

**F9000N, F8000N AND
F2004N SERIES
ROBOTS
OPERATING MANUAL**

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

SECTION 1: SAFETY.....7

- 1. CE CERTIFICATION REQUIREMENTS8
- 2. SAFETY RULES.....8
- 3. GENERAL CONDITIONS FOR SAFETY9
- 4. SAFETY DURING OPERATION.....9

SECTION 2: SYSTEM INSTALLATION11

- 1. CONTROLLER.....12
 - 1.1. Controller Specifications.....12
 - 1.2. Controller Structure13
 - 1.3. Front View.....14
 - 1.4. Rear View.....16
 - 1.5. Connection Cable.....17
 - 1.6. User I/O Cable.....17
 - 1.7. System I/O Cable.....18
- 2. INSTALLATION OF THE PRODUCT18
 - 2.1. Initial Considerations.....18
 - 2.2. Connecting the Controller to the Manipulator.....19
 - 2.3. Turning ON the Controller for the first time.....21

SECTION 3: TEACHING OVERVIEW.....22

- 1. TEACHING OVERVIEW23
- 2. USING THE TEACH PENDANT25
 - 2.1. Key Selection.....25
 - 2.2. Key Assignments.....26
 - 2.3. Navigation Menu.....27
 - 2.4. Jogging.....28
 - 2.5. Data Entry.....28
 - 2.6. LED Panel.....29
- 3. TEACH BOX KEY ASSIGNMENTS.....29
- 4. POINT TYPE FUNCTIONS SUMMARY.....32
 - 4.1. Point Menu.....32
 - 4.2. Setup Menu.....34
 - 4.3. Condition Menu.....36
 - 4.4. Menu 1.....38
 - 4.5. Menu 2.....39

SECTION 4: PROGRAMMING EXAMPLE.....41

- 1. PROGRAMMING EXAMPLE.....42
- 2. EDITING A PROGRAM47
 - 2.1. Changing a Point XYZ location.....47
 - 2.2. Insert / Delete an Instruction.....48
- 3. CHANGING THE PROGRAM NUMBER48
- 4. CHANGING FROM TEACH MODE TO RUN MODE.....48
- 5. AUTO MODE AND STEP MODE IN RUN MODE.....49

SECTION 5: DISPENSING PARAMETERS.....50

1.	DISPENSE END SETUP	51
2.	Z CLEARANCE.....	52
3.	LINE DISPENSE SETUP.....	53
4.	RETRACT	54
5.	ADJUST ORIGIN	58
6.	AUTO PURGE	59
SECTION 6: POINT TYPE & FUNCTION REFERENCE.....		61
1.	POINT MENU.....	62
1.1.	Dispense Dot.....	62
1.2.	Line Start.....	62
1.3.	Line Passing.....	63
1.4.	Arc Point.....	63
1.5.	Circle.....	63
1.6.	Center.....	63
1.7.	Line End.....	65
1.8.	Dummy.....	66
1.9.	End Program.....	66
1.10.	Dispense ON / OFF.....	66
1.11.	Home Point.....	67
1.12.	Wait Point	67
1.13.	Stop Point.....	67
1.14.	Brush Area	68
1.15.	If.....	70
1.16.	Output.....	70
1.17.	Input.....	70
1.18.	Pulse.....	70
1.19.	Point.....	70
1.20.	Load Point.....	71
2.	CONDITION MENU	72
2.1.	Goto Address.....	72
2.2.	Step & Repeat Y.....	72
2.3.	Step & Repeat X.....	77
2.4.	Call Subroutine	77
2.5.	Call Program	78
2.6.	Loop Address.....	79
2.7.	Label.....	79
2.8.	Arm.....	79
2.9.	FixR.....	80
2.10.	Calculation.....	80
2.11.	Jmov/Lmov	85
2.12.	IncJ / IncL.....	85
2.13.	Offset	85
2.14.	PALLET.....	86
2.15.	Pattern / Pattern End	89
2.16.	Xmov.....	91
2.17.	COMM	91
3.	SETUP MENU	92
3.1.	Line Speed.....	92
3.2.	Line Dispense Setup	92
3.3.	Point Dispense Setup.....	92
3.4.	Dispense End Setup.....	92
3.5.	Z Clearance.....	93
3.6.	X/Y Move Speed.....	93
3.7.	Z Move Speed.....	93
3.8.	Home Position.....	93
3.9.	Retract.....	94
3.10.	Auto Purge	94

3.11.	Adjust Point # 1	94
3.12.	Adjust Point # 2	94
3.13.	Round	94
3.14.	Z Lift	96
3.15.	Move Cond	97
3.16.	Disp Cond	97
4.	MENU 1	98
4.1.	Program Name	98
4.2.	Z Axis Limit	98
4.3.	Initial Output	98
4.4.	Cycle Counter	99
4.5.	Set Password	100
4.6.	Jog Speed	100
4.7.	Run Mode	100
4.8.	Adjust Position	100
4.9.	Parameter	101
4.10.	Auto Tool Setting	103
4.11.	Resume	105
4.12.	Origin Searching	105
4.13.	Hour Meter	105
4.14.	PLC File Edit	106
4.15.	I/O Monitor	106
4.16.	Set Variable	106
4.17.	EMG Mode	106
5.	MENU 2	107
5.1.	Point Utility	107
5.1.1.	MDI Mode	107
5.1.2.	Numerical Move	107
5.1.3.	Save Temp Point	107
5.1.4.	Retrieve Temp Point	107
5.2.	Group Utility	107
5.2.1.	Group Edit	107
5.2.1.1.	Copy	108
5.2.1.2.	Delete	109
5.2.1.3.	Move	110
5.2.1.4.	Line SP (Line Speed)	111
5.2.1.5.	Dispen.TM (Dispense Time)	112
5.2.1.6.	Offset	113
5.2.2.	Expand Step & Repeat	114
5.2.3.	Relocate Data	115
5.2.4.	Adjust Origin	116
5.3.	Program Utility	118
5.3.1.	Copy Program / Delete Program	118
5.3.2.	Auto Offset	118
5.4.	Memory Utility	119
5.4.1.	Delete Memory	119
5.4.2.	Program BackUp	119
5.4.3.	Delete Memory	119
SECTION 7: PLC		120
1.	CREATION OF A PLC FILE	121
2.	RUNNING A PLC FILE	124
3.	PLC PROGRAM EXAMPLES	125
3.1.	LD / LDNOT /OUT	125
3.2.	AND / ANDNOT	126
3.3.	OR / ORNOT	127
3.4.	ANDBK	128
3.5.	ORBK	129
3.6.	MC/MCR	130
3.7.	SET/RESET	131
3.8.	PULS/PULSNOT	132

3.9.	<i>T (Timer)</i>	133
3.10.	<i>C (Counter)</i>	134
3.11.	<i>MOV / DMOV</i>	135
3.12.	<i>ADD / DADD</i>	136
3.13.	<i>SUB / DSUB</i>	136
3.14.	<i>MUL/DMUL</i>	137
3.15.	<i>DIV / DDIV</i>	137
4.	SYSTEM CONTROL CONTACT	138
4.1.	<i>System Control Input</i>	138
4.2.	<i>System Control Output</i>	139
SECTION 8: ERROR LIST		140
1.	ERROR LIST FOR HARDWARE	141
2.	ERROR LIST FOR PROGRAM	149
2.1.	<i>Need LINE START</i>	149
2.2.	<i>Need LINE END</i>	149
2.3.	<i>Need Step & Repeat</i>	149
2.4.	<i>PROGRAM END ERROR</i>	149
2.5.	<i>LABL Not Exist</i>	149
SECTION 9: APPENDIX		150
1.	APPENDIX A: USER I/O BOARD	151
2.	APPENDIX B: SYSTEM I/O BOARD	161
3.	APPENDIX C: EQUIPMENT (MACHINE) CONNECTION	169
4.	APPENDIX D : MACHINE DIMENSIONS	180
4.1.	<i>F8100N</i>	180
4.2.	<i>F8104N</i>	181
4.3.	<i>F8800N</i>	182
4.4.	<i>F8804N</i>	183
4.5.	<i>F9300N</i>	184
4.6.	<i>F9304N</i>	185
4.7.	<i>F9600N</i>	186
4.8.	<i>F9604N</i>	187
4.9.	<i>F9800N</i>	188
4.10.	<i>F9804N</i>	189
4.11.	<i>F2004N</i>	190
5.	APPENDIX E : TABLE DIMENSIONS	191
5.1.	<i>F9300N/F9304N</i>	191
5.2.	<i>F9600N/F9604N</i>	192
5.3.	<i>F9800N/F9804N</i>	193
6.	APPENDIX F: COORDINATES (AXES) OF ROBOTS	194
6.1.	<i>Manipulator of F9000N Series</i>	194
6.2.	<i>Manipulator of F2004N Series</i>	195
6.2.1.	<i>Joint Coordinates</i>	195
6.2.2.	<i>Cartesian Coordinates</i>	196

Section 1: Safety

F8000N/F9000N Gantry and SCARA TMB200R/300R robots use the same hardware controller, programming software and teach pendant. Therefore, the instructions and programming information are presented in this Operating Manual for both of them.

1. CE Certification Requirements

1. In order to meet the safety requirements of the CE directives (applicable in the countries of European Union) the robots must be placed in an enclosure which can be supplied by the Fisnar Inc. distributors.
2. The enclosure must prevent the access to the moving parts except through the enclosure door.
3. The enclosure door switch must be connected to the door switch connector on the robot I/O cable.

2. Safety Rules

1. In order to use a robot in safety conditions, the user should prepare the **safety work regulations** under the careful consideration of line layout and side-line establishments where the robot is installed, and the operator must keep strictly to the safety work regulations to prevent accidents. Also, standard operation procedure about the robot must be written-up for safety, and appropriate measures for safety operation must be taken, such as safety training of the operators.
2. Teaching operation and maintenance procedure of the robot should be set according to the standards of the Industrial Safety and Health Law and Industrial Safety Regulations.
3. The user should prepare the safety operation regulations of the overall system and abide by them.
4. In order to secure the robot's safety, please observe the general provisions related to the safety operation of an industrial robot.
5. Prepare a safety management system, such as appointing operators responsible for the safe operation of robot or deciding on safety supervisors, and give them thorough safety training.

3. *General Conditions for Safety*

1. Please use robot within the standard requirements (such as payload, speed, operational range, user environment) as stated in the specification. Make sure specifically that the single phase is not over AC 230 V / 15A before turning the power on.
2. Make sure the operator has read the operation manual and other materials thoroughly, so that all problems can be solved, thus minimizing damage during operation.
3. Do not attach or detach the power cord while the power switch of the controller is turned ON.
4. Do not drop the teach pendant.
5. Install the robot firmly so that it will not be shaken.
6. Install a safety fence around the robot's working area for a safe work environment.
7. Check electrical connections before turning on the electrical power of the controller. The machine may not work properly due to incorrect connection of electrical wires.
8. Install FG (Frame Ground) in order to prevent electric shocks.

4. *Safety During Operation*

1. To start operation of the robot, turn on the power switch of the robot controller device. Please read the following conditions for safety during operation.
2. Before starting the operation, make sure that there is no person or obstacle in the robot's working area.
3. Be ready to push the emergency stop switch if the robot does not function normally.
4. Before starting a repetitive operation, make sure that nobody and no obstacle is in the robot's working area enclosed by the safety fence.

5. When several people are working together simultaneously, check for mutual safety, especially during the power is ON/OFF and during the manual operation.

6. During maintenance and inspection of the robot, pull out the power plug of the controller.

Section 2: System Installation

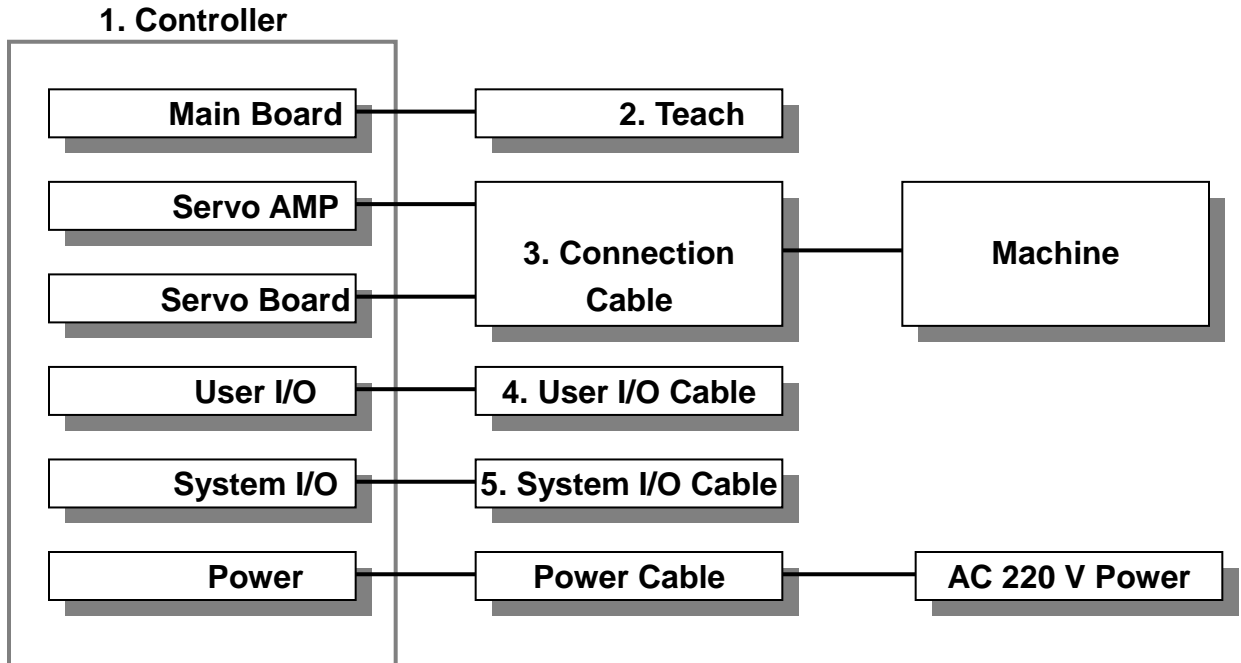
1. Controller

1.1. Controller Specifications

Item	Type	Specification
Axis Control	Number of Control axes	Min. 1 axis, Max. 4 axes
	Control Method	AC Full Digital Servo
	Positioning Unit	Cartesian Coordinates: mm Axis Coordinates : Degree (Deg)
	Motion Method	PTP motion (Joint interpolation), CP motion, linear motion
	Control Interpolation	Linear interpolation, circular interpolation
	Speed	1 ~ 100%
	Encoder Type	Incremental (9 wires) / Absolute Type usable (Tamakawa Motor Type)
	Position Precision Degree	Within $\pm 1/4$ Encoder Pulse
General Condition	Size	368 mm x 302 mm x 140 mm
	Weight	Max 10kg
	Max. Power Capacity	4 axes Total 2.4 kw
	Operation Method	Teach Pendant / Front Panel
	Program Volume	3,000 Steps / Program 1,000 Steps x 20 programs
	External Input/ Output	General 32/32 System 24/6
	External Communication	RS-232C
	Input Power	Single Phase AC 220V 50/60Hz, $\pm 10\sim 15\%$
	Running Temperature and Humidity	0 ~ 45 Degree C; 20 ~ 80%RH
	Components	Controller, Connection cables, Teach Pendant, User I/O Cable, System I/O Cable, Power Cable, Dispenser cable, I/O Port Connector, External Control Connector, Enclosure (for units delivered to E.U. countries – not included), Spacers (4-axis robots only)

Abnormality	Abnormalities	Over Current, Over Heat, Following Error, Encoder error, Board malfunction, Over Speed, Position deviation abnormality Brake error, etc.
-------------	---------------	--

1.2. Controller Structure

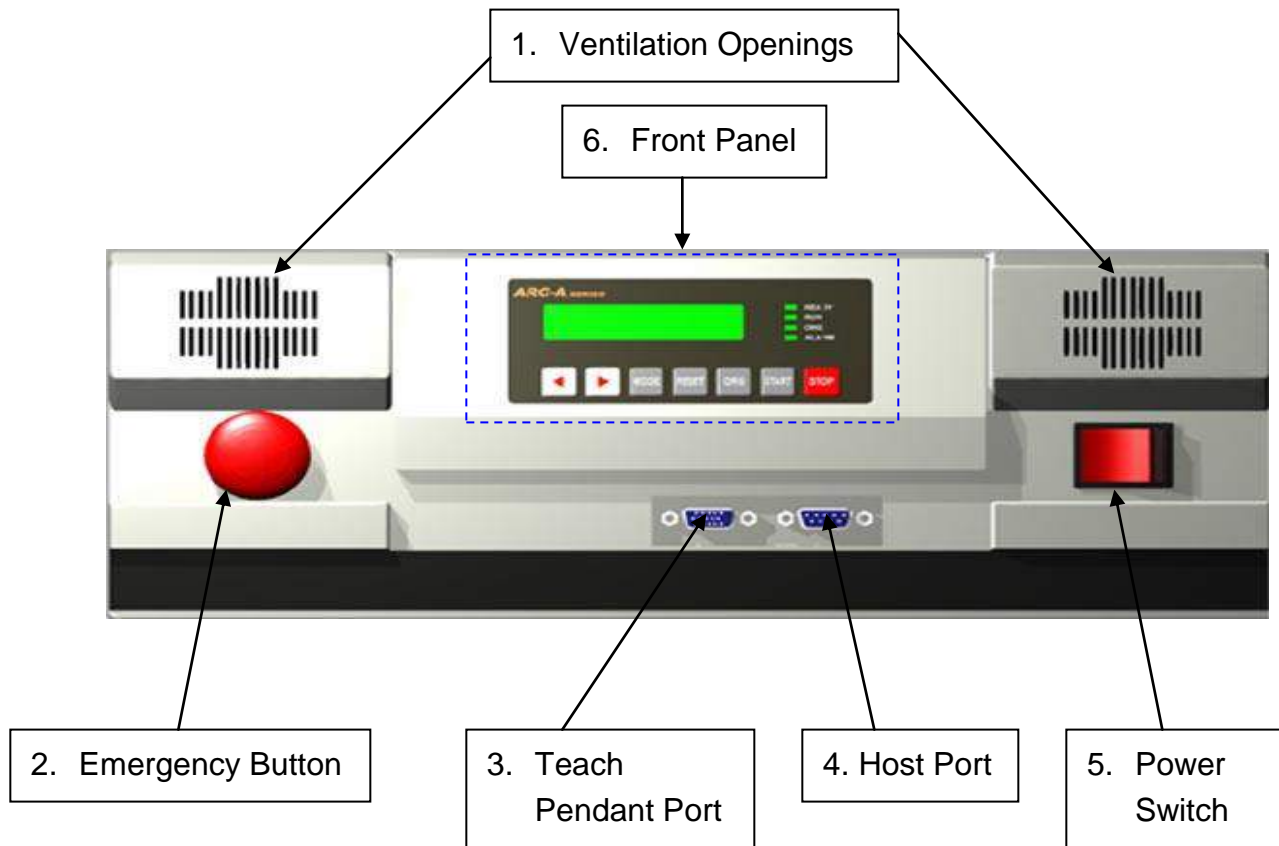


- 1. **Controller** : Consists of different types of boards and AMP, as shown above. P/N: F9000N-C-LF OR F9000NR-C-LF
- 2. **Teach Pendant** : Creates a work program and changes the configurations of the system. P/N: F9000N-T
- 3. **Connection Cable** : Connects each channel of the controller and each axis of the manipulator.
- 4. **User I/O Cable** : Connects the user I/O board and other equipment.
- 5. **System I/O Cable** : Connects the system I/O board and other equipment.

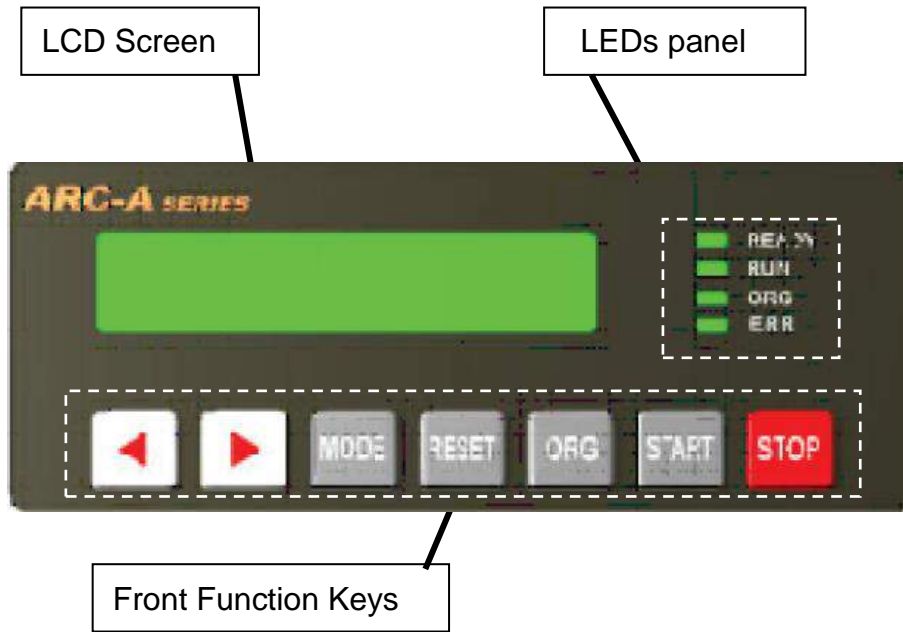
According to the user's needs, the specifications of the Controller or the structure of the machine can be changed.

The aspects and functions of each part of the Controller are as follows:

1.3. Front View



1. **Ventilation openings:** Allow the entrance of cold air into the unit to cool the inner heat produced by the controller during operation. Do not block these openings.
2. **Emergency Stop Button:** Used when the controller is needed to be stopped immediately.
3. **Teach pendant (T-Box) port:** Used to connect the Teach Pendant. Be careful not to connect the HOST cable.
4. **Host port (for RS-232C communication):** Used for MMI Serial communication.
5. **Power switch:** Used to turn on or off the power of the controller. (The light comes on when the power is ON.)
6. **Front Panel** is used to operate the controller without a teach pendant.



- **LCD Screen** is a 2 line x 16 letter screen and indicates the current state and the program being operated or selected, as well as the current step.
- **LEDs Panel** shows the current status.



READY LED : Light is turned on when the robot is ready to start.

RUN LED : Light is turned on while the robot is operating.

ORG LED : Light is turned on after an origin-searching operation is performed.

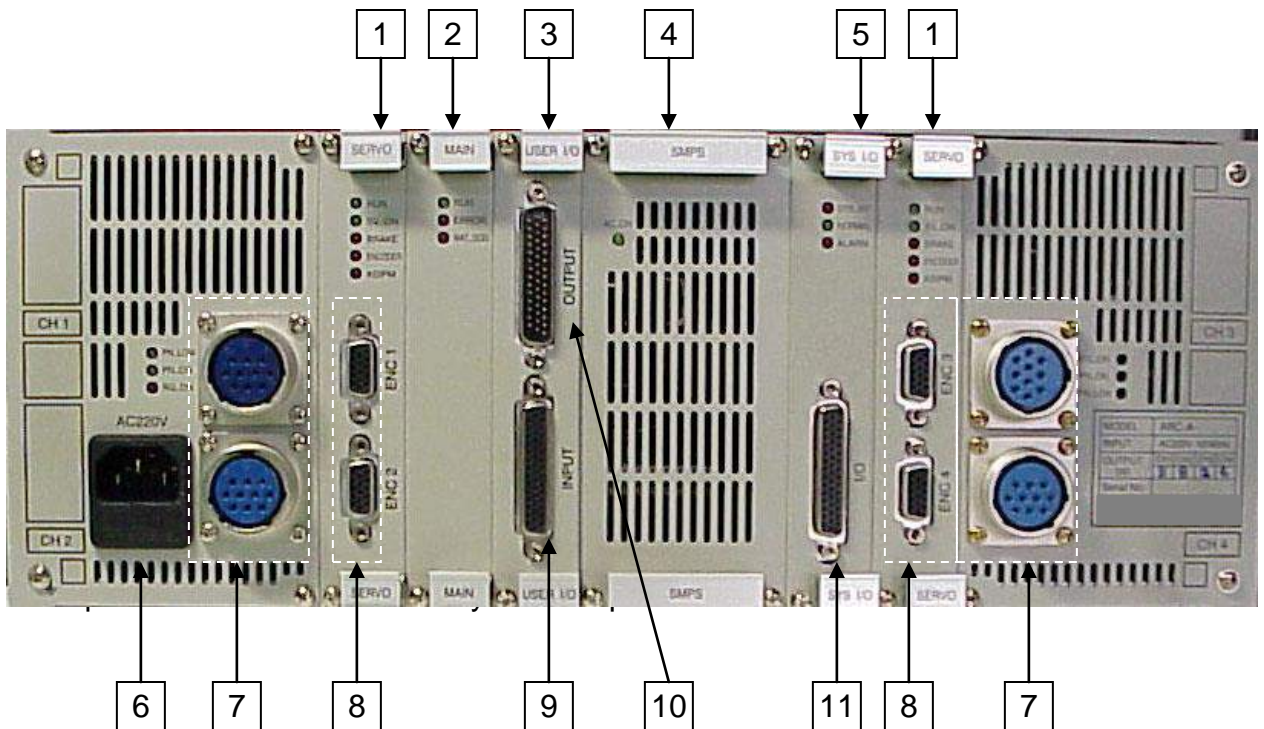
ERR LED : Light is turned on when an error occurs during operation.

- **Front Panel keys** are used to choose and operate program.

KEY	Description
	This key is used for selecting a Program . When selecting a Program , press this key to increase the program number by ten
	This key is used for selecting a Program . When selecting a Program , press this key to increase the program number one by one.

MODE	This key is used to change the controller to Teach Mode or Run Mode.
RESET	This key is used to reset the error caused.
ORG	This key is used to perform an origin-searching operation (Go to position of the origin).
START	This key is used to start program running.
STOP	This key is used to stop program running.

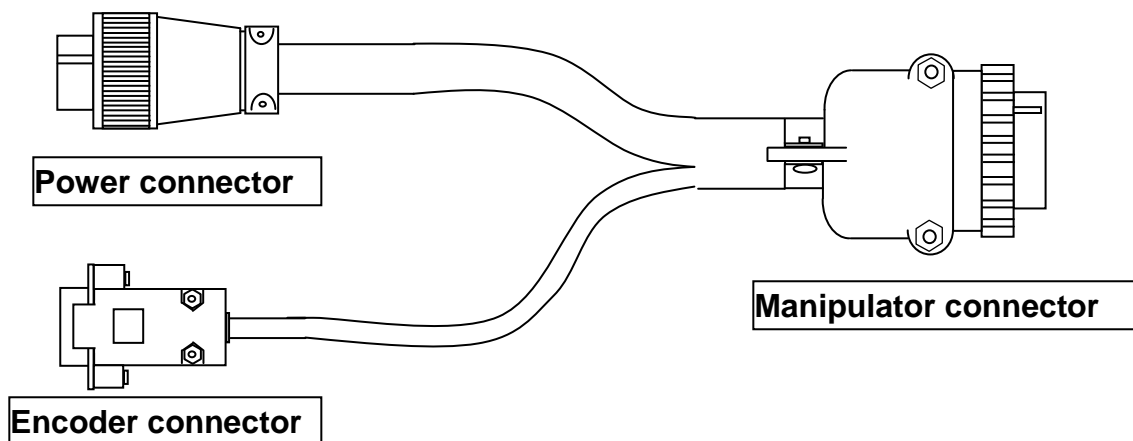
1.4. Rear View



6. **AC Inlet** is used to input the power, AC 220V.
7. **Motor Power Connector** is used to connect motors of the manipulator with connecting cables, supplying high voltage to control the motor.
8. **Encoder Connector** is used to connect the encoder of the motor with the connecting cables, allowing the current value of the encoder to be read.
9. **Input Connector of User I/O board** is used to connect with an external device. This is only for input from the external device
10. **Output Connector of User I/O board** is used to connect with an external device. This is only for the output to the external device.
11. **System I/O Connector** is used to connect with an external device or with the External Control Connector. This is input or output for the system only.

1.5. Connection Cable

This cable is used to connect the machine and the channels of the controller. This cable consists of two outlets - **encoder connector** and **power connector** – on the controller side, while it has one outlet on the machine side.



For proper connection, please refer to **Appendix C: Equipment (machine) Connection.**

1.6. User I/O Cable

This cable is used to connect the user input/output of the external device and the input/output port of the User I/O board. There are independent connectors for the user Input and the user Output. Each connector has 32/32 contacts.

For more details, please refer to **Appendix A.**

1.7. System I/O Cable

This cable is used to connect the input/output ports of the external device and the input/output ports of the system I/O board. The input and output are connected with only one connector. The system input has 24 contacts and the output has 6 contacts. For more details, please refer to **Appendix B**.

2. Installation of the Product

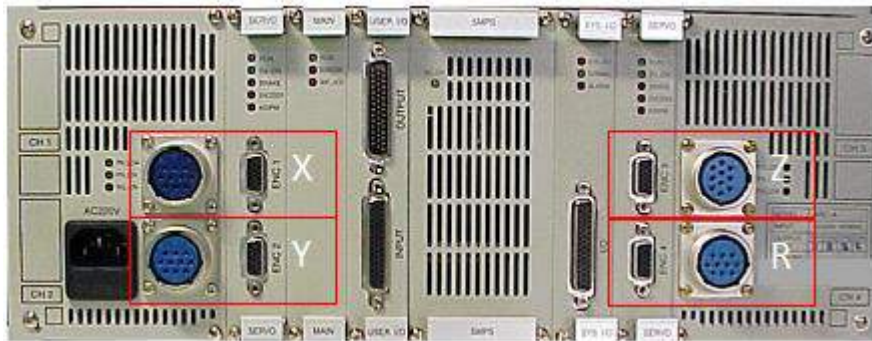
2.1. Initial Considerations.

1. Install the product in a well-ventilated area to avoid overheating.
2. Prevent vibration of the unit. Too much vibration can do considerable damage to the controller.
3. Keep moisture level low. Avoid direct contact between water and the unit.
4. Protect unit against atmospheric agents.
5. Make all connected cables free from vibration.
6. Please install FG (Frame Ground).
7. Make sure that the motor specification indicated on the backside of the controller and the one in the machine are matched.
8. Make sure that the power voltage is AC 220V.
9. Connect all cables appropriately and tie them up to prevent disconnection.

Note: To be seen Appendix F for the coordinates (axes) of the robots.

2.2. Connecting the Controller to the Manipulator

1. Connect the cables between the controller and the manipulator. Depending if the manipulator has 3 or 4 axes, there will be 3 or 4 sets of cables respectively. Each set of cables is labeled with the corresponding connections: CH1-ENC1 (for X), CH2-ENC2 (for Y), CH3-ENC3 (for Z), and CH4-ENC4 (for R). Each channel in the back of the controller is also labeled with the corresponding connections: CH1-ENC1 (for X), CH2-ENC2 (for Y), CH3-ENC3 (for Z), and CH4-ENC4 (for R).



In each set of cables, the end that connects to the controller has two connectors.

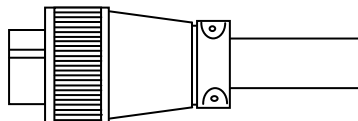
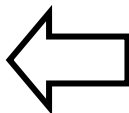
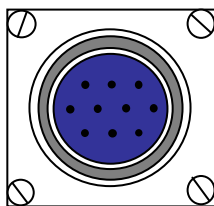
- Power Connector
- Encoder Connector

In the back of the controller, for each axis there are two outlets

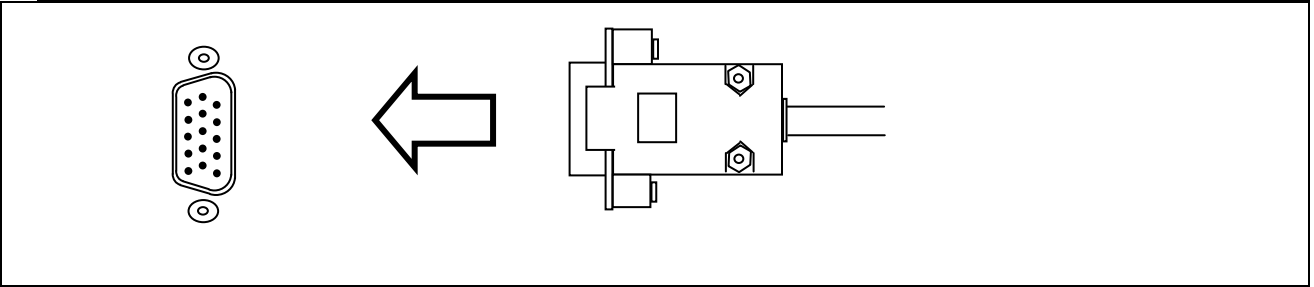
- Power outlet
- Encoder outlet

Connect as follows, matching the connecting symbols labeled on the cables (CH1-ENC1, CH2-ENC2, CH3-ENC3, CH4-ENC4) with the corresponding connecting symbols on the controller.

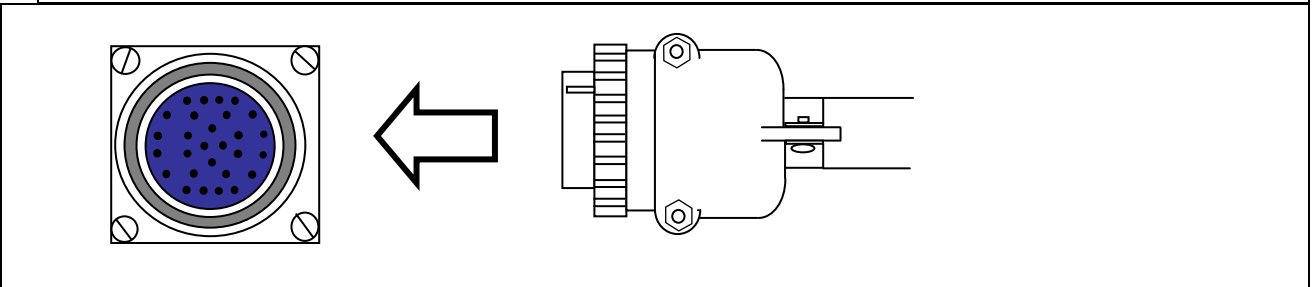
The power connector is connected to the power outlet.



The encoder connector is connected to the encoder outlet.



The manipulator connector is connected to the connecting outlet.



When you connect each cable and the External Control Connector, be sure to tighten the screws so that to avoid any disconnection.

2. Connect the system input/output cable, or the External Control Connector.
3. Connect the user input/output cable (if necessary)
4. Connect the Teach pendant cable to the Teach Pendant (T-Box) outlet on the controller.
5. Connect the power cable. The power should be single-phase AC 220V.

2.3. *Turning ON the Controller for the first time.*

Check that all the cables have been properly connected, and then turn ON the controller. Usually, the controller is preconfigured with the specific parameters for the manipulator. If that is the case, the Teach Pendant will display the following sequence of messages.

**DATA LOADING.....
PLEASE WAIT**

**Press Home Key
To Find Origin**

If the controller has not been preconfigured with the parameters for your manipulator, the Teach Pendant will display the following screen.

**Select Robot Type
TYPE [F3300N]
Change: UP/DOWN
Save: ENT Skip: ESC**

- Using the ▲▼ keys on the Teach Pendant, select your type of the manipulator from the list. Your manipulator's type can be read on the back side of the manipulator.
- Press the ENT key on the Teach Pendant. The Teach Pendant will display the following sequence of messages:

**DATA LOADING.....
PLEASE WAIT**

**Press Home Key
To Find Origin**

Section 3: Teaching Overview

1. Teaching Overview

A program consists of a series of instructions stored in the main unit memory. Each instruction is stored in a numbered memory address. A memory address may record a point location, with an X, Y, Z and R (for 4-axis robots) value and point type or it may store an instruction which sets a parameter, such as a dispensing time or a line speed.

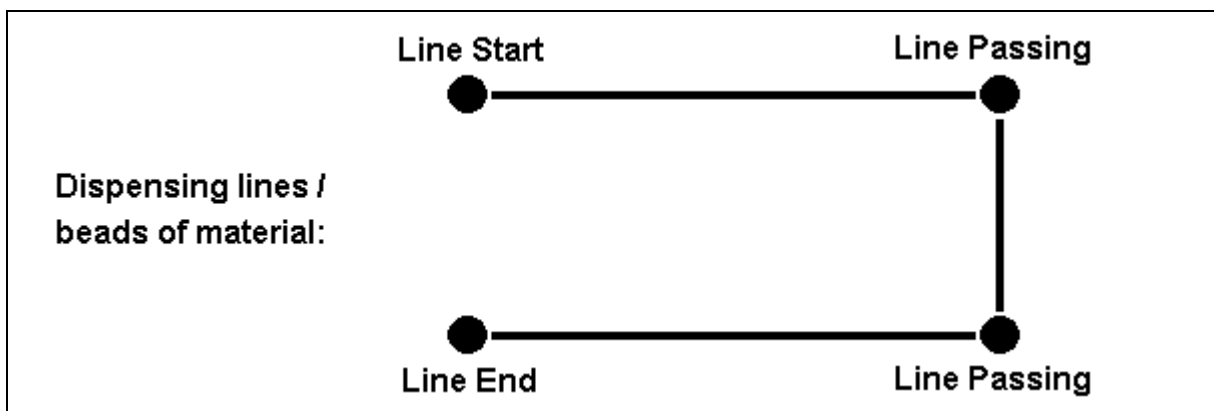
When the program is run, the robot will step through each memory address in sequence and execute the instruction found there. If the memory address contains a point location, the robot will move the X, Y and Z axes to that location and also will execute the rotation corresponding to the value of the R in that point. Depending on the type of point registered at that location, the robot may also perform other functions, such as turn the dispenser on or off.

The most commonly used point types are: Dispense Dot, Line Start, Line Passing, Arc Point, and Line End.

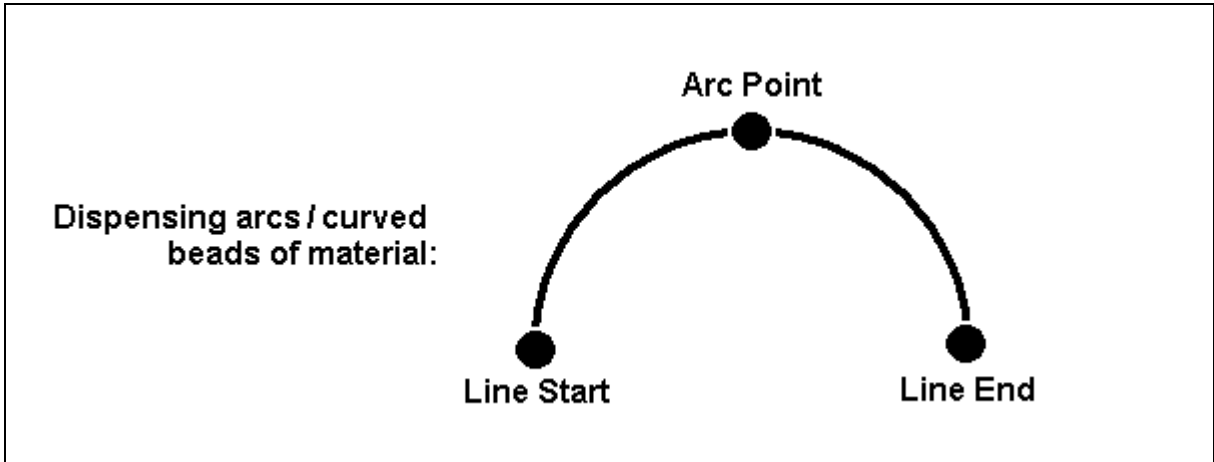
To program the robot to dispense a 'dot' of material, the dispensing tip must be jogged to the desired XYZ location (and in the desired R position of the tip), then that location is registered as a DISPENSE DOT type by pressing the appropriate key on the Teach Pendant.



