

# **TT-300**

## Operating Manual

### **IAI America, Inc.**

357 Van Ness Way, Suite 20  
Torrance, CA 90501-1483  
310-320-3978 / 310-320-4553 FAX

Publication No. IAI-021-C  
Publication Date: May 1994

---

This publication was written to assist you in better understanding this part of your IA system. If you require further assistance, please contact IA Technical Support. For Central and East Coast Time Zones, please call our Elk Grove Village, IL office at 1-800-944-0333 or FAX 708-290-0351. For Mountain and Pacific Time Zones, please call our Torrance, CA office at 1-800-736-1712 or FAX 310-320-4553; Monday thru Friday from 8:00 AM to 5:00PM.



INTELLIGENT ACTUATOR

---

**IAI America, Inc.**

U.S. Headquarters  
2360 W. 205th Street  
Torrance, CA 90501  
310-320-3978 / 310-320-4553 FAX

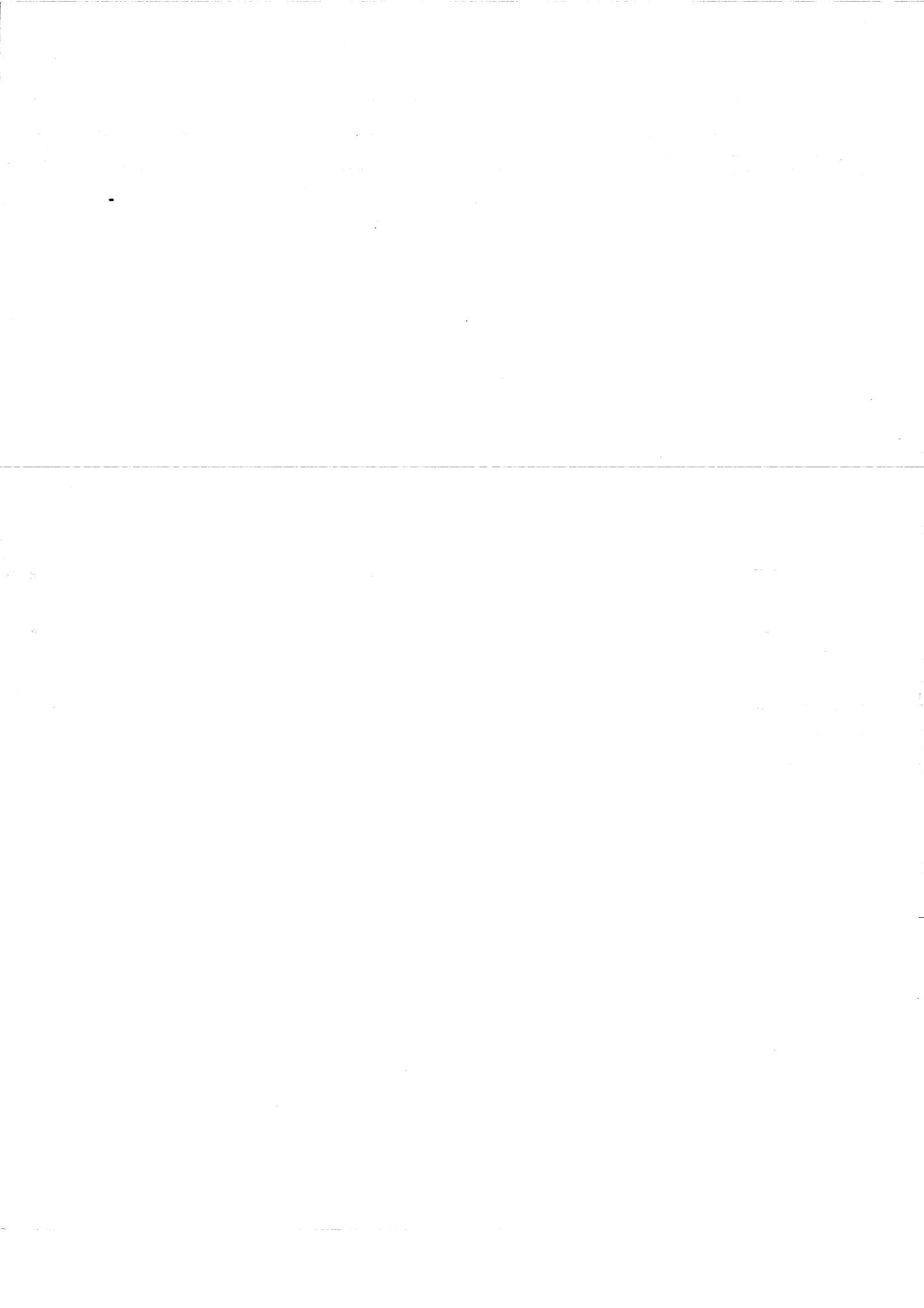
**IAI America, Inc.**

Midwest Regional Office  
1850 Jarvis Avenue  
Elk Grove Village, IL 60007  
708-290-0003 / 708-290-0351 FAX

---

**Disclaimer**

The information and technical data contained herein are subject to change without notice. IAI America, Inc. assumes no responsibility for any errors or omissions regarding the accuracy of the information contained in this publication.



# Table of Contents

---

<b>1.</b>	<b>Introduction.....</b>	<b>1</b>
<b>2.</b>	<b>Summary.....</b>	<b>2</b>
<b>3.</b>	<b>Safety Precautions.....</b>	<b>3</b>
<b>4.</b>	<b>Operation.....</b>	<b>4</b>
4.1	Features and Functions.....	4
4.2	Teaching Pendant (Option).....	5
<b>5.</b>	<b>Specifications.....</b>	<b>30</b>
5.1	Mechanical System.....	30
5.2	Controller System.....	30
5.3	I/O Summary.....	31
5.4	Teaching/RS232C Connector.....	32
5.5	TT-300 Dimention.....	33
<b>6.</b>	<b>Programming.....</b>	<b>34</b>
6.1	Super SEL Controller Language Description.....	34
6.2	Super SEL Language Structure.....	41
6.3	Command.....	46
6.4	Parameter List.....	79
6.5	Error Codes.....	82

---

# 1. Introduction

---

This publication was written to assist you in better understanding the tabletop type of your IA system. If you require further assistance, please contact IA Technical Support at 1-800-736-1712 or Fax 310-320-4553: Monday through Friday from 8:00 AM to 6:00 PM Pacific Time.

## 2. Summary

---

We have developed TT-300, X-Y 2 axes tabletop type, for positioning the end-effector to the designated location (any desirable points) repeatedly with high accuracy. TT-300 is a compact automation apparatus with rectangular coordinate system. The TT-300 has a high-sophisticated built-in controller and DV Servo motor to enhance assembling automation and labor-saving.

### 3. Safety Precautions

---

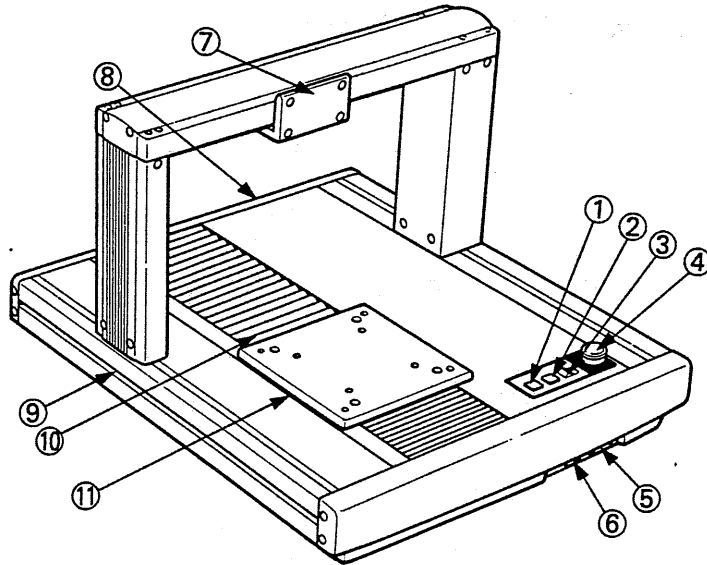
The TT-300 is all in one type and it can be used independently. It is capable of controlling more than one peripheral device simultaneously, using the built-in high sophisticated controller and Input / Output connector attachment. As any system becomes more extensive, the possibility of accidents also increased. Please take sufficient care when operating your system. The IA TT-300 as automated driving part should not be operated beyond its torque and speed limit.

Please follow the following safety precautions when operating your IA System:

- [1] Any operation not specifically addressed in this manual should not be attempted. If you have any questions, please contact your IA sales representative or contact IA Technical Support at: 1-800-736-1712.
- [2] Stand clear of your IA System when operating or preparing to operate. Surround your IA System with safety partitions if there is any possibility that someone may become injured by an operating IA System.
- [3] Before assembling, adjusting, or performing maintenance on your IA System, please unplug the power cable. Please make sure that people around you are aware that the system is not to be powered up or turned on. You may want to place big warning signs where everyone sees, keep the power cable close to the operator, or use a safety plug to ensure that the power cable will not be plugged in inadvertently.
- [4] When more than one person is working on your IA System, use signs or give a word of command to inform everyone of the operating status of the Equipment. Make sure that everyone stands clear prior to operation. Operate your system only after you are sure that everyone knows that you are initiating System start-up and that everyone is clear of the System.
- [5] In situations where the cables must be lengthened, be sure to double check all connections before powering up your IA System.

# 4. Operation

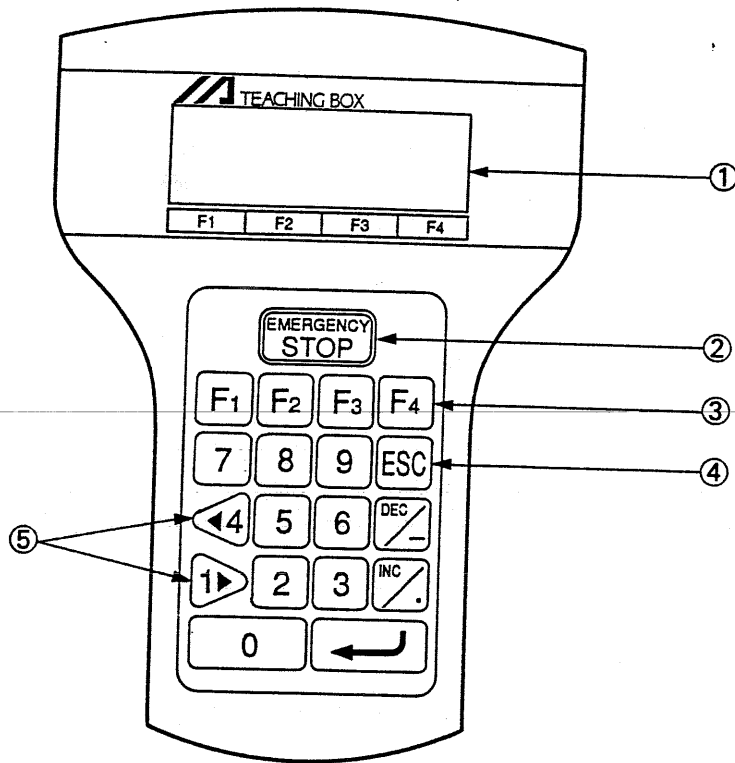
## 4.1 Features and Functions



- [1] **Start**  
Starts the program and operate.
- [2] **Stop (Temporary Stop)**  
When the stop switch is pressed, the program stops. When the stop switch is pressed again the program starts back.  
**Amendment:** The stop switch does not work unless the command "HOLD 5" is included at the beginning of the program. The HOLD command will only stop movements. The hold command does not turn off output ports.
- [3] **Program Transfer Switch**  
Transfers (change-over) up to 32 programs.
- [4] **Emergency Stop**  
When the emergency stop button is pressed, servos will disengage and all programmable outputs will be turned off.
- [5] **Teaching/RS232C**  
Connector for Teaching Pendant and PC.
- [6] **Power ON/OFF**  
Power Switch. When the main power is turned OFF, all the programs and data will be deleted after 14 days (back-up period).
- [7] **Y-Axis Slider**  
Slider where tool(s) are attached to.
- [8] **Input/Output (General I/O : Rear Panel Side)**  
General Input/Output connector that connects peripheral devices.
- [9] **T-Slot**  
Auxiliary tools can be placed using T-slots (on both sides).
- [10] **Bellows**  
Prevents small parts from getting into the system (not waterproof).
- [11] **X-Axis Table**  
Working table. When a larger working platform is used, please make sure that the switches and buttons are not interfered.



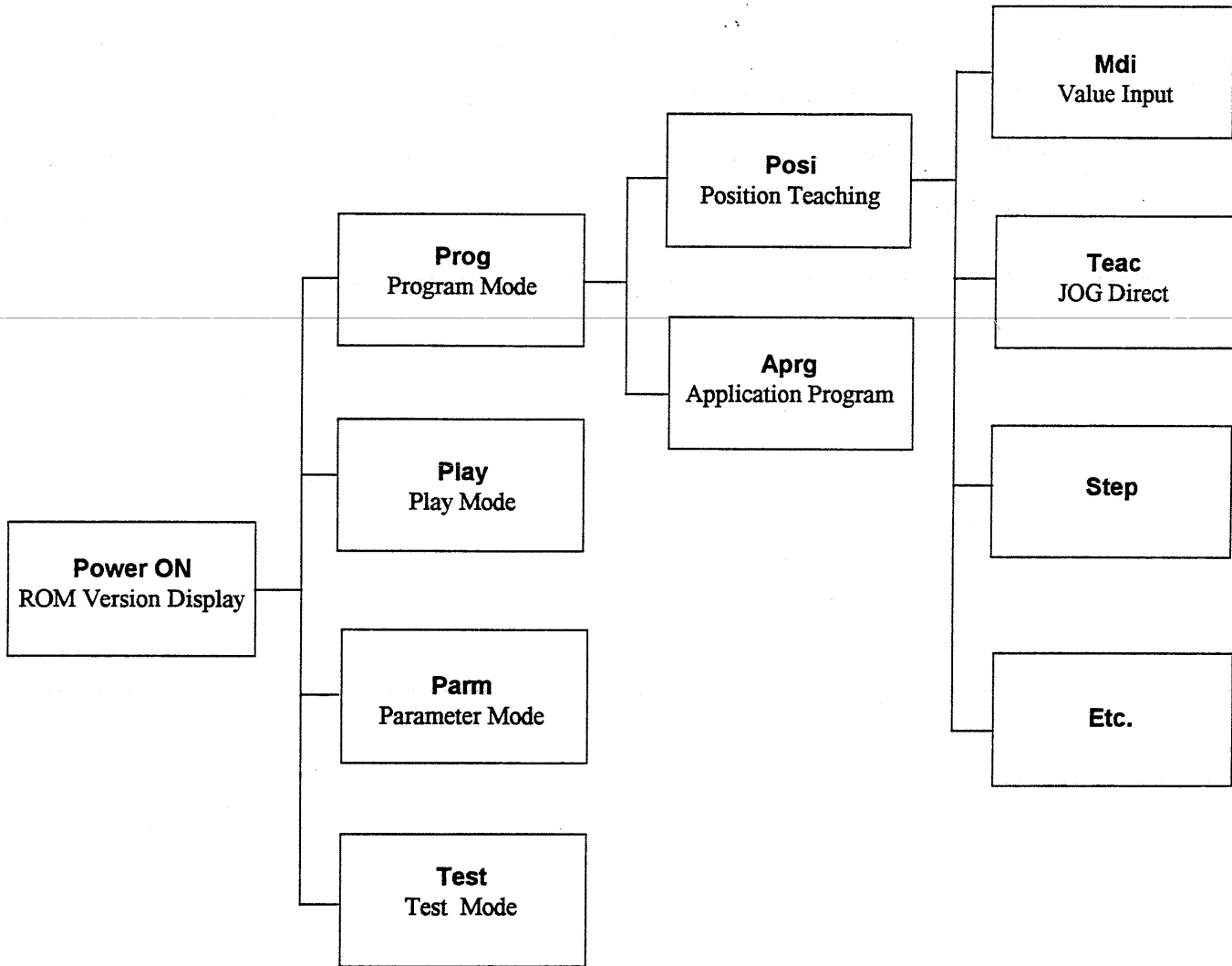
## 4.2 Teaching Pendant (Option)



- [1] **LCD Display**  
4 lines with a 20 character per line capacity display. Shows program and Motion status.
- [2] **Emergency Stop**  
When the emergency stop button is pressed, servos will disengage and all programmable outputs will be turned OFF. To Reset Emergency Stop, Turn OFF the Power Switch, and then turn it back ON.
- [3] **F1, F2, F3, F4 (Function Key)**  
Multi-function keys which correspond with the LCD Display.
- [4] **ESC (Escape)**  
The escape key allows the operator to go backwards in one-step increments to previous displays to make corrections or to switch to different modes.
- [5] ◀ ▶  
Dual function keys for use in data input and axis Jog functions.

