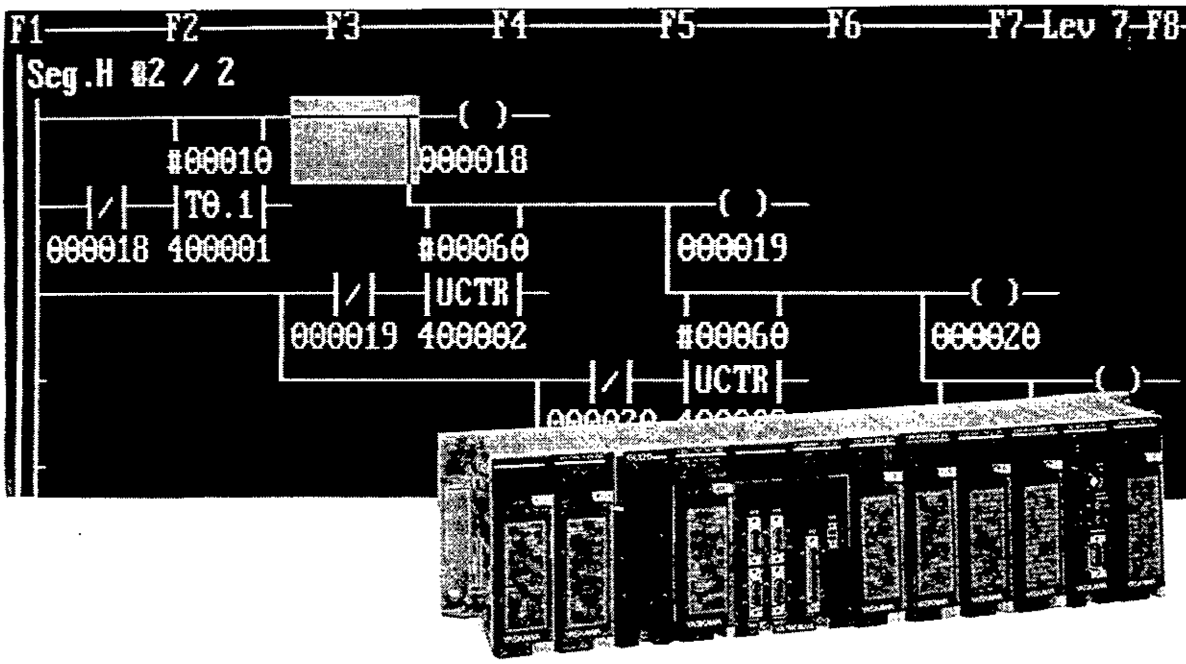
YASKAWA

MEMOCON GL120, GL130

Motion Module MC20

SOFTWARE USER'S MANUAL



YASKAWA

MANUAL NO. SIEZ-C825-20.52C

Manual Contents

This manual describes the installation and operation of the MC20 Motion Module, and explains how to create motion programs (stored in the MC20 Module) and motion ladder logic programs (stored in the CPU Module) for use with the MC20 Motion Module, in conjunction with the MEMOCON GL120 or GL130 PLCs.

This manual also explains new step-2 functions and version B08 functions. Systems that support step-2 functions and version B08 functions are listed in *1.1 Overview of Manual.*

Please read this manual carefully and be sure you understand the information provided before attempting to install and operate the MC20 Motion Module.

Visual Aids

The following aids are used to indicate certain types of information for easier reference.

Ver. B08	Indicates additional information on version B08 functions.
STEP 2	Indicates additional information on step-2 functions.
☞	Indicates references for additional information.
IMPORTANT	Indicates important information that should be memorized.
◀EXAMPLE▶	Indicates application examples.
INFO	Indicates supplemental information.
♦ SUMMARY	Indicates a summary of the important points of explanations.
Note	Indicates inputs, operations, and other information required for correct operation but that will not cause damage to the device.
TERMS	Indicates definitions of terms used in the manual.

NOTICE

The following conventions are used to indicate precautions in this manual. Failure to heed precautions provided in this manual can result in injury to people or damage to the products.

⚠**WARNING**	Indicates precautions that, if not heeded, could possibly result in loss of life or serious injury.
⚠**Caution**	Indicates precautions that, if not heeded, could result in relatively serious or minor injury, damage to the product, or faulty operation.

OUTLINE TABLE OF CONTENTS

Note The sections of this manual are classified into the following three categories. These categories can be used as guides to which sections are relevant to the user.

1 Sections that are helpful for beginning programming.

2 Sections that are helpful for intermediate-level and advanced programming.

3 Sections that are can be read as required regardless of the level of programming.

CONTENTS

CONTENTS

CONTENTS

APPENDIXES

Introduction and Precautions

This chapter gives precautions and warnings concerning the use of this product and the manual. **You must read this chapter before reading the rest of the manual or using the product.**

I.1 Overview of Manual

- This manual describes the installation and operation of the Motion Module MC20. Read this manual carefully to ensure the proper use of the MEMOBUS. Also, keep this manual in a safe place so that it can be used whenever necessary.

- Refer to the following manuals for related information.

Name	Document number	Contents
Motion Module MC20 Hardware User's Manual	SIEZ-C825-20.51	Explains the startup and maintenance of the MC20 Module system. Together with this manual, it provides the information required for understanding the MC20 Module.
MEMOCON GL120, GL130 P120 Programming Panel (MEMOSOFT) User's Manual	SIEZ-C825-60.7	Describes the functions, specifications, and operating methods of the P120 Programming Panel with the MEMOSOFT built in.
MEMOCON GL120, GL130 MEMOSOFT for DOS User's Manual	SIEZ-C825-60.10	Explains programming for the GL120 and GL130 using the MEMOSOFT programming software on a DOS computer.
MEMOCON GL120, GL130 Hardware User's Manual	SIEZ-C825-20.1	Describes the models, specifications, and functions of the devices that make up a GL120 or GL130 system configuration, and explains the startup and maintenance of the GL120 or GL130 system.
MEMOCON GL120, GL130 Software User's Manual (Vol. 1)	SIEZ-C825-20.11	Explains the basic instructions and principles of GL120 or GL130 system operation.
MEMOCON GL120, GL130 Software User's Manual (Vol. 2)	SIEZ-C825-20.12	Explains the expansion instructions and the floating-point instructions of the GL120 or GL130 system.
MEMOCON GL120, GL130 Teach Pendant TB120 User's Manual	SIEZ-C825-60.3	Describes the operating methods of the Teach Pendant for on-site operation of MC20 Modules.

Document Title	Document Number	Content
AC Servo Drive M, F, G, S and D Series User's Manual: Technical Sheets (Incremental Encoders)	TSE-S800-11.1	Describes the functions, specifications, and handling methods for CACR-SR□□BE Servopacks with Incremental Encoders.
AC Servo Drive M, F, G, S and D Series User's Manual: Technical Sheets (Absolute Encoders)	TSE-S800-11.2	Describes the functions, specifications, and handling methods for CACR-SR□□BY Servopacks with Absolute Encoders.
Σ Series SGM□/SGDA User's Manual	TSE-S800-15	Describes the functions, specifications, and handling methods for SGDA-□□□S Servopacks with Incremental or Absolute Encoders.
Σ Series SGM□/SGDB User's Manual	TSE-S800-16	Describes the functions, specifications, and handling methods for SGDB-□□ Servopacks with Incremental or Absolute Encoders.
Σ Series SGM□/DR2 User's Manual	TSE-S800-17	Describes the functions, specifications, and handling methods for DR2-□□ Servopacks.

- First-time users of the Motion Module MC20 should first read chapter 1 of the *Motion Module MC20 Hardware User's Manual.* This chapter provides a relatively quick understanding of the basics.

- Thoroughly check the specifications and conditions or restrictions of the product before use.

- This manual also provides information on version B08 functions. Systems that support version B08 functions are listed below. Information related to step-2 functions is referred to by notes in the text or is indicated by the following icon.

  Ver. B08

Ver. B08

- **Systems Supporting Version B08 Functions**
  The following Modules used with MEMOCON GL120 and GL130 Programmable Controllers and with the version numbers listed in the table support step-2 functions.

Module	Name	Model	Version Numbers Enabling Version B08 Functions	Location of Version Number
CPU Module (8 kW)	CPU10	DDSCR-120CPU14200	□□A01 and later	Nameplate *1
CPU Module (16 kW)	CPU20	DDSCR-120CPU34100	□□B05 and later	Nameplate *1
CPU Module (16 kW)	CPU21	DDSCR-120CPU34110	□□A02 and later	Nameplate *1
CPU Module (32 kW)	CPU30	DDSCR-120CPU54100	□□B05 and later	Nameplate *1
Four-axis Motion Module	MC20	JAMSC-120MMB10400	□□B08 and later	Nameplate *1
MEMOSOFT		FMSGL-DV3 (MEMOSOFT for DOS)	1.40□ and later*2	Displayed at the center bottom of the MEMOSOFT startup screen.
		FMSGL-PP3 (MEMOSOFT for P120 Programming Panel)		
		FMSGL-PP3E (MEMOSOFT for P120 Programming Panel)		

Note (1) The nameplates are on the right side of the Modules.

(2) Override functions cannot be set using version 1.40 of the MEMOSOFT. Use the PRM instruction from the ladder program to set override functions.

- This manual also provides information on the new step-2 functions. Systems that support step-2 functions are listed below. Information related to step-2 functions is referred to by notes in the text or is indicated by the following icon.

STEP2

STEP2

- **Systems Supporting Step-2 Functions**
The following Modules used with MEMOCON GL120 and GL130 Programmable Controllers and with the version numbers listed in the table support step-2 functions.

Module	Name	Model	Version Numbers Enabling Step-2 Functions	Location of Version Number
CPU Module (8 kW)	CPU10	DDSCR-120CPU14200	□□A01 and later	Nameplate *1
CPU Module (16 kW)	CPU20	DDSCR-120CPU34100	□□A08 and later	Nameplate *1
CPU Module (16 kW)	CPU21	DDSCR-120CPU34110	□□A02 and later	Nameplate *1
CPU Module (32 kW)	CPU30	DDSCR-120CPU54100	□□A07 and later	Nameplate *1
Four-axis Motion Module	MC20	JAMSC-120MMB10400	□□B01 and later	Nameplate *1
MEMOSOFT		FMSGL-DV3 (MEMOSOFT for DOS)	1.30□ and later	Displayed at the center bottom of the MEMOSOFT startup screen.
		FMSGL-PP3 (MEMOSOFT for P120 Programming Panel)		

Note (1) The nameplates are on the right side of the Modules.

I.2 Safety Precautions

This section outlines general precautions that apply to using this manual and the product. **You must read this section first before reading the remainder of the manual.**

I.2.1 General Precautions

The functions and performance of industrial machinery will not be determined by the MC20 Motion Module alone. The final control system will depend on a combination of mechanical systems, servo drivers, and so on, so be sure that you have carefully read the manuals provided by the manufacturers of all the equipment before attempting to operate it.

I.2.2 Application Precautions

POSITIONING (MOV)

Unlike normal linear interpolation, the path of movement based on the POSITIONING (MOV) command is often not a straight line. When programming, be absolutely sure to check the path to make sure that there are no tools or other obstacles in the way of the workpiece. Failure to do this may result in damage to equipment, serious personal injury, or even death.

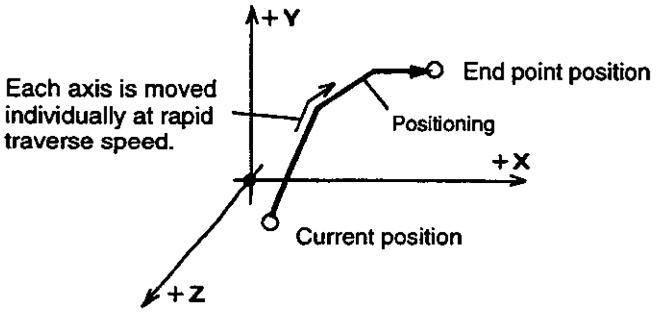


Figure 1.1 Basic Path of Movement for MOV

LINEAR INTERPOLATION (MVS)

Linear interpolation can be executed for either linear or rotary axes. If rotary axes are included, however, the path of movement will not be in a straight line. When programming, be absolutely sure to check the path to make sure that there are no tools or other obstacles in the way of the workpiece. Failure to do this may result in damage to equipment, serious personal injury, or even death.

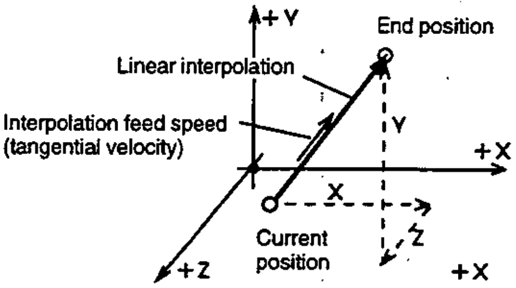


Figure 1.2 Basic Path of Movement for MVS

HELICAL INTERPOLATION (MCW, MCC)

The HELICAL INTERPOLATION command can be executed for linear interpolation for either linear or rotary axes. The path of movement will be a helical line if the axis movement in the linear interpolation portion is correct. When programming, be absolutely sure to check the path to make sure that there are no tools or other obstacles in the way of the workpiece. Failure to do this may result in damage to equipment, serious personal injury, or even death.

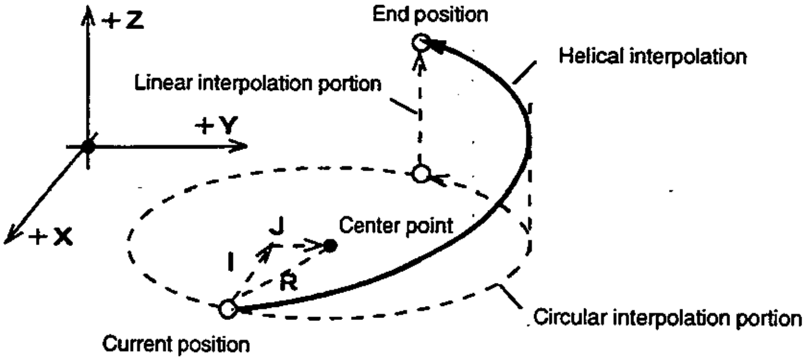


Figure 1.3 Basic Path of Movement for MCW and MCC

ABSOLUTE (ABS) and INCREMENTAL PROGRAMMING MODE (INC)

The ABSOLUTE PROGRAMMING MODE (ABS) command causes the coordinate words in all subsequently designated axis control commands to be treated as absolute values and the INCREMENTAL PROGRAMMING MODE (INC) command causes the coordinate words in all subsequently designated axis control commands to be treated as incremental values. The meaning of a coordinate word treated as an absolute value is entirely different from that of the same word treated as an incremental value. Before operating the MC20 Module with a program, be absolutely sure to check the program to make sure that these commands are used properly. Failure to do this may result in damage to equipment, serious personal injury, or even death.

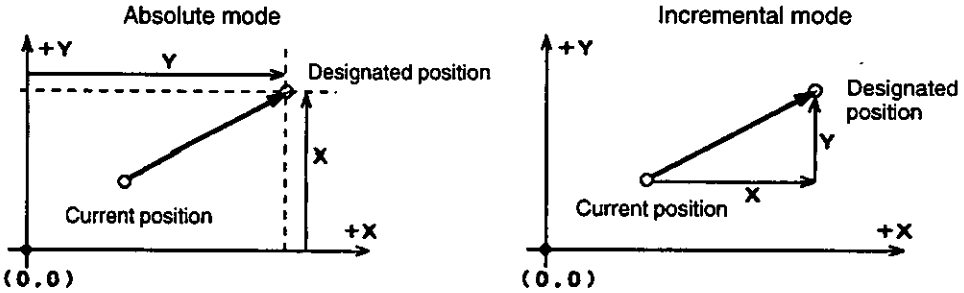


Figure 1.4 Coordinate Words Treated as Absolute or Incremental Values

CURRENT POSITION SET (POS)

The CURRENT POSITION SET (POS) command is used to create a new coordinate system called a "workpiece coordinate system." All move commands designated after this command are carried out in the workpiece coordinate system. Before operating the MC20 Module with a program, be absolutely sure to check the program to make sure that this command is used properly. Failure to do this may result in damage to equipment, serious personal injury, or even death.

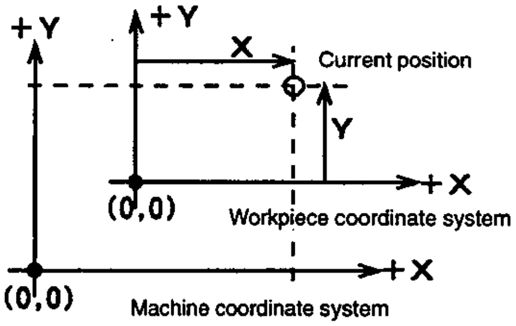


Figure 1.5 Workpiece Coordinate System Created with POS

MOVE ON MACHINE COORDINATES (MVM)

The MOVE ON MACHINE COORDINATES (MVM) command is used to temporarily move axes in a machine coordinate system. Before executing this command, be absolutely sure to check the home position of the machine coordinate system to make sure that the position designated with this command is correct. Failure to do this may result in damage to equipment, serious personal injury, or even death.

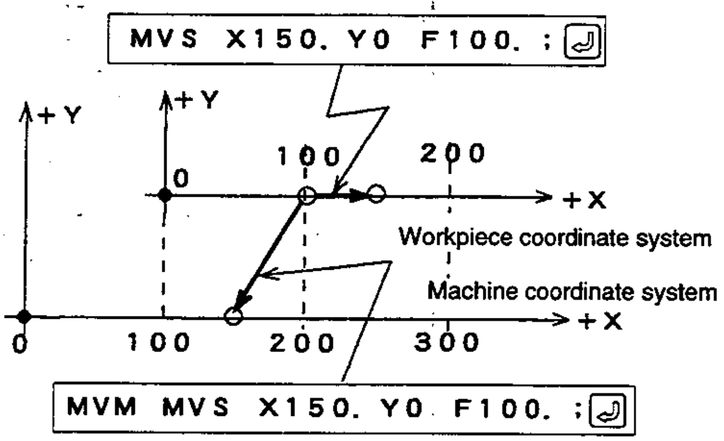


Figure 1.6 Axis Movement on Machine Coordinate System with MVM

COORDINATE SETTING (POS)

The COORDINATE SETTING (POS) command is used to create a new workpiece coordinate system. Before executing this command, be absolutely sure to check the workpiece coordinate system to make sure that the designation of the workpiece coordinate system is correct. Failure to do this may result in damage to equipment, serious personal injury, or even death.

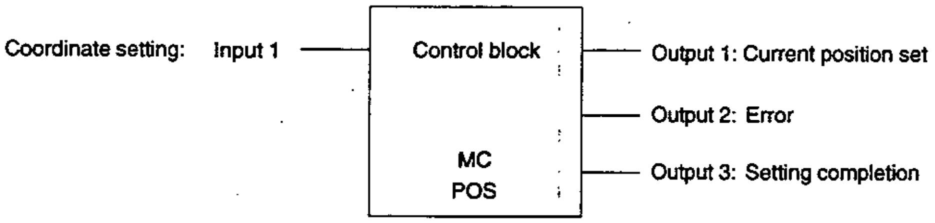

Coordinate setting: Input 1
Control block
MC
POS
Output 1: Current position set
Output 2: Error
Output 3: Setting completion

Figure 1.7 COORDINATE SETTING (POS) Command

HOME POSITION SETTING (ZST)

This command is used to set the machine coordinate home position of an absolute positioning system. Before executing this command, be absolutely sure to check the machine coordinate home position to make sure that the designation of the machine coordinate home position is correct. Failure to do this may result in damage to equipment, serious personal injury, or even death.

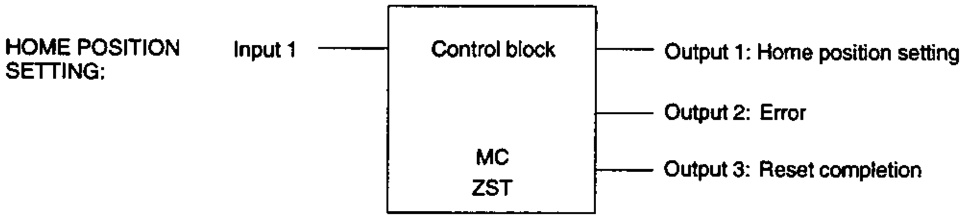
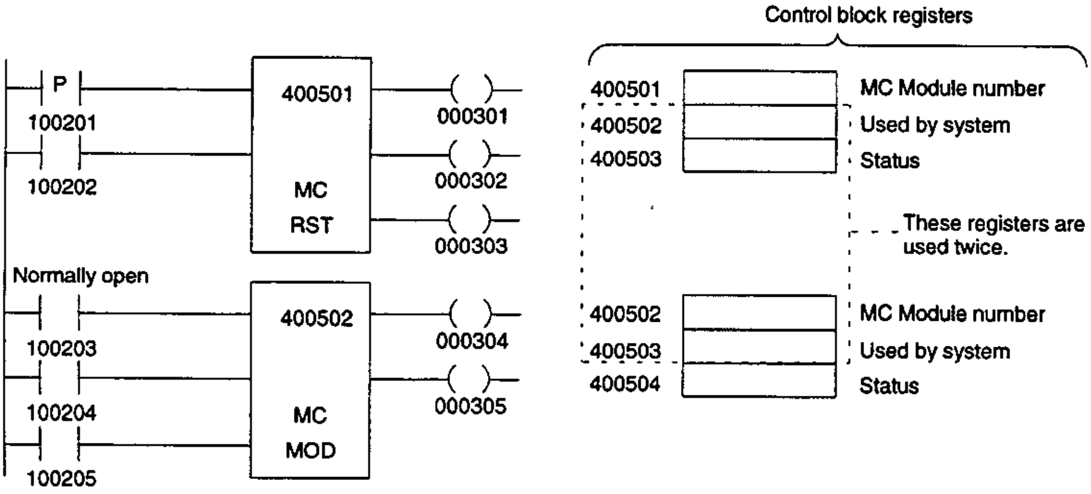


Figure 1.8 HOME POSITION SETTING (ZST) Instruction

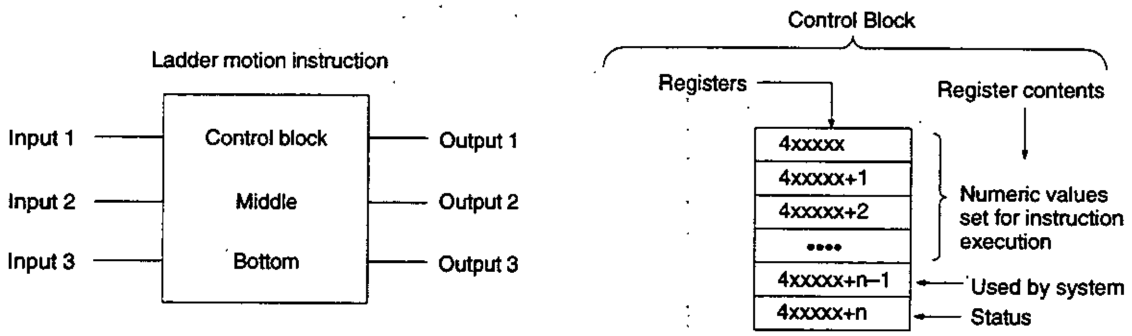
⚠**Caution** Do not use the registers in control blocks for ladder motion instructions for more than one purpose (i.e., more than one instruction). The ladder motion instructions may not function properly if one or more registers in the control block for one ladder motion instruction are also used in the control block for another ladder motion instruction. Never use even one register in a control block for more than one ladder motion instruction.

Example: Incorrect Application



⚠**Caution** Never write data to the control block for a ladder motion instruction that is being executed in the program. If another part of the ladder logic program writes data to the control block of a ladder motion instruction being executed in the program, execution of the ladder motion instruction will be force-ended. Never write data to the control block for a ladder motion instruction that is being executed. (You can read data from the control block from other portions of the ladder logic program without any problem.)

⚠**Caution** Do not write data to the registers used by the system even if the instructions for which the registers are used are not currently being executed.



If data is written to the register specified "used by system" from other parts of the ladder logic program, the above ladder motion instruction will not operate and no alarms will be given.

I.3 Using this Manual

The sections of this manual are classified into three categories, as shown below, and the classifications are indicated in the *Outline Table of Contents*. These categories can be used as guides to which sections are relevant to the user.

$\boxed{1}$ Sections that are helpful for beginning programming.

$\boxed{2}$ Sections that are helpful for intermediate-level and advanced programming.

$\boxed{3}$ Sections that can be read as required regardless of the level of programming.

- **Meaning of Basic Terms**

In this manual, the following terms indicate the meanings as described below, unless otherwise specified.

- **PLC = Programmable (Logic) Controller**

- **CPU = CPU Module**

- **; CR LF = A character string that indicates the end of a block. In this manual this is also sometimes indicated by "; ↵ ".**

Creating Motion Programs

1

This chapter explains how to create motion programs and describes each of the motion commands that can be used. Motion programs are stored in the MC20 Module and are executed to control its operation.

1.1 Programming Methods

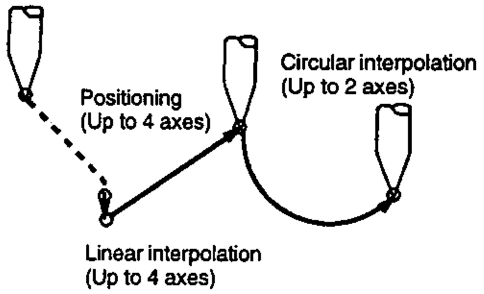
This section describes the basic rules for creating motion programs. Read through this section before attempting to write a program.

1.1.1 Motion Control Capabilities

One or two MC20 Modules can be used to program the specific motions required for industrial equipment. The following examples illustrate the main types of motion that can be programmed.

One MC20 Module

◀EXAMPLE▶



Basic motions, such as rapid traverse positioning, linear interpolation, and circular interpolation, can be easily programmed.

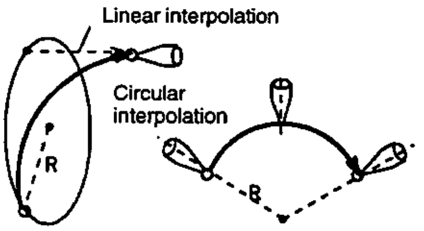
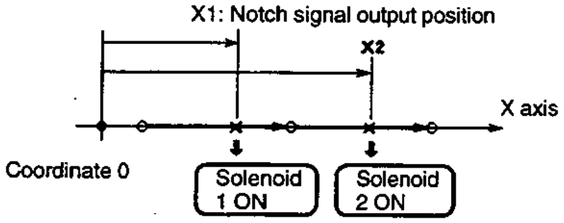


Fig. A Helical Interpolation

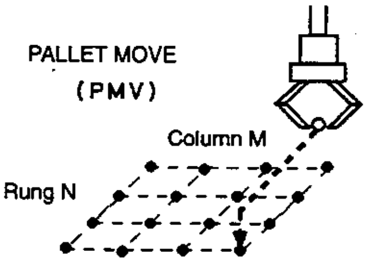
Fig. B Normal Line Control

- Helical interpolation can be programmed to combine linear and circular interpolation (Fig. A).
- Helical interpolation can also be used by applying the linear interpolation portion to the rotary axis to trace an arc using **normal line control** (Fig B).

Using PASS NOTCH SIGNAL OUTPUT (PNT)

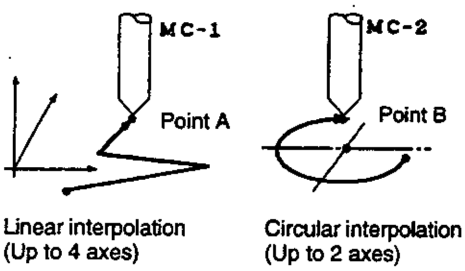


The M code is output when the specified position (i.e., the notch signal output position) is passed during axis movement by a block command. This M code can be used, for example, to turn ON a particular solenoid.



- Positioning can be carried out at rapid traverse speed to a particular grid point position on the pallet. The pallet number and grid point data are set in memory in advance.
- Palletizing and depalletizing operations can be easily programmed.

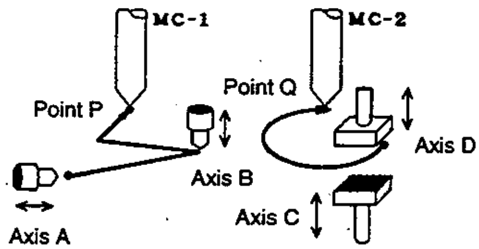
◀EXAMPLE▶ **Two MC20 Modules**



Linear interpolation (Up to 4 axes)

Circular interpolation (Up to 2 axes)

- Two MC20 Modules are used.
- Independent locus control by separate programs is possible for points A and B.



- Movement to points P and Q is simultaneously controlled from the MC20 Module along two axes.
- Axes A to D are controlled independently from the CPU Module.
- A total of eight axes are controlled all together.

TERMS

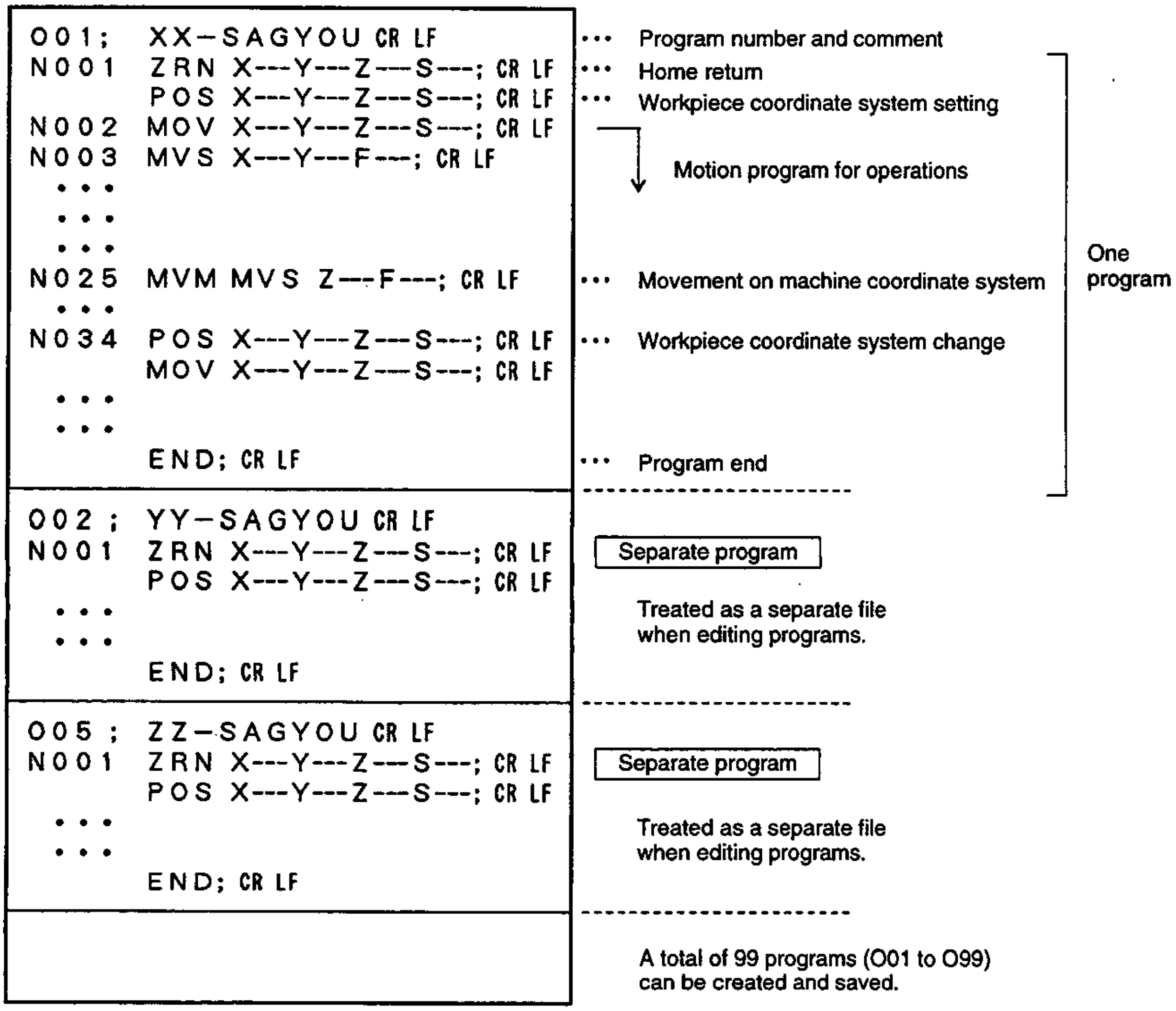
Normal line control

The control of an object such as a paint gun or the tip of a welding torch in the normal line direction with respect to the forward direction is called "normal line control."

1.1.2 Program Outline

1) The memory capacity for storing motion programs in a single MC20 Module is 60 kilobytes.

2) The basic program format for an **incremental position detecting system** is as shown below.



```
O01;  XX-SAGYOU CR LF
N001  ZRN X---Y---Z---S---; CR LF
      POS X---Y---Z---S---; CR LF
N002  MOV X---Y---Z---S---; CR LF
N003  MVS X---Y---F---; CR LF
 • • •
 • • •
 • • •
N025  MVM MVS Z---F---; CR LF
 • • •
N034  POS X---Y---Z---S---; CR LF
      MOV X---Y---Z---S---; CR LF
 • • •
 • • •
      END; CR LF

O02 ; YY-SAGYOU CR LF
N001  ZRN X---Y---Z---S---; CR LF
      POS X---Y---Z---S---; CR LF
 • • •
 • • •
      END; CR LF

O05 ; ZZ-SAGYOU CR LF
N001  ZRN X---Y---Z---S---; CR LF
      POS X---Y---Z---S---; CR LF
 • • •
 • • •
      END; CR LF
```

··· Program number and comment
··· Home return
··· Workpiece coordinate system setting
Motion program for operations
··· Movement on machine coordinate system
··· Workpiece coordinate system change
··· Program end
One program
Separate program
Treated as a separate file when editing programs.
Separate program
Treated as a separate file when editing programs.
A total of 99 programs (O01 to O99) can be created and saved.

Program memory capacity = 60 kilobytes = 61,440 words (characters)
= approximately 3,000 blocks (when one block = 20 words)

TERMS

Incremental position detecting system

A system in which a HOME RETURN (ZRN) instruction or a home position return operation is executed immediately after power-up and in which a machine coordinate system must be set. This system is possible for a normal servodriver with an incremental encoder. Individual axis parameter PA402 must be set to "0."

3) The basic program format for an **absolute position detecting system** is as shown below.

```
O01;  XX-SAGYOU CR LF                 ··· Program number and comment
N001  ZRN X---Y---Z---S---; CR LF     ··· Move to operation starting point
      POS X---Y---Z---S---; CR LF     ··· Workpiece coordinate system setting
N002  MOV X---Y---Z---S---; CR LF
N003  MVS X---Y---F---; CR LF              Motion program for operations
 •••
 •••
 •••                                                                              One
N025  MVM MVS Z---F---; CR LF         ··· Movement on machine coordinate system       program
 •••
N034  POS X---Y---Z---S---; CR LF     ··· Workpiece coordinate system change
      MOV X---Y---Z---S---; CR LF
 •••
 •••
      END; CR LF                      ··· Program end
-------------------------------------------------------------
O02 ; YY-SAGYOU CR LF
N001  ZRN X---Y---Z---S---; CR LF       Separate program
      POS X---Y---Z---S---; CR LF
 •••                                       Treated as a separate file
 •••                                       when editing programs.
      END; CR LF
-------------------------------------------------------------
O05 ; ZZ-SAGYOU CR LF
N001  ZRN X---Y---Z---S---; CR LF       Separate program
      POS X---Y---Z---S---; CR LF
 •••                                       Treated as a separate file
 •••                                       when editing programs.
      END; CR LF
-------------------------------------------------------------
                                           A total of 99 programs (O01 to O99)
                                           can be created and saved.
```

Program memory capacity = 60 kilobytes = 61,440 words (characters)
= approximately 3,000 blocks (when one block = 20 words)

4) The program number selection and the instruction to start operation are executed from the CPU Module's ladder logic program.

TERMS

Absolute position detecting system

A system in which the encoder's absolute position data is retrieved and the machine coordinate system is set at power-up. There is thus no need to return to the home position before the program is run. A servodriver with an absolute encoder is required for this system. In addition, parameter PA402 must be set to "3" and a home position setting operation must be executed.

1.1.3 Programming Format

1) Motion programs are created in a variable-length block format.

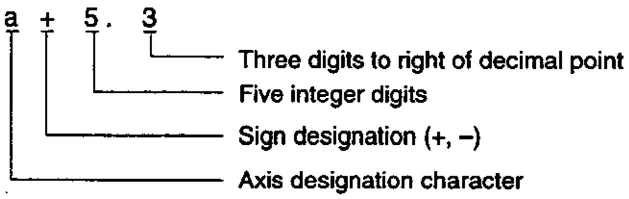
2) The programming format is shown in Table 1.1. The numbers that are entered show the maximum number of digits allowable for the input data.

Table 1.1 Programming Format

Item	One Decimal Digit in Coordinate Word	Two Decimal Digits in Coordinate Word	Three Decimal Digits in Coordinate Word
	Parameter P0005 = 1	Parameter P0005 = 2	Parameter P0005 = 3
Program number	O2	O2	O2
Sequence number	N3	N3	N3
Motion command	Three letters	Three letters	Three letters
Coordinate word (see notes)	a + 7.1	a + 6.2	a + 5.3
Feed per minute	F6	F6	F6
Dwell time	P5.3	P5.3	P5.3
Pallet number	P3	P3	P3
Grid point number	C6	C6	C6
Sub-program number	P2	P2	P2
Number of times	L2	L2	L2
M code output	M2	M2	M2
Interpolation command T designation	T1.3	T1.3	T1.3
Program end	; CR LF	; CR LF	; CR LF

Note (1) "a" represents the axis designation character (X, Y, Z, S, R, I, J, K, L, U, V, W, T, XF, YF, ZF, SF).

(2) An example coordinate word is shown below.


a + 5. 3
Three digits to right of decimal point
Five integer digits
Sign designation (+, −)
Axis designation character

IMPORTANT The explanations from this point on in the manual will be given in terms of three digits to the right of the decimal point (i.e., P0005 = 3).

TERMS

Parameter P0005

One of the sixteen parameters (P0000 to P0015) shared by the controlled axes. This parameter sets the position of the decimal point in coordinate words. The default setting (i.e., the initial value when the power is turned ON) is "3."

3) Leading zeroes can be omitted from numbers following the address, including program numbers and sequence numbers. Plus (+) signs can also be omitted, but minus (–) signs cannot.

Example

X00123 ⇨ X123
X+123 ⇨ X123
X–123 ⇨ X–123

4) The characters that can be used are shown in Table 1.2, along with their meanings.

Table 1.2 Characters

Character	Meaning	Character	Meaning
C	Grid point number	R	Circle radius
E	Point number	S	S-axis coordinate
F	Interpolation feed speed	X	X-axis coordinate
H	H variable	Y	Y-axis coordinate
I	Circle center point X coordinate[*2]	Z	Z-axis coordinate
J	Circle center point Y coordinate[*2]	U	X-axis grid point pitch[*1]
K	Circle center point Z coordinate[*2]	V	Y-axis grid point pitch[*1]
L	Circle center point S coordinate Number of times to repeat sub-program[*2]	W	Z-axis grid point pitch[*1]
M	M code output for wait-for-end external output and notch signal	T	S-axis grid point pitch[*1] Interpolation command T designation
		SF	Helical interpolation linear portion (S axis)[*3]
N	Sequence number	XF	Helical interpolation linear portion (X axis)[*3]
O	Program number	YF	Helical interpolation linear portion (Y axis)[*3]
P	Dwell time, pallet number, sub-program number	ZF	Helical interpolation linear portion (Z axis)[*3]

[*1] This characters are used for the matrix setting command in PALLET SET (PMV).

[*2] I, J, K, and L are also used to designate the number of grids on each axis.

[*3] Coordinate words designated from SF to ZF are always expressed in incremental values. They are not affected by the **ABSOLUTE PROGRAMMING MODE (ABS) and INCREMENTAL PROGRAMMING MODE (INC) commands.**

5) The function characters that can be used are shown in Table 1.3, along with their meanings.

TERMS

ABS and INC commands

When ABSOLUTE PROGRAMMING MODE (ABS) is executed, subsequently designated coordinate words are treated as absolute values. When INCREMENTAL PROGRAMMING MODE (INC) is executed, subsequently designated coordinate words are treated as incremental values.

Table 1.3 Function Characters

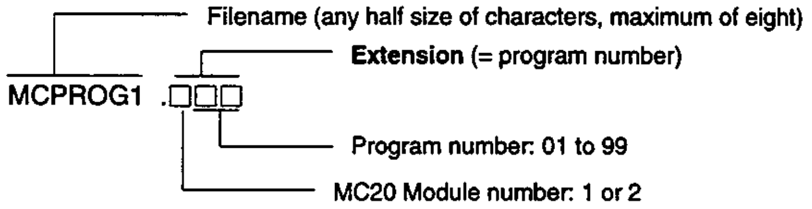
Character	Meaning	Character	Meaning
SP	Space	/	Operator
CR, LF	One block completion*1	.	Decimal point
;	End of block*1	#	Variable
+	Positive sign, operator	*	Operator
–	Negative sign, operator	=	Equal
0 to 9	Number	>	Not equal (greater than)
A to Z	Letter of alphabet	<	Not equal (less than)

*1: A block must be ended by "; CR LF." A comment can be inserted between ";" and "CR LF" as required.

6) Program numbers are handled as described below.

a) Program numbers serve to identify programs. Unlike numbers for normal programs, however, there are no numbers indicated by the character "O." This character is handled as described below.

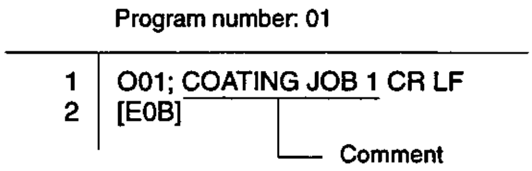
b) When creating a new program in the Programming Device's online edit mode, the filename and the extension are input first. The **extension** includes the program number, as shown in the following example.


Filename (any half size of characters, maximum of eight)
Extension (= program number)
MCPROG1 .□□□
Program number: 01 to 99
MC20 Module number: 1 or 2

In other words, when this new program is created, it is stored in the Programming Device memory under the filename of "MCPROG1.201" (i.e., MC Module 2, program number 01) and the database name of "MCPROG1."

c) First key in the filename and the extension and press the ⏎ key. Then enter a comment between ";" and "CR LF." The comment can contain up to 128 characters/blocks, including program coding. Alphanumeric characters can be used.

Comment Example



```
1   O01; COATING JOB 1 CR LF
2   [E0B]
```

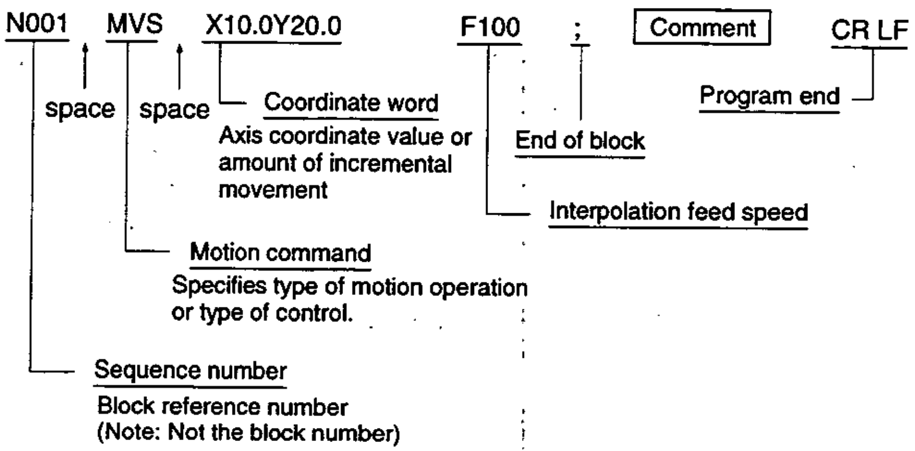
Extension

A computer term which refers to the three characters added at the end of a filename following a period. The extension normally indicates the type of file.

d) It is possible to change the program number of a newly-created program by changing the extension to the desired program number by means of the COPY or RENAME commands at the A:/ MS-DOS command prompt. The filename can also be changed at that time if required. If that is done, the O and number in the first line of the program will not be changed.

7) Create a one-block command as described below.

a) Create a one-block command according to *Table 1.1 Programming Format.* A representative example is shown in the following illustration.


N001
MVS
X10.0Y20.0
F100
;
Comment
CR LF
space
space
Coordinate word
Axis coordinate value or amount of incremental movement
Program end
End of block
Interpolation feed speed
Motion command
Specifies type of motion operation or type of control.
Sequence number
Block reference number (Note: Not the block number)

b) At least one space should be inserted after the sequence number so that the one–block command can be seen easily. The one-block command will operate normally even if there is no space after the sequence number. Do not, however, insert any spaces after the sequence number when editing the program on the screen of the Programming Device.

c) At least one space must be inserted between the motion command and the coordinate word.

d) The sequence number serves as a reference to the block and it can be omitted if not needed. It serves as the destination number, however, when BRANCH (IF...GOTO) is executed. For details, refer to *1.7 Control Commands.*

e) In the Programming Device's offline edit display, one block is written on one or two lines containing no more than 128 half size of characters (i.e., 128 bytes). All characters, including SP, CR, LF, and comments, must be included in the 128-character limit.

f) The "; CR LF" series is required at the end of the block even if no comment has been inserted. (In this manual, "CR LF" is sometimes represented by the ↵ symbol.)

8) Sequence numbers are handled as described below.

a) A sequence number can be entered at the beginning of the block, consisting of three integer digits following the character "N." Leading zeroes can be omitted. Leading zeroes cannot, however, be omitted when editing the program on the screen of the Programming Device

b) Sequence numbers can be entered irrespective of **block numbers** (i.e., edit display line numbers). It is recommended, however, that the sequence numbers and block numbers are related to each other to aid in program management and execution, e.g., so that the program can be executed by designating block numbers from the CPU Module.

c) In general, sequence numbers do not affect the order in which the program is executed. It does not matter whether consecutive numbers, non-consecutive numbers, overlapping numbers, or no sequence numbers at all are used. The one case in which the order of execution is affected is when BRANCH (IF... GOTO) commands are used. For details, refer to *1.7.1 BRANCH (IF... GOTO).*

Note An error will be generated if a sequence number is longer than three digits.

9) Multiblock prereading and preread parallel execution are handled as described below.

a) **Multiblock Prereading**
Normally several blocks are read in advance to control operation when programmed operation is being performed.

b) **Preread Parallel Execution**
When an axis is travelling as the result of an axis move command (i.e., while pulses are being distributed), and when the next block after the one that was preread is not an axis move command, that block will be executed and the preread operation will be repeated for the next block. (This applies only when **IN-POSITION CHECK (PFN)** is not in effect.) This function is called preread parallel execution, and it can be executed in parallel with axis move commands, arithmetic commands, and so on. This is a special feature of this device. It prevents interrupting axis operation during operations such as execution of arithmetic commands.

TERMS

Block numbers

The line numbers in a program edit display are called "block numbers." The line which contains the program number is block no. 1, and the next line is block no. 2. Block numbers are created automatically when the program is edited.

IN-POSITION CHECK (PFN)

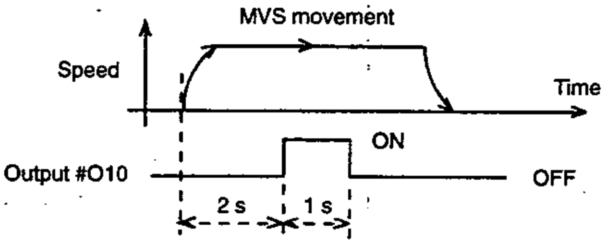
When PFN is executed after an axis move command, program execution will proceed to the next block after detecting that the position completion range has been entered. Therefore, pre-read parallel processing is temporarily interrupted when PFN is executed.

◀EXAMPLE▶ **Preread Parallel Execution Program Example**

MVS X200. Y300. F150. ; ↵ Movement using linear interpolation.

(The following commands are executed in parallel with the above linear interpolation.)

