1112, TFDEN-092-001, 001

NACHI

FD CONTROLLER INSTRUCTION MANUAL COMMAND REFERENCE

1st edition

- Before attempting to operate the robot, please read through this operating manual carefully, and comply with all the safety-related items and instructions in the text.
- The installation, operation and maintenance of this robot should be undertaken only by those individuals who have attended one of our robot course.
- When using this robot, observe the low related with industrial robot and with safety issues in each country.
- This operating manual must be given without fail to the individual who will be actually operating the robot.
- Please direct any queries about parts of this operating manual which may not be completely clear or any inquiries concerning the after-sale service of this robot to any of the service centers listed on the back cover.

NACHI-FUJIKOSHI CORP.

Chapter 1 Command Reference

This document is a reference manual for application commands (Functions) and move commands.

1.1 What is command?

1.1.1 Outline of the command

There are two categories for the commands in a playback program. One is motion command that can be taught using [REC] key and the other is application command that can be taught using [FN] key. However, these are simply treated as "Commands" in the robot language program.

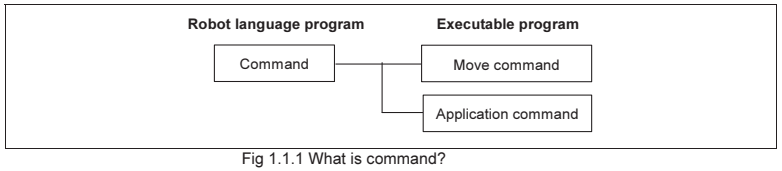


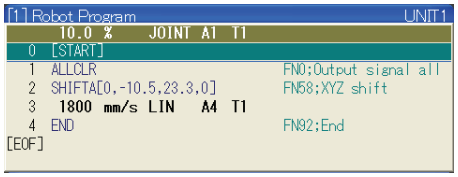
Fig 1.1.1 What is command?

Excepting three motion commands (MOVE, MOVEJ and MOVEX), the all command are called as "Application command (function)".

Application commands (functions) have code No. that starts from "FN". Motion commands do not have FN code No.

There are 100 or more various application commands for some kinds of applications or optional functions. For details. please refer to the respective option manuals.

1.1.2 Format of Move command and application command



Move command (step 3)
Application command (step 1,2 and 4)

- Parameters are surrounded by [ ] after command.
- FN code No. and title is displayed on the right side.

Fig 1.1.2 Format of move command and application command for executable program

```
ALLCLR - - - - - (1)
SHIFTA 0,-10.5,23.3,0 - - - - - (2)
MOVEX A=4,M1X,L,(*,*,*,*,*,*),S=1800,H=1,MS,CONF=0000 - - - (3)
END - - - - - (4)
```

Move command (3)
Application command (1) (2) (4)

- Parameters are displayed after command.
- Respective parameters are separated by comma.

Fig 1.1.3 Format of move command and application command for robot language program

1.2 Command list (order of SLIM command)

The command sorted in an order of SLIM command is described hereinafter.

The respective outlines are described in short. For more details, please refer to 1.3 Detail of each command (order of FN code No.).

IMPORTANT

Because this manual covers all the commands without any distinguishments like standard or option, please be sure that there are some cases where some commands are not available depending on the specification of the controller.
If the command is not displayed on the teach pendant screen, it is a command that is protected by option protect or non-supported command in an old system software version.

SLIM command	FN code	Name	Description
*	***601***	Label	Label. This is used as a label that can be referred by GOTO command etc.
ABOVE	***163***	Elbow config.(above)	The above-the-elbow (less than 180-degree angle formed by J2 axis and J3 axis) posture is forcibly selected for calculating the robot postures.
ABS	***657***	Let ABS function	Calculates the absolute value of real number.
ACOS	***649***	LET ACOS function	Get ACOS variables
ADAPTOFF	***365***	Adaptive Motion OFF	Ends Adaptive Motion. (Option)
ADAPTON	***364***	Adaptive Motion ON	Starts Adaptive Motion with the specified condition. (Option)
ADDP	***635***	Add pose variable	Adds the value of pose variable.
ADDR	***69***	Add shift value	The specified values in the specified shift register are added up. (Option)
ADDVF	***638***	Add real variable	Adds the value of real variable.
ADDVI	***637***	Add integer variable	Adds the value of integer variable.
AE	***415***	Arc end	Terminates arc welding with the specified conditions. This is available only when connected with a weld power interface.
AEM	***419***	Multi Pass Welding End	This ends the multi-pass welding under the designated conditions.
ALIGNL	***321***	Left alignment	This command controls the start and end of the left alignment function. (Glass panel handling)
ALIGNMENT	***304***	Alignment	This command controls the start and end of the alignment function. (Glass panel handling)
ALIGNR	***320***	Right alignment	This command controls the start and end of the right alignment function. (Glass panel handling)
ALIGNT	***322***	Slider alignment	This command controls the start and end of the Slider alignment function. (Glass panel handling)
ALLCLR	***0***	Output signal all reset	This command is used to set all the output signals to OFF.
ANG2ENC	***820***	Set encoder Variable (angle)	Set an angle variable(As angle) to encoder variable(As encoder)
ANG2POS	***813***	Set position Variable (angle)	Set an angle variable(As angle) to position variable(As position)
ANG2POSE	***810***	Set Pose Variable (Angle)	Set an Angle variable(As Angle)to Pose variable Pn
AOUT	***46***	Analog output	The TCP (robot tool center point) linear speed and other data are output as analog voltages. Offset can be designated using distance or time. (OPTION)
AS	***414***	Arc start	Starts arc welding with the specified conditions. This is available only when connected with a weld power interface.
ASIN	***648***	LET ASIN function	Get ASIN variables
ASM	***418***	Multi Pass Welding Start	This starts the multi-pass welding under the designated conditions.
ATN	***655***	Let ATN function	Calculates the ATN value of real number.

SLIM command	FN code	Name	Description
ATN2	***656***	Let ATN2 function	Calculates the ATN2 value of real number.
AUTOZERO	***319***	Analog input auto zero set	Auto zero the analog input signal is executed. (Option)
BARC	***613***	Draw the arc	This command is used for the user screen to draw the arc.
BELOW	***164***	Elbow config.(below)	The below-the-elbow (180-degree angle or more formed by J2 axis and J3 axis) posture is forcibly selected for calculating the robot postures.
BGCOLOR	***617***	Designate back ground color	This can designate the background color used in color graphics command.(CLS, PRINT) Total 16 colors (0 to 15) are available.
BREAK	***688***	BREAK	End the execution of an innermost instruction that encloses this with the loop or the condition structure. The control shifts to the instruction immediately after the ended instruction.
CALL	***21***	Step call	This command is used to call the step which has been specified in the same program.
CALLFAR	***454***	CallFar Program	This command is used to call the program of other unit. (Option)
CALLFARI	***455***	CallFarI Program	Using an input signal, this command is used to call the program of other unit. (Option)
CALLFARN	***456***	CallFarN Program	Using a pass count (number of passes), this command is used to call the program of other unit. (Option)
CALLI	***24***	Step call(I-condition)	Using an input signal, this command is used to call the step which has been specified in the same program.
CALLMCR	***671***	Call usertask Program	This command is used to call the specified user task program.
CALLN	***27***	Step call(freq. condition)	Using a pass count (number of passes), this command is used to call a step specified in the same program.
CALLP	***80***	Program call	This command is used to call the specified program.
CALLPBCD	***402***	Program call(external BCD prog.)	This command enables to call the program externally designated by the BCD code.
CALLPBIN	***403***	Program call(external BIN prog.)	The robot calls the program externally designated by the binary code.
CALLPI	***81***	Program call(I-condition)	Using an input signal, this command is used to call the specified program.
CALLPN	***82***	Program call(freq. condition)	Using a pass count (number of passes), this command is used to call the specified program.
CALLPR	***102***	Relative program call	This command is used to call a subprogram and makes the first step position and orientation the same as the current step in the base program and all point positions in the subprogram become relative to that step position.
CALLPRI	***103***	Conditional relative program call	Using an input signal, this command is used to call the specified program.
CALLPRN	***104***	Relative program call (freq. condition)	Using a pass count (number of passes), this command is used to call the specified program.
CallProc	***806***	Call User Procedure	Call User procedure
CASE	***687***	CASE	Two or more conditions are judged.

SLIM command	FN code	Name	Description
CHGCOORD	***113***	Change coord. No.(shift)	This makes it possible to select the number of the user coordinate system used to implement shifts based on the coordinate system.The number of the user coordinate system must be selected without fail before implementing shift-related commands based on the coordinate system.
CHGENDLESS	***373***	Change endless control	The control of the endless rotation axis to change.
CHGGUN	***95***	Mount Mechanism2	Connect or disconnect mechanism2(Option) (dedicated to mechanism 2 only)
CHGMEC	***301***	Mount Mechanism	Connect or disconnect the designated mechanism(Option)
CHGXXGUN	***238***	Change Mechanism2	This is the function command used in mechanism change without electric disconnection, or mechanism change without removing and mounting.
CHGXXMEC	***302***	Change Mechanism	As for the change mechanism, refer to the function command CHGXXGUN: Change Mechanism (FN238). CHGXXGUN (FN238) is a command exclusive for the mechanism 2, meanwhile, CHXXMEC (FN302) allows you to designate an arbitrary mechanism. Except this point, it is the same command as CHGXXGUN (FN238).
CLRREGWR	***699***	Clear register of written sts	Clear the written flag of shift register. (Option)
CLS	***609***	Clear user screen	This is to clear the user screen. (Paint screen with back ground color.)
CNVI	***550***	Conveyor interlock	Robot waits until conveyor register reaches up to the designated distance, stationarily. (Option)
CNVSYNC	***55***	Conveyor counter reset	Reset conveyor counter (Option)
CNVSYNCCHG	***274***	Conveyor synchronize select	Used to select a mechanism that is synchronized with the conveyor with the conveyor synchronization function.
CNVSYNCI	***562***	Conveyor interlock(sync.)	Robot waits until conveyor register reaches up to the designated distance, synchronizing to the conveyor. (Option)
COLOR	***616***	Designate color	This can designate the color used in color graphics command. Total 16 colors (0 to 15) are available. 0; Black 1: Gray 2: Dark Blue 3: Blue 4: Dark Green 5: Green 6: Dark Sky Blue 7: Sky Blue 8: Dark Red 9: Red 10:Purple 11:Pink 12:Dark Yellow 13:Yellow 14:Light Gray 15:White
COLSEL	***230***	Set interference detection level	The threshold value to be considered as interference can be switched during playback.
COMPOFF	***207***	Compliance OFF	When this function command is executed, the soft compliance control function becomes disabled. (Option)
COMPON	***206***	Compliance ON	When this function command is executed, the soft compliance control function becomes enabled, and the robot can be moved according to external force. (Option)
COS	***653***	Let COS function	Calculates the COS value of real number.
CVTCOORDPOS	***821***	Coord. Trans (position)	Translate a position variable(As Position) to a designated coordinate.
DELAY	***50***	Timer delay	This command is used to place the robot in the standby status.
DIM	***801***	Any variable	You can define some variables as Integer, real and Array, as you like.

SLIM command	FN code	Name	Description
DIVVF	***644***	Divide real variable	Divides the value of real variable.
DIVVI	***643***	Divide integer variable	Divides the value of integer variable.
DOUT	***278***	Digital output	The TCP (robot tool center point) linear speed and other data are output using general-purpose output signals.
DPRESETM	***280***	Output signal(Distance)	This command is used to set one of the general-purpose output signals with advancing distance.
DSPALLET	***65***	Direction select palletize	This limits the shift direction, and it is started by the palletizing work. (Option)
DYNCALIBROB	***295***	Robot calibration	Start/End calibration cycle of Dynalog PC (Option)
DYNMESPOS	***296***	Measurement point	The Dynalog PC to take a measurement of the current position of the robot's TCP. (Option)
ELSE	***678***	Condition	Move the control to the following instruction.
ELSEIF	***677***	Condition	Move the control to the following instruction when the condition consists. Move the control to ELSE and ENDIF for the failure.
ENC2ANG	***817***	Set angle Variable (encoder)	Set an encoder variable(As encoder) to Angle variable(As Angle)
ENC2POS	***814***	Set position Variable (encoder)	Set an encoder variable(As encoder) to position variable(As position)
ENCS2POSE	***811***	Set Pose Variable (encoder)	Set an encoder variable(As Encoder)to Pose variable Pn
END	***92***	End	This command is used to end program playback. If the program is a called program, return to the original program.
ENDIF	***679***	Condition end	End IF-ENDIF.
EndProc	***804***	End Procedure	Finish and exit Procedure, and back to source procedure
ENDS	***689***	SWITCH end	It is a terminator of the SWITCH-ENDS structure.
ENDW	***664***	WHILE end	It is terminator of the WHILE-ENDW structure.
EP	***498***	Execution Pass Specification	This designates per pass whether the function commands are to be executed or not in the multi-pass section.
EQUALIZE	***287***	Equalize value	This command is used to the equalizing motion as defined by the servo gun.
EQUALIZECLR	***248***	Equalize clear	The equalize setting clear.
ET	***486***	End tracking	This ends the seam tracking. This is used when the arc sensor (AX-AR) is connected.
EXIT	***619***	User task end	This can terminate the user task.
ExitProc	***803***	Exit User Procedure	Stop procedure routine and back to source procedure
FBUSREL	***312***	Field bus release	In the field bus master, error detection Enabled/Disabled of the specified node is switched.
FCASEEND	***88***	Case jump end	This command is used to end the case jump(FCASEI, FCASEN).
FCASEI	***87***	Case jump(I-condition)	Using an input signal, this command is used to select one of a multiple number of steps and executes it.

SLIM command	FN code	Name	Description
FCASEN	***86***	Case jump(freq. condition)	Using a pass count (number of passes), this command is used to select one of a multiple number of steps and execute it.
FETCH	***528***	Fetch Input cond.	Determine judgment the input condition of a following function.
FHCLAMP	***362***	FLEXhand Clamp	Execute clamp moiton by FLEXhand. (Option)
FHCLAMP2	***366***	New FH Clamp	Execute clamp moiton by FLEXhand
FHCLAMPDCT	***368***	FH Clamping Detection	Detect clamp status by FLEXhand
FHUNCLAMP	***363***	FLEXhand Unclamp	Execute unclamp moiton by FLEXhand (Option)
FHUNCLAMP2	***367***	New FH Unclamp	Execute unclamp moiton by FLEXhand
FLIP	***165***	Wrist config.(flip)	The wrist-flip posture is forcibly selected for calculating the robot postures.
FOR	***604***	Loop Start	This is loop command. Loop starts here. See also; NEXT(FN605)
FORK	***450***	Fork Program	This command is used to start the program of other unit. (Option)
FORKI	***451***	ForkI Program	Using an input signal, this command is used to start the program of other unit. (Option)
FORKMCR	***670***	Fork Usertask Program	This command is used to start the specified user task program.
FORKMCRDST	***673***	Fork User Task Program (distance)	This command is used to start the specified user task program with advancing distance.
FORKMCRTM	***672***	Fork User Task Program (time)	This command is used to start the specified user task program. Furthermore, the command enables advance execution to be specified.
FORKN	***452***	ForkN Program	Using a pass count (number of the passes), this command is used to start the program of other unit.(Option)
FORKWAIT	***453***	Wait Fork-Program	This command leads the robot to await the completion of the task program of the other unit which was started up by the FORK, FORKI or FORKN command. (Option)
FRANGE	***202***	Flange axis rot. config.	The rotational direction of the J6 axis is specified for calculating the robot postures
GARC	***623***	Display ellipse	This command is used for the user screen to draw the ellipse.
GBOX	***612***	Draw the box	This command is used for the user screen to draw the box.
GE	***413***	Gas OFF	Stops the shield gas.
GETANG	***823***	Set angle variable (pos.data)	Substitute robot position to angle variables
GETANGLE	***157***	Set real variable (angle)	This command is used to store the current angle value of each axis in a real number variable.
GETBYTE	***587***	Get buffer (byte)	This command is used to read one byte data from the buffer, and stored the integer variable.
GETENC	***824***	Set encoder variable (pos.data)	Substitute robot position to encoder variables
GETFIGURE	***158***	Set real variable (figure)	The robot figure is used to store in real number variables.

SLIM command	FN code	Name	Description
GETFORCE	***360***	Get force/torque	The force/torque data are acquirable to a real variable. (Option)
GETINT	***585***	Get buffer (integer)	This command is used to read data from the buffer, and stored the integer variable.
GETP	***142***	Set real variable (coordinate)	This command is used to store the current coordinate values (RPY angle expression) in real number variables.
GETPELR	***94***	Set real variable(Euler pos)	This command is used to store the current coordinate values (Eulerian angle expressions) in the real number variables
GETPOS	***822***	Set position variable (pos.data)	Substitute robot position to positional variables
GETPOSE	***143***	Set real variable (pose)	This stores the pose variable Pn into the real variable V!.
GETREAL	***586***	Get buffer (real)	This command is used to read data from the buffer, and stored the real variable.
GETSFT	***145***	Set real variable (shift)	This command replaces the values of the specified shift register with the specified real number variables (7 consecutive variables are used).
GETSTR	***584***	Get buffer (string)	This command is used to read data from the buffer, and to store data in the string variable.
GETTIPCON	***306***	Get tip consumption	Used to get tip consumption amount.
GFONT	***683***	Set the font	The font of the user screen is set.
GLINE	***611***	Display position specification	This command is used for the user screen to draw the straight line.
GMSGBOX	***685***	Create message box	Create a message box on the user screen.
GOSUB	***91***	Line call	Execute a sub-routine call by a designated line No. or label.
GOTO	***90***	Line jump	This is used to jump to a designated line or label.
GPAINT	***614***	Paint	This command is used to paint out the enclosed area on the user screen.
GS	***412***	Gas ON	Starts to output the shield gas.
GSEA	***167***	Servo gun search	This command is used to detect the electrode tip consumption of the servo gun.
GSEA_ORDER	***229***	Servo gun search order	Servo gun search2 is execute before servo gun search1
GSETP	***615***	Draw the pixel	This command is used for the user screen to draw a pixel.
GSOFTKEY	***684***	Create soft key	Create a soft key on the user screen.
GUNOPEN	***218***	Gun Open	This command is used to change stroke of air gun.
ICH	***410***	Inching	Performs inch the wire with specified time and wire speed
IF	***602***	Condition	If condition is satisfied then command(jump/call) after “THEN” is executed, else command(jump/call) after “ELSE” is executed.
IF	***676***	Condition	Move the control to the following instruction when the condition consists. Move the control to ELSEIF, ELSE, and ENDIF for the failure.
INCLUDE	***697***	Translate table included (file)	The conversion rule is read from “inc file”

SLIM command	FN code	Name	Description
INCLUDEIO	***698***	Translate table included (I/O)	The conversion rule is read from “I/O NAME”.
INH	***310***	Inhibit	This determines to inhibit the fetch control.
INPUT	***271***	Strings input	This receives the character string data from the specified communication (serial) port, and holds it in the specified character string variable. (Option)
JMP	***20***	Step jump	The robot jumps to the step specified in the same program.
JMPI	***23***	Step jump(I-condition)	Using an input signal, this command causes the robot to jump to the step specified in the same program.
JMPN	***26***	Step jump(freq. condition)	Using a pass count (number of passes), the robot jumps to the step specified in the same program.
JMPP	***83***	Program jump	This command is used to jump to the start of the specified program.
JMPPBCD	***400***	Program jump(to ext. BCD prog.)	This command enables to externally jump to the program designated by the BCD code.
JMPPBIN	***401***	Program jump(to ext. BIN prog.)	The robot jumps to the program externally designated by the binary code.
JMPPI	***84***	Program jump(I-condition)	Using an input signal, this is used to jump to the start of the specified program.
JMPPN	***85***	Program jump(freq. condition)	Using a pass count (number of passes), this command is used to jump to the start of the specified program.
LCALLMCR	***593***	Call User Task Program with Arguments	This command is used to call the specified user task program. At this time, ten real numbers can pass the arguments to the program.
LCALLP	***590***	Program call with Arguments	This command is used to call the specified program. At this time, ten real numbers can pass the arguments to the program.
LCALLPI	***591***	Conditional program call with Arguments	Using an input signal, this command is used to call the specified program. At this time, ten real numbers can pass the arguments to the program.
LCALLPN	***592***	Conditional program call after specified number of passes with Arguments.	Using a pass count (number of passes), this command is used to call the specified program. At this time, ten real numbers can pass the arguments to the program.
LEFTY	***161***	Arm config.(left/front)	The left-arm system posture is forcibly selected for calculating the robot postures
LET	***634***	Let variable	Sets the variable of the same type.
LETC	***647***	Integer variable setting	Used to make setting of values to integer variable registers specified.
LETCOORDP	***630***	Let pose variable	Stores the pose data recorded by the specified rectangular coordinates value in the pose variables.
LETLF	***629***	Set local real variable	Sets the value into the specified local real variable register.
LETLI	***628***	Set local integer variable	Sets the value into the specified local integer variable register.
LETPE	***632***	Let pose element	Stores the pose element recorded by the specified rectangular coordinates value in the pose variables.

SLIM command	FN code	Name	Description
LETPOSE	***144***	Set pose variable	This stores the real variable V!. into the pose variable.
LETR	***68***	Set shift value	The shift amount data is set in the specified shift register. (Option)
LETRE	***633***	Let shift element	Sets the shift element in the specified shift register.
LETVF	***76***	Set real variable	Substitute a value for a global float variable. Can not substitute for a local variable.
LETVI	***75***	Set integer variable	Substitute a value for a global integer variable. Can not substitute for a local variable.
LETVS	***77***	Set strings variable	Substitute a value(string) for a global string variable. Can not substitute for a local variable.
LETX	***71***	Pose X	Substitute a value for the X component of a pose. LETX, LETY, and LETZ are available only for an already recorded pose. These functions are used in a case where only 1 pose is made and parallel shift is applied for the pose.
LETY	***72***	Pose Y	Substitute a value for the Y component of a pose.
LETZ	***73***	Pose Z	Substitute a value for the Z component of a pose.
LOCATE	***610***	Locate the display pos	This command is used to specify the position of the character displayed on the user screen.
LOCCVT	***53***	Coord. trans(shift value)	It is possible to proceed with playback while offsetting each recorded point based on the difference (skew amount) measured beforehand between the recorded position of the three points serving as the reference and the actual position obtained from the visual device, etc. (OPTION)
LOCCVT1	***54***	Coord. trans(posi. value)	It is possible to proceed with playback while offsetting each recorded point based on the difference (skew amount) measured beforehand between the recorded position of the three points serving as the reference and the actual position obtained from the visual device, etc.
LOCCVT3	***275***	Base angle shift	The start or end of the shift operation is specified. When shift operation start has been specified, the shift operation is performed on the basis of the shift amount stored in the specified shift register.
MAPPING	***305***	Mapping	This command controls the start and end of the mapping function. (Glass panel handling)
MAX	***659***	Let MAX function	Calculates a larger real number out of two.
MIN	***658***	Let MIN function	Calculates a smaller real number out of two.
MODUSRCOORD	***626***	Modify User coordinate	Modifies the existent user coordinates using pose variables
MOVE	**-**	Movement (Cartesian coordinates or pose designation)	Move the robot based on a traditional Cartesian coordinate system of NACHI AW controller.(XYZRPY)
MOVEJ	**-**	Movement (Axis values)	Move the robot using joint values(angles) in NACHI AW controller format.
MOVEX	**-**	Movement (Unified format)	Move the robot based on this controller original format. Position data can be given in any of Cartesian coordinates, joint values and encoder data.

SLIM command	FN code	Name	Description
MPE	***497***	Multi Pass Section End	This represents the end position of the section where a series of movements for multi-pass welding is repeated.
MPS	***496***	Multi Pass Section Start	This represents the start position of the section where a series of movements for multi-pass welding is repeated.
MULTIM	***264***	Multi output signal	This command is used to set the pre-defined multiple output signals to ON or OFF using the binary format.
MULVF	***642***	Multiply real variable	Multiplies the real variables.
MULVI	***641***	Multiply integer variable	Multiplies the integer variables.
NEXT	***605***	Loop End	Please refer to "FOR"(FN604).
NONFLIP	***166***	Wrist config.(non-flip)	The wrist-non-flip posture is forcibly selected for calculating the robot postures.
NOP	***600***	NOP	No operation
NRLCRD	***171***	Select robot language coordinate system	Used to switch functions to a specified user coordinate system.
OFFSET	***499***	Multi Offset Specification	This sets offset in the movement steps in the multi-pass welding section.
ONGOTO	***603***	ON GOTO Jump	Next command(jump/call) is determined by the value of condition. It's order is 1,2,3… from left.
OPEANG	***827***	Extraction angle Variable	Substitute or Extract an angle variable to a global real variable (V!) or local real variable (L!).
OPEENC	***828***	Extraction encoder Variable	Substitute or Extract an encoder variable to a global integer variable (V%) or local integer variable (L%), any integer variable.
OPEPOS	***826***	Extraction position Variable	Substitute or Extract a position variable to a global real variable (V!) or local real variable (L!).
OPEPOSE	***825***	Extraction pose Variable	Substitute or Extract a pose variable to a global real variable (V!) or local real variable (L!).
OUT	***44***	Binary output signal	This command is used to set the general-purpose output signals in any group to ON or OFF using the binary format.
OUTDIS	***43***	Discrete output signal	This command is used to set the general-purpose output signals in any group to ON or OFF using the discrete format.
PALLET2	***47***	Palletize start	Start palletizing based on the pre-designed palletizing pattern. (OPTION)
PALLET2_END	***48***	Palletize end	Finish palletizing based on the pre-designed palletizing pattern. Confirmation signal can be output. (OPTION)
PALLET2_RESET	***49***	Palletize reset	When a condition signal has been input, the palletize counter is forcibly reset. (palletizing operation is forcibly terminated) (OPTION)
PALLET3	***249***	Palletize start	Start palletizing based on the pre-designed palletizing pattern.
PALLET3_APR	***374***	Palletize approach selection	Starts approaching motion.
PALLET3_END	***250***	Palletize end	Finish palletizing based on the pre-designed palletizing pattern.
PALLET3_GETREG	***377***	Get palletize register	Store some palletize register to some variables.
PALLET3_OPT	***375***	Palletize optimize path	Optimize Step position based on the locus of previous step and following step
PALLET3_RESET	***251***	Palletize reset	The palletize counter is forcibly reset. (palletizing operation is forcibly terminated)

SLIM command	FN code	Name	Description
PALLET3_SELGR	***376***	Palletize select grasp position	Select Work grasp position from registered by Palletize pattern.
PALLET3_SELZ	***388***	Palletize select height(Z)	The function compares the Z-axis value of target step and the Z-axis value of reference step after palletizing shift by using target palletizing number set by the first parameter of the function, and then Z-axis value of target step will be adjust higher value of both.
PALLET3_SETREG	***378***	Set palletize register	Set some variables to some palletize registers.
PAUSE	***620***	Pause user task	This can make a brief stop of user task.
PAUSEINPUT	***252***	Pause Input	The robot is pause when the designated [Pause input] signal is turned off.
POS2ANG	***816***	Set angle Variable (position)	Set a position variable(As position) to Angle variable(As Angle)
POS2ENC	***819***	Set encoder Variable (position)	Set a position variable(As position) to encoder variable(As encoder)
POS2POSE	***809***	Set Pose Variable (position)	Set a position variable(As Position)to Pose variable Pn
POSAUTO	***160***	Disable posture control	Used to disable the posture control for robot posture calculation.
POSE2ANG	***815***	Set angle Variable (pose)	Set Pose variable Pn to Angle variable(As Angle)
POSE2ENC	***818***	Set encoder Variable (pose)	Set Pose variable Pn to encoder variable(As encoder)
POSE2POS	***812***	Set position Variable (pose)	Set a Pose variable Pn to position variable(As position)
POSESAVE	***74***	Pose file save	Pose variables are stored to the pose file.
PRINT	***101***	Output strings	The character string data is output to the screen or specified RS232C serial port.
PRINT	***606***	Print String	Please refer to "PRINT"(FN101).
PRINTF	***669***	Print string with format	Draw the string data on the screen with form. Or Output string data with form via RS232C.
PRSD	***308***	Read press data	This reads a press brake synchronization setting file, in the press brake synchronization function.
PRSI	***564***	Press interlock	This sets interlock, in the press brake synchronization function.
PRSS	***307***	Press brake shelter	This executes retreat actions after work process, in the press brake synchronization function.
REGC	***224***	Shift register copy	Data is copied between shift registers. (Option)
RELMOV	***407***	Move of External axis	The designated external axis moves the specified distance from the current position.
REM	***99***	Comment	This command is used to provide comments inside programs.
RESET	***34***	Output signal reset	This command is used to set one of the general-purpose output signals to OFF.
RETI	***25***	Step return(I-condition)	Using an input signal, this command is used to return the robot to the step following the one which executed the step call command in the same program.
RETN	***28***	Step return(freq. condition)	Using a pass count (number of passes), this command is used to return the robot to the step following the one which executed the step call command in the same program.

SLIM command	FN code	Name	Description
RetProc	***805***	Return User Procedure	Set a return value of user procedure
RETURN	***22***	Step return	This command is used to return the robot to the step following the one which executed the step call command in the same program. The commands that call a sub-routine are CALL, CALLI, CALLN, and GOSUB. Normally, only GOSUB is used in robot language.
RIGHTY	***162***	Arm config.(right/back)	The right-arm system posture is forcibly selected for calculating the robot postures
RINT	***29***	Robot interrupt(I-condition)	Executing step is interrupted by input signal
RSCLR	***111***	RS232C buffer clear	The send/receive buffer inside the specified RS232C port is cleared. (Option)
RTC	***411***	Retract	Performs to retract the wire with specified time and wire speed.
SEA	***59***	Search	Detect the work position shift amount, and store those data to the shift register (Option)
SEAMEND	***246***	Seam weld end	Stop the seam welding.
SEAMOV	***313***	Seam override	Set the electrode rotation speed override.
SEAMSPD	***247***	Seam electrode speed	Set the electrode rotation speed while welding.
SEAMST	***245***	Seam weld start	Start the seam welding.
SEAMTHICK	***311***	Seam panel thick	Set the thickness of the work.
SET	***32***	Output signal set	This command is used to set one of the general-purpose output signals to ON.
SETBYTE	***583***	Set buffer (byte)	This command is used to stored byte data at an arbitrary position in the buffer.
SETC	***646***	Set output signal	Consecutive output signal is output.
SETINT	***581***	Set buffer (integer)	This command is used to stored integer value at an arbitrary position in the buffer.
SETM	***105***	Output signal	This command is used to set any general-purpose output signal to ON or OFF
SETMD	***35***	Output sig(ON/OFF/delay/ pulse)	This command is used to set one of the general-purpose output signals to come with a pulse or delay and to ON or OFF.
SETO	***100***	Consecutive output signal ON/OFF	This command is used to set any number of consecutive general-purpose output signals to ON or OFF altogether.
SETREAL	***582***	Set buffer (real)	This command is used to stored real value at an arbitrary position in the buffer.
SETSTR	***580***	Set buffer (string)	This command is used to stored string at an arbitrary position in the buffer.
SF0	***470***	Wire Extension	This detects and corrects the wire extension. This is used when the touch sensor (AX-WD) is connected.(OPTION)

SLIM command	FN code	Name	Description
SF1	***471***	One Direction Search(Touch)	This detects the deviation of a workpiece by a touch sensor. *This is used when the touch sensor (AX-WD) is connected. (Option)
SF3	***473***	Deviation call	This receives the stored deviation and execute a compensation. (Option) *This is used when the touch sensor (AX-WD) and the laser search (AX-RD) are connected.
SF4	***474***	Dev. vector composition	This calculates a new deviation on the basis of stored deviation. This is used when the touch sensor (AX-WD) and the laser search (AX-RD) are connected.
SF9	***479***	Generation of a GAP.file	This stores variable values to a gap file. This is used when the touch sensor (AX-WD) and the laser search (AX-RD) are connected.
SGSPRT	***279***	Servo gun separation	Servo gun separated status is changed.
SGTIPRST	***270***	Reset consumption	Reset the tip consumption of designated servo gun
SHIFTA	***58***	XYZ shift	The playback position is shifted in parallel (Option)
SHIFTR	***52***	Shift	The start or end of the shift operation is specified. When shift operation start has been specified, the shift operation is performed on the basis of the shift amount stored in the specified shift register.
SIN	***652***	Let SIN function	Calculates the SIN value of real number.
SLEND	***351***	Seal end	Stop the dispensing process
SLPRS	***355***	Seal press ctrl	Start/Stop pressure control in the dispensing process.
SLPRSG	***356***	Seal press ctrl 2	Start/Stop pressure control in the dispensing process. Control level can be designated.
SLREADY	***353***	Flow ready	Pressure in the pump is controlled to a specified value.
SLRELOAD	***352***	Reload	Refill the booster pump.
SLSTART	***350***	Seal start	Start the dispensing.
SOCKBIND	***572***	Bind the socket	This command is used to assign a socket an port No.
SOCKCLOSE	***571***	Close the socket	This command is used to close the socket.
SOCKCONNECT	***574***	Connect to server	This command is used to connect to server.
SOCKCREATE	***570***	Create Socket	This command is used to create the socket.
SOCKRECV	***577***	Receive data	This command is used to receive the data.
SOCKSEND	***575***	Send data	This command is used to transmit the data stored in the specified buffer.
SOCKSENDSTR	***576***	Send string	This command is used to transmit the specified string.
SOCKWAIT	***573***	Wait for connect	This command is waited for until the connection from the client is done to the allocated port.
SPDDOWNA	***169***	Analog input speed override	The playback speed of the robot is changed in accordance with the input voltage. (Option)

SLIM command	FN code	Name	Description
SPDDOWND	***277***	Digital input speed override	The playback speed of the robot is changed in accordance with digital input signals
SPF	***439***	Servo OFF	This turns OFF the servo power source in unit of mechanism.
SPN	***438***	Servo ON	This turns ON the servo power source in unit of mechanism.
SPOT	***119***	Spot welding	Execute spot welding in accordance with pre-defined sequence.
SPOT2	***268***	Spot welding	By recording the spot welding function to the welding step, spot welding can be carried out by a designated sequence
SPOTIWB1	***199***	Spot welding	Execute spot welding in accordance with pre-defined sequence.（dedicated to welding I/F＝MEDbus only）(Option)
SQR	***651***	Let SQR function	Calculates the square root of the real number.
SREQ	***51***	Shift data request	The command requests the shift amount data from the external device using the serial port. Once it has been input from the external device, the shift amount data is stored in the specified shift register.
SREQ2	***315***	Shift amount request (binary)	The command requests the shift amount data (binary data) from the external device using the serial port. Once it has been input from the external device, the shift amount data is stored in the specified shift register.
ST	***485***	Start tracking	This starts seam tracking. This is used when the arc sensor (AX-AR) is connected.
STOOL	***67***	Select the stated tool No.	This command is used to select the coordinate system of the stationary tool number from among the user coordinate systems
STOP	***41***	Robot stop	This command is used to stop the robot.
STOPI	***42***	Robot stop(I-condition)	Using an input signal, this command is used to stop the robot.
SUBVF	***640***	Subtract real variable	Subtracts the real variable.
SUBVI	***639***	Subtract integer variable	Subtracts the value of integer variable.
SWITCH	***686***	SWITCH	Two or more conditions are judged.
SYNCSPOT	***303***	Sync spot welding	Enabling to perform synchronous welding with two servo guns.
SYNCSPOTIWB	***316***	Sync spot welding	Enabling to perform synchronous welding with two servo guns.（dedicated to welding I/F＝MEDbus only）(option)
TAN	***654***	Let TAN function	Calculates the TAN value of real number.
TIMER	***650***	Let TIMER function	Sets the time value passed since the power-on into the specified real variable register.
TIPDRESS	***265***	Tip dress	Execute the tip dress of spot welding gun.
TITLE	***608***	User screen title	This is to draw the title of user screen.

SLIM command	FN code	Name	Description
USE	***98***	Select pose file	This function is used to select a pose file. Pose data is controlled as a file, and poses P1 - P9999 can be recorded into a file. For example, it is useful when to change only the position data of the robot according to the type of workpiece, and when to play back a same program.
USRERR	***467***	User Error Output	Output the user customized error, alarm and information.
UsrProc	***802***	User Procedure	Define User Procedure
VCHKGRP	***336***	Vision group check	Check the measurement group of the vision sensor.(Option)
VDATA	***334***	Vision data	Get the data from the vision sensor.(Option)
VGROUP	***335***	Vision group change	Change the measurement group of the vision sensor.(Option)
VLOCCVT	***342***	Vision location convert	The start or end of the shift operation with the vision sensor are specified. When shift operation start has been specified, the shift operation is performed on the basis of the shift amount stored in the specified shift register. (Option)
VRESET	***330***	Vision reset	Data of the vision sensor is cleared. (Option)
VSHIFT	***333***	Vision shift	Get the shift value from the vision sensor. (Option)
VSTART	***331***	Vision start	Start the measurement of vision sensor. (Option)
VWAIT	***343***	Vision measure wait	The measurement completion of the vision sensor is waited for. (Option)
VWORK	***332***	Vision work	Distinguish work with the vision sensor. (Option)
WAIT	***552***	Wait I-cond with timer	This command is used to wait for any one general-purpose input signal for up to the specified time.
WAITA	***553***	Wait I-group(AND) with timer	This command is used to wait for any of group general-purpose input signal (AND logic) with designated time.
WAITAD	***558***	Wait I-group BCD(AND) with timer	This command is used to wait for any of group general-purpose input signal (AND logic) with designated time. The condition is written in BCD format.
WAITE	***555***	Wait I-group with timer	This command is used to wait for any of group general-purpose input signal with designated time.
WAITED	***560***	Wait I-group BCD with timer	This command is used to wait for any of group general-purpose input signal with designated time. The condition is written in BCD format.
WAITI	***525***	Wait Input cond	This command is used to wait for any one general-purpose input signal.
WAITJ	***526***	Wait not Input cond	This command is used to wait for any one general-purpose input signal using negative logic.
WAITO	***554***	Wait I-group(OR) with timer	This command is used to wait for any of group general-purpose input signal (OR logic) with designated time.
WAITOD	***559***	Wait I-group BCD(OR) with timer	This command is used to wait for any of group general-purpose input signal (OR logic) with designated time. The condition is written in BCD format.

SLIM command	FN code	Name	Description
WAITR	***127***	Wait shift value receive	This initiates a jump to the shelter step when the robot has been waiting for the shift amount data to be input from the external source into the specified shift register and the data has not been input within the specified time. (Option)
WAX	***441***	Axis Weaving	Starts weaving with the simple harmonic motion of the axes.
WE	***443***	Weaving End	Terminates weaving
WELDCND	***33***	Spot condition output	This function outputs signals assigned to the "weld condition output."
WELDGRP	***282***	Weld condition with group	When this function command is executed, the welding machine number to be used by the welding function (FN119) and the welding condition group number are designated.
WFP	***440***	Fix Pattern Weaving	Starts weaving with the specified waveform, attitude, and frequency.
WHILE	***663***	WHILE loop	Execute the instruction in WHILE-ENDW repeatedly until the condition doesn't consist.
WINDOW	***607***	User screen open/close	Open user screen, or close user screen.
WSF	***442***	Taught Weaving	This carries out weaving in the taught pattern.
ZF1	***480***	One Direction Search(Laser)	This detects the setting deviation of a workpiece. This is used when the laser search (AX-RD) is connected.
ZG1	***483***	High-speed groove search	This searches the groove information at high speed. This s used when the laser search (AX-RD) is connected.

1.3 Detail of each command (order of FN code No.)

Detail of each command is described hereinafter, sorted in an order of FN code number.

IMPORTANT

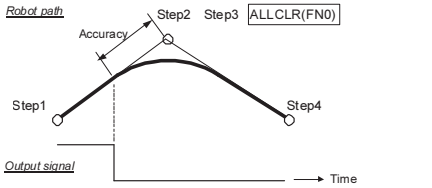
Because this manual covers all the commands without any distinguishments like standard or option, please be sure that there are some cases where some commands are not available depending on the specification of the controller.
If the command is not displayed on the teach pendant screen, it is a command that is protected by option protect or non-supported command in an old system software version.

Function commands (FN codes)

Command name	ALLCLR
FN code	0
Title name	All output signals clear
General description	This command is used to set all the general-purpose output signals to OFF.

General description
When this function command is executed, it is possible to set all the general-purpose output signals (O1 to O2048) to OFF. However, the command cannot set any of the status signals (signals with pre-assigned applications such as the gun signals and starting signal) to OFF.
Which signals have been assigned as status signals can be identified on the monitor screen. Signals whose numbers are displayed in the bold italics are status signals so any of the other signals can be set to OFF altogether.

■ **Example of operation**



The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

■ **Parameter**
None

■ **Example of screen display**

ALLCLR	FN0: All output signals clear

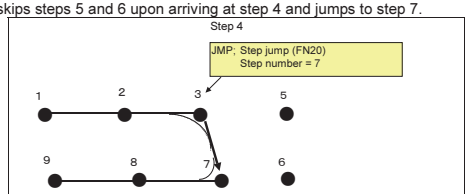
See
SET; Output signal ON (FN32)
RESET; Output signal OFF (FN34)
SETO; Consecutive output signal ON/OFF (FN100)
SETM ; Output signal ON/OFF (FN105)

Function commands (FN codes)

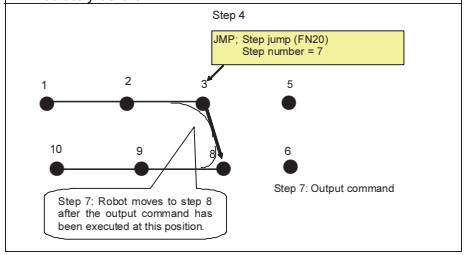
Command name	JMP
FN code	20
Title name	Step jump
General description	The robot jumps to the step specified in the same program.

■ **General description**
When this function command is executed, the robot is able to jump to the step specified in the same program. It makes no difference whether the jump destination step is a move command or function command.
Bear in mind that if the jump destination step is a function command, the function command at the jump destination will be executed as soon as the jump command has been executed.

■ **Example of operation**
In step 4, record JMP: step jump (FN20), and 7 as the number of the jump destination step. When this is played back, the robot skips steps 5 and 6 upon arriving at step 4 and jumps to step 7.



The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.



■ **Parameter**

Parameter No. 1	Step No.	This specifies the number of the step serving as the robot's jump destination. (1-9999)

■ **Example of screen display**

JMP[35]	FN20; Step jump

See
JMPI: Conditional step jump (FN23)
JMPN; Conditional step jump after specified number of passes (FN26)

Function commands (FN codes)

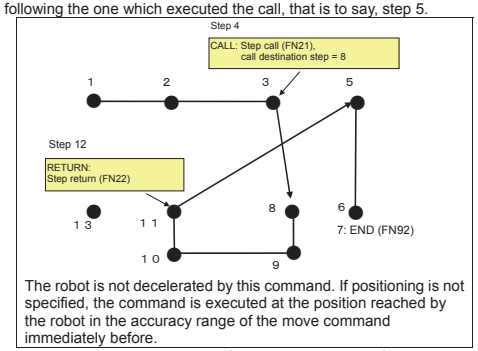
Command name	CALL
FN code	21
Title name	Step Call
General description	This command is used to call the step which has been specified in the same program.

■ General description

When this function command is executed, a step specified in the same program is called. It makes no difference whether the call destination step is a move command or function command.
Bear in mind that if the call destination step is a function command, the function command at the call destination will be executed as soon as the call command has been executed.
When the step return command is subsequently executed, operation returns to the step following the one which executed the call.

■ Example of operation

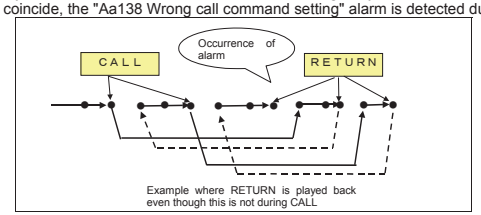
The step call and step return commands are used in tandem.
In step 4, record CALL: step call (FN21), 8 as the number of the call destination step, and in step 12, record RETURN: step return (FN22). When this is played back, the robot skips steps 5 to 7 upon arriving at step 4 and jumps to step 8. Then it advances to steps 9 through 12, and after the RETURN: step return (FN22) command is executed, it returns to the step following the one which executed the call, that is to say, step 5.



The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

The robot is returned to the step following the one with CALL; step call (FN21) command by the RETURN: step return (FN22) command, the RETI: conditional step return (FN25) command or by the RETN: conditional step return after specified number of passes (FN28) command.

The step call command can be executed again at the call destination (between steps 8 and 11 in the above figure). Up to 8 layers of calls can be executed. If calls exceeding 8 layers are executed or the number of step calls and step returns does not coincide, the "Aa138 Wrong call command setting" alarm is detected during playback, and the robot stops.



■ Parameter

Parameter No. 1	Step No.	This specifies the number of the call destination step. (1-9999)

■ Example of screen display

CALL[12]	FN21: Step call

See
CALLI: Conditional step call (FN24) command
CALLN: Conditional step call after specified number of passes (FN27) command
RETURN: Step return (FN22)
RETI: Conditional step return (FN25) command
RETN: Conditional step return after specified number of passes (FN28) command

Function commands (FN codes)

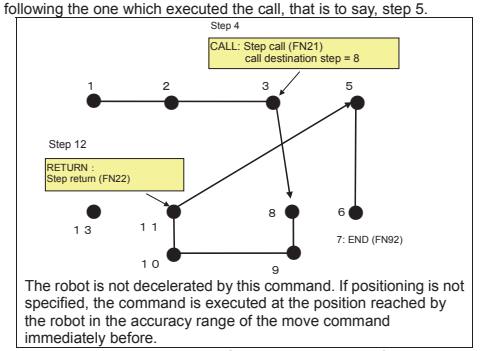
Command name	RETURN
FN code	22
Title name	Step return
General description	This command is used to return the robot to the step following the one which executed the step call command in the same program.

■ General description

When this function command is executed, the robot is returned to the step following the one which executed the step call command in the same program. It makes no difference whether the return destination step is a move command or function command.
Bear in mind that if the return destination step is a function command, the function command at the return destination will be executed as soon as the return command has been executed.

■ Example of operation

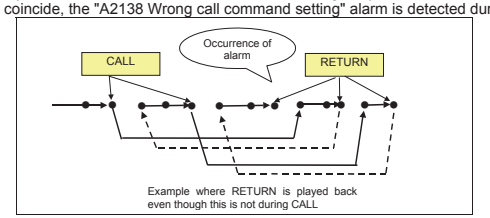
The step call and step return commands are used in tandem.
In step 4, record CALL: step call (FN21), 8 as the number of the call destination step, and in step 12, record RETURN: step return (FN22). When this is played back, the robot skips steps 5 to 7 upon arriving at step 4 and jumps to step 8. Then it advances to steps 9 through 12, and after the RETURN: step return (FN22) command is executed, it returns to the step following the one which executed the call, that is to say, step 5.



The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

The robot is returned to the step following the one with CALL; step call (FN21) command by the RETURN: step return (FN22) command, the RETI: conditional step return (FN25) command or by the RETN: conditional step return after specified number of passes (FN28) command.

The step call command can be executed again at the call destination (between steps 8 and 11 in the above figure). Up to 8 layers of calls can be executed. If calls exceeding 8 layers are executed or the number of step calls and step returns does not coincide, the "A2138 Wrong call command setting" alarm is detected during playback, and the robot stops.



■ Parameter

None

■ Example of screen display

RETURN	FN22; Step return

See
CALL; Step call (FN21)
CALLI: Conditional step call (FN24) command
CALLN: Conditional step call after specified number of passes (FN27) command
RETI: Conditional step return (FN25) command
RETN: Conditional step return after specified number of passes (FN28) command

Function commands (FN codes)

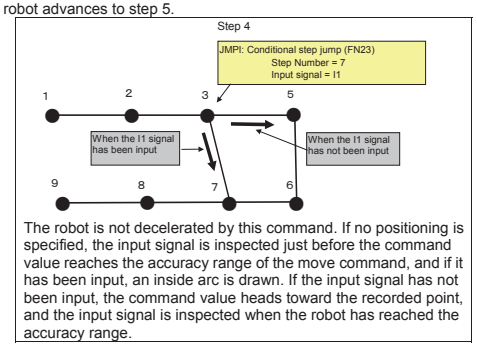
Command name	JMPI
FN code	23
Title name	Conditional step jump
General description	Using an input signal, this command causes the robot to jump to the step specified in the same program.

■ General description

When this function command is executed, the robot is able to jump to the step specified in the same program. It makes no difference whether the jump destination step is a move command or function command.
When the specified input signal has been input, the robot jumps; when it has not been input, it does not jump and the command is passed by.
Bear in mind that if the jump destination step is a function command, the function command at the jump destination will be executed as soon as the jump command has been executed.

■ Example of operation

In step 4, record JMPI: step jump (FN23), 7 as the number of the jump destination step and I1 as the input signal.
If, when this is played back, input signal I1 has been input, the robot jumps to step 7, but if the signal has not been input, the robot advances to step 5.



The robot is not decelerated by this command. If no positioning is specified, the input signal is inspected just before the command value reaches the accuracy range of the move command, and if it has been input, an inside arc is drawn. If the input signal has not been input, the command value heads toward the recorded point, and the input signal is inspected when the robot has reached the accuracy range.

■ Parameter

Parameter No. 1	Step No.	This specifies the number of the step serving as the robot's jump destination. （1-9999）
Parameter No. 2	Input signal	This records the number of the input signal which is to serve as the condition for executing the jump. When number 5101 or above is specified, multiple input signals can be specified. （1-2048、5101～5196）

■ Example of screen display

JMPI [7,I1]　　FN23: Conditional step jump

See
JMP: Step jump (FN20)
JMPN: Conditional step jump after specified number of passes (FN26)

Function commands (FN codes)

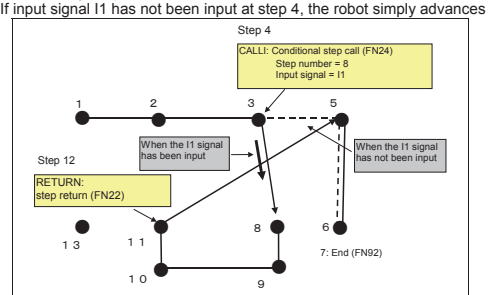
Command name	CALLI
FN code	24
Title name	Conditional step call
General description	Using an input signal, this command is used to call the step which has been specified in the same program.

■ General description

When this function command is executed, a step specified in the same program is called. It makes no difference whether the call destination step is a move command or function command.
When the specified input signal has been input, the step is called; when it has not been input, the step is not called and the robot passes the command by.
Bear in mind that if the call destination step is a function command, the function command at the call destination will be executed as soon as the call command has been executed.
When the step return command is subsequently executed, operation returns to the step following the one which executed the call.

■ Example of operation

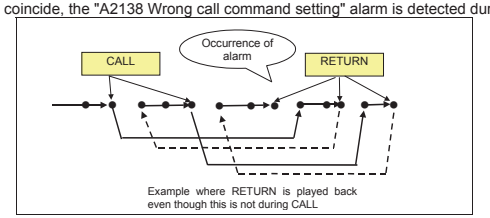
In step 4, record CALLI: conditional step call (FN24), 8 as the number of the call destination step and I1 as the input signal.
If, when this is played back, input signal I1 has been input, the robot jumps to step 8 and advances to steps 9 through 12, and after the RETURN: step return (FN22) command is executed, it returns to the step following the one which executed the call, that is to say, step 5.
If input signal I1 has not been input at step 4, the robot simply advances to step 5.



The robot is not decelerated by this command. If no positioning is specified, the input signal is inspected just before the command value reaches the accuracy range of the move command, and if it has been input, an inside arc is drawn. If the input signal has not been input, the command value heads toward the recorded point, and the input signal is inspected when the robot has reached the accuracy range.

The robot is returned to the step following the one with the CALLI: conditional step call (FN24) command by the RETURN: step return (FN22) command, the RETI: conditional step return (FN25) command or by the RETN: conditional step return after specified number of passes (FN28) command.

The step call command can be executed again at the call destination (between steps 8 and 11 in the above figure). Up to 8 layers of calls can be executed. If calls exceeding 8 layers are executed or the number of step calls and step returns does not coincide, the "A2138 Wrong call command setting" alarm is detected during playback, and the robot stops.



■ **Parameter**

Parameter No. 1	Step No.	This specifies the number of the call destination step.（1-9999）
Parameter No. 2	Input signal	This records the number of the input signal which is to serve as the condition for executing the jump. When number 5101 or above is specified, multiple input signals can be specified. （1-2048、5101～5196）

■ **Example of screen display**

CALLI [8,I1]	FN24; Conditional step call

See
CALL; Step call (FN21)
CALLN: Conditional step call after specified number of passes (FN27) command
RETURN: Step return (FN22)
RETI: Conditional step return (FN25) command
RETN: Conditional step return after specified number of passes (FN28) command

Function commands (FN codes)

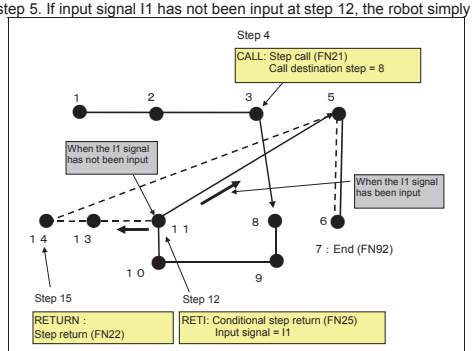
Command name	RETI
FN code	25
Title name	Conditional step return
General description	Using an input signal, this command is used to return the robot to the step following the one which executed the step call command in the same program.

■ **General description**

When this function command is executed, the robot returns to the step following the one which executed the step call command in the same program when the specified signal has been input. It makes no difference whether the return destination step is a move command or function command.
When the specified input signal has been input, the robot returns; when it has not been input, it does not return but passes the command by.
Bear in mind that if the return destination step is a function command, the function command at the return destination will be executed as soon as the return command has been executed.

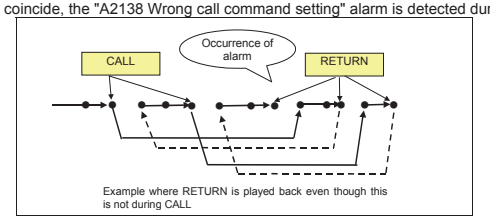
■ **Example of operation**

In step 4, record CALL: step call (FN21) and in step 11, record RETI: conditional step return (FN25), and I1 as the input signal. When this is played back, the robot advances to steps 3 through 8 through 12, and if input signal I1 has been input where the RETI: conditional step return command was executed, the robot returns to the step following the call source step, that is to say, step 5. If input signal I1 has not been input at step 12, the robot simply advances to step 13.



The robot is not decelerated by this command. If no positioning is specified, the input signal is inspected just before the command value reaches the accuracy range of the move command, and if it has been input, an inside arc is drawn. If the input signal has not been input, the command value heads toward the recorded point, and the input signal is inspected when the robot has reached the accuracy range.

The step call command can be executed again at the call destination (between steps 8 and 11 in the above figure). Up to 8 layers of calls can be executed. If calls exceeding 8 layers are executed or the number of step calls and step returns does not coincide, the "A2138 Wrong call command setting" alarm is detected during playback, and the robot stops.



Example where RETURN is played back even though this is not during CALL

■ **Parameter**

Parameter No. 1	Input signal	This records the number of the input signal which is to serve as the condition for executing the return. When number 5001 or above is specified, multiple input signals can be specified. （1-2048、5001～5196）

■ **Example of screen display**

```
RETI [I1]                    FN25; Conditional step return
```

See
CALL; Step call (FN21)
CALLI: Conditional step call (FN24) command
CALLN: Conditional step call after specified number of passes (FN27) command
RETURN: Step return (FN22)
RETN: Conditional step return after specified number of passes (FN28) command

Function commands (FN codes)

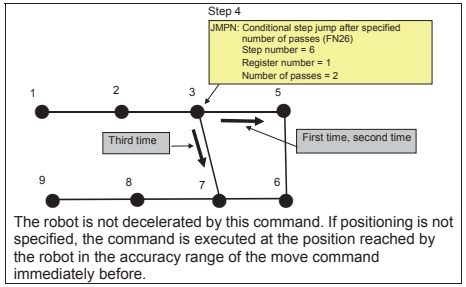
Command name	JMPN
FN code	26
Title name	Conditional step jump after specified number of passes
General description	Using a pass count (number of passes), the robot jumps to the step specified in the same program.

■ **General description**

When this function command is executed, the robot jumps to the step specified in the same program in accordance with the number of passes. It makes no difference whether the jump destination step is a move command or function command.
The robot passes for the specified number of passes, and on the next time (specified number of passes 1) the jump command is executed. (For instance, if "2" is specified as the number of passes, the robot passes twice, and on the third time the jump command is executed.)
Bear in mind that if the jump destination step is a function command, the function command at the jump destination will be executed as soon as the jump command has been executed.

■ **Example of operation**

In step 4, record JMPN: conditional step jump after specified number of passes (FN26), 6 as the number of the jump destination step, 1 as the register number, and 2 as the number of passes.
When this is played back, the robot passes on the first and second times, and it advances to step 5; on the third time, however, it jumps to step 7.



The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

A global integer variable common to all units is used for the number of passes.
The current number of passes can be referenced using monitor/integer variables.

■ **Parameter**

Parameter No. 1	Step No.	This specifies the number of the step serving as the robot's jump destination. （1-9999）
Parameter No. 2	Register number	A "register" refers to the memory used for counting. Since an integer variable (1 to 200) is used, this parameter specifies its number. （1-200）
Parameter No. 3	Number of passes	This records the number of passes which is to serve as the condition for executing the jump. The robot passes for the specified number of passes, and on the next time (specified number of passes 1) the jump command is executed. （0-10000）

■ **Example of screen display**

```
JMPN [6, V1%, 2]     FN26; Conditional step jump after specified
                           number of passes
```

See
JMP; Step jump (FN20)
JMPI: Conditional step jump (FN23)

Function commands (FN codes)

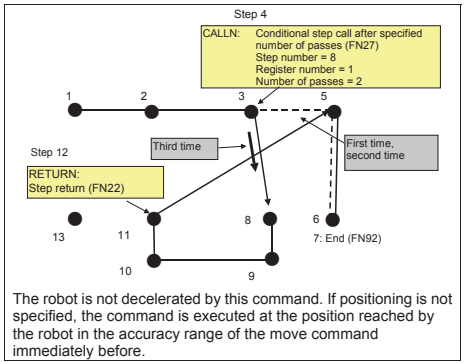
Command name	CALLN
FN code	27
Title name	Conditional step call after specified number of passes
General description	Using a pass count (number of passes), this command is used to call a step specified in the same program.

■ General description

When this function command is executed, the step specified in the same program is called in accordance with the number of passes. It makes no difference whether the call destination step is a move command or function command.
The robot passes for the specified number of passes, and on the next time (specified number of passes 1) the call command is executed. (For instance, if "2" is specified as the number of passes, the robot passes twice, and on the third time the call command is executed.)
Bear in mind that if the call destination step is a function command, the function command at the call destination will be executed as soon as the call command has been executed.
When the step return command is subsequently executed, operation returns to the step following the one which executed the call.

■ Example of operation

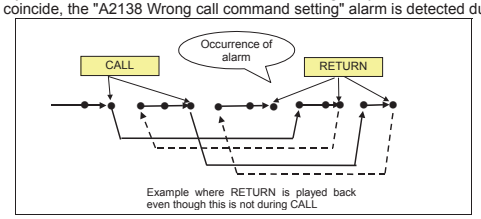
The step call and step return commands are used in tandem.
In step 4, record CALLN: conditional step call after specified number of passes (FN27), 8 as the number of the call destination step, 1 as the register number, and 2 as the number of passes. In step 12, record RETURN: step return (FN22). When this is played back, the robot passes on the first and second times, and it advances to step 5; on the third time, however, it jumps to step 8, and returns to the step following the step which executed the RETURN: step return FN22) command, that is to say, step 5.



The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

The robot is returned to the step following the one with CALLN: conditional step call after specified number of passes (FN27) command by the RETURN: step return (FN22) command, the RETI: conditional step return (FN25) command or by the RETN: conditional step return after specified number of passes (FN28) command.

The step call command can be executed again at the call destination (between steps 8 and 11 in the above figure). Up to 8 layers of calls can be executed. If calls exceeding 8 layers are executed or the number of step calls and step returns does not coincide, the "A2138 Wrong call command setting" alarm is detected during playback, and the robot stops.



A global integer variable common to all units is used for the number of passes.
The current number of passes can be referenced using monitor/integer variables.

■ Parameter

Parameter No. 1	Step No.	This specifies the number of the call destination step.（1-9999）
Parameter No. 2	Register number	A "register" refers to the memory used for counting. Since an integer variable (1 to 200) is used, this parameter specifies its number.（1-200）
Parameter No. 3	Number of passes	This records the number of passes which is to serve as the condition for executing the call. The robot passes for the specified number of passes, and on the next time (specified number of passes 1) the call command is executed.（0-10000）

■ Example of screen display

CALLN [8, V1%, 2] FN27; Conditional step call after specified number of passes

See
CALL; Step call (FN21)
CALLI: Conditional step call (FN24) command
RETURN: Step return (FN22)
RETI: Conditional step return (FN25) command
RETN: Conditional step return after specified number of passes (FN28) command

Function commands (FN codes)

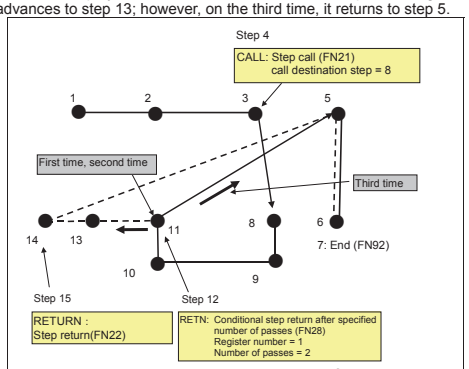
Command name	RETN
FN code	28
Title name	Conditional step return after specified number of passes
General description	Using a pass count (number of passes), this command is used to return the robot to the step following the one which executed the step call command in the same program.

■ General description

When this function command is executed, the robot is returned to the step following the one which executed the step call command in the same program in accordance with the number of passes. It makes no difference whether the return destination step is a move command or function command.
The robot passes for the specified number of passes, and on the next time (specified number of passes 1) the return command is executed. (For instance, if "2" is specified as the number of passes, the robot passes twice, and on the third time the return command is executed.)
Bear in mind that if the return destination step is a function command, the function command at the return destination will be executed as soon as the return command has been executed.

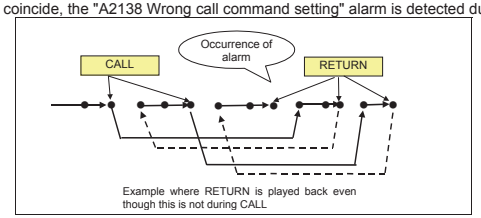
■ Example of operation

In step 4, record CALL: step call (FN21) and in step 11, record RETN: conditional step return after specified number of passes (FN26), 1 as the register number, and 2 as the number of passes.
When this is played back, the robot advances to steps 3 through 8 through 12, and for the first and second times it simply advances to step 13; however, on the third time, it returns to step 5.



The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

The step call command can be executed again at the call destination (between steps 8 and 11 in the above figure). Up to 8 layers of calls can be executed. If calls exceeding 8 layers are executed or the number of step calls and step returns does not coincide, the "A2138 Wrong call command setting" alarm is detected during playback, and the robot stops.



A global integer variable common to all units is used for the number of passes.
The current number of passes can be referenced using monitor/integer variables.

■ Parameter

Parameter No. 1	Register number	A "register" refers to the memory used for counting. Since an integer variable (1 to 200) is used, this parameter specifies its number. (1-200)
Parameter No. 2	Number of passes	This records the "number of passes" which is to serve as the condition for executing the return. The robot passes for the specified number of passes, and on the next time (specified number of passes 1) the return command is executed. (0-10000)

■ Example of screen display

RETN [V1%, 2] FN28; Conditional step return after specified number of passes

See
CALL; Step call (FN21)
CALLI: Conditional step call (FN24) command
CALLN: Conditional step call after specified number of passes (FN27) command
RETURN: Step return (FN22)
RETI: Conditional step return (FN25) command

Command name	RINT
FN code	29
Title name	Robot Interrupt (I-condition)
General description	Executing step is interrupted by input signal

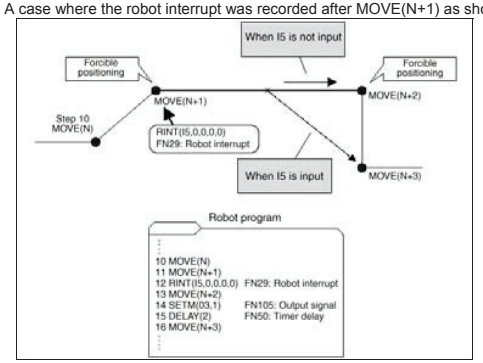
■ General description

Normal application commands are executed after the recording position has been reached. For this reason, it is not possible to conduct such operations as changing the movement path or out-putting the output signals until the target step has been reached. By using this application command, such operations as detecting obstacles and changing the movement path or outputting general-purpose output signals can be conducted while moving between MOVE commands.

The normal procedure to be followed when handling work with a poor positioning accuracy is to use the inputs of the sensors attached to the gripper or other end effecter as the robot interrupt input signals and perform the grip/release operations at the position where contact is made with the work.

■ Example of operation

A case where the robot interrupt was recorded after MOVE(N+1) as shown in the figure is explained as an example.



- The robot interrupt section (range within which the interrupt processing is valid) extends from MOVE(N+1), which is the movement command immediately before the robot interrupt command RINT step, to MOVE(N+2) which is the movement command immediately after this step.
- When the input of the input signal I5 is detected in this section, the process of moving to MOVE(N+2) is exited immediately, and operation transfers to executing the command which has been recorded following MOVE(N+2). Since, in the case of the example provided, the command recorded after the MOVE(N+2) movement command is the SETM application command and the next one is the DELAY application command, these application commands are executed at the position where I5 is detected, and the robot next heads from that position toward the position of the MOVE(N+3) movement command. (Path indicated by the dotted line in the figure)
- The steps at both ends of the robot interrupt section (the MOVE(N+1) and MOVE(N+2) movement commands in the case of the figure) forcibly entail the positioning of the robot irrespective of what has been recorded.
- If the input of the input signal (I5) has not been detected by the time when the position where MOVE(N+2) was recorded is reached, the robot moves as usual up to the position where MOVE(N+2) was recorded, and the next SETM (FN105) command is executed.
- When the input of the input signal (I5) is detected as the robot heads toward MOVE(N+2) and then the robot heads toward the position where MOVE(N+3) was recorded, it moves along the positioning path.
- Record the MOVE(N+2) movement command at a low recording speed. If the robot is operated at a high speed, the response to the interrupt detection may become slow.

■ Parameter

Parameter No. 1	Input signal	This is used to specify the number of the input signal serving as the robot interrupt condition. (1 to 2048): General-purpose input signal number (5101 to 5196): Multiple input signal number	
Parameter No. 2	Search position	Search basis write setting status 0: No done 1: Done	These parameters are used by the search function. When a robot interrupt command is used on its own, use the “0” setting for all the parameters. **For details on the search function, refer to SEA: Search(FN59).**
Parameter No. 3	Standard X	X,Y and Z coordinate value when a search basis write has been performed (−3000～3000)	
Parameter No. 4	Standard Y		
Parameter No. 5	Standard Z		

■ Example of screen display

```
RINT[34,0,0,0,0]   FN29: Robot Interrupt (I-condition)
```

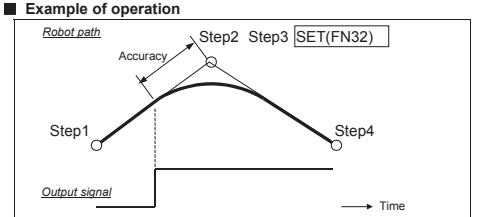
See
SEA; Search (FN59)

Command name	SET
FN code	32
Title name	Output signal ON
General description	This command is used to set one of the general-purpose output signals to ON.

■ General description

When this function command is executed, it is possible to set any one of the general-purpose output signals (O1 to O2048) to ON. However, the command cannot set any of the status signals (signals with pre-assigned applications such as the gun signals and starting signal) to ON.
Which signals have been assigned as status signals can be identified on the monitor screen. Signals whose numbers are displayed in the bold italics are status signals so any of the other signals can be set to ON.

■ Example of operation



The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

■ Parameter

Parameter No. 1	Output signal number	This specifies the number of the general-purpose output signal which is to be set to ON. (1-2048)

■ Example of screen display

```
SET[O12]          FN32; Output signal ON
```

See
ALLCLR; All output signals clear (FN0)
RESET; Output signal OFF (FN34)
SETO; Consecutive output signal ON/OFF (FN100)
SETM ; Output signal ON/OFF (FN105)

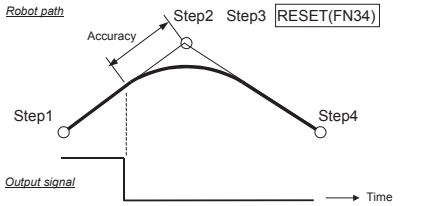
Function commands (FN codes)

Command name	RESET
FN code	34
Title name	Output signal off
General description	This command is used to set one of the general-purpose output signals to OFF.

■ General description

When this function command is executed, it is possible to set any one of the general-purpose output signals (O1 to O2048) to OFF. However, the command cannot set any of the status signals (signals with pre-assigned applications such as the gun signals and starting signal) to OFF.
Which signals have been assigned as status signals can be identified on the monitor screen. Signals whose numbers are displayed in the bold italics are status signals so any of the other signals can be set to ON.

■ Example of operation



The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

■ Parameter

Parameter No. 1	Output signal number	This specifies the number of the general-purpose output signal which is to be set to OFF.（1-2048）

■ Example of screen display

```
RESET [O12]          FN34; Output signal OFF
```

See
ALLCLR: All output signals clear (FN0)
SET; Output signal OFF (FN32)
SETO: Consecutive output signal ON/OFF (FN100)
SETM: Output signal ON/OFF (FN105)

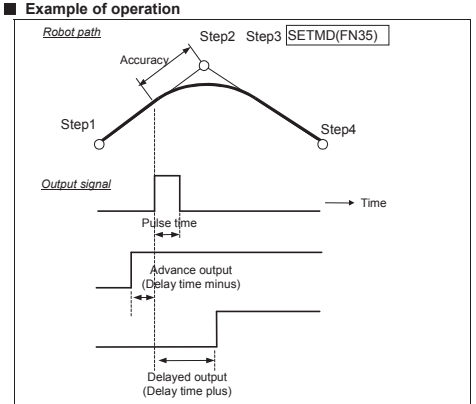
Function commands (FN codes)

Command name	SETMD
FN code	35
Title name	Output with pulse or delay ON/OFF
General description	This command is used to set one of the general-purpose output signals to come with a pulse or delay and to ON or OFF.

■ General description

When this function command is executed, it is possible to set any one of the general-purpose output signals (O1 to O2048) to ON or OFF. Furthermore, the command enables pulse output, advance output or delayed output to be specified.
However, the command cannot set any of the status signals (signals with pre-assigned applications such as the gun signals and starting signal) to ON or OFF. Which signals have been assigned as status signals can be identified on the monitor screen. Signals whose numbers are displayed in the bold italics are status signals so any of the other signals can be set to ON or OFF.

■ Example of operation



The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

If an advanced output or delayed output has been specified, the output can cover both the move command immediately before (step 1 in the above figure) and the move command immediately after (step 4 in the above figure).

■ Parameter

Parameter No. 1	Output signal number	This specifies the number of the general-purpose output signal which is to be set to ON or OFF. (1-2048)
Parameter No. 2	ON/OFF	"1" is specified for ON, and "0" for OFF. (0-1)
Parameter No. 3	Delay time	If "0.0" is specified as the time, the command is executed at the timing which coincides with the recorded point. If a minus value is specified, the command is output ahead of the original execution timing by the amount equivalent to the delay time setting. Conversely, if a plus value is specified, it is output after the timing by the amount equivalent to the delay time setting. (Increment: seconds) In either case, it can be executed beyond the time lapse to the step before or after. (-10.0 – 10.0)
Parameter No. 4	Pulse time	This is set when the output signal is to be output as a pulse signal. It is used to specify the width of the pulse signal. When "0.0" is specified as the time, a level signal is output. (Increment: seconds) (0.0 – 10.0)

■ Example of screen display

```
SETMD [O12, 1, -5, 3]   FN35;     Output with pulse or delay ON/OFF
```

See
ALLCLR: All output signals clear (FN0)
SET: Output signal ON (FN32)
RESET: Output signal OFF (FN34)
SETO: Consecutive output signal ON/OFF (FN100)
SETM: Output signal ON/OFF (FN105)

Function commands (FN codes)

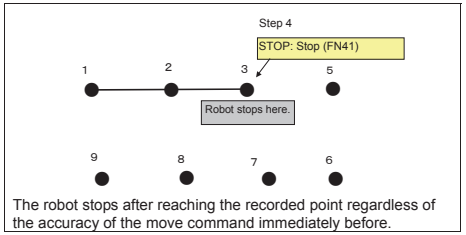
Command name	STOP
FN code	41
Title name	Stop
General description	This command is used to stop the robot.

■ General description

When this function command is executed, the robot is stopped.
The robot will not start unless the start command is input again.

■ Example of operation

In step 4, record STOP: stop (FN41).
When this is played back, the robot stops at step 4. This command does not establish the program end status (status established by executing the END: FN92 command to end the program): this means that the robot will head to step 5 without returning to the first step if it is restarted at the step 4 position. This should be borne in mind.



The robot stops after reaching the recorded point regardless of the accuracy of the move command immediately before.

■ Parameter

None

■ Example of screen display

STOP FN41; Stop

See
STOPI: Conditional stop (FN42)
END: End (FN92)

Function commands (FN codes)

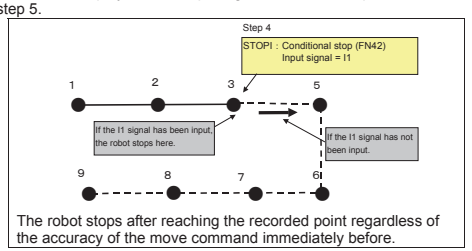
Command name	STOPI
FN code	42
Title name	Conditional stop
General description	Using an input signal, this command is used to stop the robot.

■ General description

When this function command is executed, the robot is stopped.
When the specified input signal has been input, the robot stops; when it has not been input, the robot does not stop and passes.
When the robot has stopped, it will not start unless the start command is input again.

■ Example of operation

In step 4, record STOPI: conditional stop (FN41), and I1 as the input signal.
If, when this is played back, input signal I1 has been input, the robot stops at step 4, and if it has not been input, it advances to step 5.



The robot stops after reaching the recorded point regardless of the accuracy of the move command immediately before.

■ Parameter

Parameter No. 1	Input signal	This records the number of the input signal which is to serve as the condition for executing the stop. When number 5101 or above is specified, multiple input signals can be specified. (1-2048、5101～5196)

■ Example of screen display

STOPI[I1] FN42; Conditional stop

See
STOP: Stop (FN41)

Function commands (FN codes)

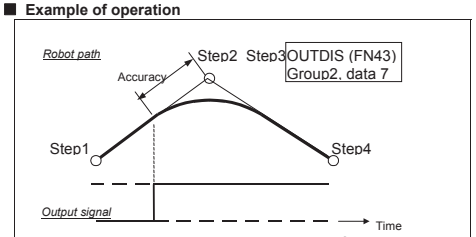
Command name	OUTDIS
FN code	43
Title name	Discrete format output signals
General description	This command is used to set the general-purpose output signals in any group to ON or OFF using the discrete format.

■ General description

When this function command is executed, it is possible to set the general-purpose output signals (O1 to O2048) in any group to ON or OFF using the discrete format.
The general-purpose output signals are divided up into groups of ten: "group" here refers to such a group. With the discrete format, by specifying a number, any one signal inside a group of ten signals can be set to ON while the other signals are set to OFF.
However, the command cannot set any of the status signals (signals with pre-assigned applications such as the gun signals and starting signal) to ON or OFF.
Which signals have been assigned as status signals can be identified on the monitor screen. Signals whose numbers are displayed in the bold italics are status signals so any of the other signals can be set to ON or OFF.

Group	Output signal	Group	Output signal	Group	Output signal
1	1～10	11	101～110	21	201～210
2	11～20	12	111～120	…	…
3	21～30	13	121～130	30	291～300
4	31～40	14	131～140	…	…
5	41～50	15	141～150	50	491～500
6	51～60	16	151～160	…	…
7	61～70	17	161～170	100	991～1000
8	71～80	18	171～180	…	…
9	81～90	19	181～190	…	…
10	91～100	20	191～200	204	2031～2040

■ Example of operation



The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

When "group 2, data 7" has been specified

Output signal number	11	12	13	14	15	16	17	18	19	20
ON/OFF status	○	○	○	○	○	○	●	○	○	○

○: OFF, ●: ON

■ Parameter

Parameter No. 1	Group Number	This specifies the number of the group which is to be output. (1–204)
Parameter No. 2	Data	This specifies the data which is to be output. Since ten signals per group are specified using the discrete format, the maximum setting range for the data is 10. (1–10)

■ Example of screen display

OUTDIS [Group2, 7] FN43; Discrete format output signals

See
OUT: Binary format output signal (FN44)

Function commands (FN codes)

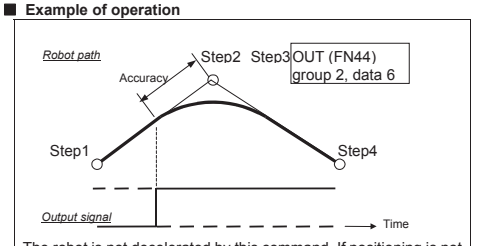
Command name	OUT
FN code	44
Title name	Binary format output signals
General description	This command is used to set the general-purpose output signals in any group to ON or OFF using the binary format.

■ General description

When this function command is executed, it is possible to set the general-purpose output signals (O1 to O2048) in any group to ON or OFF using the binary format.
The general-purpose output signals are divided up into groups of ten: "group" here refers to such a group. With the binary format, a number is specified, this number is then converted into a binary number and used to set the signals inside the corresponding group to ON or OFF.
However, the command cannot set any of the status signals (signals with pre-assigned applications such as the gun signals and starting signal) to ON or OFF.
Which signals have been assigned as status signals can be identified on the monitor screen. Signals whose numbers are displayed in the bold italics are status signals so any of the other signals can be set to ON or OFF.

Group	Output signal	Group	Output signal	Group	Output signal
1	1～10	11	101～110	21	201～210
2	11～20	12	111～120	…	…
3	21～30	13	121～130	30	291～300
4	31～40	14	131～140	…	…
5	41～50	15	141～150	50	491～500
6	51～60	16	151～160	…	…
7	61～70	17	161～170	100	991～1000
8	71～80	18	171～180	…	…
9	81～90	19	181～190	…	…
10	91～100	20	191～200	204	2031～2040

■ Example of operation



The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

When "group 2, data 6" has been specified
As a binary number "6" is 00 0000 0110.

Binary number	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Output signal number	20	19	18	17	16	15	14	13	12	11
ON/OFF status	○	○	○	○	○	○	○	●	●	○

○: OFF, ●: ON

■ Parameter

Parameter No. 1	Group number	This specifies the number of the group which is to be output. (1-204)
Parameter No. 2	Data	This specifies the data which is to be output. Since ten signals per group are specified using the binary format, the maximum setting range for the data is 1023. (0-1023)

■ Example of screen display

OUT[OB2, 6] FN44; Binary format output signals

See
OUTDIS: Discrete format output signals (FN43)

Function commands (FN codes)

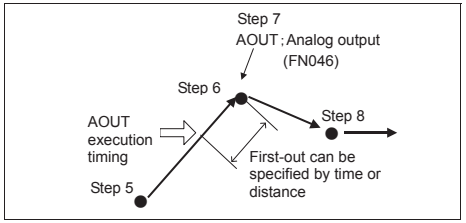
Command name	AOUT
FN code	46
Title name	Analog output
Outline	The TCP (robot tool center point) linear speed and other data are output as analog voltages

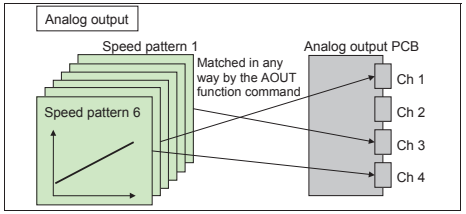
■ General description

By using these function commands, the TCP (robot tool center point) linear speed and other data can be output externally as analog voltages. They are useful for sealing and other applications. The TCP speed, a direct specification or OFF can be selected as the output voltage. If TCP speed has been specified, the TCP speed output data in the constant setting mode must be designed ahead of time. (→ Constants/TCP speed data)

There are 4 channels for the analog output, and the data can be output from any of the ports. The command cannot be used if the analog output PCB (option) has not been installed.

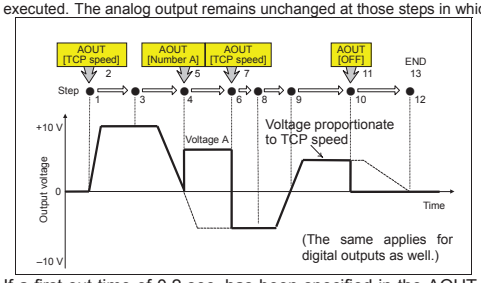
This command can be executed before the recorded point is reached. Specify its amount as a time or distance.





■ Example of operation

Record the AOUT command (FN46) in steps 2, 5, 7 and 11 in the figure shown below.
When these steps are played back, the analog output changes to the specified data each time the AOUT command (FN46) is executed. The analog output remains unchanged at those steps in which the AOUT command (FN46) has not been recorded.



If a first-out time of 0.2 sec. has been specified in the AOUT command of step 7 in this example, the analog voltage changes from numerical data A to the TCP speed 0.2 sec. before the robot reaches step 6 (the previous movement command).

In the teach mode, any analog voltage can be output by performing a manual operation (shortcut R206).
The analog voltage which is output last is held even if the mode is switched between teach and playback.

■ Parameter

Parameter No. 1	Channel number	This is used to specify the number of the port from which the analog values will be output. (1 to 4)
Parameter No. 2	Output signal type	This is used to specify the type of data to be output as analog data. (0 to 2) 0: OFF (0 V is output) 1: TCP speed 2: Directly specified
Parameter No. 3	Output data	When output signal type = 0: This recorded data is not used. When output signal type = 1: Specify the number of the pre-designed TCP speed pattern.(0 to 6) When "0" is specified, OFF (0 V) is established. When output signal type = 2: The parameter is used to directly specify the output voltage. (–10 V to 10 V)
Parameter No.4	Pre-out type	This enables the output to be started before the recorded point is reached. Select the specification method. (0 to 1) 0: Specified as a time 1: Specified as a distance
Parameter No. 5	Pre-out data	This is used to initiate pre-out using a negative numerical value. When pre-out type = 0: Specify it as a time. (–1.0 to 0 sec.) When pre-out type = 1: Specify it as a distance. (–500 to 0 mm)

■ Example of screen display

```
AOUT[2,1,6,1,-100]        FN46; Analog output
```

See
DOUT: Digital output (FN278)
DPRESETM: Distance specification output preset (FN280)

Command name	PALLET2
FN code	47
Title name	Palletize start
Outline	Start palletizing based on the pre-designed palletizing pattern.

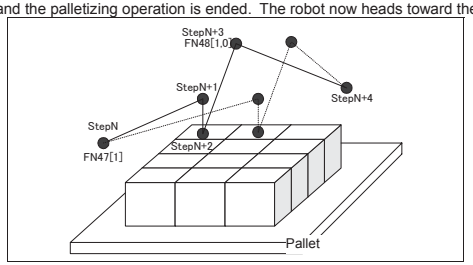
■ General description

When this function command is executed, the specified palletizing can be started.
It is executed in tandem with the FN48 "Palletize end" command which is used to end palletizing.

■ Example of operation

As shown in the figure below, record “PALLET2: Palletize start” (FN47) in step N and “PALLET2 END: Palletize end” (FN48) in step N+3.

When the program is played back and the robot reaches step N, the shift amount is calculated from the palletize number specified by FN47 and the palletize counter, and the shift operation is performed. Step N+1 and step N+2 are shifted. (The robot passes along the path indicated by the dotted lines in the figure below.) When it reaches step N+3, FN48 is executed, and the palletizing operation is ended. The robot now heads toward the point where step N+4 is recorded.



■ Parameter

	Data	Description, setting range
Parameter No. 1	Palletize No.	This specifies the palletizing number to be executed. (1 to 200)

■ Example of screen display

```
PALLET2[999]              FN47: Palletize start
```

See
PALLET2_END: Palletize end (FN48)
PALLET2_RESET: Palletize reset (FN49)
DSPALLET: Direction select palletize (FN65)

Command name	PALLET2__END
FN code	48
Title name	Palletize end
Outline	Finish palletizing based on the pre-designed palletizing pattern.

■ General description

When this function command is executed, the specified palletizing work can be completed.
It is executed in tandem with the FN47 "Palletize start" command which is used to start palletizing.

■ Example of operation

As shown in the figure below, record “PALLET2: Palletize start” (FN47) in step N and “PALLET2 END: Palletize end” (FN48) in step N+3.

When the program is played back and the robot reaches step N, the shift amount is calculated from the palletize number specified by FN47 and the palletize counter, and the shift operation is performed. Step N+1 and step N+2 are shifted. (The robot passes along the path indicated by the dotted lines in the figure below.)
When it reaches step N+3, FN48 is executed, and the palletizing operation is ended. The robot now heads toward the point where step N+4 is recorded.

■ Parameter

	Data	Description, setting range
Parameter No. 1	Palletize No.	This specifies the number of the palletizing operation which is to be ended. (1 to 200)
Parameter No. 2	Output signal	This specifies the number of the output signal. When all the palletizing tasks specified have been completed, the general-purpose output signal specified is set to ON. (0 to 2048)

■ Example of screen display

```
PALLET2_END[200,01]              FN48: Palletize end
```

See
PALLET2: Palletize start (FN47)
PALLET2_RESET: Palletize reset (FN49)
DSPALLET: Direction select palletize (FN65)

Command name	PALLET2_RESET
FN code	49
Title name	Palletize reset
Outline	When a condition signal has been input, the palletize counter is forcibly reset. (palletizing operation is forcibly terminated)

■ **General description**

If the condition signal (input signal) is at the ON status when the command is executed, the specified palletize counter can be forcibly reset (cleared to zero).

The fact that the counter has been reset can be output to an external source using an output signal. Furthermore, it is possible to jump to a specified step only when the counter has been reset.

■ **Example of operation**

As shown in the figure below, record "PALLET2: Palletize start" (FN47) in step N, "PALLET2_END: Palletize end" (FN48) in step N+3, and "PALLET2_RESET: Palletize reset" (FN49) in step N+2.

When the program is played back and the robot reaches step N, the shift amount file corresponding to the palletize number specified by FN47 is loaded, and the shift operation is performed. Normally, step N+1 and step N+2 are shifted, and when the robot reaches step N+3, the FN49 command is executed to end the palletizing operation.

However, if the reset condition signal (the I1 signal in the example) has been input when the robot reached step N+2, an acknowledge signal (O1 in the example) is output, the palletize counter is reset, and the robot heads toward the point where step N+4 is recorded.

■ **Parameter**

	Data	Description, setting range
Parameter No. 1	Palletize No.	This is used to specify the palletizing number for which the counter is to be reset. (1 to 200)
Parameter No. 2	Input Signal	This is used to specify the number of the input signal serving as the condition for resetting the palletizing. By specifying numbers from 5101 to 5196, a multiple number of input conditions can be specified. (1 to 5196)
Parameter No. 3	Output signal number	This is used to specify the number of the output signal (acknowledge signal) which indicates that the palletizing has been reset. (1 to 2048) * If <All output> has been selected as the setting for the "Palletize acknowledge signals" under "Handling teach/playback conditions," the output signals which have been specified here are output regardless of whether the reset process is to be executed or not.
Parameter No.4	Step No.	This is used to specify the number of the jump destination step. Operation jumps to the step specified here only when palletizing has been reset. (0 to 9999)

■ **Example of screen display**

```
PALLET2_RESET[200, I1, O5, 99]    FN49: Palletize reset
```

See

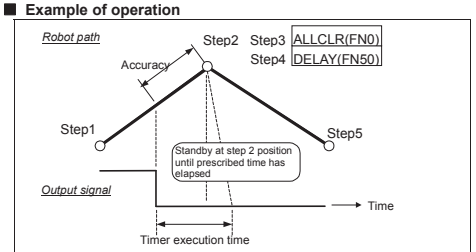
PALLET2: Palletize start (FN47)
PALLET2_END: Palletize end (FN48)
DSPALLET: Direction select palletize (FN65)

Command name	DELAY
FN code	50
Title name	Timer
General description	This command is used to place the robot in the standby status.

■ **General description**

When this function command is executed, the robot is placed in the standby status. During standby, the robot rests at a recorded point.

■ **Example of operation**



If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

With the move command immediately before the timer command, the in-position check is not conducted so that the command value passes through the recorded point without fail. This means that even when "0 seconds" has been set for the timer, the cycle time will be longer compared with when no recording is performed.

■ **Parameter**

Parameter No. 1	Standby time	This specifies the standby time in increments of seconds. (0－60. 0)

■ **Example of screen display**

```
DELAY [3.0]                    FN50; Timer
```

See

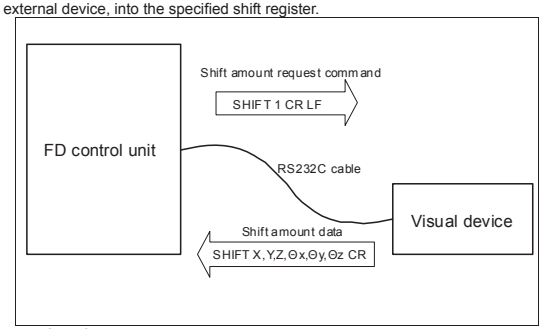
WAIT: Input signal wait with timer (FN552)

Function commands (FN codes)

Command name	SREQ
FN code	51
Title name	Shift amount request
Outline	The command requests the shift amount data from the external device using the serial port. Once it has been input from the external device, the shift amount data is stored in the specified shift register.

■ General description

This command sends a character string (command) requesting a shift amount to the external device connected to the controller by the RS232C cable, and it sets the shift amount data character string, which is input as the response from the external device, into the specified shift register.



The RS232C cable is an optional accessory.

When this function command is executed, the character string data below is output from the RS232C port, and the shift amount data is thereby requested.

SHIFT *1 CR LF (*1=Register number, CR=0x0d, LF=0x0a)

The robot continues to operate even after the request data has been output.

When the shift amount data is input from the external device in the following format, it is stored in the specified shift register.

SHIFT X,Y,Z,dX,dY,dZ CR (all values are real numbers)

The receive time, wait time and communication conditions are set by selecting "8 Communication" from "Constant Setting" and then "1 Serial port." The data input wait time can be set in "Timeout time," and if this is set to "0," the serial port will remain in the wait status until an input arrives. (The robot continues to operate.)

■ Example of operation

As shown in the figure, when SREQ (FN51) is executed at step N+3, the shift amount request command is sent to the external device through the serial port, and it is ensured that the data can be received at any time. The robot continues to operate as is and heads toward step N+4. When the shift data is received from the external device at any position, the legitimacy of the data is checked, and the data is stored in the specified shift register only when the data is found to be legitimate.

But received shift data is committed after executing the function “WAITR: Wait shift value receive” (FN127) which is programmed at step N+6.

This shift register value is used for the shift operations by the "SHIFTR: Shift2" (FN52) or such other command which is programmed in step N+7.

IMPORTANT

If you use SREQ, you must put “RSCLR: Buffer clear” (FN111) at the step before SREQ.
If you don’t use “RSCLR: Buffer clear” (FN111), robot might move as you don’t want.

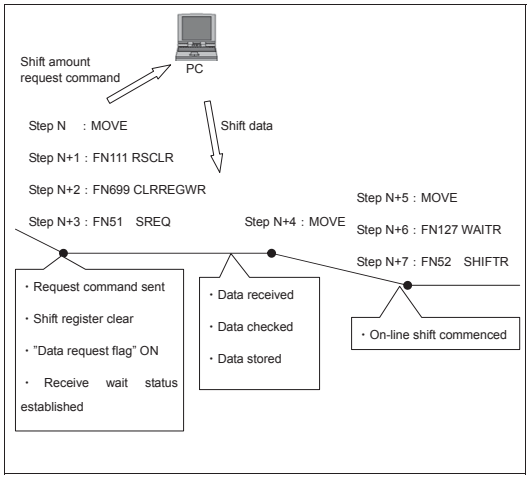
IMPORTANT

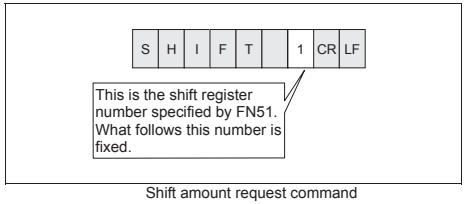
If you use SREQ and there is possibility that the specified shift register is written, you must put “CLRREGWR: Clear register of written sts” (FN699) at the step before SREQ.
If you don’t use “CLRREGWR: Clear register of written sts” (FN699), Received shift data might be not used.
The case CLRREGWR needed

1. “Shift clearance on step 0” is set “No clear.
2. The specified shift register at SREQ is used “LETR; Set shift value” (FN68) etc.
3. In case that you use SREQ command again at the shelter step which is specified by WAITR; Wait shift value receive” (FN127) etc.

IMPORTANT

If you use SREQ, you must put “WAITR: Wait shift value receive” (FN127) at the step before using the specified shift register.
If you don’t use “WAITR: Wait shift value receive” (FN127), Received shift data aren’t used.





Shift amount request command

«Example of response 1»
For shift amounts when only X value is 10 mm

S H I F T 1 0 . 0 . 0 . 0 . 0 . 0 CR

«Example of response 2»
For shift amounts when only θz value is 5 degrees (abbreviation)

S H I F T . . . . . 5 CR

Response data

If the response data from the external device is sent in any other format, the values will not be set correctly in the shift register.