## Summary of Teaching Pendant Operation

The tree structure below illustrates the Teaching Pendant mode structure.



Press F1 ~ F4 to move through the branch modes and press ESC to go back to the main branch.

---

## Teaching Pendant Key Functions

Teaching Pendant Key Functions are as follows:

- 01**      **Cursor:** Numbers can be changed when the cursor is positioned underneath.
- "Inc"**    **Increment:** This key increases the Step Number or Point Number.
- "Dec"**    **Decrement:** This key decreases the Step Number or Point Number.
- "Esc"**    **Escape:** This key is used to go back to the previous display.
- ". "**      **Decimal Point Key:** When creating an application program, the display will go to the next command menu.
- " - "**     **Minus Key:** When creating an application program, the display will return to the previous command menu.

Note: If the main power is turned OFF, all programs and data will be stored for a guaranteed period of 14 days.

## Teaching Pendant Operation

### (1) Mode Selection

*(LCD Screen Display)*

IA. Super. SEL TeachV1.00 02/16/94			
Start (Blinking)			
F1	F2	F3	F4

*(Operation)*

#### 1. Initial Display

Press F1 (Start)

IA. Super SEL TeachV1.00 02/16/94 Main V1.00 02/19/94 Start (Blinking)			
F1	F2	F3	F4

#### 2. Controller ROM Version Display

Press F1 (Start)

Mode Select			
Prog	Play	Parm	Test
F1	F2	F3	F4

#### 3. Mode Selection Display

- F1 - Prog (Program) Mode
- F2 - Play (Play/Execution) Mode
- F3 - Parm (Parameter) Mode
- F4 - Test (Test) Mode

**\*Note\*** Pressing **ESC** key will bring the display back to this screen.

(2) Program Mode

(LCD Screen Display)

(Operation)

Prog			
Posi	Aprg		
F1	F2	F3	F4

4. Program Mode (Press F1 at Step 3)

- F1 - (Posi: Position) Position Data Input
- F2 - (Aprg: Application Program) Programming
- Esc - Return to Mode Selection Display (Step 3)

Position Data Input

Posi			
Mdi	Teac	Step	Etc
F1	F2	F3	F4

5. Position (Position Data) Input Display (Step 4. Press F1)

- F1 - Mdi Mode (Position Data Direct Input)
- F2 - Teac (Position Data Teaching) Mode
- F3 - Step (Position Data Step) Mode
- F4 - Posi (Position Data) Display 2
- Esc - Return to Program Mode (4.)

Mdi - <u>1</u> No 1[1] - 2			
1.234		12.345	
Inc	Dec	Clr	Del
F1	F2	F3	F4

6. Mdi Mode

6-1. Position Number Input Mode (Step 5. Press F1)

Input 2-digit numbers.

- F1 - (Inc: Increment) Increments the Position Number displayed by 1.
- F2 - (Dec: Decrement) Decrement the Position Number displayed by 1.
- F3 - (Clr: Clear) Clears all data (return to 0) for re-entry.
- F4 - (Del: Delete) Deletes input data.
- CR (Carriage Return) key - Position Data Input
- Esc - Returns to Position (Position Data) Input Display (Step 5.)

**\*Note\*** Mdi - 1 No 1 [1] - 2  
 Cursor Position      Axis #      Axis Name      Total Axis Number

Posi			
	Copy	Clr	Etc
F1	F2	F3	F4

6-2 Position (Position Data) Input Display 2 (Go back to the display in Step 5. Press F4)

- F2 - (Copy) Position Data Copy
- F3 - (Clr: Clear) Position Data Clear (Deletion)
- F4 - Position (Position Data) Input Display (5.)
- Esc - Return to Program Mode Display (Step 4.)

Mdi - 1 No 1 [1] - 2			
	1.23 <u>4</u>		12.345
Wrt	Can	Clr	Etc
F1	F2	F3	F4

7. Position Data Input Mode (Step 6-1, CR Carriage Return)

Direct Input (5-digit integers before decimal point and 3-digits after the decimal point.)

- F1 - (Wrt: Write) Writing in (Saving) Position Data
- F2 - (Can: Cancel) displays xxxx.xxx.
- F3 - (Clr: Clear) displays 0.000 for data re-entry.
- F4 - Position Data Input additional menu.

Mdi - 1 No 1[1] - 2			
	1.23 <u>4</u>		12.345
Axis+	Axis-	Vel	Etc
F1	F2	F3	F4

8. Position Data Input Sub Menu (Step 7. F4)

Direct Input (5-digit integers before decimal point and 3-digits after the decimal point)

- F1 - (Axis+) Increments Axis Number by 1.
- F2 - (Axis-) Decrement Axis Number by 1.
- F3 - (Vel: Velocity) Set Velocity and Acceleration for the Position Number.
- Etc - Returns to Position Data Input Mode (Step 7.)

Mdi - 1 No 2[2] - 2			
	1.234		12.34 <u>5</u>
Axis+	Axis-	Vel	Etc
F1	F2	F3	F4

8-1 Increments Axis Number by 1. (Step 8, F1)

Mdi - 1 No 2[2] - 2			
	2345.678		1.234
Axis+	Axis-	Vel	Etc
F1	F2	F3	F4

8-2. Decrements Axis Number by 1. (Step 8, F2)

Mdi - 1 No 2[2] - 2			
500.000		600.000	
Wrt	Can	Clr	Run
F1	F2	F3	F4

8-3 After input 2-axis, save data by pressing the F1 key (Wrt).

If "Run" is blinking, data will not be saved because it means a program is being executed at that moment. In this case, press F4, and then switch to Test Mode or Play Mode and monitor the program being executed.

If "Wrt" (F1) is pressed while programming, "Run" (F4) starts to blink to alert the operator.

F1 - Wrt: Write      F2 - Can: Cancel

F3 - Clr: Clear      F4 - Run : Program Execution

Mdi - 1 No. 1[1] - 2			
1.234		12.345	
Vel [300]	Axis	Acc[0.99]	
Wrt		Clr	
F1	F2	F3	F4

9. Position Data Velocity and Acceleration Setting (Step 8. F3)

(Velocity: 3-digits, Acceleration 9.99, 2-digits after decimal point.)

Step 1 No 1[1] - 2			
1.234		12.345	
Inc	Dec	Go	JVel
F1	F2	F3	F4

10. Position Step Mode (Go back to the display in Step Number 5, then press F3.)

F1 - Position No. +1

F2 - Position No. -1

F3 - Move to Designated Position

F4 - Designate Velocity

Clear All Position			
Clr			
F1	F2	F3	F4

11. Position Clear Mode (Go back to the display in the Step 6-2. Then, press F3.)

F3 - Clears all Position Numbers.

F1 - Execute

Teac- 1 No 1[1] - 2			
1.234		12.345	
Inc	Dec	Go	JVel
F1	F2	F3	F4

12. Teaching Mode (Go back to the display in Step 5, then press F2.)

Position Number Input Mode (2-digit Number Input)

F1 - Increments Position Number by 1.

F2 - Decrements Position Number by 1.

F3 - Moves from current position to the position designated in the display.

F4 - Sets Velocity,

CR - Performs teaching for each axis.

13. Teaching Selection Mode (12. CR Carriage Return)

Select Jog or SvOf (Servo OFF, Manual)

The Cursor is located under Axis Number 1. Press Etc to switch to another axis.

F1 - Select Jog Velocity

F2 - Set for Jog Movement

F3 - Set for Manual (direct) Teaching.

F4 - Velocity Setting and Axis Changing Mode

13-1. Jog Mode (Step 13. F2)

Turns ON all Servos and also indicates the present position. When Wrt is selected, the present position will be saved.

Teac- 1 No 1[1] - 2			
1.23 <u>4</u>		12.345	
JVel	Jog	SvOf	Etc
F1	F2	F3	F4

Jog- 1 No 1[1] - 2			
1111.12 <u>3</u>		12.345	
Wrt	JVel	SvOf	Etc
F1	F2	F3	F4

13.2 Jog Mode (Step 13-1, F4)  
Velocity, Axis Selection

When selecting Axes to jog:

F1 - Axis+ moves cursor forward 1 Axis.

F2 - Axis- moves cursor backward 1 Axis.

13.3 Teaching Velocity Set, Axis Change Mode (Step 13. F4)

F1 - Teaching Axis No. + 1

F2 - Teaching Axis No. - 1

F3 - Set Velocity and Acceleration for Position Data.

F4 - Return to the display before selection.

Jog- 1 No 1[1] - 2			
1111.12 <u>3</u>		12.345	
Axis +	Axis -	Vel	Etc
F1	F2	F3	F4

Teac- 1 No 1[1] - 2			
1.23 <u>4</u>		12.345	
Axis+	Axis-	Vel	Etc
F1	F2	F3	F4



SvOf - 1 No 1[1] - 2			
: 1111.123		12.345	
Wrt	JVel	Jog	Etc
F1	F2	F3	F4

14. SvOf (Servo OFF, Manual • Direct Teaching) Mode (Go back to the display in Step 13, then press F3)

All connected axes Servo OFF  
 Moves axis position manually (Direct Teaching).  
 Wrt: Saves data  
 CR: Changes axis number.

SvOf - 1 No 1[1] - 2			
1111.123		12.345	
Axis+	Axis-	Vel	Etc
F1	F2	F3	F4

14 -1. SvOF Mode (Step 14. F4)

Velocity Setting, Axis No. Selection  
 (Use to set Point Position Velocity or to change Axis No. manually)

SvOf - 1 No 1[1] - 2			
1111.123		12.345	
Wrt	Vel[ 30]	Acc[0.30]	
	Axis	Clr	
F1	F2	F3	F4

14-2. SvOf Mode (Step 14-1. F3)

Velocity Setting Selection  
 Input Velocity and Accerelation, then press Return (ENTER) key  
 F2 - Return to SvOf Mode  
 F3 - Clr (Clear) clears input value, for re-entry.

Application Programming

Aprg			
Edit		Copy	
F1	F2	F3	F4

Edit 1- 1 [ 50]			
MOVP 1			
699 A N499			
Inc	Dec	Clr	Del
F1	F2	F3	F4

15. Programming Edit Display (Go back to the display in Step 4, then press F2)

F1 - (Edit Mode) Perform Programming, Editing, Addition, Insertion, and Deletion.

F2 - (Copy Mode) Copies or overwrites programs.

16. Programming Edit Mode (15. F1)

16-1. Program Number Input Mode

F1 - Increments Program Number by 1.

F2 - Decrements Program Number by 1.

F3 - Clears input value for re-entry.

F4 - Deletes Program Number.

CR - Switches to Step Number Input Mode when the program number input is within the appropriate range.

When a program number exists, the step number 1 will be displayed. When a program number does not exist, a space will be displayed.

( Edit 1- 1 [ 50] )  
 |            |            |  
 Prog. No. Step No. Step Number

( MOVP 1 )  
 |            |  
 Command Operation 1

(            699 A N499 )  
 |            |            |  
 Operation 2 Result Continuous Condition

Edit 1- <u>1</u> [ 50]			
MOVP 1			
699 A N499			
Inc	Dec	Clr	etc
F1	F2	F3	F4

16-2. Step Number Input Mode (Step 16-1. CR: Carriage Return)

- F1 - (Inc) Increments Step Number by 1.
- F2 - (Dec) Decrements Step Number by 1.
- F3 - (Clr) Clears data for re-entry
- F4 - Additional Mode (Insertion mode, deletion mode etc.)

Edit 1- <u>1</u> [ 50]			
MOVP 1			
699 A N499			
Inc	Del		etc
F1	F2	F3	F4

16-2-1. Step Number Input Mode 1 (Step 16-2. F4)

- F1 - (Insertion) Add Steps.
- F2 - (Delete) Delete Steps.

Edit 1- <u>1</u> [ 50]			
MOVP 1			
699 A N499			
ABPG	ACC	ADD	AND
F1	F2	F3	F4

16-3. Edit Command Input Mode (Step 16-2. CR: Carriage Return)

4 commands will be displayed from F1 to F4.

. (decimal point) - By pressing the decimal point, commands will be displayed in alphabetical order.

- (minus sign) - By pressing the minus sign, commands will go backwards in reverse alphabetical order.

Select a command by pressing F1 ~ F4, . (decimal point), or - (minus) sign.

Press CR (Return key). Operation 1 Input Mode (Step 16-4.) will be displayed.

Edit 1- <u>1</u> [ 50]			
MOVP <u>1</u>			
699 A N499			
*	BS	Clr	
F1	F2	F3	F4

16-4. Operation 1 Input Mode (Step 16-3. CR: Return key)

Input Operation 1 of the selected command.

F1 - (\*) Designate variables indirectly.

F2 - (BS: Back Space) Clears last input value. Cursor will move backwards.

F3 - (Clr: Clear) Clears all the values input for new data entry.

CR (Return key) - Input values will be saved, then Operation 2 Input Mode will be displayed.

ESC - Command input mode

Edit 1- 1 [ 50]			
MOVP 1			
		BS	699 A N499 Clr
F1	F2	F3	F4

16-5. Operation 2 Input Mode (Step 16-4, CR: Return key)

Same as 16-4

16-6. Result Input Mode (16-4. CR: Return key)

Input result, output and flag.

F2 - Clears the last entry by BS (Back Space).

F3 - Clears all data

CR - Continuous Condition Input Mode

Edit 1- 1 [ 50]			
MOVP 1			
And	Or	Clr	699 A N499 Not
F1	F2	F3	F4

16-7. Continuous Condition Input Mode (Step 16-6, CR: Return key)

F1 - Select And

F2 - Select Or

F3 - Clears data entry.

F4 - Select Not

CR - Confirmation Mode

Edit 1- 1 [ 50]			
MOVP 1			
		Clr	699 A N499 Wrt
F1	F2	F3	F4

16-8. Confirmation Mode (Step 16-7. CR: Return key)

F3 - Re-input from command

F4 - Saves (Wrt) current Step No. After saving, Step Number will proceed to the next Step Number. (In the case of Insertion, Step Number will remain the same.)