TIM P2.0 ; ↵	Timer, 2 seconds
#O10=1 ; ↵	Turns output #O10 ON.
TIM P1.0 ; ↵	Timer, 1 second
#O10=0; ↵	Turns output #O10 OFF.



1.1.4 Axis Control

Axis Numbers and Axis Designation Characters

As shown in Table 1.4, **Parameter numbers** P0001 to P0004 can by set from the Programming Device to link the axis numbers (axis 1 to axis 4) corresponding to the MC20 Module hardware and the axis designation characters (X, Y, Z, etc.) that are used to set axes in the motion program.

Table 1.4 Axis Numbers and Axis Designation Characters

Axis Number	Parameter Number	Axes Designation Characters
Axis 1	P0001	Any of X, Y, Z, S, A, B, C, D, or –
Axis 2	P0002	Any of X, Y, Z, S, A, B, C, D, or –
Axis 3	P0003	Any of X, Y, Z, S, A, B, C, D, or –
Axis 4	P0004	Any of X, Y, Z, S, A, B, C, D, or –
	Classification	X, Y, Z, S: Axes that can be designated from the motion program A, B, C, D: Independent axes*1

*1: Independent axes can only be designated from the ladder logic program.

Note (1) The same axis designation character can not be set for more than one parameter.

(2) Set "–" (minus sign) for the axis number of any unused axes.

TERMS

Parameter numbers

Parameter settings are made using the Programming Device.

Coordinate Words

The combination of an axis designation character and the amount of movement (with either a plus or minus sign) is called a "coordinate word." The meanings of the axis designation characters for the coordinate words used in this system are shown in Table 1.5.

Table 1.5 Coordinate Word Meanings

Axis Designation Characters for Coordinate Words		Meaning
Basic axes	X, Y, Z, S	Coordinate value or incremental move distance for X, Y, Z, and S axes.
Auxiliary data for circular or helical interpolation	R	Circular interpolation radius (fixed for incremental value)
	I, J, K, L	Respective X, Y, Z, and S axes components of center point coordinates for circular interpolation.
	XF, YF, ZF, SF	Linear travel distance of helical interpolation in X, Y, Z, and S axes directions respectively.
Independent axes	A, B, C, D	Coordinate value or incremental move distance for axes that can be designated from the ladder logic program.
PALLET MOVE (PMV) command	U, V, W, T	Grid point pitch in the X, Y, Z, and S axes directions for the PALLET SET (PST) command.

Independent Axes

1) Any or all of an MC20 Module's four axes can be designated as independent axes. The independent axes are called axis A, axis B, axis C, and axis D, and they can be operated independently from the CPU Module. They cannot be operated from the motion program.

2) Independent axis settings can only be made by means of parameter settings (P0001 to P0004). It is not possible to change from independent axes to basic axes, or vice versa, from the CPU Module during operation.

3) The following tables show examples of basic and independent axis settings when using two, three, and four axes.

Example 1: Two Axes

Case	Axis 1	Axis 2	Method of Operation
1	X	Y	Axes X and Y controlled from motion program.
2	A	B	Axes A and B controlled independently from ladder logic program.
3	X	A	Axis X controlled from motion program; axis A controlled independently from ladder logic program.

Example 2: Three Axes

Case	Axis 1	Axis 2	Axis 3	Method of Operation
1	X	Y	Z	Axes X, Y, and Z controlled from motion program.
2	X	Y	A	Axes X and Y controlled from motion program; axis A controlled independently from ladder logic program.
3	X	A	B	Axis X controlled from motion program; axes A and B controlled independently from ladder logic program.

Example 3: Four Axes

Case	Axis 1	Axis 2	Axis 3	Axis 4	Method of Operation
1	X	Y	Z	S	Axes X, Y, Z, and and S controlled from motion program.
2	X	Y	A	B	Axes X and Y controlled from motion program; axes A and B controlled independently from ladder logic program.
3	A	B	C	D	Axes A, B, C, and D controlled independently from ladder logic program.

Number of Simultaneously Controlled Axes

The number of axes that can be controlled simultaneously from the motion program are shown in Table 1.6.

Table 1.6 Number of Simultaneously Controlled Axes

Command	Number of Simultaneous Control Axes	
POSITIONING (MOV)	Four (axes X, Y, Z, and S).	
LINEAR INTERPOLATION (MVS)	Four (axes X, Y, Z, and S).	
CIRCULAR INTERPOLATION (MCW, MCC)	Two on X–Y, Y–Z, Z–X, X–S, Y–S, Z–S planes.	
HELICAL INTERPOLATION (MCW, MCC)	Three, as follows:	
	XY plane circular + axis Z linear	XY plane circular + axis S linear
	YZ plane circular + axis X linear	YZ plane circular + axis S linear
	ZX plane circular + axis Y linear	ZX plane circular + axis S linear
	XS plane circular + axis Z linear	XS plane circular + axis Y linear
	YS plane circular + axis X linear	YS plane circular + axis Z linear
	ZS plane circular + axis X linear	ZS plane circular + axis Y linear

Programming and Movable Resolution

1) The smallest reference unit that can be programmed is set by parameter P0005, as shown in *Table 1.7.* (In this manual, “reference unit” is sometimes referred to as “designated units.”)

Table 1.7 Reference Unit

Parameter Setting	Reference Unit	
	Linear Axis	Rotary Axis
P0005 = 1	0.1 mm	0.1°
P0005 = 2	0.01 mm	0.01°
P0005 = 3	0.001 mm	0.001°

2) The smallest movable resolutions that can be set for machine system movements are shown in Table 1.8.

Table 1.8 Movable Resolution

Linear Axis	Rotary axis
0.1 mm	0.1°
0.01 mm	0.01°
0.001 mm	0.001°

IMPORTANT

In principal, reference unit should be set to match movable resolution. Parameters **PA501** through PA505 can be used to set movable resolution.

3) When programming and movable resolutions are properly set, their relationship is as shown in Table 1.9.

Table 1.9 Relation between Programming and Movable Resolution

Programming Method		Output Side	
Parameter Setting	**Program Example**	**Movable Resolution for Setting**	**Actual Amount of Movement**
P0005 = 1	0.1	0.1 mm 0.1°	0.1 mm 0.1°
P0005 = 2	0.1	0.01 mm 0.01°	0.1 mm 0.1°
P0005 = 3	0.1	0.001 mm 0.001°	0.1 mm 0.1°

Maximum Programmable Values

1) The maximum values for a single move command are shown in Table 1.10.

Table 1.10 Maximum Programmable Values

		Parameter Setting	Linear Axis	Rotary Axis
Finite length		P0005=1	±9999999.9 mm	±9999999.9°
		P0005=2	±999999.99 mm	±999999.99°
		P0005=3	±99999.999 mm	±99999.999°
Infinite length	Absolute mode	P0005=1	±[PA503]–1	±[PA503]–1
		P0005=2	±[PA503]–1	±[PA503]–1
		P0005=3	±[PA503]–1	±[PA503]–1
	Incremental mode	P0005=1	±9999999.9 mm	±9999999.9°
		P0005=2	±999999.99 mm	±999999.99°
		P0005=3	±99999.999 mm	±99999.999°

TERMS

Parameter PA501

When the letter "A" is included in a parameter number, it means that settings can be made individually for each axis. These parameters are called individual axis parameters. When actually making the parameter setting, substitute the axis number (1 to 4) in place of "A."

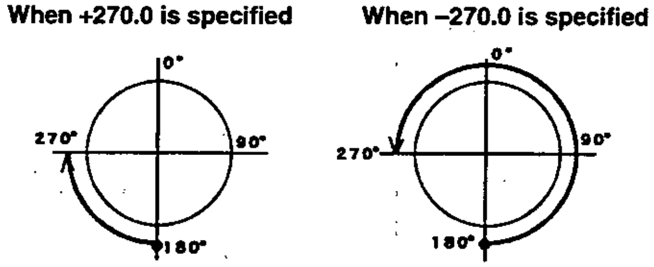
TERMS

±[PA503]–1

This indicates a number which is the value set for PA503 (one machine revolution/reference unit) minus one, with a plus (+) or minus (–) sign attached.

INFO

In the absolute mode for infinite length axis, as shown in the following illustration, the designated sign indicates the direction of revolution, and the designated angle indicates the absolute position. In this example, the position is specified from a current value of 180°.



Note An alarm will be generated if the absolute coordinate designation exceeds the value set for **parameter PA503**.

2) When designating incremental values, be sure that the designated values do not exceed the maximum programmable values shown above. Also, when absolute values are designated, the amount of individual axis movement specified must not exceed the maximum programmable values.

3) Accumulated values for move commands must not exceed the values shown in Table 1.11.

Table 1.11 Accumulated Maximum Command Values

	Parameter Setting	Linear Axis	Rotary Axis
Finite length	P0005=1	±9999999.9 mm	±9999999.9°
	P0005=2	±999999.99 mm	±999999.99°
	P0005=3	±99999.999 mm	±99999.999°
Infinite length		No limit	No limit

Inputting Decimal Points

1) Numbers with decimal points can be used for designating coordinates (coordinate values or amount of movement), speed, or time. The numbers for which decimal points can be used are the numbers following the characters shown below.

Coordinate words:	X, Y, Z, S, I, J, K, L, R, U, V, W, T, XF, YF, ZF, SF
Interpolation feed speed:	F
Time:	P

TERMS

Parameter PA503

This is an individual axis parameter called "one machine revolution/reference unit." In the case of a rotary axis, the value of one revolution (360°) divided into movable resolution is set for this parameter.

2) The number of digits to the right of the decimal point for coordinate words is set by parameter P0005, as shown in Table 1.12. (The selection of linear axis or rotary axis is set by bit 2 of parameter PA506.)

Table 1.12 Reference Unit

Parameter Setting	Reference Unit	
	Linear Axis*1	Rotary Axis*1
P0005 = 1	0.1 mm	0.1°
P0005 = 2	0.01 mm	0.01°
P0005 = 3	0.001 mm	0.001°

*1: The linear or rotary axis designation is specified in bit b2 of parameter PA506.

Note (1) Decimal points are not used in when setting parameters. They are only valid in motion programs.

(2) In this manual explanations are given in terms of PA0005=3 (i.e., with the reference unit set to 0.001 mm) as the standard setting.

3) The following table shows examples of actual amounts of movement for each command when the reference unit is changed by switching the parameter P0005 setting.

◀EXAMPLE▶

Program Command Value Examples		Actual Distance Moved or Coordinate Values	
		P0005=2 (0.01 mm)	P0005=3 (0.001 mm)
(With decimal point)			
1. Axis X	X15.	15.00 mm	15.000 mm
2. Axis Y	Y20.5	20.50 mm	20.500 mm
3. Rotary axis	S20.5	20.5°	20.5°
4. Dwell time	P1.0	1.000 s	1.000 s
5. Feed speed	F25.	25.00 mm/min	25.00 mm/min
(Without decimal point)			
6. Axis X	X15	0.15 mm	0.015 mm
7. Axis Y	Y20	0.20 mm	0.020 mm
8. Rotary axis	S30	0.3°	0.03°
9. Dwell time	P1	0.001 s (fixed)	0.001 s (fixed)
10. Feed speed	F25	25.00 mm/min	25.000 mm/min

1.1.5 Feed Speeds

Rapid Traverse Speed

1) Rapid traverse speed can be used in the following operations: POSITIONING (MOV), JOG (JOG), and STEP (STP).

2) Set the rapid traverse speed independently for each axis with parameter PA202.

Parameter No.	Name	Range	Unit
PA202	Rapid traverse speed	1 to 240,000	mm/min deg/min
PA201	Maximum feed speed	1 to 240,000	mm/min deg/min

Set the maximum allowable speed for each axis with parameter PA201 (maximum feed speed). This setting will automatically be regarded as the upper limit.

3) An **override** can be set for the rapid traverse speed in 16 steps between 0% and 100% or in increments of 0.1% between 0% and 3276.7%. The overrides can also be enabled and disabled.

Overrides for the rapid traverse speed are enabled and disable in parameter P0016.

a) Override: Enabled for **MC control coils** (16 steps)

Ver. B08

b) Override: Enabled for **MC link registers** (in increments of 0.1%)

c) Override: Disabled

TERMS

Override

Use this function to change the set value by the selected percentage.

MC control coils

MC control coils are special signals, with fixed allocations, which control the MC20 Module from the CPU Module. Their reference numbers are Q1001 to Q1160 for MC20 Module 1 and Q2001 to Q2160 for MC20 Module 2.

MC link registers

MC link registers are located in the CPU Module and they can be set from the ladder program. They can also be accessed from the MC20 Module using link I/O variables. Their reference numbers are 409851 to 409914 for MC20 Module 1 and 409924 to 409987 for MC20 Module 2.

Refer to *1.5.7* and *1.5.6* for details.

The overrides are disabled in the default settings.

When an override is enabled, the percentage can be switched as shown below.

MC Control Coils (16 Steps)
0, 1, 2, 4, 6, 8, 10, 20, 30, 40, 50, 60, 60, 80, 90, or 100 (%)

Ver. B08

MC Link Registers (0.1% Increments)
0.0 to 3276.7 (%)

When an override is disabled, the speed will be 100% of the set value or reference value.

Rapid traverse speed overrides are set using the following MC control coils and MC link registers.

Ver. B08

Coil/Register	Use	Reference	
MC control coils 16 steps from 0% to 100%	Rapid traverse speed	QN0137 to QN0140	N is the MC20 Module number (N = 1 or 2)
MC link registers 0.0% to 3276.7% in 0.1% increments	Rapid traverse speed	409909 and 409982	MC Modules 1 and 2 (default allocations)

Set the MC link register to 10 times the override percentage when setting an override in 0.1% increments.

Refer to *3.2 MC Control Coil Functions* for details on MC control coils and to *1.5.6 Link Input Variables (#□□□□)* for details on MC link registers.

Interpolation Feed Speed

1) Feed speeds for **interpolation commands** are specified by six-digit numbers following the letter "F." They are sometimes referred to as "F designations."

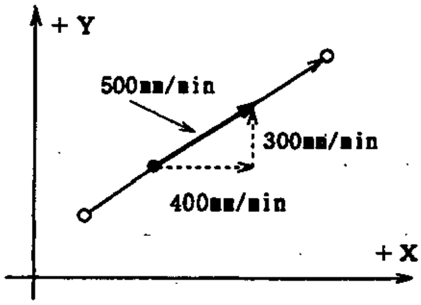
2) Interpolation feed speeds can be designated within a range of F1 to F240000 (mm/min or deg/min), regardless of the parameter P0005 setting.

3) The F designation for linear and circular interpolation specifies the tangential feed speed.

◀EXAMPLE▶ **Example 1: Tangential Velocity for 2-axis Linear Interpolation**

INC MVS X1200 Y900 F500 ; ⏎

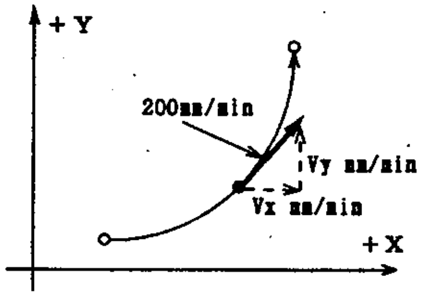
$$F = 500 = \sqrt{400^2 + 300^2} \quad \text{[mm/min]}$$



Example 2: Tangential Velocity for Circular Interpolation

MCC X--- Y--- I--- J--- F200 ; ⏎

$$F = 200 = \sqrt{Vx^2 + Vy^2} \quad \text{[mm/min]}$$



TERMS

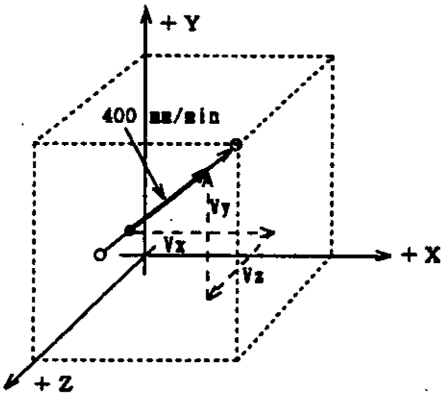
Interpolation commands

The interpolation commands are LINEAR INTERPOLATION (MVS), CIRCULAR INTERPOLATION (MCW, MCC), and HELICAL INTERPOLATION (MCW, MCC).

◀EXAMPLE▶ **Example 3: Tangential Velocity for 3-axis Linear Interpolation**

INC MVS X100 Y100 Z100 F400 ; ↵

$$F = 400 = \sqrt{Vx^2 + Vy^2 + Vz^2} \quad \text{[mm/min]}$$



Example 4: Tangential Velocity for 4-axis Linear Interpolation

MVS X--- Y--- Z--- S--- F600 ; ↵

$$F = 600 = \sqrt{Vx^2 + Vy^2 + Vz^2 + Vs^2} \quad \text{[mm/min]}$$

4) The maximum upper limit that can be set for the feed speed depends on the capacity of the mechanical system and the servo system. The setting is made in parameter P0006. An alarm will be generated (alarm code 039: F designation value exceeded) if an F designation is made which exceeds that limit.

Parameter No.	Name	Range	Unit
P0006	Maximum interpolation feed speed setting	1 to 240,000	mm/min deg/min

5) An override can be set for the maximum interpolation feed speed in 16 steps between 0% and 100% or in increments of 0.1% between 0% and 3276.7%. The overrides can also be enabled and disabled.

Overrides for the rapid traverse speed are enabled and disable in parameter P0016.

a) Override: Enabled for MC control coils (16 steps)

Ver. B08

b) Override: Enabled for MC link registers (in increments of 0.1%)

c) Override: Disabled

The overrides are disabled in the default settings.

When an override is enabled, the percentage can be switched as shown below.

MC Control Coils (16 Steps)
0, 1, 2, 4, 6, 8, 10, 20, 30, 40, 50, 60, 60, 80, 90, or 100 (%)

Ver. B08

MC Link Registers (0.1% Increments)
0.0 to 3276.7 (%)

When an override is disabled, the speed will be 100% of the set value or reference value.

Maximum interpolation feed speed overrides are set using the following MC control coils and MC link registers.

Ver. B08

Coil/Register	Use	Reference	
MC control coil 16 steps from 0% to 100%	Maximum interpolation feed speed	QN0141 to QN0144	N is the MC20 Module number (N = 1 or 2)
MC link registers 0.0% to 3276.7% in 0.1% increments	Maximum interpolation feed speed	409910 and 409983	MC Modules 1 and 2 (default allocations)

Set the MC link register to 10 times the override percentage when setting an override in 0.1% increments.

Refer to *3.2 MC Control Coil Functions* for details on MC control coils and to *1.5.6 Link Input Variables (#□□□□)* for details on MC link registers.

Note Be careful of the following points when using F designations:

d) If a rotary axis is included among the axes operated by an interpolation command, the mechanical velocity will not be equivalent to the tangential velocity specified by the F designation.

e) An error (alarm code 010) will be generated if "F0" is specified as the F designation.

f) Do not specify an F designation with a minus sign (F–□□□). Doing so will cause an alarm to be generated (alarm code 004: address error).

Automatic Acceleration/Deceleration Control

1) Automatic acceleration/deceleration control parameters can be set for operations such as interpolation, positioning, and manual operation. The acceleration/deceleration control combinations possible for the various commands are shown in Table 1.13.

Table 1.13 Automatic Acceleration/Deceleration Control Combinations

Motion Command	Type of Automatic Acceleration/Deceleration					
	Single Step Linear A/D	Double Step Linear A/D	Asymmetric A/D	Exponential A/D	Moving Average A/D	S-curve A/D[*2]
Interpolation (MVS, MCW/MCC)	O	O	O	O	O	O
POSITIONING (MOV)	O	O	O	O	O	O
JOG/STEP (JOG/STP)	O	O	O	O	O	O
HOME RETURN (ZRN)	O	X	X	X	X	O
Independent axis operation (MVA to MVD)	O	O	O	O	O	O

*1: "A/D" stands for "acceleration/deceleration."

*2: S-curve acceleration/deceleration results from using single step and moving average acceleration/deceleration in conjunction.

*3: O: Combinations in which acceleration/deceleration is possible.
X: Combinations in which acceleration/deceleration is not possible.

2) The parameters for setting acceleration/deceleration controls and constants for interpolation are shown in *Table 1.14.*

Table 1.14 Acceleration/Deceleration Control Parameters for Interpolation

Parameter	Name/Meaning
P0006	Maximum interpolation feed speed
P0007	Time constant for linear acceleration/deceleration for interpolation (1)
P0008	Time constant for linear acceleration/deceleration for interpolation (2)
P0009	Linear acceleration/deceleration constant switch speed for interpolation
P0010	Deceleration constant for asymmetric acceleration/deceleration for interpolation
P0011	Time constant for exponential acceleration/deceleration for interpolation
P0012	Bias speed for exponential acceleration/deceleration for interpolation
P0013	Time constant of moving average acceleration/deceleration for interpolation
P0014	Acceleration/deceleration type for interpolation
P0015	Filter selection for interpolation

The above parameters are rather complicated. Refer to the section on parameters for designating speed, acceleration, and deceleration for interpolation in the *Motion Module MC 20 Hardware User's Manual* for details.

3) The parameters for setting acceleration/deceleration controls and constants when not interpolating are shown in *Table 1.15.*

Table 1.15 Automatic Acceleration/Deceleration Control Parameters (No Interpolation)

Parameter	Name/Meaning
PA201	Maximum feed speed
PA202	Rapid traverse speed
PA203	Not used.
PA204	Linear acceleration/deceleration constant (1)
PA205	Linear acceleration/deceleration constant (2)
PA206	Linear acceleration/deceleration constant switch speed
PA207	Deceleration constant for asymmetric acceleration/deceleration
PA209	Time constant for exponential acceleration/deceleration
PA210	Bias speed for exponential acceleration/deceleration
PA211	Time constant for moving average acceleration/deceleration
PA213	Acceleration/deceleration type for positioning (MOV/STP)
PA214	Acceleration/deceleration type for jogging (JOG)
PA215	Acceleration/deceleration type for independent axis operation (MVA to MVD)
PA216	Acceleration/deceleration type for HOME RETURN (ZRN)
PA217	Filter selection

A: Axis numbers (1 to 4)

The above parameters are rather complicated: Refer to the section on parameters for designating speed, acceleration, and deceleration in the *Motion Module MC 20 Hardware User's Manual* for details.

1.1.6 Motion Command Tables

1) Motion commands are specified by three letters, and they specify the task to be performed in that block.

2) Motion commands are divided into modal and non-modal group commands.

Classification	Meaning
Modal group commands	Once a command in this group has been executed, it remains in effect until another command of the same type is executed.
Non-modal group commands	Commands in this group are valid only within the block in which they are executed.

3) There are some commands that can be used together in the same block, and some that cannot. Refer to *Table 1.18 Commands Compatible within the Same Block*.

4) At least one space must be inserted before and after the motion command.

```
N010  MOV  X1000Y2000 ·  ·  ·  ·  ;↵
```

A space must be inserted here.

At least one space should be inserted after the sequence number so that the motion command can be seen easily. The motion command will operate normally even if there is no space after the sequence number. Do not, however, insert any space after the sequence number when editing the program on the screen of the Programming Device.

5) Table 1.16 provides a list of MC20 Module motion commands and brief descriptions of their functions. For more detailed explanations, refer to the relevant sections in this manual. M1 through M6 stand for modal groups 1 through 6, and NM stands for the non-modal group of commands.

Table 1.16 MC20 Module Motion Commands

In the "class" column, M1 through M6 stand for modal groups 1 through 6, and NM stands for the non-modal group of commands.

Command	Name	Class	Command Format	Function/Meaning	Page
MOV	POSITIONING	NM	MOV X–Y–Z–S–; ↵ Target position	Executes simultaneous positioning for maximum four axes at rapid traverse speed.	1-34
MVS	LINEAR INTERPOLATION	NM	MVS X–Y–Z–S–F–T–; ↵ Target position	Executes linear travel at tangential velocity F for maximum four axes simultaneously.	1-37
MCW MCC	CIRCULAR INTERPOLATION CW CIRCULAR INTERPOLATION CCW	NM	MCW X–Y–R–F–; ↵ MCC X–Y–I–J–F–T–; ↵	Execute circular travel at tangential velocity F for two axes simultaneously, following radius R or center-point coordinates I and J.	1-40
MCW MCC	HELICAL INTERPOLATION CW HELICAL INTERPOLATION CCW	NM	MCW X–Y–R–ZF–F–; ↵ MCC X–Y–I–J–Z–F–T–; ↵	Move three axes simultaneously in a combination of circular interpolation and linear interpolation outside of the circular interpolation plane. Speed designation F becomes the circular interpolation tangential velocity.	1-46
PXY PYZ PZX PXS PZS PYS	PLANE XY PLANE YZ PLANE ZX PLANE XS PLANE ZS PLANE YS	M1	MCW PXY X–Y–R–Z–F–; ↵ MCC PXY X–Y–I–J–ZF–F–; ↵	Designate the plane in which circular interpolation is to be executed. Also valid for designating circular interpolation plane in the helical interpolation command.	1-42
		M6	PST PXY X–Y–I–J–U–V–; ↵	Also used for designating the PALLET SET (PST) matrix plane for use in PALLET MOVE (PMV).	1-55
ZRN	HOME RETURN	NM	ZRN X–Y–Z–S–; ↵ Intermediate position	Returns each axis to its home position after positioning to the intermediate position. For the first time after power-up, positioning will return directly to the home position without travelling to the intermediate position.	1-49

1.1.6 Motion Command Tables cont.

Command	Name	Class	Command Format	Function/Meaning	Page
PMV	PALLET MOVE	NM	PMV P– C–; ↵ Grid point number Pallet number	Positions at rapid traverse speed to the position of the grid point number for the specified pallet number. The grid point data must be saved in advance by means of PALLET SET (PST).	1-54
PST	PALLET SET	M5	PST PXY P– X–Y– I–J– U–V–; ↵ Grid point pitch Number of grid points	Saves in memory, for the specified plane, the pallet number and its grid point data. This must be done before PALLET MOVE (PMV) is executed.	1-54
SKP	SKIP	NM	SKP X–Y–Z–S–F–T–; ↵	When the SKIP signal turns ON while the axes are travelling, skips the remaining movement and operation proceeds to the next block. The position at which the SKIP signal turned ON is saved.	1-57
ABS	ABSOLUTE PROGRAMMING MODE	M2	ABS; ↵	Treats all subsequent coordinate words as absolute values.	1-61
INC	INCREMENTAL PROGRAMMING MODE	M2	INC; ↵	Treats all incremental coordinate words as absolute values.	1-63
POS	CURRENT POSITION SET	NM	POS X–Y–Z–S–; ↵ Desired coordinate values	Changes the current position to the desired coordinate values. Subsequent move commands utilize the new coordinate system.	1-64
MVM	MOVE ON MACHINE COORDINATE	NM	MVM MVS X–Y–Z–S– F–; ↵	Goes to the target position on **machine coordinate system.** The coordinate system set automatically at the completion of the home position return is called a machine coordinate system. This coordinate system is not affected by CURRENT POSITION SET (POS).	1-66
TIM	DWELL TIME	NM	TIM P–; ↵	Operation is paused for the amount of time specified by P, and then proceeds to the next block.	1-68
STP	PROGRAM STOP	NM	STP; ↵	Stops the motion program until it is restarted by means of the start operation.	1-69
END	PROGRAM END	NM	END; ↵	Ends the motion program.	1-69
PFN	IN-POSITION CHECK	NM	MVS X–F– PFN; ↵ or PFN; ↵	Proceeds to the next block after the positioning which is commanded in the same or previous block enters the positioning completion range (parameter setting).	1-70

Command	Name	Class	Command Format	Function/Meaning	Page
INP	SECOND IN-POSITION RANGE SETTING	M3	INP $\underbrace{X\text{–}Y\text{–}Z\text{–}S\text{–}}_{\text{Second positioning completion range}}$; ↵	Proceeds to the next block after subsequently commanded interpolations enter second positioning completion range.	1-73

TERMS

Machine coordinate system

- In an incremental position detecting system, this is a coordinate system which is automatically set according to the first home return operation after turning the power ON. That is, it takes the home position (0, 0, 0, 0) as the home return position.

- In an absolute position detecting system, this is a coordinate system that is automatically set by turning the power ON.

- A machine coordinate system is not affected by the CURRENT POSITION SET (POS) command.

Command	Name	Class *1	Command Format	Function/Meaning	Page
SET	SET EXTERNAL OUTPUT	NM	SET M□□ ;↵ M code (01 to 96)	Proceeds to the next block after the M code is output and the MFIN signal from the CPU Module is returned.	1-74
PNT	PASS NOTCH SIGNAL OUTPUT	M4	PNT X–Y–Z–S–M□□;↵ Transit point for each axis MOV X–Y–Z–S;↵	Outputs the specified M code (01 to 96) when the transit points for all axes have been passed during movement in subsequent blocks.	1-77
SNG	IGNORE SINGLE-BLOCK SIGNAL	NM	SNG MOV X–;↵ SNG MVS Z–F–;↵	A block with this command will be operated continuously even in the single-block operation mode.	1-79
IOW	I/O WAIT	NM	IOW (front output variable = O) (condition I/O variable = Δ) (back output variable C = □);↵	After the front output variable "O" has been output, outputs the back output variable "□" when the condition I/O variable reaches the "Δ" state.	1-80
GSB	SUB-PROGRAM CALL	NM	GSB P□□ L□□;↵	Executes the program number specified by P as a sub-program for the number of times specified by L.	1-82
RET	SUB-PROGRAM END	NM	RET;↵	Designates the end of the sub-program.	1-83
PCN *2	PCON SIGNAL OUTPUT	NM	PCN X–Y–Z–S–; ↵	Turns the PCON□ signal output for the specified axis ON or OFF.	1-113
VCC *2	VOLTAGE OUTPUT	NM	VCC X–Y–Z–S–T–; ↵	Outputs an analog voltage as the output for the specified axis.	1-116
EXM *2	EXTERNAL POSITIONING	NM	EXM X–I–U–Y–J–V–; ↵	When the external positioning signal turns ON, the system moves the external positioning travel distance and then ends.	1-134
PGS *2	RATIO OPERATION	NM	PGS X–I–Y–J–Z–K– MS□;↵	Operates the slave axis for a travel distance equivalent to the travel distance of the master axis multiplied by a ratio.	1-139
PGR *2	RATIO OPERATION CANCEL	NM	PGR;↵ or PGR X0 Y0; ↵	Cancels ratio operation for all axes. Cancels ratio operation for the specified axis or axes.	1-139
TSS *2	TRAILING SYNCHRONOUS OPERATION	NM	TSS a–P– MS□;↵ X, Y, Z or Z	After synchronizing the trailing axis with the master axis, switches to trailing synchronous mode.	1-151
TSR *2	TRAILING SYNCHRONOUS OPERATION CANCEL	NM	TSR; ↵	Cancels the trailing synchronous operation.	1-151

*1: M1 to M6: Commands for modal groups 1 to 6
NM: Commands for non–modal groups

*2: Step-2 functions

Command	Name	Command Format	Function/Meaning	Page
	Variables			
#□□□	Common variable	#1 to #199	• Used for general purposes.	1-85
#I□□□	Input variable	#I1 to #1256	• Reads **MC coil** status.	1-85
#O□□□	Output variable	#O1 to #O256	• Outputs **MC relay** status.	1-86
#□□□□	System variable	#1001 to #1018	• Reads system variables such as current position and saved SKIP position.	1-87
#□□□□	Link input variable	#1101 to #1116	• Reads **MC link register** value.	1-93
#□□□□	Link output variable	#1201 to #1216	• Transmits values to MC link register.	1-97
H□	H variable	H1 to H8	• Used for positions and speeds.	1-99
	Arithmetic Commands[*1]			
=	DEFINE	#i = 100, #j = #i	Numeric range: 0 to ±99,999,999 Integers only. Digits to the right of the decimal point are discarded. Calculations are carried out from left to right with no order of priority.	1-100
+	ADD	#i = #j + #k		1-100
–	SUBTRACT	#i = #j – #k		1-100
*	MULTIPLY	#i = #j * #k		1-101
/	DIVIDE	#i = #j / #k		1-101
	COMBINE	#i = #j – #k / #m		1-101
	Control Commands			
IF GOTO	BRANCH	IF <condition> GOTO n;↵	• When the condition is realized, the program jumps to block n.	1-102
WHILE DO DEND	REPEAT	WHILE <condition> DOm;↵ • • • DEND m;↵	• While the condition is in effect, the block from DO m to DEND m is repeated.	1-103
#E□□□	POINT TABLE POSITION	MOV –#E–;↵ MOV X#E–Y#E–F–;↵	A point table for storing the position data for the four axes is created. It is then possible to move to a given position by specifying a point number along with the motion command.	1-106

[*1]: If arithmetic commands (+, –, *, /) are used in the context of the I/O WAIT (IOW) command, they will become logical operation commands (OR, NOR, AND, NAND).

TERMS

MC coil

An internal coil that can be used for general purposes in the CPU Module. For details, refer to *1.5.3 Input Variables.*

MC relay

An internal relay that can be used for general purposes in the CPU Module. For details, refer to *1.5.4 Output Variables.*

MC link register

A group of registers at the CPU Module, the values for which can be set by the ladder logic program. These registers can also be accessed from the MC20 Module by using link I/O variables. For details, refer to *1.5.6 Link Input Variables and 1.5.7 Link Output Variables.*

Table 1.17 Characters Usable with Motion Commands

In the "class" column, M1 through M6 stand for modal groups 1 through 6, and NM stands for the non-modal group of commands.

Command	Name	Class	Command Overlap	Characters that Can Be Designated			
				N X Y Z S	R I J K L	XF YF ZF SF	F P C L M T
MOV	POSITIONING	NM	Yes	O O O O O			
MVS	LINEAR INTERPOLATION			O O O O O			O O
MCW	CIRCULAR INTERPOLATION CW			O O O O O	O O O O O		O O
MCC	CIRCULAR INTERPOLATION CCW			O O O O O	O O O O O		O O
MCW	HELICAL INTERPOLATION CW			O O O O O	O O O O O	O O O O	O O
MCC	HELICAL INTERPOLATION CCW			O O O O O	O O O O O	O O O O	O O
PXY PYZ PZX PXS PZS PYS	PLANE XY PLANE YZ PLANE ZX PLANE XS PLANE ZS PLANE YS	M1 M6 (Note 3)	Yes Yes	O O O O O O			
ZRN	HOME RETURN	NM	No	O O O O O			
PMV	PALLET MOVE	NM	No	O			O O
PST	PALLET SET	M5	Yes	O O O O O	O O O O	(U, V, W, T)	O
SKP	SKIP	NM	No	O O O O O			O O
ABS	ABSOLUTE PROGRAMMING MODE	M2	Yes	O			
INC	INCREMENTAL PROGRAMMING MODE			O			
POS	CURRENT POSITION SET	NM	No	O O O O O			
MVM	MOVE ON MACHINE COORDINATES	NM	Yes	O			
TIM	DWELL TIME	NM	No	O			O
STP	PROGRAM STOP	NM	No	O			
END	PROGRAM END	NM	No	O			
PFN	IN-POSITION CHECK	NM	Yes	O			
INP	SECOND IN-POSITION RANGE SETTING	M3	No	O			
SET	SET EXTERNAL OUTPUT	NM	Yes	O			O
PNT	PASS NOTCH SIGNAL OUTPUT	M4	Yes	O O O O O			O
SNG	IGNORE SINGLE-BLOCK SIGNAL	NM	Yes	O			
IOW	I/O WAIT	NM	No	O			
GSB	SUB-PROGRAM CALL	NM	No	O			O O
RET	SUB-PROGRAM END	NM	No	O			
PCN	PCON SIGNAL OUTPUT	NM	No	O O O O O			
VCC	VOLTAGE OUTPUT	NM	No	O O O O O			O
EXM	EXTERNAL POSITIONING	NM	No	O O O O O	O O O O	(U, V, W, T)	
PGS	RATIO OPERATION	NM	No	O O O O O	O O O O	(MS□)	
PGR	RATIO OPERATION CANCEL	NM	No	O O O O O			

Command	Name	Class	Command Overlap	Characters that Can Be Designated			
				N X Y Z S	R I J K L	XF YF ZF SF	F P C L M T
TSS	TRAILING SYNCHRONOUS OPERATION	NM	No	O O O O O		(MS□)	O
TSR	TRAILING SYNCHRONOUS OPERATION CANCEL	NM	No	O			

Note (1) Commands marked with "Yes" in the "Command Overlap" column can be designated in the same block with other commands. Commands in the same group, (e.g., MOV to MCC, PXY to PYS, and ABS to INC) cannot be used together in the same block. For details, refer to *Table 1.18 Commands Compatible Within the Same Block.*

(2) A numeral (0 to 9), H variable (H1 to H8) and common variable (#1 to #199) can be added after the designated character.

(3) The plane designation for CIRCULAR INTERPOLATION is in modal group M1.
The matrix designation for PALLET MOVE is in modal group M6.

Table 1.18 Commands Compatible Within the Same Block

The following table shows which commands can be used together in the same block. An "O" indicates that the two commands can be used together and an "X" indicates that they cannot.

Command	Compatible Commands																																		
	MOV	MVS	MCW	MCC	PXY	PYZ	PZX	PXS	PYS	PZS	ZRN	PMV	PST	SKP	ABS	INC	POS	MVM	TIM	STP	END	PFN	INP	SET	PNT	SNG	IOW	GSB	RET	=+−*/	IF	GOTO	WHILE	DO	DEND
MOV	X	X	X	X	O	O	O	O	O	O	X	X	X	X	O	O	X	O	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
MVS	X	X	X	X	O	O	O	O	O	O	X	X	X	X	O	O	X	O	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
MCW*1	X	X	X	X	O	O	O	O	O	O	X	X	X	X	O	O	X	O	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
MCC*1	X	X	X	X	O	O	O	O	O	O	X	X	X	X	O	O	X	O	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
PXY	O	O	O	O	X	X	X	X	X	X	X	O	O	X	O	O	X	O	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
PYZ	O	O	O	O	X	X	X	X	X	X	X	O	O	X	O	O	X	O	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
PZX	O	O	O	O	X	X	X	X	X	X	X	O	O	X	O	O	X	O	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
PXS	O	O	O	O	X	X	X	X	X	X	X	O	O	X	O	O	X	O	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
PYS	O	O	O	O	X	X	X	X	X	X	X	O	O	X	O	O	X	O	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
PZS	O	O	O	O	X	X	X	X	X	X	X	O	O	X	O	O	X	O	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
ZRN	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
PMV	X	X	X	X	O	O	O	O	O	O	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
PST	X	X	X	X	O	O	O	O	O	O	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
SKP	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
ABS	O	O	O	O	O	O	O	O	O	O	O	O	O	O	X	X	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O
INC	O	O	O	O	O	O	O	O	O	O	O	O	O	O	X	X	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O
POS	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
MVM	O	O	O	O	O	O	O	O	O	O	X	X	X	X	O	O	X	X	X	X	X	O	X	X	O	O	X	X	X	X	X	X	X	X	X
TIM	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
STP	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
END	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
PFN	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	X	O	O	O	O	O	O	O	O	O	O	O	O	O
INP	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
SET	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
PNT	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	O	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
SNG	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	X	O	O	O	O	O	O	O	O	O
IOW	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
GSB	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
RET	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
PCN *2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
VCC *2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
EXM *2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
PGS *2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
PGR *2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	X	X	X	X	O	X	X	X	X	X	X	X	X	X
TSS *2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	X	X	X	X	O	X	X	X	X	X	X	X	X	X
TSR *2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
=, +, −, *, /	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
IF	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	O	X	X	X
GOTO	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	O	X	X	X	X

Command	Compatible Commands																																		
	MOV	MVS	MCW	MCC	PXY	PYZ	PZX	PXS	PYS	PZS	ZRN	PMV	PST	SKP	ABS	INC	POS	MVM	TIM	STP	END	PFN	INP	SET	PNT	SNG	IOW	GSB	RET	= + – * /	IF	GOTO	WHILE	DO	DEND
WHILE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	O	X
DO	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	O	X	X
DEND	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X

*1: The circular and helical interpolation versions of MCW and MCC are compatible with the same commands.

*2: Step-2 functions

*3: Refer to *Table 1.16 MC20 Module Motion Commands* for a list of command names.

1.2 Commands for Axis Movement

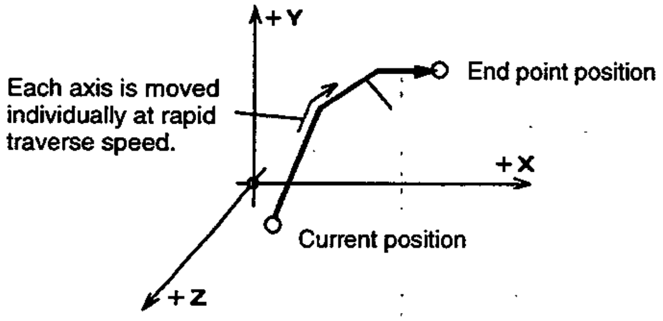
This section explains the programming methods for those motion commands that move axes. This section covers the most basic commands, so be sure to familiarize yourself with it before proceeding further.

1.2.1 POSITIONING (MOV)

1) Specify POSITIONING MOVE (MOV) as follows:

$$\text{MOV}\ \ \underset{\text{End position}}{\underline{X\cdots Y\cdots Z\cdots S\cdots}}\ \ ;\boxed{\hookleftarrow}$$

This command will simultaneously move maximum four axes from the current position to the end position at rapid traverse speed. Any axis for which this command is omitted will not be moved. This axis movement operation is called "positioning." The path of movement based on the POSITIONING (MOV) command is not normally a straight line, as in the linear interpolation described in *1.2.2 LINEAR INTERPOLATION (MVS)*. The following illustration shows an example of movement along three axes (X, Y, and Z).



2) The end point position is determined in advance by ABSOLUTE PROGRAMMING MODE (**ABS**) or INCREMENTAL PROGRAMMING MODE (**INC**).

Mode	End Point Position
ABS (absolute mode)	Positioning to coordinate values (X· ·Y· ·Z· ·S· ·).
INC (incremental mode)	Positioning to incremental distance (X· ·Y· ·Z· ·S· ·) from current position.

⚠ **Caution** The path of movement based on the POSITIONING (MOV) command is often not a straight line, as in linear interpolation. When programming, be absolutely sure to check the path to make sure that there are no tools or other obstacles in the way of the workpiece. Failure to do this may result in damage to equipment, serious personal injury, or even death.

Ver. B08

3) The rapid traverse speeds are set in axis parameters PA202, where A = axis number 1 to 4. The parameter numbers for the four axes are shown below.

Axis Number	Parameter Number of Rapid Traverse Speed
Axis 1	P1202
Axis 2	P2202
Axis 3	P3202
Axis 4	P4202

The actual rapid traverse speed will be the result of the value set for the parameter (PA202, where A = axis number 1 to 4) times the override.

Rapid traverse speed overrides are set using the following MC control coils and MC link registers.

Ver. B08

Coil/Register	Use	Reference	
MC control coil 16 steps from 0% to 100%	Maximum interpolation feed speed	QN0137 to QN0140	N is the MC20 Module number (N = 1 or 2)
MC link registers 0.0% to 3276.7% in 0.1% increments	Maximum interpolation feed speed	409909 and 409982	MC Modules 1 and 2 (default allocations)

Set the MC link register to 10 time the override percentage when setting an override in 0.1% increments.

Refer to *1.1.5 Feed Speeds* for details on the rapid traverse speed..

TERMS

ABS and INC commands

These commands determine whether a coordinate word is to be treated as an absolute value or an incremental value. They are modal commands, so once one has been designated it remains in effect until the other one is designated.

4) An **in-position check** is carried out with respect to the axis movement based on POSITIONING (MOV), to check whether the axis has entered the positioning completion range. The positioning completion range is set by parameter PA103. After the in-position check has been completed, the next move command block is executed. If the following block is not a move command block, that block is executed without waiting for the in-position check. To have execution of a block that is not a move command block delayed until after the in-position check has been completed, designate IN-POSITION CHECK (PFN) in the previous block.

5) Parameter settings can be made to select any of the following types for the automatic acceleration/deceleration control.

   a) Single step linear acceleration/deceleration

   b) Double step linear acceleration/deceleration

   c) Asymmetric acceleration/deceleration

   d) Exponential acceleration/deceleration

   e) Moving average acceleration/deceleration

   f) S-curve acceleration/deceleration

◀EXAMPLE▶ **Program Example (Reference Unit: 0.001 mm)**

ABS ; ↵

MOV X4000. Y3000. Z2000. ; ↵

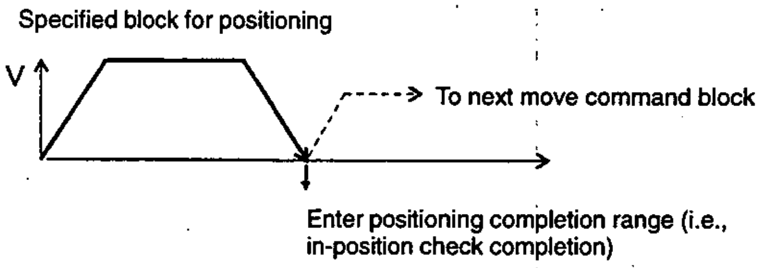
Current position: X = Y = Z = 0

Rapid traverse speed: X = Y = Z = 8 m/min (parameter PA202)

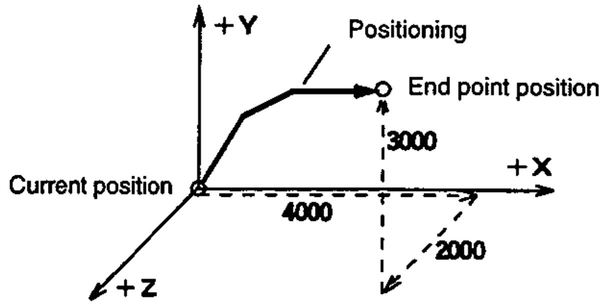
TERMS

In-position check

This function detects whether the actual axis movement has entered the positioning completion range after pulse distribution has been completed for the specified block.


Specified block for positioning
V
To next move command block
Enter positioning completion range (i.e., in-position check completion)

Decimal point setting: Parameter P0005 = 3

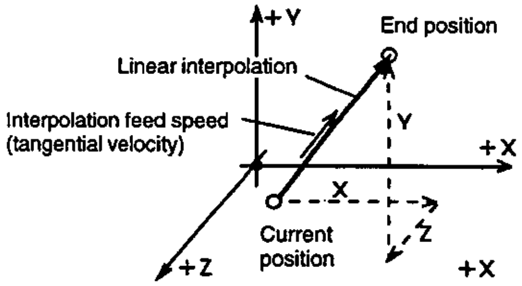


1.2.2 LINEAR INTERPOLATION (MVS)

1) Specify LINEAR INTERPOLATION (MVS) as follows:

```
MVS  X···Y···Z···S···   F···   T···  ;  ⏎
     End position        │     Interpolation command
                         │     T designation
                         └─ Interpolation feed speed
```

This command moves maximum four axes simultaneously from the current position to the end position in a straight line at interpolation feed speed (F). The MVS command will move only the axes that are specified. This axis movement operation is called "linear interpolation." Coordinate words can be designated for any of the four axes X, Y, Z, and S. The following illustration shows an example of operation along three axes (X, Y, and Z).



2) The end position will be determined according to the current mode as set by ABSOLUTE PROGRAMMING MODE (ABS) or INCREMENTAL PROGRAMMING MODE (INC).

Mode	End Position
ABS (Absolute mode)	Positions in a straight line to coordinate values (X..Y..Z..).
INC (Incremental mode)	Positions in a straight line to an incremental distance (X..Y..Z..) from current position.

⚠ **Caution** Linear interpolation can be executed for either linear or rotary axes. If rotary axes are included, however, the path of movement will not be in a straight line. When programming, be absolutely sure to check the path to make sure that there are no tools or other obstacles in the way of the workpiece. Failure to do this may result in damage to equipment, serious personal injury, or even death.

3) The interpolation feed speed is designated by character F, so it is called the "F designation." The F designation specified in the previous block remains in effect, and there is no need to designate it again unless it needs changed. An alarm will be generated (alarm code 010: Undefined F designation) if interpolation is attempted when no F designation has been specified for an interpolation command since the power was turned ON.

4) The tangential feed speeds for all designated axes are related to the F designation value.