■ Parameter

Parameter No. 1	Shift register number	This is used to specify the number of the shift register in which to store the shift amount received from the external device. (1 to 9)
Parameter No. 2	Port number	This is used to specify the number of the port to be used to transfer the data. At the present time, only port 1 can be used. (1 to 1)

■ Example of screen display

```
SREQ[R1, 1]                FN51: Shift amount request
```

See
SHIFTR: Shift2 (FN52)
RSCLR: Buffer clear (FN111)
WAITR: Wait shift value receive (FN127)
CLRREGWR: Clear register of written sts (FN699)

Function commands (FN codes)

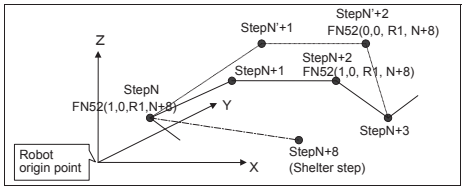
Command name	SHIFTR
FN code	52
Title name	Shift
Outline	The start or end of the shift operation is specified. When shift operation start has been specified, the shift operation is performed on the basis of the shift amount stored in the specified shift register.

■ General description

This function command proceeds with playback while shifting the recorded position in the robot program on the basis of the shift amount data stored in the specified shift register. One of four options — “Machine coordinates (robot coordinates)”, “Tool coordinates,” “User coordinates” or “Absolute coordinates (world coordinates)” — can be selected for the coordinate system to be moved.
If the shift amount data has not been set in the specified shift register, it is possible to jump to the shelter step. Alternatively, the robot can be stopped immediately without escaping.

■ Example of operation

As shown in the figure, shift start is recorded at the position (step N) where the shift is to start, and shift end is recorded at the position (step N+2) where shift is to end.
When the program is played back, the robot reads the contents of the shift register specified by FN52 after it has reached step N, and it then moves toward the position (step N'+1) established by shifting the next target position (step N+1). The position established by similarly shifting the recorded position as far as the position (step N+2) where shift end has been recorded serves as the target position. (Path of dotted line in figure below)



If, when FN52 is executed at step N, the shift data has not been set at the time when the specified shift register is read, the robot moves toward the shelter step in the event that an shelter step has been set by the FN52 command. (Path of alternate long and short dash line in figure above)

■ Parameter

Parameter No. 1	Start/end	This is used to specify the start or end of the shift operation. 1: Start / 0: End
Parameter No. 2	Coordinate system	This is used to specify the coordinate system to be shifted. 0: Machine coordinates (robot coordinates) 1: Tool coordinates 2: User coordinate 3: Absolute coordinates (world coordinates) If user coordinates are to be specified, they must first be registered in "10 User coordinate system registration" selected on the Service Utilities menu, and then the number of the user coordinates to be used by the shift must be selected using "FN113 shift coordinate system selection" in the application command.
Parameter No. 3	Shift register number	This is used to specify the shift register number. (1 to 9)
Parameter No.4	Shelter step	This is used to specify the number of the shelter step when the shift amount data was not set in the specified shift register. (0 to 10000) When 10000 is specified as the shelter step number, an alarm (A2118: “No data has been input in shift register.”) results immediately with no escape operation performed, and the robot can be stopped.

■ Example of screen display

```
SHIFTR[0, 0, R1, 10]          FN52:Shift2
```

See
SREQ: Shift amount request (FN51)

Function commands (FN codes)

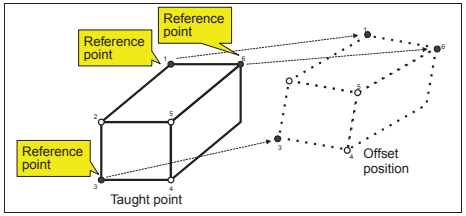
Command name	LOCCVT
FN code	53
Title name	Coordinate transform (Offset amount)
Outline	It is possible to proceed with playback while offsetting each recorded point based on the difference (skew amount) measured beforehand between the recorded position of the three points serving as the reference and the actual position obtained from the visual device, etc.

■ General description

When this function command is used, it is possible to proceed with playback while offsetting the skew based on the skew amount (mm) between the recorded position of the three points serving as the reference and the actual position obtained from the visual device or other external device.
Any already taught three points can be specified as the three points serving as the reference. The skew amounts are stored in shift registers 1, 2 and 3.
Before this application command is executed, the SREQ; shift amount request (FN51) command, etc. must be played back and the skew amount entered.

■ Example of operation

As shown in the figure, the LOCCVT command is recorded with the three points of steps 1, 3 and 6 which were originally taught serving as the reference points. Before the FN53 command is played back, the skew amount corresponding to the offset position (at the right in the figure below) must first be set in shift registers 1, 2 and 3. When FN53 is now played back, a coordinate transform matrix is generated from the three reference points at taught and the three reference points at offset, and the recorded positions are shifted according to this matrix and played back.



When the shift operation end is executed by FN53, the shift playback ends, and the recorded positions are played back.
* As the three reference points, specify the points in such a way that other recorded positions are enclosed.
* Specify the three reference points in such a way that they are not aligned on a single straight line.
* The θx, θy and θz data in the shift registers are not used.

■ Parameter

Parameter No. 1	Start/end	This is used to specify the start or end of the shift operation. 1: Start / 0: End
Parameter No. 2	Reference step 1	This is used to specify the first step to be used as the reference. (1 to 9999)
Parameter No. 3	Reference step 2	This is used to specify the second step to be used as the reference. (1 to 9999)
Parameter No.4	Reference step 3	This is used to specify the third step to be used as the reference. (1 to 9999)

■ Example of screen display

```
LOCCVT[1, 1, 2, 3]     FN53:Coordinate transform (Offset amount)
```

See
LOCCVT1: Coordinate transform (Coordinate values) (FN54)

Function commands (FN codes)

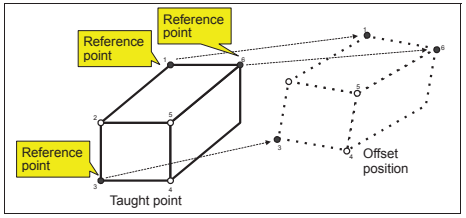
Command name	LOCCVT1
FN code	54
Title name	Coordinate transform (Coordinate values)
Outline	It is possible to proceed with playback while offsetting each recorded point based on the difference (skew amount) measured beforehand between the recorded position of the three points serving as the reference and the actual position obtained from the visual device, etc.

■ General description

When this function command is used, it is possible to obtain the actual coordinates of the three points serving as the reference from the visual device or other external device, and proceed with playback while offsetting the coordinates on the basis of the data obtained.
Any already taught three points can be specified as the three points serving as the reference.
The coordinates are stored in shift registers 1, 2 and 3.
Before this function command is executed, the "SREQ: Shift amount request" (FN51) command, etc. must be played back and the skew amount entered.

■ Example of operation

As shown in the figure, the LOCCVT1 command is recorded with the three points of steps 1, 3 and 6 which were originally taught serving as the reference points. Before the FN54 command is played back, the coordinates of the offset position (at the right in the figure below) must first be set in shift registers 1, 2 and 3. When FN54 is now played back, a coordinate transform matrix is generated from the three reference points at taught and the three reference points at offset, and the recorded positions are shifted according to this matrix and played back.



When the shift operation end is executed by FN54, the shift playback ends, and the recorded positions are played back.

* As the three reference points, specify the points in such a way that other recorded positions are enclosed.
* Specify the three reference points in such a way that they are not aligned on a single straight line.
* The θx, θy and θz data in the shift registers are not used.

■ Parameter

Parameter No. 1	Start/end	This is used to specify the start or end of the shift operation. 1: Start / 0: End
Parameter No. 2	Reference step 1	This is used to specify the first step to be used as the reference. (1 to 9999)
Parameter No. 3	Reference step 2	This is used to specify the second step to be used as the reference. (1 to 9999)
Parameter No.4	Reference step 3	This is used to specify the third step to be used as the reference. (1 to 9999)

■ Example of screen display

```
LOCCVT1[1,1,2,3]   FN54:Coordinate transform (Coordinate values)
```

See
LOCCVT: Coordinate transform (Offset amount) (FN53)

Function commands (FN codes)

Command name	CNVSYNC
FN code	55
Title name	Conveyor Counter Reset
General description	Reset conveyor counter

■ General description

This function is used in conveyor synchronizing application.
Conveyor counter (read value of conveyor pulse) is reset at step0 forcibly, but this function allows to clear conveyor counter in any step.

Please refer to the **"Conveyor Synchronization Manual"** (option) for detail operations.

■ Example of operation

Conveyor Counter is reset where the step CNVCYNC：Conveyor Counter Reset（FN55）is executed.

■ Parameter

Parameter No. 1	Conveyor number	Conveyor number which counter should be reset.(1-*) (* is conveyor quantity defined)

■ Example of screen display

```
CNVSYNC[1]          FN55: Conveyor Counter Reset
```

See
CNVI; Conveyor Interlock (FN550)
CNVYSYNCI; Synchronizing Conveyor Interlock (FN562)

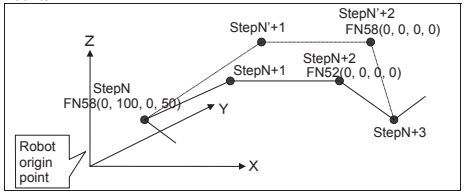
Command name	SHIFTA
FN code	58
Title name	XYZ shift
Outline	The playback position is shifted in parallel

■ General description

The XYZ shift function shifts the position which has already been taught in a parallel direction for each of the 3 dimensions. During the shift operations, the tool poses are maintained. One of four options — "Machine coordinates (robot coordinates)", "Tool coordinates," "User coordinates" or "Absolute coordinates (world coordinates)" — can be selected for the coordinate system to be moved.
The shift amounts are recorded as parameters. In other words, this command is useful when the amounts of the parallel movement by which the robot is to be shifted are already known.
The shift operation is considered to have ended when 0.0 mm is specified for all the shift amounts recorded in FN58.
Use of this shift function will not update the actual recorded positions in the robot program.

■ Example of operation

In the example shown in the figure, the XYZ shift (FN58) command with the shift amounts specified is recorded at step N at which the shift operation is to be started, and the XYZ shift command with zero recorded for all the shift amounts is recorded at the shift end step.
When the program is played back, the robot, after it has reached step N, plays back toward the target position, step N'+1, which is obtained by adding the shift amount in the specified coordinate system recorded in FN58 to the next recorded position (step N+1). After the robot has reached step N+2, the shift operation is ended by executing FN58 with 0 mm set for all the shift amounts.



If positions are to be shifted based on the user coordinate system, "User coordinate system registration" must be performed, and the user coordinate numbers must be selected using the "CHGCOORD: Change coord. No. (shift)" (FN113) command beforehand.

If a specified shift amount exceeds the value specified in "Shift Amount Limit" selected from "Machine constants" under "Constant Setting," an error results, and the robot is stopped.

This function can be used together with other shift functions. If shift amounts in the reverse direction from the ones specified by FN58 have been specified by another shift-related command, they will be set to 0.0 mm as a result.

■ Parameter

Parameter No. 1	Coordinate system	The parallel movement amounts specified by the parameters No. 2 to 4 serve as the movement amounts in the coordinate system which is specified here. A value from 0 to 3 is specified by this parameter for the coordinates. 0: Machine coordinates (robot coordinates) 1: Tool coordinates 2: User coordinates 3: Absolute coordinates (world coordinates) If user coordinates are to be specified, they must first be registered in "10 User coordinate system registration" selected on the Service Utilities menu, and then the number of the user coordinates to be used by the shift must be selected using "FN113 shift coordinate system selection" in the function command.
Parameter No. 2	Shift value X	This is used to specify the shift amount in the X direction as defined by the coordinate system specified by the parameter No.1. (–3000 mm to 3000 mm)
Parameter No. 3	Shift value Y	This is used to specify the shift amount in the Y direction as defined by the coordinate system specified by the parameter No.1. (–3000 mm to 3000 mm)
Parameter No.4	Shift value Z	This is used to specify the shift amount in the Z direction as defined by the coordinate system specified by the parameter No.1. (–3000 mm to 3000 mm)

■ Example of screen display

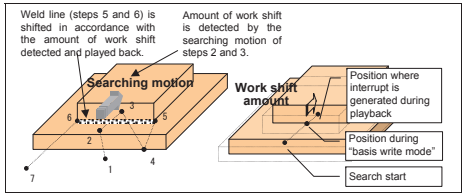
```
SHIFTA[0, -10.5, 23.3, 0.0]        FN58: XYZ Shift
```

See

Command name	SEA
FN code	59
Title name	Search
General description	Detect the work position shift amount, and store those data to the shift register

■ General description

By using this application command, it is possible to detect shift amounts when there are variations in the work positions and store those amounts in the shift register. By using the shift amounts stored in the shift register, any multiple number of steps can be shifted altogether by the shift command. When shift amounts in the work positions are to be detected, the search command must be used together with the robot interrupt signal. This is the case so as to capture the position where contact is made with the work.

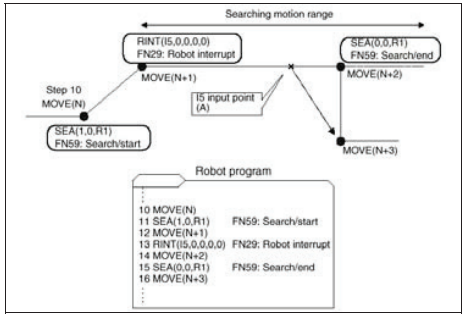


Before calculating the shift amounts, the work position serving as the basis must be obtained. This is called **the basis write mode**. The difference between the basis position obtained in the basis write mode and the actual work positions obtained in the normal mode are stored in the specified shift register as the shift amount. The recording positions are then shifted using the shift amount stored in the shift register.
For details on the robot interrupt application command, refer to 2.1 RINT: Robot interrupt (I-condition) (FN29), and for details on the shift application command, refer to 2.3 SHIFTR: Shift2 (FN52).

Please refer to the **"Application Manual / Material Handling"** for detail operations.

■ Example of operation

1) Teaching	Set the "Search range," and perform the teaching for the searching motion.
↓	
2) Search basis write	Obtain the position to be used the basis in the "Search basis write" mode.
↓	
3) Search execution	The searching motion is executed with normal playback. The shift amounts obtained are stored in the shift register.
↓	
4) Shift execution	Playback is performed in accordance with what is stored in the shift register while the prescribed steps are shifted.



- The teaching of the searching motion must be recorded without fail in the following sequence: **Search start (SEA: FN59) _ Robot interrupt (RINT: FN29) _ Search end (SEA: FN59)**. The search start command declares the processes performed when a recorded robot interrupt has been subsequently implemented to be “search processes.”
- The number of robot interrupts (RINT: FN29) permitted between the search start (SEA: FN59) and search end (SEA: FN59) is limited to one only. If two or more robot interrupts are recorded, the search function will not work correctly.
- The search section (range within which the interrupt processing is valid) is between MOVE(N+1), which is the movement command immediately before the robot interrupt command RINT step, and MOVE(N+2) which is the movement command immediately after this step, and it extends beyond MOVE(N+2) along the straight line which links these two points up to the position specified by “Search range.”
- Movement command MOVE(N+2) is recorded at a low speed using linear interpolation ON. If the speed at which a searching motion is performed is too high, the current position data obtained when the interrupt was detected will vary, resulting in a poor accuracy.
- When an interrupt is detected during a searching motion, it will be interpreted that the robot has reached MOVE(N+2) at that point, and after search end (SEA: FN59) has been executed, the next step, namely, movement command MOVE(N+3) will be executed.

Upon completion of the teaching, perform “search basis write.” “Search basis write” refers to the task that remembers the “basis position.” The remembered basis position serves as the basis for future searching motions, and the difference between this position and the position obtained by a normal searching motion is used as the shift amount.
” Search basis write” selection menu is in Service/22 Handling application/1 Handling teach/playback condition.

■ **Parameter**

Parameter No. 1	Start2/ Start/ End	This is used to specify the start and end of the search. 2 (start2): The machine coordinates at which the shift amounts are not canceled when the position was detected by FN29 (robot interrupt) are obtained. 1 (start): The machine coordinates at which the shift amounts were canceled when the position was detected by FN29 (robot interrupt) are obtained. 0 (end): This is used to end the search.
Parameter No. 2	Coordinate	This is used to specify the coordinate system that serves as the basis for the shift amounts to be set in the shift register. 0 (machine coordinates): The shift amounts are set in the shift register as the shift amounts in the machine coordinate system. 1 (tool): The shift amounts are set in the shift register as the shift amounts in the tool coordinate system.
Parameter No. 3	Register No.	This is used to specify the number of the shift register in which the shift amounts are to be set. (1 to 9)

■ **Example of screen display**

```
SEA[2, 0, 9]              FN59: Search
```

See
RINT; Robot Interrupt (I-condition) (FN29)
SHIFTR; Shift2 (FN52)

Function commands (FN codes)

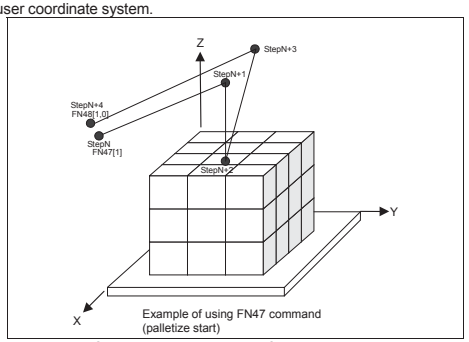
Command name	DSPALLET
FN code	65
Title name	Direction select palletize
Outline	This limits the shift direction, and it is started by the palletizing work.

■ **General description**

The direction in which the position is to be shifted during the palletizing tasks can be specified using this function command.

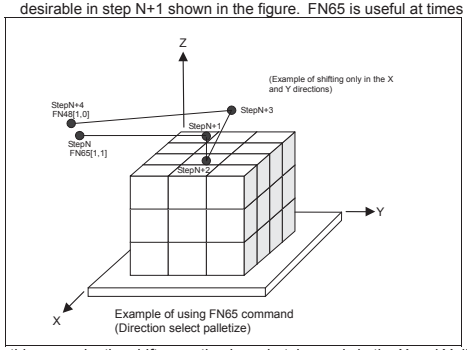
■ **Example of operation**

When the FN47 "Palletize start" command is used, the shift is normally implemented in all the X, Y and Z axis directions for the user coordinate system.



Example of using FN47 command
(palletize start)

However, if there is an obstruction, for instance, above the Z axis direction, shifting in the Z axis direction would not be desirable in step N+1 shown in the figure. FN65 is useful at times like this.



Example of using FN65 command
(Direction select palletize)

In this example, the shift operation is undertaken only in the X and Y directions of the user coordinate system.

■ **Parameter**

	Data	Description, setting range
Parameter No. 1	Palletize No.	This specifies the palletizing number to be executed. (1 to 200)
Parameter No. 2	Shift direction No.	This is used to specify the shift direction. The shift direction numbers are as follows. Here all the X, Y and Z values denote the values used in the user coordinate system. (0 to 6) 0: All directions (Same shift as with FN47) 1: XY plane (Only the X and Y values for the shift amounts are used) 2: YZ plane (Only the Y and Z values for the shift amounts are used.) 3: ZX plane (Only the Z and X values for the shift amounts are used) 4: X direction (Only the X value for the shift amounts is used) 5: Y direction (Only the Y value for the shift amounts is used) 6: Z direction (Only the Z value for the shift amounts is used)

■ **Example of screen display**

```
DSPALLET[1, 1]            FN65: Direction select palletize
```

See
PALLET2: Palletize start (FN47)
PALLET2_END: Palletize end (FN48)
PALLET2_RESET: Palletize reset (FN49)

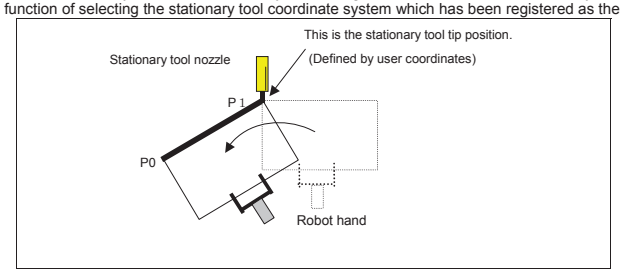
Command name	STOOL
FN code	67
Title name	Stationary tool number selection
General description	This command is used to select the coordinate system of the stationary tool number from among the user coordinate systems.

■ General description

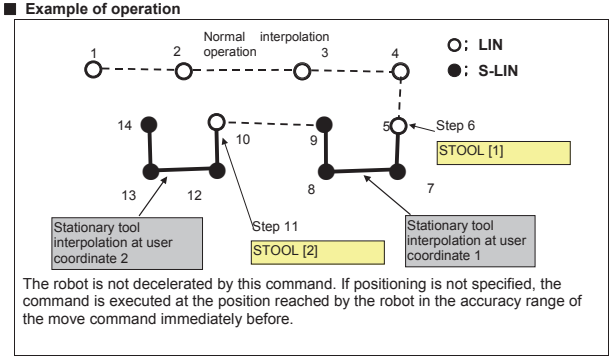
A stationary tool is not one of the robot's tools but one which is settled externally. If "stationary tool interpolation" is specified as the interpolation type for the move command, the step concerned will conduct the interpolation playback using the coordinates of the stationary tool.

In the case of linear interpolation:
"LIN" for normal interpolation, "S-LIN" for stationary tool interpolation

The coordinate system of the stationary tool is registered as the user coordinate system. The STOOL command has the function of selecting the stationary tool coordinate system which has been registered as the user coordinate system.



■ Example of operation



The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

Once the number of the stationary tool is selected, the interpolation operation is performed by that stationary tool until another stationary tool number is selected.

The normal interpolation tool number remains unaffected by this command.

If the move command for the stationary tool interpolation has been played back without the STOOL command having been recorded, Alarm No.2971 “Stationary tool is not selected” occurs.

■ Parameter

Parameter No. 1	User coordinate No.	This specifies the number of the user coordinate system which is to be used as the stationary tool coordinate system. (1-100)

■ Example of screen display

```
STOOL [1]      FN67;   Stationary tool number selection
```

See
None

Command name	LETR
FN code	68
Title name	Set shift value
Outline	The shift amount data is set in the specified shift register

■ General description

When this function command is used, the shift amount data can be set in the specified shift register.

Even while the specified shift register is being used by the "SHIFTR: Shift2" (FN52) or other command, the new values specified by LETR will be set in the shift register without adversely affecting the operation of the other command being executed. The changed shift values become valid when the next "SHIFTR: Shift2" (FN52) command is played back.

■ Example of operation

Shift register values when LETR[R1,100,100,100,10,11,12] has been executed

① When the command has been executed while shift register values were already set

	Before LETR is executed	After LETR is executed
Request flag	0	0
Setting flag	1	1
X	110	100
Y	120	100
Z	130	100
θX	5	10
θY	6	11
θZ	7	12

② When the command has been executed with the shift registers in the initial status

	Before LETR is executed	After LETR is executed
Request flag	0	0
Setting flag	0	1
X	0	100
Y	0	100
Z	0	100
θX	0	10
θY	0	11
θZ	0	12

After this command has been executed, the request flag of the shift register is always set to "0," and the setting flag is always set to "1." The values specified by the parameters are stored in X through θz.

■ Parameter

Parameter No. 1	Shift register number	This is used to specify the number of the shift register in which the shift amount data specified by the parameters No.2 to 7 is to be assigned. (1 to 9)
Parameter No. 2	Shift value X	This is used to specify the value of the shift amount in the X direction. (–3000.0 to 3000.0 mm)
Parameter No. 3	Shift value Y	This is used to specify the value of the shift amount in the Y direction. (–3000.0 to 3000.0 mm)
Parameter No.4	Shift value Z	This is used to specify the value of the shift amount in the Z direction. (–3000.0 to 3000.0 mm)
Parameter No. 5	Angle shift X	This is used to specify the amount of rotation around the X axis. (–360.0 to 360.0 deg)
Parameter No. 6	Angle shift Y	This is used to specify the amount of rotation around the X axis. (–360.0 to 360.0 deg)
Parameter No. 7	Angle shift Z	This is used to specify the amount of rotation around the X axis. (–360.0 to 360.0 deg)

■ Example of screen display

```
LETR[R1,100.0,-100.0,50.0,10.0,0.0,0.0]  FN68:Set shift value
```

See
SREQ: Shift amount request (FN51)
SHIFTR: Shift2 (FN52)
ADDR: Add shift value (FN69)

Function commands (FN codes)

Command name	ADDR
FN code	69
Title name	Add shift value
Outline	The specified values in the specified shift register are added up

■ General description

When this function command is used, the specified values in the specified shift register can be added up.
Even while the specified shift register is being used by the "SHIFTR: Shift2" (FN52) or other command, the new values specified by ADDR will be set in the shift register without adversely affecting the operation of the other command being executed. The changed shift values become valid when the next "SHIFTR: Shift2" (FN52) command is played back.

■ Example of operation

Shift register values when ADDR[R1,100,100,100,10,11,12] has been executed

① When the command has been executed while shift register values were already set

	Before ADDR is executed	After ADDR is executed
Request flag	0	0
Setting flag	1	1
X	110	210
Y	120	220
Z	130	230
θX	5	15
θY	6	17
θZ	7	19

② When the command has been executed with the shift registers in the initial status

	Before ADDR is executed	After ADDR is executed
Request flag	0	0
Setting flag	0	1
X	0	100
Y	0	100
Z	0	100
θX	0	10
θY	0	11
θZ	0	12

After this command has been executed, the request flag of the shift register is always set to "0," and the setting flag is always set to "1." The values specified by the parameters are added to the original values and stored in X through θz.

■ Parameter

Parameter No. 1	Shift register number	This is used to specify the number of the shift register in which the shift amount data specified by the parameters No.2 to 7 is to be added.
Parameter No. 2	Shift value X	This is used to specify the value of the shift amount in the X direction. (–3000.0 to 3000.0 mm)
Parameter No. 3	Shift value Y	This is used to specify the value of the shift amount in the Y direction. (–3000.0 to 3000.0 mm)
Parameter No.4	Shift value Z	This is used to specify the value of the shift amount in the Z direction. (–3000.0 to 3000.0 mm)
Parameter No. 5	Angle shift X	This is used to specify the amount of rotation around the X axis. (–360.0 to 360.0 deg)
Parameter No. 6	Angle shift Y	This is used to specify the amount of rotation around the X axis. (–360.0 to 360.0 deg)
Parameter No. 7	Angle shift Z	This is used to specify the amount of rotation around the X axis. (–360.0 to 360.0 deg)

■ Example of screen display

```
ADDR[R1,100.0,-100.0,50.0,10.0,0.0,0.0,0.0]    FN69: Add shift value
```

See
SREQ: Shift amount request (FN51)
SHIFTR: Shift2 (FN52)
LETR: Set shift value (FN68)

Application Command (FN Code)

Command name	LETX
FN code	71
Title name	Assign X component of pose
Outline	Used to assign a value to the X component of pose.

■ Outline

Use this command in robot languages.
Executing this command makes it possible to set shift amount to a specified pose variable. The commands LETX (FN71), LETY (FN72), and LETZ (FN73) are only enabled for already-recorded positional data. Use these commands to record only a single point in the pose and move the robot in parallel with the point.

■ Parameters

Parameter 1	Pose variable	Used to make setting of pose variable. (Setting range: 1 to 9999)
Parameter 2	Assigning value	Used to make setting of value of the X component to be assigned to pose variable. (Setting range: –3000.0 to 3000.0 mm)

■ Example of screen display

```
LETX[P1,5]      FN71: Assign X component of pose
```

Related commands
LETY: Assign Y component of pose (FN72)
LETZ: Assign Z component of pose (FN73)

Application Command (FN Code)

Command name	LETY
FN code	72
Title name	Assign Y component of pose
Outline	Used to assign a value to the Y component of pose.

■ Outline

Use this command in robot languages.
Executing this command makes it possible to set shift amount to a specified pose variable. The commands LETX (FN71), LETY (FN72), and LETZ (FN73) are only enabled for already-recorded positional data. Use these commands to record only a single point in the pose and move the robot in parallel with the point.

■ Parameters

Parameter 1	Pose variable	Used to make setting of pose variable. (Setting range: 1 to 9999)
Parameter 2	Assigning value	Used to make setting of value of the Y component to be assigned to pose variable. (Setting range: −3000.0 to 3000.0 mm)

■ Example of screen display

```
LETY[P1,10]     FN72: Assign Y component of pose
```

Related commands
LETY: Assign X component of pose (FN71)
LETZ: Assign Z component of pose (FN73)

Application Command (FN Code)

Command name	LETZ
FN code	73
Title name	Assign Z component of pose
Outline	Used to assign a value to the Z component of pose.

■ Outline

Use this command in robot languages.
Executing this command makes it possible to set shift amount to a specified pose variable. The commands LETX (FN71), LETY (FN72), and LETZ (FN73) are only enabled for already-recorded positional data. Use these commands to record only a single point in the pose and move the robot in parallel with the point.

■ Parameters

Parameter 1	Pose variable	Used to make setting of pose variable. (Setting range: 1 to 9999)
Parameter 2	Assigning value	Used to make setting of value of the Z component to be assigned to pose variable. (Setting range: −3000.0 to 3000.0 mm)

■ Example of screen display

```
LETX[P1,10]     FN73: Assign Z component of pose
```

Related commands
LETX: Assign X component of pose (FN71)
LETY: Assign Y component of pose (FN72)

Application Command (FN Code)

Command name	POSESAVE
FN code	74
Title name	Pose file save
Outline	Pose variables are stored to the pose file.

■ Outline

Pose variables are not saved when the main power is down.
This command can save pose variables to the pose file.

Before this command, pose file must be selected by FN98 USE function.
If pose file is not selected when executing FN74, information “I2151 : The program or the file does not exist.” is detected and robot stops immediately.

■ Parameters

None

■ Example of screen display

POSESAVE FN74: Pose file save

Function commands (FN codes)

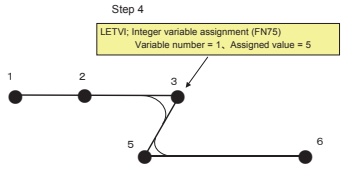
Command name	LETVI
FN code	75
Title name	Integer variable assignment
General description	This command is used to assign a value to the specified integer variable register.

■ General description

When this function command is executed, a value is assigned to the specified (global) integer variable. The (global) integer variables are registers used to count the number of passes by the conditional functions, which are initiated after the specified number of passes, of the flow control system such as the conditional program call after the specified number of passes command. They can be referenced from all the units.
This command is used for initial resetting and for setting the initial values.

■ Example of operation

In step 4, record LETVI: integer variable assignment (FN75), "1" as the variable number, and "5" as the assigned value. When this is played back, "5" is set in the first integer variable. The variables which have been set can be checked on the integer variable register screen of the monitor screen.



The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

■ Parameter

Parameter No. 1	Integer variable number	This specifies the integer variable number to which the value will be assigned. (1-200)
Parameter No. 2	Assigned value	This specifies the value which is to be set in the integer variable. (-2147483647～ +2147483647)

■ Example of screen display

LETVI [V1%,5] FN75; Integer variable assignment

See
LETVF: Set real var. (FN76)
LETVS: Set strings variable (FN77)

Command name	LETVF
FN code	76
Title name	Real number variable assignment
General description	This command is used to assign a value to the specified real number variable register.

■ General description

When this function command is executed, a value is assigned to the specified (global) real number variable. The (global) real number variables are registers used to store values by the robot axis coordinate acquisition commands and timer variable read commands. They can be referenced from all the units.
This command is used for initial resetting and for setting the initial values.

■ Example of operation

In step 4, record LETVF: real number variable assignment (FN76), "1" as the variable number, and "5.5" as the assigned value. When this is played back, "5.5" is set in the first real number variable. The variables which have been set can be checked on the real number variable register screen of the monitor screen.

The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

■ Parameter

Parameter No. 1	Real number variable number	This specifies the number of the real number variable to which the value will be assigned. (1－200)
Parameter No. 2	Assigned value	This specifies the value which is to be set in the real number variable. (–1.0E38～+1.0E38)

■ Example of screen display

```
LETVF [V1!,5.5]        FN76; Real number variable assignment
```

See
LETVI: Set real var. (FN75)
LETVS: Set strings variable (FN77)

Command name	LETVS
FN code	77
Title name	Character string variable assignment
General description	This command is used to assign a character string to the specified character string variable register.

■ General description

When this function command is executed, a character string is assigned to the specified (global) character string variable. The (global) character string variables are registers used to store the characters displayed on the screen using user macros, and the character string data transmitted from the peripheral devices. They can be referenced from all the units.
This command is used for initial resetting and for setting the initial values.

■ Example of operation

In step 4, record LETVS: character string variable assignment (FN77), "1" as the variable number, and "1A" as the value to be assigned. When this is played back, "1A" is set in the first character string variable. The variables which have been set can be checked on the character string variable register screen of the monitor screen.

The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

■ Parameter

Parameter No. 1	Character string variable number	This specifies the number of the character string variable to which the value will be assigned. (1-50)
Parameter No. 2	Assigned value	This specifies the value which is to be assigned to the character string variable using characters with a length of up to 199 characters.

■ Example of screen display

```
LETVS [V1$,"1A"]    FN77; Character string variable assignment
```

See
LETVI: Set integer variable (FN75)
LETVF: Set real var. (FN76)

Function commands (FN codes)

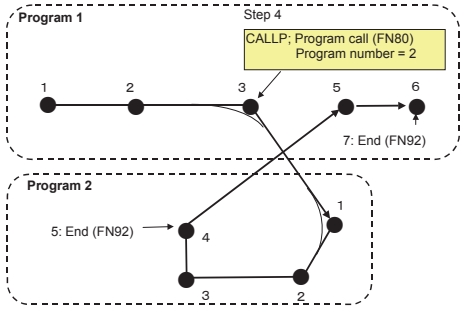
Command name	CALLP
FN code	80
Title name	Program call
General description	This command is used to call the specified program.

■ General description

When this function command is executed, the specified program is called.
Bear in mind that if a function command has been recorded in the first step in the call destination program, the function command at the jump destination will be executed as soon as the call command has been executed.
When the playback of the program at the call destination is completed (in the status established by executing the END command), the robot returns to the step following the step with the call command of the call source program.

■ Example of operation

In step 4, record CALLP: program call (FN80) and "2" as the program number. When this is played back, the robot skips steps 5 and 6 upon arriving at step 4 and jumps to the first step in program 2. When the playback of program 2 is completed (in the status established by executing the END command), the robot returns to step 5 following the step with the call command of call source program 1.



The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

The program call can be executed again at the call destination (during program 2 in the above figure.) Up to 8 layers of calls can be executed. If calls exceeding 8 layers are executed, the "A2138 Wrong call command setting" alarm is detected during playback, and the robot stops.

■ Parameter

Parameter No. 1	Program No.	This specifies the number of the program serving as the call destination. (1-9999)

■ Example of screen display

CALLP [2] FN80; Program call

See
CALLPI: Conditional program call (FN81)
CALLPN: Conditional program call after specified number of passes (FN82)

Function commands (FN codes)

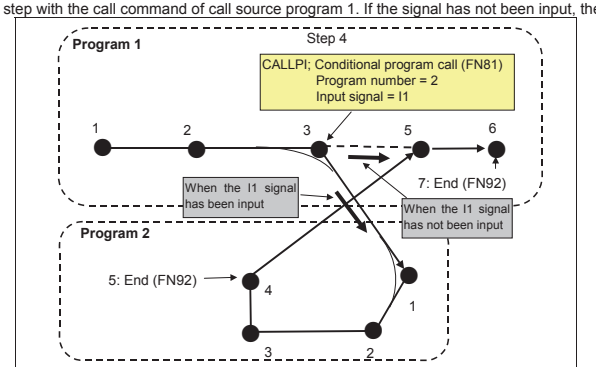
Command name	CALLPI
FN code	81
Title name	Conditional program call
General description	Using an input signal, this command is used to call the specified program.

■ General description

When this function command is executed, the specified program is called. When the specified input signal has been input, the step is called; when it has not been input, the step is not called and the robot passes the command by.
Bear in mind that if a function command has been recorded in the first step in the call destination program, the function command at the jump destination will be executed as soon as the call command has been executed.
When the playback of the program at the call destination is completed (in the status established by executing the END command), the robot returns to the step following the step with the call command of the call source program.

■ Example of operation

In step 4, record CALLPI: conditional program call (FN81), "2" as the program number, and I1 as the input signal.
When this is played back, the robot arrives at step 4, and if input signal I1 has been input, it jumps to the first step in program 2, and when the playback of program 2 is completed (the END command is executed), the robot returns to step 5 following the step with the call command of call source program 1. If the signal has not been input, the robot does not jump to program 2.



The robot is not decelerated by this command. If no positioning is specified, the input signal is inspected just before the command value reaches the accuracy range of the move command, and if it has been input, an inside arc is drawn. If the input signal has not been input, the command value heads toward the recorded point, and the input signal is inspected when the robot has reached the accuracy range.

The program call can be executed again at the call destination (during program 2 in the above figure.) Up to 8 layers of calls can be executed. If calls exceeding 8 layers are executed, the "A2138 Wrong call command setting" alarm is detected during playback, and the robot stops.

■ Parameter

Parameter No. 1	Program No.	This specifies the number of the program serving as the call destination. (1-9999)
Parameter No. 2	Input signal	This records the number of the input signal which is to serve as the condition for executing the call. When number 5101 or above is specified, multiple input signals can be specified. (1-2048、5101—5196)

■ Example of screen display

CALLP [2,I1] FN81; Conditional program call

See
CALLP: Program call (FN80)
CALLPN: Conditional program call after specified number of passes (FN82)

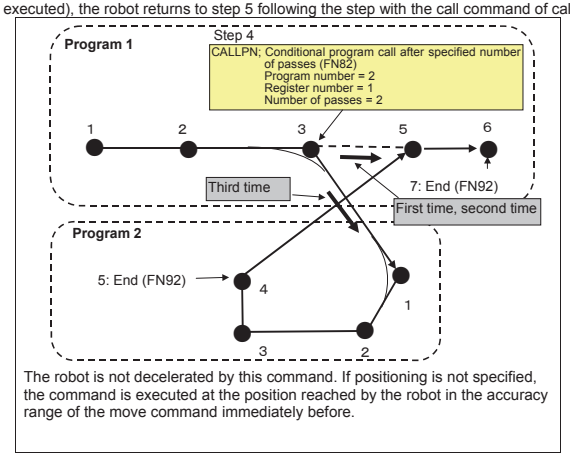
Command name	CALLPN
FN code	82
Title name	Conditional program call after specified number of passes
General description	Using a pass count (number of passes), this command is used to call the specified program.

■ General description

When this function command is executed, the specified program is called. The robot passes for the specified number of passes, and on the next time (specified number of passes +1) the call command is executed. (For instance, if "2" is specified as the number of passes, the robot passes twice, and on the third time the call command is executed.)
Bear in mind that if a function command has been recorded in the first step in the call destination program, the function command at the jump destination will be executed as soon as the call command has been executed.
When the playback of the program at the call destination is completed (in the status established by executing the END command), the robot returns to the step following the step with the call command of the call source program.

■ Example of operation

In step 4, record CALLPN: conditional program call after specified number of passes (FN82), "2" as the program number, "1" as the register number, and "2" as the number of passes.
When this is played back, the robot passes by for the first and second times, and then advances to steps 5; however, on the third time, it jumps to the first step in program 2. When the playback of program 2 is completed (the END command is executed), the robot returns to step 5 following the step with the call command of call source program 1.



The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

The program call can be executed again at the call destination (during program 2 in the above figure.) Up to 8 layers of calls can be executed. If calls exceeding 8 layers are executed, the "A2138 Wrong call command setting" alarm is detected during playback, and the robot stops.

A global integer variable common to all units is used for the number of passes.
The current number of passes can be referenced using monitor/integer variables.

■ Parameter

Parameter No. 1	Program No.	This specifies the number of the program serving as the call destination. (1-9999)
Parameter No. 2	Register number	A "register" refers to the memory used for counting. Since an integer variable (1 to 200) is used, this parameter specifies its number. (1-200)
Parameter No. 3	Number of passes	This records the number of passes which is to serve as the condition for executing the call. The robot passes for the specified number of passes, and on the next time (specified number of passes +1) the call command is executed. (0-10000)

■ Example of screen display

```
CALLPN [2, V1%, 2]    FN82; Conditional program call after
                            specified number of passes
```

See
CALLP: Program call (FN80)
CALLPI: Conditional program call (FN81)

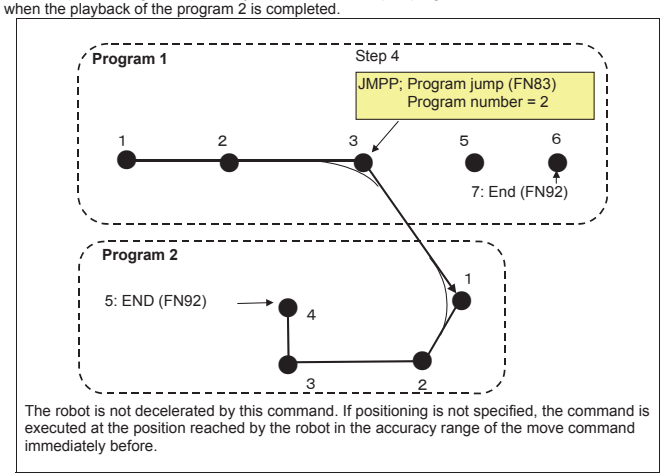
Command name	JMPP
FN code	83
Title name	Program jump
General description	This command is used to jump to the start of the specified program.

■ General description

When this function command is executed, the robot jumps to the specified program.
Bear in mind that if a function command has been recorded in the first step in the jump destination program, the function command at the jump destination will be executed as soon as the jump command has been executed.
The robot does not return to the source program even when the playback of the program at the jump destination is completed.

■ Example of operation

In step 4, record JMPP: program jump (FN83), and "2" as the program number. When this is played back, the robot skips steps 5 and 6 upon arriving at step 4 and jumps to the first step in program 2. The robot does not return to the source program even when the playback of the program 2 is completed.



The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

■ Parameter

Parameter No. 1	Program No.	This specifies the number of the program which is to serve as the jump destination. (1-9999)

■ Example of screen display

```
JMPP [2]              FN83; Program jump
```

See
JMPPI: Conditional program jump (FN84)
JMPPN: Conditional program jump after specified number of passes (FN85)

Function commands (FN codes)

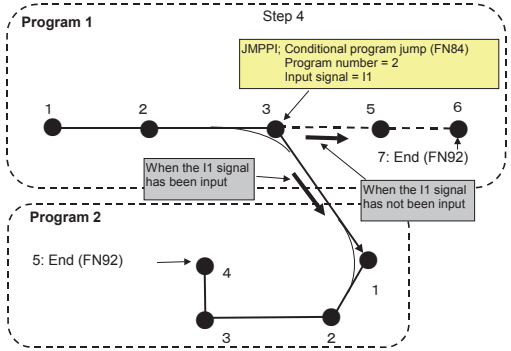
Command name	JMPPI
FN code	84
Title name	Conditional program jump
General description	Using an input signal, this is used to jump to the start of the specified program.

■ General description

When this function command is executed, the specified program is called. When the specified input signal has been input, the robot jumps; when it has not been input, it does not jump and the command is passed by.
Bear in mind that if a function command has been recorded in the first step in the jump destination program, the function command at the jump destination will be executed as soon as the jump command has been executed.
The robot does not return to the jump source program even when the playback of the program at the jump destination is completed (the END command is executed).

■ Example of operation

In step 4, record JMPPI: conditional program jump (FN84), "2" as the program number, and "1" as the input signal. If, when this is played back, input signal I1 has been input, the robot jumps to the first step in program 2; and if it has not been input, it advances to steps 5 and 6. The robot does not return to the jump source program even when the playback of program 2 is completed (the END command is executed).



The robot is not decelerated by this command. If no positioning is specified, the input signal is inspected just before the command value reaches the accuracy range of the move command, and if it has been input, an inside arc is drawn. If the input signal has not been input, the command value heads toward the recorded point, and the input signal is inspected when the robot has reached the accuracy range.

Unlike the program call command, the robot can jump any number of times.

■ Parameter

Parameter No. 1	Program No.	This specifies the number of the program which is to serve as the jump destination. (1-9999)
Parameter No. 2	Input signal	This records the number of the input signal which is to serve as the condition for executing the jump. When number 5101 or above is specified, multiple input signals can be specified. (1-2048、5101—5196)

■ Example of screen display

```
JMPPI [2,I1]          FN84; Conditional program jump
```

See
JMPP: Program jump (FN83)
JMPPN: Conditional program jump after specified number of passes (FN85)

Function commands (FN codes)

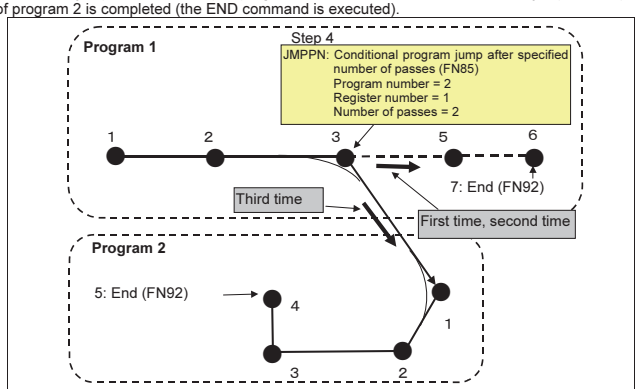
Command name	JMPPN
FN code	85
Title name	Conditional program jump after specified number of passes
General description	Using a pass count (number of passes), this command is used to jump to the start of the specified program.

■ General description

When this function command is executed, the robot jumps to the specified program. The robot passes for the specified number of passes, and on the next time (specified number of passes +1) the jump command is executed. (For instance, if "2" is specified as the number of passes, the robot passes twice, and on the third time the jump command is executed.)
Bear in mind that if a function command has been recorded in the first step in the jump destination program, the function command at the jump destination will be executed as soon as the jump command has been executed.
The robot does not return to the jump source program even when the playback of the program at the jump destination is completed (the END command is executed).

■ Example of operation

In step 4, record JMPPN: conditional program jump after specified number of passes (FN85), "2" as the program number, "1" as the register number, and "2" as the number of passes.
When this is played back, the robot passes by for the first and second times, and then advances to steps 5; however, on the third time, it jumps to the first step in program 2. The robot does not return to the jump source program even when the playback of program 2 is completed (the END command is executed).



The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

A global integer variable common to all units is used for the number of passes.
The current number of passes can be referenced using monitor/integer variables.

■ Parameter

Parameter No. 1	Program No.	This specifies the number of the program which is to serve as the jump destination. (1-9999)
Parameter No. 2	Register number	A "register" refers to the memory used for counting. Since an integer variable (1 to 200) is used, this parameter specifies its number. (1-200)
Parameter No. 3	Number of passes	This records the number of passes which is to serve as the condition for executing the jump. The robot passes for the specified number of passes, and on the next time (specified number of passes +1) the jump command is executed. (0-10000)

■ Example of screen display

```
JMPPN [2, V1%, 2] FN85; Conditional program jump after
                        specified number of passes
```

See
JMPP: Program jump (FN83)
JMPPI: Conditional program jump (FN84)

Function commands (FN codes)

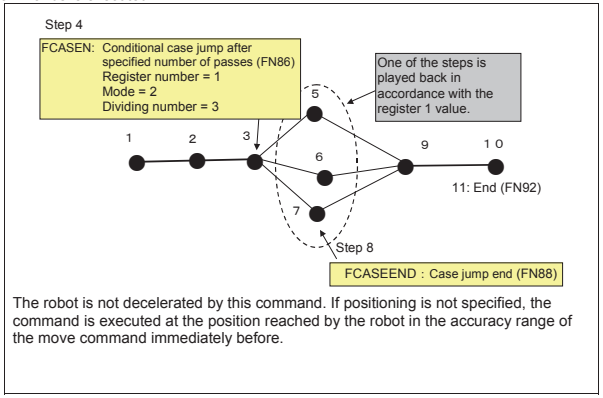
Command name	FCASEN
FN code	86
Title name	Conditional case jump after specified number of passes
General description	Using a register, this command is used to select one of a multiple number of steps and execute it.

■ General description

When this function command is executed, one of a multiple number of steps is selected and executed. It makes no difference whether the selected step is a move command or function command. As the condition under which the step selection is executed, Register is divided by a "dividing number" and, based on its "quotient" or "remainder", just the one step to be executed is decided on.
Bear in mind that if the step selected is a function command, the function command at the jump destination will be executed as soon as the conditional case jump after the specified number of passes command has been executed.

■ Example of operation

The FCASEN: conditional case jump after specified number of passes (FN86) command must always be paired with the FCASEEND: case jump end (FN88) command for use.
In the example presented below, only one step among steps 5, 6 and 7 enclosed between the FCASEN and FCASEEND commands is executed.



The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

Operation proceeds as follows in numerical sequence starting from the step following the step in which FCASEN is recorded: step executed when the quotient (or remainder) is 0 → step executed when the quotient (or remainder) is 1 → step executed when the quotient (or remainder) is 2 and so on.

Step 4: FCASEN: Conditional case jump after specified number of passes (FN86)
Step 5: (Selected when quotient or remainder is 0)
Step 6: (Selected when quotient or remainder is 1)
Step 7: (Selected when quotient or remainder is 2)
Step 8: FCASEEND: Case jump end (FN88)

Mode	Explanation
"Quotient" assignment	The "quotient" of the [pass count divided by the dividing number] serves as the condition of the command. In the example of the table presented below, step 5 is executed when the Register is 0, 1 or 2, and step 6 is executed when it is 3, 4 or 5.
"Remainder" assignment	The "remainder" of [pass count divided by the dividing number] serves as the condition of the command. In the table below, Register is add, the steps are executed in numerical order as follows: 5 → 6 → 7 → 5 → 6 → 7.

"Quotient" and "remainder" when 3 is the dividing number

Register	0	1	2	3	4	5	6	7	8	9	10	11
Quotient	0	0	0	1	1	1	2	2	2	3	3	3
Remainder	0	1	2	0	1	2	0	1	2	0	1	2

Any number of steps between the FCASEN and FCASEEND commands can be recorded provided that this number does not exceed the maximum number of 999 steps allowed per program. Either move commands or function commands can be the steps enclosed. If a step to be executed has not been recorded, an alarm is detected, and the robot stops.

Global integer variable is used for Register.
The current number of Register can be referenced using monitor/integer variables.

■ Parameter

Parameter No. 1	Register number	Global integer variable (1 to 200) is used, this parameter specifies its number. (1-200)
Parameter No. 2	Mode	1: The quotient of the [register value divided by the dividing number] is used as the condition for the command. 2: The remainder of the [register value divided by the dividing number] is used as the condition for the command. (1-2)
Parameter No. 3	Dividing number	This specifies the number that will divide the register value. (1-127)

■ Example of screen display

```
FCASEN [V1%, 2, 3]        FN86; Conditional case jump after
                                specified number of passes
```

See
FCASEEND: Case jump end (FN88)

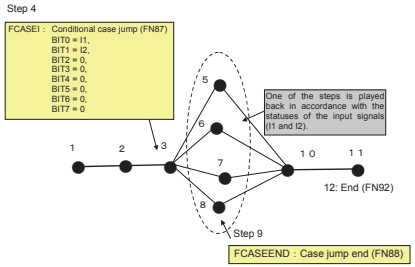
Command name	FCASEI
FN code	87
Title name	Conditional case jump
General description	Using an input signal, this command is used to select one of a multiple number of steps and executes it.

■ General description

When this function command is executed, one of a multiple number of steps is selected and executed. It makes no difference whether the selected step is a move command or function command. The "ON/OFF statuses of up to 8 input signals" are used as the condition under which the step selection is executed. The input signals are converted into an 8-bit binary value and, based on this, just the one step to be executed is decided on. Bear in mind that if the step selected is a function command, the function command at the jump destination will be executed as soon as the conditional case jump command has been executed.

■ Example of operation

The FCASEI: conditional case jump (FN87) command must always be paired with the FCASEEND: case jump end (FN88) command for use. In the example presented below, only one step among the four steps (steps 5, 6, 7 and 8) enclosed between the FCASEI and FCASEEND commands is executed.



The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

Operation proceeds as follows in numerical sequence starting from the step following the step in which FCASEI is recorded: step executed when the input signal is 0 → step executed when the input signal is 1 → step executed when the input signal is 2 and so on.

Step 4: FCASEI: Conditional case jump (FN87)
Step 5: (Selected when the input signal is 0)
Step 6: (Selected when the input signal is 1)
Step 7: (Selected when the input signal is 2)
Step 8: (Selected when the input signal is 3)
Step 9: FCASEEND: Case jump end (FN88)

These steps correspond in sequence to BIT 0, BIT 1, BIT 2 and so on from the first parameter of FCASEI. If, for instance, in the example presented in the previous figure, input signals I1 and I2 are both set to ON:

Bit 7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	—	—	I2	I1
0	0	0	0	0	0	1	1

= decimal number "3"

Therefore, step 8 will be selected.

Any number of steps between the FCASEI and FCASEEND commands can be recorded provided that this number does not exceed the maximum number of 999 steps allowed per program. Either move commands or function commands can be the steps enclosed. If a step to be executed has not been recorded, an alarm is detected, and the robot stops.

■ Parameter

Parameter No. 1	Input signal BIT 0	The numbers of the input signals corresponding to bits 0 through 7 are specified in sequence starting from the first parameter. If an input signal is not going to be used, 0 is specified. When number 5101 or above is specified, multiple input signals can be specified. (1–2048, 5101–5196)
Parameter No. 2	Input signal BIT 1	
Parameter No. 3	Input signal BIT 2	
Parameter No. 4	Input signal BIT 3	
Parameter No. 5	Input signal BIT 4	
Parameter No. 6	Input signal BIT 5	
Parameter No. 7	Input signal BIT 6	
Parameter No. 8	Input signal BIT 7	

Pack the input signals in sequence starting from the lower bits. If the registration of any input signal is missing, an alarm results during playback, and the robot stops.

■ Example of screen display

```
FCASEI [I1,I2,I3,I0,I0,I0,I0,I0]      FN87; Conditional case jump
```

See
FCASEEND: Case jump end (FN88)

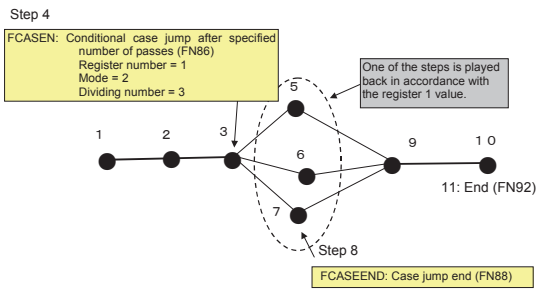
Command name	FCASEEND
FN code	88
Title name	Case jump end
General description	This command is used to end the case jump.

■ General description

This function command is paired with the FCASEN: conditional case jump after specified number of passes (FN86) command or FCASEI: conditional case jump (FN87) command for use. It signals the end of the case jump.

■ Example of operation

Refer to the FCASEN: conditional case jump after specified number of passes (FN86) or FCASEI: conditional case jump (FN87) commands.



The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

■ Parameter

None

■ Example of screen display

```
FCASEEND                    FN88; Case jump end
```

See
FCASEN: Conditional case jump after specified number of passes (FN86)
FCASEI: Conditional case jump (FN87)

Application Command (FN Code)

Command name	GOTO
FN code	90
Title name	Line jump
Outline	Used to jump to a specified line or label.

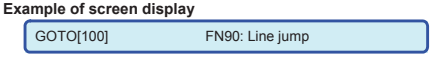
■ **Outline**

Executing this command transfers control to a line specified by line number or label without conditions.

■ **Parameters**

Parameter 1	Line No./ Label	Used to make setting of line number or label of jump destination. (Setting range: 1 to 9999)

■ **Example of screen display**



Related commands

∗LABEL: Label (FN601)

Application Command (FN Code)

Command name	GOSUB
FN code	91
Title name	Line call
Outline	Used to subroutine-call a specified line or label.

■ **Outline**

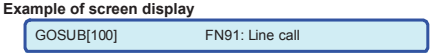
Use this command in robot languages.

Executing this command makes the program jump to a specified line. Executing any of the commands RETURN: Step return (FN22), RETI: Step return with conditions (FN25), and RETN: Step return with conditions for number of times (FN28) return to a line next to the call in which call was made.

■ **Parameters**

Parameter 1	Line No./ Label	Used to make setting of line number or label of jump destination. (Setting range: 1 to 9999)

■ **Example of screen display**



Related commands

RETURN: Step return (FN22)

RETI: Step return with conditions (FN25)

RETN: Step return with conditions for number of times (FN28)

Function commands (FN codes)

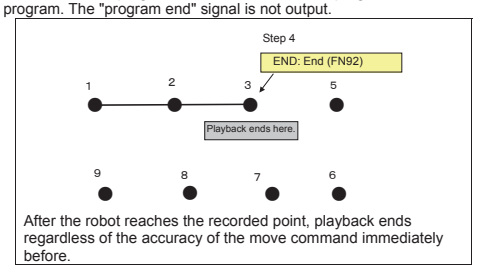
Command name	END
FN code	92
Title name	END
General description	This command is used to end program playback.

■ General description

When this function command is executed, the playback of the program is ended.
In the single cycle mode, operation stops immediately; in the continuous mode, operation returns to the start of the program and continues.
At least one END command is required in a program. This command does not signify the very end of the file so steps can still be recorded after this function. It is also permissible to record a multiple number of END commands in a program.

■ Example of operation

In step 4, record END: End (FN92).
When this is played back in the single cycle mode, the robot stops at step 4. The program end status is established, and the "program end" signal is output. If operation is now started again immediately, the robot returns to the first step.
However, if this program is the call destination program, the robot does not stop after END but returns to the call source program. The "program end" signal is not output.



After the robot reaches the recorded point, playback ends regardless of the accuracy of the move command immediately before.

■ Parameter

None

■ Example of screen display

END	FN92; End

"Program end" basic output signal

See
STOP: Stop (FN41)

Function commands (FN codes)

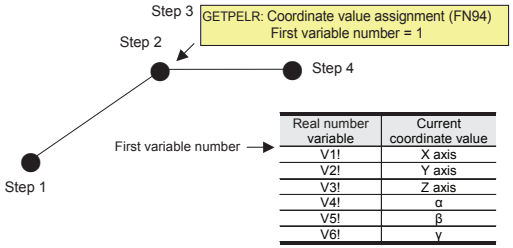
Command name	GETPELR
FN code	94
Title name	Real number variable assignment (coordinate values)
General description	This command is used to store the current coordinate values (Eulerian angle expressions) in the real number variables.

■ General description

When this function command is executed, the current axis coordinate values of the robot expressed as Eulerian angles are stored in the real number variables of the specified numbers.
(The GETP: real number variable assignment (coordinate values) (FN142) command is used when the wrist posture is to be expressed as the R (roll), P (pitch) and Y (yaw) angles.)

■ Example of operation

In the example of the figure presented below, when the robot arrives at the recorded point in step 2 and positioning is completed, the GETPELR command in step 3 is executed, and six real number variables are assigned with the coordinate values in sequence starting from the specified real number variable, as shown below.



Real number variable	Current coordinate value
V1!	X axis
V2!	Y axis
V3!	Z axis
V4!	α
V5!	β
V6!	γ

After the robot reaches the exact recorded point, the GETPELR command is executed regardless of the accuracy of the move command immediately before and regardless of whether positioning is specified.

The assigned values can be referenced by monitor/real number variables.

■ Parameter

Parameter No. 1	Real number variable number	This specifies the first number of the register (real number variable) in which a coordinate value is to be stored. Six variables are then used in succession. (1-195)

■ Example of screen display

GETPELR [V1!]	FN94; V! Set real var. (cur.pos)

See
GETP: Real number variable assignment (coordinate values) (FN142)
GETANGLE: Real number variable assignment (axis angles) (FN157)

Function commands (FN codes)

Command name	CHGGUN
FN code	95
Title name	Mechanism change
General description	Connect or disconnect mechanism2 (dedicated to mechanism 2 only)

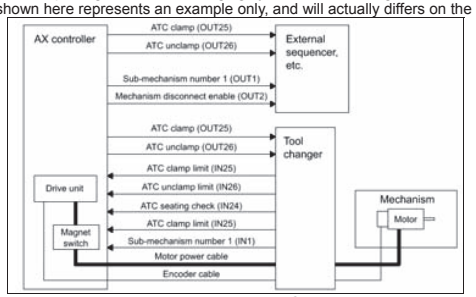
■ General description

There is an application that uses a tool changer to change the spot welding gun, hand grippers or other tools so that one robot can be employed for a multi purpose use. If the tool to be changed is a spot welding gun or hand gripper and it is air-driven, it can be changed by I/I sequences. On the other hand, if it is a servo gun or other servo-driven tool, it is not possible to disconnect it in order to implement the change while still keeping the power supplied to their motors. Therefore, changing servo guns and other such tools requires a function that makes it possible to electrically connect and disconnect the motor. This is called the mechanism change function.
This command can implement this utility. (**dedicated to mechanism 2 only**)

Connecting and disconnecting the tools mechanically must be implemented using I/O sequences.

■ Example of operation

An example is given for tool changing system where the signals have been allocated as shown in the figure below. The method shown here represents an example only, and will actually differs on the environment that depends on the customer's facility.

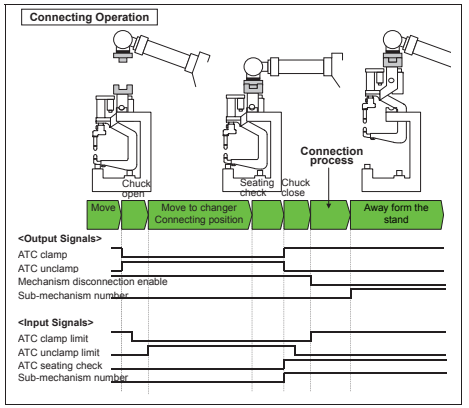


Connection with tool changer (sample)

Connecting operation

STEP N	500mm/s LIN A1 T31		
STEP N+1	RESET[025]	FN34; Output OFF	Unclamp
STEP N+2	SET[026]	FN32; Output ON	
STEP N+3	WAIT[126]	FN525; Wait input (positive)	Waiting for unclamp finish
STEP N+4	100mm/s LIN A1 T31		Move to the connect/disconnect position
STEP N+5	DELAY[0.5]	FN50; Delay	
STEP N+6	WAIT[124]	FN525; Wait input (positive)	Waiting for seating
STEP N+7	RESET[026]	FN34; Output OFF	Clamp
STEP N+8	SET[025]	FN32; Output ON	
STEP N+9	WAIT[125]	FN525; Wait input (positive)	Waiting for clamp finish
STEP N+10	CHGGUN[1]	FN95; Mechanism change	Electrical connection
STEP N+11	100mm/s LIN A1 T1		Move away

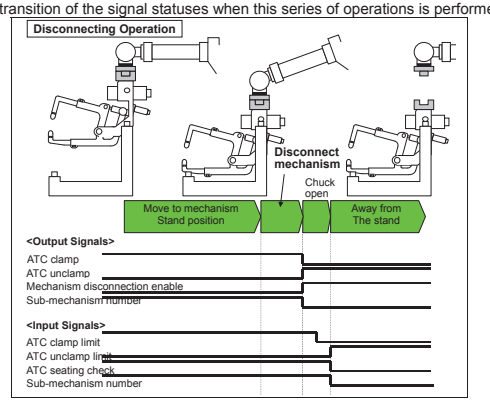
At step N, the mechanism is moved by the following linear movement to the position where the changer can be connected. At this position, the changer is set to the unclamp status by the way or precaution.
At step N+4, the mechanism is moved to the position where the changer is connected. At first its seating is checked and the changer is clamped. After it has been clamped, the mechanism change function is executed.
At step N+11, the mechanism is disengaged from the stand.
Shown below is the transition of signal statuses when this series of operations is performed.



Disconnecting Operation

STEP N	500mm/s LIN A1 T1		
STEP N+1	100mm/s LIN A1 T1		Move to the connect/disconnect position
STEP N+2	CHGGUN[0]	FN95; Mechanism change	Electrical disconnection
STEP N+3	RESET[025]	FN34; Output OFF	Unclamp
STEP N+4	SET[026]	FN32; Output ON	
STEP N+5	WAIT[126]	FN525; Wait input (positive)	Waiting for unclamp finish
STEP N+6	100mm/s LIN A1 T31		

At step N, the mechanism is moved by the following movement to the position where it can be brought to the stand. At step N+1, the mechanism is moved to the changer disconnect position. At this position, the mechanism function is executed, and the mechanism is electrically disconnected. Next, the changer is set to the unclamp status.
After the unclamp status has been checked, the mechanism is electrically disconnected at the step N+6. Shown below is the transition of the signal statuses when this series of operations is performed.



■ Parameter

Parameter No. 1	Connect / Disconnect	This is used to specify whether the mechanism is to be connected or disconnected 0; Connect 1; Disconnect

■ Example of screen display

```
CHGGUN[0]              FN95: Mechanism change
```

See
CHGMEC; Mechanism Change（mechanism can be designated）(FN301)

Application Command (FN Code)

Command name	USE
FN code	98
Title name	Pose file select
Outline	Used to select pose file.

■ Outline

Pose data are controlled as files and one file can record poses P1 to P999. For example, this command is useful in playing back the same program by changing only the data on robot positions according to the types of workpieces.

■ Parameters

Parameter 1	Pose file number	Used to make setting of pose file number. (Setting range: 1 to 9999)

■ Example of screen display

```
USEL[2]                FN098: Pose file select
```

Function commands (FN codes)

Command name	REM
FN code	99
Title name	Comment
General description	This command is used to provide comments inside programs.

■ General description

This command is used to provide comments inside programs.
Using the software keyboard, alphanumerics and symbols can be input.
A comment provided in step 0 is handled specially as the "program name," and it appears on the program directory by the R17 short-cut and on the status window at the top of the screen.

Software keyboard

Entire text of comment entered using 

■ Example of operation

The robot is not decelerated by this command.
More than one comment may be recorded in a program.

■ Parameter

Parameter No. 1	Comment	Comments consisting of up to 230 characters can be recorded.

■ Example of screen display

```
REM ["Test program"]        FN99; Comment
```

See
None

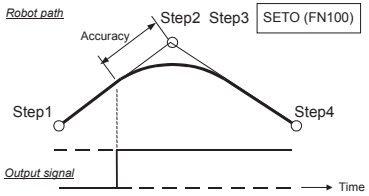
Function commands (FN codes)

Command name	SETO
FN code	100
Title name	Consecutive output signal ON/OFF
General description	This command is used to set any number of consecutive general-purpose output signals to ON or OFF altogether.

■ General description

When this function command is executed, any number of consecutive general-purpose output signals (O1 to O2048) are set to ON or OFF altogether. It makes no difference how many consecutive signals are involved.
However, the command cannot set any of the status signals (signals with pre-assigned applications such as the gun signals and starting signal) to ON or OFF. Which signals have been assigned as status signals can be identified on the monitor screen. Signals whose numbers are displayed in the bold italics are status signals so any of the other signals can be set to ON or OFF.

■ Example of operation



The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

■ Parameter

Parameter No. 1	Output signal number	This specifies the number of the first consecutive output signals which are to be turned ON or OFF. (1–2048)
Parameter No. 2	Output signal number	This specifies the number of the last consecutive output signals which are to be turned ON or OFF. (1–2048)
Parameter No. 3	ON/OFF	"1" specified for ON, and "0" for OFF. (0–1)

■ Example of screen display

```
SETO [O1, O512, 1]   FN100; Consecutive output signal ON/OFF
```

See
SETM: Output signal ON/OFF (FN105)

Function commands (FN codes)

Command name	PRINT
FN code	101
Title name	Strings output
Outline	The character string data is output to the specified serial port

■ General description

When this function command is used, character string data can be output to the specified communication (serial) port. A character string consisting only of single-byte alphanumerics (up to 199) can be output.
At the present time, only port 1 (RS232C) can be used.

■ Example of operation

When the robot reaches the step in which the PRINT (FN101) command is recorded, the character string is sent from the specified serial port.
The robot continues moving while the data is being sent.

■ Parameter

Parameter No. 1	Port number	This is used to specify the number of the port from which the character string is to be output. At the present time, only port 1 (RS232C) can be used. (1 to 1)
Parameter No. 2	Output character string	This is used to specify the character string to be output. (Up to 199 single-byte alphanumerics)

The character string is input using the soft keyboard.

■ Example of screen display

```
PRINT[#1, "TEST WORK"]                FN101: Strings output
```

T	E	S	T		W	O	R	K	CR	LF

Example of data sent (TEST WORK)

See
SREQ: Shift amount request (FN51)
RSCLR: RS232C Buffer clear (FN111)

Function commands (FN codes)

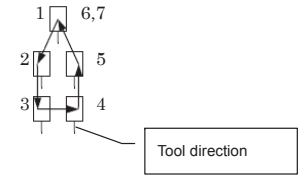
Command name	CALLPR
FN code	102
Title name	Relative program call
General description	This command is used to call a subprogram and makes the first step position and orientation the same as the current step in the base program and all point positions in the subprogram become relative to that step position.

■ General description

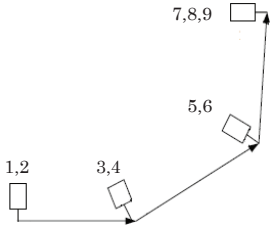
[Relative Program Call] calls a subprogram and makes the first step position and orientation the same as the current step in the base program and all point positions in the subprogram become relative to that step position. （Refer to the illustration below）
This is a very convenient function for spot welding, weld gun sticking, grinding, cutting and other frequently repeated simple task actions.

■ Example of operation

When you make the program to operate as follows

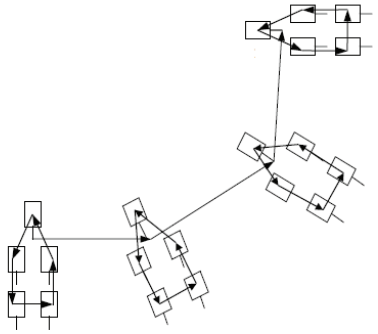


Program 1
(subprogram to be called)
1 MOVE
2 MOVE
3 MOVE
4 MOVE
5 MOVE
6 MOVE
7 END



Program 2
(Main(Base) program)
1 MOVE
2 CALLPR 1
3 MOVE
4 CALLPR 1
5 MOVE
6 CALLPR 1
7 MOVE
8 CALLPR 1
9 END

Below shows a task(program 1) to be repeated numerous times.



When CALLPR is recorded in program 2 as above the robot will operate as shown here. CALLPR function cannot be executed again while executing CALLPR function. After other shifts are applied, the shift by CALLPR is done.
The CALLPR outputs abnormality when move command has never been executed before CALLPR is executed. Please note immediately after having turned on the power supply of controller and immediately after selecting program.
This function cannot be used with two manipulators or more included in a unit. There is a possibility of doing the operation that the robot doesn't anticipate.

■ Parameter

Parameter No. 1	Program No.	This specifies the number of the program serving as the call destination. (0-9999)

■ Example of screen display

CALLPR [2]	FN102; Relative program call

See
CALLPRI: Conditional relative program call (FN103)
CALLPRN: Relative program call (freq. condition) (FN104)

Function commands (FN codes)

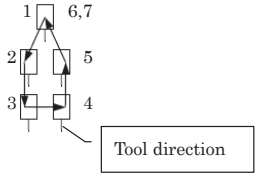
Command name	CALLPRI
FN code	103
Title name	Conditional relative program call
General description	Using an input signal, this command is used to call the specified program.

■ General description

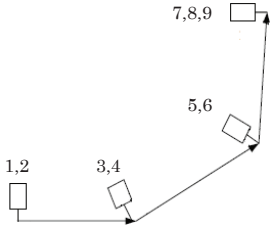
If the designated input signal is being input, [Relative Program Call] calls a subprogram and makes the first step position and orientation the same as the current step in the base program. Then all point positions in the subprogram become relative to that step position. (Refer to the illustration below)
If the input signal is not being input the main program simply continues on. This is a very convenient function for spot welding, weld gun sticking, grinding, cutting and other frequently repeated simple task actions.

■ Example of operation

When you make the program to operate as follows

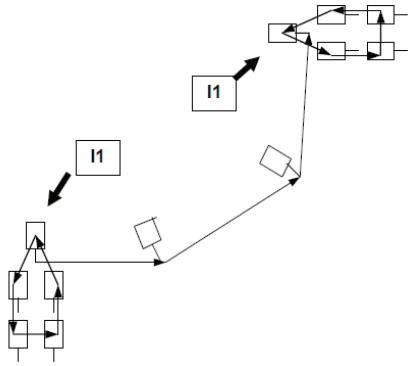


Program 1
1 MOVE
2 MOVE
3 MOVE
4 MOVE
5 MOVE
6 MOVE
7 END



Program 2
1 MOVE
2 CALLPRI 1,I1 ← I1 ON
3 MOVE
4 CALLPRI 1,I1 ← I1 OFF
5 MOVE
6 CALLPRI 1,I1 ← I1 OFF
7 MOVE
8 CALLPRI 1,I1 ← I1 ON
9 END

Below shows a task(program 1) to be repeated numerous times.



When CALLPRI function is executed and the designated I signal is being input program 2 is played and robot will operate as shown here. CALLPRI function cannot be executed again while executing CALLPRI function. After other shifts are applied, the shift by CALLPRI is done.
The CALLPRI outputs abnormality when move command has never been executed before CALLPRUI s executed. Please note immediately after having turned on the power supply of controller and immediately after selecting program.
This function cannot be used with two manipulators or more included in a unit. There is a possibility of doing the operation that the robot doesn't anticipate.

■ Parameter

Parameter No. 1	Program No.	This specifies the number of the program serving as the call destination. (0-9999)
Parameter No. 2	Input signal	This records the number of the input signal which is to serve as the condition for executing the call. When number 5101 or above is specified, multiple input signals can be specified. (1-2048, 5101-5196)

■ Example of screen display

```
CALLPRI [2,I1]      FN103; Conditional relative program call
```

See
CALLPR : Relative program call (FN102)
CALLPRN: Relative program call (freq. condition) (FN104)

Function commands (FN codes)

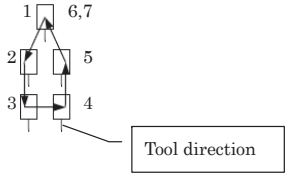
Command name	CALLPRN
FN code	104
Title name	Relative program call (freq. condition)
General description	Using a pass count (number of passes), this command is used to call the specified program.

■ General description

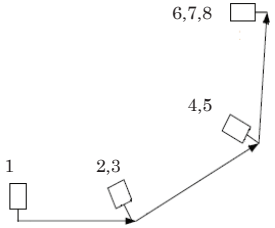
Depending on the number of times CALLPRN function has been passed, [Relative Program Call] calls a subprogram and makes the first step position and orientation the same as the current step in the base program. Then all point positions in the subprogram become relative to that step position. （Refer to the illustration below） This is a very convenient function for spot welding, weld gun sticking, grinding, cutting and other frequently repeated simple task actions.
When the cycle (frequency. + 1) is reached, [Relative Program Call] calls a subprogram, if the cycle is not (frequency + 1) the main program will simply continue on.

■ Example of operation

When you make the program to operate as follows

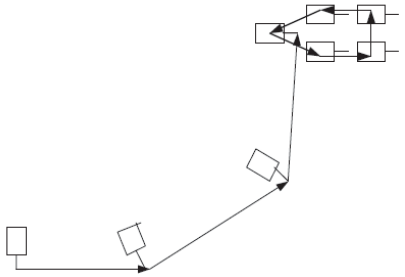


Program 1
1 MOVE
2 MOVE
3 MOVE
4 MOVE
5 MOVE
6 MOVE
7 END



Program 2
1 MOVE
2 MOVE
3 CALLPRN 1,V1%,2
4 MOVE
5 CALLPRN 1,V1%,2
6 MOVE
7 CALLPRN 1,V1%,2
8 END

Below shows a task(program 1) to be repeated numerous times.



When CALLPRN function is being passed on the 3rd (Freq. + 1) cycle program 2 will operate, as shown here. CALLPRN function cannot be executed again while executing CALLPRN function. After other shifts are applied, the shift by CALLPRN is done.

A global integer variable common to all units is used for the number of passes.
The current number of passes can be referenced using monitor/integer variables.

The CALLPRN outputs abnormality when move command has never been executed before CALLPRN is executed. Please note immediately after having turned on the power supply of controller and immediately after selecting program.
This function cannot be used with two manipulators or more included in a unit. There is a possibility of doing the operation that the robot doesn't anticipate.

■ Parameter

Parameter No. 1	Program No.	This specifies the number of the program serving as the call destination. (0-9999)
Parameter No. 2	Register number	A "register" refers to the memory used for counting. Since an integer variable (1 to 200) is used, this parameter specifies its number. (1-200, 301-500)
Parameter No. 3	Number of passes	This records the number of passes which is to serve as the condition for executing the call. The robot passes for the specified number of passes, and on the next time (specified number of passes +1) the call command is executed. (0-10000)

■ Example of screen display

```
CALLPN [2, V1%, 2]      FN104; Relative program call (freq. condition)
```

See
CALLPR: Relative program call (FN102)
CALLPRI: Conditional relative program call (FN103)

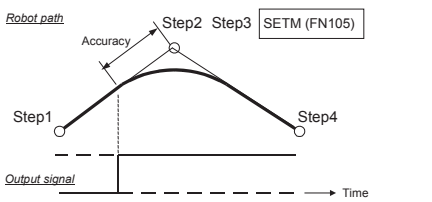
Function commands (FN codes)

Command name	SETM
FN code	105
Title name	Output signal ON/OFF
General description	This command is used to set any general-purpose output signal to ON or OFF.

■ General description

When this function command is executed, any general-purpose output signal (O1 to O2048) is set to ON or OFF.
However, the command cannot set any of the status signals (signals with pre-assigned applications such as the gun signals and starting signal) to ON or OFF. Which signals have been assigned as status signals can be identified on the monitor screen. Signals whose numbers are displayed in the bold italics are status signals so any of the other signals can be set to ON or OFF.
Single-action recording is possible using the [OUT] key on the teach pendant.

■ Example of operation



The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

■ Parameter

Parameter No. 1	Output signal number	This specifies the number of the output signal which is to be turned ON or OFF. (1–2048)
Parameter No. 2	ON/OFF	"1" specified for ON, and "0" for OFF. (0–1)

■ Example of screen display

```
SETM [O17, 1]              FN105; Output signal ON/OFF
```

See
SETO: Consecutive output signal ON/OFF (FN100)

Function commands (FN codes)

Command name	RSCLR
FN code	111
Title name	RS232C Buffer clear
Outline	The send/receive buffer inside the specified RS232C port is cleared

■ General description

When this function command is used, the send/receive buffer inside the specified RS232C port can be cleared.
At the present time, only port 1 can be used as the RS232C port.

■ Example of operation

When the robot reaches the step in which the RSCLR (FN111) command is recorded, the send/receive buffer inside the specified RS232C port is cleared. The robot continues moving while the buffer is being cleared.

■ Parameter

Parameter No. 1	Port number	This is used to specify the number of the serial port whose send/receive buffer is to be cleared. (1 to 1) At the present time, only port 1 can be specified.

■ Example of screen display

```
RSCLR[#1]                  FN111: RS232C Buffer clear
```

See
SREQ: Shift amount request (FN51)
PRINT: Strings output (FN101)

Function commands (FN codes)

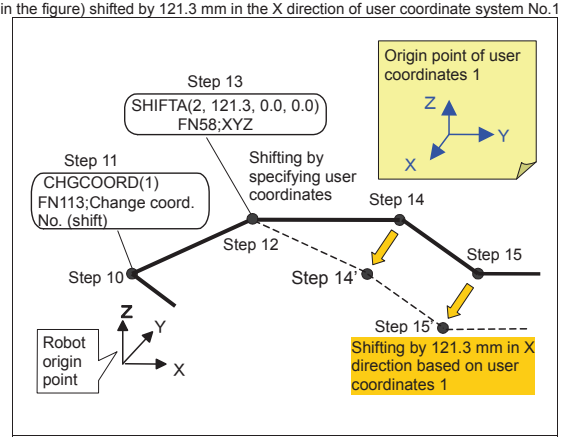
Command name	CHGCOORD
FN code	113
Title name	Change coord. No. (shift)
Outline	This makes it possible to select the number of the user coordinate system used to implement shifts based on the coordinate system. The number of the user coordinate system must be selected without fail before implementing shift-related commands based on the coordinate system.

■ General description

The user coordinate system can be selected as the coordinate system to serve as the reference for shifting when a command such as the "SHIFTR: Shift 2 (FN52)" or "SHIFTA: XYZ shift (FN58)" is used to implement shift operations. The CHGCOORD command specifies the number of the user coordinate system to be used at this time.
When a shift operation has been executed using a user coordinate system with no user coordinate system number selected by this function command, an alarm results, and the robot is stopped.
The user coordinate system must be registered ahead of time using Service Utilities/User coordinates. Up to a hundred coordinate systems can be registered.

■ Example of operation

In the figure below, the function command “SHIFTA: XYZ shift” (FN58) has been recorded in step 13, and a user coordinate system has been selected as the coordinate system to be used for this. Therefore, the command “CHGCOORD: Change coord. No. (shift)” (FN113) is recorded in step 11 which comes before, and user coordinate system No.1 has been specified.
The move commands executed in step 14 and the subsequent steps operate with the positions (positions of steps 14' and 15' in the figure) shifted by 121.3 mm in the X direction of user coordinate system No.1 serving as the movement target.



■ Parameter

Parameter No. 1	User coordinate number	This is used to specify the user coordinate number to be used by the shift-related function commands. (0 to 100) When "0" is specified, the user coordinates are not selected.

■ Example of screen display

```
CHGCOORD[ 5]          FN113: Change coord. No. (shift)
```

See
SHIFTR: Shift2 (FN52)
SHIFTA: XYZ shift (FN58)

Function commands (FN codes)

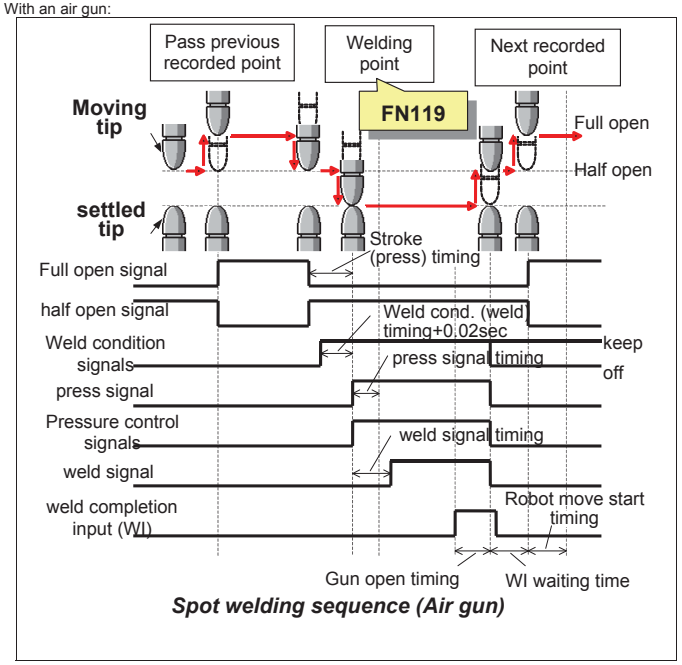
Command name	SPOT
FN code	119
Title name	Spot welding execution
General description	Execute spot welding in accordance with pre-defined sequence.

■ General description

Spot welding can be performed in the specified step by recording the spot welding function in the welding step. By using this function, the output timings of the pressurizing signal, power-on signal, stroke signal and other welding control signals can be programmed by the controller without having to use an external sequencer. A higher level of welding control tailored to each weld point can be exercised by setting a multiple number of welding conditions and welding sequences.
This function is used with either an air gun or servo gun.

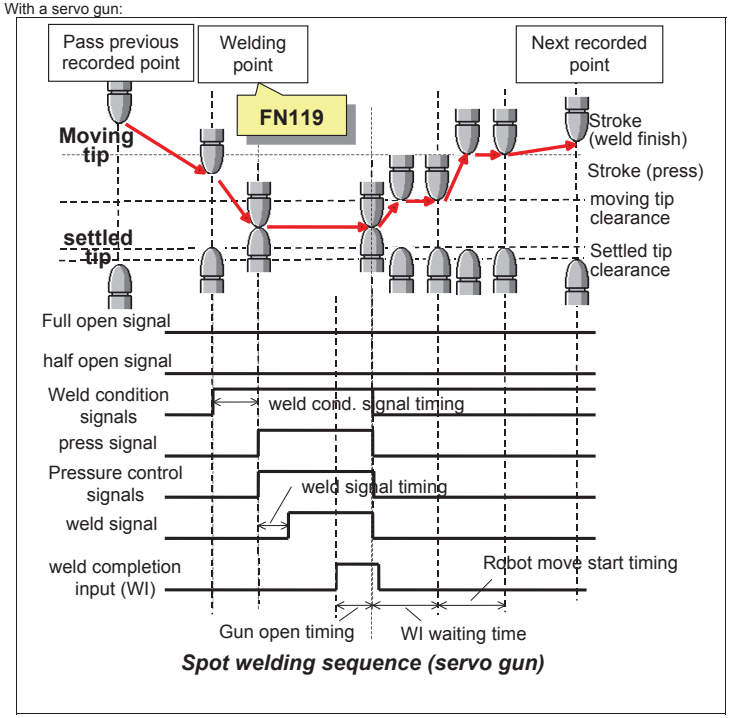
■ Example of operation

With an air gun:



Spot welding sequence (Air gun)

With a servo gun:



Spot welding sequence (servo gun)

■ **Parameter**

Parameter No. 1	Welder number	This parameter specifies the number of the first welder for which the welding control signal is output. (1–6)
Parameter No. 2	Condition#	This specifies the number of the welding condition. It establishes the welding force and welding condition signal. (1–255)
Parameter No. 3	Welding sequence number	This specifies the number of the welding sequence. It establishes the output timing of the pressurizing signal, power-on signal and stroke signal. (1－64)
Parameter No. 4	Welding point number	This specifies the number of the weld point. Use this parameter when controlling the welding points. If welding trouble has occurred, this weld point number is output. In the case of automatic recording, the step number is recorded. (0–16000)

■ **Example of screen display**

```
SPOT [1, 1, 2, 3]          FN119; Spot welding
```

See
GSEA: Servo gun search (FN167)

Function commands (FN codes)

Command name	WAITR
FN code	127
Title name	Wait shift reg. receive
Outline	This initiates a jump to the shelter step when the robot has been waiting for the shift amount data to be input from the external source into the specified shift register and the data has not been input within the specified time.

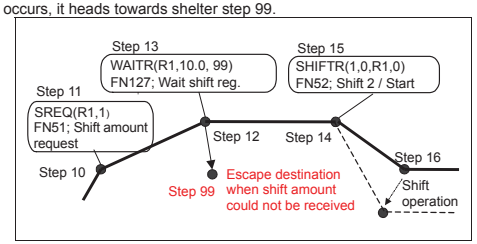
■ **General description**

When this function command is used, the robot operation is stopped and the wait status is established until the shift amount data is input into the specified shift register or until the specified time arrives. As soon as the input of the shift amount data has been detected within the specified time, the next step is executed. The input is judged to have been detected by the setting of "input completed" data in the specified shift register to "1."
When the input of the shift amount data has not been detected within the specified time, operation jumps to the shelter step.

The input of the shift amount data can be confirmed even more definitely by having this function command before the shift-related command that use the shift registers.
If the robot is stopped during shift amount receive wait and then restarted, the remaining time in the specified time serves as the wait time.

■ **Example of operation**

In the figure, the "WAITR: Wait shift reg. receive" (FN127) command is recorded in step 13.
When the program is played back, the robot, after reaching step 13, is set to the wait status until the shift amount data is input in the specified shift register. When the data input is confirmed, the robot heads toward the next step, and when a timeout occurs, it heads towards shelter step 99.



■ **Parameter**

Parameter No. 1	Shift register number	This is used to specify the number of the shift register into which the input of the shift amount data is awaited. (1 to 9)
Parameter No. 2	Wait time	This is used to specify the wait time. (0.0 to 60.0 sec.) When 0.0 sec. is specified, the wait status is established until the shift amount data is input.
Parameter No. 3	Shelter step number	This is used to specify the step to which the robot is to escape in the event that the data was not input during the specified time. (0 to 10000) When 10000 is specified as the shelter step number, an alarm (A2118: “No data has been input in shift register.”) results immediately with no escape operation performed, and the robot can be stopped.

■ **Example of screen display**

```
WAITR[R1, 10.5, 15]          FN127: Wait shift reg. receive
```

See
SREQ: Shift amount request (FN51)
SHIFTR: Shift2 (FN52)

Function commands (FN codes)

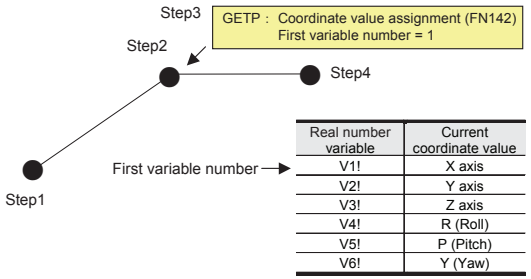
Command name	GETP
FN code	142
Title name	Real number variable assignment (coordinate values)
General description	This command is used to store the current coordinate values (RPY angle expression) in real number variables.

■ General description

When this function command is executed, the current coordinate value expressed as RPY (roll, pitch and yaw) angle of each of the robot's axes is stored in the real number variables with the specified numbers.
(If a wrist posture is to be expressed as Eulerian angles, use the GETPELR: real number variable assignment (coordinate values) (FN94) instead.)

■ Example of operation

In the example of the figure presented below, when the robot arrives at the recorded point in step 2 and positioning is completed, the GETP command in step 3 is executed, and six real number variables are assigned with the coordinate values in sequence starting from the specified real number variable, as shown below.



Real number variable	Current coordinate value
V1!	X axis
V2!	Y axis
V3!	Z axis
V4!	R (Roll)
V5!	P (Pitch)
V6!	Y (Yaw)

The robot reaches the exact recorded point, the GETP command is executed regardless of the accuracy of the move command immediately before and regardless of whether positioning is specified.

The assigned values can be referenced by monitor/real number variables.

■ Parameter

Parameter No. 1	Real number variable number	This specifies the first number of the register (real number variable) in which a coordinate value is to be stored. Six variables are then used in succession. (1–195)

■ Example of screen display

```
GETP [V1!]          FN142; V! Set real var. (cur.pos)
```

See
GETPELR: Real number variable assignment (Eulerian angle coordinate value) (FN94)
GETANGLE: Real number variable assignment (axis angles) (FN157)

Application Command (FN Code)

Command name	GETPOSE
FN code	143
Title name	Assign real variable (pose variable)
Outline	Used to save pose variable Pn in the real variable V!.

■ Outline

This command is used to save pose variable Pn in the real variable V!.

■ Parameters

Parameter 1	Leading variable number	Used to make setting of leading numbers of real variables in which the current robot coordinate values (X, Y, Z, r, p, y) and also pose file numbers. (Setting range: 1 to 195) The coordinate values are saved in six real variables in the order in which the variables were specified. V! [Leading variable number] = X V! [Leading variable number + 1] = Y V! [Leading variable number + 2] = Z V! [Leading variable number + 3] = r V! [Leading variable number + 4] = p V! [Leading variable number + 5] = y
Parameter 2	Pose variable number	Used to make setting of robot posture (pose). (Setting range: 1 to 9999)

■ Example of screen display

```
GETPOSE[V1!, 2]     FN143: Assign real variable (pose variable)
```

Application Command (FN Code)

Command name	LETPOSE
FN code	144
Title name	Assign pose variable
Outline	Used to assign real variable V! to pose variable.

■ Outline

This command is used to assign the real variable V! to pose variable.

■ Parameters

Parameter 1	Pose variable number	Used to make setting of pose variable number to be assigned. (Setting range: 1 to 9999)
Parameter 2	Leading variable number	Used to make setting of leading numbers of real variables. (Setting range: 1 to 195) Taking the values of specified real variables as coordinate values, assign these values to the pose coordinate values in order. X = V! [Leading variable number] Y = V! [Leading variable number + 1] Z = V! [Leading variable number + 2] r = V! [Leading variable number + 3] p = V! [Leading variable number + 4] y = V! [Leading variable number + 5]

■ Example of screen display

```
LETPOSE[2, V1!]      FN144: Assign pose variable
```

Function commands (FN codes)

Command name	GETSFT
FN code	145
Title name	Real number variable assign (shift)
Outline	This command replaces the values of the specified shift register with the specified real number variables (7 consecutive variables are used).

■ General description

When this function command is used, the values (X, Y, Z, θx, θy, θz) recorded in the specified shift register can be assigned in sequence from the number of the specified real number variable. Furthermore, the numerical value (1) indicating that the values have been assigned is set in the number of the next specified real number variable.
The numerical values at the time concerned are assigned regardless of whether or not the data has been set in the specified shift register.

■ Example of operation

When the robot reaches the step where the "GETSTF: Set real var. (shift)" (FN145) command is recorded, the contents of the shift register are immediately read, copied into the real number variables, and set. The robot does not stop.

The numerical values are set (written over the existing data) in 7 consecutive real number variables regardless of their usage status up to this point.

■ Parameter

Parameter No. 1	Real number variable number	This is used to specify the starting number of the real number variables which will store the numerical values inside the shift register. The numerical values are set (overwritten) in 7 consecutive real number variables whose number starts with this. (1 to 94)
Parameter No. 2	Shift register number	This is used to specify the shift register number. (1 to 9)

■ Example of screen display

```
GETSFT[ V1!, R1]          FN145: Set real var.(shift)
```

See
SREQ: Shift amount request (FN51)
SHIFTR: Shift2 (FN52)

Function commands (FN codes)

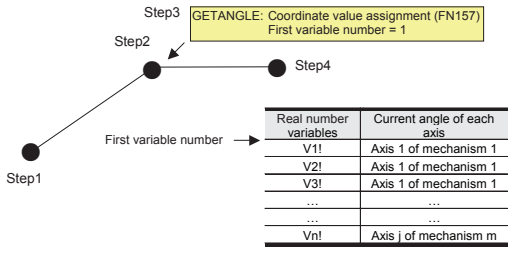
Command name	GETANGLE
FN code	157
Title name	Real number variable assignment (axis angles)
General description	This command is used to store the current angle value of each axis in a real number variable.