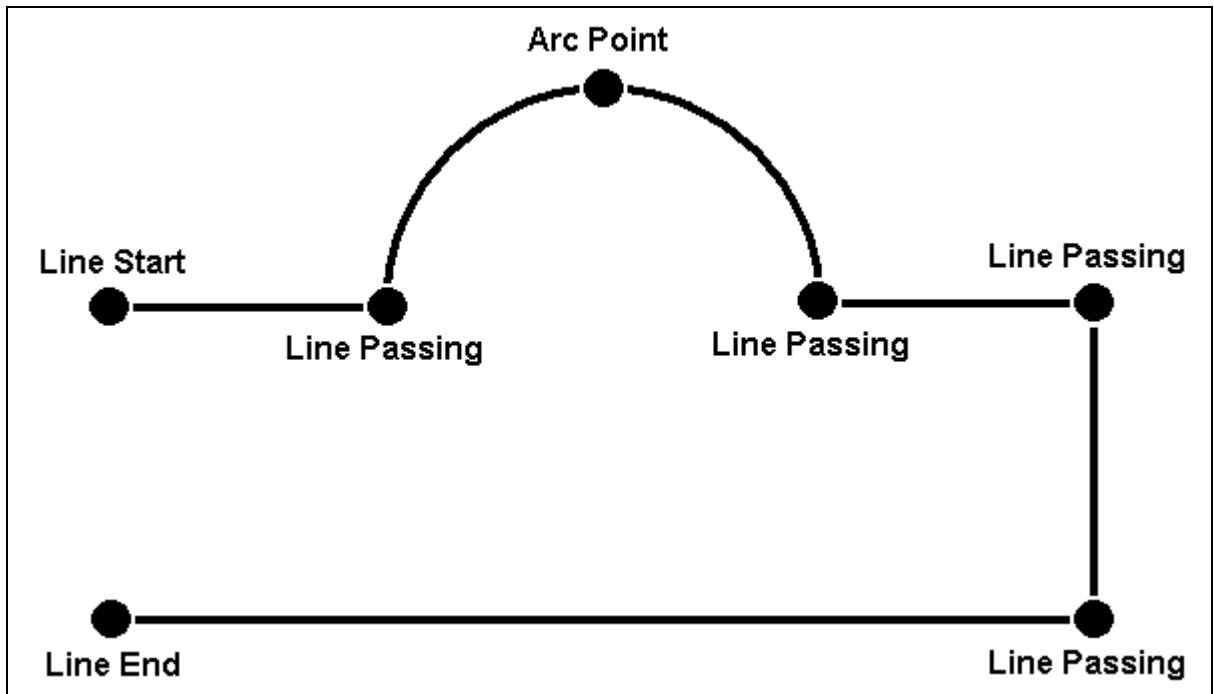
To program the robot to dispense a bead of material along a linear path, the XYZ location (and R position of the tip) of the start of the line is registered as a LINE START point. The locations where the tip changes direction (and position) are registered as LINE PASSING points. The end of the line is registered as a LINE END point:



To dispense a bead of material in an arc, the XYZ location (and R position of the tip) of the start of the line is registered as a **LINE START** point. The high point of the arc is registered as an **ARC Point**. The end of the arc is registered as a **LINE END** point:



Lines and arcs can also be combined to dispense a bead of material along a complex path:



Once the required point locations for your program have been taught, the teach pendant is no longer required. The unit can be switched to RUN mode and operated using the buttons and switches on the main unit control panel.

2. Using the Teach Pendant

The teach pendant enables the user to jog robot to input program data.



When entering numbers, this key is 7

If **Shift/Char** is pressed, released, and next the Speed key is pressed, **Speed** is executed. When entering alphabetical Characters, if **Shift/Char** is pressed, released, this key is 'M'.

2.1. Key Selection














There are several functions assigned to most keys on the teach pendant. When such a key is pressed alone, the function shown in the white colored area on the key is executed. The functions **MENU 1**, **MENU 2**, **Setup**, and **Condition** are all the default key functions that are executed when their keys are pressed alone.



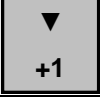
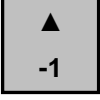

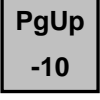
To access the function shown at the top of a blue key, press and release the **Shift /Char** key first, then press the desired key. To select a function shown in the black area of a key, like – for example - the **Speed** function, press and release **Shift/Char**, then press the **Speed** key.

When a number is required, the teach pendant will automatically switch to numeric entry mode. The number represented by each key is shown in the lower left corner of the key.



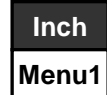
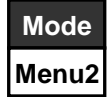


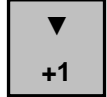

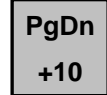

When an Alphabetical character is required, press the **Shift/Char** key first. The character represented by each key is shown in the lower side or in the lower right side of the key.

2.2. Key Assignments


Menu Keys	
	Opens the Point registration menu.
	Opens the Setup menu.
	Opens the Condition Menu.
	Opens Menu # 1. It is also used for the Inch Jog Mode by pressing the Shift/Char key first.
	Opens Menu # 2. It is also used for the Mode Change by pressing the Shift/Char key first.
Jog Keys	
	Jogs the X axis in the forward direction.
	Jogs the X axis in the backward direction.
	Jogs the Y axis in the left direction.
	Jogs the Y axis in the right direction.
	Jogs the Z axis UP.
	Jogs the Z axis DOWN.
	Jogs the Rotation axis.
	

 	<p>Changes jog speed. Right arrow button is used for increasing jog speed. Left arrow button is used for decreasing jog speed.</p>
<p>Navigation Keys</p>	
	<p>Moves forward (1) memory address.</p>
	<p>Moves backward (1) memory address.</p>
	<p>Moves forward (10) memory addresses.</p>
	<p>Moves backward (10) memory addresses.</p>

2.3. Navigation Menu

<p>Pressing any of the keys shown on the right will open the corresponding menu.</p>	    
<p>Once the menu is open, use the up and down arrows to move through the items on the menu.</p>	 
<p>Use the Page Up and Page Down keys to change to the next page or previous page of the menu.</p>	 
<p>Press ENTER to select the current item.</p>	

2.4. Jogging

<p>The tip is jogged by pressing the jog buttons after the Servo button is pressed.</p> <div style="text-align: center;"> <table border="1" data-bbox="683 426 781 522"> <tr><td>Servo</td></tr> <tr><td>A</td></tr> </table> </div>	Servo	A	<table border="1" data-bbox="1151 331 1370 726"> <tr><td>- 1X</td><td>+ 1X</td></tr> <tr><td>S</td><td>X</td></tr> <tr><td>- 2Y</td><td>+ 2Y</td></tr> <tr><td>T</td><td>Y</td></tr> <tr><td>- 3Z</td><td>+ 3Z</td></tr> <tr><td>U</td><td>Z</td></tr> <tr><td>- 4W</td><td>+ 4W</td></tr> <tr><td>V</td><td>W</td></tr> </table>	- 1X	+ 1X	S	X	- 2Y	+ 2Y	T	Y	- 3Z	+ 3Z	U	Z	- 4W	+ 4W	V	W
Servo																			
A																			
- 1X	+ 1X																		
S	X																		
- 2Y	+ 2Y																		
T	Y																		
- 3Z	+ 3Z																		
U	Z																		
- 4W	+ 4W																		
V	W																		
<p>Jog speed has three velocity levels: low, middle and high.</p> <p>If the right arrow button is pressed, the jog speed changes to a faster level.</p> <p>If the left arrow button is pressed, the jog speed changes to a slower level.</p> <p>The value of the jog speed (at the velocity level: high) can be set using the function Jog Speed (Menu 1). (See Section 6: 4.6 Jog Speed).</p> <p>The Speed led display on the Teach Pendant shows the velocity level: high, middle or low.</p>	<div style="text-align: center;"> <table border="1" data-bbox="1135 804 1386 903"> <tr><td>◀</td><td>▶</td></tr> <tr><td>SPD -</td><td>SPD +</td></tr> </table> </div> <div style="text-align: center;">  </div>	◀	▶	SPD -	SPD +														
◀	▶																		
SPD -	SPD +																		
<p>Changes to Inch Jog Mode.</p>	<table border="1" data-bbox="1138 1270 1370 1367"> <tr><td>Shift</td><td>+</td><td>Inch</td></tr> <tr><td>/Char</td><td></td><td>Menu1</td></tr> </table>	Shift	+	Inch	/Char		Menu1												
Shift	+	Inch																	
/Char		Menu1																	

2.5. Data Entry

The Teach Pendant is used also to enter numeric data. If a numeric value is required, the Teach Pendant will automatically switch to numeric mode. Use the keys 0 – 9, (.), and (-) to enter the values.













2.6. LED Panel

This indicates the current system status and speed.

- **SERVO LED** : The light is turned on when the robot is run or ready to move after **Servo is turned ON**.
- **ORG LED** : The light is turned on after the function of **Origin** is performed.
- **CHAR LED** : The light is turned on when the **CHAR** key is pressed.
- **ERR LED** : The light is turned on when the **Error** occurs in the robot.
- **INCH LED** : The light is turned on when the current mode is **Inch Mode**.
- **LOW LED** : The light is turned on when the current jog speed is **low**.
- **MED LED** : The light is turned on when the current jog speed is **medium**.
- **HIGH LED** : The light is turned on when the current jog speed is **high**.

3. Teach Box Key Assignments

<i>Key</i>	<i>Function</i>
<div style="border: 1px solid black; padding: 2px; width: 40px; margin: 0 auto;">F1</div> <div style="border: 1px solid black; padding: 2px; width: 40px; margin: 0 auto;">Setup</div>	Opens Setup Menu.
<div style="border: 1px solid black; padding: 2px; width: 40px; margin: 0 auto;">F2</div> <div style="border: 1px solid black; padding: 2px; width: 40px; margin: 0 auto;">Cond</div>	Opens Condition Menu.
<div style="border: 1px solid black; padding: 2px; width: 40px; margin: 0 auto;">Inch</div> <div style="border: 1px solid black; padding: 2px; width: 40px; margin: 0 auto;">Menu1</div>	Opens Menu #1. With Shift/Char key, it is used for Inch Jog Mode.
<div style="border: 1px solid black; padding: 2px; width: 40px; margin: 0 auto;">Mode</div> <div style="border: 1px solid black; padding: 2px; width: 40px; margin: 0 auto;">Menu2</div>	Opens Menu #2. With Shift/Char key, it is used for Mode Change.
<div style="border: 1px solid black; padding: 2px; width: 40px; margin: 0 auto;">Servo</div> <div style="border: 1px solid black; padding: 2px; width: 40px; margin: 0 auto;">A</div>	Turns the servo motor On/Off.
<div style="border: 1px solid black; padding: 2px; width: 40px; margin: 0 auto;">Brake</div> <div style="border: 1px solid black; padding: 2px; width: 40px; margin: 0 auto;">B</div>	Releases or locks the Brake.
<div style="border: 1px solid black; padding: 2px; width: 40px; margin: 0 auto;">Jump</div> <div style="border: 1px solid black; padding: 2px; width: 40px; margin: 0 auto;">C</div>	Jumps to a specified memory address.
<div style="border: 1px solid black; padding: 2px; width: 40px; margin: 0 auto;">STOP</div>	Stops program running.

Key	Function
	Goes to another program.
	Moves the tip to the point location currently in the display.
	Inserts a memory address before the current address.
	Deletes the current memory address.
	Changes to character mode or shift mode.
	Resets current error.
	Changes from Point List display mode to single point display when teaching point data. If pressed once, clears the current numeric value.
	Jog speed has three velocity levels: low, middle and high. If the right arrow button is pressed, jog speed is changed to a faster level. If the left arrow button is pressed, jog speed is changed to a slower level. The Speed led display on the Teach Pendant shows the velocity level.
	Moves backward (10) memory addresses.
	Moves backward (1) memory address.
	Moves forward (10) memory addresses.
	Moves forward (1) memory address.

Key	Function	
ENT	Confirms data entries. Also opens the Point registration menu.	
Shift /Char	● Dot 1 / G	Shortcut for registering a Dispense Dot.
Shift /Char	—Start 2 / H	Shortcut for registering a Line Start point.
Shift /Char	—Pass 3 / I	Shortcut for registering a Line Passing point.
Shift /Char	—End 4 / J	Shortcut for registering a Line End point.
Shift /Char	—Arc 5 / K	Shortcut for registering an Arc Point.
Shift /Char	End Pr 6 / L	Shortcut for registering the End Program command.
Shift /Char	Speed 7 / M	Shortcut for registering Line Speed.
Shift /Char	● Setup 8 / N	Shortcut for registering a Point Dispense Setup.
Shift /Char	— Setup 9 / O	Shortcut for registering a Line Dispense Setup.
First 0 / P	Changes the display to memory address number 0.	
End ./ Q	Changes the display to the last used memory address in the program.	
Shift /Char	Mode Menu2	Changes to Run / Teach Mode.
Home - / R	'Home' the robot. Initializes all axes and moves to (0,0,0,0).	

4. Point Type Functions Summary

4.1. Point Menu

Below is a list of Point type functions that are found under the **Enter** key (**Point** menu):

Function	Description
Dispense Dot	Registers the current XYZ location (and R position of tip) as a Dispense point for dot dispensing.
Line Start	Registers the current XYZ location (and R position of tip) as a Line Start point for line dispensing.
Line Passing	Registers the current XYZ location (and R position of tip) as a Line Passing point. This is a location on the line where the tip changes direction, such as at the corner of a rectangle.
Line End	Registers the current XYZ location (and R position of tip) as a Line End point.
Arc Point	Registers the current XYZ location (and R position of tip) as an Arc Point. Arc Points are used to dispense material in an arc or circle.
Circle	Registers the current XYZ location (and R position of tip) as a Circle point. Circle points are used to dispense material in a circle.
Center	Registers the current XYZ location (and R position of tip) as a center point of circle. Center points are used to dispense material in a circle.
Dummy	Registers the current XYZ location (and R position of tip) as a Dummy point. The tip will simply pass through this point. This is useful for avoiding obstacles on the work piece.
End Program	Registers the current memory address as the end of the program.

<i>Function</i>	<i>Description</i>
Dispense ON / OFF	Registers an instruction which turns the dispenser on or off at the current memory address.
Home Point	Sets robot to home position.
Wait Point	Registers a wait time. When executed, the tip will wait for the specified period of time.
Stop Point	Registers a Stop Point at the current X, Y, Z location (and R position of tip). When executed, the tip will move to that location and wait until the start button is pressed.
Brush Area	Causes the tip to 'paint' the defined area. The painted area can be in the form of a rectangle or a circle / spiral.
If	Registers an instruction that either sets the value of an output signal or checks the status of an input signal.
Output	Registers an instruction that sets the value of an output signal.
Input	Registers an instruction that waits for an input signal.
Pulse	Registers an instruction that sets the value of an output signal and Output Time.
Point	Sets point variable P0 ~ P99 by saving current position or input numerical data.
Load Point	Registers the current XYZ location (and R position of tip) as a Load point. The tip will simply move this point. The point will not be affected by 'Step & Repeat' or 'Offset' instruction.

4.2. Setup Menu

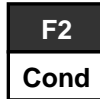
Below is a list of functions that are found under the **Setup** key (**Setup** menu):

<i>Function</i>	<i>Description</i>
Line Speed	Registers the LINE SPEED used for all lines from the current memory address forward until another Line Speed instruction is found.
Line Dis. Setup	Registers the LINE DISPENSE SETUP values which set dispensing wait time at the start of lines ('head' time) waiting time at the end of lines ('tail' time), and dispense off length ('head' length and 'tail' length). Output ('Output') The registered values will be used from the current memory address forward until another Line Dispense Setup instruction is found.
Point Dis. Setup	Registers POINT DISPENSE SETUP values which set dispensing time and waiting time at the end of dispensing ('tail' time) for dots. Output ('Output'), The registered values will be used from the current memory address forward until another POINT DISPENSE SETUP instruction is found.
Dispense End Setup	Registers the height and speed the tip should rise at the end of dispensing. The registered values will be used from the current memory address forward until another DISPENSE END SETUP instruction is found.
Z Clearance	Registers the additional distance the tip should rise, beyond the height set in Dispense End Setup, to allow obstacles to be cleared as the tip moves from one figure to another. Values will be used until another Z Clearance instruction is found.
XY Move Speed	Sets the movement speed of the X and Y axes when moving from one figure to another in the program.
Z Move Speed	Sets the movement speed of the Z axis when moving from one figure to another in the program.

<i>Function</i>	<i>Description</i>
Home Position	Changes the position the robot moves to at the end of a program cycle.
Retract	Registers Retract values at the current XYZ location. Retract causes the tip to move up and back over the dispensed bead after line dispensing.
Auto Purge	Registers Wait time and Purge time, for purging the system at the end of a program.
Adjust Point #1	Saves current position and steps as a first data for Relocate Data function. Saves current position to temporary point #1.
Adjust Point #2	Saves current position and steps as a second data for Relocate Data function. Saves current position to temporary point #2.
CCD Shot	Used for Vision Application
Relocate	Used for Vision Application
Round	Sets radius of a line at a Line Passing point.
Z Lift	Select whether lifting Z axis or not, when robot MOVE to point in TEACHING MODE.
MarkPoint #1 / #2	Set Base Position for Vision System.
Move Cond	Register 'Line Speed', 'Z clearance(Abs)', 'Round'.
Disp Cond	Register Dispensing time('Disp time', 'Tail time') and Dispense length('Head Lehgh', 'Tail length')

4.3. Condition Menu

Below is a list of functions that are found under the



key:

Function	Description
Goto Address	Causes the program to jump to the specified memory address when executed.
Step & Repeat X	Registers an instruction that will re-run a selected group of memory addresses, stepping by a user-defined distance in the X or Y axis after each copy. The matrix of parts is defined by specifying the number of rows, the number of columns, the X offset and the Y offset. Step & Repeat X indicates that the robot will give priority to the X axis, running the parts along the X axis first.
Step & Repeat Y	Registers an instruction that will re-run a selected group of memory addresses, stepping by a user-defined distance in the X or Y axis after each copy. The matrix of parts is defined by specifying the number of rows, the number of columns, the X offset and the Y offset. Step & Repeat Y indicates that the robot will give priority to the Y axis, running the parts along the Y axis first.
Call Subroutine	Causes the machine to jump to a specified memory address and execute the instructions found there. When the end program instruction is reached, program execution will continue at address just after the call Subroutine instruction.
Call Program	Executes the specified program number from within the current program. After the called program completes, the current program will continue execution.
Loop Address	Causes the program to execute a group of memory addresses a user-specified number of times.

Label	Sets Label. Label can be used instead of Address Number.
Arm	For TMB Series, it determines the position of the arm. For F9000N Series, it determines the XYZ coordinate or RYZ coordinate.
FixR	Fixes/unfixes the axis rotation while a linear movement is executed.
Calc	Arithmetic Instruction.
Jmov/ Lmov	Registers current XYZ position or Point variable to Moving position.
IncJ / IncL	Registers current XYZ position or Point variable as an Offset position.
XMov	While moving, if a sensor signal is turned ON, robot will stop immediately, and save current position to P98.
XCOPY	It Is Used for DualX system. it duplicate the XYZ coordinate Position to RYZ coordinate with Y Axis Offset.
Pallet	Registers a pallet movement setting.
Offset	Set Offset value or point number. Every position after Offset command will be increased by Offset Value.
Pattern / End Pattern	It is similar with the Step & Repeat Instruction, but it can change the Repeat order.
COMM	Send String through RS232C port.

4.4. Menu 1

Below is a list of functions that are found under the **Menu 1** key:

Function	Description
Program Name	Allows the user to register a name for the current program number.
Z Axis Limit	Sets the limits between which the Z axis will move during a program.
Initial Output	Sets the status of the output signals when the machine is initialized.
Cycle Counter	Enables or disables the program cycle counter shown in the display when in run mode.
Set Password	Sets Password to protect all programs from unwilling editing.
Jog Speed	Sets the value of the jogging speed, both the linear speed and the rotation speed (for 4-axis robots).
Run Mode	Determines whether the robot operates in Standalone mode (default) or Slave mode. Slave mode allows the robot to be controlled by commands sent over the RS232 port.
Adjust Position	Sets Adjust Position for Adjust Origin.
Parameter	Sets all parameters about machine and controller.
Resume	Determines if a program will restart from point #0 or the point at which it was interrupted if a program is stopped by an emergency stop or the enclosure door switch open signal.
Origin searching	Executes origin searching, making the robot to go to the position established as home position.
Hour Meter	Shows the working time and running time of the robot.
PLC File Edit	Edits PLC File.
I/O Monitor	Views Input/Output status.

<i>Function</i>	<i>Description</i>
Set Variable	Sets Variable that is used in arithmetic instruction.
Emg Mode	Selects SYS EMG mode.
Vision Set	N/A

4.5. Menu 2

Below is a list of functions that are found under the **Menu 2** key:

<i>Function</i>	<i>Description</i>
Point Utility	
MDI Mode	Changes Position value of each address by numerical key input.
Numerical Move	Allows the tip to be positioned numerically by entering numbers for the X, Y, Z and R values.
Save Temp Point	Saves the current XYZR position in a temporary memory area numbered 1 – 9.
Retr. Temp Point	Retrieves an XYZR position, which was stored with Save Temp Point.
Group Utility	
Group Edit	Allows a function to be applied to a user-defined group of memory addresses. Functions include copy, delete, move, multiply line speed, multiply dispense times, apply X offset, apply Y offset, apply Z offset.
Expand Step & Repeat	Expands the memory address locations which would be performed at a Step & Repeat instruction so individual memory addresses of the repeated instructions can be edited.
Relocate Data	Allows the position of a program to be corrected using two reference points. Corrects for X offset, Y offset, Z offset and angle of rotation.

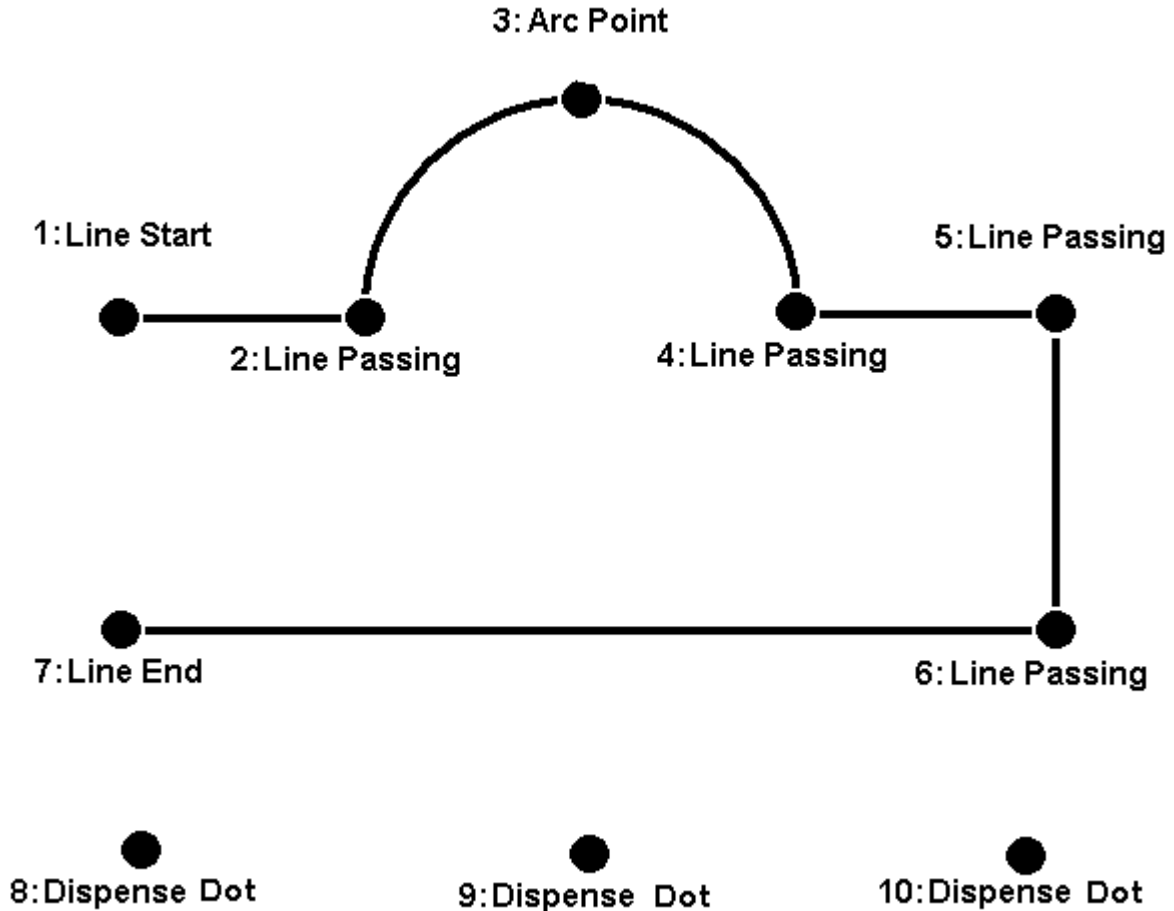
<i>Function</i>	<i>Description</i>
Adjust Origin	Adjusts origin position.
Program Utility	
Copy Program	Allows programs to be copied.
Delete Program	Allows programs to be deleted.
Auto Offset	Adjust position quickly.
Memory Utility	
Delete Memory	Clears whole memory.

Note: Certain functions shown on the display are not applicable to these units and consequently they are marked in the manual as "N/A".

Section 4: Programming Example

1. Programming Example




To help you become familiar with programming the robot, please follow the instructions below to create a program that dispenses in the following pattern:



Notes:

- We will create the pattern above in the Program # 10.
- We will use a Line Speed of 40 mm/sec for the lines and arcs in the program.
- For the line Line Dispense Setup, Head Time 0.1 sec, Tail Time 0.1 sec, Wait Time 0 sec, Tail Length 0 mm and Output is 0. This setting is preliminary and is subject to change depending on the material properties (i.e. viscosity, etc.).
- For dots, we will use a dispensing time of 0.50 seconds and a waiting (tail) time of 0.1 seconds after dispensing.

	<i>Instruction</i>	<i>Display Shows</i>
1	Turn ON the controller.	
2	<p>After the robot initializes, the Teach Pendant will show either one of the two displays shown on the right.</p> <p>Press the HOME button. The robot will move to the home position.</p> <p>If the robot starts up in Run mode, select EXIT by pressing the Menu2 button to switch to Teach mode.</p>	<p><u>Run Mode Screen:</u></p> <pre> PROG:00 AUTO Press Move Key Cycle Counter: 0 MODE PLC RESET EXIT </pre> <p><u>Teach Mode Screen</u></p> <pre> ----- Press Home Key To Find Origin </pre>
3	The Teach Pendant should display the following upon initialization.	<pre> ADDR:0 PROG:1 EMPTY X:0 Y:0 Z:0 R:0 </pre>
4	Press the Program number key (Prog Num) then use the keypad and type 10.	<pre> --Change Prog No.-- Input No. 10 </pre>
5	You should see the program number on the upper right corner of the Teach Pendant screen change to "10". You are now ready to create a new program.	<pre> ADDR:0 PROG:10 EMPTY X:0 Y:0 Z:0 R:0 </pre>
6	Press the F1/Setup key, then 1 to select Line Speed (from page 1/4 of the Setup Menu) to register a line speed of 40 mm/second at memory address number 0. Press 40 , then ENTER to register a speed of 40 mm / second.	<pre> Line Speed ----- Speed: unit: mm/sec </pre>

	<i>Instruction</i>	<i>Display Shows</i>
7	<p>Pressing ENTER automatically saves the speed. The address number (ADDR) will change to "1".</p> <p>Note: The robot automatically goes to the next address line once a command has been saved. To check if the command is saved properly, use the navigation keys:</p> 	<p>ADDR:1 PROG:10 EMPTY</p>
8	<p>Press the F1/Setup key, then 2 to select Line Dispense Setup (from page 1/4 of the Setup Menu) to register a Line Dispense Setup at memory address number 1.</p>	<p>Line Disp Setup Head Time:0.1 Tail Time:0.1 Wait Time:0 Head Length:0 Tail Length:0 Output:0</p>
9	<p>The display shows that we are at memory address 1 and that it is empty.</p> <p>Jog the dispense tip to the first location in the diagram above (1: Line Start).</p> <p>To jog the X, Y, Z axes and R position of tip, press the respective Jog keys.</p>  <p>Press the  key to jog fast. See</p> <p>Teaching Overview Section 3:2.4 Jogging for more information.</p>	<p>ADDR: 2 PROG: 10</p> <p>X: 0 Y: 0 Z: 0 R: 0</p>

	<i>Instruction</i>	<i>Display Shows</i>
10	Once the tip is at the correct X, Y, Z, R location for the first point (1: Line Start), press the ENTER key, then 2 (from page 1/5 of the Point Menu) to register the location as a Line Start point. (Note: From now on, the symbols and the values displayed for the X, Y, Z axes and R position will not be written in the column “Display Shows” of this example).	ADDR: 3 PROG: 10 EMPTY
11	The display will show that we are at memory address 2 and it is empty. Jog the tip to the X, Y, Z, R location of the second point (2: Line Passing). When the location is correct, press the ENT key, then 3 (from page 1/5 of the Point Menu) to register the location as a Line Passing point.	ADDR: 4 PROG: 10 EMPTY
12	Now jog the tip to the location of the third point (3: Arc Point). When the location is correct, press the ENT key, then 1 (from page 2/5 of the Point Menu) to register the location as an Arc Point.	ADDR: 5 PROG: 10 EMPTY
13	Jog the tip to the location of the fourth point (4: Line Passing). When the location is correct, press the ENT key, then 3 (from page 1/5 of the Point Menu) to register the location as a Line Passing point.	ADDR: 6 PROG: 10 EMPTY
14	Jog the tip to the location of the fifth point (5: Line Passing). When the location is correct, press the ENT key, then 3 (from page 1/5 of the Point Menu) to register the location as a Line Passing point.	ADDR: 7 PROG: 10 EMPTY
15	Jog the tip to the location of the sixth point (6: Line Passing). When the location is correct, press the ENT key, then 3 (from page 1/5 of the Point Menu) to register the location as a Line Passing point.	ADDR: 8 PROG: 10 EMPTY

	<i>Instruction</i>	<i>Display Shows</i>
16	Jog the tip to the location of the seventh point (7: Line End). When the location is correct, press the ENT key, then 4 (from page 1/5 of the Point Menu) to register the location as a Line End point.	ADDR:9 PROG:10 EMPTY
17	The line is now complete. The next step is to register the dispense settings for the dots. Press the F1/SETUP key, then 3 to register the Point Dispensing Setup.	Point Disp Setup Dis. Time: sec Tail Time: sec unit: sec
18	Type 0.5 to register a dispensing time of 0.5 seconds, and then press ENT . Type 0.1 to register a waiting (tail) time after dispensing of 0.1 seconds, then press ENT .	ADDR:10 PROG:10 EMPTY
19	Jog the tip to the location of the first dispense dot (8: Dispense Dot). When the location is correct, press the ENT key, then 1 (from page 1/5 of the Point Menu) to register the location as a Dispense Dot.	ADDR:11 PROG:10 EMPTY
20	Jog the tip to the location of the second dispense dot (9: Dispense Dot). When the location is correct, press the ENT key, then 1 (from page 1/5 of the Point Menu) to register the location as a Dispense Dot.	ADDR:12 PROG:10 EMPTY
21	Jog the tip to the location of the third dispense dot (10: Dispense Dot). When the location is correct, press the ENT key, then 1 (from page 1/5 of the Point Menu) to register the location as a Dispense Dot.	ADDR:13 PROG:10 EMPTY
22	The program is now complete. Press ENT , then 4 (from page 3/6 of the Point Menu) to register address 12 as the END of the program.	ADDR:14 PROG:10 EMPTY

To Run the Program:

Press “**Shift/Char**” button, then “**Mode/Menu 2**” button and then “**Move/D**” button.

2. *Editing a Program*

You can move through the instructions in an existing program by using the following keys:

Key	Function
▼/+1	Moves forward (1) memory address
▲/-1	Moves backward (1) memory address
FIRST	Moves to the first memory address in the program
END	Moves to the last programmed memory address in the program
PgDn/+10	Jumps forward (10) memory addresses
PgUp/-10	Jumps backward (10) memory addresses
MOVE	Moves the tip to the X,Y,Z,R point location of the selected point
JUMP	Jumps to display the specified memory address

2.1. *Changing a Point XYZ location*

To change the X,Y,Z location of a point, press the **▼/+1** or **▲/ -1** key until the point you want to change is shown in the display.

You can confirm that the correct point is in the display by pressing the **MOVE** key. This will cause the tip to move to the X, Y, and Z position shown in the display.

Now use the Jog keys to jog the tip to the new location.

Once the location is correct, simply re-register the point as you did when it was first taught, by pressing the **ENT** and selecting the point type. The point will be re-registered at the new location.

2.2. *Insert / Delete an Instruction*

To insert an instruction, press the **INS/E** key. The instruction currently shown in the display will be moved forward one memory address. A new memory address can be inserted at the current memory address, after pressing the **INS/E** key when the letter “I” appears on the right up corner of the display. Type the data of the new address to be inserted, then press the **ENT** key.


To delete an instruction currently shown in the display, press the **DEL/F** key, then press the **1/G** key (when you will be asked to confirm).

3. *Changing the Program Number*




The program number is selected using the program number selection switches on the main unit control panel (controller).



Press the ◀ and ▶ buttons to select the program number.

In Teach Pendant mode, press the  key to change the program number, then type the new program number and press the **ENT** key.

4. *Changing from Teach Mode to Run Mode*

To switch between Teach mode and Run mode, press the  key in the main unit control panel. Or press the  +  keys on the Teach Pendant.

When the machine is in Run mode, the Teach Pendant is not required. Programs can be selected and run using the switches on the front control panel of the main control unit (controller).

5. *AUTO Mode and STEP Mode in Run Mode*

In Run Mode have two modes that are Auto mode and Step Mode.

Auto Mode means that robot will run all program step automatically.

Step Mode means that robot will run each step by start Input one by one.

In Auto Mode, pressing start key will robot run program from start address to End address.

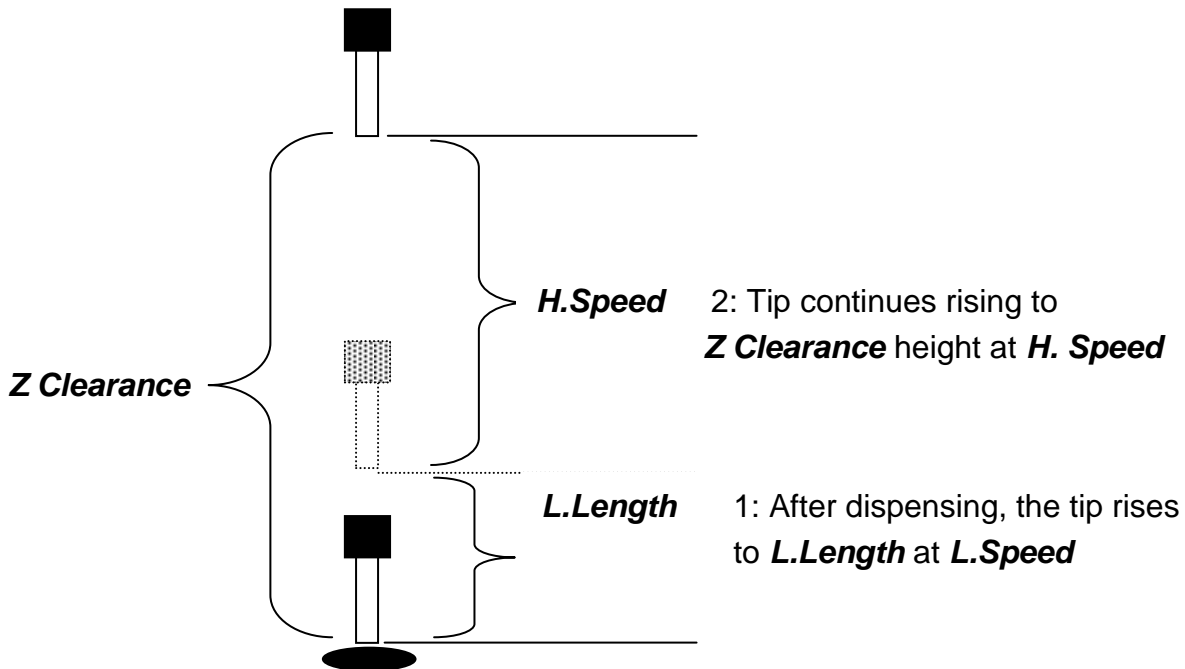
Section 5: Dispensing Parameters

1. Dispense End Setup

After dispensing a dot or line, it is often required to raise the tip a short distance at a slow speed. This allows the material to cleanly break free from the tip, without 'dragging' material where it is not wanted.

The distance and speed which the tip rises after dispensing is controlled by the **L.Length** and **L.Speed** settings.

After the tip rises the length specified by **L.Length** at the speed specified by **L.Speed**, the tip will continue rising to the **Z Clearance** height at the speed specified by **H.Speed**. The purpose of specifying a Z Clearance height is to allow the tip to rise high enough to clear any obstacles it may encounter on the way to the next point.



Values for **H.Speed**, **L.Speed** and **L.Length** are registered with the **Dispense End Setup** function by pressing the **SETUP** key, then choosing **Dispense End Setup**.

Once **Dispense End Setup** values have been registered at a memory address, all points after that memory address will use the values specified. If **Dispense End Setup** values are registered again, at a higher memory address, all the points from that memory address forward will use the new values.

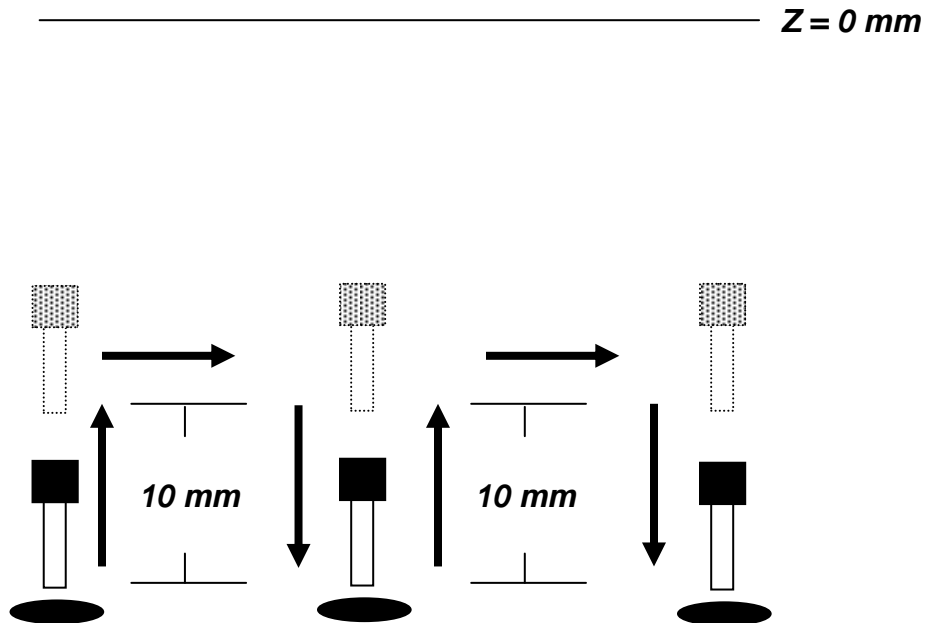
2. Z Clearance

The purpose of the Z Clearance function is to cause the tip to rise high enough to clear all obstacles as it moves from one point to another. If there are no obstacles between any of the program points, a small Z Clearance value, such as 5 mm, can be used to minimize the program cycle time.

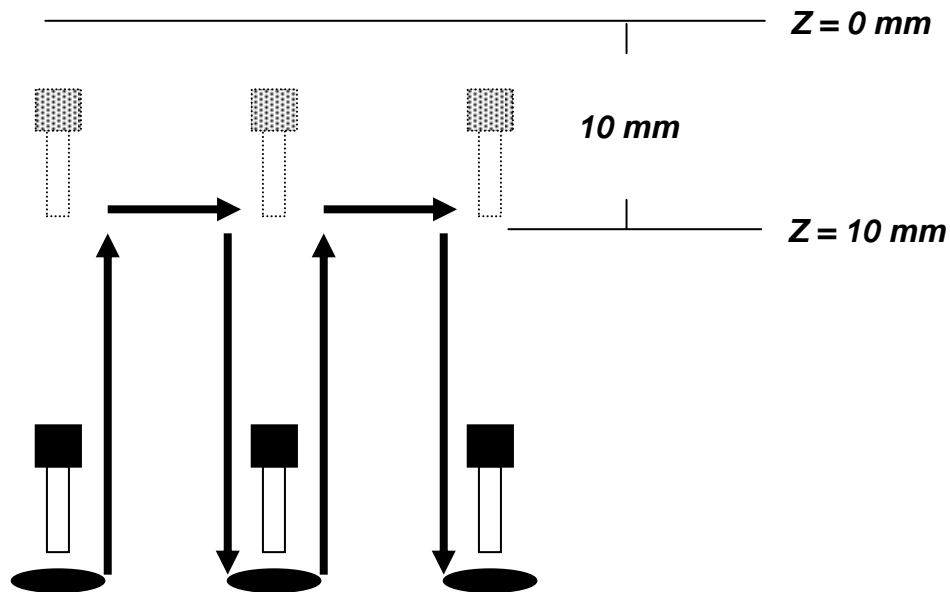
Values for the *Z Clearance* are registered by pressing the **SETUP** key, then choosing **Z Clearance**. All the points from that memory address forward will use the Z Clearance value until another *Z Clearance* value is found. Normally, a Z clearance instruction should be registered in the beginning of a program, at one of the first few memory addresses.