Prog Copy			
From		1	
To		Clr	
F1	F2	F3	F4

17. Copy Mode (Go back to the display in Step No. 15, then press F2.)

17-1. Copies program. (Step 15. F2)

Input program number to be copied after "From", then input a new program number after "To".

F3 - Clears data for data entry.

Prog Copy			
From		1	[ 50]
To		1	[ 50]
Copy	OWrt		
F1	F2	F3	F4

17-2. Copy, Overwrite Selection Mode (17-1. CR: return key)

F1 - Copies the program

F2 - Overwrites the program.

(3) Parameter Mode

Para			
Axis	Sys		
F1	F2	F3	F4

18. Parameter Mode (Step 3, F3)

- F1 - Parameter Mode for each axis
- F2 - System Parameter Mode

Para Axis			
Srvo	Home	Motr	Name
F1	F2	F3	F4

18-1. Parameter Mode for each axis (Step 18. F1)

- F1 - Servo Parameter Mode for each axis
- F2 - Home Parameter Mode for each axis  
Axis Activation Mode
- F3 - Motor Parameter Mode for each axis
- F4 - Axis Name Parameter Mode for each axis

Para System			
Prog	Pos	Srvo	Etc
F1	F2	F3	F4

18-2. System Parameter Mode (Step 18. F2)

- F1 - System Program Parameter Mode
- F2 - System Point Parameter Mode
- F3 - System Servo Parameter Mode
- F4 - Additional selection mode

Para System			
Sio	Cir	Etc	
F1	F2	F3	F4

18-3. System parameter additional selections (Step 18-2. F4)

- F1 - (Sio: Serial I/O RS232C) System Sio Parameter Mode
- F2 - (Cir: Circle) System Arc/Circular Parameter Mode
- F4 - (Etc) Returns to Step 18-2.

Para Axis 1[1] Srvo			
1. Numerator			
Inc [ < = ]		1	
Axis*	Dec	Clr	Wrt
F1	F2	F3	F4

19-1. Servo Parameter Mode by axis (18-1. F1)

- F1 - Goes to next axis no. (+1) Axis name will also be changed.
- F2 - Goes backward -1
- F3 - Clears data for data entry
- F4 - Direct number input
- CR - (Inc[<=]) adds +1

Para Axis 2[2] Srvo			
1. Numerator			
Inc [=<]		<u>1</u>	
Axis+	Dec	Clr	Wrt
F1	F2	F3	F4

19-1-1. Press F1 to proceed to one axis number.

Para Axis 1[1] Srvo			
9. Soft Limit (-)			
Inc [=<]		<u>0</u>	
Axis+	Dec	Clr	Wrt
F1	F2	F3	F4

19-1-2. Press F2 to go back one axis number.

Para Axis 1[1] Home			
1. Home Dir			
Inc[=<]		<u>1</u>	
Axis+	Dec	Clr	Wrt
F1	F2	F3	F4

19-2. Home Parameter Mode for each axis, Axis Activation Mode (Step 18-1, F2)

Para Axis 1[1] Home			
3. Home Sequence			
Inc[=<]	<u>1</u>		
Axis+	Dec	Clr	Wrt
F1	F2	F3	F4

19-2-1. Home Parameter Mode for each axis, Axis Activation Mode (Step 19-2, Press the RETURN key twice)

By inputting 0, the axis will be inactivated. Homing or Jog will not be performed. (Position indicator will display Δ)

By inputting a number (1 ~ 2), order of Homing of all activated axes can be designated.

Para Axis 1[1] Motr			
1. Motor RPM Max			
Inc[=<]		<u>4000</u>	
Axis+	Dec	Clr	Wrt
F1	F2	F3	F4

19-3. Motor Parameter Mode by axis (Step 18-1. F3)

Para Axis 1[1] Name			
Axis+	Axis -	Name+	Name-
F1	F2	F3	F4

19-4. Parameter Name Mode by axis (Step 18-1. F4 Key)

- F1 - Increments axis number by 1.
- F2 - Decrements axis number by 1.
- F3 - Increments axis name by 1. (range: 1 ~ 2)
- F4 - Decrements axis name by 1. (range: 1 ~ 2)

Para Axis 1[2] Name			
Wrt		Name+	Name-
F1	F2	F3	F4

19-4-1. Axis Name Change (Step 19-4. F3)

- F1 - (Wrt) Change axis name.

Para System			
Prog	Pos	Srvo	Etc
F1	F2	F3	F4

20. System Parameter Mode (Step 18. F2)

Para System		Prog	
1.Auto Start PRG			
Inc[<=]		<u>0</u>	
Inc	Dec	Clr	Wrt
F1	F2	F3	F4

20-1. System Program Parameter Mode (Step 20. F1)

Press the RETURN key to increment the item. (Inc[<=])

Para System		Pos	
1.Point Size			
Inc[<=]		<u>1000</u>	
Inc	Dec	Clr	Wrt
F1	F2	F3	F4

20-2. System Point Parameter Mode (Step 20. F2)

Para System		Srvo	
1. Axis Size			
Inc [<=]		<u>2</u>	
Inc	Dec	Clr	Wrt
F1	F2	F3	F4

20-3. System Servo Parameter Mode (Step 20. F3)

20-4. System Sio Parameter Mode (18-3. F1 Key))

Para System		Sio	
1. Terminal ID			
Inc[<=]		99	
Inc	Dec	Clr	Wrt
F1	F2	F3	F4

20-5. System Circle, Arc, Parameter Mode (Step 18-3. F2)

Para System		Cir	
1. Circle Angle			
Inc[<=]		15.9	
Inc	Dec	Clr	Wrt
F1	F2	F3	F4



(4) Test Mode

Test			
	In	Out	Ver
F1	F2	F3	F4

21. Test Mode (Step3. F4)

F1 - Press F1 and "1" key simultaneously. RamCL (Ram Clear) Mode will appear on the display.

F2 - (In) Input Port Display.

F3 - (Out) Output Port output management.

F4 - (Ver: Version) Displays current version of Servo, Main and Teac (teaching Pendant).

Test			
RamCL	In	Out	Ver
F1	F2	F3	F4

21-1. While holding down both the F1 and "1" keys, the display will remain like the one on the left. When the keys are released, Memory Clear Mode (RamCL) will appear.

Test 0123456789 (In)			
000-> 1000000001 <-009			
010-> 0000001100 <-019			
Inc	Dec		
F1	F2	F3	F4

21-2. Test Input Port Display (Step 21. F2)

Displays input port status.

F1 - Port displayed. +10

F2 - Port displayed. -10

Test 0123456789 (Out)			
300->0000000000<-309			
310->0000001100<-319			
Inc	Dec	Clr	0/1
F1	F2	F3	F4

21-3. Test output Port Display (Step 21. F3)

F1 - Output Port +10

F2 - Output Port -10

F3 - 0 Output of all the displayed Ports

F4 - Output 0 -->1, 1-->0

. (decimal point), CR (RETURN) key - cursor moves to the right.

- (minus key) - cursor moves to the left.

"0" - cursor position port 0 Output

"1" - cursor position port 1 Output.

Test 0123456789 (Out)			
310->0000000000<-319			
320->0000001100<-329			
Inc	Dec	Clr	0/1
F1	F2	F3	F4

21-3-1. F1 (Output Port +10)

Test Version		1[1]	
MotorV1.00		01/26/94	
Main V1.00		2/19/94	
Axis+	Axis-	Main	Teac
F1	F2	F3	F4

21-4. Test Version Mode (Step 21. F4)

RamCL			
All	Para	Prog	Pos
F1	F2	F3	F4

21-5. Memory Clear (RamCL) Mode (Step 21. F1 + I key)

- F1 - Clears system parameter, program and, position area
- F2 - Clears system parameter
- F3 - Clears system program area
- F4 - Clears system position area

RamCL All			
CLROK?			
F1	F2	F3	F4

21-5-1. F1(All) is selected.

Press F1 (CLROK?: Clear OK?)  
 Note: This is the same as RESET.

RamCL Para			
CLROK?			
F1	F2	F3	F4

21-5-2. F2 (Para) is selected.

Press F1.  
 Note: This is the same as RESET.

RamCL Prog			
CLROK?			
F1	F2	F3	F4

21-5-3. F3 (Prog) is selected.

Press F1

RamCL Pos			
CLROK?			
F1	F2	F3	F4

21-5-4. F4 (Pos) is selected.

Press F1

(5) Play Mode

Play	1-1	[ 50]	
MOVP		1	
		699 A N499	
Inc	Dec	Clr	
F1	F2	F3	F4

22. Play (Play) Mode (Step 3, F2)

22-1. Play Program Input Mode

Input Program number to execute (or stop) Program status can be seen by pressing START then PROG.

Play	1-1	[ 50]	
MOVP		1	
		699 A N499	
Show	Go	Stat	Stop
F1	F2	F3	F4

22-2. Program Execution/Stop Selection (Step 22-1. CR: return key)

F1 - (Show) Monitoring (when the designated program is executing.)

F2 - (Go) Program execution

F3 - (Status) show status (after checking the status, select execution or stop)

F4 - (Stop)

Play	1-1	[ 50]	
MOVP		1	
		699 A N499	
Posi	Play	Stat	Etc
F1	F2	F3	F4

22-3. Execution Status Selection (Step 22-2. F2) (Step 22-2. F4)

F1 - Axis status display

F2 - Other program execution, or stops currently executing program.

F3 - Shows program status

F4 - Input port, Output port, Flags

Play 1-	No 1 [1]-2		
1111.123		2222.234	
Axis+	Axis-		
F1	F2	F3	F4

22-4. Axis Status Display Mode (Step 22-3. F1)

F1 - Checks other axis status (+)

F2 - Checks other axis status (-)

( Play 1- No 1 [2]-2)  
|  
Starting axis

Play 1-1 [ 50]			
MOVP 1			
699 A N499			
Inc	Dec	Clr	
F1	F2	F3	F4

22-5. Execution, Stop Mode (Step 22-3, F2)

Input program number to be executed or stopped.  
Returns to Step 22-3.

Play 1 [Run 1- 0]			
Inc	Dec		
F1	F2	F3	F4

22-6. System movement and program number display (Step 22-3, F3)

Play 1- 1 [ 50]			
MOVP 1			
699 A N499			
In	Out	Flag	Etc
F1	F2	F3	F4

22-7. I/O Port, Flag Status Selection Mode (Step 22-3, F4)

F1 - Input port change display, select  
F2 - Output port change display, select  
F3 - Flag change display, select  
F4 - Execute, status selection screen (Step 22-3)

Play 0123456789 (In)			
000 ->1000000001 <-009			
010->1000001100<-019			
Inc	Dec		
F1	F2	F3	F4

22-8. Play Input Port display (Step 22-7, F1)

F1 - Input Port +10  
F2 - Input Port -10

Play 0123456789 (Out)			
300 ->1000000001 <-309			
310->0000001100<--319			
Inc	Dec		
F1	F2	F3	F4

22-9. Play Output Port display (Step 22-7, F2)

F1 - Output Port +10  
F2 - Output Port -10

Play 0123456789 (Fig)			
600->1000000001<-609			
610->0000001100<-619			
Inc	Dec		
F1	F2	F3	F4

22-10. Play Flag display (Step 22-7, F3)

F1 - Flag +10  
F2 - Flag -10

PlayHLT [Run 1-1]			
1-2			
			No
			Stop
F1	F2	F3	F4

22-11. Escape Play Mode by pressing the ESC key (Step 22-3. ESC.) If a program is being executed, it will change to the stop selection mode. (PlayHLT: Play Halt)

F3 - (No) Returns to Execution/Stop Designation Mode (Step 22-5) (This does not stop the program.)

F4 - (Stop) Program Stop Selection Mode

PlayHLT [Run 1-1]			
1-2			
Inc	Dec	All	Stop
F1	F2	F3	F4

22-12. Program Stop Selection Mode (Step 22-11. F4)

F1 - Select program to be stopped. (+1)

F2 - Select program to be stopped (-1)

F3 - Select F3 to stop all the programs. (Display will proceed to Stop Program Confirmation Mode)

F4 - Select F4 to stop the program currently displayed on screen. (Display will proceed to Stop Program Confirmation Mode.)

ESC key - Returns to Execution/Stop Designation Mode (Step 22-5) (This does not stop any program.)

PlayHLT [Run 1- 0]			
hit [ ESC]			
F1	F2	F3	F4

22-13. Stop-Program Confirmation Mode

F1 - (hit) Execute stop of the selected program

ESC - Returns to Execution/Stop Mode (This does not stop any program.)

(6) Simple Application Program Example

2-Axes Actuators run between point 1 and point 2, back and forth.

• Application Program

Command	Operand 1	Comment
HOME	11	X and Y Axis Homing (Servo ON)
VEL	100	Velocity Setting 100mm/sec
ACC	0.3	Acceleration Setting 0.3G
TAG	1	Assigns the target for GOTO Command
MOVL	1	Moves to position number 1
MOVL	2	Moves to position number 2
GOTO	1	Jumps to TAG1

• Position Data

Position Number	X Axis	Y Axis
1	100.000	100.000
2	10.000	10.000

(LCD Display)

(Operation)

IA Super.SEL TeachV1.00      02/16/94  Start (blinking)			
F1	F2	F3	F4

1. Initial Display

Press F1 to start.

IA Super.SEL TeachV1.00      02/16/94 Main V1.00      02/19/94 Start (blinking)			
F1	F2	F3	F4

2. Controller ROM Version Display

Press F1 to start.

Mode Select			
Prog	Play	Parm	Test
F1	F2	F3	F4

3. Mode Selection Display

Press F1 to select Prog Mode.

Prog			
-			
Posi		Aprg	
F1	F2	F3	F4

4. Program Mode Display

Press F2 to select Aprg (Application Program).

Aprg			
Edit		Copy	
F1	F2	F3	F4

5. Editing, Programming Display

Press F1 to select Edit Mode to bring up program, modify, add, insert, or erase.

Edit 1- 0 [ 0]			
Inc	Dec	Clr	Dec
F1	F2	F3	F4

6. Edit Mode

Press 1 (Program number 1), then press return key. When the program number selected exists, the display shows the first step number. If the program number selected does not exist, a space will appear.

Edit 1- 1 [ 0]			
Inc	Dec	Clr	etc
F1	F2	F3	F4

7. Press return key.

Edit 1- 1 [ 0]			
ABPG	ACC	ADD	AND
F1	F2	F3	F4

8. Press "." (decimal point) repeatedly until HOME appears.

Edit 1- 1 [ 0]			
HOME	IN	INB	LET
F1	F2	F3	F4

9. Press F1 to select HOME.

Edit 1- 1 [ 0]			
HOME			
HOME	IN	INB	LET
F1	F2	F3	F4

10. Press RETURN key.

Edit 1- 1 [ 0]			
HOME _			
*	BS	Clr	
F1	F2	F3	F4

11. Press "1" two times to home X and Y axes simultaneously.

Edit 1- 1 [ 0]			
HOME 11			
*	BS	Clr	
F1	F2	F3	F4

12. Press return key 3 times.

Edit 1- 1 [ 0]			
HOME 11			
		Clr	Wrt
F1	F2	F3	F4

13. Press F4 (Wrt: Writing) to input a command.

Edit 1- 2 [ 1]			
-			
HOME	IN	INB	LET
F1	F2	F3	F4

14. Press "." (decimal point) repeatedly until VEL appears on display.

Repeat the same process after step 9 to input commands.



When programming is completed, press ESC key repeatedly until Mode Selection Display appears. Next, position data will be input.