■ General description

When this function command is executed, the current angle value (in degrees) for each axis of all the mechanisms in the unit is stored in a real number variable with the specified number.

■ Example of operation

In the example of the figure presented below, when the robot arrives at the recorded point in step 2 and positioning is completed, the GETANGLE command in step 3 is executed, and the number of real number variables corresponding to the number of axes starting from the specified real number variable are assigned in sequence to the axis angle values (in degrees), as shown below.



Real number variables	Current angle of each axis
V1!	Axis 1 of mechanism 1
V2!	Axis 1 of mechanism 1
V3!	Axis 1 of mechanism 1
…	…
…	…
Vn!	Axis j of mechanism m

n: Total number of axes for mechanisms configuring unit, n: 1 to 54

The robot reaches the exact recorded point, the GETANGLE command is executed regardless of the accuracy of the move command immediately before and regardless of whether positioning is specified.

The assigned values can be referenced by monitor/real number variables.

■ Parameter

Parameter No. 1	Real number variable number	This specifies the first number of the register (real number variable) in which a coordinate value is to be stored. Following this, the number of variables that corresponds to the "total number of axes of the mechanisms configuring the unit" are used in succession. (1–195)

Since the total number of axes is high, the number of variables may exceed 200, in which case an error occurs in playback.

■ Example of screen display

```
GETANGLE [V1!]      FN157; V! Set real var. (cur.ang)
```

See
GETPELR: Real number variable assignment (Eulerian angle coordinate value) (FN94)
GETP: Real number variable assignment (RPY angle coordinate value) (FN142)

Function commands (FN codes)

Command name	GETFIGURE
FN code	158
Title name	Set real variable(figure)
General description	The robot figure is used to store in real number variables.

■ General description

When this function command is executed, the position of the mechanism is stored in the V! variable.
You can choice the kind and type of the position of the mechanism as follow.

Kind

Angle	The current angle of each of the mechanism's axes is stored in the V! variable. Unit [deg] or [mm]
TCP Coordinate (RPY)	The current TCP coordinate of each of the mechanism is stored in six chosen V! variables in （X,Y,Z, R,P,Y）order. Output is in user coordinate if you use “FN171 NRLCRD” or else is in the mechanism coordinate Unit [mm]/[deg]
TCP Coordinate (EULER)	The current TCP coordinate of each of the mechanism is stored in six chosen V! variables in （X,Y,Z, α, β, γ）order. Output is in user coordinate if you use “FN171 NRLCRD” or else is in the mechanism coordinate Unit [mm]/[deg]

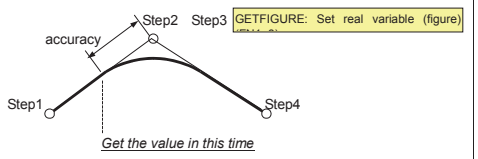
Type

Com	Get the command value in the time.
Current	Get the current value in the time.
Object	Get the object value in the time. This is not depend on accuracy of the move command. The value reflects “shift” if you use “shift” and so on.

This function is carried out while passing. Was that, if the step that an accuracy of the move command is large, gotten value in the time when enter accuracy extent.

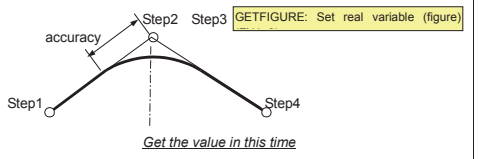
■ Example of operation

case Type =com or cur



Gotten value in the time when enter accuracy extent.
Gotten the value after positioning If you command it.

case Type =obj



This is not depend on accuracy of the move command.
The value reflects “shift” if you use “shift” and so on.

■ **Parameter**

Parameter No. 1	Top Variable (figure)	Select a register used to store in result. （1-195）
Parameter No. 2	Mechanism No.	Select the Mechanism No. (1-9)
Parameter No. 3	Kind	0：ang 1：TCP（RPY） 2：TCP（EULER）
Parameter No. 4	Type	0：Com 1：Cur 2：Obj

If valables is over 200, make an error in the time starting.

■ **Example of screen display**

```
GETFIGURE[V1!, 1, 1, 2]   FN158; Set real variable(figure)
```

See
GETPELR: Real number variable assignment (Eulerian angle coordinate value) (FN94)
GETP；Set real variable（Set real variable(pos)（FN142）
GETANGLE: Real number variable assignment (axis angles) (FN157)
NRLCRD；Change coord. for R-Lang(FN171)

Application Command (FN Code)

Command name	POSAUTO
FN code	160
Title name	Disable posture control
Outline	Used to disable the posture control for robot posture calculation.

■ **Outline**
In order to perform a robot posture calculation, this command disables setting used to find the calculation result in a posture that is forcedly set.

■ **Parameters**

Parameter 1	Select posture control	Used to select posture control to be disabled. (Selection range: 0 to 4) 0: All / 1: Right and left arm system / 2: System above/below elbow / 3: Wrist posture system / 4: Flange shaft

■ **Example of screen display**

```
POSAUTO[0]          FN160: Disable posture control
```

Related commands
LEFTY: Left arm system (FN161)
RIGHTY: Right arm system (FN162)
ABOVE: System above elbow (FN163)
BELOW: System below elbow (FN164)

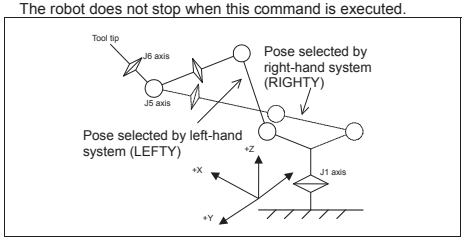
Command name	LEFTY
FN code	161
Title name	Arm config. (left/front)
Outline	The left-arm system posture is forcibly selected for calculating the robot postures

■ General description

This command obtains the angles of the robot axes by calculating them from the head coordinates when shift-related operations are executed. The angles of two patterns capable of expressing the same position are obtained by the calculation. Which angle is to be selected is normally determined automatically and sometimes the robot is not set to the desired pose. When this function command is used, the poses of the left-arm system are forcibly selected from among the calculation results in the subsequent steps.

■ Example of operation

When the "LEFTY: Arm config. (left/front)" (FN161) command is played back, the poses of the left-arm system are forcibly selected when shift-related operations are executed in the subsequent steps.
The robot does not stop when this command is executed.



As soon as the first step (step 0) in the program is played back, the forced selection status is released, and the process of automatically determining which angle to select is restored. However, the forced selection status continues at step 0 of a program which has been called.

■ Parameter

None.

■ Example of screen display

```
LEFTY          FN161: Arm config.(left/front)
```

See
RIGHTY: Arm config. (right/back) (FN162)
ABOVE: Elbow config. (above) (FN163)
BELOW: Elbow config. (below) (FN164)

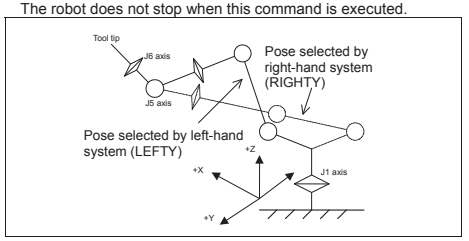
Command name	RIGHTY
FN code	162
Title name	Arm config. (right/back)
Outline	The right-arm system posture is forcibly selected for calculating the robot postures

■ General description

This command obtains the angles of the robot axes by calculating them from the head coordinates when shift-type operations are executed. The angles of two patterns capable of expressing the same position are obtained by the calculation. Which angle is to be selected is normally determined automatically and sometimes the robot is not set to the desired pose. When this function command is used, the poses of the right-arm system are forcibly selected from among the calculation results in the subsequent steps.

■ Example of operation

When the "RIGHTY: Arm config. (right/back)" (FN162) command is played back, the poses of the right-arm system are forcibly selected when shift-related operations are executed in the subsequent steps.
The robot does not stop when this command is executed.



As soon as the first step (step 0) in the program is played back, the forced selection status is released, and the process of automatically determining which angle to select is restored. However, the forced selection status continues at step 0 of a program which has been called.

■ Parameter

None.

■ Example of screen display

```
RIGHTY         FN162: Arm config.(right/back)
```

See
LEFTY: Arm config. (left/front) (FN161)
ABOVE: Elbow config. (above) (FN163)
BELOW: Elbow config. (below) (FN164)

Function commands (FN codes)

Command name	ABOVE
FN code	163
Title name	Elbow config. (above)
Outline	The above-the-elbow (less than 180-degree angle formed by J2 axis and J3 axis) posture is forcibly selected for calculating the robot postures

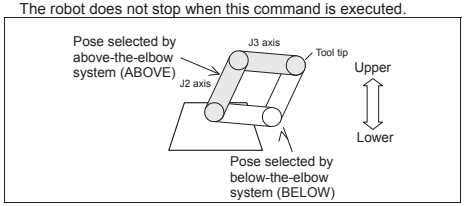
■ General description

This command obtains the angles of the robot axes by calculating them from the head coordinates when shift-related operations are executed. The angles of two patterns capable of expressing the same position are obtained by the calculation. Which angle is to be selected is normally determined automatically and sometimes the robot is not set to the desired pose. When this function command is used, the poses of the above-the-elbow system (less than 180-degree angle formed by J2 axis and J3 axis) are forcibly selected from among the calculation results in the subsequent steps.

■ Example of operation

When the "ABOVE: Elbow config. (above)" (FN163) command is played back, the poses of the above-the-elbow system (less than 180-degree angle formed by J2 axis and J3 axis) are forcibly selected when shift-related operations are executed in the subsequent steps.

The robot does not stop when this command is executed.



As soon as the first step (step 0) in the program is played back, the forced selection status is released, and the process of automatically determining which angle to select is restored. However, the forced selection status continues at step 0 of a program which has been called.

■ Parameter

None.

■ Example of screen display

```
ABOVE          FN163: Elbow config. (above)
```

See
LEFTY: Arm config. (left/front) (FN161)
RIGHTY: Arm config. (right/back) (FN162)
BELOW: Elbow config. (below) (FN164)

Function commands (FN codes)

Command name	BELOW
FN code	164
Title name	Elbow config. (below)
Outline	The below-the-elbow (180-degree angle or more formed by J2 axis and J3 axis) posture is forcibly selected for calculating the robot postures

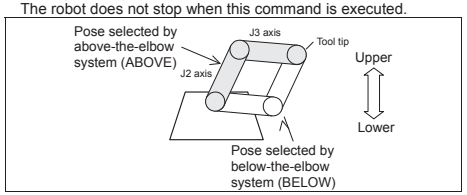
■ General description

This command obtains the angles of the robot axes by calculating them from the head coordinates when shift-related operations are executed. The angles of two patterns capable of expressing the same position are obtained by the calculation. Which angle is to be selected is normally determined automatically and sometimes the robot is not set to the desired pose. When this function command is used, the postures of the below-the-elbow system (180-degree angle or more formed by J2 axis and J3 axis) are forcibly selected from among the calculation results in the subsequent steps.

■ Example of operation

When the "BELOW: Elbow config. (below)" (FN164) command is played back, the poses of the below-the-elbow system (180-degree angle or more formed by J2 axis and J3 axis) are forcibly selected when shift-related operations are executed in the subsequent steps.

The robot does not stop when this command is executed.



As soon as the first step (step 0) in the program is played back, the forced selection status is released, and the process of automatically determining which angle to select is restored. However, the forced selection status continues at step 0 of a program which has been called.

■ Parameter

None.

■ Example of screen display

```
BELOW          FN164: Elbow config. (below)
```

See
LEFTY: Arm config. (left/front) (FN161)
RIGHTY: Arm config. (right/back) (FN162)
ABOVE: Elbow config. (above) (FN163)

Function commands (FN codes)

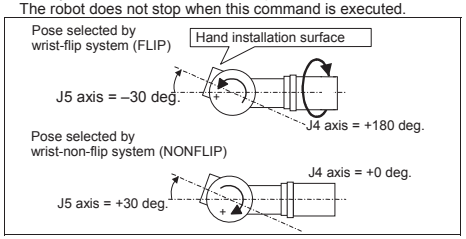
Command name	FLIP
FN code	165
Title name	Wrist config. (flip)
Outline	The poses of the wrist-flip system (where the J5 axis angle is negative) is forcibly selected for calculating the robot postures

■ General description

This command obtains the angles of the robot axes by calculating them from the head coordinates when shift-related operations are executed. The angles of two patterns capable of expressing the same position are obtained by the calculation. Which angle is to be selected is normally determined automatically and sometimes the robot is not set to the desired pose. When this application command is used, the poses at which the angle of the J5 axis is negative are forcibly selected from among the calculation results in the subsequent steps.

■ Example of operation

When the "FLIP: Wrist config. (flip)" (FN165) command is played back, the poses at which the angle of the J5 axis is negative are forcibly selected when shift-related operations are executed in the subsequent steps.
The robot does not stop when this command is executed.



As soon as the first step (step 0) in the program is played back, the forced selection status is released, and the process of automatically determining which angle to select is restored. However, the forced selection status continues at step 0 of a program which has been called.

■ Parameter

None.

■ Example of screen display

```
FLIP          FN165: Wrist config.(flip)
```

See
LEFTY: Arm config. (left/front) (FN161)
RIGHTY: Arm config. (right/back) (FN162)
ABOVE: Elbow config. (above) (FN163)
BELOW: Elbow config. (below) (FN164)
NONFLIP: Wrist config. (nonflip) (FN166)

Function commands (FN codes)

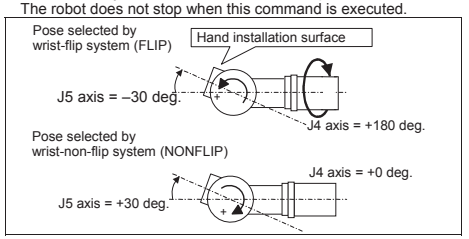
Command name	NONFLIP
FN code	166
Title name	Wrist config. (nonflip)
Outline	The poses of the wrist-non-flip system (where the J5 axis angle is positive) is forcibly selected for calculating the robot postures

■ General description

This command obtains the angles of the robot axes by calculating them from the head coordinates when shift-related operations are executed. The angles of two patterns capable of expressing the same position are obtained by the calculation. Which angle is to be selected is normally determined automatically and sometimes the robot is not set to the desired pose.
When this function command is used, the poses at which the angle of the J5 axis is positive are forcibly selected from among the calculation results in the subsequent steps.

■ Example of operation

When the "NONFLIP: Wrist config. (nonflip)" (FN166) command is played back, the poses at which the angle of the J5 axis is positive are forcibly selected when shift-related operations are executed in the subsequent steps.
The robot does not stop when this command is executed.



As soon as the first step (step 0) in the program is played back, the forced selection status is released, and the process of automatically determining which angle to select is restored. However, the forced selection status continues at step 0 of a program which has been called.

■ Parameter

None.

■ Example of screen display

```
NONFLIP          FN166: Wrist config.(nonflip)
```

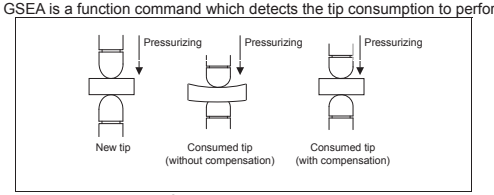
See
LEFTY: Arm config. (left/front) (FN161)
RIGHTY: Arm config. (right/back) (FN162)
ABOVE: Elbow config. (above) (FN163)
BELOW: Elbow config. (below) (FN164)
FLIP: Wrist config. (flip) (FN165)

Function commands (FN codes)

Command name	GSEA
FN code	167
Title name	Servo gun search
General description	This command is used to detect the electrode tip consumption of the servo gun.

■ General description

As welding is performed over and over again, the electrodes gradually start to wear down, and their length shortens. If welding is performed after the electrode lengths have changed, the settled side electrode can no longer be pressed against the work piece: as a result, the work piece is subjected to extra stress, the welding force at the settled side becomes inadequate, etc. The tip consumption must be compensated in order to eliminate these discrepancies.
GSEA is a function command which detects the tip consumption to perform this compensation.



There are three kinds of detection operations: gun search 1 which detects the total tip consumption, gun search 2 which detects the tip consumption on the moving side, and gun search 3 which adds simplified compensation to the tip consumption obtained by gun searches 1 and 2.

Search system	Description of features
Gun Search 1	**Total tip consumption detection** The total tip consumption is measured and subdivided into the settled side tip consumption and moving side tip consumption according to a constant ratio. When gun search 1 is used independently, there is no need for a reference stand, and the tip consumption can be measured in a short period of time. On the other hand, the measurement accuracy is not as good as when both gun search 1 and gun search 2 are performed.
Gun Search2	**Moving side tip consumption detection** This is used together with gun search 1. The moving side tip consumption is measured. The settled side tip consumption is calculated by subtracting the moving side tip consumption from the total tip consumption obtained by gun search 1. This has the highest measurement accuracy. It is necessary to provide a reference stand.
Gun Search 3	**Simplified tip consumption detection** This is used together with gun search 1 and gun search 2. It measures the overall tip consumption. With gun search 3, the tip consumption can be compensated in a simplified way for the tip consumption after gun search 1 and gun search 2 have been executed. The measurement accuracy is not as good as when gun search 3 is performed together with gun search 1 and gun search 2. However, its advantage is that the search operation is completed in one go except when the tips have been replaced. Reference write is not required.

In accordance with the trouble detection level set by constants/spot welding application/servo gun tip consumption detection, an error or information is output, and the corresponding alarm signal is output for the tip consumption obtained.

■ Parameter

Parameter No. 1	Welder number	This specifies the number of the welder to which the servo gun whose electrode tip consumption are to be detected is connected. (1–6)
Parameter No. 2	Gun search number	This specifies the type of gun search to be executed. (1–3) 1: Gun search 1 = Total tip consumption detection 2: Gun search 2 = Moving side tip consumption detection 3: Gun search 3 = Total tip consumption detection (simplified tip consumption detection)
Parameter No. 3	Pressure	This specifies the welding force when detecting the tip consumption. (1.0 – 100.0) [kN]

For detail, refer to information in the FD CONTROLLER OPERATING MANUAL “SPOT WELDING Chapter 4 Servo gun wear amount measurements”.

■ Example of screen display

```
GSEA [1,1,1.5]          FN167; Servo gun search
```

See
SPOT: Spot welding execution (FN119)

Function commands (FN codes)

Command name	SPDDOWNA
FN code	169
Title name	Analog input speed override
Outline	The playback speed of the robot is changed in accordance with the input voltage

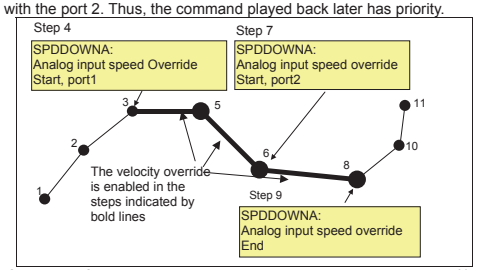
■ General description

With this function command, the playback speed of the robot can be changed in accordance with the analog input voltage. It is necessary to set a velocity override pattern beforehand in the "Constant Setting" mode. (→ Constant Setting / Signals / Velocity override depend on input)
Four analog input channels are provided, any of which can be specified.

This function, however, cannot be used when an analog input board (option) is not installed.

■ Example of operation

In the following diagram, record "SPDDOWNA (FN169) / Start" in steps 4 and 7, and "SPDDOWNA (FN169) / End" in step 9. When this is played back, the velocity override is performed by analog input signals in the sections indicated by the bold lines in the diagram. As far as step 6 the operation is performed in accordance with the port 1, and as far as step 8 in accordance with the port 2. Thus, the command played back later has priority.



If the rate of velocity override by F keys and R codes is set to 50% and that by input signals set to 50%, the actual rate of velocity override is 25% (50% x 50%).
The velocity override by digital input (FN277) is also available, yet this cannot be used together with the SPDDOWNA command.

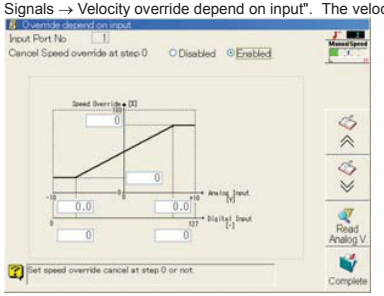
■ Parameter

Parameter No. 1	Start/end	Specify whether to start or end the velocity override by input signals. (1: Start, 0: End)
Parameter No. 2	Input port No.	Select the analog input port to refer to from the four ports. It is ignored when "0: End" is selected as the setting of the first parameter. (1 to 4)

■ Example of screen display

```
SPDDOWNA[1,4]          FN169: Analog input speed override
```

The correspondence between input voltage and speed override must be designed beforehand using "Constant Setting → Signals → Velocity override depend on input". The velocity override by input signal cannot be set over 100%.



If no pattern is defined (all the parameters on the screen are set to 0), the velocity override function is disabled.

See
SPDDOWND: Digital input speed override (FN277)

Application Command (FN Code)

Command name	NRLCRD
FN code	171
Title name	Select robot language coordinate system
Outline	Used to switch functions to a specified user coordinate system.

■ Outline

Using this command makes it possible to switch the following functions to a specified user coordinate system.
- LETX (FN71): Assign X component of pose
- LETY (FN72): Assign Y component of pose
- LETZ (FN73): Assign Z component of pose
- GETP (FN142): Assign real variable (coordinate value)
- GETPOSE (FN143): Assign real variable (pose variable)
- LETPOSE (FN144): Assign pose variable

■ Parameters

Parameter 1	User coordinate system number	Used to make setting of user coordinate system number. (Setting range: 0 to 100)

*When "0" is set to this parameter, the robot will run in the machine coordinate system, but not in the user coordinate system.

■ Example of screen display

```
NRLCRD[1]  FN171: Select robot language coordinate system
```

Related commands

LETX: Assign X component of pose (FN71)
LETY: Assign Y component of pose (FN72)
LETZ: Assign Z component of pose (FN73)
GETP: Assign real variable (coordinate value) (FN142)
GETPOSE: Assign real variable (pose variable) (FN143)
LETPOSE: Assign pose variable (FN144)

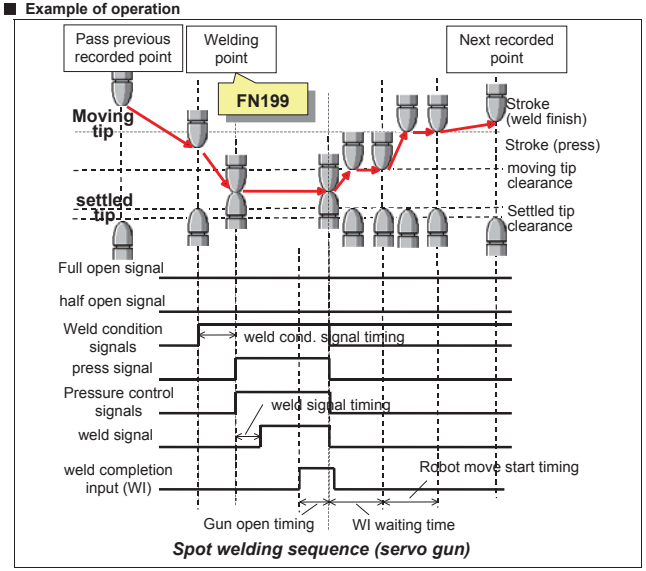
Function commands (FN codes)

Command name	SPOTIWB1
FN code	199
Title name	Spot welding
General description	Execute spot welding in accordance with pre-defined sequence. (dedicated to welding I/F＝MEDbus only)

■ General description

This function is dedicated to the MEDAR spot welder (IWB timer controller) interface application only. **(welding I/F＝MEDbus)**
Spot welding can be performed in the specified step by recording the spot welding function in the welding step. By using this function, the output timings of the pressurizing signal, power-on signal, stroke signal and other welding control signals can be programmed by the controller without having to use an external sequencer. A higher level of welding control tailored to each weld point can be exercised by setting a multiple number of welding conditions and welding sequences.

■ Example of operation



Spot welding sequence (servo gun)

■ Parameter

Parameter No. 1	Welder number	This parameter specifies the number of the first welder for which the welding control signal is output. (1–1)
Parameter No. 2	Welding sequence number	This specifies the number of the welding sequence. It establishes the output timing of the pressurizing signal, power-on signal and stroke signal. (1-64)
Parameter No. 3	Welding point number	This specifies the number of the weld point. Use this parameter when controlling the welding points. If welding trouble has occurred, this weld point number is output. In the case of automatic recording, the step number is recorded. (0–16000)

■ Example of screen display

```
SPOTIWB1[1,2,3]       FN199: Spot welding
```

See

GSEA; Servo gun search (FN167)

Function commands (FN codes)

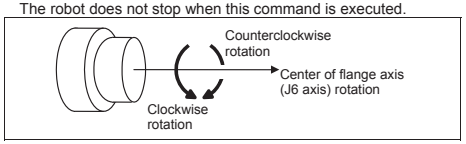
Command name	FRANGE
FN code	202
Title name	Flange axis rot. config.
Outline	The rotational direction of the J6 axis is specified for calculating the robot postures

■ General description

Restrictions stemming from the wiring that accompanies the tools and the installation conditions of the robot sometimes make it preferable for the rotational direction of the tools to be fixed. During palletizing operations and rotational shifts based on the robot language, etc., the poses that minimize the rotational amount are often selected automatically. This sometimes results in operations which differ from the desired rotational direction.
When this finction command is used, the rotational direction of the J6 axis can be forcibly specified.

■ Example of operation

When the "FRANGE: Flange axis rot. config." (FN202) command is played back, the poses are forcibly selected with the rotational direction of the J6 axis specified when shift-related operations are executed in the subsequent steps.
The robot does not stop when this command is executed.



As soon as the first step (step 0) in the program is played back, the forced selection status is released, and the process of automatically determining which angle to select is restored. However, the forced selection status continues at step 0 of a program which has been called.

■ Parameter

Parameter No. 1	ON/OFF	This is used to specify whether to specify the rotational direction or release the specification. (1: ON/2: OFF)
Parameter No. 2	Rotational direction	This is used to specify the rotational direction of the J6 axis. (1: Clockwise rotation/ –1: counterclockwise rotation)

■ Example of screen display

```
FRANGE[1,1]          FN202: Flange axis rot. config.
```

See
LEFTY: Arm config. (left/front) (FN161)
RIGHTY: Arm config. (right/back) (FN162)
ABOVE: Elbow config. (above) (FN163)
BELOW: Elbow config. (below) (FN164)
FLIP: Wrist config.(flip) (FN165)
NONFLIP: Wrist config. (nonflip) (FN166)

Application Command (FN Code)

Command name	COMPON
FN code	206
Title name	Software compliance ON
Outline	Used to enable the software compliance control function.

■ Outline

Using this command enables the software compliance control function, thus making it possible to run the robot according to external force.

■ Parameters

Parameter 1	Condition number	Used to make setting of condition number that is selected within the software compliance data. (Setting range: 1 to 10)
Parameter 2	Command position replace	Used to select whether or not to retrieve the current position and take it as a command position. (1: ON / 2: OFF) 1: Used to retrieve the current position that made a change due to external force in a step in progress and perform a calculation of locus to the subsequent step according to this position. 0: Used to perform a normal locus calculation in a step in progress without retrieving the current position.

■ Example of screen display

```
COMPON[1,1]          FN206: Software compliance ON
```

Related commands
COMPOFF: Software compliance OFF (FN207)

Application Command (FN Code)

Command name	COMPOFF
FN code	207
Title name	Software compliance OFF
Outline	Used to disable the software compliance control function.

■ Outline
Using this command disables the software compliance control function.

■ Parameters
None

■ Example of screen display
```
COMPOFF          FN207: Software compliance OFF
```

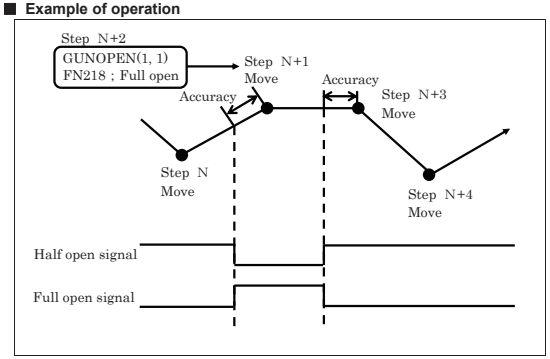
Related commands
COMPON: Software compliance ON (FN206)

Function commands (FN codes)

Command name	GUNOPEN
FN code	218
Title name	Gun Open
General description	This command is used to change stroke of air gun.

■ General description
This function changes a stroke of air gun to half open or full open.
If you select half open, half open signal is on and full open signal off.
If you select half open, half open signal is on and full open signal off.

If you use this function, you must set “Full/half gun open function” of [Constant Setting][13 Spot welding application][1 Spot welder setting] is enabled.

■ Example of operation


■ Parameter

Parameter No. 1	Welder number	The number of Welder number. (1–6)
Parameter No. 2	Half open / Full open	"1" means full open stroke, and "0" means half open stroke. (0–1)

■ Example of screen display
```
GUNOPEN [1, 1]    FN218; Gun Open
```

Function commands (FN codes)

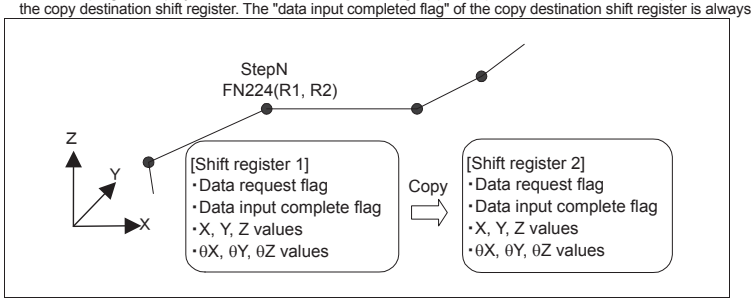
Command name	REGC
FN code	224
Title name	Shift register copy
Outline	Data is copied between shift registers

■ General description

When this function command is used, data can be copied between shift registers.

■ Example of operation

The "REGC: Shift register copy" (FN224) command is recorded in step N.
When the program is played back, the robot, after reaching step N, copies the contents of the copy source shift register into the copy destination shift register. The "data input completed flag" of the copy destination shift register is always set to "1."



■ Parameter

Parameter No. 1	Copy source shift register number	This is used to specify the number of the copy source shift register. (1 to 9)
Parameter No. 2	Copy destination shift register number	This is used to specify the number of the copy destination shift register. (1 to 9)

■ Example of screen display

```
REGC[ R1, R4]          FN224: Shift register copy
```

See
SREQ: Shift data request (FN51)
SHIFTR: Shift2 (FN52)

Function commands (FN codes)

Command name	GSEA_ORDER
FN code	229
Title name	Servo gun search order
General description	Servo gun search2 is execute before servo gun search1

■ General description

When gun search 1 and gun search 2 are to be used together, their execution sequence can be specified. The execution sequence is specified using a function command. Under normal circumstances, set so that gun search 1 is executed first followed by gun search 2. In this case, there is no need to register the function command. Use this sequence when, due to the system configuration or other factors, it is better to execute gun search 2 first.

Please refer to GSEA: Servo gun search (FN167) for the detail of Servo gun search.

■ Example of operation

First Search 1, and next search 2

```
 1 REM[search]      FN99: comment
 2 300 mm/s LIN     A1   T1
 3 300 mm/s LIN     A1   T1
 4 GSEA[1,1,10]     FN167: Servo gun search
 5 300 mm/s LIN     A4   T1
 6 300 mm/s LIN     A1   T1
 7 GSEA[1,2,10]     FN167: Servo gun search
 8 300 mm/s LIN     A4   T1
 9 300 mm/s LIN     A1   T1
10 END              FN99:end
```

First Search 2, and next search 1

```
 1 REM[search]      FN99: comment
 2 300 mm/s LIN     A1   T1
 3 300 mm/s LIN     A1   T1

 4 GSEA_ORDER[1]    FN229: Servo gun search order

 5 GSEA[1,2,10]     FN167: Servo gun search
 6 300 mm/s LIN     A4   T1
 7 300 mm/s LIN     A1   T1
 8 GSEA[1,1,10]     FN167: Servo gun search
 9 300 mm/s LIN     A4   T1
10 300 mm/s LIN     A1   T1
11 END              FN99:end
```

If what has been specified in the search sequence and the gun search sequence differ, an alarm is detected during playback, and the robot stops.

■ Parameter

Parameter No. 1	Gun number	This parameter specifies the number of the gun which is to measure the wear amount. (1–32)
Parameter No. 2	Execution sequence	This parameter specifies the execution sequence of gun search 1 and gun search 2. (0–1) 0; Gun search 1 is executed first followed by gun search 2. 1; Gun search 2 is executed first followed by gun search 1.

■ Example of screen display

```
GSEA_ORDER[1,1]     FN229: Servo gun search order
```

See
SPOT: Spot welding (FN119)
GSEA: Servo gun search (FN167)
SGTIPRST: Reset servo gun tip consumption (FN270)

Application Command (FN Code)

Command name	COLSEL
FN code	230
Title name	Select interference detection level
Outline	Used to select a detection level.

■ Outline

Using this command makes it possible to select a threshold used to determine interference.
For example, use this command to reduce interference detection sensitivity or prevent detection errors in cases where you want to put a robot tip in narrow places, have a close look at interference, keep threshold levels low, increase detection sensitivity, and make the robot conduct contact work.

■ Parameters

Parameter 1	Level number	Used to make setting of detection level number. (Setting range: 1 to 3) 0: Normal sensitivity 1: High sensitivity 2: Low sensitivity 3: Disabled

∗ The operator class ***Expert*** or greater is required.

■ Example of screen display

Application Command (FN Code)

Command name	CHGXXGUN
FN code	238
Title name	Mechanism change
Outline	Used to select a mechanism to be operated manually. (Command dedicated to Mechanism 2 / Mechanism change type “Unlock”)

■ Outline

This application command is used for mechanism changes without electrical disconnection or unlocked mechanism changes (e.g. to control two gun arms with a single motor for servo gun application).
Use this application command to make mechanism changes.
(Command dedicated to Mechanism 2 / Mechanism change type “Unlock”)

■ Example of motion

When the mechanism change type is set to “Unlock”, teaching will be conducted as shown below.

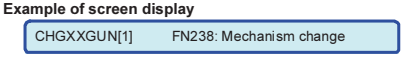
STEP N	**500mm/s LIN A1 T2**		
STEP N+1	**100mm/s LIN A1 T2**		
STEP N+2	**CHGXXGUN[1]**	FN238: Mechanism change	Change to sub-mechanism 1
STEP N+3	**100mm/s LIN A1 T1**		Move to weld point
STEP N+4	**SPOT[1,1,1,1,]**	FN119: Spot welding	
STEP N+5	**100mm/s LIN A1 T1**		
STEP N+6	**100mm/s LIN A1 T1**		
STEP N+7	**CHGXXGUN[2]**	FN238: Mechanism change	Change to sub-mechanism 2
STEP N+8	**100mm/s LIN A1 T2**		Move to weld point
STEP N+9	**SPOT[1,1,1,1,]**	FN119: Spot welding	
STEP N+10	**100mm/s LIN A1 T2**		

For steps in which mechanism changes are made, teach positions in which both guns can move. If one gun is located outside the motion range even though the other gun is located within the motion range (software limit), “software stroke over” error may be detected when executing the mechanism change function.
Set a correct tool number to the spot welding function execution step. Not doing so will disable proper equalizing operation. If the “Tool number check” parameter on the Servo Gun usage condition screen is set to “Enabled”, the tool number recorded and that set with the “Spot welding gun setting” parameter will be verified. If different, an error will be detected to stop playback of mechanism change.

■ Parameters

Parameter 1	Sub-mechanism number	Used to make setting of sub-mechanism number. (Setting range: 1 to 31)

■ Example of screen display



Function commands(FN codes)

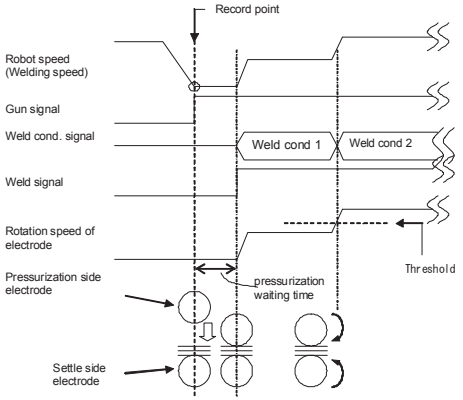
Command name	SEAMST
FN code	245
Title name	Seam weld start
General description	Start the seam welding.

■ General description

Execution of this command starts seam welding.
First, pressurization operation of the electrode by the side of pressurization is carried out, and a work is put. Next, after applying delay by pressurization waiting time, the weld signal and a welding condition signal are outputted. Furthermore, rotation of electrodes which synchronized with a robot's speed is started. The rotation direction is specified by a parameter.
By this command, welding conditions can be automatically changed according to welding speed. It can have a maximum of 5 sets of correspondences of the welding speed and a welding condition number. This correspondence data is recorded as a parameter of a function. Moreover, the welding condition output signals can be outputted by the WELDCND function(FN33) to use this automatic welding condition change function. In this case, the parameter 1 of a function is set as 0.
Furthermore, when the parameter 1 of a function is set as 2, electrodes are rotated at the speed specified with the parameters, without carrying out pressurization. Pressurization side electrode rotates at the speed set as the parameter 3, and settle side electrode rotates at the speed set as the parameter 5.
This is used in dressing of electrodes.

■ Example of operation

The robot shall move to a specified position according to specified motion command and start seam welding as follow time chart.



■ Parameter

Parameter No.1	0-2	Welding condition (0:Manu/1:Auto)
Parameter No.2	0-1	Rotation direction (0:Normal/1:Reverse)
Parameter No.3	0.0-10.0	Press wait time(sec)
Parameter No.4	0-65535	Weld condition 1 (Auto)
Parameter No.5	0-1000	Weld speed 1 (Auto)
	0-1000	Speed of electrode (Dress)
Parameter No.6	0-65535	Weld condition 2 (Auto)
Parameter No.7	0-1000	Weld speed 2 (Auto)
	0-1000	Speed of electrode (Dress)
Parameter No.8	0-65535	Weld condition 3 (Auto)
Parameter No.9	0-1000	Weld speed 3 (Auto)
Parameter No.10	0-65535	Weld condition 4 (Auto)
Parameter No.11	0-1000	Weld speed 4 (Auto)
Parameter No.12	0-65535	Weld condition 5 (Auto)
Parameter No.13	0-1	Seam gun type (0:Both type/1:One type)
Parameter No.14	0-30	Press value(mm) (One type)
Parameter No.15	0-1	Weld operation (0:Continue/1:Pitch/2:Dress)
Parameter No.16	0-5000	Welding distance(mm) (Pitch)
Parameter No.17	0-5000	Idle distance(mm) (Pitch)

The Unit of Parameter No.5, No.7 and No.9 are changed by the parameter,“unit of velocity” in [Constant menu] [Spot Welding Application] [Seam Welding] monitor. Details is explained by the manual of Seam Welding.

Example of screen display

```
SEAMST[1, 0, 0.6, 1, 5, 2, 10, 3, 15, 4, 20, 5, 0, 0, 0, 0, 0]   FN245:Seam weld start
```

See
SEAMEND: Seam weld end (FN246)
SEAMSPDD: Seam electrode speed (FN247)
WELDCND: Spot condition output (FN33)
SEAMOV: Seam override (FN313)

Function commands(FN codes)

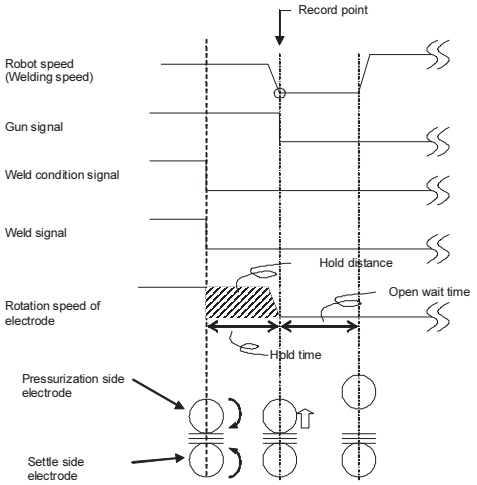
Command name	SEAMEND
FN code	246
Title name	Seam weld end
General description	Stop the seam welding.

■ General description

Execution of this command ends seam welding started by the seam welding start function SEAMST.
From position attainment of the move command on which the function was recorded, weld signal turns off previously only the order to time or order to distance specified with the parameter, and it ends welding. Then, rotations of electrodes are stopped at the same time a robot is positioned by hold time progress. Next, gun signals are made to turn off and between electrodes are opened wide. Furthermore, the following step is performed after the open waiting time progress specified with the parameter.

■ Example of operation

The robot shall move to a specified position according to specified motion command and end seam welding.



■ Parameter

Parameter No.1	0-10.0	Time (sec)
	0-5000.0	Distance (mm)
Parameter No.2	0-10	Open wait time (sec)
Parameter No.3	0-1	Pre-out timing of electric off 0:Order to time / 1:Order to distance

■ Example of screen display

```
SEAMEND[1,1,0]                FN246:Seam weld end
```

See
SEAMST: Seam weld start (FN245)
SEAMSPD : Seam electrode speed (FN247)
WELDCND: Spot condition output (FN33)

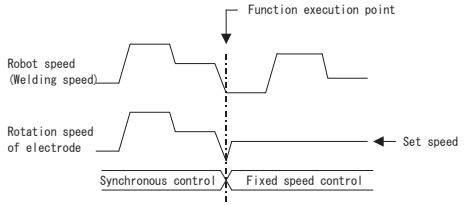
Function commands(FN codes)

Command name	SEAMSPD
FN code	247
Title name	Seam electrode speed
General description	Set the electrode rotation speed while welding.

■ General description

Fundamentally, the rotation speed of electrodes are automatically controlled after seam welding start function execution to become the speed corresponding to a robot's speed. However, the speed of electrodes can be specified by this command to tune speed only for a specific part finely.

■ Example of operation



■ Parameter

Parameter No.1	0-1000	Speed 1 Welding speed of move-side electrode.
Parameter No.2	0-1000	Speed 2 Welding speed of settle-side electrode.

The Unit of Parameter No.1 and No.2 are changed by the parameter, “unit of velocity” in [Constant menu] [Spot Welding Application] [Seam Welding] monitor. Details is explained by the manual of Seam Welding.

■ Example of screen display

```
SEAMSPD[30,30]                FN247:Seam electrode speed
```

See
SEAMST: Seam weld start (FN245)
SEAMEND: Seam weld end (FN246)

Function commands (FN codes)

Command name	EQUALIZECLR
FN code	248
Title name	Equalize clear
General description	The equalize setting clear.

■ General description

The equalize setting set by the EQUALIZE; Equalize value (FN287) function is cleared.

■ Parameter

Parameter No. 1	Welder No.	This parameter specifies the welder number. (1–6)

■ Example of screen display

```
EQUALIZECLR[1]                                FN248 ; Equalize clear
```

See

SPOT ; Spot welding (FN119)
TIPDRESS ; Tip dress (FN265)
EQUALIZE ; Equalize value(FN287)

Function commands (FN codes)

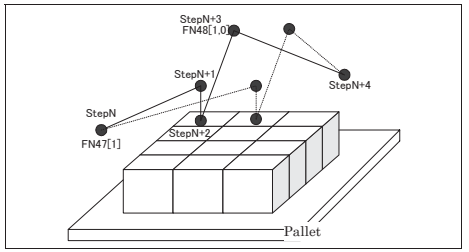
Command name	PALLET3
FN code	249
Title name	Palletize start
General description	Start palletizing based on the pre-designed palletizing pattern.

■ General description

When this function command is executed, the specified palletizing can be started.
It is executed in tandem with the FN250 "Palletize end" command which is used to end palletizing.

■ Example of operation

As shown in the figure below, record “PALLET3: Palletize start” (FN249) in step N and “PALLET3 END: Palletize end” (FN250) in step N+3.
When the program is played back and the robot reaches step N, the shift amount is calculated from the palletize number specified by FN249 and the palletize counter, and the shift operation is performed. Step N+1 and step N+2 are shifted. (The robot passes along the path indicated by the dotted lines in the figure below.) When it reaches step N+3, FN250 is executed, and the palletizing operation is ended. The robot now heads toward the point where step N+4 is recorded.



■ Parameter

Parameter No.1	Palletize No.	This specifies the palletizing number to be executed. (1 to 100)
Parameter No.2	Palletize type	0:palletize 1:de-palletize
Parameter No.3	Layer count signal	O-signal No. for layer counter. (1 to 1024) The counter is outputted by binary data. This number shows the last bit.
Parameter No.4	Work count signal	O-signal No. for work counter. (1 to 1024) The counter is outputted by binary data. This number shows the last bit.

■ Example of screen display

```
PALLET3[1, 0, O1, O11]          FN249 Palletize start
```

See

PALLET3_END: Palletize end (FN250)
PALLET3_RESET: Palletize reset (FN251)
PALLET3_APR: Palletize approach selection(FN374)
PALLET3_OPT: Palletize optimize path(FN375)
PALLET3_SELGR: Palletize select grasp position(FN376)
PALLET3_GETREG: Get palletize register(FN377)
PALLET3_SETREG: Set palletize register(FN378)

Function commands (FN codes)

Command name	PALLET3_END
FN code	250
Title name	Palletize end
General description	Finish palletizing based on the pre-designed palletizing pattern.

■ General description

When this function command is executed, the specified palletizing work can be completed.
It is executed in tandem with the FN249 "Palletize start" command which is used to start palletizing.

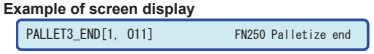
■ Example of operation

As shown in the figure below, record “PALLET3: Palletize start” (FN249) in step N and “PALLET3 END: Palletize end” (FN250) in step N+3.
When the program is played back and the robot reaches step N, the shift amount is calculated from the palletize number specified by FN249 and the palletize counter, and the shift operation is performed. Step N+1 and step N+2 are shifted. (The robot passes along the path indicated by the dotted lines in the figure below.) When it reaches step N+3, FN250 is executed, and the palletizing operation is ended. The robot now heads toward the point where step N+4 is recorded.
Refer to the picture of FN249 PALLET3’s Help screen.

■ Parameter

Parameter No.1	Palletize No.	This specifies the number of the palletizing operation which is to be ended. (1 to 100)
Parameter No.2	Completion signal	When the palletizing finishes all works, this O-signal is outputted. (1 to 1024)

■ Example of screen display



See
PALLET3: Palletize start (FN249)
PALLET3_RESET: Palletize reset (FN251)
PALLET3_APR: Palletize approach selection(FN374)
PALLET3_OPT: Palletize optimize path(FN375)
PALLET3_SELGR: Palletize select grasp position(FN376)
PALLET3_GETREG: Get palletize register(FN377)
PALLET3_SETREG: Set palletize register(FN378)

Function commands (FN codes)

Command name	PALLET3_RESET
FN code	251
Title name	Palletize reset
General description	The palletize counter is forcibly reset. (palletizing operation is forcibly terminated)

■ General description

The specified palletize counter can be forcibly reset (cleared to zero).
If this function is executed while palletizing is under execution, the palletizing is forcibly terminated.

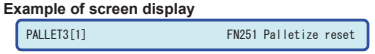
● Signals

When this function is executed, layer counter signal and work counter signal are cleared. And ACK signal is turned on.

■ Parameter

Parameter No.1	Palletize No.	This specifies the palletizing number to be reset. (1 to 100)

■ Example of screen display



See
PALLET3: Palletize start (FN249)
PALLET3_END: Palletize end (FN250)
PALLET3_APR: Palletize approach selection(FN374)
PALLET3_OPT: Palletize optimize path(FN375)
PALLET3_SELGR: Palletize select grasp position(FN376)
PALLET3_GETREG: Get palletize register(FN377)
PALLET3_SETREG: Set palletize register(FN378)

Function commands (FN codes)

Command name	PAUSEINPUT
FN code	252
Title name	Pause Input
General description	The robot is pause when the designated [Pause input] signal is turned off.

■ General description

The robot enters the state of the standby if the designated input signal is turned off when this function is effective. Moreover, when the designated input signal is turned on, the robot restarts operation.

■ Parameter

Parameter No. 1	1～4	Signal No.
Parameter No. 2	0～1	enabled/disabled(1/0)

■ Example of screen display

```
PAUSEINPUT[4,1]          FN252;Pause Input
```

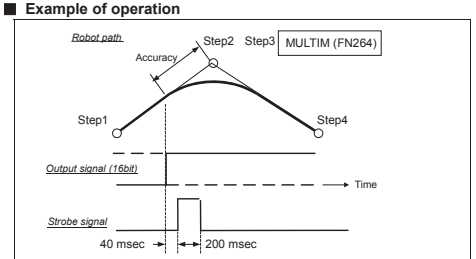
Function commands (FN codes)

Command name	MULTIM
FN code	264
Title name	Multi Output signal
General description	This command is used to set the pre-defined multiple output signals to ON or OFF using the binary format.

■ General description

When this function command is executed, it is possible to set any general-purpose output signals (O1 to O2048) to ON or OFF using the binary format.
Up to 16 general-purpose output signals (O1 to O2048) are combined in any way, and by defining them as a group (multiple output signals) and specifying a number, these multiple output signals are set to ON or OFF using the binary format. This command differs from the OUT: binary format output signal (FN44) command in that up to 16 general-purpose output signals forming a group can be combined in any way.
The group must have already been defined in the constants mode.
However, the command cannot set any of the status signals (signals with pre-assigned applications such as the gun signals and starting signal) to ON or OFF.
Which signals have been assigned as status signals can be identified on the monitor screen. Signals whose numbers are displayed in the bold italics are status signals so any of the other signals can be set to ON or OFF.

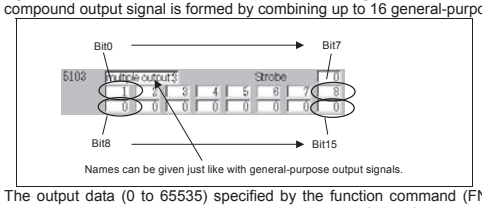
■ Example of operation



The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

As soon as the accuracy is matched, a 16-bit signal is output, and 40 ms later the strobe signal is output by a 200 ms pulse.
If the output data setting has exceeded the number of defined bits of the general-purpose output signals, the data is output within the range of the number of bits defined. (Error stop does not occur.)

Definition of multiple output signals (constants mode)
The multiple output signals are defined ahead of time by input/output signals/output signal assignment/compound output signals in the constants mode. Compound output signals can be defined in 96 different ways from 5101 to 5196, and one compound output signal is formed by combining up to 16 general-purpose output signals.



The output data (0 to 65535) specified by the function command (FN264) is converted into a 16-bit binary value, and the general-purpose output signals concerned are set to ON or OFF. In order to provide the read timing of the output signals, one strobe signal can be defined in addition to the 16 general-purpose output signals. (If this signal is not going to be used, its number is zero.) It is not possible to specify other multiple output signals as signals which configure multiple output signals.

■ Parameter

Parameter No. 1	Output signal number	This specifies the numbers of the multiple output signals to be output. It is defined in advance in the constants mode. (5101 – 5196)
Parameter No. 2	Output data	This specifies the data to be output in the binary format by the multiple output signals. (0 – 65535)

■ Example of screen display

```
MULTIM [05102, 65535]     FN264; Multiple output signals
```

See
OUTDIS: Discrete format output signals (FN43)
OUT: Binary format output signal (FN44)

Function commands (FN codes)

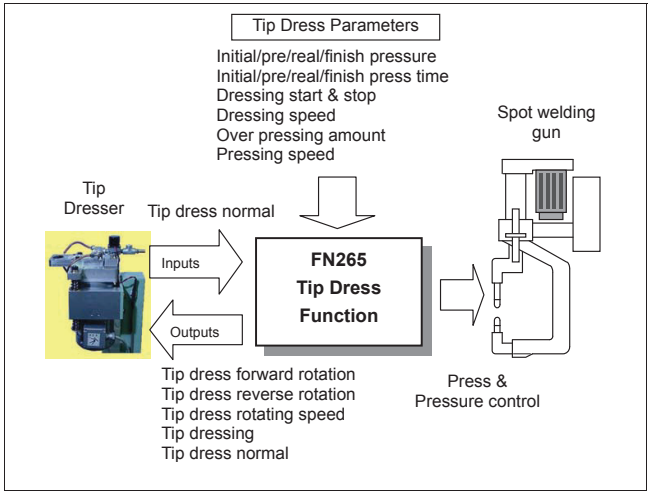
Command name	TIPDRESS
FN code	265
Title name	Tip dress
General description	Execute the tip dress of spot welding gun

■ General description

During the course of welding, the tips of the electrodes eventually wear down and their shapes change. Since the shape of the electrode tips determines the current density of the current-carrying path, proper welding cannot be performed if the electrodes have worn down.
Therefore, before the wear amounts become significant and the welding quality is impaired, the electrodes must be ground and reshaped. This work is known as “tip dressing”. A machine called a “tip dresser” is used to dress the tips. It has a rotary blade matching the shape of the electrodes.

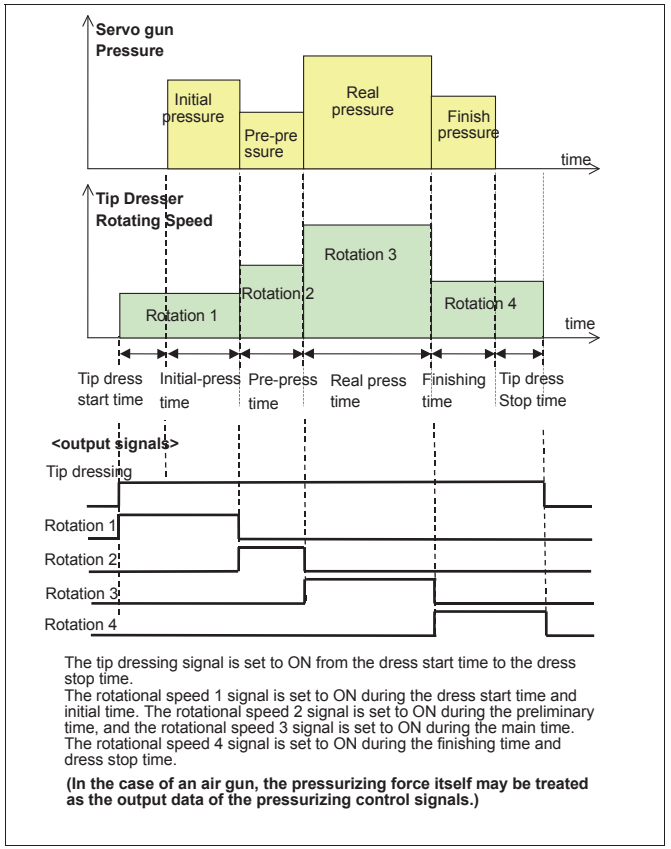
The electrodes are ground by rotating the rotary blade while pressure remains applied to the blade by the spot welding gun. Normally, tip dressing is taught so that it is performed when the prescribed number of welding times or cycles have been executed.
The tips are generally dressed by executing welding operations in the welding OFF status. For the rotational direction and speed of the dresser and the time to be controlled, an interface using I/O signals between the robot controller and tip dresser is required.

This function, which utilizes the characteristics inherent to a servo gun, divides the tip dressing cycle into six periods, two of which are the tip dressing start time and the initial pressurizing time. By specifying the servo gun welding force and rotational speed of the tip dresser for each of these periods, the function achieves high-quality tip dressing without using an external sequencer.



■ Example of operation

Proceed with the same teaching as for the spot welding function (FN119). Record the positions where contact has been made on the tip dresser at both the moving and fixed sides, and record the tip dressing function TIPDRESS (FN265) in the step concerned.
When TIPDRESS(FN265) is executed, pressure and rotating speed of tip dresser is controller under the condition that is defined in Service / 20 Spot welding application / 3 Tip dress.



The tip dressing signal is set to ON from the dress start time to the dress stop time.
The rotational speed 1 signal is set to ON during the dress start time and initial time. The rotational speed 2 signal is set to ON during the preliminary time, and the rotational speed 3 signal is set to ON during the main time. The rotational speed 4 signal is set to ON during the finishing time and dress stop time.

(In the case of an air gun, the pressurizing force itself may be treated as the output data of the pressurizing control signals.)

Input & output signals should be assigned in Constant setting / 3 Input & output / 2 Input signal assignment or 3 Output signal assignment.

■ Parameter

Parameter No. 1	Welder number	This is used to specify the number of the tip dressing welder. (1-6)
Parameter No. 2	Tip dressing sequence number	is used to specify the sequence number to be used for tip dressing.(1-8)

■ Example of screen display

```
TIPDRESS[1,3]          FN265: Tip dress
```

See
SPOT; Spot welding (FN119)

Application Command (FN Code)

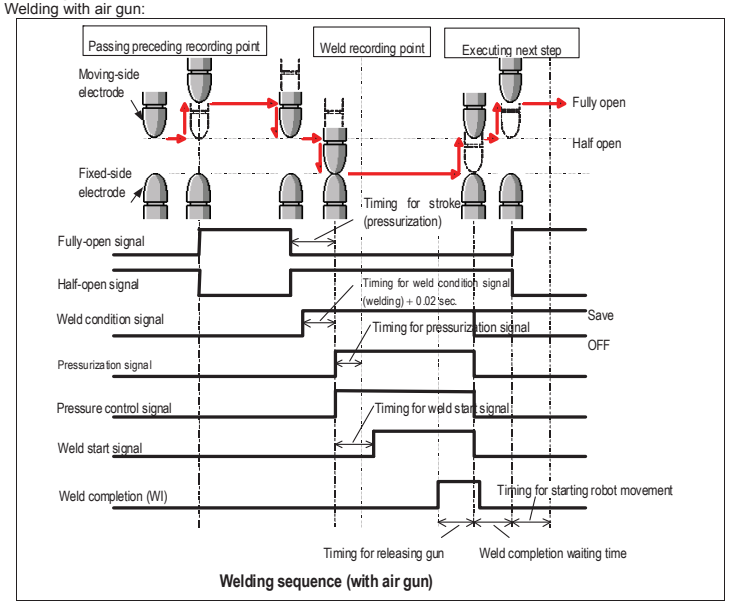
Command name	SPOT2
FN code	268
Title name	Spot welding
Outline	Used to output welding signals according to the set sequence.

■ Outline

By recording the spot welding function in steps in which welding is carried out, spot welding can be carried out in the steps specified. Using this function enables the controller to program timing for outputting welding control signals such as pressurization signal, weld start signal, and stroke signal without using any external programmable logic controller. In addition, making setting of multiple welding conditions and welding sequence enables detailed welding control by welding point.
This function is used for both air guns and servo guns.

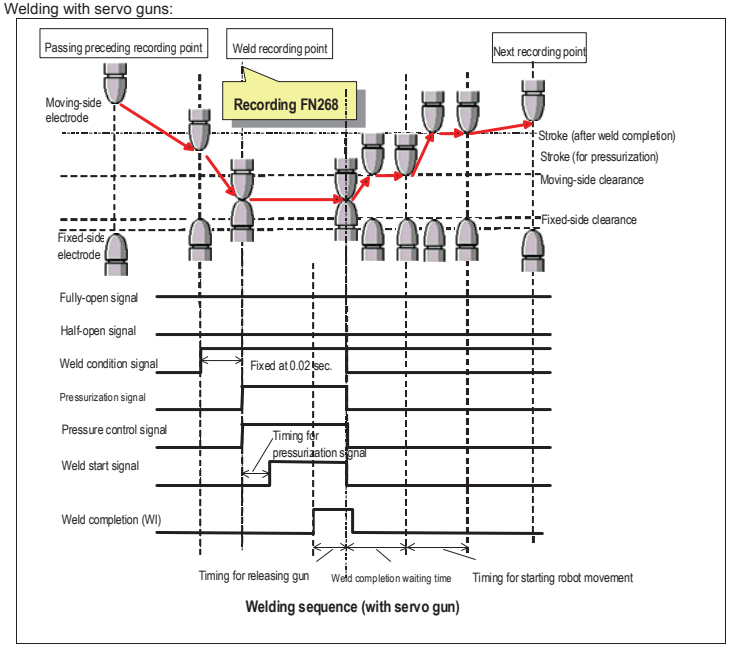
■ Examples of motions

Welding with air gun:



Welding sequence (with air gun)

Welding with servo guns:



Welding sequence (with servo gun)

■ Parameters

Parameter 1	Weld condition output data	Used to make setting of welding condition output data. (Setting range: 0 to 65535)
Parameter 2	Weld sequence No.	Used to make setting of welding sequence number and also determine timing for outputting pressurization signal, weld start signal, and stroke signal. (Setting range: 1 to 64)
Parameter 3	Pressure	Used to make setting of gun pressure for pressurization. This parameter is ignored for welding with air gun. (Setting range: 0.5 to 10.0 [kN], or 50 to 1,020 [kgf] depending on the selection of unit)
Parameter 4	Thickness	Used to make setting of thickness of plate to be welded. (Setting range: 0 to 20.00 [mm])

■ Example of screen display

```
SPOT2[1, 2, 1.5, 0.7]          FN268: Spot welding
```

Related commands

GSEA: Servo gun weld point search (FN167)

Function commands (FN codes)

Command name	SGTIPRST
FN code	270
Title name	Reset servo gun tip consumption
General description	Reset the tip consumption of designated servo gun

■ General description

The tip consumption amounts can be cleared to zero using a function command. (They cannot be preset to the desired values.)
When the electrodes are to be replaced automatically, record this function command in the electrode replacement program.
Please refer to GSEA: Servo gun search (FN167) for the detail of Servo gun search.

■ Example of operation

When SGTIPRST (FN270) is executed, the moving tip or settled tip or both tip consumption of designated servo gun is reset.

Also, the tip consumption amounts can be cleared to 0 by the following method.

- R code (R339)
- Manual operation
- Input signal

■ Parameter

Parameter No. 1	Gun number	This parameter specifies the number of the gun whose tip consumption amounts are to be reset. (1–31)
Parameter No. 2	Tip selection	This parameter specifies the tip whose amounts is to be reset. (0–2) 0: The consumption amount of only the moving side electrode is reset. 1: The consumption amount of only the settled side electrode is reset. 2: The consumption amounts of both the moving side and settled side electrode are reset.

■ Example of screen display

```
SGTIPRST[1,0]  FN270: Reset servo gun tip consumption
```

See
SPOT: Spot welding (FN119)
GSEA: Servo gun search (FN167)
GSER_ORDER: Servo gun search order (FN229)

Function commands (FN codes)

Command name	INPUT
FN code	271
Title name	Strings input
Outline	This receives the character string data from the specified communication (serial) port, and holds it in the specified character string variable.

■ General description

When this function command is used, a character string which is input from the specified serial port can be read and then stored in the specified character string variable. Only character strings consisting of up to 100 single-byte alphanumerics per string can be read.
The robot stops until the input process of the character string is completed. If the process of inputting the character string could not be completed even after the timeout time has elapsed, it is aborted, and execution is transferred to the next step.

■ Example of operation

When the step where the "INPUT: Strings input" (FN271) command is recorded is played back, the robot checks the positioning and then waits for the data input from the specified serial port.
When any data is received during the specified timeout time, the received data is stored in the specified character string variable. The standby status is released, and the robot moves to the next step.
If no data is input even after the timeout time has elapsed, the robot simply moves to execute the next step.

■ Parameter

Parameter No. 1	Port number	This is used to specify the number of the input destination port. At the present time, only port 1 can be used. (1 to 1)
Parameter No. 2	Character string variable number	This is used to specify the number of the character string variable in which the received character string is to be stored. (1 to 100)
Parameter No. 3	Connection time out	This is used to specify the timeout time. (1.0 to 60.0 sec.)

■ Example of screen display

```
INPUT[#1, V100$, 45.5]          FN271: Strings input
```

See
PRINT: Strings output (FN101)
RSCLR: RS232C Buffer clear (FN111)

Application Command (FN Code)

Command name	CNVYSYNCCHG
FN code	274
Title name	Conveyor synchronize select
Outline	Used to select a mechanism that is synchronized with the conveyor with the conveyor synchronization function.

■ Outline

Using this application command makes it possible to specify a mechanism that follows the conveyor with the conveyor synchronization function.
If this application command is not used, such mechanism will be determined in the order of slider and manipulator included in the targeted unit. The mechanism is automatically selected in such a manner that the slider moves to the motion range in synchronization with the conveyor, and then the manipulator is synchronized with it.

■ Example of motion

Enter a mechanism number that you want to synchronize with the conveyor.

■ Parameters

Parameter 1	Mechanism No.	Used to make setting of mechanism that is synchronized with the conveyor. (Setting range: 0 to 9)

Setting “0” to the mechanism number that you want to synchronize will return the system to the default state:
Default: The mechanism is automatically selected in such a manner that the slider moves to the motion range in synchronization with the conveyor, and then the manipulator is synchronized with it.

■ Example of screen display

CNVYSYNCCHG[2] FN274: Conveyor synchronize select

Function commands (FN codes)

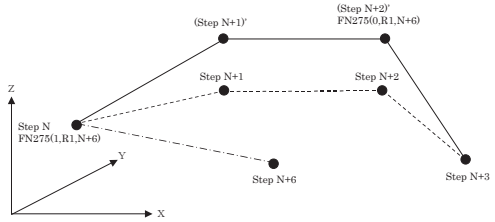
Command name	LOCCVT3
FN code	275
Title name	Base angle shift
Outline	The start or end of the shift operation is specified. When shift operation start has been specified, the shift operation is performed on the basis of the shift amount stored in the specified shift register.

■ General description

This function command proceeds with playback while shifting the recorded position in the robot program on the basis of the shift amount data stored in the specified shift register. If the shift amount data has not been set in the specified shift register, it is possible to jump to the shelter step. Alternatively, the robot can be stopped immediately without escaping.
It is similar to FN52 (shift 2) excluding the thing that the shift processing method and the shift are always done in the machine coordinate system.

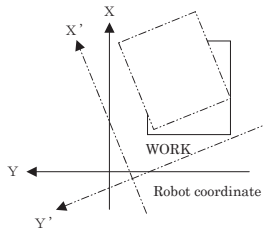
■ Operation example

The shift beginning is recorded at position Step N where the shift wants to begin as shown in the figure below and the shift end is recorded in position (Step N+2) 'which wants to end shifting. The robot reads the content of the shift register specified with FN275 after it reaches step N, and faces position (Step N+1) 'by which the following target position Step N+1 is shifted when reproducing. The robot works as a target position, position where the record position was similarly shifted to position (Step N+2) 'where the shift end is recorded.(solid line tracks of figure below)
The shift is done in the machine coordinate system regardless of the coordinate system selection.



When FN275 is executed in step N, if the shift data is not set when the specified shift register is read, the robot faces the shelter step (Step N+6) which is set with FN275.

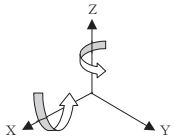
■ About the shift operation



The work when teaching and the position of the robot are shown by the solid line. (Hereafter, explain by two dimensions for the simplification of the explanation)
Two point chain line shows the thing which shifts more than the position of work teaches, and also shows the appearance by which robot coordinate system (X'-Y') is described to become as the work when the shifting work is taught is the same as the relative position between robots.

In the base angle shift,

1. It is assumed the one to show the amount of conversion of the robot coordinate system from robot coordinate system (X-Y) shown by the solid line from which showed in 2 point chain line in amount (X,Y,Z,θx,θy,θz) of the shift. X, Y, and Z correspond to the position of "Starting point of robot coordinate system (X'-Y')" in robot coordinate system (X-Y).
2. θx, θy, and θz are the amounts of the rotation in X axis, Y axis, and Z axis circumference respectively. The plus of ± of the rotation is a direction to each coordinates axis as shown in the figure below where a right screw proceeds.



3. Robot coordinate system (X-Y) sequentially
   (1) θx rotation X axially of robot coordinate system (X-Y)
   (2) θy rotation Y axially of robot coordinate system (X-Y)
   (3) θz rotation Z axially of robot coordinate system (X-Y)
   (4) X, Y, and Z translation in robot coordinate system (X-Y).
   It becomes robot coordinate system (X'-Y') in robot coordinate system

■ **Parameters**

Parameter No. 1	Start/End	This is used to specify the start or end of the shift operation. 1: Start / 0: End
Parameter No. 2	Shift register number	This is used to specify the shift register number. (1 to 9)
Parameter No. 3	Shelter step	This is used to specify the number of the shelter step when the shift amount data was not set in the specified shift register. (0 to 10000) When 10000 is specified as the shelter step number, an alarm (A2118: "No data has been input in shift register.") results immediately with no escape operation performed, and the robot can be stopped.

■ **Example of screen display**

```
LOCCVT3 [1,R1,100]              FN275: Base angle shift
```

See
This function begins processing before the record point reaches. Therefore, it is necessary to have been set before the shift amount data processing.
When shift amount data is set by using SREQ: Shift amount request (FN51) or SREQ2: Shift amount request 2(FN315), it is better to record WAITR: Wait shift value receive(FN127) before this function, to confirm the shift amount data is surely input.

Function commands (FN codes)

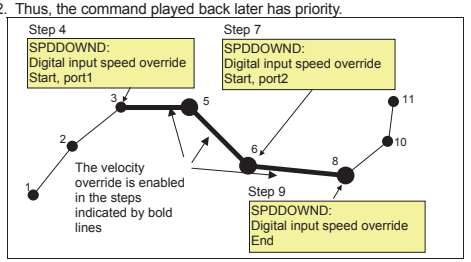
Command name	SPDDOWND
FN code	277
Title name	Digital input speed override
Outline	The playback speed of the robot is changed in accordance with digital input signals

■ **General description**

With this function command, the playback speed of the robot can be changed in accordance with digital input signals. It is necessary to design the velocity override patterns (Constant Setting / Signals / Velocity override depend on input) and allocate input signals beforehand in the "Constant Setting" mode.
Four digital input ports are provided, any of which can be specified.

■ **Example of operation**

In the following diagram, record "SPDDOWND (FN277) / Start" in steps 4 and 7, and "SPDDOWND (FN277) / End" in step 9. When this is played back, the velocity override is performed by digital input signals in the sections indicated by the bold lines in the diagram. As far as step 6 the operation is performed in accordance with the port 1, and as far as step 8 in accordance with the port 2. Thus, the command played back later has priority.



If the rate of velocity override by f keys and R codes is set to 50% and that by input signals set to 50%, the actual rate of velocity override is 25% (50% x 50%).
The velocity override by analog input (FN169) is also available, yet this cannot be used together with the SPDDOWND command.

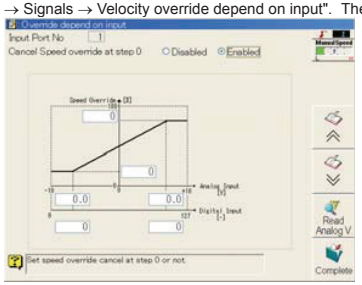
■ **Parameter**

Parameter No. 1	Start/end	Specify whether to start or end the velocity override by input signals. (1: Start, 0: End)
Parameter No. 2	Input port No.	Select the digital input port to refer to from the four ports. It is ignored when "0: End" is selected as the setting of the parameter No.1. (1 to 4)

■ **Example of screen display**

```
SPDDOWND[1,4]       FN277: Digital input speed override
```

The correspondence between digital input signals and speed override must be designed beforehand using "Constant Setting → Signals → Velocity override depend on input". The velocity override by input signal cannot be set over 100%.



In addition, 7 bits per port must be allocated to the digital input signals (Constant Setting → Signals). The speed override is executed by reading out this signal in the binary format and referencing the above table.
If not even one digital input signal has been allocated, the speed override will not function.

See
SPDDOWNA: Analog input speed override (FN169)

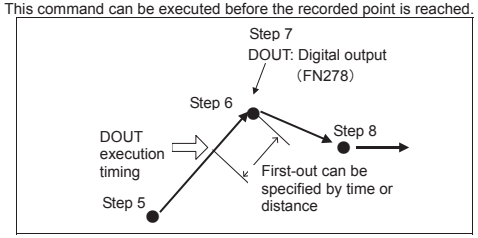
Function commands (FN codes)

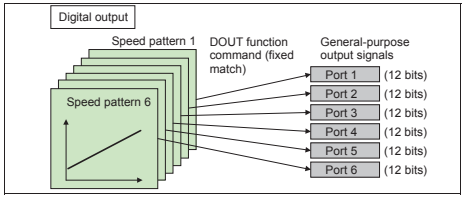
Command name	DOUT
FN code	278
Title name	Digital output
Outline	The TCP (robot tool center point) linear speed and other data are output using general-purpose output signals.

■ General description

When this function command is used, the TCP (robot tool center point) linear speed and other data can be output to an external device as 12-bit digital values (0 to 4095) using general-purpose output signals. They are useful for sealing and other applications.
The TCP speed, a direct specification or OFF can be selected as the output data.
The output signals must be allocated in advance (⇨ Constant Setting/Signals). When the TCP speed has been specified, the TCP speed output data must be designed ahead of time (⇨ Constants/TCP speed data).
One output port is provided for each TCP speed output pattern. (6 ports)

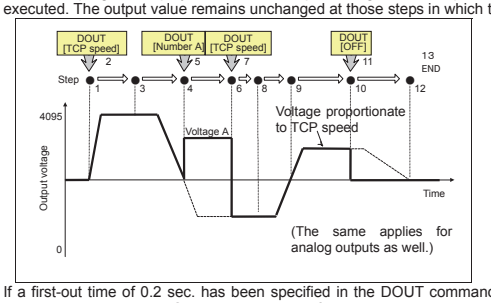
This command can be executed before the recorded point is reached. Specify its amount as a time or distance.





■ Example of operation

Record the DOUT command (FN278) in steps 2, 5, 7 and 11 in the figure shown below.
When the program is played back, the output changes to the specified data each time the DOUT command (FN278) is executed. The output value remains unchanged at those steps in which the DOUT command (FN278) has not been recorded.



If a first-out time of 0.2 sec. has been specified in the DOUT command of step 7 in this example, the output changes from numerical data A to the TCP speed 0.2 sec. before the robot reaches step 6 (the previous movement command).

In the teach mode, any data can be output by performing a manual operation (shortcut R278).
The data which is output last is held even if the mode is switched between teach and playback mode.

■ Parameter

Parameter No. 1	Port number	This is used to specify the number of the output signal port from which data will be digitally output. (1 to 6)
Parameter No. 2	Output signal type	This is used to specify the type of data to be output digitally. (0 to 2) 0: OFF (“0” is output) 1: TCP speed 2: Directly specified
Parameter No. 3	Output data	When output signal type = 0: This recorded data is not used. When output signal type = 1: This recorded data is not used. (With digital outputs, the speed pattern numbers and output signal port numbers correspond on a 1:1 basis.) When output signal type = 2: Directly specify the output data. (0 to 4095)
Parameter No.4	Pre-out type	This enables the output to be started before the recorded point is reached. Select the specification method. (0 to 1) 0: Specified as a time 1: Specified as a distance
Parameter No. 5	Pre-out data	This is used to initiate pre-out using a negative numerical value. When pre-out type = 0: Specify it as a time. (–1.0 to 0 sec.) When pre-out type = 1: Specify it as a distance. (–500 to 0 mm)

■ Example of screen display

```
DOUT[2,1,0,1,-100]      FN278: Digital output
```

See
AOUT: Analog output (FN046)
DPRESETM: Distance specification output preset (FN280)

Function commands (FN codes)

Command name	SGSPRT
FN code	279
Title name	Servo gun separation
General description	Servo gun separated status is changed.

■ General description

Separate the servo gun from the robot.
During servo gun separating, because the servo gun axis is separated from the robot, at the external signal it becomes possible to operate.
At external play stop and emergency stop, it stops also the servo gun axis.

External operation)
Ext. squeeze, Wide open, Small open, Ext. gun search1～3

⚠ CAUTION	During servo gun separating, because the servo gun axis does not operate to record position of the program, step set and program set etc. please keep noting sufficiently.

■ Parameter

Parameter No. 1	1～6	Welder No.
Parameter No. 2	0～1	Separation/Release(1/0)

■ Example of screen display

```
SGSPRT[1,1]          FN279;Servo gun separation
```

■ Status display

During servo gun separating, 

icon is indicated.

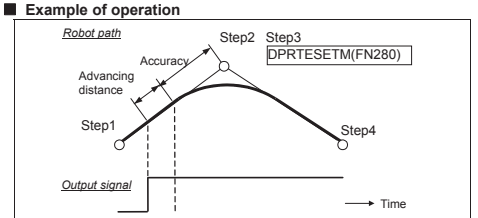
Function commands (FN codes)

Command name	DPRESETM
FN code	280
Title name	Advanced output (distance)
General description	This command is used to set one of the general-purpose output signals with advancing distance

■ General description

When this function command is executed, it is possible to set any one of the general-purpose output signals (O1 to O2048) to ON or OFF. And this enables advanced output designated by distance from the recorded point.
However, the command cannot set any of the status signals (signals with pre-assigned applications such as the gun signals and starting signal) to ON or OFF. Which signals have been assigned as status signals can be identified on the monitor screen. Signals whose numbers are displayed in the bold italics are status signals so any of the other signals can be set to ON or OFF.

■ Example of operation



The robot is not decelerated by this command.

In advanced output, the output can cover plural MOVE steps. Advancing distance is calculated linear length from recorded point, so if MOVE step is recorded by JOINT output point differs from the real moving length of robot TCP.

■ Parameter

Parameter No. 1	Output signal number	This specifies the number of the general-purpose output signal which is to be set to ON or OFF. (1-2048)
Parameter No. 2	ON/OFF	"1" is specified for ON, and "0" for OFF. (0-1)
Parameter No. 3	Output timing	This specifies the output advancing timing that is designated by the linear length from the recorded point. (-500 - 0) [mm]

■ Example of screen display

```
DPRESETM[2048,1,-100]  FN280: Advanced output (distance)
```

See
ALLCLR; All output signals clear (FN0)
SET; Output signal ON (FN32)
RESET; Output signal OFF (FN34)
SETO; Consecutive output signal ON/OFF (FN100)
SETM; Output signal ON/OFF (FN105)
SETMD; Output with pulse or delay ON/OFF (FN35)

Function commands (FN codes)

Command name	WELDGRP
FN code	282
Title name	Weld condition with group
General description	Used to designate the welding condition group number for the welding function (FN119).

■ General description

When this function command is executed, the weld controller number to be used by the welding function (FN119) and the welding condition group number are designated.

■ Example of operation

```
1  REM[TEST]    FN99: Comment
2  WELDGRP[1,2] FN282: Weld condition w
3  100 %    LIN   A1  T1
4  SPOT[1,1,1,0]  FN119: Spot welding
5  100 %    LIN   A4  T1
6  100 %    LIN   A4  T1
7  WELDGRP[1,3] FN282: Weld condition w
8  100 %    LIN   A4  T1
9  SPOT[1,1,1,0]  FN119: Spot welding
10 100 %    LIN   A1  T1
11 END          FN99: End
```

Group No.2 designated in step 2 is enabled until step 6.
Group No.3 is enabled from step 7 to step 11.

Caution of check back
Beware that weld group No. may be different from that of normal playback.
- In case of check-go from step 0
  FN119 of step 4 is executed on the group No.2.
- In case of check back from step 11
  FN119 of step 4 is executed on the group No.3.

Utilize the shortcut R371 that can change group No. manually.

If spot welding command is executed without designating spot weld condition group number,
Alarm “A2063: Weld condition group number is not set.” is detected and robot stops. FN282 must be recorded before FN119.

■ Parameter

Parameter No. 1	Welder number	This specifies the weld controller number. Its maximum number is according to the registered data.
Parameter No. 2	Group number	This specifies the weld condition group number. (1 to 16)

■ Example of screen display

```
WELDGRP[1,1]      FN282: Weld condition w/group
```

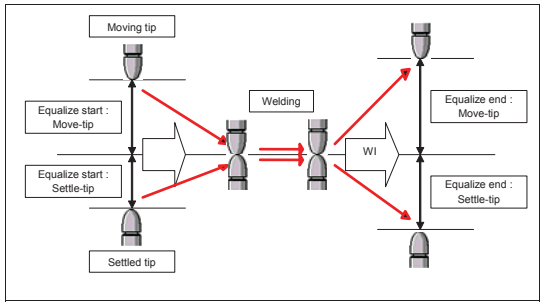
See

Function commands (FN codes)

Command name	EQUALIZE
FN code	287
Title name	Equalize value
General description	This command is used to the equalizing motion as defined by the servo gun.

■ General description

This command is used to the equalizing motion as defined by the servo gun.
1. The equalize start is order position to before the pressing motion.
2. The equalize end is order the open position to after the pressing motion.



Spot welding command and tip dress command is referring to after this command.

A Step zero, EQUALIZECLR; Equalize clear (FN248) is executed to the release of this command.

And, you can confirm current equalize setting by Servo gun data monitor and Equalize confirmation / clear (R376).

■ Parameter

Parameter No. 1	Welder No.	This parameter specifies the welder number. (1–6)
Parameter No. 2	Equalize Start : Move-tip	This is used to specify the equalizing motion as defined by the pressing start position by moving tip. Range: Clearance Limit : move-tip – Large opening end(#1) Initial value: Clearance : move-tip
Parameter No. 3	Equalize Start : Settle-tip	This is used to specify the equalizing motion as defined by the pressing start position by settled tip. Range: Clearance Limit : settle-tip – Large opening end(#1) Initial value: Clearance : settle-tip
Parameter No. 4	Equalize End : Move-tip	This is used to specify the equalizing motion as defined by the open position by moving tip. Range: Clearance Limit : Move-tip – Large opening end(#2) Initial value: Clearance : move-tip
Parameter No. 5	Equalize End : Settle-tip	This is used to specify the equalizing motion as defined by the open position by settled tip. Range: Clearance Limit : settle-tip – Large opening end(#2) Initial value: Clearance : settle-tip

#1［Equalize Start: Move-tip + Equalize Start: Move-tip］value is limited the Large opening end value.
#2［Equalize End: Move-tip + Equalize End: Move-tip］value is limited the Large opening end value.

■ Example of screen display

```
EQUALIZE[1,5,5,20,20]                FN287 ; Equalize value
```

See
SPOT ; Spot welding (FN119)
TIPDRESS ; Tip dress (FN265)
EQUALIZECLR ; Equalize clear(FN248)

Function commands (FN codes)

Command name	DYNCALIBROB
FN code	295
Title name	Robot calibration
Outline	Start/End calibration cycle

■ General description

This function calibrates the robot using the DynalogFMS. This makes it possible to improve the accuracy of a robot program.
IF the [FN296 Measurement point] is executed after starting a calibration cycle, the current position of the TCP will be measured. If the calibration cycle is finished after the measurement of each point, the characteristic of the robot will be calculated automatically and the robot program will be compensated based on the result.

■ Example of operation

When the step where the "DYNCALIBROB: Robot calibration" (FN295) command is recorded is played back, the robot checks the positioning and then waits for the Reply data from Dynalog PC.
When any data is received during the specified timeout time, the standby status is released, and the robot moves to the next step.
If no data is input even after the timeout time has elapsed, the robot jumps to the specified escape step.

■ Parameter

Parameter No. 1	Calibration Start/Finish	When “1” is designated, a calibration cycle will be started. When “0” is designated, a calibration cycle will be finished. (0 –1)
Parameter No. 2	Robot program name	Alphanumeric character (Max 8 characters)
Parameter No. 3	Vision model name	Alphanumeric character (Max 8 characters)
Parameter No. 4	Wait time	This specifies the wait unit in seconds. IF “0” is designated, the robot will forever. (0.0 – 300.0)
Parameter No. 5	Escape destination step	This specifies the number of the step to which the robot is to jump if condition is not satisfied even though the wait time has elapsed. Any step in the same program can be designated. (1 – 9999)

■ Example of screen display

DYNCALIBROB [1,program1,modelA,5,15]	FN295; Robot Calibration

See
DYNMESPOS: Measurement point (FN296)

Function commands (FN codes)

Command name	DYNMESPOS
FN code	296
Title name	Measurement point
Outline	The Dynalog PC to take a measurement of the current position of the robot’s TCP

■ General description

This function calibrates the robot using the DynalogFMS. This makes it possible to improve the accuracy of a robot program.
If this function is executed after starting a calibration cycle [FN295 Robot calibration], the current position of the TCP will be measured. If the calibration cycle is finished [FN295 Robot calibration] after the measurement of each point, the characteristic of the robot will be calculated automatically and the robot program will be compensated based on the result.

■ Example of operation

When the step where the "DYNMESPOS: Measurement point" (FN296) command is recorded is played back, the robot checks the positioning and then waits for the measurement completion data from Dynalog PC.
When measurement completion data is received during the specified timeout time, the standby status is released, and the robot moves to the next step.
If no data is input even after the timeout time has elapsed, the robot jumps to the specified escape step.

■ Parameter

Parameter No. 1	Robot position name	Alphanumeric character (Max 8 characters)
Parameter No. 2	Vision target name	Alphanumeric character (Max 8 characters)
Parameter No. 3	Pin number	This specifies the number of the pin (1 – 9)
Parameter No. 4	Segment name	Alphanumeric character (Max 8 characters)
Parameter No. 5	Wait time	This specifies the wait unit in seconds. IF “0” is designated, the robot will forever. (0.0 – 300.0)
Parameter No. 6	Escape destination step	This specifies the number of the step to which the robot is to jump if condition is not satisfied even though the wait time has elapsed. Any step in the same program can be designated. (1 – 9999)

■ Example of screen display

DYNMESPOS [point1,targetA,1,segmentA,5,10]	FN296; Measurement Point

See
DYNCALIBROB: Robot calibration (FN295)

Function commands (FN codes)

Command name	CHGMEC
FN code	301
Title name	Mechanism change
General description	Connect or disconnect the designated mechanism

■ General description

About mechanism change itself, please refer to description of function CHGGUN; mechanism change (FN95).
CHGGUN (FN95) is dedicated to mechanism 2 only, but on the other hand mechanism can be designated on CHGMEC (FN301).
This point is only difference between CHGGUN (FN95) and CHGMEC (FN301).

■ Example of operation

Please refer to CHGGUN; mechanism change (FN95).

■ Parameter

Parameter No. 1	Mechanism number	Specify the mechanism number to be connected or disconnected. (1-9)
Parameter No. 2	Connect / Disconnect	This is used to specify whether the mechanism is to be connected or disconnected. O; Connect 1; Disconnect

■ Example of screen display

See
CHGGUN; Mechanism Change（designated to mechanism 2 only）(FN95)

Application Command (FN Code)

Command name	CHGXXMEC
FN code	302
Title name	Mechanism change
Outline	Used to select a mechanism to be operated manually.

■ Outline

For mechanism change, refer to information on the Application Command CHGXXGUN: Mechanism change (FN238).
While the CHGXXGUN (FN238) is a command dedicated to Mechanism 2, the CHXXMEC (FN302) can specify arbitrary mechanisms.
Except for that point, this command is the same as the CHGXXGUN (FN238).

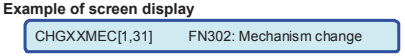
■ Example of motion

For detail, refer to information on the Application Command CHGXXGUN: Mechanism change (FN238).
Setting the Parameter 1 to Mechanism 2 will provide exactly the same motion as that provided by the CHGXXGUN (FN238).

■ Parameters

Parameter 1	Mechanism No.	Used to make setting of mechanism number to perform change operation. (Setting range: 1 to 9)
Parameter 2	Sub-mechanism No.	Used to make setting of sub-mechanism number. (Setting range: 1 to 31)

■ Example of screen display



Related commands
CHGXXGUN: Mechanism change (dedicated to Mechanism 2) (FN238)

Application Command (FN Code)

Command name	SYNCSPOT
FN code	303
Title name	Spot sync welding
Outline	Used to weld with two servo guns at a time.

■ Outline

This is a command to control special servo gun that enables welding at two weld points at a time as one of multi gun system patterns. Since this command enables spot welding at two points at a time, cycle time can be significantly reduced.
There are limits to workpiece shapes that can be welded.
For detail of multi-gun synchronous welding, refer to information in the “Multi-gun synchronous welding” section in the “APPLICATION MANUAL / SPOT WELDING”.

■ Parameters

Parameter 1	Welder No.	Used to makes setting of welder number used for welding. A gun set with this parameter serves as the master for synchronous welding. (Setting range: 0 to 6)
Parameter 2	Weld condition No.	Used to make setting of welding condition number that determines the pressure, welding condition signal, and others of a gun set with the Parameter 1. (Setting range: 1 to 255)
Parameter 3	Weld sequence No.	Used to make setting of welding sequence number that determines timing for outputting the pressurization signal, weld start signal, and stroke signal a gun set with the Parameter 1. (Setting range: 1 to 64)
Parameter 4	Welder No.	Used to makes setting of welder number used for welding. (Setting range: 0 to 6)
Parameter 5	Weld condition No.	Used to make setting of welding condition number that determines the pressure, welding condition signal, and others of a gun set with the Parameter 4. (Setting range: 1 to 255)
Parameter 6	Weld sequence No.	Used to make setting of welding sequence number that determines timing for outputting the pressurization signal, weld start signal, and stroke signal a gun set with the Parameter 4. (Setting range: 1 to 64)
Parameters 7 to 18	Welder No. Weld condition No. Weld sequence No.	Used to make settings of welder number, welding condition number, and welding sequence number for the third and later welders following the setting procedure for the first and second welders.
Parameter 19	Weld point No.	Used to control weld points. If a weld fault occurs, the relevant weld point number will be output, thus making it possible to identify the weld point. To disable this function, set this parameter to “0”. (Setting range: 0 to 16000)

■ Example of screen display

SYNCSPOT[1,1,1,2,1,1 →] FN303: Spot sync welding

Related commands
SPOT: Spot welding (FN119)

Function commands (FN codes)

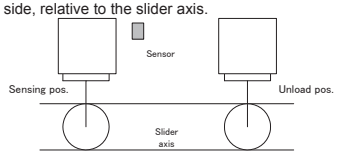
Command name	ALIGMENT
FN code	304
Title name	Alignment
Outline	This command controls the start and end of the alignment function.

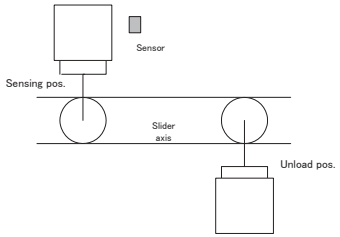
■ General description

This command controls the start and end of the alignment function.
From the Parameter No.3 onward, the values are automatically substituted in when the alignment standard is written.
When the function is recorded, they must be set to zero.

■ Operation example

[Unload direction 1]
In case the sensing position and the unload position are on the same side, relative to the slider axis.

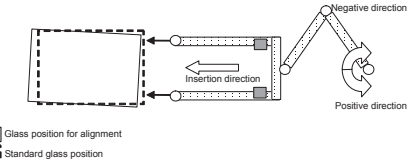




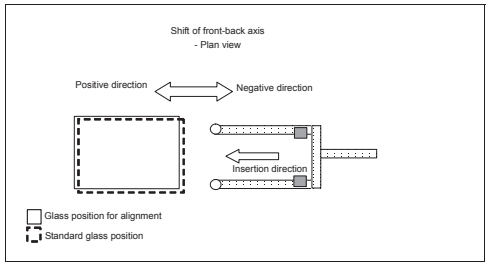
[Unload direction 2]
In case the sensing position and the unload position are on the opposite side, relative to the slider axis.

About rotary alignment shift action (The same applies to both right and left alignment.)
The rotary alignment shift action is used to carry out the shifts (1) to (3) shown below at a time.

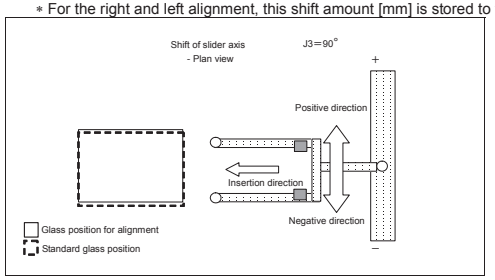
(1) Used to carry out a shift by the amount of misplacement of the glass in the rotation direction (the direction of the J3 axis).
∗ For the right and left alignment, this shift amount [DEG] is stored to the set real variable number.



(2) Used to carry out a shift by the amount of misplacement of the glass in the front and back directions (the direction of the J5 axis for the hand on the right side or that of the J4 axis for that on the left side).
∗ For the right and left alignment, this shift amount [DEG] is stored to the set real variable number.



(3) Used to carry out a shift by the amount of misplacement of the glass in the lateral direction (the direction of the slider axis).
∗ For the right and left alignment, this shift amount [mm] is stored to the number of the set real variable number plus 2.



■ **Parameters**

Parameter No.1	Start/ end	Set 1 to start alignment and 0 to stop it.
Parameter No.2	Shift type	Set the type of shift. 0: Rotary shift 1: Parallel shift, fixed sensor 1, unload direction 1. 2: Parallel shift, fixed sensor 2, unload direction 1. 3: Parallel shift, fixed sensor 1, unload direction 2. 4: Parallel shift, fixed sensor 2, unload direction 2.
Parameter No.3	Set standard value	Indicates whether alignment standard writing has been completed. When the function is set, it must be set to zero.
Parameter No.4	Standard 1	When the function is set, it must be set to zero.
Parameter No.5	Standard 2	When the function is set, it must be set to zero.

■ **Example of screen display**

ALIGNMENT[1,0,0,0,0]	FN304; Alignment

See
ALIGNR : Right alignment (FN320)
ALIGNL : Left alignment (FN321)
ALIGNT : Slider alignment (FN322)

Function commands (FN codes)

Command name	MAPPING
FN code	305
Title name	Mapping
Outline	This command controls the start and end of the mapping function.

■ **General description**
This command controls the start and end of mapping.

■ **Parameters**

Parameter No.1	Start/ end	Set 1 to start mapping and 0 to stop it.
Parameter No.2	Cassette number	Set the cassette number. (1 - 32)

■ **Example of screen display**

MAPPING[1,0,0,0,0]	FN305; Mapping

See

Application Command (FN Code)

Command name	GETTIPCON
FN code	306
Title name	Get tip consumption
Outline	Used to get tip consumption amount.

■ **Outline**

Using this command makes it possible to get the consumption amounts of the moving- and fixed-side gun tips.

■ **Parameters**

Parameter 1	Gun No.	Used to make setting of gun number which you want to detect the tip consumption amount. (Setting range: 1 to 31)
Parameter 2	Moving-side save variable No.	Used to make setting of variable in which the consumption amount of the moving-side tip is saved. (Setting range: 1 to 200)
Parameter 3	Fixed-side save variable No.	Used to make setting of variable in which the consumption amount of the fixed-side tip is saved. (Setting range: 1 to 200)

■ **Example of screen display**

```
GETTIPCON[1,V1!,V1!]        FN306: Get tip consumption
```

Related commands

Short-cut (R code)

R code	307
Title name	Press brake shelter
Outline	This executes retreat actions after work process, in the press brake synchronization function.