The Z Clearance value may be specified as a relative value or as an absolute value. When specified as a relative value, it is the distance to rise relative to the taught point location. When it is specified as an absolute value, it is the distance from the Z axis zero position which the tip will rise to, regardless of the Z axis value of the taught point location. For example:

Z Clearance = 10 mm RELATIVE:



Z Clearance = 10 mm ABSOLUTE:



3. *Line Dispense Setup*

When dispensing high viscosity materials, there is often a delay from the moment the dispenser is turned on until the material begins to flow. The following parameters are set under this function: Head Time, Tail Time, Head Length, Tail Length and Output.

The **Head Time** setting is a delay time used at the start of a line dispensing to prevent the tip from moving along the line path until the material is flowing.

The tip will move to the start of the line, turn on the dispenser and wait for the time period specified in the *head time* setting before moving. The time value can be adjusted to ensure that the material begins flowing at the same time as the line movement begins.

At the end of dispensing, a delay is often required after the dispenser is turned off, to allow the barrel pressure to equalize, before moving to the next point location. This prevents material from being 'spilled' where it is not wanted. This time delay at the end of dispensing is called the **Tail Time**.

Head Length defines the distance between the Line Start point and the point where the dispensing of the material has to begin. When the value of the Head Length is “0” the dispensing will begin from the Line Start point (when the value of the Head Time is “0”).

Tail Length defines the distance between the point where the dispensing is turned off (at the user’s choice) and the Line End point, thus being prevented the excess of the material to be deposited at the end of the line. This is necessary because usually the material continues to flow for a while after the dispenser is turned off, due to the pressure built in the system.

Output defines the dispense output port number for Line dispense and Arc, Circle dispense work. The default dispense output port number is zero. If the value of “Output” is “1”, Output port 1 is on from Line Start point and off on the Line End point.

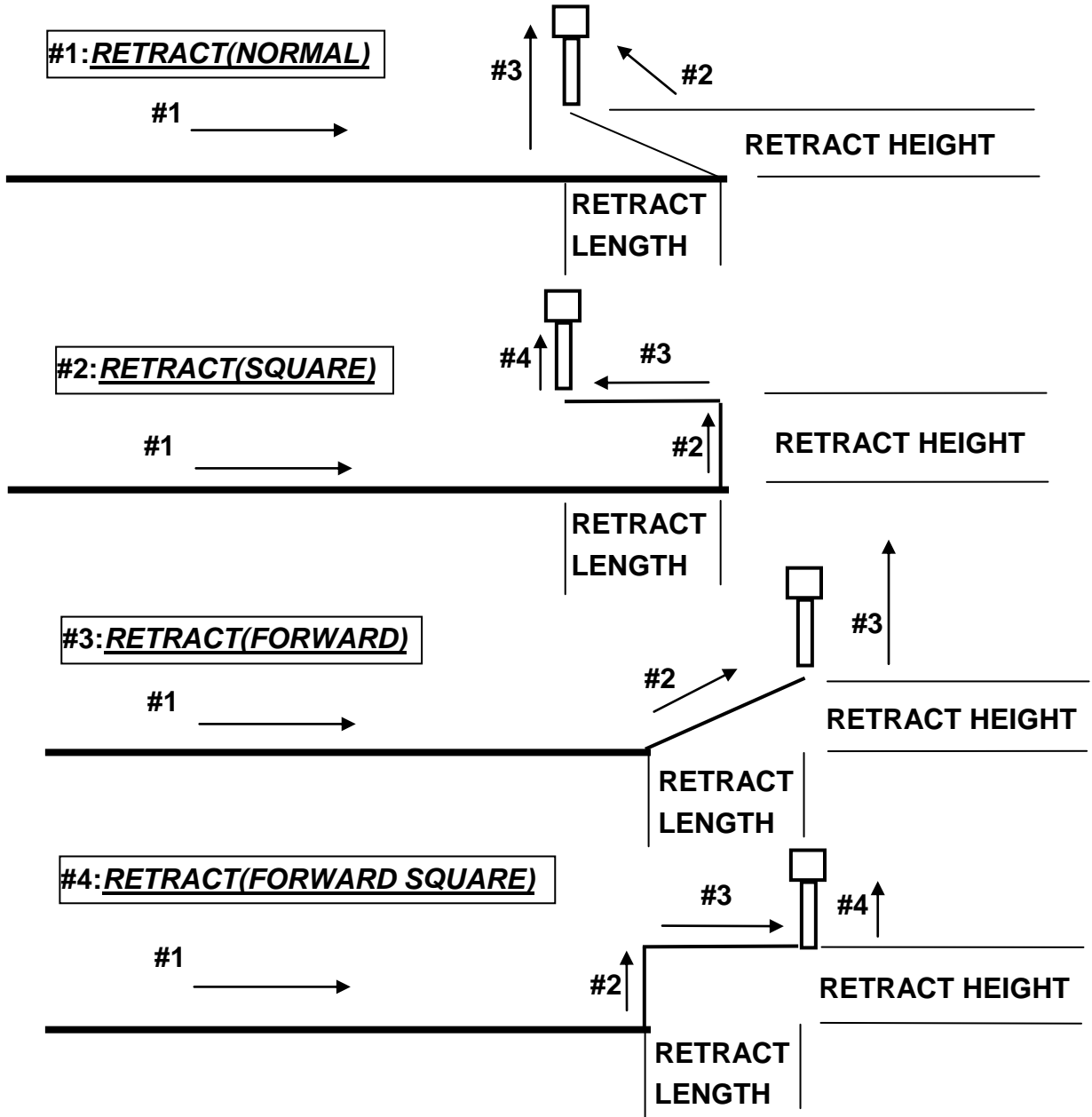
Values for the *Head Time*, *Tail Time*, *Head Length* and *Tail Length* used when performing line dispensing are registered by pressing the **SETUP** key, then selecting **Line Dispense Setup**. The set values will be used by all lines from that memory address forward until a new set of *Line Dispense Setup* values is found.

4. Retract

The Retract function gives the programmer the possibility to control the tip at the end of the dispensed line. It allows the tip to retract not only upward but also backward, forward or at an angle. This is useful when dispensing high viscosity or ‘stringy’ materials as it will lay the material tail down on the dispensed bead.

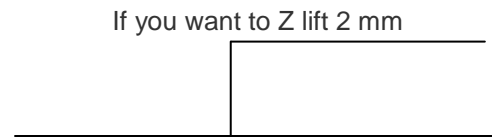
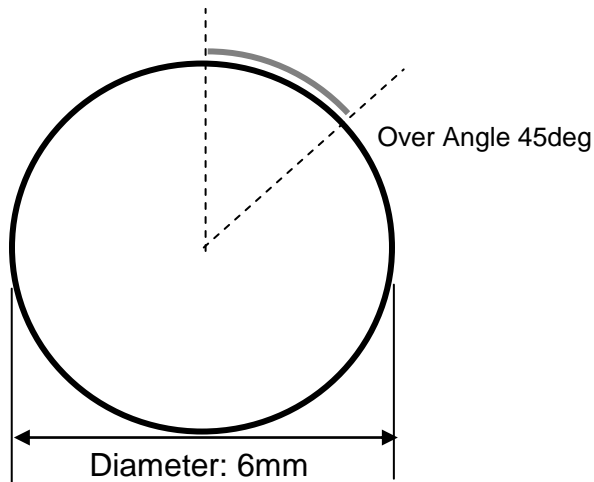
There are four retract forms under this Retract function: Normal, Square, Forward and Forward Square. The Retract forms can be selected after entering the values for: Retract Height, Retract Length and Retract Speed. The value of retract height must be smaller than the value of Z clearance in that point.

If “Cancel” option displayed is selected the tip will retract straight upwards (this could be considered as being the fifth retract form).



#5: ***RETRACT(Over Ang.)***

Over Ang. Function only used for the center function. Please see section 6.1.6 more detailed information



End of the circle lift
the Z axis and
rotates the tip about

I wrote the over angle 45 degree so when the robot finish the circle, it lifts the Z-axis and rotates the tip 45 degree on the circle without dispensing so it is not going to be more material end of the circle

After insert all column, press ENTER, then two addresses are saved.

Retract type: OverAngle Z Lift: 2mm

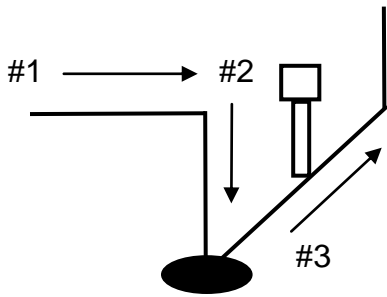
Center X: 100 Y: 100 Z: 100 Diameter: 6

If Over Angle value is zero, Retract step will not be saved.

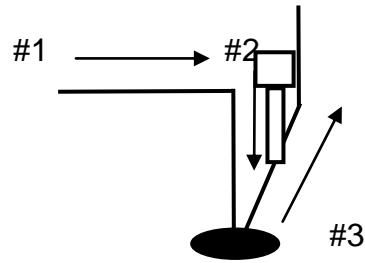
#6: RETRACT(DOT X, DOT Y)

Dot X, Dot Y is retract option for Dispensing Dot Motion.

If programmer need some asymmetrical Dot, this option can be used.



DOT X



DOT Y

5. Adjust Origin

When the dispensing barrel or tip is removed and replaced, the new tip or barrel is often in a slightly different XYZR position than the old tip or barrel was.

The robot has a software utility to adjust a program's origin, thereby correcting the tip offset problem.

A reference should be chosen someplace on the work piece fixture or on the work piece itself. The reference point must be registered in the program data. This only needs to be done one time, for example when the program is originally created.

	<i>Instruction</i>	<i>Display Shows</i>
1	Jog the tip to the reference point (i.e.: X=10, Y=20, Z=30, R=40) OR If the reference point is an existing point in your program, press the MOVE key to bring the tip to that XYZ R location.	
2	Press the Menu1 key, then on page two select 4.ADJUST POSITION to save the location.	Set Adjust Position X: 10 Y: 20 Z: 30 R: 40

When the tip or barrel is changed, use the following procedure to adjust the program origin for the new tip location.

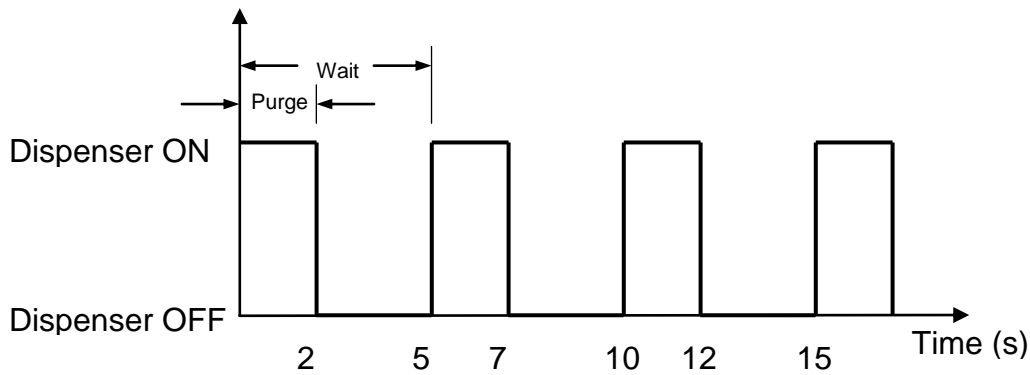
	<i>Instruction</i>	<i>Display Shows</i>
1	While in TEACH mode, press the Menu 2 key, then 2 to select Group Utility menu.	1.Group Edit 2.Expand Step&Repeat 3.Relocate Data 4.Adjust Origin
2	Press 4 to select <i>Adjust Origin</i> .	Move to First Point X: Y: Z: R: Press any key

	<i>Instruction</i>	<i>Display Shows</i>
3	Press any key. The tip will move to the reference point that was recorded in the Adjust Position. If the new tip location is slightly different from the last tip location, you should see that the tip is not exactly at the reference point.	Adjust the First Point X: 10 Y: 20 Z: 30 R: 40
4	Jog the tip to the correct location for the reference point. When the location is correct, press ENT . The program origin will be adjusted for the new tip location.	Program Data Adjusted !!!

6. Auto Purge

After the end of a program, the tip will go to the home position and material will be purged in a continuous loop according to the parameters registered in the Auto Purge Setup command.

This command is very useful for two part materials that have a very short pot life. For example, if a Wait Time of 5s and a Purge Time of 2s is registered in the Auto Purge Setup, the following chart shows the Purge pattern.



Section 6: Point Type & Function Reference

1. *Point Menu*

Below is a list of functions which are found under the **ENTER** key. These functions are 'point-type' functions. The values applied will occupy one memory address.

1.1. *Dispense Dot*

Registers the current XYZR location as a Dispense point for dispensing a dot.

The dispense time and wait time must be set in a previous memory address by registering a ***Point Dispense Setup*** instruction with the **SETUP** key.

The upward motion of the tip after dispensing can be controlled by registering a ***Dispense End Setup*** instruction and / or a ***Z Clearance*** instruction in a previous memory address. ***Dispense End Setup*** and ***Z Clearance*** instructions are registered using the **SETUP** key.

See also ***Section 6:3.3 Point Dispense Setup***, ***Section 6:3.4 Dispense End Setup*** and ***Section 6:3.5 Z Clearance***.

1.2. *Line Start*

Registers the current XYZR location as a Line Start point for dispensing a line.

The line speed must be set in a previous memory address by registering a ***Speed*** instruction using the **SETUP** key.

Dispense delay times used at Line Start point can be controlled by registering a ***Line Dis. Setup*** instruction in a previous memory address. The ***Line Dis. Setup*** instruction is registered by pressing the **SETUP** key.

See also ***Section 6:3.1 Line Speed*** and ***Section 6:3.2 Line Dispense Setup***.

1.3. Line Passing

Registers the current XYZR location as a Line Passing point, this being a location on the line where the tip changes direction, such as at the corner of a rectangle.

1.4. Arc Point

Registers the current XYZR location as an Arc point, this being used to dispense material along an arc or a circular path.

See Section 4: **Programming Example**, for an example of the use of an Arc Point.

1.5. Circle

Registers the current XYZR location as a Circle point in order to dispense material along a circular path. For doing that it is necessary to enter three points on the circle to be dispensed and to register them as: Line Start point (the first point), Circle point (the second point) and Line End point (the third point).

1.6. Center

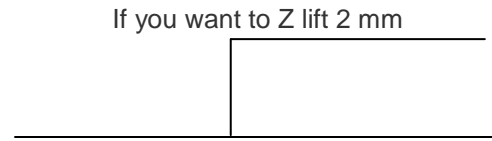
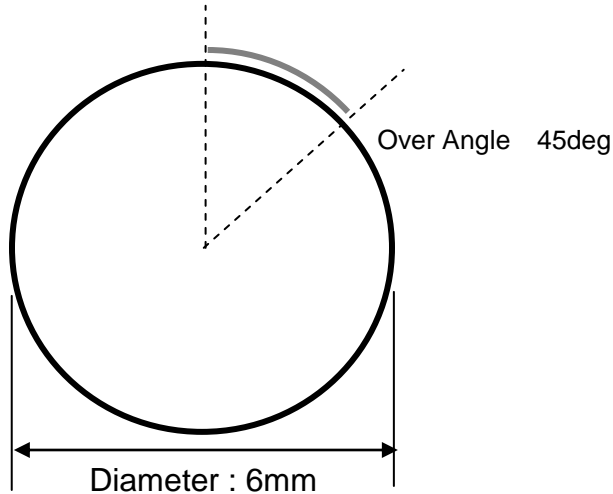
Centers function registers the current XYZR location as a Center point of circle in order to dispense material along a circular path. For doing that it is necessary to enter diameter, over angle and lift Z. Value.

Over Angle means that after dispensing the circle, while going through an extra part of the circle defined by the Over Angle, the dispenser will be turned OFF

If you want to lift the Z at the end of the dispense, input Z lift value bigger than zero.

Center

Diameter:	6 mm	{Type the diameter of the circle}
Over Angle:	45 degree	{Type the angle between the 1 to 360}
Z lift:	2 mm	{If you want to lift the Z end of the dispense}
Mode	0 (NONE)	{No tool correction}
	1 (Outer)	{Tip draw circle on the outside of circle}
	2 (Inner)	{Tip draw circle on the inside of circle}



End of the circle lift
the Z axis and
rotates the tip about

I wrote the over angle 45 degree so when the robot finish the circle, it lifts the Z-axis and rotates the tip 45 degree on the circle without dispensing so it is not going to be more material end of the circle

After insert all column, press ENTER, then two addresses are saved.

Retract type: OverAngle Z Lift: 2mm

Center X: 100 Y: 100 Z: 100 Diameter: 6

If Over Angle value is zero, Retract step will not be saved.

Mode option is used for robot has a Rotation Axis.

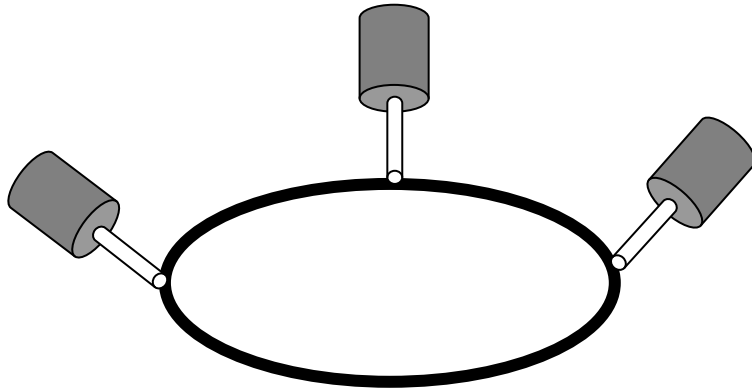
Mode option can be used after tool correction

See **Section 5.3. Auto Tool Setting** for a detailed description of Tool correction.

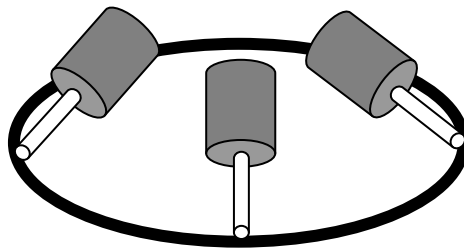
This function is used to prevent barrel from crashing by inner or outer obstacle.

if circle has an obstacle inside of the circle or outside of the circle, the barrel must be crashed by obstacle.

CASE 1: there is an obstacle inside of the circle.
Set the Mode as a 1.Outer. Then tip will move like next figure.



CASE 2: there is an obstacle outside of the circle.
Set the Mode as a 2.Inner. Then tip will move like next figure.



1.7. Line End

Line end registers the current XYZR location as a Line End point.

The dispense setting used at the end of the line can be controlled by registering a **Line Dis. Setup** instruction in a previous memory address. The **Line Dis. Setup** instruction is registered by pressing the **SETUP** key.

See also **Section 6:3.2 Line Dispense Setup**.

The upward motion of the tip after dispensing can be controlled by registering a **Dispense End Setup** instruction and / or a **Z Clearance** instruction and/or a **Retract** instruction in a previous memory address.

Dispense End Setup and **Z Clearance** instructions are registered using the **SETUP** key.

1.8. Dummy

Registers the current XYZR location as a Dummy point, where the tip will simply pass through this point. A dummy point is useful for avoiding obstacles on the work piece.

1.9. End Program

Registers the current memory address as the end of the program. The End Program instruction will cause the tip to return to the home position at the end of the program cycle.

1.10. Dispense ON / OFF

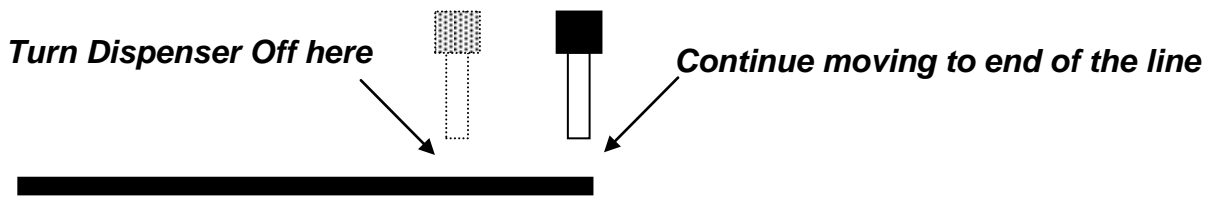
The Dispense ON / OFF instruction will allow the user to program an instruction which will turn the dispenser ON or OFF.

This is useful for turning the dispenser OFF before the end of a line in order to prevent an excess of material at the end of that line.

Problem: Too much material at the end of the line



Solution: Turn the dispenser OFF before the end of the line



To register a Dispense OFF instruction, jog the tip to the XYZR location where you want the dispenser OFF, enter that location as a line passing point, then press the **ENTER** key, select **Dispense OFF** and press the **ENTER** key again.

Press **1** to select dispenser ON or press **2** to select dispenser OFF.

Please note that: ARC point does not recognize the **Dispense ON/OFF** functions. In order to use dispenser ON/OFF function in the ARC, you must insert dispenser ON/OFF function before the Arc.

1.11. Home Point

Registers an instruction to 'home' all axes, sending them to the home position. See the **Setup** Menu for instructions on changing the location of the home position.

1.12. Wait Point

Registers a Wait time. When executed, the tip will wait for the specified period of time.

1.13. Stop Point

Registers a Stop Point at the current XYZR location which must be entered as a Line Passing point. When executed, the tip will move to that location and wait until the start button is pressed.

1.14. Brush Area

Brush Area causes the tip to 'paint' the defined area
There are two Brush area shapes: rectangle and circle.

Brush Area: Rectangle

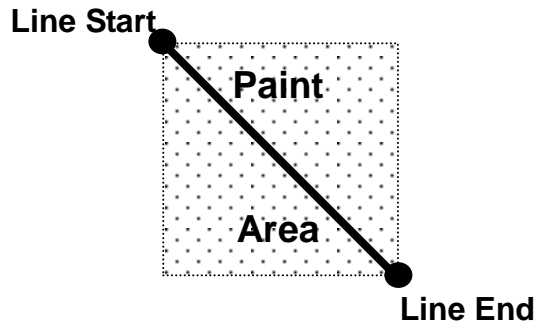
Press **ENT**, and then press the **PgDn** key to scroll down to page 4/5 of the **Point** Menu.
Select **Brush Area**, then press **1** to select **Rectangle**

You will be prompted to enter the *brush width* and the *brush distance*. The brush width is the distance between two consecutive passing of the tip during brushing.

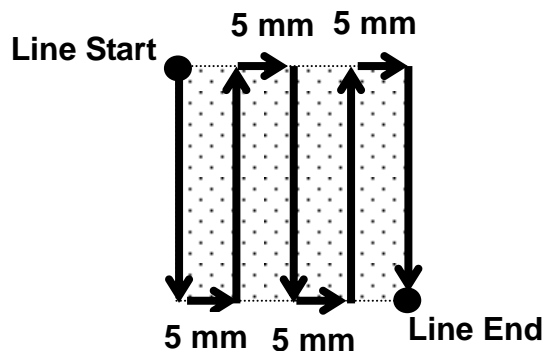
The brush distance is the distance between the first and the last passing of the tip during brushing.

Enter the value you wish to use for the brush width and brush distance and press **ENT**.
(If "0" is entered for the brush distance, the entire area of the rectangle will be brushed).

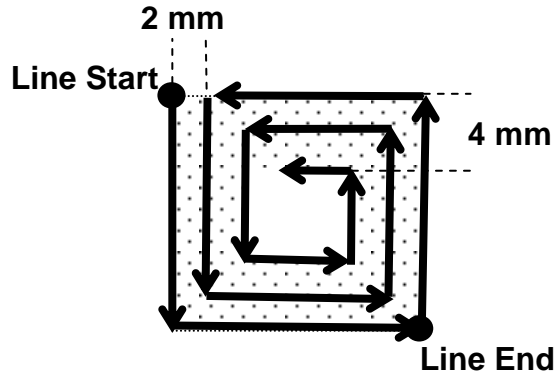
After registering the brush width and the brush distance, teach a Line Start point at the top left corner of the area to be brushed and a Line End point at the bottom right corner of that area (the tip will not dispense a straight line between these two points):



If, for example, a brush width of 5 mm and brush distance of 0 mm are entered, the tip will take the following path when the program is run:



If for example, a brush width of 2 mm and brush distance of 4 mm are entered, the tip will make the following path when the program is run:

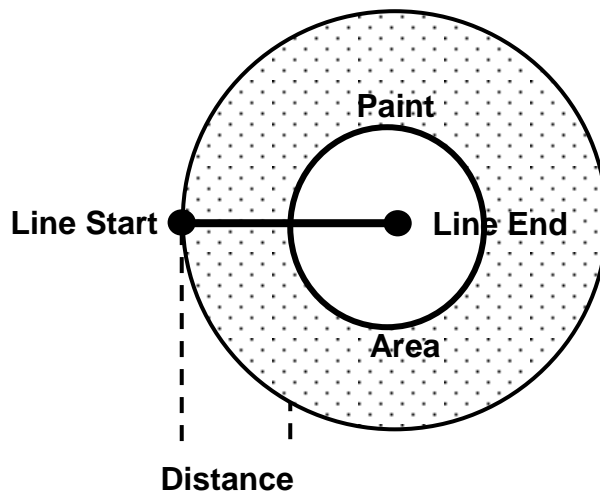


Brush Area: Circle

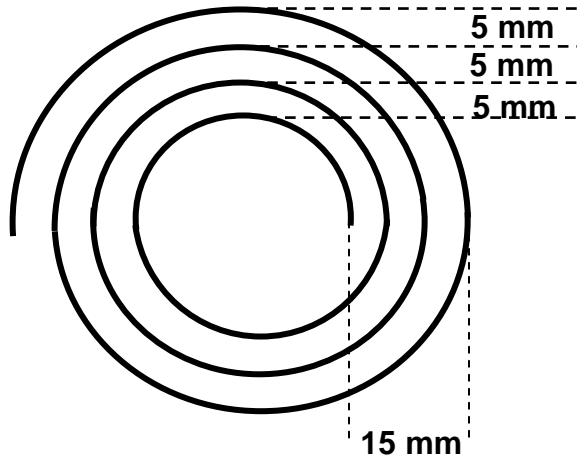
Press **ENT**, and then press the **PgDn** key to scroll down to page 4/5 of the **Point** Menu. Select **Brush Area**, then press **2** to select **Circle**.

You will be prompted to enter the *brush width* and the *brush distance*. The brush width is the distance between two consecutive passings of the tip during brushing. The brush distance is the distance between the first and the last passing of the tip during brushing. Type the value you wish to use for the brush width and brush distance and press **ENT**. (If "0" is entered for the brush distance, the entire area of the circle will be brushed).

Then teach a Line Start point on the exterior line of the circular surface to be brushed and a Line End point in the center of that surface (the tip will not dispense a straight line between these two points):



If for example, a brush width of 5 mm and brush distance of 15 mm are entered, the tip will take the following path when the program is run:



1.15. *If*

If registers is an instruction that either sets the value of an output signal or checks the status of an input signal.

If *If* is selected, the user can enter the input port (input # 0 – 31), the input status (1 for open or 0 for closed) and the address to go to or Label if that input status occurs.

1.16. *Output*

If *Output* is selected, the user can enter the output port (output # 0 – 31), and whether the output should be turned **ON** or **OFF**.

1.17. *Input*

If *Input* is selected, robot waits till the status of an input signal is set at user's selection.

1.18. *Pulse*

Register Output Port to be turned ON and Output time. After Output time, Output Port will be turned OFF.

1.19. *Point*

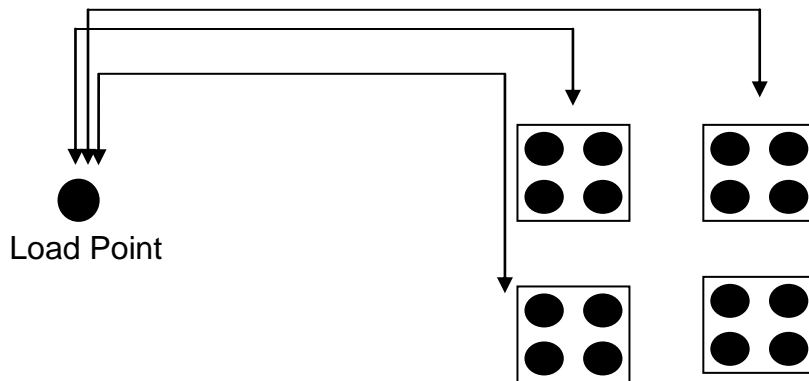
Point sets Point Variable P0 – P99 by saving current position or input numerical data.

- ※ P99 is current position.
P98 is used for Auto Tip Align.

1.20. Load Point

Registers the current XYZ location (and R position of tip) as a Load point. The tip will simply move this point. The point will not be affected by 'Step & Repeat' or 'Offset' instruction

Address	Instruction
0	Load Point
1	Dispensing Dot
2	Dispensing Dot
3	Dispensing Dot
4	Dispensing Dot
5	Step & Repeat from address 0



2. Condition Menu

2.1. Goto Address

Goto Address causes the program to jump to the specified memory address or Label.

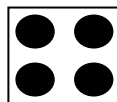
2.2. Step & Repeat Y

Step & Repeat Y allows a group of instructions to be run repeatedly, stepping a given distance in the X axis or Y axis between each cycle.

Step & Repeat Y is useful when a fixture is mounted on the robot that holds many identical work pieces aligned in rows and columns. The user needs only to create a program for the first work piece in the fixture, and then use the Step & Repeat function to dispense to the other work pieces.

The Step & Repeat function will allow the user to define the number of rows, the number of columns, the X offset between each part and the Y offset between each part.

If, for example, we have a program (for instance: program number 20) which has to dispense four dots of material on a work piece:

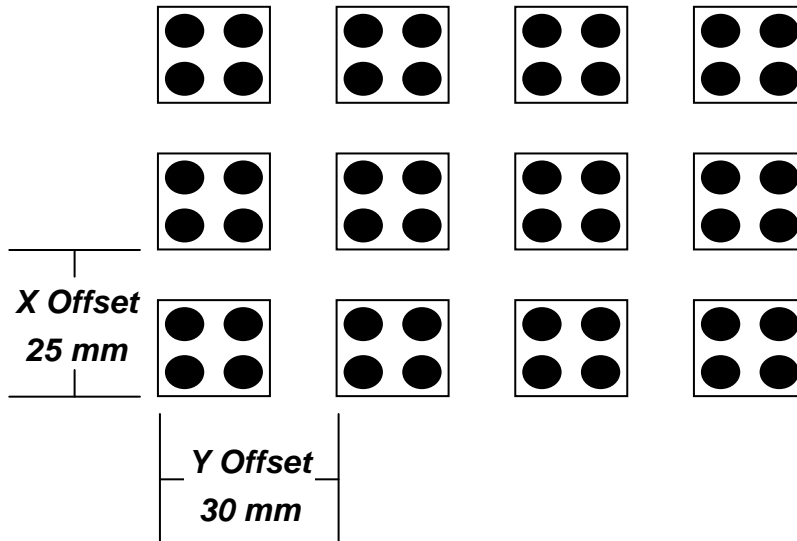


The program would consist of the following eight instructions:

Address	Instruction
0	Dispense End Setup: H.Speed = 100 mm/s, L.Speed = 15 mm/s, L.Length = 5mm
1	Z Clearance: Relative 10 mm
2	Point Dispense Setup: Dis.Time = 0.25 s Tail Time = 0.10 s

3	Dispense Dot (the first dot)
4	Dispense Dot (the second dot)
5	Dispense Dot (the third dot)
6	Dispense Dot (the fourth dot)
7	End Program

If a fixture is made to hold (12) work pieces, in four columns by three rows:



A Step & Repeat Y instruction can be used to repeat the program at the additional (11) locations.

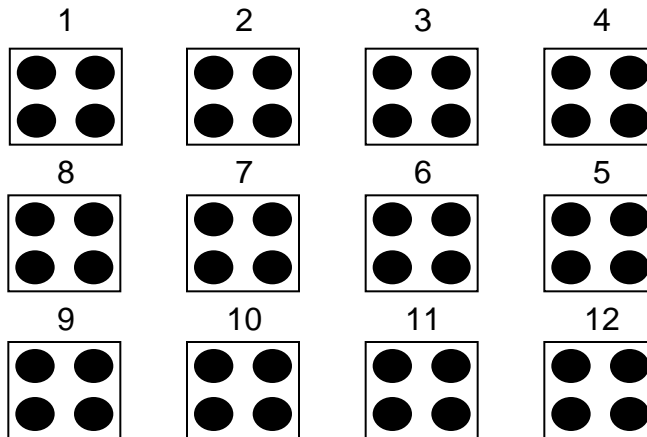
The instruction at memory address 7 should be changed from End Program to Step & Repeat Y. To register a Step & Repeat Y instruction at memory address 7, it is necessary to be done the following instructions:

	<i>Instruction</i>	<i>Display Shows</i>
1	Press the ▼/+1 or ▲/ -1 keys until memory address 7 is shown in the display	ADDR:7 PROG:20 End Program

	<i>Instruction</i>	<i>Display Shows</i>
2	Press the F2/Cond key, select Step & Repeat Y , select the Address, enter the number of the start address and press ENT . The start address is the memory address of the first instruction which is part of this Step & Repeat group. In our example, we want to repeat all instructions starting with memory address number 3. Enter 3 to specify this memory address and then press ENT .	Step & Repeat Y Columns (X) : Rows (Y) :
3	Enter 3 again, this time to specify the three locations on the X axis direction and then press ENT . Although we are prompted here to enter the number of the columns, we will have to take into account the significance of the X axis displayed together with the word “columns” i.e. we will have to enter the number of locations on the X axis direction.	Step & Repeat Y Columns (X) :3 Rows (Y) :
4	Enter 4 to specify the four locations on the Y axis direction and then press ENT . Although we are prompted here to enter the number of the rows, we will have to take into account the significance of the Y axis displayed together with the word “rows” i.e. we will have to enter the number of locations on the Y axis direction.	Step & Repeat Y X Offset: (mm) Y Offset: (mm)
5	In the above example, the Y Offset between parts is 30 mm. Type 30 to specify an Y Offset of 30 mm, then press ENT .	Step & Repeat Y X Offset:30 (mm) Y Offset: (mm)
6	In the above example, the X Offset between parts is 25 mm. Type 25 to specify an X Offset of 25 mm, and then press ENT .	Step & Repeat Y 1. N Path 2. S Path Select:

	<i>Instruction</i>	<i>Display Shows</i>
7	<p>The display will prompt you to select between: 1. N Path and 2. S Path. Selecting N Path will determine the first row (columns 1 to 4) to be dispensed first, then the second row (columns 1 to 4), then the third row (columns 1 to 4). Selecting S Path will determine the first row (columns 1 to 4) to be dispensed first, then the second row (this time columns 4 to 1), then the third row (columns 1 to 4). Press 2 to select S Path.</p>	<p>ADDR: 8 PROG: 20</p> <p>EMPTY</p>
8	<p>The program is now complete. Press ENT, then select End Program to mark address 8 as the new End Program instruction.</p>	<p>ADDR: 9 PROG: 20</p> <p>EMPTY</p>

The program will run in the following pattern and consists of the following instructions:

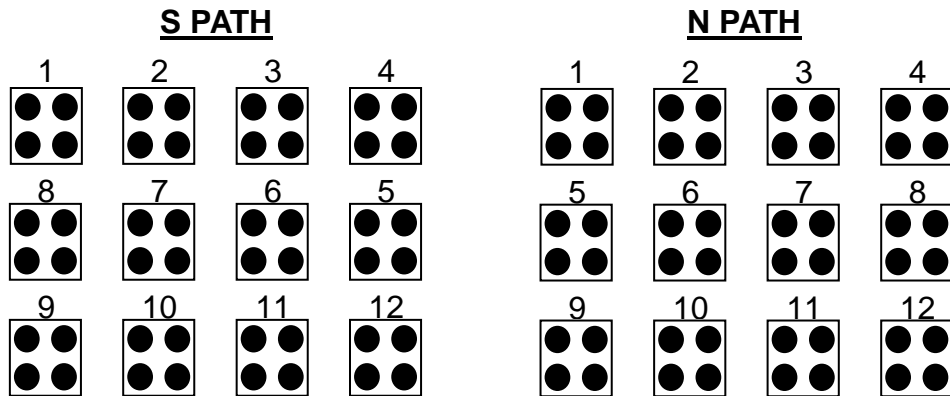


<i>Address</i>	<i>Instruction</i>
0	<p>Dispense End Setup: H.Speed = 100 mm/s, L.Speed = 15 mm/s, L.Length = 5mm</p>
1	<p>Z Clearance: Relative 10 mm</p>

2	Point Dispense Setup: Dis.Time = 0.25 s Tail Time = 0.10 s
3	Dispense Dot
4	Dispense Dot
5	Dispense Dot
6	Dispense Dot
7	Step & Repeat Y: Cols(X):3, Rows(Y):4, X Off: 25mm, Y Off: 30mm, Addr:3, S Path
8	End Program

The above example was done using S Path. The difference between S Path and N Path is the order in which the pieces are run:

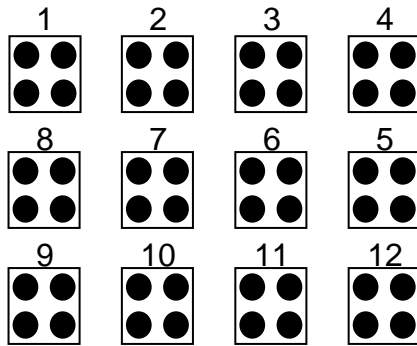
Step & Repeat Y:



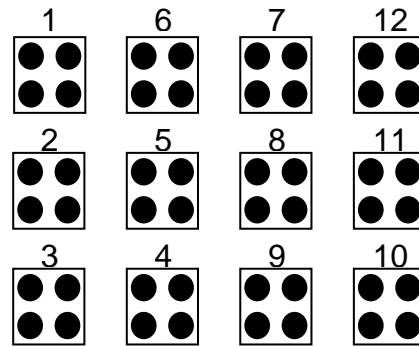
2.3. Step & Repeat X

Step & Repeat X works just as Step & Repeat Y does, with one difference: priority is given to the X axis instead of the Y axis.

Step & Repeat Y - S PATH



Step & Repeat X - S PATH



2.4. Call Subroutine

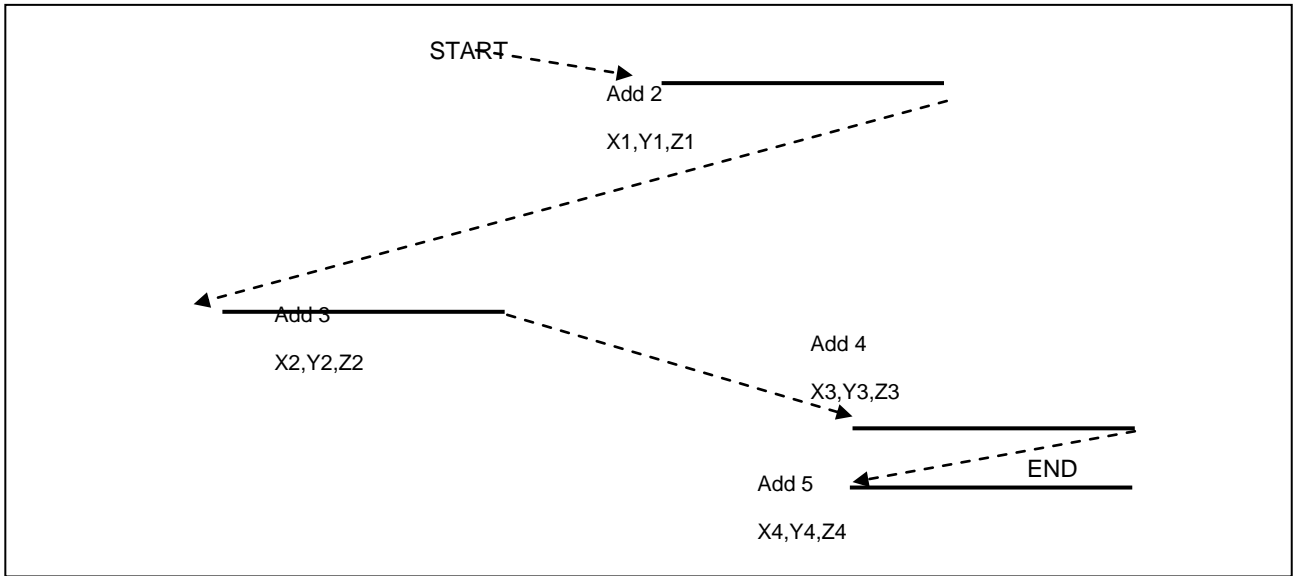
Call Subroutine causes the machine to jump to a specified memory address and execute the instructions found there using coordinates specified at the **Call Subroutine** instruction. When the *End Program* instruction for the subroutine is reached, program execution will continue at the address immediately after the **Call Subroutine** instruction.