Mode Select			
Prog	Play	Parm	Test
F1	F2	F3	F4

15. Mode Selection Display

Press F1 to select Prog (Program) Mode

Prog			
Posi	Aprg		
F1	F2	F3	F4

16. Program Mode

Press F1 to select Position data.

Posi			
Mdi	Teac	Step	Etc
F1	F2	F3	F4

17. Position Data Editing (Go back to the display in Step 4, then press F1).

Press F1 to select Mdi Mode (position data direct input).

Mdi - <u>1</u> No 1[1]- 2			
Inc	Dec	Clr	Del
F1	F2	F3	F4

18. Mdi Mode

Press 1 for position data number 1, then press the RETURN key.

Mdi - <u>1</u> No 1[1]- 2			
xxx.xxx		xxx.xxx	
Wrt	Can	Clr	Etc
F1	F2	F3	F4

19. Press 100, then press RETURN key for X axis position data. Press 100, then press RETURN for Y axis position data.

Press F1 (Wrt: Writing) to save position data number 1 (100,100).

\* Repeat the same process to input position data number 2.

# 5. Specifications

## 5.1 Mechanical System

		X	Y
Stroke	mm	300	
Rated Velocity	mm/sec	750	
Rated Thrust	N(kgf)	29(3)	
Repeatability	mm	±0.08	
Body Weight	kg	32(Total Body Weight including controller)	
Base, Side Cover, Base Frame		Extruded Aluminum	
Table, Slider		Extruded Aluminum White Almite	
Front Cover, Rear Cover		ABS Resin	
Bellows		Chloroprene Rubber	
Max. Velocity	mm/sec	1000	
Max. Thrust (*1)	N(kgf)	68(7)	
Table, Slider Payload (*2)	kg	5	
X,Y Axis Thrust	N(kgf)	118(12)	

\*1 At a speed of 10mm/sec for 10 seconds.

\*2 Load uniformly distributed on the table. Conditions: 0.5G acceleration, 750mm/sec speed

## 5.2 Controller System

Power Voltage	AC100V±10%
Power Frequency	50/60Hz
Power Capacity	Approximately 600W
Ambient Temperature, Humidity	5~40°C (No Condensation)
Ambient Environment	No Corrosive Gas, Minimal Dust
Isolation Resistance	500V 10M Ω or more
Noise Immunity	1500V μ sec pulse by Noise Simulator
Protective Function	Driver Temperature Check Overload Check Soft Limit Check
Motor Capacity	DC Servo Motor 35W
Memory Capacity	Approximately 1000 Steps, 500 Positions
Number of Program	32 Programs (4 Multi Task Programs)
Memory Device	CMOS RAM Battery Back-Up 128KBYTE
Input/Output (DC24V)	Dedicated Input, Emergency Stop General Input 8 Points General Output 8 Points DC24V 0.5A Built-In Output Max Loading Current 100mA/1Point (20mA/1Point recommended) (Applied to TD62083AF)
Data Input Method	Teaching Pendant or RS232C Communications
Communications	EIA RS 232C Asynchronous

### 5.3 I/O Summary

Port No.	Function	Pin No.	Cable
-	Built-In Power +24V Output	1	Brown
000	(Start Switch on Topface Panel)	-	-
005	(Use for Stop Switch on Topface Panel: Program HOLD)	-	-
016	General Input	2	Red
017	General Input	3	Orange
018	General Input	4	Yellow
019	General Input	5	Green
020	General Input	6	Blue
021	General Input	7	Purple
022	General Input	8	Gray
023	General Input	9	White
002	Emergency Stop Input (Point b Input)	10	Black
316	General Output	16	Blue
317	General Output	17	Purple
318	General Output	18	Gray
319	General Output	19	White
320	General Output	20	Black
321	General Output	21	Brown
322	General Output	22	Red
323	General Output	23	Orange
-	Built-In Power 0V	24	Yellow

\* To release Emergency Stop, connect PinNo.10 and Pin No.24 to I/O Connector.  
 (A demo-program is in Program No.32)

## 5.4 Teaching/RS232C Connector

Pin No.	Signal
1	FG
2	TXD
3	RXD
4	(RTS)
5	(CTS)
6	DSR
7	SG
8	NC
9	NC
10	NC
11	NC
12	NC
13	NC

Pin No.	Signal
14	NC
15	NC
16	NC
17	NC
18	+6.2V Output
19	NC
20	DTR
21	NC
22	NC
23	Emergency Stop Switch (EMG.SW)
24	NC
25	0V (+6.2V)

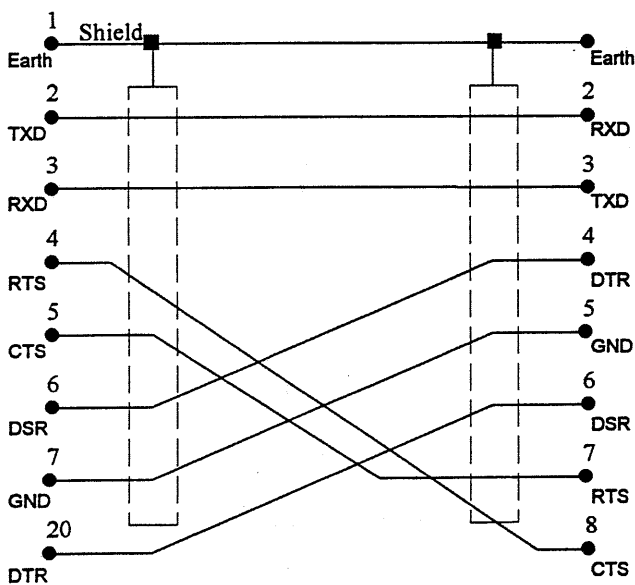
- Pin numbers 18, 21, 23, 25 are for use with the Teaching Pendant signal. Do not connect these pins.
- Pin numbers 4 and 5 are short-circuited.

### RS232C Cable

Use RS232C Cable Pin Configuration (Between Controller and Computer Serial Port)

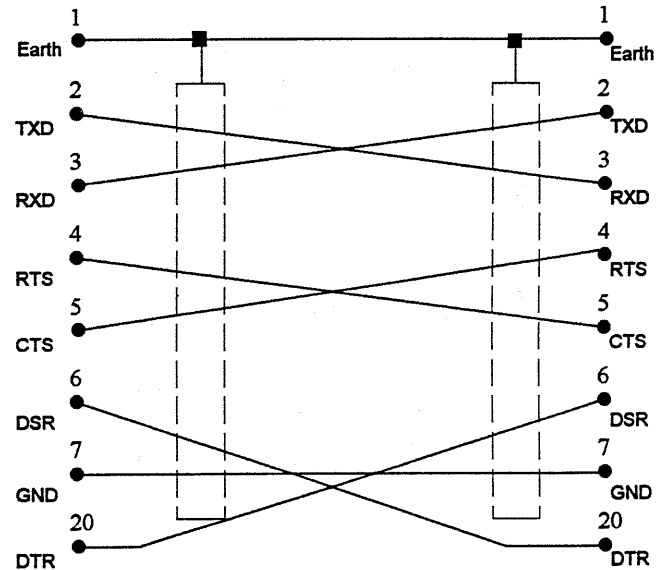
RS232C Adapter  
(25 Pin Male)

IBM PC  
(9 Pin Female)

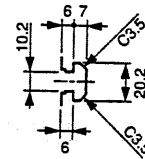
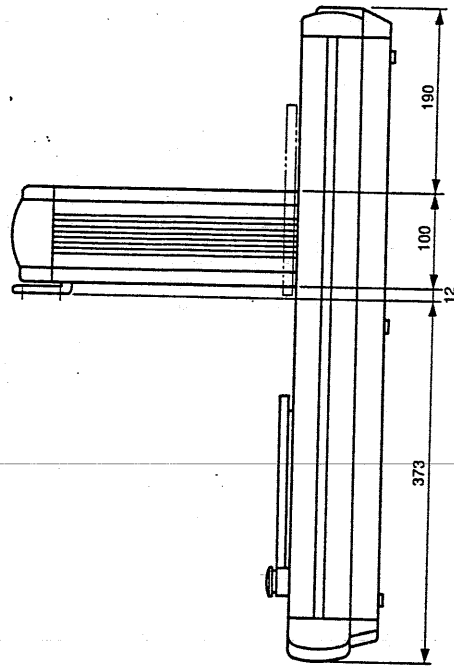
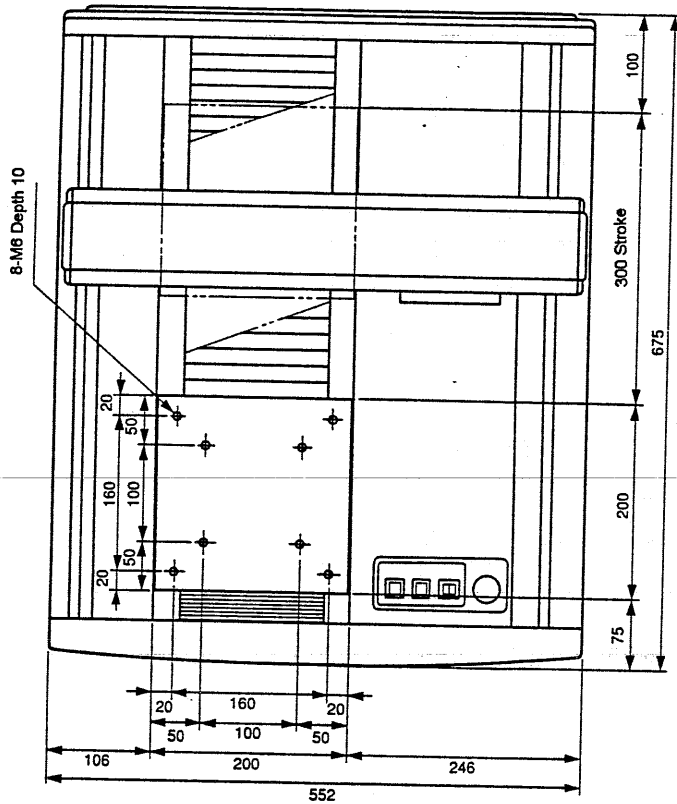


RS232C Adapter  
(25 Pin Male)

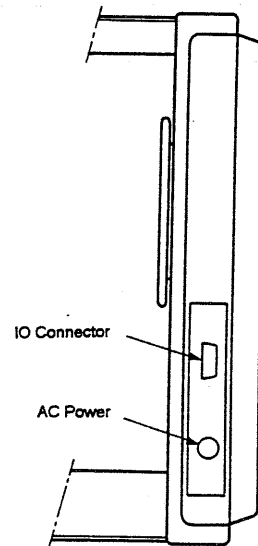
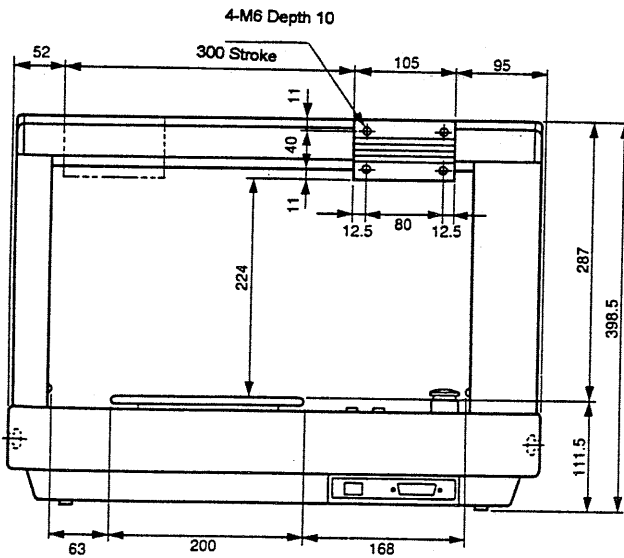
IBM PC  
(25 Pin Female)



## 5.5 TT-300 Dimensions



T-Slot Cross Section



Back Side

# 6. Programming

## 6.1 Super SEL Controller Language Description

The Super SEL Language used in Tabletop type is not exactly the same but almost as powerful as the Super SEL Language used in Super SEL Controller, which is capable of giving strong commands.

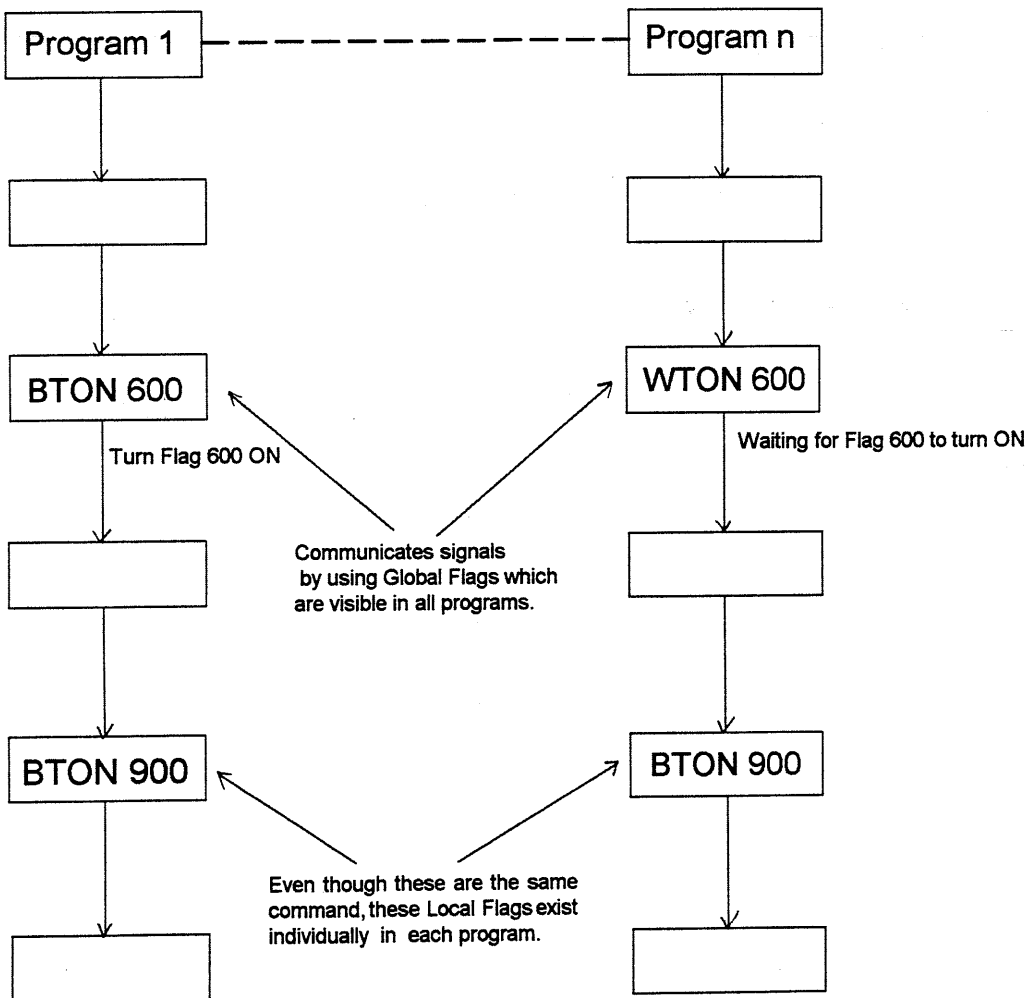
### Flags

The function of flags is to set and reset data within "Memory". This is analagous to "internal relays" or "coils".