■ **Outline**

When this function is used, the tool tip coordinates of the first manipulator registered in the unit are taken into real variables. The robot moves to the position with addition of the "retreat distance" of the press brake data to the coordinates.

At movement, it retreats at the speed designated by the "retreat distance" of the press brake data.

■ **Action example**

Parameter No.1	Conveyor number	This designates a conveyor number registered as press brake. (1-2)
Parameter No.2	Real variable number	This designates the start number of the real variables to store the current robot position values (X, Y, Z, R, P, Y). (1-195) The position values are stored to 6 real variables sequentially from the designated real variables. V![Start variable number] = X V![Start variable number+1] = Y V![Start variable number+2] = Z V![Start variable number+3] = R V![Start variable number+4] = P V![Start variable number+5] = Y

■ **Screen display example**

```
PRSS[1,V1!]                 FN307; Press brake shelter
```

Reference

PRSD; Read press data (FN308)

Function commands (FN codes)

Command name	PRSD
FN code	308
Title name	Read press data
Outline	This reads a press brake synchronization setting file, in the press brake synchronization function.

■ **Outline**

When this function command is executed, the synchronization data registered in PRESS.CON file is read.

■ **Action example**

Parameter No.1	Conveyor number	This designates a conveyor number registered as press brake. (1-2)
Parameter No.2	Press data number	This designates a press condition number to be read. (1-10)

■ **Screen display example**

PRSD[1,10] FN308; Read press data

Reference

PRSS; Press brake shelter (FN307)

Function commands (FN codes)

Command name	SETVELO
FN code	309
Title name	Set velocity
General description	This command is used to set the speed in an endless axis.

■ **General description**

Only one-axis mechanism can use the function. And it must be set an endless axis. Allowed endless type is an speed control type or change control type.
The rotation speed of an endless axis can be freely changed between arbitrary record points.
When this function is used for a change control type endless axis, it is necessary to change the velocity control by the CHGENDLESS; Change endless control (FN373).

■ **Parameter**

First parameter	Explanation
Mechanism No.	This specifies the mechanism number of an endless axis. (1-9)

Second parameter	Explanation
Velocity	This specifies the number of rotations of deceleration machine output sides. Maximum speed is different depending on the specification of the axis.

■ **Example of screen display**

SETVELO [2,500] FN309; Set velocity

See

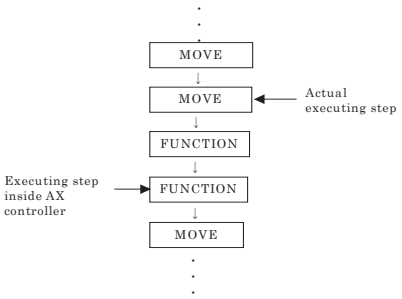
CHGENDLESS; Change endless control(FN373)

Function commands (FN codes)

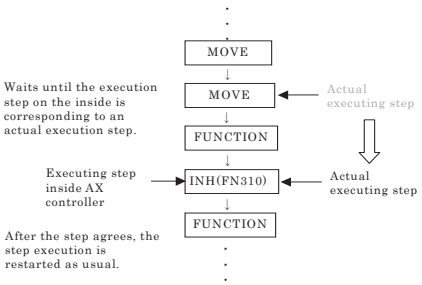
Command name	INH
FN code	310
Title name	INHIBIT
General description	Execution is controlled ahead.

■ General description

This function code is used combine with the input functions such as INP, INPB, GETSIG, and GETSIGB, of the robot language and the signal acquisition timing may be matched to actual step execution.
In the AX controller, the step execution processing is done in AX to make the robot work smoothly, and the robot works follows to it.
Therefore, a part of application instruction is executed earlier than the displays executing. (execution step ahead)



When this function is executed waits until execution is controlled the step ahead, and the execution step in AX is corresponding to an actual execution step.
Afterwards, the step execution is restarted as usual.



■ Parameter

None

■ Example of screen display

```
INH                FN310: Inhibit
```

Function commands(FN codes)

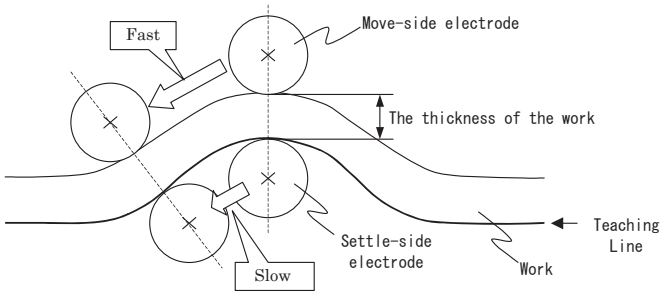
Command name	SEAMTHICK
FN code	311
Title name	Seam panel thick
General description	Set the thickness of the work.

■ General description

When performing seam welding on 3-dimensional space, a difference appears in the welding speed of move-side electrode and settle-side electrode by the thickness of a work. According to this speed difference, the rotation speed of electrode can be adjusted automatically, and welding by the optimal electrode rotation can be performed. In this command, the thickness of the work used as the basement of the amount of compensation is specified.

■ Example of operation

When welding the work of 3-dimensional form as shown in the following figure by setting work board thickness by this command, according to board thickness, the rotation speed of electrode can be changed automatically, and it can weld at the electrode rotation speed optimal at a work contact surface.



■ Parameter

Parameter No.1	-100.0 - 100.0	Panel thickness[mm] Set the thickness of the work.

■ Example of screen display

```
SEAMTHICK[0.5]          FN311: Seam panel thick
```

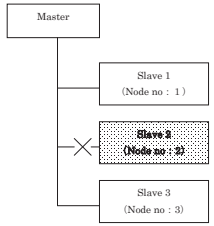
See
SEAMST Seam weld start (FN245)

Function commands (FN codes)

Command name	FBUSREL
FN code	312
Title name	Field bus release
General description	In the field bus master, error detection Enabled/Disabled of the specified node is switched.

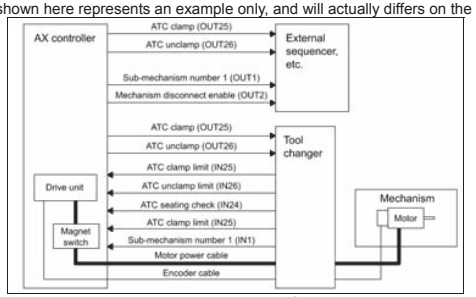
■ General description

In the field bus function, when AX is used as master, the error detection is done in each slave connected with it. This function is used to invalidate the error detection of a specific slave temporarily because of the mechanism change etc.
Enabled/Disabled of the error detection is switched specifying the slave node of each field bus master by using this function.
In the example of the figure below, the example of invalidating the error detection of slave 2 which leads to mastering is shown by using this application instruction. When slave 1 occurs the error, mastering detects the error in the figure below. However, even if slave 2 occurs the error, mastering does not detect the error.



■ Example of operation

An example is given for tool changing system where the signals have been allocated as shown in the figure below. The method shown here represents an example only, and will actually differs on the environment that depends on the customer's facility.



Connection with tool changer (sample)

STEP N	**500mm/s LIN A1 T31**		
STEP N+1	**100mm/s LIN A1 T1**		Move to the connect/disconnect position
STEP N+2	**FBUSREL[1,1,0]**	FN312;Fieldbus release	Error detection Disabled.
STEP N+3	**CHGGUN[0]**	FN95; Mechanism change	Electrical disconnection
STEP N+4	**RESET[025]**	FN34; Output OFF	Unclamp
STEP N+5	**SET[026]**	FN32; Output ON	
STEP N+6	**WAIT[I26]**	FN525; Wait input (positive)	Waiting for unclamp finish
STEP N+7	**100mm/s LIN A1 T31**		Move to the connect/disconnect position
STEP N+8	**DELAY[0.5]**	FN50; Delay	
STEP N+9	**WAIT[I24]**	FN525; Wait input (positive)	Waiting for seating
STEP N+10	**RESET[026]**	FN34; Output OFF	Clamp
STEP N+11	**SET[025]**	FN32; Output ON	
STEP N+12	**WAIT[I25]**	FN525; Wait input (positive)	Waiting for clamp finish
STEP N+13	**CHGGUN[1]**	FN95; Mechanism change	Electrical connection
STEP N+14	**100mm/s LIN A1 T1**		Move away
STEP N+15	**FBUSREL[1,1,1]**	FN312;Fieldbus release	Error detection enabled

The error does not occur even if it is a field bus disengaging (error detection invalidity) and the specified slave node is physically cut off in step N+2 before the disengaging of the mechanism.
It is a field bus disengaging (effective the error detection) in STEP N+15 after enough time passes until the mechanism is connected in step N+13 and the link of the field buses establishes, and the error detection of the field bus is restarted.

< Note >
It is necessary to see enough time until the link establishes before executing the field bus disengaging (effective the error detection) after reconnecting the field bus. Error "E0960 A part or all I/O links are stopping." is generated when becoming the error detection effective before the field bus link is established.
Time until the link establishes after reconnecting the field bus is different according to the system environment etc. which the customer uses.

When field bus master is connected with the field bus slave by one-to-one, one example of the average time until the link establishes is shown as follows.

Protocol type	Average time until the link establishes
Device net	8-10[sec]
Device net (Quick Connect Enable)	2[sec]
CC-Link	2[sec]
JEMANET	1[sec]
PROFIBUS	2[sec]

■ Parameter

Parameter No. 1	Channel No.	Channel number which the field bus master is using is specified.(1-4)
Parameter No. 2	Slave Node No.	The slave node number of the object of the error detection is specified. (0-127)
Parameter No. 3	Error detection Enabled/Disabled	Enabled/Disabled of the error detection is specified. (0=Disabled/1=Enabled)

■ Example of screen display

```
FBUSREL[1,3,0]                FN312:Fieldbus release
```

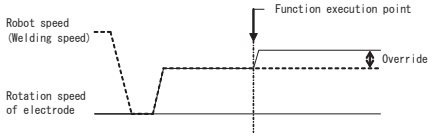
Function commands(FN codes)

Command name	SEAMOV
FN code	313
Title name	Seam override
General description	Set the electrode rotation speed override.

■ General description

This function overrides the rotational speed of the electrode to the welding speed according to the correction rates specified by the parameter.
The override values are cleared when step 0 of the unit that contains the electrode mechanism executions.
Note)
The rotational speed is not corrected in the welding off.

■ Example of operation



If this function is executed before welding, the override values which are specified by this function are applied at time when the welding is begun.
When this function is executed again during welding at the override speed, the rotational speed of the electrodes is changed at the override speed newly specified.

■ Parameter

Parameter No.1	-20-20	Override (Move) [%]
Parameter No.2	-20-20	Override (Settle) [%]

■ Example of screen display

```
SEAMOV[6,6]                FN313:Seam override
```

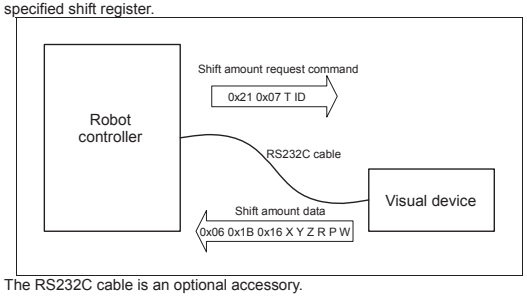
See
SEAMST: Seam weld start (FN245)
SEAMEND: Seam weld end (FN246)

Function commands (FN codes)

Command name	SREQ2
FN code	315
Title name	Shift amount request (binary)
Outline	The command requests the shift amount data (binary) from the external device using the serial port. Once it has been input from the external device, the shift amount data is stored in the specified shift register.

■ General description

This command sends binary code data (command) requesting a shift amount to the external device connected to the controller by the RS232C cable, and it sets the shift amount binary data, which is input as the response from the external device, into the specified shift register.



The RS232C cable is an optional accessory.

When this function command is executed, following binary data is output from the RS232C port, and the shift amount data is thereby requested.

0x21 0x07 T ID

Here, T and ID is as followed.

T = time out (4 bytes integer [msec])
Designated time value is transmitted from lower byte to upper byte.
ID = ID of virtual frame (1 byte integer)

The robot continues to operate even after the request data has been output.
When the shift amount data is input from the external device in the following format, it is stored in the specified shift register.

0x06 0x1B 0x16 X Y Z R P W

Here, from X to W is following binary data received from lower byte to upper byte. XYZ direction is right hand coordinate system.

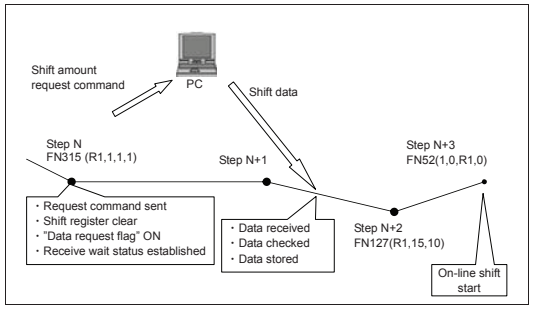
X = X direction shift amount (4 byte floating real value [mm])
Y = Y direction shift amount (4 byte floating real value [mm])
Z = Z direction shift amount (4 byte floating real value [mm])
R = Rotating amount on Z axis (4 byte floating real value [degree])
P = Rotating amount on Y axis (4 byte floating real value [degree])
W = Rotating amount on X axis (4 byte floating real value [degree])

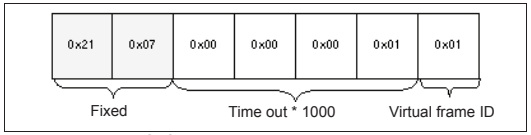
CAUTION	**Shift request command, time out shift data must be manages by little-endian control (sending from lower byte to upper byte). When treating these data in personal computer, beware to follow the little-endian control.**

The receive time, wait time and communication conditions are set by selecting "8 Communication" from "Constant Setting" and then "1 Serial port." The data input wait time can be set in "Timeout time," and if this is set to "0," the serial port will remain in the wait status until an input arrives. (The robot continues to operate.) If this function is again executed while waiting for the previous receive, next request command is not sent until receiving is completed. But time out is set to “0” and FN111(RSCLR) is executed, waiting status is abort.

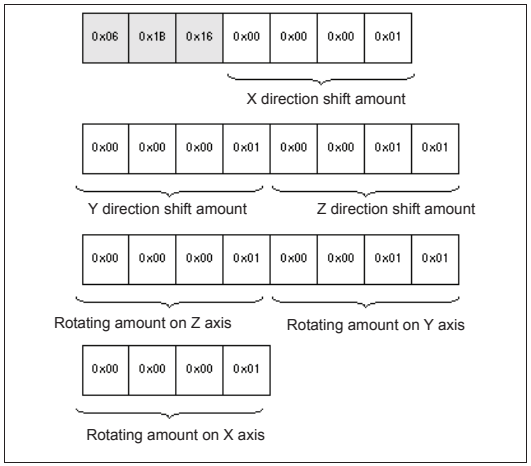
■ Example of operation

As shown in the figure, when SREQ2 (FN315) is executed, the shift amount request command is sent to the external device through the serial port, and it is ensured that the data can be received at any time. The robot continues to operate as is and heads toward step N+1. When the shift data is received from the external device at any position, the legitimacy of the data is checked, and the data is stored in the specified shift register only when the data is found to be legitimate.
But received shift data is committed after executing the function “WAITR: Wait shift value receive” (FN127) which is programmed at step N+2.
This shift register value is used for the shift operations by the "SHIFTR: Shift2" (FN52) or such other command.





Shift amount request command



Response data

If the response data from the external device is sent in any other format, the values will not be set correctly in the shift register.

■ **Parameter**

Parameter No. 1	Shift register number	This is used to specify the number of the shift register in which to store the shift amount received from the external device. (1 to 9)
Parameter No. 2	Port number	This is used to specify the number of the port to be used to transfer the data. At the present time, only port 1 can be used. (1 to 1)
Parameter No. 3	Time out	This is used to specify the waiting time. (0.0 to 3600.0 sec) This is sent as the time out data of sending command to the vision sensor.
Parameter No. 4	Virtual frame ID	This is used to specify the virtual frame ID of the vision sensor. (0 to 255)

■ **Example of screen display**

SREQ2[R1,1,1.5,1] FN315: Binary Shift data

See
SHIFTR: Shift2 (FN52)
RSCLR: Buffer clear (FN111)
WAITR: Wait shift value receive (FN127)
CLRREGWR: Clear register of written sts (FN699)

Application Command (FN Code)

Command name	SYNCSPOTIWB
FN code	316
Title name	Spot sync welding
Outline	Used to weld with two servo guns at a time. （dedicated to welding I/F＝MEDbus only）

■ **Outline**

This function is dedicated to the MEDAR spot welder (IWB timer controller) interface application only. **(welding I/F＝MEDbus)**
This is a command to control special servo gun that enables welding at two weld points at a time as one of multi gun system patterns. Since this command enables spot welding at two points at a time, cycle time can be significantly reduced.
There are limits to workpiece shapes that can be welded.
For detail of multi-gun synchronous welding, refer to information in the "Multi-gun synchronous welding" section in the "APPLICATION MANUAL / SPOT WELDING".

■ **Parameters**

Parameter 1	Welder No.	Used to makes setting of welder number used for welding. A gun set with this parameter serves as the master for synchronous welding. (Setting range: 0 to 6)
Parameter 2	Weld sequence No.	Used to make setting of welding sequence number with the Parameter 1. (Setting range: 1 to 255)
Parameter 3	Welder No.	Used to makes setting of welder number used for welding. (Setting range: 0 to 6)
Parameter 4	Weld sequence No.	Used to make setting of welding sequence number with the Parameter 3. (Setting range: 1 to 255)
Parameters 5 to 12	Welder No. Weld sequence No.	Used to make settings of welder number and welding sequence number for the third and later welders following the setting procedure for the first and second welders.
Parameter 13	Weld point No.	Used to control weld points. If a weld fault occurs, the relevant weld point number will be output, thus making it possible to identify the weld point. To disable this function, set this parameter to "0". (Setting range: 0 to 16000)

■ **Example of screen display**

SYNCSPOTIWB[1,1,2,1 →] FN316: Spot sync welding

Related commands
SPOTIWB1: Spot welding (FN199)

Function commands (FN codes)

Command name	AUTOZERO
FN code	319
Title name	Analog input auto zero set
Outline	Auto zero the analog input signal is executed.

■ General description

The analog signal input is prepared two channels for the servo amplifier. The analog input signal has the unavoidable offset value when it is 0V. The measurement accuracy of the analog input signal decreases if the influence of the offset is not removed.
This function executes the autozero of the analog input, and removes the influence of the offset.

■ Example of operation

This function is used as follows.

```
SETM   O4,1    Digital output port 4 resets the analog signal
DELAY 0.5      wait until analog input becomes 0.
SETM   O4,0
AUTOZERO       Execute auto zero
```

We recommend that this function is executed at the beginning of the program when the analog input signal is used.

For the convenience, the above-mentioned functions is used as the user macro.

Please execute this function when the analog input signal is always 0V. The measurement accuracy of the analog input signal decreases when executing it, except when the analog input signal is 0V.

■ Parameter

None.

■ Example of screen display

```
AUTOZERO          FN319: Analog input auto zero set
```

Function commands (FN codes)

Command name	ALIGNR
FN code	320
Title name	Right Alignment
Outline	This command controls the start and end of the right alignment function.

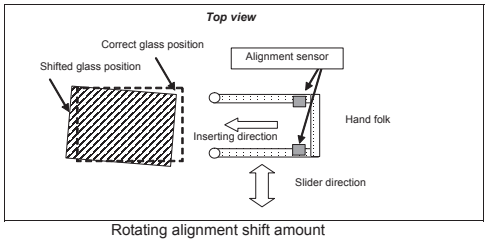
■ General description

Glass panel transferring allocation only.
This command controls the start and end of the right alignment function.
Right alignment is performed by the right arm seen from the robot rear.
From the Parameter No.3 onward, the values are automatically substituted in when the alignment standard is written.
When the function is recorded, they must be set to zero.

■ Operation example

Alignment shift is carried out in two steps. One shift is glass rotating direction and another one is slider direction.
Alignment shift is carried out according to the deviation compared with the base position. So at first, base position registration is necessary using the master work (glass).

Alignment shift amount is stored into the designated global real variables on the order of J3 axis shift amount [DEG] -> front/back axis (J5) shift amount [mm] -> slider shift amount [mm].
For example, when real variable No.100 is designated, each data is stored as followed.
V![100] is occupied by J3 axis shift amount [DEG]
V![101] is occupied by front/back axis (J5) shift amount [mm]
V![102] is occupied by slider shift amount [mm].
So free real variables is maximum 198.



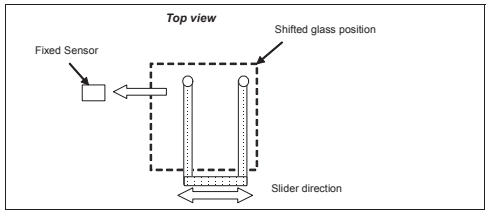
Rotating alignment shift amount

3 shift amount calculated by alignment are stored into the real variables designated by argument.

n	designated real variables number
V![n]	Rotating (J3) axis shit amount (plus/minas is based on monitor data)
V![n+1]	Front/back (J5) axis shit amount (plus/minas is based on monitor data)
V![n+2]	Slider shit amount (plus/minas is based on monitor data)

<Rotating shift when extracting the glass>
Robot program sample

```
1    100% JOINT AT T1
            ↑ Alignment start
2    ALIGNR[1, 1, 0, 0, 0]      ;FN320
3    100% JOINT AT T1
            ↑ Move gripper under the glass (Getting input from alignment sensor, but shift is never carried out)
4    100% JOINT AT T1
            ↑ Sane position as step 3 (shift is carried out)
5    20% JOINT AT T1
            ↑ Move with shifting
6    SET[O1]           ;FN32
            ↑ Vacuum ON and alignment is completed
7    ALIGNR[0, 1, 0, 0, 0]     ;FN320
8    20% JOINT AT T1
            ↑ Lift up the glass (back to the taught position because no shift)
9    WAITI[I3]                 ;FN525
            ↑ Waiting for vacuum complete input I3
10   100% JOINT AT T1
            ↑ Hand returns
```



■ **Parameters**

Parameter No.1	Start/ end	Set 1 to start alignment and 0 to stop it.
Parameter No.2	Real variable number	Set the real variable number to store the shift amount. (1 to 198)
Parameter No.3	Set standard value	Indicates whether alignment standard writing has been completed. When recording, this must be set to “0”.
Parameter No.4	Standard 1	When recording, this must be set to “0”.
Parameter No.5	Standard 2	When recording, this must be set to “0”.

■ **Example of screen display**

ALIGNMENR[1,0,0,0,0]	FN320; Right alignment

See
ALIGNMENT : Alignment (FN304)
ALIGNL : Left alignment (FN321)
ALIGNT : Slider alignment (FN322)

Function commands (FN codes)

Command name	ALIGNL
FN code	321
Title name	Left Alignment
Outline	This command controls the start and end of the left alignment function.

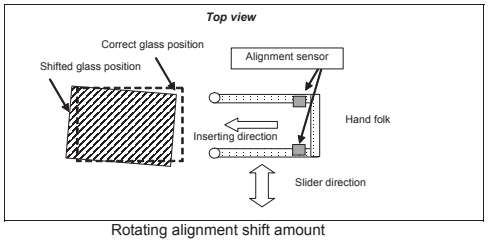
■ **General description**

Glass panel transferring allocation only.
This command controls the start and end of the left alignment function.
Left alignment is performed by the left arm seen from the robot rear.
From the Parameter No.3 onward, the values are automatically substituted in when the alignment standard is written.
When the function is recorded, they must be set to zero.

■ **Operation example**

Alignment shift is carried out in two steps. One shift is glass rotating direction and another one is slider direction.
Alignment shift is carried out according to the deviation compared with the base position. So at first, base position registration is necessary using the master work (glass).

Alignment shift amount is stored into the designated global real variables on the order of J3 axis shift amount [DEG] -> front/back axis (J5) shift amount [mm] -> slider shift amount [mm].
For example, when real variable No.100 is designated, each data is stored as followed.
V![100] is occupied by J3 axis shift amount [DEG]
V![101] is occupied by front/back axis (J5) shift amount [mm]
V![102] is occupied by slider shift amount [mm].
So free real variables is maximum 198.

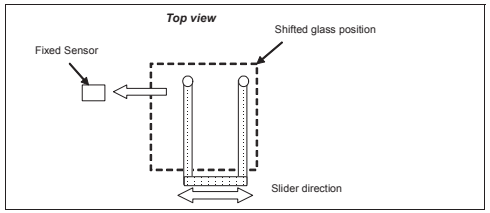


Rotating alignment shift amount

3 shift amount calculated by alignment are stored into the real variables designated by argument.
n designated real variables number
V![n] Rotating (J3) axis shit amount (plus/minas is based on monitor data)
V![n+1] Front/back (J4) axis shit amount (plus/minas is based on monitor data)
V![n+2] Slider shit amount (plus/minas is based on monitor data)

<Rotating shift when extracting the glass>
Robot program sample

```
1    100% JOINT AT T1
            ↑ Alignment start
2    ALIGNL[1, 1, 0, 0, 0]      ;FN321
3    100% JOINT AT T1
            ↑ Move gripper under the glass (Getting input from alignment sensor, but shift is never carried out)
4    100% JOINT AT T1
            ↑ Sane position as step 3 (shift is carried out)
5    20% JOINT AT T1
            ↑ Move with shifting
6    SET[O1]          ;FN32
            ↑ Vacuum ON and alignment is completed
7    ALIGNL[0, 1, 0, 0, 0]     ;FN321
8    20% JOINT AT T1
            ↑ Lift up the glass (back to the taught position because no shift)
9    WAITI[I3]                 ;FN525
            ↑ Waiting for vacuum complete input I3
10   100% JOINT AT T1
            ↑ Hand returns
```



■ **Parameters**

Parameter No.1	Start/ end	Set 1 to start alignment and 0 to stop it.
Parameter No.2	Real variable number	Set the real variable number to store the shift amount. (1 to 198)
Parameter No.3	Set standard value	Indicates whether alignment standard writing has been completed. When recording, this must be set to “0”.
Parameter No.4	Standard 1	When recording, this must be set to “0”.
Parameter No.5	Standard 2	When recording, this must be set to “0”.

■ **Example of screen display**

ALIGNMENL[1,0,0,0,0]	FN321; Left alignment

See
ALIGNMENT : Alignment (FN304)
ALIGNR : Right alignment (FN320)
ALIGNT : Slider alignment (FN322)

Function commands (FN codes)

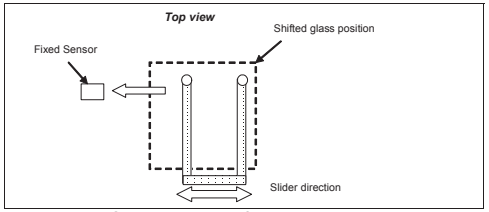
Command name	ALIGNT
FN code	322
Title name	Slider Alignment
Outline	This command controls the start and end of the slider alignment function.

■ **General description**

Glass panel transferring allocation only.
This command controls the start and end of the slider alignment function.
Only fixed sensor 1 is used. Fixed sensor 2 is not used.
From the Parameter No.3 onward, the values are automatically substituted in when the alignment standard is written.
When the function is recorded, they must be set to zero.
(If the direction of sensing position and unloading position is different, shift amount must to be used after multiplying -1.)

■ **Operation example**

Alignment shift is carried out according to the deviation compared with the base position. So at first, base position registration is necessary using the master work (glass).
Real variables are available 1 to 200.

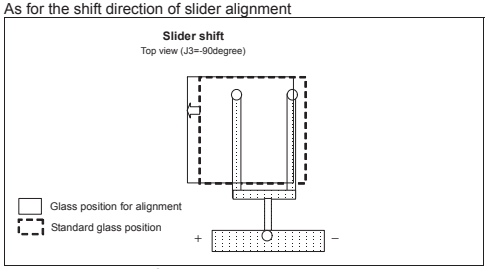


Slider alignment shift amount

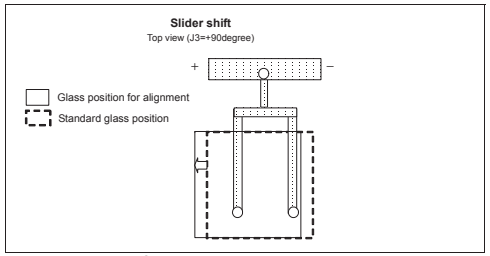
Robot program sample

```
1    100% JOINT AT T1
          ↑ Alignment start
2    ALIGNT[1, 1, 0, 0, 0]     ;FN322
3    100% JOINT AT T1
          ↑ Path through the fixed sensor position (Getting input from alignment sensor, but shift is never carried out)
4    100% JOINT AT T1
          ↑ Shifting is carried out from here
5    20% JOINT AT T1
 ・・
X    100% JOINT AT T1
          ↑ Alignment end (next step is not shifted)
10   ALIGNT[0, 1, 0, 0, 0]     ;FN322
```

As for the shift direction of slider alignment



In this case, minus shift amount is stored into the real variable.
When the glass must be loaded on the cassette at J3=-90 degree side (direction of sensing position and unloading position is same), slider axis is shifted by the shift amount as it was.
When the glass must be loaded on the cassette at J3=+90 degree side (direction of sensing position and unloading position is different), slider axis is shifted by the shift amount after changing the sign of shift amount programmed by robot language.



In this case, plus shift amount is stored into the real variable.
When the glass must be loaded on the cassette at J3=+90 degree side (direction of sensing position and unloading position is same), slider axis is shifted by the shift amount as it was.
When the glass must be loaded on the cassette at J3=-90 degree side (direction of sensing position and unloading position is different), slider axis is shifted by the shift amount after changing the sign of shift amount programmed by robot language.

■ **Parameters**

Parameter No.1	Start/ end	Set 1 to start alignment and 0 to stop it.
Parameter No.2	Real variable number	Set the real variable number to store the shift amount. (1 to 200)
Parameter No.3	Set standard value	Indicates whether alignment standard writing has been completed. When recording, this must be set to “0”.
Parameter No.4	Standard 1	When recording, this must be set to “0”.
Parameter No.5	Standard 2	When recording, this must be set to “0”.

■ **Example of screen display**

```
ALIGNMENT[1,0,0,0,0]          FN322; Slider alignment
```

See
ALIGNMENT : Alignment (FN304)
ALIGNR : Right alignment (FN320)
ALIGNL : Left alignment (FN321)

Function commands(FN codes)

Command name	WELDCND
FN code	33
Title name	Spot condition output
General description	Output spot condition signals on manual.

■ **General description**
This function outputs signals assigned to the "weld condition output." The signals are outputted by the binary of 0-65535.
Notes -- this function is not concerned with a setup of "Check with function " of "Teach/Playback condition", but is performed. However, it does not execute at the time of step back operation.

■ **Example of operation**
Please refer to SETM (FN32) about the timing of outputs.

■ **Parameter**

Parameter No.1	1～6	Welder number
Parameter No.2	0～65535	Weld condition data

■ **Example of screen display**

```
WELDCND[1,1]                  FN33: Spot condition output
```

See
SEAMST Seam weld start (FN245)

Function commands (FN codes)

Command name	VRESET
FN code	330
Title name	Vision reset
General description	Data of the vision sensor is cleared.

■ General description
This instruction used to clear data of the vision sensor built into AX controller.
In general, it is necessary to execute this instruction before measuring.

Operations and parameters are different from which vision sensor uses.

[Vision sensor type1]

■ Example of operation
This instruction can clear the measurement data of the vision sensor.
If it is not possible to communicate with the vision sensor, the error is detected.

■ Parameter

None

■ Example of screen display

```
VRESET          FN330:Vision reset
```

[Vision sensor type2]

■ Example of operation
You can clear the measure result of vision. This function will be done at before actual function execution timing (pre-execution).

■ Parameter

Parameter No.1	0～999	Measure condition No.

If you set 0, all measure results are cleared.

■ Example of screen display

```
VRESET[1]       FN330: FN330:Vision reset
```

See
VSTART: Vision start (FN331)
VWORK: Vision discrimination (FN332)
VSHIFT: Vision shift (FN333)
VDATA: Vision data (FN334)
VLOCCVT: Vision location convert (FN342)
VWAIT: Vision wait (FN343)

Function commands (FN codes)

Command name	VSTART
FN code	331
Title name	Vision start
General description	Start the measurement of vision sensor.

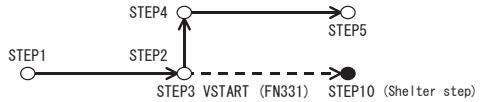
■ General description
[High-speed synchro vision is unavailable]
When this instruction is executed, the AX controller request to start the measurement of the vision sensor.
The next step is not executed until the measurement is completed.
Jumps to the shelter step specified by the parameter the vision sensor notifies abnormality.

[High-speed synchro vision is available]
When this instruction is executed, the AX controller request to start the measurement of the vision sensor.
The next step is executed without waiting to complete the measurement.
VWAIT(FN343) is used to wait to complete the measurement

Operations and parameters are different from which vision sensor uses.

[Vision sensor type1]

■ Example of operation



[High-speed synchro vision is unavailable]
After checking position, this function is executed.
The step executed based as follows on the completion notification received from the vision sensor is decided. When it is normal completion, the following STEP4 is executed. When it is abnormal completion of being not able the measurement etc., jumps to the shelter step
Because the operation moved to the shelter step is done when abnormality occurs when measuring with the vision sensor, Robot error is not detected.
Please turn on the output signal by the routine in the shelter step when it is necessary to detect error on the outside. It is necessary to confirm an error content on the display of the vision sensor.

[High-speed synchro vision is available]
This function is executed without checking position.The shelter step specified by the parameter is ignored. Please use VWAIT(FN343) when you want to jump to the shelter step when failure occurs when vision measuring.

■ Parameter

Parameter No.1	1-256	Measurement No.
Parameter No.2	1-9999	Shelter step No.

■ Example of screen display

```
VSTART[1,10]    FN331:Vision start
```

[Vision sensor type2]

■ Example of operation

Case: Disabled hardware trigger
The function will be done after robot position will be at the step.
Next target step will be judged after receiving the completion note from vision sensor.
Robot moves to the next step if the note from vision sensor is complete.
Robot moves to the shelter step if note from vision sensor is abnormal, for example, vision couldn't measure for some reason.
You should check what kind of error happened by the logging data of vision monitor.

Case: Enabled hardware trigger
This function is optional of the vision sensor. Set appropriate signal number to the 3rd parameter(Trigger No.) when measuring with the robot moving. Please wait the measurement completion with VWAIT(FN343) when you use the trigger signal. The shelter step specified by the parameter in this function is disregarded, and the shelter step of VWAIT(FN343) is used.

■ Parameter

Parameter No.1	1～999	Measure condition No.
Parameter No.2	0～16	Image Proc. No. / Measure point No.
Parameter No.3	0～3	Trigger No.
Parameter No.4	1～10000	Shelter step No.

Robot will stop with recognition error if you set [10000] for shelter step.

■ Example of screen display

```
VSTART[1,10,0,10000]      FN331: Vision start
```

See
VRESET: Vision reset (FN330)
VWORK: Vision discrimination (FN332)
VSHIFT: Vision shift (FN333)
VDATA: Vision data (FN334)
VLOCCVT: Vision location convert (FN342)
VWAIT: Vision wait (FN343)

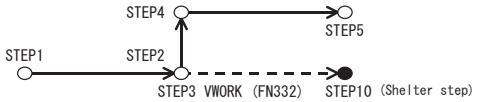
Function commands (FN codes)

Command name	VWORK
FN code	332
Title name	Vision work
General description	Distinguish work with the vision sensor

■ General description

When this instruction is executed, the AX controller request to start the distinction of the work with the vision sensor.
The next step is not executed until the distinction is completed.
Jumps to the shelter step specified by the parameter the vision sensor notifies abnormality.

■ Example of operation



After checking position, this function is executed.
The step executed based as follows on the completion notification received from the vision sensor is decided. When it is normal completion, the following STEP4 is executed. When it is abnormal completion of being not able the measurement etc., jumps to the shelter step
Because the operation moved to the shelter step is done when abnormality occurs when measuring with the vision sensor, Robot error is not detected.
Please turn on the output signal by the routine in the shelter step when it is necessary to detect error on the outside. It is necessary to confirm an error content on the display of the vision sensor.

■ Parameter

Parameter No.1	1-256	Start Measure No.
Parameter No.2	1-256	End Measure No.
Parameter No.3	1-9999	Shelter step No.

■ Example of screen display

```
VWORK[1,5,10]      FN332:Vision work
```

See
VRESET: Vision reset (FN330)
VSTART: Vision start (FN331)
VSHIFT: Vision shift (FN333)
VDATA: Vision data (FN334)

Function commands (FN codes)

Command name	VSHIFT
FN code	333
Title name	Vision shift
General description	Get the shift value from the vision sensor

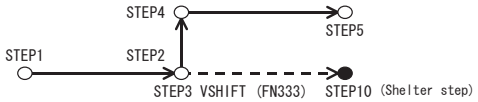
■ **General description**

The shift values are acquired from the last measurement result of the vision sensor when this function is executed, and stores the shift values in the shift register specified by the parameter.
The next step is not executed until the function is completed.
Jumps to the shelter step specified by the parameter the vision sensor notifies abnormality.

Operations and parameters are different from which vision sensor uses.

[Vision sensor type1]

■ **Example of operation**



[High-speed synchro vision is unavailable]
After checking position, this function is executed.
The step executed based as follows on the completion notification received from the vision sensor is decided. When it is normal completion, the following STEP4 is executed. When it is abnormal completion of being not able the measurement etc., jumps to the shelter step
Because the operation moved to the shelter step is done when abnormality occurs when measuring with the vision sensor, Robot error is not detected.
Please turn on the output signal by the routine in the shelter step when it is necessary to detect error on the outside. It is necessary to confirm an error content on the display of the vision sensor.

[High-speed synchro vision is available]
This function is executed without checking position. The following STEP4 is executed when the calculation of the shift amount succeeds, but it jumps to the shelter step when failure occurs.
Please turn on the output signal by the routine in the shelter step when it is necessary to detect error on the outside. It is necessary to confirm an error content on the display of the vision sensor.

■ **Parameter**

Parameter No.1	1-9	Shift register No.
Parameter No.2	1-9999	Shelter step No.

■ **Example of screen display**

```
VSHIFT[1,10]     FN333:Vision shift
```

[Vision sensor type2]

■ **Example of operation**

Robot moves to the next step if the note from vision sensor is complete.
Robot moves to the shelter step if note from vision sensor is abnormal, for example, vision couldn't measure for some reason.
You should check what kind of error happened by the logging data of vision monitor.

■ **Parameter**

Parameter No.1	1～999	Measure Condition No.
Parameter No.2	0～16	Detect No.
Parameter No.3	1～9	Shift register No.
Parameter No.4	1～3	Base coordinate
Parameter No.5	1～10000	Shelter step No.

Robot will stop with recognition error if you set [10000] for shelter step.

■ **Example of screen display**

```
VSHIFT[1,1,1,1,10000]        FN333: Vision shift
```

See
VRESET: Vision reset (FN330)
VSTART: Vision start (FN331)
VWORK: Vision discrimination (FN332)
VDATA: Vision data (FN334)
VLOCCVT: Vision location convert (FN342)
VWAIT: Vision wait (FN343)

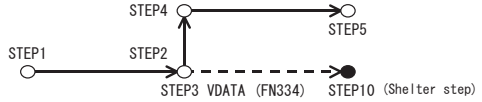
Function commands (FN codes)

Command name	VDATA
FN code	334
Title name	Vision data
General description	Get the data from the vision sensor

■ **General description**

The kind of data specified that this function is executed from the last measurement result of the vision sensor by the parameter is acquired, and stores data in the real variable register. Three kinds of data of X scale, Y scale, and the score can be selected.
The next step is not executed until the function is completed.
Jumps to the shelter step specified by the parameter the vision sensor notifies abnormality.

■ **Example of operation**



After checking position, this function is executed.
The step executed based as follows on the completion notification received from the vision sensor is decided. When it is normal completion, the following STEP4 is executed. When it is abnormal completion of being not able the measurement etc., jumps to the shelter step
Because the operation moved to the shelter step is done when abnormality occurs when measuring with the vision sensor, Robot error is not detected.
Please turn on the output signal by the routine in the shelter step when it is necessary to detect error on the outside. It is necessary to confirm an error content on the display of the vision sensor.

■ **Parameter**

Parameter No.1	1-16	Data select 1: X Scale 2: Y Scale 3: Score
Parameter No.2	1-200	Real variable No.
Parameter No.3	1-9999	Shelter step No.

■ **Example of screen display**

```
VDATA[1,1,10]      FN334:Vision data
```

See
VRESET: Vision reset (FN330)
VSTART: Vision start (FN331)
VWORK: Vision discrimination (FN332)
VSHIFT: Vision shift (FN333)

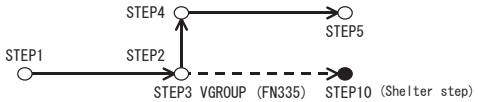
Function commands (FN codes)

Command name	VGROUP
FN code	335
Title name	Vision group change
General description	Change the measurement group of the vision sensor

■ **General description**

This function change the measurement group of the vision sensor.
The group number after this processing is stored in the specified integer variable. The next step is not executed until processing is completed.
Jumps to the shelter step specified by the parameter when the measurement group is not correctly changed, or when the vision sensor notifies abnormality.

■ **Example of operation**



After checking position, this function is executed.
The step executed based as follows on the completion notification received from the vision sensor is decided. When it is normal completion, the following STEP4 is executed. When it is abnormal completion of being not able the measurement etc., jumps to the shelter step
Because the operation moved to the shelter step is done, Robot error is not detected.
Please turn on the output signal by the routine in the shelter step when it is necessary to detect error on the outside. It is necessary to confirm an error content on the display of the vision sensor.

■ **Parameter**

Parameter No.1	1-20	Measurement group No.
Parameter No.2	1-200	Integer variable No.
Parameter No.3	1-9999	Shelter step No.

■ **Example of screen display**

```
VGROUP[2,1,10]        FN335:Vision group change
```

See
VRESET: Vision reset (FN330)
VSTART: Vision start (FN331)
VWORK: Vision discrimination (FN332)
VSHIFT: Vision shift (FN333)
VDATA: Vision data (FN334)

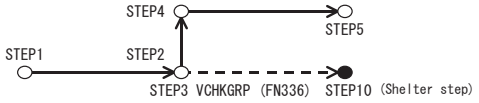
Function commands (FN codes)

Command name	VCHKGRP
FN code	336
Title name	Vision group check
General description	Check the measurement group of the vision sensor

■ **General description**

This function acquires the measurement group of the vision sensor in the integer variable specified by the parameter.
The next step is not executed until processing is completed.
Step jumps to the shelter step specified by the parameter when the vision sensor notifies abnormality. Moreover, Step jumps to the shelter step when this value and the acquired measurement number are different when the values other than 0 are set in the measurement group number of the parameter.

■ **Example of operation**



After checking position, this function is executed.
The step executed based as follows on the completion notification received from the vision sensor is decided. When it is normal completion, the following STEP4 is executed. When it is abnormal completion of being not able the measurement etc., jumps to the shelter step
Because the operation moved to the shelter step is done, Robot error is not detected.
Please turn on the output signal by the routine in the shelter step when it is necessary to detect error on the outside. It is necessary to confirm an error content on the display of the vision sensor.

■ **Parameter**

Parameter No.1	0-20	Measurement group No.
Parameter No.2	1-200	Integer variable No.
Parameter No.3	1-9999	Shelter step No.

■ **Example of screen display**

```
VCHKGRP[2,1,10]       FN336:Vision group check
```

See
VRESET: Vision reset (FN330)
VSTART: Vision start (FN331)
VWORK: Vision discrimination (FN332)
VSHIFT: Vision shift (FN333)
VDATA: Vision data (FN334)

Function commands (FN codes)

Command name	VLOCCVT
FN code	342
Title name	Vision location convert
Outline	The start or end of the shift operation with the vision sensor are specified. When shift operation start has been specified, the shift operation is performed on the basis of the shift amount stored in the specified shift register.

■ **General description**

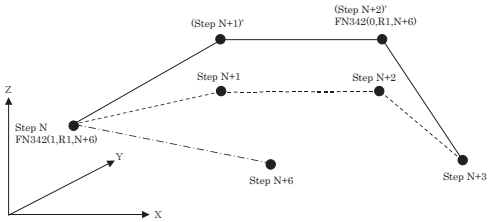
This function command proceeds with playback while shifting the recorded position in the robot program on the basis of the shift amount data stored in the specified shift register. If the shift amount data has not been set in the specified shift register, it is possible to jump to the shelter step.

The parameters are different from which vision sensor uses.

■ **Operation example**

The shift beginning is recorded at position Step N where the shift wants to begin as shown in the figure below and the shift end is recorded in position (Step N+2) 'which wants to end shifting. The robot reads the content of the shift register specified with FN342 after it reaches step N, and faces position (Step N+1) 'by which the following target position Step N+1 is shifted when reproducing. The robot works as a target position, position where the record position was similarly shifted to position (Step N+2) 'where the shift end is recorded.(solid line tracks of figure below)

The shift is done in the machine coordinate system regardless of the coordinate system selection.



When FN342 is executed in step N, if the shift data is not set when the specified shift register is read, the robot faces the shelter step (Step N+6) which is set with FN342.

[Vision sensor type1]

■ **Parameters**

Parameter No. 1	Start/End	This is used to specify the start or end of the shift operation. 1: Start / 0: End
Parameter No. 2	Shift register number	This is used to specify the shift register number. (1 to 9)
Parameter No. 3	Shelter step	This is used to specify the number of the shelter step when the shift amount data was not set in the specified shift register. (0 to 9999)

■ **Example of screen display**

```
VLOCCVT[1,R1,100]              FN342: Vision location
```

[Vision sensor type2]

■ **Parameters**

Parameter No. 1	Start/End	This is used to specify the start or end of the shift operation. 1: Start / 0: End
Parameter No. 2	Shift register number	This is used to specify the shift register number. (1 to 9)
Parameter No. 3	Base coordinate	Set a base coordinate for calculation of correction. 0:World 1:Flange
Parameter No. 4	Shelter step	This is used to specify the number of the shelter step when the shift amount data was not set in the specified shift register. (0 to 10000)

Robot will stop with recognition error if you set [10000] for shelter step.

■ **Example of screen display**

```
VLOCCVT[1,R1,0,100]            FN342: Vision location
```

See

VRESET: Vision reset (FN330)
VSTART: Vision start (FN331)
VSHIFT: Vision shift (FN333)
VWAIT: Vision wait (FN343)

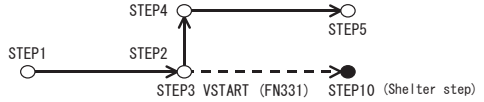
Function commands (FN codes)

Command name	VWAIT
FN code	343
Title name	Vision measure wait
General description	The measurement completion of the vision sensor is waited for.

■ **General description**

It waits until the measurement of the vision sensor is completed when this function is executed.
The next step is not executed until the measurement is completed.
It jumps to the shelter step when failure occurs when measuring by the vision sensor.

■ **Example of operation**



When passing over the position, this function is executed.
The step executed based as follows on the completion notification received from the vision sensor is decided. When it is normal completion, the following STEP4 is executed. When it is abnormal completion of being not able the measurement etc., jumps to the shelter step
Because the operation moved to the shelter step is done when abnormality occurs when measuring with the vision sensor, Robot error is not detected.
Please turn on the output signal by the routine in the shelter step when it is necessary to detect error on the outside. It is necessary to confirm an error content on the display of the vision sensor.

■ **Parameter**

Parameter No.1	1-256	Measurement No.
Parameter No.2	1-9999	Shelter step No.

The maximum value of the shelter step is 10000 for vision sensor type 2. Robot will stop with recognition error if you set [10000] for shelter step.

■ **Example of screen display**

```
VWAIT[1,10]          FN343:Vision measure wait
```

See

VRESET: Vision reset (FN330)
VSTART: Vision start (FN331)
VSHIFT: Vision shift (FN333)
VLOCCVT: Vision location convert (FN342)

Function commands(FN codes)

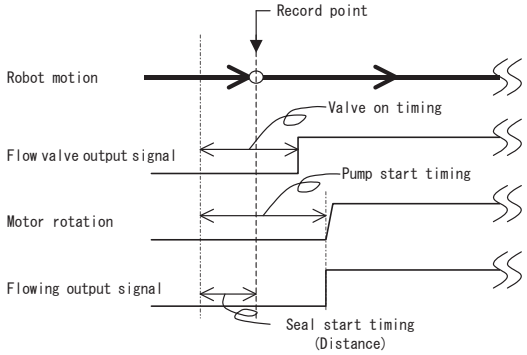
Command name	SLSTART
FN code	350
Title name	Seal start
General description	Start the dispensing.

■ General description

This instruction used to actuate the dispense applicator and start the dispensing.
Flow rate and execution timing parameters are set in “Sealing Schedule.”
This instruction contains a schedule number used as an input parameter.

■ Example of operation

The robot shall move to a specified position according to specified motion command and start dispensing according to the parameters specified in the sealing schedule.



■ Parameter

Parameter No.1	1-6	Gun No.
Parameter No.2	1-255	Seal Schedule No.

■ Example of screen display

```
SLSTART[1,1]          FN350:Seal start
```

See
SLEND: Seal end (FN351)
SLRELOAD: Reload (FN352)
SLREADY： Flow ready (FN353)
SLPRS： Seal press ctrl (FN355)

Function commands(FN codes)

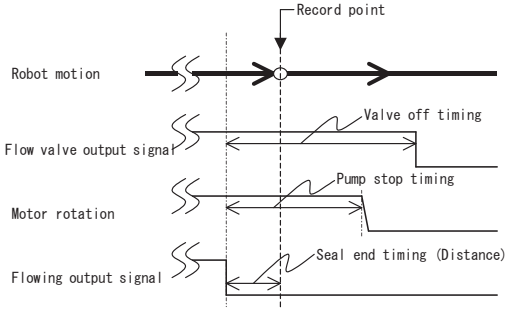
Command name	SLEND
FN code	351
Title name	Seal end
General description	Stop the dispensing process

■ General description

This instruction used to stop the dispensing process.
This instruction is executed according to the timing described to the seal schedule specified by the last executed seal start function.
The pressure correction is turned off.

■ Example of operation

The robot shall move to a specified position according to specified motion command and start dispensing according to the parameters specified in the sealing schedule.



■ Parameter

Parameter No.1	1-6	Gun No.

■ Example of screen display

```
SLEND[1]              FN351:Seal end
```

See
SLSTART: Seal start (FN350)
SLRELOAD: Reload (FN352)
SLREADY： Flow ready (FN353)
SLPRS： Seal press ctrl (FN355)

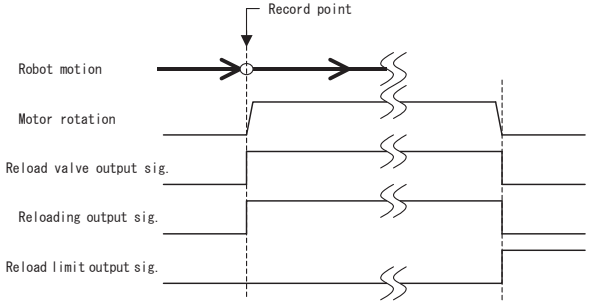
Function commands(FN codes)

Command name	SLRELOAD
FN code	352
Title name	Reload
General description	Refill the booster pump.

■ General description

This instruction used to refill the booster pump. The reload operation ends automatically when reaching the reload limit.
The robot can move while reloading.

■ Example of operation



■ Parameter

Parameter No.1	1-6	Gun No.

■ Example of screen display

```
SLRELOAD[1]                FN352:Reload
```

See
SLSTART: Seal start (FN350)
SLEND: Seal end (FN351)
SLREADY : Flow ready (FN353)
SLPRS : Seal press ctrl (FN355)

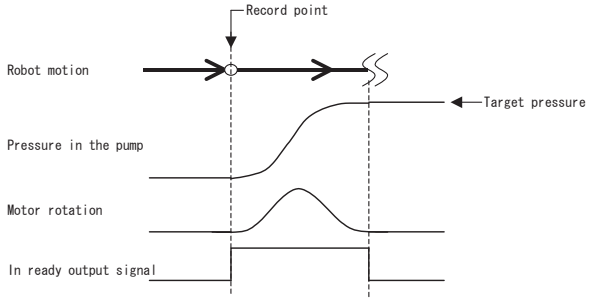
Function commands(FN codes)

Command name	SLREADY
FN code	353
Title name	Flow ready
General description	Pressure in the pump is controlled to a specified value.

■ General description

The motor which controls the pump is moved when this instruction is executed, and controls to the value for which pressure in the pump is specified by the parameter of the function.
The change of the bead shape when beginning to dispense in controlling pressure in the pump can be controlled.
The robot can move in the flow ready control.

■ Example of operation



■ Parameter

Parameter No.1	1-6	Gun number
Parameter No.2	0-50.0	Pre-pressure[MPa]

■ Example of screen display

```
SLREADY[1,1.2]             FN353: Flow ready
```

See
SLSTART Seal start (FN350)
SLEND: Seal end (FN351)
SLRELOAD: Reload (FN352)
SLPRS: Seal press ctrl (FN355)

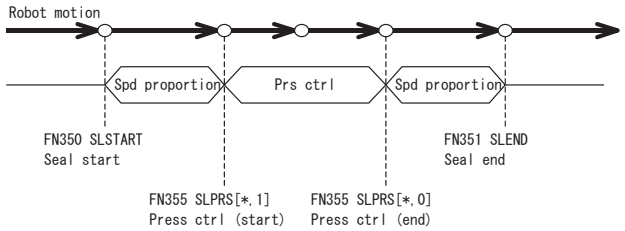
Function commands(FN codes)

Command name	SLPRS
FN code	355
Title name	Seal press ctrl
General description	Start/Stop pressure control in the dispensing process

■ General description

This instruction declares start/end of pressure control as flow control.
When this instruction for which start(ON) is specified by the parameter is executed, the dispensing will be executed by the pressure control. When this instruction is executed specifying end (OFF), the pressure control is canceled. Even if the pressure control end is not executed, the pressure control is canceled by the seal end function.

■ Example of operation



■ Parameter

Parameter No.1	1-6	Gun number
Parameter No.2	0-1	ON/OFF(1/0)

■ Example of screen display

```
SLPRS[1,1]                FN355: Seal press ctrl
```

See
SLSTART Seal start (FN350)
SLEND: Seal end (FN351)
SLRELOAD: Reload (FN352)
SLREADY : Flow ready (FN353)

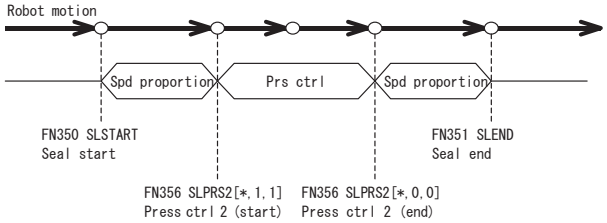
Function commands(FN codes)

Command name	SLPRSG
FN code	356
Title name	Seal press ctrl 2
General description	Start/Stop pressure control in the dispensing process

■ General description

This instruction declares start/end of pressure control as flow control.
When this instruction for which start (ON) is specified by the parameter is executed, the dispensing will be executed by the pressure control.
During pressure control, the control level is corrected by the value specified by the parameter.
It is happen the pulsation or the oscillation though the response improves if a big value is set.
When this instruction is executed specifying end (OFF), the pressure control is canceled. Even if the pressure control end is not executed, the pressure control is canceled by the seal end function. It is possible to end with FN355.

■ Example of operation



■ Parameter

Parameter No.1	1-6	Gun number
Parameter No.2	0-1	ON/OFF(1/0)
Parameter No.3	0.0-10.0	Rate

■ Example of screen display

```
SLPRSG[1,1,1.2]           FN356: Seal press ctrl 2
```

See
SLSTART Seal start (FN350)
SLEND: Seal end (FN351)
SLRELOAD: Reload (FN352)
SLREADY : Flow ready (FN353)
SLPRS : Seal press ctrl (FN355)

Function commands(FN codes)

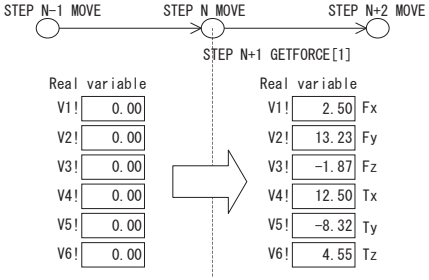
Command name	GETFORCE
FN code	360
Title name	Get force/torque
General description	The force/torque data are acquirable to a real variable.

■ General description

By executing this command, the force / torque data inputted from the force sensor are acquirable to a real variable.

■ Example of operation

The force / torque data are stored in six variables from the specified variable number.



■ Parameter

Parameter No.1	1-195	Variable No.

■ Example of screen display

```
GETFORCE[1]                    FN360:Get force/torque
```

Function commands(FN codes)

Command name	FHCLAMP
FN code	362
Title name	FH Unclamp
General description	Execute clamp moiton by FLEXhand

■ General description

By this command,the FLEXhand clamps the work.

■ Parameter

Parameter No.1	1-2	FLEXhand Number
Parameter No.2	1-2	Clamping direction 1:close 2:open
Parameter No.3	0-1	Air force mechanism 0:not use 1:use
Parameter No.4	0-50[kN]	Clamping force

■ Example of screen display

```
FHCLAMP[1,1,1,2]               FN362: FLEXhand clamp
```

Function commands(FN codes)

Command name	FHUNCLAMP
FN code	363
Title name	FH Unclamp
General description	Execute unclamp moiton by FLEXhand

■ **General description**
By this command,the FLEXhand unclamps the work.

■ **Parameter**

Parameter No.1	1-2	FLEXhand Number
Parameter No.2	0-1	Specifying method of opened finger position 0:absolute 1:relative
Parameter No.3	-1000-1000[mm]	Opened finger position

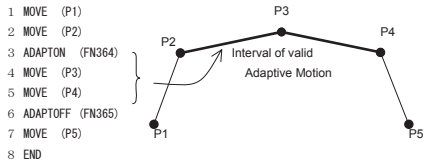
■ **Example of screen display**

```
FHUNCLAMP[1,1,1,10]                    FN363: FLEXhand unclamp
```

Function commands (FN codes)

Command name	ADAPTON
FN code	364
Title name	Adaptive Motion ON
General description	Starts Adaptive Motion with the specified condition.

■ **General description**
During recording steps from Adaptive Motion ON function (FN364) to Adaptive Motion OFF function (FN365), Adaptive Motion function becomes valid and robot moves in accordance with external force.



■ **Parameter**

Parameter No.1	Condition No.	Conditional numbers selected in “Constants-39 Adaptive Motion” shall be set up. (1~10)
Parameter No.2	Command position replacing ON/OFF(1/0)	1: After taking in present position changed by external force, trace calculation to next step is done based on its position. 0: Trace is normally calculated without taking in present position halfway in the step.

■ **Example of screen display**

```
ADAPTON[1,1]                    FN364: Adaptive Motion ON
```

You can change the condition number by executing ADAPTON again while Adaptive Motion is running.

See
ADAPTOFF: Adaptive Motion OFF (FN365)

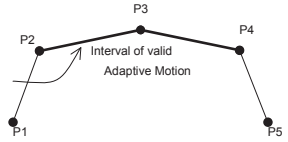
Function commands (FN codes)

Command name	ADAPTOFF
FN code	365
Title name	Adaptive Motion OFF
General description	Ends Adaptive Motion.

■ General description

This command enables to end Adaptive Motion function.

```
1 MOVE  (P1)
2 MOVE  (P2)
3 ADAPTON  (FN364)
4 MOVE  (P3)
5 MOVE  (P4)
6 ADAPTOFF  (FN365)
7 MOVE  (P5)
8 END
```



When this function is executed, a trajectory to the next step is created form robot's present position.

■ Parameter

Nil		

■ Example of screen display

```
ADAPTOFF                    FN365: Adaptive Motion OFF
```

See
ADAPTON: Adaptive Motion ON (FN364)

Function commands(FN codes)

Command name	FHCLAMP2
FN code	366
Title name	New FH Clamp
General description	Execute clamp moiton by FLEXhand

■ General description

By this command,the FLEXhand clamps the work.

■ Parameter

Parameter No.1	1-2	FLEXhand Number
Parameter No.2	1-2	Clamping direction 1:close 2:open
Parameter No.3	0-1	Air force mechanism 0:not use 1:use
Parameter No.4	0-50[kN]	Clamping force
Parameter No.5	0-500	Clamping speed
Parameter No.6	0-1	Brake during clamping 0:not use 1:use
Parameter No.7	0-2[s]	Brake delay time

■ Example of screen display

```
FHCLAMP2[1,1,0,1,0,0,1]                    FN366: New FLEXhand clamp
```

Function commands(FN codes)

Command name	FHUNCLAMP 2
FN code	367
Title name	New FH Unclamp
General description	Execute unclamp moiton by FLEXhand

■ General description

By this command,the FLEXhand unclamps the work.

■ Parameter

Parameter No.1	1-2	FLEXhand Number
Parameter No.2	0-1	Specifying method of opened finger position 0:absolute 1:relative
Parameter No.3	-1000-1000[mm]	Opened finger position

■ Example of screen display

```
FHUNCLAMP2[1,1,1,10]                    FN367: New FLEXhand unclamp
```

Function commands(FN codes)

Command name	FHCLAMPDCT
FN code	368
Title name	FH Clamping Detection
General description	Detect clamp status by FLEXhand

■ General description

This command detects that the FLEXhand clamps the work.

■ Parameter

Parameter No.1	1-2	FLEXhand Number
Parameter No.2	0-1000[mm]	Clamping position
Parameter No.3	0-60[sec]	Wait time
Parameter No.4	1-9999	Escape destination step
Parameter No.5	1-200 or 301-500	Real variable number for clamping position value.

■ Example of screen display

```
FHCLAMPDCT[1,85,1,7,V1!]                FN368:Fhand Clamping Detection
```

Function commands (FN codes)

Command name	CHGENDLESS
FN code	373
Title name	Change endless control
General description	The control of the endless rotation axis to change.

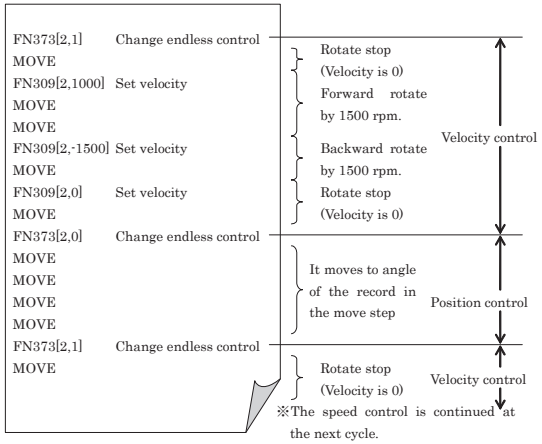
■ General description

It is a function to change the control of the endless rotation axis to the position control or the speed control.
If you use the position control, endless rotation axis moves to recorded position.
if you use the velocity control, endless rotation axis doesn't move to recorded position. The axis rotates by the speed setting of SETVELO; Set velocity (FN309).

Only endless axis set by "Change control" can use the function. Please set the endless rotation axis to "Change" by [Constant setting][Machine constant][Endless Rotation].

This function becomes in position only when the control changes, and robot stops one moment. Robot doesn't stop when the control doesn't change.
Please record the function between the start and the end section.

■ Example of operation



The example is the mechanism 2 as an endless aixs.

■ Parameter

Parameter No. 1	Mechanism Number	This specifies the mechanism number of a change type endless axis. (1 – 9)

Parameter No. 2	Control	The changed control is specified. 0:Potision control 1:Velocity control

■ Example of screen display

CHGENDLESS[2,0]	FN373;Change endless control

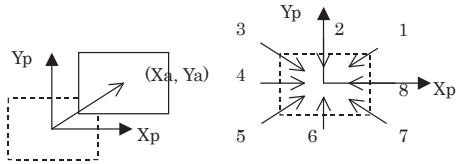
See
SETVELO;Set velocity(FN309)

Function commands (FN codes)

Command name	PALLET3_APR
FN code	374
Title name	Palletize approach selection
General description	Starts approaching motion.

■ General description

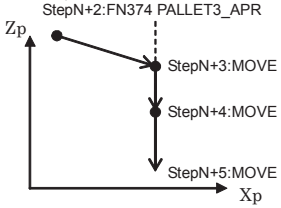
'Approaching motion' is for preventing from collision with works already stacked at the time of stacking a new work.
'Approach distance' is defined in palletize pattern, and 'Approach direction' is desgined each works. When this function is executed, robot performs 'Approaching motion' according to the setting from next step.



■ Example of operation

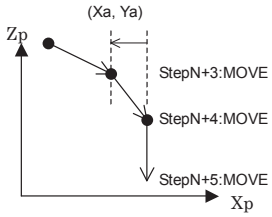
Following figure shows FN249 palletize start is recorded at step N+1. And FN374 Palletize approach selection is recorded at step N+1. And Move step N+2, N+3 and N+4 are also recorded.

StepN :MOVE
StepN+1:FN249 PALLET3
StepN+2:FN374 PALLET3_APR

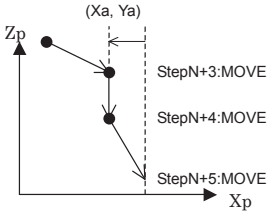


Program sample of approach

Robot starts palletizing based on the palletize pattern of FN249 and the palletize counter. At next step of FN374, Robot moves with palletizing shift and approach shift that includes some value and direction.



Approach motion 1
(Appr-down)



Approach motion 2
(Down-appr)

You can select Appr-down or Down-appr by the parameter of function.

■ **Set Approach direction**

Approach direction 1to 8 is set by pallet coordinate. Approach direction doesn't change even if work rotates.

■ **Approach motion**

The next step of PALLET3_APR will move with approach shift. In case of Down-appr, robot has some approach shift from next two step of function. Please be careful when you add some steps.

■ **Parameter**

Parameter No.1	Palletize No.	This specifies the palletizing number to be reset. (1 to 100)
Parameter No.2	Route selection	Select Appr-down(0) or Down-appr(1)

■ **Example of screen display**

```
PALLET3_APR[1, 0]      FN374 Palletize approach selection
```

See

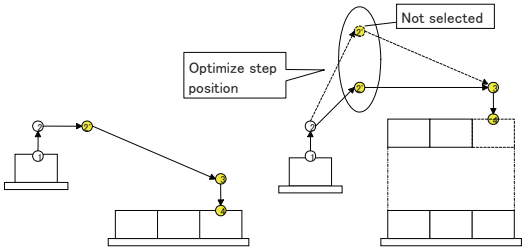
PALLET3: Palletize start (FN249)
PALLET3_END: Palletize end (FN250)
PALLET3_RESET: Palletize reset (FN251)
PALLET3_OPT: Palletize optimize path(FN375)
PALLET3_SELGR: Palletize select grasp position(FN376)
PALLET3_GETREG: Get palletize register(FN377)
PALLET3_SETREG: Set palletize register(FN378)

Function commands (FN codes)

Command name	PALLET3_OPT
FN code	375
Title name	Palletize optimize path
General description	Optimize Step position based on the locus of previous step and following step

■ **General description**

Palletizing motion often has some useless motion when steps are recorded as avoiding the interference in case of big cargo. This function optimizes and adjusts the next position automatically based on the locus of previous step and following step.



■ **Example of operation**

Concretely a Z axis value of "Next move position" will be replaced with higher Z axis value of two command positions.
Regarding two command positions, one is the previous step of the function, another is the reference move step where is set by parameter of the function.
A Z axis value is calculated by the pallet coordinates used as reference Palletize Pattern.

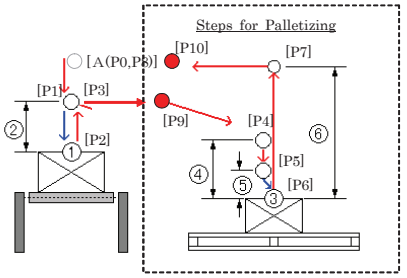
■ **Parameter**

Parameter No.1	Palletize No.	This specifies the palletizing number to be executed. (1 to 100)
Parameter No.2	Reference step	Reference move step (1 to 9999) Z axis value will be compared.

■ **Example of screen display**

```
PALLET3_OPT[1, 20]      FN375 Palletize optimize path
```

■ **Example of teaching**




```
1    REM                    Program Name
```

```
2    *TOP
3    RESET [01]
4    SET[02]
5    100% LIN A8  T2            Move to P1
6    WAITI[13]
7    WAITI[12]
8     30% LIN A1P T2            Move to P2
9    RESET[02]
10   SET[01]
11   WAITI[11]
12   100% LIN A8  T1            Move to P3   (Compared with ①)
13   PALLET3[1,0,10,020]        Palletizing START
14   PALLET3_APR[1,1]           Approach function Enabled
15   PALLET3_OPT[1,17]          Optimized locus (Refer step 17)
16   100% LIN A8  T1            Move to P9
17   100% LIN A8  T1            Move to P4   (Compared with ②)
18   100% LIN A8  T1            Move to P5
19    30% LIN A1P T1            Move to P6
20   RESET[01]
21   SET[02]
22   100% LIN A8  T2            Move to P7   (Compared with ①)
23   PALLET3_OPT[1,26]          Optimized locus (Refer step 26)
24   100% LIN A8  T1            Move to P10
25   PALLETIZE_END[1,030]       Palletizing END
26   100% LIN A8  T2            Move to P8   (Compared with ②)
27   GOTO *TOP                  Go to top of loop
28   END
```

See
PALLET3: Palletize start (FN249)
PALLET3_END: Palletize end (FN250)
PALLET3_RESET: Palletize reset (FN251)
PALLET3_APR: Palletize approach selection(FN374)
PALLET3_SELGR: Palletize select grasp position(FN376)
PALLET3_GETREG: Get palletize register(FN377)
PALLET3_SETREG: Set palletize register(FN378)

Function commands (FN codes)

Command name	PALLET3_SELGR
FN code	376
Title name	Palletize select grasp position
General description	Select Work grasp position from registered by Palletize pattern.

■ General description

This function selects one Work grasp position from “Work grasp position” which is set by Palletize pattern which has maximum 4 positions.
Suppose that robot grasps some cargo at once. Robot moves toward to cargo position set by Palletize pattern, but robot cannot put a cargo correctly if every cargo has different grasp position.
If you select the grasp position for every cargo correctly, robot target position would be the center of cargo.

If you want to change a work grasp position for cargo (work) number, please use FN377 PALLET_GETREG. This function can get a current work number and store V% variable etc. Then please record SWICH command and FN376 PALLET3_SELGR to move correctly.

■ Default

If this function never runs before, automatically work grasp position is set to one. After changing work grasp position, the position would be kept until the palletize END.

■ Parameter

Parameter No.1	Palletize No.	This specifies the palletizing number to be executed. (1 to 100)
Parameter No.2	Work grasp position	Work grasp position (1 to 4) which is set by Palletize pattern.

■ Example of screen display

```
PALLET3_SELGR[1, 1]   FN376 Palletize select grasp position
```

See
PALLET3: Palletize start (FN249)
PALLET3_END: Palletize end (FN250)
PALLET3_RESET: Palletize reset (FN251)
PALLET3_APR: Palletize approach selection(FN374)
PALLET3_OPT: Palletize optimize path(FN375)
PALLET3_GETREG: Get palletize register(FN377)
PALLET3_SETREG: Set palletize register(FN378)

Function commands (FN codes)

Command name	PALLET3_GETREG
FN code	377
Title name	Get palletize register
General description	Store some palletize register to some variables.

■ General description

Palletize register means internal variables that shows some status of palletize.
You don't have to use the function normally because it is for system software. If you want to check some status and use them for changing sequence of robot program, the function offers some useful status to you.

(Example)
Change locus depended on the palletize counter.
Skip palletize counter with intension.
Modify shift values.

This function stores some palletize registers to some variables. If you want to change back some palletize registers from some variables, you use FN378 PALLET3_SETREG.

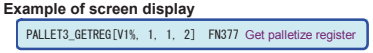
■ Palletize Registers

No.	Register name	Description	Type	Range
1	Palletize No.	Target palletize No.	Int.	1～100
2	Pallet No.	Target pallet No.	Int.	1～100
3	Running status	Stop or Running	Int.	0/1
4	Type	Palletizing or De-palletizing	Int.	0/1
5	Layer counter	Layer No. in running	Int.	0～50
6	Work counter	Work No. in running	Int.	0～99
7	Total counter	Total work counter in running （）	Int.	0～4950
8	Grasp position sel.	Work grasp position No. in use	Int.	1～4
9	Approach dir.	Approach direction in use	Int.	0～8
10	Layer count signal	LSB Signal No. of Layer count signal	Int.	0～1024
11	Work count signal	LSB Signal No. of work count signal	Int.	0～1024
12	Work complete signal	Work complete signal No.	Int.	0～1024
13	Shift value X	Current shift value of X axis based on the pallet coordinate	Real	—
14	Shift value Y	Current shift value of Y axis based on the pallet coordinate	Real	—
15	Shift value Z	Current shift value of Z axis based on the pallet coordinate	Real	—
16	Shift value Rz	Current shift value of Rz axis based on the pallet coordinate	Real	—

■ Parameter

Parameter No.1	Variable No.	Target variable No. to store value Select from V!, V%, L! or L%
Parameter No.2	Palletize No.	This specifies the palletizing number to be executed. (1 to 100)
Parameter No.3	Register type (Start)	Set number of Palletize Registers(1 - 255)
Parameter No.4	Register type (End)	Set number of Palletize Registers(1 - 255) You can select and get consecutive register values from Start to End

■ Example of screen display



See
PALLET3: Palletize start (FN249)
PALLET3_END: Palletize end (FN250)
PALLET3_RESET: Palletize reset (FN251)
PALLET3_APR: Palletize approach selection(FN374)
PALLET3_OPT: Palletize optimize path(FN375)
PALLET3_SELGR: Palletize select grasp position(FN376)
PALLET3_SETREG: Set palletize register(FN378)

Function commands (FN codes)

Command name	PALLET3_SETREG
FN code	378
Title name	Set palletize register
General description	Set some variables to some palletize registers.

■ General description

Palletize register means internal variables which shows some status of palletize.
You don't have to use the function normally because it is for system software. If you want to check some status and use them for changing sequence of robot program, the function offers some useful status to you.

(Example)
Change locus depended on the palletize counter.
Skip palletize counter with intension.
Modify shift values.

This function sets some variables to some palletize registers. If you want to change back some variables from some palletize registers, you use FN377 PALLET3_GETREG.

Some registers cannot written. Please refer to table below "Palletize Registers" which register is writable.
When the register that cannot be written is specified, it is not changed though alarm doesn't occur.

■ Cautions

Changing registers effect the motion of robot. Do not replace the registers if you don't use them.

■ Palletize Registers

No.	Register name	Description	Type	Range	Write
1	Palletize No.	Target palletize No.	Int.	1～100	
2	Pallet No.	Target pallet No.	Int.	1～100	
3	Running status	Stop or Running	Int.	0/1	
4	Type	Palletizing or De-palletizing	Int.	0/1	○
5	Layer counter	Layer No. in running	Int.	0～50	○
6	Work counter	Work No. in running	Int.	0～99	○
7	Total counter	Total work counter in running	Int.	0～4950	
8	Grasp position sel.	Work grasp position No. in use	Int.	1～4	○
9	Approach dir.	Approach direction in use	Int.	0～8	○
10	Layer count signal	LSB Signal No. of Layer count signal	Int.	0～1024	○
11	Work count signal	LSB Signal No. of work count signal	Int.	0～1024	○
12	Work complete signal	Work complete signal No.	Int.	0～1024	○
13	Shift value X	Current shift value of X axis based on the pallet coordinate	Real	—	○
14	Shift value Y	Current shift value of Y axis based on the pallet coordinate	Real	—	○
15	Shift value Z	Current shift value of Z axis based on the pallet coordinate	Real	—	○
16	Shift value Rz	Current shift value of Rz axis based on the pallet coordinate	Real	—	○

■ Parameter

Parameter No.1	Palletize No.	This specifies the palletizing number to be executed. (1 to 100)
Parameter No.2	Register type (Start)	Set number of Palletize Registers(1 - 255)
Parameter No.3	Register type (End)	Set number of Palletize Registers(1 - 255) You can select and get consecutive register values from Start to End
Parameter No.4	Variable No.	Target variable No. to store value Select from V!, V%, L! or L%

■ Example of screen display

See
PALLET3: Palletize start (FN249)
PALLET3_END: Palletize end (FN250)
PALLET3_RESET: Palletize reset (FN251)
PALLET3_APR: Palletize approach selection(FN374)
PALLET3_OPT: Palletize optimize path(FN375)
PALLET3_SELGR: Palletize select grasp position(FN376)
PALLET3_GETREG: Get palletize register(FN377)

Function commands (FN codes)

Command name	PALLET3_SELZ
FN code	388
Title name	Palletize select height (z)
General description	The function compares the Z-axis value of target step and the Z-axis value of reference step after palletizing shift by using target palletizing number set by the first parameter of the function, and then Z-axis value of target step will be adjust higher value of both.

■ General description

Optimize path selection adjusts not only the Z direction (height) shift value but also X and Y direction shift because target step is inside palletize section. This function can be placed the target step at outside of palletize section, and then you can get only Z direction shift.

A target step of Z-axis direction is the next move step of the function.

The function compares the Z-axis value of target step and the Z-axis value of reference step after palletizing shift by using target palletizing number set by the first parameter of the function, and then Z-axis value of target step will be adjust higher value of both.

■ Parameter

Parameter No.1	Palletize No.	This specifies the palletizing number to be executed. (1 to 100)
Parameter No.2	Reference Step	Compared move step. (1 to 9999)

■ Example of screen display

```
PALLET3_SELZ[1, 12]  FN388 Palletize select height(z)
```

See

PALLET3: Palletize start (FN249)
PALLET3_END: Palletize end (FN250)
PALLET3_RESET: Palletize reset (FN251)
PALLET3_APR: Palletize approach selection (FN374)
PALLET3_OPT: Palletize optimize path (FN375)
PALLET3_SELGR: Palletize select grasp position (FN376)
PALLET3_GETREG: Get palletize register (FN377)
PALLET3_SETREG: Set palletize register (FN378)

Function Commands (FN Codes)

Command name	JMPPBCD
FN code	400
Title name	Program jump (Selecting ext.BCD prog.)
General description	This command enables to externally jump to the program designated by the BCD code.

■ General description

This command enables to jump to the program externally designated by the BCD code.

However, if the designated input signal has not been entered, it will not jump but only pass. Alternatively, it is also available to wait in the current step until the signal is entered.
For the branch program No., enter it in “Branch Prg.No.(BCD)” in the standard input signal. To use this function command, the signal must be assigned in advance. If not assigned, the program No. is to be designated to 0 and to jump to the head step of the program No. 0.

Note that, if a function command is stored in the head step of the destination program, the function command in the destination is to be executed in the step where the jump command was executed.

Even after completing the playback of the destination program, the robot will not return to the original program.

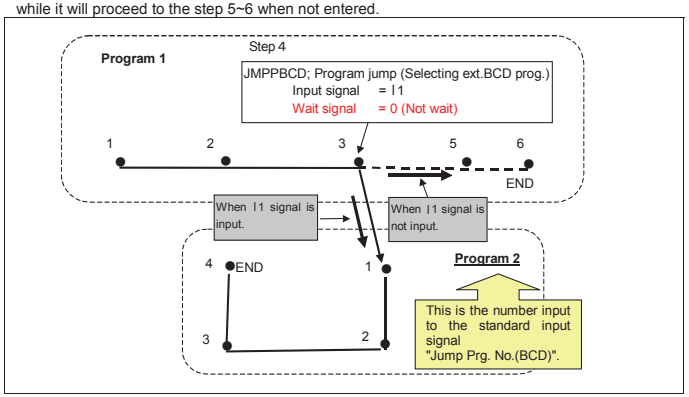
In the multi-unit specifications, the program in the destination is confined to the one within the same unit. (It is not available to jump to the program in the different unit.)

■ Example of operation

● With the wait input signal “OFF”

Store the JMPPBCD command, the input signal=1 and the wait signal=0 in the step 4. (0: Input wait ”OFF”/ 1: Input wait “ON”) And during playback, the program 2 is supposed be specified by the external signal (BCD code).

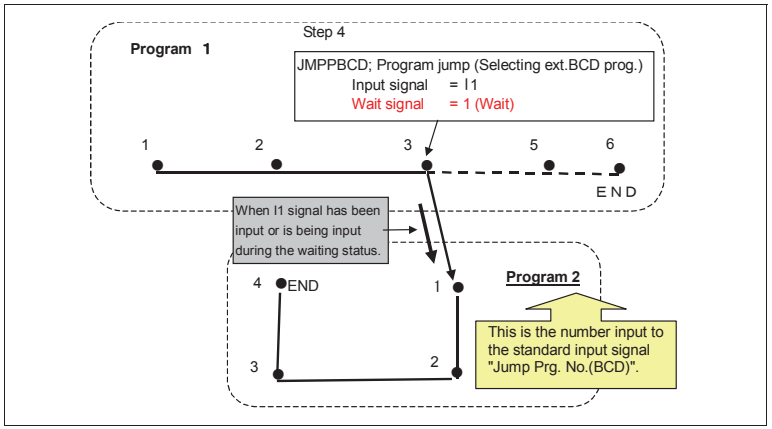
By performing playback, the robot will jump to the head step of the program 2 when the input signal I1 has been entered; while it will proceed to the step 5~6 when not entered.



● With the wait input signal “ON”

Store the JMPPBCD command, the input signal=1 and the wait signal=0 in the step 4. (0: Input wait ”OFF”/ 1: Input wait “ON”) And during playback, the program 2 is supposed be specified by the external signal (BCD code).

By performing playback, the robot will jump to the head step of the program 2 when the input signal I1 has been entered; while it will wait in the step 3 until the signal is entered in the input signal I1 when not entered. If the input signal I1 is entered while waiting, the robot will jump to the head of the program 2.



■ **Parameters**

Parameter No.1	Input signal	This parameter stores the input signal No. as the jump execution condition. (1～2048, 5001～5196) Multiple signals can be specified by designating 5001～5196.
Parameter No.2	Wait signal	The robot jumps when the designated input signal above has been entered; while it will not jump but only pass when not entered. Alternatively, the robot can even wait in the current step until the signal is entered. 0: Wait signal "OFF" 1: Wait signal "ON"

■ **Example of screen display**

```
JMPPBCD[I1, 1]   FN400:Program jump (Selecting ext.BCD prog.)
```

See
CALLPBCD ; Program call (Selecting ext.BCD prog.) (FN402)
JMPPBIN ; Program jump (Selecting ext.BIN prog.) (FN401)
CALLPBIN ; Program call (Selecting ext.BIN prog.) (FN403)

Function Commands (FN Codes)

Command name	JMPPBIN
FN code	401
Title name	Program jump (Selecting ext.BIN prog.)
General description	The robot jumps to the program externally designated by the binary code.

■ **General description**

This command enables to jump to the program externally designated by the binary code.

However, if the designated input signal has not been entered, it will not jump but only pass. Alternatively, it is also available to wait in the current step until the signal is entered.
For the branch program No., enter it in "Branch Prg.No.(BCD)" in the standard input signal. To use this function command, the signal must be assigned in advance. If not assigned, the program is to be designated to 0 and to jump to the head step of the program No. 0.

Note that, if the function command is stored in the head step of the destination program, the function command in the destination is to be executed in the step where the jump command was executed.

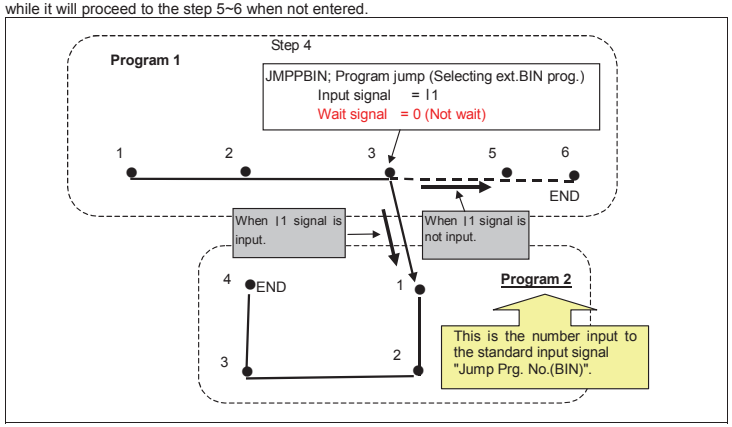
After completing the playback of the destination program, the robot will not return to the original program.

In the multi-unit specifications, the program in the destination is confined to the one within the same unit. (It is not available to jump to the program in the different unit.)

■ **Example of operation**

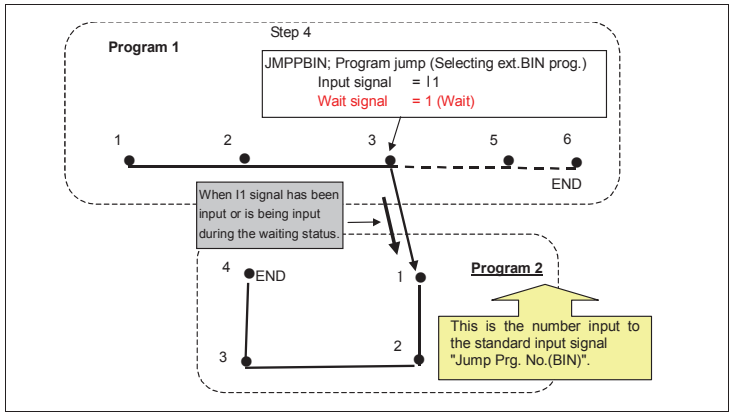
● **With the wait input signal "OFF"**

Store the JMPPBIN command, the input signal=1 and the wait signal=0 in the step 4. (0: Input wait "OFF"/ 1: Input wait "ON") And during playback, the program 2 is supposed be specified by the external signal (Binary code).
By performing playback, the robot will jump to the head step of the program 2 when the input signal I1 has been entered; while it will proceed to the step 5~6 when not entered.



● **With the wait input signal "ON"**

Store the JMPPBIN command, the input signal=1 and the wait signal=0 in the step 4. (0: Input wait "OFF"/ 1: Input wait "ON") And during playback, the program 2 is supposed be specified by the external signal (Binary code).
By performing playback, the robot will jump to the head step of the program 2 when the input signal I1 has been entered; while it will wait in the step 3 until the signal is entered in the input signal I1 when not entered. If the input signal I1 is entered while waiting, the robot will jump to the head of the program 2.



■ **Parameters**

Parameter No.1	Input signal	This parameter stores the input signal No. as the jump execution condition. (1～2048, 5001～5196) Multiple signals can be specified by designating 5001～5196.
Parameter No.2	Wait signal	The robot jumps when the designated input signal above has been entered; while it will not jump but only pass when not entered. Alternatively, the robot can even wait in the current step until the signal is entered. 0: Wait signal "OFF" 1: Wait signal "ON"

■ **Example of screen display**

```
JMPPBIN[I1, 1]   FN401:Program jump (Selecting ext.BIN prog.)
```

See
CALLPBIN ; Program call (Selecting ext.BIN prog.) (FN403)
JMPPBCD ; Program jump (Selecting ext.BCD prog.) (FN400)
CALLPBCD ; Program call (Selecting ext.BCD prog.) (FN402)

Function Commands (FN Codes)

Command name	CALLPBCD
FN code	402
Title name	Program call (Selecting ext.BCD prog.)
General description	This command enables to call the program externally designated by the BCD code.

■ **General description**

This command enables to call the program externally designated by the BCD code.

However, if the designated input signal has not been entered, it will not call but only pass. Alternatively, it is also available to wait in the current step until the signal is entered.
For the branch program No., enter it in "Branch Prg.No.(BCD)" in the standard input signal. To use this function command, the signal must be assigned in advance. If not assigned, the program No. is to be designated to 0 and to jump to the head step of the program No. 0.

Note that, if a function command is stored in the head step of the call destination program, the function command in the destination is to be executed in the step where the call command was executed.

Even after completing the playback of the destination call program, the robot will not return to the original program.

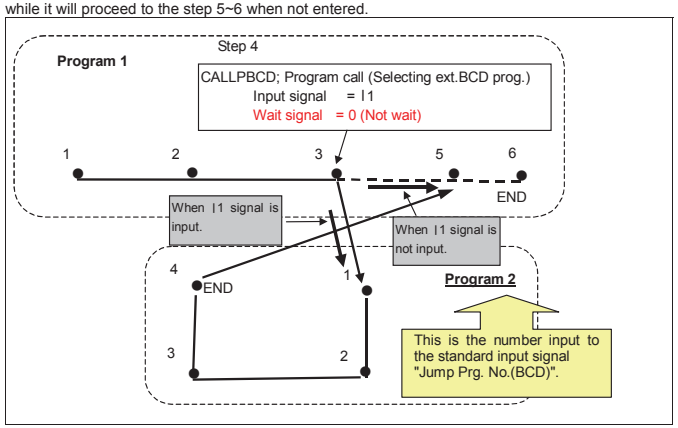
In the multi-unit specifications, the program in the call destination is confined to the one within the same unit. (It is not available to cal the program in the different unit.)

■ **Example of operation**

● **With the wait input signal "OFF"**

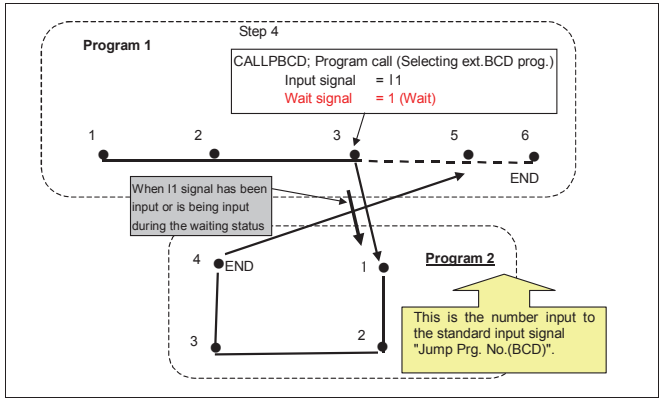
Store the CALLPBCD command, the input signal=1 and the wait signal=0 in the step 4. (0: Input wait "OFF"/ 1: Input wait "ON") And during playback, the program 2 is supposed be specified by the external signal (BCD code).

By performing playback, the robot will jump to the head step of the program 2 when the input signal I1 has been entered; while it will proceed to the step 5~6 when not entered.



● **With the wait input signal "ON"**

Store the CALLPBCD command, the input signal=1 and the wait signal=0 in the step 4. (0: Input wait "OFF"/ 1: Input wait "ON") And during playback, the program 2 is supposed be specified by the external signal (BCD code).
By performing playback, the robot will jump to the head step of the program 2 when the input signal I1 has been entered; while it will wait in the step 3 until the signal is entered in the input signal I1 when not entered. If the input signal I1 is entered while waiting, the robot will jump to the head of the program 2.



■ **Parameters**

Parameter No.1	Input signal	This parameter stores the input signal No. as the call execution condition. (1～2048、5001～5196) Multiple signals can be specified by designating 5001～5196.
Parameter No.2	Wait signal	The robot calls when the designated input signal above has been entered; while it will not call but only pass when not entered. Alternatively, the robot can even wait in the current step until the signal is entered. 0: Wait signal “OFF” 1: Wait signal “ON”

■ **Example of screen display**

```
CALLPBCD[I1, 1]   FN402:Program call (Selecting ext.BCD prog.)
```

See
JMPPBCD ; Program jump (Selecting ext.BCD prog.) (FN400)
JMPPBIN ; Program jump (Selecting ext.BIN prog.) (FN401)
CALLPBIN ; Program call (Selecting ext.BIN prog.) (FN403)

Function Commands (FN Codes)

Command name	CALLPBIN
FN code	403
Title name	Program call (Selecting ext.BIN prog.)
General description	The robot calls the program externally designated by the binary code.

■ **General description**

This command enables to call the program externally designated by the binary code.

However, if the designated input signal has not been entered, it will not call but only pass. Alternatively, it is also available to wait in the current step until the signal is entered.
For the branch program No., enter it in “Branch Prg.No.(BCD)” in the standard input signal. To use this function command, the signal must be assigned in advance. If not assigned, the program is to be designated to 0 and to jump to the head step of the program No. 0.

Note that, if the function command is stored in the head step of program in the call destination. Note that, if the function command is stored in the head step of the destination program, the function command in the destination is to be executed in the step where the call command was executed.

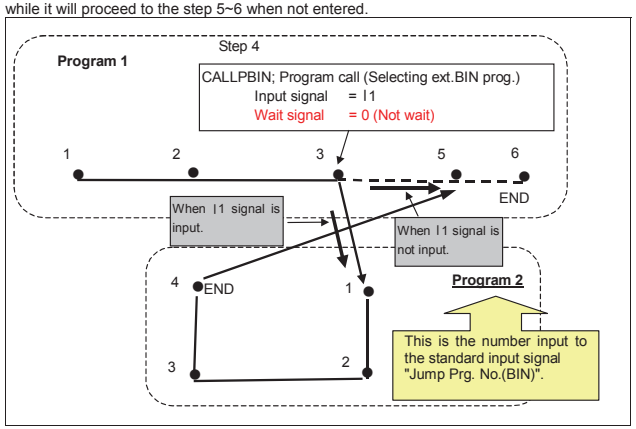
After completing the playback of the call destination program, the robot will not return to the original program.

In the multi-unit specifications, the program in the call destination is confined to the one within the same unit. (It is not available to call the program in the different unit.)