The **Call Subroutine** function is most useful to repeat a pattern anywhere on the work-piece, as opposed to the *Step & Repeat* function where the pattern must be repeated in straight lines, at fixed distances from each other.

The following example illustrates the use of the *Call Subroutine* instruction. An explanation follows.

Address	Instruction
0	Line Speed = 20
1	Call Subroutine (X1,Y1,Z1) address 6
2	Call Subroutine (X2,Y2,Z2) address 6
3	Call Subroutine (X3,Y3,Z3) address 6

4	Call Subroutine (X4,Y4,Z4) address 6
5	End Program
6	Line Start (Xs,Ys,Zs)
7	Line End (Xe,Ye,Ze)
8	End Program



2.5. Call Program

Call Program will jump to the specified program number and execute the program data in the destination program until the End Program command is reached. When the destination program is executed, the robot will return to the calling program.

2.6. *Loop Address*

Registers an instruction that will execute a group of instructions a user-specified number of times.

When the **Loop Address** instruction is registered, the display will prompt for an **Address/Label** and **Count**.

Address is the memory address to jump to from the current address. This address must be less than the current memory address.

Label is the address name to jump to from the current address. This address must be less than the current memory address.

Count is the number of times to execute the loop.

2.7. *Label*

Label is used instead of Address Number. A maximum of 64 Labels is permitted per program, and each label can have up to 8 characters.

2.8. *Arm*

Arm command is used both for TMB Series and for F9000N Series. The use of this command for TMB Series is different from that for F9000N Series.

In case of TMB Series, when robot moves with the command of **Line/Arc**, **Arm** command determines position of the first and the second arm of the robot.

When **Arm Left** command is inserted before **Line** move command, the robot will move **Line/Arc** in the minus direction of the second arm. On the contrary, **Arm Right** will move in the opposite direction.

To disable **Arm** command, **Arm No** command is used.

In case of F9000N Series, it allows to select between XYZ interpolation and RYZ interpolation. **Arm Left/No** selects XYZ interpolation. **Arm Right** selects RYZ interpolation.

2.9. *FixR*

FixR command is used for **rotational axis** which 4-axis robot.

FixR Fix indicates that during **Line/Arc** movement R axis is fixed in the same direction.

FixR No indicates that R axis moves to the point whose position was saved while robot was moving in the **Linear/Arc** interpolation.

2.10. *Calculation*

For calculation, F9000N Series have 4 types of variables:

100 Integer Variables (I0 ~ I99),

100 Float Variables (F0 ~ F99),

2 Timer Variables (T0,T1) ,

100 Position Variables (P0 ~ P99).

All numeric formula must have a format like this:

Variables = Variables (OK)

or

Variables = Variables (Operator) Variables (OK).

But it is not allowed:

Variables = Variables (Operator) Variables (Operator) Variables (NO)

Example1)

I0=1

I1=I0+1

F0=I1+123.45

F1=F2/3

P0=P1

But it is not allowed:

I0=I1+I2X I3 (NO)

I4=I2X I3

I0=I1+I4 (OK)

Position Variable can be set like this:

Example2)

P0=P1

P0.X=P3.X+13

P4.Y=I3/10

P3.Z=123.45

It is possible to use array.

Only Integer Variable can be Index for array

Example3)

I0=5

F[I0]=3 (F5=3)

P[I1].X=123.4

Point Variable is used for IncJ, IncL, Jmov, Lmov

Example4)

Jmov P0

Jmov P[I0]

Variable I, F, T can be used for If command

Example5)

IF I < 50

IF F[I10] = 10

IF T0 < 100

IF I1 < I2

1. Input sequence for “I10=I11+123.45”.

	<i>Instruction</i>	<i>Display Shows</i>
1	Press Cond key and then press PgDn/+10 key until page 3/5 is displayed.	1. FixR 2. Calc 3. Jmov 4. Lmov
2	Select Calc command, by pressing 2 key.	Select Type ----- █ I F P T
3	Select Variable Type, by pressing F1/Setup key corresponding to the letter “I” on the display.	Input Number ----- I█ [I]
4	Input Variable Number (10) by Numeric key and then press ENT.	Select Type ----- I10=█ I F P T
5	Select Variable Type, by pressing F1/Setup key.	Input Number ----- I10=I█ [I]
6	Input Variable Number (11) by Numeric key and then press ENT.	Select Operator ----- I10=I11█ + - * /
7	Select Operator, by pressing F1/Setup key for “+”. If there is no need to use operator, press ENT.	Select Type ----- I10=I11+█ I F P T
8	Input Number (123.45) by Numeric key and then press ENT.	Input Number ----- I10=I11+123.45█ [I]

You can set variables in Menu1.

	<i>Instruction</i>	<i>Display Shows</i>
1	Press Menu 1 key and then go to page 4 of the Menu.	1.Plc File Edit 2.I/O Monitor 3. Set Variable 4. Emg Mode
2	Select 3 Set Variable from the above mentioned page. First view is Integer Variables.	I0 [1]INT I1 [2] I2 [13] I3 [2]
3	To see Float Variables, press F2.	F0 [1.2]REAL F1 [2.4] F2 [13.5] F3 [2.0]
4	To see POINT Variables, press F3 <i>i.e. press Inch/Menu1 key located under F3 sign.</i>	B0 [123.0]BUF B1 [200.1] B2 [13.2] B3 [2.5]
5	To Change value, Input Value by numeric key and press ENT.	B0 [23.0]BUF B1 [200.1] B2 [13.2] B3 [2.5]
6	To see another page of the Menu, press PgDn/PgUp.	I4 [1]INT I5 [2] I6 [13] I7 [2]
7	To jump specified Variable, press F4 (i.e. Mode/Menu 2 key) and input variable Number by numeric key.	Select Variable Num Var NO[]

If Command can use variables. Here is a useful example:

If I0 < 100

If F[I10] < 123.45

If I12 < I13

	<i>Instruction</i>	<i>Display Shows</i>
1	Select ENT key and find If command and select it.	If ----- 2. Address 3. Label
2	Select 1 Address and input address number, then press ENT.	Select Type ----- Input I F T
3	To use user input, press F1/Setup key (for "Input"). If not, press Function key to select Type (I:F2 F:Inch T:Mode).	If Input ----- Input No: On/Off (1/0)
4	Input Variable Number by pressing numeric key and then ENT key.	Select Operator ----- I12 = < > !
5	Select Compare Operator. = (Equal), <(Less Than), >(Greater than),!(Not Equal)	Select Compare Type ----- I12< I F
6	Press Function key to select Type (I:F1 F:F2)	Input Number ----- I12<I [I]
7	Input Variable Number by pressing numeric key.	Input Number ----- I12<I13 [I]

2.11. *Jmov/Lmov*

This command moves to Position which is saved in the Point Variable.
In case of Jmov P0, robot moves to the point that is stored in P0.

2.12. *IncJ / IncL*

IncJ/IncL command moves robot to the absolute position resulted from adding a relative position specified as an argument to the current position.

2.13. *Offset*

Set Offset value or Point number. All saved position after Offset command will be increased by Offset Value(except 'Load Point').

Offset Value can be edited by numeric key of Point position

ex) Z Axis offset 10mm.

Offset X:0 Y:0 Z:10 R:0

Ex) Offset with Point P0. Increase Offset of Z Axis

P0.X = 0

P0.Y = 0

P0.Z = 0

P0.R= 0

Labl A

P0.Z=P0.Z+5

Offset P0

Loop Address 3

2.14. PALLET

Calculate Pallet position and input the result to Position Variable.

First of all, input the number of columns and rows of pallet.

Three corner points of the Pallet are saved as three continuous Position Variables.

The first point is called "Base Point".

Base Point can be selected from P0 to P97.

If Base Point is in P1, last unit position of column direction is saved in P2. Last unit position of row direction is saved in P3.

Set Point is Position Variable that will save Pallet position calculation results.

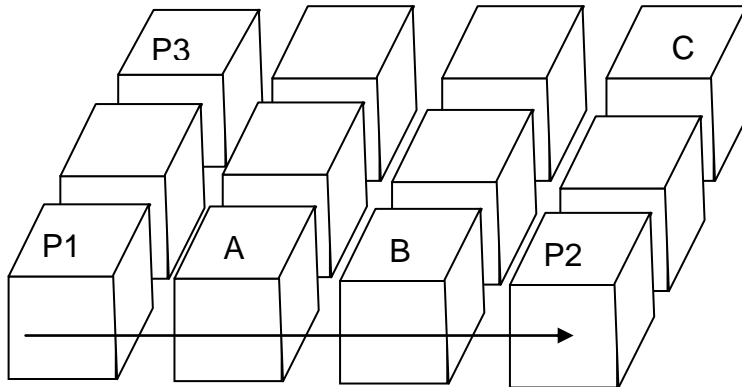
Input index variable to determine the work sequence number. Index variables use integer variables. Index variables can be selected from I0 to I99.

Next figure shows: 4 X 3 Pallet. Base Point is P1, Set Point is P0, Index is I0.

If value of I0 is 0, P0 has position of P1. If I0 is 1, P0 saves position of unit A. If I0 is 2, P0 saves position of unit C.

Value of I0 is not added automatically. Program must have command like: "I0=I0+1".

To save position of P1, P2, P3:



	<i>Instruction</i>	<i>Display Shows</i>
1	Move to P1 position and press ENT. Select Point Command (page 5 of the Menu).	Point Point No :
2	Input "1" by numeric key and press ENT.	Point No 1 X:0 Y:0 Z:0 R:0 [Cur]
3	Display shows current saved position of P1. To save current position, press F1. To edit current saved position, insert new value by numeric key. Then press ENT key	----- Current Position Saved -----
4	Current position is saved for P1. For P2, P3 to be used the same way.	

<i>Address</i>	<i>Instruction</i>
1	I0=0
2	Jmov P4
3	Pallet Row:4 Col: 3 Base Point: P1 Set Point P0 Index I0
4	Jmov P0
5	I0=I0+1
6	If I0<12 Goto Address 2
7	End Program

	<i>Instruction</i>	<i>Display Shows</i>
1	Press Cond key and select Pallet Command (page 4 of the Menu).	Pallet ----- Columns :4 Row :3
2	Input column, row number.	Pallet ----- Base Point : Set Point : Index :
3	Input Base Point number.	Pallet ----- Base Point :1 Set Point : Index :
4	Input Set Point number.	Pallet ----- Base Point :1 Set Point :0 Index :
5	Input Index number.	Pallet ----- Base Point :1 Set Point :0 Index :0

2.15. Pattern / Pattern End

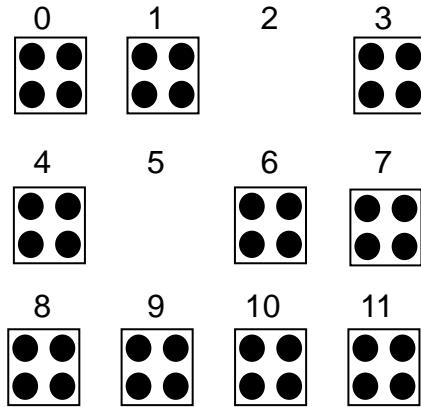
Pattern command works like “Step & Repeat X/Y” command, but it is different only because it can skip or change the order of the repeat.

Pattern command is useful for arranging similar shapes.

At the end of Pattern, to be inserted Pattern End Command.

Pattern command has Index Variable. Index Variable can change the order of the repeat. Index number can be used form I0 to I99.

For example, if the third (number 2 below) and the sixth (number 5 below) repeats must be skipped:



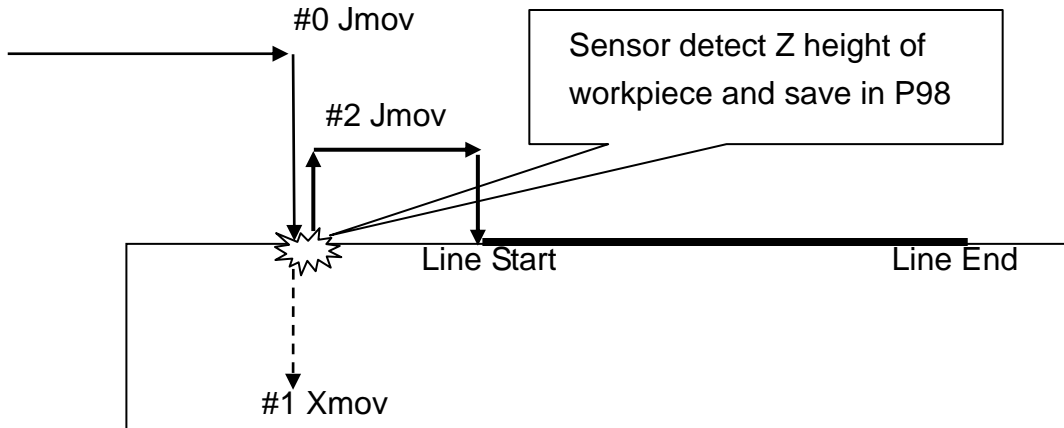
Address	Instruction
0	I0=0
1	Pattern X Cols(X): 3, Rows(Y): 4, X Off: 25mm, Y Off: 30mm, Index IO
2	Dispense Dot
3	Dispense Dot
4	Dispense Dot
5	Dispense Dot
6	Labl A
7	I0=I0+1
8	IF I0==2 Goto Labl A (skip third work)
9	IF I0==5 Goto Labl A(skip sixth work)
10	Dispense Dot
11	Pattern End
12	End Program

	Instruction	Display Shows
1	Press Cond key and select Pattern command.	Pattern ----- X Offset: Y Offset:
2	Insert X Offset, Y Offset.	Col : Row : Index:
3	Input col, row value and Index Number.	1. X Start 2. Y Start
4	Select X Start or Y Start	

2.16. Xmov

Xmov can be used for sensor detecting of different work piece.

If sensor input is on while moving, robot will stop immediately and save current position in P98.



Address	Instruction
0	Jmov (0,0,0)
1	Xmov (0,0,100) In0 sensor input 0
2	Jmov (0,0,10)
3	P0.X = 0
4	P0.Y = 0
5	P0.Z= P98.Z- <u>50</u>
6	Offset P0
7	Line Start (10,0, <u>50</u>)
8	Line End (100,0, <u>50</u>)

Ex) if Z height of work piece is 45mm, after sensor detection, value of P98 is(0,0,45)

So value of P0 is (0,0,-5)

By offset command, Line Start is (10,0,45) and Line End is (100,0,45)

2.17. COMM

COMM Command can send string through RS232c port.

'COMM abcd' command send 'a"b"c"d' character.

3. Setup Menu

Below is a list of functions that are found under the **F1/ SETUP** key. These functions are all related to the setup of dispensing parameters.

3.1. Line Speed

Registers the LINE SPEED used for all lines from the current memory address forward until another Line Speed instruction is found.

3.2. Line Dispense Setup

Registers the LINE DISPENSE SETUP values, which set the dispensing wait time at the start of lines ('head' time), waiting time at the end of lines ('tail' time), dispense off distance ("tail length") and output port number for Line and Arc dispense ("output"). The registered values will be used from the current memory address forward until another Line Dispense Setup instruction is found.

See **Section 5:3. Line Dispense Setup** for a detailed description of this function.

3.3. Point Dispense Setup

Registers POINT DISPENSE SETUP values which set dispensing time and waiting time at the end of dispensing ('tail' time) for dots, output port number for dot dispense ("output") The registered values will be used from the current memory address forward until another POINT DISPENSE SETUP instruction is found.

3.4. Dispense End Setup

Dispense End Setup allows the **L. Length**, **L. Speed** and **H. Speed** values to be registered at a memory address. These values will affect how far and how fast the tip rises after dispensing. Please see **Section 5:1. Dispense End Setup** for a detailed description of this function.

3.5. Z Clearance

Z Clearance command sets the distance the tip will rise after dispensing, in order to avoid obstacles. It is registered at a memory address. Please see **Section 5:1. Dispense End Setup** and **Section 5:2. Z Clearance** for a detailed description of this function.

3.6. X/Y Move Speed

X/Y Move Speed sets default X and Y axes movement speed as the tip moves between figures in a program, such as from one dispensing dot to another or from the end of one dispensing line to the start of the next dispensing line.

3.7. Z Move Speed

Z Move Speed sets default Z axis movement speed as the tip moves between figures in a program, such as from one dispensing dot to another or from the end of one dispensing line to the start of the next dispensing line.

3.8. Home Position

Home Position allows the user to change the location of the program home position. The home position is the location to which the tip will move at the end of a program cycle.

To change the home position, jog the tip to the desired location for the new home position, press the **Setup** key, then press



and select **Home Position**.

NOTE: When executing a program in TEACH MODE, the robot will move to the mechanical home position (X=0, Y=0, Z=0, R=0) at the start of every program cycle. When executing a program in RUN MODE, the robot will move to the home position set under the Setup: Home Position.

3.9. Retract

The *Retract* function causes the tip to reverse direction backward along the dispensing path, and upward after a line dispensing. This is useful when dispensing high viscosity or 'stringy' materials, as it will lay the material tail down on the dispensed bead.

Please see **Section 5: 4.Retract** for additional information on the retract function.

3.10. Auto Purge

The *Auto Purge* function prevents the dispensing material to become harden while robot is running. If there is no dispensing during 'Wait time', the robot will purge the material by 'Purge time'.

3.11. Adjust Point # 1

Saves current position and steps as a first data for Relocate Data function. Saves current position to temporary Point # 1.

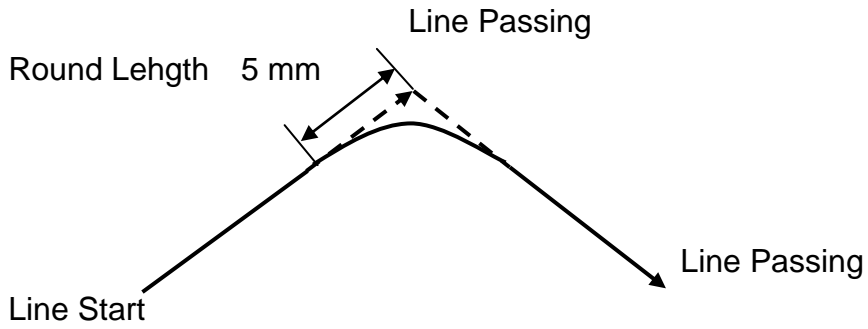
3.12. Adjust Point # 2

Saves current position and steps as a second data for Relocate Data function. Saves current position to temporary Point # 2.

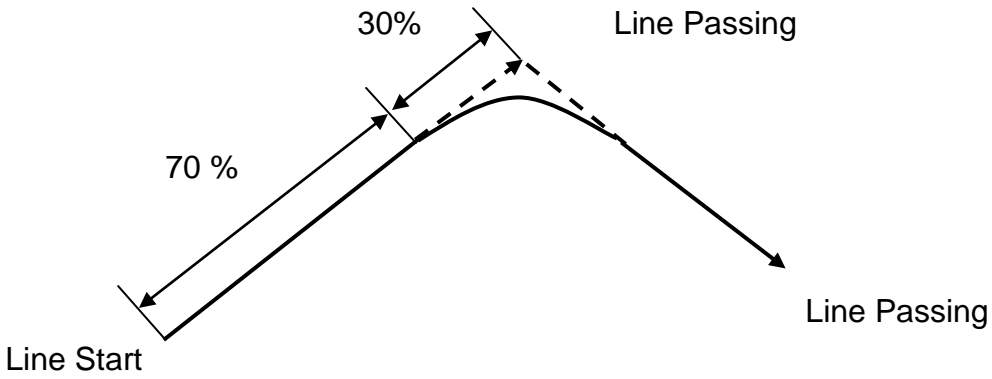
3.13. Round

A continuous path motion follows a path that smoothly connects two separate motions of a robot without stopping by inserting a round path between the two motions. In order to make such a smooth motion, a robot assumes that it has already moved to the target point if the robot has moved to a certain range of the first target point and executes motion plan for the next target point. It is Round commands that are used for setting the range

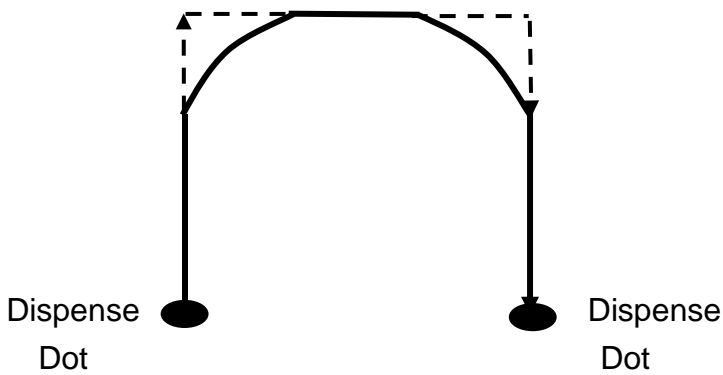
If value of Round Length is 5mm



If value of Round Percent is 70%



Round Percent also affect dispense dot motion.

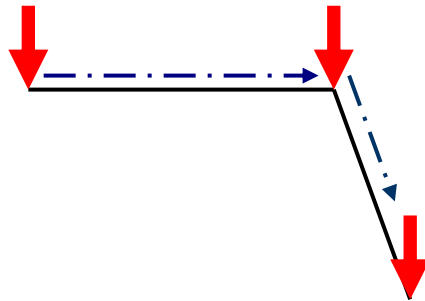


3.14. Z Lift

The Z-Lift function allows the user to move through the program from point to point without raising the Z-axis. The function also allows the user to set the distance the Z-axis will move before going to the next point. This function is especially useful when adjusting pre-existing points which are defined but are not accessible through direct Z-axis movement. The default value is OFF.

1. Lifting Z : **OFF**

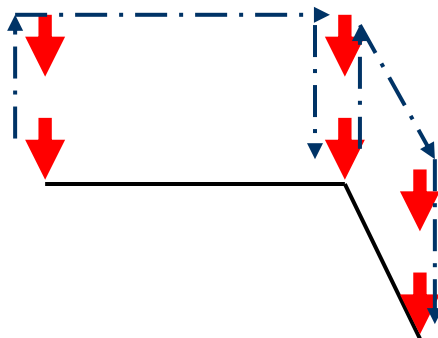
When Lifting Z is off, moving the tip from point to point will require the Z-axis to rise according to the specified lifting length. If lifting length is set to zero, the tip will not do a Z-axis movement to get to the next point.



Z-axis lift is **off** with the lifting length set to zero.

2. Lifting Z : **ON**

When lifting Z is on, moving the tip from point to point will require the Z-axis to rise to its maximum height ($Z = 0$) before traveling the XY plane to get to the next point.



Z-axis lift is **on** so the tip moves to the home position first before going to the next point.

3.15. *Move Cond*

Setting Line Speed, XYMove Speed, Z Clearence(Absolute), Round Value in One Command.

3.16. *Disp Cond*

Setting Dispensing time, Head time, Tail time, Head Length, Tail Length in One Command.

4. Menu 1

4.1. Program Name

Program Name allows the user to register a name for the current program. If a program name is registered, it will appear on the display when that program is selected in Run mode. The name of a program can be registered only after that program has been created.

4.2. Z Axis Limit

Z Axis Limit allows the user to limit the range of the Z axis values.

Press **Menu 1** key and select 2 (on page 1/5) for Z Axis Limit. You will be prompted to enter the two new values for the Z axis range.

When you are prompted to enter the High Value, you will have to enter the desired value of the Z axis in its higher position, which will be – in fact – the lower value of the new Z axis range.

When you are prompted to enter the Low Value, you will have to enter the desired value of the Z axis in its lower position, which will be – this time – the higher value of the new Z axis range.

The Z axis range will be limited to these new values.

4.3. Initial Output

Initial Output Status sets the ON/OFF status of the output signals at the start of each program cycle.

Initial Output Status value is the hexadecimal representation of an 8 character value controlling the 32 output signals.

For example,

<i>HexaDecimal Value</i>	<i>Output Status</i> (1 = on, 0 = OFF)							
	Bit31							Bit 0
0	0000	0000	0000	0000	0000	0000	0000	0000
1	0000	0000	0000	0000	0000	0000	0000	0001

2	0000	0000	0000	0000	0000	0000	0000	0010
4	0000	0000	0000	0000	0000	0000	0000	0100
8	0000	0000	0000	0000	0000	0000	0000	1000
A	0000	0000	0000	0000	0000	0000	0000	1010
B	0000	0000	0000	0000	0000	0000	0000	0000
AAAA	0000	0000	0000	0000	1010	1010	1010	1010
555555	0000	0000	0101	0101	0101	0101	0101	0101

4.4. Cycle Counter

When in Run mode, the third line of the display shows the number of program cycles that have run to completion for the current program number:

Cycle Counter : X

It is possible to reset this number by selecting **Menu 1, Cycle Counter**. When prompted for the **New Value:**, press **0** and **ENT** to reset the cycle counter.

It is also possible to cause an alarm to be generated after a fixed number of program cycles. Select Menu 1, **Cycle Counter**. When prompted for the **New Value:**, type the number of program cycles to complete before generating the alarm and press **ENT**.

In Run mode, when the set number of program cycles has been completed, the display will show **Counter Full!** And the start button will be disabled. The Cycle Counter must be reset before the program can be run again. The Cycle Counter can also be reset using the following procedure:

	<i>Instruction</i>	<i>Display Shows</i>
1	While in Run mode, press F3 (i.e. Inch/Menu 1) key.	Prog: AUTO 1. Reset counter 2. Adjust position
2	Press 1 key to chose "Reset Counter" shown on display.	Prog: AUTO Press Move key Cycle Counter: 0

4.5. Set Password

Set Password allows the user to protect a program from editing. If the program is locked, the user will not be able to change any of the program data. Unlocking the program will allow the data to be changed again.

4.6. Jog Speed

Sets the jog speed of the axes of the robot (namely, at the velocity level: high), both the linear speed and the rotation speed (for 4-axis robots). The level of the velocity can be reduced to other two limits: middle (50% of high level) and low (10% of high level). Although only mm/sec is displayed on the Teach Pendant as measure unit of the speeds, the rotation speed (when applicable) will be considered as being expressed in degrees/sec.

Note: For 4-axis robots, the rotation axis is marked with W on the keys and with R on the display of the Teach Pendant.

4.7. Run Mode

Determines whether the robot operates in Standalone mode (default) or Host mode or Console mode. Host mode allows the robot to be controlled by commands sent over the RS232 port.

4.8. Adjust Position

Sets the position for Adjust Origin.

4.9. *Parameter*

Sets the parameters of the unit. To have access to these data you are prompted to insert the current password of the unit.

Robot Type Set

Selects the robot model / type.

Motor Setting

Sets: motor power, motor brake, encoder (pulse/rev), other spec.

Reduction Ratio

Sets the reduction ratio of each axis.

Origin Offset Set

Moves the Origin Position to proper position.

Gain Setting

Changes the Motor Gain.

Speed Set

Changes the maximum speed, acceleration time, inching speed, round set for velocity,
acceleration option set.

Brake Time Set

Selects the brake On/Off delay.

Tool Setting

Sets: Tool Offset.

RS-232C

Changes the serial port setting.

Origin Setting

Changes the origin speed, origin process, origin mode.

H/W Test

Checks the hardware status: servo func., system input, servo R/W test, sensor, origin,
servo led.

Encoder Preset (TMB Series Only)

Resets the absolute encoder error.

Inposition Set

Sets INPOS output usage: Inpos (pulse), Inpos (time), Inrange, Inpos < - - >
INRNG Output by [Inpos].

Direction Setting

Sets the positive/the negative direction of each axis.

Axis Change

Selects the hardware channel and each axis

Limit Enable

Determines whether to be used limit sensor or not.

Change Password

Changes the password to enter the Parameter menu.

Set PLC Autorun

Sets PLC file autorun option.

S/W Limit Set

Selects the Software limit area.

Bcode Set

Changes/Edits the Buffer.

Parallel Set

Sets the parallel axis.

Protocol Set

Selects the Protocol while robot is running.

Stop Output

Selects Output status when robot stops moving.

Flash Burning

Upgrades the robot firmware.

4.10. Auto Tool Setting

This function can be used in Menu1>Parameter>Tool Setting.

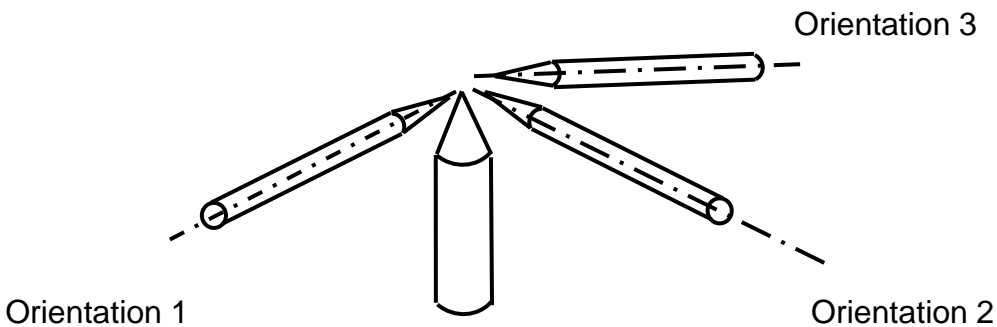
In order to get an accurate offset of a tool attached to the end of the rotational axis of a manipulator, three positions are necessary to be taught for one point with different orientation of the manipulator.

This function automatically calculates and sets X and Y value of tool offset from the taught three positions.

Teach three position for one same position with the tool mounted on the end of a manipulator one by one. the angle between three positions should be greater than 90 degree in the following figure so that the offset value of the tool can be calculated as accurate as possible.

In the menu, after teaching a point, press F4 PNT key to save the current position. The message "PNT1" for the first position will change to "PNT2" for the second position and "PNT3" for the third position each time the F4 key is pressed

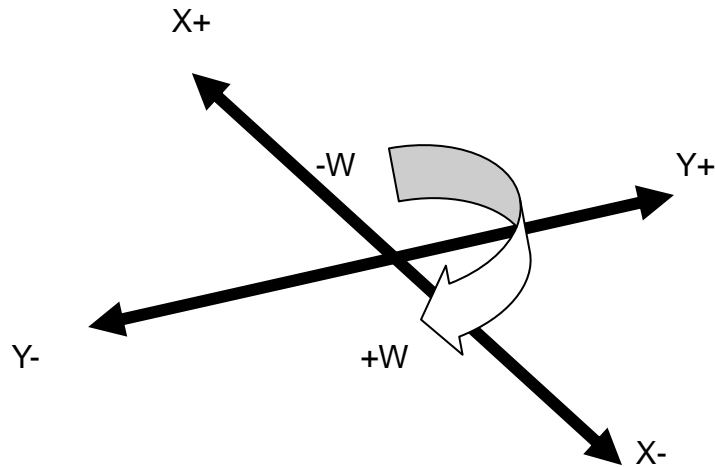
After teaching all three position with different orientations, press ESC key and select 1.Yes to calculate the offset of the attached tool.



	<i>Instruction</i>	<i>Display Shows</i>
1	Press the MENU 1 key, then select Parameter and find Tool Setting Menu for tool correction function.	[TOOL OFFSET SET] 1.INPUT TOOL OFFSET 2.AUTO TOOL SET
2	Press 2 for Auto tool set.	[AUTO TOOL SETTING] A:5.3 B:4.5 Z:39.4 R:3.8 PNT
3	Move Tip to Orientation 1 position by hand or Jog key. And Press F4 to save current Position.	[AUTO TOOL SETTING] A:8.3 B:3.5 Z:39.4 R:6.8 [PNT1]
4	Move Tip to Orientation 2 position by hand or Jog key. And Press F4 to save current Position.	[AUTO TOOL SETTING] A:48.3 B:43.2 Z:39.4 R:36.8 [PNT2]
5	Move Tip to Orientation 3 position by hand or Jog key. And Press F4 to save current Position.	[AUTO TOOL SETTING] A:118.3 B:33.5 Z:39.4 R:36.8 [PNT3]
6	Press ESC key to calculate Tool offset. Then Select 1.Yes	[AUTO TOOL SETTING] A:8.3 B:3.5 Z:39.4 R:6.8 1.YES 2.NO
7	Check the value of tool offset in Input Tool Offset menu.	[TOOL SETTING] X Length : 10.3 mm Y Length : 4.5 mm Z Length : 0 mm
8	In Teach mode, select rectangular coordination by shift+F2 key. And press R+/- Jog key. If tool correction is right. Tip will move around current position as a center.	ADDR: 0 PROG:0 EMPTY X:113.2 Y:23.5 Z:14.3 R:8.5

For tool setting, next conditions must be satisfied.

1. The rotation direction of R axis must be from +X to +Y.



2. The angle between three Orientation positions should be greater than 90 degree.

4.11. Resume

Once a program has been interrupted, the **Resume** setting will determine if the program restarts from point # 1 or from the point at which it was interrupted, such as the middle of a program.

If **Resume** is set to **1 and** a program is stopped by an emergency stop signal or by the enclosure door switch open signal, the program will restart from the point at which it was interrupted when the start button is pressed.

If **Resume** is set to **0** and a program is stopped by an emergency stop signal or by the enclosure door switch open signal, the program will restart from point #1.

The default value is **0**.

4.12. Origin Searching

This function makes the robot to go to the home position.

4.13. Hour Meter

This function shows the total time the controller has been on, as well as how long the manipulator has run.

4.14. PLC File Edit

Allows the user to create and edit PLC files. For more information refer to Section 7:PLC.

4.15. I/O Monitor

Displays current Input/Output status.

4.16. Set Variable

Allows to insert value to Variable which is used in CALC(arithmetic).

4.17. EMG Mode

EMG Mode selects System Emergency signal of System Input as: Interlock signal or Emergency Stop signal.

5. *Menu 2*

5.1. *Point Utility*

5.1.1. *MDI Mode*

MDI Mode allows to modify the position of each address by numerical key input. After pressing **Menu 2** key and then entering Point Utility and MDI Mode, using the ▲/-1, ▼/+1 keys, find the address number that you want and press the **ENT** key. The cursor will appear in the X position. Enter the value using the key pad on the Teach Pendant and press **ENT**. Then the cursor will move to the Y position. Enter the value using the key pad on the Teach Pendant and press **ENT**. The same procedure will be followed for Z and R (for 4-axis robots) positions. When finished, you will be asked if want to save these changes.

5.1.2. *Numerical Move*

Allows the tip to be positioned numerically by entering the desired numbers for the X, Y, Z and R (for 4-axis robots) values.

5.1.3. *Save Temp Point*

Saves the current XYZR (for 4-axis robots) positions in a temporary memory area numbered 1 – 9. The Save Temp Point function is also used with the Relocate Data function. See **Section 6:5.2.3 Relocate Data**.

5.1.4. *Retrieve Temp Point*

Retrieves an XYZR (for 4-axis robots) position that was stored with Save Temp Point.

5.2. *Group Utility*

5.2.1. *Group Edit*

Group Edit is a powerful utility that allows several different functions to be applied to a user-defined group of addresses. These functions include: copy, delete, move, multiply line speed, multiply dispense time, apply X Offset, apply Y Offset, apply Z Offset, apply angle of rotation (for 4-axis robots).

5.2.1.1. Copy

For example, to use *Group Edit* to copy addresses 1 – 20 in the current program to memory addresses 21 – 40:

	<i>Instruction</i>	<i>Display Shows</i>
1	Press the MENU 2 key, then 2 to select <i>Group Utility</i> and then 1 to select <i>Group Edit</i> . The display will prompt the user to enter the starting memory address of the group to edit (From) and the ending number of the group to edit (To).	GROUP EDIT FROM: █ TO : (0 <-> 2999)
2	Type 1 then press ENT to register 1 in <i>From</i> . Type 20 then press ENT to register 20 in <i>To</i> .	GROUP EDIT 1-20 1.Copy 4.Line SP 2.Delete 5.Disp.TM 3.Move 6.Offset
3	The <i>Group Edit</i> menu will then appear, allowing the user to select which function will apply to the range of points. Press 1 to select <i>Copy</i> .	GROUP COPY SOURCE 1-20 Destination:
4	The display will prompt the user to type the destination memory address where the data will be copied. Press the Del key when the cursor is on the old figure to erase it, then type 21 and press ENT to select destination memory address number 21. For moving the cursor to be used /SPD- and /SPD+ keys.	GROUP COPY SOURCE 1-20 Destination:21 1.Yes 2.No
5	The display will now prompt the user to confirm the copy. Press 1 to select <i>Yes</i> and perform the copy.	

5.2.1.2. Delete

For example, to use *Group Edit* to delete addresses 15 – 25 in the current program:

	<i>Instruction</i>	<i>Display Shows</i>
1	Press the MENU 2 key, then 2 to select <i>Group Utility</i> and then 1 to select <i>Group Edit</i> . The display will prompt the user to enter the starting memory address of the group to edit (From) and the ending number of the group to edit (To).	GROUP EDIT FROM : █ TO : (0 <-> 2999)
2	Type 15 then press ENT to register 15 in <i>From</i> . Type 25 then press ENT to register 25 in <i>To</i> .	GROUP EDIT 15-25 1.Copy 4.Line SP 2.Delete 5.Dispen.TM 3.Move 6.Offset
3	The <i>Group Edit</i> menu will then appear, allowing the user to select which function will apply to the range of points. Press 2 to select <i>Delete</i> .	GROUP DELETE SOURCE 15-25 SELECT 1. Yes 2.No
4	The display will now prompt the user to confirm the deletion. Press 1 to select Yes and delete the data.	

5.2.1.3. Move

For example, to use *Group Edit* to move addresses 10 – 20 in the current program to memory addresses 50 – 60:

	<i>Instruction</i>	<i>Display Shows</i>
1	Press the MENU 2 key, then 2 to select <i>Group Utility</i> and then 1 to select <i>Group Edit</i> . The display will prompt the user to enter the starting memory address of the group to edit (From) and the ending number of the group to edit (To).	GROUP EDIT FROM : █ TO : (0 <-> 2999)
2	Type 10 then press ENT to register 10 in <i>From</i> . Type 20 then press ENT to register 20 in <i>To</i> .	GROUP EDIT 10-20 1.Copy 4.Line SP 2.Delete 5.Dispen.TM 3.Move 6.Offset
3	The <i>Group Edit</i> menu will then appear, allowing the user to select which function will apply to the range of points. Press 3 to select <i>Move</i> .	GROUP MOVE SOURCE 10-20 Destination:1
4	The display will prompt the user to type the destination memory address where the data will be moved. Press the Del key when the cursor is on the old figure to erase it, then type 50 and press ENT to select destination memory address number 50.	GROUP MOVE SOURCE 10-20 Destination:50 1.Yes 2.No
5	The display will now prompt the user to confirm the move. Press 1 to select Yes and move the data.	