In general, there are two (2) types of flags: Global Flags 600 ~ 887 which can be used in all programs; and Local Flags 900 ~ 999 which can be used *only* in individual programs.

Global Flags can be saved when the power is turned OFF (battery backup). Local Flags are erased when the power is turned OFF.

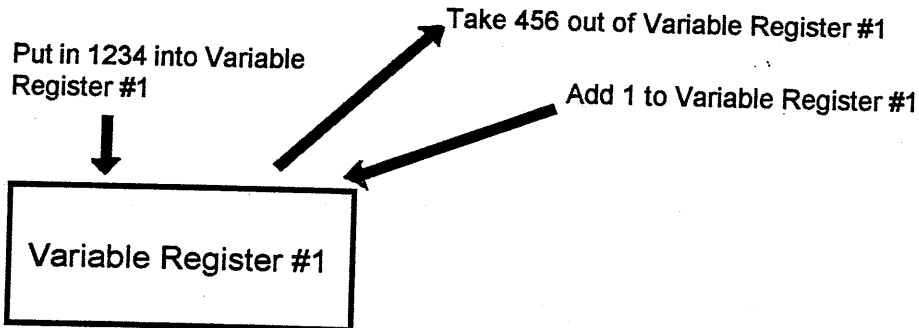
Flag Number	600 ~ 887	Global Flag : Visible To All Program
Flag Number	900 ~ 999	Local Flag : Visible Only Within A Program



## Variable Register

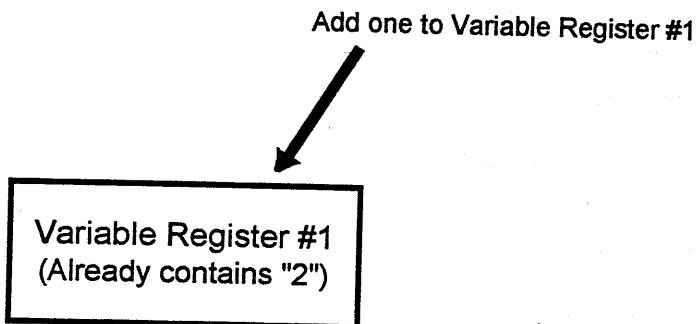
### (1) What are Variables?

The term "Variable Register" is a software term. Imagine a box that holds numbers. Numbers can be put in and taken out, added, subtracted, and so on.



Command	Operand 1	Operand 2
Add	1	1

This command adds 1 to Variable Register #1. If the Register contains 2, then the variable becomes 3.



---

(2) Types of Variables

2-1 Integer Variable. Whole numbers do not have decimal points. (For Example: -2, -1, 0, 1, 2, 3)

Integer Variable  
Register #1  
(1234)

Integer Variable No.	200 ~ 299	Global Integer Variable
Integer Variable No.	1 ~ 99	Local Integer Variable

2-2 Real Variable. For example: 1234.567

Integer Variable  
Register #1  
(1234.567)

Real Variable No.	300 ~ 399	Global Real Variable
Real Variable No.	100 ~ 199	Local Real Variable

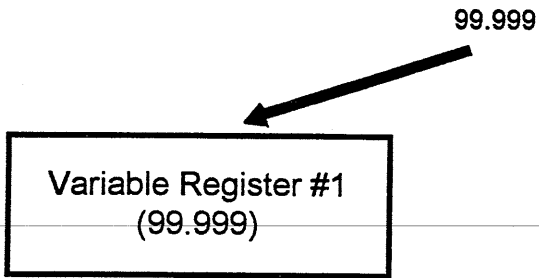


---

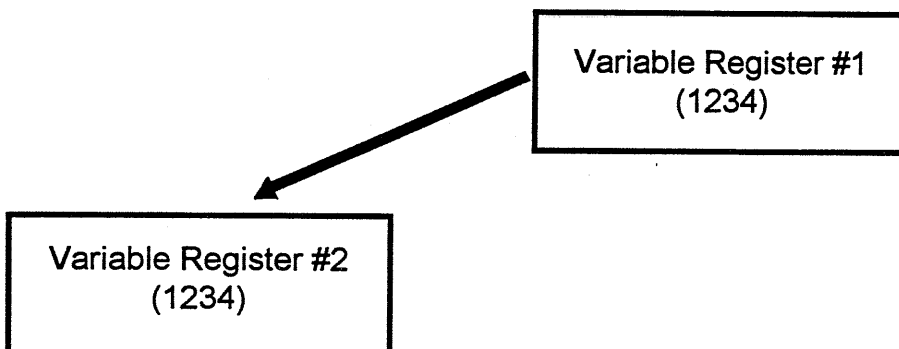
### 2-3 Asterisk

The Asterisk symbol (\*) is used to represent contents of the Variable Register which can be data in another program.

Command	Operand 1	Operand 2
LET	1	99.999



Command	Operand 1	Operand 2
LET	2	*1

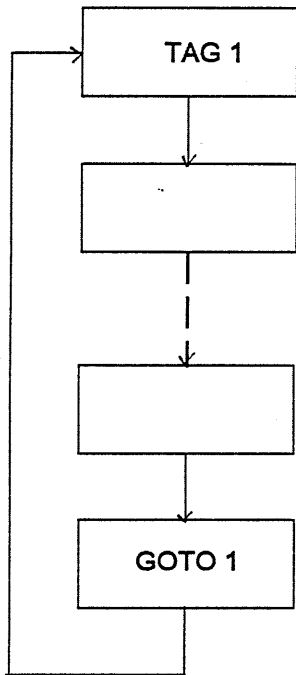
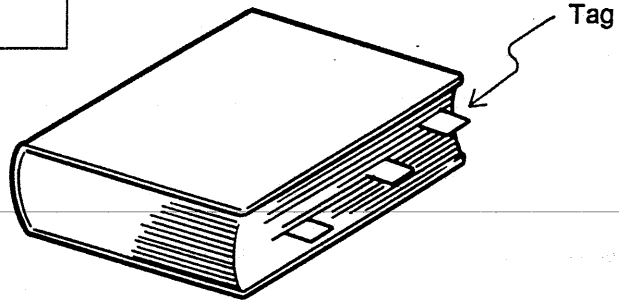


## Tag

"Tag" means heading. A TAG can be thought of as the same as placing labels on important pages. The TAG as it is used in the Super SEL programming language is the "return to" area and is used in conjunction with the GOTO command to provide programming loops.

Command	Operand 1
TAG	TAG No. (Integers 1 ~ 99)

Local to each program.



## Subroutine

Frequently repeated steps in a program can be expressed as subroutines in order to simplify the entire application program. These subroutines are individually usable in each program.

Command	Operand 1
BGSR	Subroutine No. (1 ~ 99 Integers)

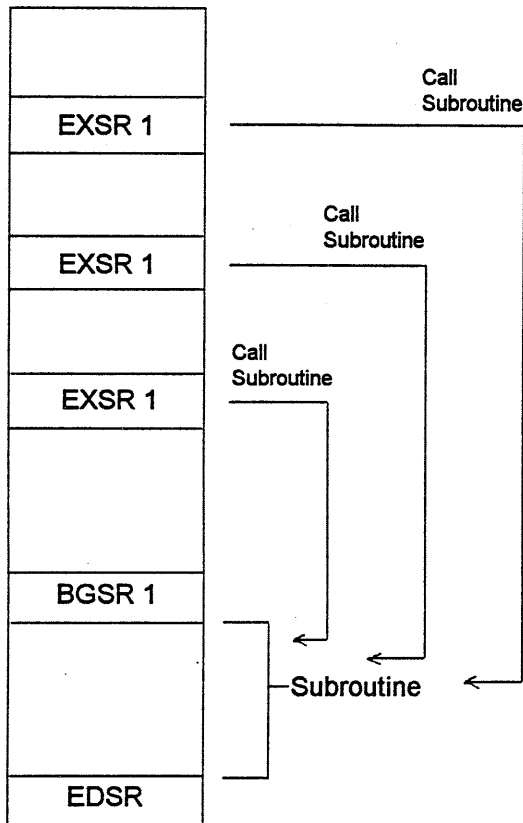
Begin subroutine command

Command	Operand 1
EXSR	Subroutine No. (1 ~ 99 Integers, or Variables)

Execute subroutine command

Command	Operand 1
EDSR	-----

End subroutine command



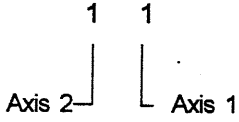
---

### Axis Selection

Selection of an Axis is specified by either "1" or "0"

### For Example

If Axis 1 and Axis 2 are in use, then this is signified by...



---

## 6.2 Super SEL Language Structure

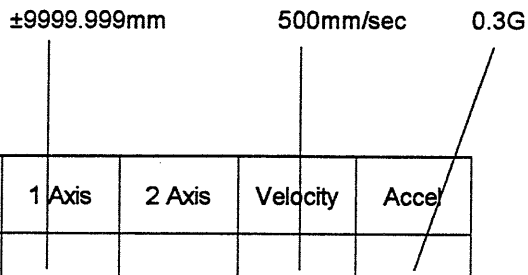
The Super SEL programming consists of a Position and Application Program (command) section.

### Position Program

In the Position Section, we have Coordinates, Velocity, Acceleration, and Variables

Position No.	1 Axis	2 Axis	Velocity	Accel
1				
2				
3				
4				
497				
498				
499				
500				

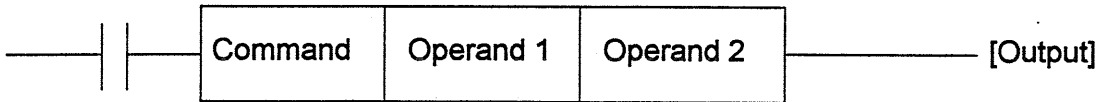
±9999.999mm      500mm/sec      0.3G



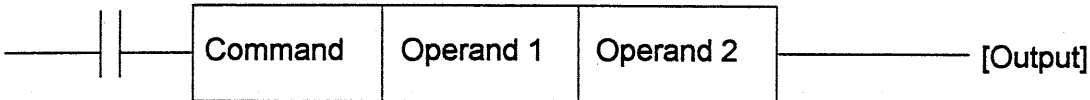
**Application Program (Commands)**

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	

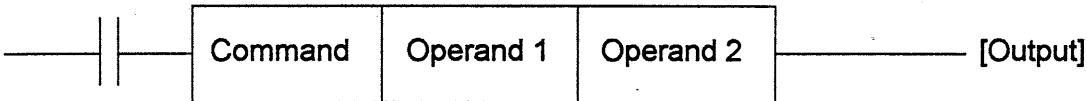
**Ladder Diagram Display**



The conditions before the commands are equivalent to "if ~ then" statements in BASIC language.



1. Carry out a command when an input condition is established. When not established, then go on to the next step.
2. If there is no conditioning set up, carry out command unconditionally.
3. If condition is used as "negative condition", then place an "N".
4. Input/output port & flag can be used for condition.



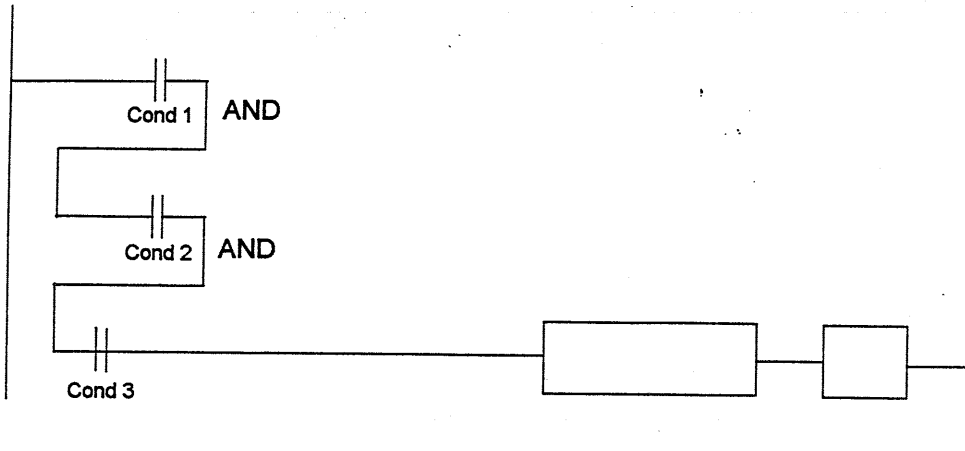
Output is set based on the result of the command execution.

---

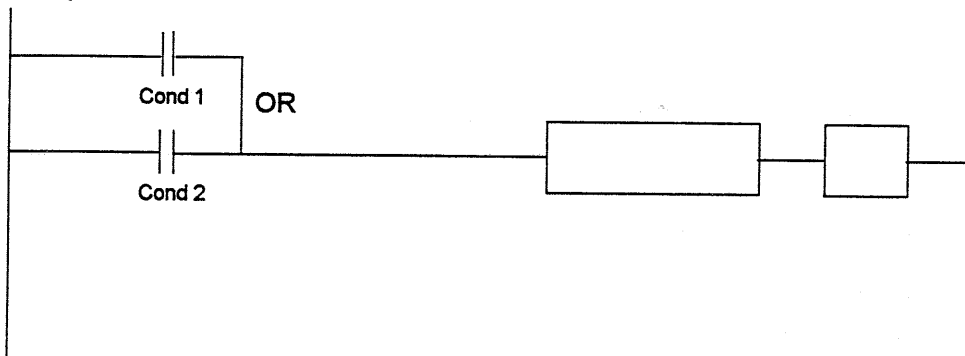
## Expansion Condition

It is possible to combine conditions to make more complicated conditions as follows:

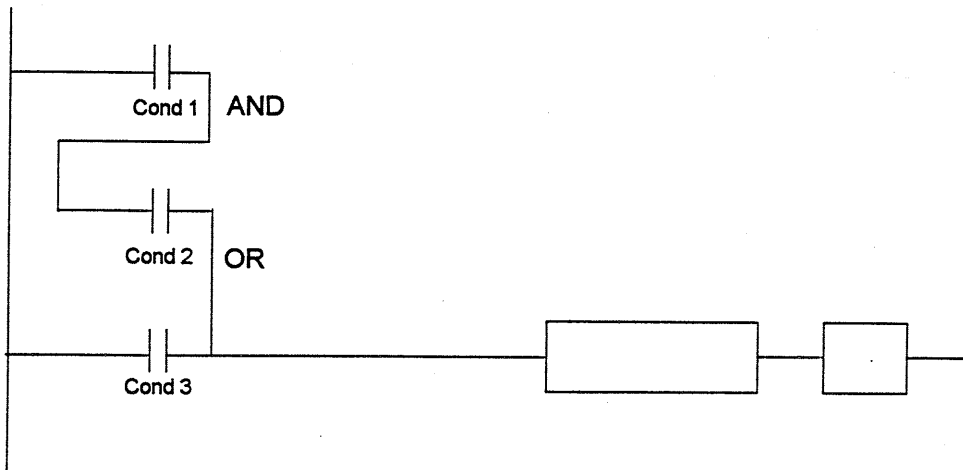
### AND Expansion



### OR Expansion



### AND/OR Expansion

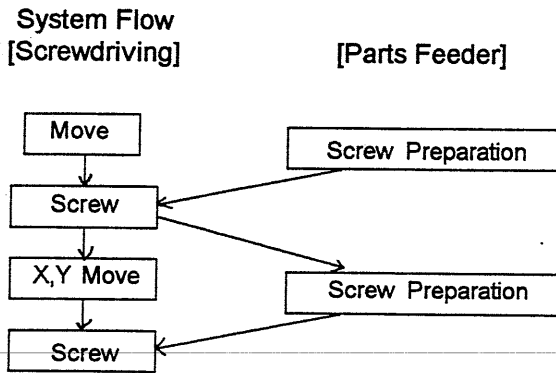


---

## Multi-tasking

"Multi Tasking" simply means running multiple programs concurrently.