■ **Example of operation**

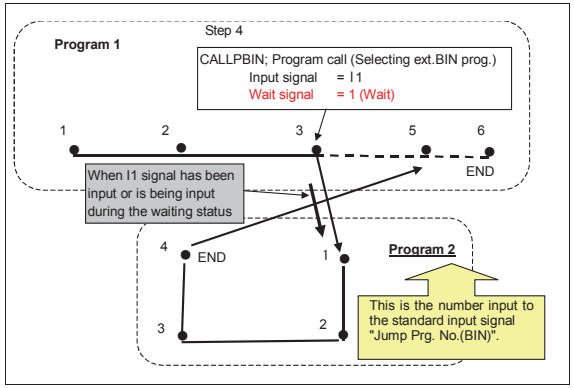
● **With the wait input signal “OFF”**

Store the CALLPBIN command, the input signal=1 and the wait signal=0 in the step 4. (0: Input wait "OFF"/ 1: Input wait “ON”) And during playback, the program 2 is supposed be specified by the external signal (Binary code).
By performing playback, the robot will jump to the head step of the program 2 when the input signal I1 has been entered; while it will proceed to the step 5~6 when not entered.



● **With the wait input signal “ON”**

Store the CALLPBIN command, the input signal=1 and the wait signal=0 in the step 4. (0: Input wait "OFF"/ 1: Input wait “ON”) And during playback, the program 2 is supposed be specified by the external signal (Binary code).
By performing playback, the robot will jump to the head step of the program 2 when the input signal I1 has been entered; while it will wait in the step 3 until the signal is entered in the input signal I1 when not entered. If the input signal I1 is entered while waiting, the robot will jump to the head of the program 2.



■ **Parameters**

Parameter No.1	Input signal	This parameter stores the input signal No. as the call execution condition. (1～2048, 5001～5196) Multiple signals can be specified by designating 5001～5196.
Parameter No.2	Wait signal	The robot calls when the designated input signal above has been entered; while it will not call but only pass when not entered. Alternatively, the robot can even wait in the current step until the signal is entered. 0: Wait signal “OFF” 1: Wait signal “ON”

■ **Example of screen display**

```
CALLPBIN[I1, 1]   FN403:Program call (Selecting ext.BIN prog.)
```

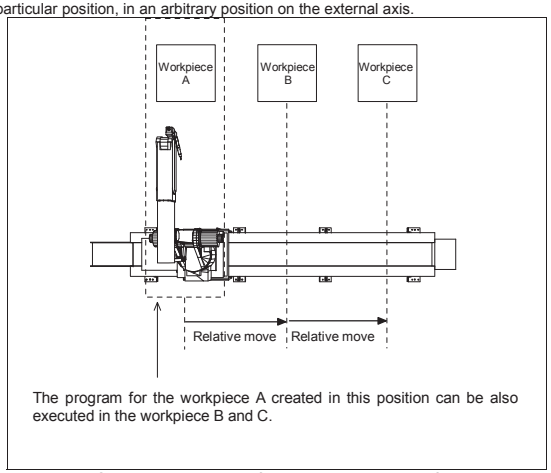
See
JMPPBIN ; Program jump (Selecting ext.BIN prog.) (FN401)
JMPPBCD ; Program jump (Selecting ext.BCD prog.) (FN400)
CALLPBCD ; Program call (Selecting ext.BCD prog.) (FN402)

Function Commands (FN Codes)

Command name	RELMOV
FN code	407
Title name	Move of External axis
General description	The designated external axis moves the specified distance from the current position.

■ **General description**

This command enables an external axis such as a slider, a positioner etc. (auxiliary axis) to move the specified distance or angle from the current position. This is to be used for repeatedly using the program, created when the external axis was in a particular position, in an arbitrary position on the external axis.

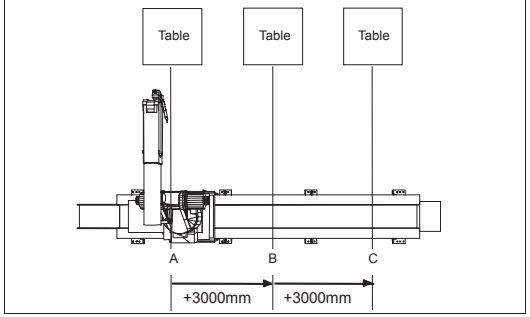


A single RELMOV command can specify a single relative move of the external axis.
The type of interpolation from the current position to the target position is the axis interpolation motion.

■ **Example of operation**

This function command enables the designated external axis to move the specified distance or angle from the current position. The simplest way of executing this command is to call the program, created when the external axis was in a particular position, again after executing this RELMOV command.

Steps	Operation of Slider	Destination of Slier
100% JOINT	Slider moves to point A.	A
CALLP [100]	Program 100 is called.	
RELMOV +3000	Slider shifts +3000mm from point A.	B
CALLP [100]	Program 100 is called.	
RELMOV +3000	Slider shifts +3000mm from point B.	C
CALLP [100]	Program 100 is called.	



■ **Parameters**

Parameter No.1	Mechanism No.	This specifies the mechanism No. defined in the current unit. (1~9)
Parameter No.2	Axis No.	This designates the axis No. for move in the mechanism specified above. For the 1st axis slider : 1 For the 2nd axis positioner : 2
Parameter No.3	Rate of Move Speed	This specifies the move speed. (1~100%)
Parameter No.4	Distance/ Angle of Movement	For slider, this specifies the move distance from the current position. (-9999~9999mm) For positioner, this specifies the rotation angle from the current position. (-999.9~999.9deg)

■ **Example of screen display**

```
RELMOV[4, 1, 100, 3000]    FN407: Move of External axis
```

Function Commands (FN Codes)

Command name	ICH
FN code	410
Title name	Inching
General description	Performs inch the wire with specified time and wire speed.

■ **General description**

The robot performs to inch the wire with specified time and wire speed.
Specified time and speed determines the inching value.

The robot does not stop operating while the inching or retract command is executed.
Furthermore, if the arc start command is placed immediately after an inching or retract command in the teaching, welding will be started as soon as the inching or retract operation is completed.

The inching or retract command is not executed in the following situations.

- When welding ON/OFF has been set to OFF
- During a welding section (If TIG filler wire is used by the APDA-301, the retract command can be executed. In the case of other power supplies, neither inching commands nor retract commands can be executed in the welding section.)
- If operation has been temporarily stopped and restarted during inching or retraction

■ **Parameters**

Conditions	Range	Unit
Time	0.0 ～ 9.9	sec
Wire speed	1 ～ 9999	cm/min

■ **Example of screen display**

```
ICH[W1,2.0s,  6%]          FN410:Inching
```

[ ] shows the welder number, time, and wire speed from the left.

See
RTC ; Retract (FN411)

Function Commands (FN Codes)

Command name	RTC
FN code	411
Title name	Retract
General description	Performs to retract the wire with specified time and wire speed.

■ General description

The robot performs to retract the wire with specified time and wire speed.
Specified time and speed determines the retract value.

The robot does not stop operating while the inching or retract command is executed.
Furthermore, if the arc start command is placed immediately after an inching or retract command in the teaching, welding will be started as soon as the inching or retract operation is completed.

The inching or retract command is not executed in the following situations.

- When welding ON/OFF has been set to OFF
- During a welding section (If TIG filler wire is used by the APDA-301, the retract command can be executed. In the case of other power supplies, neither inching commands nor retract commands can be executed in the welding section.)
- If operation has been temporarily stopped and restarted during inching or retraction

■ Parameters

Conditions	Range	Unit
Time	0.0 ～ 9.9	sec
Wire speed	1 ～ 9999	cm/min

■ Example of screen display

```
RTC[W1,2.0s,  6%]        FN411:Retract
```

[ ] shows the welder number, time, and wire speed from the left.

See
ICH ;Inching (FN410)

Function Commands (FN Codes)

Command name	GS
FN code	412
Title name	Gas ON
General description	Starts to output the shield gas.

■ General description

The robot starts to output the shield gas.
Prior to starting welding, the robot normally performs pre-flow for the duration which was set in the welding constants ("pre-flow time"). However, pre-flow can be started from any position by teaching the following commands.

The robot performs pre-flow from the position of the operation command that was taught immediately before the gas ON command (GS). When automatic operation is temporarily stopped during pre-flow initiated by the gas ON command (GS), pre-flow is suspended but it is then resumed after operation has been restarted.

■ Parameters

Parameter No.1	Welder No.	This specifies the welder number (1 to 12) to be controlled to output the shield gas.

■ Example of screen display

```
GS[W1]                   FN412:Gas ON
```

See
GE; Gas OFF(FN413)

Command name	GE
FN code	413
Title name	Gas OFF
General description	Stops the shield gas.

■ General description

The robot stops the shiled gas performed to start with Gas ON command (GS).

■ Parameters

Parameter No.1	Welder No.	This specifies the welder number (1 to 12) to be controlled to stop the shield gas.

■ Example of screen display

```
GE[W1]                    FN413:Gas OFF
```

See
GS; GasON(FN412)

Command name	AS
FN code	414
Title name	Arc start
General description	Starts arc welding with the specified conditions.

■ General description

This command enables to start Arc welding with the specified conditions. If it has been already started, the welding condition is changed to the specified one.

The welding start position is a move command shortly before AS(FN414). To select the command, designate the FN code “414” or press F key <AS>.

■ Parameters

●Control parameters except welding conditions

Conditions	Range
Welder	When multiple welders are connected to a single controller, select a target welder to start welding.
AS Cond. File	When designating the welding condition directly by figures : “0” When designating the welding condition by files : File No. (1～999) Enter the number and press [Enter]→F11<Input value> key. Then, the conditions stored in the previously created condition files are to be called. If you entered the file number not created yet, the initial conditions are to be called. Also, called conditions can be modified. If writing after modification, the modified details are reflected in the files. If newly creating the files, they will be stored in the internal memory.
Retry cond. no.	Arc retry function is currently under development. Enter“0” and then, the standard Arc retry function will be performed.

●Welding conditions

Welders can be divided into 3 types as follows, and the welding conditions that must be set differ depending on the type. (The type of welding power supply used is set ahead of time in the welding characteristics data.)

- Individual adjustments welder … This type of welder controls the current and voltage separately.
- Unified adjustments welder … This type of welder automatically outputs the voltage level which is optimally suited to the current value.
- TIG welder … This type of welder is used to perform TIG welding.

Type of welder	Conditions	Range	Unit
Individual adjustment welder	Welding current	1 ～ rated.	A
	Welding voltage	0.1 ～ rated.	V
	Welding speed	1 ～ 999	cm/min
Unified adjustment welder	Welding current	1 ～ rated.	A
	Voltage tuning	-100 ～ +100	%
	Welding speed	1 ～ 999	cm/min
TIG welder	Welding current	1 ～ rated.	A
	Filler speed	1 ～ rated.	cm/min
	Welding speed	1 ～ 999	cm/min

Checkpoints for setting the conditions

Concerning the setting range of the welding speed

Although any speed from 1 to 999 cm/min. can be set as the welding speed, the maximum speed is actually 600 cm/min. or so. Since this is the maximum operable speed and not a speed which can be used for actual welding, it must be adjusted to ensure optimum welding.

Voltage tuning

The voltage tuning amount is the amount by which the voltage is adjusted to the voltage command value which is automatically output by the welding power supply. For instance, when this is set to "+10%," the voltage will be slightly higher than the voltage level that should be output. Conversely, when it is set to "–10%," the voltage will be slightly lower. Set the fine adjustment amount to establish the appropriate condition.

Pre-flow time

The pre-flow time cannot be set using the arc start command (AS). Set it by selecting [Arc constant] – [Constant of Weld] – [Pre-flow time].

■ Example of screen display

```
AS[W1,OFF,0,150A, 21.0V, 80cm/m, →]
```

[ ] shows Welder No., Condition file No. (OFF indicates no number is specified), Arc retry condition No., and welding conditions (Current/Voltage/Speed) from the left.

See
AE; Arc end (FN415)

Command name	AS
FN code	414
Title name	Arc start
General description	Starts arc welding with the specified conditions.

■ **General description**

This command enables to start Arc welding with the specified conditions. If it has been already started, the welding condition is changed to the specified one. The welding start position is a move command shortly before AS(FN414). To select the command, designate the FN code “414” or press F key <AS>.

■ **Parameters**

● **Parameters other than welding conditions**

Conditions	Range
Welder	This specifies the target welder when a multiple number of welders are connected. It need not be set if only one welder is being used.
AS Cond. File	"0" is selected if the welding conditions are to be specified directly using numbers; a file number (1 to 999) is selected if they are to be specified by a file. Enter the number and press [Enter]→F11<Input value> key. Then, the conditions stored in the previously created condition files are to be called. If you entered the file number not created yet, the initial conditions are to be called. Also, called conditions can be modified. If writing after modification, the modified details are reflected in the files. If newly creating the files, they will be stored in the internal memory.
Retry cond. no.	Arc retry function is currently under development. Enter“0” and then, the standard Arc retry function will be performed.

●**Welding conditions**

Conditions	Range	Unit
Current condition	Wire Speed / Current	-
Welding current (When the Current condition is "Current".)	1 ～ rated.	A
Wire speed (When the Current condition is "Wire Speed".)	1 ～ 9999	cm/min.
Welding voltage (When Individual adjusting)	0.1 ～ rated.	V
Arc length tuning (When Unified adjusting)	-100 ～ +100	
Welding speed	1 ～ 999	cm/min.
Arc characteristic	-100 ～ +100	%
Slow down speed	1 ～ 9999	cm/min.
Start current	1 ～ 600	A
Hot start time	10 ～ 990	msec
Hot start voltage	0.0 ～ 9.9	V

※ Since incorrect settings may cause the poor welding quality, some of the parameters above may not be displayed depending on the operator qualification.

Checkpoints for setting the conditions

Concerning the setting range of the welding speed

Although any speed from 1 to 999 cm/min. can be set as the welding speed, the maximum speed is actually 600 cm/min. or so. Since this is the maximum operable speed and not a speed which can be used for actual welding, it must be adjusted to ensure optimum welding.

Arc length tuning

When the CPDRA-351/501 is used as a unified adjustment power supply, the optimal voltage for the current setting is output automatically. The "voltage adjustment value" is the value for increasing or reducing the voltage which is output automatically. Setting a "+" value yields a voltage which is on the high side; conversely, setting a "–" value yields a voltage which is on the low side.

Arc characteristic

The arc characteristics value is expressed in the form of a numerical value which enables the arc hardness or softness to be set. When this value is increased gradually in the "+" direction, a concentrated hard arc is obtained; conversely, when it is reduced gradually in the "–" direction, an expansive soft arc is obtained.

A "+" value tends to minimize the arc heat and is thus ideal for upward or sideways welding.

A "–" value tends to minimize spatter.

Bear in mind that setting an excessively high value will cause the welding to become unstable.

Slow down speed/ Start current/ Hot start time/ Hot start voltage

These conditions are supposed to be automatically set to their optimum values based on the input welding current (or wire feed rate). Do not change these default values except inevitability. When obliged to change it by necessity, perform the fine adjustment carefully based on the optimum value. Major change may cause the welding failure, abnormality and the poor quality.

Pre-flow time

The pre-flow time cannot be set using the arc start command (AS). Set it by selecting [Arc constants setting] – [Welding constants] – [Pre-flow time].

■ **Example of screen display**

```
AS[W1,OFF,0,150A, 21.0V, 80cm/m, →]
```

[ ] shows Welder No., Condition file No. (OFF indicates no number is specified), Arc retry condition No., and welding conditions (Current/Voltage/Speed) from the left.

See
AE; Arc end (FN415)

Command name	AS
FN code	414
Title name	Arc start
General description	Starts arc welding with the specified conditions.

■ **General description**

This command enables to start Arc welding with the specified conditions. If it has been already started, the welding condition is changed to the specified one.

The welding start position is a move command shortly before AS(FN414). To select the command, designate the FN code “414” or press F key <AS>.

■ **Parameters**

After selecting the Arc start instructions (AS), specify the “Arc start control conditions” first. “Arc start control conditions” are the ones displayed on the page 1. Specify the “Arc start conditions” on the page 2 and after.

● **Arc start control conditions**

Conditions	Explanation
Welder	When multiple welders are connected to a single controller, select a target welder to start welding.
AS Cond. File	When designating the welding condition directly by figures : “0” When designating the welding condition by files : File No. (1～999) Enter the number and press [Enter]→F11<Input value> key. Then, the conditions stored in the previously created condition files are to be called. If you entered the file number not created yet, the initial conditions are to be called. Also, called conditions can be modified. If writing after modification, the modified details are reflected in the files. If newly creating the files, they will be stored in the internal memory.
Retry cond. no.	Arc retry function is currently under development. Enter“0” and then, the standard Arc retry function will be performed.
Welding process	This specifies the welding process. The items displayed here are the ones registered as the welding characteristics data.
Current cond.	This gives the alternative of “current” or “wire feed rate” for the welding condition.
Slope cond.	This gives the alternative for the slope control at Arc start of “time designation” (performing at stop) or “distance designation” (performing with move).
Welding control	Normally, this is fixed at “Standard”. And, even if the optional software “Synchro MIG” is set, both “Synchro” and “FC” can be selected. To perform the Synchro MIG welding, select “Synchro” and to perform the FC-MIG welding, select “FC2.

● **Arc start conditions**

Conditions	Range	Unit	DC Pulse	DC Wave Pulse	AC Pulse	AC Wave Pulse
Welding current (When "Current cond." is "Current.")	1～200	A	○	○	○	○
Wire speed (When "Current cond." is "Wire speed".)	1 ～ 9999	cm/min				
Arc length tuning	-100～100	-	○	○	○	○
Welding speed	1～999	cm/min	○	○	○	○
Arc characteristic	-10～10	-	○	○	○	○
Wave frequency	0.5～32.0	Hz	—	○	—	○
Penetration adjust.	-150～150	-	—	—	○	○
Slow down speed	1～9999	cm/min	○	○	○	○
Start current	1～600	A	○	○	○	○
Slope time (When "Slope cond." is "Time".)	0.0～9.9	sec.	○	○	○	○
Slope distance (When "Slope cond." is "Distance".)	0～99	mm				
Initial current (When "Current cond." is "Current".)	1～rated	A	○	○	○	○
Initiall wire speed (When "Current cond." is "Wire speed".)	1～9999	cm/min				
Initial arc length tuning	-100～100	-	○	○	○	○
Base current	20～200	A	○	○	○	○
Standup time	0.4～3.0	m sec	○	○	○	○
Falling time	0.4～3.0	m sec	○	○	○	○
Peak current	20～600	A	○	○	○	○
Peak time	0.4～3.0	m sec	○	○	○	○
EN current	20～300	A	—	—	○	○
EN time	0.0～30.0	m sec	—	—	○	○

○ : Item to be set — : Item not to be set

Checkpoints for setting the conditions

Concerning the welding current and wire speed

The average current (or average feed speed) during pulsed welding serves as the welding current (or wire feed speed) which is initially input as the welding condition. The base current, peak current and other pulse conditions for pulsed welding

are automatically calculated on the basis of the welding current (or wire feed speed) which has been input.

Arc length tuning

Since the CPDACA-201 is a unified adjustment welder, the optimum voltage for the current which has been set is output automatically. The "arc length fine adjustment" is an adjustment value for increasing or reducing the voltage which is output automatically. For example, when it is set to +5, the output voltage is increased by approximately 0.5V.

Concerning the setting range of the welding speed

Although any speed from 1 to 999 cm/min. can be set as the welding speed, the maximum speed is actually 600 cm/min. or so. Since this is the maximum operable speed and not a speed which can be used for actual welding, it must be adjusted to ensure optimum welding.

EN current and EN time

The EN current and EN time are parameters for controlling the heat input to the workpiece during AC pulsed welding. "EN" stands for "electrode negative." Outputting EN during AC pulsed welding makes the penetration in the base metal shallower and increases the amount of the wire melted. However, when adjusting the penetration in the base metal, do not change the EN current or EN time but change the penetration adjustment conditions.

Concerning the pulse conditions

To adjust the pulse conditions, proceed by changing "Arc characteristics," "Wave frequency" and "Penetration adjustment" .

Pulse conditions "Base current", "Standup time", "Falling time", "Peak current ", and "Peak time" can be set provided that the operator qualifications level is set to Expert or above. However, the welding quality may deteriorate if they are set carelessly. In principle, therefore, do not change these conditions.

To prevent the conditions from being changed in error, select [Arc Constant] – [Constant of weld] – [Pulse condition], and set "Recommendation" ("Teaching" or "Recommendation" can be set, and "Teaching" is set initially.)

If "Recommendation" is set, these values will always be used when welding is executed even if the pulse conditions are changed in error, thus safeguarding the welding quality from deterioration.

Arc characteristic

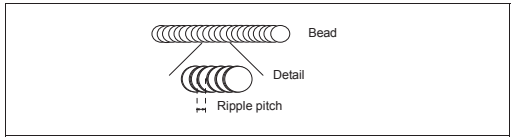
The pulsed arc characteristics are a parameter used for adjusting the pulse rise time and pulse fall time internally. When its value is increased, an expansive soft arc is obtained; conversely, when it is reduced, a concentrated hard arc is obtained.

Penetration adjust.

The penetration adjustment is a parameter for adjusting the EN time. Increasing its numerical value makes for a shallower penetration; conversely, reducing it makes for a deeper penetration.

Wave frequency

The wave frequency is a parameter used for adjusting the ripple pitch of beads which are shaped like fish scales and which occur in the DC wave pulsed method. The ripple pitch can be adjusted as desired by a combination of the welding speed and wave frequency. Increasing the wave frequency while keeping the welding speed fixed reduces the pitch width; conversely, reducing it increases the pitch width.



Pre-flow time

The pre-flow time cannot be set using the arc start command (AS). Set it by selecting [Arc constant] – [Constant of Weld] – [Pre-flow time].

Slop control

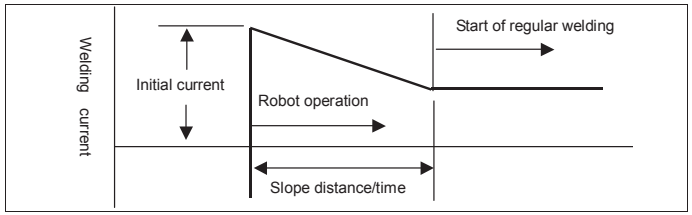
Slope control involves changing the conditions (welding current, welding voltage) in the form of a slope (that is to say, gradually) rather than switching the welding conditions straight away to the specified values. It enables the sputter amount, weld defects, etc. that occur especially when the conditions are switched to be reduced. It can be used when starting the welding, changing the conditions or ending the welding.

Slope control when welding is started

The figure below shows the slope control that is initiated from the initial welding condition settings to the regular welding conditions.

It shows what happens with the welding current although it is the same for the welding voltage.

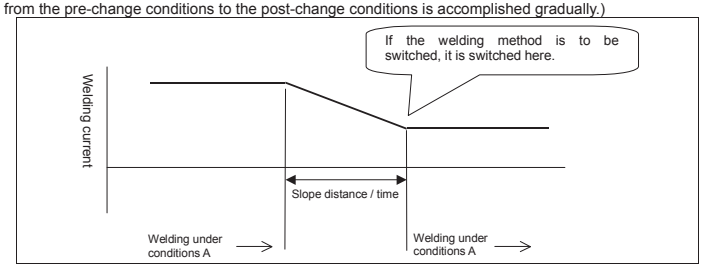
The control can be specified as a distance or time for the slope section.



Slope control when conditions are changed

When conditions are changed, slope control is conducted from the pre-change conditions to the post-change conditions.

The initial current and initial voltage set as the slope conditions are not used when the conditions are changed. (Switching from the pre-change conditions to the post-change conditions is accomplished gradually.)



■ **Example of screen display**

```
AS[W1,OFF,0,150A, 21.0V, 80cm/m, →]
```

[ ] shows Welder No., Condition file No. (OFF indicates no number is specified), Arc retry condition No., and welding conditions (Current/Voltage/Speed) from the left.

See
AE; Arc end (FN415)

Function Commands (FN Codes)

Command name	AS
FN code	414
Title name	Arc start
General description	Starts arc welding with the specified conditions.

■ General description

This command enables to start Arc welding with the specified conditions. If it has been already started, the welding condition is changed to the specified one.

The welding start position is a move command shortly before AS(FN414). To select the command, designate the FN code “414” or press F key <AS>.

■ Parameters

After selecting the Arc start instructions (AS), specify the “Arc start control conditions” first. “Arc start control conditions” are the ones displayed on the page 1. Specify the “Arc start conditions” from on and after the page 2.

● Arc start control conditions

Conditions	Explanation
Welder	When multiple welders are connected to a single controller, select a target welder to start welding.
AS Cond. File	When designating the welding condition directly by figures : “0” When designating the welding condition by files : File No. (1～999) Enter the number and press [Enter]→F11<Input value> key. Then, the conditions stored in the previously created condition files are to be called. If you entered the file number not created yet, the initial conditions are to be called. Also, called conditions can be modified. If writing after modification, the modified details are reflected in the files. If newly creating the files, they will be stored in the internal memory.
Retry cond. no.	Arc retry function is currently under development. Enter“0” and then, the standard Arc retry function will be performed.
Welding process	This specifies the welding process. The items displayed here are the ones registered as the welding characteristics data.
Current cond.	This gives the alternative of “current” or “wire feed rate” for the welding condition.
Slope cond.	This gives the alternative for the slope control at Arc start of “time designation” (performing at stop) or “distance designation” (performing with move).
Welding control	Normally, this is fixed at “Standard”. And, even if the optional software “Synchro MIG” is set, both “Synchro” and “FC” can be selected. To perform the Synchro MIG welding, select “Synchro” and to perform the FC-MIG welding, select “FC2.

● Arc start conditions

Conditions	Range	Unit	DF	DC Pulse	DC Wave Pulse
Welding current (When "Current cond." is "Current.")	1 ～ rated	A	○	○	○
Wire speed (When "Current cond." is "Wire speed".)	1 ～ 9999	cm/min			
Welding voltage (When individual adjustment)	0.1 ～ rated	V	○	○	○
Arc length tuning (When unified adjustment)	-100 ～ 100	-			
Welding speed	1 ～ 999	cm/min	○	○	○
Arc characteristic	-10 ～ 10	—	—	○	○
Wave frequency	0.5 ～ 32.0	Hz	—	—	○
Slow down speed	1 ～ 9999	cm/min	○	○	○
Start current	1 ～ 600	A	—	○	○
Slope time (When "Slope cond." is "Time".)	0.0～9.9	sec.	○	○	○
Slope distance (When "Slope cond." is "Distance".)	0～99	mm			
Initial current (When "Current cond." is "Current".)	1～rated	A	○	○	○
Initiall wire speed (When "Current cond." is "Wire speed".)	1～9999	cm/min			
Initial welding voltage (When individual adjustment)	-100～100	-	○	○	○
Initial arc length tuning (When unified adjustment)	0.1 ～ rated	V			
Base current	20 ～ 200	A	—	○	○
Standup time	0.1 ～ 3.0	m min	—	○	○
Falling time	0.1 ～ 3.0	m min	—	○	○
Peak current	20 ～ 600	A	—	○	○
Peak time	0.1 ～ 3.0	m min	—	○	○

○ : Item to be set — : Item not to be set

Checkpoints for setting the conditions

Concerning the welding current and wire speed

The average current (or average feed speed) during pulsed welding serves as the welding current (or wire feed speed) which is initially input as the welding condition. The base current, peak current and other pulse conditions for pulsed welding are automatically calculated on the basis of the welding current (or wire feed speed) which has been input.

Arc length tuning

Since the CPDPAS-501 is a unified adjustment welder, the optimum voltage for the current which has been set is output automatically. The "arc length fine adjustment" is an adjustment value for increasing or reducing the voltage which is output automatically. For example, when it is set to +5, the output voltage is increased by approximately 0.5V.

Concerning the setting range of the welding speed

Although any speed from 1 to 999 cm/min. can be set as the welding speed, the maximum speed is actually 600 cm/min. or so. Since this is the maximum operable speed and not a speed which can be used for actual welding, it must be adjusted to ensure optimum welding.

Concerning the pulse conditions

To adjust the pulse conditions, proceed by changing "Arc characteristics" and "Wave frequency" .

Pulse conditions "Base current", "Standup time", "Falling time", "Peak current ", and "Peak time" can be set provided that the operator qualifications level is set to Expert or above. However, the welding quality may deteriorate if they are set carelessly. In principle, therefore, do not change these conditions.

To prevent the conditions from being changed in error, select [Arc Constant] – [Constant of weld] – [Pulse condition], and set "Recommendation" ("Teaching" or "Recommendation" can be set, and "Teaching" is set initially.)

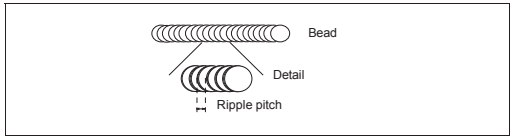
If "Recommendation" is set, these values will always be used when welding is executed even if the pulse conditions are changed in error, thus safeguarding the welding quality from deterioration.

Arc characteristic

The pulsed arc characteristics are a parameter used for adjusting the pulse rise time and pulse fall time internally. When its value is increased, an expansive soft arc is obtained; conversely, when it is reduced, a concentrated hard arc is obtained.

Wave frequency

The wave frequency is a parameter used for adjusting the ripple pitch of beads which are shaped like fish scales and which occur in the DC wave pulsed method. The ripple pitch can be adjusted as desired by a combination of the welding speed and wave frequency. Increasing the wave frequency while keeping the welding speed fixed reduces the pitch width; conversely, reducing it increases the pitch width.



Pre-flow time

The pre-flow time cannot be set using the arc start command (AS). Set it by selecting [Arc constant] – [Constant of Weld] – [Pre-flow time].

Slop control

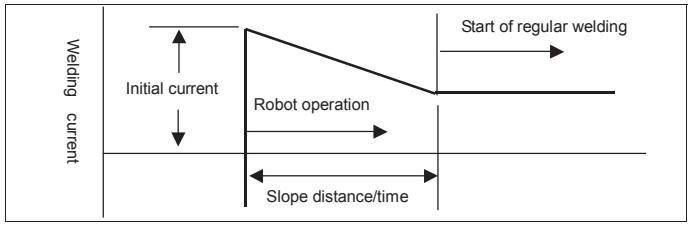
Slope control involves changing the conditions (welding current, welding voltage) in the form of a slope (that is to say, gradually) rather than switching the welding conditions straight away to the specified values. It enables the sputter amount, weld defects, etc. that occur especially when the conditions are switched to be reduced. It can be used when starting the welding, changing the conditions or ending the welding.

Slope control when welding is started

The figure below shows the slope control that is initiated from the initial welding condition settings to the regular welding conditions.

It shows what happens with the welding current although it is the same for the welding voltage.

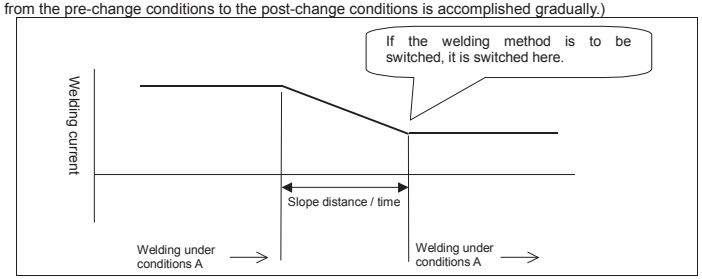
The control can be specified as a distance or time for the slope section.



Slope control when conditions are changed

When conditions are changed, slope control is conducted from the pre-change conditions to the post-change conditions.

The initial current and initial voltage set as the slope conditions are not used when the conditions are changed. (Switching from the pre-change conditions to the post-change conditions is accomplished gradually.)



■ **Sample of screen display**

```
AS[W1,OFF,0,150A, 21.0V, 80cm/m, →]
```

[ ] shows Welder No., Condition file No. (OFF indicates no number is specified), Arc retry condition No., and welding conditions (Current/Voltage/Speed) from the left.

See
AE; Arc end (FN415)

Function Commands (FN Codes)

Command name	AS
FN code	414
Title name	Arc start
General description	Starts arc welding with the specified conditions.

■ **General description**

This command enables to start Arc welding with the specified conditions. If it has been already started, the welding condition is changed to the specified one.

The welding start position is a move command shortly before AS(FN414). To select the command, designate the FN code “414” or press F key <AS>.

■ **Parameters**

After selecting the Arc start instructions (AS), specify the “Arc start control conditions” first. “Arc start control conditions” are the ones displayed on the page 1. Specify the “Arc start conditions” from on and after the page 2.

● **Arc start control conditions**

Conditions	Explanation	
Welder	This specifies the target welder when a multiple number of welders are connected. It need not be set if only one welder is being used.	
AS Cond. File	"0" is selected if the welding conditions are to be specified directly using numbers; a file number (1 to 999) is selected if they are to be specified by a file. Enter the number and press [Enter]→F11<Input value> key. Then, the conditions stored in the previously created condition files are to be called. If you entered the file number not created yet, the initial conditions are to be called. Also, called conditions can be modified. If writing after modification, the modified details are reflected in the files. If newly creating the files, they will be stored in the internal memory.	
Retry cond. no.	Arc retry function is currently under development. Enter“0” and then, the standard Arc retry function will be performed.	
Output current	DC	This is for welding using DC waveforms.
	AC	This is for welding using AC waveforms.
	DC pulse (*1)	This is for welding using DC pulsed waveforms.
	AC pulse (*1)	This is for welding using AC pulsed waveforms.
	AC-DC (*1)	This is for hybrid welding (a method that switches between AC and DC in the specified cycle).
	DC synchro (*2)	This is for welding while synchronizing the peak current (DC) with the end point of weaving.
	AC synchro (*2)	This is for welding while synchronizing the peak current (AC) with the end point of weaving.
	AC-DC synchro (*2)	This is for TIG welding while synchronizing the AC output or DC output timing with the end point of weaving.
Wire control	OFF	The filler is not fed. This is selected for TIG fusion welding.
	ON	This is for feeding the filler at a constant speed.
	Pulse (*3)	This is for feeding the filler at a pulse rate.
Starting current	Auto	Specifying the starting current by “strong” or “week”.
	Manual	Specifying the starting current by figures.
Preheating	ON	Preheating control is performed.
	OFF	Preheating control is not performed.
Slope control	ON	Up-slope control is performed.
	OFF	Up-slope control is not performed.
AC wave	Standard	This enables welding across a wide range from thin sheets to thick sheets.
	Hard	・ This enables concentrated arcs as with DC to be produced. ・ It is useful for thin-sheet fillet welding, etc.
	Soft	・ This enables soft arcs to be produced. ・ It is useful for thin-sheet butt welding, etc.

(*1) These modes require the optional "TIG pulsed welding" software.
(*2) The optional board “RS-422”, the connection cable and other items are required.
(*3) An AC servo feeder (L7142 for steel, L7143 for aluminum) is required to feed the pulses. This cannot be selected when AC or DC is selected as the current output.

●Welding start conditions

Classification	Condition		Range	Unit
Welding condition	Current	Welding current	1 ～ 300	A
		Base current	4 ～ 300	A
		Peak current	4 ～ 300	A
	Filler	Wire speed	0 ～ rated	cm/min
		Base wire speed	0 ～ rated	cm/min
		Peak wire speed	0 ～ rated	cm/min
		AC wire speed	0 ～ rated	cm/min
		DC wire speed	0 ～ rated	cm/min
	Welding speed		1 ～ 999	cm/min
	AC frequency		50 ～ 200	Hz
	Cleaning width		-10 ～ +10	-
	Pulse control	Frequency	0.5 ～ 500.0	Hz
		Pulse ratio	1 ～ 99	%
	AC-DC swithing	AC-DC frequency	0.5 ～ 30.0	Hz
		AC-DC ratio	1 ～ 99	%
	Weaving	Weaving signal	1pulse / 2pulses	-
		Weaving edge	AC/DC	-
		Phase adjust.	0.00 ～ 9.99	sec
Start current	Starting current	Auto setting	Weak/Strong	-
		Manual setting	4 ～ 300	A
	Starting current time	Auto setting	Set automatically	-
		Manual setting	1 ～ 20	m min
Filler timing adjustment	Wire timing (Standup)		0.00 ～ 0.99	sec
	Wire timing (Falling)		0.00 ～ 0.99	sec
Pre- heating	Preheating time		0.0 ～ 30.0	m min
	Preheating current		4 ～ 300	A
	Preheating speed		0 ～ rated	cm/min
Up slope	Slope time or Slope distance		0.0 ～ 9.9	sec
			0 ～ 99	mm
	Robot stop time		0.0 ～ 9.9	sec
	Wire delay time		0.0 ～ 9.9	sec

Checkpoints for setting the conditions

Preheating

This is a process for stabilizing arc start. A constant preheating current is output while the robot is stopped. (The start current is output for an instant at the very start of this process.) The preheating current does not become a pulsive current even if DC pulsed or AC pulsed is selected as the welding mode.

Start current

This process outputs a constant high current at the beginning in order for preheating to be conducted smoothly. Specify "Strong" or "Weak" at the teaching stage. At the "Strong" setting, the start current is 200 A. (If the current for regular welding is higher than 200 A, this higher value is used.)

Specify "Weak" for thin sheet welding. At the "High" setting, burn-through may occur.

Up-slope

This process is for stabilizing the arc start, and it keeps raising the output current step by step. It performs the 2-step processing described below.

① Upon completion of preheating, up-slope is started at the arc start point with the torch stopped.
② Next, up-slope is conducted while the torch is being moved.

Wire delay time

This is the time by which the wire feed start timing is delayed from the up-slope start.

The delay time is set using a time shorter than the up-slope time. If the delay time is longer than the up-slope time, wire feed will not start even when up- slope is completed.

Pulse ratio

The pulse ratio is expressed as a percentage of the peak current duration to one period, and it is defined by the following equation.

Cleaning width

The cleaning width can be set in the case of AC welding. Any value from -10 to +10 can be set. The following symptoms raise when the numerical value of the cleaning width is changed.

Symptom \ Setting value	-10 ⟷ +10	
	Minus direction	Plus direction
Cleaning width	Becomes narrower	Becomes wider
Penetration depth	Becomes deeper	Becomes shallower
Electrode wear	Decreases	Increases

Wire timing

The current output and filler feed timing can be adjusted. This parameter is set in cases where the response of equipment relating to filler feed is to be compensated or filler is to be inserted for an instant within the peak current period, for example. Set the following conditions for the rise and fall of the output waveforms.

AC-DC frequency/ AC-DC ratio

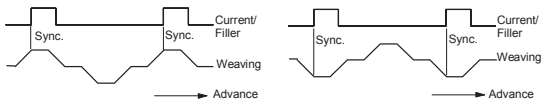
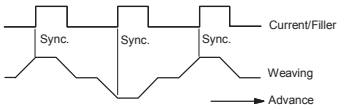
AC/DC hybrid welding involves the output of AC (alternating current) and DC (direct current) alternately. Set the following conditions when AC/DC hybrid has been selected as the welding mode.

The DC ratio is expressed as a percentage of the AC output duration to one period, and it is defined by the following equation.

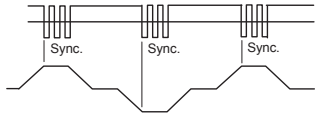
When performing Syncho-TIG

The type of welding in which the weaving period is synchronized with the timing of the current and filler feed output pulses is called synchro TIG welding. The following 3 synchro TIG welding modes are provided.

Either the 1-pulse or 2-pulse method can be selected as the way to synchronize with the weaving.

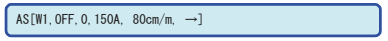
Synchronization method	Details
1 pulse	The current and filler feed are output once (=1 pulse) during a single weaving cycle. The amplitude for peak welding is based on the weaving start phase (oscillation start direction). In other words, when weaving is to start from the left amplitude, the pulses are output at the left amplitude side. Conversely, when weaving is to start from the right amplitude, the pulses are output at the right amplitude side.  (a) Left amplitude side (b) Right amplitude side
2 pulses	The current and filler feed are output twice (=2 pulses) during a single weaving cycle. 

In the case of AC/DC synchro welding, specify whether the above synchronization timing is to be implemented with the AC output or DC output.



In this case, the AC output is synchronized by two pulses.

■ Example of screen display



[ ] shows Welder No., Condition file No. (OFF indicates no number is specified), Arc retry condition No., and welding conditions (Current//Speed) from the left.

See
AE; Arc end (FN415)

Command name	AS
FN code	414
Title name	Arc start
General description	Starts arc welding with the specified conditions.

■ General description

This command enables to start Arc welding with the specified conditions. The welding start position is a move command shortly before AS(FN414). To select the command, designate the FN code “414” or press F key <AS>.

■ Parameters

● Parameters other than welding conditions

Conditions	Description
Welder	When multiple welders are connected to a single controller, select a target welder to start welding.
AS Cond. file	When designating the welding condition directly by figures : “0” When designating the welding condition by files : File No. (1～999) Enter the number and press [Enter]→F11<Input value> key. Then, the conditions stored in the previously created condition files are to be called. If you entered the file number not created yet, the initial conditions are to be called. Also, called conditions can be modified. If writing after modification, the modified details are reflected in the files. If newly creating the files, they will be stored in the internal memory.
Retry cond. no.	Arc retry function is currently under development. Enter“0” and then, the standard Arc retry function will be performed.

●Welding conditions

Conditions	Range	Unit
Current cond.	Speed/ Current	-
Welding current (When "Current cond." is "Current".)	1 ～ rated	A
Wire speed (When "Current cond." is "Speed".)	1 ～ 9999	cm/min
Welding voltage (When individual adjustment)	0.1 ～ rated	V
Arc length tuning (When unified adjustment)	-100 ～ +100	-
Welding speed	1 ～ 999	cm/min
Arc characteristic	-99 ～ +99	%
Slow down speed	100 ～ 9999	cm/min
Start current	1 ～ 600	A

※ Since incorrect settings may cause the poor welding quality, some of the parameters above may not be displayed depending on the operator qualification.

Checkpoints for setting the conditions

Concerning the setting range of the welding speed

Although any speed from 1 to 999 cm/min. can be set as the welding speed, the maximum speed is actually 600 cm/min. or so. Since this is the maximum operable speed and not a speed which can be used for actual welding, it must be adjusted to ensure optimum welding.

Arc length tuning

Since the DM-350/500 is a unified adjustment welder, the optimum voltage for the current which has been set is output automatically. The "arc length fine adjustment" is an adjustment value for increasing or reducing the voltage which is output automatically. For example, when it is set to +5, the output voltage is increased by approximately 0.5V.

Arc characteristic

The arc characteristics value is expressed in the form of a numerical value which enables the arc hardness or softness to be set. When this value is increased gradually in the "+" direction, a concentrated hard arc is obtained; conversely, when it is reduced gradually in the "–" direction, an expansive soft arc is obtained.

A "+" value tends to minimize the arc heat and is thus ideal for upward or sideways welding.

A "–" value tends to minimize spatter.

Bear in mind that setting an excessively high value will cause the welding to become unstable.

Slow down speed/ Start current

These conditions are supposed to be automatically set to their optimum values based on the input welding current (or wire feed rate). Do not change these default values except inevitability. When obliged to change it by necessity, perform the fine adjustment carefully based on the optimum value. Major change may cause the welding failure, abnormality and the poor quality.

Pre-flow time

The pre-flow time cannot be set using the arc start command (AS). Set it by selecting [Arc constant] – [Constant of Weld] – [Pre-flow time].

■ Example of screen display

```
AS[W1,OFF,0,150A, 21.0V, 80cm/m, →]
```

[ ] shows Welder No., Condition file No. (OFF indicates no number is specified), Arc retry condition No., and welding conditions (Current/Voltage/Speed) from the left.

See
AE; Arc end (FN415)

Command name	AE
FN code	415
Title name	Arc end
General description	Terminates arc welding with the specified conditions.

■ General description

This command enables to end Arc welding, performing the crater process with the specified conditions. The welding end position is a move command shortly before AE(FN415). To select the command, designate the FN code “415” or press F key <AE>.

■ Parameters

● Parameters other than welding conditions

Conditions	Details
Welder	This specifies the target welder when a multiple number of welders are connected. It need not be set if only one welder is being used.
AS Cond. File	"0" is selected if the welding conditions are to be specified directly using numbers; a file number (1 to 999) is selected if they are to be specified by a file. Enter the number and press [Enter]→F11<Input value> key. Then, the conditions stored in the previously created condition files are to be called. If you entered the file number not created yet, the initial conditions are to be called. Also, called conditions can be modified. If writing after modification, the modified details are reflected in the files. If newly creating the files, they will be stored in the internal memory.

●Welding end conditions

The welding conditions must be set different depending on the type of welders specified in the welding characteristics data (Individual adjustment・Unified adjustment・TIG).

Type of welder	Conditions	Range	Unit
Individual adjustment welder	Crater current	1～rated	A
	Crater voltage	0.1 ～ rated	V
	Crater time	0.0 ～ 9.9	sec
	Postflow time	0.0 ～ 9.9	sec
Unified adjustment welder	Crater current	1～rated	A
	Voltage tuning	-100 ～ +100	%
	Crater time	0.0 ～ 9.9	sec
	Postflow time	0.0 ～ 9.9	sec
TIG welder	Crater current	1～rated	A
	Retract time	0.0 ～ 9.9	sec
	Crater time	0.0 ～ 9.9	sec
	Postflow time	0 ～ 99	sec
	Retract speed	0 ～ 100	%

Checkpoints for setting the conditions

Crater conditions

Crater process denotes the process to reform the bead shape after welding performance by applying the arc for “a certain period of time”. At this time, the arc must be applied by “smaller condition than the real welding”.

The condition “smaller condition than the real welding” is called “*crater current/ crater voltage*” (voltage fine adjustment) and “a certain period of time” to apply the arc, “*crater time*”.

Postflow time

“Postflow time” is the action to keep supplying the gas for a certain period of time to prevent the welded part from being oxidized.

■ Example of screen display

```
AE[W1,OFF,130A, 18.0V, 0.1s,0.1s]
```

[ ] shows Welder No., Condition file No. (OFF indicates no number is specified), and arc end conditions (Current/ Voltage/ Crater time/ Postflow time) from the left.

See
AS; Arc start (FN414)

Function Commands (FN Codes)

Command name	AE
FN code	415
Title name	Arc end
General description	Terminates arc welding with the specified conditions.

■ General description

This command enables to end Arc welding, performing the crater process with the specified conditions. The welding end position is a move command shortly before AE(FN415). To select the command, designate the FN code “415” or press F key <AE>.

■ Parameters

●Parameters other than welding conditions

Conditions	Details
Welder	This specifies the target welder when a multiple number of welders are connected. It need not be set if only one welder is being used.
AS Cond. file	"0" is selected if the welding conditions are to be specified directly using numbers; a file number (1 to 999) is selected if they are to be specified by a file. Enter the number and press [Enter]→F11<Input value> key. Then, the conditions stored in the previously created condition files are to be called. If you entered the file number not created yet, the initial conditions are to be called. Also, called conditions can be modified. If writing after modification, the modified details are reflected in the files. If newly creating the files, they will be stored in the internal memory.

●Welding end conditions

Conditions	Range	Unit
Current cond.	Speed/ Current	-
Welding current (When the Current condition is "Current".)	1 ～ rated.	A
Wire speed (When the Current condition is "Wire Speed".)	1 ～ 9999	cm/min
Welding voltage (When Individual adjusting)	0.1 ～ rated.	V
Arc length tuning (When Unified adjusting)	−100 ～ +100	-
Crater time	0.0 ～ 9.9	sec
Postflow time	0.0 ～ 9.9	sec
Arc characteristic	-100 ～ +100	%
Burnback time	10 ～ 990	msec

※ Since incorrect settings may cause the poor welding quality, some of the parameters above may not be displayed depending on the operator qualification.

Checkpoints for setting the conditions

Crater conditions

Crater condition denotes the process to reform the bead shape after welding performance by applying the arc for “a certain period of time”. The arc applied here is “less than the real welding”.

The condition “less than the real welding” is called “*crater current/ crater voltage*” (arc length fine adjustment) and likewise, “a certain period of time” to apply the arc, “*crater time*”.

Arc length tuning

When the CPDRA-351/501 is used as a unified adjustment power supply, the optimal voltage for the current setting is output automatically. The "voltage adjustment value" is the value for increasing or reducing the voltage which is output automatically. Setting a "+" value yields a voltage which is on the high side; conversely, setting a "–" value yields a voltage which is on the low side.

Arc characteristic

The arc characteristics value is expressed in the form of a numerical value which enables the arc hardness or softness to be set. When this value is increased gradually in the "+" direction, a concentrated hard arc is obtained; conversely, when it is reduced gradually in the "–" direction, an expansive soft arc is obtained.

A "+" value tends to minimize the arc heat and is thus ideal for upward or sideways welding.

A "–" value tends to minimize spatter.

Bear in mind that setting an excessively high value will cause the welding to become unstable.

Postflow time

“Postflow time” is to keep supplying the gas for a while to prevent the welded part from oxidized.

Burnback time

The optimum values of this parameter are set automatically depending on the specified welding current (or wire feed speed). Do not change this condition, basically. If changing, adjust the values slightly based on the optimum value. If making a big change, welding cannot be executed and welding error may occur, or welding quality may become worse.

■ Example of screen display

```
AE[W1,OFF,130A, 18.0V, 0.1s,0.1s]
```

[ ] shows Welder No., Condition file No. (OFF indicates no number is specified), and arc end conditions (Current/ Voltage/ Crater time/ Postflow time) from the left.

See
AS; Arc start (FN414)

Function Commands (FN Codes)

Command name	AE
FN code	415
Title name	Arc end
General description	Terminates arc welding with the specified conditions.

■ General description

This command enables to end Arc welding, performing the crater process with the specified conditions. The welding end position is a move command shortly before AE(FN415). To select the command, designate the FN code “415” or press F key <AE>.

■ Parameters

Set "Welding end control conditions" first, after selecting this command (AE).
"Welding end control conditions" are the conditions displayed on Page 1.
"Welding end conditions" are displayed on Page 2 and after.

●Welding end control conditions

Conditions	Details
Welder	This specifies the target welder when a multiple number of welders are connected. It need not be set if only one welder is being used.
AS Cond. file	"0" is selected if the welding conditions are to be specified directly using numbers; a file number (1 to 999) is selected if they are to be specified by a file. Enter the number and press [Enter]→F11<Input value> key. Then, the conditions stored in the previously created condition files are to be called. If you entered the file number not created yet, the initial conditions are to be called. Also, called conditions can be modified. If writing after modification, the modified details are reflected in the files. If newly creating the files, they will be stored in the internal memory.
Welding process	This is used to set the welding method to be used for the welding. The items displayed here are the welding methods which have been registered as the welding characteristics data.
Current cond.	This is used to select whether to specify the welding conditions by current or by wire feed speed.
Slope cond.	This is used to select whether to exercise slope control at the time of arc end with the robot stopped (in which case it is specified as a time) or while the robot is still advancing (in which case it is specified as a distance).

●Welding end conditions

Conditions	Range	Unit	DC Pulse	DC Wave Pulse	AC Pulse	AC Wave Pulse
Welding current (When "Current cond." is "Current.")	1～200	A	○	○	○	○
Wire speed (When "Current cond." is "Wire speed".)	1 ～ 9999	cm/min				
Arc length tuning	-100～100		○	○	○	○
Crater time	0.0～9.9	sec	○	○	○	○
Postflow time	0.0～9.9	sec	○	○	○	○
Arc characteristic	-10～10	-	○	○	○	○
Wave frequency	0.5～32.0	Hz	—	○	—	○
Penetration adjust.	-150～150	-	—	—	○	○
Burnback time	0～999	m sec	○	○	○	○
Burnback voltage	0.1～27.0	V	○	○	○	○
Slope time (When "Slope cond." is "Time".)	0.0～9.9	sec	○	○	○	○
Slope distance (When "Slope cond." is "Distance".)	0～99	mm				
Base current	20～200	A	○	○	○	○
Standup time	0.4～3.0	m sec	○	○	○	○
Falling time	0.4～3.0	m sec	○	○	○	○
Peak current	20～600	A	○	○	○	○
Peak time	0.4～3.0	m sec	○	○	○	○
EN current	20～300	A	—	—	○	○
EN time	0.0～30.0	m sec	—	—	○	○

○ : Item to be set — : Item not to be set

Checkpoints for setting the conditions

Concerning the welding current and wire speed

The average current (or average feed speed) during pulsed welding serves as the welding current (or wire feed speed) which is initially input as the welding condition. The base current, peak current and other pulse conditions for pulsed welding are automatically calculated on the basis of the welding current (or wire feed speed) which has been input.

When welding without a crater process

If performing welding without a crater process, set 0 as the crater time. In this case, note that the setting of welding current or wire feed speed must be the same as that for actual welding. Otherwise, the optimum value of burnback voltage may not be displayed properly when performing teaching.

Arc length tuning

Since the CPDACA-201 is a unified adjustment welder, the optimum voltage for the current which has been set is output automatically. The "arc length fine adjustment" is an adjustment value for increasing or reducing the voltage which is output automatically. For example, when it is set to +5, the output voltage is increased by approximately 0.5V.

EN current and EN time

The EN current and EN time are parameters for controlling the heat input to the workpiece during AC pulsed welding. "EN" stands for "electrode negative." Outputting EN during AC pulsed welding makes the penetration in the base metal shallower and increases the amount of the wire melted. However, when adjusting the penetration in the base metal, do not change the EN current or EN time but change the penetration adjustment conditions.

Arc characteristic

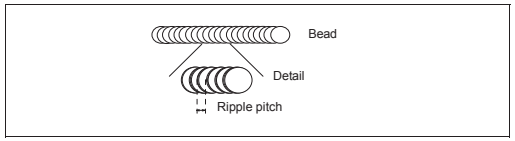
The pulsed arc characteristics are a parameter used for adjusting the pulse rise time and pulse fall time internally. When its value is increased, an expansive soft arc is obtained; conversely, when it is reduced, a concentrated hard arc is obtained.

Penetration adjust.

The penetration adjustment is a parameter for adjusting the EN time. Increasing its numerical value makes for a shallower penetration; conversely, reducing it makes for a deeper penetration.

Wave frequency

The wave frequency is a parameter used for adjusting the ripple pitch of beads which are shaped like fish scales and which occur in the DC wave pulsed method. The ripple pitch can be adjusted as desired by a combination of the welding speed and wave frequency. Increasing the wave frequency while keeping the welding speed fixed reduces the pitch width; conversely, reducing it increases the pitch width.



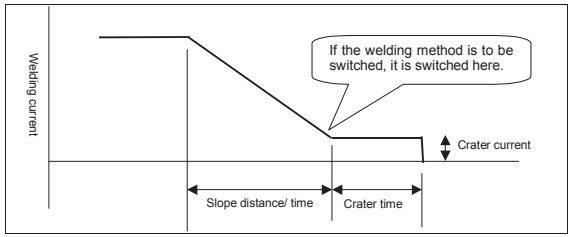
Slope control

Slope control involves changing the conditions (welding current, welding voltage) in the form of a slope (that is to say, gradually) rather than switching the welding conditions straight away to the specified values. It enables the sputter amount, weld defects, etc. that occur especially when the conditions are switched to be reduced. It can be used when starting the welding, changing the conditions or ending the welding.

Slope control when welding is ended

When welding is ended, slope control is conducted as the conditions are changed from the regular welding conditions to the crater conditions.

Even when crater treatment is not performed, the crater conditions (crater current, crater voltage) must be input if slope control is to be conducted.



■ **Example of screen display**

```
AE[W1, OFF, 130A,  18.0V,  0.1s, 0.1s]
```

[ ] shows Welder No., Condition file No. (OFF indicates no number is specified), and arc end conditions (Current/ Voltage/ Crater time/ Postflow time) from the left.

See
AS; Arc start (FN414)

Function Commands (FN Codes)

Command name	AE
FN code	415
Title name	Arc end
General description	Terminates arc welding with the specified conditions.

■ **General description**

This command enables to end Arc welding, performing the crater process with the specified conditions. The welding end position is a move command shortly before AE(FN415). To select the command, designate the FN code “415” or press F key <AE>.

■ **Parameters**

Set "Welding end control conditions" first, after selecting this command (AE).
"Welding end control conditions" are the conditions displayed on Page 1.
"Welding end conditions" are displayed on Page 2 and after.

●**Welding end control conditions**

Conditions	Details
Welder	This specifies the target welder when a multiple number of welders are connected. It need not be set if only one welder is being used.
AS Cond. file	"0" is selected if the welding conditions are to be specified directly using numbers; a file number (1 to 999) is selected if they are to be specified by a file. Enter the number and press [Enter]→F11<Input value> key. Then, the conditions stored in the previously created condition files are to be called. If you entered the file number not created yet, the initial conditions are to be called. Also, called conditions can be modified. If writing after modification, the modified details are reflected in the files. If newly creating the files, they will be stored in the internal memory.
Welding process	This is used to set the welding method to be used for the welding. The items displayed here are the welding methods which have been registered as the welding characteristics data.
Current cond.	This is used to select whether to specify the welding conditions by current or by wire feed speed.
Slope cond.	This is used to select whether to exercise slope control at the time of arc end with the robot stopped (in which case it is specified as a time) or while the robot is still advancing (in which case it is specified as a distance).

●**Welding end conditions**

Conditions	Range	Unit	DC	DC Pulse	DC Wave Pulse
Welding current (When "Current cond." is "Current.")	1 ～ rated	A	○	○	○
Wire speed (When "Current cond." is "Wire speed".)	1 ～ 9999	cm/min			
Welding voltage (When Individual adjusting)	0.1 ～ rated	V	○	○	○
Arc length tuning (When Unified adjusting)	-100 ～ 100	-			
Crater time	0.0 ～ 9.9	sec	○	○	○
Postflow time	0.0 ～ 9.9	sec	○	○	○
Arc characteristic	-10 ～ 10	-	—	○	○
Wave frequency	0.5 ～ 32.0	Hz	—	—	○
Penetration adjust.	0.1 ～ 50.0	V	△	○	○
Burnback time	10 ～ 999	m sec	○	○	○
Slope time (When "Slope cond." is "Time".)	0.0～9.9	秒	○	○	○
Slope distance (When "Slope cond." is "Distance".)	0～99	mm			
Base current	20 ～ 200	A	—	○	○
Standup time	0.1 ～ 3.0	m sec	—	○	○
Falling time	0.1 ～ 3.0	m sec	—	○	○
Peak current	20 ～ 600	A	—	○	○
Peak time	0.1 ～ 3.0	m sec	—	○	○

○ : Item to be set — : Item not to be set
△ : Item to be set only when aluminum is the wire material

Checkpoints for setting the conditions

Concerning the welding current and wire speed

The average current (or average feed speed) during pulsed welding serves as the welding current (or wire feed speed) which is initially input as the welding condition. The base current, peak current and other pulse conditions for pulsed welding are automatically calculated on the basis of the welding current (or wire feed speed) which has been input.

Arc length tuning

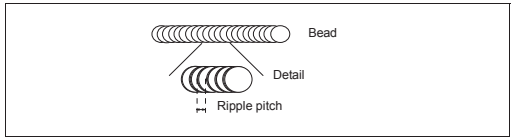
Since the CPDPAS-501 is a unified adjustment welder, the optimum voltage for the current which has been set is output automatically. The "arc length fine adjustment" is an adjustment value for increasing or reducing the voltage which is output automatically. For example, when it is set to +5, the output voltage is increased by approximately 0.5V.

Arc characteristic

The pulsed arc characteristics are a parameter used for adjusting the pulse rise time and pulse fall time internally. When its value is increased, an expansive soft arc is obtained; conversely, when it is reduced, a concentrated hard arc is obtained.

Wave frequency

The wave frequency is a parameter used for adjusting the ripple pitch of beads which are shaped like fish scales and which occur in the DC wave pulsed method. The ripple pitch can be adjusted as desired by a combination of the welding speed and wave frequency. Increasing the wave frequency while keeping the welding speed fixed reduces the pitch width; conversely, reducing it increases the pitch width.

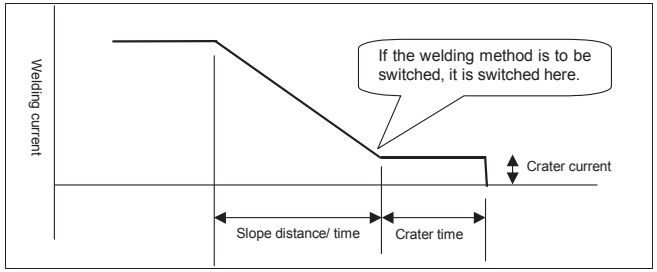


Slope control involves changing the conditions (welding current, welding voltage) in the form of a slope (that is to say, gradually) rather than switching the welding conditions straight away to the specified values. It enables the sputter amount, weld defects, etc. that occur especially when the conditions are switched to be reduced. It can be used when starting the welding, changing the conditions or ending the welding.

Slope control when welding is ended

When welding is ended, slope control is conducted as the conditions are changed from the regular welding conditions to the crater conditions.

Even when crater treatment is not performed, the crater conditions (crater current, crater voltage) must be input if slope control is to be conducted.



■ **Example of screen display**

```
AE[W1, OFF, 130A,  18.0V,  0.1s, 0.1s]
```

[ ] shows Welder No., Condition file No. (OFF indicates no number is specified), and arc end conditions (Current/ Voltage/ Crater time/ Postflow time) from the left.

See
AS; Arc start (FN414)

Function Commands (FN Codes)

Command name	AE
FN code	415
Title name	Arc end
General description	Terminates arc welding with the specified conditions.

■ **General description**

This command enables to end Arc welding, performing the crater process with the specified conditions. The welding end position is a move command shortly before AE(FN415). To select the command, designate the FN code "415" or press F key <AE>.

■ **Parameters**

Set "Welding end control conditions" first, after selecting this command (AE).
"Welding end control conditions" are the conditions displayed on Page 1.
"Welding end conditions" are displayed on Page 2 and after.

●**Welding end control conditions**

Conditions		Explanation
Welder		This specifies the target welder when a multiple number of welders are connected. It need not be set if only one welder is being used.
AS Cond. file		"0" is selected if the welding conditions are to be specified directly using numbers; a file number (1 to 999) is selected if they are to be specified by a file. Enter the number and press [Enter]→F11<Input value> key. Then, the conditions stored in the previously created condition files are to be called. If you entered the file number not created yet, the initial conditions are to be called. Also, called conditions can be modified. If writing after modification, the modified details are reflected in the files. If newly creating the files, they will be stored in the internal memory.
Output current	DC	This is for welding using DC waveforms.
	AC	This is for welding using AC waveforms.
	AC-DC (*1)	This is for hybrid welding (a method that switches between AC and DC in the specified cycle).
Slope control	ON	Down-slope control is performed.
	OFF	Down-slope control is not performed.
AC wave	Standard	This enables welding across a wide range from thin sheets to thick sheets.
	Hard	・ This enables concentrated arcs as with DC to be produced. ・ It is useful for thin-sheet fillet welding, etc.
	Soft	・ This enables soft arcs to be produced. ・ It is useful for thin-sheet butt welding, etc.

(*1) This mode requires the optional "TIG pulsed welding" software.

●**Welding end conditions**

Classification	Condition		Range	Unit
Welding end condition	Crater	Crater current	4 ～ 300	A
		Crater time	0.0 ～ 9.9	sec
	Postflow time		0.0 ～ 60.0	sec
	Postflow robot operation		Stop/ Move	-
	Filler	Retract speed	0 ～ rated	cm/min
		Retract time	0.0 ～ 9.9	sec
	AC frequency		50 ～ 200	Hz
	Cleaning width		-10 ～ +10	-
	AC-DC swith-ing	AC-DC frequency	0.5 ～ 30.0	Hz
		AC-DC ratio	1 ～ 99	%
Down slope	Slope time or Slope distance		0.0 ～ 9.9	sec
			0 ～ 99	mm
	Robot stop time		0.0 ～ 9.9	sec
	Wire precedence time		0.0 ～ 9.9	sec

Checkpoints for setting the conditions

Down slope

This process is for preventing crater cracks from forming at arc end, and it achieves this by lowering the output current step by step. It performs the 2-step processing described below.
(1) Before the arc end point is reached, down-slope is conducted while the torch is being moved.
(2) Next, down-slope is conducted at the arc end point with the robot stopped.

Wire precedence time

This is the time by which the wire feed end timing is brought ahead of the down-slope end.

The advance time is set using a time shorter than the down-slope time. If the advance time is longer than the down-slope time, down-slope is not conducted. If down-slope is specified when the crater time is 0 seconds, down-slope to the crater conditions is conducted.

A maximum of 99 mm can be specified for the down-slope distance. However, if the arc end point distance from the teach point immediately before is less than 99 mm, this distance will serve as the maximum down-slope distance instead.

Postflow

"Stop" or "Move" can be selected as the operation performed by the robot during after-

Stop ... The robot is stopped until after-flow or retract ends.
Move ... Upon completion of crater welding, the robot moves to the next sequence while performing post-flow (which also includes retract).

Set to "stop" initially. When a long time is needed to release the gas in order to prevent oxidation of the tungsten electrodes, select "Move." The tact time can be shortened since the gas is released at the same time as the robot is operating.
However, since the gas release is controlled by the robot, the following restrictions apply when "Operate" has been set.

(1) If a welding command has been executed during post-flow, post-flow is stopped immediately, and the welding command is executed.
(2) If the work was completed (if the END command was executed) during post-flow, post-flow is stopped.

In regard to restriction (2) , the timer command (DELAY) is input before the work is completed (before the END command) to set enough time for after-flow to be completed. By this means, after-flow is conducted as per the time set.
However, since this method may affect the tact time, it is recommended that the step jump command (JMP) be added before the END command and teaching be conducted to ensure that a return is made to the head of the work program if useless robot operations are to be avoided.

■ **Example of screen display**

```
AE[W1,OFF,130A,  0.1s,0.1s]
```

[ ] shows Welder No., Condition file No. (OFF indicates no number is specified), and arc end conditions (Current/ Crater time/ Postflow time) from the left.

See
AS; Arc start (FN414)

Function Commands (FN Codes)

Command name	AE
FN code	415
Title name	Arc end
General description	Terminates arc welding with the specified conditions.

■ **General description**

This command enables to end Arc welding, performing the crater process with the specified conditions. The welding end position is a move command shortly before AE(FN415). To select the command, designate the FN code "415" or press F key <AE>.

■ **Parameters**

●**Parameters other than welding conditions**

Conditions	Details
Welder	This specifies the target welder when a multiple number of welders are connected. It need not be set if only one welder is being used.
AS Cond. File	"0" is selected if the welding conditions are to be specified directly using numbers; a file number (1 to 999) is selected if they are to be specified by a file. Enter the number and press [Enter]→F11<Input value> key. Then, the conditions stored in the previously created condition files are to be called. If you entered the file number not created yet, the initial conditions are to be called. Also, called conditions can be modified. If writing after modification, the modified details are reflected in the files. If newly creating the files, they will be stored in the internal memory.

●**Welding end conditions**

Conditions	Range	Unit
Current cond.	Speed/ Current	-
Welding current (When "Current cond." is "Current".)	1 ～ rated	A
Wire speed (When "Current cond." is "Speed".)	1 ～ 9999	cm/min
Welding voltage (When individual adjustment)	0.1 ～ rated	V
Arc length tuning (When unified adjustment)	-100 ～ +100	-
Crater time	0.0 ～ 9.9	sec
Postflow time	0.0 ～ 9.9	sec
Arc characteristic	-99 ～ +99	%
Burnback voltage	10 ～ rated	V

※ Since incorrect settings may cause the poor welding quality, some of the parameters above may not be displayed depending on the operator qualification.

Checkpoints for setting the conditions

Crater conditions

Crater condition denotes the process to reform the bead shape after welding performance by applying the arc for "a certain period of time". The arc applied here is "less than the real welding".
The condition "less than the real welding" is called "*crater current/ crater voltage*" (arc length fine adjustment) and likewise, "a certain period of time" to apply the arc, "*crater time*".

Arc length tuning

Since the DM-350/500 is a unified adjustment welder, the optimum voltage for the current which has been set is output automatically. The "arc length fine adjustment" is an adjustment value for increasing or reducing the voltage which is output automatically. For example, when it is set to +5, the output voltage is increased by approximately 0.5V.

Arc characteristic

The arc characteristics value is expressed in the form of a numerical value which enables the arc hardness or softness to be set. When this value is increased gradually in the "+" direction, a concentrated hard arc is obtained; conversely, when it is reduced gradually in the "–" direction, an expansive soft arc is obtained.
A "+" value tends to minimize the arc heat and is thus ideal for upward or sideways welding.
A "–" value tends to minimize spatter.
Bear in mind that setting an excessively high value will cause the welding to become unstable.

Postflow time

"Postflow time" is to keep supplying the gas for a while to prevent the welded part from oxidized.

Burnback voltage

The optimum values of this parameter are set automatically depending on the specified welding current (or wire feed speed). Do not change this condition, basically. If changing, adjust the values slightly based on the optimum value. If making a big change, welding cannot be executed and welding error may occur, or welding quality may become worse.

■ **Example of screen display**

```
AE[W1,OFF,130A,  18.0V,  0.1s,0.1s]
```

[ ] shows Welder No., Condition file No. (OFF indicates no number is specified), and arc end conditions (Current/ Voltage/ Crater time/ Postflow time) from the left.

See
AS; Arc start (FN414)

Function commands (FN codes)

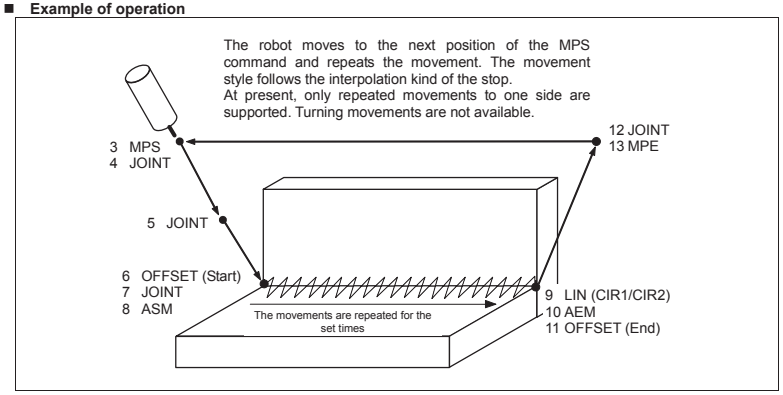
Command name	ASM
FN code	418
Title name	Multi Pass Welding Start
General description	This starts the multi-pass welding under the designated conditions.

■ General description

This starts the multi-pass welding under the designated conditions. For teach and playback, the optional software "Multi-pass Welding function" is required.

As for details, refer to the instruction manual for multi-pass welding function (optional).

■ Example of operation



Step	Command
1	100% JOINT
2	100% JOINT
3	MPS [3 times]
4	100% JOINT
5	100% JOINT
6	OFFSET [Start]
7	100% JOINT
8	ASM [1]
9	300cm/m LINE
10	AEM [1]
11	OFFSET [End]
12	100% JOINT
13	MPE
14	100% JOINT
15	END

Multi-pass welding start condition file (ASM*ARCW.001)

Pass	Multi-pass welding start condition file number	Weaving condition file number
First pass	1 (AS**ARCW.001)	1 (WFP.001)
Second pass	2 (AS**ARCW.002)	2 (WFP.002)
Third pass	3 (AS**ARCW.003)	3 (WFP.003)
Fourth pass	4 (AS**ARCW.004)	4 (WFP.004)
Fifth pass	5 (AS**ARCW.005)	5 (WFP.005)
:	:	:

■ Parameter

Parameter No. 1	AS Cond. file	This designates the multi-pass welding start condition file number. (1 - 999) This file is to be created in advance by <Arc Condition> - [6 Arc start in multi-pass welding].

■ Example of screen display

```
ASM[W1,001,        ->]    FN418;MultiPass Welding Start
```

See
AEM: Multi Pass Welding End (FN419)
MPS; Multi Pass Section Start (FN496)
MPE; Multi Pass Section End (FN497)
EP; Execution Pass Specification (FN498)
OFFSET; Multi Offset Specification (FN499)

Function commands (FN codes)

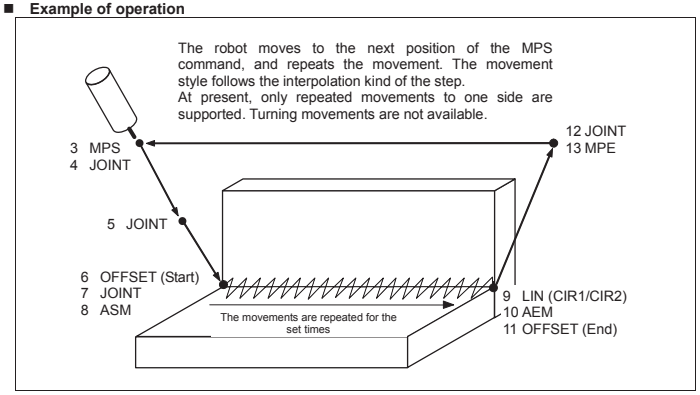
Command name	AEM
FN code	419
Title name	Multi Pass Welding End
General description	This ends the multi-pass welding under the designated conditions.

■ General description

This ends the multi-pass welding under the designated conditions. For teach and playback, the optional software "Multi-pass Welding function" is required.

As for details, refer to the instruction manual for multi-pass welding function (optional).

■ Example of operation



Step	Command
1	100% JOINT
2	100% JOINT
3	MPS [3 times]
4	100% JOINT
5	100% JOINT
6	OFFSET [Start]
7	100% JOINT
8	ASM [1]
9	300cm/m LINE
10	AEM [1]
11	OFFSET [End]
12	100% JOINT
13	MPE
14	100% JOINT
15	END

Multi-pass welding end condition file (AEM*ARCW.001)

Pass	Multi-pass welding end condition file number
First pass	1 (AE**ARCW.001)
Second pass	2 (AE**ARCW.002)
Third Pass	3 (AE**ARCW.003)
Fourth pass	4 (AE**ARCW.004)
Fifth pass	5 (AE**ARCW.005)
:	:

■ Parameter

Parameter No. 1	AS Cond. file	This designates the multi-pass welding end condition file number. (1 - 999) This file is to be created in advance by <Arc Condition> - [7 Arc end in multi-pass welding].

■ Example of screen display

```
AEM[W1,001,        ->]    FN419;Multi Pass Welding End
```

See
ASM: Multi Pass Welding Start (FN418)
MPS; Multi Pass Section Start (FN496)
MPE; Multi Pass Section End (FN497)
EP; Execution Pass Specification (FN498)
OFFSET; Multi Offset Specification (FN499)

Function commands (FN codes)

Command name	SPN
FN code	438
Title name	Servo ON
General description	This turns ON the servo power source in unit of mechanism.

■ General description

This turns ON the servo power source in unit of designated mechanism. For teach and playback, the optional software "Mechanism-by-Mechanism Servo ON/OFF function" is required.

- At check go operation (when <Service Utilities> - [1 Teach / Playback Condition] - [9 Check with function] is set to "Enabled")
- At automatic operation (at playback)

To turn OFF the servo power source, use SPF (FN439).

■ Parameter

Parameter No. 1	Mechanism No.	This designates the mechanism number to turn ON the servo power source. (1 - 9) When a mechanism that does not exist, or a mechanism that does not belong to the unit of teach objective is designated, though there is no error at teaching, it is detected as an alarm "A4911" at check go operation, and an error "E4911" at automatic operation.

■ Example of screen display

```
SPN[1]                    FN438;Servo ON
```

See
SPF; Servo OFF (FN439)

Function commands (FN codes)

Command name	SPF
FN code	439
Title name	Servo OFF
General description	This turns OFF the servo power source in unit of mechanism.

■ General description

This turns OFF the servo power source in unit of designated mechanism. For teach and playback, the optional software "Mechanism-by-Mechanism Servo ON/OFF function" is required.

- At check go operation (when <Service Utilities> - [1 Teach / Playback Condition] - [9 Check with function] is set to "Enabled")
- At automatic operation (at playback)

To turn ON the servo power source, use SPN (FN438).

■ Parameter

Parameter No. 1	Mechanism No.	This designates the mechanism number to turn OFF the servo power source. (1 - 9) When a mechanism that does not exist, or a mechanism that does not belong to the unit of teach objective is designated, though there is no error at teaching, it is detected as an alarm "A4911" at check go operation, and an error "E4911" at automatic operation.

■ Example of screen display

```
SPF[1]                    FN439;Servo OFF
```

See
SPN; Servo ON (FN438)

Function Commands (FN Codes)

Command name	WFP
FN code	440
Title name	Fixed pattern weaving
General description	Starts weaving with the specified waveform, attitude, and frequency.

■ General description

This command enables to start with the specified waveform, attitude, and frequency.
The robot starts weaving from a position of the movement command taught immediately before this command (WFP).

■ Parameters

Conditions	Range
WFP cond. file	0 ～ 999
Frequency	0.0 ～ 20.0 [Hz]
Function Type	Linear func./ Sine wave / Circle
Amplitude (right amplitude, left amplitude) * When the linear or sine wave function has been set as the operation pattern	0.0 ～ 50.0 [mm]
Radius (right radius, left radius) * When the arc has been set as the operation pattern	0.0 ～ 50.0 [mm]
Stopping time (center, 1/4 period, 3/4 period)	0.0 ～ 9.9 [sec.]
Move at Stop Time	Not exist/ Exist
Keep weaving time	Not keep/ Keep
Weav Start Direct.	Right/ Left
Weav Angle (right angle of inclination, left angle of inclination)	-180 ～ 180 [deg.]
Torch Angle (right angle of inclination, left angle of inclination)	-180 ～ 180 [deg.]
Push Angle (right crosswise angle, left crosswise angle) * When the linear or sine wave function has been set as the operation pattern	-180 ～ 180 [deg.]
Circle center ratios (front circle center ratio, back circle center ratio) * When the arc has been set as the operation pattern	1 ～ 100 [%]

WFP cond. file

"0" is selected if the welding conditions are to be specified directly using numbers; a file number (1 to 999) is selected if they are to be specified by a file.

Enter the number and press [Enter]→F11<Input value> key. Then, the conditions stored in the previously created condition files are to be called. If you entered the file number not created yet, the initial conditions are to be called.

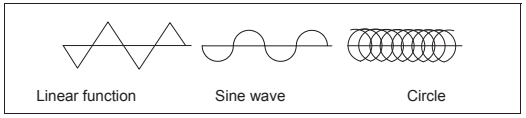
Also, called conditions can be modified. If writing after modification, the modified details are reflected in the files. If newly creating the files, they will be stored in the internal memory.

Frequency

This is the weaving frequency (number of waveforms per second).

Function Type

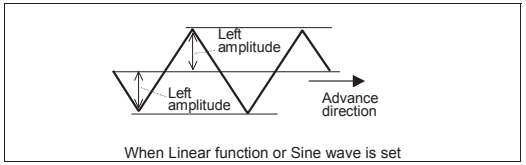
One of the following can be selected as the weaving operation pattern (waveform).



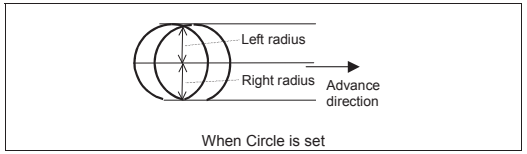
Amplitude

This condition is for setting the weaving amplitude when the linear function or trigonometric function has been set as the operation pattern. Both the left and right amplitudes relative to the advance direction can be set.

The radius from the center of the circle is set when the arc has been set as the operation pattern. Both the left and right radii relative to the advance direction can be set.



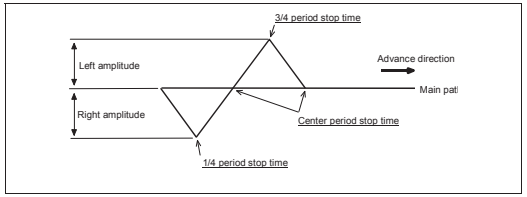
When Linear function or Sine wave is set



When Circle is set

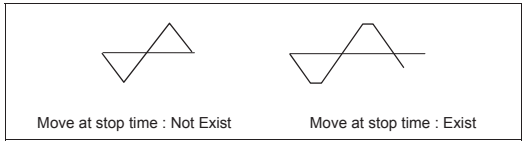
Stopping time

The center stop time, 1/4 period stop time and 3/4 period stop time are set.
Deeper penetration can be obtained by setting the stopping time.



Move at stop time

This condition is for selecting whether the robot is to move forward in the advance direction or stop during the weaving stop time when weaving stop time has been set. The initial setting is "Not Exist"

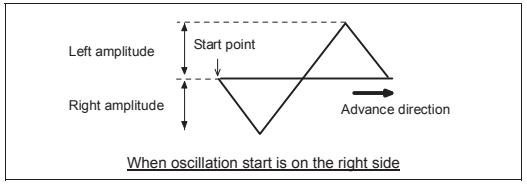

Move at stop time : Not Exist　　Move at stop time : Exist

Keep weaving time

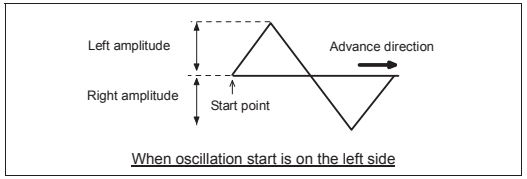
The condition is for setting whether the actual welding time is to be maintained even when the weaving stop time has been set. If the weaving stop time is not set, the condition will be devoid of meaning.

Weave start direction

This condition is for setting whether the weaving is to start on the right or left relative to the advance direction. Right is the initial setting, and weaving starts from the right side relative to the advance direction.



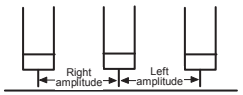
When oscillation start is on the right side



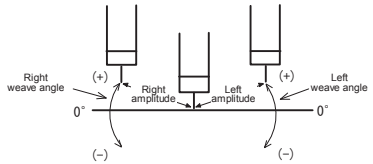
When oscillation start is on the left side

Weave angle

This condition is for setting angle of the weaving from the main path. It can be set both for the left and right amplitude. The initial value is 0 degrees, and the weaving plane is perpendicular to the torch.



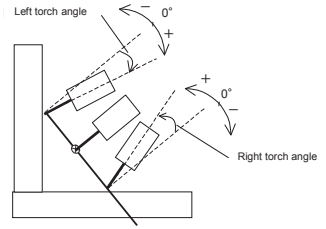
Weaving at an angle of inclination of 0 degrees



Weaving when there is an angle of inclination

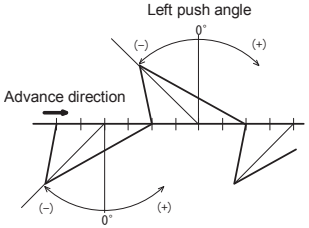
Torch angle

When the angle of inclination of the torch is set, this condition makes it possible to determine the welding posture in respect of the work piece surface at the weaving end point.



Push angle

When the crosswise angle is set, this enables a change into a waveform such as the one shown in the figure below. However, when the crosswise angle is set, the amplitude is tilted in the advance direction and is thus shortened. If, for instance, the angle is set to -45 degrees, the amplitude will be about 70% of what it would be if the angle were 0 degrees.



Circle rate

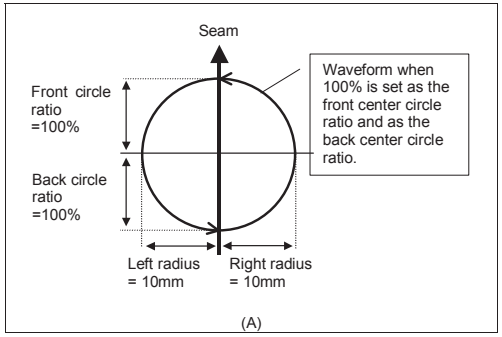
The circle center ratio is set when the arc has been set as the operation pattern. This ratio is for determining the percentage of the advance direction component to be reflected in the arc radius (for determining to what extent the arc is to be distorted).

For instance, if it is assumed that the circle in Fig. (A), when the front and back center circle ratios are 100% has:

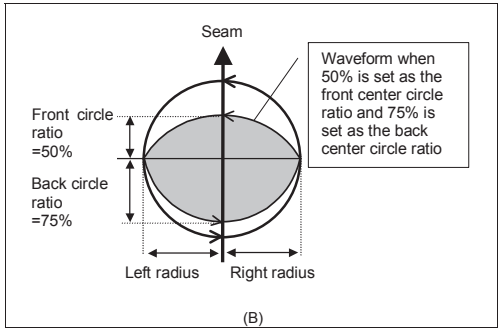
- A left radius and right radius of the same length
- A front circle center ratio and back circle center ratio of 100%

Then the circle will have a perfectly round shape. (Perfect circles are formed in cases where weaving is performed immediately. Normally, they are not formed since the speed component of the advance direction is added.)

The center circle ratio is what determines the extent to which the arcs are to be distorted in the advance direction.
The circle shown in Fig.(B) is formed if 50% is set as the front center circle ratio and 75% is set as the back center circle ratio.



(A)



(B)

■ **Example of screen display**

```
WFP[OFF, 5.0Hz      →]    FN440:Fix Pattern Weaving
```

[ ] shows Condition file number, and Frequency from the left.

See
WAX; Axis weaving (FN441)
WE; Weaving end(FN443)

Function Commands (FN Codes)

Command name	WAX
FN code	441
Title name	Axis Weaving
General description	Starts weaving with the simple harmonic motion of the axes.

■ General description

This command enables to start weaving with the simple harmonic motion of the axes.
The robot starts weaving from a position of the movement command taught immediately before this command (WAX).

■ Parameters

Conditions	Range
WFP cond. file	0 ～ 999
Frequency	0.0 ～ 20.0 [Hz]
Stopping time (center, 1/4 period, 3/4 period)	0.0 ～ 9.9 [sec.]
Move at Stop Time	Not exist/ Exist
Keep weaving time	Not keep/ Keep
Axis No.	1 ～ 6
Amplitude (right amplitude, left amplitude)	0.0 ～ 50.0 [mm]

WFP cond. file

"0" is selected if the welding conditions are to be specified directly using numbers; a file number (1 to 999) is selected if they are to be specified by a file.

Enter the number and press [Enter]→F11<Input value> key. Then, the conditions stored in the previously created condition files are to be called. If you entered the file number not created yet, the initial conditions are to be called.

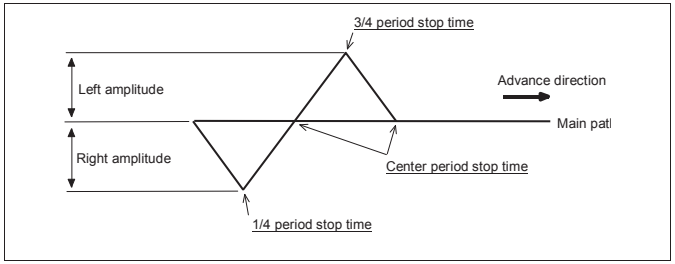
Also, called conditions can be modified. If writing after modification, the modified details are reflected in the files. If newly creating the files, they will be stored in the internal memory.

Frequency

This is the weaving frequency (number of waveforms per second).

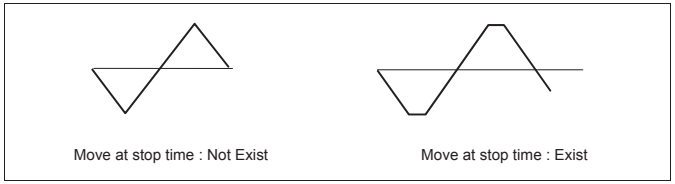
Stopping time

The center stop time, 1/4 period stop time and 3/4 period stop time are set. Deeper penetration can be obtained by setting the stopping time.



Move at stop time

This condition is for selecting whether the robot is to move forward in the advance direction or stop during the weaving stop time when weaving stop time has been set. The initial setting is "Not Exist"



Keep weaving time

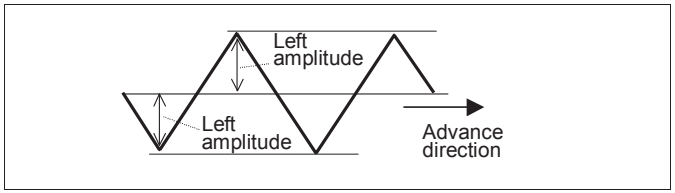
The condition is for setting whether the actual welding time is to be maintained even when the weaving stop time has been set. If the weaving stop time is not set, the condition will be devoid of meaning.

Axis number

This condition specifies the number of the axis which will be used to conduct the weaving.

Amplitude

This condition is for setting the weaving amplitude. Both the left and right amplitudes relative to the advance direction can be set.



■ Example of screen display

```
WAX[OFF, 5.0Hz          →]    FN441:Axis Weaving
```

[ ] shows Condition file number, and Frequency from the left.

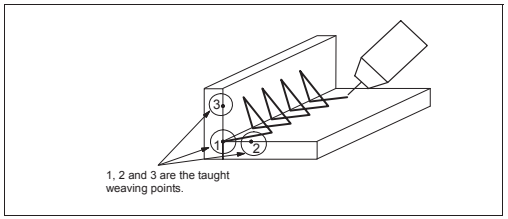
See
WFP; Fixed Pattern weaving (FN440)
WE; Weaving end(FN443)

Function commands (FN codes)

Command name	WSF
FN code	442
Title name	Taught Weaving
General description	This carries out weaving in the taught pattern.

■ General description

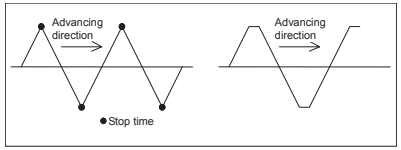
Taught weaving (optional) is the weaving operation where points for weaving can be taught according to the groove shape.
A weaving pattern as shown in the figure below can be created arbitrarily.



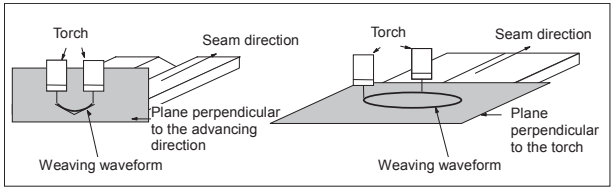
As for details of taught weaving, refer to the instruction manual for taught weaving. This help explains its outline.

■ Parameter

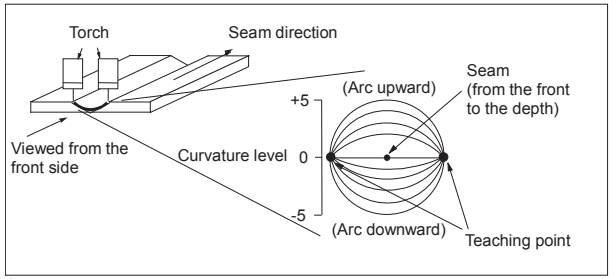
Condition	Description	Setting range
Weaving condition		
Control of Weaving speed	This sets whether the weaving speed is designated by the frequency or by the speed between points.	Frequency / Point speed
Frequency	This sets the weaving frequency in the case when "Control of Weaving speed" is set to "Frequency".	0.1～10.0 [Hz]
Point speed	This sets the speed in the case when "Control of Weaving speed" is set to "Point speed".	1～999 [cm/min]
Motion type	This sets whether to loop or repeat between taught points.	Loop / Repeat
Move at Stop Time	This sets whether or not to move the robot in the main track direction during stop time, in the case when stop time is set in teaching point.	Not exist / Exist



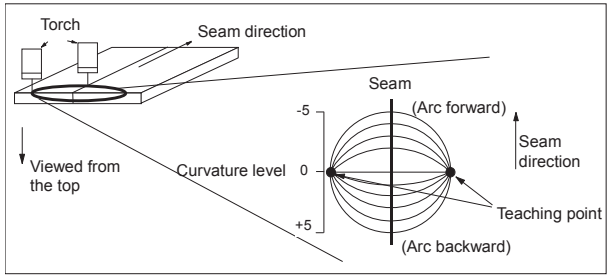
Keep weaving time	This sets whether or not to maintain the actual welding time even in the case where stop time is set in teaching point.	Not keep / Keep
Weaving plane	This sets "User" when the number of teaching points is 3 points or more. This sets "Advance" or "Torch" when the number of teaching points is 2 points. Advance ····· The plane perpendicular to the advancing direction is the weaving plane. Torch ··· ····· The plane perpendicular to the torch direction is the weaving plane.	User / Advance / Torch



Condition	Description	Setting range
Comment	By [ENABLE] and [EDIT], a comment may be added.	Arbitrary character string
Parameter at teaching points		
Stopping time	This sets the stop time at each point.	0.0 ~ 9.9 [sec]
Curveture	Curvature level becomes valid only when the number of teaching points is 2 points, and this sets what a curvature should be drawn. (When the curvature level is 0, a straight line results.) When the number of teaching points is 3 points or more, the torch moves in a straight line between points, so even if this is set, setting is ignored. (Normally, leave it "0".)	-5 ~ 5



Curvature level when the weaving plane is set to "Advance"



Curvature level when the weaving plane is set to "Torch"

Positional data Posture data	This sets the point position and posture with the seam coordinate system as reference.	

■ Example of screen display

```
WSF[001, 60cm/min          ->]    FN442;Taught Weaving
```

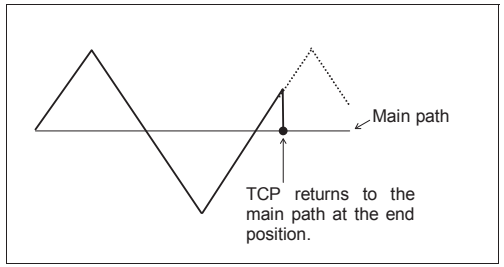
See
WE; Weaving End (FN443)

Function Commands (FN Codes)

Command name	WE
FN code	443
Title name	Weaving End
General description	Terminates weaving

■ General description

This command is used to end the weaving while it is being executed.
TCP returns to the main path if it is midway through a weaving waveform.



■ Parameters

None

■ Example of screen display

```
WE[1]                    FN443:Weaving End
```

See
WFP; Fixed Pattern weaving (FN440)
WAX; Axis weaving (FN441)

Application Command (FN Code)

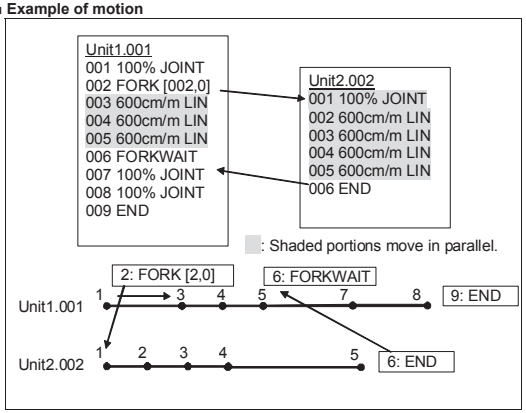
Command name	FORK
FN code	450
Title name	Start other unit
Outline	Used to start the work program of other unit.