5.2.1.4. Line SP (Line Speed)

For example, to use *Group Edit* to increase all line speed commands in memory addresses range 1 – 200 by 20 %:

	<i>Instruction</i>	<i>Display Shows</i>
1	Press the MENU 2 key, then 2 to select <i>Group Utility</i> and then 1 to select <i>Group Edit</i> . The display will prompt the user to enter the starting memory address of the group to edit (From) and the ending number of the group to edit (To).	GROUP EDIT FROM : █ TO : (0<-> 2999)
2	Type 1 then press ENT to register 1 in <i>From</i> . Type 200 then press ENT to register 200 in <i>To</i> .	GROUP EDIT 1-200 1.Copy 4.Line SP 2.Delete 5.Dispen.TM 3.Move 6.Offset
3	The <i>Group Edit</i> menu will then appear, allowing the user to select which function will apply to the range of points. Press 4 to select <i>Line SP</i> .	GROUP LINE SP Multiple Value:
4	The display will prompt the user to type the multiple value to be applied to the line speeds. For example, a value of 1.2 will increase all speeds by 20%. A value of 0.8 will decrease all speeds by 20%. Type 1.2 and press ENT to select a multiplier of 1.2.	GROUP LINE SP Multiple Value:1.2 SELECT 4. Yes 2.No
5	The display will now prompt the user to confirm the change. Press 1 to select Yes. All line speed instructions in the selected range of addresses will now be multiplied by 1.2.	

5.2.1.5. Dispen.TM (Dispense Time)

For example, to use *Group Edit* to increase all dispensing times (Point Dispense Setup) in memory addresses range 1 – 200 by 15%:

	Instruction	Display Shows
1	Press the MENU 2 key, then 2 to select <i>Group Utility</i> and then 1 to select <i>Group Edit</i> . The display will prompt the user to enter the starting memory address of the group to edit (From) and the ending number of the group to edit (To).	GROUP EDIT FROM : █ TO : (0 <-> 2999)
2	Type 1 then press ENTER to register 1 in <i>From</i> . Type 200 then press ENTER to register 200 in <i>To</i> .	GROUP EDIT 1-200 1.Copy 4.Line SP 2.Delete 5.Dispen.TM 3.Move 6.Offset
3	The <i>Group Edit</i> menu will then appear, allowing the user to select which function will apply to the range of points. Press 5 to select <i>Dispen.TM</i> .	GROUP DISPENSE TM SOURCE 1 – 200 Multiple Value:
4	The display will prompt the user to type the multiple value to be applied to the dispense times. For example, a value of 1.15 will increase all dispense times by 15%. A value of 0.85 will decrease all dispense times by 15%. Type 1.15 and press ENTER to select a multiplier of 1.15.	GROUP DISPENSE TM SOURCE 1 – 200 SELECT 5. Yes 2.No
5	The display will now prompt the user to confirm the change. Press 1 to select Yes. All the Point Dispense Setup instructions in the select range of addresses will now be multiplied by 1.15.	

5.2.1.6. Offset

The Offset function allows all XYZR (for 4-axis robots) locations in a program to be shifted in new XYZR locations at distances (with values) defined by the user.

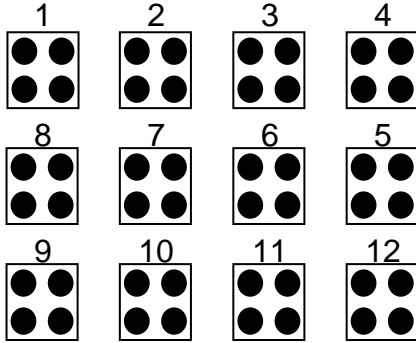
For example, to use *Group Edit* for adding 15 mm to all X axis values in memory addresses range 1 – 200:

	<i>Instruction</i>	<i>Display Shows</i>
1	Press the MENU 2 key, then 2 to select <i>Group Utility</i> and then 1 to select <i>Group Edit</i> . The display will prompt the user to enter the starting memory address of the group to edit (From) and the ending number of the group to edit (To).	GROUP EDIT FROM : █ TO : (1 <-> 2999)
2	Type 1 then press ENTER to register 1 in <i>From</i> . Type 200 then press ENTER to register 200 in <i>To</i> .	GROUP EDIT 1-200 1.Copy 4.Line SP 2.Delete 5.Dispen.TM 3.Move 6.Offset
3	The <i>Group Edit</i> menu will then appear, allowing the user to select which function will apply to the range of points. Press 6 to select <i>Offset</i> .	GROUP OFFSET1-200 X: Y: Z: R: Unit: mm
4	The display will prompt the user to type the X, Y, Z and R offset amounts to be added to all points in address range 1 – 200. Type 15 and press ENTER to add 15 mm to the X axis values. Press ENTER three more times to leave the Y, Z and R offsets at zero.	GROUP OFFSET1-200 SELECT: 6. Yes 2.No
5	The display will now prompt the user to confirm the change. Press 1 to select Yes. 15 mm will be added to all of the X axis values in the selected range of addresses.	

5.2.2. Expand Step & Repeat

Expand Step & Repeat will expand a step and repeat instruction to the actual data it represents. For example, if the following program was created:

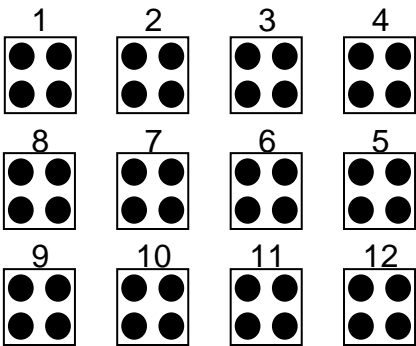
Before:



Address	Instruction
0	Dispense End Setup
1	Z Clearance
2	Point Dispense Setup
3	Dispense Dot
4	Dispense Dot
5	Dispense Dot
6	Dispense Dot
7	Step & Repeat X, Addr=4
8	End Program

The original program occupies 9 memory addresses. If the user brings memory address number 7 into the display and then selects **MENU 2→Group Utility→ Expand Step & Repeat**, address 7 will be expanded into the 44 points which it represents, bringing the total number of memory addresses used to 51 (plus the End Program instruction at address 52).

After:



Address	Instruction
0	Dispense End Setup
1	Z Clearance
2	Point Dispense Setup
3	Dispense Dot
4	Dispense Dot
5	Dispense Dot
6	Dispense Dot
7	Dispense Dot
8	Dispense Dot
9	Dispense Dot
.	.
.	.
.	.
50	Dispense Dot
51	End Program

The Expand Step & Repeat function is useful in situations where the user must edit selected elements in a Step & Repeat group, although an expanded Step & Repeat instruction will occupy more memory space than an un-expanded instruction.

5.2.3. Relocate Data

The *Relocate Data* function allows the position of a program to be corrected, including correction for X offset, Y, offset, Z offset and angle of rotation.

If, for example, the work piece fixture has been changed, the program position can be adjusted automatically for the new fixture.

The *Relocate Data* function requires two reference points for the calculations. Choose two point locations from your program that will be used as reference points. For example, to relocate the program after a fixture change:

	<i>Instruction</i>	<i>Display Shows</i>
1	In this example, memory address 3 is the first reference point and memory address 4 is the second reference point. Go to memory address 3.	
2	Jog the tip to the new, correct position for the first reference point.	ADDR:3 PROG:20 Line Start X:100.00 Y:12.54 Z:23.65 R:
3	Press the Setup key, then select Adjust point#1 to save the location and address. This location is saved as location in temporary position #1.	Adjust Positon 1 Saved -----

	<i>Instruction</i>	<i>Display Shows</i>
4	Go to memory address 4.	ADDR:4 PROG:20 Line Passing X:200.00 Y:112.54 Z:23.65 R:
5	Jog the tip to the new, correct position for the second reference point.	
5	Press the Setup key, and then select Adjust point#2 to save the location and address. This location is saved as location in temporary position #2.	Adjust Position 2 Saved -----
6	In Run mode. Press Menu2 key , then select 2.Group Utility > 2. Adjust Position.	1. All Points 2. Some Points SELECT:
7	Press 1 to select All Points.	Relocate All Point S1-> S2->
8	Input Address number for Adjust point #1 and #2 to S1 and S2.	Point Relocated
9	The program location will be adjusted for X offset, Y offset, Z offset and angle of rotation.	

5.2.4. Adjust Origin

When the dispensing barrel or tip is removed and replaced, the new tip/barrel is often in a slightly different XYZR (for 4-axis robots) position than the old tip/barrel was.

The robot has a software utility to adjust a program's origin, thereby correcting the tip offset problem.

A reference should be chosen someplace on the work piece fixture or on the work piece itself. The reference point must be registered in the program data. This only needs to be done one time, for example when the program is originally created.

	<i>Instruction</i>	<i>Display Shows</i>
1	Jog the tip to the reference point or if the reference point is an existing point in your program, press the MOVE key to bring the tip to that XYZR location.	ADDR: PROG: EMPTY X: Y: Z: R:
2	Press the Menu 1 key, and then on page two select ADJUST POSITION to save the location.	Set Adjust Position

When the tip/barrel is changed, use the following procedure to adjust the program origin for the new tip location.

	<i>Instruction</i>	<i>Display Shows</i>
1	While in TEACH mode, press the MENU 2 key, then 2 to select <i>Group Utility</i> menu.	1.Group Edit 2.Expand Step&Repeat 3.Relocate Data 4.Adjust Origin
2	Press 4 to select <i>Adjust Origin</i> .	Move to First Point X: Y: Z: R: Press Any Key
3	Press any key. The tip will move to the reference point that was recorded in the Adjust Position. If the new tip location is slightly different from the last tip location, you should see that the tip is not exactly at the reference point.	Adjust the First Point X: Y: Z: R:

	<i>Instruction</i>	<i>Display Shows</i>
4	<p>Jog the tip to the correct location for the reference point.</p> <p>When the location is correct, press ENT.</p> <p>The program origin will be adjusted for the new tip location.</p>	<p>Program Data Adjusted !!!</p>

5.3. Program Utility

The *Program Utility* menu includes two options: *Copy Program* and *Delete Program*.

5.3.1. Copy Program / Delete Program

1. **Copy** Copies the current program number to a different program number
2. **Delete** Erases data in the current program number.

5.3.2. Auto Offset

Move the tip using the original coordinates stored in program memory with the MOVE button. The next step is to jog the tip to the where it should be and the robot does all the math on all data points to align the tip/fixture with the original program

	<i>Instruction</i>	<i>Display Shows</i>
1	<p>While in TEACH mode, press the MENU 2 key, then 3 to select <i>Program Utility</i> menu.</p>	<p>1.Copy Porgram 2.Delete Program 3.Auto Offset</p>
2	<p>Press 3 to select <i>Auto Offset</i>.</p> <p>Then go to address that store reference position with cursor key..</p>	<p>ADDR:0 PROG: 0 L Line Start X: 100 Y:100 Z: 100 R:100</p>

	<i>Instruction</i>	<i>Display Shows</i>
3	<p>Press Move key. The tip will move to the saved position that was recorded in the current Address.</p> <p>If the new tip location is slightly different from the last tip location, you should see that the tip is not exactly at the reference point.</p>	<p>Move New Point</p> <p>X: 100 Y:100</p> <p>Z: 100 R:100</p>
4	<p>Jog the tip to the correct location for the reference point.</p> <p>When the location is correct, press ENT.</p> <p>The program origin will be adjusted for the new tip location.</p>	<p>Program Data</p> <p>Adjusted !!!</p>

5.4. Memory Utility

5.4.1. Delete Memory

Erases all programs.

5.4.2. Program BackUp

Save program and Parameter from main memory to Flash Memory.

5.4.3. Delete Memory

Load program and Parameter from Flash Memory to main memory.

Section 7: PLC

1. Creation of a PLC file

The robot controller has the mini- PLC function. Up to 10 programs can be created, each program containing up to 1000 Steps.

The contacts in PLC are classified as follows.

S : System Input	S0 ~ S23
X : General Input	X0 ~ X31
Y : General Output	Y0 ~ Y31
M : Internal Contact I/O	M32 ~ M352
T : Timer	64 ,Set from 1 to 9999
C : Counter	64, Set from 1 to 9999
B : 16 bit Register (Nonvolatile Memory)	B0 ~ B1023
D : 16 bit Register (Volatile Memory)	D0 ~ D255
H : System Control Contact	H0 ~ H31

The sampling time for the PLC program is 30msec.

To create a PLC file, follow the next steps:

	<i>Instruction</i>	<i>Display Shows</i>
1	Turn On the controller and set it in Run Mode.	PROG:00 AUTO Press Move Key Cycle Counter: 0 AU/ST PLC Clear Mode
2	Select Mode by pressing the Mode/Menu 2 key.	----- Press Home Key To Find Origin -----
3	Press the Home/-/R key. The robot will move to the home position.	ADDR:0 PROG:00 EMPTY X:0 Y:0 Z:0 R:0

	<i>Instruction</i>	<i>Display Shows</i>
4	Press Menu 1 key, then press PgDn key three times, to go to page 4 of the menu. Finally, press the numeric key 1 (for: PLC File Edit).	PLC FILE NO []
5	Press any numeric key from 0 to 9, for example 1, and press the ENT key.	ADDR:0 PLC_1 i EDIT GROUP PGDEL
6	Select EDIT by pressing the F1/Setup key.	ADDR:0 PLC_1 i PROG CTRL MV DMV
7	The different commands can be accessed using the F1, F2, Menu1 and Menu2 keys.	

The following is a list of the commands found under **PROG**, **CTRL**, **MV**, **DMV**.
(Use /-1 and /+1 keys to go from one group of commands to another. Use ESC key to exit from the displayed group of commands).

PROGRAM

NAME	MEANING	FUNCTION
AND	And	Serial connection of logical operation (A contact).
OR	Or	Parallel connection of logical operation (A contact).
NOT	Negation	
LD	Load	Start operation at A (Normal open) contact.
OUT	Out	Output of the operation result.
PULS	Pulse	If input contact value changes to On status, this output will be given during 1 Scan Time.
SET	Set	If input is turned On, the output contact is maintained with On state.
RESET	Reset	If input is turned On, the output contact is maintained with Off state.

CTRL

NAME	MEANING	FUNCTION
T	Timer	This is used to control time and is On Delay Timer.
C	Counter	This is used to count the number and is Down Counter. When the enable signal is turned Off, the current value will be reset to the setting value. Input after counting is finished will be ignored.
MC		Master Control Set.
MCR	MC Reset	Master Control Reset.
D	Data	Sets Data value for counter and timer.
BK	Block	
END	End	Marks the end of the program.

MV

NAME	MEANING	FUNCTION
MOV	Move 16bit data	This is used to move data from a 16 bit register (B or D) to another 16 bit Register (B or D) ex: MOV B100 D100 (D100 = B100)
ADD	Add two 16bit data	This is used to add data of two 16 bit register (B or D) and store the result to another 16 bit Register (B or D) ex: ADD B100 D30 D40 (D40 = B100 + D30)
SUB	Subtract two 16bit data	This is used to subtract data of two 16 bit register (B or D) and store the result to another 16 bit Register (B or D) ex: SUB B100 D30 D40 (D40 = B100 - D30)
MUL	multiply two 16bit data	This is used to multiply data of two 16 bit register (B or D) and store the result to another 16 bit Register (B or D) ex: MUL B100 D30 D40 (D40 = B100 x D30)
DIV	divide two 16bit data	This is used to divide data of two 16 bit register (B or D) and store the quotient to another 16 bit Register (B or D), remainder to Next 16bit Register (B or D) ex: DIV B100 D30 D40 (D40 = B100 / D30 D41 = B100 % D30)

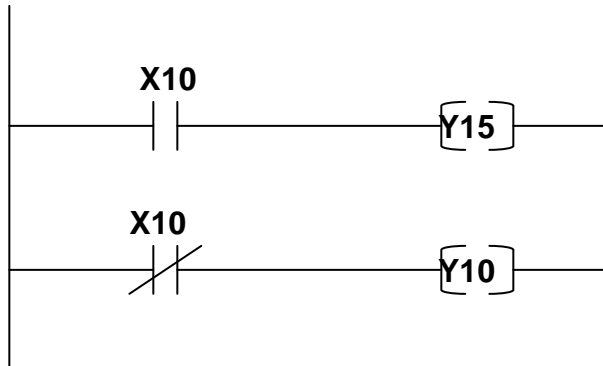
DMV is 32 bit operation of MV function.

2. Running a PLC file

	<i>Instruction</i>	<i>Display Shows</i>
1	Turn On the controller and set it in Run Mode. Press the Shift/Char + Menu2 keys.	PROG:00 AUTO Press Move Key Cycle Counter: 0 AU/ST PLC Clear Mode
2	Select PLC by pressing the F2 key.	PROG: PLC_0 [STOP] RUN
3	Use the arrow keys ▲▼ to select the program number, for example 1.	PROG: PLC_1 [STOP] RUN
4	Press F1 to execute the program. If the ESC key is pressed, it will exit from the current display and return to Program selection display. However, the current PLC program will continue running.	PROG: PLC_1 [PLAY] RUN
5	To stop the program, press the STOP button.	PROG: PLC_1 [STOP] RUN

3. PLC Program Examples

3.1. LD / LDNOT / OUT



[Content]

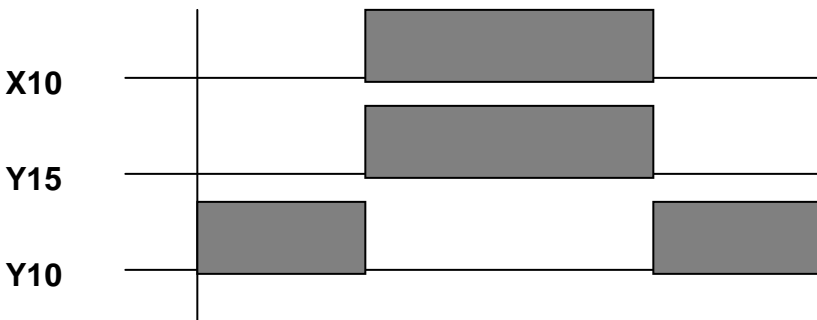
If the general input 10 is turned ON, then the general output 15 is turned ON
and the general output 10 is turned OFF.

If the general input 10 is turned OFF, then the general output 15 is turned OFF
and the general output 10 is ON.

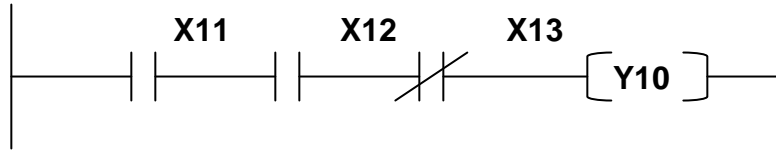
[Program]

```
LD X10
OUT Y15
LDNOT X10
OUT Y10
END
```

[Time Chart]



3.2. AND / ANDNOT



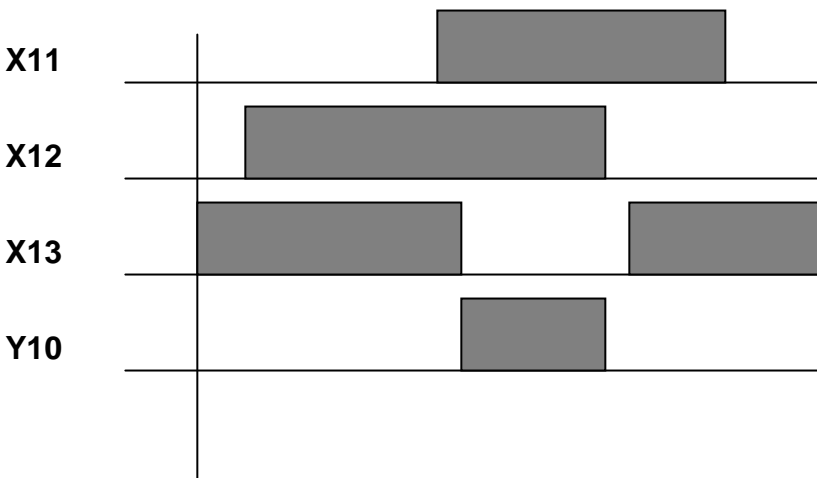
[Content]

If the general input 11 is turned ON and
The general input 12 is turned ON and
The general input 13 is turned OFF,
Then the general output 10 will be turned ON.

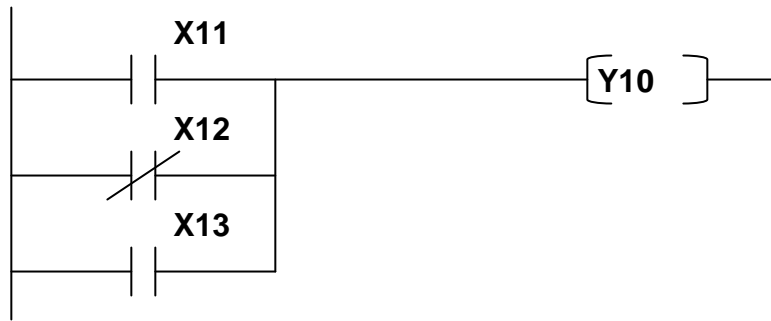
[Program]

```
LD X11  
AND X12  
ANDNOT X13  
OUT Y10  
END
```

[Time Chart]



3.3. OR / ORNOT



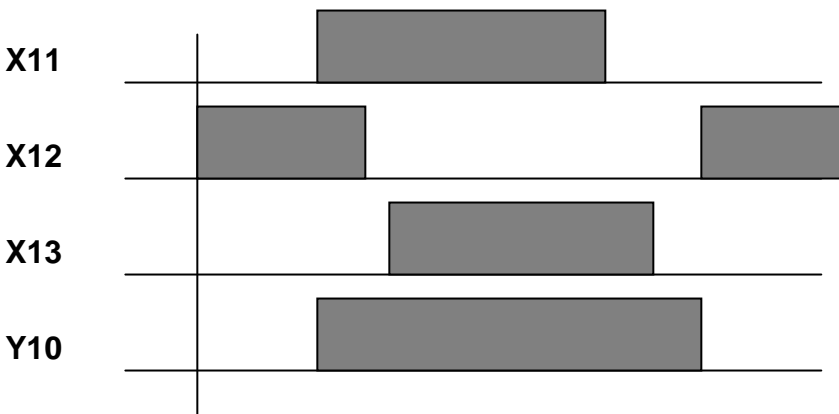
[Content]

If the general input 11 is turned ON or
The general input 12 is turned OFF or
The general input 13 is turned ON,
Then the general output 10 will be turned ON.

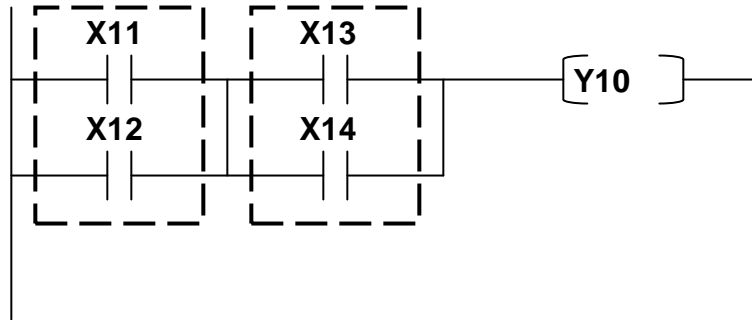
[Program]

```
LD X11
ORN X12
OR X13
OUT Y10
END
```

[Time Chart]



3.4. ANDBK



[Content]

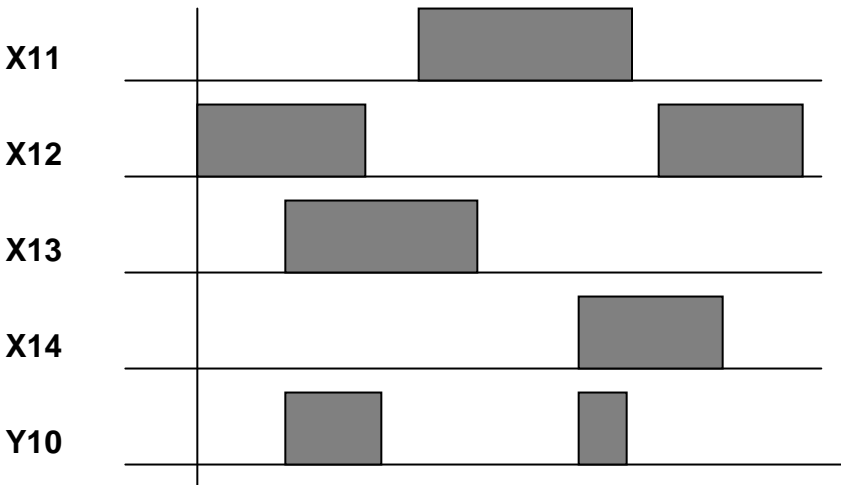
If one of the general inputs 11 or 12 is turned ON and
 One of the general inputs 13 or 14 is turned ON,
 Then the general output 10 will be turned ON.

[Program]

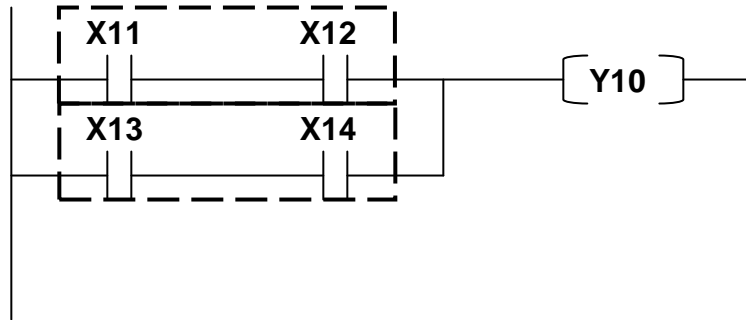
```

LD X11
OR X12
LD X13
OR X14
ANDBK
OUT Y10
END
    
```

[Time Chart]



3.5. ORBK



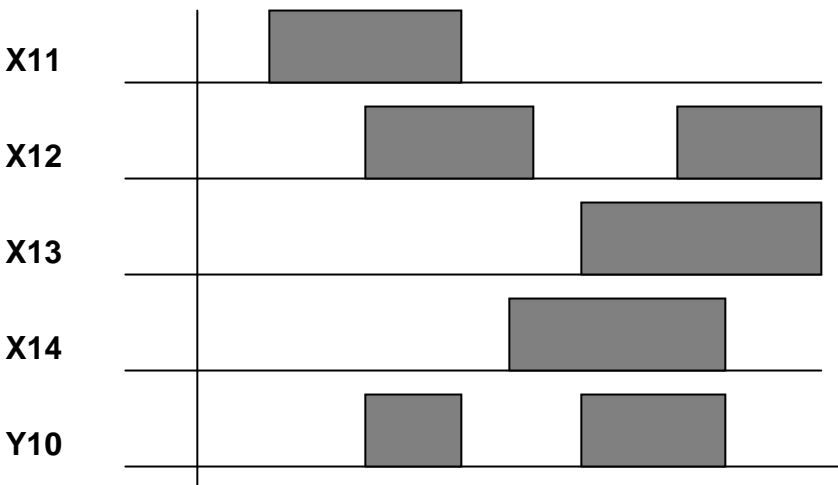
[Content]

If the general input 11 is turned ON and 12 is turned ON or
The general input 13 is turned ON and 14 is turned ON,
Then the general output 10 will be turned ON.

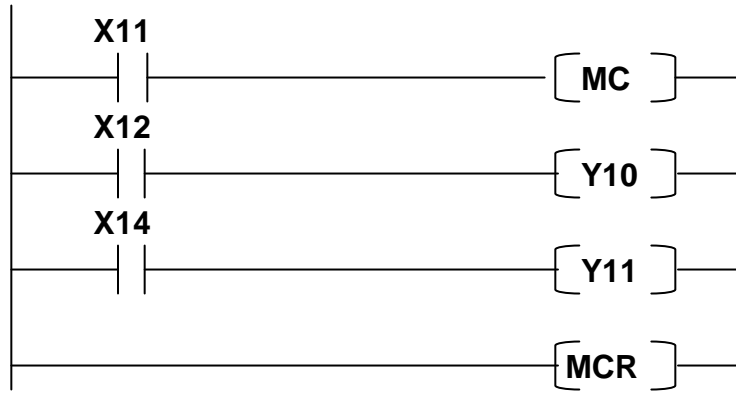
[Program]

```
LD X11
AND X12
LD X13
AND X14
ORBK
OUT Y10
END
```

[Time Chart]



3.6. MC/MCR



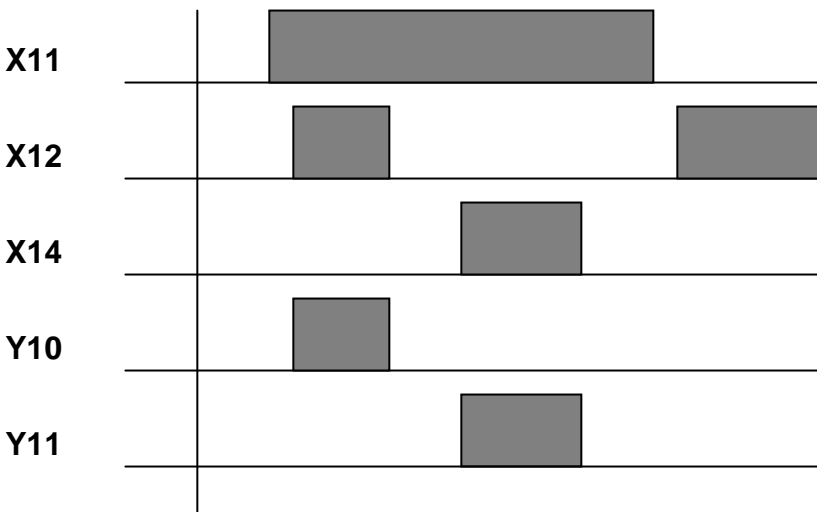
[Content]

Only when the input condition of MC is turned ON, it will be run until the same MCR.

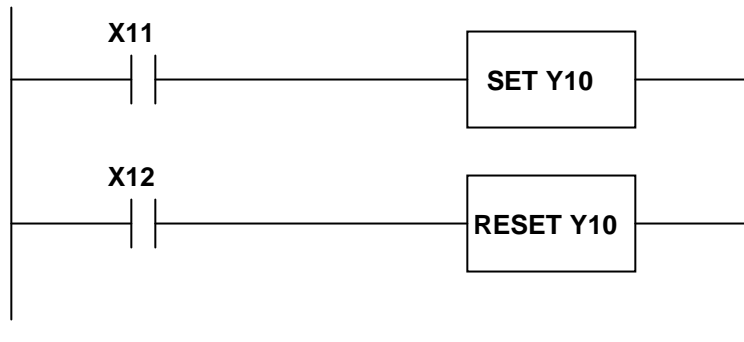
[Program]

```
LD X11
MC
LD X12
OUT Y10
LD X14
OUT Y11
MCR
END
```

[Time Chart]



3.7. SET/RESET



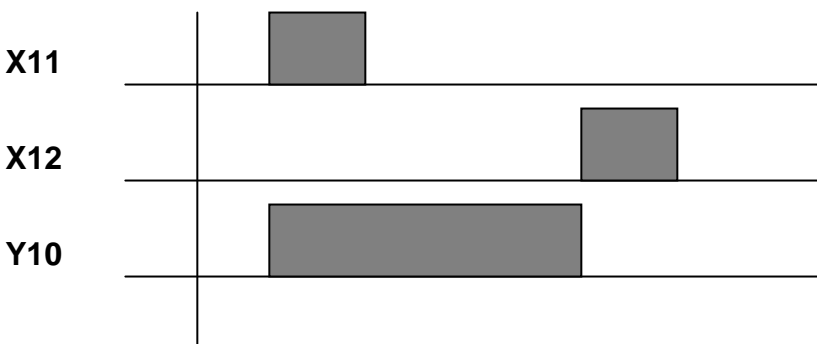
[Content]

If the general input 11 is turned ON, the general output 10 will be turned ON and maintained in the ON state. **(Even if input is turned OFF, it will remain in the ON state).** If the general input 12 is turned ON, the general output 10 will be turned OFF and maintained in the OFF state. **(Even if input is turned OFF, it will remain in the OFF state).**

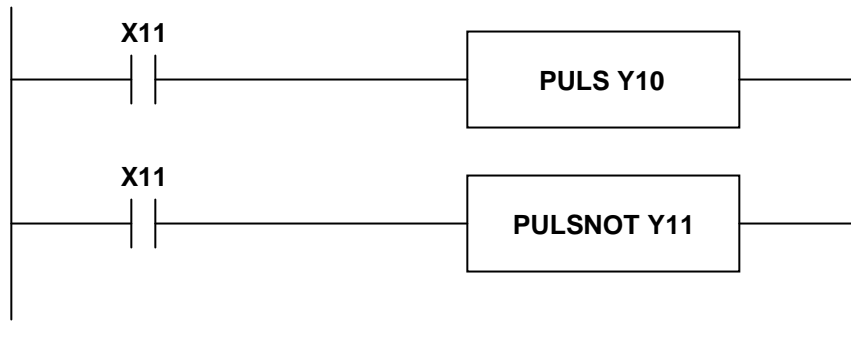
[Program]

```
LD X11
SET Y10
LD X12
RESET Y10
END
```

[Time Chart]



3.8. PULS/PULSNOT



[Content]

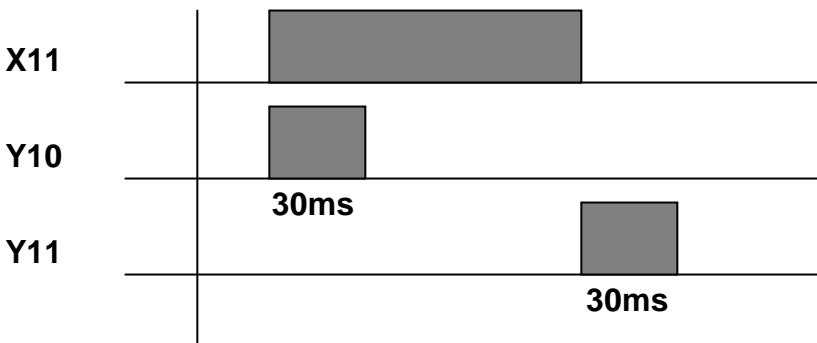
If the general input 11 is turned ON, then general output 10 will be turned ON for 1PULSE (30ms).

If the general input 11 is turned OFF, the general output 11 will be turned ON for 1PULSE (30ms).

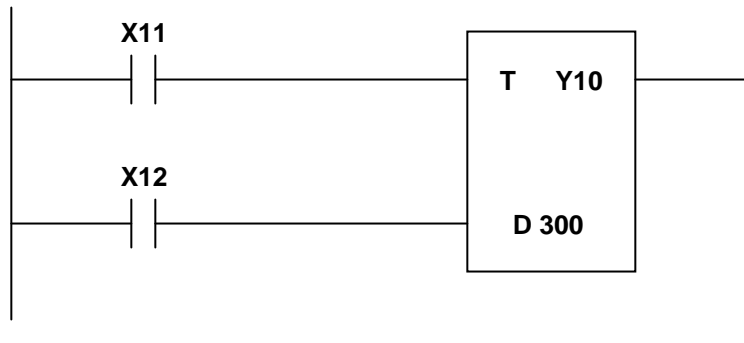
[Program]

```
LD X11  
PULS Y10  
LD X11  
PULSNOT Y11  
END
```

[Time Chart]



3.9. T (Timer)



[Content]

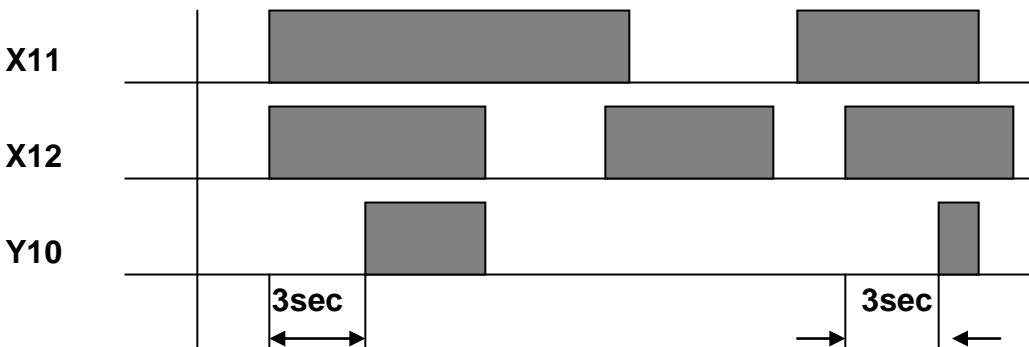
When the general input 11 is turned ON and if the general input 12 is turned ON for more than 3 seconds, the general output 10 is turned ON.

Timer is based on 10ms.

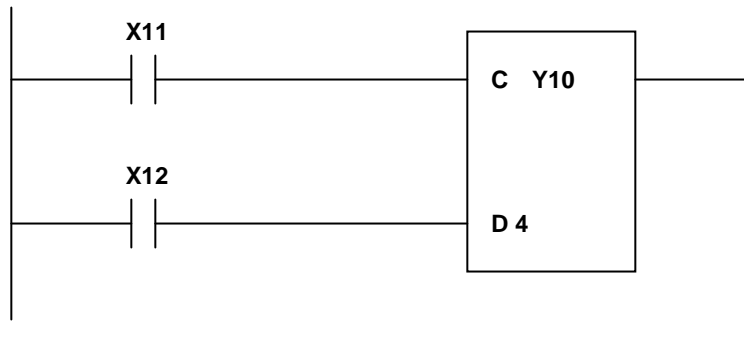
[Program]

```
LD X11
LD X12
T Y10
D 300 (D differs from D register. It means set timer 3 seconds)
END
```

[Time Chart]



3.10. C (Counter)



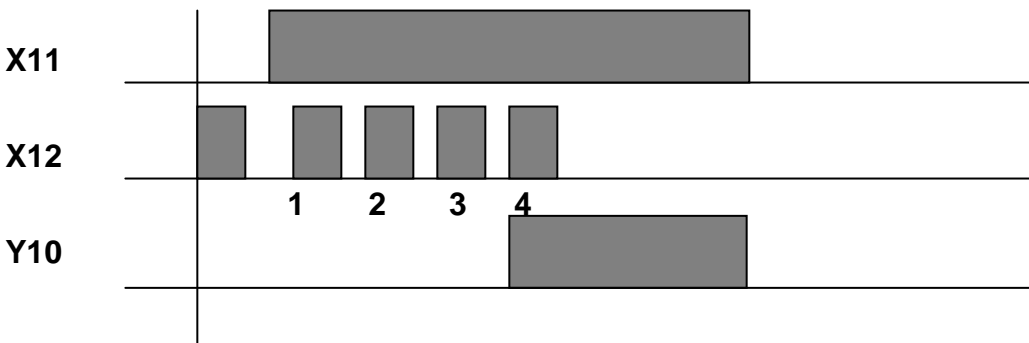
[Content]

When the general input 11 is turned ON,
and if more than 4 PULSES come to the general input 12, the general output 10 is turned ON.

[Program]

```
LD X11
LD X12
C Y10
D 4      (D differs from D register. It means set counter 4 )
END
```

[Time Chart]



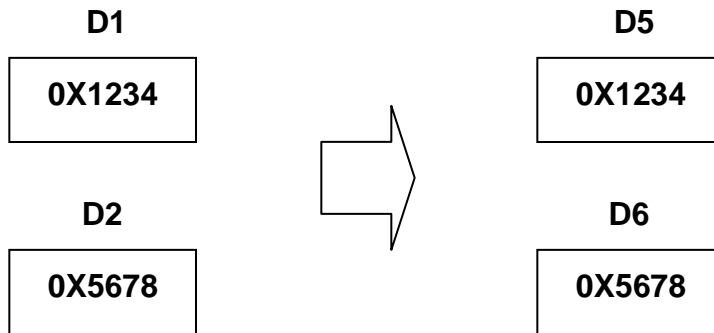
3.11. MOV / DMOV

This is used to move data from a 16/32 bit register (B or D) to another 16/32 bit register (B or D).

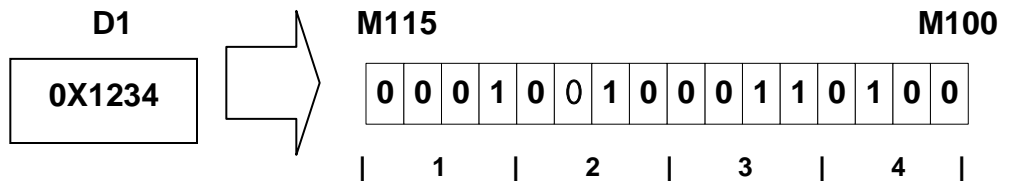
MOV D1 D5



DMOV D1 D5



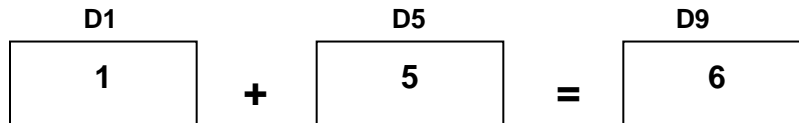
MOV D1 M100



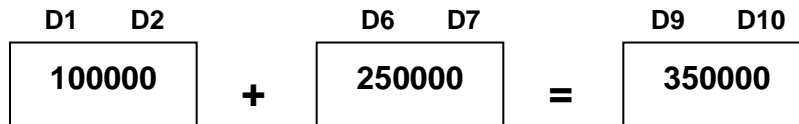
3.12. ADD / DADD

This is used to add data of two 16 bit register (B or D) and store the result to another 16 bit register (B or D).