When axes X and Y are designated: $F = \sqrt{Vx^2 + Vy^2}$ [mm/min]

When axes X, Y, and Z are designated: $F = \sqrt{Vx^2 + Vy^2 + Vz^2}$ [mm/min]

When axes X, Y, Z, and S are designated: $F = \sqrt{Vx^2 + Vy^2 + Vz^2 + Vs^2}$ [mm/min]

An alarm will be generated (alarm code 039: F designation value exceeded) if an F designation is made which exceeds the limit set in parameter P0006.

Parameter No.	Name	Range	Unit
P0006	Maximum interpolation feed speed setting	1 to 240,000	mm/min deg/min

5) An override of 0% to 200% can be applied with respect to the F designation value in actual program execution. For details, refer to *1.1.5 Feed Speed.*

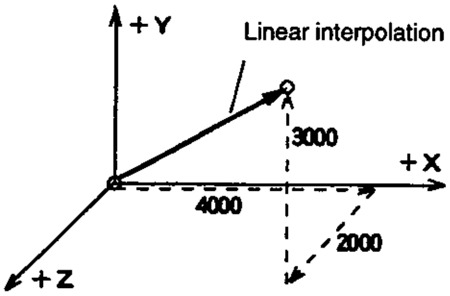
6) The interpolation command T designation makes it possible to program a time constant for acceleration/deceleration for interpolation with the character T if necessary. If the interpolation command T designation is not used or T is set to zero, the time constant for acceleration/deceleration set with parameters P0011 or P0013 will be valid. For details, refer to *1.2.8 Interpolation Command T Designation.*

7) **An in-position check is not executed with respect to axis movement designated by LINEAR INTERPOLATION (MVS).** The next move command block is executed when pulse distribution for the designated block is completed. If the next block is not a move command block, it is executed during interpolation block pulse distribution.
When executing the in-position check, designate IN-POSITION CHECK (PFN) in the previous block. The next block (whether it is a move command or not) is executed after the in-position check has been completed.

Program Example (Unit: 0.001 mm)

```
ABS ; ↵
MVS X4000. Y3000. Z1000. F1000 ; ↵
```

Current position:	X = Y = Z = 0
Interpolation feed speed:	F = 1,000 mm/min
Decimal point setting:	Parameter P0005 = 3



8) Parameter settings can be made to select any of the following types of automatic acceleration/deceleration control.

   a) Single step linear acceleration/deceleration

   b) Double step linear acceleration/deceleration

   c) Asymmetric acceleration/deceleration

   d) Exponential acceleration/deceleration

   e) Moving average acceleration/deceleration

   f) S-curve acceleration/deceleration

   Refer to the section on parameters for designating speed, acceleration, and deceleration for interpolation in the *Motion Module MC 20 Hardware User's Manual* for details.

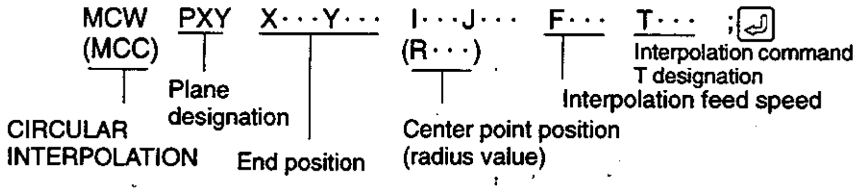
TERMS

In-position check

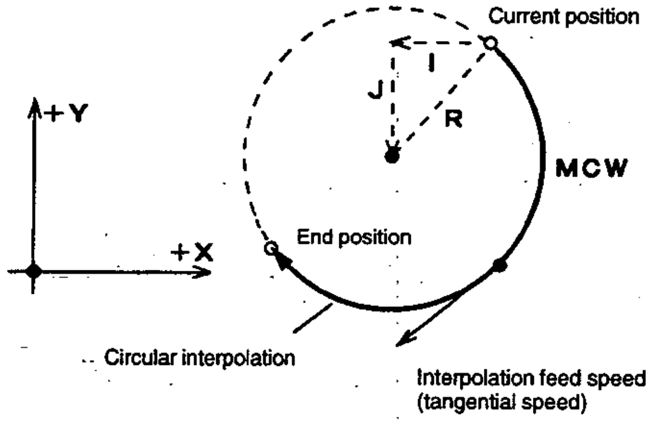
This function detects whether the actual axis movement has entered the positioning completion range after pulse distribution has been completed for the specified block.

1.2.3 CIRCULAR INTERPOLATION (MCW, MCC)

1) Specify CIRCULAR INTERPOLATION (MCW, MCC) as follows:



This command moves two axes simultaneously from the current position to the end position on the specified plane and in a circle using either the center point position or the radius value. The movement is executed at the interpolation feed speed (F). This axis movement operation is called "circular interpolation." The following illustration shows an example.



When using CIRCULAR INTERPOLATION, it is necessary to specify the direction of movement. Specify either MCW for clockwise or MCC for counterclockwise movement.

The end position can be on any of the four axes (X, Y, Z, or S). Specify axes that match the designated plane. For the characters (I, J, K, L), specify the X, Y, Z, and S axis components, respectively.

2) The interpretation of the coordinate words (X, Y, Z, S, I, J, K, L) is determined in advance by ABSOLUTE PROGRAMMING MODE (ABS) or INCREMENTAL PROGRAMMING MODE (INC).

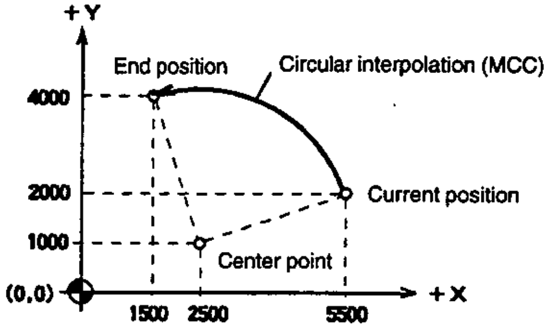
Mode	End Position
ABS (Absolute mode)	Indicates end position coordinate values (X..,Y..) or center point coordinate value (I.., J..).
INC (Incremental mode)	Indicates an incremental amount from current position to end position or center point position.

◀EXAMPLE▶

Coordinate Word Program Examples

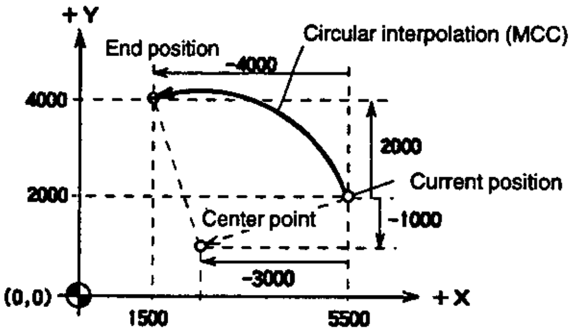
Absolute Mode

```
ABS ; ↵
MCC PXY X1500. Y4000. I2500. J1000. F150 ; ↵
```



Incremental Mode

```
INC ; ↵
MCC PXY X-4000. Y2000. I-3000. J-1000. F150 ; ↵
```

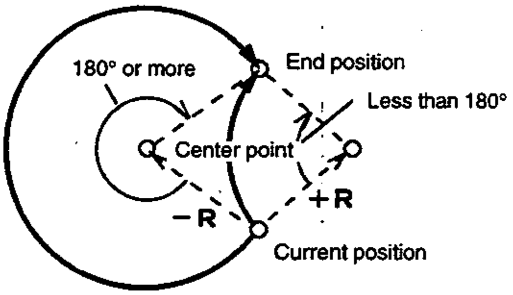


3) Circular interpolation can be specified by means of the circle's radius (R) instead of the center point position (I, J, K, L). The following illustration shows an example.

MCW PXY X...Y...R±...F... ; ↵

When R > 0: Circular interpolation with arc angle of less than 180°.

When R < 0: Circular interpolation with arc angle of 180° or more.



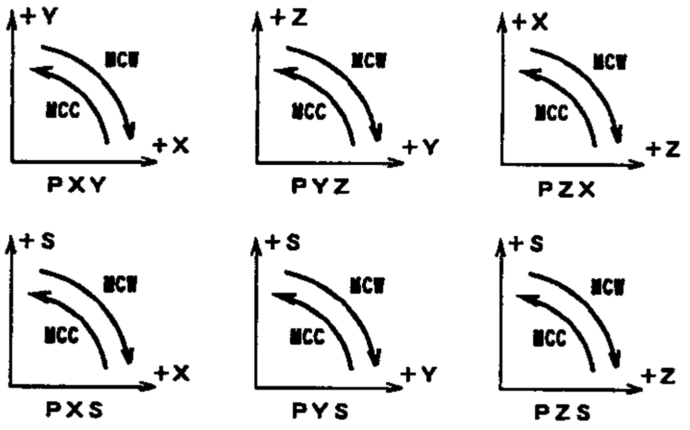
Note An alarm will be generated if R = 0.

4) As a general principle, specify the plane when designating circular interpolation. Table 1.19 shows the designation format for each circular interpolation plane.

Table 1.19 Circular Interpolation Designation Format

Plane	Circular Interpolation	Plane Setting	End Position	Center Point	Feed Speed and T Designation
1 XY plane	MCW (or MCC)	PXY	X Y	I J (R)	F T; ↵
2 ZX plane	MCW (or MCC)	PZX	Z X	K I (R)	F T; ↵
3 YZ plane	MCW (or MCC)	PYZ	Y Z	J K (R)	F T; ↵
4 XS plane	MCW (or MCC)	PXS	X S	I L (R)	F T; ↵
5 YS plane	MCW (or MCC)	PYS	Y, S	J L (R)	F T; ↵
6 ZS plane	MCW (or MCC)	PZS	Z S	K L (R)	F T; ↵

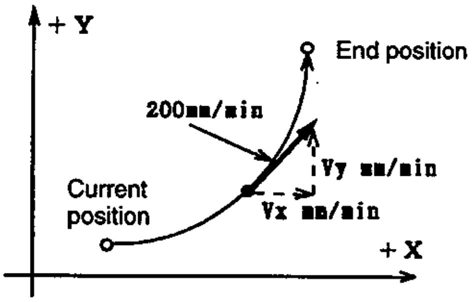
The directions of revolution for each plane are as shown below.



5) The interpolation feed speed is designated by character F, so it is called the "F designation." The F designation specified in the previous block remains in effect, and there is no need to designate it again unless it needs changed. An alarm will be generated (alarm code 010: Undefined F designation) if interpolation is attempted when no F designation has been specified for an interpolation command since the power was turned ON.

6) The tangential feed speed for the specified circle is related to the F designation value.

When axes X and Y are designated: $F = \sqrt{Vx^2 + Vy^2}$ [mm/min]



An alarm will be generated (alarm code 039: F designation value exceeded) if an F designation is made which exceeds the limit set in parameter P0006.

Parameter No.	Name	Range	Unit
P0006	Maximum interpolation feed speed setting	1 to 240,000	mm/min deg/min

7) An override of 0% to 200% can be applied with respect to the F designation value in actual program execution. For details, refer to *1.1.5 Feed Speed.*

8) The interpolation command T designation makes it possible to program a time constant for acceleration/deceleration for interpolation with the character T if necessary. If the interpolation command T designation is not used or T is set to zero, the time constant for acceleration/deceleration set with parameters P0011 or P0013 will be valid. For details, refer to *1.2.8 Interpolation Command T Designation.*

IMPORTANT

9) **An in-position check is not executed with respect to axis movement designated by CIRCULAR INTERPOLATION (MCW, MCC).** The next move command block is executed when pulse distribution for the designated block is completed. If the next block is not a move command block, it is executed during interpolation block pulse distribution. When executing the in-position check, designate IN-POSITION CHECK (PFN) in the previous block. The next block (whether it is a move command or not) is executed after the in-position check has been completed.

10) Parameter settings can be made to select any of the following types of automatic acceleration/deceleration control.

a) Single step linear acceleration/deceleration

b) Double step linear acceleration/deceleration

c) Asymmetric acceleration/deceleration

d) Exponential acceleration/deceleration

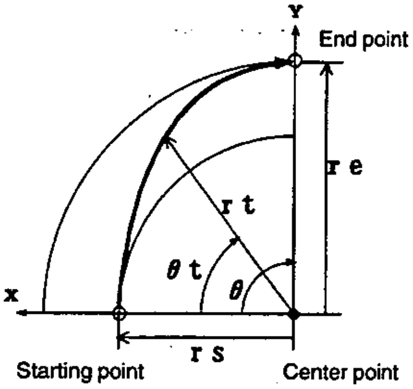
e) Moving average acceleration/deceleration

f) S-curve acceleration/deceleration

Refer to the section on parameters for designating speed, acceleration, and deceleration for interpolation in the *Motion Module MC 20 Hardware User's Manual* for details.

Note (1) An alarm (alarm code 011: No radius designation for circular interpolation) will be generated when zero is specified as the radius in CIRCULAR INTERPOLATION (MCW, MCC). This would occur, for example, if "I=0" and "J=0" were specified in incremental mode for the XY plane.

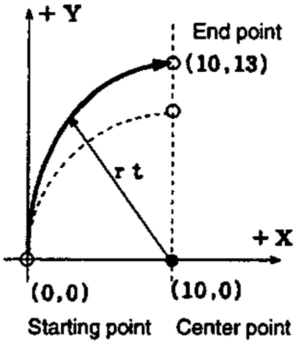
(2) If an end point is specified which deviates from the circumference of the circle, the interpolation will occur as shown in the following illustration.



If the specified end point is not on the specified circle circumference, the interpolation proceeds from the starting point to the end point in a spiraling motion while gradually adjusting the radius.

◀EXAMPLE▶ **Program Example A: When the Circle's Center Point is Designated**

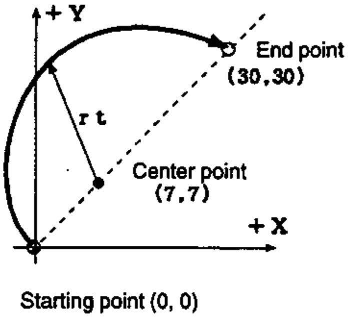
```
N010 ABS ; ↵
N011 MOV X0 Y0 ; ↵
N012 MCW PXY X10. Y13. I10. J0. F100 ; ↵
```



This program executes the interpolation as shown in the illustration.

◀EXAMPLE▶ **Program Example B: When the Circle's Radius (R) is Designated**

```
N010 ABS ; ↵
N011 MOV X0 Y0 ; ↵
N012 MCW PXY X30. Y30. R10. F100 ; ↵
```

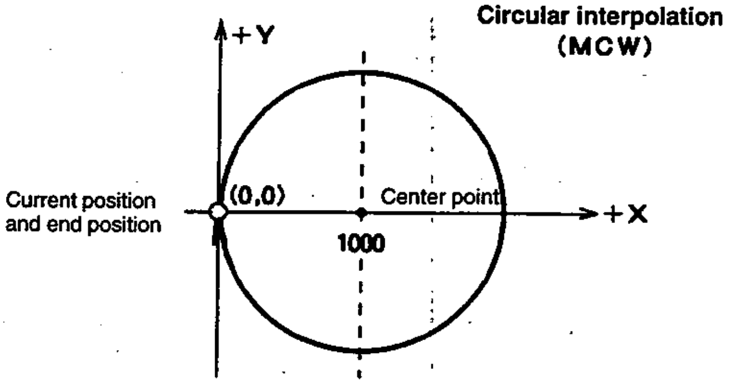


This program executes the interpolation as shown in the illustration.

◀EXAMPLE▶ **Program Example C: When a Full Circle is Designated**

A completely closed circle can be designated in one block by matching the starting point and the end point. It is not possible to designate a full circle by means of the radius (R).

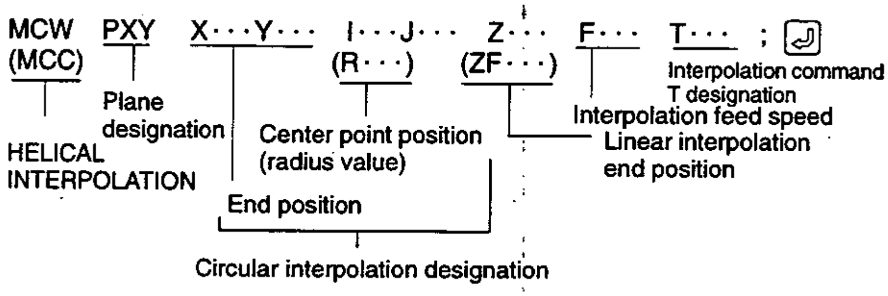
```
ABS ; ⏎
MOV X0 Y0 ; ⏎
MCW PXY X0 Y0 I1000. J0. F100 ; ⏎
```



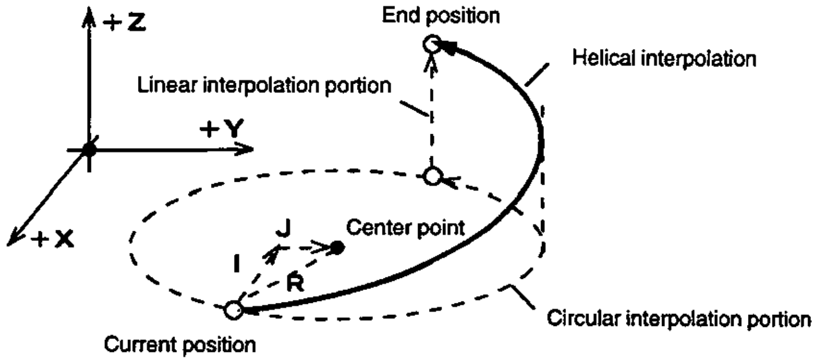
1.2.4 HELICAL INTERPOLATION (MCW, MCC)

1) Helical interpolation is an extension of circular interpolation. Refer to the description of circular interpolation for designations related to circular interpolation. They have been omitted here.

2) Specify HELICAL INTERPOLATION (MCW, MCC) as follows:


MCW PXY X···Y··· I···J··· Z··· F··· T··· ; ⏎
(MCC)
(R···)
(ZF···)
Plane designation
HELICAL INTERPOLATION
Center point position (radius value)
End position
Circular interpolation designation
Linear interpolation end position
Interpolation feed speed
Interpolation command T designation

This command moves two axes simultaneously from the current position to the end position on the specified plane and in a circle using either the center point position or the radius value while executing simultaneous linear interpolation. The movement is executed at the interpolation feed speed (F). This axis movement operation is called "helical interpolation." The following illustration shows an example.



3) An axis not specified for the plane designation can be specified by the linear interpolation axis. The axis does not necessarily have to be at a right angle to the interpolation plane.

Axis Characters for Linear Interpolation Portion	Effect of Absolute or Incremental Mode
X, Y, Z, S	These follow the ABS and INC commands, and can be used for either absolute or incremental values.
XF, YF, ZF, SF	These are not affected by the ABS and INC commands. Always use incremental value.

⚠ **Caution** The HELICAL INTERPOLATION command can be executed for linear interpolation for either linear or rotary axes. The path of movement will be a helical line if the axis movement in the linear interpolation portion is correct. When programming, be absolutely sure to check the path to make sure that there are no tools or other obstacles in the way of the workpiece. Failure to do this may result in damage to equipment, serious personal injury, or even death.

4) Specify the plane when designating helical interpolation. If no plane is specified, the plane previously specified for circular interpolation will remain in effect. Table 1.20 shows the designation format for each interpolation plane.

Table 1.20 Helical Interpolation Designation Format

Plane	Helical Interpolation	Plane Setting	End Position	Center Point	Linear Interpolation	Feed Speed and T designation
1 XY plane	MCW (or MCC)	PXY	X Y	I J (R)	Z (ZF)	F T; ↵
2 XY plane	MCW (or MCC)	PXY	X Y	I J (R)	S (SF)	F T; ↵
3 ZX plane	MCW (or MCC)	PZX	Z X	K I (R)	Y (YF)	F T; ↵
4 ZX plane	MCW (or MCC)	PZX	Z X	K I (R)	S (SF)	F T; ↵
5 YZ plane	MCW (or MCC)	PYZ	Y Z	J K (R)	X (XF)	F T; ↵
6 YZ plane	MCW (or MCC)	PYZ	Y Z	J K (R)	S (SF)	F T; ↵
7 XS plane	MCW (or MCC)	PXS	X S	I L (R)	Y (YF)	F T; ↵
8 XS plane	MCW (or MCC)	PXS	X S	I L (R)	Z (ZF)	F T; ↵
9 YS plane	MCW (or MCC)	PYS	Y S	J L (R)	X (XF)	F T; ↵
10 YS plane	MCW (or MCC)	PYS	Y S	J L (R)	Z (ZF)	F T; ↵
11 ZS plane	MCW (or MCC)	PZS	Z S	K L (R)	X (XF)	F T; ↵
12 ZS plane	MCW (or MCC)	PZS	Z S	K L (R)	Y (YF)	F T; ↵

Note (1) Specify the circular interpolation position within 360°.

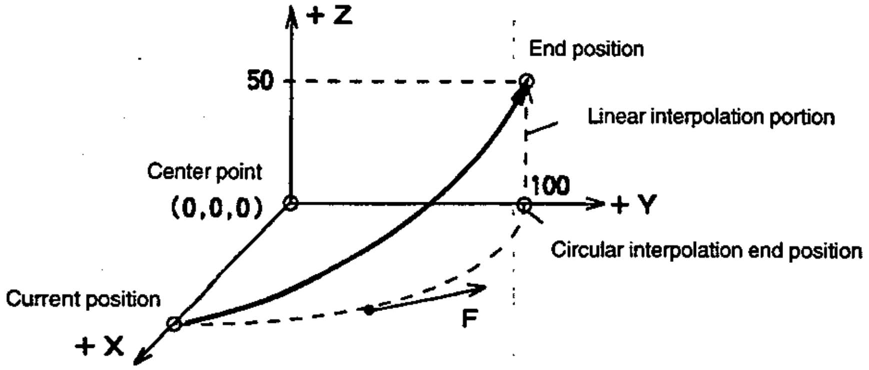
(2) The interpolation feed speed (F) is the tangential speed for the circle on the circular interpolation plane. The feed speed for the linear interpolation portion (F') is found by means of the following equation:

F' = Fx (length of linear interpolation axis) / (length of circular interpolation portion)

Parameter P0006 (maximum interpolation feed speed setting) is valid for the F designation.

◀EXAMPLE▶ **Program Example**

```
ABS ; ↵
MOV X100. Y0 Z0 ; ↵
MCC PXY X0 Y100. I0 J0 Z50. F100. ; ↵
```


+ Z
End position
50
Linear interpolation portion
Center point
(0,0,0)
100
+ Y
Circular interpolation end position
Current position
F
+ X

5) The interpolation command T designation makes it possible to program a time constant for acceleration/deceleration for interpolation with the character T if necessary. If the interpolation command T designation is not used or T is set to zero, the time constant for acceleration/deceleration set with parameters P0011 or P0013 will be valid. For details, refer to *1.2.8 Interpolation Command T Designation.*

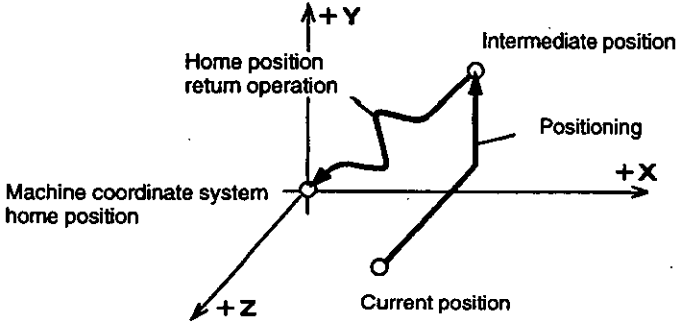
1.2.5 HOME RETURN (ZRN)

1. Incremental Position Detecting System

1) Specify HOME RETURN (ZRN) as follows:

```
ZRN  X···Y···Z···S···  ;↵
     Intermediate position
```

This command executes the home position return operation after positioning to an intermediate position by means of a POSITIONING (MOV) operation. The resultant stop position is set as the home position of the machine coordinate system. (More details concerning home position return operations are provided later in this section.) The first time that HOME RETURN (ZRN) is executed after the power has been turned ON, however, it executes the home position return operation without first travelling to the intermediate position.



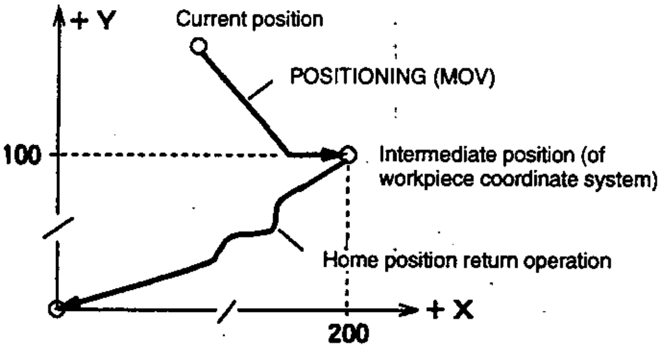
This command moves maximum four axes simultaneously. If an axis is not specified for HOME RETURN, positioning to the intermediate point and the home position return will not be executed for that axis.

After the home position return operation has been completed for all of the designated axes, program execution proceeds to the next block.

◀EXAMPLE▶ **Program Example**

ABS ; ↵

ZRN X200. Y100. ; ↵


+ Y
Current position
POSITIONING (MOV)
100
Intermediate position (of workpiece coordinate system)
Home position return operation
+ X
200

Stop position set as machine coordinate system home position (0, 0).

2) When HOME RETURN (ZRN) is executed in an incremental position detecting system, the return position is set as the home position of the machine coordinate system. At the same time, the workpiece coordinate system previously set by CURRENT POSITION SET (POS) is cancelled.

IMPORTANT

When HOME RETURN (ZRN) is executed, the machine coordinate system matches the workpiece coordinate system. Until the next time that CURRENT POSITION SET (POS) is executed, MOVE ON MACHINE COORDINATES (MVM) has no effect even if it is executed.

3) Table 1.21 shows the four types of home position return operation. The selection can be made by means of parameter PA301.

Table 1.21 Types of Home Position Return Operation

Parameter Setting	Name	Method
(1) PA301 = 0	Home position return operation 1	Three-step deceleration method using **deceleration limit switch** and C-phase pulse.
(2) PA301 = 1	Home position return operation 2	Two-step deceleration method using **deceleration limit switch.**
(3) PA301 = 2	Home position return operation 3	Three-step deceleration method using **deceleration limit switch** and home position limit switch.
(4) PA301 = 3	Home position return operation 4	Two-step deceleration method using C-phase pulse.

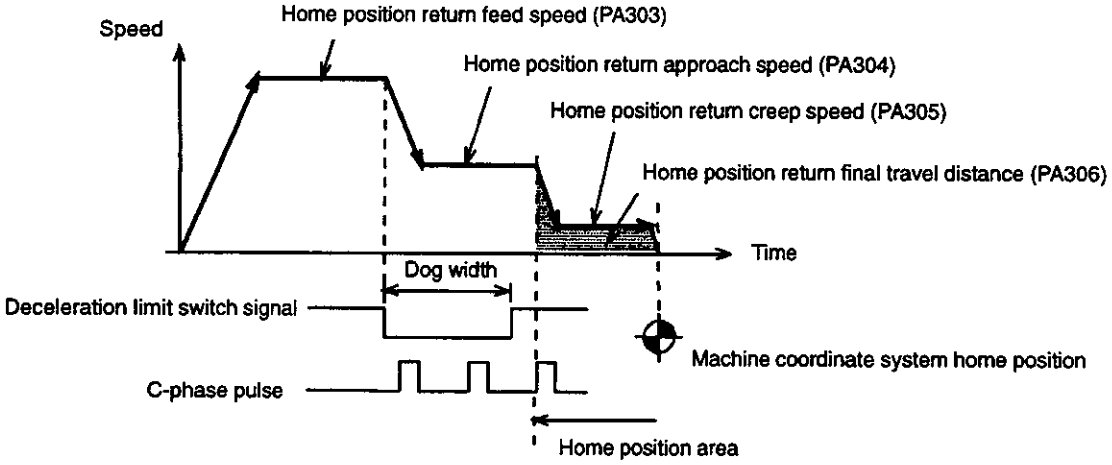
TERMS

Deceleration limit switch

A limit switch used for deceleration purposes. The deceleration timing in a home position return operation is obtained depending on the mounting positions of the deceleration limit switch and the dog.

4) **Home Position Return Operation 1 (PA301 = 0)**

The following operation chart shows the execution of a home position return operation. The return position is taken to be the machine coordinate system home position (0, 0, 0, 0).


Speed
Home position return feed speed (PA303)
Home position return approach speed (PA304)
Home position return creep speed (PA305)
Home position return final travel distance (PA306)
Time
Dog width
Deceleration limit switch signal
C-phase pulse
Machine coordinate system home position
Home position area

a) After reaching the intermediate position, positioning then begins in the direction specified by parameter PA302 (home position return direction), at the speed set by parameter PA303 (home position return feed speed).

b) When the dog that is set for deceleration turns ON the deceleration limit switch, the feed speed is reduced to the value set for parameter PA304 (home position return approach speed).

c) The speed is further reduced to value set for parameter PA305 (home position return creep speed) at the position where the pulse encoder's C-phase pulse is first detected after the dog leaves the deceleration limit switch.

d) From the position where the C-phase pulse is detected, the axis travels for only the distance set for parameter PA306 (home position return final travel distance) and then stops. That position is set as the machine coordinate system home position (0, 0, 0, 0). The travel is not executed and the home position is not set for any axis that has not been designated.

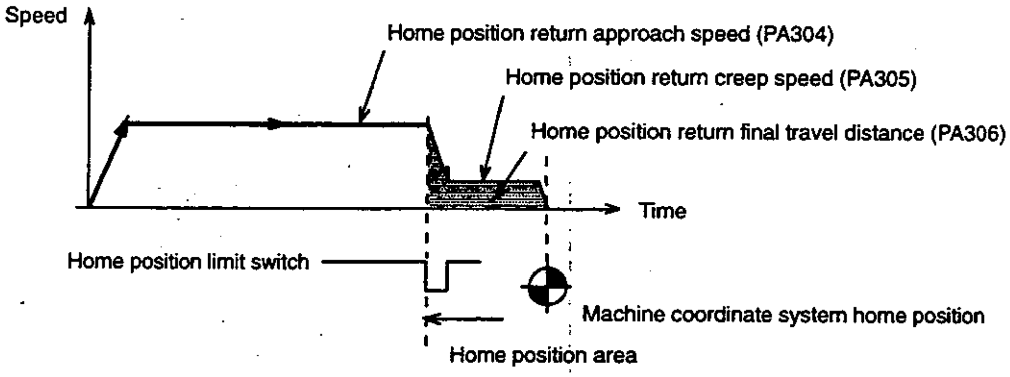
Note (1) The home position return operation will not be executed correctly if the machine is already within the home position area when the power is turned ON. Move it out of the home position area before executing the home position return operation.

(2) Set the dog width to a width at least equal to the distance required for decelerating from the home position return feed speed to the home position return approach speed.

(3) If the distance set for parameter PA306 (home position return final travel distance) is too short, the axis will pass the home position and then return to the home position from the opposite side.

5) **Home Position Return Operation 2 (PA301 = 1)**

The following operation chart shows the execution of a home position return operation. The return position is taken to be the machine coordinate system home position (0, 0, 0, 0).



a) After reaching the intermediate position, positioning then begins in the direction specified by parameter PA302 (home position return direction), at the speed set by parameter PA304 (home position return approach speed).

b) When the dog that is set for the home position turns ON the home position limit switch, the feed speed is reduced to the value set parameter PA305 (home position return creep speed).

c) After the dog turns ON the home position limit switch, the dog will move for the distance designated with parameter PA306 (home position returning final travelling distance) and stop. This position is set as the machine coordinate system home position. An axis not designated will not move and no home position setting will be performed for the axis.

Note

(1) The first time that HOME RETURN (ZRN) is executed after the power has been turned ON, it executes the home position return operation without first travelling to the intermediate position even if an intermediate position has been specified. Therefore, the home position return operation will not be executed correctly if the machine is already within the home position area when the power is turned ON. Move it out of the home position area before executing the home position return operation.

(2) If the distance set for parameter PA306 (home position return final travel distance) is too short, the axis will stop travelling at a position which is one revolution too far. If that should occur, adjust the final travel distance.

6) **Home Position Return Operation 3 (PA301 = 2)**

In place of the C-phase pulse, it is possible to use another pulse signal based on a separately installed home position limit switch. To do that, set parameter PA301 to "2."

7) **Home Position Return Operation 4 (PA301 = 3)**

In place of the home position limit switch, it is possible to use the C-phase pulse. To do that, set parameter PA301 to "3."

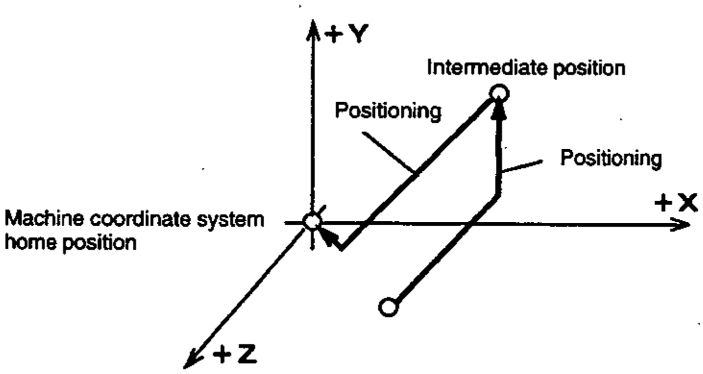
2. Absolute Position Detecting System

1) Specify HOME RETURN (ZRN) as follows:

ZRN $\underbrace{X\cdots Y\cdots Z\cdots S\cdots}_{\text{Intermediate position}}$; ↵

Based on this command, a POSITIONING (MOV) operation positions to the intermediate position and then to the machine coordinate system home position. If the workpiece coordinate system has been set by CURRENT POSITION SET (POS), the intermediate position is taken to be the position in the workpiece coordinate system.

The program cannot be started, however, if the machine coordinate system home position has not been set by means of the home position setting operation. If an attempt is made to start the program without that setting having been made, an alarm will be generated (alarm code A16: Home position setting incomplete).



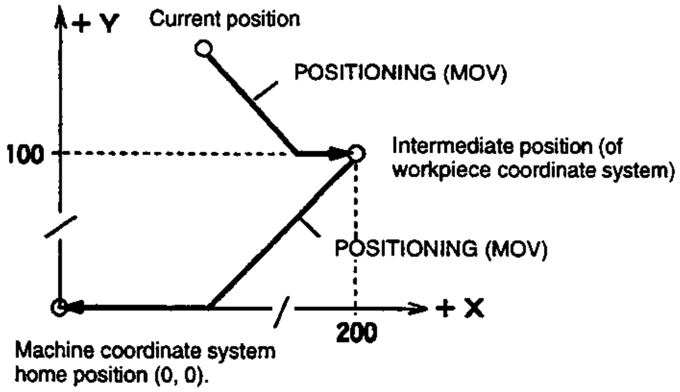
This command moves maximum four axes simultaneously. If HOME RETURN is not designated for any of the four axes, neither positioning to the intermediate point or the home position return operation is executed for that axis.

After positioning to the machine coordinate system home position has been completed for all of the designated axes, program execution proceeds to the next block.

◀EXAMPLE▶ **Program Example**

ABS ; ↵

ZRN X200. Y100. ; ↵


+ Y
Current position
POSITIONING (MOV)
100
Intermediate position (of workpiece coordinate system)
POSITIONING (MOV)
+ X
200
Machine coordinate system home position (0, 0).

2) When HOME RETURN (ZRN) is executed in an absolute position detecting system, positioning is ultimately executed to the home position of the machine coordinate system. At the same time, the workpiece coordinate system previously set by CURRENT POSITION SET (POS) is cancelled.

IMPORTANT

When HOME RETURN (ZRN) is executed, therefore, the machine coordinate system matches the workpiece coordinate system. Until the next time that CURRENT POSITION SET (POS) is executed, MOVE ON MACHINE COORDINATES (MVM) has no effect even if it is executed.

Note If alarm code A16 (home position setting incomplete) or any alarm related to the absolute encoder is generated, then execute an operation such absolute encoder initialization or home position setting. After the machine coordinate system home position has been correctly set by this means, then HOME RETURN (ZRN) can be properly executed. Refer to the section on absolute position detection in the *Motion Module MC20 Hardware User's Manual* for details.

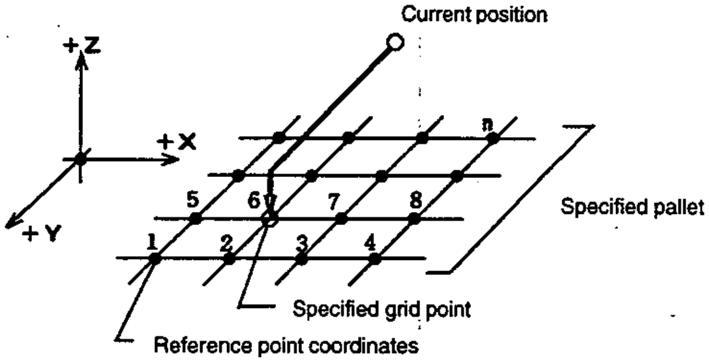
1.2.6 PALLET MOVE (PMV)

1) Basic position data can be stored in memory by executing PALLET SET (PST) to establish imaginary positions on pallets. Positioning to particular grid points on a pallet can then be executed by specifying those grid points in move command blocks.

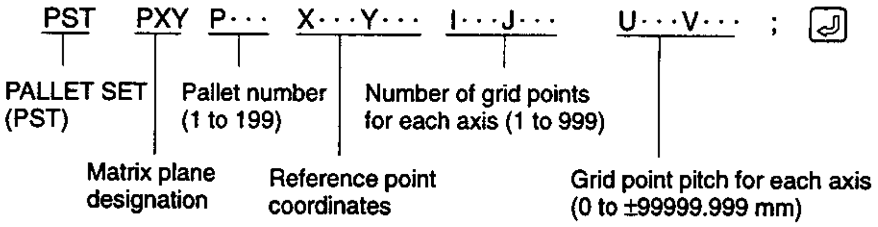
2) Specify the grid point position as follows:

```
PMV  P···  C···  ;↵
     |     |
     |     └──── Grid point number (1 to 999 x 999)
     └──── Pallet number (1 to 199)
```

This command positions at rapid traverse speed to the grid point, specified by "C," on the pallet specified by "P." The rapid traverse speed is the speed set by parameter PA202 (rapid traverse speed). The travel path is generally not linear. The grid point position is read from the basic position data stored in memory in advance by means of PALLET SET (PST).



3) Before executing PALLET MOVE (PMV), specify PALLET SET (PST) as shown blow. If this command is not executed, an alarm will be given.



a) Executing this command stores the basic matrix data in memory. The common variables #1 to #199 and the compensation values H1 to H8 can also be used for the numeric portions.

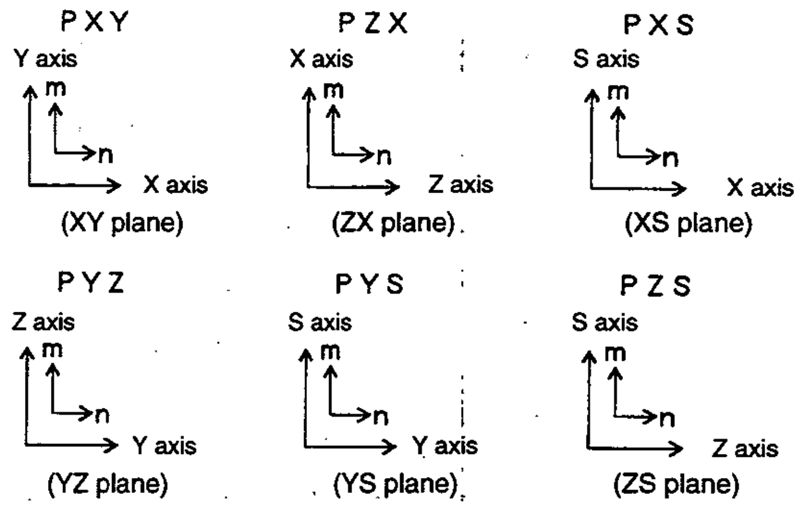
b) If there is no matrix plane setting in the block containing the PALLET SET (PST) command, the plane specified in the previous block with a PALLET SET (PST) command remains in effect. An error will be generated if PALLET SET (PST) is executed without a plane specification has been made since power was turned ON.

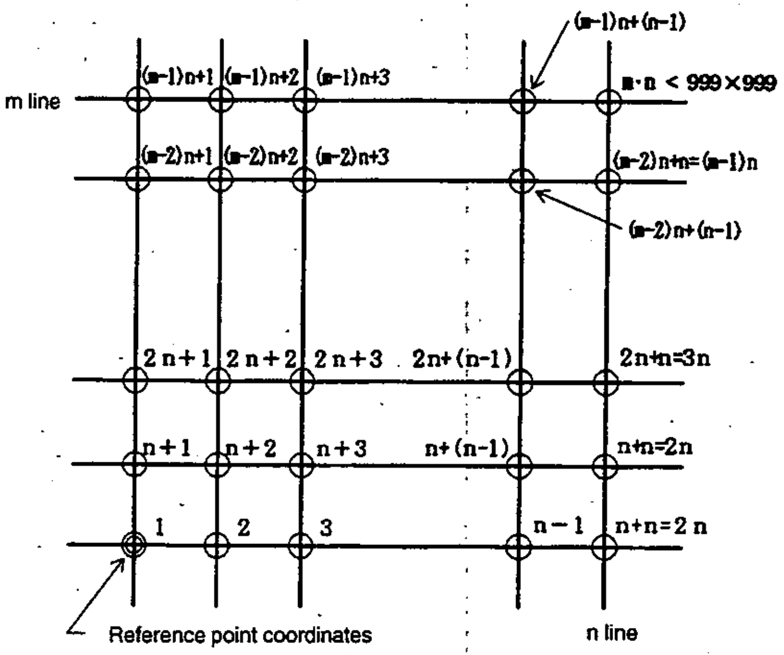
c) The formats for each matrix plane are shown in Table 1.22.

Table 1.22 Matrix Plane Command Designation Format

Plane	Matrix Setting Command	Plane Setting	Pallet Number	Reference Point Coordinates	Number of Grid Points	Grid Point Pitch
1 XY plane	PST	PXY	P	X Y	I J	U V; ↵
2 ZX plane	PST	PZX	P	Z X	I K	U W; ↵
3 XS plane	PST	PXS	P	X S	I L	U T; ↵
4 YZ plane	PST	PYZ	P	Y Z	J K	V W; ↵
5 YS plane	PST	PYS	P	Y S	J L	V T; ↵
6 ZS plane	PST	PZS	P	Z S	K L	W T; ↵

d) Assign grid point numbers for each plane as shown in the following illustration.





1.2.7 SKIP (SKP)

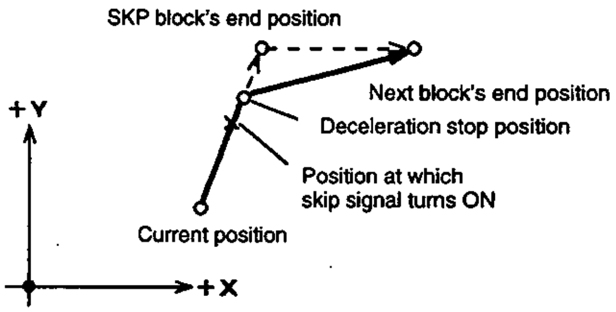
1) When the skip input signal turns ON during axis movement in a SKIP (SKP) block, axis movement is decelerated to a stop. The remaining amount of axis movement for that block is cancelled, and operation proceeds directly to the next block. Meanwhile, the position at which the skip input signal turned ON is automatically retained in memory. This command thus enables the programming of motion control which can respond to external conditions.

2) Specify SKIP (SKP) as follows:

```
SKP  X···Y···Z···S···   F···   T···  ; ⏎
       End position       │     Interpolation command T designation
                          └─ Interpolation feed speed
```

This command uses linear interpolation to move maximum four axes simultaneously at interpolation feed speed from the current position to the end position. If the skip input signal turns ON during positioning, the position at which the signal turned ON is stored in memory and the moving axes are decelerated to a stop. The command to the remaining amount of axis movement is cancelled, and program execution proceeds to the next block.

Any or all of the four axes (X, Y, Z, and S) can be designated. Any of these axes that are not designated will not be moved.



3) The interpolation command T designation makes it possible to program a time constant for acceleration/deceleration for interpolation with the character T if necessary. If the interpolation command T designation is not used or T is set to zero, the time constant for acceleration/deceleration set with parameters P0011 or P0013 will be valid. For details, refer to *1.2.8 Interpolation Command T Designation.*

4) The skip input signal is one of the MC20 Module's I/O signals. The MC20 Module's I/O connector specifications are as follows:

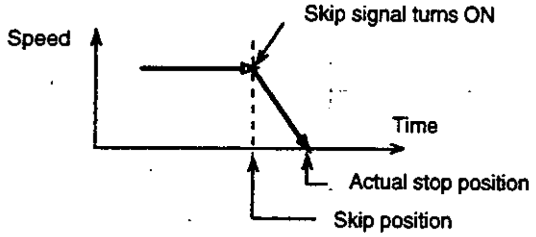
Pin Numbers	Signal Name
17	Skip input signal (normally closed contact)
21 to 25	Common lines for inputs

The point at which the skip input signal turns ON is saved in memory for all four axes, including those axes not designated by SKIP (SKP).

5) As mentioned above, the current position at which the skip input signal turns ON is saved in memory. That data is called the "skip position." It is stored as a system variable, and it can be retrieved for use later during motion program execution.

   Moving axes are decelerated to a stop after the skip input signal turns ON, so the position at which axis movement actually stops is different from the position that is saved in memory.

   When SKIP (SKP) is not being executed, the skip position is not saved in memory even if the skip signal turns ON.



6) Skip positions are stored as system variables #1005 to #1008. Meanwhile, current position data is stored moment by moment in variables #1001 to #1004. The current position values are constantly being refreshed.

   All of these values can be retrieved and substituted for variables in the motion program. The names of the variables and types of data are as follows:

#1001:	Axis 1 current position
#1002:	Axis 2 current position
#1003:	Axis 3 current position
#1004:	Axis 4 current position
#1005:	Axis 1 (SKIP) memorized position
#1006:	Axis 2 (SKIP) memorized position
#1007:	Axis 3 (SKIP) memorized position
#1008:	Axis 4 (SKIP) memorized position

Note Be careful when using these variables. The current positions and memorized positions shown above are positions in the workpiece coordinate system. If the workpiece coordinate system is changed by means of CURRENT POSITION SET (POS), the contents of variables #1001 to #1008 will also be changed.

◀EXAMPLE▶ **Program Example**

The axis 1 current position and the axis 1 memorized position respectively can be retrieved by means of the following equations:

```
#10 = #1001    (axis 1 current position)
#20 = #1005    (axis 1 memorized position)
```

The variable, however, cannot be placed in the left-hand side of the equation as shown below:

```
#1001 = #10
#1001 = 999
```

1.2.8 Interpolation Command T Designation

1) **Summary**

A time constant can be designated from the program for automatic acceleration/deceleration for interpolation for each of the following.

- LINEAR INTERPOLATION (MVS)

- HELICAL INTERPOLATION (MCW, MCC)

- CIRCULAR INTERPOLATION (MCW, MCC)

- SKIP (SKP)

2) Specify interpolation command T designation as follows:

```
MVS  X···Y···Z···S···F···T···  ;  ⏎  LINEAR INTERPOLATION
MCW  PXY X···Y···I···J···
(MCC)        (Z···)  F···  T···  ;  ⏎  HELICAL INTERPOLATION and
                                        CIRCULAR INTERPOLATION
SKP  X···Y···Z···S···F···T···  ;  ⏎  SKIP
                         ↑
                   Interpolation command T designation
```

T can be set to a value within the following range: 0.002 to 1.999 s

The interpolation command T designation is a modal specification. Once it is designated, the MC20 Module will keep the designation as valid data.

3) A time constant designated one command is valid for the other commands, e.g., one designated for acceleration/deceleration for linear interpolation is valid for circular interpolation, helical interpolation, and SKIP.

4) **Canceling An Interpolation Command T Designation**
If any of the following is executed, the time constant for acceleration/deceleration specified by the interpolation command T designation will be made invalid and the time constants for acceleration/deceleration set in the following parameters will be valid again.

a) If T is set to zero.

b) If the PROGRAM END (END) command is executed.

c) If the PROGRAM RUN (MVL) command is executed.

d) If the MACHINE RESET (RST) command is executed.