■ Outline

This command is used to start the work program of other unit. (The command starts a specified work program in addition to a work program currently being played back.)

If the FORK command is taught, teach the FORKWAIT command that waits for the completion of the FORK command to a proper position.
Even though the FORK command and the FORKWAIT command are not necessarily taught by the set, it is recommended to teach them by the set for the safety purpose in order to avoid resource conflict or multiple execution of the FORK command.

■ Example of motion



If the FORKWAIT command is not used in the source work program of the FORK command, the source work program of the FORK command may be completed before its destination work program is completed.
In this case, the handling of the start signal for the source work program of the FORK command varies with the setting of "Service" → "1 Teach/Playback Condition" → "27 Start during FORK".

When the "Start during FORK" parameter is set to "Prohibited":
The error message "E6080: The system attempted to start a unit including a resource that is in starting mode" will be output. The system accepts no starting signal unless the FORM command is completed.

When the "Start during FORK" parameter is set to "Allowed":
The system will accept the starting signal and play back it.

■ Parameters

Parameter 1	Program No.	Used to make setting of program number to be started. (Setting range: 0 to 9999)
Parameter 2	Resource conflict waiting time	Used to make setting of a period of time waiting for a mechanism to be used in the work program that is started to be released if this mechanism is being started in a different unit, in seconds. (Setting range: ∞ (–1) or 0 to 99) If the relevant mechanism is released within the set period of time, the set program will be started. If it is not released, a fault will be output.

■ Example of screen display

```
FORK[1,-1]                FN450: Start other unit
```

Related commands
FORKI: Start other unit (Input) (FN451)
FORKN: Start other unit (Number of times) (FN452)
FORKWAIT: Fork complete waiting (FN453)

Application Command (FN Code)

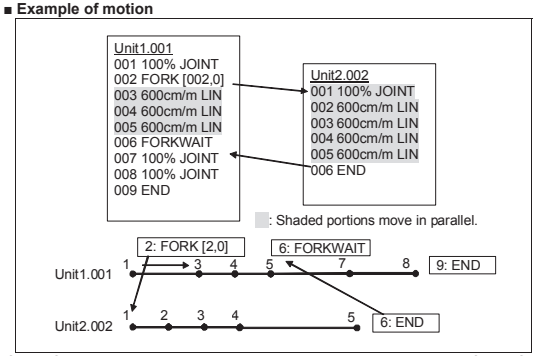
Command name	FORKI
FN code	451
Title name	Start other unit (Input)
Outline	Used to start the work program of other unit according to input signal.

■ Outline

This command is used to start the work program of other unit according to input signal. (The command starts a specified work program in addition to a work program currently being played back.)
Without input signal, the work program cannot be started.

If the FORKI command is taught, teach the FORKWAIT command that waits for the completion of the FORKI command to a proper position.
Even though the FORKI command and the FORKWAIT command are not necessarily taught by the set, it is recommended to teach them by the set for the safety purpose in order to avoid resource conflict or multiple execution of the FORKI command.

■ Example of motion



If the FORKWAIT command is not used in the source work program of the FORK command, the source work program of the FORK command may be completed before its destination work program is completed.
In this case, the handling of the start signal for the source work program of the FORK command varies with the setting of "Service" → "1 Teach/Playback Condition" → "27 Start during FORK".

When the "Start during FORK" parameter is set to "Prohibited":
The error message "E6080: The system attempted to start a unit including a resource that is in starting mode" will be output. The system accepts no starting signal unless the FORK command is completed.

When the "Start during FORK" parameter is set to "Allowed":
The system will accept the starting signal and play back it.

■ Parameters

Parameter 1	Program No.	Used to make setting of program number to be started. (Setting range: 0 to 9999)
Parameter 2	Resource conflict waiting time	Used to make setting of a period of time waiting for a mechanism to be used in the work program that is started to be released if this mechanism is being started in a different unit, in seconds. (Setting range: ∞ (–1) or 0 to 99) If the relevant mechanism is released within the set period of time, the set program will be started. If it is not released, a fault will be output.
Parameter 3	Input signal No.	Used to make setting of input signal number that determines whether or not to start the work program. (Setting range: 1 to 2048 or 5001 to 5064)

■ Example of screen display

FORKI[1,-1,I1]	FN451: Start other unit (Input)

Related commands
FORK: Start other unit (FN450)
FORKN: Start other unit (Number of times) (FN452)
FORKWAIT: Fork complete waiting (FN453)

Application Command (FN Code)

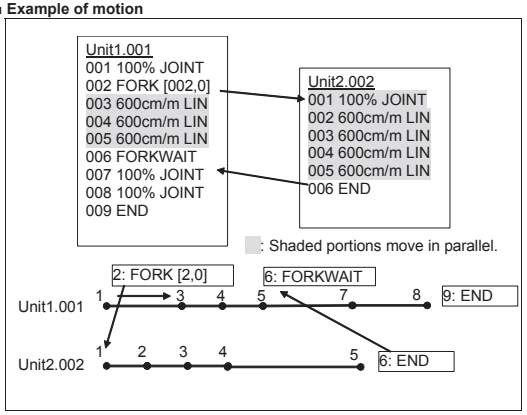
Command name	FORKN
FN code	452
Title name	Start other unit (Number of times)
Outline	Used to start the work program of other unit according to the number of pass times.

■ Outline

This command is used to start the work program of other unit. (The command starts a specified work program in addition to a work program currently being played back.)
This command makes the unit pass by the specified number of times and starts the work program of a different unit when the number of pass times reaches "specified number of pass times + 1".
For example, when the "Number of times" parameter is set to "2", the unit will pass two times, and the work program of a different unit will be started at the third pass.
When the number of pass times is less than the set value, such work program will not be started.

If the FORKN command is taught, teach the FORKWAIT command that waits for the completion of the FORKN command to a proper position.
Even though the FORKN command and the FORKWAIT command are not necessarily taught by the set, it is recommended to teach them by the set for the safety purpose in order to avoid resource conflict or multiple execution of the FORKN command.

■ Example of motion



If the FORKWAIT command is not used in the source work program of the FORK command, the source work program of the FORK command may be completed before its destination work program is completed.
In this case, the handling of the start signal for the source work program of the FORK command varies with the setting of "Service" → "1 Teach/Playback Condition" → "27 Start during FORK".

When the "Start during FORK" parameter is set to "Prohibited":
The error message "E6080: The system attempted to start a unit including a resource that is in starting mode" will be output. The system accepts no starting signal unless the FORK command is completed.

When the "Start during FORK" parameter is set to "Allowed":
The system will accept the starting signal and play back it.

■ **Parameters**

Parameter 1	Program No.	Used to make setting of program number to be started. (Setting range: 0 to 9999)
Parameter 2	Resource conflict waiting time	Used to make setting of a period of time waiting for a mechanism to be used in the work program that is started to be released if this mechanism is being started in a different unit, in seconds. (Setting range: ∞ (–1) or 0 to 99) If the relevant mechanism is released within the set period of time, the set program will be started. If it is not released, a fault will be output.
Parameter 3	Register No.	The term “register” means a memory used to count. Since integer variables are used, this parameter makes setting of the register number. (Setting range: 1 to 200)
Parameter 4	Number of times	Used to record the number of pass times that serves as condition for executing jump. The relevant unit passes by the specified number of times and executes jump when the number of pass times reaches “specified number of pass times + 1”. (Recording range: 0 to 10000)

■ **Example of screen display**

FORKN[1,-1,V1%,5]	FN452: Start other unit (Number of times)

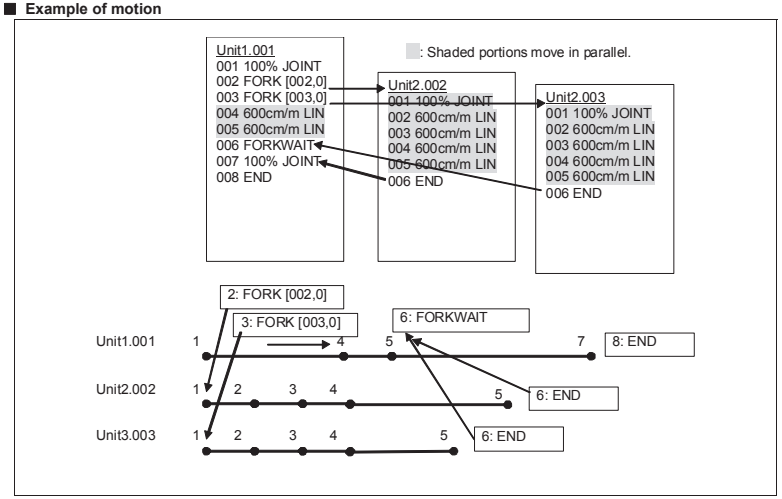
Related commands
FORK: Start other unit (FN450)
FORKI: Start other unit (Input) (FN451)
FORKWAIT: Fork complete waiting (FN453)

Application Command (FN Code)

Command name	FORKWAIT
FN code	453
Title name	Fork complete waiting
Outline	Used to wait for the completion of the work programs of other unit started by the FORK, FORKI and FORKN commands.

■ **Outline**
This command is used to wait for the completion of all programs started by the FORK, FORKI and FORKN commands.

■ **Example of motion**



■ **Parameters**
N/A

■ **Example of screen display**

FORKWAIT	FN453: Fork complete waiting

Related commands
FORK: Start other unit (FN450)
FORKI: Start other unit (Input) (FN451)
FORKN: Start other unit (Number of times) (FN452)

Application Command (FN Code)

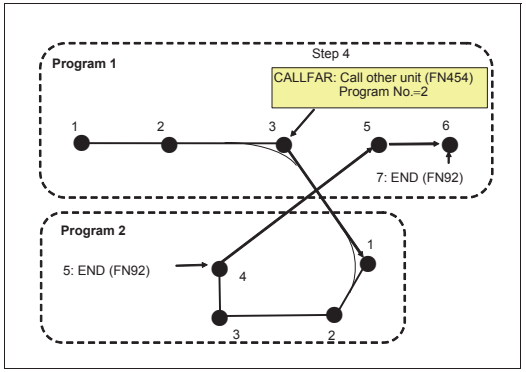
Command name	CALLFAR
FN code	454
Title name	Call other unit
Outline	Used to call the work program of other unit.

■ Outline

Using this command makes it possible to call the work program of other unit.
When the work program of other unit is called, the relevant unit will stop executing its own work program, and resume it when the execution of the program called is completed.
It is impossible to further call the program of other unit with the program of call destination.
However, it is possible to call the program within the unit using the CALLP command with the program of call destination (for a maximum of eight hierarchies).

■ Example of motion

Record “CALLFAR: Call other unit (FN454), Program No.=2” in Step 4.
When playing back this program, the robot will reach Step 4, and then skip Steps 5 and 6 to jump to the leading step of the Program 2.
When the playback of the Program 2 is completed (the END: End command is executed), the robot will return to the next step, Step 5 of call command of the call source program 1.



This command will not cause the robot to decelerate.
If no step positioning is specified, this command will be executed when the robot reaches the accuracy range of the immediately preceding movement command.

■ Parameters

Parameter 1	Program No.	Used to make setting of program number to be called. (Setting range: 0 to 9999)
Parameter 2	Resource conflict waiting time	Used to make setting of a period of time waiting for a mechanism to be used in the work program that is called to be released if this mechanism is being started in a different unit, in seconds. (Setting range: ∞ (–1) or 0 to 99) If the relevant mechanism is released within the set period of time, the set program will be called. If it is not released, a fault will be output.

■ Example of screen display

CALLFAR[1,-1]	FN454: Call other unit

Related commands
CALLFARI: Call other unit (Input) (FN455)
CALLFARN: Call other unit (Number of times) (FN456)

Application Command (FN Code)

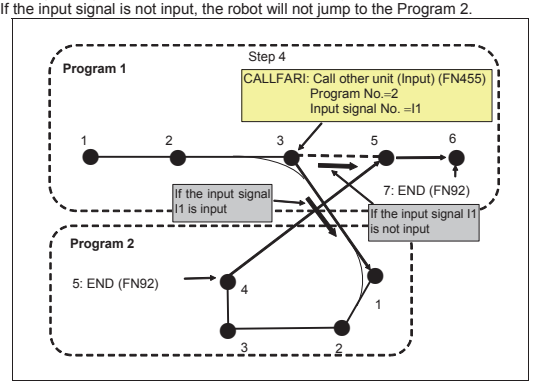
Command name	CALLFARI
FN code	455
Title name	Call other unit (Input)
Outline	Used to call the work program of other unit if input signal is input.

■ Outline

Using this command makes it possible to call the work program of other unit if input signal is input.
This command makes the unit pass by the specified number of times and calls the work program of other unit when the number of pass times reaches “specified number of pass times + 1”.
For example, when the “Number of times” parameter is set to “2”, the unit will pass two times, and the work program of other unit will be called at the third pass.
Note that, if application command is recorded in the leading step of the program of call destination program, the application command of jump destination will be executed on the spot where the call command is executed.
When the work program of other unit is called, the relevant unit will stop executing its own work program, and resume it when the execution of the program called is completed.
It is impossible to further call the program of other unit with the program of call destination.
However, it is possible to call the program within the unit using the CALLP command with the program of call destination (for a maximum of eight hierarchies).

■ Example of motion

Record “CALLFARI: Call other unit (Input) (FN455), Program No. of other unit=2, Input signal=I1)” in Step 4.
When playing back this program, the robot will reach Step 4, jump to the leading step of the Program 2 when the input signal I1 is input, and return to the next step, Step 5 of call command of the call source program 1, when the playback of the Program 2 is completed (the END: End command is executed).
If the input signal is not input, the robot will not jump to the Program 2.



This command will not cause the robot to decelerate.
If no step positioning is specified, the input signal will be inspected just before the command value reaches the accuracy range of the movement command. If the input signal is input, the robot will make an inward turn.
If the input signal is not input, the command value will move toward the recorded point. And when the robot reaches the accuracy range, the input signal will be inspected.

■ Parameters

Parameter 1	Program No.	Used to make setting of program number to be called. (Setting range: 0 to 9999)
Parameter 2	Resource conflict waiting time	Used to make setting of a period of time waiting for a mechanism to be used in the work program that is called to be released if this mechanism is being started in a different unit, in seconds. (Setting range: ∞ (–1) or 0 to 99) If the relevant mechanism is released within the set period of time, the set program will be called. If it is not released, a fault will be output.
Parameter 3	Input signal No.	Used to make setting of input signal number that determines whether or not to start the work program. (Setting range: 1 to 2048 or 5001 to 5064)

■ Example of screen display

CALLFARI[1,-1,I1]	FN455: Call other unit (Input)

Related commands
CALLFAR: Call other unit (FN454)
CALLFARN: Call other unit (Number of times) (FN456)

Application Command (FN Code)

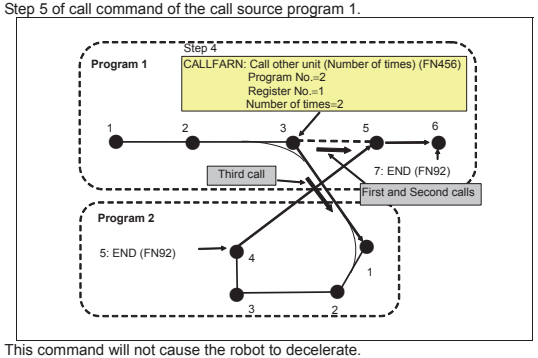
Command name	CALLFARN
FN code	456
Title name	Call other unit (Number of times)
Outline	Used to call the work program of other unit according to the number of pass times.

■ Outline

This command is used to call the work program of other unit.
This command makes the unit pass by the specified number of times and calls the work program of other unit when the number of pass times reaches “specified number of pass times + 1”.
For example, when the “Number of times” parameter is set to “2”, the unit will pass two times, and the work program of other unit will be called at the third pass.
Note that, if application command is recorded in the leading step of the program of call destination program, the application command of jump destination will be executed on the spot where the call command is executed.
When the work program of other unit is called, the relevant unit will stop executing its own work program, and resume it when the execution of the program called is completed.
It is impossible to further call the program of other unit with the program of call destination.
However, it is possible to call the program within the unit using the CALLP command with the program of call destination (for a maximum of eight hierarchies).

■ Example of motion

Record “CALLFARN: Call other unit (Number of times) (FN456), Program No. of other unit =2, Register No.=1, Number of times=2)” in Step 4.
When playing back this program, the robot will pass the first and second calls to move to the Step 5, but jump to the leading step of the Program 2 at the third call.
When the playback of the Program 2 is completed (the END: End command is executed), the robot will return to the next step, Step 5 of call command of the call source program 1.



This command will not cause the robot to decelerate.
If no step positioning is specified, this command will be executed when the robot reaches the accuracy range of the immediately preceding movement command.

■ Parameters

Parameter 1	Program No.	Used to make setting of program number to be called. (Setting range: 0 to 9999)
Parameter 2	Resource conflict waiting time	Used to make setting of a period of time waiting for a mechanism to be used in the work program that is called to be released if this mechanism is being started in a different unit, in seconds. (Setting range: ∞ (−1) or 0 to 99) If the relevant mechanism is released within the set period of time, the set program will be called. If it is not released, a fault will be output.
Parameter 3	Register No.	The term “register” means a memory used to count. Since integer variables are used, this parameter makes setting of the register number. (Setting range: 1 to 200)
Parameter 4	Number of times	Used to record the number of pass times that serves as condition for executing jump. The relevant unit passes by the specified number of times and executes jump when the number of pass times reaches “specified number of pass times + 1”. (Recording range: 0 to 10000)

■ Example of screen display

```
CALLFARN[1,-1,V1%,5]      FN456: Call other unit (Number of times)
```

Related commands
CALLFAR: Call other unit (FN454)
CALLFARI: Call other unit (Input) (FN455)

Function commands (FN codes)

Command name	USRERR
FN code	467
Title name	User Error Output
General description	Output the user customized error

■ General description

When this function is played, this can output the user error (or alarm or information) that is defined in Service / 25.Root Diagnosis / 6.User error definition. Total 997 errors (or alarm or information) can be defined.

Error	Playback aborted, and motor power OFF.
Alarm	Playback aborted, but motor power is still ON.
Information	Playback continued.

■ Example of operation

When this function is played, this can output the user error (or alarm or information). If designated error is not defined, default error is detected.

■ Parameter

Parameter No. 1	Error code	This specifies the error number to be happened in recorded step. (7003-7999)

■ Example of screen display

```
USRERR[7003]      FN467: User error
USRERR[7004]      FN467: User alarm
USRERR[7005]      FN467: User information
```

Function commands (FN codes)

Command name	SF0
FN code	470
Title name	Wire Extension
General description	This detects and corrects the wire extension. * This is used when the touch sensor (AX-WD) is connected.

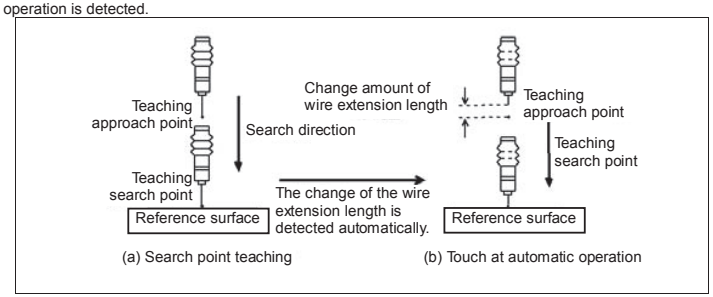
■ General description

SF0 is the command to detect the wire extension so that the search motions and deviation detection of SF1 and SF2 should be carried out correctly. By this command, for example, even if the wire extension length changes, it is possible to limit mistake in detection or deviation caused by that to minimum.

As for details on the respective functions, teach examples and so forth of the touch sensor (AX-WD), refer to the instruction manual for the touch sensor. This help explains its outline.

■ Example of operation

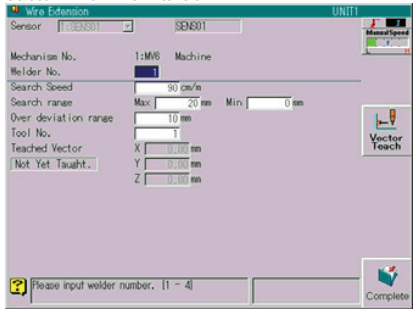
When SF0 is executed, deviation between the search point at teaching and the touch point detected during automatic operation is detected.



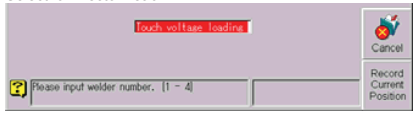
In SF1 and SF2 after SF0, the wire top positions are corrected so as to be always constant.
As for the correction of SF0, until welding is executed, the search command and the motion command just before it is corrected automatically.

■ SF0 Teaching Method

(1) Select FN470 "Wire Extension".

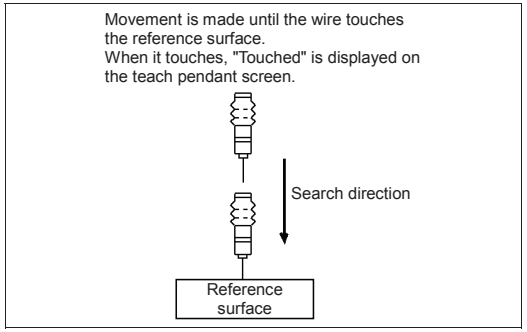


(2) Set necessary parameters (to be described later).
(3) Select f9 <Vector Teach>.



Warning	**During teaching of search vectors, voltage of DC50V is impressed. Do not touch the wire or electricity conductive portions, otherwise electric shock may result.**

(4) By manual operation, movements are made until the wire touches the reference surface.



(5) When the wire touches, select f12 <Record Current Position>, and record the reference position.
(6) Press f12 <Complete> and record SF0.

■ Parameter

Parameter name	Description	Range
Sensor	This selects the sensor to become a teach objective in the case where plural sensors are connected.	
Mechanism No.	The mechanism number to use the sensor is displayed. This cannot be edited.	—
Welder No.	This sets the welding machine number.	1 ~ 4
Search speed	This is the speed of search motions at playback. When the search speed is increased, that tact time is shortened, but the detection precision decreases, and the coasting run amount after touching increases. (The wire may be bent.) Set around 60cm/min (10mm/sec) as a standard amount. Set an appropriate search speed according to your purpose.	6 ~ 360 [cm/min]
Search range (Max.)	This sets the maximum searching distance so as to detect the search idle run at execution of SF0. When after start of search, touch is not detected in motions on set distance from the reference position, a failure is detected.	0 ~ 999 [mm]
Search range (Min.)	This sets the minimum searching distance so as to detect that the wire has already touched at execution of SF0. When after start of search, touch is detected in motions on set distance, a failure is detected.	0 ~ 999 [mm]
Over deviation range	This sets the allowable range of wire extension deviation detected by SF0. When the wire extension deviation after touch detection is larger than the set value, a failure is detected.	0 ~ 999 [mm]
Tool No.	This sets the tool number.	1 ~ 32
Taught Vector	When the search vector is taught, a value is set. The value cannot be edited.	-

■ Example of screen display

SF0[150cm/m, M1] FN470;Wire Extension

Function commands (FN codes)

Command name	SF1
FN code	471
Title name	One Direction Search (Touch)
General description	This detects the deviation of a workpiece. * This is used when the touch sensor (AX-WD) is connected.

■ General description

SF1 is the command that detects the deviation of a workpiece from the positional difference between the search point at teaching and the touch point at automatic operation, and records it to the deviation correction file.

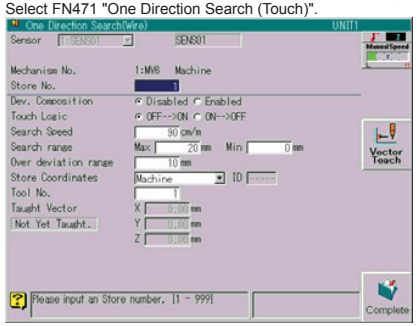
- The deviation of a workpiece to be detected by SF1 is in one direction. When a workpiece displaces in 3 dimensional manners, carry out SF1 in 3 directions.
- Only by executing SF1, the deviation of a workpiece cannot be corrected. To correct the deviation, teaching of SF3 (receiving deviation amount) is necessary.
- In SF1, there are the following 2 methods.
  - Search motion to move the torch attached to manipulator
  - Search motion to rotate or slide the external axis, while keeping the torch standstill

  The latter method is specially called "external axis search".
  The external axis search is available whether the external axis is a slider or a positioner. (However, it is not available in other case than the robot's external axis.) And it is available in both the control methods (coordinate control and simultaneous control).

As for details on the respective functions, teaching examples and so forth of the touch sensor (AX-WD), refer to the instruction manual for the touch sensor. This help explains its outline.

■ SF1 Teach Method

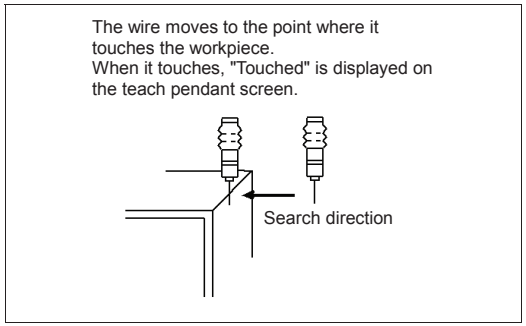
(1) Select the mechanism for search motions as the manual operation mechanism.
(2) Select FN471 "One Direction Search (Touch)".



(3) Set necessary parameters (to be described later).
(4) Select f9 <Vector Teach>.

⚠Warning	**During teaching of search vectors, voltage of DC50V is impressed. Do not touch the wire or electricity conductive portions, otherwise electric shock may result.**

(5) By manual operation, movements are made until the wire touches the reference surface.



(6) When the wire touches, select f12 <Record Current Position>, and record the reference position.
(7) Press f12 <Complete> and record SF1.

■ Parameter

Parameter name	Description	Range
Sensor	This selects the sensor to become a teach objective in the case where plural sensors are connected.	-
Mechanism No.	The manual operation mechanism at the moment when a command is selected is selected as the mechanism to carry out search motions, and is displayed here. When the manipulator is selected, it becomes the one direction search, and when the external axis is selected, it becomes the external axis search. Editing is not available, so if a wrong mechanism is displayed, exit teaching by reset once, and newly select the correct mechanism and select the command once again.	-
Store No.	This sets the number of deviation length detected by SF1. Detected deviation is stored per number. Stored deviation can be received by SF3 command.	1 ~ 999
Dev. Composition	When deviation has been already stored by the same store number, when the deviation composition is set to "ON", the detected deviation and the already stored deviation can be composed. When the deviation composition is set to "OFF", the detected deviation is overwritten.	Disabled/ Enabled
Touch Logic	This designates the logic of touch detection. This judges the touch detection from changes from the wire off status to the wire touch status by "OFF → ON", while from the wire touch status to the wire off status by "ON → OFF".	OFF → ON/ ON → OFF
Search Speed	This is the speed of search motions at playback. When the search speed is increased, that tact time is shortened, but the detection precision decreases, and coasting amount after touching increases. (The wire may be bent.) Set around 60cm/min (10mm/sec) as a standard amount. Set an appropriate search speed according to your purpose.	Manipulator: 6 ~ 360 [cm/min] External axis: 1 ~ 100 [%]
Search range (Max.) Search range (Max.)	This sets the maximum searching distance so as to detect the search idle run at execution of SF1. When after start of search, touch is not detected in motions on set distance from the reference position, a failure is detected.	Manipulator: 0 ~ 999 [mm] Slider: 0 ~ 9999 [mm] Positioner: 0 ~ 180 [deg]
Search range (Min.)	This sets the minimum searching distance so as to detect that the wire has already touched at execution of SF1. When after start of search, touch is detected in motions on set distance, a failure is detected.	Manipulator: 0 ~ 999 [mm] Slider: 0 ~ 9999 [mm] Positioner: 0 ~ 180 [deg]
Over deviation range	This sets the allowable range of wire extension deviation detected by SF1. When the wire extension deviation after touch detection is larger than the set value, a failure is detected.	Manipulator: 0 ~ 999 [mm] Slider: 0 ~ 9999 [mm] Positioner: 0 ~ 180 [deg]
Store Coordinates	This designates the coordinate system to store the deviation. In the list box, coordinate systems that can be selected are displayed, so select one. At the external axis search, it is fixed to each axis coordinate system.	Manipulator: Machine / Tool / Other

		External axis: Axis
Tool No.	This sets the tool number.	1 ~ 32
Taught Vector	When search vectors are taught, the value is set. The value cannot be edited.	—

■ **Example of screen display**

SF1[150cm/m, No.1, M1] FN471; One Direction Search

Parameters displayed on the screen are "Search speed", "Store No.", and "Mechanism No." from the left.

See
SF3; Deviation call (FN473)
SF4; Dev. vector composition (FN474)

Function commands (FN codes)

Command name	SF3
FN code	473
Title name	Deviation call
General description	This receives the stored deviation. * This is used when the touch sensor (AX-WD) and the laser search (AX-RD) are connected.

■ **General description**

In SF3, this receives the stored deviation, and corrects positions. To the teaching position of the manipulator, the position of the manipulator is corrected by the deviation received by SF3.

- The objective to be corrected by SF3 is the move command (LIN / CIR / JOINT).
- By receiving the deviation by the touch sensor, only the position is corrected. The torch posture is not changed.
- SF3 is a section command. Designate receiving before the step at which you want to start correction, and teach the end of receiving before the step at which you want to end the correction.
- In the receiving section, irresponsive of presence or absence of an arc command, all the teaching points become objectives for receiving.
- Receiving deviation can be made only between devices of the same mechanism kind. For example, deviation stored by a positioner cannot be received by a manipulator.
- Deviation receiving section can be designated per mechanism. Use this function with sufficient care on up to where and which deviation should be reflected.
- Receiving deviation is made in the coordinate system where deviation is stored, which please note. Especially, as for deviation files, files stored by various kinds of mechanisms exist. For example, if a receiving number is wrong, deviation stored by the neighbor manipulator is received.
- In the case where deviation is received between units or so, there may be the case when there is no stored coordinate system. In such a case, receiving is not carried out (movement toward the teaching point), which please note.
- Only one deviation amount to be received by one mechanism in a section. When to simultaneously reflect (receive) plural search results, convert plural search results into one deviation composed by SF4 in advance.

As for details on the respective functions, teach examples and so forth of the touch sensor (AX-WD), and the laser search (AX-RD), refer to the respective instruction manuals. This help explains its outline.

■ **Parameter**

Parameter name	Description	Range
Mechanism No.	This sets the mechanism to receive deviation in the case where there are plural mechanisms. It cannot be set in a single manipulator machine.	-
Call No.	This sets the number of the deviation to be received. It is necessary to store the deviation by SF1 in advance.	1 ~ 999
Posture calling	This select whether or not to receive the posture deviation. Select only "Disabled". Even if "Enabled" is selected, the posture deviation is not received.	Disabled / Enabled
SF3 Section	This selects start / end / all end of SF3 section.	Start / End / All End
X direction offset	This is set when to make offset further to the received deviation.	-99.9 ~ 99.9
Y direction offset		
Z direction offset		

■ **Example of screen display**

SF3[No.1, ON, M1] FN473;Deviation call

Parameters displayed on the screen show "Call No.", "Section Start or End", and "Mechanism No." from the left.

See
When using the touch sensor:
SF1; One Direction Search (Touch) (FN471)
SF4; Dev. vector composition (FN474)

When using the laser search:
ZF1; One Direction Search (Laser) (FN480)
SF4; Dev. vector composition (FN474)

Function commands (FN codes)

Command name	SF4
FN code	474
Title name	Dev. vector composition
General description	This calculates a new deviation on the basis of stored deviation. * This is used when the touch sensor (AX-WD) and the laser search (AX-RD) are connected.

■ **General description**

In SF4, a new deviation is calculated on the basis of stored deviation. And at the same time, by receiving this deviation, positions may be corrected.

As for details on the respective functions, teach examples and so forth of the touch sensor (AX-WD), and the laser search (AX-RD), refer to the respective instruction manuals. This help explains its outline.

■ **Parameter**

Parameter name	Description	Range
DEV file 1 No.	This designates the number of the DEV file 1 to calculate the deviation.	1 ~ 999
DEV file 1 Rate	This designates the composition ratio of the DEV file 1 to calculate the deviation.	-100 ~ 100 [%]
DEV file 2 No.	This designates the number of the DEV file 2 to calculate the deviation.	1 ~ 999
DEV file 2 Rate	This designates the composition ratio of the DEV file 2 to calculate the deviation.	-100 ~ 100 [%]
DEV file 3 No.	This designates the number of the DEV file 3 to calculate the deviation.	1 ~ 999
DEV file 3 Rate	This designates the composition ratio of the DEV file 3 to calculate the deviation.	-100 ~ 100 [%]
Dev. storage No.	This select start / end / all end of SF3 section.	1 ~ 999

SF4 is calculated on the basis of the above parameters by the following equation.

Deviation to be calculated (stored)
＝（DEV file 1×Rate 1）＋（DEV file 2× Rate 2）＋（DEV file 3×Rate 3）

■ **Example of screen display**

SF4[001,002,003,No.004] FN474;Dev.vector composition

Parameters displayed on the screen show "DEV file 1 No.", "DEV file 2 No.", "DEV file 3 No.", and "Store No." from the left.

See
When using the touch sensor:
SF1; One Direction Search (Touch) (FN471)
SF3; Deviation call (FN473)

When using the laser search:
ZF1; One Direction Search (Laser) (FN480)
SF3; Deviation call (FN473)

Function commands (FN codes)

Command name	SF9
FN code	479
Title name	Generation of a GAP file
General description	This stores variable values to a gap file. * This is used when the touch sensor (AX-WD) and the laser search (AX-RD) are connected.

■ **General description**

By use of this function command, the values of designated actual variables (global actual numbers, local actual numbers) can be stored into a gap file (GAP***) as gap values. In the case when there is not a gap file, one is newly created and gap values are stored into it. And gap values designated from gap file to variables can be obtained.

Thereby, in the case where there is restriction in available "gap file" or 3 gap values or more are to be composed, gap values can be saved to variables (and can be composed by calculation by register), accordingly more gap values can be handled.

As for details on the respective functions, teach examples and so forth of the touch sensor (AX-WD), and the laser search (AX-RD), refer to the respective instruction manuals. This help explains its outline.

■ **Parameter**

Parameter name	Description	Range
Storing direction	This designates the method (storage direction) to handle generated gap value.	Register->File / File->Register
Register Kind	This designates the kind of actual variable to store gap value.	Local / Global
Register No.	This designates the number of actual variable to store gap value.	1 ~ 200
GAP file No.	This designates the number of a gap file to store gap value.	1 ~ 999
Data kind	This designates the kind of data to be obtained.	Root Gap / Gap Depth / Gap Angle 1 / Gap Angle 2

■ **Example of screen display**

SF9[R->F,G001,GAP001] FN479;Generation of a GAP file

See
SF1; One Direction Search (Touch) (FN471)
SF3; Deviation call (FN473)

Function commands (FN codes)

Command name	ZF1
FN code	480
Title name	One Direction Search (Laser)
General description	This detects the setting deviation of a workpiece. * This is used when the laser search (AX-RD) is connected.

■ General description

ZF1 is the command that detects the deviation of a workpiece from the positional difference between the search point at teaching and the touch point at automatic operation, and records it to the deviation correction file.

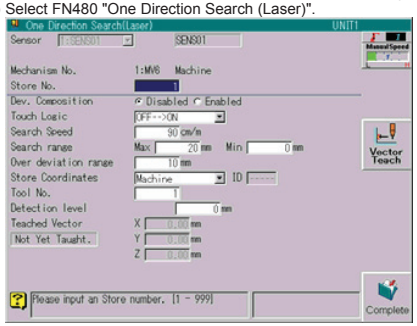
- The deviation of a workpiece to be detected by ZF1 is in one direction. When a workpiece displaces in 3 dimensional manners, carry out ZF1 in 3 directions.
- Only by executing ZF1, the deviation of a workpiece cannot be corrected. To correct the deviation, teaching of SF3 (receiving deviation amount) is necessary.
- In ZF1, there are the following 2 methods.
  • Search motion to move the manipulator
  • Search motion to rotate or slide the external axis, while keeping the manipulator standstill.

  The latter method is specially called "external axis search".
  The external axis search is available whether the external axis is a slider or a positioner. (However, it is not available in other case than the robot's external axis.) And it is available in both the control methods (coordinate control and simultaneous control).

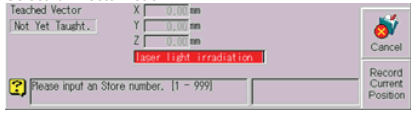
As for details on the respective functions, teach examples and so forth of the laser search (FD-RD), refer to the instruction manual for the laser search. This help explains its outline.

■ ZF1 Teach Method

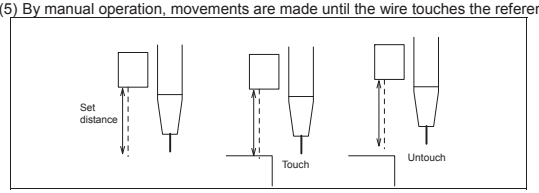
(1) Select the mechanism for search motions as the manual operation mechanism.
(2) Select FN480 "One Direction Search (Laser)".



(3) Set necessary parameters (to be described later).
(4) Select f9 <Vector Teach>.



(5) By manual operation, movements are made until the wire touches the reference surface.



(6) When the wire touches, select f12 <Record Current Position>, and record the reference position.
(7) Press f12 <Complete> and record SF1.

■ Parameter

Parameter name	Description	Range
Sensor	This selects the sensor to become a teach objective in the case where plural sensors are connected.	-
Mechanism No.	The manual operation mechanism at the moment when a command is selected is selected as the mechanism to carry out search motions, and is displayed here. When the manipulator is selected, it becomes the one direction search, and when the external axis is selected, it becomes the external axis search. Editing is not available, so if a wrong mechanism is displayed, exit teaching by reset once, and newly select the correct mechanism and select the command once again.	-
Store No.	This sets the number of deviation length detected by ZF1. Detected deviation is stored per number. Stored deviation can be received by SF3 command.	1 ~ 999
Dev. Composition	When deviation has been already stored by the same store number, when the deviation composition is set to "ON", the detected deviation and the already stored deviation can be composed. When the deviation composition is set to "OFF", the detected deviation is overwritten.	Disabled/ Enabled
Touch Logic	This designates the logic of touch detection. This judges the touch detection from changes from the wire off status to the wire touch status by "OFF → ON", while from the wire touch status to the wire off status by "ON → OFF".	OFF → ON/ ON → OFF
Search Speed	When the search speed is increased, the tact time is shortened, but the detection precision declines. Set an appropriate search speed according to your purpose.	Manipulator: 6 ~ 360 [cm/min] External axis: 1 ~ 100 [%]
Search range (Max.)	This sets the maximum searching distance so as to detect the search idle run at execution of ZF1. When after start of search, touch is not detected in motions on set distance from the reference position, a failure is detected.	Manipulator: 0 ~ 999 [mm] Slider: 0 ~ 9999 [mm] Positioner: 0 ~ 360 [deg]
Search range (Min.)	This sets the minimum searching distance so as to detect laser touch is already made at execution of ZF1. When after start of search, touch is detected in motions on set distance, a failure is detected.	Manipulator: 0 ~ 999 [mm] Slider: 0 ~ 9999 [mm] Positioner: 0 ~ 360 [deg]
Over deviation range	This sets the allowable range of wire extension deviation detected by ZF1. When the wire extension deviation after touch detection is larger than the set value, a failure is detected.	Manipulator: 0 ~ 999 [mm] Slider: 0 ~ 9999 [mm] Positioner: 0 ~ 360 [deg]
Store Coordinates	This designates the coordinate system to store the deviation. In the list box, coordinate systems that can be selected are displayed, so select one. At the external axis search, it is fixed to each axis coordinate system.	Manipulator: Machine / Tool / Other External axis: Axis
Tool No.	This sets the tool number.	1 ~ 32
Detecting level	This sets the reference distance for touch detection. From the sensor nozzle 176 mm → reference distance 0mm 276 mm → reference distance 100mm Normally, the height of the tool top becomes same as that of the reference distance 0mm.	0 ~ 99
Taught Vector	When search vectors are taught, the value is set. The value cannot be edited.	-

■ Example of screen display

ZF1[90cm/m, No.1, M1] FN471;OneDirectionSearch(Laser)

Parameters displayed on the screen shows "Search Speed", "Store No.", and "Mechanism No." from the left.

See
SF3; Deviation call (FN473)
SF4; Dev. vector composition (FN474)

Function commands (FN codes)

Command name	ZG1
FN code	483
Title name	High-speed groove search
General description	This searches the groove information at high speed. * This is used when the laser search (AX-RD) is connected.

■ General description

This carries out a search motion crossing the groove (seam) once, and obtains the groove (seam) shape, and searches the welding position by simple image processing method, and detects deviation from teaching of the welding portion.

As for details on the respective functions, teach examples and so forth of the laser search (FD-RD), refer to the instruction manual for the laser search. This help explains its outline.

■ Parameter

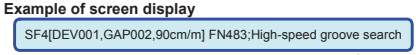
Parameter name	Description	Range
Sensor	This selects the sensor to become a teach objective in the case where plural sensors are connected.	-
Mechanism No.	The manual operation mechanism at the moment when a command is selected is selected as the mechanism to carry out search motions, and is displayed here. When the manipulator is selected, it becomes the one direction search, and when the external axis is selected, it becomes the external axis search. Editing is not available, so if a wrong mechanism is displayed, exit teaching by reset once, and newly select the correct mechanism and select the command once again.	-
File Number DEV	This sets the number of deviation detected by ZG1. Detected deviation is stored per number. Stored deviation amount can be received by SF3 command.	1 ~ 999
File Number GAP	This sets the number of the gap value detected by ZG1.	1 ~ 999
File Number GFF	This sets the number of the GFF file (image processing parameter file) to be used.	1 ~ 999
Search Speed	When the search speed is increased, the tact time is shortened, but the detection precision declines. Set an appropriate search speed according to your purpose.	Manipulator: 6 ~ 360 [cm/min] External axis: 1 ~ 100 [%]
Store Coordinates	This designates the coordinate system to store the deviation. In the list box, coordinate systems that can be selected are displayed, so select one. At the external axis search, it is fixed to each axis coordinate system.	Manipulator: Machine / Tool / Other External axis: Axis
Tool No.	This sets the tool number.	1 ~ 32
Dev. composition	When deviation has been already stored by the same store number, when the deviation composition is set to "ON", the detected deviation and the already stored deviation can be composed. When the deviation composition is set to "OFF", the detected deviation is overwritten.	Disabled/ Enabled
Auto modify	In the trial operation of the soft key, if the trail operation is selected to ON, trial operation mode gets in. However, in the section where this parameter is set to OFF, acquisition of the reference position is not carried out (trial operation is not carried out), and normal deviation detection mode gets in. Speaking reversely, only the section where this parameter is set to ON become the objective for reference position acquisition. The above contents are summarized as shown below.	ON/OFF

Automatic correction setting	Trial operation setting	Result
OFF (Not designated)	OFF	Deviation detection motion (normal motion)
OFF (Not designated)	ON	Deviation detection motion (normal motion)
ON (Designated)	OFF	Deviation detection motion (normal motion)
ON (Designated)	ON	Reference position acquisition motion (without deviation detection)

At teaching, this parameter becomes valid. And even in trail operation, it does not get invalid automatically, therefore, if the trail operation is left ON by the setting of the soft key, acquisition of reference position is always executed in the next playback motion, which please note.

Stable waiting time	This sets the stability waiting time at distance measurement.	0.3 ~ 9.9 [sec]
Vertical Base Distance	This is the position to be the reference of deviation store. Coordinates of the reference position are displayed, which can be confirmed easily. • The reference position cannot be registered by teaching. Be sure to execute trial operation to register it. • When correction is erroneous, by checking the reference position, it may be easy to specify the problem.	
Over deviation range	This sets the allowable range of wire extension deviation detected by ZF1. When the wire extension deviation after touch detection is larger than the set value, a failure is detected.	Manipulator: 0 ~ 999 [mm] Slider: 0 ~ 9999 [mm] Positioner: 0 ~ 360 [deg]
Max. Gap value	This parameter is the one to be used in the image processing portion. This sets the maximum expected gap value. Designate the maximum value and the minimum value of the gap value obtained as the result of image processing by the parameter "gap watch value".	0.0 ~ 99.9 [mm]
Min. Depth value	This sets the minimum depth value to be used in the process of image processing. As same as the above "maximum gap value", this may be used or not according to applicable shapes.	0.0 ~ 99.9 [mm]
Gap watch range (Max.)	When the gap value detected in the image processing exceeds the value of this parameter, it is judged as a failure. As for the troubleshooting, designation can be made by sensor constant.	-999 ~ 999 [mm]
Gap watch range (Min.)	When the gap value detected in the image processing is below the value of this parameter, it is judged as a failure. As for the troubleshooting, designation can be made by sensor constant.	-999 ~ 999 [mm]
Groove depth range (Max.)	When the groove depth detected in the image processing exceeds the value of this parameter, it is judged as a failure. As for the troubleshooting, designation can be made by sensor constant.	-999 ~ 999 [mm]
Groove depth range (Min.)	When the groove depth detected in the image processing is below the value of this parameter, it is judged as a failure. As for the troubleshooting, designation can be made by sensor constant.	-999 ~ 999 [mm]
Groove angle 1 range (Max.)	When the groove angle 1 detected in the image processing exceeds the value of this parameter, it is judged as a failure. As for the troubleshooting, designation can be made by sensor constant.	-360 ~ 360 [deg]
Groove angle 1 range (Min.)	When the groove angle 1 detected in the image processing is below the value of this parameter, it is judged as a failure. As for the troubleshooting, designation can be made by sensor constant.	-360 ~ 360 [deg]
Groove angle 2 range (Max.)	When the groove angle 2 detected in the image processing exceeds the value of this parameter, it is judged as a failure. As for the troubleshooting, designation can be made by sensor constant.	-360 ~ 360 [deg]
Groove angle 2 range (Min.)	When the groove angle 2 detected in the image processing is below the value of this parameter, it is judged as a failure. As for the troubleshooting, designation can be made by sensor constant.	-360 ~ 360 [deg]
Allowable arc dev	This parameter is the one to designate the allowable deviation at circular detection, to be used in the process of image processing. As same as the above "maximum gap value", this may be used or not according to applicable shapes.	0 ~ 100 [%]

■ Example of screen display



Parameters displayed on the screen show "DEV file No.", "GAP file No.", and "Search speed" from the left.

See
SF3; Deviation call (FN473)
SF4; Dev. vector composition (FN474)

Function commands (FN codes)

Command name	ST
FN code	485
Title name	Start Tracking
General description	This starts seam tracking. *This is used when arc sensor (AX-AR/AX-AR2) or TIG arc sensor (AX-TR/AX-TR2) is connected.

■ General description

ST is the start command of seam tracking.
In tracking section, it detects the workpiece deviation while welding, and corrects taught seam.

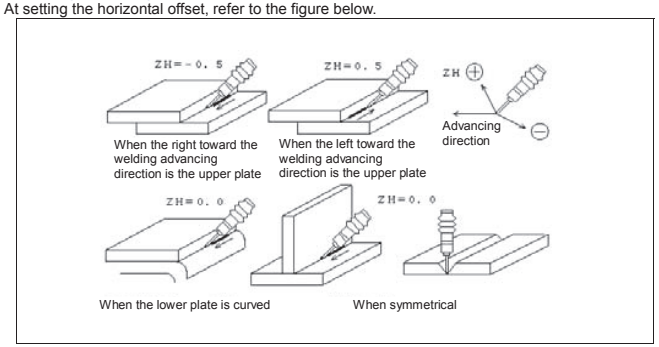
- Teach the move command in tracking section by linear interpolation (LIN) or circular interpolation (CIR).
- Teach ST in the following section.
  Arc Sensor: Welding section and weaving section.
  TIG Arc Sensor: Welding section
- Deviation stored by ET, SF4, and SF5 is data of the designated coordinate system. In the case to shift a task program including motion commands of different coordinate systems, shift may not be available by the coordinate system to be the reference or shift may be made in unexpected direction, which please note.

■ Parameter

Parameter name	Description	Range
Sample data No.	This is the parameter necessary for the arc sensor, and the basic data to set the values to be used in common according to mainly joint kinds and so forth.	1 ~ 50
Chasing sensibility (Horizontal)	<Arc Sensor> This sets the speed to correct the detected deviation. 1 (slow) → 5 (fast) When 0 is set here, correction is not carried out. When the chasing speed is fast, position is corrected fast to deviation occurrence, but when it is too fast, bead snakes by seam tracking becomes conspicuous. On the contrary, when the chasing speed is slow, bead snaking decreases and becomes near taught track, but large deviation may not be corrected. In the arc sensor, current deviation is detected, and reflected to positions after that, so in the case of a workpiece with a large rotation deviation, tool center may see offset. To make the status as small as possible, it is necessary to set the left and right chasing speed large. <TIG Arc Sensor> Set 0 here.	0 ~ 5
Chasing sensibility (Vertical)	This sets the speed to correct the detected deviation. 1 (slow) → 5 (fast) When 0 is set here, correction is not carried out. When the chasing speed is fast, position is corrected fast to deviation occurrence, but when it is too fast, changes of extension by seam tracking becomes conspicuous. On the contrary, when the chasing speed is slow, extension changes from the desired value (changes of extension cannot be corrected).	0 ~ 5
Offset (Horizontal)	<Arc Sensor> This sets the offset amount to the detected deviation. <TIG Arc Sensor> Set 0 here.	-9.999 ~ 9.999 [mm]
Offset (Vertical)	This sets the offset amount to the detected deviation.	-9.999 ~ 9.999 [mm]
Chasing coordinates	This sets the chasing method of manipulator.	torch / work
Tracking deviation range	This sets the maximum value of correction in tracking function. Erroneous deviation of a set workpiece, and tracking deviation by some problem can be detected. Set about +5 ~ -10mm to the expected maximum deviation of workpiece. When the total of the deviation corrected by tracking function (total of vertical direction and horizontal direction) exceeds this value, "processing at correction amount monitor value over" is executed.	1.0 ~ 99.9 [mm]
Undetecting range	Not in use at present	
Arc stable surveillance value	Not in use at present	
Wire speed stable surveillance value	Not in use at present	
Arc Standard Voltage	<TIG Arc sensor only> Input the actual arc voltage. Set the value to realize the desired stand-off. Normally, the value should be between 5V and 15V (10V in average). TIG arc sensor controls the distance between the tip point of electrode and workpiece according to the arc refenece value.	0.1 ~ 99.9[V]

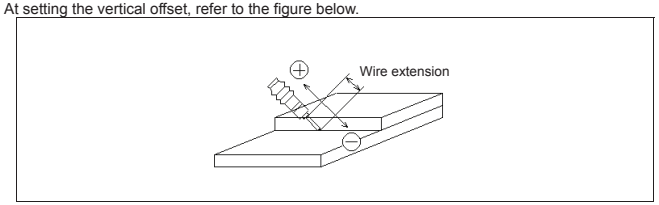
● Offset (Horizontal)

At setting the horizontal offset, refer to the figure below.



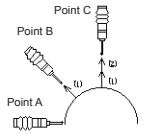
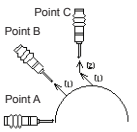
● Offset (Vertical)

At setting the vertical offset, refer to the figure below.



● Chasing coordinates

At setting the chasing method, refer to the figure below.

	Torch reference	Workpiece reference
Correction motion example	 (1) Correction value at point B (2) Correction value at point C	 (1) Correction value at point B (2) Correction value at point C
Correction motion image	Scale	Horizontal movement
Applicable workpiece	- Straight line - Moderate curve - Circle or arc with diameter over 100mm	- Straight line including corner - Circle or arc with diameter below 100mm

■ Example of screen display

```
ST[S1, No.1          ]        FN485;Start Tracking
```

See
SF3; Deviation call (FN473)
SF4; Dev. vector composition (FN474)
SF5; Dev. vector composition (FN475)
ET ; End Tracing (FN486)

Command name	ET
FN code	486
Title name	Tracking end
General description	This ends the seam tracking. * This is used when the arc sensor (AX-AR) is connected.

■ **General description**

ET is the command to end the seam tracking.
In ET, deviation (touch sensor deviation + arc sensor deviation, or arc sensor deviation) in the designated coordinate system in the execution of ET is stored into the deviation correction file of the designated store number.

- Teach ET in the tracking section and the welding section and the weaving section.
- Teach the move command in the tracking section by linear interpolation (LIN) or circular interpolation (CIR).
- When the stored number is further stored, a new deviation is overwritten.
- When to carry out corner winding process or so in the section by use of the deviation stored by ET, designate receiving by SF3 after ET.

■ **Parameter**

Parameter name	Description	Range
Store No.	This sets the number of deviation to be stored. Contents of store may be selected by "Basement of Store". Detected deviation is stored per number. Stored deviation can be received by SF3 command.	1 ~ 999
Endpoint detection	This sets the end point detection whether disabled or enabled. Be sure to set this to "Disabled".	Disabled/ Enabled
Endpoint detection offset	Not in use at present	0 ~ 999 [mm]
Endpoint detection range	Not in use at present	0 ~ 999 [mm]
Store Coordinates	This selects an store coordinate system. Coordinate systems for selection are displayed, so select the one you want.	-
Basement of Store	This designates the contents to be stored. Tracking only : Only the amount corrected by the arc sensor is stored as deviation. In the receiving section of SF3, received amount becomes subtracted amount. Teaching Point : The deviation of the present torch position to the teaching point is stored. In the receiving section of SF3, its amount is also included. It is convenient for carrying out corner winding process or so.	Tracking only / Teaching Point
Keep compensation	This designates whether to keep the tracking correction amount at the moment of ET, or to move as taught toward the next teaching point. Correction is kept until arc end command.	Disabled/ Enabled

● **"K**eep compensation**"**

As for "Keep compensation" parameter movements, refer to the table below.

Keep compensation	SF3 receiving	Movement
Disabled	None	It moves toward the next teaching point. (Without correction)
	In section	Only correction in SF3 is continued. (It moves to the point corrected by only SF3.)
	Receiving changed to just after ET	Only correction amount newly designated is continued. At this moment, do not designate the store number by ET. (It may go to an unexpected position.)
Enabled	None	It is continuously corrected by the deviation corrected by the arc sensor.
	In section	The deviation corrected by the arc sensor and the deviation designated in SF3 are continued simultaneously.
	Receiving changed to just after ET	The newly designated correction amount and the deviation corrected by the arc sensor are continued simultaneously. At this moment, do not designate the store number by ET. (It may go to an unexpected position.)

■ **Example of screen display**

```
ET[S1, Store No.1          ]        FN486;End Tracking
```

See
SF3; Deviation call (FN473)
SF4; Dev. vector composition (FN474)
ST ; Start Tracing (FN485)

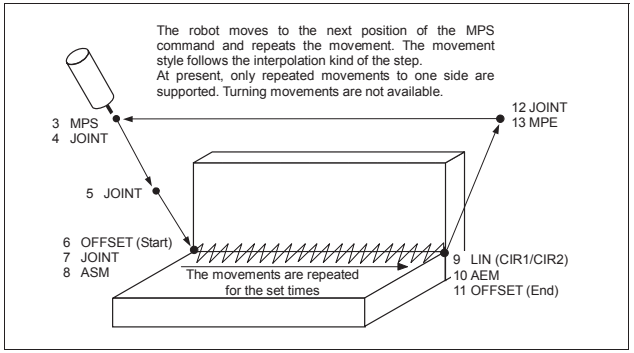
Command name	MPS
FN code	496
Title name	Multi Pass Section Start
General description	This represents the start position of the section where a series of movements for multi-pass welding is repeated.

■ **General description**

This is the command to represent the start of multi-pass section. For teach and playback, the optional software "Multi-pass Welding function" is required.
The multi-pass section means the section where taught movements including not only welding but also approach movements to workpiece and retreat movements are carried out repeatedly.

As for details, refer to the multi-pass welding function instruction manual (optional).

■ **Example of operation**



(1) Joint interpolation movements are carried out from the position of step 2 to the teach position of step 4. In the conventional DAIHEN multi-pass welding function, since MPS had position data, the robot moved to the teach position of MPS. While, in the AX controller, MPS does not have position data, the robot moves to the next step of MPS.

(2) The robot moves to the arc start point of step 7, and starts welding under the multi-pass welding condition taught by ASM.

(3) After completion of welding, the robot moves to the position of step 12.

(4) If the number of repetitions in the multi-pass section is smaller than the number of repetitions set by MPS (3 times), the step execution shifts to MPS. The robot moves to the position of step 4.

(5) The robot carries out the steps 5 ~ 12 once again.

(6) After execution of MPE of step 13, if repetition movements are completed by the set number, the robot moves to the position of step 14, and completes its movements.

■ **Parameter**

Parameter No. 1	Variable No.	The number of the integer variable to count the number of passes is designated here. (1 - 200) As for the integer variable to be designated here, designate a number different from the integer variable to be used for other applications. The pass number now under execution may be confirmed on the integer variable monitor.
Parameter No. 2	Number of the pass	"How many times to be repeated" is designated as the number of passes. (1 - 100)

■ **Example of screen display**

```
MPS[V1%,1,0]          FN496;Multi Pass Section Start
```

See
MPE; Multi Pass Section End (FN497)

Function commands (FN codes)

Command name	MPE
FN code	497
Title name	Multi Pass Section End
General description	This represents the end position of the section where a series of movements for multi-pass welding is repeated.

■ General description

This is the command to represent the end of multi-pass section. For teach and playback, the optional software "Multi-pass Welding function" is required.

As for details, refer to the multi-pass welding function instruction manual (optional).

■ Example of operation

Refer to the multi-pass welding start MPS (FN496).

■ Parameter

None

■ Example of screen display

See
MPS; Multi Pass Section Start (FN496)

Function commands (FN codes)

Command name	EP
FN code	498
Title name	Execution Pass Specification
General description	This designates per pass whether the function commands are to be executed or not in the multi-pass section.

■ General description

This is the command to designate per pass whether the function commands are to be executed or not in the multi-pass section. In the case when this command is not taught, the function commands in the multi-pass area are executed at every execution of each pass.
For teach and playback, the optional software "Multi-pass Welding function" is required.

As for details, refer to the multi-pass welding function instruction manual (optional).

■ Example of operation

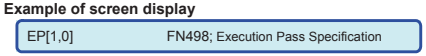
By teaching EP <FN498>, the following movements are available. The designated parameter is shown in [ ].

Step	Command	Contents
:	:	
3	MPS [10 times]	
4	EP [==, 5]	At the stage when the number of passes becomes 5, the waiting program for checking the forming condition is called.
5	CALLP [100]	
6	EP [==, 10]	At the stage when the number of passes becomes 10 (final pass), signal is output to the outside.
7	SET [200]	
8	100% JOINT	
9	100% JOINT	
10	OFFSET [Start]	
11	100% JOINT	
12	ASM	
13	300cm/m LINE	
14	AEM	
15	OFFSET [End]	
16	100% JOINT	
17	MPE	
:	:	

■ Parameter

Parameter No. 1	Condition	One of the following conditions for executing the function commands taught in the next step is selected here. • == (execution only at the designated number of pass) • <= (execution at one over the designated number of pass) • >= (execution at one below the designated number of pass) • Forward (not supported at present) • Return (not supported at present) (0: == / 1: <= / 2: >= / 3: Forward / 4: Return)
Parameter No. 2	Number of the pass	The number of pass for comparison with the first parameter is designated here. (1 - 100)

■ Example of screen display



See
MPS; Multi Pass Section Start (FN496)
MPE; Multi Pass Section End (FN497)

Function commands (FN codes)

Command name	OFFSET
FN code	499
Title name	Multi Offset Specification
General description	This sets offset in the movement steps in the multi-pass welding section.

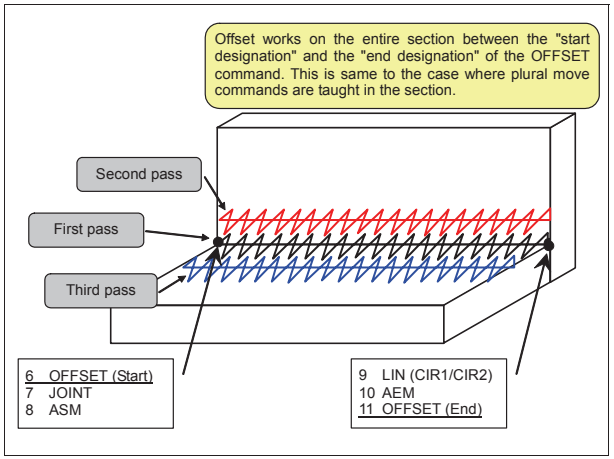
■ General description

This is the command to set the offset in the movement steps in the multi-pass welding section. This is used when to weld the second pass and passes after that with shift from the first pass.
For teach and playback, the optional software "Multi-pass Welding function" is required.

As for details, refer to the multi-pass welding function instruction manual (optional).

■ Example of operation

As shown below, by teaching the OFFSET command to the step just before the arc start point, and further the OFFSET command again to the step just after the arc end point, the entire multi-pass welding section is shifted by the designated offset amount.



■ Parameter

Parameter No. 1	Multi Offset No.	The number of the multi offset condition file is designated here. (1 - 999) This file is to be created in advance in <Arc condition> - [9 Multi offset condition].
Parameter No. 2	Mechanism No.	The mechanism number to which offset is set is designated here. (1 - 9) Normally, since the manipulator having the welding torch is set as mechanism 1, designate "1".
Parameter No. 3	Section Kind	Since the offset becomes valid over the entire multi-pass welding section, whether to "start" or "end" the offset is designated here. (1 : start, 0 : end) Designate "start" for the start position of the multi-pass welding section, while designate "end" for the end position.

■ Example of screen display

```
OFFSET[1,1,1]          FN499; Multi Offset Specification
```

See
MPS; Multi Pass Section Start (FN496)
MPE; Multi Pass Section End (FN497)

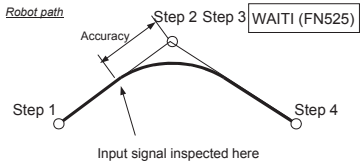
Function commands (FN codes)

Command name	WAITI
FN code	525
Title name	Input signal wait (positive logic)
General description	This command is used to wait for any one general-purpose input signal.

■ General description

When this function command is executed, the robot is made to stand by until the specified general-purpose input signal is input.
It can be recorded in a single action using the [IN] dedicated key on the teach pendant.
It is not possible to wait for a status signal (a signal such as the welding finish signal and start signal whose application has already been assigned). Which signals have been assigned as status signals can be identified on the monitor screen. Signals whose numbers are displayed in the bold italics are status signals so any of the other signals can be awaited.

■ Example of operation 1

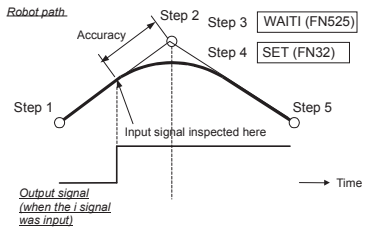


The robot is not decelerated by this command. If no positioning is specified, the input signal is inspected just before the command value reaches the accuracy range of the move command, and if it has been input, an inside arc is drawn. If the input signal has not been input, the command value heads toward the recorded point, and the input signal is inspected when the robot has reached the accuracy range.

In the case of an interlock signal requiring a stringent accuracy, either reduce the accuracy level in step 2 or record positioning "P" in step 2 in the figure above.

■ Example of operation 2

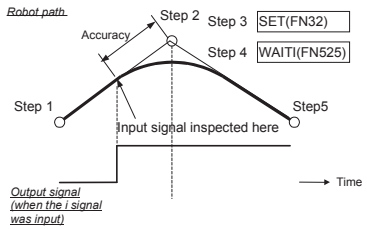
If SET (FN32) has been recorded after WAITI (FN525)



If the input signal has been input, step 4 is executed at the time when the accuracy is matched.
If it has not been input, after arriving at step 2, the input signal is checked, and after it has been input, step 4 is executed.

■ **Example of operation 3**

If SET (FN32) has been recorded before WAITI (FN525)



If the input signal has been input, step 3 is executed at the time when the accuracy is matched.
If it has not been input, step 3: SET is not executed at the above location; instead, after arriving at step 2, step 3: SET is executed. Then, the input signal is checked again, and after it has been input, the robot moves toward step 5.

■ **Parameter**

Parameter No. 1	Input signal number	This specifies the number of input signal to be awaited. When number 5101 or above is specified, multiple input signals can be awaited. (1–2048, 5101–5196)

■ **Example of screen display**

WAITI [I7] FN525; Input signal wait (positive logic)

See
WAITJ: Input signal wait (negative logic) (FN526)
WAIT: Input signal wait with timer (FN552)

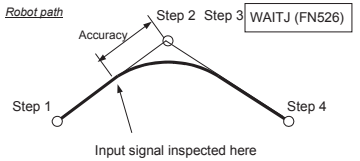
Function commands (FN codes)

Command name	WAITJ
FN code	526
Title name	Input signal wait (negative logic)
General description	This command is used to wait for any one general-purpose input signal using negative logic.

■ **General description**

When this function command is executed, the robot is made to stand by until the one specified general-purpose input signal is set to OFF.
It is not possible to wait for a status signal (a signal such as the welding finish signal and start signal whose application has already been assigned). Which signals have been assigned as status signals can be identified on the monitor screen. Signals whose numbers are displayed in the bold italics are status signals so any of the other signals can be awaited.

■ **Example of operation**



The robot is not decelerated by this command. If no positioning is specified, the input signal is inspected just before the command value reaches the accuracy range of the move command, and if it has been input, an inside arc is drawn. If the input signal has not been input, the command value heads toward the recorded point, and the input signal is inspected when the robot has reached the accuracy range.

In the case of an interlock signal requiring a stringent accuracy, either reduce the accuracy level in step 2 or record positioning "P" in step 2 in the figure above.

■ **Parameter**

Parameter No. 1	Input signal number	This specifies the number of input signal to be awaited. When number 5101 or above is specified, multiple input signals can be awaited. (1 – 2048, 5101 – 5196)

■ **Example of screen display**

WAITJ [I7] FN526; Input signal wait (negative logic)

See
WAITI: Input signal wait (positive logic) (FN525)
WAIT: Input signal wait with timer (FN552)

Function commands (FN codes)

Command name	FETCH
FN code	528
Title name	Fetch Input cond.
General description	Determine judgment the input condition of a following function.

■ General description

This has a parameter I-signal No., it decides judgment the input condition of a following function. Input condition function (ex. WAITI, JMPI, etc.) might cause robot deceleration, because pathway might not be decided immediately before.
'FN258 FETCH' function decides judgment the input condition in any timing, it also prevents from deceleration.

■ Target functions of FETCH function

It is effective for the functions with parameter I signal No.

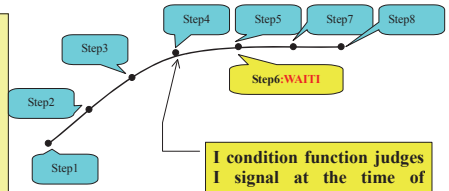
■ Out of target functions of FETCH function

Follows function is off the subject.
・FN29 RINT Robot interrupt (I-condition)

■ Example of operation

Functions with I-cond. in short pitch

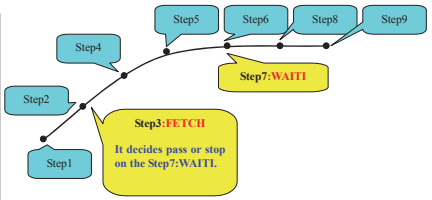
```
1 100% LIN A1 T1
2 100% LIN A8 T1
3 100% LIN A8 T1
4 100% LIN A8 T1
5 100% LIN A8 T1
6 WAITI[I1]
7 100% LIN A8 T1
8 100% LIN A8 T1
…
```



I condition function judges I signal when robot passes a previous move command. Therefore, when the movement section is short the migration pathway might not be decided immediately before, and the robot decelerate.

Using 'FETCH' function

```
1 100% LIN A1 T1
2 100% LIN A8 T1
3 100% LIN A8 T1
3 FETCH[I1]
4 100% LIN A8 T1
5 100% LIN A8 T1
6 100% LIN A8 T1
7 WAITI[I1]
8 100% LIN A8 T1
9 100% LIN A8 T1
…
```



I-condition is decided at the time of executing 'FETCH' function.
It prevents from robot deceleration because the pathway is decided.

■ Parameter

Parameter No. 1	Input signal	I signal No. to fetch.(1-5196)

■ Example of screen display

FETCH[I1] FN528; Fetch Input cond.

■ Effective span of FETCH function

The result of 'FETCH' function keeps until execution a function with fetched I-signal condition. However, when follows occur the effect of FETCH is lost.
(1) Stopping playback. (include Motor-OFF and power-OFF)
(2) Executing 'END'.
(3) Program No. is changed by executing program jump, call, return, and so on.
(4) Executing step jump.

■ Plural execution of FETCH function (1)

When some 'FETCH' functions with same I-signal are executed, the judgment of I-signal is revised whenever it executes.
For example, in case of execution FETCH[I1] - FETCH[I1] - WAITI[I1], the second FETCH is effective for WAITI.

■ Plural execution of FETCH function (2)

Some pairs of FETCH and I-cond. function can be recorded. These works independently.

```
FETCH[I1]
FETCH[I2]
MOVE
WAITI[I1]
MOVE
WAITI[I2]
```

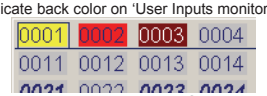
■ Settings

Some error is supported.
Refer to 'Constants: 6 Signals: 1 Signal Condition'

Setting	function
Detect fetch error	Error occurs when a fetched signal status by function FETCH(Fn528) is changed before the intended function working. Choice: Disabled / Enabled
Effective span of a fetched signal	Detect an error when a fetched signal by a function FETCH(Fn528) spends the designated number of move steps. Range: 0～9999（0: disabled）

■ Monitor

Fetched signals indicate back color on 'User Inputs monitor'.



dark red Fetched signal OFF
red Fetched signal ON

The monitor shows state of current unit.

■ Others

'FETCH' function does not work while operating 'Check back'.

'FETCH' function is managed separately each units. Executing 'FETCH' in one unit does not affect other unit.

Function commands (FN codes)

Command name	CNVI
FN code	550
Title name	Conveyor Interlock
General description	Robot waits until conveyor register reaches up to the designated distance, stationarily

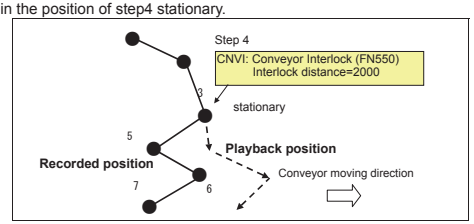
■ General description

This function is used in conveyor synchronizing application.
Robot waits until conveyor register reaches up to the designated distance (= interlock distance). While waiting, robot is stationary. When conveyor register reaches, robot proceeds to the next step. If conveyor register had already reached to the interlock distance before the start of waiting, robot immediately proceeds to the next step.

Please refer to the "Conveyor Synchronization Manual" (option) for detail operations.

■ Example of operation

In case of CNVI : Conveyor Interlock (FN550) is recorded at step4, robot will wait until conveyor register reaches up to 2000mm in the position of step4 stationary.



■ Parameter

Parameter No. 1	Interlock distance	Input the conveyor interlock distance. Linear conveyor; (0-30,000) mm Circular conveyor; (0-180) degree

■ Example of screen display

```
CNVI [3000]          FN550: Conveyor Interlock
```

See
CNVSYNC; Conveyor Counter Reset (FN55)
CNVYSYNCI; Synchronizing Conveyor Interlock (FN562)

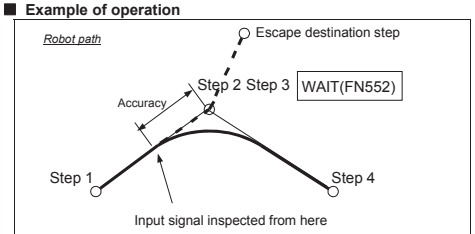
Function commands (FN codes)

Command name	WAIT
FN code	552
Title name	Input signal wait with timer
General description	This command is used to wait for any one general-purpose input signal for up to the specified time.

■ General description

When this function command is executed, the robot is made to stand by until the specified general-purpose input signal is input.
If the input signal is not input even after the specified time has elapsed, the robot jumps to the specified escape step.
The escape step can be designated with a move command or a function command. Bear in mind that when a function command has been designated, the command will be executed straight away.
It is not possible to wait for a status signal (a signal such as the welding finish signal and start signal whose application has already been assigned). Which signals have been assigned as status signals can be identified on the monitor screen. Signals whose numbers are displayed in the bold italics are status signals so any of the other signals can be awaited.

■ Example of operation



The robot is not decelerated by this command. If no positioning is specified, the input signal is inspected just before the command value reaches the accuracy range of the move command, and if it has been input, an inside arc is drawn. If the input signal has not been input, the command value heads toward the recorded point, and the input signal is inspected when the robot has reached the accuracy range.
If the input signal is not set to ON even after the specified time has elapsed, the robot jumps to the specified escape step.
In the case of an interlock signal requiring a stringent accuracy, either reduce the accuracy level in step 2 or record positioning "P" in step 2 in the figure above.

■ Parameter

Parameter No. 1	Input signal number	This specifies the number of input signal to be awaited. When number 5101 or above is specified, multiple input signals can be awaited. (1 – 2048, 5101 – 5196)
Parameter No. 2	Wait time	This specifies the wait unit in seconds. (0.0 – 60.0)
Parameter No. 3	Escape step number	This specifies the number of the step to which the robot is to jump if the input signal is not input even though the wait time has elapsed. Any step in the same program can be designated. (1 – 9999)

■ Example of screen display

```
WAIT [I7, 3.0, 7]          FN552; Timer input signal
```

See
WAITI: Input signal wait (positive logic) (FN525)
WAITJ: Input signal wait (negative logic) (FN526)

Function commands (FN codes)

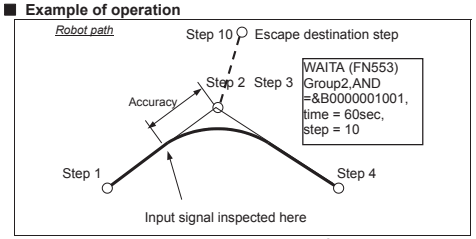
Command name	WAITA
FN code	553
Title name	Wait group input (AND) with timer
General description	This command is used to wait for any of group general-purpose input signal (AND logic) with designated time

■ General description

When this function command is executed, the robot is stand by until the specified group input satisfies the condition that is all designated bit (I signal) should be ON. One “group” is constructed by 10 general-purpose input signals (I1~I2048) as shown below.
It is not possible to wait for a status signal (a signal such as the welding finish signal and start signal whose application has already been assigned). Which signals have been assigned as status signals can be identified on the monitor screen. Signals whose numbers are displayed in the bold italics are status signals so any of the other signals can be awaited.

Group	Input Sig.	Group	Input Sig.	Group	Input Sig.
1	1～10	11	101～110	21	201～210
2	11～20	12	111～120	…	…
3	21～30	13	121～130	30	291～300
4	31～40	14	131～140	…	…
5	41～50	15	141～150	50	491～500
6	51～60	16	151～160	…	…
7	61～70	17	161～170	100	991～1000
8	71～80	18	171～180	…	…
9	81～90	19	181～190	…	…
10	91～100	20	191～200	204	2031～2040

■ Example of operation



The robot is not decelerated by this command. If no positioning is specified, the input signal is inspected just before the command position reaches the accuracy range of the move command, and if it has been input, an inside arc is drawn.
If the input signal has not been input, the robot toward the recorded point and the condition is inspected when the robot has reached.
If the condition is not satisfied even after the specified time has elapsed, the robot jumps to the specified escape step. On this sample as followed table, signals described “ON” should be “ON”, and signals described “any” does not matter ON or OFF.

In case of “GROUP2, AND=&B0000001001”

Output signals	20	19	18	17	16	15	14	13	12	11
ON/OFF status	Any	Any	Any	Any	Any	Any	**ON**	Any	Any	**ON**

In the case of an interlock signal requiring a stringent accuracy, either reduce the accuracy level in step 2 or record positioning "P" in step 2 in the figure above.

■ Parameter

Parameter No. 1	Group number	This specifies the group number. (1-204)
Parameter No. 2	AND condition	This specifies the condition of 10 input signals. “1” should be ON, and “0” does not matter ON/OFF. (0000000000 - 1111111111)
Parameter No. 3	Wait time	This specifies the wait unit in seconds. (0.0 – 60.0)
Parameter No. 4	Escape destination step	This specifies the number of the step to which the robot is to jump if condition is not satisfied even though the wait time has elapsed. Any step in the same program can be designated. (1 – 9999)

■ Example of screen display

```
WAITA[Group2, &B0000101, 60.0, 10]  FN553:Wait group input (AND) with timer
```

See
WAITO; Wait group input (OR) with timer（FN554）
WAITE; Wait group input with timer（FN555）

Function commands (FN codes)

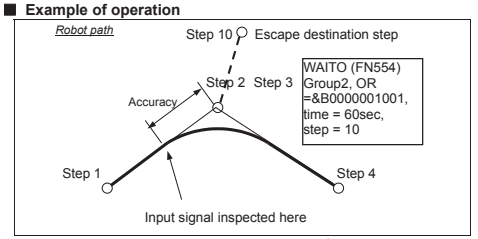
Command name	WAITO
FN code	554
Title name	Wait group input (OR) with timer
General description	This command is used to wait for any of group general-purpose input signal (OR logic) with designated time

■ General description

When this function command is executed, the robot is stand by until the specified group input satisfies the condition that is at least one designated bit (I signal) should be ON. One “group” is constructed by 10 general-purpose input signals (I1~I2048) as shown below.
It is not possible to wait for a status signal (a signal such as the welding finish signal and start signal whose application has already been assigned). Which signals have been assigned as status signals can be identified on the monitor screen. Signals whose numbers are displayed in the bold italics are status signals so any of the other signals can be awaited.

Group	Input Sig.	Group	Input Sig.	Group	Input Sig.
1	1～10	11	101～110	21	201～210
2	11～20	12	111～120	…	…
3	21～30	13	121～130	30	291～300
4	31～40	14	131～140	…	…
5	41～50	15	141～150	50	491～500
6	51～60	16	151～160	…	…
7	61～70	17	161～170	100	991～1000
8	71～80	18	171～180	…	…
9	81～90	19	181～190	…	…
10	91～100	20	191～200	204	2031～2040

■ Example of operation



The robot is not decelerated by this command. If no positioning is specified, the input signal is inspected just before the command position reaches the accuracy range of the move command, and if it has been input, an inside arc is drawn.
If the input signal has not been input, the robot toward the recorded point and the condition is inspected when the robot has reached.
If the condition is not satisfied even after the specified time has elapsed, the robot jumps to the specified escape step. On this sample as followed table, at least one signal described “ON” should be “ON”, and signals described “any” does not matter ON or OFF.

In case of “GROUP2, OR=&B0000001001”

Output signals	20	19	18	17	16	15	14	13	12	11
ON/OFF status	Any	Any	Any	Any	Any	Any	**ON**	Any	Any	**ON**

In the case of an interlock signal requiring a stringent accuracy, either reduce the accuracy level in step 2 or record positioning "P" in step 2 in the figure above.

■ Parameter

Parameter No. 1	Group number	This specifies the group number. (1-204)
Parameter No. 2	OR condition	This specifies the condition of 10 input signals. At least one “1” signal should be ON, and “0” does not matter ON/OFF. (0000000000 - 1111111111)
Parameter No. 3	Wait time	This specifies the wait unit in seconds. (0.0 – 60.0)
Parameter No. 4	Escape destination step	This specifies the number of the step to which the robot is to jump if condition is not satisfied even though the wait time has elapsed. Any step in the same program can be designated. (1 – 9999)

■ Example of screen display

```
WAITO[Group2, &B0000101, 60.0, 10]  FN554:Wait group input (OR) with timer
```

See
WAITA; Wait group input (AND) with timer（FN553）
WAITE; Wait group input with timer（FN555）

Command name	WAITE
FN code	555
Title name	Wait group input with timer
General description	This command is used to wait for any of group general-purpose input signal with designated time

■ General description

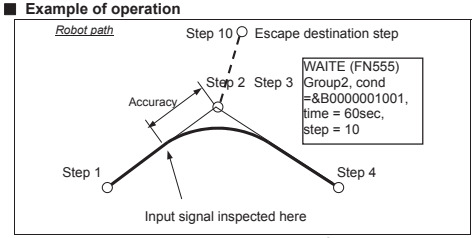
When this function command is executed, the robot is stand by until the specified group input satisfies the condition that is all bit (I signal) ON/FF should coincident. One “group” is constructed by 10 general-purpose input signals (I1~I2048) as shown below.
It is not possible to wait for a status signal (a signal such as the welding finish signal and start signal whose application has already been assigned). Which signals have been assigned as status signals can be identified on the monitor screen. Signals whose numbers are displayed in the bold italics are status signals so any of the other signals can be awaited.

Group	Input Sig.
1	1～10
2	11～20
3	21～30
4	31～40
5	41～50
6	51～60
7	61～70
8	71～80
9	81～90
10	91～100

Group	Input Sig.
11	101～110
12	111～120
13	121～130
14	131～140
15	141～150
16	151～160
17	161～170
18	171～180
19	181～190
20	191～200

Group	Input Sig.
21	201～210
…	…
30	291～300
…	…
50	491～500
…	…
100	991～1000
…	…
…	…
204	2031～2040

■ Example of operation



The robot is not decelerated by this command. If no positioning is specified, the input signal is inspected just before the command position reaches the accuracy range of the move command, and if it has been input, an inside arc is drawn.
If the input signal has not been input, the robot toward the recorded point and the condition is inspected when the robot has reached.
If the condition is not satisfied even after the specified time has elapsed, the robot jumps to the specified escape step. On this sample as followed table, all signals described “ON” should be “ON”, and all signals described “OFF” should be “OFF”.

In case of “GROUP2, cond.=&B0000001001”

Output signals	20	19	18	17	16	15	14	13	12	11
ON/OFF status	Off	Off	Off	Off	Off	Off	**ON**	Off	Off	**ON**

In the case of an interlock signal requiring a stringent accuracy, either reduce the accuracy level in step 2 or record positioning "P" in step 2 in the figure above.

■ Parameter

Parameter No. 1	Group number	This specifies the group number. (1-204)
Parameter No. 2	condition	This specifies the condition of 10 input signals. “1” signal should be ON, and “0” signal should be “OFF”. (0000000000 - 1111111111)
Parameter No. 3	Wait time	This specifies the wait unit in seconds. (0.0 – 60.0)
Parameter No. 4	Escape destination step	This specifies the number of the step to which the robot is to jump if condition is not satisfied even though the wait time has elapsed. Any step in the same program can be designated. (1 – 9999)

■ Example of screen display

```
WAITE[Group2, &B0000010I, 60.0, 10]  FN555:Wait group input with timer
```

See
WAITA; Wait group input (AND) with timer（FN553）
WAITO; Wait group input (OR) with timer（FN554）

Command name	WAITAD
FN code	558
Title name	Wait group input BCD (AND) with timer
General description	This command is used to wait for any of group general-purpose input signal (AND logic) with designated time

■ General description

When this function command is executed, the robot is stand by until the specified group input satisfies the condition that is all designated bit (I signal) should be ON.
The only one difference from WAITA (FN553) is how to designate condition that is here BCD (Binary Coded Decimal).

■ Example of operation

In case of “condition = 63”, signals described “ON” should be “ON”, and signals described “any” does not matter ON or OFF.

In case of “GROUP2, AND=63”

Output signals	20	19	18	17	16	15	14	13	12	11
ON/OFF status	Any	Any	Any	**ON**	**ON**	Any	Any	Any	**ON**	**ON**
BCD	0		6				3			

■ Parameter

Parameter No. 1	Group number	This specifies the group number. (1-204)
Parameter No. 2	AND condition	This specifies the condition of 10 input signals. “1” should be ON and “0” does not matter ON/OFF, then describe it by BCD, (0 - 399)
Parameter No. 3	Wait time	This specifies the wait unit in seconds. (0.0 – 60.0)
Parameter No. 4	Escape destination step	This specifies the number of the step to which the robot is to jump if condition is not satisfied even though the wait time has elapsed. Any step in the same program can be designated. (1 – 9999)

■ Example of screen display

```
WAITAD[Group2, 63,  60.0,  10]    FN558; Wait group input BCD (AND) with timer
```

See
WAITA; Wait group input (AND) with timer（FN553）

Function commands (FN codes)

Command name	WAITOD
FN code	559
Title name	Wait group input BCD (OR) with timer
General description	This command is used to wait for any of group general-purpose input signal (OR logic) with designated time

■ General description

When this function command is executed, the robot is stand by until the specified group input satisfies the condition that is at least one designated bit (I signal) should be ON.
The only one difference from WAITO (FN554) is how to designate condition that is here BCD (Binary Coded Decimal).

■ Example of operation

In case of “condition = 63”, at least one signal described “ON” should be “ON”, and signals described “any” does not matter ON or OFF.

In case of “GROUP2, OR=63”

Output signals	20	19	18	17	16	15	14	13	12	11
ON/OFF status	Any	Any	Any	**ON**	**ON**	Any	Any	Any	**ON**	**ON**
BCD	0		6				3			

■ Parameter

Parameter No. 1	Group number	This specifies the group number. (1-204)
Parameter No. 2	AND condition	This specifies the condition of 10 input signals. At least one “1” should be ON and “0” does not matter ON/OFF, then describe it by BCD, (0 - 399)
Parameter No. 3	Wait time	This specifies the wait unit in seconds. (0.0 – 60.0)
Parameter No. 4	Escape destination step	This specifies the number of the step to which the robot is to jump if condition is not satisfied even though the wait time has elapsed. Any step in the same program can be designated. (1 – 9999)

■ Example of screen display

WAITOD[Group2, 63, 60.0, 10] FN559; Wait group input BCD (OR) with timer

See
WAITO; Wait group input (OR) with timer（FN554）

Function commands (FN codes)

Command name	WAITED
FN code	560
Title name	Wait group input BCD with timer
General description	This command is used to wait for any of group general-purpose input signal with designated time

■ General description

When this function command is executed, the robot is stand by until the specified group input satisfies the condition that is all bit (I signal) should be coincident.
The only one difference from WAITE (FN555) is how to designate condition that is here BCD (Binary Coded Decimal).

■ Example of operation

In case of “condition = 63”, signals described “ON” should be “ON”, and signals described “OFF” should be “OFF”.

In case of “GROUP2, cond.=63”

Output signals	20	19	18	17	16	15	14	13	12	11
ON/OFF status	Off	Off	Off	**ON**	**ON**	Off	Off	Off	**ON**	**ON**
BCD	0		6				3			

■ Parameter

Parameter No. 1	Group number	This specifies the group number. (1-204)
Parameter No. 2	AND condition	This specifies the condition of 10 input signals. “1” should be ON and “0” should be “OFF”, then describe it by BCD, (0 - 399)
Parameter No. 3	Wait time	This specifies the wait unit in seconds. (0.0 – 60.0)
Parameter No. 4	Escape destination step	This specifies the number of the step to which the robot is to jump if condition is not satisfied even though the wait time has elapsed. Any step in the same program can be designated. (1 – 9999)

■ Example of screen display

WAITED[Group2, 63, 60.0, 10] FN558; Wait group input BCD with timer

See
WAITE; Wait group input with timer（FN555）

Application Command (FN Code)

Command name	CNVYSYNCI
FN code	562
Title name	Sync conveyor interlock
Outline	Used to make the robot wait until the conveyor travels a distance specified by the conveyor register while following the conveyor.

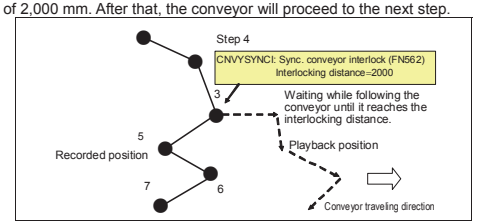
■ Outline

The CNVYSYNCI command is an application command dedicated to synchronization with the conveyor.
Recording this application command enables the conveyor to wait for interlock until the conveyor travels a specified distance while in playback mode. The robot also moves while following the conveyor even in interlock waiting mode. When the conveyor reaches an interlocking distance, the robot will proceed to the next step. If the conveyor already reaches the interlocking distance when the robot starts waiting for interlock, the robot will not wait but immediately proceed to the next step.

For detail of the conveyor synchronization function, refer to information in the **FD CONTROLLER OPERATING MANUAL "CONVEYOR SYNCHRONIZATION FUNCTION" (OPTION)**.

■ Example of motion

Record "CNVYSYNCI: Sync. conveyor interlock (FN562)" in Step 4.
When playing back this command, the robot will complete positioning in the Step 4, and subsequently wait while following the conveyor until the conveyor register (the current conveyor position found by the current conveyor counter) reaches a distance of 2,000 mm. After that, the conveyor will proceed to the next step.



■ Parameters

Parameter 1	Interlocking distance	Used to enter an interlocking distance. Setting range: 0 to 30000 mm for straight conveyors 0 to 180 deg. for circular conveyors.

■ Example of screen display

```
CNVYSYNCI[3000]    FN562: Sync. conveyor interlock
```

Related commands
CNVSYNC: Conveyor counter reset (FN55)
CNVI: Conveyor interlock (FN550)

Function commands (FN codes)

Command name	PRSI
FN code	564
Title name	Press interlock
Outline	This sets interlock, in the press brake synchronization function.

■ Outline

When the parameter 1 (downward) is designated, "synchronization start position" of the press brake setting data is used, and when the conveyor register becomes larger than the value, the interlock is released. At waiting for interlock, the robot stops and waits.

When the parameter 0 (upward) is designated, "retreat start position" is used, and when the conveyor register becomes smaller than the value, the interlock is released. At waiting for interlock, the robot waits in synchronization.

■ Parameters

Parameter No.1	Conveyor number	This designates a conveyor number registered as press brake. (1-2)
Parameter No.2	Upward/ Downward	This designates punch upward/downward. (Upward: 0 / Downward: 1)

■ Screen display example

```
PRSI[1,0]                FN564; Press interlock
```

Function commands (FN codes)

Command name	SOCKCREATE
FN code	570
Title name	Create Socket
General description	This command is used to create the socket.

■ General description

The socket is made. It is necessary to execute this function before other socket functions are executed when communicating with a socket.

■ Parameter

Parameter No. 1	Socket No.	The socket number used is specified. When the socket number that has already been made is specified, it becomes an error. (1 – 16)
Parameter No.2	TCP/UDP	Whether the specified socket is used with TCP or it uses it with UDP is specified. 0 If TCP and 1 is specified, it becomes UDP if is specified. (0 – 1)

■ Example of screen display

```
SOCKCREATE[1,0]                    FN570; Create the socket
```

See
SOCKCLOSE ; Close the socket (FN571)
SOCKBIND ; Bind the socket (FN572)
SOCKWAIT ; Wait for connect (FN573)
SOCKCONNECT ; Connect to server (FN574)
SOCKSEND ; Send data (FN575)
SOCKSENDSTR ; Send string data (FN576)
SOCKRECV ; Receive data (FN577)

Function commands (FN codes)

Command name	SOCKCLOSE
FN code	571
Title name	Close the socket
General description	This command is used to close the socket.

■ General description

The socket made with SOCKCREATE is cleared, and it puts it into the state that can be used again. When the user task program ends, it is closed automatically.

■ Parameter

Parameter No. 1	Socket No.	The socket number used is specified. When the socket number not made is specified, it becomes an error. (1 – 16)

■ Example of screen display

```
SOCKCLOSE[1]                       FN571; Close the socket
```

See
SOCKCREATE ; Open the socket (FN570)
SOCKBIND ; Bind the socket (FN572)
SOCKWAIT ; Wait for connect (FN573)
SOCKCONNECT ; Connect to server (FN574)
SOCKSEND ; Send data (FN575)
SOCKSENDSTR ; Send string data (FN576)
SOCKRECV ; Receive data (FN577)

Function commands (FN codes)

Command name	SOCKBIND
FN code	572
Title name	Bind the socket
General description	This command is used to assign a socket an port No.

■ General description

When AX Controller is used as a server, the port where the connection from the client is waited is specified for the socket.

■ Parameter

Parameter No. 1	Socket No.	The socket number used is specified. When the socket number not made is specified, it becomes an error. (1 – 16)
Parameter No. 2	Port No.	The waiting port when operating as a server is specified. When the port number used is specified, it becomes an error. (1 – 65535)

■ Example of screen display

```
SOCKBIND[1,48952]                    FN572; Bind the socket
```

See
SOCKCREATE ; Open the socket (FN570)
SOCKCLOSE ; Close the socket (FN571)
SOCKWAIT ; Wait for connect (FN573)
SOCKCONNECT ; Connect to server (FN574)
SOCKSEND ; Send data (FN575)
SOCKSENDSTR ; Send string data (FN576)
SOCKRECV ; Receive data (FN577)

Function commands (FN codes)

Command name	SOCKWAIT
FN code	573
Title name	Wait for connect
General description	This command is waited for until the connection from the client is done to the allocated port.

■ General description

SOCKWAIT is waited for until the connection from the outside is done to the allocated port. Meanwhile, the program stops. When the time-out is done, the reproduction of the program is restarted. SOCKWAIT is used only in the case of TCP.

■ Parameter

Parameter No. 1	Waiting Socket No.	The socket number used is specified. When the socket number not made is specified, it becomes an error. (1 – 16)
Parameter No. 2	Communication Port No.	The socket to connect it with the client that has been connected is specified. When the client has connected it, the socket is made by the specified socket. Therefore, after the communication ends, it is necessary to close by using the SOCKCLOSE function. (1 – 16)
Parameter No. 3	Timeout	The time of connected waiting time-out is set every second. It keeps waiting for the connection without doing the time-out when 0 is set. (0 – 20)

■ Example of screen display

```
SOCKWAIT[1,2,5]                      FN573; Wait for connect
```

See
SOCKCREATE ; Open the socket (FN570)
SOCKCLOSE ; Close the socket (FN571)
SOCKBIND ; Bind the socket (FN572)
SOCKCONNECT ; Connect to server (FN574)
SOCKSEND ; Send data (FN575)
SOCKSENDSTR ; Send string data (FN576)
SOCKRECV ; Receive data (FN577)

Function commands (FN codes)

Command name	SOCKCONNECT
FN code	574
Title name	Connect to server
General description	This command is used to connect to server.

■ General description

This command is connected with the server for TCP. It is necessary to set IP address with SOCKCONNECT though this command doesn't connect with the server for UDP.