ADD D1 D5 D9



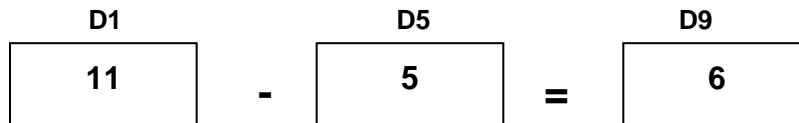
DADD D1 D5 D9



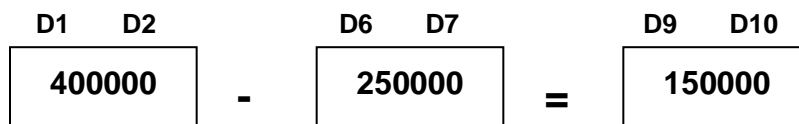
3.13. SUB / DSUB

This is used to subtract data of two 16/32 bit register (B or D) and store the result to another 16 /32 bit register (B or D).

SUB D1 D5 D9



DSUB D1 D5 D9



3.14. MUL/DMUL

This is used to multiply data of two 16/32 bit register (B or D) and store the result to another 16/32 bit register (B or D).

MUL D1 D5 D9

$$\begin{array}{c} \text{D1} \\ \boxed{11} \end{array} \times \begin{array}{c} \text{D5} \\ \boxed{5} \end{array} = \begin{array}{c} \text{D9} \\ \boxed{55} \end{array}$$

DMUL D1 D5 D9

$$\begin{array}{c} \text{D1} \quad \text{D2} \\ \boxed{4000} \end{array} \times \begin{array}{c} \text{D6} \quad \text{D7} \\ \boxed{250} \end{array} = \begin{array}{c} \text{D9} \quad \text{D10} \\ \boxed{1000000} \end{array}$$

3.15. DIV/DDIV

This is used to divide data of two 16/32 bit register (B or D) and store the quotient to another 16/32 bit register (B or D), remainder to next 16bit register (B or D).

DIV D1 D5 D9

$$\begin{array}{c} \text{D1} \\ \boxed{11} \end{array} \div \begin{array}{c} \text{D5} \\ \boxed{5} \end{array} = \begin{array}{c} \text{quotient} \\ \text{D9} \\ \boxed{2} \end{array} \quad \begin{array}{c} \text{remainder} \\ \text{D10} \\ \boxed{1} \end{array}$$









DDIV D1 D5 D9

$$\begin{array}{c} \text{D1} \quad \text{D2} \\ \boxed{100000} \end{array} \div \begin{array}{c} \text{D6} \quad \text{D7} \\ \boxed{3} \end{array} = \begin{array}{c} \text{quotient} \\ \text{D9} \quad \text{D10} \\ \boxed{30000} \end{array} \quad \begin{array}{c} \text{remainder} \\ \text{D11} \quad \text{D12} \\ \boxed{10000} \end{array}$$

4. System Control Contact

4.1. System Control Input

In Repeat Mode, like Front Panel Key, System Input Signal or H Contact Input can operate robot controller. H Contact can be set/reset by PLC program. Next is function of each operation.

Front Panel Key	Description	System Input Signal	H Contact Input
	This key is used for selecting a Program. It increases Program Number by ten.	System Input #11 (Pin No.23) On	-
	This key is used for selecting a Program. It increases Program Number by one.	System Input #12 (Pin No.9) On	-
	This key is used to change the controller to Teach Mode or Repeat Mode.	System Input #10 (Pin No.8) On	-
	This key is used to reset the error caused.	System Input #17 (Pin No.26) On	H42 On
	This key is used to perform an origin-searching operation	System Input #6 (Pin No.6) On	H32 On
	This key is used to start program running.	System Input #16 (Pin No.11) On	H40 On
	This key is used to stop program running.	System Input #7 (Pin No.21) On	H41 On
	Emergency Stop	System Input #13 (Pin No.13) Off	H33 On

External Program Number Call	Select Program by External Program Number	System Input #5 (Pin No. 20)	H39 On
	External Program Number	System Input #0 ~4, #18,#22,#23	number in B0

External Program Number Call

Input #23	Input #22	Input #18	Input #4	Input #3	Input #2	Input #1	Input #0	Program Number
Pin 29	Pin 14	Pin 12	Pin 5	Pin 19	Pin 4	Pin18	Pin3	
1				3				
0	0	0	1	0	0	1	1	13

4.2. System Control Output

Front Panel LED	Description	System Input Signal	H Contact output
Ready	Robot is ready to start.	System Output #0 (Pin No.30)	H2 On
Run	Robot is moving.	System Output #1 (Pin No.44)	H7 On
Org	Origin-searching operation is performed.	System Output #2 (Pin No. 43)	H0 On
Err	An error occurs during operation.	System Output #3 (Pin No. 39)	H3 On

Section 8: Error list

1. Error list for Hardware

The following Error List is about Hardware or Mechanical error.

No	Error Display	Contents	
01	ENVLOP ERR	Meaning	The tracking error of the position/velocity profile exceeds the commanded value.
		Cause	1. The gains of position/velocity control loop are too low. 2. Motor connection is abnormal. 3. Servo Amp is damaged.
		Action	1. Increase the proportional gains of position / velocity control loop to more than 10% of the present value (PP/V_P). 2. Reconnect the motor connector. 3. Contact the manufacturer/sales agent.
02	OVER SPEED	Meaning	The velocity is over the commanded value.
		Cause	Electrical power lines of motor & encoder are connected to other channels.
		Action	Connect the connector properly and if error reoccurs, contact the manufacturer.
03	ENC READ ERR	Meaning	1. Failure to detect stimulus position of initial encoder. 2. Failure to detect position of initial absolute encoder.
		Cause	1. Disconnection or bad connection in encoder line. 2. Encoder power related abnormality.
		Action	1. Check encoder line. 2. Check encoder power line.

04	ENC COUNT ERR	Meaning	Difference in the encoder position during one motor rotation is over 5 pulses.
		Cause	1. Disconnection or bad connection in encoder line while driving. 2. Encoder setting abnormality.
		Action	1. Check/ Reconnect the encoder line. 2. Change encoder setting related to the counts of a rotation.
07	MOTOR CUR ERR	Meaning	Excessive electric current flow in motor.
		Cause	1. Acceleration and deceleration time is too short for inertia load. 2. The electric current loop gain is either too small or too big.
		Action	1. Reset parameter related to acceleration and deceleration. 2. Reset parameter related to gain.
08	ENC ERR	Meaning	Encoder Connection is abnormal.
		Cause	Encoder Cable Connection is bad.
		Action	Check Encoder Connection.
09	OVER LOAD	Meaning	Over load on rated load.
		Cause	1. Friction may be too high between machine and moving parts. 2. Acceleration and deceleration may be frequent in short distances. 3. Inertia load may be too high. 4. Brake malfunction.
		Action	1. Check moving parts in the machine. 2. Increase the acceleration and deceleration time. 3. Reduce inertia load. 4. Check motion or change brakes.
10	MISS ABS INC	Meaning	Encoder Type is not exact.
		Cause	Parameter setting is wrong.
		Action	Correct parameter.

11	H/W LIMIT	Meaning	Came in contact with limit sensor.
		Cause	<ol style="list-style-type: none"> 1. Machine came in contact with the limit sensor. 2. Disconnection or Bad connection of limit sensor line. 3. Set limit sensor as point of contact A (N,O). 4. Improper parameter setting.
		Action	<ol style="list-style-type: none"> 1. Move machine. 2. Check limit sensor line, dismount connector and then connect again. 3. Change limit sensor to point of contact B. 4. Change parameters related to limit setting.
12	S/W LIMIT	Meaning	Exceeded software limit.
		Cause	<ol style="list-style-type: none"> 1. Machine has surpassed software limit area. 2. When point of teach is close to the software limit.
		Action	<ol style="list-style-type: none"> 1. Move the machine. 2. Change parameters related to software limit or change point of teach.
13	INPOS ERR	Meaning	The tracking error of the position/ velocity profile exceeds over the In position value.
		Cause	<ol style="list-style-type: none"> 1. The gains of position/velocity control loop are too low. 2. In position Value in Parameter is too small.
		Action	<ol style="list-style-type: none"> 1. Increase the proportional gains of position/ velocity control loop to more than 10% of the present value. (PP/V_P). 2. Increase In position value in Parameter.
14	ORG ERR	Meaning	Opposite direction signal input of limit sensor during point of origin return process.
		Cause	Incorrect setting of origin sensor/direction.
		Action	Correct the parameter related to origin.

15	ENC BATT ERR	Meaning	Absolute encoder battery has been discharged. (Only SCARA robots)
		Cause	1. The encoder battery voltage may be lower than 2.6V. 2. Bad wiring of encoder line or bad connection.
		Action	1. Check the voltage. 2. Check the encoder line.
16	ENC OVERFLOW	Meaning	Overflow of absolute encoder count.
		Cause	1. The motor rotates continuously in one direction. 2. Preset was not initiated during initial setting.
		Action	1. Limit the motor rotation. 2. Initiate PRESET and reset zero point position.
17	BRAKE ERR	Meaning	Brake disconnection.
		Cause	1. Brake has not been properly set. 2. Actual brake line has bad connection or is disconnected. 3. Brake damage.
		Action	1. Reset brake parameter. 2. Check brake line. 3. Change brake.
18	A LINE ERR	Meaning	Disconnection/bad connection of encoder A phase line.
		Cause	Disconnection/bad connection of encoder line.
		Action	Check encoder line.
19	B LINE ERR	Meaning	Disconnection/bad connection of encoder B phase line.
		Cause	Disconnection/bad connection of encoder line.
		Action	Check encoder line.
20	Z LINE ERR	Meaning	Disconnection/bad connection of encoder Z phase line.
		Cause	Disconnection/bad connection of encoder line.
		Action	Check encoder line.

21	ABS LINE ERR	Meaning	Disconnection/bad connection of absolute encoder line.
		Cause	Disconnection/bad connection of encoder line.
		Action	Check encoder line.
22	SENSOR ERR	Meaning	Limit sensor is disconnected.
		Cause	1. Disconnection/bad connection of actual limit sensor line. 2. Incorrect parameter setting.
		Action	1. Check limit sensor line. 2. Reset parameters related to limit sensor.
42	REGEN ERR	Meaning	Revision time overtime.
		Cause	1. The inertia load may be excessive. 2. Too short acceleration/deceleration. 3. Abnormality in AMP board.
		Action	1. Reduce the inertia load. 2. Lengthen the time of acceleration/deceleration. 3. Change the AMP board.
43	IPM ERR	Meaning	Excessive electric current flow in IPM.
		Cause	1. Acceleration and deceleration times are too short for inertial load. 2. Bad wiring in motor. 3. Motor damage. 4. Problem in AMP board.
		Action	1. Increase acceleration and deceleration times. 2. Check wiring of motor power. 3. Contact the manufacturer/sales agent.
44	CUR SEN ERR U	Meaning	A Current feedback value is abnormal.
		Cause	1. Abnormality/bad connection of servo board. 2. Abnormality/bad connection of AMP board. 3. Abnormality in current sensor setting.
		Action	1. Reconnect servo board. 2. Change AMP board.

45	CUR SEN ERR V	Meaning	A Current feedback value is abnormal.
		Cause	1. Abnormality/bad connection of servo board. 2. Abnormality/bad connection of AMP board. 3. Abnormality in current sensor setting.
		Action	1. Reconnect servo board. 2. Change AMP board.
62	PN 240V LOW	Meaning	Voltage abnormality among PN.
		Cause	1. Bridge diode abnormality. 2. Disconnection/bad connection of PN line. 3. AMP board abnormality.
		Action	1. Change bridge diode. 2. Check PN line. 3. Change AMP.
63	HEAT SINK ERR	Meaning	Heat Sink is overheated.
		Cause	Acceleration and deceleration time is too short for inertia load.
		Action	Reset parameters related to acceleration and deceleration.
64	TMS TIMEOVER	Meaning	The servo board is not answering.
		Cause	1. Incorrect parameter setting. 2. Abnormality in servo board.
		Action	1. Change parameter setting. 2. Change the servo board.
65	SEMA ERR SRV	Meaning	The servo and the main board do not correlate.
		Cause	1. An unused axis may have been set. 2. Bad connection in servo board.
		Action	1. Reset parameters related to axis setting. 2. Dismount the servo board and reinstall.
66	P12V FAIL	Meaning	Abnormality in +12V power (Less than 10.5V).
		Cause	1. Bad connection. 2. Voltage drop (Abnormal SMPS).
		Action	1. Reconnect (SMPS/ Servo board). 2. Change SMPS.

67	M12V FAIL	Meaning	Input voltage -12V abnormality (Less than -10.5V).
		Cause	1. Bad connection. 2. Voltage drop (Abnormal SMPS).
		Action	1. Reconnect (SMPS/ Servo board). 2. Change SMPS.
68	SEMA ERR MAIN	Meaning	Main and servo board do not activate.
		Cause	1. Unused axes have been set as being in use. 2. Bad connection of servo board.
		Action	1. Reset axis parameter. 2. Dismount the servo board and then reinstall.
69	AMP BD NONE	Meaning	The AMP board does not exist.
		Cause	1. Incorrect axis setting. 2. Bad connection of board.
		Action	1. Change axis setting parameter. 2. Reconnect board.
80	SYS EMG	Meaning	System EMG signal is On.
		Cause	System EMG signal input.
		Action	Cancel system EMG.
81	FRONT EMG	Meaning	Front EMG signal is On.
		Cause	Front EMG signal input.
		Action	Cancel front EMG.
82	T/P EMG	Meaning	TP EMG signal is On.
		Cause	TP EMG signal input.
		Action	Cancel TP EMG.
84	MAIN BATT ERR	Meaning	The main board battery has been discharged.
		Cause	The main board battery voltage may be lower than 2.4 V.
		Action	Check / change the main board battery voltage.
86	HOST EMG	Meaning	Host EMG is On.
		Cause	Host EMG is On.
		Action	Reset Host EMG.

87	AC POWER FAIL	Meaning	Input voltage 220V abnormality (Less than 180V).
		Cause	1. Bad connection. 2. Voltage drop (Abnormal SMPS).
		Action	1. Reconnect (SMPS/ Servo board). 2. Change SMPS.
88	5V POWER FAIL	Meaning	Input power 5V is abnormal (less than 4.8V).
		Cause	1. Bad connection. 2. Voltage drop (Abnormal SMPS).
		Action	1. Reconnect (SMPS/ Servo Board). 2. Change SMPS.
89	3.3V POWER FAIL	Meaning	Input power 3.3V is abnormal (less than 3.1V).
		Cause	1. Bad connection. 2. Voltage drop (Abnormal SMPS).
		Action	1. Reconnect (SMPS/ Servo Board). 2. Change SMPS.

2. Error List For Program

2.1. Need LINE START

An attempt was made to register a Line Passing point, an Arc point or a Line End point without first registering a Line Start point.

Register a Line Start point before registering a Line Passing point, an Arc point or a Line End point.

2.2. Need LINE END

An attempt was made to run a program that registered a Line Passing point, an Arc point or a Line Start point without registering a Line End point.

Register a Line End point after registering a Line Passing point, an Arc point or a Line Start point.

2.3. Need Step & Repeat

The Expand Step & Repeat command was given but there is no Step & Repeat instruction currently in the display. Move to the Step & Repeat function you want to expand before selecting Expand Step & Repeat.

2.4. PROGRAM END ERROR

An attempt was made to run a program without registering a Program End.

2.5. LABEL Not Exist

An attempt was made to call a label that does not exist

Section 9: Appendix

1. Appendix A: User I/O Board

1. Outline

User I/O board is the general Input/Output board used to interface controller and external device using I/O signal. The form of signal is as follows:

- Digital Input: 32 contacts
- Digital Output: 32 contacts

2. Structure

Input is received through photo coupler and output given by photo coupler as well.

3. Components of the structure

- Device for the input: photo coupler
- Device for the output: photo coupler (Darlington Transistor Output)

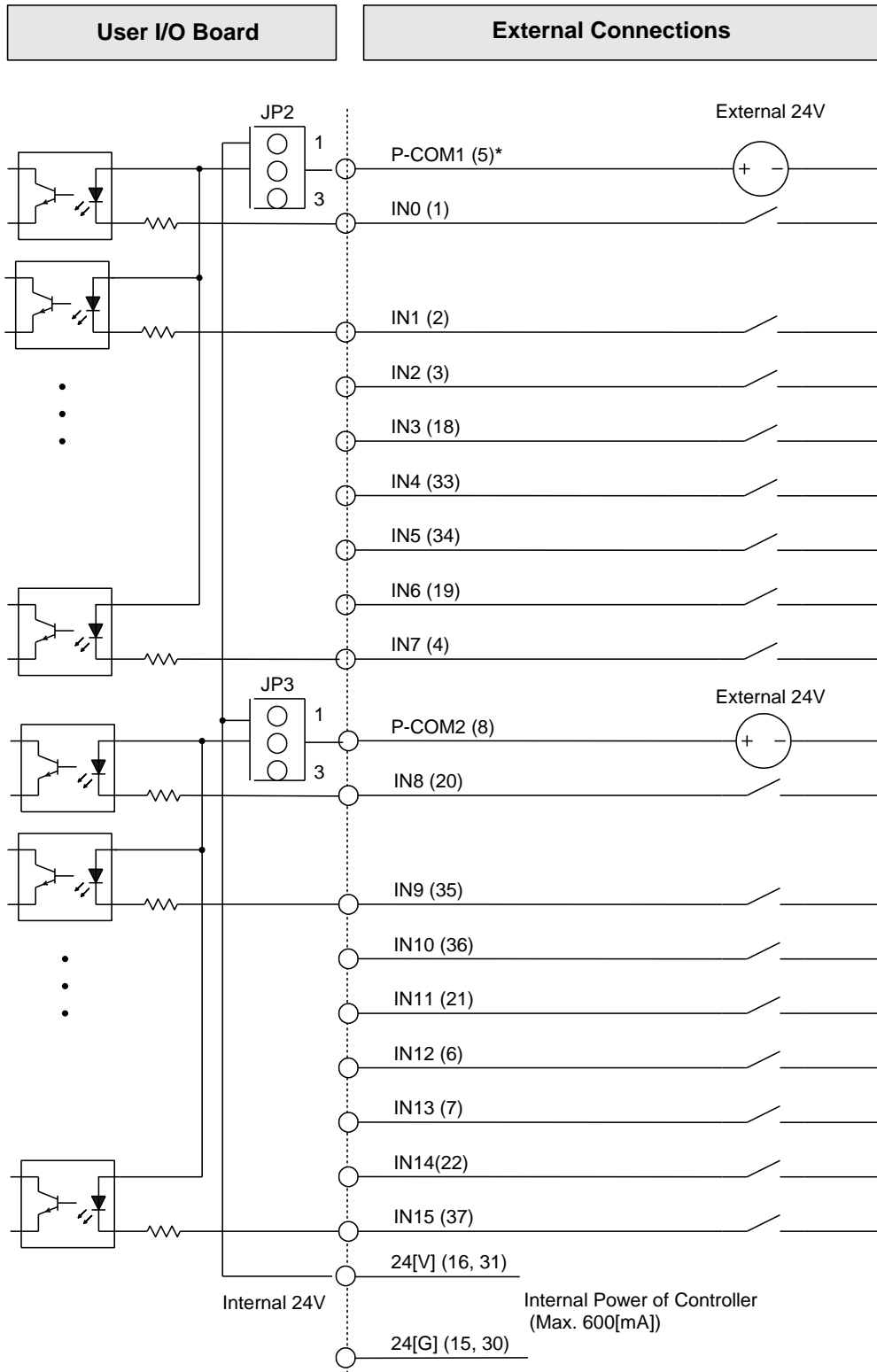
4. Specifications

Items	User Input	User Output
Rated Input/Output voltage	D.C 24[V]	
Rated Input/Output current	Min. 5[mA] / 1 contact	Max. 50 [mA] / 1 contact
Insulation method (Photo coupler)	Input: Insulation voltage (2000 [Vrms]) Output: Insulation voltage (2000 [Vrms])	
Number of Input/Output contacts	32 contacts	32 contacts
Internal Input/Output voltage (Output current)	D.C 24[V] (Max. 600[mA])	

5. Connectors and Input/Output Circuit Diagram

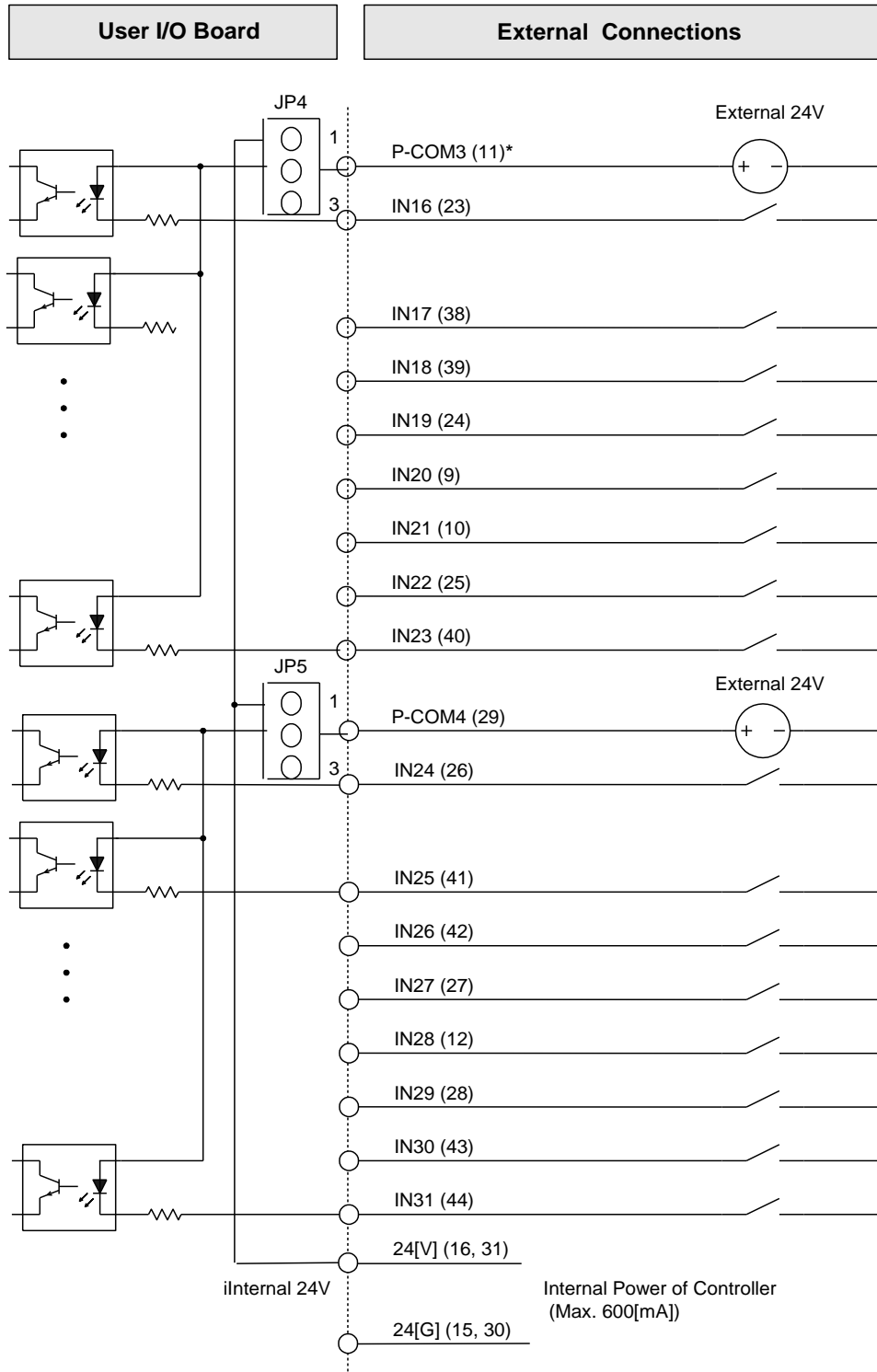
- CN2: External Input Interface Connector
- CN3: External Output Interface Connector

1) Input Circuit and External Interface Circuit (1st ~16th contact): **CN2 Connector**



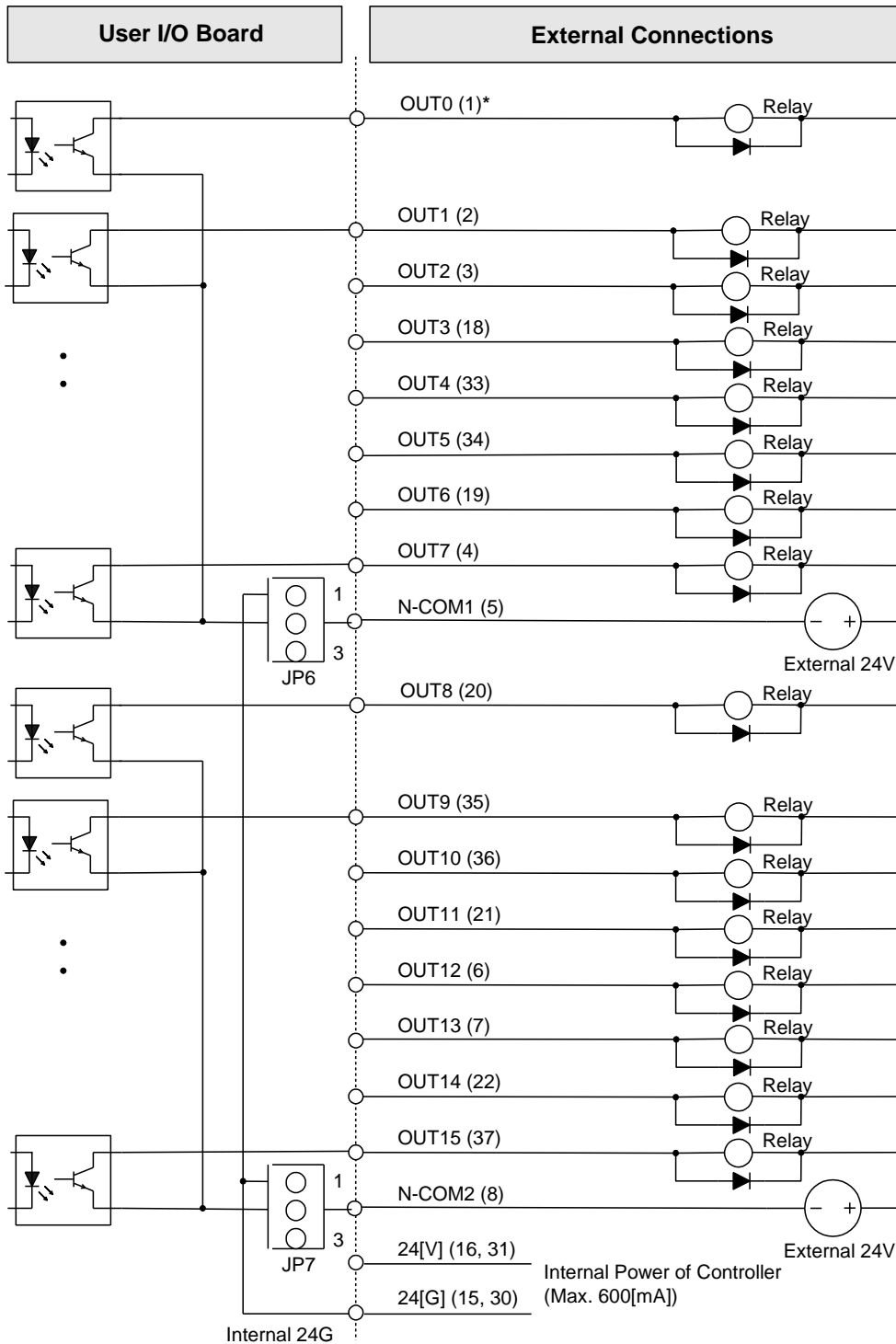
Note: ()* is the Pin Number of the Connector.

Input Circuit and External Interface Circuit (17th ~ 32nd contact): **CN2 Connector**



Note: ()* is the Pin Number of the Connector.

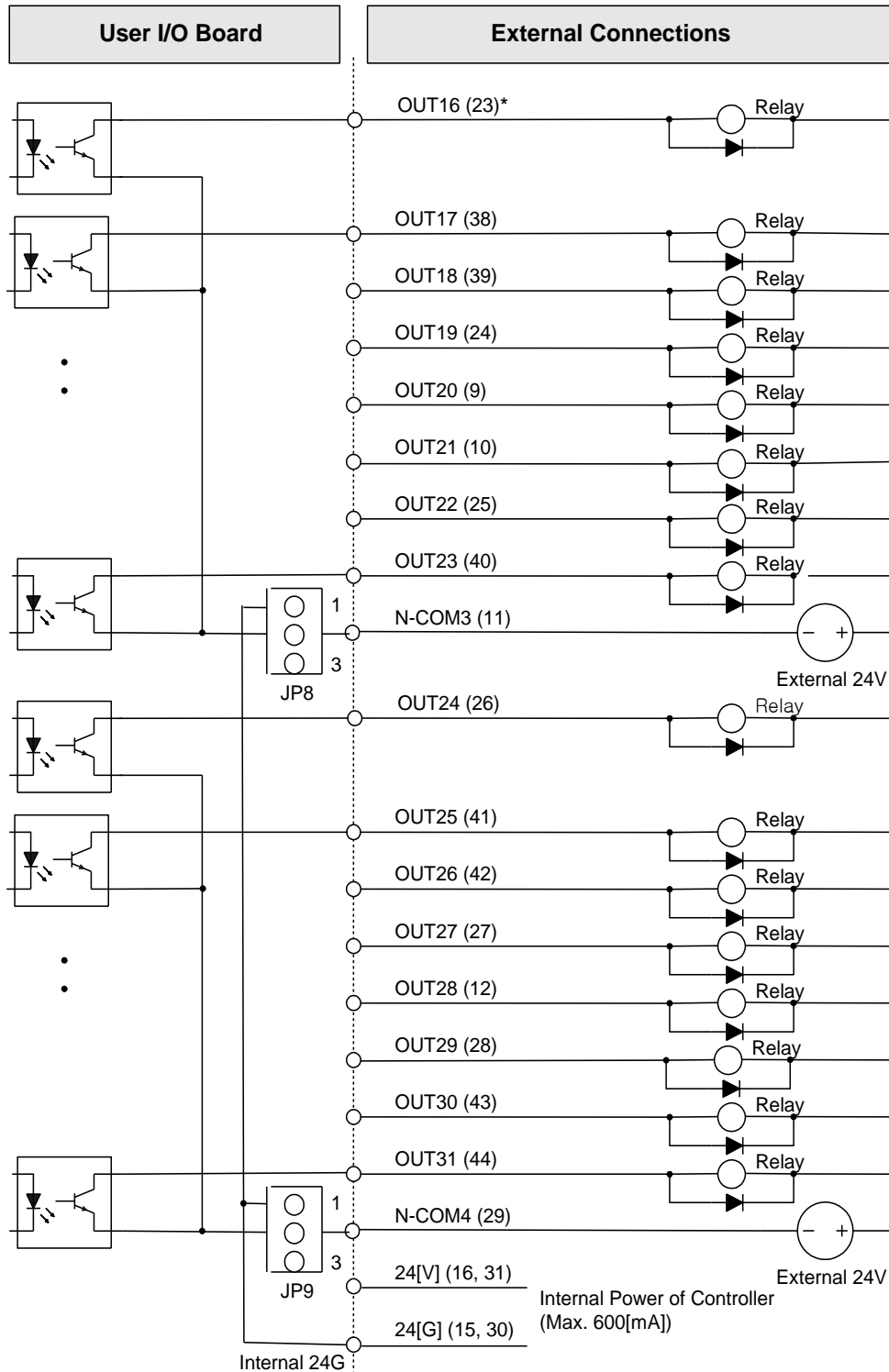
2) Output Circuit and External Interface circuit (1st~16th contact): **CN3 Connector**



Note) In case of inductive load(ex: relay), insert a diode as shown in the figure.

Note: ()* is the Pin Number of the Connector.

Output Circuit and External Interface Circuit (17th ~32nd contact): **CN3 Connector**



Note) In case of inductive load(ex: relay), insert a diode as shown in the figure.

Note: ()* is the Pin Number of the Connector.

6. In/Output Signal Name

1) Input Signal Name

No.	I/O	Sign	Explanation
16, 31	Internal power	24[V]	D.C 24[V] Internal Power
15, 30	Internal grounding	24[G]	D.C 24[G] Internal Ground
1	Input	IN 0	INPUT 1
2	Input	IN 1	INPUT 2
3	Input	IN 2	INPUT 3
18	Input	IN 3	INPUT 4
33	Input	IN 4	INPUT 5
34	Input	IN 5	INPUT 6
19	Input	IN 6	INPUT 7
4	Input	IN 7	INPUT 8
5		P-COM1	24[V] Common
20	Input	IN 8	INPUT 9
35	Input	IN 9	INPUT 10
36	Input	IN 10	INPUT 11
21	Input	IN 11	INPUT 12
6	Input	IN 12	INPUT 13
7	Input	IN 13	INPUT 14
22	Input	IN 14	INPUT 15
37	Input	IN 15	INPUT 16
8		P-COM2	24[V] Common
23	Input	IN 16	INPUT 17
38	Input	IN 17	INPUT 18
39	Input	IN 18	INPUT 19
24	Input	IN 19	INPUT 20
9	Input	IN 20	INPUT 21
10	Input	IN 21	INPUT 22
25	Input	IN 22	INPUT 23
40	Input	IN 23	INPUT 24
11		P-COM3	24[V] Common

No.	I/O	Sign	Explanation
26	Input	IN 24	INPUT 25
41	Input	IN 25	INPUT 26
42	Input	IN 26	INPUT 27
27	Input	IN 27	INPUT 28
12	Input	IN 28	INPUT 29
28	Input	IN 29	INPUT 30
43	Input	IN 30	INPUT 31
44	Input	IN 31	INPUT 32
29		P-COM4	24[V] Common

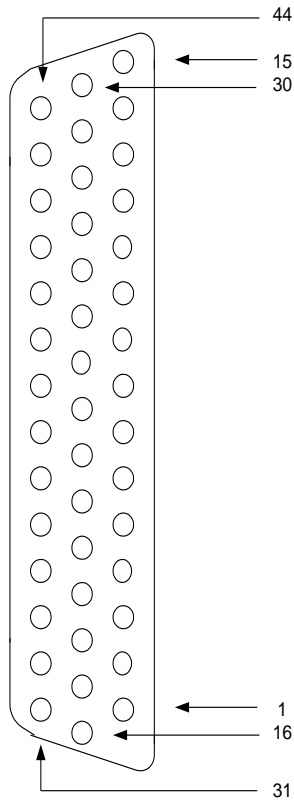
2) Output Signal Name

No.	I/O	Sign	Explanation
16, 31	Internal power	24[V]	D.C 24[V] Internal Power
15, 30	Internal Grounding	24[G]	D.C 24[G] Internal Ground
1	Output	OUT 0	OUTPUT 1
2	Output	OUT 1	OUTPUT 2
3	Output	OUT 2	OUTPUT 3
18	Output	OUT 3	OUTPUT 4
33	Output	OUT 4	OUTPUT 5
34	Output	OUT 5	OUTPUT 6
19	Output	OUT 6	OUTPUT 7
4	Output	OUT 7	OUTPUT 8
5		N-COM1	24[G] Ground Common
20	Output	OUT 8	OUTPUT 9
35	Output	OUT 9	OUTPUT 10
36	Output	OUT 10	OUTPUT 11
21	Output	OUT 11	OUTPUT 12
6	Output	OUT 12	OUTPUT 13
7	Output	OUT 13	OUTPUT 14
22	Output	OUT 14	OUTPUT 15
37	Output	OUT 15	OUTPUT 16
8		N-COM2	24[G] Ground Common

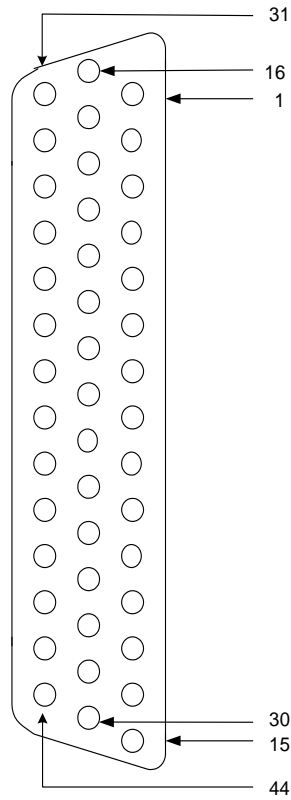
No.	I/O	Sign	Explanation
16, 31	Internal Power	24[V]	D.C 24[V] Internal Power
15, 30	Internal Grounding	24[G]	D.C 24[G] Internal Ground
23	Output	OUT 16	OUTPUT 17
38	Output	OUT 17	OUTPUT 18
39	Output	OUT 18	OUTPUT 19
24	Output	OUT 19	OUTPUT 20
9	Output	OUT 20	OUTPUT 21
10	Output	OUT 21	OUTPUT 22
25	Output	OUT 22	OUTPUT 23
40	Output	OUT 23	OUTPUT 24
11		N-COM3	24[G] Ground Common
26	Output	OUT 24	OUTPUT 25
41	Output	OUT 25	OUTPUT 26
42	Output	OUT 26	OUTPUT 27
27	Output	OUT 27	OUTPUT 28
12	Output	OUT 28	OUTPUT 29
28	Output	OUT 29	OUTPUT 30
43	Output	OUT 30	OUTPUT 31
44	Output	OUT 31	OUTPUT 32
29		N-COM4	24[G] Ground Common

7. Pin Arrangement and Specification of the Connector

Input Connector (CN2)



Output Connector (CN3)



Connector specification:

CN2: D-Sub, 44Pin, Female, Right Angle Type

CN3: D-Sub, 44Pin, Male, Right Angle Type

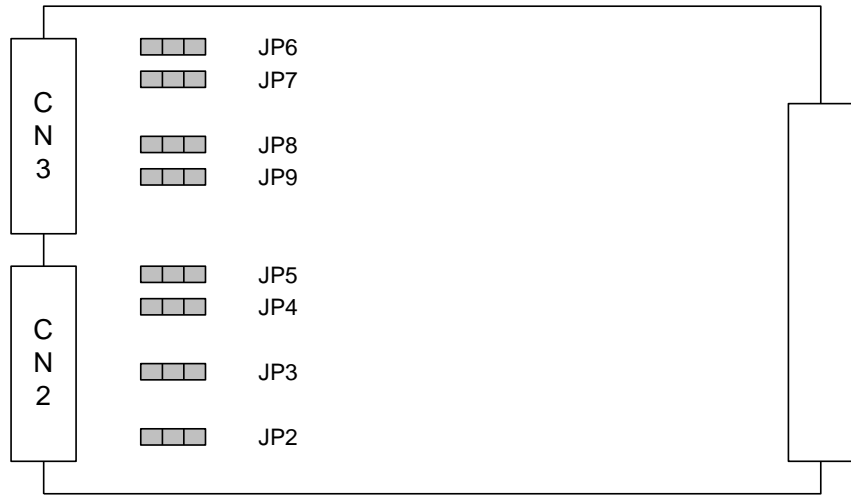
Note 1: When you connect the CN2 connector with external device, use D-Sub, 44Pin, Male, Solder Type.

Note 2: When you connect the CN3 connector with external device, use D-Sub, 44Pin, Female, Solder Type.

Note 3: When you wire the external power, please make sure the connection of D.C 24[V] polarity is properly assigned.

Note 4: If wiring is not properly done, the internal circuit element may be damaged. Take precaution of the polarity of the common terminal.