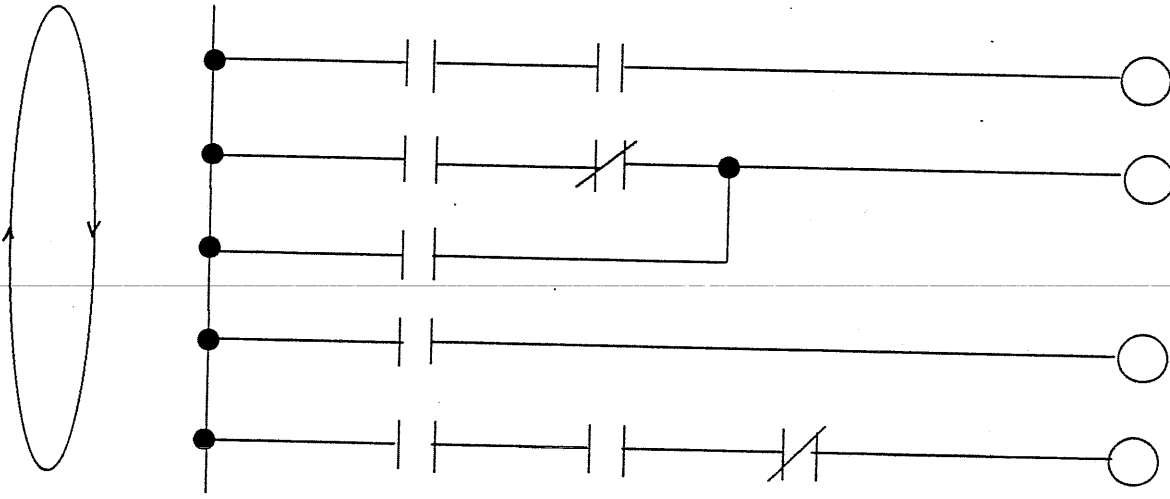
Let us consider a screwdriving robot system as an example. The following screwdriving system is composed of two actuators (X axis and Y axis) and a screwdriver with the part feeder.





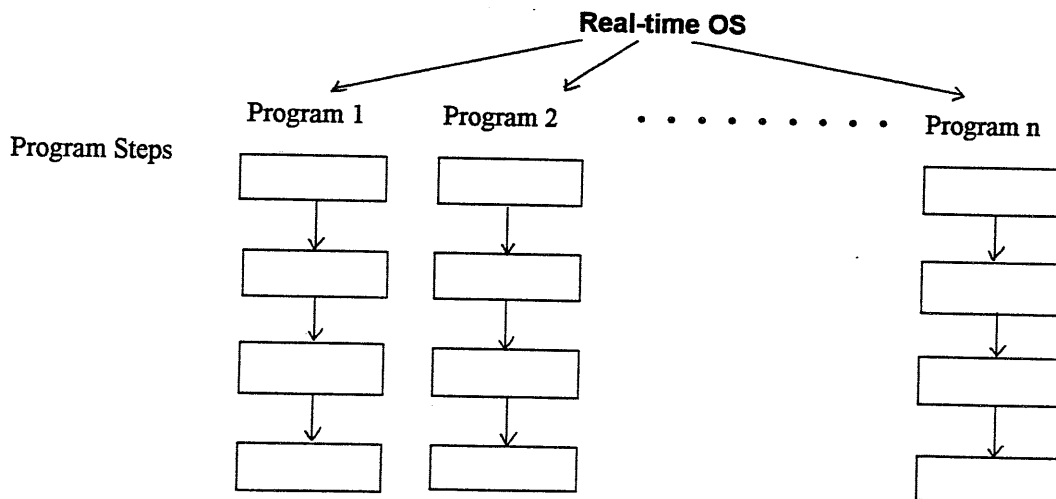
## Difference from PLC Operation

In the past, parallel processing was accomplished by relay ladder circuitry. It was subsequently replaced by PLC which is equipped with micro processors. There is considerable scanning time involved in PLCs. The following shows that PLC scans the entire program, then sets the post if certain conditions are met.



The Super SEL Operating System (OS) handles programs at a high speed and executes programs only when a certain event occurs (Event Driven), without scanning. Also, each parallel processing program is composed of simple one-line steps. This makes programming and debugging easy.

With the Super SEL controller, the user will be able to develop a multi tasking program without thinking about parallel processing, which is handled by the SEL Operating System (OS).



## 6.3 Command

### Command Table

Category	Function	Command
Actuator Control Designation	Set Velocity	VEL
	Set Acceleration	ACC
	Designate Velocity Coefficient	OVRD
	Designate Offset	OFST
	Designate Arc Degrees	DEG
	Designate Axis Temp. Stop Port	HOLD
	Designate Axis Stop Port	CANC
Actuator Control Commands	Designated Axis Servo ON	SVON
	Designated Axis Servo OFF	SVOF
	Designated Axis Execute Homing	HOME
	Designated Group Axes Point Move	MOV P
	Designated Group Axes Point Move	MOV L
	Circular Movement	CIR
	Arc Movement	ARC
	Path Movement	PATH
Input/Output Flag Operation Commands	Designated Output Port Flag ON	BTON
	Designated Output Port Flag OFF	BTOF
	Designated Output Port Flag Reverse	BTNT
	Designated Output Port Flag ON Waiting	WTON
	Designated Output Port Flag OFF Waiting	WTOF
	Read Binary Value	IN
	Read BCD Value	INB
	Binary Output	OUT
	BCD Output	OUTB
Timer Commands	Time (sec) Waiting	TIMW
	Other Program Timer Cancel	TIMC

Category	Function	Command
Program Control Commands	Program End	EXIT
	Start Other Program	EXPG
	Other Program Forced End	ABPG
Tag Command Designation	Jump	GOTO
	Designate Tag for Jump	TAG
Subroutine Control Designation Commands	Designate Start Subroutine	BGSR
	Designate End Subroutine	EDSR
	Call Subroutine	EXSR
Calculation Commands	Assign	LET
	Add	ADD
	Subtract	SUB
	Multiply	MULT
	Divide	DIV
	Rounding Off Numbers	MOD
	Clears Desig. Variable Range	CLR
Logic Commands	Logic Multiplication	AND
	Logic Addition	OR
	Exclusive Logic Addition	EOR
Comparison Commands	Equal	CMPE
	Greater Than	CMPG
	Less Than	CMPL
Point Data Operation Commands	Assign Point Data to Designated Axis	PPUT
	Read Point Data of Designated Axis	PGET
	Checks for Data in Designated Axis	PTST
	Copies Point Data	PCPY
	Clears Point Data	PCLR
	Reads Designated Point Data	PRED
	Point Data Range	PSIZ

## Commands

### (1) Actuator Control Designations

#### VEL (Velocity)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	VEL	Velocity		

[Function] The setup velocity of the Actuator is in mm/sec.

#### [Example]

VEL 1000  
1000mm/sec (Velocity Setting)

VEL 1000  
MOVP 1  This velocity is 1000mm/sec between these two points.  
MOVP 2   
VEL 500  
MOVP 3  This velocity is 500mm/sec between these two points.  
MOVP 4

#### ACC (Acceleration)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	ACC	Acceleration		

[Function] The setup acceleration of the actuator is expressed in G (Gravity).

#### [Example]

ACC 0.3  
0.3G (Acceleration setting at 0.3G)

OVRD (Override)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	OVRD	Velocity Ratio Value		

[Function] This command decreases the velocity according to the designated ratio. (Velocity coefficient setting). The range of the ratio settings is from 1 ~ 100%.

[Example]

VEL 100 100mm/sec setting  
 OVRD 50 100mm/sec is reduced by 50% and the actual velocity becomes 50mm/sec.

OFST (Offset)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	OFST	Axis Pattern	Offset Value	

[Function] During movement, this command adds an offset value to the specified Axes. Resolution is 0.001mm

[Example]

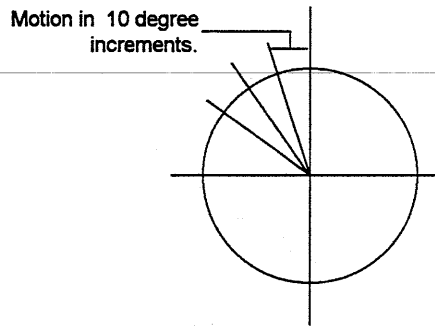
OFST 11 50.000  
 50mm is added to the movement amount of Axis 1 and Axis 2.

DEG (Degree Setting)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	DEG	Angle		

[Function] This command sets up the motion increments for use with CIR (Circular Movement) and ARC (Arc Movement) commands.

[Example] DEG 10

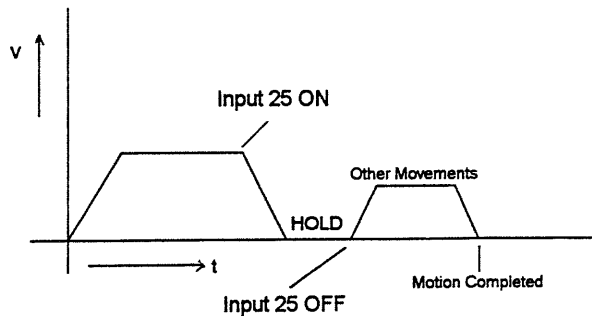


HOLD (Hold : Axis Temporary Stop)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	HOLD	Input Port		

[Function] If the actuator is moving and the designated Input Port turns ON, then velocity decreases until all motion stop. When the Input Port turns OFF, then motion begins again.

[Example] HOLD 25 (When Input Port 25 turns ON, velocity decreases until all motion stops.)

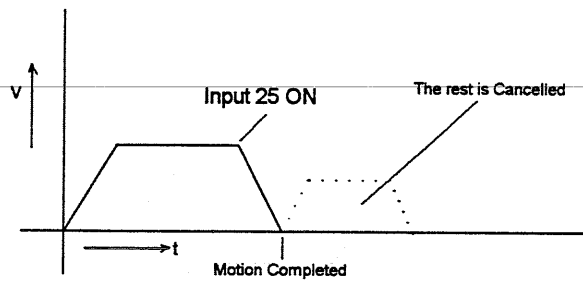


**CANC (Cancel : Cancelling the next steps after axis stop motion)**

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	CANC	Input Port		

**[Function]** If the actuator is moving and the designated input port turns ON, then velocity decreases until all motion stops. Any other programmed motion thereafter is cancelled and not executed.

**[Example]** CANC 25



(2) Actuator Control Commands

SVON (Servo ON)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	SVON	Axis Pattern		

[Function] This commands turns ON the Servos of the designated axes.

[Example] SVON 11

SVOF (Servo OFF)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	SVOF	Axis Pattern		

[Function] This commands turns OFF the Servos of the designated axes.

[Example] SVOF 11



HOME (Return Home)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	HOME	Axis Pattern		OPTIONAL

[Function] This command executes homing of the designated axes. Servos turn ON automatically.

[Example] HOME 11 Axis 1 and Axis 2 execute homing.

MOVP (Point-to-point Position Data)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	MOVP	Position No.		OPTIONAL

[Function] This command moves the actuator to the designated position number from point to point without interpolation.

[Example] MOVP 100 Moves to Position No. 100 (point to point)

MOVP \*1 (If the variable 1 is 150, then the actuator moves to position number 150 (point to point))

Position No.	1	2	3
1			
2			
3			
thru			
100	100.00	100.00	xxx.xx
thru			
150	200.00	200.00	xxx.xx
thru			

**MOVL (Position Data with Interpolation)**

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	MOVL	Position No.		OPTIONAL

[Function] Moves the actuator to the designated point along using interpolation (not point to point).

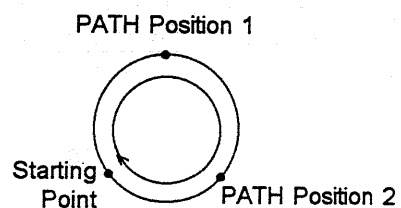
[Example] **MOVL 100** Move to position No. 100 using interpolation.

**MOVL \*1** If variable 1 is 150, then the actuator moves to position 150 using interpolation.

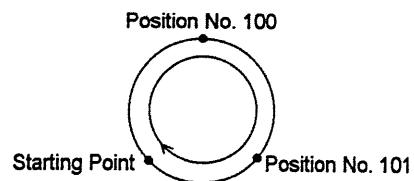
**CIR (Circular Movement)**

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	CIR	Passing Through Position 1	Passing Through Position 2	OPTIONAL

[Function] Three points determine circular motion. The starting point, Position 1, and Position 2. The rotation direction is determined by the position data. The following diagram shows CW (clockwise) motion but this can be changed to CCW (counterclockwise) by exchanging positions 1 and 2)



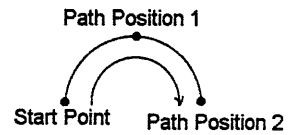
[Example] **CIR 100 101**  
The following diagram shows circular motion passing through position Numbers 100 and 101.



**ARC (Arc Movement)**

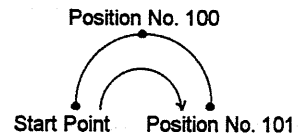
Expansion Condition (AND/OR)	Input Condition (Input/Ouput, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	ARC	Passing Through Position 1	Passing Through Position 2	OPTIONAL

[Function] Circular motion that starts from the present point and which passes through position No. 1 and 2 is called ARC motion.



[Example] ARC 100 101

The following diagram shows Arc motion passing through position numbers 100 and 101.



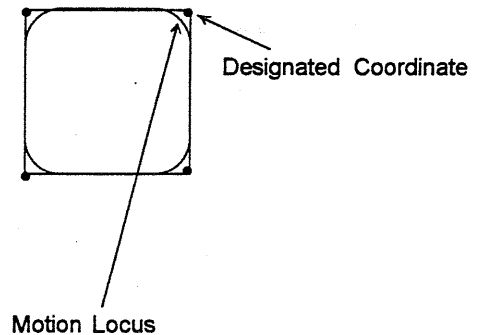
**PATH (Path Movement)**

Expansion Condition (AND/OR)	Input Condition (Input/Ouput, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	PATH	Starting Position	Ending Position	OPTIONAL

[Function] Actuator moves continuously between the designated starting point and the finishing point. the locus is a B-spline-type, free-form curve which passes through the inside of the designated coordinate. It is possible for the actuator to move close to the designated coordinate by increasing the aceleration. However, when it exceeds the maximum acceleration, an error will occur.

[Example] PATH 100 120

Moves to position 100 ~ 120, Continuously.



(3) I/O, Flag Operation Commands

BTON (Output Port, Flag ON)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	BTON	Output Port, Flag	Output Port, Flag	

[Function] Turns ON the designated output port or flag, or the range of ports or flags.

[Example] BTON 300 Output Port 300 turns ON  
 BTON 300 310 Output Port 300 ~ 310 turns ON  
 BTON 600 Output Port 600 turns ON  
 BTON 600 610 Output Port 600 ~ 610 turns ON

BTOF (Output Port, Flag OFF)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	BTOF	Output Port, Flag	Output Port, Flag	

[Function] Turns OFF the designated output port or flag, or the range of ports or flags.