The following time constants for acceleration/deceleration will be valid again for any of the above.

Parameter	Name	Filter Selection for Interpolation
P0011	Time constant for exponential acceleration/deceleration for interpolation	• Exponential acceleration/deceleration • Exponential acceleration/deceleration with bias
P0013	Time constant for moving average acceleration/deceleration for interpolation	• Moving average acceleration/deceleration • S-curve acceleration/deceleration

If the interpolation command T designation is not used, the time constant for acceleration/deceleration set with parameters P0011 or P0013 will be valid.

Program Example

ABS ; ↵	
MVS X500.Y600.F1000T0.5 ; ↵	T is set to 0.5 s.
SET M30 ; ↵	
MVS X100.Y100.T0 ; ↵	The interpolation command T designation is canceled and the value is set to the one specified in the parameter.

Note (1) The decimal point position specified by the interpolation command T designation is always between the first and second digit without being influenced by parameter P0005.

(2) If the interpolation command T designation is used, the previous block used for axis movement will automatically execute an in-position check.

1.3 Basic Control Commands

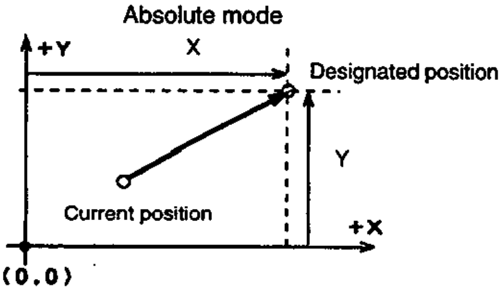
This section explains the programming methods for the basic commands that control motion. This section covers basic commands, so be sure to familiarize yourself with it before proceeding further.

1.3.1 ABSOLUTE PROGRAMMING MODE (ABS)

1) Specify ABSOLUTE PROGRAMMING MODE (ABS) as follows:

ABS ; ↵
Absolute mode

This command causes the coordinate words in all subsequently designated axis control commands to be treated as absolute values.



2) ABSOLUTE PROGRAMMING MODE (ABS) is a **modal group command**, so once it has been executed, absolute mode remains in effect until the next time INCREMENTAL PROGRAMMING MODE (INC) is executed. Absolute mode is the default mode when power is turned ON.

TERMS

Modal group command

Once a modal group command has been executed, it remains continuously in effect until another command is executed. The term "modal" refers to the way in which these commands function as modes.

3) The coordinate words that are affected by the ABS and INC commands are shown in the following table.

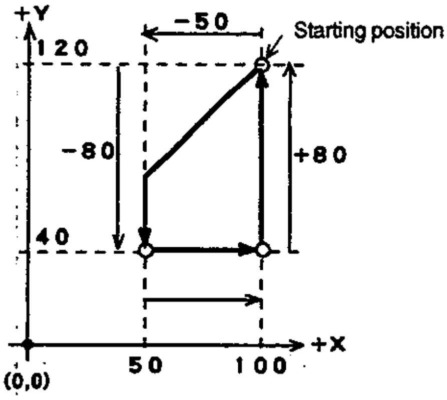
Coordinate words affected by ABS/INC	Coordinate word characters X, Y, Z, S, I, J, K, L
Coordinate words not affected by ABS/INC	R, XF, YF, ZF, SF, U, V, W, T (fixed at incremental values)

⚠ **Caution** The ABSOLUTE PROGRAMMING MODE (ABS) command causes the coordinate words in all subsequently designated axis control commands to be treated as absolute values and the INCREMENTAL PROGRAMMING MODE (INC) command causes the coordinate words in all subsequently designated axis control commands to be treated as incremental values. The meaning of a coordinate word treated as an absolute value is entirely different from that of the same word treated as an incremental value. Before operating the MC20 Module with a program, be absolutely sure to check the program to make sure that these commands are used properly. Failure to do this may result in damage to equipment, serious personal injury, or even death.

4) In this manual, absolute mode is sometimes abbreviated as "ABS mode."

◀EXAMPLE▶ **Program Example**

Program examples (A) and (B) indicate the same motion, shown on the right.



(A) ABS Mode

```
ABS MOV X100. Y120. ; ↵
(ABS ; ↵ ... This line may be omitted.)
MOV X50. Y40. ; ↵
MOV X100. ; ↵
MOV Y120. ; ↵
```

(B) INC Mode

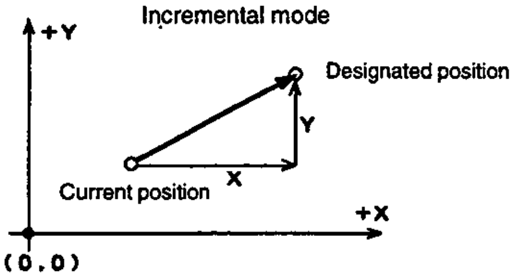
```
ABS MOV X100. Y120. ; ↵
INC ; ↵
MOV X–50. Y–80. ; ↵
MOV X50. ; ↵
MOV Y80. ; ↵
```

1.3.2 INCREMENTAL PROGRAMMING MODE (INC)

1) Specify INCREMENTAL PROGRAMMING MODE (INC) as follows:

INC ; ↵
Incremental mode

This command causes the coordinate words in all subsequently designated axis control commands to be treated as incremental values.



2) INCREMENTAL PROGRAMMING MODE (INC) is a modal group command, so once it has been executed absolute mode remains in effect until the next time ABSOLUTE PROGRAMMING MODE (ABS) is executed. Absolute mode is the default mode when power is turned ON.

3) The coordinate words that are affected by the ABS and INC commands are shown in the following table.

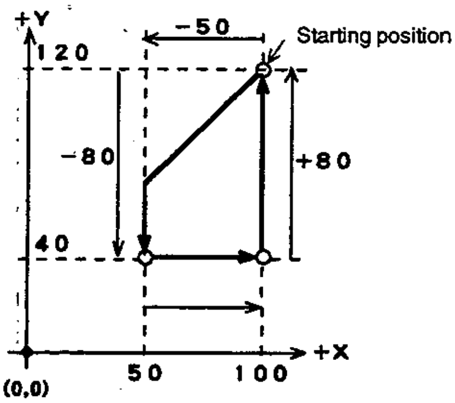
Coordinate words affected by ABS/INC	Coordinate word characters X, Y, Z, S, I, J, K, L
Coordinate words not affected by ABS/INC	R, XF, YF, ZF, SF, U, V, W, T (fixed at incremental values)

⚠ **Caution** The ABSOLUTE PROGRAMMING MODE (ABS) command causes the coordinate words in all subsequently designated axis control commands to be treated as absolute values and the INCREMENTAL PROGRAMMING MODE (INC) command causes the coordinate words in all subsequently designated axis control commands to be treated as incremental values. The meaning of a coordinate word treated as an absolute value is entirely different from that of the same word treated as an incremental value. Before operating the MC20 Module with a program, be absolutely sure to check the program to make sure that these commands are used properly. Failure to do this may result in damage to equipment, serious personal injury, or even death.

4) In this manual, incremental mode is sometimes abbreviated as “INC mode.”

◀EXAMPLE▶ **Program Example**

Program examples (A) and (B) indicate the same motion, shown on the right.



(A) INC Mode

```
ABS MOV X100. Y120. ; ↵
INC ; ↵
MOV X–50. Y–80. ; ↵
MOV X50. ; ↵
MOV Y80. ; ↵
```

(B) ABS Mode

```
ABS MOV X100. Y120. ; ↵
(ABS ; ↵ ... This line may be omitted.)
MOV X50. Y40. ; ↵
MOV X100. ; ↵
MOV Y120. ; ↵
```

1.3.3 CURRENT POSITION SET (POS)

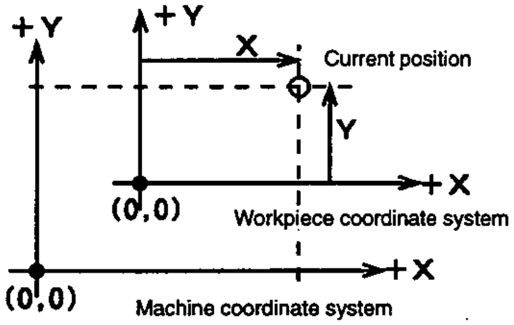
1) Specify CURRENT POSITION SET (POS) as follows:

$$\text{POS } \underbrace{\text{X}\cdots\text{Y}\cdots\text{Z}\cdots\text{S}\cdots}_{\text{Coordinate value to be changed}} \;;\, ↵$$

This command is used to change the current position to the specified new coordinates, and to create a new coordinate system. This newly specified coordinate system is called a "**workpiece coordinate system.**"

IMPORTANT

All move commands designated after CURRENT POSITION SET (POS) are carried out in the workpiece coordinate system.



2) The workpiece coordinate system can be switched as often as desired using CURRENT POSITION SET (POS), but the machine coordinate system must be set in advance. The machine coordinate system is not affected by CURRENT POSITION SET (POS).

3) CURRENT POSITION SET (POS) can be designated for maximum four axes. The current position is not set for any axis which is not designated.

⚠ **Caution** The CURRENT POSITION SET (POS) command is used to create a new coordinate system called a "workpiece coordinate system." All move commands designated after this command are carried out in the workpiece coordinate system. Before operating the MC20 Module with a program, be absolutely sure to check the program to make sure that this command is used properly. Failure to do this may result in damage to equipment, serious personal injury, or even death.

4) The status of the machine coordinate system and the workpiece coordinate system are shown in Table 1.23 according to the status of the device.

Table 1.23 Coordinate System Status

Status	Incremental Position Detecting System	Absolute Position Detecting System
After power turned ON	Machine coordinate system valid (see note a) Workpiece coordinate system cancelled (see note b)	Machine coordinate system valid Workpiece coordinate system cancelled (see note b)
After HOME RETURN (ZRN) executed	Machine coordinate system set Workpiece coordinate system cancelled (see note b)	Workpiece coordinate system cancelled (see note b)
After CURRENT POSITION SET (POS) executed	Workpiece coordinate system set	Workpiece coordinate system set
After home position set	————	Machine coordinate system set Workpiece coordinate system cancelled (see note b)

TERMS

Workpiece coordinate system

As opposed to a machine coordinate system, a workpiece coordinate system is a coordinate system set by the CURRENT POSITION SET (POS) command. The workpiece coordinate system is cancelled when HOME RETURN (ZRN) is executed, and the machine coordinate system is restored.

Note (a) When the power is turned ON, the machine coordinate system which the home position is taken as the current position is set. If HOME RETURN (ZRN) is then not executed, however, the stored stroke limit function will not be effective.

(b) "Workpiece coordinate system cancelled" means that the workpiece coordinate system becomes the same as the machine coordinate system.

5) The **current position display** on the Programming Device always indicates values in a workpiece coordinate system.

IMPORTANT

Even when the workpiece coordinate system is the same as the machine coordinate system, e.g., after HOME RETURN (ZRN) is executed, the current position display indicates values in a workpiece coordinate system.

Note Move commands in a workpiece coordinate system cannot be converted in a machine coordinate system to exceed the maximum programmable value. For example, when parameter P0005 is set to "3," do not exceed the values shown below for the machine coordinate system.

±99999.999 mm (linear axis)
±99999.999° (rotary axis)

1.3.4 MOVE ON MACHINE COORDINATES (MVM)

1) This command is used to temporarily move axes in a **machine coordinate system** after a workpiece coordinate system has been set by CURRENT POSITION SET (POS).

2) Specify MOVE ON MACHINE COORDINATES (MVM) as follows:

```
MVM  MOV X···Y···Z···S···  ; ⏎
     POSITIONING command area
MVM  MVS X···Y···Z···  F···  ; ⏎
     LINEAR INTERPOLATION command area
```

When this command is executed, positioning to the absolute coordinate position is temporarily executed in the machine coordinate system by either POSITIONING (MOV) or LINEAR INTERPOLATION (MVS). This command is always executed in absolute mode regardless of the ABS/INC mode setting.

3) MOVE ON MACHINE COORDINATES (MVM) is a **non-modal command**, so it is valid only for the block in which is is designated. Normal move commands in subsequent blocks will be executed in the workpiece coordinate system.

TERMS

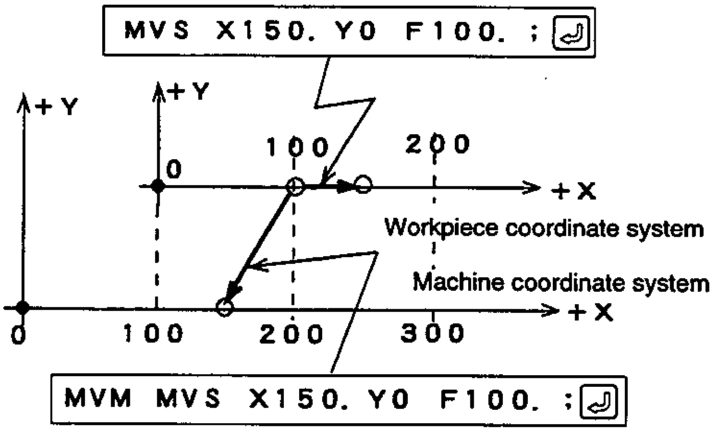
Current position display

There is either a current position value or a position display shown in the display area of a Programming Device or Teach Pendant. The coordinate values show in these display areas are always for a workpiece coordinate system.

4) The only axis movement commands that can be designated in the same block as MOVE ON MACHINE COORDINATES (MVM) are POSITIONING (MOV), LINEAR INTERPOLATION (MVS), CIRCULAR INTERPOLATION (MCW, MCC), HELICAL INTERPOLATION (MCW, MCC) and PASS NOTCH SIGNAL OUTPUT (PNT). MOVE ON MACHINE COORDINATES (MVM) must be placed at the beginning of the block.

⚠ **Caution** The MOVE ON MACHINE COORDINATES (MVM) command is used to temporarily move axes in a machine coordinate system. Before executing this command, be absolutely sure to check the home position of the machine coordinate system to make sure that the position designated with this command is correct. Failure to do this may result in damage to equipment, serious personal injury, or even death.

◀EXAMPLE▶ **Program Example 1**



```
MVS X150. Y0 F100. ;↵
```

```
MVM MVS X150. Y0 F100. ;↵
```

Program Example 2

In this program, the workpiece coordinate system is paired with the machine coordinate system.

MVM MOV X(a)Y(b)Z(c)S(d); ↵ Positions in machine coordinate system.

POS X(a)Y(b)Z(c)S(d); ↵ Sets that position in workpiece coordinate system.

TERMS

Machine coordinate system

The machine coordinate system is the basic coordinate system. It is set either by execution of HOME RETURN (ZRN) in an incremental position detecting system, or by means of a home position setting operation in an absolute position detecting system. A workpiece coordinate system can be changed and it can be set on top of the machine coordinate system.

Non-modal command

Commands that are effective only for the block in which they are designated are called "non-modal commands." The functions of these commands do not apply to subsequent blocks.

1.3.5 DWELL TIME (TIM)

Specify DWELL TIME (TIM) as follows:

```
TIM  P···; ⏎
     Dwell time
```

This command causes execution to pause for the period of time specified by "P" before proceeding to the next block.

The range of time that can be specified for "P" is 0.001 to 99999.999 seconds.

DWELL TIME (TIM) is a **non-modal command**. No other commands can be designated in the same block as TIM.

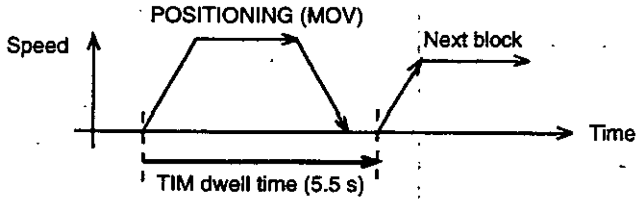
◀EXAMPLE▶ **Programming Examples**

Two programming examples are shown here. In the first example, IN-POSITION CHECK (PFN) is not in effect for the POSITIONING (MOV) command, so DWELL TIME (TIM) is executed during positioning for MOV. In this example, "P5500" sets a dwell time of 5.5 seconds.

In the second example, IN-POSITION CHECK (PFN) is in effect for the POSITIONING (MOV) command, so DWELL TIME (TIM) is executed only after positioning has been completed. In this example, "P2.5" sets a dwell time of 2.5 seconds.

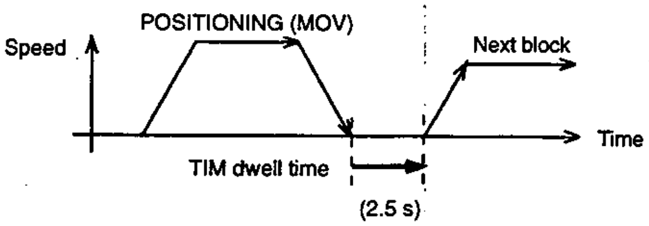
Example 1: TIM Executed during Positioning

```
MOV  X100.  ;⏎ ← IN-POSITION CHECK (PFN) not in effect.
TIM  P5500  ;⏎
       |
  5.5 seconds
```



Example 2: TIM Executed after Positioning

```
MOV  X100.  ;⏎ ] ← IN-POSITION CHECK (PFN) in effect.
PFN  ;⏎        ]
TIM  P2.5  ;⏎
      |
  2.5 seconds
```



INFO

When the time "P" is specified with no decimal point, the time will be calculated according to a formula of 1=0.001 second. This is not affected by the setting of the decimal point position parameter P0005.

1.3.6 PROGRAM STOP (STP)

1) Specify PROGRAM STOP (STP) as follows:

```
STP ;⏎
 └──── Program stop
```

This command causes program execution to be temporarily stopped after completion of the current block. No other commands can be designated to overlap this command in the same block.

2) To resume program execution from the next block after the PROGRAM STOP (STP) command, turn ON the PROGRAM RUN (MVL) start signal (input 1) from the PLC.

1.3.7 PROGRAM END (END)

1) Specify PROGRAM END (END) as follows:

```
END ;⏎
 └──── Program end
```

This command causes program execution to end after completion of the current block. If there was a move command in the previous block, a **in-position check** will be executed before program execution is completed. No other commands can be designated in the same block as PROGRAM END (END).

2) If PROGRAM END (END) is executed, the program will be executed from the beginning the next time execution is started.

TERMS

In-position check

This function detects whether the actual axis movement has entered the positioning completion range after deceleration has begun for the specified block.

Preread parallel execution

Preread parallel execution is a function in which the next block is preread while axis movement based on a move command is still underway. If there is no move command in the next block, the block is executed while the next block after it is preread. This operation is continuously repeated.

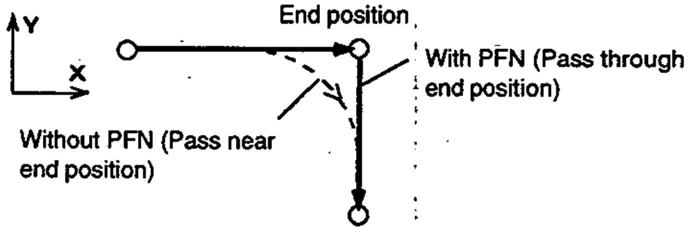
1.4 High-level Control Commands

This section explains how to program using high-level motion control commands. These are relatively complex commands, used for more advanced programming. Beginners can skip directly to *Chapter 3 Creating Ladder Logic Programs.*

1.4.1 IN-POSITION CHECK (PFN)

1) IN-POSITION CHECK (PFN) is used in the following types of situations:

a) To pass through the specified end position for the first movement when moving around a corner using interpolation commands.


Y
X
End position
With PFN (Pass through end position)
Without PFN (Pass near end position)

b) To cancel **preread parallel execution.**

2) Specify IN-POSITION CHECK (PFN) as follows:

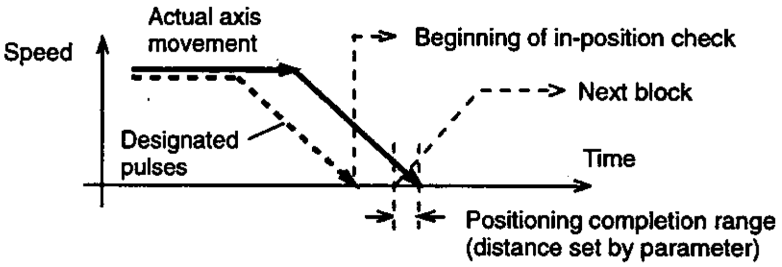
```
Positioning or interpolation block ; ↵
PFN ; ↵
Next block ; ↵
```

When this command is designated, program execution proceeds to the next block after the positioning has been completed for axis movement in the previous block (i.e., the block before the one with IN-POSITION CHECK (PFN)).

The same operation will be carried out if the command is designated as follows:

```
Positioning or interpolation block  PFN ; ↵
Next block ; ↵
```

3) The positioning completion range is set by individual parameter PA103. The range that can be set is from 0 to 250, in the specified unit.


Speed
Actual axis movement
Beginning of in-position check
Next block
Designated pulses
Time
Positioning completion range (distance set by parameter)

4) As shown above, designate IN-POSITION CHECK (PFN) in the block after the command for which **in-position check** is to be executed. It can also be designated at the end of the block for which the in-position check is to be executed.

5) Table 1.24 shows the effects of IN-POSITION CHECK (PFN) on the positioning and interpolation commands.

Table 1.24 Effects of PFN on Positioning and Interpolation Commands

PFN	(1) Move Command		(2) Next Block
	POSITIONING (MOV)	INTERPOLATION (MVS, MCW, MCC)	
Not used between (1) and (2)	Proceeds to next block after positioning completion.	Proceeds to next block when pulse distribution is completed for that block.	Move command.
	Executes the next block while command pulses are output if the next block does not contain a move command.		Not move command.
Used between (1) and (2)	Proceeds to next block after in-position check.		Move command.
			Not move command.

TERMS

In-position check

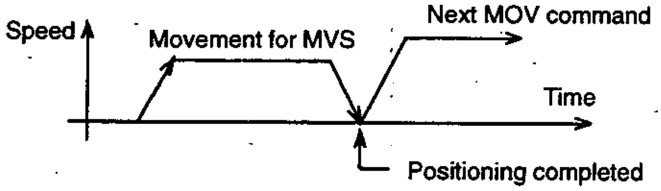
After pulse distribution has been completed for the specified block, this function proceeds to the next operation after detecting that the actual axis movement has entered the positioning completion range.

◀EXAMPLE▶

Programming Examples

Example 1

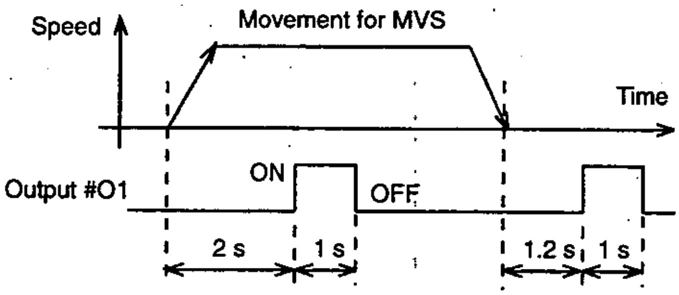
MVS X200. F100; ↵	Move using linear interpolation.
PFN; ↵	Wait for movement to end.
MOV X400Z400.; ↵	Move using positioning.



Note If there is no IN-POSITION CHECK (PFN) command, execution of the next block begins after completion of INTERPOLATION (MVS) pulse distribution.

Example 2

MVS X800. F8000; ↵	Move using linear interpolation.
TIM P2.0; ↵	Timer: 2 seconds
#O01=1; ↵	**Output #O01=ON**
TIM P1.0; ↵	Timer: 1 second
#O01=0; ↵	Output #O01=OFF
PFN; ↵	Wait for movement to end.
TIM P1.2; ↵	Timer: 1.2 seconds
#O01=1; ↵	Output #O01=ON
TIM P1.0; ↵	Timer: 1 second
#O01=0; ↵	Output #O01=OFF



TERMS

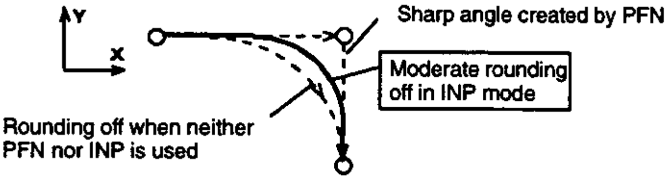
Output #O01=ON

Output #O01 is one of the output variables (#O01 to #O256). Output variables are designated from the motion program and used to turn MC relays ON and OFF in the ladder logic program in the CPU Module. "#O01=1" turns ON the MC relay, and "#O01=0" turns OFF the MC relay.

INFO It is meaningless to place the IN-POSITION CHECK (PFN) command immediately after POSITIONING (MOV). This is because the in-position check is always carried out for MOV regardless of whether or not the IN-POSITION CHECK (PFN) command is used.

1.4.2 SECOND IN-POSITION RANGE SETTING (INP)

1) SECOND IN-POSITION RANGE SETTING (INP) is used to moderately round off the end position when movement is based on interpolation.

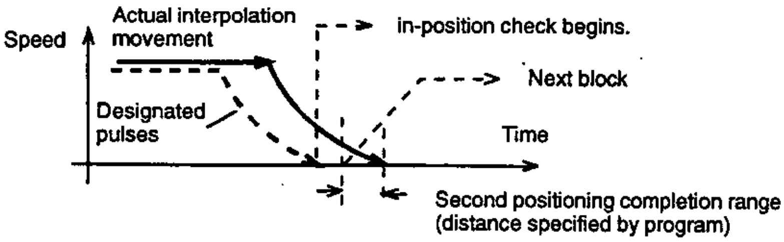

Y
X
Sharp angle created by PFN
Moderate rounding off in INP mode
Rounding off when neither PFN nor INP is used

2) Specify SECOND IN-POSITION RANGE SETTING (INP) as follows:

$$\text{INP } \underbrace{X\cdots Y\cdots Z\cdots S\cdots}_{\text{Second positioning completion range}} \;;\ \boxed{\hookleftarrow}$$

This command proceeds to the next block when it detects that the second positioning completion range has been entered during positioning by subsequently designated LINEAR INTERPOLATION (MVS) or CIRCULAR INTERPOLATION (MCW, MCC) commands. The IN-POSITION CHECK (PFN) command is not required. When there is more than one moving axis, INP proceeds to the next block after the second positioning completion range designated for each of those axes has been entered.

3) The second positioning completion range can be designated from +0.001 to +99999.999 mm. The second in-position check is started, however, when pulse distribution for the specified block has been completed, as shown in the following illustration. Therefore, if a fairly large value is set for the second positioning completion range, the second in-position check will end immediately at the deceleration starting point, and execution will proceed to the next block immediately.



4) SECOND IN-POSITION RANGE SETTING (INP) is a modal command. When any positive number other than "0" is set for the second positioning completion range, the second in-position check will remain in effect for all subsequent interpolation commands until it is cancelled.

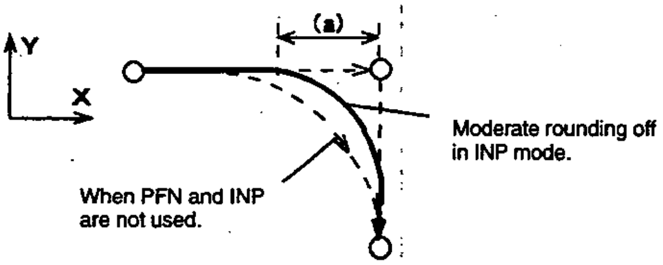
5) To cancel the second in-position setting, specify "0" for the second positioning completion range for the relevant axis or axes. The second in-position setting will be cancelled for only those axes for which "0" is specified.

The second in-position setting is cancelled for all axes when the power is turned ON.

6) The IN-POSITION CHECK (PFN) operates normally even while the second in-position setting is in effect. If it is executed, the in-position check is executed according to the positioning completion range set by parameter PA103.

◀EXAMPLE▶ **Programming Example**

INP X(a)Y(b); ⏎	Sets second positioning completion range.
MVS X100. F600; ⏎	Linear interpolation in X-axis direction
MVS Y100; ⏎	Linear interpolation in Y-axis direction


(a)
Y
X
Moderate rounding off in INP mode.
When PFN and INP are not used.

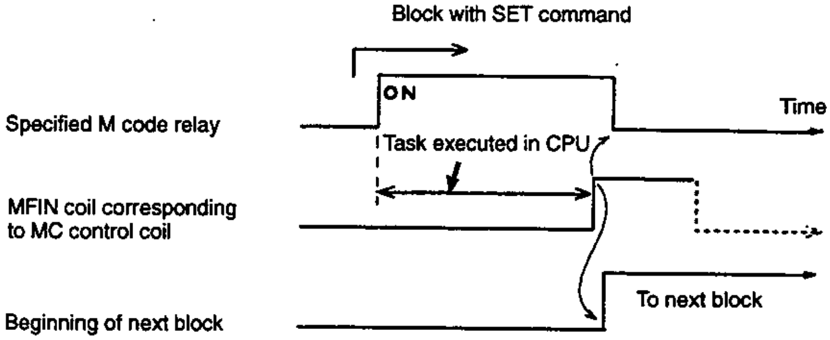
1.4.3 SET EXTERNAL OUTPUT (SET)

1) SET EXTERNAL OUTPUT (SET) is used to output an M code as a wait-for-completion signal when a specific block in the program has been executed and then to restart program execution when a response signal is received from the CPU Module.

2) Specify SET EXTERNAL OUTPUT (SET) as follows:

SET $\underbrace{\text{M□□}}$; ⏎
M code output
(□□=01 to 96)

When this command is designated, the M code (M□□) is output to the CPU Module and program execution is placed on standby. The M code relay with the number corresponding to the M code is turned ON in the CPU Module. When this signal is received, the task specified by the ladder logic program is executed. Turn ON the MFIN coil with the number corresponding to the **MC control coil** when the task executed in the CPU module is completed. When this ON status is detected, motion program execution will proceed to the next block.



3) **M Code Relays (CPU Module Side)**
M code relays are CPU Module side relays corresponding to the number of M code outputs. They can be used during ladder logic program execution. Two MC20 Modules can be connected, and the M code relay reference numbers are determined as follows:

M code relay (CPU Module side): M□00□□
- 01 to 96 = M code number
- 1, 2 = MC20 Module number

4) **M Code Sampling (MFIR) Signal (MC Control Relay)**
When the MC control relay (M code sampling: MFIR) signal is received from the MC20 Module, the CPU Module turns ON the M code relay with the number corresponding to the M code output. Use the ON signal to start the proper CPU Module tasks.

5) **MFIN Coils, MF01 to MF96 (MC Control Coils)**
The MC20 Module motion program will proceed to the next block when the MC control coil (MFIN coil: MF□□) corresponding to the specified M code relay is turned ON at the completion of the CPU Module task.

TERMS

MC control coil

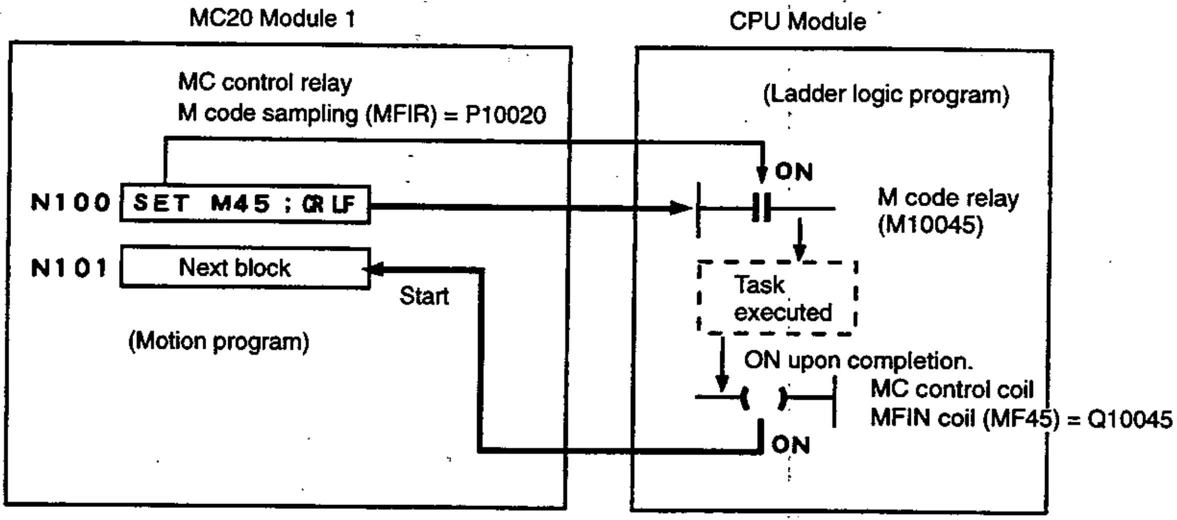
MC control coils are special signals, with fixed allocations, which serve to control the MC20 Module from the CPU Module. Their reference numbers are Q10001 to Q10160 for MC20 Module 1 and Q20001 to Q20160 for MC20 Module 2.

MC control relay

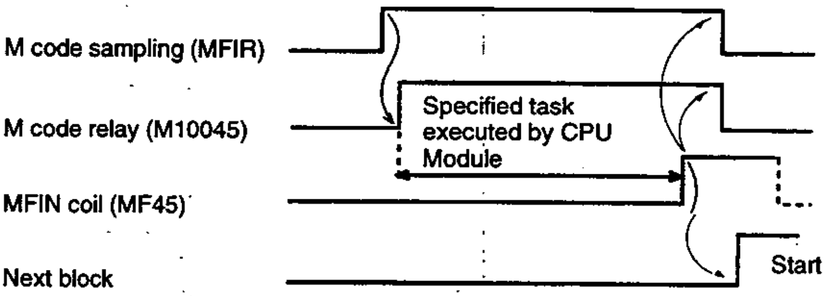
MC control relays are special signals, with fixed allocations, which transmit MC Module status to the CPU Module. Their reference numbers are P10001 to P10256 for MC Module 1 and P20001 to P20256 for MC Module 2.

6) The signals related to M code outputs described above are shown in simplified form in the following illustration. In this example, SET EXTERNAL OUTPUT (SET) is used to output M code 45 (M45) from MC20 Module 1.

Signals Related to M Code Output

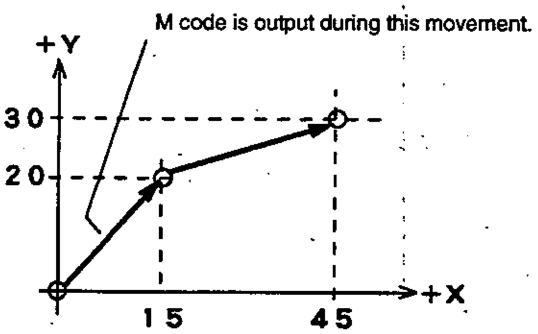


The following timing chart shows the timing for the signals in the example above.



◀EXAMPLE▶ **Programming Example**

N001 MVS X15Y20F100; ↵	Move using linear interpolation.
N002 SET M25; ↵	Output M code (M25).
N003 MVS X45Y30; ↵	Move using linear interpolation.



Note At the same time as the N001 block movement is being executed, the next SET command is also executed (i.e., via preread parallel execution). To have SET EXTERNAL OUTPUT (SET) execution delayed until after N001 movement has been completed, designate IN-POSITION CHECK (PFN) either between the N001 and N002 blocks or at the end of the N001 block.

1.4.4 PASS NOTCH SIGNAL OUTPUT (PNT)

1) PASS NOTCH SIGNAL OUTPUT (PNT) is used to output a specified M code when the designated transit point is passed. This M code output is called a "notch signal output."

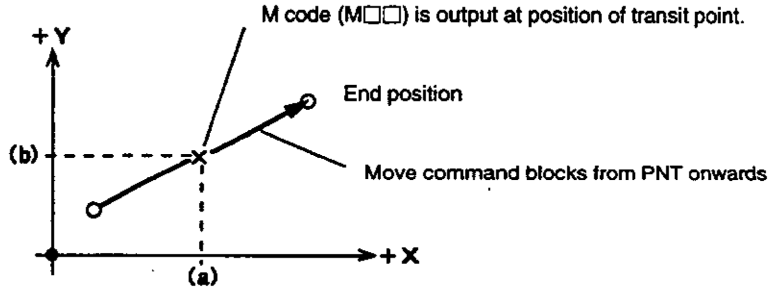
2) Specify PASS NOTCH SIGNAL OUTPUT (PNT) as follows:

```
PNT  X···Y···Z···S···              M□□ ; ↵
     Individual axis transit points  M code output
                                   (□□=01 to 96)
```

When this command is designated, the M code (M□□) is output to the CPU Module after all of the designated transit points have been passed during axis movement by subsequent move commands. Axis movement will continue to the end position.

The M code relay with the number corresponding to the M code is turned ON in the CPU Module. When the signal goes ON, perform the proper processing for the notch signal output in the ladder logic program and turn ON the MFIN coil with the number corresponding to the MC control coil when that processing has been finished. The motion program will proceed to the next block when an ON status is detected. If axis movement is not finished, the program will wait for it to finish before proceeding to the next block.

PNT X(a)Y(b) M□□; ↵

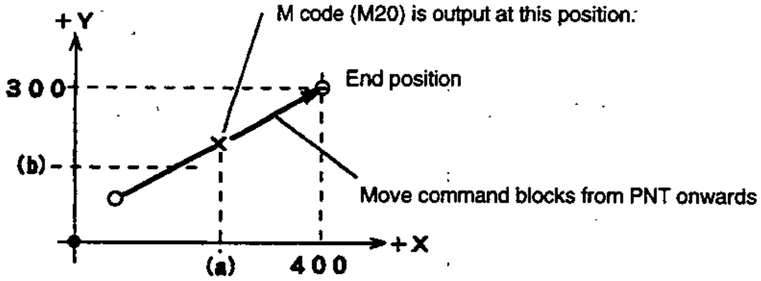

M code (M□□) is output at position of transit point.
+Y
End position
(b)
Move command blocks from PNT onwards
(a)
+X

3) The M code (M□□) is output to the CPU Module after all of the transit points designated by PASS NOTCH SIGNAL OUTPUT (PNT) have been passed. In other words, the M code is output when an AND of all of the conditions (i.e., passing the transit points for each axis) has been satisfied.

Programming Example

PNT X(a)Y(b) M20; ↵

MVS X400.Y300.F400; ↵


+Y
M code (M20) is output at this position.
300
End position
(b)
Move command blocks from PNT onwards
+X
(a)
400

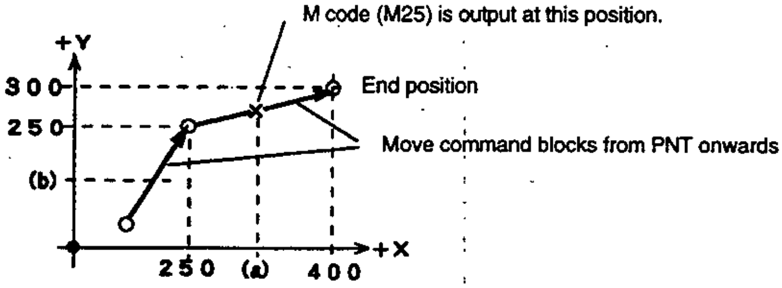
4) After PASS NOTCH SIGNAL OUTPUT (PNT) is executed, the transit points for each axis can be passed using multiple blocks. The M code is output when all of the transit points have been passed.

Programming Example

PNT X(a)Y(b) M25; ↵

MVS X250.Y250.F600; ↵

MVS X400.Y300.; ↵


M code (M25) is output at this position.
+Y
300
End position
250
Move command blocks from PNT onwards
(b)
+X
250
(a)
400

5) PASS NOTCH SIGNAL OUTPUT (PNT) is cancelled under the following circumstances:

   a) When the M code is output by the PNT command.

   b) When a new PNT command is executed without the M code having been output by the PNT command. (The new PNT command goes into effect in place of the previous one.)

6) Details on the reception of signals related to M code outputs are the same as for SET EXTERNAL OUTPUT (SET).

7) The coordinates for the individual axis transit points are values in the workpiece coordinate system. To designate them in the machine coordinate system, designate them in the same block as MOVE ON MACHINE COORDINATES (MVM).

INFO

1) A single notch signal output (i.e., M code) can be designated with a single PNT command. The position at which the notch signal is output, however, will vary depending on subsequent move commands.

2) Coordinate words for specifying individual axis transit points depend on the ABSOLUTE PROGRAMMING MODE (ABS) and INCREMENTAL PROGRAMMING MODE (INC) commands. In the INC mode, incremental values are designated from the current position at the point where PASS NOTCH SIGNAL OUTPUT (PNT) is executed.

1.4.5 IGNORE SINGLE-BLOCK SIGNAL (SNG)

1) IGNORE SINGLE-BLOCK SIGNAL (SNG) is used to continuously execute specific blocks without stopping while executing the program in **single-block operation mode.**

2) Specify IGNORE SINGLE-BLOCK SIGNAL (SNG) as follows:

```
SNG  Command that moves axes         ; ↵
           or
SNG  Command that does not move axes ; ↵
```

A block with this command is executed immediately after the previous block (according to normal execution conditions) without a single-block stop even in the single-block operation mode. Operation is continuous between the block with the SNG command and the previous block.

3) Just as in normal program execution, preread parallel execution is carried out for that portion of the program for which continuous operation is designated using IGNORE SINGLE-BLOCK SIGNAL (SNG).

◀EXAMPLE▶ **Programming Example**

```
MVS  X0Y0F1000 ; ↵                     Linear interpolation
SNG  MVS  X100.Y100. F100 ; ↵          SNG applied to linear interpolation.
SNG  #O11=1 ; ↵                        SNG applied to output variable.
#O12=1 ; ↵                             Output variable ON
#O13=1 ; ↵                             Output variable ON
```

Continuous operation carried out even in single-block operation mode.

TERMS

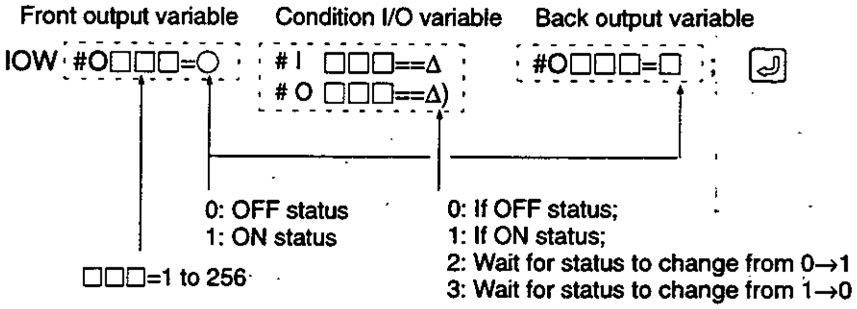
Single-block operation mode

In single-block operation mode, a stop is executed after every block. In normal program execution, a subsequent block that does not move the axes is executed while axes are being moved by means of a move command. In single-block operation mode, on the other hand, the single-block stop is executed after completion of the axis movement only.

1.4.6 I/O WAIT (IOW)

1) I/O WAIT (IOW) is used to output a front output variable, wait for a condition to be met for an I/O variable, and then output a back output variable.

2) Specify I/O WAIT (IOW) as follows:



This command executes the following sequence with regard to I/O variables:

(DO) 1) Front output variable output as "O" status.

⇩

(IF) 2) Condition **input variable** or condition **output variable** checked for "Δ" status.

⇩

(DO) 3) Back output variable output as "□" status when condition is met and execution proceeds to next block.

3) The front output variable and back output variable designations can be omitted. Outputs do not occur for the omitted parts, and program execution proceeds to the next block.

4) Be sure to use "= =" for the condition I/O variable. Use "=" for the front and back output variables.

◀EXAMPLE▶

Programming Examples

Example 1

IOW #O250==1; ↵

Program execution proceeds to the next block when the status of output variable #O250 is "1."

TERMS

Input variables and output variables

The ON/OFF status of the MC coils in the ladder logic program can be read from motion programs through input variables #I1 to #I256 and the ON/OFF status of the MC relays in ladder logic programs can be specified from motion programs through output variables #O1 to #O256.

Example 2

IOW #O250==0 #O255=1; ↵

When the status of output variable #O250 is "0," "1" is output to output variable #O255 and program execution proceeds to the next block. If the status of input variable #I250 is "1," then execution waits until the status changes to "0."

Example 3

IOW #O240=1 #I250==2; ↵

"1" is output to output variable #O240, and then execution waits until the status of input variable #I250 changes from "0" to "1" before proceeding to the next block.

Note In the status of input variable #I250 is already "1," execution does not immediately proceed to the next block. Rather, it waits until after the status is next changed to "1" and then proceeds to the next block.

Example 4

IOW #O240=0 #I250==3 #O255=1; ↵

After "0" is output to output variable #O240, execution waits until the status of input variable #I250 to change from "1" to "0." "1" is then output to output variable #O255, and execution proceeds to the next block.

Note In the status of input variable #I250 is already "0," execution does not immediately proceed to the next block. Rather, it waits until after the status is next changed to "0" and then proceeds to the next block.

5) A logical operation formula can be placed to the left of a condition I/O variable.

```
        Condition I/O Variable
          #I □□□==Δ
          #O □□□==Δ)
               ↓
#I···+#O···–#O···∗#I···/#I···==Δ
```

Use arithmetic operators (+, –, ∗, /) as operators for logical operations.

+ = OR

– = NOR

∗ = AND

/ = NAND

The definitions of the logical operations are as shown in the following table.

A	B	Results of A and B Logical Operations			
		OR	NOR	AND	NAND
0	0	0	1	0	1
0	1	1	0	0	1
1	0	1	0	0	1
1	1	1	0	1	0

Logical operations are executed in order from left to right, with no order of priority. A maximum of 11 variables can be specified.

◀EXAMPLE▶ **Programming Example**

```
IOW #O250=1 #I200+#O100*#220/#I130==3 #O120=1 ; ↵
```

1.4.7 SUB-PROGRAM CALL (GSB)

1) A sub-program stored in motion program memory can be called by the **main program** and executed a specified number of times.

2) Specify SUB-PROGRAM CALL (GSB) as follows:

```
GSB  P□□   L□□  ;↵
     │       └──────── Number of times to be repeated: 1 to 99
     │
Sub-program no. to be called: 01 to 99
```

When this command is designated, the sub-program specified by "P" is executed for the number of times specified by "L." If the "L" designation is omitted, the sub-program is executed only once.

3) Calls can be made consecutively up to four more sub-programs. In other words, the maximum number of sub-programs that can be nested in a given sub-program is four.

4) Treat sub-program numbers just the same as ordinary program numbers. That is, treat the two numeric digits following the MC20 module number (1 or 2), as the program filename extension. For details, refer to the explanation of program numbers in *1.1.3 Programming Format.*

5) A SUB-PROGRAM END (RET) command must be designated at the end of the sub-program.

TERMS

Main program

The main program is the normal program, i.e., not a sub-program.

1.4.8 SUB-PROGRAM END (RET)

1) SUB-PROGRAM END (RET) must be designated at the end of a sub-program.

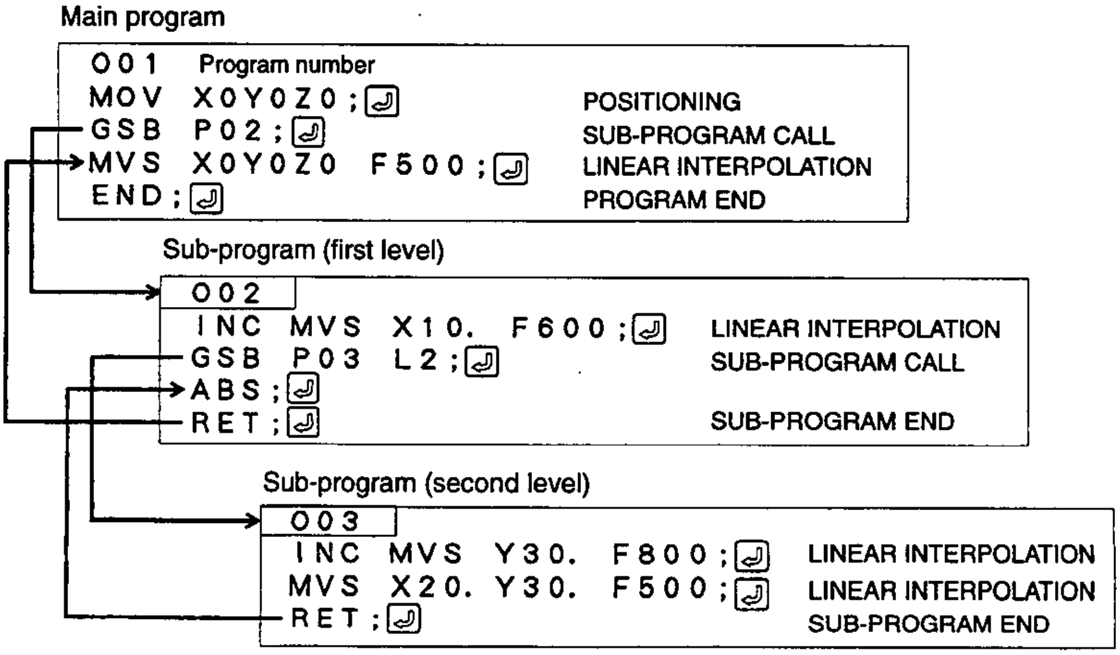
2) Specify SUB-PROGRAM END (RET) as follows:

```
RET ; ⏎
└──── Sub-program end
```

When this command is designated, program execution proceeds to the next block of the program (either the main program or a sub-program) after the SUB-PROGRAM CALL (GSB) command that called this sub-program.