8. Jumper Setting



No.	Sign	Setting
JP2	24[V] Common (P-COM1)	No. 1-2 Short : internal power use No. 2-3 Short : external power use
JP3	24[V] Common (P-COM2)	No. 1-2 Short : internal power use No. 2-3 Short : external power use
JP4	24[V] Common (P-COM3)	No. 1-2 Short : internal power use No. 2-3 Short : external power use
JP5	24[V] Common (P-COM4)	No. 1-2 Short : internal power use No. 2-3 Short : external power use
JP6	24[G] Ground Common (N-COM1)	No. 1-2 Short : internal grounding use No. 2-3 Short : external grounding use
JP7	24[G] Ground Common (N-COM2)	No. 1-2 Short : internal grounding use No. 2-3 Short : external grounding use
JP8	24[G] Ground Common (N-COM3)	No. 1-2 Short : internal grounding use No. 2-3 Short : external grounding use
JP9	24[G] Ground Common (N-COM4)	No. 1-2 Short : internal grounding use No. 2-3 Short : external grounding use

Note: When the product is delivered, the jumper of the external power and grounding has been set in advance. Please use the jumper setting as external power and grounding if possible, and refrain from using internal power and grounding.

2. Appendix B: System I/O Board

1. Outline

This is the exclusive I/O Board used to interface the controller with I/O signal of the external device.

The form of the signal is as follows.

- Exclusive Digital Input : 24 contacts (3 contacts : Spare)
- Exclusive Digital Output : 6 contacts

2. Structure

Input is received through the photo coupler and output given through photo coupler.

3. Structure Components

- Device for Input Circuit : Photo coupler
- Device for Output Circuit : Photo coupler

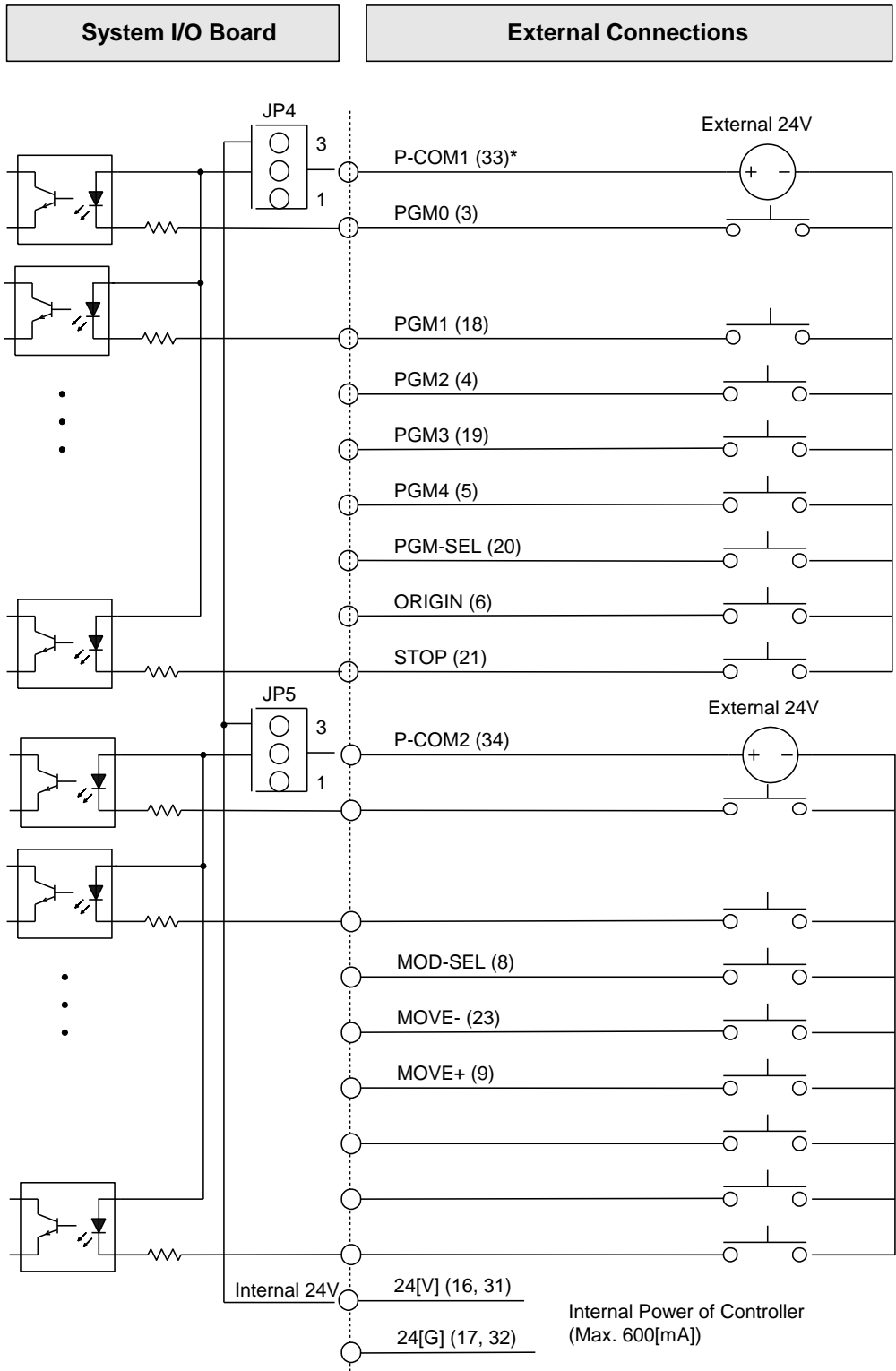
4. Specifications

Items	User Input	User Output
Rated Input/Output Voltage	D.C 24[V]	
Rated Input/Output Current	Min. 5[mA] / 1 contact	Max. 50 [mA] / 1 contact
Insulation Method (Photo Coupler)	Input : Insulation voltage(2000 [Vrms]) Output : Insulation voltage(2000 [Vrms])	
Number of Input/Output Contacts	24 contacts	6 contacts
Internal In/Output Voltage (output current)	D.C 24[V] (Max. 600[mA])	

5. Connector and Input/Output Circuit Diagram

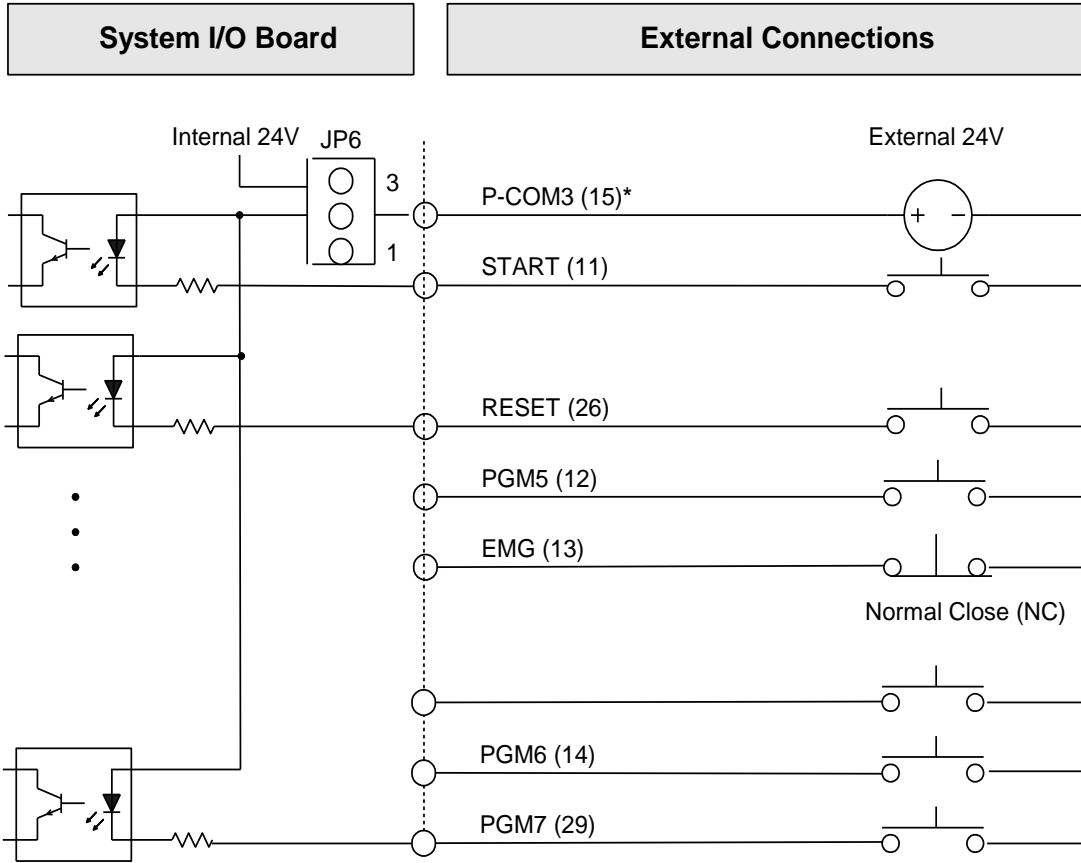
- CN2: External Input/Output Interface Connector

1) Input Circuit and External Interface Circuit (1st ~16th contact)



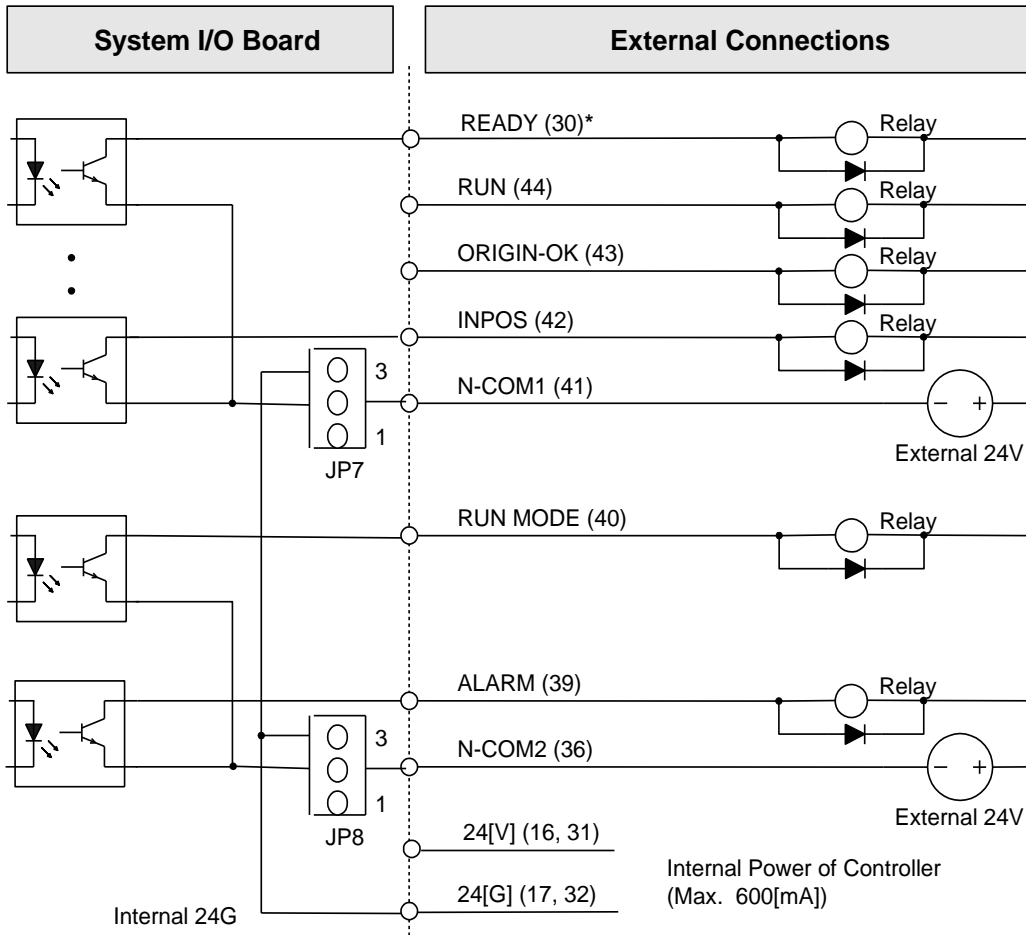
Note: ()* is the Pin Number of the Connector.

Input Circuit and External Interface Circuit (17th ~ 23rd Contact)



Note 1: ()* is the Pin Number of the Connector.

2) Output Circuit and External Interface Circuit (1st ~ 6th contact)



Note) In case of inductive load(ex: relay), insert a diode as shown in the figure.

Note 2: ()* is the Pin Number of the Connector.

6. In/Output Signal Name

1) Input Signal Name

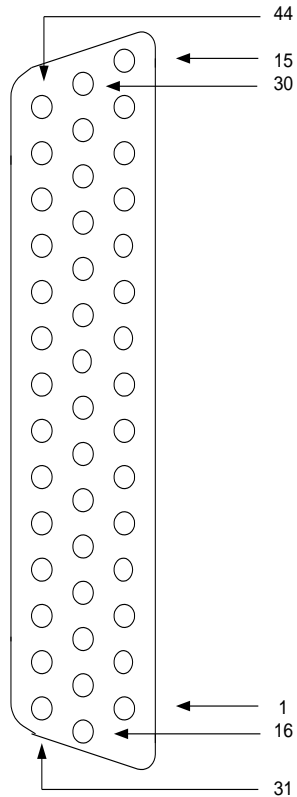
Pin No.	I/O	Sign	Explanation
16, 31	Internal power	24[V]	D.C 24[V] Internal Power
17, 32	Internal grounding	24[G]	D.C 24[G] Internal Ground
3	Input	PGM0	Program #0
18	Input	PGM1	Program #1
4	Input	PGM2	Program #2
19	Input	PGM3	Program #3
5	Input	PGM4	Program #4
20	Input	PGM-SEL	Program selection
6	Input	ORIGIN	Point of origin
21	Input	STOP	Stop
33		P-COM1	24[V] Common
7	Input		Reserved
22	Input		Reserved
8	Input		Reserved
23	Input		Reserved
9	Input		Reserved
24	Input		Reserved
10	Input		Reserved
25	Input		Reserved
34		P-COM2	24[V] Common
11	Input	START	Start
26	Input	RESET	Reset
12	Input	PGM5	Program #5
13	Input	EMG	Emergency stop
28	Input		Reserved
14	Input	PGM6	Program #6
29	Input	PGM7	Program #7
15		P-COM3	24[V] Common

2) Output Signal Name

No.	I/O	Sign	Explanation
30	Output	READY	Preparation complete
44	Output	RUN	Robot Running
43	Output	ORIGIN-OK	Point of origin
42	Output	INPOS	In Position
41	Output	N-COM1	24[G] Ground Common
40	Output	RUN MODE	In Run Mode
39	Output	ALARM	Error
36	Output	N-COM2	24[G] Ground Common

7. Pin Arrangement and Specification of the Connector

Input/Output Connector (CN2)



Connector Specification

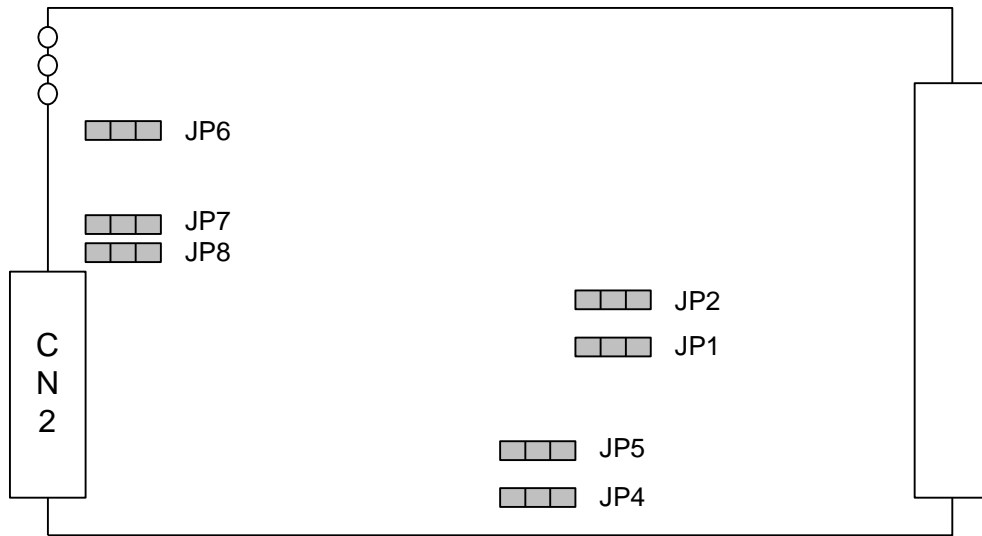
CN2: D-Sub, 44Pin, Female, Right Angle Type

Note 1: When you connect the CN2 connector with the external device, use D-Sub, 44Pin, Male, Solder Type.

Note 2: Do not use internal power. Contact the manufacturer if internal power is needed.

Note 3: If wiring is not properly done, the internal circuit element may be damaged. Note the polarity of the common terminal.

8. Jumper Setting



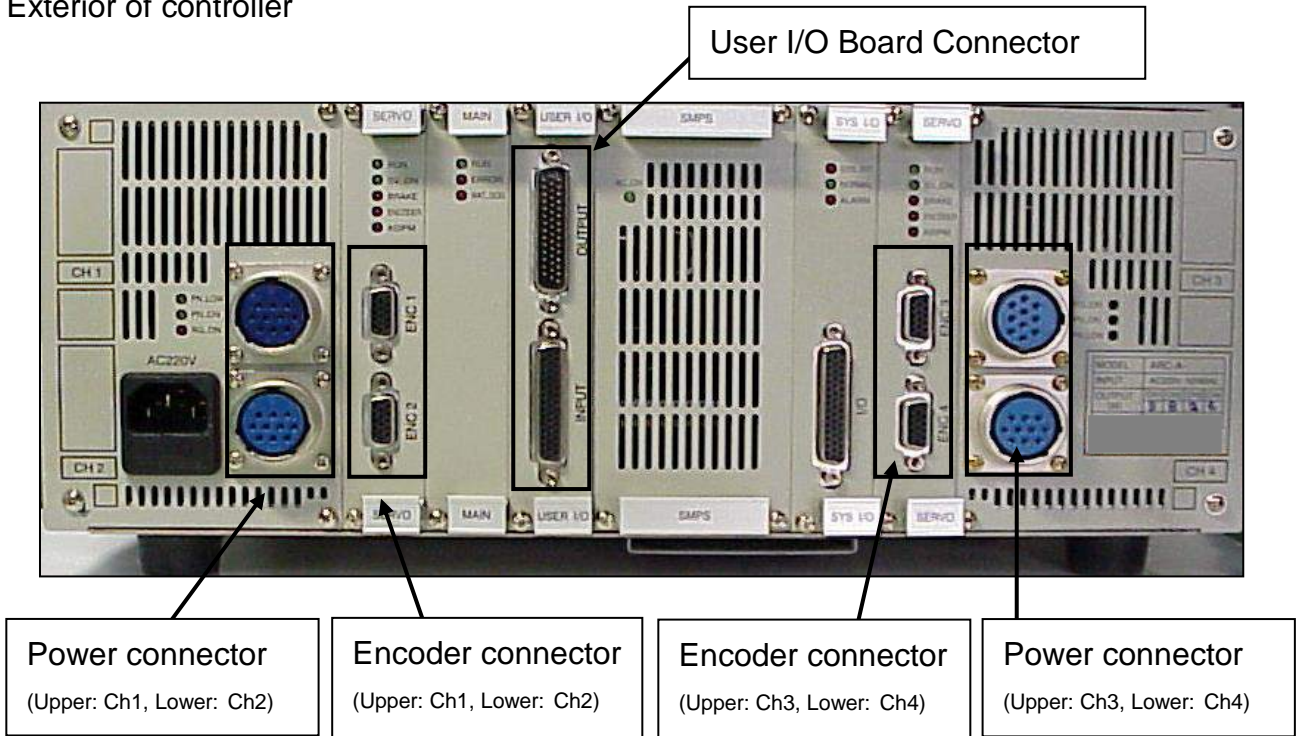
No.	Sign	Setting
JP1	External → Control power On	Unused
JP2	Emergency Stop	1-2 Short: Normal check 2-3 Short: No check
JP4	24[V] Common (P-COM1)	1-2 Short: external power use 2-3 Short: internal power use
JP5	24[V] Common (P-COM2)	1-2 Short: external power use 2-3 Short: internal power use
JP6	24[V] Common (P-COM3)	1-2 Short: external power use 2-3 Short: internal power use
JP7	24[G] Ground Common (N-COM1)	1-2 Short: external grounding use 2-3 Short: internal grounding use
JP8	24[G] Ground Common (N-COM2)	1-2 Short: external grounding use 2-3 Short: internal grounding use

Note: When the product is delivered, the jumpers of the external power and grounding have been set in advance. Please use jumper setting as external power and grounding if possible, and refrain from using internal power and grounding.

3. Appendix C: Equipment (machine) Connection

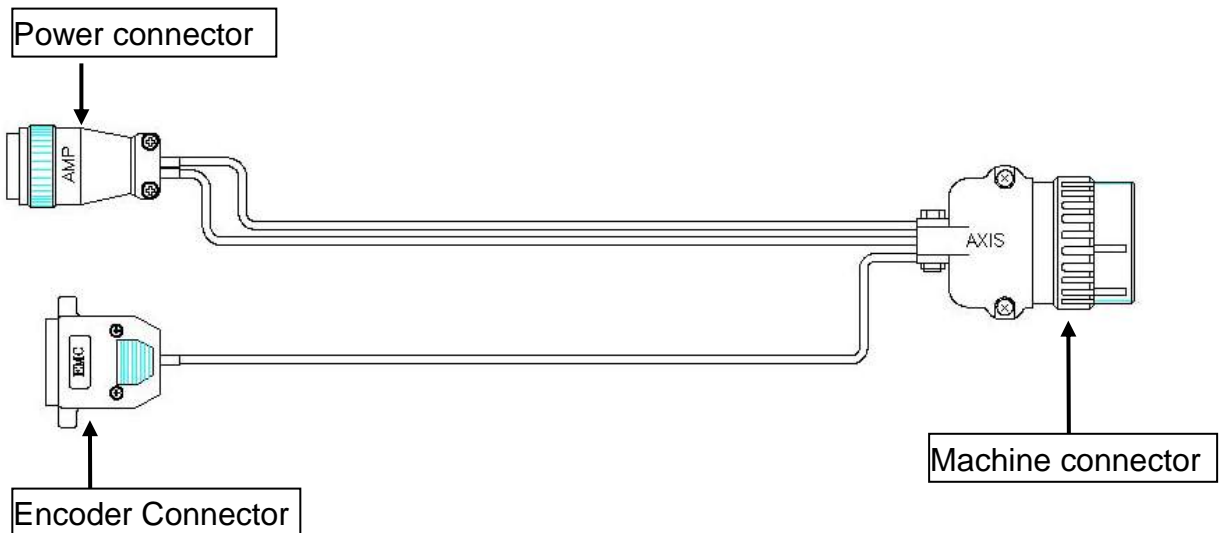
A Separate Harness connecting cable is used to connect the manipulator and controller.

1. Exterior of controller



2. Cables for the F9000N Cartesian coordinate robot

1) Exterior View of the Separate Harness



2) Input/Output signal name of the Separate Harness

The encoder connector is connected with the **servo board** connector of the controller.

No	Signal Name	No	Signal Name	No	Signal Name
1	A	6	/A	11	-
2	B	7	/B	12	-
3	Z	8	/Z	13	-
4	-	9	-	14	GND
5	VCC(+5V)	10	-	15	-

The power connector is connected with the **amp. board** of the controller.

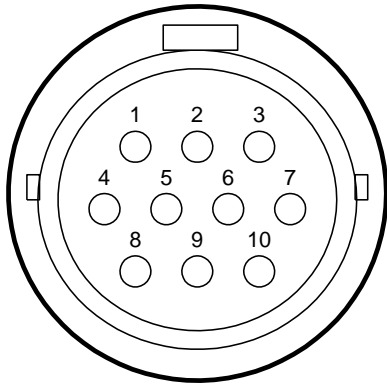
No	Signal Name	No.	Signal Name
1	U phase	6	Brake
2	+24[V]	7	Right Limit Sensor
3	Left Limit Sensor	8	W phase
4	V phase	9	24GND
5	F.G	10	Origin Sensor

The machine connector is connected with the connector of Cartesian coordinate robot.

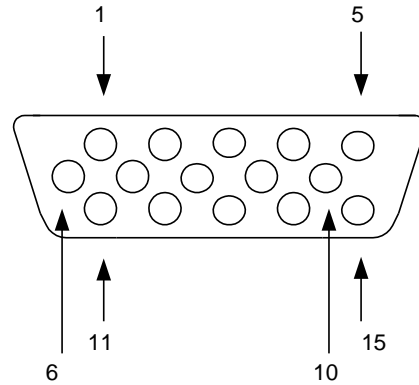
No	Signal Name	No	Signal Name
1	VCC(+5V)	15	-
2	-	16	-
3	GND	17	Left Limit Sensor (N.C)
4	-	18	Right Limit Sensor (N.C)
5	A	19	+24[V]
6	/A	20	24GND
7	-	21	Origin Sensor (N.O)
8	-	22	+24[V]
9	B	23	Brake
10	/B	24	-
11	-	A	U phase
12	-	B	V phase
13	Z	C	W phase
14	/Z	D	F.G

3) Pin assignment of the connector for the Separate Harness

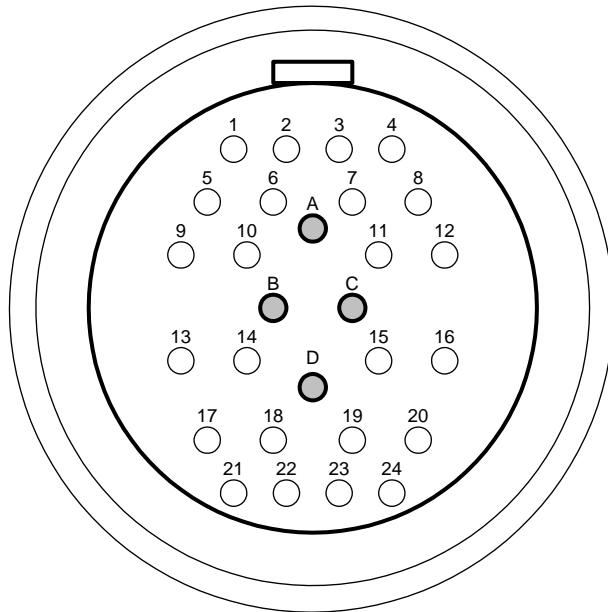
• Power connector



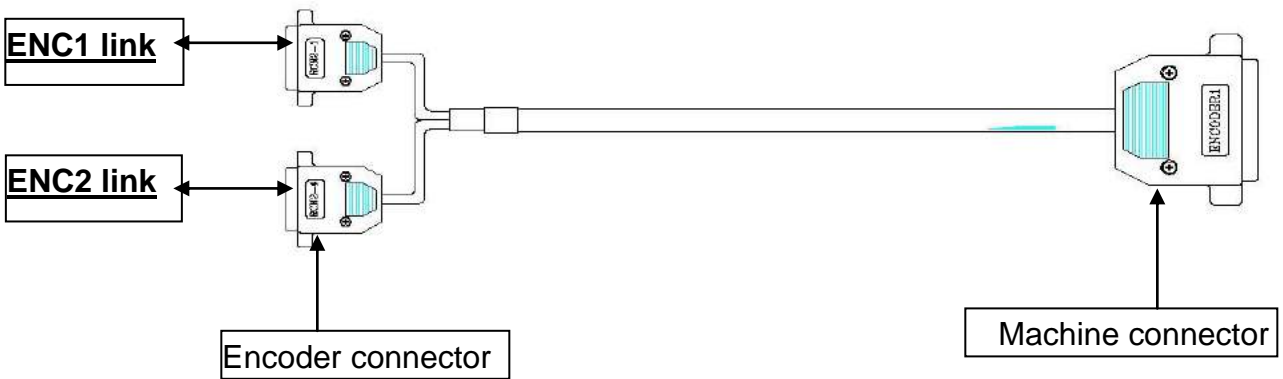
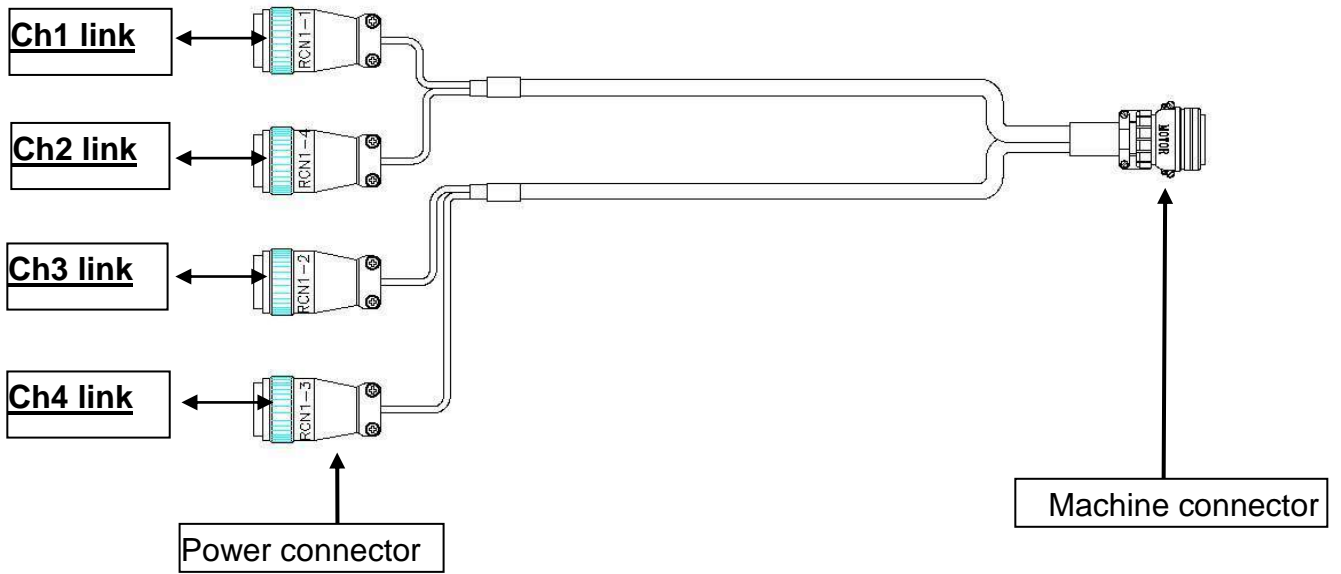
• Encoder connector

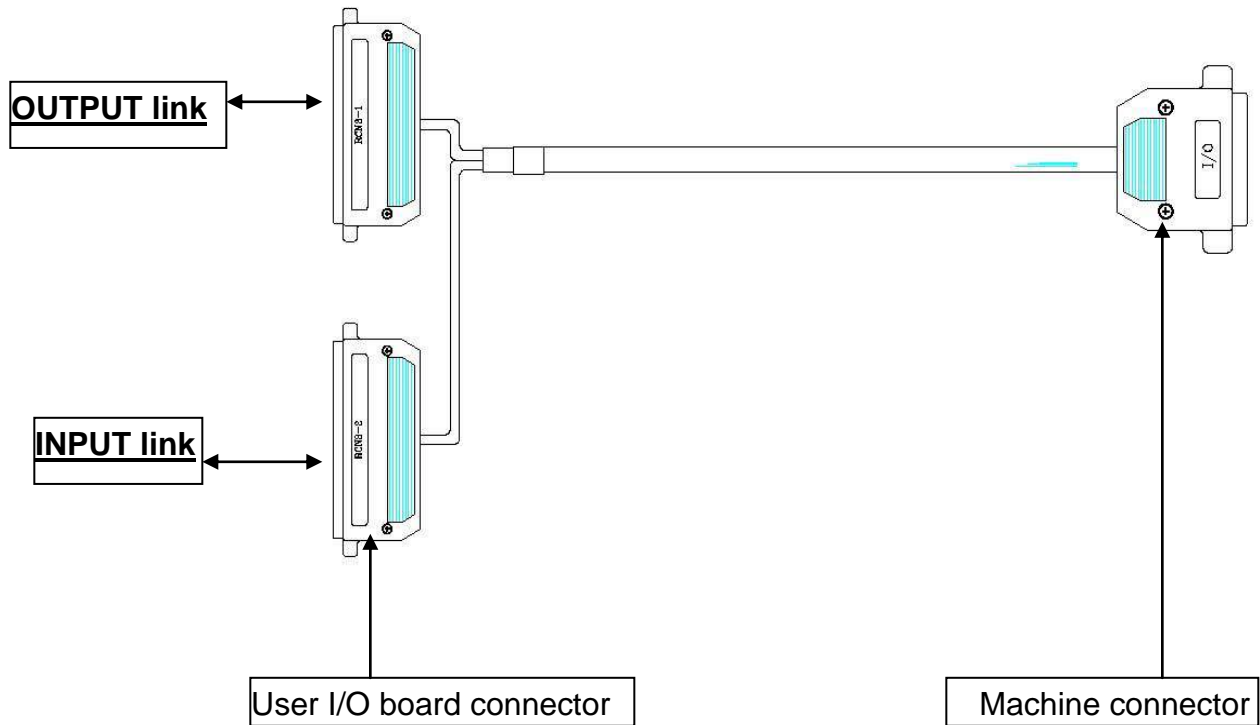
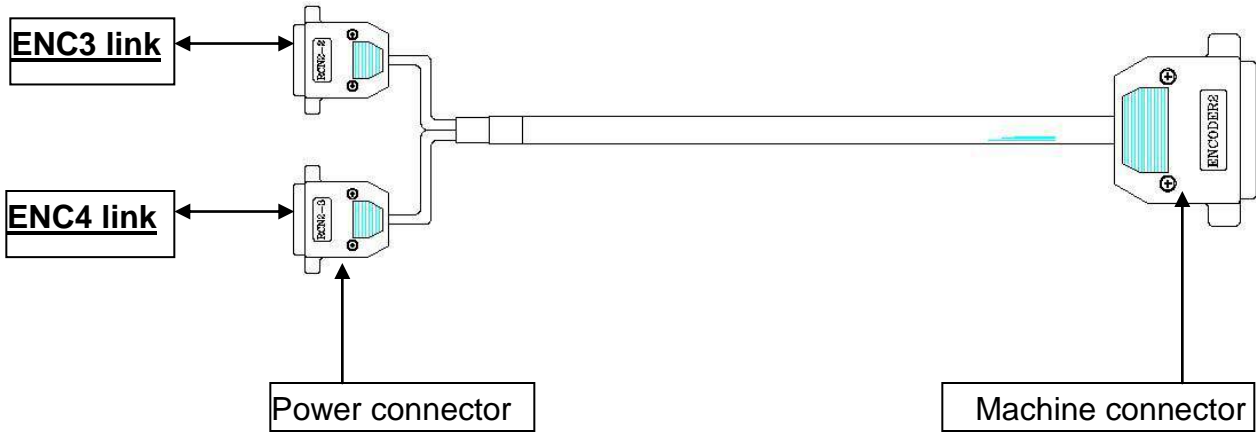


• Machine connector



3. Cables for the TMB200R/300R SCARA robot
1) Exterior view of the Separate Harness





2) Input/Output Signal Name of the Separate Harness

A) Separate harness for the Encoder

Encoder connector (**RCN2-1**) is connected with the Servo board connector (**ENC1**).

No	Signal Name	No	Signal Name	No	Signal Name
1	1_A	6	1_/A	11	1_VCC(+5V)
2	1_B	7	1_/B	12	-
3	1_Z	8	1_/Z	13	-
4	1_Rx	9	1_/Rx	14	-
5	1_PRESET	10	-	15	1_GND

Encoder connector (**RCN2-4**) is connected with the Servo board connector (**ENC2**).

No	Signal Name	No	Signal Name	No	Signal Name
1	4_A	6	4_/A	11	4_VCC(+5V)
2	4_B	7	4_/B	12	-
3	4_Z	8	4_/Z	13	-
4	4_Rx	9	4_/Rx	14	-
5	4_PRESET	10	-	15	4_GND

The machine connector is connected with the **ENCODER1** connector.

No	Signal Name	No	Signal Name
1	1_A	17	4_A
2	1_B	16	4_B
3	1_Z	15	4_Z
4	1_Rx	14	4_Rx
9	1_PRESET	18	4_PRESET
5	1_/A	22	4_/A
6	1_/B	21	4_/B
7	1_/Z	20	4_/Z
8	1_/Rx	19	4_/Rx
10	1_VCC(+5V)	24	4_VCC(+5V)
11	1_GND	23	4_GND
25	Shield	25	Shield

Encoder connector (**RCN2-2**) is connected to the **Servo board** connector (**ENC3**).

No	Signal Name	No	Signal Name	No	Signal Name
1	2_A	6	2_/A	11	2_VCC(+5V)
2	2_B	7	2_/B	12	-
3	2_Z	8	2_/Z	13	-
4	2_Rx	9	2_/Rx	14	-
5	2_PRESET	10	-	15	2_GND

The encoder connector (**RCN2-3**) is connected with the **Servo board** connector (**ENC4**).

No	Signal Name	No	Signal Name	No	Signal Name
1	3_A	6	3_/A	11	3_VCC(+5V)
2	3_B	7	3_/B	12	-
3	3_Z	8	3_/Z	13	-
4	3_Rx	9	3_/Rx	14	-
5	3_PRESET	10	-	15	3_GND

The machine connector is connected with the **ENCODER2** connector.

No	Signal Name	No	Signal Name
1	2_A	17	3_A
2	2_B	16	3_B
3	2_Z	15	3_Z
4	2_Rx	14	3_Rx
9	2_PRESET	18	3_PRESET
5	2_/A	22	3_/A
6	2_/B	21	3_/B
7	2_/Z	20	3_/Z
8	2/Rx	19	3_/Rx
10	2VCC(+5V)	24	3_VCC(+5V)
11	2GND	23	3_GND
25	Shield	25	Shield

B) Separate harness for the Power

Power connector (**RCN1-1**) is connected with the **Amp. Board** connector (**Ch1**).

No	Signal Name	No	Signal Name
1	1_U phase	8	1_W phase
4	1_V phase	5	F.G

Power connector (**RCN1-4**) is connected with the **Amp. Board** connector (**Ch2**).

No	Signal Name	No	Signal Name
1	4_U phase	8	4_W phase
4	4_V phase		

Power connector (**RCN1-2**) is connected with the **Amp. Board** connector (**Ch3**).

No	Signal Name	No	Signal Name
1	2_U phase	8	2_W phase
4	2_V phase		

Power connector (**RCN1-3**) is connected with the **Amp. Board** connector (**Ch4**).

No	Signal Name	No	Signal Name
1	3_U phase	2	+24[V]
4	3_V phase	9	24GND
8	3_W phase	6	Brake

The machine connector is connected with the **Motor** connector.

No	Signal Name	No	Signal
A	1_U phase	E	2_V phase
B	1_V phase	G	2_W phase
C	1_W phase	F	3_U phase
S	F.G	J	3_V phase
D	2_U phase	L	3_W phase
M	+24[V]	H	4_U phase
P	24GND	K	4_V phase
R	Brake	N	4_W phase

C) Separate harness for the User I/O

RCN3-1 is connected with the **User I/O board** connector (**CN2**).

No	Signal Name	No	Signal Name
16	+24[V]	43	INPUT 30
12	INPUT 28	44	INPUT 31
28	INPUT 29	15	24GND

RCN3-2 is connected with the **User I/O board** connector (**CN3**).

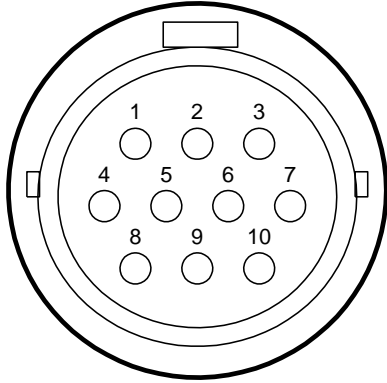
No	Signal	No	Signal Name
26	OUTPUT 24	12	OUTPUT 28
41	OUTPUT 25	28	OUTPUT 29
42	OUTPUT 26	43	OUTPUT 30
27	OUTPUT 27	44	OUTPUT 31

Machine connector is connected with the **I/O** connector.