[Example] BTOF 300 Output Port 300 turns OFF  
 BTOF 300 310 Output Port 300 ~ 310 turns OFF  
 BTOF 600 Output Port 600 turns OFF  
 BTOF 600 610 Output Port 600 ~ 610 turns OFF

**BTNT (Output Port, Flag Invert)**

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	BTNT	Output Port, Flag	Output Port, Flag	

**[Function]** Inverts the designated output port or flag, or range of ports or flags. When the command is performed, ON changes to OFF and OFF changes to ON.

**[Example]**

BTNT	300		Reverse Output Port 300
BTNT	300	310	Reverse Output Port 300 ~ 310
BTNT	600		Reverse Flag 600
BTNT	600	610	Reverse Flag 600 ~ 610

**WTON (Waiting Input/Output Port, Flag ON)**

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	WTON	Input Port, Flag	Time Out	OPTIONAL

**[Function]** Wait until the designated I/O port or flag turns ON. Do not go to the next step. I/O waiting can be cut off at the time that was set by the time out setting. The time out setting is in units of seconds (0.01 ~ 99 sec).

**[Example]**

WTON	25		Wait until Input Port 25 turns ON
WTON	25	10	Wait until Input Port 25 turns ON. After 10 seconds, it becomes time out, then go on to the next step.

WTOF (Waiting for Input/Output Port, Flag OFF)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	WTOF	Input Port, Flag	Time Out	OPTIONAL

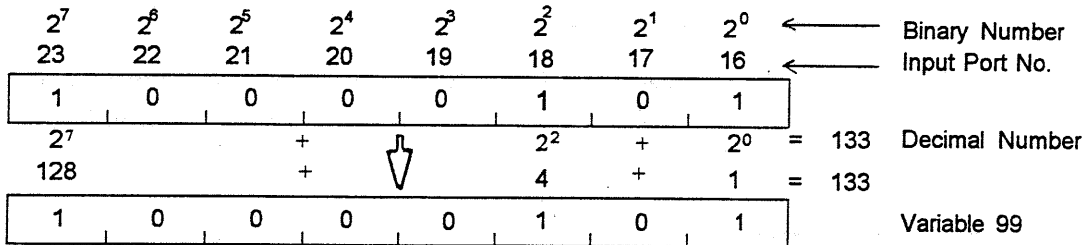
[Function] Wait until the designated I/O port or flag turns OFF. Do not go on to the next step. I/O waiting can be cut off at the time that was set by the time out setting. The time out setting is in units of seconds (0.01 ~ 99 sec).

[Example] WTOF 25                      Input Port 25 turns OFF  
 WTOF 25      10      Input Port 25 turns OFF

IN (Binary Numbers Input/Output, Read in)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	IN	Start Input/Output Port, Flag	Finish Input/Output Port, Flag	

[Function] Read the value as a binary number based on the designated I/O port or flag, then load the Variable Register 99 with the value read.

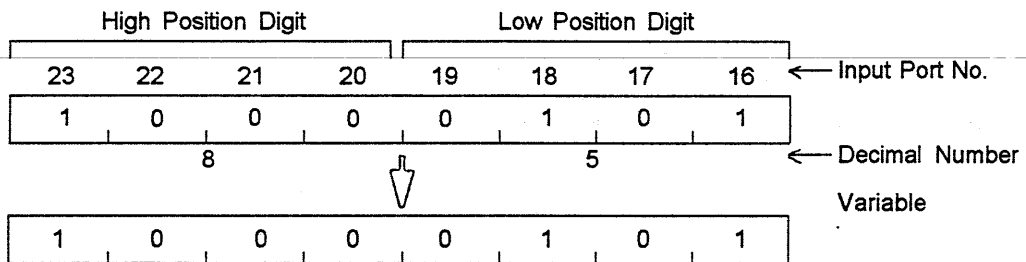


[Example] IN      16      23  
 Read the binary values based on input ports 16 ~ 23. Load Variable Register 99 with the value.

INB (BCD Input/Output, Read in)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	INB	Start Input/Output Port, Flag	BCD Value	

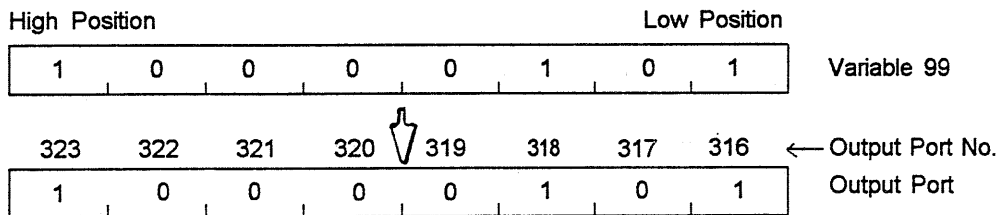
[Function] Read BCD value from designated I/O port or flag, then load Variable Register 99 with the binary value read.



OUT (Binary Number Output)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	OUT	Output Port, Flag	Finish Output Port, Flag	

[Function] Output the binary value of Variable Register 99 to the designated output port or flag (range).



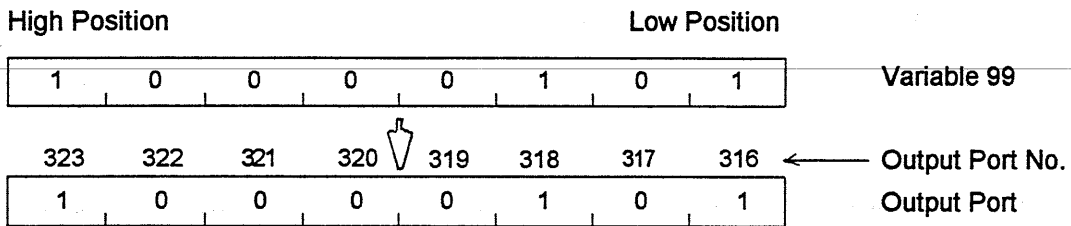
[Example] OUT 316 323

Output the binary value of Variable Register 99 to output ports 316 ~ 323.

OUTB (BCD Output)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	OUTB	Output Port, Flag	BCD Value	

[Function] Output the BCD value of the Binary Variable Register 99 to the designated output port or flag by BCD value.





(4) Timer Command

TIMW (Timer Command)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	TIMW	Time		OPTIONAL

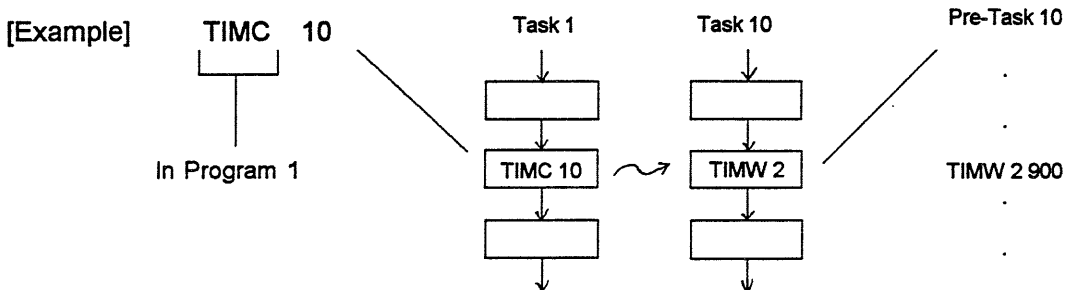
[Function] Waiting according to time set. The time set unit is expressed in seconds (0.01 ~ 99 sec)

[Example] TIMW 1.5 Wait for 1.5 seconds

TIMC (Timer Cancel)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	TIMC	Program No.		

[Function] Cancel (set) the timer of the other concurrent task.



(5) Program Control Commands

EXIT (Program Exit)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	EXIT			

[Function] Finish program.

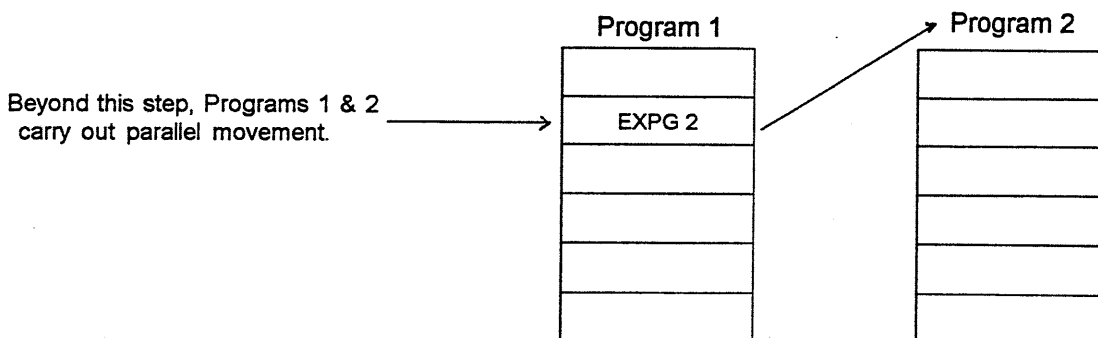
[Example] EXIT Program Finish

EXPG (Execute Other Program)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	EXPG	Program No.		

[Function] Start another program, conduct parallel processing (other programs).

[Example] EXPG 2 Start program No. 2, then conduct parallel processing.

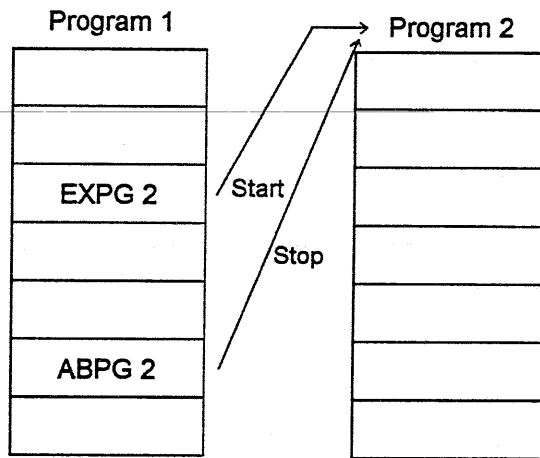


**ABPG (Abandon Other Program)**

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	ABPG	Program No.		

**[Function]** Make the other program actually in practice complete.

**[Example]**



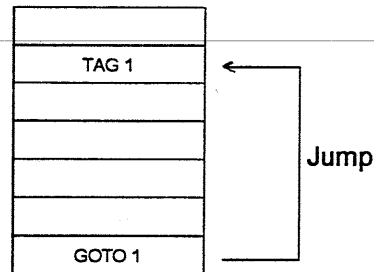
(6) Turnout Commands

GOTO (Jump)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	GOTO	Tag No.		

[Function] Go to the designated step by Tag No.

[Example] Go to Tag 1 Step.

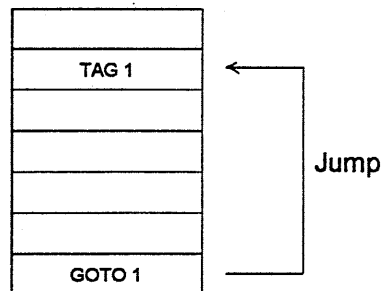


TAG (Tag Designation)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	TAG	Tag No.		

[Function] Designate the jumping place of the GOTO Command via Tag No.

[Example] Prepare Tag 1 as the jumping place for GOTO 1.



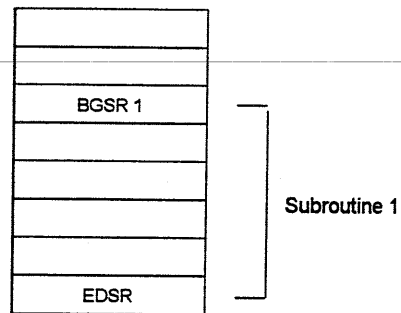
(7) Subroutine Control Commands

BGSR (Begin Subroutine)

Expansion Condition (AND/OR)	Input Condition (I/O, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
		BGSR	Subroutine No.		

[Function] Declare subroutine starting

[Example]

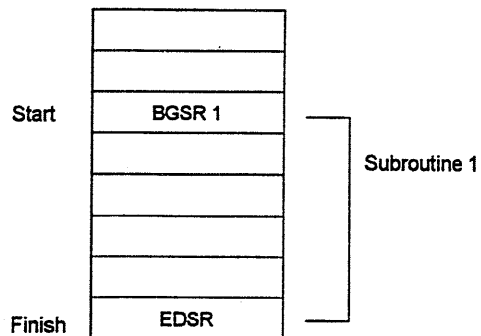


EDSR (End Subroutine)

Expansion Condition (AND/OR)	Input Condition (I/O, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
		EDSR			

[Function] Declare subroutine complete

[Example] Declare Subroutine complete. EDSR is always used as the very last step.

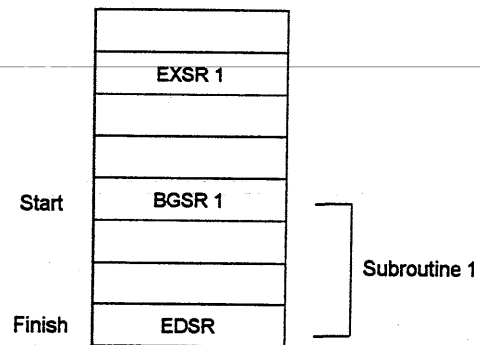


**EXSR (Execute Subroutine)**

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	EXSR	Subroutine No.		

[Function] Perform the designated numbered subroutine.

[Example] Perform subroutine 1



(8) Calculation Commands

LET (Assignment)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	LET	Variable No.	Data or Variable Register	OPTIONAL

[Function] Substitute variable for data.

[Example] LET 1 10 Assign 10 to Variable Register 1.

ADD (Addition)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	ADD	Variable No.	Data Variable No.	OPTIONAL

[Function] If there is a post instruction (flag, output port), it turns ON when the computing result becomes zero.

[Example] ADD 1 10 When the variable content in Operand 1 is 10, 10 (variable 1 content in Operand 1) + 10 (data in Operand 2) equals 20, then after performing the command, the variable 1 becomes 20.

ADD 1 \*2 When variable 1 content in Operand 1 and variable 2 content in Operand 2 are both 10, then 10 (variable 1 content in Operand 1) + 10 (variable 2 content in Operand 2) equals 20, then after performing the command, variable 1 becomes 20.

SUB (Subtraction)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	SUB	Variable No.	Data Variable No.	OPTIONAL

[Function] Subtract the data in Operand2 from the variable content in Operand 1, then load the Operand 1 variable with it. Subtract the variable content in Operand 2 from the variable content in Operand 1, then load the Operand 1 variable with it.

[Example] SUB 1 10 When variable 1 content in Operand 1 is 20, 20 (variable 1 content in Operand 1) -10 (data in Operand 1) equals 10. After performing the command, the variable becomes 10.

SUB 1 \*2 When variable 1 content in Operand 1 is 20 and variable 2 content in Operand 2 is 10, then 20 (variable 1 content in Operand 1) - 10 (variable 2 content in Operand 2) equals 10. After performing the command, the variable 1 becomes 10.

MULT (Multiplication)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	MULT	Variable No.	Data Variable No.	OPTIONAL

[Function] Multiply the variable content in Operand 1 by the data in Operand 2, then load the Operand 1 variable with it. Multiply the variable content in Operand 1 by the variable contents in Operand 2, then load the Operand 1 variable with it. If there is an instruction (flag, output port), it turns ON when the computing result is zero.

[Example] MULT 1 10 When the variable 1 content in Operand 1 is 10, -10 (variable 1 content in Operand 1) x 10 (data in Operand 2) equals 100, then after performing the command, the variable 1 becomes 100.