3) SUB-PROGRAM END (RET) can be designated and executed within the main program to return to the beginning of the main program and start execution again.

◀EXAMPLE▶ **Programming Example (Sub-program)**



Main program

```
O01   Program number
MOV  X0Y0Z0 ;⏎              POSITIONING
GSB  P02 ;⏎                 SUB-PROGRAM CALL
MVS  X0Y0Z0  F500 ;⏎        LINEAR INTERPOLATION
END ;⏎                      PROGRAM END
```

Sub-program (first level)

```
O02
INC  MVS  X10.  F600 ;⏎     LINEAR INTERPOLATION
GSB  P03  L2 ;⏎             SUB-PROGRAM CALL
ABS ;⏎
RET ;⏎                      SUB-PROGRAM END
```

Sub-program (second level)

```
O03
INC  MVS  Y30.  F800 ;⏎     LINEAR INTERPOLATION
MVS  X20.  Y30.  F500 ;⏎    LINEAR INTERPOLATION
RET ;⏎                      SUB-PROGRAM END
```

1.5 Using Variables

This section explains commands that manipulate variables in a motion program. These are relatively complex commands used for more advanced programming. Beginners can skip directly to *Chapter 3 Creating Ladder Logic Programs.*

1.5.1 Summary of Variables

1) Variables can be inserted in place of numbers in a motion program. At the time of actual program execution using variables, the numbers stored in the variable area are retrieved.

2) Depending on the type of variable, variables are indicated by a maximum of either three or four digits following the character #.

3) The various types of variables are shown in Table 1.25.

Table 1.25 Types of Variables

Variable Types	Format	Explanation
Common variables	#1 to #199	General-purpose variables that can be used in motion programs. They can also be used for calculations in motion programs. Common variables are reset to "0" when the power is turned OFF.
Input variables	#I1 to #I256	Variables that can read the status of the MC coils in ladder logic programs. When the power is turned ON, the status of input variables depends on the ladder logic program.
Output variables	#O1 to #O256	Variables that can turn ON or OFF the MC relays in ladder logic programs. Output variables are reset to "0" when the power is turned OFF.
System variables	#1001 to #1018	Variables that can read current positions and SKIP positions. System variables are reset to "0" when the power is turned OFF.
Link input variables	#1101 to #1116	MC link register values in ladder logic programs are read by these variables and used for motion programs. The MC link registers stores the values when the MC20 module is turned OFF.
Link output variables	#1201 to #1216	Values set for these variables are automatically transferred to the corresponding MC link registers. Link output variables are reset to "0" when the power is turned OFF.

1.5.2 Common Variables (#□□□)

1) General-purpose variables that can be used in motion programs are called "common variables." Up to 199 common variables can be used.

2) Common variables #1 to #199 can be used in a motion program in place of specific values. Variables can be specified within a range of integers from 0 to ±99,999,999. Define a variable as follows:

```
#1=12345; ↵
```

3) Variables can be used for the various types of calculations, and the results of calculations can also be defined as variables.

```
#10=#20+500; ↵
#100=#30·#40/#50; ↵
```

4) Values stored as variables are stored in memory until power is turned OFF, so variables can be used in common for multiple motion programs. The variables are cleared to "0" when the power is turned OFF.

◀EXAMPLE▶ **Programming Example**

#10=100; ↵	Definition of variable
#11=200; ↵	Definition of variable
#12=300; ↵	Definition of variable
#13=1000; ↵	Definition of variable
MVS X#10 Y#11 Z#12 F#13; ↵	Movement using linear interpolation

This programming has the same result as the following:

```
MVS X100 Y200 Z300 F1000; ↵
```

Note Common variables cannot be used to specify data for the following:

a) Program numbers (O)

b) Sequence numbers (N)

c) Identification numbers (m) for repeat commands (WHILE <> DO m;↵)

1.5.3 Input Variables (#I□□□)

1) The ON/OFF status of **MC coils** can be read as input variables. Up to 256 variables can be used.

2) It is possible to use 256 input variables (#I1 to #I256) to read the ON/OFF status of the following MC coils.

Classification	Input Variables	Status Read From
MC20 Module 1	#I1 to #I256	Y10001 to Y10256 (MC coils)
MC20 Module 2	#I1 to #I256	Y20001 to Y20256 (MC coils)

3) If an MC coil is ON, the input variable of the MC coil is set to 1. If the MC coil is OFF, the input variable of the MC coil is set to 0.

ON status of coil = 1   OFF status of coil = 0

◀EXAMPLE▶ **Programming Example**

#10=I100; ⏎	The status (1 or 0) of input variable #I100 is assigned to common variable #10.
#11=I100+#I101; ⏎	A logical OR of the status of input variable #I100 and input variable #I101 is assigned to common variable #11.
#12=I100*#I102; ⏎	A logical AND of the status of input variable #I100 and input variable #I102 is assigned to common variable #12.
IF#I1==0 GOTO 20; ⏎	BRANCH command
MOV X10. ; ⏎	POSITIONING
N020 MVS Y10. ; ⏎	LINEAR INTERPOLATION

If input variable #I1 is "0" (OFF), execution jumps to N020. If input variable #I1 is "1" (ON), the next block after the BRANCH command is executed.

1.5.4 Output Variables (#O□□□)

1) **MC relays** can be output as either ON or OFF using output variables. Up to 256 variables can be used.

2) It is possible to use 256 output variables (#O1 to #O256) to turn the following MC coils ON or OFF.

Classification	Output Variables	Status Output To
MC20 Module 1	#O1 to #O256	X10001 to X10256 (MC relays)
MC20 Module 2	#O1 to #O256	X20001 to X20256 (MC relays)

3) An MC coil will be turned ON when the input variable is 1 and turned OFF when the input variable value is 0.

Output variable 1 = ON   Output variable 0 = OFF

TERMS

MC control coils

MC control coils are special signals, with fixed allocations, which serve to control the MC20 Module from the CPU Module. Their reference numbers are Q1001 to Q1160 for MC20 Module 1 and Q2001 to Q2160 for MC20 Module 2.

◀EXAMPLE▶ **Programming Example**

```
#O150=0; ↵  Output variable #O150 is output as OFF.
#O155=1; ↵  Output variable #O155 is output as ON.
IOW #O10=0 #I16==1 #O20=1; ↵
```

In the last block in the above programming, "0" is output to output variable #O10, "1" is output to output variable #O20 when input variable #I16 is "1," and then execution proceeds to the next block.

1.5.5 System Variables (#□□□□)

1) System variables are variables that can periodically read current positions, **SKIP positions,** and so on.

2) System variables can be specified within a range of #1001 to #1018. The current positions and SKIP positions that can be read for each axis are shown in the following table. These are all positions in the workpiece coordinate system.

TERMS

MC relays

MC relays are special signals, with fixed allocations, which provide MC20 Module status to the CPU Module. Their reference numbers are P10001 to P10256 for MC20 Module 1 and P20001 to P20256 for MC20 Module 2.

Skip positions

When an external skip signal turns ON during axis movement in a block with SKIP (SKP), the axis movement is decelerated to a stop. The remaining amount of movement is cancelled and program execution proceeds to the next block. The position at which the skip signal turned ON is saved in memory. This position at which the skip signal is memorized is called SKIP position. This function enables motion control to respond to external conditions. For details, see *1.2.7 SKIP (SKP).*

System Variable	Meaning
#1001	Axis 1 current position
#1002	Axis 2 current position
#1003	Axis 3 current position
#1004	Axis 4 current position
#1005	Axis 1 SKIP position
#1006	Axis 2 SKIP position
#1007	Axis 3 SKIP position
#1008	Axis 4 SKIP position
#1009	Axis 1 current position in machine coordinate system
#1010	Axis 2 current position in machine coordinate system
#1011	Axis 3 current position in machine coordinate system
#1012	Axis 4 current position in machine coordinate system
#1013 to #1016	Not used.
#1017	Error program number
#1018	Error block number

The above current and SKIP positions are all possible in workpiece coordinate systems and the current positions in machine coordinate systems are all possible in machine coordinate systems.

3) If an alarm occurs during program execution, the program number and block number that are being executed are stored in system variables #1017 and #1018 respectively. Use these variables for troubleshooting when an alarm is generated.

◀EXAMPLE▶ **Programming Example**

#1=#1001; ↵	Axis 1 current position read to common variable.
#1=#1+1000; ↵	A value of 1000 added to the current position.
MOV X#1; ↵	Axis X is moved to that position.

#2=#1008; ↵	Axis SKIP position is read.
#2=#1-1000; ↵	A value of 1000 is subtracted from the SKIP position.
MOV Y#1; ↵	Axis Y is moved to that position.

Note (1) Current positions and SKIP positions are all positions in the workpiece coordinate system. Therefore, when the workpiece coordinate system is changed by CURRENT POSITION SET (POS), the contents of variables #1001 to #1008 are changed too.

(2) System variables cannot be placed on the left side of the equation as shown below:

```
#1001 = #10
#1008 = 999
```

4) Current positions and other data can be referenced by CPU Module ladder logic programs. Holding registers can be grouped and allocated for special applications, as shown in the following diagram, by specifying the leading address of the holding registers. These holding registers are refreshed every scan.

Specify this leading address. →

CPU Module holding registers	Contents of registers		
4xxxxx	Rightmost bits: Axis 1 current position	= #1001	
4xxxxx+1	Leftmost bits: Axis 1 current position		
4xxxxx+2	Rightmost bits: Axis 2 current position	= #1002	
4xxxxx+3	Leftmost bits: Axis 2 current position		
4xxxxx+4	Rightmost bits: Axis 3 current position	= #1003	
4xxxxx+5	Leftmost bits: Axis 3 current position		
4xxxxx+6	Rightmost bits: Axis 4 current position	= #1004	
4xxxxx+7	Leftmost bits: Axis 4 current position		
4xxxxx+8	Alarm code		
4xxxxx+9	Rightmost bits: **MC link register**	= #1201	**Link input variable** (16)
4xxxxx+10	Leftmost bits: MC link register		
.....	Rightmost bits: MC link register	to	
.....	Leftmost bits: MC link register		
4xxxxx+39	Rightmost bits: MC link register	= #1216	
4xxxxx+40	Leftmost bits: MC link register		
4xxxxx+41	Rightmost bits: MC link register	= #1101	**Link output variable** (16)
4xxxxx+42	Leftmost bits: MC link register		
.....	Rightmost bits: MC link register	to	
.....	Leftmost bits: MC link register		
4xxxxx+71	Rightmost bits: MC link register	= #1116	
4xxxxx+72	Leftmost bits: MC link register		

Ver. B08

Any of the following five types of current position can be used.

a) **Reference Current Position**
The position on a workpiece coordinate system that does not include any servo system or mechanical system delay.

b) **Feedback Current Position**
The position on a workpiece coordinate system that includes servo system or mechanical system delay.

c) **Reference Speed**
The speed that does not include any servo system or mechanical system delay.

TERMS

MC link registers, link input variables, link output variables

These are names of registers in the CPU Module and variables in the MC20 Module used for transferring data between the CPU Module and the MC20 Module. For instructions on using them, see *1.5.6 Link Input Variables* and *1.5.7 Link Output Variables.*

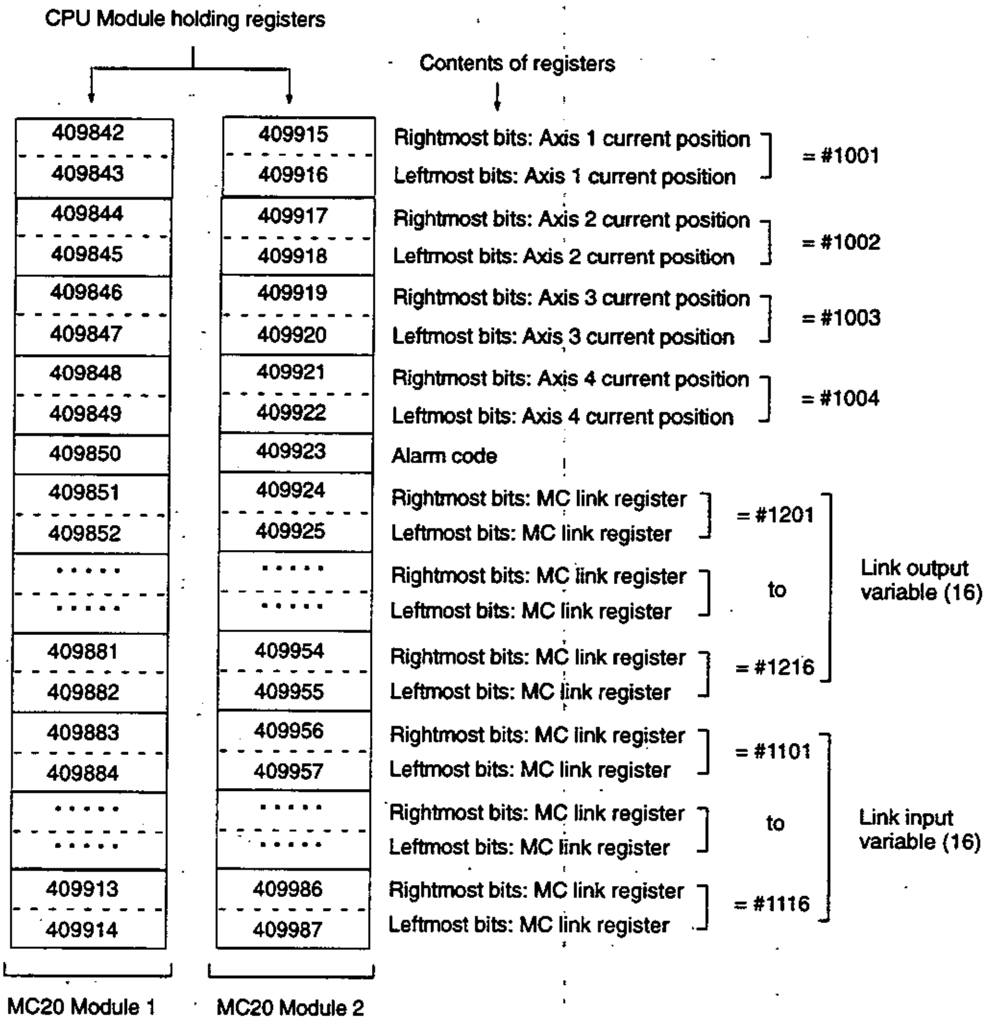
d) **Feedback Speed**
The speed that includes servo system or mechanical system delay.

e) **Deviation**
The number of pulses by which the reference current position and the feedback current position differ.

Refer to item *6) Changing the Type of Current Position* for information on changing the type of current position.

5) Special holding register groups can be allocated by specifying the leading addresses of the registers independently for MC20 Module 1 and MC20 Module 2. The factory settings (i.e., the default settings) are shown in the following figure.

Figure 2.1 Default Allocations



CPU Module holding registers

Contents of registers

MC20 Module 1	MC20 Module 2	Contents of registers		
409842	409915	Rightmost bits: Axis 1 current position	= #1001	
409843	409916	Leftmost bits: Axis 1 current position		
409844	409917	Rightmost bits: Axis 2 current position	= #1002	
409845	409918	Leftmost bits: Axis 2 current position		
409846	409919	Rightmost bits: Axis 3 current position	= #1003	
409847	409920	Leftmost bits: Axis 3 current position		
409848	409921	Rightmost bits: Axis 4 current position	= #1004	
409849	409922	Leftmost bits: Axis 4 current position		
409850	409923	Alarm code		
409851	409924	Rightmost bits: MC link register	= #1201	Link output variable (16)
409852	409925	Leftmost bits: MC link register		
·····	·····	Rightmost bits: MC link register	to	
·····	·····	Leftmost bits: MC link register		
409881	409954	Rightmost bits: MC link register	= #1216	
409882	409955	Leftmost bits: MC link register		
409883	409956	Rightmost bits: MC link register	= #1101	Link input variable (16)
409884	409957	Leftmost bits: MC link register		
·····	·····	Rightmost bits: MC link register	to	
·····	·····	Leftmost bits: MC link register		
409913	409986	Rightmost bits: MC link register	= #1116	
409914	409987	Leftmost bits: MC link register		

Ver. B08

6) **Changing the Type of Current Position**

a) The MC control coils from QN0110 to QN0112, where N is MC20 Module number 1 or 2, are set as shown in the following table to change the type of current position. The type of current position is changed at the same time for all axes. It cannot be change separately for individual axes.

MC Control Coil (N = Module Number)			Current Position Type	Unit
QN0112 (ERRREQ)	QN0111 (VELREQ)	QN0110 (FBREQ)		
OFF	OFF	OFF	Reference current position	Reference unit
OFF	OFF	ON	Feedback current position	Reference unit
OFF	ON	OFF	Reference speed	mm/min or deg/min
OFF	ON	ON	Feedback speed	mm/min or deg/min
ON	---	---	Deviation	pulses

b) The type of current position for all axes not set as servo axes in the parameters will be as shown in the following table.

Axis Type	Parameter Setting	Current Position Type				
		Reference Current Position (unit)	Feedback Current Position (unit)	Reference Speed (unit)	Feedback Speed (unit)	Deviation (unit)
Not used	P000A A = –	0	0	0	0	0
External axis	P000A A = P or Q	Count pulses (pulses)	Count pulses (pulses)	0	External axis speed (mm/min or deg/min)	0
Voltage output axis (counter not used)	PA506 b3 = 1 b4 = 0	Voltage output value (V/reference unit)	0	Voltage output value (V/reference unit)	0	0
Voltage output axis (counter used)	PA506 b3 = 1 b4 = 1	Voltage output value (V/reference unit)	Count pulses (pulses)	Voltage output value (V/reference unit)	External axis speed (mm/min or deg/min)	0

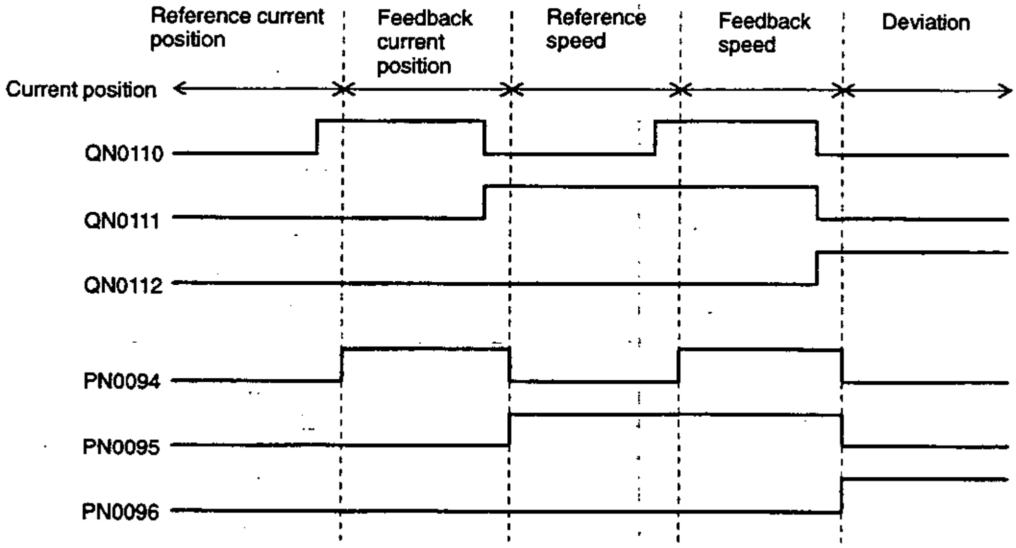
c) The values of system variables #1001 to #1004 and the data referenced for monitor number 0001 for the MONITOR (MON) command will remain as the reference current position even if the type of current position is changed.

Ver. B08

7) Confirming the Type of Current Position

MC control relays PN0094 to PN 0096, when N is MC20 Module number 1 or 2, can be used to confirm the type of current position that is being monitored. The type of current value being monitored will change as soon as the MC control relays change. Confirm the status of the MC control relays before using the monitored current position data in calculations or other operations.

MC Control Relays (N = MC Module Number)			Current Position Type
PN0096 (ERRMON)	PN0095 (VELMON)	PN0094 (FBMON)	
OFF	OFF	OFF	Reference current position
OFF	OFF	ON	Feedback current position
OFF	ON	OFF	Reference speed
OFF	ON	ON	Feedback speed
ON	OFF	OFF	Deviation



1.5.6 Link Input Variables (#□□□□)

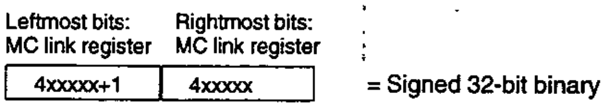
1) Link input variables are variables that read values from the MC link register in the CPU Module and which can be used to designate data in motion programs, such as position data.

2) Link input variables can be specified within a range of #1101 to #1116. These 16 variables can be used in a motion program in place of specific values. The contents of these variables is the values set for the corresponding MC link register.

3) An example of the relationships between link input variables and MC link registers is shown in the following table. The values shown here are the default factory settings. These allocations can be changed. (For details, see item 4 in *1.5.5 System Variables (#□□□□)*.

Link input Variables	MC Link Registers in CPU Module	
	MC20 Module 1	MC20 Module 2
#1101	Rightmost bits: 409883 data	Rightmost bits: 409956 data
	Leftmost bits: 409884 data	Leftmost bits: 409957 data
#1102	Rightmost bits: 409885 data	Rightmost bits: 409958 data
	Leftmost bits: 409886 data	Leftmost bits: 409959 data
#1103	Rightmost bits: 409887 data	Rightmost bits: 409960 data
	Leftmost bits: 409888 data	Leftmost bits: 409961 data
#1104	Rightmost bits: 409889 data	Rightmost bits: 409962 data
	Leftmost bits: 409890 data	Leftmost bits: 409963 data
#1105	Rightmost bits: 409891 data	Rightmost bits: 409964 data
	Leftmost bits: 409892 data	Leftmost bits: 409965 data
#1106	Rightmost bits: 409893 data	Rightmost bits: 409966 data
	Leftmost bits: 409894 data	Leftmost bits: 409967 data
#1107	Rightmost bits: 409895 data	Rightmost bits: 409968 data
	Leftmost bits: 409896 data	Leftmost bits: 409969 data
#1108	Rightmost bits: 409897 data	Rightmost bits: 409970 data
	Leftmost bits: 409898 data	Leftmost bits: 409971 data
#1109	Rightmost bits: 409899 data	Rightmost bits: 409972 data
	Leftmost bits: 409900 data	Leftmost bits: 409973 data
#1110	Rightmost bits: 409901 data	Rightmost bits: 409974 data
	Leftmost bits: 409902 data	Leftmost bits: 409975 data
#1111	Rightmost bits: 409903 data	Rightmost bits: 409976 data
	Leftmost bits: 409904 data	Leftmost bits: 409977 data
#1112	Rightmost bits: 409905 data	Rightmost bits: 409978 data
	Leftmost bits: 409906 data	Leftmost bits: 409979 data
#1113	Rightmost bits: 409907 data	Rightmost bits: 409980 data
	Leftmost bits: 409908 data	Leftmost bits: 409981 data
#1114	Rightmost bits: 409909 data	Rightmost bits: 409982 data
	Leftmost bits: 409910 data	Leftmost bits: 409983 data
#1115	Rightmost bits: 409911 data	Rightmost bits: 409984 data
	Leftmost bits: 409912 data	Leftmost bits: 409985 data
#1116	Rightmost bits: 409913 data	Rightmost bits: 409986 data
	Leftmost bits: 409914 data	Leftmost bits: 409987 data

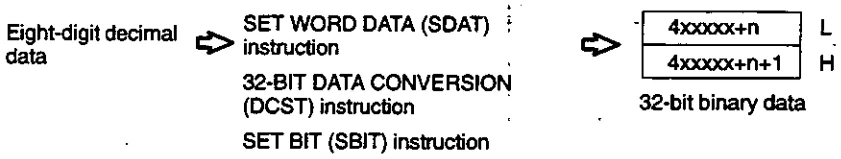
4) A single MC link register uses two holding registers, so signed 32-bit binary values can be set.



Leftmost bits: MC link register | Rightmost bits: MC link register

4xxxxx+1 | 4xxxxx = Signed 32-bit binary

It is convenient to use 32-BIT DATA CONVERSION (DCST) instruction when carrying out direct designation of 32-bit binary values.



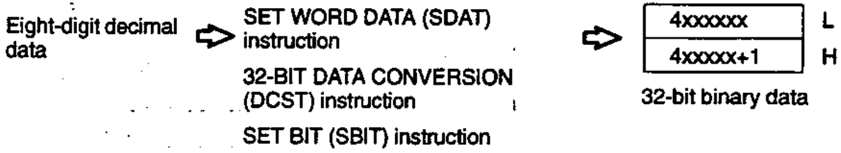
Eight-digit decimal data ⇨ SET WORD DATA (SDAT) instruction; 32-BIT DATA CONVERSION (DCST) instruction; SET BIT (SBIT) instruction ⇨ 4xxxxx+n L; 4xxxxx+n+1 H

32-bit binary data

5) The range of values that can be used for link input variables is 0 to ±99,999,999.

6) Changing 8-digit Decimal Data to 32-bit Binary Data

The following examples show how to change 8-digit decimal data into 32-bit binary data, and store it in two 16-bit holding registers using the instructions shown below in a ladder program.



Eight-digit decimal data ⇨ SET WORD DATA (SDAT) instruction; 32-BIT DATA CONVERSION (DCST) instruction; SET BIT (SBIT) instruction ⇨ 4xxxxxx L; 4xxxxx+1 H

32-bit binary data

◀EXAMPLE▶ **Programming Examples (MC20 Module 1)**

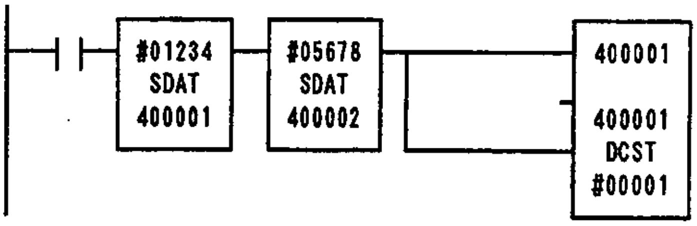
#1=#1101; ↵ — Assigns the values in MC link register 409883 and 409884 to common variable #1. (The link input variable cannot be placed on the left side of a substitution equation.)

MOV X#1102 Y#1103; ↵ — Executes positioning with the values in MC link registers 409885 and 409886 as the X-axis position data, and with the values in MC link registers 409887 and 409888 as the Y-axis position data.

PMV P#1104 C#1105; ↵ — Executes PALLET MOVE (PMV) with values in MC link register 409889 and 409890 as the pallet number, and with the values in MC link registers 409891 and 409892 as the grid point number.

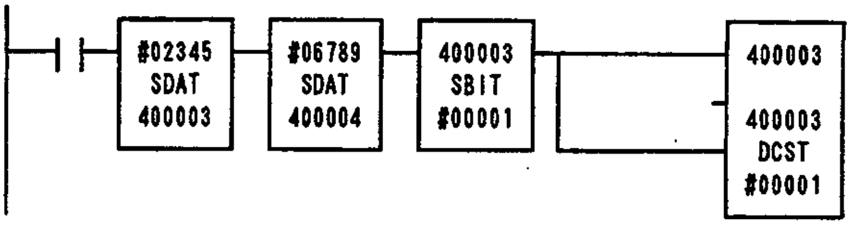
Example 1

Decimal data (12, 345, 678) is changed into 32-bit binary data and stored in holding registers 400001 and 400002.



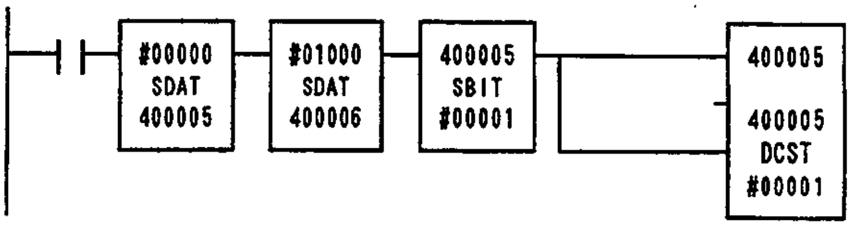
Example 2

Decimal data (-23,456,789) is changed into 32-bit binary data and stored in holding registers 400003 and 400004.



Example 3

Decimal data (-1,000) is changed into 32-bit binary data and stored in holding registers 400005 and 400006.



7) **Using MC Link Registers to Set Overrides**

MC link registers can be used to set overrides for feed speeds. Set parameter P0016 to enable the use of MC link registers.

a) Overrides are enabled and disable in parameter P0016.

(1) Override: Enabled for MC control coils (16 steps)

Ver. B08

(2) Override: Enabled for MC link registers (in increments of 0.1%)

(3) Override: Disabled

The overrides are disabled in the default settings.

b) When an override is enabled, the percentage can be switched as shown below.

(1) **MC Control Coils (16 Steps)**
0, 1, 2, 4, 6, 8, 10, 20, 30, 40, 50, 60, 60, 80, 90, or 100 (%)

Ver. B08

(2) **MC Link Registers (0.1% Increments)**
0.0 to 3276.7 (%)

When an override is disabled, the speed will be 100% of the set value or reference value.

c) Overrides are set using the following MC link registers. The registers that are shown are the default allocations for these settings. The allocations can be changed by the customer.

Use		MC Link Registers		Setting
		Module 1	Module 2	
Rapid traverse speed		409909	409982	Set value: 0 to 32767 Override: 0.0% to 3276.7%
Maximum interpolation feed speed		409910	409983	
Independent axes and manual	Axis 1	409911	409984	
	Axis 2	409912	409985	
	Axis 3	409913	409986	
	Axis 4	409914	409987	

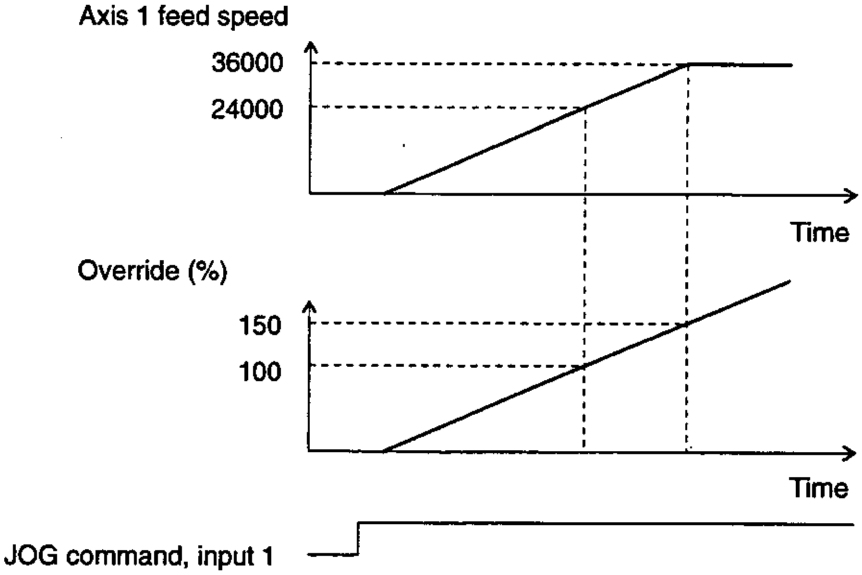
Set the MC link register to 10 times the override percentage.

d) Set a signed binary number (2's complement) in the MC link register. An override of 0 will be used is 0 or a negative value is set directly.

e) The set speed times the override will be clamped to the maximum interpolation speed parameter (P0006) or the maximum feed speed parameter for each axis (PA201, where A is axis number 1 to 4).

For example, assume that axis 1 is jogged when the following parameters are set as shown.
P1201 (max. feed speed for axis 1) = 36000 mm/min
P1202 (feed speed for axis 1) = = 24000 mm/min

If the override percentage is gradually increased from 0, the feed speed for axis 1 will be clamped to 36000 mm/min, as shown below.



1.5.7 Link Output Variables (#□□□□)

1) When values are set for link output variables in a motion program, they are automatically transferred to the corresponding MC link registers in the CPU Module.

2) Link output variables can be specified within a range of #1201 to #1216. Values that are set are automatically transferred to MC link registers in the CPU Module every 10 ms. These 16 variables can be used in a motion program in place of specific values.

3) An example of the relationships between link output variables and MC link registers is shown in the following table. The values shown here are the default factory settings. These allocations can be changed. (For details, see item 4 in *1.5.5 System Variables (#□□□□)*.

Link Output Variables	MC Link Registers in CPU Module	
	MC20 Module 1	MC20 Module 2
#1201	Rightmost bits: 409851 data	Rightmost bits: 409924 data
	Leftmost bits: 409852 data	Leftmost bits: 409925 data
#1202	Rightmost bits: 409853 data	Rightmost bits: 409926 data
	Leftmost bits: 409854 data	Leftmost bits: 409927 data
#1203	Rightmost bits: 409855 data	Rightmost bits: 409928 data
	Leftmost bits: 409856 data	Leftmost bits: 409929 data
#1204	Rightmost bits: 409857 data	Rightmost bits: 409930 data
	Leftmost bits: 409858 data	Leftmost bits: 409931 data
#1205	Rightmost bits: 409859 data	Rightmost bits: 409932 data
	Leftmost bits: 409860 data	Leftmost bits: 409933 data
#1206	Rightmost bits: 409861 data	Rightmost bits: 409934 data
	Leftmost bits: 409862 data	Leftmost bits: 409935 data
#1207	Rightmost bits: 409863 data	Rightmost bits: 409936 data
	Leftmost bits: 409864 data	Leftmost bits: 409937 data
#1208	Rightmost bits: 409865 data	Rightmost bits: 409938 data
	Leftmost bits: 409866 data	Leftmost bits: 409939 data
#1209	Rightmost bits: 409867 data	Rightmost bits: 409940 data
	Leftmost bits: 409868 data	Leftmost bits: 409941 data
#1210	Rightmost bits: 409869 data	Rightmost bits: 409942 data
	Leftmost bits: 409870 data	Leftmost bits: 409943 data
#1211	Rightmost bits: 409871 data	Rightmost bits: 409944 data
	Leftmost bits: 409872 data	Leftmost bits: 409945 data
#1212	Rightmost bits: 409873 data	Rightmost bits: 409946 data
	Leftmost bits: 409874 data	Leftmost bits: 409947 data
#1213	Rightmost bits: 409875 data	Rightmost bits: 409948 data
	Leftmost bits: 409876 data	Leftmost bits: 409949 data
#1214	Rightmost bits: 409877 data	Rightmost bits: 409950 data
	Leftmost bits: 409878 data	Leftmost bits: 409951 data
#1215	Rightmost bits: 409879 data	Rightmost bits: 409952 data
	Leftmost bits: 409880 data	Leftmost bits: 409953 data
#1216	Rightmost bits: 409881 data	Rightmost bits: 409954 data
	Leftmost bits: 409882 data	Leftmost bits: 409955 data

4) A single MC link register uses two holding registers, so signed 32-bit binary values can be set.

Leftmost bits: MC link register	Rightmost bits: MC link register	
4xxxxx+1	4xxxxx	= Signed 32-bit binary

5) The range of numbers that can be used for link output variables is 0 to ±99,999,999.

◀EXAMPLE▶ **Programming Examples (MC20 Module 1)**

#1201=-12345678; ↵
Transfers the value "-12,345,678" to MC link registers 409851 and 409852.
(The link output variable can also be placed on the right side of a substitution equation.)

#1202=#1001; ↵
Transfers the axis 1 current position to MC link registers 409853 and 409854.
(A variable can be used on the right side of an equation, as shown here.)

MOV X#1203 Y#1204; ↵
Executes positioning with the value in link output variable #1203 as the X-axis position data, and with the value in link output variable #1204 as the Y-axis position data.

PMV P#1205 C#1206; ↵
Executes PALLET MOVE (PMV) with the value in link output variable #1205 as the pallet number, and with the value in link output variable #1206 as the grid point number. By programming in this way, the pallet number is available in MC link registers 409891 and 409892, and the grid point number is available in MC link registers 409893 and 409894 in the CPU Module. This data can be used in ladder logic programs.

1.5.8 H Variables (H□)

1) H variables (H□) can be used in place of specific position and speed data.

2) H variables can be specified within a range of H1 to H8. The following example shows how H variables can be used.

   MOV XH1 YH2 ZH3 FH4; ↵

3) H variables are set in advance using the CPU Module's H VARIABLE SETTING (VAR) command before the program is executed. If no setting is made, the previous data remains in effect. All H variables are factory set to "0." Once an H variable has been set, the value is backed up by battery even when power is turned OFF.

4) H variables can be used multiple times within a program.

5) H variables can be assigned to common variables, as shown below:

   #1=H1; ↵

6) H variables can also be used in calculations. Here, they can also be placed on the left side of equations.

   #1=H1+H2; ↵
   #2=H3+100; ↵
   H8=#100+300; ↵

Note Two or more H variables cannot be used in combinations like the following one.

MOV XH1+H2; ↵ **Incorrect**

1.6 Arithmetic Commands

This section explains the arithmetic calculations that are executed on variables or between variables and constants. These are used in relatively complex programming. Beginners can skip directly to *Chapter 3 Creating Ladder Logic Programs.*

1.6.1 DEFINE (=)

1) Specify DEFINE (=) in either of the following ways:

#i=#j; ↵

or

#i=100; ↵

When this command is used, the variable on the left side of the equation is defined either as another variable or as a constant.

2) A variable that is defined in advance can be assigned to yet another variable.

#10=100; ↵

#20=#10; ↵ Command for changing variable

Here, the common variable #20 is defined as the number "100."

1.6.2 ADD (+) and SUBTRACT (–)

1) Specify ADD (+) and SUBTRACT (–) as follows:

#i=#j+#k; ↵ ADD

#i=#j–#k; ↵ SUBTRACT

When these commands are used, the variable on the left side of the equation is defined as the result of the addition or subtraction operation on the right side. Constants can be used in place of variables in the addition and subtraction operations.

◀EXAMPLE▶ **Programming Example**

First variables #11 and #12 are defined:

#11=200; ↵

#12=300; ↵

Then variables #100 and #120 are defined as the sum and difference of the first variables:

#100=#11+#12; ↵ Common variable #100 is defined as "500."

#120=#11–#12; ↵ Common variable #120 is defined as "–100."

1.6.3 MULTIPLY (*) and DIVIDE (/)

1) Specify MULTIPLY (*) and DIVIDE (/) as follows:

#i=#j*#k; ↵ MULTIPLY

#i=#j/#k; ↵ DIVIDE

When these commands are used, the variable on the left side of the equation is defined as the result of the multiplication or division operation on the right side. Constants can be used in place of variables in the addition and subtraction operations.

◀EXAMPLE▶ **Programming Example**

First variables #13 and #14 are defined:

#13=5; ↵

#14=3; ↵

Then variables #130 and #140 are defined as the product and quotient of the first variables:

#130=#13*#14; ↵ Common variable #130 is defined as “15.”

#140=#13/#14; ↵ Common variable #140 is defined as “1.”
(Digits to the right of the decimal point are truncated.)

1.6.4 Combined Arithmetic Operations

Arithmetic commands can be combined as follows:

```
#i=#j+#k-#I*#m/#n; ↵
```

Note (1) **The operations are executed from left to right, with no order of priority of operations.**

(2) If there are more than 11 terms on the right side of the equation, alarm #040 (i.e., number of terms in expression exceeded) will result.

◀EXAMPLE▶ **Programming Example**

#150=10+5*4; ↵ Common variable #150 is defined as “60.”

1.6.5 Calculation Precision

1) Only integers can be used in calculations. Decimal points cannot be used even in variables.

2) The range of integers that can be used is 0 to ±99,999,999.

3) When a decimal occurs in a calculation result, all digits to the right of the decimal point are truncated.

1.7 Control Commands

This section explains the BRANCH (IF...GOTO) and REPEAT (WHILE...DO) commands that change the flow of the program. These are used in relatively complex programming. Beginners can skip directly to *Chapter 3 Creating Ladder Logic Programs.*

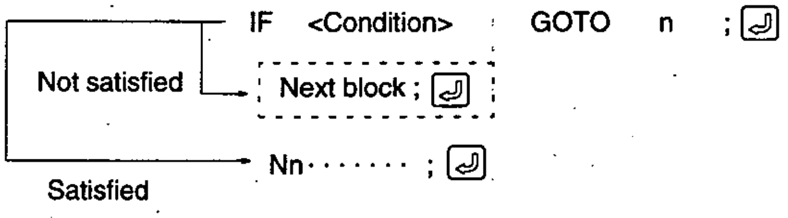
1.7.1 BRANCH (IF...GOTO)

1) This command causes program execution to jump to the block with the specified sequence number if the specified conditions are satisfied.

2) Specify BRANCH (IF...GOTO) as follows:

```
IF   <Condition>     GOTO   n   ;↵
                            |
                  Sequence number (□□□)
```

If the condition is satisfied, this command causes execution to jump to the block in the same program with sequence number "n." If the condition is not satisfied, program execution proceeds to the next block after IF.



3) The following table shows the types of discriminants that can be entered for the condition.

<Condition>	Meaning	<Condition>	Meaning
#i==#j	#i=#j	#i<#j	#i<#j
#i<>#j	#i≠#j	#i>=#j	#i≧#j
#i> #j	#i>#j	#i<=#j	#i≦#j

Constants, common variables, input variables, output variables, H variables, and point tables (#E) can all be used for #i and #j.

4) The Module determines the sequence number of the jump destination in the following ways.

a) When there is no sequence number in the IF block,
First a search is carried out from the BRANCH command downward through the program. If the sequence number has not been discovered by the time PROGRAM END is reached, the search is then carried out from the BRANCH command upward through the program. If the same sequence number exists more than once in the program, the first one that is found will be used.

b) When there is a sequence number in the IF block:

(1) If the sequence number in the IF block is smaller than "n," the search is carried out downward through the program. If no sequence number is found in that direction, the search is then carried out upward through the program.

(2) If the sequence number in the IF block is larger than "n," the search is carried out upward through the program. If no sequence number is found in that direction, the search is then carried out downward through the program.

5) If the "IF <Condition>" is omitted, the BRANCH command becomes a simple jump command that will cause a jump unconditionally.

```
┌── GOTO n ; ↵      Simple jump command
│    . . . .
│    . . . .
│    . . . .
└── Nn· · · · ·  ;↵
```

◀EXAMPLE▶ **Programming Example**

N011 #10=#I20; ↵	Input variable #I20 read.
N012 IF #10==1 GOTO 14; ↵	If #I20 is ON, program execution jumps to block N14. If it is not ON, execution proceeds to block N13.
N013 MVS Z100. F600; ↵	
N014 MOV X0 Y0 Z0; ↵	

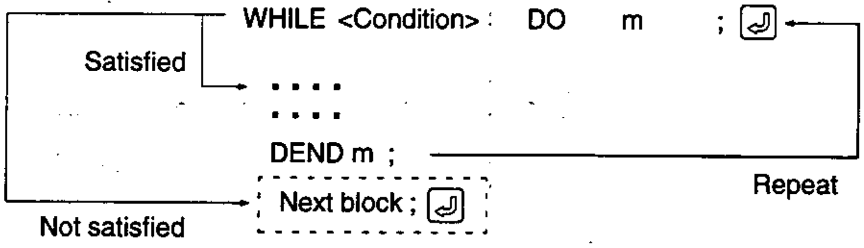
1.7.2 REPEAT (WHILE...DO)

1) This function causes execution of a specified range of blocks to be repeated as long as a specified condition is being satisfied.

2) Specify REPEAT (WHILE...DO) as follows:

```
WHILE <Condition>   DO   m    ; ↵
. . . .                  │
. . . .              REPEAT identification
                     number (m = 1, 2, 3)
DEND   m  ; ↵
  │
  End of REPEAT command with identification number m
```

When this command is used, the blocks from WHILE to DEND are repeated as long as the condition is satisfied. If the condition is not satisfied, program execution proceeds to the next block after DEND.

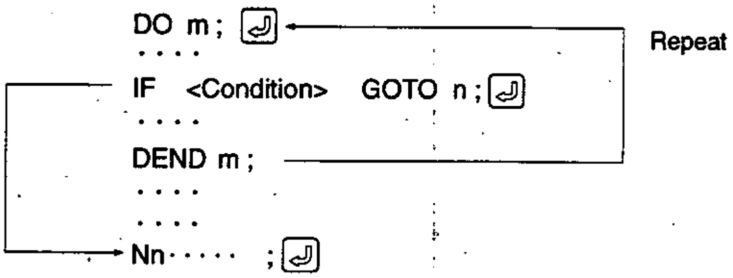


```
WHILE <Condition>  DO   m   ; ⏎
  . . . .
  . . . .
  DEND m ;
  Next block ; ⏎
```

3) Only "1," "2," or "3" can be used for the REPEAT identification number. Use the same number for "m" in both the "DO m" and "DEND m" for the same loop.

4) The same identification number can be used a number of times. If the range that is to be repeated is the same, however, use a different identification number.

5) If "WHILE <Condition>" is omitted, the blocks from "DO m" to "DEND m" are repeated endlessly (i.e., an infinite loop is created). Insert a BRANCH (IF) in the loop in order to provide some condition to escape from the loop.



```
DO  m ; ⏎
. . . .
IF   <Condition>   GOTO  n ; ⏎
. . . .
DEND  m ;
. . . .
. . . .
Nn· · · · ·   ; ⏎
```

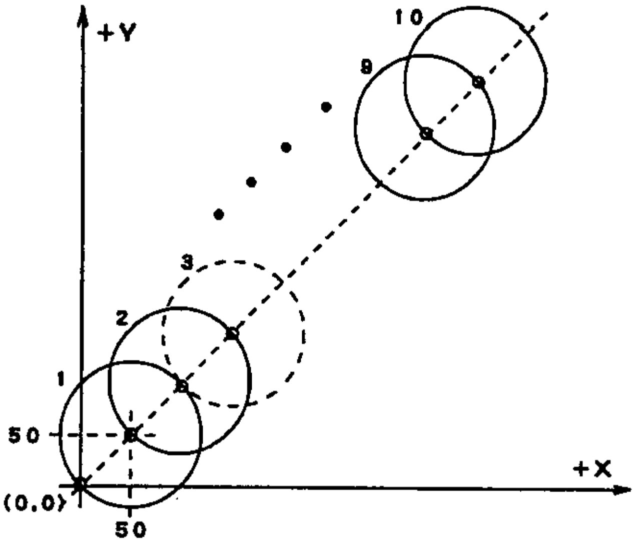
Note (a) REPEAT commands can be nested up to three levels. That is, "1," "2," and "3" can be used for identification number "m" to create loops within loops.

(b) BRANCH (IF) can be used to exit from a repeating loop, but it cannot be used to enter into one.

◀EXAMPLE▶ **Programming Example**

```
N020 MOV X0 Y0; ↵                         POSITIONING
N021 #10=1; ↵                             Defines common variable.
N022 INC; ↵                            INCREMENTAL PROGRAMMING MODE
N023 WHILE #10<=10 DO 1; ↵                REPEAT
N024 MCW PXY X0 Y0 I50.J50.F80; ↵         CIRCULAR INTERPOLATION
N025 MOV X50. Y50.; ↵                     POSITIONING
N026 #10=#10+1; ↵                         Adds 1 to common variable.
N027 DEND 1; ↵                            Ends REPEAT command.
```

This program section will draw ten circles from circle 1 to circle 10, as shown in the following illustration.