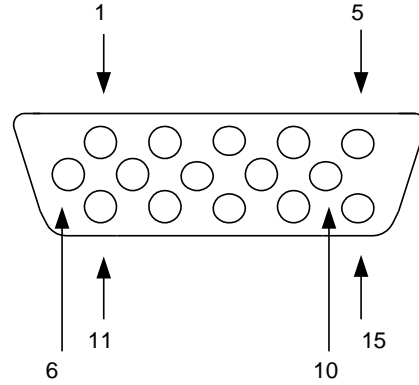
No	Signal Name	No	Signal Name
1	+24[V]	3	OUTPUT 25
10	INPUT 28	4	OUTPUT 26
11	INPUT 29	5	OUTPUT 27
12	INPUT 30	6	OUTPUT 28
13	INPUT 31	7	OUTPUT 29
9	24GND	8	OUTPUT 30
2	OUTPUT 24	15	OUTPUT 31

3) Pin assignment of the connector for the Separate Harness

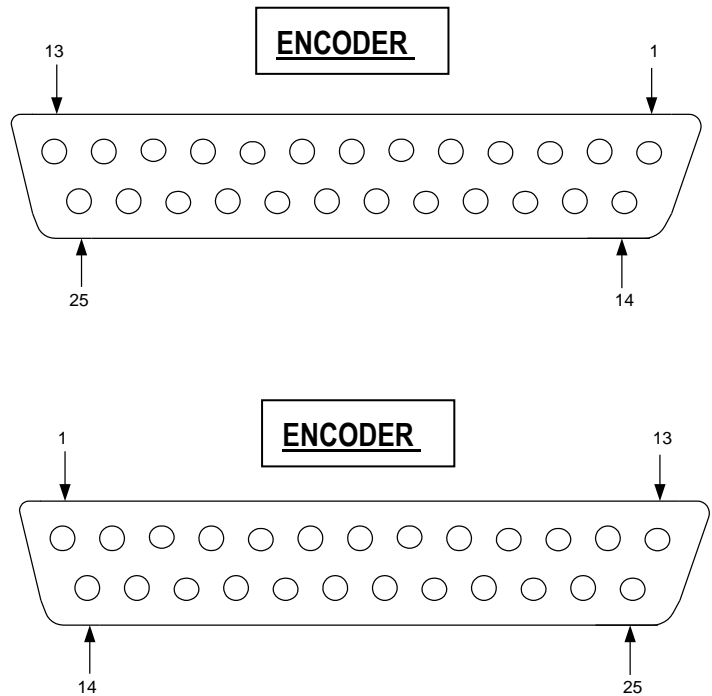
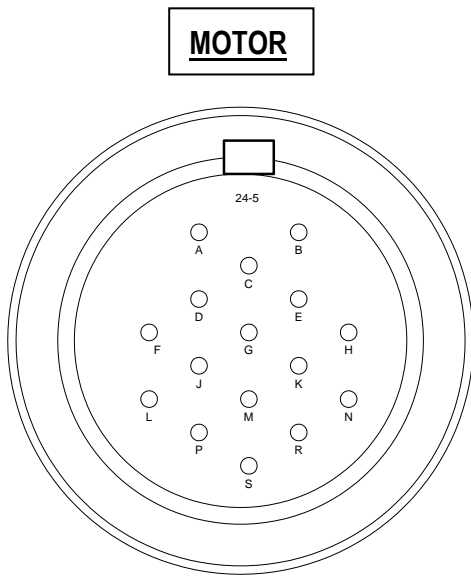
• Power connector



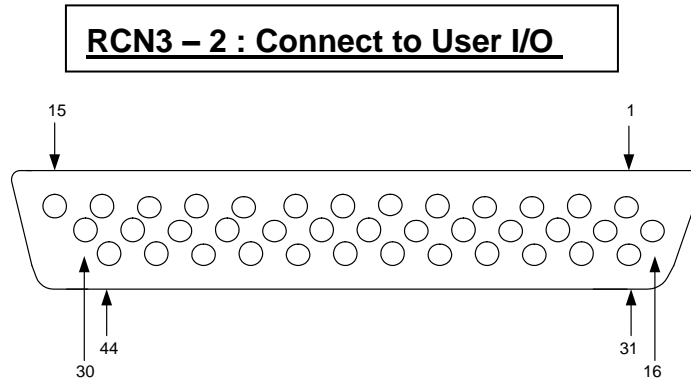
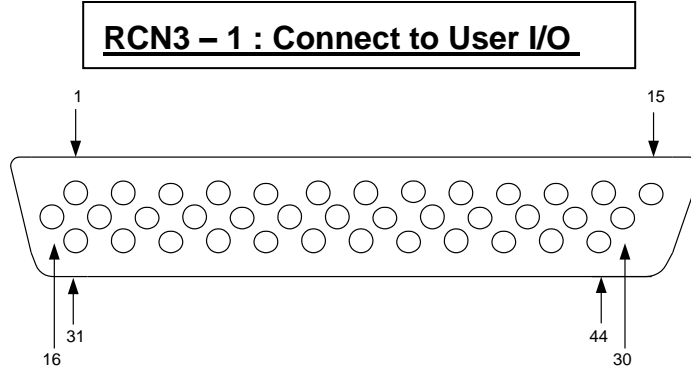
• Encoder connector



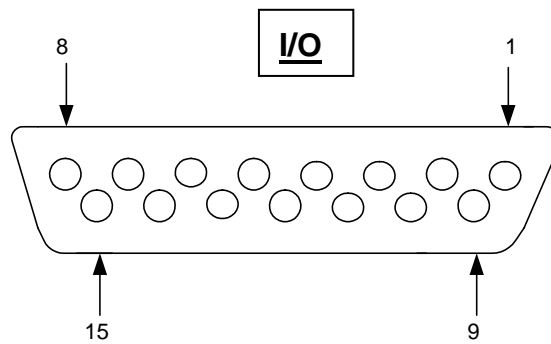
• Machine connector



- Connector for the User I/O Board (Controller side)

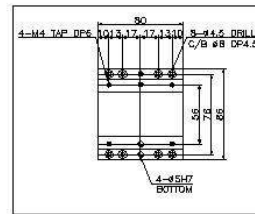
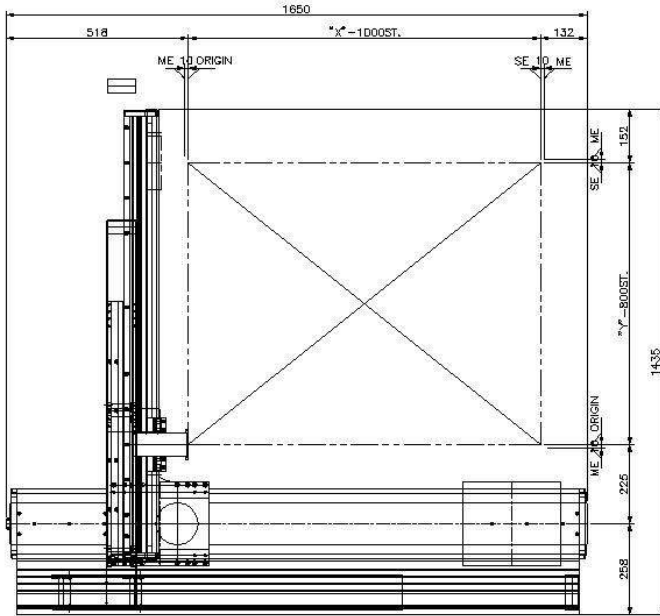


- Connector for the User I/O Board (Machine side)

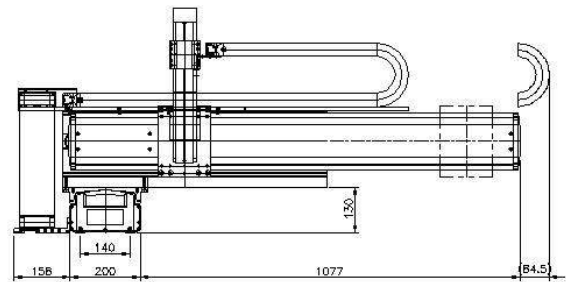
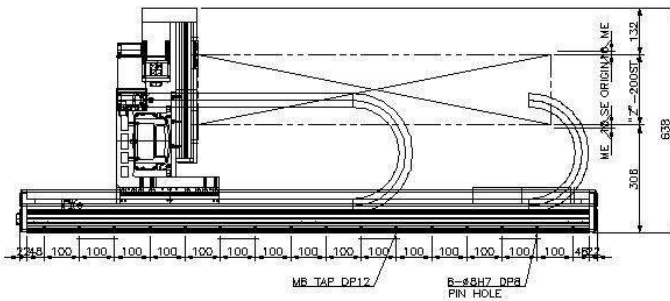


4. Appendix D : Machine Dimensions

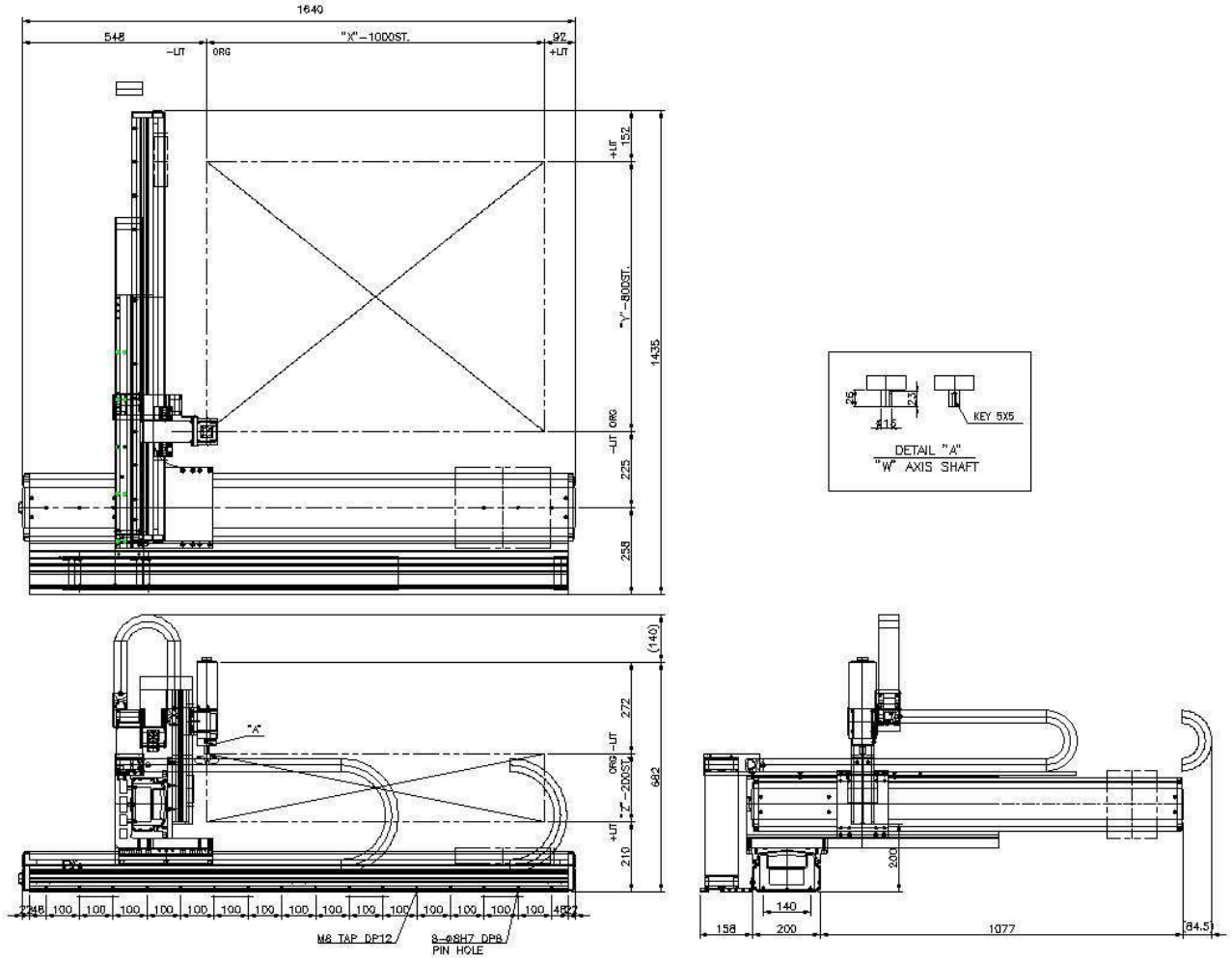
4.1. F8100N



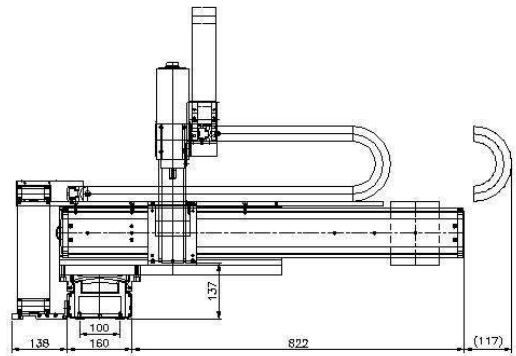
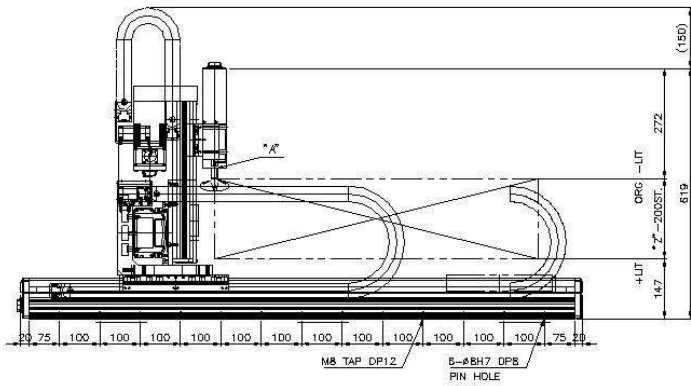
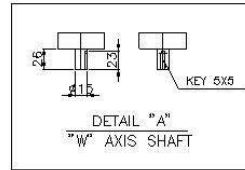
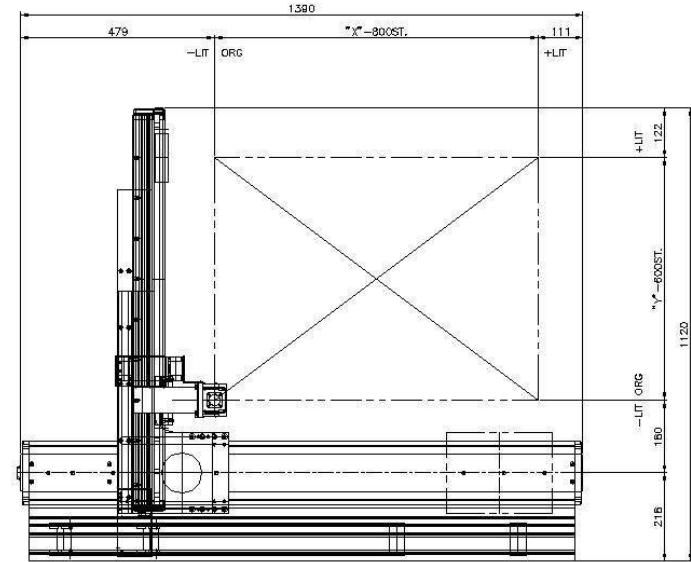
"Z" SLIDE TABLE
 SCALE 3:1



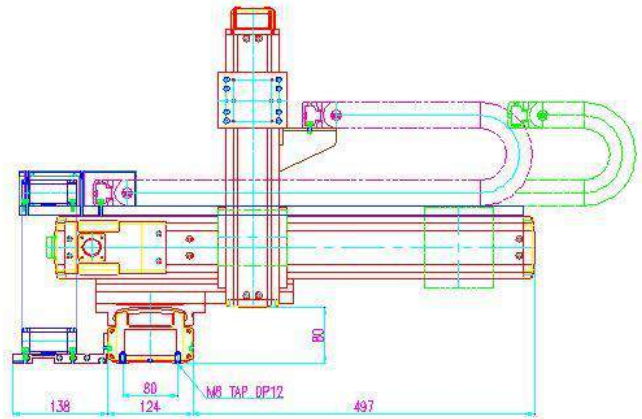
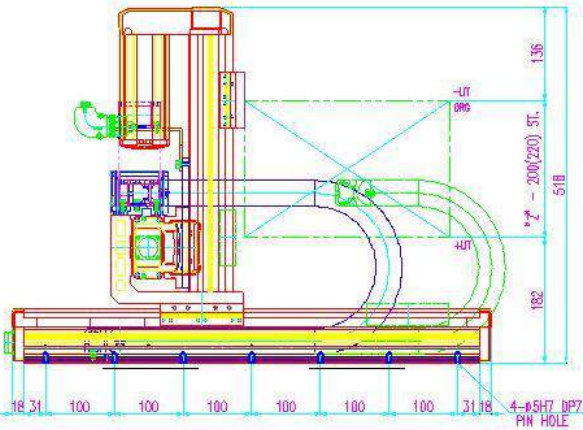
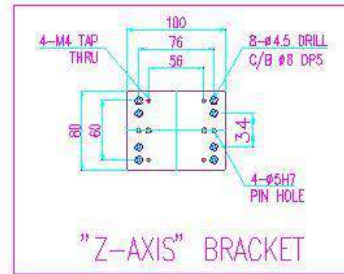
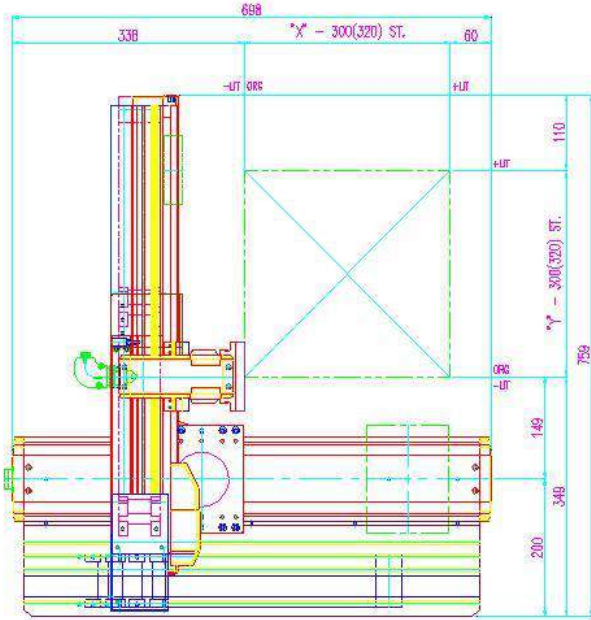
4.2. F8104N



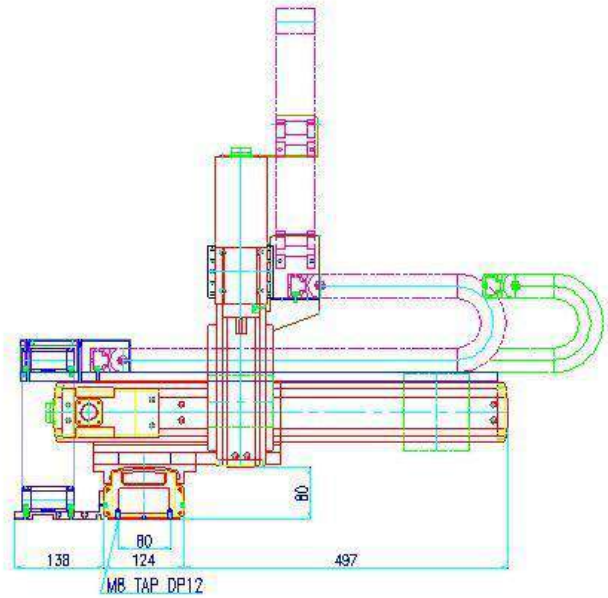
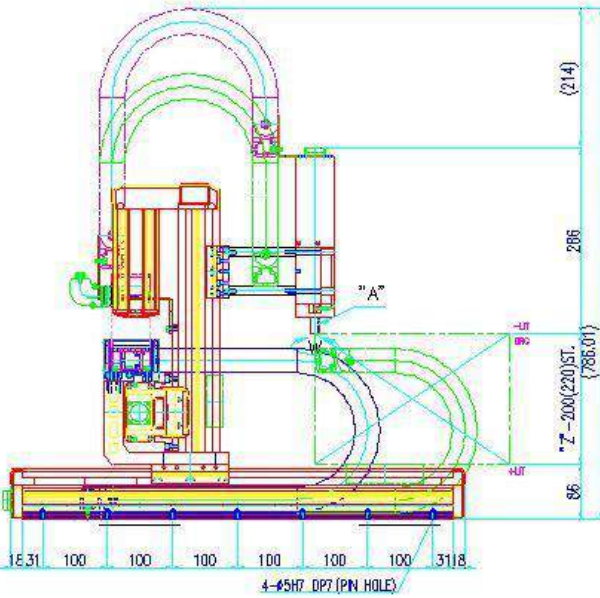
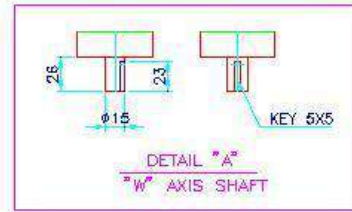
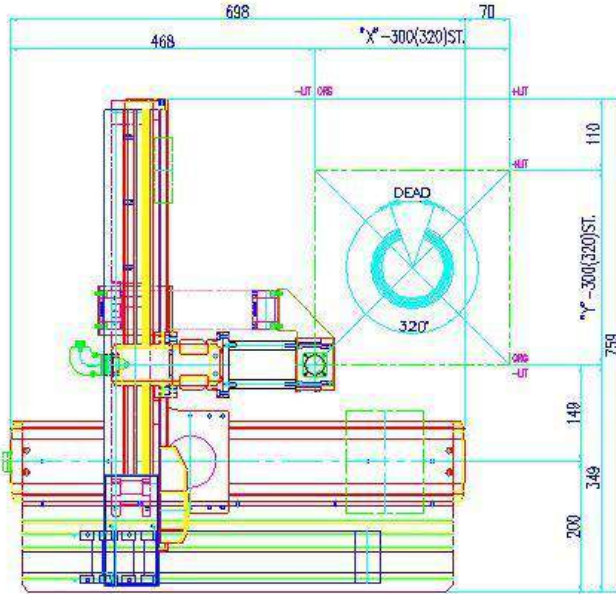
4.4. F8804N



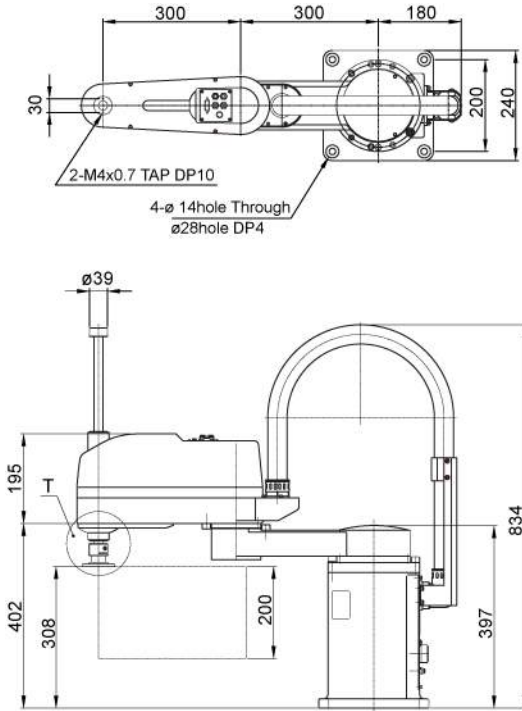
4.5. F9300N



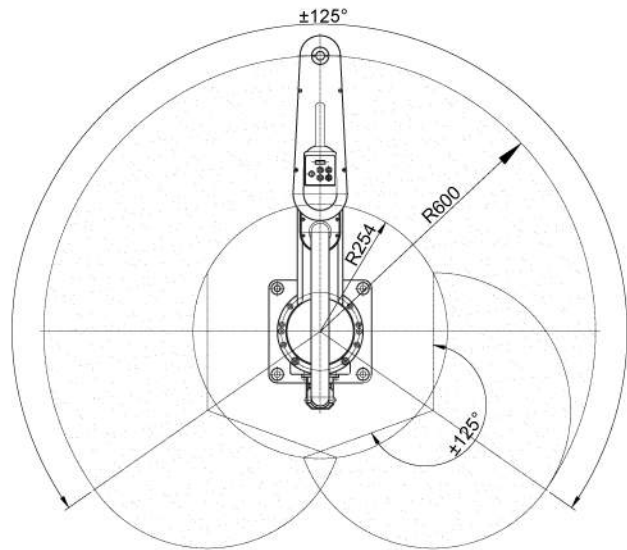
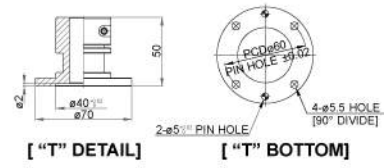
4.6. F9304N



4.11. F2004N



(UNITS: mm)

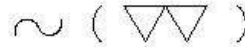
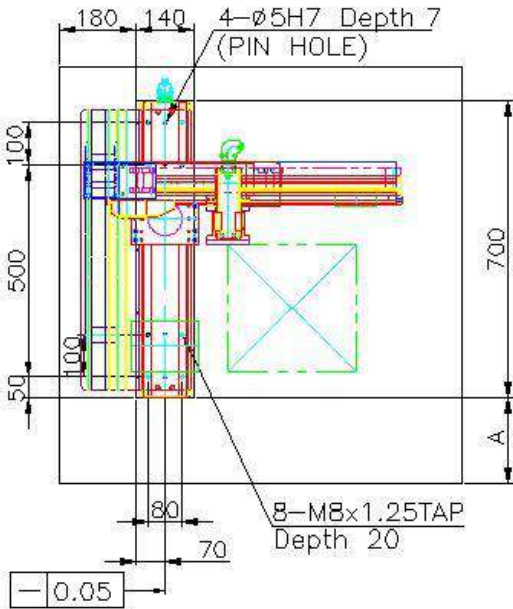


(WORKING ENVELOPE)

5. Appendix E : Table Dimensions

(Optional)

5.1. F9300N/F9304N

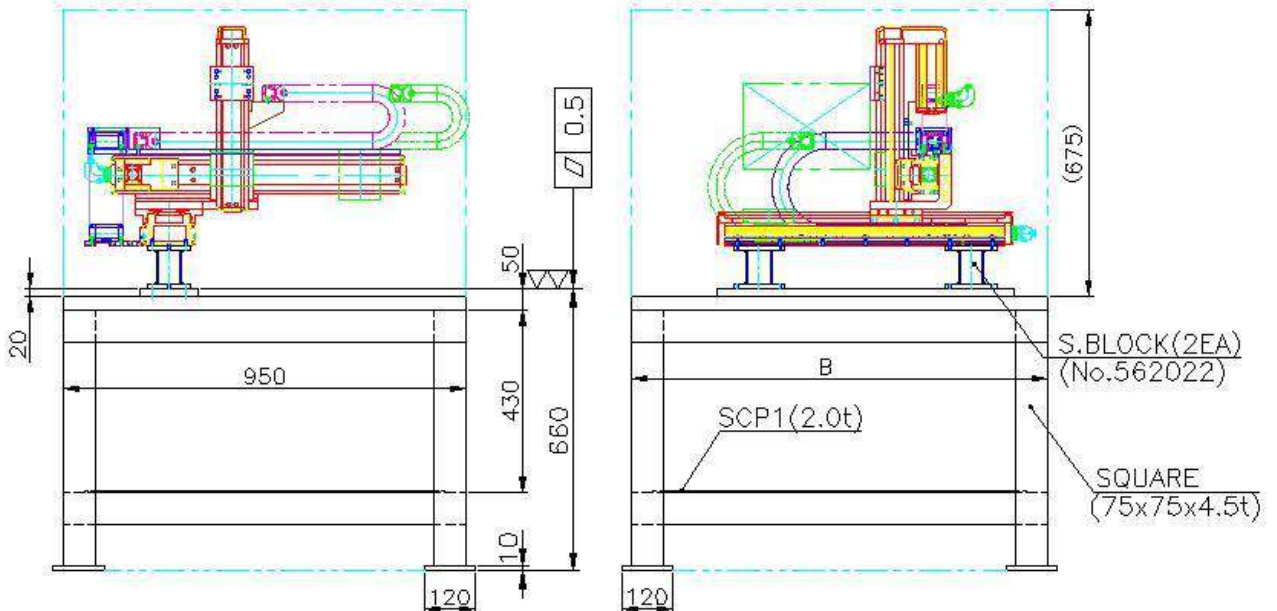


NOTE

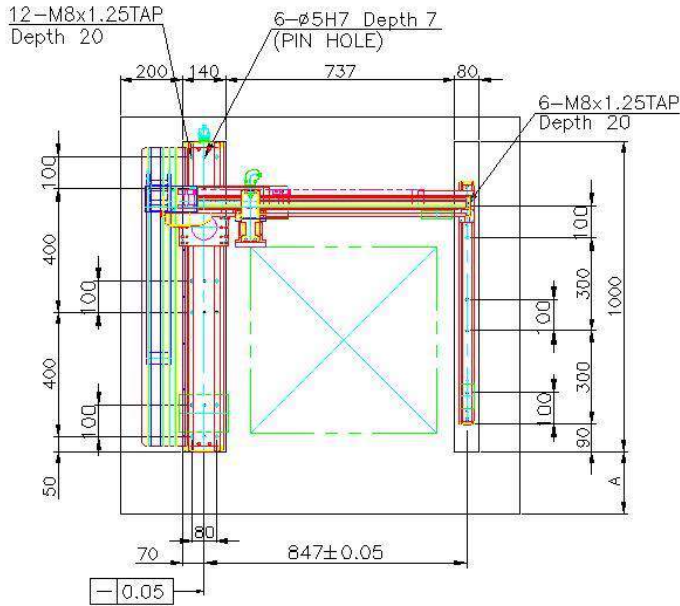
1. Chamfers without designation; C1
2. Surface treatment; Black painting
3. WEIGHT; Over 270kg

MODEL	F9300N	F9304N
A	100	200
B	880	980

(UNITS: mm)



5.2. F9600N/F9604N



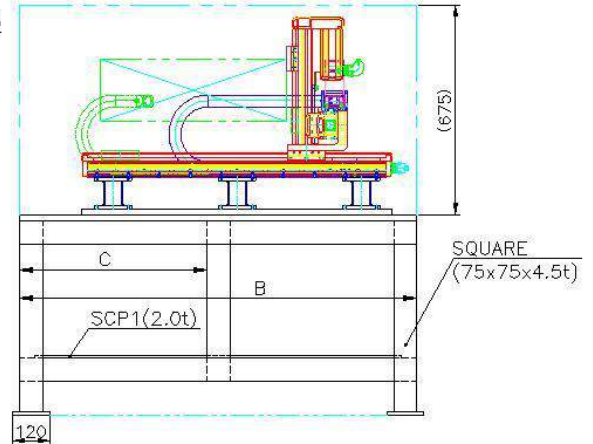
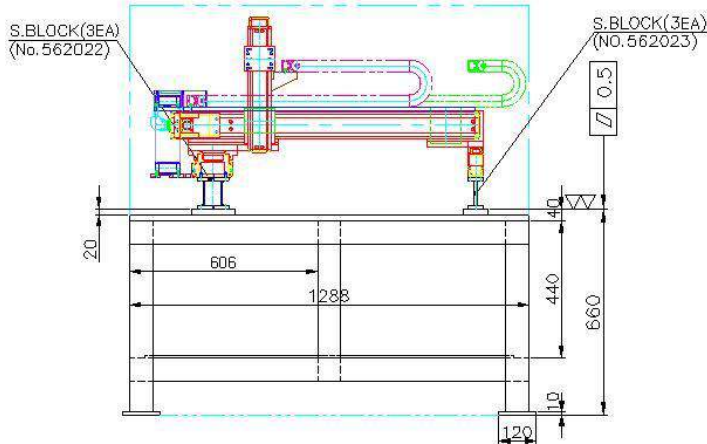
$\sim (\nabla)$

NOTE

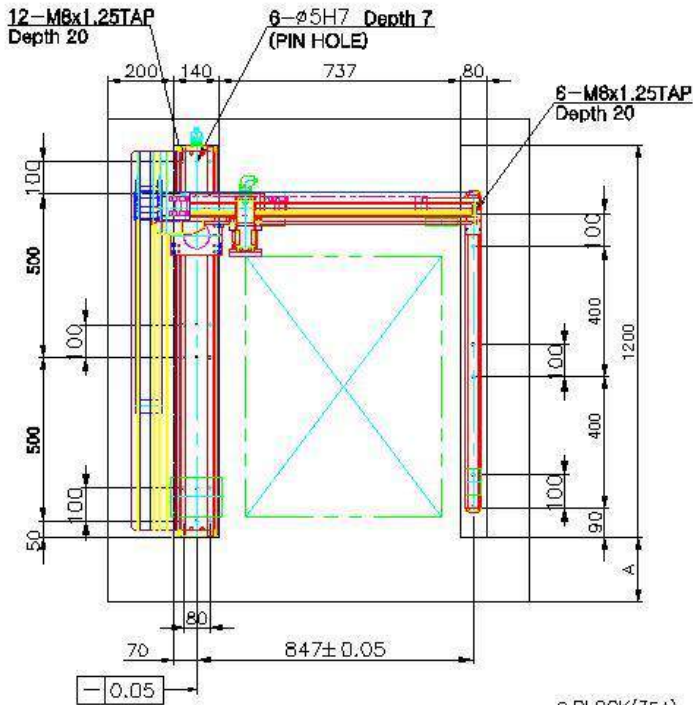
1. Chamfers without designation; C1
2. Surface treatment; Black painting
3. Weight; OVER 300kg

MODEL	F9600N	F9604N
A	100	200
B	1180	1280
C	552	602

(UNITS: mm)



5.3. F9800N/F9804N

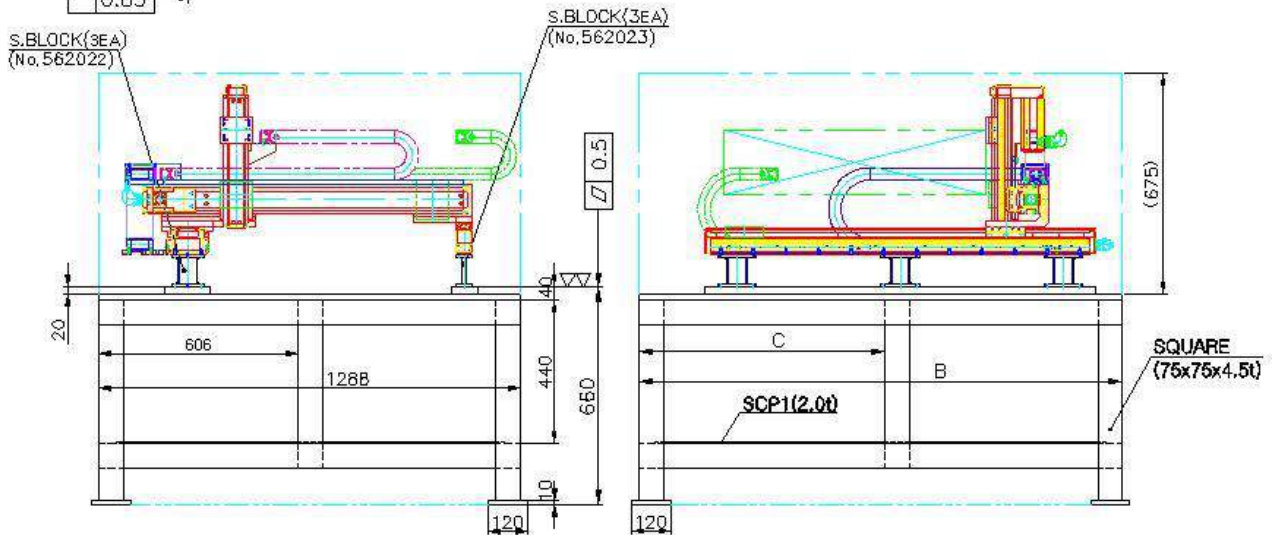


NOTE

1. Chamfers without designation; C1
2. Surface treatment; Black painting
3. Weight; OVER 320kg

MODEL	F9800N	F9804N
A	100	200
B	1380	1480
C	652	702

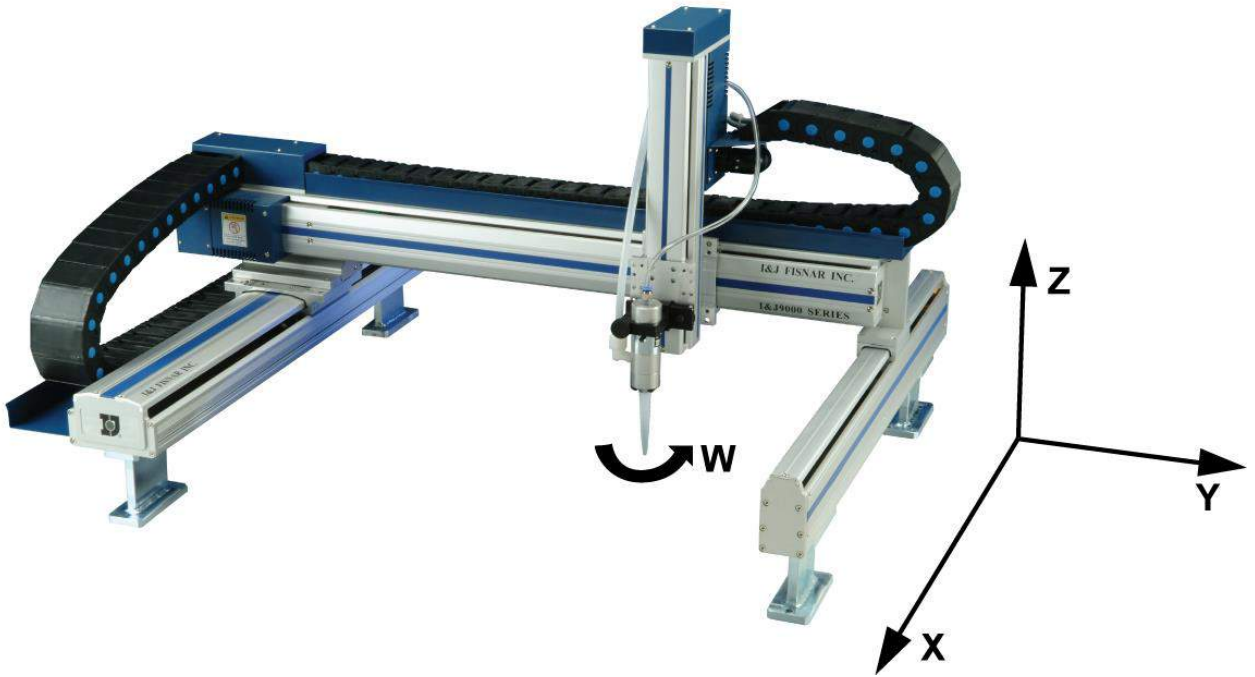
(UNITS: mm)



6. Appendix F: Coordinates (axes) of robots

6.1. Manipulator of F9000N Series

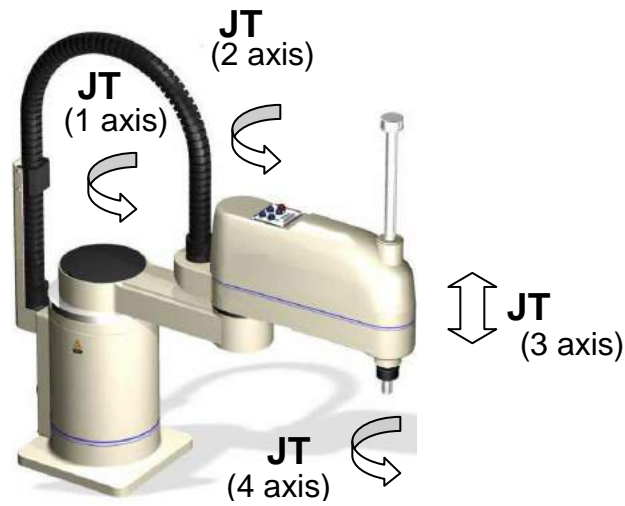
From Zero (origin) point at the Manipulator base, X coordinate is designated far forward and backward directions, Y coordinate for right and left directions, Z coordinate for upward and downward directions and W coordinates for clockwise and counterclockwise rotations.



6.2. Manipulator of F2004N Series

6.2.1. Joint Coordinates

Joint coordinate indicate rotation angle for each axis. Axis 1,2, and 4 coordinates are designated in degrees and axis 3 coordinate is designed in mm. During operation, the rotation speed of each axis is indicated as a percentage from its maximum motor speed.



6.2.2. Cartesian Coordinates

From zero (origin) point at the manipulator base, X coordinate is designated for forward and backward directions, Y for right and left directions, Z for upward and downward directions and W for rotation direction (clockwise and counterclockwise).