MULT 1 \*2 When variable 1 content in Operand 1 and variable 2 content in Operand 2 are both 10, 10 (variable 1 content in Operand 1) x 10 (variable 2 content in Operand 2) = 100, then after performing the command, the variable 1 becomes 100.



**DIV (Division)**

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	DIV	Variable No.	Data Variable No.	OPTIONAL

**[Function]** Divide the variable content in Operand 1 by the data in Operand 2, then load the Operand 1 variable with it. Divide the variable content in Operand 1 by the variable content in Operand 2, then load the Operand 1 variable with it.

**[Example]**     **DIV   1     5**                    When variable 1 in Operand 1 is 10, 10 (variable 1 content in Operand 1) divided by 5 (data in Operand 2) equals 2.

**DIV   1     \*2**                When the variable 1 in Operand 1 is 10 and variable 2 in Operand 2 is 5, 10 (variable 1 content in Operand 1) divided by 5 (variable \*2 content in Operand 2) equals 2. After performing the command, the variable 1 becomes 2.

**MOD (Module)**

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	MOD	Variable No.	Data Variable No.	OPTIONAL

**[Function]** If there is an instruction (flag, output port) at output area, it turns ON when the computing result becomes zero.

**[Example]**     **MOD   1     3**                    When the Variable 1 content in Operand 1 is 10, 10 (variable 1 content in Operand 1) divided by 3 (data in Operand 2) = 3 and 1 over. After performing the command, the variable 1 becomes 1.

**MOD   1     \*2**                When variable 1 in Operand 1 is 10 and variable 2 content in Operand 2 is 3, 10 (variable 1 content in Operand 2) divided by 3 (variable 2 content in Operand 2) is 3 and 1 over. After performing the command, the variable 1 becomes 1.

**CLR (Clear Variables)**

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	CLR	Variable No.	Variable No.	OPTIONAL

[Function] Clear the designated range of variables to zero.

[Example] CLR 1 Clear the variable 1 to zero.

CLR 1 10 Clear the variables 1 ~ 10 to zero.

(9) Logical Operation Commands

AND (Bitwise AND)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	AND	Variable No.	Data Variable No.	OPTIONAL

[Function] Load the variable in Operand 1 with AND result of variable content in Operand 1 and data in Operand 2. Load variable in Operand 1 with AND result of variable content in Operand 1 and variable content in Operand 2. If there is an instruction (flag, output port) at output area, it turns ON when the computing result is zero.

[Example] AND 1 3 When the variable 1 content in Operand 1 is 131 by the decimal (1000011) and the data condition in Operand 2 is 3 by the decimal, the result of AND is 3 (by decimal (0000011) by binary number the variable 1 becomes 3.

Variable 1 (Operand 1)	1000011	AND
Data (Operand 2)	0000011	
Result (Load Variable 1)	0000011 (3 by decimal, the variable becomes 3)	

OR (Bitwise OR)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	OR	Variable No.	Data Variable No.	OPTIONAL

[Function] Load variable in Operand 1 with OR result of variable content in Operand 1 and data in Operand 2. Load variable in Operand 1 with OR result of variable content in Operand 1 and variable content in operand 2. If there is an instruction (flag, output port) at output area, it turns ON when the computing result is zero.

[Example] OR 1 3 When the variable 1 content in Operand 1 is 128 by decimal (1000000 by binary number), and the data in Operand 2 is 3 by the decimal (0000011 by binary number) the result of OR is 131 by decimal (10000011 by binary number).

Variable 1 (Operand 1)	1000000	OR
Data (Operand 2)	0000011	
Result (Load Variable 1)	1000011 (131 by decimal. The variable 1 becomes 131)	

EOR (Bitwise Exclusive OR)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	EOR	Variable No.	Data Variable No.	OPTIONAL

[Function] Load variable in Operand 1 with EOR result of variable content in Operand 1 and data in Operand 2. Load variable in Operand 1 with EOR result of variable content in Operand 1 and variable content in Operand 2. If there is any instruction (flag, output port) at output area, it turns ON when the computing result is zero.

[Example] EOR 1 3 When the variable 1 content in Operand 1 is 128 by decimal (10000000 by binary number) and the data in Operand 2 is 3 by decimal (00000011 by binary number) the result of EOR is 124 by decimal (01111100 by binary number). Therefore, the variable 1 becomes 124.

Variable 1 (Operand 1)      10000000  EOR  
 Data (Operand 2)          00000011   
 Result (Load Variable 1)    01111100 (124 by decimal. The variable 1 becomes 124)

(10) Calculation Comparison

CMPE (Compare if Equal)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	CMPE	Variable No.	Data Variable No.	NECESSARY

[Function] If the values in Operand 1 and 2 are the same (equal), the output flag and output port will be turned ON.

[Example] CMPE 1 5 600 When the variable 1 content in Operand 1 becomes 5 and if it matches the value of Operand 2 data, then it turns ON output 600.

CMPE 1 \*2 600 When variable 1 content in Operand 1 matches the variable 2 content in Operand 2, then output 600 will be turned ON.

CMPG (Compare if Greater Than)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	CMPG	Variable No.	Data Variable No.	NECESSARY

[Function] When the Operand 1 value is greater than the Operand 2 value, then the output flag or output port turns ON.

[Example] CMPG 1 5 600 When variable 1 content in Operand 1 is greater than the value of Operand 2 data, then the output 600 turns ON.

CMPG 1 \*2 600 When the variable 1 content in Operand 1 is greater than the variable 2 content in Operand 2, then output 600 turns ON.

CMPL (Compare if Less Than)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	CMPL	Variable No.	Data Variable No.	NECESSARY

[Function] When the value in Operand 1 is smaller than the value in Operand 2, output flag and output port will be turned ON.

[Example] CMPL 1 5 600 When variable 1 content in Operand 1 is smaller than the data value in Operand 2, then output 600 will be turned ON.

CMPL 1 \*2 600 When the variable 1 content in Operand 1 is smaller than the variable 2 content in Operand 2, then output 600 will be turned ON.

(11) Position Data Operation Commands

PPUT (Axis Data Assignment)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	PPUT	Axis No.	Position No.	

[Function] Substitute the variable 199 coordinate for the designated axis position data.

[Example] PPUT 2 3 Substitute the variable 199 coordinate for the axis no. 2 position No. 3.

Variable 199 (Common Variable)

50.00 Load the variable 199 with coordinate calculated by computing results.

Position No.	1Axis	2Axis
1		
2		
3		
4		

PGET (Axis Data Read)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	PGET	Axis No.	Position No.	

[Function] Read the designated axis data of position data to the variable 199.

[Example] PGET 2 3 Read the data of axis No. 2, Position No. 3 at variable 199.

PTST (Position Data Check)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	PTST	Axis No.	Position No.	Necessary

[Function] Check to see whether effective data exists in the designated pattern and the position No. If there is any data, it turns on a flag in the output area or output port.

[Example] PTST 11 10 600

If there is data in position 10 of axis 1 and 2, it turns on flag 600.

Position No.	1Axis	2Axis
1		
2		
3		
10	50.000	100.000
11	XX.XXX	XX.XXX



PCPY (Position Data Copy)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	PCPY	Position No.	Position No.	

[Function] Copy data in the designated position No. (copy data from Operand 1 to Operand 2).

[Example] Copy data from position 10 in Operand 1 to position 20 in Operand 2.

Copy

Position No.	1Axis	2Axis
10	50.000	100.000
20	50.000	100.000





PCLR (Position Data Clear)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	PCLR	Position No.	Position No.	

[Function] Clear the position data in the designated area from Operand 1 to Operand 2.

[Example] PCPY 10 20 Clear data from position 10 in Operand 1 through position 20 in Operand 2.

PRED (Coordinate Read)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	PRED	Axis Pattern	Position No.	

[Function] Read the present coordinate of the designated axis in Operand 1. Write it in the designated position in Operand 2.

[Example] PRED 11 10 Write the present coordinates of the designated axis 1 and axis 2 in Operand 1 into position 10.

PSIZ (Position Data Size Check)

Expansion Condition (AND/OR)	Input Condition (Input/Output, Flag)	Commands			Output Section (Output Port, Flag)
		Commands	Operand 1	Operand 2	
OPTIONAL	OPTIONAL	PSIZ	Variable (Assigned To)		

[Function] Check the maximum size of the usable position data.

[Example] PSIZ 1 The maximum of position data goes in the variable 1 (substituted number) in Operand 1.

## 6.4 Parameter List

### (1) Axis Parameters

#### a) Servo Control

No.	Parameter Name	Default	Content
1	Numerator	1	Numerator
2	Denominator	1	Denominator
3	Override (%)	100	Velocity Scale Factor
4	Acceler (0.01G)	0.30	Acceleration
5	Jog Vel	30	Jog Velocity
6	Pend Band	10	Position End Band (Pulse)
7	Soft Limit Off	2000	Software Limit Offset
8	Soft Limit (+)	9999	Software Limit (+)
9	Soft Limit (-)	0	Software Limit (-)

#### b) Homing

No.	Parameter Name	Default	Contents
1	Home Dir	0	Home Direction
2	Home Type	0	Home Type
3	Home Sequence	0 ~ 9	Sequence (Axis Used, Unused)
4	Home SW Pol	1	Limit Input Polarity
5	Home Z Edge	1	Z-phase detect edge
6	Home Creep Vel	100	Creep Velocity
7	Home Back Vel	10	Run-in Velocity
8	Home Z Vel	5	Z-Phase Search Velocity
9	Home Offset	0	Offset Move Amount (Length)
10	Home Deviation	667	Deviation (Pulse)
11	Home Current	60	Current Limit

c) Motor

No.	Parameter Name	Default	Content
1	Motor RPM Max	4000	Motor RPM Maximum
2	Encoder Pulse	400	Encoder Pulse Per Rev.
3	Screw Lead	8	Screw Lead (mm)
4	Multiple	4	Encoder Pulse Multiplier
5	Brake Time	0.1	Brake Time (sec)
6	Position Gain	60	Position Gain
7	Speed Gain	80	Speed Gain
8	F/F Gain	0	Feed Forward Gain
9	Integral Gain	30	Integral Gain
10	Total Gain	150	Total Gain
11	Int. Volt. Lmt.	60	Integral Voltage Limit
12	Over Speed	410	Over Speed Constant
13	Error Range	2666	Error Range
14	Motor Max Cur	90	Motor Maximum Current
15	Motor Over Load	16300	Motor Overload Lower Limit

d) Axis Name

No.	Parameter Name	Default	Content
1	Axis 1	1	Axis Name 0~1, A-Z Setting
2	Axis 2	2	Axis Name 0~1, A-Z Setting
3	Axis 3	3	Axis Name 0~1, A-Z Setting
4	Axis 4	4	Axis Name 0~1, A-Z Setting
5	Axis 5	5	Axis Name 0~1, A-Z Setting
6	Axis 6	6	Axis Name 0~1, A-Z Setting
7	Axis 7	7	Axis Name 0~1, A-Z Setting
8	Axis 8	8	Axis Name 0~1, A-Z Setting

(2) System Parameters

a) Application Program

No.	Parameter Name	Setting Value	Contents	TT300
1	Auto Start PRG	0	Auto Start Program Number	-
2	Emergency PRG	0	Emergency Stop Program Number	-
3	Program Size	48	Program Size	32
4	Task Size	8	Task Size	4
5	Step Size	2000	Program Step Size	1000
6	Time Slice	0.10	Time Slice Check	-

b) Position Data

No.	Parameter Name	Setting Value	Contents	TT300
1	Point Size	1000	Position Data Number	500

c) Servo Device

No.	Parameter Name	Default	Content
1	Axis Size	8	Axis Number
2	Numerator	1	Numerator
3	Denominator	1	Denominator
4	Override (%)	100	Override
5	Acceler (0.01G)	0.30	Acceleration
6	Acc Max (0.01G)	1.00	Maximum Acceleration
7	Drive Vel	100	Drive Velocity (mm/sec)
8	Drive Vel Max	1000	Drive Velocity Max (mm/sec)

d) Communication

No.	Parameter Name	Default	Contents
1	Terminal ID	99	Multi-Drop Code
2	Time Out (sec)	0	Time Out
3	Baud Rate	3	Baud Rate
4	Char Length	0	Character Length
5	Parity	1	Parity
6	Stop Bit	0	Stop Bit

e) Circular/Path

No.	Parameter Name	Default	Contents
1	Circle Angle	150	Slice Angle (1/10 Degrees)
2	Circle Delt	0	Velocity Increment

## 6.5 Error Codes

Error Code	Error Name	Explanation
A1	External Interrupt Error	1. Motor over current 2. Over regenerative current (Over negative load)
A2	Motor Overload Error	Mechanical overload of motor
A3	Deviation Error	Motor is unable to perform properly due to mechanical overload
A4	Software Limit Error	Exceeded software limit
B0	No Program Error	Program does not exist
B1	Program Execution Error	Execution of a currently executing program
B2	Program Over Error	Number of tasks exceeds those set as parameters
B3	Double Subroutine Number Error	Two or more of the same subroutine number are used
B4	Double Tag Number Error	Two or more of the same tag number are used
B5	Undefined Subroutine Number	Subroutine number is not defined.
B6	Undefined Tag Number	Tag number is not defined
B7	Subroutine Pair Error	BGSR and EDSR are not the same quantity
B8	Step 1 BGSR Error	Step 1 is a BGSR Error
C0	No Homing Error	Homing was not performed before running actuators
C1	Point Data Error	Attempt has been made to execute unregistered Point Data
C2	Axis Double Execution Error	Move command given to axis currently moving
C3	Software Limit Error	Software limit exceed in program
D0	Acceleration Error	Acceleration exceeds limits
D1	No Velocity Error	Velocity has not been set
D2	Override Error	Override was set outside the range of 0~100%
D3	Angle Error	Angle was set outside the range of 0.1~120 degrees
D4	Axis Pattern Error	Axis Pattern was not set correctly
D5	Axis Number Error	Axis Number was set outside the range of 1~8
D6	Axis Error	More than 3 axes are designated in circular/arc motion
D7	Program Number Error	Program number exceeds the limit
D8	Position Number Error	Position number exceeds the limit
D9	Point Number Error	Negative number was input in the point number
DA	PIO Number Error	Input, Output, and Flag are not assigned correctly
DB	Variable Error	Variable is not assigned correctly
DC	Digits Over Error	Assigned number exceed 8 digits (binary 32 bit)
DD	Division (0) Error	Result of the division is "0"
DE	Circular Motion Computation Error	Position data that cannot perform circular motion was input
E0	Undefined Command Error	Undefined command execution attempted
E1	Subroutine Over Nesting Error	Nesting of more than 15 subroutines
E2	Subroutine Under Nesting Error	EXSR and EDSR are not making a pair
E3	Controlling Column Error	Use of conditions is not correct

Step	A/O	N	OP-CODE	OPRND1	OPRND2	POST	COMMENT
1							
2							
3							
4							
5							
6							
7							
8							
9							
0							
1							
2							
3							
4							
5							
6							
7							
8							
9							
0							
1							
2							
3							
4							
5							
6							
7							
8							
9							
0							

No.	1 (x)	2 (y)	COMMENT
1			
2			
3			
4			
5			
6			
7			
8			
9			
0			
1			
2			
3			
4			
5			
6			
7			
8			
9			
0			
1			
2			
3			
4			
5			
6			
7			
8			
9			
0			