1.8 Point Tables

This section explains how to execute motion commands using a position data array called a "point table," and how to use a point table to specify variables. These commands are used in relatively complex programming. Beginners can skip directly to *Chapter 3 Creating Ladder Logic Programs.*

1.8.1 POINT TABLE POSITION (#E)

1) POINT TABLE POSITION (#E) is used to create a point table in advance. Position data for any or all of the four axes can be stored in the point table. After the point table has been created, operations in the program can be executed by specifying a point number in motion commands for axis movements to positions recorded in the point table.

2) Specify a point number as follows:

```
a±#E□□□
│    └──── Point number  (□□□=1 to 500)
│
└──── Axis designation character (X, Y, Z, S, etc.)
```

Once recorded in the point table, the position data for the point number specified by "#E□□□" can be used as a coordinate word in move commands such as POSITIONING (MOV), LINEAR INTERPOLATION (MVS), etc.

If the "a" axis designation is omitted, the point number applies to all valid axes (except for independent axes A, B, C, and D).

3) Before specifying a point number as described above, it is first necessary to create the point table. As shown in Table 1.26, the maximum point table memory capacity is 500 points per axis for up to four axes. Only integers can be saved in memory, and the maximum values are as shown in the table.

Table 1.26 Point Tables and Maximum Values

Point No.	Axis No.	Axis 1	Axis 2	Axis 3	Axis 4
	a	Y Axis	Z Axis	X Axis	A Axis (Note a)
E001		±99,999,999	±99,999,999	±99,999,999	±99,999,999
E002		±99,999,999	±99,999,999	±99,999,999	±99,999,999
E003		±99,999,999	±99,999,999	±99,999,999	±99,999,999
E···		·····	·····	·····	·····
E498		±99,999,999	±99,999,999	±99,999,999	±99,999,999
E499		±99,999,999	±99,999,999	±99,999,999	±99,999,999
E500		±99,999,999	±99,999,999	±99,999,999	±99,999,999

Note (a) Axis designation characters (X, Z, Y, A, etc.) are set in parameters P0001 to P0004. The axis designation characters in the table are examples only.

(b) Point table position data is retained even when the power is turned OFF.

4) There are three methods for creating point table position data.

a) **Using the Teach Pendant**
First select the point table edit mode for the Teach Pendant. Then manually move each axis to the specified position and press the Teach Key to record that current position data in the point table.

b) **Using the Programming Device**
First select the point table edit mode for the Programming Device. Then directly input the data for each axis to create the point table. Finally, transfer the point table file to the MC20 Module.

c) **Executing POINT TABLE SETTING (PTBL)**
The point table can be created by using the CPU Module's POINT TABLE SETTING (PTBL) command to transfer data from registers to the specified point number. (Only one point can be specified at a time.)

5) Point tables can be designated by means of motion commands. The underlined MOV is just an example of the various motion commands used for axis movement for which point numbers can be specified.

MOV a+#E□□□; ↵ — Designates axis data by means of axis designation character.

or

MOV #E□□□; ↵ — Designates data for all valid axes (except independent axes A, B, C, and D).

When these commands are used, positioning is executed to the data position designated by the specified point number, either for a designated axis or for all valid axes.

Plus (+) signs can be omitted, but minus (–) signs cannot.

6) Point numbers can be used with the following motion commands:

- MOV, MVS, ZRN, PNT, INP, POS, SKP: All axes can be designated.
- MCW, MCC, PST: Designating all axes generates an alarm.

◀EXAMPLE▶ **Programming Examples**

1) **Conditions**
All of the subsequent programming examples use the following conditions. That is, the S, B, C, and D axes are not valid axes.

Table 1.27 Point Table Data Example

Point No.	Axis No.	Axis 1	Axis 2	Axis 3	Axis 4
	a	Y Axis	Z Axis	X Axis	A Axis
....					
E123		10000	20000	30000	40000
....					

The reference unit is three digits to the right of the decimal point (P0005 = 3).

2) **Examples**
The programming examples shown on the left express the same values as those on the right.

```
MOV X#E123; ⏎          →     MOV X30; ⏎
MOV Y–#E123; ⏎         →     MOV Y–10; ⏎
MOV #E123; ⏎           →     MOV X30. Y10. Z20; ⏎
```

Note The following programming examples will generate alarms for the above conditions.

```
MOV A#E123; ⏎          Alarm 006: Character error
                       (Independent axes A, B, C, D cannot be designated.)
MOV S#E123; ⏎          Alarm 024: Undefined axis error
                       (When S axis is not set as valid axis)
```

1.8.2 Using Position Data as Variables (#E)

1) The point table position data array created in the previous section can be treated as variables using #E□□□, and can be used in arithmetic commands and control commands.

2) Specify a point number as follows:

```
a#E□□□
│   └──── Point number  (□□□=1 to 500)
└──── Axis designation character (X, Y, Z, S, A, B, C, D, etc.)
```

The data for the axes designated for the point number specified by #E□□□ can be used in the following commands:

a) The right side of **variable definitions** in arithmetic commands.

b) Control commands such as IF and WHILE.

3) Define a variable as follows:

```
#i=a#E□□□; ↵
```

This defines the designated axis data for the point number specified by #E□□□ as a variable (#i). No sign (+ or –) can be designated in front of "#."

◀EXAMPLE▶ **Programming Examples**

The programming examples shown on the left express the same values as those on the right.

```
#1=X#E123; ↵        →      #1=30000; ↵
#2=Z#E123; ↵        →      #2=20000; ↵
#3=Y#E123; ↵        →      #3=10000; ↵
```

As shown below, polynomial expressions can also be used.

```
#1=X#E123+Y#E123; ↵         →      #1=40000; ↵
#2=#1–Y#E123+Z#E123; ↵      →      #2=40000; ↵
```

The calculation results are obtained using the point table data example shown in Table 1.27.

Note (a) The point table position cannot be designated on the left side of the definition statement.

(b) The following programming examples will generate alarms for the example conditions.

TERMS

Variable definitions

- #i=#j; ↵

This command defines the variable on the left side of the equation as the variable on the right. For details, refer to *1.6.1 DEFINE (=)*.

#1=S#E123; ⏎	Alarm 024: Undefined axis error (When S axis is not set as valid axis)
#1=#E123; ⏎	Alarm 005: Numerical value designation error (All axes cannot be designated for definition statement.)
#1=X–#E123; ⏎	Alarm 005: Numerical value designation error (Sign cannot be designated in front of "#.")

The following example, however, is correct:

#1=–X#E123; ⏎

4) Specify **control commands** as follows:

IF a#E□□□==□ GOTO n; ⏎

or

WHILE a#E□□□==□ DO n;

When these commands are used, BRANCH (IF...GOTO) and REPEAT (WHILE...DO) are executed depending on whether or not the conditions using the point number are satisfied.

◀EXAMPLE▶ **Programming Examples**

IF Y#E123==10000 GOTO 10; ⏎	Satisfied
IF Z#E123==20000 GOTO 10; ⏎	Satisfied
IF X#E123==30000 GOTO 10; ⏎	Satisfied

WHILE Z#E123==20000 DO 1; ⏎	Satisfied
WHILE X#E123==30000 DO 2; ⏎	Satisfied
WHILE Y#E123==10000 DO 3; ⏎	Satisfied

The calculation results are obtained using the point table data example shown in Table 1.27.

TERMS

Control commands

- Conditional branching is executed by the following command:
  IF <Condition> GOTO n; ⏎
- The specified range of the program is repeated by the following command:
  WHILE <Condition> DO m; ⏎

For details, refer to 1.7 *Control Commands.*

Note The following programming examples will generate alarms for the example conditions.

IF B#E123==0 GOTO 10; ↵	Alarm 024: Undefined axis error (When B axis is not set as valid axis)
WHILE #E123==0 DO 1; ↵	Alarm 005: Numerical value designation error (All axes cannot be designated control command.)
IF X–#E123==0 GOTO 20; ↵	Alarm 005: Numerical value designation error (Sign cannot be designated in front of axis designation.)
IF –X#E123==0 GOTO 20; ↵	Alarm 005: Numerical value designation error (Sign cannot be designated in front of "#.")

INFO

5) General notes and precautions

a) Point numbers can be designated as the following characters:

(1) Center point coordinate designation characters for circular interpolation: I, J, K, L

(2) Linear interpolation designation characters for helical interpolation: XF, YF, ZF, SF

(3) Individual axis grid point and pitch characters for PALLET SET (PST) for use in PALLET MOVE (PMV): I, J, K, L, U, V, W, T

The names of the axes corresponding to the characters are as follows:

I→X, J→Y, K→Z, L→S, XF→X, YF→Y, ZF→Z, SF→S, U→X, V→X, W→Z, T→S

Programming Examples

MCW X#E10Y#E10I#E20J#E20 F100; ↵ CIRCULAR INTERPOLATION

MCW PXY X10.Y20.R30.ZF#E10 F100; ↵ HELICAL INTERPOLATION

PST P1X0Y0I#E10J#E10U#E20V#E20; ↵ PALLET SET

b) Besides integers 1 to 500, common variables #1 to #199 can be used to specify point numbers (E□□□).

Programming Example

#1=100

MOV X#E#1; ↵ expresses the same value as:

MOV X#E100; ↵

Note Only integers and common variables can be used for specifying point numbers. If other variables are used, an alarm (alarm code 006: character error) will be generated.

Using common variables allows data saved for independent axes A, B, C, and D, and axes that not valid in the point table to be used in motion commands.

c) Point numbers cannot be designated for the following character data:

(1) Program numbers:	O–
(2) Sequence numbers:	N–
(3) Circular interpolation radius:	R–
(4) Pallet numbers:	PST P–
(5) Dwell times:	TIM P–
(6) M codes:	SET M–
(7) M codes:	PNT M–
(8) I/O variables:	IOW #O– #I– #O–
(9) Sub-program numbers or number of repetitions:	GSB P– L–
(10) Interpolation command T designation:	T–
(11) Grid points on palletizing surfaces	

d) Polynomial expressions cannot be used when point numbers are designated in motion commands.

Programming Example

MOV X#E10+Y#E20; ↵

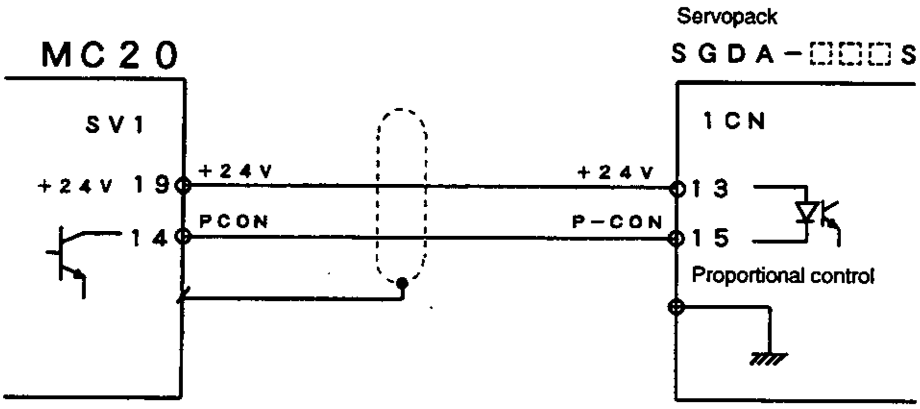
Alarm 008: Command error
(Polynomial expression not allowed.)

1.9 Advanced Signal Output Commands

This chapter explains the commands used for advanced signal output.

1.9.1 PCON SIGNAL OUTPUT (PCN)

1) The PCN command turns ON or OFF the PCON□ signal outputs on MC20 Module servo connectors SV1 to SV4. The PCON signal outputs connect to a Yaskawa Servopack Servo Amp, and is used to control its P-CON input.



The above diagram shows a connection example with a typical Yaskawa SGDA-□□□S Servopack. The Servopack P-CON input is mainly used for switching between proportional control and proportional/integration control. It also has other uses. For details, refer to the User's Manual for the Servopack model being connected.

When the remote device is not a Servo Amp, it can be used as an ordinary signal output.

2) Specify the PCON SIGNAL OUTPUT (PCN) command as follows:

```
PCN   X 1 Y 1 Z 1 S 1 ; ⏎
      │ └──────── Specifies PCON signal output ON/OFF.
      │           (1 = ON, 0 = OFF)
      └──────── Specifies the axis for which the signal is to be output.
```

Depending on the command operands, the PCON□ signal output in servo connector SVn output of the specified axis is turned ON or OFF. The axis number of the controlled axis is n = 1 to 4.

The relationship between the axis number n and the axis designation character (X, Y, Z, or S) is set in parameter numbers P0001 to P0004. The PCON signal output for an axis for which the reference is omitted is not refreshed.

3) The PCON□ signal output specifications are as follows:

Pin Number	Signal Name
14	PCON□ signal output (open collector output)
17	Common line for output (also servo alarm GND:ALM0□)

4) The PCN command is a preread command that is executed parallel with a move command.

5) This command is completed when the PCON□ signal is output. The PCON□ signal output status changed by the PCN command is maintained even if program operation ends. The status is also maintained even if the program is interrupted by an alarm or other cause. In manual mode and edit mode, the status of the MC control coils (PCNn) depends on this status. (For details, see the supplementary information at the end of *1.9.1.PCON SIGNAL OUTPUT (PCN)*)

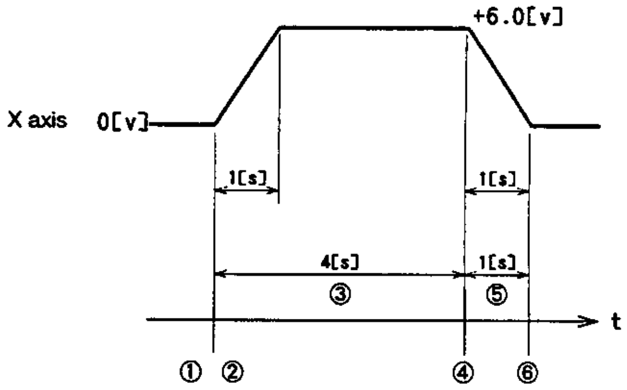
6) PCON□ signal output can be executed for any axis, regardless of whether a servo axis or a voltage output axis is specified in the parameter.

◀EXAMPLE▶ **Programming Example**

Set the X axis as a voltage output axis (parameter PA506 b3=1).

It is assumed that the PCON□ signal in servo connector SVn for this axis is connected to the P-CON input of a Yaskawa SGDA-□□□S Servopack.

1	PCN X1 ; ↵	Sets the Servopack to the proportional control status. (P control)
2	VCC X6.0T1.0 ; ↵	Operate the X axis at the speed reference in the following diagram, using the voltage output function.
3	TIM P4.0 ; ↵	See above.
4	VCC X0 ; ↵	See above.
5	TIM P1.0 ; ↵	See above.
6	PCN X0 ; ↵	Sets Servopack to proportional/integration control. (PI control)



Setting the memory switches of SGDA-□□□S Servopack as shown below enables the P-CON input function to switch P/PI control.

- Cn-01 Bit A = 0, Bit B = 0
- Cn-02 Bit 2 = 0

Note With the PCN command, an indirect specification using a variable cannot be made for the 1 (ON), 0 (OFF) designation.

INFO

1) The PCN command can be executed using the single-block operation of the Teach Pendant (TB120).

TERMS

Voltage output axis and voltage output function

By setting parameter PA506 b3 to 1 for the voltage axis function, the voltage output function is enabled for the specified controlled axis. For details, see *1.9.2 VOLTAGE OUTPUT (VCC)*. Setting PA506 b3 to 0 specifies a normal servo axis.

2) Besides the PCN motion command, the PCON signal output can be turned ON or OFF by turning ON or OFF the MC control coil (PCNn, where n = 1 to 4) from the CPU Module. Section 1.9.4 of shows which of the two methods is given priority.

3) The final output status of the PCON□ signal for each axis can be monitored using an MC control relay (PCNLn, where n = 1 to 4).

4) The status of the PCON□ signal output is maintained as follows:

   a) For a Basic Axis (X, Y, Z, or S Axis)

      (1) In Edit Mode or Manual Mode:
      Depends on the MC control coil (PCNn) status.

      (2) In Automatic Mode, Online Edit Mode, and Teach Mode:
      The last PCN command status specified by the motion program is maintained. If the status changes to any of these modes from edit mode or manual mode, the PCON□ status is maintained in the mode before the change.

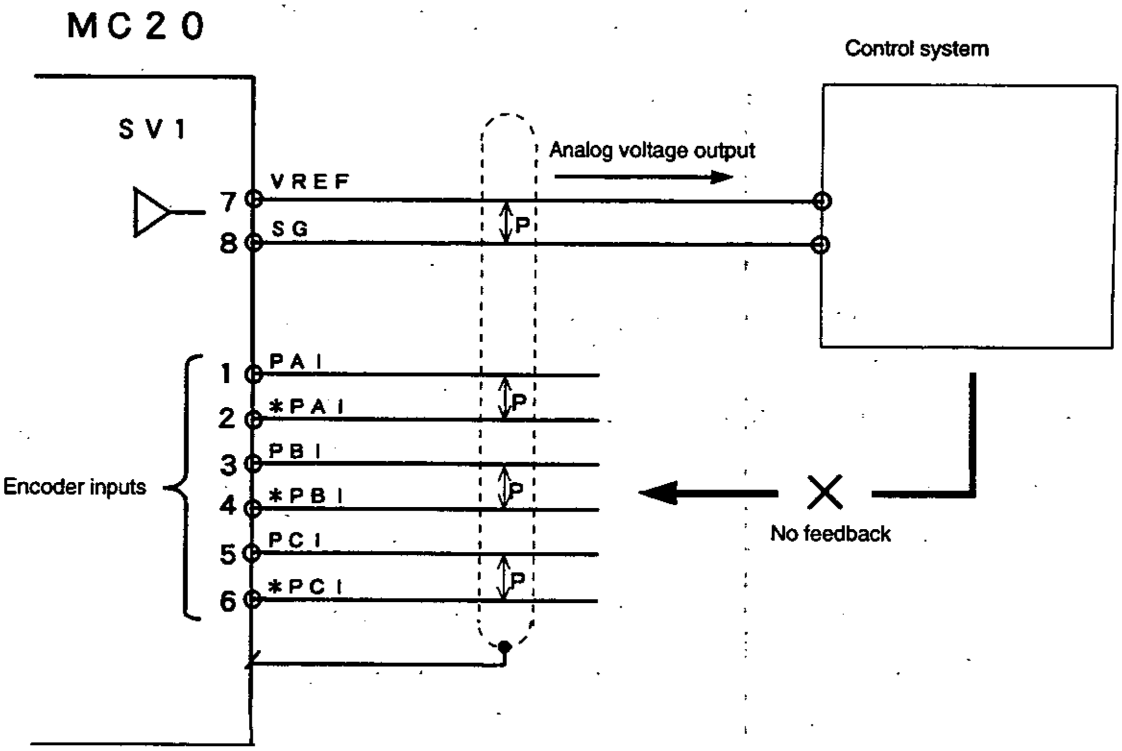
   b) For an Independent Axis (A, B, C, or D Axis)
   Depends on the MC control coil (PCNn) status, regardless of the operation mode.

5) When the connected device is not a Servo Amp, the PCON□ signal outputs can be used as ordinary signal outputs.

1.9.2 VOLTAGE OUTPUT (VCC)

STEP 2

1) The VCC command outputs an analog voltage that changes with time from the VREF□ speed reference outputs on servo connectors SV1 to SV4 of the MC20 Module.



The voltage output function can be enabled for the desired controlled axis by setting parameters. As shown in the previous diagram, feedback control is not provided for a controlled axis for which the voltage output function has been enabled. Therefore, parameters can be set to use the encoder inputs to servo connectors SV1 to SV4 of the MC20 Module as follows:

a) To not use the encoder inputs.

b) To use the encoder inputs as ordinary counter functions.

2) To enable the voltage output function, first set b3 of parameter PA506 to 1 (voltage output axis), and then turn ON the power supply again.

Parameter No.	Name	Setting Range
PA506	Mode setting b3: Axis type b4: Counter function	 0: Servo axis 1: Voltage output axis 0: Disabled 1: Enabled

Parameter No.	Name	Parameter Rewrite Enabled/Disabled		Effective Timing	Initial Value	Parameter Type
		From Ladder	From Teach Pendant			
PA506	b3	Disabled	Disabled	At power-up	0	C
	b4	Disabled	Disabled		0	B

3) Specify the VOLTAGE OUTPUT (VCC) command as shown below. (The following example assumes all four axes are set for voltage output axes.)

```
VCC  X···  Y···  Z···  S···  T···  ; ↵
     ┬ ┬                     ┬
     │ │                     └ Linear acceleration/deceleration time (0 to 99.999 s)
     │ └─ Voltage reference value (0 to ±10.000 V)
     └── Specifies the axis on which the voltage is output
```

Depending on the command, VCC outputs an analog voltage for the specified linear acceleration/deceleration time as a VREF□ signal in servo connector SVn of the specified axis. (Replace n with the axis number (1 to 4) of the controlled axis.) The relationship between the axis number n and the axis designation character (X, Y, Z, or S) is set using parameter numbers P0001 to P0004.

The voltage reference is not refreshed for axes for which a reference is omitted.

4) The VREF□ signal output specifications are as follows:

Pin No.	Signal Name
7	VREF□: speed reference output (operational amplifier output)
8	SG: Signal ground

5) The VCC command is a preread command that is executed parallel to a move command. This command is completed when the VREF□ signal is output.

6) The VCC command is valid only while a program is being executed. When the program execution is completed or the program is interrupted by an alarm, etc., the voltage output value automatically becomes 0 V.

7) The voltage reference value is set as follows:

a) With a decimal point, it can be set in the following range.

Setting range: 0 to +10.000 V

b) Without a decimal point, it can be set in the following range. The position of the decimal point is not affected by the setting of parameter P0005.

Setting range:		-10000 to +10000
Reference unit	:	1 = 0.001 V (fixed)

c) The voltage reference value is set in advance using ABSOLUTE PROGRAMMING MODE (ABS) or INCREMENTAL PROGRAMMING MODE (INC).

Mode	Meaning of Voltage Reference Value
ABS (absolute mode)	Outputs the specified voltage value as the target reference value.
INC (incremental mode)	Outputs the target reference value as an increment from the current reference voltage.

The voltage reference initial value is 0 V.

d) When voltage output is executed simultaneously for multiple controlled axes, the locus is close to interpolation, but it does not become a strict interpolation operation.

8) The linear acceleration/deceleration time is specified as follows:
The only acceleration/deceleration method available is linear acceleration/deceleration.

a) With a decimal point, it can be set in the following range.

Setting range: 0 to 99.999 s

b) Without a decimal point, it can be set in the following range. The position of the decimal point is not affected by the parameter (P0005) setting.

Setting range: 0 to 99999
Reference unit: 1 = 0.001 V (fixed)

c) The linear acceleration/deceleration time specified for T is the time it takes to accelerate or decelerate from the current voltage reference value to the target voltage reference value.

d) The T designation is a modal designation. When the T designation is omitted in the VCC command, the last executed T designation will be used. The default value for is 0 (no acceleration/deceleration). When program execution is completed or the program is interrupted, e.g., by an alarm, T will be reset to 0.

e) The linear acceleration/deceleration time can be specified separately for each axis. When multiple axes are specified simultaneously, however, one T designation is valid for all the referenced axes.

◀EXAMPLE▶ **Programming Example 1**

The following example shows the modal operation of the T designation.

Set the X axis and Y axis as voltage output axes (parameter PA506 b3 = 1).

ABS ; ↵

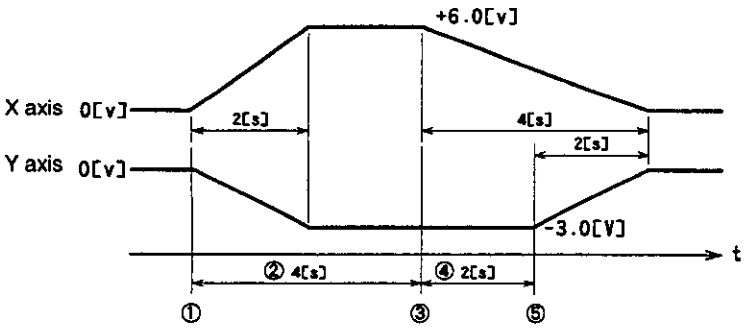
1 VCC X6Y-3.0T2.0 ; ↵ The linear acceleration/deceleration time T is valid for both the X axis and Y axis.

2 TIM P4.0 ; ↵

3 VCC X0T4.0 ; ↵ This T designation is valid only for the X axis.

4 TIM P2.0 ; ↵

5 VCC Y0 ; ↵ The T designation is omitted here, and so the T designation specified in 1, above, will be used.



◀EXAMPLE▶ **Programming Example 2**

The following example shows the affect of the absolute (ABS) and incremental modes (INC) and the affect of deceleration during acceleration.

Set the X axis as a servo axis (parameter PA506 b3 = 0).

Set the Y axis as a voltage output axis (parameter PA506 b3 = 1).

ABS ; ↵

1 MOV X100. ; ↵ — The VCC command is a preread command, and so is executed simultaneously with this MOV command.

2 VCC Y-4.0T2.0 ; ↵ — See above.

3 TIM P4.0 ; ↵

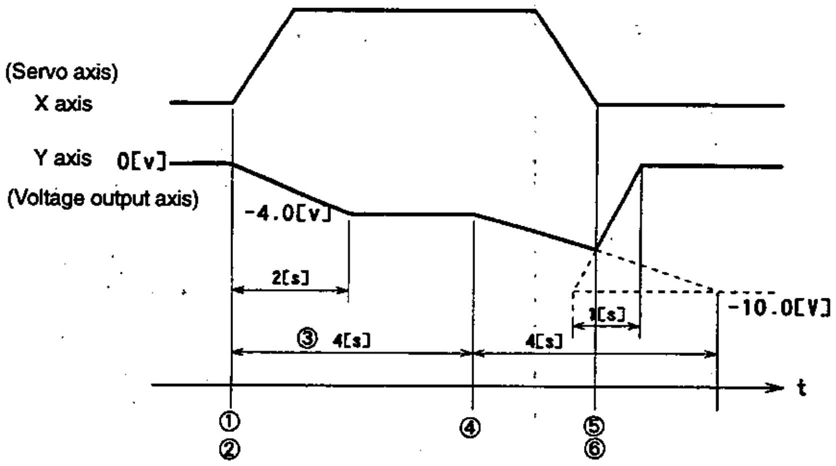
INC ; ↵

4 VCC Y-6.0T4.0 ; ↵ — Because the system is in incremental (INC) mode, the target reference value is -10 V.

ABS ; ↵

5 PFN ; ↵

6 VCC Y0 T1 ; ↵ — During acceleration by the command in 4, deceleration is started by this command. The acceleration is calculated based on the target reference value (-10 V) in 4.



9) Overrides

An override can be set for the voltage reference for voltage output axes in 16 steps between 0% and 100% or in increments of 0.1% between 0% and 3276.7%. The overrides can also be enabled and disabled.

a) Overrides for voltage references are enabled and disable in parameter P0016.

(1) Override: Enabled for MC control coils (16 steps)

Ver. B08

(2) Override: Enabled for MC link registers (in increments of 0.1%)

(3) Override: Disabled

The overrides are disabled in the default settings.

b) When an override is enabled, the percentage can be switched as shown below.

(1) **MC Control Coils (16 Steps)**
0, 1, 2, 4, 6, 8, 10, 20, 30, 40, 50, 60, 60, 80, 90, or 100 (%)

Ver. B08

(2) **MC Link Registers (0.1% Increments)**
0.0 to 3276.7 (%)

When an override is disabled, the speed will be 100% of the set value or reference value. If the override is not 100%, however, the linear acceleration/deceleration time will not agree with the specified value and the specified rate of acceleration (i.e., the analog voltage reference/linear acceleration/deceleration time) will be saved.

c) Voltage reference overrides are set using the following MC control coils and MC link registers.

Coil/Register	Use		References	
MC control coils 16 steps from 0% to 100%	Independent axes or manual	Axis 1	QN0145 to QN0148	N is the MC20 Module number (N = 1 or 2)
		Axis 2	QN0149 to QN0152	
		Axis 3	QN0153 to QN0156	
		Axis 4	QN0157 to QN0160	
MC link registers 0.0% to 3276.7% in 0.1% increments	Independent axes or manual	Axis 1	409911 and 409984	MC Modules 1 and 2 (default allocations)
		Axis 2	409912 and 409985	
		Axis 3	409913 and 409986	
		Axis 4	409914 and 409987	

Ver. B08

Set the MC link register to 10 times the override percentage when setting an override in 0.1% increments.

Refer to *3.2 MC Control Coil Functions* for details on MC control coils and to *1.5.6 Link Input Variables (#□□□□)* for details on MC link registers.

d) **Function and Timing of Overrides for Voltage References**

(1) The speed overrides for independent axis operation or for jogging or stepping operations in manual mode can be changed. An override can even be changed during operation.

(2) All operations, including programmed operation, for a control axis set as a voltage output axis will be affected by the override percentage set with the MC control coil or MC link register. The rapid traverse speed and maximum interpolation feed speed overrides are not used for voltage output axes.

Note (1) When the VOLTAGE OUTPUT (VCC) command is used in incremental mode, the target reference voltage may exceed ±10 V. If so, an alarm will be generated.
→ Alarm: 004 (address error)

(2) When the VOLTAGE OUTPUT (VCC) command is used together with a BRANCH (IF...GOTO) command, a command to set the voltage output to 0 may not be executed.

INFO

1) The voltage reference value and linear acceleration/deceleration time can be indirectly specified by means of variables.

2) The axis set in the voltage output axis parameter can be moved using the Teach Pendant (TB120) JOG (JOG) command. In this case, the voltage reference and the linear acceleration/deceleration depend on the parameters listed below.

- Voltage reference: Parameter PA202 (rapid traverse speed)
  Reference unit: 1 = 0.001 [V]

- Linear acceleration/deceleration:
  Parameter PA204 (linear acceleration/deceleration constant 1)

3) With an independent axis (A, B, C, or D) set as a voltage output axis, the analog voltage output function can be executed from the CPU Module's ladder logic program as follows:

a) Executed using an INDEPENDENT AXIS OPERATION (MVA, MVB, MVC, MVD) instruction.

b) Executed using the JOG (JOG) instruction.

4) The parameters that are enabled by selecting the voltage output function are listed in the following table.

a) Valid Parameters

Only the parameters listed in the following table are valid. Other parameters are invalid.

Parameter No.	Name
P0001 to P0004	Axis numbers 1 to 4 axis name specification
P0016	Override enabled/disabled
PA202	Rapid traverse speed
PA204	Linear A/D constant 1
PA501	Number of encoder pulses
PA502	Encoder pulse signal selection
PA503	One machine rotation/reference unit
PA504	Gear ratio (motor rev.)
PA505	Gear ratio (load rev.)
PA506-b0	Motor revolution direction (reverse revolution setting)
PA506-b3	Servo axis/voltage output axis selection
PA506-b4	Counter function enabled/disabled

A: Axis numbers (1 to 4)

b) Invalid Functions

Functions relating to parameters other than those listed in the previous table are invalid.

c) Valid Alarms

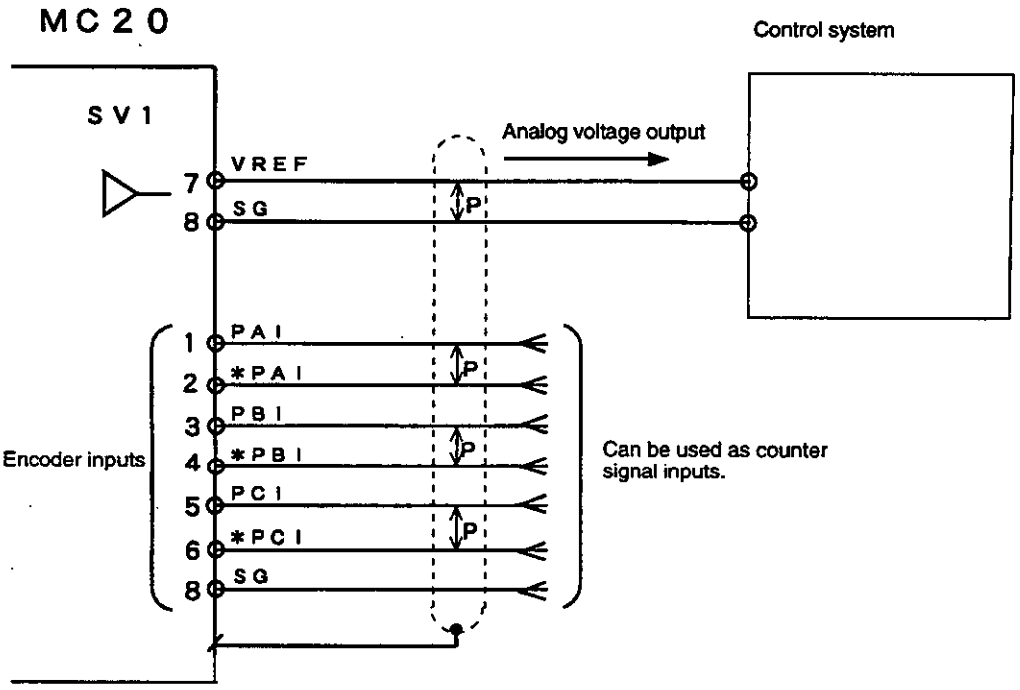
Alarms relating to parameters other than those listed in the previous table are invalid.

5) The VCC command can be used regardless of the ON/OFF status of the MC control relays (axis n servo ON status (PN0025 to PN0028)). This means that the VCC command cannot be cancelled by setting the servo status to OFF.

1.9.3 Counter Function

STEP 2

1) A controlled axis set as a voltage output axis does not perform position loop control. Therefore, the encoder inputs of the MC20 Module can be used as ordinary counter signal inputs. This is called the counter function. Its use is illustrated in the following diagram.



Note The PCI and *PCI signals are not used by the counter function, and they need not be connected.

2) Initial Preparations

a) To enable the counter function, first set the parameters shown in the following table.

Parameter No.	Name	Setting Range
PA506	Mode setting	
	b3: Axis type	0: Servo axis 1: Voltage output axis
	b4: Counter function	0: Disabled 1: Enabled

(1) Set the controlled axis type to 1: Voltage output axis.

(2) Set the controlled axis counter function to 1 (enabled), and then turn ON the power supply again. Doing this enables the encoder input for the controlled axis to be used as counter input.

b) Select the multiplication factor of the encoder pulse signal from the parameters in the following table. This sets the number of times each input pulse will be counted.

Parameter No.	Name	Setting Range
PA502	Encoder pulse signal selection	AB phase x 1
		AB phase x 2
		AB phase x 4

3) Connectors and Pin Arrangements for Counter Signal Inputs

The following table shows the connectors and pins that can be used for counter signal inputs.

- Servo connector: SVn

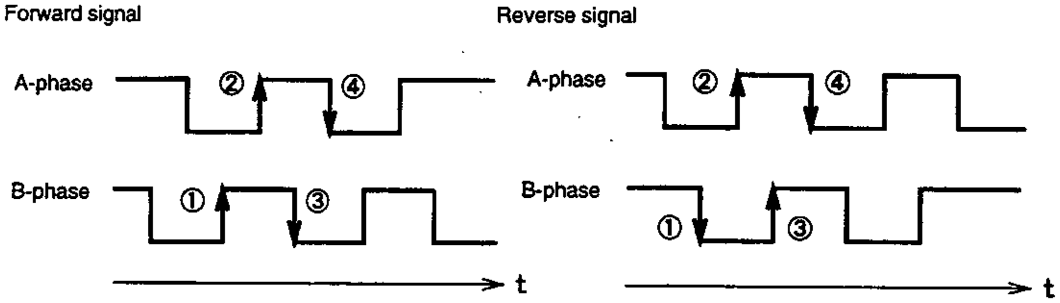
Connector model: 10220-L8A9-VE

Pin No.	Signal Name	Name
1	PAIn	A-phase signal input (axis n)
2	*PAIn	A-phase signal input (PAI reverse signal; axis n)
3	PBIn	B-phase signal input (axis n)
4	*PBIn	B-phase signal input (PBI reverse signal; axis n)
8	SG1n	Signal ground

Note n = Axis no. (1 to 4).

4) Counter Signal Input and Counting Methods

a) The input signal to the counter has the signal waveform shown in the following diagram.



1) When A-phase is OFF,
B-phase turns ON.

2) When B-phase is ON,
A-phase turns ON.

3) When A-phase is ON,
B-phase turns OFF.

4) When B-phase is OFF,
A-phase turns OFF.

1) When A-phase is OFF,
B-phase turns OFF.

2) When B-phase is OFF,
A-phase turns ON.

3) When A-phase is ON,
B-phase turns ON.

4) When B-phase is ON,
A-phase turns OFF.

b) Counting Method for Counter

The counter operates according to the encoder pulse signal selection parameters described previously, as shown in the following table. The numbers in the right column correspond to the numbers in the above diagram.

Parameter Setting	Meaning
AB phase x 1 (PA502=1)	1, 2, 3, and 4 are handled as one group, making a total of one count.
AB phase x 2 (PA502=1)	1 and 2 are handled as one group, and 3 and 4 are handled as one group, making a total of two counts.
AB phase x 4 (PA502=1)	1,2,3, and 4 are each handled as one group, making a total of four counts.

5) Storage Location for Count Values

The count values counted for the counter signal input are stored in the system variables listed in the following table. The count unit is one pulse.

System Variable	Meaning
#1009	Counter signal input count value for axis 1
#1010	Counter signal input count value for axis 2
#1011	Counter signal input count value for axis 3
#1012	Counter signal input count value for axis 4

Note With an axis set as the normal servo axis, the current position of the machine coordinate system is stored in system variables #1009 to #1012.

6) Presetting the Count Values

Use the following methods to preset the count values for the counter.

a) The count value for the counter can be preset to a set value using the CURRENT POSITION SET (POS) motion program command.

b) The count value for the counter can be preset to a set value using the COORDINATE SETTING (POS) ladder motion instruction.

c) The count value for the counter can be preset to a set value by executing the CURRENT POSITION SET (POS) motion program command as a single-block operation from the Teach Pendant (TB120).

1.9.4 Voltage Output Function Application Examples

STEP 2

Five typical application examples using the voltage output function explained in the previous section are given in this section.

1) Operating a Servo Amp Using Speed Control

a) This example shows an MC20 Module connected to a Yaskawa Servopack Servo Amp, and the use of speed control rather than the normal position loop control.

b) The connections between an MC20 Module and a Servo Amp are the same as for normal servo axes.

c) Set the MC20 Module parameters as follows:

PA506 b3 = 1: Sets the X axis as a voltage output axis.

PA506 b4 = 0: Disables the counter function (default value) for the X axis.

d) Set the Servo Amp user constants, and set the P-CON input function to P/PI control switching. For example, with the SGDA-□□□S Servopack, set the memory switches as follows:

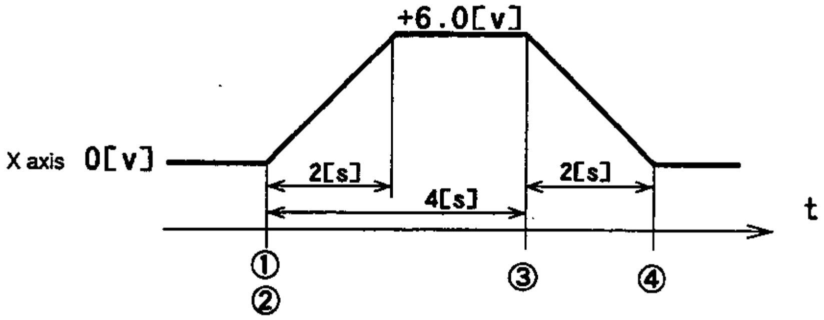
Cn-01 Bit A = 0, Bit B = 0

Cn-02 Bit 2 = 0

◀EXAMPLE▶ **Programming Example**

ABS ; ↵ Specifies absolute values for target references.

1	PCN X0 ; ⏎	Sets the Servo Amp to proportional/integration control (i.e., PI control) to improve the follow-up performance to the reference value.
2	VCC X6.0T2.0 ; ⏎	Operate the X axis at the speed references in the following diagram, using the voltage output function.
3	TIM P4.0 ; ⏎	See above.
4	VCC X0 ; ⏎	See above.
5	TIM P2.0 ; ⏎	See above.
6	PCN X1 ; ⏎	Sets the Servo Amp to proportional control (i.e., P control) to reliably stop the operation at a reference value of 0.



2) Operating an Inverter using Speed Control

a) This example shows an MC20 Module connected to an Inverter, and the use of speed control.

b) The connections between the MC20 Module and inverter are as follows:

Connect the VRDF□ speed reference output of the MC20 Module servo connector to the external frequency reference input of the inverter.

Short-circuit the ALM□ servo alarm signal input and the ALM0□ servo alarm ground of the servo connector.

Connect the MC20 Module to the inverter forward/reverse operation reference input from a GL120 and GL130 I/O Module.

c) Set the MC20 Module parameters as follows:

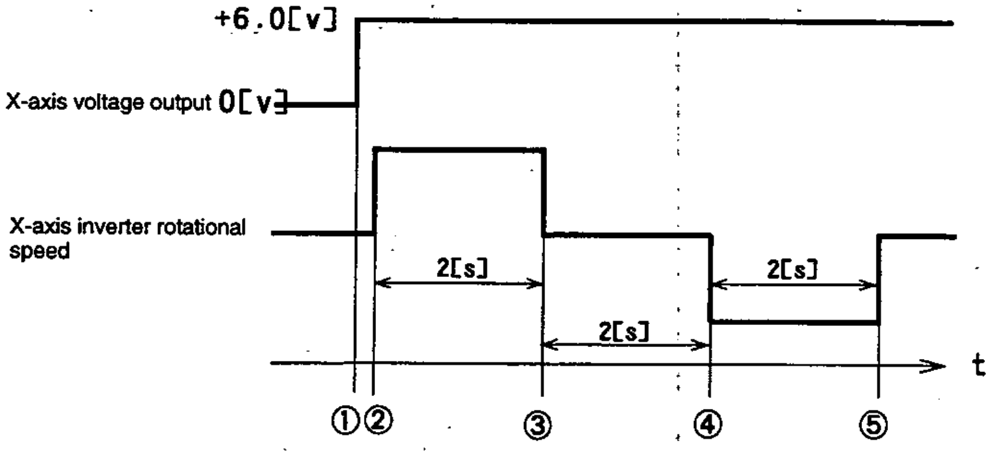
PA506 b3 = 1: Sets the X axis as a voltage output axis.

PA506 b4 = 0: Disables the counter function (default value) for the X axis.

◀EXAMPLE▶ **Programming Example**

In this example, the forward and reverse signals use output variables #03 and #04 respectively, and are output from a GL120 and GL130 I/O Module.

	ABS ; ↵	
1	VCC X6.0T0 ; ↵	Specifies the X axis operating speed. The inverter is not yet rotating at this time.
2	#03 = 1 ; ↵	Setting output variable. #03 to ON moves the inverter in forward rotation for 2 s.
	TIM P2.0 ; ↵	
3	#03 = 0 ; ↵	Setting output variable #03 to OFF stops the inverter for 2 s.
	TIM P2.0 ; ↵	
4	#04 = 1 ; ↵	Setting output variable #04 to ON moves the inverter in reverse rotation for 2 s.
	TIM P2.0 ; ↵	
5	#04 = 0 ; ↵	Setting output variable #04 to OFF stops the inverter.



3) Operating a Servo Amp Using Torque Control

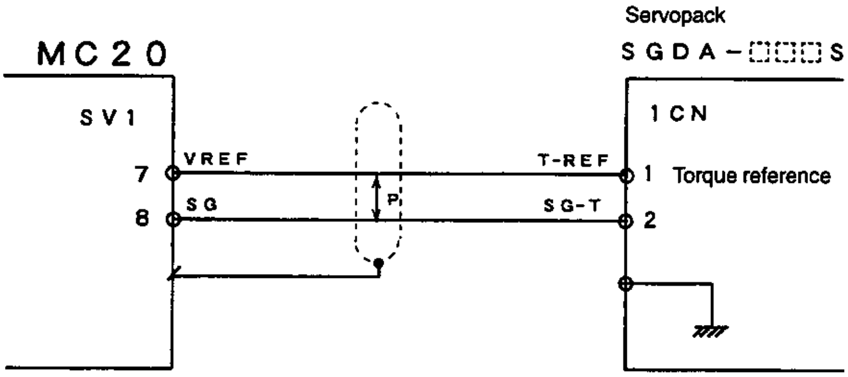
a) This example shows an MC20 Module connected to a Yaskawa Servopack Servo Amp, and the use of torque control rather than the normal position loop control.

b) The connections between an MC20 Module and a Servo Amp are as shown below. (When the Module is connected to an SGDA-□□□S Servopack)

Connect the VREF□ speed reference output of the MC20 Module servo connector to the Servopack torque reference (T-REF) input.

Connections other than the VREF□ speed reference output of the servo connector are the same as for a normal servo axis.

The MC20 Module PCON□ signal output is not used.



c) Set the MC20 Module parameters as follows:

PA506 b3 = 1: Set the X axis as a voltage output axis.

PA506 b4 = 0: Enables the counter function (default value) for the X axis.

d) Set the Servo Amp user constants as follows:

(1) Select torque control I using the memory switch for selecting the control type, as follows:
Cn-01 Bit A = 0: Select torque control I.
Bit B = 1: See above.

(2) Use the following user constants to set the limits for the torque reference gain and the motor speed in torque control I.
Cn-13 = 30 (0.1 V/rated torque) when 3 V is set as the rated torque
Cn-14 = 1000 (r/min) when 1,000 r/min is set as the upper limit

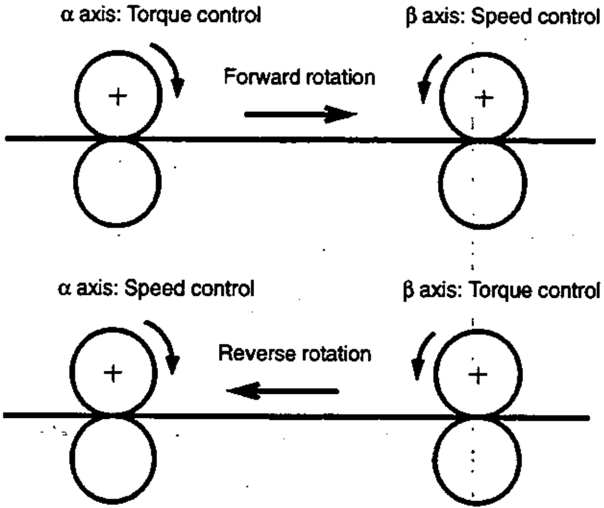
◀EXAMPLE▶ **Programming Example**

VCC X0.3 ; ↵	Generates a rated torque of 10% in the forward rotation direction for the X axis. As mentioned above, this occurs when the rated torque is set to 3 V.