■ Parameter

Parameter No. 1	Socket No.	The socket number used is specified. When the socket number not made is specified, it becomes an error. (1 – 16)
Parameter No. 2	IP Address	The least significant byte of IP address of the server is specified. Other values use the same one as the setting of TCP/IP of the AX Controller. (1 – 254)
Parameter No. 3	Port No.	The port number of the server is specified. (1 – 65535)
Parameter No. 4	Timeout	The time of connected waiting time-out is set every second. (1 – 20)

■ Example of screen display

```
SOCKCONNECT[1,2,48952,5]                FN574; Connect to server
```

See
SOCKCREATE ; Open the socket (FN570)
SOCKCLOSE ; Close the socket (FN571)
SOCKBIND ; Bind the socket (FN572)
SOCKWAIT ; Wait for connect (FN573)
SOCKSEND ; Send data (FN575)
SOCKSENDSTR ; Send string data (FN576)
SOCKRECV ; Receive data (FN577)

Function commands (FN codes)

Command name	SOCKSEND
FN code	575
Title name	Send data
General description	This command is used to transmit the data stored in the specified buffer.

■ General description

This command transmits the data stored in the specified buffer. When it doesn't connect or the other party closes, it becomes an error. When SOCKSEND is done because the other party's cutting cannot be recognized at once, it is likely not to become an error when the other party doesn't do the close processing and it cuts it. When the data of length that exceeds the size of the buffer is transmitted, it is necessary to transmit ,divided into two portions.

■ Parameter

Parameter No. 1	Socket No.	The socket number used is specified. When the socket number not made is specified, it becomes an error. (1 – 16)
Parameter No. 2	Buffer No.	The buffer number is specified. (1 – 16)
Parameter No. 3	Data Length	The size of the transmitted data is set. (1 – 1024)
Parameter No. 4	Timeout	The time of connected waiting time-out is set every second. (1 – 20)
Parameter No. 5	Integer Variable	The variable in which the transmitted data size (unit of the byte) is written is specified.

■ Example of screen display

```
SOCKSEND[1,1,10,5,V1%]                FN575; Send Data
```

See
SOCKCREATE ; Open the socket (FN570)
SOCKCLOSE ; Close the socket (FN571)
SOCKBIND ; Bind the socket (FN572)
SOCKWAIT ; Wait for connect (FN573)
SOCKCONNECT ; Connect to server (FN574)
SOCKSENDSTR ; Send string data (FN576)
SOCKRECV ; Receive data (FN577)

Function commands (FN codes)

Command name	SOCKSENDSTR
FN code	576
Title name	Send string
General description	This command is used to transmit the specified string.

■ General description

This command transmits the specified string. At that time, the terminal character that shows the end of the string can be added. When it doesn't connect or the other party closes, it becomes an error. When SOCKSENDSTR is done because the other party's cutting cannot be recognized at once, it is likely not to become an error when the other party doesn't do the close processing and it cuts it. When the data of length that exceeds the size of the buffer is transmitted, it is necessary to transmit, divided into two portions.

■ Parameter

Parameter No. 1	Socket No.	The socket number used is specified. When the socket number not made is specified, it becomes an error. (1 – 16)
Parameter No. 2	String	The transmitted string data is specified. String data can be specified by the string variable and the string constant.
Parameter No. 3	Data Length	The size of the transmitted data is set. (1 – 1024)
Parameter No. 4	Timeout	The time of connected waiting time-out is set every second. (1 – 20)
Parameter No. 5	Integer Variable	The variable in which the transmitted data size (unit of the byte) is written is specified.
Parameter No. 6	Termination character	The termination character is specified. When the terminal character is specified, the Data Length becomes long only as for the length of the terminal character. (0 – 3) 0 : The terminal character is not added. 1 : It transmits applying "¥r" as a terminal character. 2 : It transmits applying "¥n" as a terminal character. 3 : It transmits applying "¥r¥n" as a terminal character.

■ Example of screen display

```
SOCKSENDSTR[1,"ABC",3,5,V1%,3]          FN576; Send String Data
```

See
SOCKCREATE ; Open the socket (FN570)
SOCKCLOSE ; Close the socket (FN571)
SOCKBIND ; Bind the socket (FN572)
SOCKWAIT ; Wait for connect (FN573)
SOCKCONNECT ; Connect to server (FN574)
SOCKSEND ; Send data (FN575)
SOCKRECV ; Receive data (FN577)

Function commands (FN codes)

Command name	SOCKRECV
FN code	577
Title name	Receive data
General description	This command is used to receive the data.

■ General description

This command is used to receive the data. Operation changes into this command with TCP in UDP.

In the following cases, the step is completed for TCP.
1. The size of the data actually received becomes more than the receive data length.
2. Other party's telecommunications equipment closed the socket.
3. The timeout was done.

The reception of data is waited for when the received data size is smaller than the set receive data size. This command is completed, when receive data reaches the set data size.

In the following cases, the step is completed for UDP.
1. Data was received.
2. The timeout was done.

UDP stores the data that was able to be received in the buffer when data is received, and completes the function. Therefore, the data length actually received might be smaller than the specified receive data length. The excess data is cut short, when the data length actually received is larger than the specified receive data length, and the error is returned.
Moreover, it is possible to reply to the other party of the communication by doing SOCKSEND after it receives the data. It is necessary to set the destination again by using SOCKCONNECT to transmit data to other parties of the communication.

■ Parameter

Parameter No. 1	Socket No.	The socket number used is specified. When the socket number not made is specified, it becomes an error. (1 – 16)
Parameter No. 2	Buffer No.	The number of the buffer where the received data is stored is specified. (1 – 16)
Parameter No. 3	Data Length	The size of the receive data is set. (1 – 1024)
Parameter No. 4	Timeout	The time of connected waiting time-out is set every second. (1 – 20)
Parameter No. 5	Integer Variable	The variable in which the received data size (unit of the byte) is written is specified. The data size actually received might be smaller than the value specified by the receive data length.

■ Example of screen display

```
SOCKRECV[1,1,3,5,V1%]                   FN577; Receive data
```

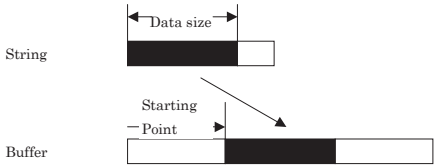
See
SOCKCREATE ; Open the socket (FN570)
SOCKCLOSE ; Close the socket (FN571)
SOCKBIND ; Bind the socket (FN572)
SOCKWAIT ; Wait for connect (FN573)
SOCKCONNECT ; Connect to server (FN574)
SOCKSEND ; Send data (FN575)
SOCKSENDSTR ; Send string data (FN576)

Function commands (FN codes)

Command name	SETSTR
FN code	580
Title name	Set buffer (string)
General description	This command is used to stored string at an arbitrary position in the buffer

■ General description

This command is used to stored string at an arbitrary position in the buffer. When the writing string overflows of the buffer (When starting position + data size exceeds the size of the buffer), the controller outputs "E2250 step data is abnormal".



■ Parameter

Parameter No. 1	Buffer No.	The number of the buffer to store data is specified. (1 – 16)
Parameter No. 2	String	The string stored in the buffer is specified. The string data can be specified by the string variable and the string constant.
Parameter No. 3	Starting point	The stored position of the buffer is specified. (0 – 1023)
Parameter No. 4	Data size	The data size of the data stored in the buffer is specified. (1 – 199)

■ Example of screen display

```
SETSTR[1,"ABC",0,3]                    FN580; Set buffer (string)
```

See
SETINT ; Set buffer(integer) (FN581)
SETREAL ; Set buffer(real) (FN582)
SETBYTE ; Set buffer(byte) (FN583)
GETSTR ; Get buffer(string) (FN584)
GETINT ; Get buffer(integer) (FN585)
GETREAL ; Get buffer(real) (FN586)
GETBYTE ; Get buffer(byte) (FN587)

Function commands (FN codes)

Command name	SETINT
FN code	581
Title name	Set buffer (integer)
General description	This command is used to stored integer value at an arbitrary position in the buffer

■ General description

This command is used to stored integer value at an arbitrary position in the buffer. The integral value with the sign is stored by four bytes in the big endian in this command.

■ Parameter

Parameter No. 1	Buffer No.	The number of the buffer to store data is specified. (1 – 16)
Parameter No. 2	Integer value	The integer value stored in the buffer is specified. The integer value can be specified by the integer variable and the integer constant.
Parameter No. 3	Starting point	The stored position of the buffer is specified. (0 – 1020)

■ Example of screen display

```
SETINT[1,53,0]                         FN581; Set buffer (integer)
```

See
SETSTR ; Set buffer(string) (FN580)
SETREAL ; Set buffer(real) (FN582)
SETBYTE ; Set buffer(byte) (FN583)
GETSTR ; Get buffer(string) (FN584)
GETINT ; Get buffer(integer) (FN585)
GETREAL ; Get buffer(real) (FN586)
GETBYTE ; Get buffer(byte) (FN587)

Function commands (FN codes)

Command name	SETREAL
FN code	582
Title name	Set buffer (real)
General description	This command is used to stored real value at an arbitrary position in the buffer

■ General description

This command is used to stored real value at an arbitrary position in the buffer.
The real number value is expressed in the form of the single precision floating point number defined by IEEE754. The data is stored in the buffer by the big endian by four bytes.

■ Parameter

Parameter No. 1	Buffer No.	The number of the buffer to store data is specified. (1 – 16)
Parameter No. 2	Real value	The real value stored in the buffer is specified. The real value can be specified by the real variable and the real constant.
Parameter No. 3	Starting point	The stored position of the buffer is specified. (0 – 1020)

■ Example of screen display

```
SETREAL[1,53.2,0]                    FN582; Set buffer (real)
```

See
SETSTR ; Set buffer(string) (FN580)
SETINT ; Set buffer(integer) (FN581)
SETBYTE ; Set buffer(byte) (FN583)
GETSTR ; Get buffer(string) (FN584)
GETINT ; Get buffer(integer) (FN585)
GETREAL ; Get buffer(real) (FN586)
GETBYTE ; Get buffer(byte) (FN587)

Function commands (FN codes)

Command name	SETBYTE
FN code	583
Title name	Set buffer (byte)
General description	This command is used to stored byte data at an arbitrary position in the buffer.

■ General description

This command is used to stored byte data at an arbitrary position in the buffer. This command is userd, when the data that cannot correspond with SETSTR, SETINT, and SETREAL is stored in the buffer.

■ Parameter

Parameter No. 1	Buffer No.	The number of the buffer to store data is specified. (1 – 16)
Parameter No. 2	Integer value	The value stored in the buffer is specified. The value can be specified by the variable and the constant.
Parameter No. 3	Starting point	The stored position of the buffer is specified. (0 – 1020)

■ Example of screen display

```
SETBYTE[1,53,0]                      FN583; Set buffer (byte)
```

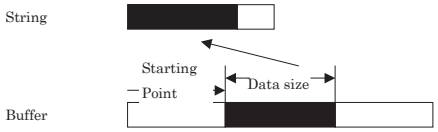
See
SETSTR ; Set buffer(string) (FN580)
SETINT ; Set buffer(integer) (FN581)
SETREAL ; Set buffer(real) (FN582)
GETSTR ; Get buffer(string) (FN584)
GETINT ; Get buffer(integer) (FN585)
GETREAL ; Get buffer(real) (FN586)
GETBYTE ; Get buffer(byte) (FN587)

Function commands (FN codes)

Command name	GETSTR
FN code	584
Title name	Get buffer (string)
General description	This command is used to read data from the buffer, and to store data in the string variable.

■ General description

This command is used to read data from specified position of the buffer, and to store data in the specified string variable. When the reading string overflows of the buffer (When starting position + data size exceeds the size of the buffer), it becomes it , saying that "E2250 step data is abnormal".



■ Parameter

Parameter No. 1	Buffer No.	The number of the buffer to read data is specified. (1 – 16)
Parameter No. 2	String variable	The string variable to store the read data is specified.
Parameter No. 3	Starting point	The read position of the buffer is specified. (0 – 1023)
Parameter No. 4	Data size	The data size of the data read in the buffer is specified. (1 – 199)

■ Example of screen display

```
GETSTR[1,V1$,0,3]                    FN584; Get buffer (string)
```

See
SETSTR ; Set buffer(string) (FN580)
SETINT ; Set buffer(integer) (FN581)
SETREAL ; Set buffer(real) (FN582)
SETBYTE ; Set buffer(byte) (FN583)
GETINT ; Get buffer(integer) (FN585)
GETREAL ; Get buffer(real) (FN586)
GETBYTE ; Get buffer(byte) (FN587)

Function commands (FN codes)

Command name	GETINT
FN code	585
Title name	Get buffer (integer)
General description	This command is used to read data from the buffer, and stored the integer variable.

■ General description

This command is used to read four bytes data from the specified position of the buffer, and stored the integer variable. When this command is used, it is necessary to confirm the big endian the data stored in the buffer, and the integer value with the sign in four bytes. It is necessary to use the GETBYTE function for other integral values.

■ Parameter

Parameter No. 1	Buffer No.	The number of the buffer to read data is specified. (1 – 16)
Parameter No. 2	Integer variable	The integer variable to store the read data from the buffer is specified.
Parameter No. 3	Starting point	The read position of the buffer is specified. (0 – 1020)

■ Example of screen display

```
GETINT[1,V1%,0]                      FN585; Get buffer (integer)
```

See
SETSTR ; Set buffer(string) (FN580)
SETINT ; Set buffer(integer) (FN581)
SETREAL ; Set buffer(real) (FN582)
SETBYTE ; Set buffer(byte) (FN583)
GETSTR ; Get buffer(string) (FN584)
GETREAL ; Get buffer(real) (FN586)
GETBYTE ; Get buffer(byte) (FN587)

Function commands (FN codes)

Command name	GETREAL
FN code	586
Title name	Get buffer (real)
General description	This command is used to read data from the buffer, and stored the real variable.

■ General description

This command is used to read four bytes data from the specified position of the buffer, and stored the real variable. When this command is used, it is necessary to confirm the big endian the data stored in the buffer, and the floating point number provided for by IEEE754 in four bytes.

■ Parameter

Parameter No. 1	Buffer No.	The number of the buffer to read data is specified. (1 – 16)
Parameter No. 2	Real variable	The real variable to store the read data from the buffer is specified.
Parameter No. 3	Starting point	The read position of the buffer is specified. (0 – 1020)

■ Example of screen display

```
GETREAL[1,V1!,0]                    FN586; Get buffer (real)
```

See
SETSTR ; Set buffer(string) (FN580)
SETINT ; Set buffer(integer) (FN581)
SETREAL ; Set buffer(real) (FN582)
SETBYTE ; Set buffer(byte) (FN583)
GETSTR ; Get buffer(string) (FN584)
GETINT ; Get buffer(integer) (FN585)
GETBYTE ; Get buffer(byte) (FN587)

Function commands (FN codes)

Command name	GETBYTE
FN code	587
Title name	Get buffer (byte)
General description	This command is used to read one byte data from the buffer, and stored the integer variable.

■ General description

This command is used to read one byte data from the buffer, and stored the integer variable.

■ Parameter

Parameter No. 1	Buffer No.	The number of the buffer to read data is specified. (1 – 16)
Parameter No. 2	Integer variable	The integer variable to store the read data from the buffer is specified.
Parameter No. 3	Starting point	The read position of the buffer is specified. (0 – 1023)

■ Example of screen display

```
GETBYTE[1,V2%,0]                    FN587; Get buffer (byte)
```

See
SETSTR ; Set buffer(string) (FN580)
SETINT ; Set buffer(integer) (FN581)
SETREAL ; Set buffer(real) (FN582)
SETBYTE ; Set buffer(byte) (FN583)
GETSTR ; Get buffer(string) (FN584)
GETINT ; Get buffer(integer) (FN585)
GETREAL ; Get buffer(real) (FN586)

Function commands (FN codes)

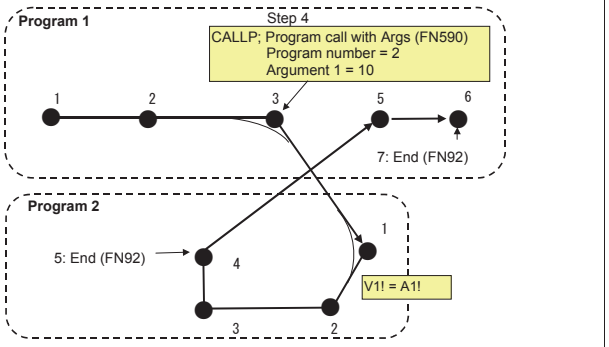
Command name	LCALLP
FN code	590
Title name	Program call with Arguments
General description	This command is used to call the specified program. At this time, ten real numbers can pass the arguments to the program.

■ General description

When this function command is executed, the specified program is called and passed the arguments.
Bear in mind that if a function command has been recorded in the first step in the call destination program, the function command at the jump destination will be executed as soon as the call command has been executed.
When the playback of the program at the call destination is completed (in the status established by executing the END command), the robot returns to the step following the step with the call command of the call source program.

■ Example of operation

In step 4, record LCALLP: program call with Arguments (FN590) and "2" as the program number, and "10" as an argument.
When this is played back, the robot calls the Program 2 upon arriving at step 4. The program2 can get the value "10" passed an argument1 in LCALLP command by using variable of argument "A1!".
When the playback of program 2 is completed (in the status established by executing the END command), the robot returns to step 5 following the step with the call command of call source program 1.



The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

The program call can be executed again at the call destination (during program 2 in the above figure.) Up to 8 layers of calls can be executed. If calls exceeding 8 layers are executed, the "A2138 Wrong call command setting" alarm is detected during playback, and the robot stops.

■ Parameter

Parameter No. 1	Program No.	This specifies the number of the program serving as the call destination. (1-9999)
Parameter No.2-No.11	Argument 1-10	This specifies the value of an argument passed to the program serving as the call destination. (-1E38 – 1E38)

■ Example of screen display

LCALLP [2,10,0,0,0,0,0,0,0,0] FN590; Program call with Arguments

See
LCALLPI: Conditional program call with Arguments (FN591)
LCALLPN: Conditional program call after specified number of passes with Arguments (FN592)

Function commands (FN codes)

Command name	LCALLPI
FN code	591
Title name	Conditional program call with Arguments
General description	Using an input signal, this command is used to call the specified program. At this time, ten real numbers can pass the arguments to the program.

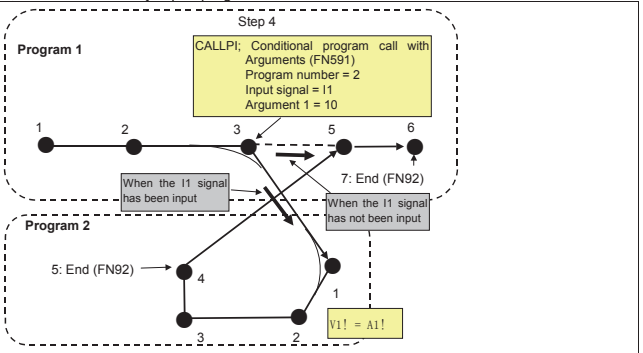
■ General description

When this function command is executed, the specified program is called and passed the arguments. When the specified input signal has been input, the step is called; when it has not been input, the step is not called and the robot passes the command by.
Bear in mind that if a function command has been recorded in the first step in the call destination program, the function command at the jump destination will be executed as soon as the call command has been executed.
When the playback of the program at the call destination is completed (in the status established by executing the END command), the robot returns to the step following the step with the call command of the call source program.

■ Example of operation

In step 4, record LCALLPI: conditional program call with Arguments (FN591), "2" as the program number, and I1 as the input signal, and "10" as an argument.

When this is played back, the robot arrives at step 4, and if input signal I1 has been input, it jumps to the first step in program 2. The program2 can get the value "10" passed an argument1 in LCALLP command by using variable of argument "A1!". When the playback of program 2 is completed (the END command is executed), the robot returns to step 5 following the step with the call command of call source program 1.
If the signal has not been input, the robot does not jump to program 2.



The robot is not decelerated by this command. If no positioning is specified, the input signal is inspected just before the command value reaches the accuracy range of the move command, and if it has been input, an inside arc is drawn. If the input signal has not been input, the command value heads toward the recorded point, and the input signal is inspected when the robot has reached the accuracy range.

The program call can be executed again at the call destination (during program 2 in the above figure.) Up to 8 layers of calls can be executed. If calls exceeding 8 layers are executed, the "A2138 Wrong call command setting" alarm is detected during playback, and the robot stops.

■ Parameter

Parameter No. 1	Program No.	This specifies the number of the program serving as the call destination. (1-9999)
Parameter No. 2	Input signal	This records the number of the input signal which is to serve as the condition for executing the call. When number 5101 or above is specified, multiple input signals can be specified. (1-2048、5101—5196)
Parameter No.3–No.12	Argument 1-10	This specifies the value of an argument passed to the program serving as the call destination. (-1E38 – 1E38)

■ Example of screen display

LCALLP [2,I1,10,0,0,0,0,0,0,0,0] FN81; Conditional program call with Args

See
LCALLP: Program call with Arguments (FN590)
LCALLPN: Conditional program call after specified number of passes with Arguments (FN592)

Function commands (FN codes)

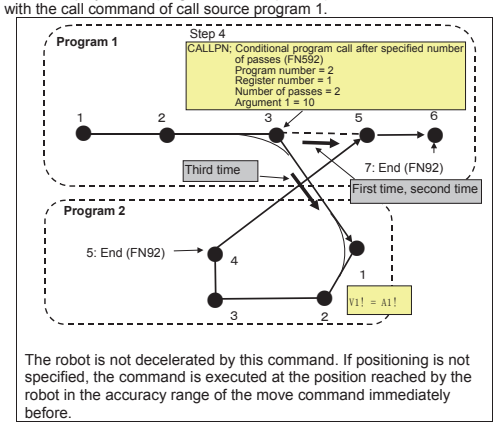
Command name	LCALLPN
FN code	592
Title name	Conditional program call after specified number of passes with Arguments.
General description	Using a pass count (number of passes), this command is used to call the specified program. At this time, ten real numbers can pass the arguments to the program.

■ General description

When this function command is executed, the specified program is called and passed the arguments. The robot passes for the specified number of passes, and on the next time (specified number of passes +1) the call command is executed. (For instance, if "2" is specified as the number of passes, the robot passes twice, and on the third time the call command is executed.)
Bear in mind that if a function command has been recorded in the first step in the call destination program, the function command at the jump destination will be executed as soon as the call command has been executed.
When the playback of the program at the call destination is completed (in the status established by executing the END command), the robot returns to the step following the step with the call command of the call source program.

■ Example of operation

In step 4, record LCALLPN: conditional program call after specified number of passes with Args (FN592), "2" as the program number, "1" as the register number, "2" as the number of passes, and "10" as the argument 1.
When this is played back, the robot passes by for the first and second times, and then advances to steps 5; however, on the third time, it jumps to the first step in program 2. The program2 can get the value "10" passed an argument1 in LCALLP command by using variable of argument "A1!".
When the playback of program 2 is completed (the END command is executed), the robot returns to step 5 following the step with the call command of call source program 1.



The robot is not decelerated by this command. If positioning is not specified, the command is executed at the position reached by the robot in the accuracy range of the move command immediately before.

The program call can be executed again at the call destination (during program 2 in the above figure.) Up to 8 layers of calls can be executed. If calls exceeding 8 layers are executed, the "A2138 Wrong call command setting" alarm is detected during playback, and the robot stops.

A global integer variable common to all units is used for the number of passes.
The current number of passes can be referenced using monitor/integer variables.

■ Parameter

Parameter No. 1	Program No.	This specifies the number of the program serving as the call destination. (1-9999)
Parameter No. 2	Register number	A "register" refers to the memory used for counting. Since an integer variable (1 to 200) is used, this parameter specifies its number. (1-200)
Parameter No. 3	Number of passes	This records the number of passes which is to serve as the condition for executing the call. The robot passes for the specified number of passes, and on the next time (specified number of passes +1) the call command is executed. (0-10000)
Parameter No.4-No.13	Argument 1-10	This specifies the value of an argument passed to the program serving as the call destination. (-1E38 – 1E38)

■ Example of screen display

LCALLPN[2,V1%,2,10,0,0,0,0,0,0,0,0,0] FN82; Conditional program call after specified number of passes with Arguments

See
LCALLP: Program call with Arguments (FN590)
LCALLPI: Conditional program call with Arguments (FN591)

Function commands (FN codes)

Command name	LCALLMCR
FN code	593
Title name	Call User Task Program with Arguments
General description	This command is used to call the specified user task program. At this time, ten real numbers can pass the arguments to the program.

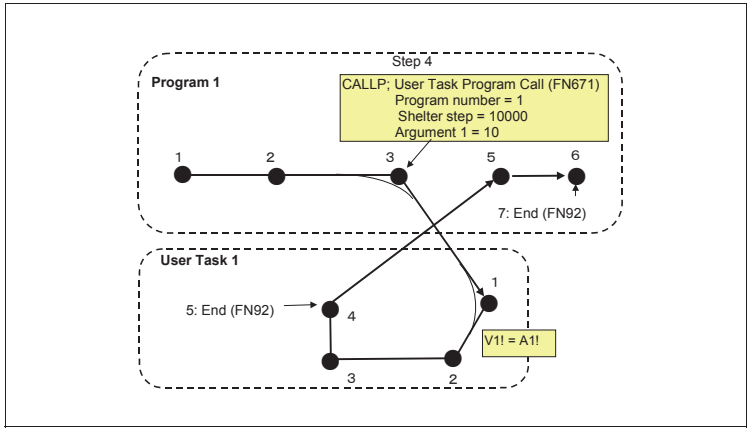
■ General description

When this function command is executed, the specified user task program is called and passed the arguments. When program call user task program, program stop one's playback and start specified user task program.
When the playback of the user task program at the call destination is completed (in the status established by executing the END command), the program returns to the step following the step with the call command of the call source program.

■ Example of operation

In step 4, record LCALLMCR: User Task Program Call with Arguments (FN593) and "1" as the program number, "10000" as the shelter step, and"10" as an argument.

When this is played back, the program1 call the user task program 1 upon arriving at step 4. The program2 can get the value "10" passed an argument1 in LCALLP command by using variable of argument "A1!". When the start of User Task 1 fails, the program1 jumps shelter step.
When the playback of user task program 1 is completed (in the status established by executing the END command), the robot returns to step 5 following the step with the call command of call source program 1.



■ Parameter

Parameter No. 1	Program No.	This specifies the number of the program serving as the call destination. (1-999)
Parameter No. 2	Shelter step	This is used to specify the number of the shelter step when the specified user task program was not starting. (1 to 10000) When 10000 is specified as the shelter step number, an alarm results immediately with no escape operation performed, and the robot can be stopped.
Parameter No.3-No.12	Argument 1-10	This specifies the value of an argument passed to the program serving as the call destination. (-1E38 – 1E38)

■ Example of screen display

LCALLMCR [1,10000,10,0,0,0,0,0,0,0,0,0] FN593; Call User Task Program with Arguments

Command name	NOP
FN code	600
Title name	Doing nothing
Outline	Doing nothing

■ **General description**

Doing nothing. This is used as a punctuation mark in program.

■ **Example of operation**

no parameters necessary

<SLIM sample> NOP

■ **Example of screen display**

```
NOP                 FN600;NOP
```


Command name	*
FN code	601
Title name	Label
Outline	Doing nothing

■ **General description**

Doing nothing.
This is used as a label that can be referred by GOTO command.

■ **Example of operation**

Beginning from *, and open software keyboard by [Enable]+[Edit] to input characters of label.

<SLIM sample> *LOADING

■ **Example of screen display**

```
*[LOADING]              FN601;Label
```

See
ONGOTO ;ONGOTO jump(FN603)

Function commands (FN codes)

Command name	IF
FN code	602
Title name	Condition (IF-THEN-ELSE)
Outline	If condition is satisfied then command after “THEN” is execute, else command after “ELSE” is executed

■ General description

This is flow control statement (IF-THEN-ELSE).
If condition is satisfied then command after “THEN” is execute, else command after “ELSE” is executed. If “ELSE” is not described and condition is not satisfied, then next step is executed.

■ Example of operation

Parameter No. 1	IF	This is condition.
Parameter No. 2	THEN	Command after this (described with the line number or label) is executed when condition is satisfied.
Parameter No. 3	ELSE	Command after this (described with the line number or label) is executed when condition is not satisfied. This is not to be described all the time.

● First Parameter Designation Method

The first parameter can be described freely by use of the soft keyboard.

At present, condition equations where integer variables or actual value variables are calculated in comparison can be described.

[Description example]

```
IF V1% = 1 THEN 10 ELSE 20
    ;  To step 10 when the integer variable "1" is "1", and
       to step 20 in other cases

IF V2! <> 0 THEN 30
    ;  To step 30 when the integer variable "2" is "other than 0", and
       to the next step in other cases

IF V3% > 1 THEN *FINISH ELSE *LOADING
    ;  To the label (*FINISH) when the integer variable "3" is "1 or
       higher", and
       to the label (*LOADING) in other cases
```

● Integer Variables

Numeric values not containing decimal point are handled.

Format	Vn%, V%[n] n=1 ~ 200 (Global integer variable) Ln%, L%[n] n=1 ~ 200 (Local integer variable) * Vn% is same in meaning as V%[n].
Range	-2147483648 ~ +2147483647
Sample	IF V1%=1 ; When the variable 1 is 1 IF V2%<>0 ; When the variable 2 is not 0
Storage	All the global integer variables are stored even if the main power source is turned off. All the local integer variables are not stored.

● Actual Value Variables

Numeric values containing decimal point are handled.

Format	Vn!, V![n] n=1 ~ 100 (Global integer variable) Ln!, L![n] n=1 ~ 100 (Local integer variable) * Vn! is same in meaning as V![n].
Range	-1.0E38 ~ +1.0E38
Sample	IF V1!=103.45 ; When the variable 1 is 103.45 IF V2!<>1.567E-2 ; When the variable 2 is not 1.567E-2
Storage	All the global integer variables are stored even if the main power source is turned off. All the local integer variables are not stored.

● Relation Calculation Equation

The relation calculation is used when to compare 2 numeric values. The result is obtained as true (1) or false (0), and is used for changing the program executional sequences in condition judgment texts and so forth.

Operator	Calculation contents	Example
=	Equal	V1%=V2%
<>	Not equal	V1%<>V2%
<	Smaller	V1%<V2%
>	Larger	V1%>V2%
<=	Smaller or equal	V1%<=V2%
>=	Larger or equal	V1%>=V2%

■ Example of screen display

```
IF V1%>1 THEN 100 ELSE 400    FN602;Condition
```

Function commands (FN codes)

Command name	ONGOTO
FN code	603
Title name	ON GOTO jump
Outline	Next command is determined by the value of condition

■ General description

This is flow control statement.
Next command is determined by the value of condition.
It's order is from 1 to 2,3,…

■ Example of operation

Parameter No. 1	ON	This is condition. One integer variable is designated. (1-200：global variables, 201-400:local variables)
Parameter No. 2	GOTO	Command after this (described with the line number or label) is executed depending on the value of condition. The order is; command executed when condition=1, command executed when condition=2, command executed when condition=3, . . . Press [Enable]+[Edit] to open software keyboard. Characters must includes comma that separates each command.

<SLIM sample>　ON V3% GOTO *CASE1,*CASE2,*CASE3

■ Example of screen display

```
ON V3% GOTO *CASE1,*CASE2…  FN603:ON GOTO jump
```

Function commands (FN codes)

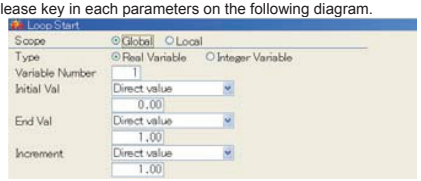
Command name	FOR
FN code	604
Title name	Loop Start
Outline	This is loop command. Loop starts here.

■ General description

Plural commands enclosed with FOR and NEXT are repeated.
This command is used with the set of NEXT command (no exception).

■ Example of operation

Please key in each parameters on the following diagram.



Parameter No. 1	Scope	This is to designate the variable to be used to determine the loop counter.
Parameter No. 2	Type	
Parameter No. 3	Variable number	
Parameter No. 4	Initial Val	This is loop starting counter.
Parameter No. 5	End Val	This is loop ending counter.
Parameter No. 6	Increments	This is the increments in every loop.

<SLIM sample>　FOR V3%=1 TO 100 STEP 2

■ Example of screen display

```
FOR V3%=1 TO 10 STEP 1     FN604;Loop Start
```

See
NEXT　;Loop End(FN605)

Command name	NEXT
FN code	605
Title name	loop End
Outline	This is loop command. Loop ends here.

■ **General description**

Plural commands enclosed with FOR and NEXT are repeated.
This command is used with the set of FOR command (no exception).

■ **Example of operation**

No parameter.
<SLIM sample> NEXT

■ **Example of screen display**

NEXT	FN605;Loop End

See
FOR ;Loop Start (FN604)

Command name	PRINT
FN code	606
Title name	Print string
Outline	The character string data is output to the screen or specified RS232C serial port

■ **General description**

When this function command is used, character string data can be displayed on the TP screen or output to the specified serial port. (RS232C is option)
Device number can designate to where output data. #0 is screen, #1 is RS232C. If device number is omitted, #0 is selected. If character is omitted, line feed code is only sent. Plural characters can be combined by ";" or ",". If last character was ";", line feed code is not sent.
FN101 is same as this function.

■ **Example of operation**

When the robot reaches the step in which the PRINT command is recorded, the character string is sent.
The robot continues moving while the data is being sent.

■ **Parameter**

Parameter No. 1	Port number	This is used to specify the number of the port from which the character string is to be output. 0; screen 1; RS232C
Parameter No. 2	Output character string	This is used to specify the character string to be output. Press [Enable]+[Edit] to open software keyboard to input characters. (Up to 199 single-byte alphanumeric)

<SLIM sample> PRINT #0,ABC

■ **Example of screen display**

PRINT[#0,TEST ABC]	FN606;Print string

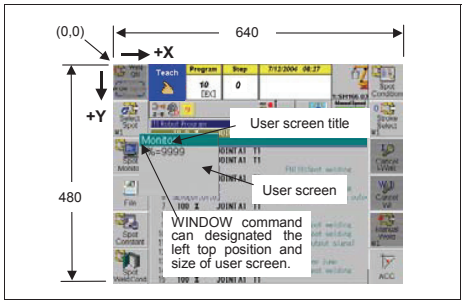
Function commands (FN codes)

Command name	WINDOW
FN code	607
Title name	User screen open/close
Outline	Open user screen, or close user screen.

■ General description

User task can draw the custom dialog on the screen, this is called “User screen”. User screen can be drawn on any place at the mode screen (where 12 F keys are displayed on left side and right side).
Size of TP screen is 640 * 480 dots, left top corner is (0,0). WINDOW command can designate the position of user screen. The position can be designated in every 16 dots. When one of width or height is defined 0, user screen can be closed.

This command is available only in user task, and is permitted being written in robot language only. Please refer to the **“User task” operating manual** for the detail of user task and user screen.



■ Parameters

Parameter No. 1	X position of left top corner	This is to define the X position of left top corner of user screen. (0 to 624)
Parameter No. 2	Y position of left top corner	This is to define the Y position of left top corner of user screen. (0 to 464)
Parameter No. 3	Width	This is to define the width of user screen. (0 to 640)
Parameter No. 4	Height	This is to define the height of user screen. (0 to 480)

<SLIM sample> WINDOW 64,160,80,64

See
TITLE ;User title (FN608)

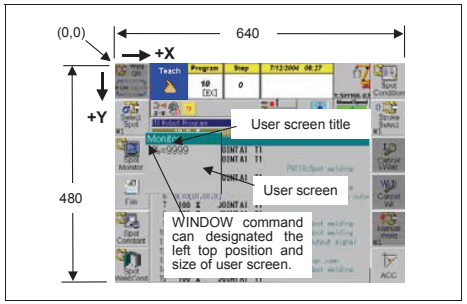
Function commands (FN codes)

Command name	TITLE
FN code	608
Title name	User screen title
Outline	This is to draw the title of user screen

■ General description

This is to draw the title of user screen.

This command is available only in user task, and is permitted being written in robot language only. Please refer to the **“User Task” operating manual** for the detail of user task and user screen.



■ Parameters

Parameter No. 1	Title	This is the character pattern data which show a title.

<SLIM sample> TITLE “MONITOR”

See
WINDOW ;User screen open/close(FN607)

Command name	CLS
FN code	609
Title name	Erase user screen
Outline	This is to erase user screen

■ **General description**

This is to erase user screen. (User screen is fully painted by background color.) Its color can be designated by BGCOLOR command.

This command is available only in user task, and is permitted being written in robot language only. Please refer to the **"User Task" operating manual** for the detail of user task and user screen.

■ **Parameters**

No parameters.

<SLIM sample> CLS

See

WINDOW ;User screen open/close (FN607)
TITLE ;User title (FN608)
BGCOLOR ;Background color (FN617)

Command name	LOCATE
FN code	610
Title name	Locate the display pos
Outline	This command is used to specify the position of the character displayed on the user screen.

■ **General description**

The position of the character displayed by the PRINT command in the user screen is specified by the row and the column.

This command is available only in user task, and is permitted being written in robot language only. Please refer to the **"User Task" operating manual** for the detail of user task and user screen.

■ **Parameters**

Parameter No. 1	Row number	The row to which the character string is displayed is specified. (0 to 22)
Parameter No. 2	Column number	The column at the character string head is specified. (0 to 80)

<SLIM sample> LOCATE 12,5

See

PRINT ; Print strings (FN606)

Command name	GLINE
FN code	611
Title name	Display position specification
Outline	This command is used for the user screen to draw the straight line.

■ **General description**

This command describes a straight-line in the user screen. A straight-line color is a value set by the COLOR command.

This command is available only in user task, and is permitted being written in robot language only. Please refer to the **"User Task" operating manual** for the detail of user task and user screen.

■ **Parameters**

Parameter No. 1	X position of start of line	X position of the point that the straight line begins is specified. (0 to 639)
Parameter No. 2	Y position of start of line	Y position of the point that the straight line begins is specified. (0 to 479)
Parameter No. 3	X position of end of line	X position of the point that the straight line ends is specified. (0 to 639)
Parameter No. 4	Y position of end of line	X position of the point that the straight line ends is specified. (0 to 479)

<SLIM sample> GLINE 10,10,100,100

See
GBOX ; Draw the box (FN612)
GSETP ; Draw the pixel (FN615)
COLOR ; Color (FN616)

Command name	GBOX
FN code	612
Title name	Draw the box
Outline	This command is used for the user screen to draw the box.

■ **General description**

This command describes a box in the user screen. A box color is a value set by the COLOR command.

This command is available only in user task, and is permitted being written in robot language only. Please refer to the **"User Task" operating manual** for the detail of user task and user screen.

■ **Parameters**

Parameter No. 1	X position of left top corner	This is to define the X position of left top corner of the box. (0 to 639)
Parameter No. 2	Y position of left top corner	This is to define the Y position of left top corner of the box. (0 to 479)
Parameter No. 3	Width	This is to define the width of the box. (0 to 640)
Parameter No. 4	Height	This is to define the height of the box. (0 to 480)
Parameter No. 5	Select of paint out	1: paint out 0: not paint out

<SLIM sample> GBOX 64,160,80,64,1

See
GLINE ; Draw the line FN611)
GSETP ; Draw the pixel (FN615)
COLOR ; Color (FN616)
BGCOLOR ; Background color (FN617)

Command name	BARC
FN code	613
Title name	Draw the arc
Outline	This command is used for the user screen to draw the arc.

■ **General description**

This command describes an arc in the user screen. An arc color is a value set by the COLOR command.

This command is available only in user task, and is permitted being written in robot language only. Please refer to the **"User Task" operating manual** for the detail of user task and user screen.

■ **Parameters**

Parameter No. 1	X position of left top corner	This is to define the X position of arc center. (0 to 639)
Parameter No. 2	Y position of left top corner	This is to define the Y position of arc center. (0 to 479)
Parameter No. 3	Arc radius	This is to define the arc radius. (0 to 800)
Parameter No. 4	Starting point of arc	This is to define the angle of starting point of the arc. (0 to 360)
Parameter No. 5	Center angle	This is to define the center angle. (0 to 360)

<SLIM sample> BARC 64,160,80,45,90

See
GLINE ; Draw the line (FN611)
GBOX ; Draw the box (FN612)
GSETP ; Draw the pixel (FN615)
COLOR ; Color (FN616)
BGCOLOR ; Background color (FN617)

Command name	GPAINT
FN code	614
Title name	Paint
Outline	This command is used to paint out the enclosed area on the user screen.

■ **General description**

This command is used to paint out the enclosed area on the user screen. An paint out color is a value set by the COLOR command.

This command is available only in user task, and is permitted being written in robot language only. Please refer to the **"User Task" operating manual** for the detail of user task and user screen.

■ **Parameters**

Parameter No. 1	X position of left top corner	This is to define the X position of paint out starting. (0 to 639)
Parameter No. 2	Y position of left top corner	This is to define the Y position of paint out starting. (0 to 479)
Parameter No. 3	Border color	This is to define the border color to paint out. (0 to 15)

<SLIM sample> GPAINT 64,160,8

See
GLINE ; Draw the line (FN611)
GBOX ; Draw the box (FN612)
GSETP ; Draw the pixel (FN615)
COLOR ; Color (FN616)
BGCOLOR ; Background color (FN617)

Function commands (FN codes)

Command name	GSETP
FN code	615
Title name	Draw the pixel
Outline	This command is used for the user screen to draw a pixel.

■ General description

This command describes a pixel in the user screen. A pixel color is a value set by the COLOR command.

This command is available only in user task, and is permitted being written in robot language only. Please refer to the **"User Task" operating manual** for the detail of user task and user screen.

■ Parameters

Parameter No. 1	X position of left top corner	This is to define the X position of pixel. (0 to 639)
Parameter No. 2	Y position of left top corner	This is to define the Y position of pixel. (0 to 479)

<SLIM sample> GSETP 64,160

See
GLINE ; Draw the line (FN611)
GBOX ; Draw the box (FN612)
COLOR ; Color (FN616)

Function commands (FN codes)

Command name	COLOR
FN code	616
Title name	Color
Outline	This can designate the color used in color graphics command

■ General description

This can designate the color used in color graphics command. Total 16 colors (0 to 15) are available.

This command is available only in user task, and is permitted being written in robot language only. Please refer to the **"User task" operating manual** for the detail of user task and user screen.

■ Parameters

Parameter No. 1	Color	0; Black
		1: Gray
		2: Dark Blue
		3: Blue
		4: Dark Green
		5: Green
		6: Dark Sky Blue
		7: Sky Blue
		8: Dark Red
		9: Red
		10:Purple
		11:Pink
		12:Dark Yellow
		13:Yellow
		14:Light Gray
		15:White

<SLIM sample> COLOR 15

See
GLINE ;Draw line (FN611)
GBOX ;Draw rectangle (FN612)
BGCOLOR ;Background color (FN616)

Command name	BGCOLOR
FN code	617
Title name	Background color
Outline	This can designate the background color used in color graphics command

■ **General description**

This can designate the background color used in color graphics command.(CLS, PRINT) Total 16 colors (0 to 15) are available.

This command is available only in user task, and is permitted being written in robot language only. Please refer to the **"User Task" operating manual** for the detail of user task and user screen.

■ **Parameters**

Parameter No. 1	Color	(refer to COLOR command)

<SLIM sample> BGCOLOR 15

See
COLOR ;Color(FN616)

Command name	EXIT
FN code	619
Title name	User task end
Outline	This can terminate the user task

■ **General description**

This can terminate the user task.
User task never ends (back to the top command after finishing the last command) unless otherwise stopped by manual operation or this EXIT command is executed.

This command is available only in user task, and is permitted being written in robot language only. Please refer to the **"User Task" operating manual** for the detail of user task and user screen.

■ **Parameters**

No parameters.
<SLIM sample> EXIT

See
OPENMCR ;Start user task(FN621)
WAITMCR ;Wait user task(FN622)

Command name	PAUSE
FN code	620
Title name	Pause user task
Outline	This can make a brief stop of user task

■ **General description**

This can pause (make a brief stop of) user task for designated time period.

This command is available only in user task, and is permitted being written in robot language only. Please refer to the **"User Task" operating manual** for the detail of user task and user screen.

This command (longer than 300msec pause) must be recorded in user task in order to avoid the CPU overload.
If not recorded, cycle time of robot program operated simultaneously with user task may varies.

■ **Parameters**

Parameter No. 1	Pause time	This is to define the pause time of user task. (0 to 999) [msec]

<SLIM sample> PAUSE 300

See
EXIT ;End user task (FN619)
OPENMCR ;Start user task (FN621)

Command name	GARC
FN code	623
Title name	Display ellipse
Outline	This command is used for the user screen to draw the ellipse.

■ **General description**

This command describes a ellipse in the user screen. A ellipse color is a value set by the COLOR command.

This command is available only in user task, and is permitted being written in robot language only. Please refer to the **"User Task" operating manual** for the detail of user task and user screen.

■ **Parameters**

Parameter No. 1	X position of left top corner	This is to define the X position of left top corner of the box to which oval is inscribed. (0 to 639)
Parameter No. 2	Y position of left top corner	This is to define the Y position of left top corner of the box to which oval is inscribed. (0 to 479)
Parameter No. 3	Width	This is to define the width of the box to which oval is inscribed. (0 to 640)
Parameter No. 4	Height	This is to define the height of the box to which oval is inscribed. (0 to 480)
Parameter No. 5	Select of paint out	1: paint out 0: not paint out

<SLIM sample> GARC 64,160,80,45,1

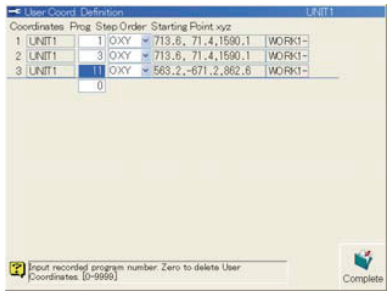
See
GLINE ; Draw the line (FN611)
GBOX ; Draw the box (FN612)
GSETP ; Draw the pixel (FN615)
COLOR ; Color (FN616)
BGCOLOR ; Background color (FN617)

Command name	MODUSRCOORD
FN code	626
Title name	Modify User coordinate
General description	Modifies the existent user coordinates using pose variables

■ General description

Executing this function command, the origin position and posture on the existent user coordinates can be modified using three pose variables.

The MODUSRCOORD command is so useful that the pose variables, that have been operated/ modified while executing the teaching programs, can be designated as a reference point on the user coordinates; however, the user coordinates that have been once modified can be never restored.
Therefore, if necessary to store the original user coordinates, be sure to duplicate the user coordinates file before executing MODUSRCOORD.



<Requirements for MODUSRCOORD>
・Existent user coordinates
…Required to register in [Service] → [10 User Coord. Definition] beforehand.
MODUSRCOORD can not create the new user coordinates.
・Pose variable
…Be sure to register three points in advance to define the user coordinates as the pose variables. For details of the definition of user coordinates, see [Service Menu] → [10 User coordinate] in this document.

■ Parameter

1st parameter	User coordinate No.	Designates the existent user coordinate No. (1~100)
2nd parameter	1st pose variable No.	Designates the 1st pose variable No. to define the user coordinates. (1~9999)
3rd parameter	2nd pose variable No.	Designates the 2nd pose variable No. to define the user coordinates. (1~9999)
4th parameter	3rd pose variable No.	Designates the 3rd pose variable No. to define the user coordinates. (1~9999)

＊ The pose variables used in the 2nd ~ 4th parameters depend on the USE command.

■ Example of screen display

```
MODUSRCOORD[1,1,2,3]          FN626: Modify User coordinate
```

The parameters shown on the screen are “User coordinate No.”, “1st pose variable No.”, “2nd pose variable No.”, and “3rd pose variable No.” from the left.

See
LETCOORDP; Let pose variable (Coordinates designation) (FN630)
LETPE; Let pose element (FN632)

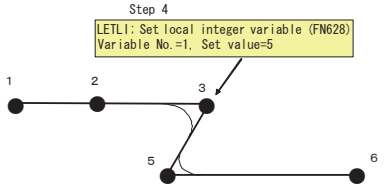
Command name	LETLI
FN code	628
Title name	Set local integer variable
General description	Sets the value into the specified local integer variable register

■ General description

This function command allows to set the value into the specified local integer variable. The local integer variables can be referred to by each unit.

■ Example of operation

The step 4 is recorded by “LETLI: Set local integer variable (FN628)”, “Variable No.=1”, and “Set value=5”. The 1st integer variable is set to 5 by playback. The set variables can be checked by the register screen of local integer variable on the monitor.



The robot will not slow down by this command. Without any positioning designation, this command is executed where the robot reaches the accuracy range of the move command right before.

■ Parameter

1st parameter	Local integer variable No.	Designates the integer variable No. to set the value. (1-200)
2nd parameter	Set value	Designates the value to set in the integer variables. (-2147483647~+2147483647)

■ Example of screen display

```
LETLI[L1%,5]          FN628;Set local integer variable
```

See
LETLF; Set local real variable (FN629)

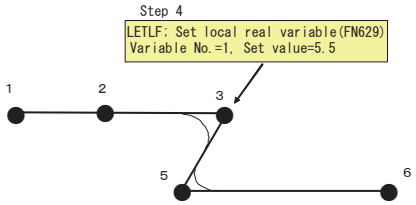
Command name	LETLF
FN code	629
Title name	Set local real variable
General description	Sets the value into the specified local real variable register

■ General description

This function command allows to set the value into the specified local real variable. The local real variables can be referred to by each unit.

■ Example of operation

The step 4 is recorded by “LETLF: Set local real variable (FN629)”, “Variable No.=1”, and “Set value=5.5”. The 1st real variable is set to 5.5 by playback. The set variables can be checked by the register screen of local real number variable on the monitor.



The robot will not slow down by this command. Without any positioning designation, this command is executed where the robot reaches the accuracy range of the move command right before.

■ Parameter

1st parameter	Local real variable No.	Designates the real variable No. to set the value. (1-200)
2nd parameter	Set value	Designates the value to set in the real variable. (-1.0E38~+1.0E38)

■ Example of screen display

See
LETLI; Set local real variable (FN628)

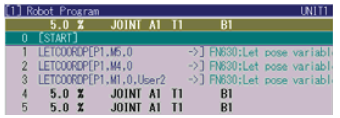
Command name	LETCOORDP
FN code	630
Title name	Let pose variable
General description	Stores the pose data recorded by the specified rectangular coordinates value in the pose variables

■ General description

This function command allows to record the pose variable set command of the coordinates designation.

- The LETCOORDP command is the pose set command by each mechanism. If there exist multiple mechanisms in the unit, execute the LETCOORDP command for the number of registered mechanisms.

  ∗ Also, in order not to forget initializing the pose variables (= Execution of LETCOORDP command for all the mechanisms), it is recommended to teach the LETCOORDP command for the number of registered mechanisms when starting to run the program.

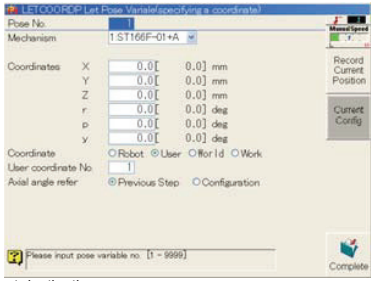


- Before executing LETCOORDP under the structure of synchronized unit, be sure to execute this command first in the operation standard mechanism.

- The robot does not move by executing LETCOORDP only. To move the robot, it is required to teach the MOVEX command.

■ LETCOORDP Teach Method

(1) Select FN630 “Let pose variable”.



(2) Enter the pose No. of the set destination.

(3) Select the mechanism No. of the pose set destination.

Be sure to teach the operation standard mechanism first. In the synchronized system of the robot and positioner, the positioner is the operation standard mechanism.

(4) Set the required parameters (described in the following).

(5) Move the robot by manual operation to the position for carrying out the pose record, and then press F8 “Record Current Position” key. Or, specify the value directly for the position to carry out the pose record.

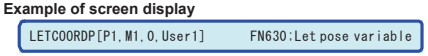
(6) Press f12 <Complete> to record LETCOORDP.

■ Parameter

Parameter name	Description	Range
Pose No.	Enters the pose variable No. of the set destination.	1~9999
Mechanism	Selects the mechanism No. of the pose set destination. → Be sure to teach the operation standard mechanism first.	1~9
Other Poses	Selects the method of how to deal with the pose data other than the record mechanism. If selecting “Initialize”, a function variable in the MOVE command step right before is set as the pose datum.	Do nothing/ Initialize in the previous step
Rectangular coordinates	Enters the rectangular coordinates value of the specified coordinates base. The current position can be also obtained by F8 “Record Current Position” key.	XYZ element: -99999.9~99999.9[mm] RPY (Rotation) element: -360.0~360.0[deg]
Coordinate	Designates the coordinates of the rectangular coordinates value. The external axis is fixed to the axis coordinates in being recorded.	Manipulator: Base/ User/ Absolute/ Workpiece

		External axis: Each axis
User coordinate No.	Enters the user coordinate No. when “User” has been selected in [Coordinate].	1~100
Axis angle refer	Designates the selection method of redundant solution in the reverse conversion, With "Configuration” selected, the radio button for redundant solution selection become ON in the J1, J3, J5, and J6 axis.	Previous step/ Configuration
J1 axis angle selection	Selects the redundant solution of the J1 axis. ・ Left arm: +45 ~ 135º ・ Right arm: -45 ~ -135º ・ Less than ±90º: -45 ~ +45º ・ ±90º or more: -135º or below or +135º or more	Left arm/ Right arm/ Less than ±90º/±90 or more
J3 axis angle selection	Selects the redundant solution of the J3 axis. ・ Upper elbow: 79.1º or below ・ Lower elbow: 79.1º or more	Upper elbow/ Lower elbow
J5 axis angle selection	Selects the redundant solution of the J5 axis. ・ Flip: 0º or below ・ Non-flip: 0º or more	Flip/ Non-flip
J6 axis angle selection	Selects the redundant solution of the J6 axis. ・ Less than ±180º: -180 ~ +180º ・ 0 ~ 360º : 0 ~ 360º ・ 0 ~ -360: 0 ~ -360º	Less than ±180º 0~360º/ 0~ -360º

■ **Example of screen display**



The parameters appeared on the screen are “Pose variable No.”, “Mechanism No.”, “Other Poses(0/1)”, and “Coordinates (+ User coordinate No.)” from the left.

See
LETPE; Let pose element (FN632)
LETPE; Let shift element (FN633)
LET; Let variable (FN634)
ADDP; Add pose variable (FN635)

Function Commands (FN Codes)

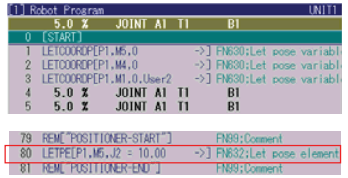
Command name	LETPE
FN code	632
Title name	Let pose element
General description	Stores the pose element recorded by the specified rectangular coordinates value in the pose variables

■ **General description**

This function command allows to record the let pose element command of the coordinates designation by each single element.

・ The LETPE command allows to specify a pose storage position by the parameter and set the pose only in that element.

∗For smooth teaching operation, it is recommended to use FN630 (the LETCOORDP command) on the pose storage of all the elements, and the LETPE command on a single element or when modifying the pose data of only 1 axis.
(The following is the example when setting 10º only into the 2nd axis of the mechanism 5 using LETPE.)

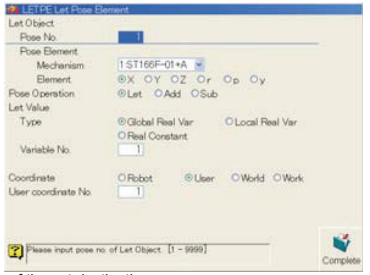


・ If there is no original pose variable, it is not available to use LETPE. In that case, execute FN630 (LETCOORDP) and create the pose data.

・ The robot does not move by executing LETPE only. To move the robot, it is required to teach MOVEX command.

■ **LETPE Teach Method**

(1) Select FN632 “Let pose element”.



(2) Enter the pose variable No. of the set destination.
(3) Select the mechanism No. of the pose set destination.
(4) Select the element of the pose set destination.
(5) Set the required parameters (described in the following).
(6) Press f12 <Complete> to record LETPE.

■ Parameter

Parameter name	Description	Range
Pose No.	Enters the pose variable No. of the set destination.	1~9999
Mechanism	Selects the mechanism No. of the pose set destination.	1~9
Element	Selects the element of the pose set destination. Selects the elements of rectangular coordinates when the mechanism is a manipulator, and the axis angle element when the mechanism is an external axis.	Manipulator: X/Y/Z/r/p/y External axis: J1/J2/J3/J4/J5/J6
Pose operation	Selects the type of pose operation. Addition and subtraction are carried out to the original pose data.	Set/ Add/ Subtract
Type	Selects a type of the pose set value. It becomes available to input [Variable No.] with a variable selected, and [Constant value] with a constant selected.	Global real variable/ Local real variable/ Real constant
Coordinate	Designates the coordinates of the rectangular coordinates value. The external axis is fixed to the axis coordinates in being recorded.	Manipulator: Robot/ User/ World/ Work External axis: Each axis
User coordinate No.	Enters the user coordinate No. when “User” has been selected in [Coordinate].	1~100

■ Example of screen display

```
LETPE[P1,M1,X = V1!,User1]     FN632:Let pose element
```

The parameters appeared on the screen are “Pose variable No.”, “Mechanism No.”, “Pose element=Pose set value”, and “Coordinates (+ User coordinate No.)” from the left.

See
LETCOORDP; Let pose variable (Coordinates designation) (FN630)
LETPE; Let shift element (FN633)
LET; Let variable (FN634)
ADDP; Add pose variable (FN635)

Function Commands (FN Codes)

Command name	LETRE
FN code	633
Title name	Let shift element
General description	Sets the shift element in the specified shift register

■ General description

This function command allows to set the shift amount data in the specified shift register by a single element.
Even if the specified shift register has been occupied by “SHIFTR; Shift2 (FN52)” or others, the value specified by LETRE can be newly set in the shift register without any influence exerted on the robot motion. The modified shift value becomes effective by the next playback of the SHIFTR; Shift2 (FN52).

- The LETRE command allows to specify the shift element storage position by the parameter and set the shift amount data only in that element.

  ∗For smooth teaching operation, it is recommended to use FN68 (LETR command) on the shift amount set of all the elements, and LETRE command on modifying the shift amount data.
  (The following is an example when setting 10mm only into the X element using LETRE.)

```
0  [START]
1  LETR[R1,100,100,100,10,11,12]  FN68:Set shift value
2   50.0 %    JOINT A1 T1     B1
3   50.0 %    JOINT A1 T1     B1
4  LETRE[R1,X = 10.00          ->] FN633:Let shift elemen
5  SHIFTR[1,0,R1,0]               FN52:Shift
6   50.0 mm/s LIN   A1 T1     B1
7  SHIFTR[0,0,R1,0]               FN52:Shift
8  END                            FN02:End
```

■ Example of operation

Shift register value when executing LETRE [R1, X = 10.00].
<1> Executing with the specified shift register value

	Before executing LETRE	After executing LETRE
Request flag	0	0
Set flag	1	1
X	110	10
Y	120	100
Z	130	100
θX	5	10
θY	6	11
θZ	7	12

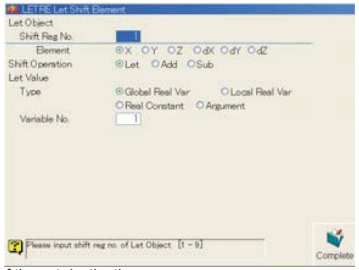
<2> Executing with the initial shift register

	Before executing LETR	After executing LETR
Request flag	0	0
Set flag	0	0
X	0	10
Y	0	0
Z	0	0
θX	0	0
θY	0	0
θZ	0	0

By executing this command, the request flag of shift register always becomes “0” and the set flag “1”. The values specified by the parameter are stored in X ~ θz.

■ LETRE Teach Method

(1) Select FN633 “Let shift element”.



(2) Enter the shift register No. of the set destination.
(3) Select the shift element of the set destination.
(4) Set the required parameters (described in the following).
(5) Press f12 <Complete> to record LETRE.

■ Parameter

Parameter name	Description	Range
Shift register No.	Enters the shift register No. of the set destination.	1~9
Element	Selects the shift element of the set destination.	X/Y/Z/θX/θY/θZ
Shift operation	Selects the type of shift operation. Addition and subtraction are carried out to the original shift amount data.	Set/ Add/ Subtract
Type	Selects a type of the shift amount substitution value. It becomes available to input [Variable No.] with a variable selected, and [Constant value] with a constant selected.	Global real variable/ Local real variable/ Real constant

■ Example of screen display

```
LETRE[R1,X = V1!]                FN633;Let shift element
```

The parameters appeared on the screen are “Shift register No.” and “Shift element=Shift amount set value”.

See
LETR; Let shift amount (FN68)
LETCOORDP; Let pose variable (Coordinates designation) (FN630)
LETPE; Let pose element (FN632)
LET; Let variable (FN634)
ADDP; Add pose variable (FN635)

Function Commands (FN Codes)

Command name	LET
FN code	634
Title name	Let variable
General description	Sets the variable of the same type

■ General description

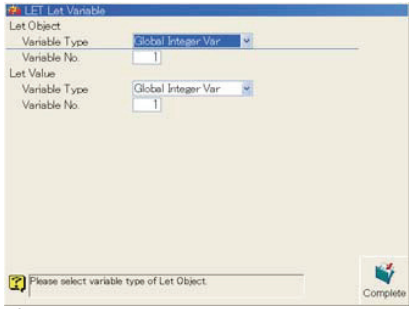
This function command allows to set the variable of the same type.
This is executed when using this command as a storage of the operation result while executing the program or copying the contents of variables.

- There are six types of variables available in the LET command; the global/local integer variable, global/local real variable, pose variable, and shift variable (shift register).

∗Values can be set each other only within the same type for the pose variable and shift variable, while it is available across the scope and type for the integer variable and real variable.

■ LET Teach Method

(1) Select FN634 “Let variable”.



(2) Select the type of variable of the set destination.
(3) Enter the variable No. of the set destination.
(4) Select the type of variable to set.
(5) Enter the variable No. of the set value.

■ Parameter

Parameter name	Description	Range
Variable type of set destination	Selects the variable type of the set destination.	Global integer variable/ Global real variable/ Local integer variable/ Local real variable/ Pose variable/ Shift variable
Variable No. of set destination	Enters the variable No. of the set destination.	Pose variable: 1~9999 Shift variable: 1~9 Others: 1~200
Variable type/ Variable No. of set value	Same as the set destination. However, if different kinds of variable type are selected for setting the pose variable/shift variable, the variable type of the set destination is applied.	-

■ Example of screen display

```
LET[V1%,V2%]                FN634:Let variable
```

The parameters appeared on the screen are “Set destination variable” and “Set value variable” from the left.

See
LETCOORDP; Let pose variable (Coordinates designation) (FN630)
LETPE; Let pose element (FN632)
LETPE; Let shift element (FN633)
ADDP; Add pose variable (FN635)

Command name	ADDP
FN code	635
Title name	Add pose variable
General description	Adds the value of pose variable

■ General description

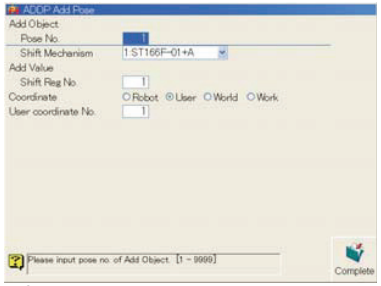
This function command allows to add the shift amount of shift register to the pose variable based on the specified coordinates.

- If there is no original pose variable, it is not available to use ADDP. In this case, execute FN630 (LETCOORDP command) and create the pose data.

The robot does not move by executing ADDP only. To move the robot, it is required to teach MOVEX command.

■ ADDP Teach Method

(1) Select FN635 “Add pose variable”.



(2) Enter the pose variable No. of the add destination.
(3) Select the shift mechanism No.
(4) Set the required parameters (described in the following).
(5) Press f12 <Complete> to record ADDP.

■ Parameter

Parameter name	Description	Range
Pose No.	Enters the pose variable No. of the add destination.	1~9999
Shift mechanism	Selects the manipulator of the shift target.	1~9
Shift register No.	Enters the shift register No. (added value).	1~9
Coordinate	Designates the coordinates of the rectangular coordinates value.	Robot/ User/ World/ Work
User coordinate No.	Enters the user coordinate No. when “User” has been selected in [Coordinate].	1~100

■ Example of screen display

```
ADDP[P1,M1,R1,User1]        FN635:Add pose variable
```

The parameters appeared on the screen are “Pose variable No.”, “Mechanism No.”, “Shift register No.”, and “Coordinates (+ User coordinate No.)” from the left.

See
LETCOORDP; Let pose variable (Coordinates designation) (FN630)
LETPE; Let pose element (FN632)
LETPE; Let shift element (FN633)
LET; Let variable (FN634)

Command name	ADDVI
FN code	637
Title name	Add integer variable
General description	Adds the value of integer variable

■ General description

This function command allows to add the integer variables.

■ Example of operation

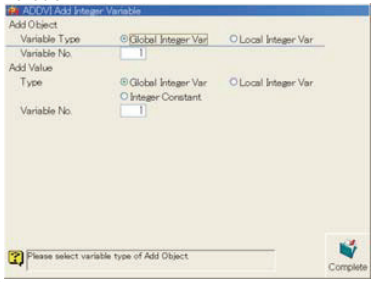
Any steps can be recorded by “ADDVI: Add integer variable (FN637)”, “Global variable No. of the add destination=1”, and “Global variable No. of the add value=2”. The 2nd variable is added to the 1st variable by playback of this command. The set variables can be checked by the register screen of global integer variable on the monitor. (Local integer variables are also available.)

Execution sample of ADDVI [V1%, V2%]

Global variable	Before executing ADDVI	After executing ADDVI
V1%	1	3
V2%	2	2

■ ADDVI Teach Method

(1) Select FN637 “Add integer variable”.



(2) Select the variable type of the add destination.
(3) Enter the variable No. of the add destination.
(4) Select the type of add value.
(5) Enter the variable No. of add value or add constant.
(6) Press F12 <Complete> to record ADDVI.

■ Parameter

Parameter name	Description	Range
Variable type of add destination	Enters the variable type of add destination.	Global integer variable/ Local integer variable
Variable No. of the add destination	Enters the variable No. of the add destination.	1~200
Type of add value	Enters the type of add value. ※ Selecting “Integer constant”, the value to add can be directly designated.	Global integer variable/ Local integer variable/ Integer constant
Variable No. of add value or Add constant	Enters the variable No. with “Variable” selected, and the constant value with “Integer constant” selected in the variable type of add value.	Variable No.: 1~200 Constant value: -2147483647~ 2147483647

■ Example of screen display

```
ADDVI[V1%,V2%]        FN637:Add integer variable
```

The parameters appeared on the screen are “Variable of the add destination”, “Add variable (or add constant)” from the left.

See
SUBVI; Subtract integer variable (FN639)
MULVI; Multiply integer variable (FN641)
DIVVI; Divide integer variable (FN643)

Command name	ADDVF
FN code	638
Title name	Add real variable
General description	Adds the value of real variable

■ General description

This function command allows to add the real variables.

■ Example of operation

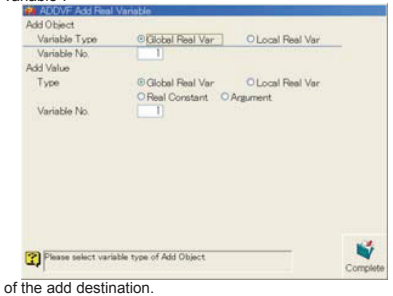
Any steps can be recorded by “ADDVF: Add real variable (FN638)”, “Global variable No. of the add destination=1”, and “Global variable No. of the add value=2”. The 2nd variable is added to the 1st variable by playback of this command. The set variables can be checked by the register screen of global real number variable on the monitor. (Local variables are also available.)

Execution sample of ADDVF [V1!, V2!]

Global variable	Before executing ADDVF	After executing ADDVF
V1!	1.1000	3.3000
V2!	2.2000	2.2000

■ ADDVF Teach Method

(1) Select FN638 “Add real variable”.



(2) Select the variable type of the add destination.
(3) Enter the variable No. of the add destination.
(4) Select the type of add value.
(5) Enter the variable No. of the add value or add constant.
(6) Press F12 <Complete> to record ADDVF.

■ Parameter

Parameter name	Description	Range
Variable type of add destination	Enters the variable type of the add destination.	Global real variable/ Local real variable
Variable No. of add destination	Enters the variable No. of the add destination.	1~200
Type of add value	Enters the type of add value. ※ Selecting “real constant”, the value to add can be directly designated.	Global real variable/ Local real variable/ Real constant
Variable No. of add value or Add constant	Enters the variable No. with “Variable” selected, and the constant value with “real constant” selected in the variable type of add value.	Variable No.: 1~200 Constant value: -1e+038~ 1e+038

■ Example of screen display

```
ADDVF[V1!,V2!]        FN638:Add real variable
```

The parameters appeared on the screen are “Variable of add destination”, “Add variable (or add constant)” from the left.

See
SUBVF; Subtract real variable (FN640)
MULVF; Multiply real variable (FN642)
DIVVF; Divide real variable (FN644)

Command name	SUBVI
FN code	639
Title name	Subtract integer variable
General description	Subtracts the value of integer variable

■ General description

This function command allows to subtract the integer variables.

■ Example of operation

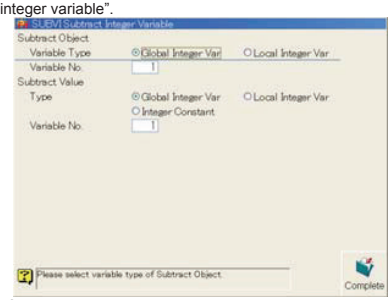
Any steps can be recorded by “SUBVI: Subtract integer variable (FN639)”, “Global variable No. of the subtract destination=1”, and “Global variable No. of the subtract value=2”. The 2nd variable is subtracted from the 1st variable by playback of this command. The set variables can be checked by the register screen of global integer variable on the monitor. (Local integer variables are also available.)

Execution sample of SUBVI [V1%, V2%]

Global variable	Before executing SUBVI	After executing SUBVI
V1%	2	1
V2%	1	1

■ SUBVI Teach Method

(1) Select FN639 “Subtract integer variable”.



(2) Select the variable type of the subtract destination.
(3) Enter the variable No. of the subtract destination.
(4) Select the type of subtract value.
(5) Enter the variable No. of the subtract value or subtract constant.
(6) Press F12 <Complete> to record SUBVI.

■ Parameter

Parameter name	Description	Range
Variable type of subtract destination	Enters the variable type of the subtract destination.	Global integer variable/ Local integer variable
Variable No. of subtract destination	Enters the variable No. of the subtract destination.	1~200
Type of subtract value	Enters the type of subtract value. ※ Selecting “Integer constant”, the value to subtract can be directly designated.	Global integer variable/ Local integer variable/ Integer constant
Variable No. of subtract value or Subtract constant	Enters the variable No. with “Variable” selected, and the constant value with “Integer constant” selected in the variable type of subtract value.	Variable No.: 1~200 Constant value: -2147483647 ~2147483647

■ Example of screen display

```
SUBVI[V1%,V2%]        FN639:Subtract integer variable
```

The parameters appeared on the screen are “Variable of the subtract destination”, “Subtract variable (or subtract constant)” from the left.

See
ADDVI; Add integer variable (FN637)
MULVI; Multiply integer variable (FN641)
DIVVI; Divide integer variable (FN643)

Command name	SUBVF
FN code	640
Title name	Subtract real variable
General description	Subtracts the real variable

■ General description

This function command allows to subtract the real variable.

■ Example of operation

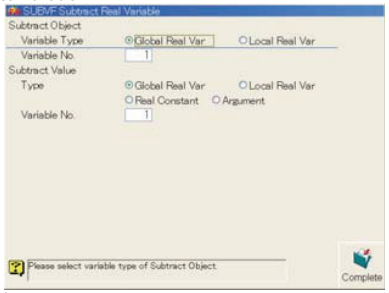
Any steps can be saved by “SUBVF: Subtract real variable (FN640)”, “Global variable No. of the subtract destination=1”, and “Global variable No. of the subtract value=2”. By playback of this command, the 2nd variable is subtracted from the 1st variable. The set variables can be checked by the register screen of global real number variable on the monitor. (Local variables are also available.)

Execution sample of SUBVF [V1!, V2!]

Global variable	Before executing SUBVF	After executing SUBVF
V1!	2.2000	1.1000
V2!	1.1000	1.1000

■ SUBVF Teach Method

(1) Select FN640 “Subtract real variable”.



(2) Select the variable type of the subtract destination.
(3) Enters the variable No. of the subtract destination.
(4) Select the type of subtract value.
(5) Enter the variable No. of the subtract value or subtract constant.
(6) Press F12 <Complete> to record SUBVF.

■ Parameter

Parameter name	Description	Range
Variable type of subtract destination	Enters the variable type of the subtract destination.	Global real variable/ Local real variable
Variable No. of subtract destination	Enters the variable No. of the subtract destination.	1~200
Type of subtract value	Enters the type of subtract value. ※ Selecting “Real constant”, the value to subtract can be directly designated.	Global real variable/ Local real variable/ Real constant
Variable No. of subtract value or Subtract constant	Enters the variable No. with “Variable” selected, and the constant value with “Real constant” selected in the variable type of subtract value.	Variable No.: 1~200 Constant value: -1e+038~ 1e+038

■ Example of screen display

```
SUBVF[V1!,V2!]          FN640:Subtract real variable
```

The parameters appeared on the screen are “Variable of subtract destination”, “Subtract variable (or subtract constant)” from the left.

See
ADDVF; Add real variable (FN638)
MULVF; Multiply real variable (FN642)
DIVVF; Divide real variable (FN644)

Command name	MULVI
FN code	641
Title name	Multiply integer variable
General description	Multiplies the integer variables

■ General description

This function command allows to multiply the integer variables.

■ Example of operation

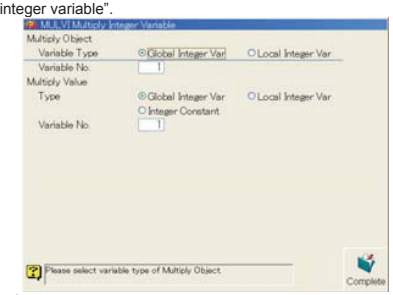
Any steps can be recorded by “MULVI: Multiply integer variable (FN641)”, “Global variable No. of the multiply destination=1”, and “Global variable No. of the multiply value=2”. The 1st and the 2nd variable are multiplied by playback of this command. The set variables can be checked by the register screen of global integer variable on the monitor. (Local integer variables are also available.)

Execution sample of MULVI [V1%, V2%]

Global variable	Before executing MULVI	After executing MULVI
V1%	2	6
V2%	3	3

■ MULVI Teach Method

(1) Select FN641 “Multiply integer variable”.



(2) Select the variable type of the multiply destination.
(3) Enter the variable No. of the multiply destination.
(4) Select the type of multiply value.
(5) Enter the variable No. of the multiply value or multiply constant.
(6) Press F12 <Complete> to record MULVI.

■ Parameter

Parameter name	Description	Range
Variable type of multiply destination	Enters the variable type of the multiply destination.	Global integer variable/ Local integer variable
Variable No. of multiply destination	Enters the variable No. of the multiply destination.	1~200
Type of multiply value	Enters the type of multiply value. ※ Selecting “Integer constant”, the value to multiply can be directly designated.	Global integer variable/ Local integer variable/ Integer constant
Variable No. of multiply value or Multiply constant	Enters the variable No. with “Variable” selected, and the constant value with “Integer constant” selected in the variable type of multiply value.	Variable No.: 1~200 Constant value: -2147483647~ 2147483647

■ Example of screen display

```
MULVI[V1%,V2%]          FN641:Multiply integer variable
```

The parameters appeared on the screen are “Variable of multiply destination”, “Multiply variable (or multiply constant)” from the left.

See
ADDVI; Add integer variable (FN637)
SUBVI; Subtract integer variable (FN639)
DIVVI; Divide integer variable (FN643)

Command name	MULVF
FN code	642
Title name	Multiply real variable
General description	Multiplies the real variables

■ General description

This function command allows to multiply the real variables.

■ Example of operation

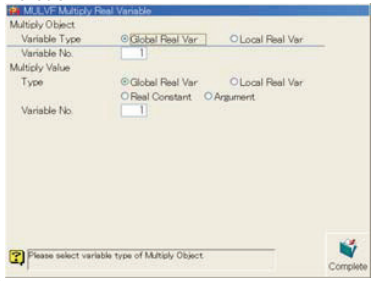
Any steps can be recorded by “MULVF: Multiply real variable (FN642)”, “Global variable No. of the multiply destination=1”, and “Global variable No. of the multiply value=2”. The 1st and the 2nd variables are multiplied by playback of this command. The set variables can be checked by the register screen of global real number variable on the monitor. (Local variables are also available.)

Execution sample of MULVF [V1!, V2!]

Global variable	Before executing MULVF	After executing MULVF
V1!	2.2000	6.6000
V2!	3.0000	3.0000

■ MULVF Teach Method

(1) Select FN642 “Multiply real variable”.



(2) Select the variable type of the multiply destination.
(3) Enter the variable No. of the multiply destination.
(4) Select the type of multiply value.
(5) Enter the variable No. of the multiply value or multiply constant.
(6) Press F12 <Complete> to record MULVF.

■ Parameter

Parameter name	Description	Range
Variable type of multiply destination	Enters the variable type of the multiply destination.	Global real variable/ Local real variable
Variable No. of multiply destination	Enters the variable No. of the multiply destination.	1~200
Type of multiply value	Enters the type of multiply value. ※ Selecting “Real constant”, the value to multiply can be directly designated.	Global real variable/ Local real variable/ Real constant
Variable No. of multiply value or Multiply constant	Enters the variable No. with “Variable” selected, and the constant value with “Real constant” selected in the variable type of multiply value.	Variable No.: 1~200 Constant value: -1e+038~ 1e+038

■ Example of screen display

```
MULVF[V1!,V2!]          FN642:Multiply real variable
```

The parameters appeared on the screen are “Variable of multiply destination”, “Multiply variable (or multiply constant)” from the left.

See
ADDVF; Add real variable (FN638)
SUBVF; Subtract real variable (FN640)
DIVVF; Divide real variable (FN644)

Command name	DIVVI
FN code	643
Title name	Divide integer variable
General description	Divides the value of integer variable

■ General description

This function command allows to divide the integer variables.

■ Example of operation

Any steps can be recorded by “DIVVI: Divide integer variable (FN643)”, “Global variable No. of the divide destination=1”, and “Global variable No. of the divide value=2”. The 1st variable is divided by the 2nd variable by playback of this command. The set variables can be checked by the register screen of global integer variable on the monitor. (Local integer variables are also available.)

Execution sample of DIVVI [V1%, V2%]

Global variable	Before executing DIVVI	After executing DIVVI
V1%	4	2
V2%	2	2

■ DIVVI Teach Method

(1) Select FN643 “Divide integer variable”.

(2) Select the variable type of the divide destination.
(3) Enters the variable No. of the divide destination.
(4) Select the type of divide value.
(5) Enter the variable No. of the divide value or divide constant.
(6) Press F12 <Complete> to record DIVVI.

■ Parameter

Parameter name	Description	Range
Variable type of divide destination	Enters the variable type of the divide destination.	Global integer variable/ Local integer variable
Variable No. of divide destination	Enters the variable No. of the divide destination.	1~200
Type of divide value	Enters the type of divide value. ※ Selecting “Integer constant”, the value to divide by can be directly designated.	Global integer variable/ Local integer variable/ Integer constant
Variable No. of divide value or Divide constant	Enters the variable No. with “Variable” selected, and the constant value with “Integer constant” selected in the variable type of divide value.	Variable No.: 1~200 Constant value: -2147483647~2147483647

■ Example of screen display

```
DIVVI[V1%,V2%]          FN643;Divide integer variable
```

The parameters appeared on the screen are “Variable of divide destination”, “Variable of divide value (or divide constant)” from the left.

See
ADDVI; Add integer variable (FN637)
SUBVI; Subtract integer variable (FN639)
MULVI; Multiply integer variable (FN641)

Function Commands (FN Codes)

Command name	DIVVF
FN code	644
Title name	Divide real variable
General description	Divides the value of real variable

■ General description

This function command allows to divide the real variables.

■ Example of operation

Any steps can be recorded by “DIVVF: Divide real variable (FN644)”, “Global variable No. of the divide destination=1”, and “Global variable No. of the divide value=2”. The 1st variable is divided by the 2nd variable by playback of this command. The set variables can be checked by the register screen of global real variable on the monitor. (Local variables are also available.)

Execution sample of DIVVF [V1!, V2!]

Global variable	Before executing DIVVF	After executing DIVVF
V1!	4.4000	2.0000
V2!	2.2000	2.2000

■ DIVVF Teach Method

(1) Select FN644 “Divide real variable”.

(2) Select the variable type of the divide destination.
(3) Enters the variable No. of the divide destination.
(4) Select the type of divide value.
(5) Enter the variable No. of the divide value or divide constant.
(6) Press F12 <Complete> to record DIVVF.

■ Parameter

Parameter name	Description	Range
Variable type of divide destination	Enters the variable type of the divide destination.	Global real variable/ Local real variable
Variable No. of divide destination	Enters the variable No. of the divide destination.	1~200
Type of divide value	Enters the type of divide value. ※ Selecting “Real constant”, the value to divide by can be directly designated.	Global real variable/ Local real variable/ Real constant
Variable No. of divide value or Divide constant	Enters the variable No. with “Variable” selected, and the constant value with “Real constant” selected in the variable type of divide value.	Variable No.: 1~200 Constant value: -1e+038~ 1e+038

■ Example of screen display

DIVVF[V1!,V2!]	FN644;Divide real variable

The parameters appeared on the screen are “Variable of divide destination”, “Variable of divide value (or divide constant)” from the left.

See
ADDVF; Add real variable (FN638)
SUBVF; Subtract real variable (FN640)
MULVF; Multiply real variable (FN642)

Function commands (FN codes)

Command name	ASIN
FN code	648
Title name	LET ASIN function
General description	Get ASIN variables

■ General description

It calculates the real value of ASIN and set the result to real variable.

■ Example of operation

Record ASIN, designated global variable1(V1!) and original global variable2(V2!). After acting the function, ASIN value of V2! is stored V1!
You can see the values by some monitors. You can use Local variables too.

Functionality of ASIN[V1!,V2!]

Global variables	Before ASIN	After ASIN
V1!	0.0000	1.5708
V2!	1.0000	0.0000

■ Parameters

Parameters	Contents	Range
Type of designated variables	Set the type of designated variable	Real Global Var. / Real Local Var.
Variable number	Set the number of variables	1～200
Type of original value	Set the type of original value ※You can set the value directly if you use “Real constant value”	Real Global Var. / Real Local Var. Real constant value
Variable number or Constant value	Set variable number if you use variable type Set real constant if you use Real constant value.	Variables : 1～200 Constant : -1e+038～1e+038

■ Example of screen display

ASIN[V1!,V2!]	FN648;ASIN

Displayed parameters on the screen from left are “Designated variables”, “Original value set by some variables, or real value”

See
ACOS; Let ACOS function (FN649)
SIN; Let SIN function (FN652)
COS; Let COS function (FN653)
TAN; Let TAN function (FN654)
ATN; Let ATN function (FN655)
ATN2; Let ATN2 function (FN656)

Function commands (FN codes))

Command name	ACOS
FN code	649
Title name	LET ACOS function
General description	Get ACOS variables

■General description
It calculates the real value of ACOS and set the result to real variable.

■ Example of operation
Record ACOS, designated global variable1(V1!) and original global variable2(V2!). After acting the function, ACOS value of V2! is stored V1!
You can see the values by some monitors. You can use Local variables too.

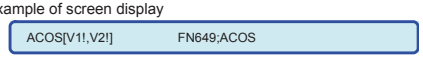
Functionality of ACOS[V1!,V2!]

Global variables	Before ACOS	After ACOS
V1!	0.0000	1.5708
V2!	1.0000	0.0000

■ **Parameters**

Parameters	Contents	Range
Type of designated variables	Set the type of designated variable	Real Global Var. / Real Local Var.
Variable number	Set the number of variables	1～200
Type of original value	Set the type of original value ※You can set the value directly if you use "Real constant value"	Real Global Var. / Real Local Var. Real constant value
Variable number or Constant value	Set variable number if you use variable type Set real constant if you use Real constant value.	Variables： 1～200 Constant： -1e+038～1e+038

■Example of screen display



Displayed parameters on the screen from left are "Designated variables", "Original value set by some variables, or real value"

See
ASIN; Let ASIN function (FN648)
SIN; Let SIN function (FN652)
COS; Let COS function (FN653)
TAN; Let TAN function (FN654)
ATN; Let ATN function (FN655)
ATN2; Let ATN2 function (FN656)

Function Commands (FN Codes)

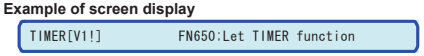
Command name	TIMER
FN code	650
Title name	Let TIMER function
General description	Sets the time value passed since the power-on into the specified real variable register

■ **General description**
This function command allows to set the time value into the specified real variable register by 1ms.

■ **Parameter**

1st parameter	Real variable No.	Designates the real variable No. to set. (1-200)

■ **Example of screen display**



Command name	SQR
FN code	651
Title name	Let SQR function
General description	Calculates the square root of the real number

■ General description

This function command allows to calculate the square root of the real number and set its return value into the specified real variable register.

■ Example of operation

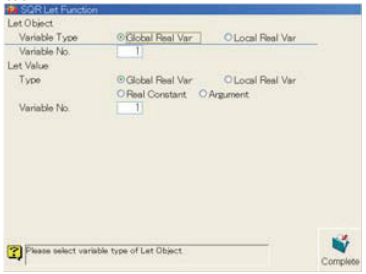
Any steps can be recorded by “SQR: Let SQR function (FN651)”, “Global variable No. of the set destination=1”, and “Global variable No. of the set value=2”. The square root of the 2nd variable is set into the 1st variable by playback of this command. The set variables can be checked by the register screen of global real number variable on the monitor. (Local variables are also available.)

Execution sample of SQR [V1!, V2!]

Global variable	Before executing SQR	After executing SQR
V1!	0.0000	2.0000
V2!	4.0000	4.0000

■ SQR Teach Method

(1) Select FN651 “Let SQR function”.



(2) Select the variable type of the set destination.
(3) Enter the variable No. of the set destination.
(4) Select the type of set value.
(5) Enter the variable No. of the set value or set constant.
(6) Press F12 <Complete> to record SQR.

■ Parameter

Parameter name	Description	Range
Variable type of set destination	Enters the variable type of the set destination.	Global real variable/ Local real variable
Variable No. of set destination	Enters the variable No. of the set destination.	1~200
Type of set value	Enters the type of set value. ※ Selecting “Real constant”, the set value can be directly designated.	Global real variable/ Local real variable/ Real constant
Variable No. of set value or Set constant	Enters the variable No. with “Variable” selected, and the constant value with “Real constant” selected in the variable type of set value.	Variable No.: 1~200 Constant value: -1e+038~ 1e+038

■ Example of screen display

```
SQR[V1!,V2!]          FN651:Let SQR function
```

The parameters appeared on the screen are “Variable of set destination” and “Variable of set value (or set constant)” from the left.

Command name	SIN
FN code	652
Title name	Let SIN function
General description	Calculates the SIN value of real number

■ General description

This function command allows to calculate the SIN value of the real number and set its return value into the specified real variable register.

■ Example of operation

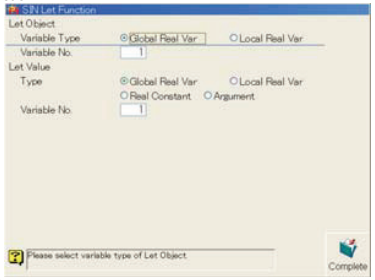
Any steps can be recorded by “SIN: Let SIN function (FN652)”, “Global variable No. of the set destination=1”, and “Global variable No. of the set value=2”. The SIN value of the 2nd variable is set into the 1st variable by playback of this command. The set variables can be checked by the register screen of global real number variable on the monitor. (Local variables are also available.)

Execution sample of SIN [V1!, V2!]

Global variable	Before executing SIN	After executing SIN
V1!	0.0000	0.8415
V2!	1.0000	1.0000

■ SIN Teach Method

(1) Select FN652 “Let SIN function”.



(2) Select the variable type of the set destination.
(3) Enter the variable No. of the set destination.
(4) Select the type of set value.
(5) Enter the variable No. of the set value or set constant.
(6) Press F12 <Complete> to record SIN.

■ Parameter

Parameter name	Description	Range
Variable type of set destination	Enters the variable type of the set destination.	Global real variable/ Local real variable
Variable No. of set destination	Enter the variable No. of the set destination.	1~200
Type of set value	Enters the type of set value. ※ Selecting “Real constant”, the set value can be directly designated.	Global real variable/ Local real variable/ Real constant
Variable No. of set value or Set constant	Enters the variable No. with “Variable” selected, and the constant value with “Real constant” selected in the variable type of set value.	Variable No.: 1~200 Constant value: -1e+038~ 1e+038

■ Example of screen display

```
SIN[V1!,V2!]          FN652:Let SIN function
```

The parameters appeared on the screen are “Variable of set destination” and “Variable of set value (or set constant)” from the left.

See
COS; Let COS function (FN653)
TAN; Let TAN function (FN654)
ATN; Let ATN function (FN655)
ATN2; Let ATN2 function (FN656)

Command name	COS
FN code	653
Title name	Let COS function
General description	Calculates the COS value of real number

■ General description

This function command allows to calculate the COS value of the real number and set its return value into the specified real variable register.

■ Example of operation

Any steps can be recorded by “COS: Let COS function (FN653)”, “Global variable No. of the set destination=1”, and “Global variable No. of the set value=2”. The COS value of the 2nd variable is set to the 1st variable by playback of this command. The set variables can be checked by the register screen of global real number variable on the monitor. (Local variables are also available.)

Execution sample of COS [V1!, V2!]

Global variable	Before executing COS	After executing COS
V1!	0.0000	0.5403
V2!	1.0000	1.0000

■ COS Teach Method

(1) Select FN653 “Let COS function”.

(2) Select the variable type of the set destination.
(3) Enter the variable No. of the set destination.
(4) Select the type of set value.
(5) Enter the variable No. of the set value or set constant.
(6) Press F12 <Complete> to record COS.

■ Parameter

Parameter name	Description	Range
Variable type of set destination	Enters the variable type of the set destination.	Global real variable/ Local real variable
Variable No. of set destination	Enter the variable No. of the set destination.	1~200
Type of set value	Enters the type of set value. ※ Selecting “Real constant”, the set value can be directly designated.	Global real variable/ Local real variable/ Real constant
Variable No. of set value or Set constant	Enters the variable No. with “Variable” selected, and the constant value with “Real constant” selected in the variable type of set value.	Variable No.: 1~200 Constant value: -1e+038~ 1e+038

■ Example of screen display

```
COS[V1!,V2!]        FN653:Let COS function
```

The parameters appeared on the screen are “Variable of set destination” and “Variable of set value (or set constant)” from the left.

See
SIN; Let SIN function (FN652)
TAN; Let TAN function (FN654)
ATN; Let ATN function (FN655)
ATN2; Let ATN2 function (FN656)

Command name	TAN
FN code	654
Title name	Let TAN function
General description	Calculates the TAN value of real number

■ General description

This function command allows to calculate the TAN value of real number and set its return value into the specified real variable register.

■ Example of operation

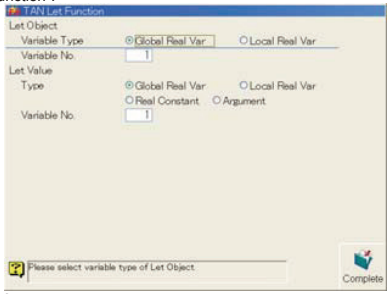
Any steps can be recorded by “TAN: Let TAN function (FN654)”, “Global variable No. of the set destination=1”, and “Global variable No. of the set value=2”. The TAN value of the 2nd variable is set into the 1st variable by performing playback of this command. The set variables can be checked by the register screen of global real number variable on the monitor. (Local variables are also available.)

Execution sample of TAN [V1!, V2!]

Global variable	Before executing TAN	After executing TAN
V1!	0.0000	1.5574
V2!	1.0000	1.0000

■ TAN Teach Method

(1) Select FN654 “Let TAN function”.



(2) Select the variable type of the set destination.
(3) Enter the variable No. of the set destination.
(4) Select the type of set value.
(5) Enter the variable No. of the set value or set constant.
(6) Press F12 <Complete> to record TAN.

■ Parameter

Parameter name	Description	Range
Variable type of set destination	Enters the variable type of the set destination.	Global real variable/ Local real variable
Variable No. of set destination	Enters the variable No. of the set destination.	1~200
Type of set value	Enters the type of set value. ※ Selecting “Real constant”, the set value can be directly designated.	Global real variable/ Local real variable/ Real constant
Variable No. of set value or Set constant	Enters the variable No. with “Variable” selected, and the constant value with “Real constant” selected in the variable type of set value.	Variable No.: 1~200 Constant value: -1e+038~ 1e+038

■ Example of screen display

```
TAN[V1!,V2!]        FN654:Let TAN function
```

The parameters appeared on the screen are “Variable of set destination” and “Variable of set value (or set constant)” from the left.

See
SIN; Let SIN function (FN652)
COS; Let COS function (FN653)
ATN; Let ATN function (FN655)
ATN2; Let ATN2 function (FN656)

Command name	ATN
FN code	655
Title name	Let ATN function
General description	Calculates the ATN value of real number

■ General description

This function command allows to calculate the ATN value of the real number and set its return value into the specified real variable register.

■ Example of operation

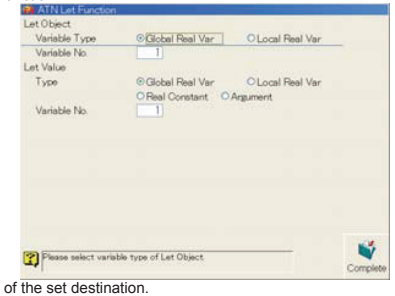
Any steps can be recorded by “ATN: Let ATN function (FN655)”, “Global variable No. of the set destination=1”, and “Global variable No. of the set value=2”. By playback of this command, the ATN value of the 2nd variable is set into the 1st variable. The set variables can be checked by the register screen of global real number variable on the monitor. (Local variables are also available.)