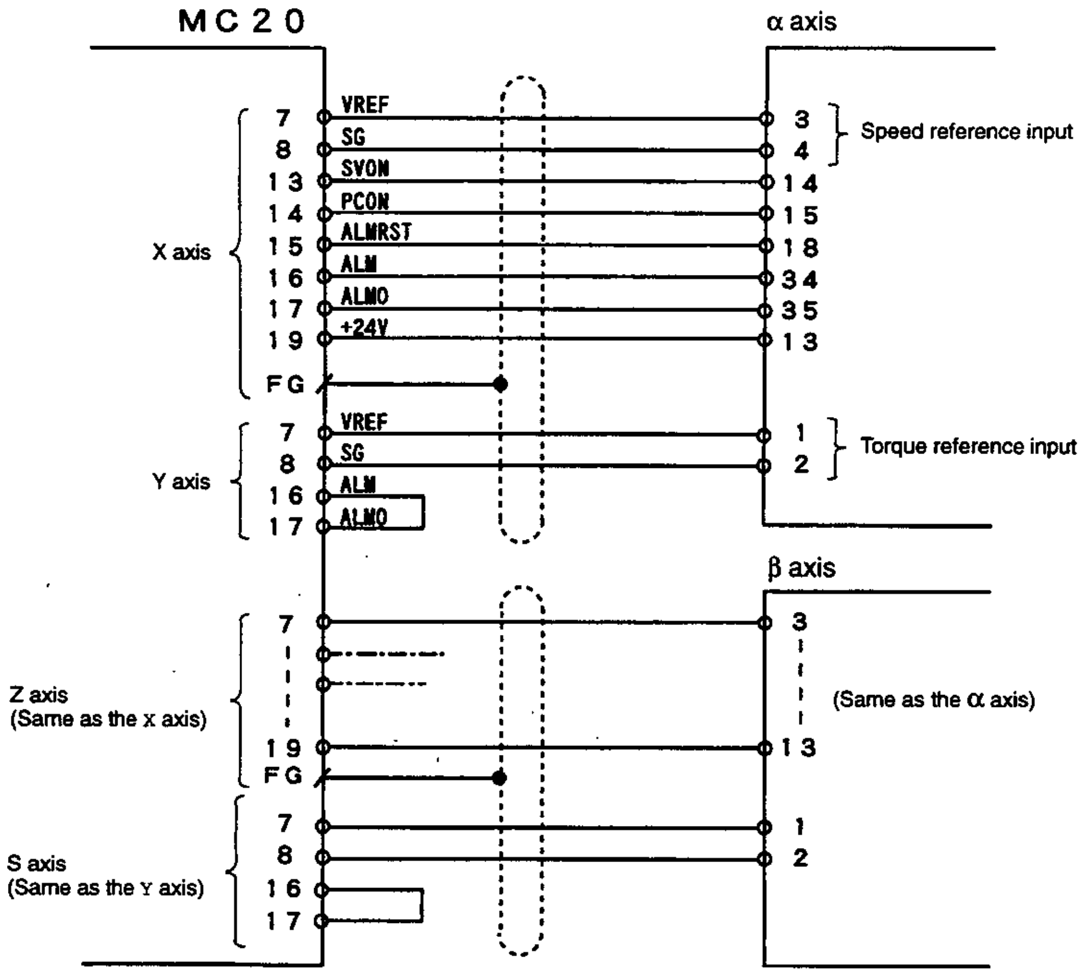
4) Operating a Servo Amp by Switching between Speed Control and Torque Control

a) This example shows the operation of an MC20 Module connected to a Yaskawa Servopack Servo Amp, and switching between speed control and torque control rather than the normal position loop control. In this example, it is assumed that the material, such as paper, is feed without slack. Torque control is performed by the axis on the feed-out side (α axis), and speed control is performed by the axis on the receiving side (β axis). To ensure that the paper feeds without slack, the torque applied to the torque control axis is the reverse of that in the feed direction.
With reverse rotation, the control methods for the feed-out side axis and the receiving

side axis are reversed. Use the PCON signal output function to switch the control method.



b) The following diagram shows the connections between an MC20 Module and a Servo Amp when the Module is connected to an SGDA-□□□S Servopack.



c) Set the X, Y, Z, and S axis parameters of the MC20 Module as follows:

PA506 b3=1: Sets a voltage output axis.

PA506 b4=0: Disables the counter function (default value).

d) Set the Servo Amp user constant as follows:

Select torque control II using the memory switch for selecting the control type, as follows:

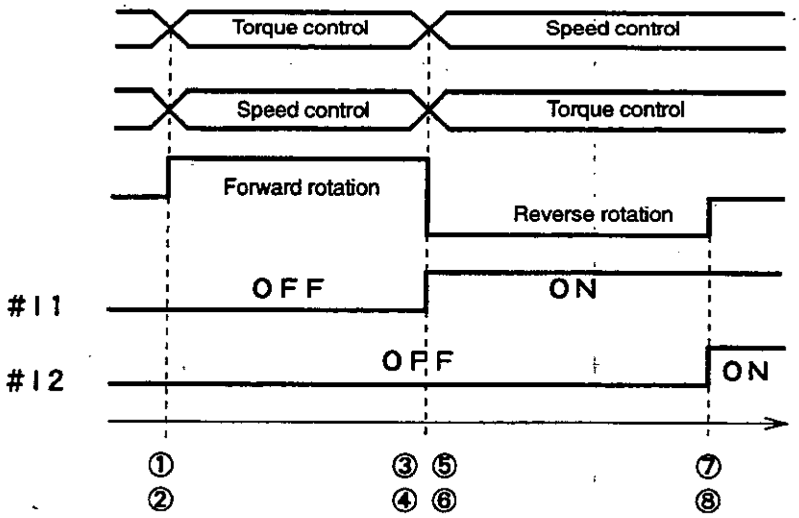
Cn-01	Bit A = 1:	Selects torque control II.
	Bit B = 1:	See above.

Torque control II operates by switching the speed control and the torque control.

◀EXAMPLE▶ **Programming Example 3**

The following example feeds a material, such as paper, without slack.

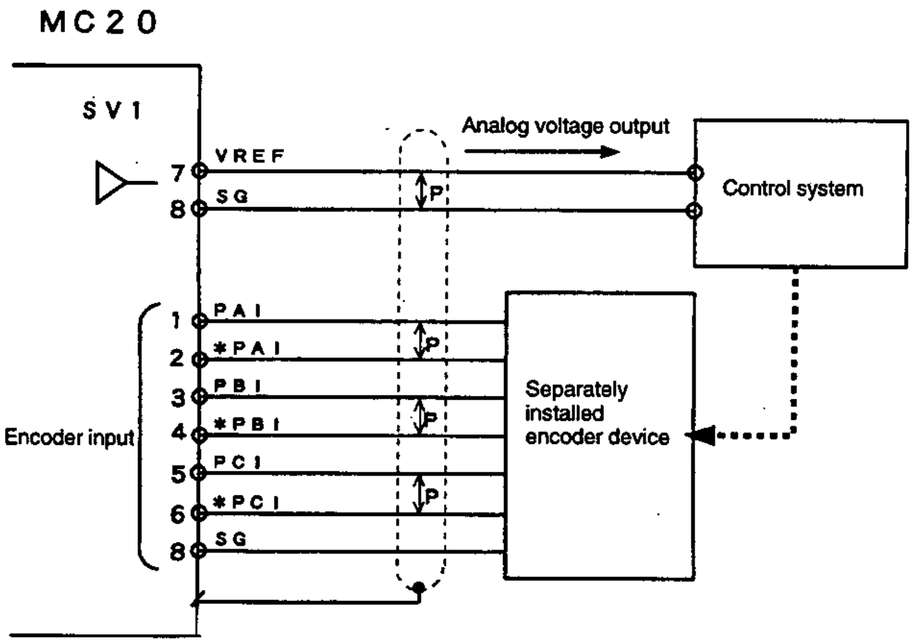
	ABS ; ⏎	
	VCC X0 Y0 Z0 S0 ; ⏎	
1	PCN X0 Z1 ; ⏎	Torque control is performed for the α axis and speed control is performed for the β axis by the PCON signal output function.
2	VCC Y-0.3Z6.0 ; ⏎	Applies a rated torque of 10% to the α axis (= Y axis control system), and the β axis (= Z axis control system) operates in forward rotation at the rated speed.
3	IOW #I1==1 ; ⏎	If input variable #11 is ON, proceeds to the next block.
4	VCC Y0 Z0 ; ⏎	
5	PCN X1 Z0 ; ⏎	Performs speed control for the α axis and torque control for the β axis.
6	VCC X-6.0S0.3 ; ⏎	The α axis (= X axis control system) operates in reverse rotation at the rated speed, and operates by applying a rated torque of 10% to the β axis (= Z axis control system).
7	IOW #I2==1 ; ⏎	
8	VCC X0 S0 ; ⏎	



5) Voltage Output Axis Encoder Input Used as a General Counter

a) This example shows how the encoder input of the controlled axis set as the voltage output axis is used as a general counter.

b) The connections between the MC20 Module and the encoder are shown in the following diagram. Connect the counter signal outputs of the separately installed Counter Module to the encoder signal inputs of the MC20 Module servo connector. For connection details, see *1.9.3 Counter Function.*



c) Set the MC20 Module parameters as follows:

PA506 b3 = 1: Sets the X axis as a voltage output axis.

PA506 b4 = 1: Enables the counter function for the X axis.

◀EXAMPLE▶ **Programming Example**

POS X0 ; ↵	Presets the counter to 0.
N100 IF #1009>=10 GOTO 200 ; ↵	Waits until the counter value reaches 10 or higher.
GOTO 100 ; ↵	See above.
N200 SET M23 ; ↵	Outputs an M code (M23) when the counter value reaches 10 or higher.

1.10 Advanced Commands for Axis Movement

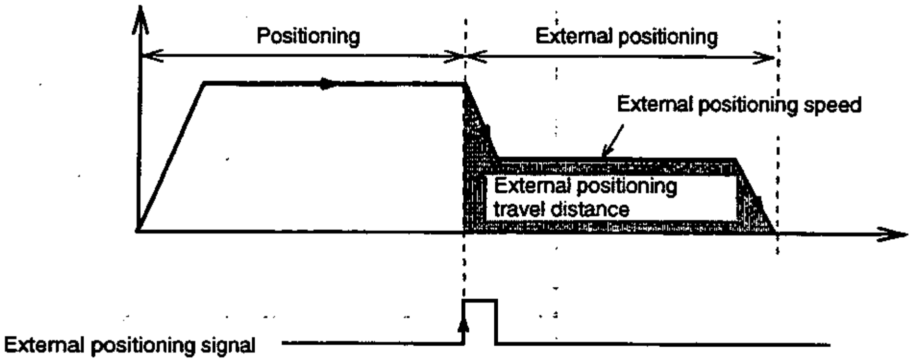
This chapter explains advanced commands for axis movement.

1.10.1 EXTERNAL POSITIONING (EXM)

STEP 2

1) Overview

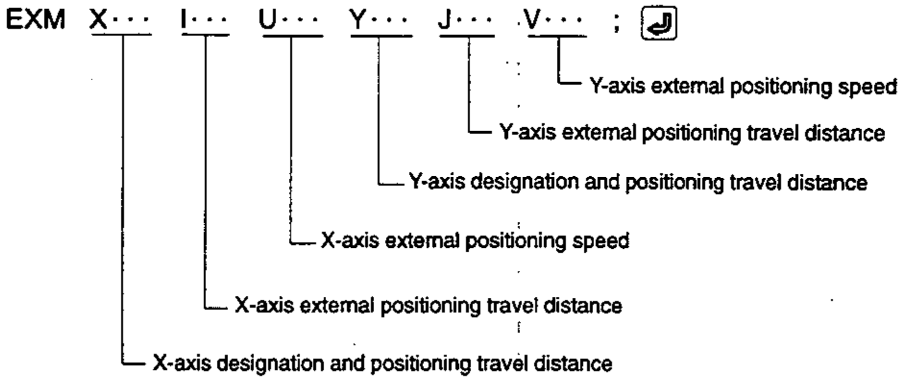
a) The following illustration is a conceptual diagram of the external positioning function.


Positioning
External positioning
External positioning speed
External positioning travel distance
External positioning signal

b) When the external positioning signal input turns ON while positioning is in progress, the external positioning function decelerates to the specified external positioning speed from that moment, and completes positioning by moving the axis only the specified external positioning travel distance.

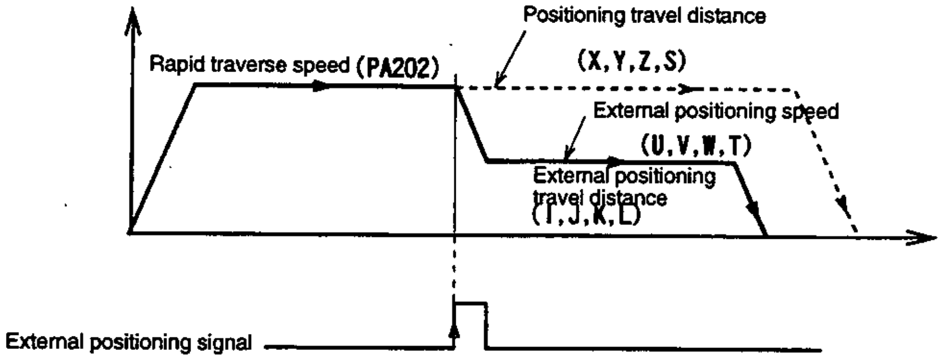
c) The external positioning function cannot be used in conjunction with the zero return function used with the home position limit switch signal, i.e., the ZEROn signal.

2) Specify the EXTERNAL POSITIONING (EXM) command as shown below. This example is for two axes.


EXM X··· I··· U··· Y··· J··· V··· ;
Y-axis external positioning speed
Y-axis external positioning travel distance
Y-axis designation and positioning travel distance
X-axis external positioning speed
X-axis external positioning travel distance
X-axis designation and positioning travel distance

This command starts normal positioning using the positioning travel distance of the specified axis. This speed depends on parameter PA202 (rapid traverse speed). When the external positioning signal input turns ON during this positioning operation (i.e., a rising signal is detected), the axis decelerates toward the specified external positioning speed, the axis moves only the specified external positioning travel distance, and the operation proceeds to the next block.

If the external positioning signal input does not turn ON, EXM completes normal positioning using the positioning travel distance for the specified axis, and the operation proceeds to the next block.


Positioning travel distance
(X,Y,Z,S)
Rapid traverse speed (PA202)
External positioning speed
(U,V,W,T)
External positioning travel distance
(I,J,K,L)
External positioning signal

3) The external positioning signal input specifications are as follows:

a) The external input signal EXPn is used as the external positioning signal. External positioning starts when a signal rise from OFF to ON is detected.

I/O Signal Connector:	I/O
Connector:	10259-52A2JL

Pin No.	Signal Name	Name
5	EXP1	External input signal for axis 1
13	EXP2	External input signal for axis 2
30	EXP3	External input signal for axis 3
38	EXP4	External input signal for axis 4
21 to 25	COM	Common input signal

b) After executing the EXM command, turn ON the external positioning signal. Even if the signal is turned ON before the EXM command is executed, external positioning will not be executed.

4) The EXM command is an axis move command, and preread parallel execution is not performed for it. If the previous block is a move command block, it is executed after execution of the previous block is completed.

5) The external positioning function can be specified for a maximum of four axes can be specified in the same block.

The following table shows the programming characters, the items controlled, and the setting ranges.

Programming Characters				Item Controlled	Absolute or Incremental Value	Setting Range
X	Y	Z	S	Positioning travel distance	The ABS or INC designation is used.	-99,999,999 to +99,999,999 (reference unit)
I	J	K	L	External positioning travel distance	Incremental value from the time the external positioning signal turns ON	-99,999,999 to +99,999,999 (reference unit)
U	V	W	T	External positioning speed	-	1 to maximum feed speed (PA201) (mm/min)

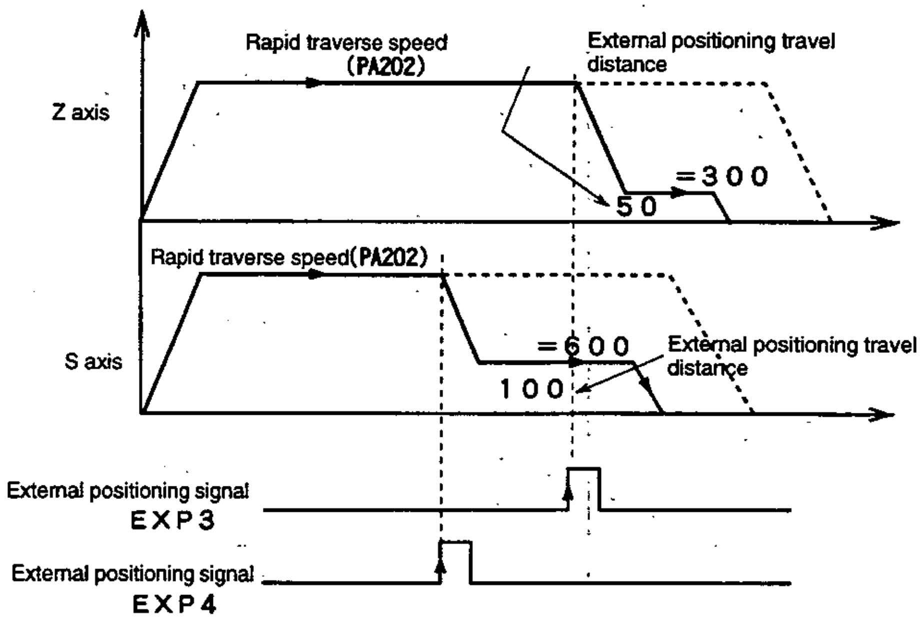
6) An in-position check is performed for the EXM command. That is, after detecting that the actual axis movement has entered the positioning-completed range, the operation proceeds to the next block.
If the EXM command is executed for multiple axes, execution will wait until the positioning of all the specified axes is completed and then the next block will be executed.

7) The rapid traverse speed overrides are valid for the external positioning function, including those for the external positioning speed. The rapid traverse speed overrides (%) can be switched using MC control coils ROV0 to ROV3.

◀EXAMPLE▶ **Programming Examples**

The following example executes external positioning for the Z and S axes.

```
1  EXM Z500.K50.W300.S400.L100.T600. ; ↵
```


Rapid traverse speed (PA202)
External positioning travel distance
Z axis
=300
50
Rapid traverse speed(PA202)
S axis
=600
100
External positioning travel distance
External positioning signal EXP3
External positioning signal EXP4

Execution proceeds to the next block after completing external positioning for axis Z.

Note (1) An absolute value that exceeds the set value in parameter PA503 (one machine rotation/reference unit) cannot be specified as the positioning travel distance set for X, Y, Z, or S for the controlled axis (PA506 b1=1) set as an infinite-length axis. If this occurs, the following alarm will be generated.

Alarm: 042 (ABSO specified range exceeded)

This restriction does not apply to the external positioning travel distance set for I, J, K, L.

(2) The EXM command cannot be specified for a controlled axis (PA506 b3 = 1) set as a voltage output axis. If EXM is specified for a voltage output axis, the following alarm will be generated.

Alarm: 024 (axis undefined)

(3) The zero return operation used by the ZEROn signal cannot be executed for another controlled axis while external positioning is being executed with the EXM command. Conversely, external positioning with the EXM command cannot be executed for another controlled axis while the zero return operation using the ZEROn signal is executing. If both functions are executed simultaneously, the following alarm will be generated.

Alarm: 044 (ZERO signal duplicated)

(4) Do not specify external positioning and trailing synchronous operation for the same axis at the same time. Correct operation cannot be guaranteed. Some specific examples of incorrect operation are provided below.

(a) Executing a trailing synchronous operation to make axis 1 the master axis while axis 1 is being externally positioned (before the external positioning signal is turned ON).

(b) Executing external positioning for axis 1 while a trailing synchronous operation to make axis 1 the master axis is being performed (before the marker sensor is turned ON).

If different controlled axes are specified, both functions will operate normally, such as in the following examples.

(c) Executing a trailing synchronous operation to make an axis other than axis 1 (i.e., axis 2, 3, or 4) the master axis while axis 1 is being externally positioned (before the external positioning signal is turned ON).

(d) Executing external positioning for axis 1 while a trailing synchronous operation to make an axis other than axis 1 (i.e., axis 2, 3, or 4) the master axis is being performed (before the marker sensor is turned ON).

INFO

1) Even if the external positioning signal turns ON during positioning deceleration for the positioning travel distance, the external positioning operation will execute normally if it is executed before positioning is completed.

2) If the external positioning signal is ON when the EXM command is executed, external positioning will not be performed. Only positioning due to the positioning travel distance will be executed.

3) If the speed specified as the external positioning speed (U, V, W, T) exceeds the set value in parameter PA201 (maximum feed speed), operation will proceed at the speed set in PA201.

4) The following variables can be used as the numeric designation for each EXM command character:

   - Common variables
   - System variables
   - Point tables used as variables
   - Link input variables
   - Link output variables

   Input and output variables cannot be used.

5) If external positioning is specified simultaneously for multiple axes, the commands may be longer than the maximum number of characters in one rung of the motion program. Use the variables listed above to reduce command length.

TERMS

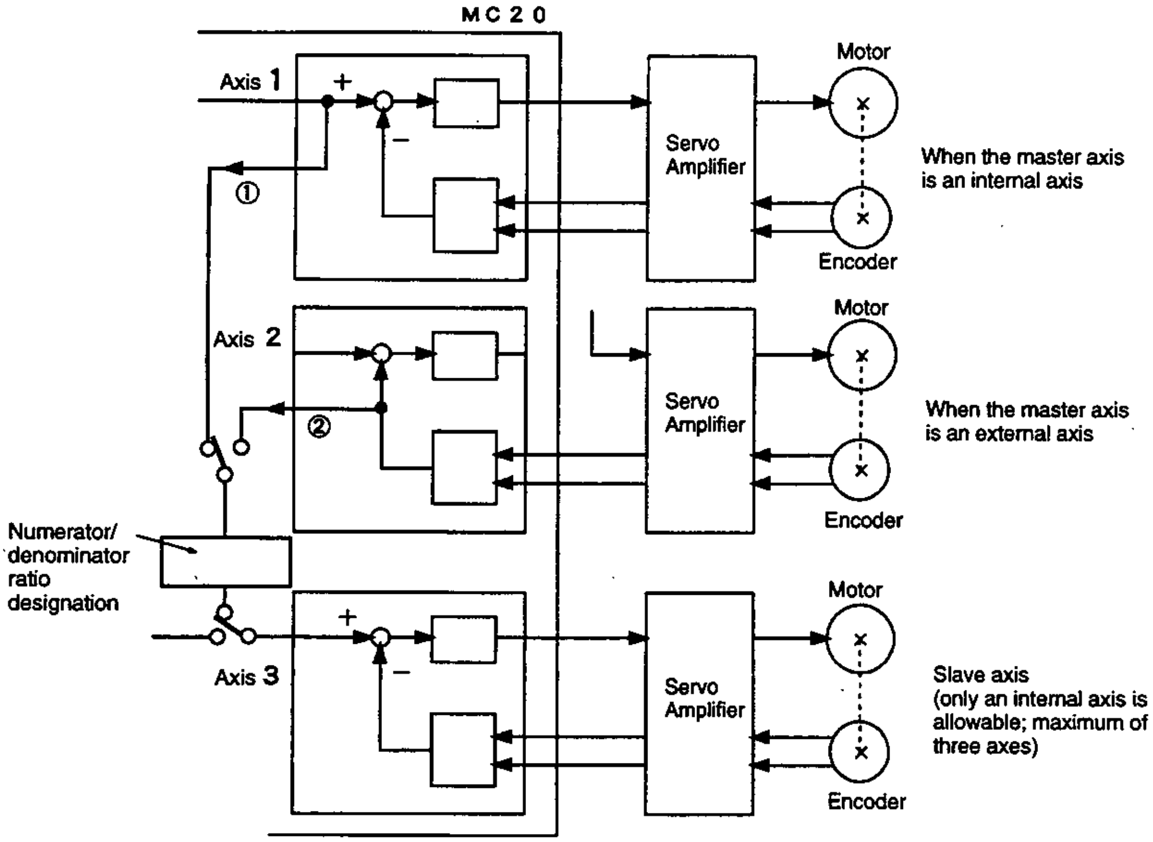
ZEROn signal

One method of executing zero return is to use the home position limit switch. The input signal for this home position limit switch signal is the ZEROn signal (replace n with axis number 1, 2, 3, or 4).

1.10.2 Ratio Operations (PGS, PGR)

1) Overview

a) The following illustration is a conceptual diagram of the ratio operation function.


MC20
Axis 1
+
−
①
Axis 2
②
Numerator/denominator ratio designation
+
−
Axis 3
Servo Amplifier
Servo Amplifier
Servo Amplifier
Motor
Encoder
Motor
Encoder
Motor
Encoder
When the master axis is an internal axis
When the master axis is an external axis
Slave axis (only an internal axis is allowable; maximum of three axes)

1: Reference travel distance per unit time (reference unit)

2: Travel distance from encoder per unit time (reference unit)

b) The controlled axis used to as the basis for ratio operations is called the master axis. An internal axis or an external axis can be specified as the master axis.

Internal axis: An axis controlled by the MC20 Module (X, Y, Z, S, A, B, C, or D).

External axis: An axis that cannot be controlled by the MC20 Module for which only the travel distance is input from the encoder (P, Q).

c) A controlled axis operated for the reference travel distance of the master axis multiplied by the specified ratio is called a slave axis. Only an internal axis can be specified as a slave axis.

d) The ratio is specified from a program using two integers; the numerator and denominator.

e) From the above, it follows that the operating speed of the slave axis will also be that of the master axis multiplied by a ratio. In this case, the override setting is 100%.

2) Specifying Master and Slave Axes

a) Up to a three slave axes can be specified for one master axis.

Example

	(Axis number)	(Ratio)
Master axis	= Axis 1	
Slave axis 1	= Axis 2	50/100
Slave axis 2	= Axis 4	40/50

In this example, if axis 1 operates at 100, then axis 2 will operate at 50.
If axis 1 operates at 100, then axis 4 will operate at 80.

b) Multiple master axes can also be set.

Example

	(Axis number)	(Ratio)
Master axis 1	= Axis 1	
Slave axis 1	= Axis 2	120/100
Master axis 2	= Axis 3	
Slave axis 2	= Axis 4	160/200

In this example, if axis 1 operates at 100, then axis 2 will operates at 120.
If axis 3 operates at 100, then axis 4 will operates at 80.

c) The same slave axis cannot have multiple master axes.
Therefore, the following cannot be specified.

Example: Not Possible

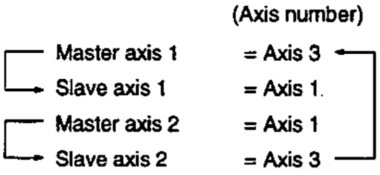
	(Axis number)
Master axis 1	= Axis 1
Master axis 2	= Axis 3
Slave axis	= Axis 4

d) A slave axis cannot function as its own master axis.
Therefore, the following cannot be specified:

Example 1: Not Possible

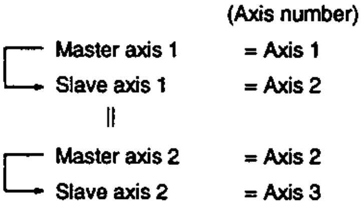
	(Axis number)
Master axis	= Axis 1
Slave axis	= Axis 1

Example 2: Not Possible



e) A slave axis cannot function as the master axis of another axis.
Therefore, the following cannot be specified:

Example: Not Possible



3) Preparations

a) Using an Internal Axis as the Master Axis

When an internal axis is to be used as the master axis, no special parameters need to be set for ratio operations. The automatic acceleration/deceleration parameters are valid as normal.

b) Using an External Axis as the Master Axis

When an external axis is to be used as the master axis, be sure to set the axis name and machine system parameters listed in the following table.

Parameter No.	Name
P0001 to P0004	Axis 1 to axis 4 names
PA201	Maximum feed speed
PA501	Number of encoder pulses
PA502	Encoder pulse signal selection
PA503	One machine rotation/reference unit
PA504	Gear ratio (motor revolutions)
PA505	Gear ratio (load revolutions)

A: Axis numbers (1 to 4)

c) Slave Axes

For ratio operations, no special parameters need to be set for the slave axes.
With a slave axis, the acceleration/deceleration parameters for automatic acceleration/deceleration will be disabled. The following filter selection parameters and their related parameters are valid.

Parameter No.	Name
PA209	Time constant for exponential acceleration/deceleration
PA210	Bias speed for exponential acceleration/deceleration
PA211	Time constant for moving average acceleration/deceleration
PA217	Filter selection

A: Axis numbers (1 to 4)

4) Specify the RATIO OPERATION (PGS) command as follows:

```
PGS  X···  I···  Y···  J···  Z···  K···  MS□  ;  ↵
                             (S··· L···)
      |     |     |     |      |           └─ Master axis designation
      |     |     |     |      └─ Third slave axis and ratio designation (see below)
      |     |     |     └─ Y axis ratio denominator
      |     |     └─ Slave axis (Y) and the ratio numerator
      |     └─ X axis ratio denominator
      └─ Slave axis (X) and the ratio numerator
```

When the command is executed, the slave axes will operate for the travel distances based on the travel distance of the master axis specified for MS□ and the ratios (numerator/denominator) specified for each slave axis.

The designations for the master axis are as follows:

- MSX, MSY, MSZ, MSS (basic axes): The X, Y, Z, or S axis is specified as the master axis.
- MSA, MSB, MSC, MSD (independent axes): The A, B, C, or D axis is specified as the master axis.
- MSP, MSQ (external axes): The P or Q axis is specified as the master axis.

Up to three of the following axis can be specified as the slave axes: X, Y, Z, and S.

5) The PGS command is a preread command that is executed in parallel with a move command. To execute the PGS command after a move command in the previous block is completed, specify the IN-POSITION CHECK (PFN) command in the previous block. The PGS command is completed as soon as execution starts, and the operation proceeds to the next block.

6) Specify the master axis and slave axis combinations and ratio changes with the PGS command.

7) The slave axis designation is modal. If the designation is not changed, the previous designation will be used. The master axis designation is non-modal. When the PGS command is specified, be sure to specify the master axis each time.

8) The ratio (numerator/denominator) is specified as follows:

a) Use addresses X, Y, Z, and S to specify the slave axis name and the ratio numerator. Use addresses I, J, K, and L to specify the ratio denominator corresponding to the X, Y, Z, and S axes respectively.

b) The setting ranges for the numerators and denominators are as follows:

Numerator setting range:	0 to ±9999
Denominator setting range:	1 to 9999

c) Besides normal integer values, the numerator and denominator can also be specified using the following variables:

Common variables

Link input variables

Point tables used as variables

d) If 0 is specified as the ratio numerator, that axis will temporarily stop for the ratio operation. To restart the ratio operation, use the PGS command to specify a value other than 0 for the numerator. When the PGS command is executed, the ratio operation continues until it is cancelled by one of the following commands.

RATIO OPERATION CANCEL (PGR) command

PROGRAM END (END) command

PROGRAM RUN (MVL) command forced end

e) The ratio can be changed by specifying a new PGS command while a PGS command is being executed. As a result, the newly specified PGS command is executed by overwriting the PGS command that was being executed up to that time. When the ratio is changed, the master axis can also be changed.

f) When the ratio is changed and the same slave axes are specified, the new ratios are valid. For the master axis designation, the new master axis specified is valid.

g) The ratio designation is an absolute value. An incremental value cannot be specified.

h) The ratio designation is valid while the PGS command is being executed, provided no change is specified.

9) Specify the RATIO OPERATION CANCEL (PGR) command as follows:

PGR ; ↵	Cancels the ratio operation for all axes.
or	
PGR X0 Y0 ; ↵	Cancels the ratio operation for the specified axes.

The command cancels the ratio operation for all axes or for the specified slave axes. To specify an axis to cancel, specify 0 together with the axis address. A controlled axis for which the ratio operation has been cancelled returns to a normal controlled axis.

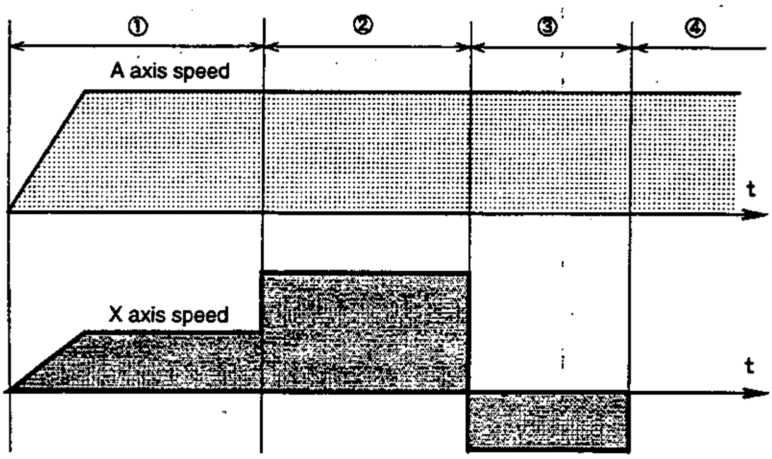
The PGR command is a preread command that is executed in parallel with a move command. To execute the PGR command after a move command in the previous block is completed, specify the IN-POSITION CHECK (PFN) command in the previous block. Nothing will be done if the PGR command is specified for a controlled axis for which ration operation is not being performed.

◀EXAMPLE▶

Programming Example 1: Single Slave Axis

The following example shows operation when the X axis is made a slave axis and the automatic acceleration/deceleration filter selection parameter (PA217) is set to 0 (none).

1	PGS X50 I100 MSA ; ↵	Master axis = independent axis A
2	PGS X100 I100 MSA ; ↵	Changes the ratio.
3	PGS X-50 I100 MSA ; ↵	Changes the ratio and the rotation direction.
4	PGR ; ↵	Cancels the ratio operation for all axes.



(The speed for the X axis is the reference value.)

◀EXAMPLE▶ **Programming Example 2: Multiple Slave Axes**

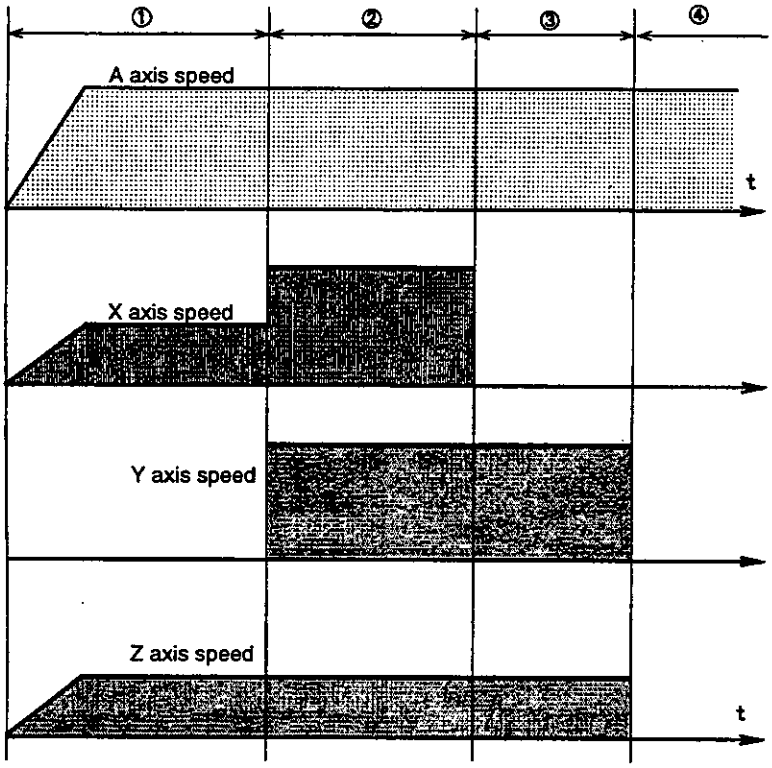
The following example shows operation when the X, Y, and Z axes are made slave axes and the automatic acceleration/deceleration filter selection parameter (PA217) is set to 0 (none).

1 PGS X5 I10 Y0 J10 Z5 K10 MSA ; ↵

2 PGS X10 I10 Y10 J10 MSA ; ↵ Changes the ratio.

3 PGR X0 ; ↵ Cancels the X axis ratio operation.

4 PGR ; ↵ Cancels the ratio operation for all axes.



(The speed for each of the axes is the reference value.)

◀EXAMPLE▶ **Programming Example 3: Changing the Master Axis**

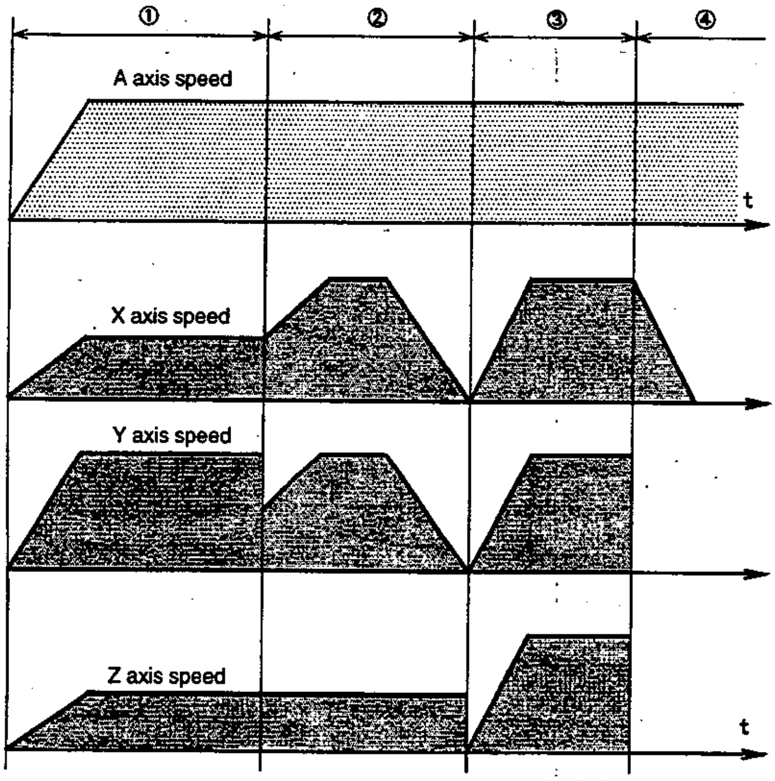
The following example shows operation when the automatic acceleration/deceleration filter selection parameter (PA217) for the Y and Z axes is set to 0 (none).

1 PGS Y100 J100 Z50 K100 MSA ; ⏎

2 PGS Y100 J100 MSX ; ⏎ Changes the master axis of axis Y.

3 PGS Z100 K100 MSX ; ⏎ Changes the master axis of axis Z.

4 PGR ; ⏎ Cancels the ratio operation for all axes.



(The speed for each of the above axes is the reference value.)

10) Overrides

a) Overrides can be set for the master and slave axes in ratio operation in 16 steps between 0% and 100% or in increments of 0.1% between 0% and 3276.7%. The feed speed for the slave axis will be as follows when overrides are set for both the master and slave axes:

Slave axis speed = (Master axis speed) x (Master axis override) x (Ratio)
x (Slave axis override)

The overrides can also be enabled and disabled.

b) Overrides for the master and slave axes are enabled and disable in parameter P0016.

(1) Override: Enabled for MC control coils (16 steps)

Ver. B08

(2) Override: Enabled for MC link registers (in increments of 0.1%)

(3) Override: Disabled

The overrides are disabled in the default settings.

c) When an override is enabled, the percentage can be switched as shown below for the basic axis feed speed and independent axis operation and manual operation speed for the master and slave axes.

(1) **MC Control Coils (16 Steps)**
0, 1, 2, 4, 6, 8, 10, 20, 30, 40, 50, 60, 60, 80, 90, or 100 (%)

Ver. B08

(2) **MC Link Registers (0.1% Increments)**
0.0 to 3276.7 (%)

When an override is disabled, the speed will be 100% of the set value or reference value.

d) When an override is enabled, the percentage can be switched as shown below for the basic axis interpolation feed speed for the master axis.

(1) **MC Control Coils (16 Steps)**
0, 1, 2, 4, 8, 10, 20, 40, 60, 80, 100, 120, 140, 160, 180, or 200 (%)

Ver. B08

(2) **MC Link Registers (0.1% Increments)**
0.0 to 3276.7 (%)

When an override is disabled, the speed will be 100% of the set value or reference value.

e) The overrides for the master axis are set using the following MC control coils and MC link registers when the master axis is an interior axis.

Ver. B08

Coil/Register		Master Axis		References	
MC control coils	16 steps from 0% to 100%	Independent axes or manual	Axis 1	QN0145 to QN0148	N is the MC20 Module number (N = 1 or 2)
			Axis 2	QN0149 to QN0152	
			Axis 3	QN0153 to QN0156	
			Axis 4	QN0157 to QN0160	
		Rapid traverse speed		QN0137 to QN0140	
		Maximum interpolation feed speed		QN0141 to QN0144	
MC link registers	0.0% to 3276.7% in 0.1% increments	Independent axes or manual	Axis 1	409911 and 409984	MC Modules 1 and 2 (default allocations)
			Axis 2	409912 and 409985	
			Axis 3	409913 and 409986	
			Axis 4	409914 and 409987	
		Rapid traverse speed		409909 and 409982	
		Maximum interpolation feed speed		409910 and 409983	

f) The overrides for the slave axis are set using the following MC control coils and MC link registers.

Coil/Register		Master Axis		References	References
MC control coils	16 steps from 0% to 100%	Independent axes or manual	Axis 1	QN0145 to QN0148	N is the MC20 Module number (N = 1 or 2)
			Axis 2	QN0149 to QN0152	
			Axis 3	QN0153 to QN0156	
			Axis 4	QN0157 to QN0160	
		Rapid traverse speed		QN0137 to QN0140	
MC link registers	0.0% to 3276.7% in 0.1% increments	Independent axes or manual	Axis 1	409911 and 409984	MC Modules 1 and 2 (default allocations)
			Axis 2	409912 and 409985	
			Axis 3	409913 and 409986	
			Axis 4	409914 and 409987	
		Rapid traverse speed		409909 and 409982	

Set the MC link register to 10 times the override percentage when setting an override in 0.1% increments.

Refer to *3.2 MC Control Coil Functions* for details on MC control coils and to *1.5.6 Link Input Variables (#□□□□)* for details on MC link registers.

Note (1) If a controlled axis specified as a slave axis is being moved by the previous block command when the PGS command is executed, the following alarm will be generated.

Alarm: 041 (axis duplicated)

If the previous block is a move command, the PGS command will be executed when execution of the previous block starts. Therefore, in the following programming example, alarm 041 will be generated.

Programming Example: Alarm 041 Generated

```
N10    INC MOV X1000 ; ↵
N20    PGS X100I100 MSA ; ↵
N30    TIM P15000 ; ↵
N40    END ; ↵
```

Axis X, which is the slave axis of the RATIO OPERATION (PGS) command, is being moved by the N10 block command, so an alarm is generated.

In the next programming example, however, no alarm is generated.

Programming Example: No Alarm is Generated, and Current Position X = 1000

```
N10    ABS MOV X1000 ; ↵
N20    PGS X100I100 MSA ; ↵
N30    TIM P15000 ; ↵
N40    END ; ↵
```

The N10 move command is completed immediately after the execution start. Therefore, axis X, which is the slave axis of the RATIO OPERATION (PGS) command, is not being moved and operation is normal.

(2) A command other than PGS cannot be specified for the slave axis during ratio operation. If a command other than PGS is specified for a slave axis, the following alarm will be generated.
Alarm: 041 (axis duplicated)

(3) If an unused axis is specified for the master axis, the following alarm will be generated.
Alarm: 024 (axis undefined)

(4) If there is no master axis specified in the PGS command, the following alarm will be generated.
Alarm: 004 (address error)

(5) If the master axis designations MSA to MSD are specified separately (example: MSA;↵), nothing will be executed.
If, however, the master axis designations MSA to MSD and the axis addresses are specified simultaneously, the following alarm will be generated.
Alarm: 004 (address error)

(6) If an address other than X, Y, Z, S, I, J, K, L, O, or N is specified in the PGS command, the following alarm will be generated.
Alarm: 004 (address error)

(7) If there is no ratio designation as listed below, the following alarm will be generated.
Alarm: 004 (address error)

(a) If the ratio numerator or denominator is outside the setting range

(b) If the ratio numerator or denominator is not specified

(8) If the final slave axis speed exceeds the value set in the maximum feed speed parameter (PA201), the following alarm will be generated.
Alarm: A11 (speed trailing disabled)

(9) If a master axis rapid traverse speed (PA202) is set that exceeds the set value for the slave axis maximum feed speed (PA201) when the PGS command is executed, the following alarm will be generated.
Alarm: A11 (speed trailing disabled)

INFO

1) The ratio operation is performed for a slave axis only while the motion program is being executed. Therefore, if the mode is changed from automatic mode to another operation mode by the MODE SET (MOD) ladder motion instruction while the motion program is being executed, the slave axis will decelerate to a stop. To restart the ratio operation, first use the MODE SET (MOD) instruction to reset automatic mode, and then start program execution.

2) Even if the master axis operates during a pause in program execution (feed hold), the slave axis will not operate. Therefore, as a result of the pause operation, the travel distance of the master axis and the slave axis may not be proportional.

3) The operation of a slave axis for which a ratio operation is being performed during a pause for the SINGLE BLOCK MODE (SMD) ladder motion command is as follows:

   a) When the master axis is a basic axis (X, Y, Z, S): Because the master axis also pauses during a single-block pause, the slave axis will not operate.

   b) When the master axis is an independent axis (A, B, C, D): If the master axis is still operating during a single-block pause, the slave axis will operate.

4) Besides the RATIO CANCEL (PGR) command, the ratio operation can also be cancelled by executing the PROGRAM END (END) motion command.

5) To operate an independent axis as a slave axis, specify using the INDEPENDENT AXIS OPERATION (MVA to MVD) ladder motion instruction.

6) A ratio operation cannot be specified from the Teach Pendant (TB120).

7) When the master axis is an internal axis and the master axis is stopped, the behavior of the master axis and the slave axis may differ. For example, if the filter selection parameter is set to anything but "none" for the master axis, and the filter selection for the slave axis is set to 0 (none), the slave axis operations may be completed sooner than the master axis operations.

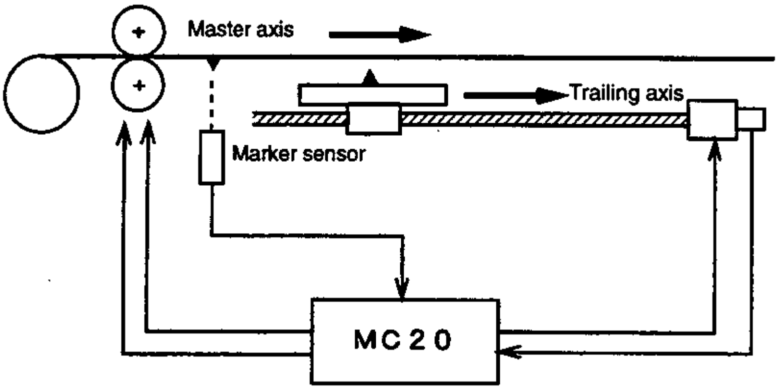
8) When the PGR command is executed, any coordinate system shift that occurred as a result of the previously executed manual operation will be cancelled.

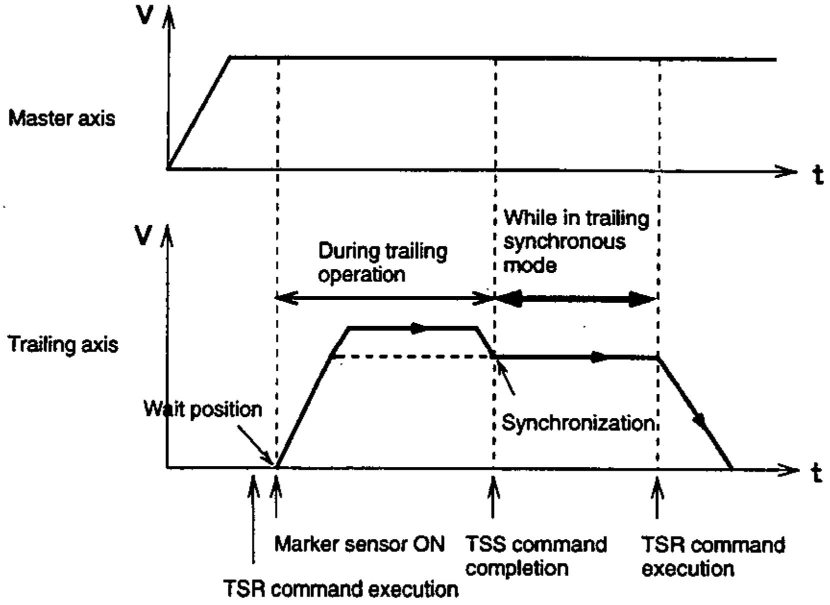
1.10.3 Trailing Synchronous Operations (TSS, TSR)

STEP 2

1) Overview

a) The following illustration is a conceptual diagram of the trailing synchronous operation function.





b) The controlled axis used as a basis for trailing synchronous operations is called the master axis. An internal axis or an external axis can be specified as the master axis.

Internal axis: An axis controlled by the MC20 Module (X, Y, Z, S, A, B, C, or D).

External axis: An axis that cannot be controlled by the MC20 Module for which only the travel distance is input from the encoder (P, Q).

c) An axis that imitates the master axis and synchronizes trailing is called a trailing axis. Only the internal axes X, Y, Z, and S can be specified as trailing axes.

d) There is one and only one master axis, and one and only one trailing axis for the trailing synchronous operation.

e) The typical operation process for trailing synchronous operation is as follows:

(1) The trailing axis is moved to a waiting position by the motion program.