Execution sample of ATN [V1!, V2!]

Global variable	Before executing ATN	After executing ATN
V1!	0.0000	0.7854
V2!	1.0000	1.0000

■ ATN Teach Method

(1) Select FN655 “Let ATN function”.



(2) Select the variable type of the set destination.
(3) Enter the variable No. of the set destination.
(4) Select the type of set value.
(5) Enter the variable No. of the set value or set constant value.
(6) Press F12 <Complete> to record ATN.

■ Parameter

Parameter name	Description	Range
Variable type of set destination	Enters the variable type of the set destination.	Global real variable/ Local real variable
Variable No. of set destination	Enter the variable No. of the set destination.	1~200
Type of set value	Enters the type of set value. ※ Selecting “Real constant”, the set value can be directly designated.	Global real variable/ Local real variable/ Real constant
Variable No. of set value or Set constant	Enters the variable No. with “Variable” selected, and the constant value with “Real constant” selected in the variable type of set value.	Variable No.: 1~200 Constant value: -1e+038~ 1e+038

■ Example of screen display

```
ATN[V1!,V2!]          FN655:Let ATN function
```

The parameters appeared on the screen are “Variable of set destination” and “Variable of set value (or set constant)” from the left.

See
SIN; Let SIN function (FN652)
COS; Let COS function (FN653)
TAN; Let TAN function (FN654)
ATN2; Let ATN2 function (FN656)

Command name	ATN2
FN code	656
Title name	Let ATN2 function
General description	Calculates the ATN2 value of real number

■ General description

This function command allows to calculate the ATN2 value of the real number and set its return value into the specified real variable register.

■ Example of operation

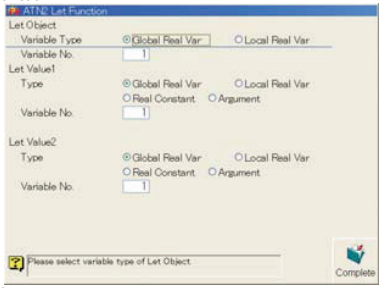
Any steps can be recorded by “ATN2: Let ATN2 function (FN656)”, “Global variable No. of the set destination=1”, “Global variable No.1 of the set value=2” and “Global variable No.2 of the set value=3”. By playback of this command, the ATN2 value of the 2nd and 3rd variable are set into the 1st variable. The set variables can be checked by the register screen of global real variable on the monitor. (Local variables are also available.)

Execution sample of ATN2 [V1!, V2!, V3!]

Global variable	Before executing ATN2	After executing ATN2
V1!	0.0000	0.4636
V2!	1.0000	1.0000
V3!	2.0000	2.0000

■ ATN2 Teach Method

(1) Select FN656 “Let ATN2 function”.



(2) Select the variable type of the set destination.
(3) Enter the variable No. of the set destination.
(4) Select the type of set value 1.
(5) Enter the variable No. of the set value 1 or the set constant.
(6) Select the type of set value 2.
(7) Enter the variable No. of the set value 2 or the set constant.
(8) Press F12 <Complete> to record ATN2.

■ Parameter

Parameter name	Description	Range
Variable type of set destination	Enters the variable type of the set destination.	Global real variable/ Local real variable
Variable No. of set destination	Enters the variable No. of the set destination.	1~200
Type of set value 1	Enters the type of set value 1. ※ Selecting “Real constant”, the set value can be directly designated.	Global real variable/ Local real variable/ Real constant
Variable No. of set value 1 or Set constant 1	Enters the variable No. with “Variable” selected, and the constant value with “Real constant” selected in the variable type of set value 1.	Variable No.: 1~200 Constant value: -1e+038~1e+038
Type of set value 2	Enters the type of set value 2. ※ Selecting “Real constant”, the set value can be directly designated.	Global real variable/ Local real variable/ Real constant
Variable No. of set value 2 or Set constant 2	Enters the variable No. with “Variable” selected, and the constant value with “Real constant” selected in the variable type of set value 2.	Variable No.: 1~200 Constant value: -1e+038~1e+038

■ Example of screen display

```
ATN2[V1!,V2!,V3!]          FN656:Let ATN2 function
```

The parameters appeared on the screen are “Variable of set destination”, “Variable of set value 1 (or set constant 1)”, and “Variable of set value 2 (or set constant 2)” from the left.

See
SIN; Let SIN function (FN652)
COS; Let COS function (FN653)
TAN; Let TAN function (FN654)
ATN; Let ATN function (FN655)

Command name	ABS
FN code	657
Title name	Let ABS function
General description	Calculates the absolute value of real number

■ General description

This function command allows to calculate the absolute value of real number and set its return value into the specified real variable register.

■ Example of operation

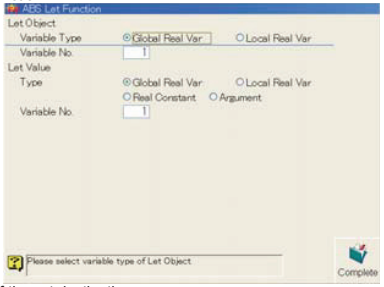
Any steps can be recorded by “ABS: Let ABS function (FN657)”, “Global variable No. of the set destination=1” and “Global variable No. of the set value=2”. By playback of this command, the absolute value of the 2nd variable is set into the 1st variable. The set variables can be checked by the register screen of global real variable on the monitor. (Local variables are also available.)

Execution sample of ABS [V1!, V2!]

Global variable	Before executing ABS	After executing ABS
V1!	0.0000	1.1000
V2!	-1.1000	-1.1000

■ ABS Teach Method

(1) Select FN657 “Let ABS function”.



(2) Select the variable type of the set destination.
(3) Enter the variable No. of the set destination.
(4) Select the type of set value.
(5) Enter the variable No. of the set value or set constant.
(6) Press F12 <Complete> to record ABS.

■ Parameter

Parameter name	Description	Range
Variable type of set destination	Enters the variable type of the set destination.	Global real variable/ Local real variable
Variable No. of set destination	Enters the variable No. of the set destination.	1~200
Type of set value	Enters the type of set value. ※ Selecting “Real constant”, the set value can be directly designated.	Global real variable/ Local real variable/ Real constant
Variable No. of set value or Set constant	Enters the variable No. with “Variable” selected, and the constant value with “Real constant” selected in the variable type of set value.	Variable No.: 1~200 Constant value: -1e+038~1e+038

■ Example of screen display

```
ABS[V1!,V2!]          FN657:Let ABS function
```

The parameters appeared on the screen are “Variable of set destination” and “Variable of set value (or substitution constant)” from the left.

Command name	MIN
FN code	658
Title name	Let MIN function
General description	Calculates a smaller real number out of two

■ General description

This function command allows to set a smaller real number out of two into the real variable register.

■ Example of operation

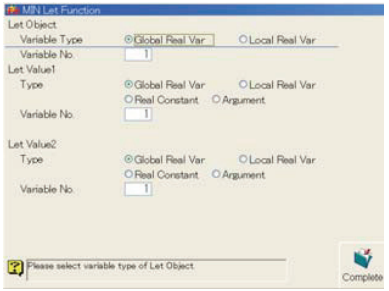
Any steps can be recorded by “MIN: Let MIN function (FN658)”, “Global variable No. of the set destination=1”, “Global variable No. of the set value 1 = 2”, and “Global variable No. of the set value 2 =3”. By playback of this command, a smaller variable of the 2nd or 3rd is set into the 1st variable. The set variables can be checked by the register screen of global real variable on the monitor. (Local variables are also available.)

Execution sample of MIN [V1!,V2!, V3!]

Global variable	Before executing MIN	After executing MIN
V1!	0.0000	1.0000
V2!	1.0000	1.0000
V3!	2.0000	2.0000

■ MIN Teach Method

(1) Select FN658 “Let MIN function”.



(2) Select the variable type of the set destination.
(3) Enter the variable No. of the set destination.
(4) Select the type of set value 1.
(5) Enter the variable No. of the set value 1 or the set constant.
(6) Select the type of set value 2.
(7) Enter the variable No. of the set value 2 or the set constant.
(8) Press F12 <Complete> to record MIN.

■ Parameter

Parameter name	Description	Range
Variable type of set destination	Enters the variable type of the set destination.	Global real variable/ Local real variable
Variable No. of set destination	Enters the variable No. of the set destination.	1~200
Type of set value 1	Enters the type of set value 1. ※ Selecting “Real constant”, the set value can be directly designated.	Global real variable/ Local real variable/ Real constant
Variable No. of set value 1 or Set constant 1	Enters the variable No. with “Variable” selected, and the constant value with “Real constant” selected in the variable type of set value 1.	Variable No.: 1~200 Constant value: -1e+038~1e+038
Type of set value 2	Enters the type of set value 2. ※ Selecting “Real constant”, the substitution value can be directly designated.	Global real variable/ Local real variable/ Real constant
Variable No. of set value 2 or Set constant 2	Enters the variable No. with “Variable” selected, and the constant value with “Real constant” selected in the variable type of set value 2.	Variable No.: 1~200 Constant value: -1e+038~1e+038

■ Example of screen display

```
MIN[V1!,V2!,V3!]          FN658:Let MIN function
```

The parameters appeared on the screen are “Variable of set destination”, “Variable of set value 1 (or set constant 1)” and “Variable of set value 2 (or set constant 2)” from the left.

See
MAX; Let MAX function (FN659)

Command name	MAX
FN code	659
Title name	Let MAX function
General description	Calculates a larger real number out of two

■ **General description**

This function command allows to set a larger real number of the two into the real variable register.

■ **Example of operation**

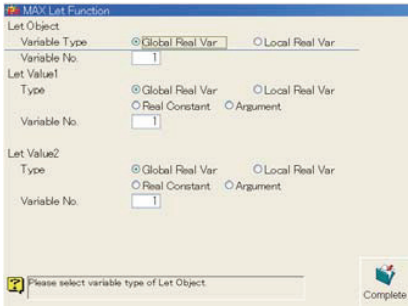
Any steps can be recorded by “MAX: Let MAX function (FN659)”, “Global variable No. of the set destination=1”, “Global variable No. of the set value 1 =2” and “Global variable No. of the set value 2 =3”. By playback of this command, a larger variable of the 2nd or 3rd is set into the 1st variable. The set variables can be checked by the register screen of global real variable on the monitor. (Local variables are also available.)

Execution sample of MAX [V1!, V2!, V3!]

Global variable	Before executing MAX	After executing MAX
V1!	0.0000	2.0000
V2!	1.0000	1.0000
V3!	2.0000	2.0000

■ **MAX Teach Method**

(1) Select FN659 “Let MAX function”.



(2) Select the variable type of the set destination.
(3) Enter the variable No. of the set destination.
(4) Select the type of set value 1.
(5) Enter the variable No. of the set value 1 or set constant.
(6) Select the type of set value 2.
(7) Enter the variable No. of the set value 2 or set constant.
(8) Press F12 <Complete> to record MAX.

■ **Parameter**

Parameter name	Description	Range
Variable type of set destination	Enters the variable type of the set destination.	Global real variable/ Local real variable
Variable No. of set destination	Enters the variable No. of the set destination.	1~200
Type of set value 1	Enters the type of set value 1. ※ Selecting “Real constant”, the set value can be directly designated.	Global real variable/ Local real variable/ Real constant
Variable No. of set value 1 or Set constant 1	Enters the variable No. with “Variable” selected, and the constant value with “Real constant” selected in the variable type of set value 1.	Variable No.: 1~200 Constant value: -1e+038～1e+038
Type of set value 2	Enters the type of set value 2. ※ Selecting “Real constant”, the substitution value can be directly designated.	Global real variable/ Local real variable/ Real constant
Variable No. of set value 2 or Set constant 2	Enters the variable No. with “Variable” selected, and the constant value with “Real constant” selected in the variable type of set value 2.	Variable No.: 1~200 Constant value: -1e+038～1e+038

■ **Example of screen display**

```
MAX[V1!,V2!,V3!]          FN659:Let MAX function
```

The parameters appeared on the screen are “Variable of set destination”, “Variable of set value 1 (or set constant 1)” and “Variable of set value 2 (or set constant 2)” from the left.

See
MIN; Let MIN function (FN658)

Command name	WHILE
FN code	663
Title name	WHILE loop
Outline	Execute the instruction in WHILE-ENDW repeatedly until the condition doesn't consist.

■ **General description**

This application instructions repeatedly executes the instruction in WHILE-ENDW until the result of evaluating the condition becomes 0. The condition is evaluated before the loop is executed every time. Therefore, the WHILE loop is executed 0 times or more. Use it together with ENDW. In the following example, while V1% is less than 10,the instructions between WHILE and ENDW are executed repeatedly.

Example:
V1%=0
WHILE V1%<10
Instruction 1
Instruction 2
V1%=V1%+1
ENDW

■ **Parameter**

Parameter No. 1	Conditional expression	It is a conditional expression. The conditional expression that compare the integer variable or the real number variable can be described now.

■ **Method of specifying the first parameter**

The first parameter can be freely described with a soft keyboard.

The conditional expression that compare the integer variable or the real number variable can be described now.

description example:

```
WHILE V1% = 1
      ;  While integer variable 1 is equal to 1
WHILE V2! < 0
      ;  While real number variable 2 is Negative
WHILE V3% > 1
      ;  While integer variable 3 is larger than 1
```

- Integer variable

Treat the numerical value that doesn't contain the decimal point.

Format	Vn% and V% n ; n=1-200 (global integer variable) Ln% and L% n ; n=1-200 (local integer variable) - Vn% and V% n are the same meanings.
Range	-2147483648～+2147483647
Sample	WHILE V1%=1 ;while variable 1 is one WHILE V2%<>0 ;while variable 2 is not 0
Preservation	Even if the main power supply is cut, all a global integer variables are preserved. All a local integer variables are not preserved.

- Real number variable

Treat the numerical value including the decimal point.

Format	Vn!, V!n ; n=1-100 (global real variable) Ln!, L!n ; n=1-100 (local real variable) - Vn! With V. n is the same meaning.
Range	-1.0E38～+1.0E38
Sample	WHILE V1!>0 While variable 1 is larger than 0
Preservation	Even if the main power supply is cut, all a global real number variables are preserved. All a local real number variables are not preserved.

- Relational operation type

Use the relational operation when you compare two numerical values. The result obtains by truth (1) and false (0), and is used for the change in the order of executing the program like the condition judgment sentence etc. etc.

Operator	Content of operation	Example
=	It is equal.	V1%=V2%
<>	It is not equal.	V1%<>V2%
<	It is small.	V1%<V2%
>	It is large.	V1%>V2%
<=	It is small or equal.	V1%<=V2%
>=	It is large or is equal.	V1%>=V2%

■ **Example of screen display**

```
WHILE V1%<10              FN663;WHILE loop
```

See
ENDW: WHILE end (FN664)

Function commands (FN codes)

Command name	ENDW
FN code	664
Title name	WHILE end
Outline	It is terminator of the WHILE-ENDW structure.

■ General description

Execute the instructions described in the WHILE-ENDW loop repeatedly.
Use it together with WHILE.

■ Parameter

None.

■ Example of screen display

ENDW	FN664;WHILE loop end

See
WHILE; WHILE loop (FN663)

Function commands (FN codes)

Command name	PRINTF
FN code	669
Title name	Print string with format
General description	Draw the string data on the screen with form. Or Output string data with form via RS232C.

■ General description

The function draws some string data and output some string data via RS232C.
You should select “Device”. Device number must start character”#”. The form is based on the “Printf” function of C language library. Corresponding sub parameters will translate to some string data and output to the device. Sub parameters will be designate max.5, and you can skip too.

■Example of operation

When the robot reaches the step where FN669 is recorded, the function will act with no motion pause.

■ Parameters

1st parameter	Device number	Set port number of string code #0： User display(It is only available by User task） #1： RS232C port #2： Not supported now #3： Not supported now #4： Playback logging monitor
2nd parameter	String data with form	Set the string data. You can set the data from Soft keyboard by ［Enable］＋［Edit］. （Max. 100 characters） Corresponding sub parameters will translate to some string data and output to the device if you use below string forms You must start 「%」character. If you use 「%」itself, you should put 「%%」 % [flag] [] []
3rd parameter	1st sub parameter	Set the variable number. You can use Integer variables, Real variables, String variables, and Optional variables (You can also skip).
4th parameter	2nd sub parameter	
5th parameter	3rd sub parameter	
6th parameter	4th sub parameter	
7th parameter	5th sub parameter	
8th parameter	6th sub parameter	

■ Example of screen display

PRINT[#4, "V1%% = %d", V1%]	FN669;Print string with format

Function commands (FN codes)

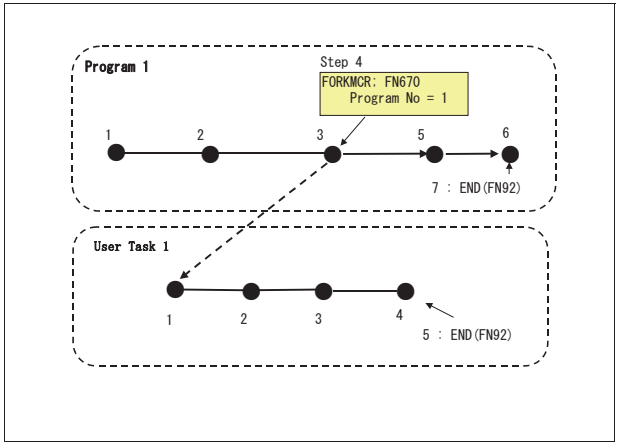
Command name	FORKMCR
FN code	670
Title name	Fork User Task Program
General description	This command is used to start the specified user task program.

■ General description

When this function command is executed, the specified user task program is started. Started user task program finish after the program is executed once from start to end.

■ Example of operation

In step 4, record FORKMCR: User Task Program Start (FN670) and "1" as the program number. When this is played back, the program1 start the user task program 1 upon arriving at step 4. (program1 and user task 1 operate concurrently.)
When the start of User Task 1 fails, the program1 jumps shelter step.



■ Parameter

Parameter No. 1	Program No.	This specifies the number of the program serving as the call destination. (1-999)
Parameter No. 2	Shelter step	This is used to specify the number of the shelter step when the specified user task program was not starting. (1 to 10000) When 10000 is specified as the shelter step number, an alarm results immediately with no escape operation performed, and the robot can be stopped.

■ Example of screen display

FORKMCR [1,6] FN670; Fork User Task Program

See
CALLMCR: Call user task program (FN671)
FORKMCRTM: Fork user task program (Time) (FN672)
FORKMCRDST: Fork user task program (Distance) (FN673)

Function commands (FN codes)

Command name	CALLMCR
FN code	671
Title name	Call User Task Program
General description	This command is used to call the specified user task program.

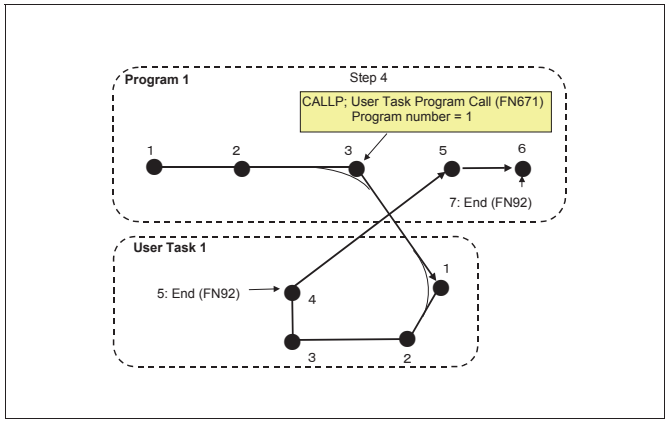
■ General description

When this function command is executed, the specified user task program is called. When program call user task program, program stop one's playback and start specified user task program.
When the playback of the user task program at the call destination is completed (in the status established by executing the END command), the program returns to the step following the step with the call command of the call source program.

■ Example of operation

In step 4, record CALLMCR: User Task Program Call (FN671) and "1" as the program number.

When this is played back, the program1 call the user task program 1 upon arriving at step 4. When the start of User Task 1 fails, the program1 jumps shelter step.
When the playback of user task program 1 is completed (in the status established by executing the END command), the robot returns to step 5 following the step with the call command of call source program 1.



■ Parameter

Parameter No. 1	Program No.	This specifies the number of the program serving as the call destination. (1-999)
Parameter No. 2	Shelter step	This is used to specify the number of the shelter step when the specified user task program was not starting. (1 to 10000) When 10000 is specified as the shelter step number, an alarm results immediately with no escape operation performed, and the robot can be stopped.

■ Example of screen display

CALLMCR [1,6] FN671; Call User Task Program

See
FORKMCR: fork user task program (FN670)
FORKMCRTM: Fork user task program (Time) (FN672)
FORKMCRDST: Fork user task program (Distance) (FN673)

Function commands (FN codes)

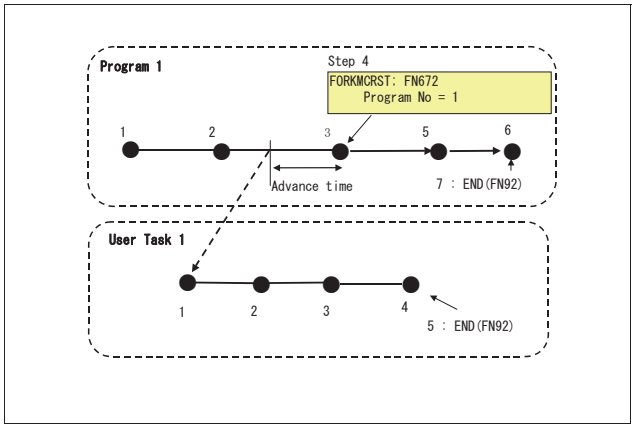
Command name	FORKMCRTM
FN code	672
Title name	Fork User Task Program (time)
General description	This command is used to start the specified user task program. Furthermore, the command enables advance execution to be specified.

■ General description

When this function command is executed, the specified user task program is started. Furthermore, the command enables advance execution earlier than at the position reached by the robot in the accuracy range of the move command.

■ Example of operation

In step 4, record FORKMCRTM : User Task Program Start with early execution (FN672) and "1" as the program number.
When this is played back, the program1 start the user task program 1 before specified seconds from arriving at step 4. (program1 and user task 1 operate concurrently.)
When the start of User Task 1 fails, the program1 jumps shelter step.



■ Parameter

Parameter No. 1	Program No.	This specifies the number of the program serving as the call destination. (1-999)
Parameter No. 2	Advance execution time	If "0.0" is specified as the time, the command is executed at the timing which coincides with the recorded point. the command is starting user task program ahead of the original execution timing by the amount equivalent to the delay time setting.(Increment: seconds) (-10.0 – 0.0)
Parameter No. 3	Shelter step	This is used to specify the number of the shelter step when the specified user task program was not starting. (1 to 10000) When 10000 is specified as the shelter step number, an alarm results immediately with no escape operation performed, and the robot can be stopped.

■ Example of screen display

```
FORKMCRTM [1,-2,6]      FN672; Fork User Task   Program(Time)
```

See
FORKMCR: Fork user task program (FN670)
CALLMCR: Call user task program (FN671)
FORKMCRDST: Fork user task program (Distance) (FN673)

Function commands (FN codes)

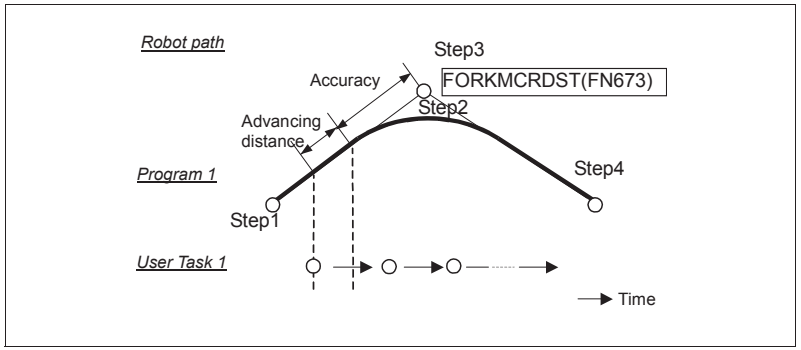
Command name	FORKMCRDST
FN code	673
Title name	Fork User Task Program (distance)
General description	This command is used to start the specified user task program with advancing distance.

■ General description

When this function command is executed, the specified user task program is started. Furthermore, the command enables advanced execute designated by distance from the recorded point. Advancing distance is calculated linear length from recorded point, so if MOVE step is recorded by JOINT output point differs from the real moving length of robot TCP.

■ Example of operation

In step 3, record FORKMCRDST : Fork User Task Program (distance) (FN672) and "1" as the program number.
When this is played back, the program1 start the user task program 1 before specified distance from arriving at step 3. (Program1 and user task 1 operate concurrently.)
When the start of User Task 1 fails, the program1 jumps shelter step.



■ Parameter

Parameter No. 1	Program No.	This specifies the number of the program serving as the call destination. (1-999)
Parameter No. 2	Advance execution timing	This specifies the output advancing timing that is designated by the linear length from the recorded point. (-1500.0 – 0.0)
Parameter No. 3	Shelter step	This is used to specify the number of the shelter step when the specified user task program was not starting. (1 to 10000) When 10000 is specified as the shelter step number, an alarm results immediately with no escape operation performed, and the robot can be stopped.

■ Example of screen display

```
FORKMCRDST [1,-100,6]      FN670; Fork User Task Program (Distance)
```

See
FORKMCR: Fork user task program (FN670)
CALLMCR: Call user task program (FN671)
FORKMCRTM: Fork user task program (Time) (FN672)

Function commands (FN codes)

Command name	IF
FN code	676
Title name	Condition
Outline	Move the control to the following instruction when the condition consists. Move the control to ELSEIF, ELSE, and ENDIF for the failure.

■ General description

It is conditional.
Use it together with ELSEIF, ELSE, and ENDIF.

IF conditional expression
Instruction 1
Instruction 2
・・
Instruction I
ELSEIF conditional expression
Instruction J
・・
Instruction K
ELSE
Instruction L
・・
Instruction M
ENDIF

If the condition of the IF instruction consists, it moves the control to instruction 1 and moves the control to ENDIF after executing instruction I.

If the condition of the IF instruction is a failure, it moves the control to ELSEIF. If the condition of the ELSEIF instruction consists, it moves the control to instruction J and moves the control to ENDIF after executing instruction K.
If the condition of the ELSEIF instruction is a failure, it moves the control to ELSE and moves to control to instruction L and moves the control to ENDIF after executing instruction M.

■ Parameter

Parameter No. 1	Conditional expression	It is a conditional expression. The conditional expression that comparing operates the integer variable or the real number variable can be described now.

■ Method of specifying the first parameter

The first parameter can be freely described with a soft keyboard.

The conditional expression that comparing operates the integer variable or the real number variable can be described now.

description example:

```
IF V1% = 1
        ;   if integer variable 1 is 1
IF V2! > 0
        ;   if real number variable 2 is larger than 0
IF V3% >= 1
        ;   When integer variable 3 is 1 or more.
```

■ Example of screen display

```
IF V1%<10                FN676;IF condition
```

See
ELSEIF; ELSEIF Condition (FN677)
ELSE; ELSE (FN678)
ENDIF; Condition end (FN679)

Function commands (FN codes)

Command name	ELSEIF
FN code	677
Title name	Condition
Outline	Move the control to the following instruction when the condition consists. Move the control to ELSE and ENDIF for the failure.

■ General description

It is conditional.
Use it together with IF, ELSEIF, ELSE, and ENDIF.

IF conditional expression
Instruction 1
Instruction 2
・・
Instruction I
ELSEIF conditional expression
Instruction J
・・
Instruction K
ELSE
Instruction L
・・
Instruction M
ENDIF

If the condition of the IF instruction consists, it moves the control to instruction 1 and moves the control to ENDIF after executing instruction I.

If the condition of the IF instruction is a failure, it moves the control to ELSEIF. If the condition of the ELSEIF instruction consists, it moves the control to instruction J and moves the control to ENDIF after executing instruction K.
If the condition of the ELSEIF instruction is a failure, it moves the control to ELSE and moves to control to instruction L and moves the control to ENDIF after executing instruction M.

■ Parameter

Parameter No. 1	Conditional expression	It is a conditional expression. The conditional expression that comparing operates the integer variable or the real number variable can be described now.

■ Method of specifying the first parameter

The first parameter can be freely described with a soft keyboard.

The conditional expression that comparing operates the integer variable or the real number variable can be described now.

description example:

```
IF V1% = 1
        ;   if integer variable 1 is 1
IF V2! > 0
        ;   if real number variable 2 is larger than 0
IF V3% >= 1
        ;   When integer variable 3 is 1 or more.
```

■ Example of screen display

```
ELSEIF V1%=10            FN677;ELSEIF condition
```

See
IF; IF Condition (FN676)
ELSEIF; ELSEIF Condition (FN677)
ELSE; ELSE (FN678)
ENDIF; Condition end (FN679)

Function commands (FN codes)

Command name	ELSE
FN code	678
Title name	Condition
Outline	Move the control to the following instruction.

■ **General description**

It is conditional.
Use it together with IF, ELSEIF, ELSE, and ENDIF.

IF conditional expression
Instruction 1
Instruction 2
・・
Instruction I
ELSEIF conditional expression
Instruction J
・・
Instruction K
ELSE
Instruction L
・・
Instruction M
ENDIF

If the condition of the IF instruction consists, it moves the control to instruction 1 and moves the control to ENDIF after executing instruction I.

If the condition of the IF instruction is a failure, it moves the control to ELSEIF. If the condition of the ELSEIF instruction consists, it moves the control to instruction J and moves the control to ENDIF after executing instruction K.
If the condition of the ELSEIF instruction is a failure, it moves the control to ELSE and moves to control to instruction L and moves the control to ENDIF after executing instruction M.

■ **Parameter**

None.

■ **Example of screen display**

```
ELSE                FN678;ELSE
```

See
IF; IF Condition (FN676)
ELSEIF; ELSEIF Condition (FN677)
ELSE; ELSE (FN678)
ENDIF; Condition end (FN679)

Function commands (FN codes)

Command name	ENDIF
FN code	679
Title name	Condition end
Outline	End IF-ENDIF.

■ **General description**

It is a terminator of IF – ENDIF structure.
Use it together with IF, ELSEIF, ELSE, and ENDIF.

IF conditional expression
Instruction 1
Instruction 2
・・
Instruction I
ELSEIF conditional expression
Instruction J
・・
Instruction K
ELSE
Instruction L
・・
Instruction M
ENDIF

If the condition of the IF instruction consists, it moves the control to instruction 1 and moves the control to ENDIF after executing instruction I.

If the condition of the IF instruction is a failure, it moves the control to ELSEIF. If the condition of the ELSEIF instruction consists, it moves the control to instruction J and moves the control to ENDIF after executing instruction K.
If the condition of the ELSEIF instruction is a failure, it moves the control to ELSE and moves to control to instruction L and moves the control to ENDIF after executing instruction M.

■ **Parameter**

None.

■ **Example of screen display**

```
ENDIF               FN679;condition end
```

See
IF; Condition (FN676)
ELSEIF; Condition (FN677)
ELSE; Condition (FN678)

Command name	GFONT
FN code	683
Title name	Set the font
Outline	The font of the user screen is set.

■ **General description**

The font of the character string where it draws to the user screen is set.
If this function is executed, PRINTF; Print string (FN606) will draw the string of the font set by this function.
It returns to the default font when WINDOW; User screen open / close (FN607) is executed.

This command is available only in user task, and is permitted being written by robot language only. Please refer to the **"User task" operating manual** for the detail of user task and user screen.

■ **Parameter**

Parameter No. 1	Height	The height of the character is specified. When 0 is set, It become same height of the current font.
Parameter No. 2	Width	The width of the character is specified. When 0 is set, It become same width of the current font.
Parameter No. 3	Bold	0:disabled 1:enabled (Bold)
Parameter No. 4	Italic	0:disabled 1:enabled (Italic)
Parameter No. 5	Under line	0:disabled 1:enabled (Under line)

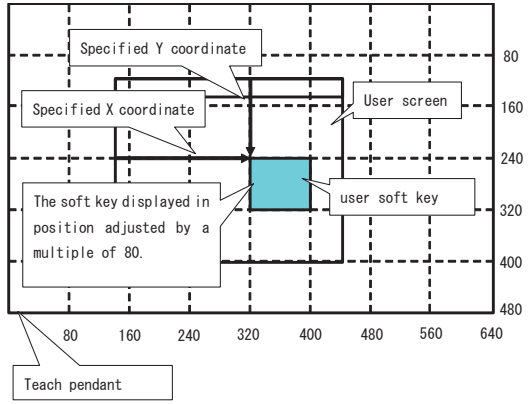
<SLIM sample> GFONT 16,8,0,0,0

See
PRINT ; Print String (FN606)
WINDOW ; Open/Close user display (FN607)
TITLE ; Set the title on user display (FN608)
CLS ; Set the title on user display (FN609)
LOCATE ; Locate the diplay pos (FN610)
GLINE ; Draw the line (FN611)
GBOX ; Draw the box (FN612)
BARC ; Draw the arc (FN613)
GPAINT ; Paint (FN614)
GSETP ; Draw the arc (FN615)
COLOR ; Ste the color (FN616)
BGCOLOR ; Set the bkgr color (FN617)
GARC ; Display ellipse (FN623)
GSOFTKEY ; Create soft key (FN684)
GMSGBOX ; Create message box (FN685)

Command name	GSOFTKEY
FN code	684
Title name	Create soft key
Outline	Create a soft key on the user screen.

■ **General description**

This function creates a soft key on the user screen.
The "AC00USK.INC file (User soft key definition file)" where the definition of a soft key is described is necessary to create a soft key by this function.
You can make or edit the user soft key definition file in [Service][User task][User task softkey] menu.
The user soft key definition file can be made or be edited by text editor because it is descried by the text form.
Please refer to "User task operation manual" for the description of the file.
The coordinates of the soft key are specified by relative coordinates on the upper-left of the user screen.
But, an actual display position of the user soft key become a position adjusted by a multiple of 80 on teach pendant.



This command is available only in user task, and is permitted being written by robot language only. Please refer to the **"User task" operating manual** for the detail of user task and user screen.

■ **Parameter**

Parameter No. 1	X coordinate	X coordinates of the user soft key are specified. (0 – 560)
Parameter No. 2	Y coordinate	Y coordinates of the user soft key are specified. (0 – 400)
Parameter No. 3	Soft key number	The soft key number described in the user soft key definition file (AC00USK.INC) is specified. When 0 is specified, a soft key to specified coordinates is deleted. (0 - 96)

<SLIM sample> GSOFTKEY 80,160,1

See
PRINT ; Print String (FN606)
WINDOW ; Open/Close user display (FN607)
TITLE ; Set the title on user display (FN608)
CLS ; Set the title on user display (FN609)
LOCATE ; Locate the diplay pos (FN610)
GLINE ; Draw the line (FN611)
GBOX ; Draw the box (FN612)
BARC ; Draw the arc (FN613)
GPAINT ; Paint (FN614)
GSETP ; Draw the arc (FN615)
COLOR ; Ste the color (FN616)
BGCOLOR ; Set the bkgr color (FN617)
GARC ; Display ellipse (FN623)
GFONT ; Set the font (FN683)
GMSGBOX ; Create message box (FN685)

Command name	GMSGBOX
FN code	685
Title name	Create message box
Outline	Create a message box on the user screen.

■ General description

This function creates a message box on the user screen.
It is for selecting some items by the user screen. When this function is executed, the user task stops processing until the button of the message box is selected. It is necessary to activate the user screen to display the message box.

This command is available only in user task, and is permitted being written by robot language only. Please refer to the **"User task" operating manual** for the detail of user task and user screen.

■ Parameter

Parameter No. 1	Display form	0: Display the "OK" button. 1: Display the "OK" and "Cancel" button. 2: Display the "YES" and "NO" button. 3: Display the "YES", "NO" and "Cancel" button. 4: Display the user definition buttons specified parameter No.6 – No.9.
Parameter No. 2	Icon	0: Not display an Icon. 1: Display the icon of warning. 2: Display the con of question. 3: Display the icon of caution. 4: Display the icon of information.
Parameter No. 3	Title	The string of the title bar is specified. (normal-width 20 characters)
Parameter No. 4	Text	The string of the text is specified. (normal-width 100 characters)
Parameter No. 5	Default	The default select button is specified. (1 -4)
Parameter No. 6	Button 1	When the user definition is specified by parameter No.1, the string of the button is specified. These parameters are not used, and please specify the empty string when specifying it by the parameter No.1 excluding the user definition. (normal-width 8 characters)
Parameter No. 7	Button 2	
Parameter No. 8	Button 3	
Parameter No. 9	Button 4	

<SLIM sample 1> GMSGBOX 2, 2, "Confirm", "Is counter reset?", 2, "", "", "", ""
<SLIM sample 2> GMSGBOX 4, 4, "Select processing",
"Please select the processing",
2, "set","all set","return","cancel"

See
PRINT ; Print String (FN606)
WINDOW ; Open/Close user display (FN607)
TITLE ; Set the title on user display (FN608)
CLS ; Set the title on user display (FN609)
LOCATE ; Locate the diplay pos (FN610)
GLINE ; Draw the line (FN611)
GBOX ; Draw the box (FN612)
BARC ; Draw the arc (FN613)
GPAINT ; Paint (FN614)
GSETP ; Draw the arc (FN615)
COLOR ; Ste the color (FN616)
BGCOLOR ; Set the bkgr color (FN617)
GARC ; Display ellipse (FN623)
GFONT ; Set the font(FN683)
GSOFTKEY ; Create soft key (FN684)

Command name	SWITCH
FN code	686
Title name	SWITCH
Outline	Two or more conditions are judged.

■ General description

Two or more conditions are judged.
Use it together with CASE, BREAK, and ENDS.

```
SWITCH V1%
CASE 1
Instruction 1
・・
Instruction I
BREAK
CASE 2
CASE 3
Instruction J
・・
Instruction K
BREAK
CASE
Instruction L
・・
Instruction M
BREAK
ENDS
```

When the value of integer variable V1% is one, it executes from instruction I to instruction K and moves the control to ENDS.
When the value of integer variable V1% is 2 or 3, it executes from Instruction J to instruction K and moves the control to ENDS.
When the value of integer variable V1% is not equal to 1, 2 and 3, it executes from Instruction L to instruction M and moves the control to ENDS.

■ Parameter

Parameter No. 1	Expression	It is an expression of the integer type.

■ Method of specifying the first parameter

The first parameter can be freely described with a soft keyboard.

■ Example of screen display

SWITCH V1%	FN686;SWITCH

See
CASE; CASE (FN687)
BREAK; BREAK(FN688)
ENDS; SWITCH end (FN689)

Command name	CASE
FN code	687
Title name	CASE
Outline	Two or more conditions are judged.

■ General description

Two or more conditions are judged.
Use it together with SWITCH, BREAK, and ENDS.

```
SWITCH V1%
CASE 1
Instruction 1
・・
Instruction I
BREAK
CASE 2
CASE 3
Instruction J
・・
Instruction K
BREAK
CASE
Instruction L
・・
Instruction M
BREAK
ENDS
```

When the value of integer variable V1% is one, it executes from instruction I to instruction K and moves the control to ENDS.
When the value of integer variable V1% is 2 or 3, it executes from Instruction J to instruction K and moves the control to ENDS.
When the value of integer variable V1% is not equal to 1, 2 and 3, it executes from Instruction L to instruction M and moves the control to ENDS.

■ Parameter

Parameter No. 1	Integer constant	It is an integer constant.

■ Example of screen display

CASE 5	FN687;CASE

See
SWITCH; SWITCH(FN686)
BREAK; BREAK(FN688)
ENDS; SWITCH end (FN689)

Command name	BREAK
FN code	688
Title name	BREAK
Outline	End the execution of an innermost instruction that encloses this with the loop or the condition structure. The control shifts to the instruction immediately after the ended instruction.

■ General description

It is possible to use it in the WHILE-ENDW structure, the FOR-NEXT structure, and the SWITCH-ENDS structure.

The control moves to the following instruction of the first ENDW instruction in the direction of the program end.

The control moves to the following instruction of the first NEXT instruction.

The control moves to the following instruction of the first ENDS instruction.

Example:

```
WHILE 1
Instruction 1
Instruction 2
V1%=V1%+1
IF V1% > 10
BREAK
ENDIF
ENDW
Instruction 3
```

It exits the WHILE-ENDW loop, and moves the control to instruction 3, when the value of V1% becomes larger than ten.

■ Parameter

None

■ Example of screen display

BREAK	FN688;BREAK

See
SWITCH; SWITCH(FN686)
CASE; CASE(FN687)
ENDS; SWITCH end(FN689)
WHILE; WHILE loop (FN663)
ENDW; WHILE loop end(FN664)
FOR; FOR(FN604)
NEXT; NEXT(FN605)

Command name	ENDS
FN code	689
Title name	SWITCH end
Outline	It is a terminator of the SWITCH-ENDS structure.

■ **General description**

It is a terminator of the SWITCH-ENDS structure.
Use it together with SWITCH, CASE, and BREAK.

```
SWITCH V1%
CASE 1
Instruction 1
. .
Instruction I
BREAK
CASE 2
CASE 3
Instruction J
. .
Instruction K
BREAK
CASE
Instruction L
. .
Instruction M
BREAK
ENDS
```

When the value of integer variable V1% is one, it executes from instruction I to instruction K and moves the control to ENDS.
When the value of integer variable V1% is 2 or 3, it executes from Instruction J to instruction K and moves the control to ENDS.
When the value of integer variable V1% is not equal to 1, 2 and 3, it executes from Instruction L to instruction M and moves the control to ENDS.

■ **Parameter**

None

■ **Example of screen display**

```
ENDS                    FN689;SWITCH end
```

See
SWITCH; SWITCH(FN686)
CASE; CASE(FN687)
BREAK; BREAK(FN688)

Command name	INCLUDE
FN code	697
Title name	Translate table included (file)
General description	The conversion rule is read from “inc file”.

■ **General description**

This command reads specified “inc file”, and adds the conversion rule to the table.

```
START_INPUT,I10
END_OUTPUT,O100
COUNT,V100%
CNT_MAX,255
```

EX) INCLDUE FILE

The conversion processing is done from the next step of INCLUDE Command according to the table of the conversion rule.
The conversion rule described in two or more files is applied by using INCLUDE Command more than once.
When reproducing, nothing is processed.

■ **Example of screen display**

```
INCLUDE [“SAMPLE.INC”]      FN670; Translate table included(file)
```

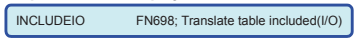
See
INCLUDEIO : Translate table included (I/O) (FN698)

Command name	INCLUDEIO
FN code	698
Title name	Translate table included (I/O)
General description	The conversion rule is read from “I/O NAME”.

■ General description

This command makes the conversion rule from “I/O NAME”, and adds it to the table. The conversion processing is done from the next step of “INCLUDEIO Command” according to the table of the conversion rule. Signal with “I/O NAME” where prohibited character is included is not added to the conversion rule. This Command can be used with “INCLUDE Command (FN697)”. The table of conversion rule can be added up to 9999.
When play backing, nothing is processed.

■ Example of screen display



See
INCLUDE : Translate table included (file) (FN697)

Command name	CLRREGWR
FN code	699
Title name	Clear register of written sts
Outline	Clear the written flag of shift register

■ General description

This command resets the written flag of the specified shift register. “SHIFTR: Shift2" (FN52) or other command, this command will reset the written flag of shift register without adversely affecting the operation of the other command being executed.

■ Example of operation

The Shift register values when CLRREGWR[R1] has been executed.

	Before LETR is executed	After LETR is executed
Request flag	0	0
Setting flag	1	0
X	110	110
Y	120	120
Z	130	130
θX	5	5
θY	6	6
θZ	7	7

Written flag of shift register set 0 when this command is executed.

Other shift register values are not changed after this command is executed.

Please see “SREQ: Shift amount request” (FN51) about how to use.

■ Parameter

Parameter No. 1	Shift register number	This is used to specify the number of the shift register in which to clear the written flag of the shift register. Setting 0 means clear written flag of all shift registers. (0 to 9)

■ Example of screen display

See
SREQ: Shift amount request (FN51)
SHIFTR: Shift2 (FN52)
RSCLR: Buffer clear (FN111)
WAITR: Wait shift value receive (FN127)

Function commands (FN codes))

Command name	DIM
FN code	801
Title name	Any variable
General description	You can define some variables as Integer, real and Array, as you like.

■General description
You can define some variables as Integer, real and Array, as you like.
And you define maximum three dimensions of array.

DIM
- Integer
- Single (Real)
- String
- Position XYZRPY
- Angle Deg or mm
- Encoder Hex.

■ **Parameters**

1st Parameter	Variables	Set Data type, Array and Variable name. Data type) Integer, Single(Real), String, Position, Angle, Encoder Array) None, One dimension, two dimension, three dimension(For only Integer, Single and String) Variable name) Initial letter must be English letter. Maximum letters are 20 which can be used some English letters, numbers and underline.

■ Example of screen display

```
DIM intData[2,2,2] As INTEGER      FN801;DIM
DIM sngData[2,2] As SINGLE         FN801;DIM
DIM strData[2] As STRING           FN801;DIM
DIM posData As POSITION            FN801;DIM
DIM angData As ANGLE               FN801;DIM
DIM encData As ENCODER             FN801;DIM
```

Function commands (FN codes))

Command name	UsrProc
FN code	802
Title name	User Procedure
General description	Define User Procedure

■General description
You can make some procedures in your programs. And you can call and use the procedures. To read and modify program easily, you should merge some similar procedures by using sub parameter of procedure.

■Example of operation

```
' AREA1 Interlock
 CallProc V1% = InToOut(V11%, V21%)   'Call User Proc
'AREA2 Interlock
 CallProc V2% = InToOut(V12%, V22%)    'Call User Proc
END

UserProc InToOut(intIN As Integer, intOUT As Integer) As Integer
  Dim intSts As Integer
  intSts = INP(intIN)              ' Get Input signal
  Output(O[intOUT], intSts)        ' Ouput signal control
  RetProc InToOut = intSts         ' Set data to return value
  EndProc
```

■ **Parameter**

1st parameter	Name	Define User procedure. Initial letter must be English letter. Maximum letters are 20 which can be used some English letters, numbers and underline
2nd parameter	Return value	Define a return value type None Integer Single(Real) String Position Angle Encoder
3rd parameter 4th 5th 6th 7th 8th 9th 10th 11th 12th	Sub parameter	Set Data type, Array, Variable name and number of array You can skip the parameter when you set none of data type.

■ Example of screen display

```
UsrProc SignalCheck(intSigNo As INTEGER) As INTEGER
```

See
ExitProc:Exit User procedure(Fn803)
EndProc:End User procedure(Fn804)
RetProc:Return User Procedure(Fn805)
CallProc:Call User procedure(Fn806)

Function commands (FN codes))

Command name	ExitProc
FN code	803
Title name	Exit User Procedure
General description	Stop procedure routine and back to source procedure

■ General description

Stop procedure routine and back to source procedure

■ Example of screen display

ExitProc	FN803;End User procedure

See

UsrProc:User procedure(Fn802)
EndProc:End User procedure(Fn804)
RetProc:Return User Procedure(Fn805)
CallProc:Call User procedure(Fn806)

Function commands (FN codes)

Command name	EndProc
Command name	804
Title name	End Procedure
General description	Finish and exit Procedure, and back to source procedure

■ General description

Finish and exit Procedure, and back to source procedure

■ Example of screen display

EndProc	FN804;End user procedure

See

UsrProc:User procedure(Fn802)
ExitProc:Exit User procedure(Fn803)
RetProc:Return User Procedure(Fn805)
CallProc:Call User procedure(Fn806)

Function commands (FN codes)

Command name	RetProc
FN code	805
Title name	Return User Procedure
General description	Set a return value of user procedure

■ General description

Set a return value of user procedure

■ Example of screen display

```
RetProc    AAAA=V1%          FN805;Return User Procedure
```

See

UsrProc:User procedure(Fn802)
ExitProc:Exit User procedure(Fn803)
EndProc:End User procedure(Fn804)
CallProc:Call User procedure(Fn806)

Function commands (FN codes)

Command name	CallProc
FN code	806
Title name	Call User Procedure
General description	Call User procedure

■ General description

Call User procedure.

■ Example of screen display

```
CallProc V1% = AAAA(V11%)          FN806;Call User Procedure
```

See

UsrProc:User procedure(Fn802)
ExitProc:Exit User procedure(Fn803)
EndProc:End User procedure(Fn804)
RetProc:Return User Procedure(Fn805)

Function commands (FN codes)

Command name	POS2POSE
FN code	809
Title name	Set Pose Variable (position)
General description	Set a position variable(As Position)to Pose variable Pn

■ General description

Set a position variable(As Position) to Pose variable Pn.

■ Parameters

1st parameter	Mechanism number	Mechanism number (0-9) 0 means all mechanism of the unit
2nd parameter	Position variable	Set position variable
3th parameter	Coordinate	Set the number of coordinates as below. 0:Base 1:Tool 2:User 3:World 4:Work
4th parameter	Coordinate number	When you select tool or user coordinate, you should select the coordinate number. It will be fixed “1”if you use other coordinates.
5rd parameter	Pose variable number	Set Pose variable number (1-9999)

■ Example of screen display

```
POS2POSE[0, posData, 0, 1, Pn]  FN809:Set pose variable(position)
```

See

DIM; Any variable (FN801)
POSE2POS; Set position variable(pose) (FN812)
ANG2POS; Set position variable(angle) (FN813)
ENC2POS; Set position variable(encoder) (FN814)
POS2ANG; Set angle variable(position) (FN816)
POS2ENC; Set encoder variable(position) (FN819)
CVTCOORDPOS; Coord. trans(position) (FN821)
GETPOS; Set position variable(pos.data) (FN822)
OPEPOS; Extraction position variable (FN826)

Function commands (FN codes)

Command name	ANG2POSE
FN code	810
Title name	Set Pose Variable (Ange)
General description	Set an Angle variable(As Angle)to Pose variable Pn

■ General description

Set Angle variable(As Angle) to Pose variable Pn.

■ Parameters

1st parameter	Mechanism number	Mechanism number (0-9) 0 means all mechanism of the unit
2nd parameter	Angle variable	Set Angle variable
3rd parameter	Pose variable number	Set Pose variable number (1-9999)

■ Example of screen display

```
ANG2POSE[0, angData, Pn]     FN810;Set Pose Variable (Angle)
```

See

DIM; Any variable (FN801)
ANG2POS; Set position variable(angle) (FN813)
POSE2ANG; Set angle variable(pose) (FN815)
POS2ANG; Set angle variable(position) (FN816)
ENC2ANG; Set angle variable(encoder) (FN817)
ANG2ENC; Set encoder variable(angle) (FN820)
GETANG; Set angle variable(pos.data) (FN823)
OPEANG; Extraction angle variable (FN827)

Function commands (FN codes)

Command name	ENC2POSE
FN code	811
Title name	Set Pose Variable (encoder)
General description	Set an encoder variable(As Encoder)to Pose variable Pn

■ General description

Set encoder variable(As Encoder) to Pose variable Pn.

■ Parameters

1st parameter	Mechanism number	Mechanism number (0-9) 0 means all mechanism of the unit
2nd parameter	Encoder variable	Set Encoder variable
3rd parameter	Pose variable number	Set Pose variable number (1-9999)

■ Example of screen display

```
ENC2POSE[0, encData, Pn]   FN811;Set Pose Variable (Encoder)
```

See

DIM; Any variable (FN801)
ENC2POS; Set position variable(encoder) (FN814)
ENC2ANG; Set angle variable(encoder) (FN817)
POSE2ENC; Set encoder variable(pose) (FN818)
POS2ENC; Set encoder variable(position) (FN819)
ANG2ENC; Set encoder variable(angle) (FN820)
GETENC; Set encoder variable(pos.data) (FN824)
OPEENC; Extraction encoder variable (FN828)

Function commands (FN codes)

Command name	POSE2POS
FN code	812
Title name	Set position Variable (pose)
General description	Set a Pose variable Pn to position variable(As position)

■ General description

Set a Pose variable Pn to position variable (As position).

■ Parameters

1st parameter	Mechanism number	Mechanism number (0-9) 0 means all mechanism of the unit
2nd parameter	Pose variable number	Set Pose variable number (1-9999)
3rd parameter	Position variable	Set position variable
4th parameter	Coordinate	Set the number of coordinates as below. 0:Base 1:Tool 2:User 3:World 4:Work
5th parameter	Coordinate number	When you select tool or user coordinate, you should select the coordinate number. It will be fixed “1”if you use other coordinates.

■ Example of screen display

```
POSE2POS[0, Pn, posData, 0, 1]  FN812:Set position variable(pose)
```

See

DIM; Any variable (FN801)
POS2POSE; Set pose variable(position) (FN809)
ANG2POS; Set position variable(angle) (FN813)
ENC2POS; Set position variable(encoder) (FN814)
POS2ANG; Set angle variable(position) (FN816)
POS2ENC; Set encoder variable(position) (FN819)
CVTCOORDPOS; Coord. trans(position) (FN821)
GETPOS; Set position variable(pos.data) (FN822)
OPEPOS; Extraction position variable (FN826)

Function commands (FN codes)

Command name	ANG2POS
FN code	813
Title name	Set position Variable (angle)
General description	Set an angle variable(As angle) to position variable(As position)

■ General description

Set an angle variable(As angle) to position variable(As position).

■ Parameters

1st parameter	Mechanism number	Mechanism number (0-9) 0 means all mechanism of the unit
2nd parameter	Angle variable	Set angle variable
3rd parameter	Position variable	Set position variable
4th parameter	Coordinate	Set the number of coordinates as below. 0:Base 1:Tool 2:User 3:World 4:Work
5th parameter	Coordinate number	When you select tool or user coordinate, you should select the coordinate number. It will be fixed “1”if you use other coordinates.

■ Example of screen display

```
ANG2POS[0, angData, posData, 0, 1] FN813:Set position variable(angle)
```

See

DIM; Any variable (FN801)
POS2POSE; Set pose variable(position) (FN809)
POSE2POS; Set position variable(pose) (FN812)
ENC2POS; Set position variable(encoder) (FN814)
POS2ANG; Set angle variable(position) (FN816)
POS2ENC; Set encoder variable(position) (FN819)
CVTCOORDPOS; Coord. trans(position) (FN821)
GETPOS; Set position variable(pos.data) (FN822)
OPEPOS; Extraction position variable (FN826)

Function commands (FN codes)

Command name	ENC2POS
FN code	814
Title name	Set position Variable (encoder)
General description	Set an encoder variable(As encoder) to position variable(As position)

■ General description

Set an encoder variable(As encoder) to position variable(As position).

■ Parameters

1st parameter	Mechanism number	Mechanism number (0-9) 0 means all mechanism of the unit
2nd parameter	encoder variable	Set encoder variable
3rd parameter	Position variable	Set position variable
4th parameter	Coordinate	Set the number of coordinates as below. 0:Base 1:Tool 2:User 3:World 4:Work
5th parameter	Coordinate number	When you select tool or user coordinate, you should select the coordinate number. It will be fixed “1”if you use other coordinates.

■ Example of screen display

```
ENC2POS[0, encData, posData, 0,1]  FN814:Set position variable(encoder)
```

See

DIM; Any variable (FN801)
POS2POSE; Set pose variable(position) (FN809)
POSE2POS; Set position variable(pose) (FN812)
ANG2POS; Set position variable(angle) (FN813)
POS2ANG; Set angle variable(position) (FN816)
POS2ENC; Set encoder variable(position) (FN819)
CVTCOORDPOS; Coord. trans(position) (FN821)
GETPOS; Set position variable(pos.data) (FN822)
OPEPOS; Extraction position variable (FN826)

Function commands (FN codes)

Command name	POSE2ANG
FN code	815
Title name	Set angle Variable (pose)
General description	Set Pose variable Pn to Angle variable(As Angle)

■ General description

Set Pose variable Pn to Angle variable(As Angle).

■ Parameters

1st parameter	Mechanism number	Mechanism number (0-9) 0 means all mechanism of the unit
2nd parameter	Pose variable number	Set Pose variable number (1-9999)
3rd parameter	Angle variable	Set Angle variable

■ Example of screen display

```
POSE2ANG[0, Pn, angData]    FN815;Set angle Variable (pose)
```

See

DIM; Any variable (FN801)
ANG2POSE; Set pose variable(angle) (FN810)
ANG2POS; Set position variable(angle) (FN813)
POS2ANG; Set angle variable(position) (FN816)
ENC2ANG; Set angle variable(encoder) (FN817)
ANG2ENC; Set encoder variable(angle) (FN820)
GETANG; Set angle variable(pos.data) (FN823)
OPEANG; Extraction angle variable (FN827)

Function commands (FN codes)

Command name	POS2ANG
FN code	816
Title name	Set angle Variable (position)
General description	Set a position variable(As position) to Angle variable(As Angle)

■ General description

Set a position variable(As position) to Angle variable(As Angle).

■ Parameters

1st parameter	Mechanism number	Mechanism number (0-9) 0 means all mechanism of the unit
2nd parameter	Position variable	Set position variable
3th parameter	Coordinate	Set the number of coordinates as below. 0:Base 1:Tool 2:User 3:World 4:Work
4th parameter	Coordinate number	When you select tool or user coordinate, you should select the coordinate number. It will be fixed "1"if you use other coordinates.
5rd parameter	Angle variable	Set Angle variable

■ Example of screen display

```
POS2ANG[0, posData, 0, 1, angData]  FN816:Set angle variable(position)
```

See

DIM; Any variable (FN801)
ANG2POSE; Set pose variable(angle) (FN810)
ANG2POS; Set position variable(angle) (FN813)
POSE2ANG; Set angle variable(pose) (FN815)
ENC2ANG; Set angle variable(encoder) (FN817)
ANG2ENC; Set encoder variable(angle) (FN820)
GETANG; Set angle variable(pos.data) (FN823)
OPEANG; Extraction angle variable (FN827)

Function commands (FN codes)

Command name	ENC2ANG
FN code	817
Title name	Set angle Variable (encoder)
General description	Set an encoder variable(As encoder) to Angle variable(As Angle)

■ General description

Set an encoder variable(As encoder) to Angle variable(As Angle).

■ Parameters

1st parameter	Mechanism number	Mechanism number (0-9) 0 means all mechanism of the unit
2nd parameter	Encoder variable	Set encoder variable
3rd parameter	Angle variable	Set Angle variable

■ Example of screen display

```
ENC2ANG[0, encData, angData]  FN817:Set angle variable(encoder)
```

See

DIM; Any variable (FN801)
ANG2POSE; Set pose variable(angle) (FN810)
ANG2POS; Set position variable(angle) (FN813)
POSE2ANG; Set angle variable(pose) (FN815)
POS2ANG; Set angle variable(position) (FN816)
ANG2ENC; Set encoder variable(angle) (FN820)
GETANG; Set angle variable(pos.data) (FN823)
OPEANG; Extraction angle variable (FN827)

Function commands (FN codes)

Command name	POSE2ENC
FN code	818
Title name	Set encoder Variable (pose)
General description	Set Pose variable Pn to encoder variable(As encoder)

■ General description

Set Pose variable Pn to encoder variable(As encoder).

■ Parameters

1st parameter	Mechanism number	Mechanism number (0-9) 0 means all mechanism of the unit
2nd parameter	Pose variable number	Set Pose variable number (1-9999)
3rd parameter	encoder variable	Set encoder variable

■ Example of screen display

```
POSE2ENC[0, Pn, encData]  FN818;Set encoder Variable (pose)
```

See

DIM; Any variable (FN801)
ENC2POSE; Set pose variable(encoder) (FN811)
ENC2POS; Set position variable(encoder) (FN814)
ENC2ANG; Set angle variable(encoder) (FN817)
POS2ENC; Set encoder variable(position) (FN819)
ANG2ENC; Set encoder variable(angle) (FN820)
GETENC; Set encoder variable(pos.data) (FN824)
OPEENC; Extraction encoder variable (FN828)

Function commands (FN codes)

Command name	POS2ENC
FN code	819
Title name	Set encoder Variable (position)
General description	Set a position variable(As position) to encoder variable(As encoder)

■ General description

Set a position variable(As position) to encoder variable(As encoder).

■ Parameters

1st parameter	Mechanism number	Mechanism number (0-9) 0 means all mechanism of the unit
2nd parameter	Position variable	Set position variable
3th parameter	Coordinate	Set the number of coordinates as below. 0:Base 1:Tool 2:User 3:World 4:Work
4th parameter	Coordinate number	When you select tool or user coordinate, you should select the coordinate number. It will be fixed “1”if you use other coordinates.
5rd parameter	Encoder variable	Set encoder variable

■ Example of screen display

```
POS2ENC[0, posData, 0, 1, encData]  FN819:Set encoder variable(position)
```

See

DIM; Any variable (FN801)
ENC2POSE; Set pose variable(encoder) (FN811)
ENC2POS; Set position variable(encoder) (FN814)
ENC2ANG; Set angle variable(encoder) (FN817)
POSE2ENC; Set encoder variable(pose) (FN818)
ANG2ENC; Set encoder variable(angle) (FN820)
GETENC; Set encoder variable(pos.data) (FN824)
OPEENC; Extraction encoder variable (FN828)

Function commands (FN codes)

Command name	ANG2ENC
FN code	820
Title name	Set encoder Variable (angle)
General description	Set an angle variable(As angle) to encoder variable(As encoder)

■ General description

Set an angle variable(As angle) to encoder variable(As encoder).

■ Parameters

1st parameter	Mechanism number	Mechanism number (0-9) 0 means all mechanism of the unit
2nd parameter	Angle variable	Set angle variable
3rd parameter	encoder variable	Set encoder variable

■ Example of screen display

```
ANG2ENC[0, angData, encData] FN820;Set encoder Variable (angle)
```

See

DIM; Any variable (FN801)
ENC2POSE; Set pose variable(encoder) (FN811)
ENC2POS; Set position variable(encoder) (FN814)
ENC2ANG; Set angle variable(encoder) (FN817)
POSE2ENC; Set encoder variable(pose) (FN818)
POS2ENC; Set encoder variable(position) (FN819)
GETENC; Set encoder variable(pos.data) (FN824)
OPEENC; Extraction encoder variable (FN828)

Function commands (FN codes)

Command name	CVTCOORDPOS
FN code	821
Title name	Coord. trans（position）
General description	Translate a position variable(As Position）to a designated coordinate.

■ General description

Translate a position variable(As Position）to a designated coordinate.

■ **Parameter**

1st parameter	Mechanism number	Mechanism number(0-9) 0 means all mechanism of the unit.
2nd parameter	Position variable1	Set an acquired position variable.
3rd parameter	Coordinate1	Set a coordinate of an acquired position variable. 0:Base 1:Tool 2:User 3:World 4:Work
4th parameter	Coordinate number1	If you select tool or user coordinate as acquired coordinate, you should select the coordinate number. It will be fixed “1”if you use other coordinates.
5th parameter	Position variable2	Set a substituted position variable.
6th parameter	Coordinate2	Set a coordinate of a substituted position variable. 0:Base 1:Tool 2:User 3:World 4:Work
7th parameter	Coordinate number1	If you select tool or user coordinate as substituted coordinate, you should select the coordinate number. It will be fixed “1”if you use other coordinates.

■ Example of screen display

```
CVTCOORDPOS[0, posData, 0, 1, posData, 0, 1]   FN821;Coord.trans(Position)
```

See

DIM:Any Variable(FN801)
POS2POSE:Set pose variable (position) (FN809)
POSE2POS: Set position variable (pose) (FN812)
ANG2POS: Set position variable (angle) (FN813)
ENC2POS:Set position variable (encoder) (FN814)
POS2ANG: Set angle variable (pos.data) (FN816)
POS2ENC:Set encoder variable (pos.data)(FN819)
GETPOS:Set position Variable (pos.data) (FN822)
OPEPOS:Extraction position variable(FN826)

Function commands (FN codes)

Command name	GETPOS
FN code	822
Title name	Set position variable (pos.data)
General description	Substitute robot position to positional variables

■ General description

The code substitutes the designated mechanism position to positional variables. You can select the mechanism position from below types.

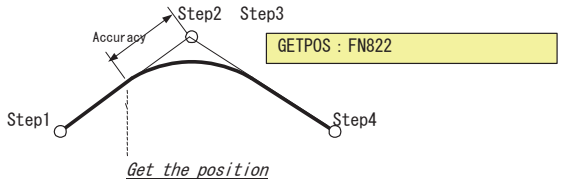
Types

Command position	Get the command position when it does.
Current position	Get the current position when it does.
Target position	Get the target command position where it is recorded. It isn’t depended on the accuracy setting of the step. If you use the shift function, the values include the shift values.

The function acts with no motion pause. Then the results are related with the accuracy of the step.

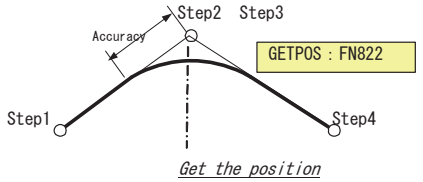
■ Example of operation

Types＝Command or Current position



Get the position when robot reach the inside range of accuracy.
If the position is pause recorded, GETPOS acts after reaching the step position.

Type＝Target position



The function gets the step position where is not related with the accuracy. If you use the shift function, the value includes the shift values.

■ **Parameters**

1st parameter	Mechanism number	Mechanism number(0-9) 0 means all mechanism of the unit.
2nd parameter	Types	0 : Command position 1 : Current position 2 : Target position
3rd parameter	Position variables	Set position variables.
4th parameter	Coordinate	Set the number of coordinates as below. 0:Base 1:Tool 2:User 3:World 4:Work
5th parameter	Coordinate number	When you select tool or user coordinate, you should select the coordinate number. It will be fixed “1”if you use other coordinates.

■ Example of screen display

GETPOS[0,0,posData,0,1] FN822;Set position variable(pos.data)

See
DIM; Any variable (FN801)
POS2POSE; Set pose variable(position) (FN809)
POSE2POS; Set position variable(pose) (FN812)
ANG2POS; Set position variable(angle) (FN813)
ENC2POS; Set position variable(encoder) (FN814)
POS2ANG; Set angle variable(position) (FN816)
POS2ENC; Set encoder variable(position) (FN819)
CVTCOORDPOS; Coord. trans(position) (FN821)
OPEPOS; Extraction position variable (FN826)

Function commands (FN codes)

Command name	GETANG
FN code	823
Title name	Set angle variable (pos.data)
General description	Substitute robot position to angle variables

■ General description

The code substitutes the designated mechanism position to angle variables. You can select the mechanism position from below types.

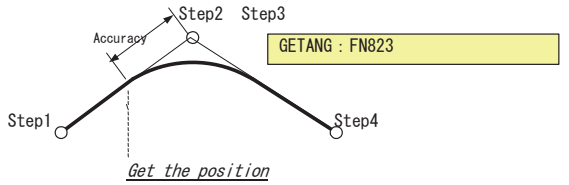
Types

Command position	Get the command position when it does.
Current position	Get the current position when it does.
Target position	Get the target command position where it is recorded. It isn’t depended on the accuracy setting of the step. If you use the shift function, the values include the shift values.

The function acts with no motion pause. Then the results are related with the accuracy of the step.

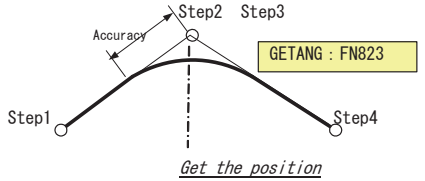
■ Example of operation

Types=Command or Current position



Get the position when robot reach the inside range of accuracy.
If the position is pause recorded, GETANG acts after reaching the step position.

Type=Target position



The function gets the step position where is not related with the accuracy. If you use the shift function, the value includes the shift values.

■ **Parameters**

1st parameter	Mechanism number	Mechanism number(0-9) 0 means all mechanism of the unit.
2nd parameter	Types	0 : Command position 1 : Current position 2 : Target position
3rd parameter	Angle variables	Set angle variables.

■ Example of screen display

GETANG[0, 0, angData] FN823;Set angle variable(pos.data)

See
DIM; Any variable (FN801)
ANG2POSE; Set pose variable(angle) (FN810)
ANG2POS; Set position variable(angle) (FN813)
POSE2ANG; Set angle variable(pose) (FN815)
POS2ANG; Set angle variable(position) (FN816)
ENC2ANG; Set angle variable(encoder) (FN817)
ANG2ENC; Set encoder variable(angle) (FN820)
OPEANG; Extraction angle variable (FN827)

Function commands (FN codes)

Command name	GETENC
FN code	824
Title name	Set encoder variable (pos.data)
General description	Substitute robot position to encoder variables

■ General description

The code substitutes the designated mechanism position to encoder variables. You can select the mechanism position from below types.

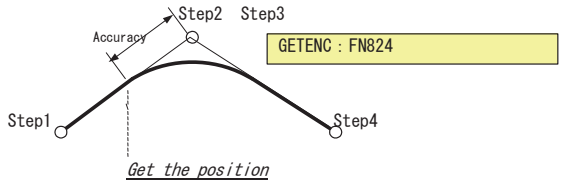
Types

Command position	Get the command position when it does.
Current position	Get the current position when it does.
Target position	Get the target command position where it is recorded. It isn't depended on the accuracy setting of the step. If you use the shift function, the values include the shift values.

The function acts with no motion pause. Then the results are related with the accuracy of the step.

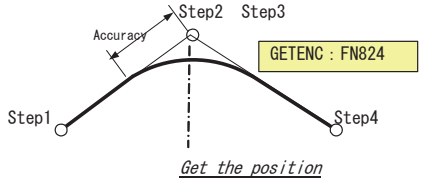
■ Example of operation

Types=Command or Current position



Get the position when robot reach the inside range of accuracy.
If the position is pause recorded, GETENC acts after reaching the step position.

Type=Target position



The function gets the step position where is not related with the accuracy. If you use the shift function, the value includes the shift values.

■ **Parameters**

1st parameter	Mechanism number	Mechanism number(0-9) 0 means all mechanism of the unit.
2nd parameter	Types	0 : Command position 1 : Current position 2 : Target position
3rd parameter	Encoder variables	Set encoder variables.

■ Example of screen display

GETENC[0, 0, encData] FN824;Set encoder variable(pos.data)

See
DIM; Any variable (FN801)
ENC2POSE; Set pose variable(encoder) (FN811)
ENC2POS; Set position variable(encoder) (FN814)
ENC2ANG; Set angle variable(encoder) (FN817)
POSE2ENC; Set encoder variable(pose) (FN818)
POS2ENC; Set encoder variable(position) (FN819)
ANG2ENC; Set encoder variable(angle) (FN820)
OPEENC; Extraction encoder variable (FN828)

Function commands (FN codes)

Command name	OPEPOSE
FN code	825
Title name	Extraction pose Variable
General description	Substitute or Extract a pose variable to a global real variable (V!) or local real variable (L!).

■ General description

Substitute or Extract a pose variable to a global real variable (V!) or local real variable (L!).

■ Parameters

1st parameter	Mechanism number	Mechanism number (0-9) 0 means all mechanism of the unit
2nd parameter	Pose variable number	Set pose variable number
3rd parameter	Real variable	Set the global real variable(V!) or Local real variable(L!), any single variable.
4th parameter	Target	Set operation target. 0 : All 1 : X 2 : Y 3 : Z 4 : R 5 : P 6 : Y
5th parameter	Operation	Select Extract or Let(Substitution) to pose variable. 0 : Extract 1 : Let

■ Example of screen display

See
DIM; Any variable (FN801)
POS2POSE; Set pose variable(position) (FN809)
ANG2POSE; Set pose variable(angle) (FN810)
ENC2POSE; Set pose variable(encoder) (FN811)
POSE2POS; Set position variable(pose) (FN812)
POSE2ANG; Set angle variable(pose) (FN815)
POSE2ENC; Set encoder variable(pose) (FN818)

Function commands (FN codes)

Command name	OPEPOS
FN code	826
Title name	Extraction position Variable
General description	Substitute or Extract a position variable to a global real variable (V!) or local real variable (L!).

■ General description

Substitute or Extract a position variable to a global real variable (V!) or local real variable (L!).

■ Parameters

1st parameter	Mechanism number	Mechanism number (0-9) 0 means all mechanism of the unit
2nd parameter	Position variable	Set position variable
3rd parameter	Real variable	Set the global real variable(V!) or Local real variable(L!), any single variable.
4th parameter	Target	Set operation target. 0 : All 1 : X 2 : Y 3 : Z 4 : R 5 : P 6 : Y 7 : Configuration
5th parameter	Operation	Select Extract or Let(Substitution) to position variable. 0 : Extract 1 : Let

■ Example of screen display

```
OPEPOS[0, posData, sgnData, 0, 0]  FN826:Extraction position variable
```

See

DIM; Any variable (FN801)
POS2POSE; Set pose variable(position) (FN809)
POSE2POS; Set position variable(pose) (FN812)
ANG2POS; Set position variable(angle) (FN813)
ENC2POS; Set position variable(encoder) (FN814)
POS2ANG; Set angle variable(position) (FN816)
POS2ENC; Set encoder variable(position) (FN819)
CVTCOORDPOS; Coord. trans(position) (FN821)
GETPOS; Set position variable(pos.data) (FN822)

Function commands (FN codes)

Command name	OPEANG
FN code	827
Title name	Extraction angle Variable
General description	Substitute or Extract an angle variable to a global real variable (V!) or local real variable (L!).

■ General description

Substitute or Extract an angle variable to a global real variable (V!) or local real variable (L!).

■ Parameters

1st parameter	Mechanism number	Mechanism number (0-9) 0 means all mechanism of the unit
2nd parameter	Angle variable	Set angle variable
3rd parameter	Real variable	Set the global real variable(V!) or Local real variable(L!), any single variable.
4th parameter	Target	Set operation target. 0 : All 1 : J1 2 : J2 3 : J3 4 : J4 5 : J5 6 : J6
5th parameter	Operation	Select Extract or Let(Substitution) to angle variable. 0 : Extract 1 : Let

■ Example of screen display

```
OPEANG[0, angData, sgnData, 0, 0]  FN827:Extraction angle variable
```

See

DIM; Any variable (FN801)
ANG2POSE; Set pose variable(angle) (FN810)
ANG2POS; Set position variable(angle) (FN813)
POSE2ANG; Set angle variable(pose) (FN815)
POS2ANG; Set angle variable(position) (FN816)
ENC2ANG; Set angle variable(encoder) (FN817)
ANG2ENC; Set encoder variable(angle) (FN820)
GETANG; Set angle variable(pos.data) (FN823)

Function commands (FN codes)

Command name	OPEENC
FN code	828
Title name	Extraction encoder Variable
General description	Substitute or Extract an encoder variable to a global integer variable (V%) or local integer variable (L%), any integer variable.

■ General description

Substitute or Extract an encoder variable to a global integer variable (V%) or integer real variable (L%), any integer variable.

■ Parameters

1st parameter	Mechanism number	Mechanism number (0-9) 0 means all mechanism of the unit
2nd parameter	Position variable	Set position variable
3rd parameter	DINT variable	Set the global integer variable(V%) or Local integer variable(L%), any integer variable.
4th parameter	Target	Set operation target. 0 : All 1 : J1 2 : J2 3 : J3 4 : J4 5 : J5 6 : J6
5th parameter	Operation	Select Extract or Let(Substitution) to encoder variable. 0 : Extract 1 : Let

■ Example of screen display

```
OPEENC[0, encData, intData, 0, 0]  FN828:Extraction encoder variable
```

See

DIM; Any variable (FN801)
ENC2POSE; Set pose variable(encoder) (FN811)
ENC2POS; Set position variable(encoder) (FN814)
ENC2ANG; Set angle variable(encoder) (FN817)
POSE2ENC; Set encoder variable(pose) (FN818)
POS2ENC; Set encoder variable(position) (FN819)
ANG2ENC; Set encoder variable(angle) (FN820)
GETENC; Set encoder variable(pos.data) (FN824)

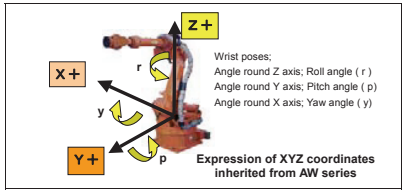
Move commands

Command name	MOVE
FN code	-
Title name	(move command)
General description	Move robot to the position expressed by the AW coordinate system

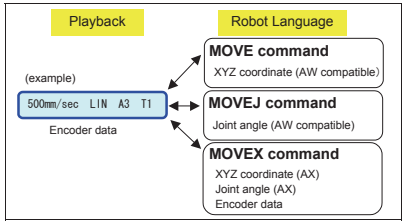
■ **General description**

This is the move command in which robot position is expressed by XYZ value of tool top and rpy (roll, pitch and yaw) angle of wrist attitude. To keep upper compatibility with NACHI-Fujikoshi AW controller, coordinate system is same as AW (front is Y). X and Y axis direction is different from FD coordinate system (= world wide standard).
Wrist attitude is fixed by the rotation on the order of r -> p -> y.
New functionalities developed in FD controller such as synchronize control, Smooth and others can not be described by this command.

<Format> MOVE Interpolation, Pose, Speed (or Time), Accuracy, Tool
<Example> MOVE L,P1,S=500,A=3



There are 3 move commands. MOVE and MOVEJ is prepared to keep upper compatibility with NACHI-Fujikoshi AW controller. Please use MOVEX as usual.

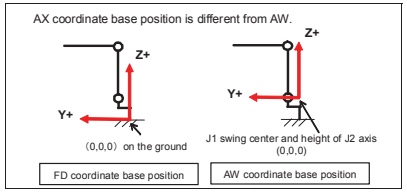


■ **Parameter**

Interpolation	This is to define the interpolation to move the desired position, by P,L,LE,C or CE description. P Joint move (interpolation OFF) L Linear interpolation LE Stationary tool linear interpolation C Circular interpolation CE Stationary tool circular interpolation
Pose	This is to define the desired position. Following list shows its variety to describe the robot pose. XYZ coordinate system is same as NACHI-Fujikoshi AW. (X,Y,Z,r,p,y) Pose constant Pn (or P[n]) Pose variable Pn + (X,Y,Z,r,p,y) Pose variable + Shift constant Pn + Rn Pose variable + Shift variable P* + (X,Y,Z,r,p,y) Current position + Shift constant P* + Rn Current position + Shift variable R,p and y angle is written in degree. If there are some aux. axes, maximum two axis angle data is added following the (X,Y,Z,r,p,y) statement. (X,Y,Z,r,p,y,J7) (X,Y,Z,r,p,y,J7,J8) J7 and J8 angle is written in degree.
Speed (or Time)	This is to define the speed or time to achieve the desired position. (1[mm/s]～5000[mm/s] or 0.1[s]～100[s]) S for TCP speed, T for time. S=100 TCP line speed is 100mm/sec. T=5 Time to achieve next position is 5 seconds.

	S=V1% Speed is designated by global variable V1%. T=L![1] Time is designated by local variable L![1]. Variables can be used as followed. Vn% (or V%[n]) global variable (integer) Vn! (or V![n]) global variable (float) Ln% (or L%[n]) local variable (integer) Ln! (or L![n]) local variable (float)
Accuracy	This is to define the accuracy number. If omitted, accuracy defined in previous step is use. When in-position check is necessary, add “P”. (1 – 8, 1P – 8P) A=1 Accuracy number is 1
Tool	This is to define the tool number. If omitted, tool defined in previous step is use. (1 - 32) H=1 Tool number is 1.

See
MOVEJ; (Move command)
MOVEX; (Move command)



Move commands

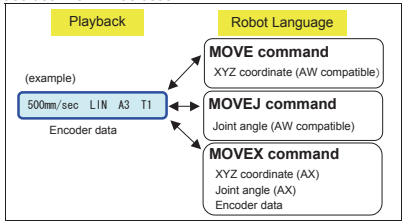
Command name	MOVEJ
FN code	–
Title name	(move command)
General description	Move robot to the position expressed by the AW axis angle

■ General description

This is the move command in which robot position is expressed by axis angle. To keep upper compatibility with NACHI-Fujikoshi AW controller, J3 value is the angle from ground level. Except J3 angle, this is same as MOVEX-J command. New functionalities developed in FD controller such as synchronize control, Smooth and others can not be described by this command.

<Format> MOVEJ Interpolation, Pose, Speed (or Time), Accuracy, Tool
<Example> MOVEJ L,(0,90,0,0,0,0),S=500,A=3

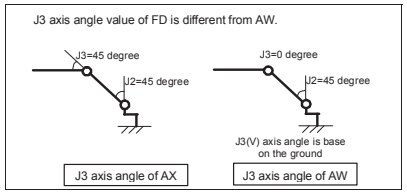
There are 3 move commands. MOVE and MOVEJ is prepared to keep upper compatibility with NACHI-Fujikoshi AW controller. Please use MOVEX as usual.



■ Parameter

Interpolation	This is to define the interpolation to move the desired position, by P,L,LE,C or CE description P Joint move (interpolation OFF) L Linear interpolation LE Stationary tool linear interpolation C Circular interpolation CE Stationary tool circular interpolation
Pose	This is to define the desired position. Axis angle is degree. (J1,J2,J3,J4,J5,J6) Pose constant If there are some aux. axes, maximum two axis angle data can be added. (J1,J2,J3,J4,J5,J6,J7,J8)
Speed (or Time)	This is to define the speed or time to achieve the desired position. (1[mm/s]～5000[mm/s] or 0.1[s]～100[s]) S for TCP speed, T for time. S=100 TCP line speed is 100mm/sec. T=5 Time to achieve next position is 5 seconds. S=V1% Speed is designated by global variable V1%. T=L![1] Time is designated by local variable L![1]. Variables can be used as followed. Vn% (or V%[n]) global variable (integer) Vn! (or V![n]) global variable (float) Ln% (or L%[n]) local variable (integer) Ln! (or L![n]) local variable (float)
Accuracy	This is to define the accuracy number. If omitted, accuracy defined in previous step is use. When in-position check is necessary, add “P”. (1 – 8, 1P – 8P) A=1 Accuracy number is 1
Tool	This is to define the tool number. If omitted, tool defined in previous step is use. (1 - 32) H=1 Tool number is 1.

See
MOVE; (Move command)
MOVEX; (Move command)



Move commands

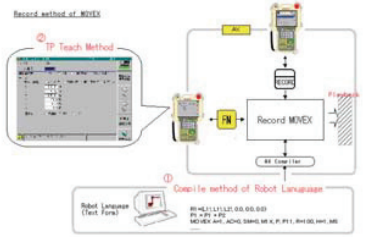
Command name	MOVEX
FN code	645
Title name	(move command)
General description	Move robot.

■ General description

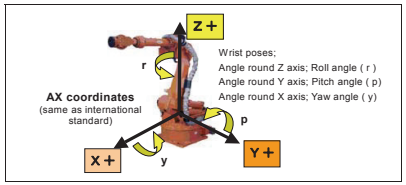
This is the move command.
MOVEX command supports following three statement of robot position.
- XYZ value of tool top and rpy (roll, pitch and yaw) angle of wrist attitude.
- Axis angle
- Axis encoder data

XYZ coordinate system is same as international standard (front = X).
Wrist attitude is fixed by the rotation on the order of r -> p -> y.
New functionalities developed in FD controller such as synchronize control, Smooth and others can be described by this command.

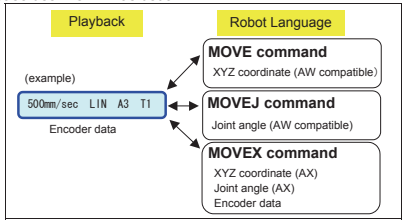
To record the MOVEX command, there are two manners provided: Robot language compile system (1) and TP teaching system (2). For the details of TP teaching system, see "■How to teach MOVEX on TP" in this document.



<Format>
MOVEX Accuracy, Acceleration, Smooth, Conveyor, Synchronize
Mechanism, Interpolation, Pose, Configuration, Speed, Tool
Mechanism, Interpolation, Pose, Speed, Tool
Mechanism, Interpolation, Pose, Speed, Tool
…
<Example>
MOVEX A=1,AC=1,SM=1,HM=1
M1W,P,P1,S=100,H=1,
M2X,L,P2,S=100,H=2,MS
M3J,L,(0,0,0,0,0,0),S=100,H=3

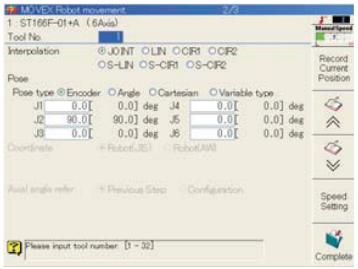


There are 3 move commands. MOVE and MOVEJ is prepared to keep upper compatibility with NACHI-Fujikoshi AW controller. Please use MOVEX as usual.



■ MOVEX Teach Method

(1) Select FN645 “Record MOVEX”.



(2) Set the required parameters (described in the following).
(3) Press F12 <Complete> to record MOVEX.

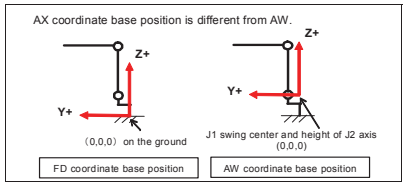
■ **Parameter**

∗ The setting example of parameter is indicated as “A” for the robot language compile system and “B” for the TP teaching system. (Without any specific indication of “B”, the description indicates the robot language compile system.)

Accuracy	This is to define the accuracy number. If omitted, accuracy defined in previous step is use. When in-position check is necessary, add “P”. (1 – 8, 1P – 8P) A : A=1 Accuracy number is 1 B : Input 1 in [Accuracy number]. Accuracy number is 1
Acceleration	This is to define the acceleration. Bigger value is smaller acceleration. No description is treated das 0. (0 - 3) A : AC=1 Acceleration number is 1 B : Input 1 in [Acceleration]. Acceleration number is 1
Smooth	This is to define the smoothness. Bigger value is larger smoothness. No description is treated das 0. (0 - 3) A : SM=1 Smoothness number is 1 B : Input 1 in [Smoothness]. Smoothness number is 1
Fine motion	Enable/disable of fine motion is switched. To make fine motion enable, “F” is applied. When nothing is specified, fine motion becomes disable. A : F Fine motion is enalbed. B : Select “F” in [Fine motion] Fine motion is enabled.
Synchronize	This is to define the master robot of synchronize control. This is significant only when plural mechanism exits. If not described, synchronizing is executed as same start control. A : HM Mechanism 1 is master B : Select “Synchronize” in [Synchronize]. Mechanism 1 is master
Mechanism	This is to define pose of each mechanism. (varies from 1 to 9) “M”, “mechanism number” and “pose definition”. A : M2X “M2” is mechanism 2, “X” means that pose is described by FD coordinate system (international standard) A : M3J “M3” is mechanism 3, “J” means that pose is described by axis angle. A : M4E “M4” is mechanism 4, “E” means that pose is described by axis encoder data. B : Select the proper mechanism and pose definition in [Pose type]. Encoder The pose is recorded by axis encoder data. Please input the axis angle to the “Edit Box”. Angle The pose is recorded by axis angle data. Please input the axis angle to the “Edit Box”. Cartesian The pose of the robot is recorded by the Cartesian coordinate value. Variable type The pose of the robot is recorded by using the value of pose variable. Before the first “M”, all statement is treated as for the mechanism 1. “Pose definition”. can be omitted, then treated as “X”. Pose definition “X” is significant only when its mechanism is the manipulator.
Interpolation	This is to define the interpolation to move the desired position, by P,L,LE,C1,C1E,C2 or C2E description. A : P Joint move (interpolation OFF) L Linear interpolation LE Stationary tool linear interpolation C1 Circular interpolation (start or middle of circle) C1E Stationary tool circular interpolation (start or middle of circle) C2 Circular interpolation (end of circle) C2E Stationary tool circular interpolation (end of circle) B : Select any of the following in [Interpolation]. JOINT Joint move (interpolation OFF) LIN Linear interpolation S-LIN Stationary tool linear interpolation CIR1 Circular interpolation (start) S-CIR1 Stationary tool circular interpolation (start) CIR2 Circular interpolation (end) S-CIR2 Stationary tool circular interpolation (end)
Pose	This is to define the desired position. Following list shows its variety to describe the robot pose. $$ In case that pose definition is “X” Pose is expressed by XYZ coordinate system same as international standard (front = X). A : (X,Y,Z,r,p,y) Pose constant B : Select “Cartesian” in [Pose type]. Pose constant A : Pn (or P[n]) Pose variable B : Select “Variable type” in [Pose type]. Pose variable A : Pn + (X,Y,Z,r,p,y) Pose variable + Shift constant A : Pn + Rn Pose variable + Shift variable A : P* + (X,Y,Z,r,p,y) Current position + Shift constant A : P* + Rn Current position + Shift variable R,p and y angle is written in degree. $$ In case that pose definition is “J” Pose is expressed by axis angle (degree). A : (J1,J2,J3,J4,J5,J6) Pose constant B : Select “Angle” in [Pose type]. Pose constant J3 is not the angle from ground level. No expression for J7 and after. They are next mechanism. $$ In case that pose definition is “E” Pose is expressed by axis encoder data (HEX). A : (E1,E2,E3,E4,E5,E6) Pose constant B : Select “Encoder” in [Pose type]. Pose constant
Configuration	This is to define the supplement information to fix robot pose. This is significant only when pose is expressed by XYZ coordinate system. A : CONF=ijkl i ; 1:FLIP / 1:NONFLIP (wrist flip / no flip) j ; 0:ABOVE / 1:BELOW (upper elbow / lower elbow) k ; 0:LEFTY / 1:RIGHTY / 2:FRONT / 3:REAR(*) (left hand / right hand/ front/ rear) l ; flange axis rotation direction specification (0:-180 [deg]~180[deg] /1:0~360[deg] /2:-360~0[deg]) B : Select “Configuration” in [Axial angle refer], and then the configuration type in each axis. [J1]: “Left”, “Right”, “Less 90deg”, “Over 90deg” (*) [J3]: “On elbow”, “Under elbow” [J5]: “Flip”, “Non Flip” (*) About “Less 90deg” and “Over90deg” These are available only when [16 Configuration J1 Extend] has been set [Enable] by proceeding to [Constant] - [5 Operation Constants] - [1 Operation condition]. When no description of configuration, robot pose is determined as the nearest pose with previous pose.
Speed (or Time)	This is to define the speed or time to achieve the desired position. (1[mm/s]~5000[mm/s] or 0.1[s]~100[s]) S for TCP speed, T for time. A : S=100 TCP line speed is 100mm/sec. A : T=5 Time to achieve next position is 5 seconds. A : D=0.1 Wrist angle speed 0.1 degree/sec A : R=50 50% ratio compared with maximum power A : S=V1% Speed is designated by global variable V1%. A : T=L![1] Time is designated by local variable L![1].

	A : D=L![1] Angle speed is designated by local variable L![1]. A : R=V1% Ratio is designated by global variable V1%. B : Press [F11 Speed Setting] to specify the speed. Variables can be used as followed. Vn% (or V%[n]) global variable (integer) Vn! (or V![n]) global variable (float) Ln% (or L%[n]) local variable (integer) Ln! (or L![n]) local variable (float)
Tool	This is to define the tool number. If omitted, tool defined in previous step is use. (1 - 32) A : H=1 Tool number is 1. B : Input 1 in [Tool number]. Tool number is 1.
Speed master definition	This is to define the speed master robot. This is significant only when synchronize control enabled. Plural mechanism can be speed master. A : MS speed master robot. B : Input the number in [Speed base mechanism]. speed master robot.

See
MOVE; (Move command)
MOVEJ; (Move command)



NACHI
NACHI-FUJIKOSHI CORP.

http://www.nachi-fujikoshi.co.jp/

JAPAN MAIN OFFICE	Phone: +81-3-5568-5245	Fax: +81-3-5568-5236	Shiodome Sumitomo Bldg. 17F, 1-9-2 Higashi-Shinbashi Minato-ku, TOKYO, 105-0021 JAPAN

NACHI NORTH AMERICA — ***http://www.nachirobotics.com/***

North America Headquarters	Phone: 248-305-6545	Fax: 248-305-6542	22285 Roethel Drive, Novi, Michigan 48375 U.S.A.
Greenville Service Office	Use 248-305-6545	Use 248-305-6542	South Carolina, U.S.A.
San Antonio Service Office	Use 248-305-6545	Use 248-305-6542	Texas, U.S.A.
Kentucky Branch Office	Phone: 502-695-4816	Fax: 502-695-4818	116 Collision Center Drive, Suite A, Frankfort, KY 40601 U.S.A
Training Office	Phone: 248-334-8250	Fax: 248-334-8270	22213 Roethel Drive, Novi, Michigan 48375 U.S.A.
Toronto Branch Office	Phone: 905-760-9542	Fax: 905-760-9477	89 Courtland Avenue, Unit 2, Vaughan, Ontario L4K3T4 CANADA
Mexico Branch Office	Phone : +52-555312-6556	Fax: +52-55-5312-7248	Urbina # 54, Parque Industrial Naucalpan, Naucalpan de Juarez, 53370, Estado de México, MEXICO
Saltillo Service Office	Phone : +52-844416-8053	Fax: +52-844416-8053	Canada 544 Privada Luxemburgo C. P. 25230, Saltillo, Coahuila, MEXICO

NACHI ROBOTIC EUROPE

Germany — ***http://www.nachi.de/***

Nachi Europe GmbH	Phone: +49-(0)2151-65046-0	Fax: +49-(0)2151-65046-90	Bischofstrasse 99, 47809, Krefeld,GERMANY

United Kingdom — ***http://www.nachi.co.uk/***

Nachi U.K. LTD.	Phone: +44-(0)121-250-1895	Fax: +44-(0)121-250-1899	Unit 7, Junction Six Industrial Estate, Electric Avenue, Birmingham B6 7JJ, U.K.

Czech Republic

Nachi Europe	Phone / Fax : +420-321-710-200	Mostni 73, Kolin 4, 28002 CZECH

NACHI ROBOTIC ASIA

Korea — ***http://www.nachi-korea.co.kr/***

Korea	Phone: +82-(0)2-469-2254	Fax: +82-(0)2-469-2264	2F Dongsan Bldg. 276-4, Sungsu 2GA-3DONG, Sungdong-ku, Seoul 133-123, KOREA

Robot Division
1-1-1, FUJIKOSHIHONMACHI, TOYAMA CITY, JAPAN 930-8511
Phone +81-76-423-5137
Fax +81-76-493-5252