(2) The master axis is started.

(3) The TRAILING SYNCHRONOUS OPERATION (TSS) command is executed. The trailing axis enters the trailing synchronous operation wait status.

(4) When the marker sensor that detects a specific position on the master axis is turned ON, the trailing axis starts the trailing operation.

(5) When the trailing operation is completed and the operation enters synchronized status, the TSS command is completed and the trailing axis enters trailing synchronous mode.

(6) While in trailing synchronous mode, any motion command for a controlled axis, including the trailing axis, can be executed. This means that work can proceed between the master axis and the slave axis during synchronous operations.

(7) TRAILING SYNCHRONOUS OPERATION CANCEL (TSR) is executed. Trailing synchronous mode ends and the trailing axis decelerates to a stop. The trailing synchronous operation function is then cancelled.

(8) If steps (1) through (8) are executed again, a second trailing synchronous operation will be performed.

2) Preparations

a) Using an Internal Axis Set as a Servo Axis as the Master Axis

If an internal axis set as a servo axis (PA506 b3 = 0) is used as the master axis, no special parameters need to be set for trailing synchronous operation. Set the normal parameters for the servo axis. The automatic acceleration/deceleration parameters are valid as normal.

b) Using an Internal Axis Set as a Voltage Output Axis as the Master Axis

If an internal axis set as a voltage output axis (PA506 b3 = 1) is used as the master axis, set the parameters shown in the following table.

Parameter No.	Name
PA506 b4 = 1	Encoder input used as counter
PA201	Maximum feed speed
PA501	Number of encoder pulses
PA502	Encoder pulse signal selection
PA503	One machine rotation/reference unit
PA504	Gear ratio (motor revolutions)
PA505	Gear ratio (load revolutions)

A: Axis numbers (1 to 4)

c) Using an External Axis as the Master Axis

When an external axis is used as the master axis, be sure to set the axis name and machine system parameters listed in the following table.

Parameter No.	Name
P0001 to P0004	Axis 1 to 4 names (external axis = P, Q)
PA201	Maximum feed speed
PA501	Number of encoder pulses
PA502	Encoder pulse signal selection
PA503	One machine rotation/reference unit
PA504	Gear ratio (motor revolutions)
PA505	Gear ratio (load revolutions)

A: Axis numbers (1 to 4)

d) Trailing Axis

No special parameters need to be set for the trailing axis used for trailing synchronous operation. Set the normal parameters for the servo axis. The automatic acceleration/deceleration in the following table is automatically applied to the trailing axis in trailing synchronous mode. It is applicable only to the trailing operation component.

Item	Acceleration Time	During Trailing Synchronous Mode	Deceleration Time
A/D type	Single step linear A/D	No acceleration/deceleration	Single step linear acceleration/deceleration
Filter selection	None	None	None

Therefore, the set value of the parameter in the following table is applied to the trailing axis acceleration/deceleration time.

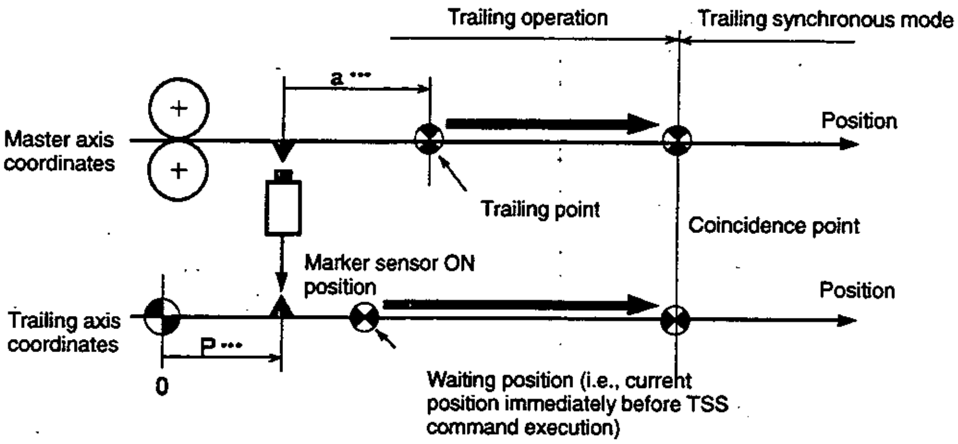
Parameter No.	Name
PA204	Linear A/D constant (1))

Automatic acceleration/deceleration according to the parameter setting is applied for the move command to the trailing axis in trailing synchronous mode.

3) Specify the TRAILING SYNCHRONOUS OPERATION (TSS) command as follows:

```
TSS  a···  P···  MS□  ; ⏎
     |     |      |
     |     |      └─ Master axis designation
     |     |
     |     └─ Marker sensor ON position
     |
     └─ Trailing axis designation and distance from marker sensor ON position to trailing point (incremental value)
```

When the command is executed, the specified trailing axis (indicated by "a") enters the trailing synchronous operation wait status. When the marker sensor for the master axis turns ON, the trailing axis starts from the current position, i.e., the waiting position, and performs trailing operation for the trailing point on the master axis. If the trailing axis waiting position is synchronized with the master axis trailing point, the trailing operation is completed and the trailing axis enters trailing synchronous mode. The TSS command is completed, and the operation proceeds to the next block.


Trailing operation
Trailing synchronous mode
a ···
Position
Master axis coordinates
Trailing point
Coincidence point
Marker sensor ON position
Position
Trailing axis coordinates
P ···
0
Waiting position (i.e., current position immediately before TSS command execution)

The designations for the master axis are shown below. Set only one of the following axes.

- MSX, MSY, MSZ, MSS (basic axes): The X, Y, Z, or S axis is specified as the master axis.

- MSA, MSB, MSC, MSD (independent axes): The A, B, C, or D axis is specified as the master axis.

- MSP, MSQ (external axes): The P or Q axis is specified as the master axis.

One of the following axes can be specified as the trailing axis ("a."): X, Y, Z, and S axes

4) The marker sensor input signal specifications are as follows:

a) The external input signal (EXPn) is used as the marker sensor input signal.
The marker sensor input signal of the controlled axis specified as the master axis detects a signal rise from OFF to ON and starts the trailing operation.

I/O Signal Connector: I/O
Connector: 10259-52A2JL

Pin No.	Signal Name	Name
5	EXP1	External input signal for axis 1
13	EXP2	External input signal for axis 2
30	EXP3	External input signal for axis 3
38	EXP4	External input signal for axis 4
21 to 25	COM	Common input signal

b) After executing the TSS command, turn ON the marker signal. Trailing synchronous operation will not be started if the marker signal is turned ON before the TSS command is executed.

5) The TSS command is a preread command that is executed parallel to a move command. If the previous block is a move command block and it is desired to execute it after the previous block is completed, specify the IN-POSITION CHECK (PFN) command in the previous block. The TSS command is completed as soon as the trailing axis enters trailing synchronous mode, and execution will proceed to the next block.

6) One and only one master axis, and one and only one trailing axis must be specified for the trailing synchronous operation.

7) When the TSS command is specified, be sure to specify the master axis, trailing axis, and character P each time. These designations are non-modal.

8) The setting ranges for the specified values are as follows:

a) Distance from Marker Sensor ON Position to the Trailing Point

The position of the trailing point on the master axis is specified as an incremental value from the position where the marker sensor turns ON. The distance is specified using the character P value for the trailing axis. This value is always an incremental value, and is not affected by absolute or incremental mode.

Setting range:	0 to +99,999,99
Unit:	Reference unit

b) Marker Sensor ON Position

The corresponding coordinate positions on the trailing axis in the marker sensor ON position are specified as character P data. The relationship between the master axis and trailing axis coordinates is dependent on this value. This value depends on whether the absolute or incremental mode is used.

Setting range:	-99,999,999 to +99,999,99
Unit:	Reference unit

9) While in trailing synchronous mode after the TSS command is completed, any motion command can be specified for a controlled axis, including the trailing axis, enabling work to be performed while in synchronous operation mode. If movement of the trailing axis is specified in absolute mode at this time, only the portion to be moved as a result of trailing synchronous operation will be applied.

The following group commands cannot be used while in trailing synchronous mode.

a) Motion Commands that Generate Alarms

If the following commands are executed while in trailing synchronous mode, an alarm will be generated.

Executing the ANALOG VOLTAGE OUTPUT (VCC) command will generate Alarm 024 (axis undefined).
Executing the EXTERNAL POSITIONING (EXM) command will generate Alarm 041 (axis duplicated).
Executing the TRAILING SYNCHRONOUS OPERATION (TSS) command will generate Alarm 041 (axis duplicated).

b) Other Unusable Motion Commands

Do not execute the following commands while in trailing synchronous mode. Although no alarm will be generated, the position of the workpiece cannot be guaranteed because the trailing axis is moving. Do not use these commands.

HOME POSITION RETURN (ZRN) command
SKIP (SKP) command
MOVE ON MACHINE COORDINATES (MVM) command

10) Overrides

An override can be set for the trailing axis in trailing synchronous operation in 16 steps between 0% and 100% or in increments of 0.1% between 0% and 3276.7%. The override, however, will apply only to the movement commands that occur during trailing synchronous operation and will not apply to the trailing synchronous operation speed component. The override can also be enabled and disabled.

a) Overrides for trailing axis in trailing synchronous operation are enabled and disable in parameter P0016.

(1) Override: Enabled for MC control coils (16 steps)

Ver. B08

(2) Override: Enabled for MC link registers (in increments of 0.1%)

(3) Override: Disabled

The overrides are disabled in the default settings.

b) When an override is enabled, the percentage can be switched as shown below.

(1) **MC Control Coils (16 Steps)**
0, 1, 2, 4, 6, 8, 10, 20, 30, 40, 50, 60, 80, 90, or 100 (%)

Ver. B08

(2) **MC Link Registers (0.1% Increments)**
0.0 to 3276.7 (%)

When an override is disabled, the speed will be 100% of the set value or reference value.

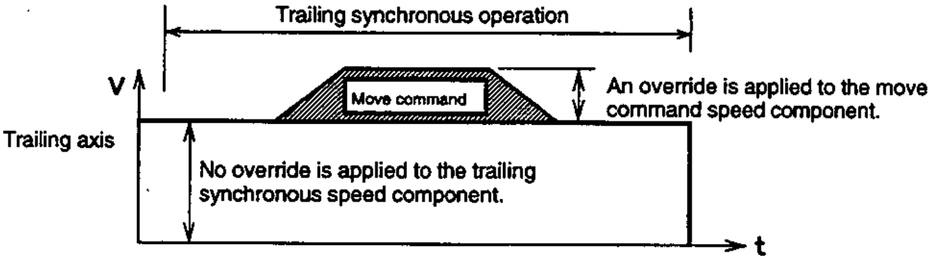
c) Overrides for trailing axis in trailing synchronous operation are set using the following MC control coils and MC link registers.

Ver. B08

Coil/Register		Use	References	
MC control coils				N is the MC20 Module number (N = 1 or 2)
	16 steps from 0% to 100%	Rapid traverse speed	QN0137 to QN0140	
	16 steps from 0% to 200%	Maximum interpolation feed speed	QN0141 to QN0144	
MC link registers 0.0% to 3276.7% in 0.1% increments		Rapid traverse speed	409909 and 409982	MC Modules 1 and 2 (default allocations)
		Maximum interpolation feed speed	409910 and 409983	

Set the MC link register to 10 times the override percentage when setting an override in 0.1% increments.

Refer to *3.2 MC Control Coil Functions* for details on MC control coils and to *1.5.6 Link Input Variables (#□□□□)* for details on MC link registers.


Trailing synchronous operation
V
Trailing axis
Move command
An override is applied to the move command speed component.
No override is applied to the trailing synchronous speed component.
t

11) Specify the TRAILING SYNCHRONOUS OPERATION CANCEL (TSR) command as follows:

TSR ; ↵ Cancels the trailing synchronous operation

When the command is executed, trailing synchronous mode for the trailing axis ends, and the trailing axis decelerates to a stop. After the trailing axis stops, the trailing synchronous mode is cancelled, and the block is completed. The trailing axis returns to a normal controlled axis.

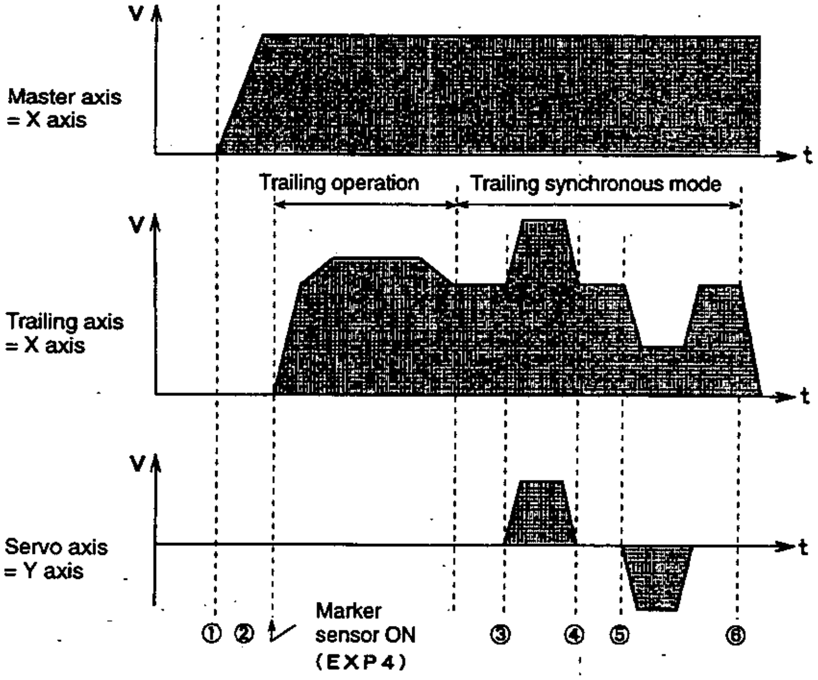
TSR is a preread command that is executed parallel to a move command. To execute after a move command in the previous block is completed, specify the IN-POSITION CHECK (PFN) command in the previous block.

Executing the TSR command resets the coordinate system. Therefore, the coordinate system shift that occurred as a result of previously executed manual operation will be cancelled.

◀EXAMPLE▶ **Programming Example**

This example shows when the external P axis (axis 4) is set as the master axis and the X axis is set as the trailing axis. The Y axis is a normal servo axis.

	ABS ; ↵	Sets absolute mode.
	MOV X0 Y0 ; ↵	Moves the jig to the wait position.
	PFN ; ↵	Performs an in-position check.
1	#03=1 ; ↵	Uses an output variable to forward rotate the master axis.
2	TSS X-50. P100. MSP ; ↵	Marker sensor ON position is X = 100. Distance from marker sensor ON position to trailing point is -50 in trailing synchronous operation.
	INC ; ↵	Sets incremental mode.
3	MOV X100. Y100. ; ↵	Moves the jig on the X and Y axes by +100.
4	SET M10 ; ↵	Executes the work specified in the SET EXTERNAL OUTPUT command.
5	MOV X-100. Y-100. ; ↵	Returns the distance travelled by the jig in step (3).
6	TSR; ↵	Ends trailing synchronous mode. The trailing axis will decelerate to a stop and trailing synchronous operation will be canceled.
	ABS ; ↵	Sets absolute mode.
	MOV X0 Y0 ; ↵	Returns the jig to the waiting position.



(The speed for each of the axes is the reference value.)

Note (1) The zero return operation for the ZEROn signal cannot be executed for another controlled axis while the trailing synchronous operation is being executed with the TSS command. Conversely, the trailing synchronous operation for the TSS command cannot be executed for another controlled axis while the zero return operation for the ZEROn signal being executed. If both functions are executed simultaneously, the following

alarm will be generated.
Alarm: 044 (ZERO signal duplicated) rank B

(2) Do not specify external positioning and trailing synchronous operation for the same axis at the same time. Correct operation cannot be guaranteed. For details, see *Note 4* of *1.10.1 EXTERNAL POSITIONING (EXM).*

(3) If a controlled axis designated as the trailing axis is being moved by a previous block command when the TSS command is executed, the following alarm will be generated.
Alarm: 041 (axis duplicated)
If the previous block is a move command, the TTS command will be executed when execution of the previous block starts. Therefore, in the next programming example, alarm 041 will be generated.

Programming Examples (Alarm 041 Generated)

```
N10   INC MOV X-200. ; ↵
N20   #03=1 ; ↵          Uses an output variable to forward rotate the master axis.
N30   TSS X50.P100.MSP ; ↵
N40   ....
```

The X axis, which is the trailing axis for the TRAILING SYNCHRONOUS OPERATION (TSS) command, is being moved by the N10 block command, and so an alarm is generated.

In the next programming example, however, no alarm will be generated.

Programming Example: No Alarm is Generated, and Current Position X = 200

```
N10   ABS MOV X200. ; ↵
N20   #03=1 ; ↵          Uses an output variable to forward rotate the master axis.
N30   TSS X50.P100.MSP ; ↵
N40   ....
```

The N10 move command is completed immediately after the execution start. Therefore, the X axis, which is the trailing axis for the TRAILING SYNCHRONOUS OPERATION (TSS) command, is not moving and so operation will be normal.

(4) If a master axis rapid traverse speed (PA202) is set that exceeds the set value for the trailing axis maximum feed speed (PA201) when the TSS command is executed, the following alarm will be generated.
Alarm: A11 (speed trailing disabled)

TERMS

ZEROn signals

One method of executing a zero return is to use the home position limit switch. The input signals for this home position limit switch signals are the ZEROn signals (replace n with axis number 1, 2, 3 or 4).

INFO

1) All the character data designations specified in the TSS command can be indirectly specified by means of variables.

2) If the individual character data for the TSS command is specified without the decimal point, the position of the decimal point follows the setting in parameter P0005 (decimal point position).

3) If a command such as the POSITIONING (MOV) command is executed while in trailing synchronous mode, no check (in-position check) is performed to determine whether the axis has entered the positioning completion range. Because the trailing axis is always moving, the operation will proceed to the next block when pulse distribution for the trailing axis is completed. The normal in-position check is performed for the MOV command for axes other than the trailing axis. The motion commands related to the in-position check are as follows:

   - POSITIONING (MOV) command

   - PALLET MOVE (PMV) command

   - IN-POSITION CHECK (PFN) command

   - SECOND IN-POSITION RANGE SETTING (INP) command

4) As a general rule, set 0 (none) for the filter selection (PA217) for a controlled axis used as the trailing axis. If 0 is not set, a trailing delay will result for the filter operation portion when a move command (such as MOV) is executed for a trailing axis in trailing synchronous mode.

5) If a trailing axis is specified as a finite-length axis (PA506 b1 = 0), set the allowable trailing range using the stored stroke limit function (PA508 to PA510).

6) Trailing synchronous operation can be executed only by specifying it in the MC20 Module program. It cannot be directly specified from the CPU Module ladder logic program.

Creating Ladder Logic Programs

2

This chapter describes how to write ladder logic programs in the CPU Module. Read this section before attempting to write a program.

2.1 Ladder Motion Instructions

This section describes the motion instructions that can be specified in CPU Module ladder logic programs, and explains their basic uses. Read this section before attempting to write a program.

2.1.1 Ladder Motion Instructions

1) The motion instructions that can be specified in CPU Module ladder logic programs are called ladder motion instructions. This group of instructions is used to control the MC20 Module from the CPU Module.

2) The 22 ladder motion instructions that are available are shown in the following table.

Table 2.1 CPU Module Ladder Motion Instructions

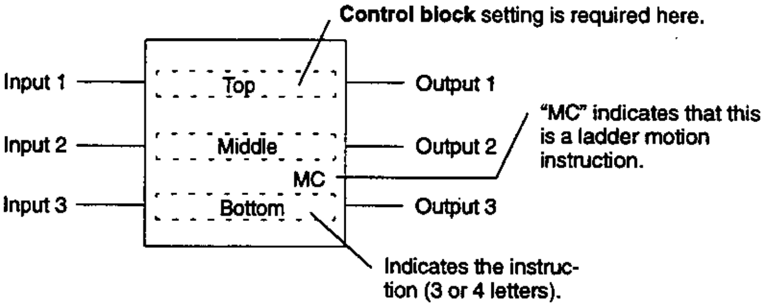
Instruction	Name	Function/Meaning	Page
MOD	MODE SET	Switches the MC20 Module operation mode.	2-31
SVN	SERVO ON	Turns the servomotor power ON and OFF.	2-34
MVL	PROGRAM RUN	Specifies the program and block numbers, and runs the program.	2-37
MVA, MVB, MVC, MVD	Independent Axis Operations	Normally operates independent axes A to D.	2-43
	Independent Axis Voltage Outputs (See note.)	Outputs the analog voltage (voltage output) from independent axes A to D.	2-51
	Independent Axis Ratio Operation (See note.)	Performs ratio operations to make independent axes A to D slave axes.	2-58
ZRN	HOME RETURN	Returns to home position in an incremental or absolute position detecting system.	2-65
JOG	JOG	Executes jogging.	2-71
STP	STEP	Executes stepping.	2-75
SMD	SINGLE BLOCK MODE	Switches to single block operation mode.	2-82
MLK	MACHINE LOCK MODE	In this mode, control axes do not move, but the current position display is changed according to the program.	2-85
MRS	MODULE RESET	Initializes the MC20 Module and resets all alarm status.	2-88
RST	MACHINE RESET	Resets part of the alarm status and enables the parameters changed with the PARAMETER SETTING (PRM) instruction.	2-92
ESP	EMERGENCY STOP NOTIFICATION	Notifies the MC20 Module that the emergency stop button has been pressed.	2-96
ARS	ALARM RESET	Resets the MC20 Module alarm.	2-98
MON	MONITOR	Monitors all internal data, including alarms, parameters, and program numbers.	2-101
POS	COORDINATE SETTING	Changes current position data.	2-109
PRM	PARAMETER SETTING	Sets data for parameters.	2-114
VAR	H VARIABLE SETTING	Sets data for H1 to H8 variables.	2-117
PTBL	POINT TABLE SETTING	Sets data for point tables.	2-122
ZST	HOME POSITION SETTING	Sets the home position for absolute position detecting system.	2-126

Note New step-2 functions.

2.1.2 Specifying Ladder Motion Instructions

1. Instruction Structure

1) The basic structure of ladder motion instructions is as follows:


Control block setting is required here.
Input 1
Input 2
Input 3
Top
Middle
MC
Bottom
Output 1
Output 2
Output 3
"MC" indicates that this is a ladder motion instruction.
Indicates the instruction (3 or 4 letters).

2) Depending on the instruction, inputs 2 and 3 are not used. Control execution of each instruction by connecting it to the specified types of input contacts.

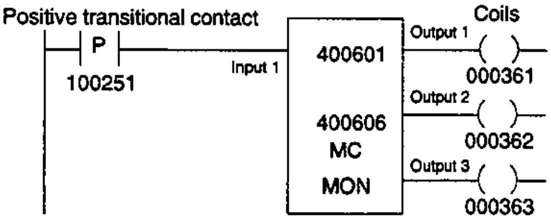
3) Depending on the instruction, output 3 is not used. Each output is turned ON and OFF depending on inputs and other conditions. Be careful of the timing when using output signals.

4) Generally, only the first (top) of the three vertical network elements (top, middle, and bottom) need to be set for ladder motion instructions. The meanings of the three elements are explained below.

Element	Setting Requirement	Meaning/Contents
Top	Setting required	Specifies, as the control block for the relevant instruction, the leading reference number for the required number of holding registers. The group of the number of holding registers required to execute a given instruction is called the "control block," and the top element specifies the leading reference number of the control block. For details, refer to the explanations for the individual instructions.
Middle	Setting required for monitor instructions only.	For details, refer to *2.3.7 MONITOR (MON).*
Bottom	—	Indicates the relevant instruction, expressed in three or four letters of the alphabet. It is automatically set when the instruction is specified.

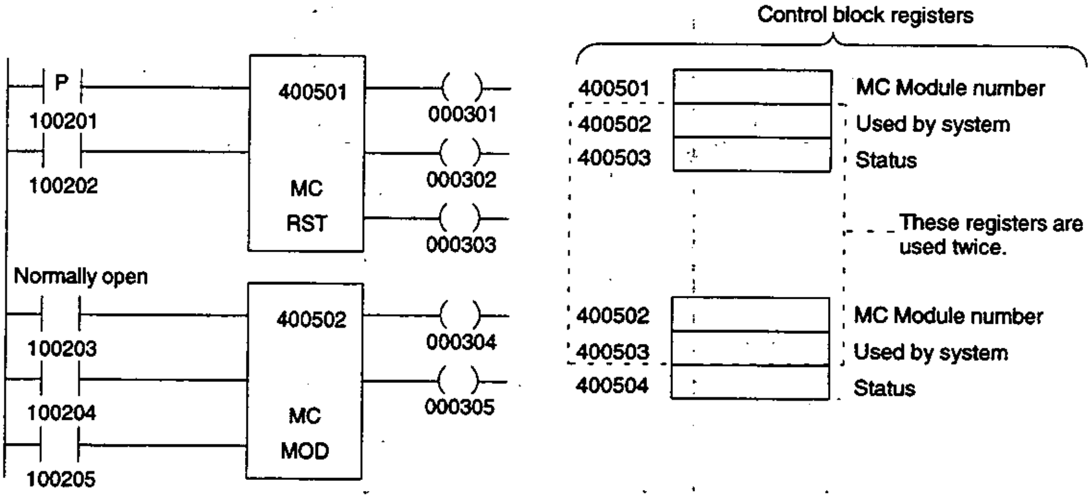
5) The following is an example of a ladder logic diagram including the MONITOR (MON) instruction.

MONITOR (MON) Instruction



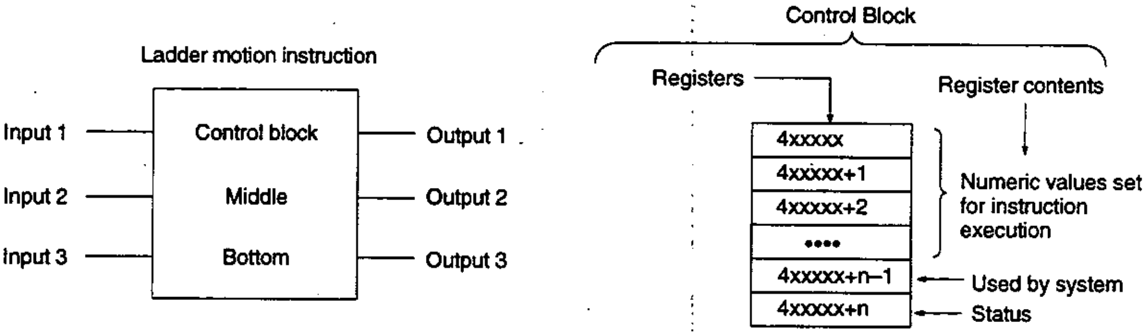
⚠ **Caution** Do not use the registers in control blocks for ladder motion instructions for more than one purpose (i.e., more than one instruction). The ladder motion instructions may not function properly if one or more registers in the control block for one ladder motion instruction are also used in the control block for another ladder motion instruction. Never use even one register in a control block for more than one ladder motion instruction.

Example: Incorrect Application



⚠ **Caution** Never write data to the control block for a ladder motion instruction that is being executed in the program. If another part of the ladder logic program writes data to the control block of a ladder motion instruction being executed in the program, execution of the ladder motion instruction will be force-ended. Never write data to the control block for a ladder motion instruction that is being executed. (You can read data from the control block from other portions of the ladder logic program without any problem.)

⚠ **Caution** Do not write data to the registers used by the system even if the instructions for which the registers are used are not currently being executed.

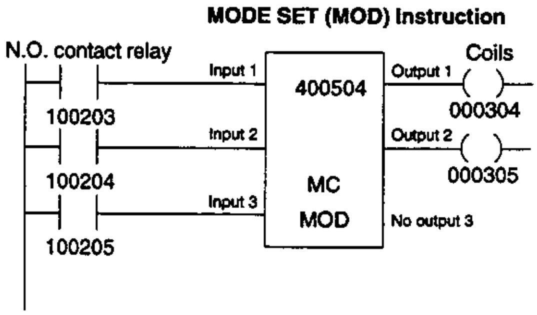


If data is written to the register specified "used by system" from other parts of the ladder logic program, the above ladder motion instruction will not operate and no alarms will be given.

2. Example of Inputting Instructions

The basic methods used to input ladder logic programs including ladder motion instructions is explained next. In the following example, the MODE SET (MOD) instruction is added to a ladder logic program.

Figure 3.1 Ladder Logic Programming Example



The following are used to input the program.

- Software: MEMOSOFT
- Mode: Offline
- Program name: TEST1
- Writing mode: Ladder editing mode
- Input to: Network #1 of high-speed segment

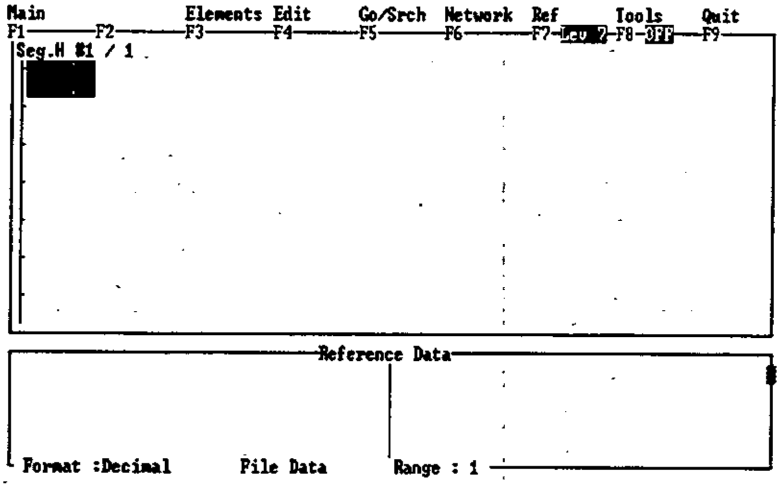
The following procedure is used to input instructions under the above conditions.

1) Start the MEMOSOFT and select the offline mode. Then go to the program selection menu or program creation menu to select or create TEST1.

Refer to the *MC20 Module Hardware User's Manual* for details on operating procedures.

2) Move the cursor to the ladder editing operation and then press the Enter Key so that the segment status screen will appear.

3) Move the cursor to select the high-speed segment and press the Enter Key. The following ladder edit screen will appear and you are now ready to input the ladder logic program to network 1 of the high-speed segment.

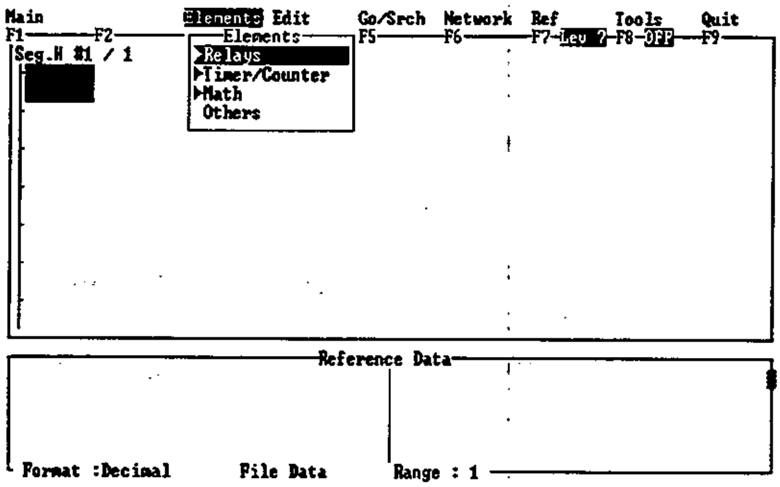


Format :Decimal

Range : 1

4) Use the following procedure to add the N.O. contact relay for input 1.

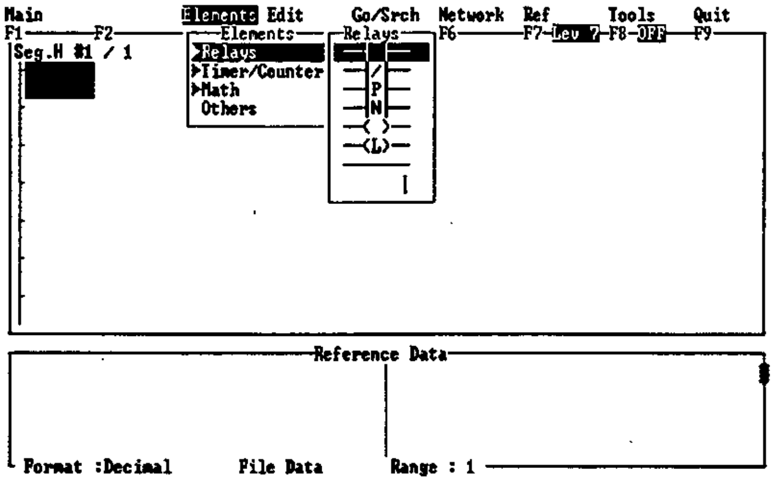
a) After pressing the Tab Key, move the cursor to select relays from the element pull-down menu.



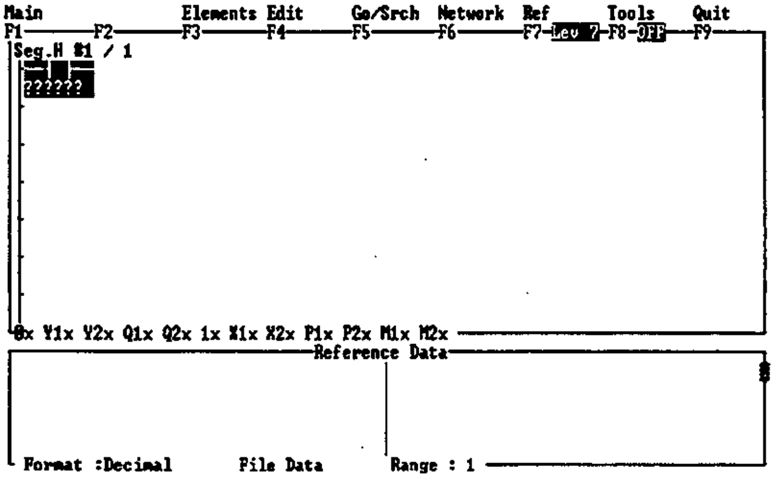
Format :Decimal

Range : 1

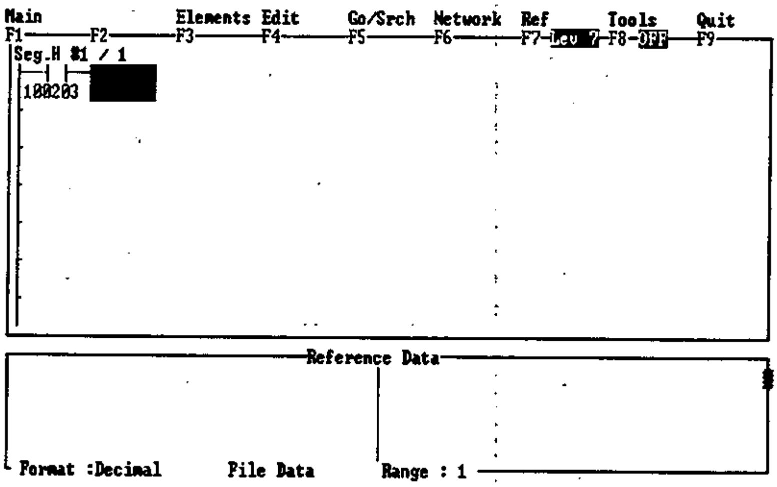
b) Press the Enter Key.
The following menu will appear.



c) Select the N.O. contact relay with the cursor and press the Enter Key.
The following screen will appear.



d) Input I203 or 100203 as the reference number of the N.O. contact relay and press the Enter Key.
The relay is written and the cursor will move to the next element position on the right.

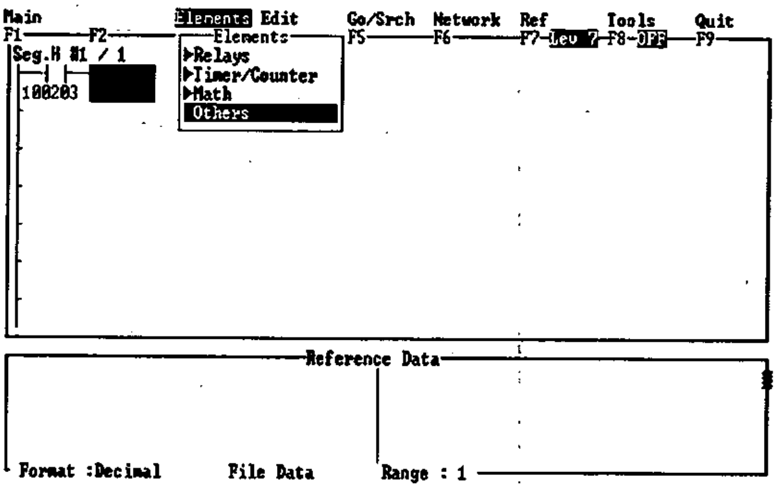


Format :Decimal

Range : 1

5) Use the following procedure to add the MODE SET (MOD) instruction.

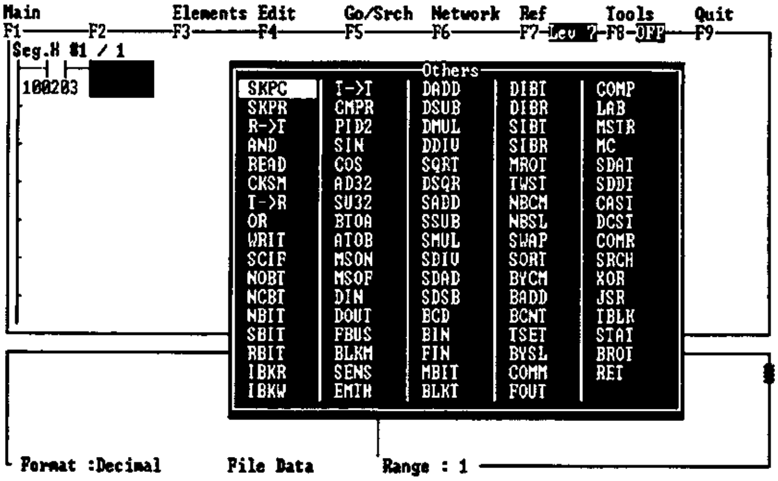
a) After pressing the Tab Key, move the cursor to select other elements from the element pulldown menu.



Format :Decimal

Range : 1

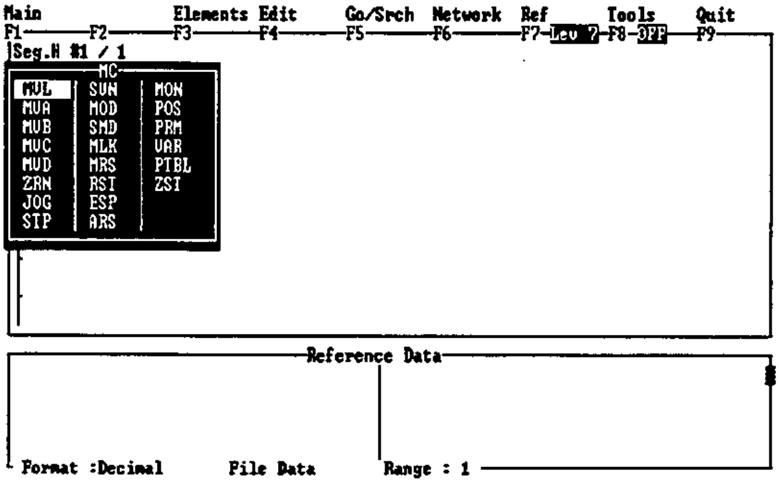
b) Press the Enter Key.
A list of the following high-level control instructions will appear.



```
Main              Elements Edit    Go/Srch  Network  Ref      Tools    Quit
F1-------F2-------F3-------F4------F5-------F6-------F7-Lev ?-F8-OFF---F9---
Seg.H #1 / 1
100203
                                   Others
                  SKPC   T->T    DADD    DIBT    COMP
                  SKPR   CMPR    DSUB    DIBR    LAB
                  R->T   PID2    DMUL    SIBT    MSTR
                  AND    SIN     DDIV    SIBR    MC
                  READ   COS     SQRT    MROT    SDAT
                  CKSM   AD32    DSQR    TWST    SDDT
                  T->R   SU32    SADD    NBCM    CASI
                  OR     BTOA    SSUB    NBSL    DCSI
                  WRIT   ATOB    SMUL    SWAP    COMR
                  SCIF   MSON    SDIV    SORT    SRCH
                  NOBT   MSOF    SDAD    BYCM    XOR
                  NCBT   DIN     SDSB    BADD    JSR
                  NBIT   DOUT    BCD     BCNT    IBLK
                  SBIT   FBUS    BIN     TSET    STAT
                  RBIT   BLKM    FIN     BYSL    BROT
                  IBKR   SENS    MBIT    COMM    RET
                  IBKW   EMTM    BLKT    FOUT

Format :Decimal      File Data      Range : 1
```

c) Move the cursor to "MC" and press the Enter Key.
The following list of ladder motion instructions will appear.

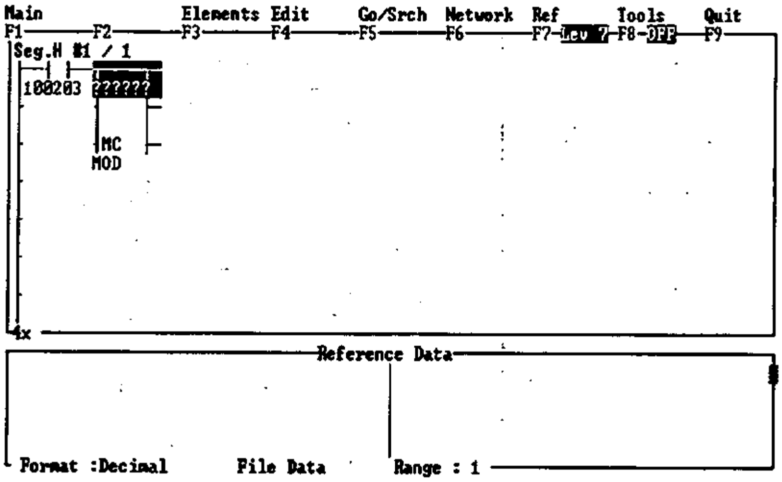


```
Main              Elements Edit    Go/Srch  Network  Ref      Tools    Quit
F1-------F2-------F3-------F4------F5-------F6-------F7-Lev ?-F8-OFF---F9---
Seg.H #1 / 1
         MC
 MVL    SVN    MON
 MVA    MOD    POS
 MVB    SMD    PRM
 MVC    MLK    VAR
 MVD    MRS    PTBL
 ZRN    RST    ZST
 JOG    ESP
 STP    ARS

                         Reference Data

Format :Decimal      File Data      Range : 1
```

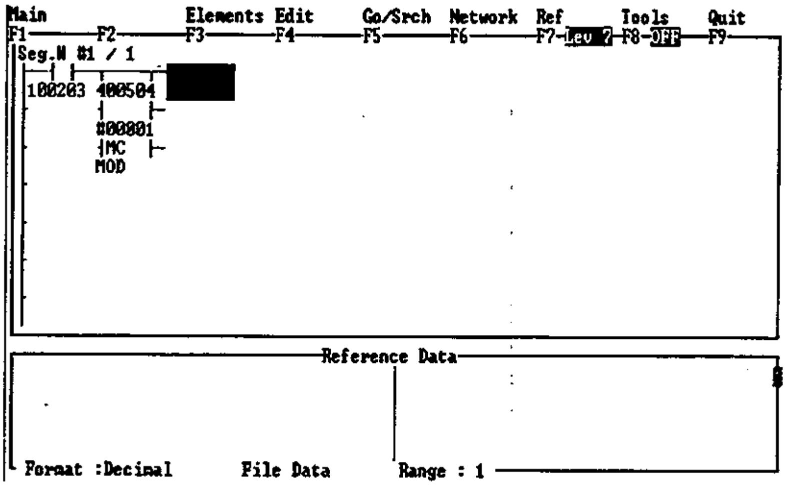
d) Move the cursor to "MOD" and press the Enter Key.
The MODE SET (MOD) instruction will appear as shown in the following illustration.



Format :Decimal

Range : 1

e) Input W504 or 400504 as the reference number of the top element (control block) of the MODE SET (MOD) instruction and press the Enter Key.
The MODE SET (MOD) instruction will be written and the cursor will move to the next element position on the right.

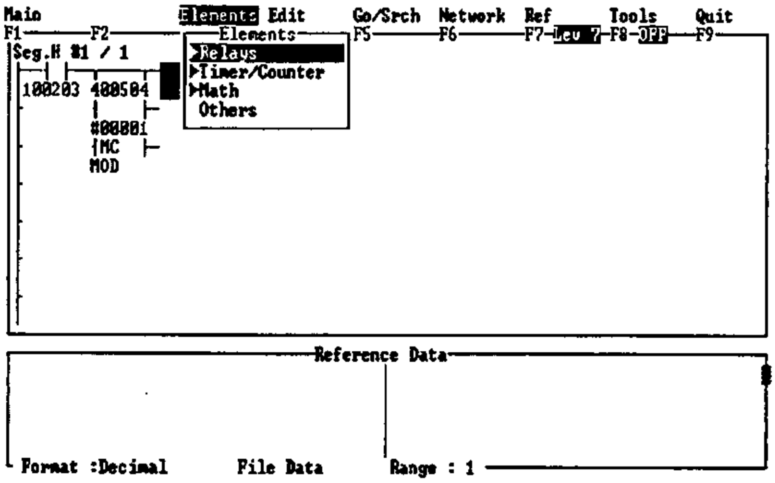


Format :Decimal

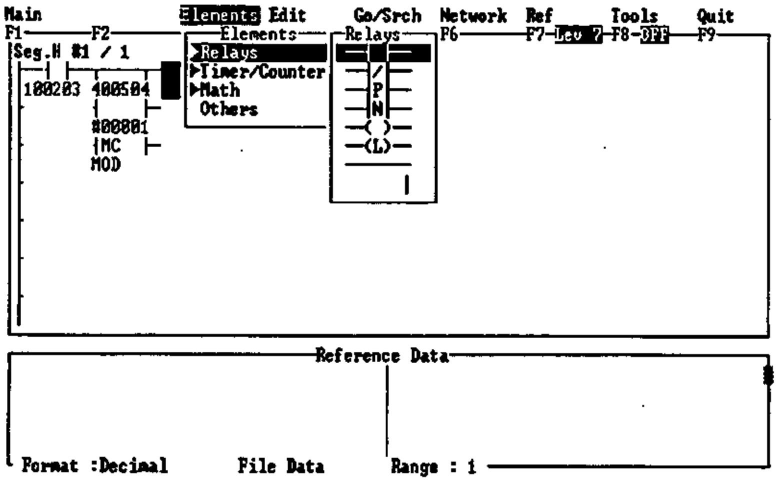
Range : 1

6) Use the following procedure to add a coil for output 1.

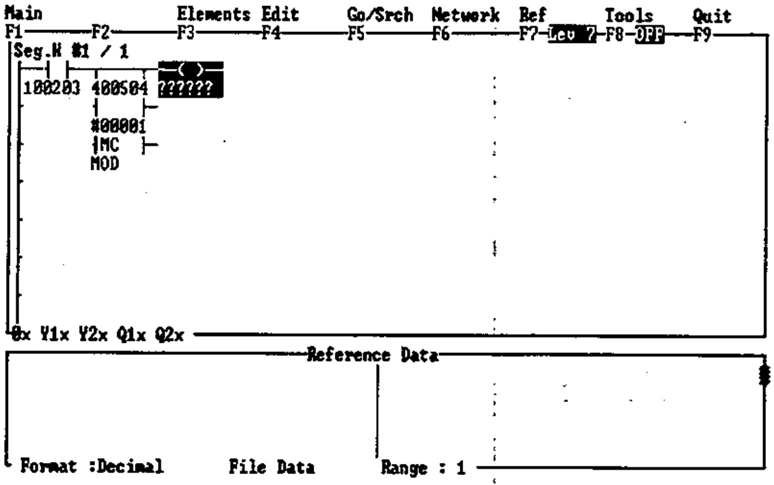
a) After pressing the Tab Key, move the cursor select relays in the element pulldown menu.



b) Press the Return Key.
The following screen will appear.



c) Select the coil with the cursor and press the Enter Key.
The following screen will appear.

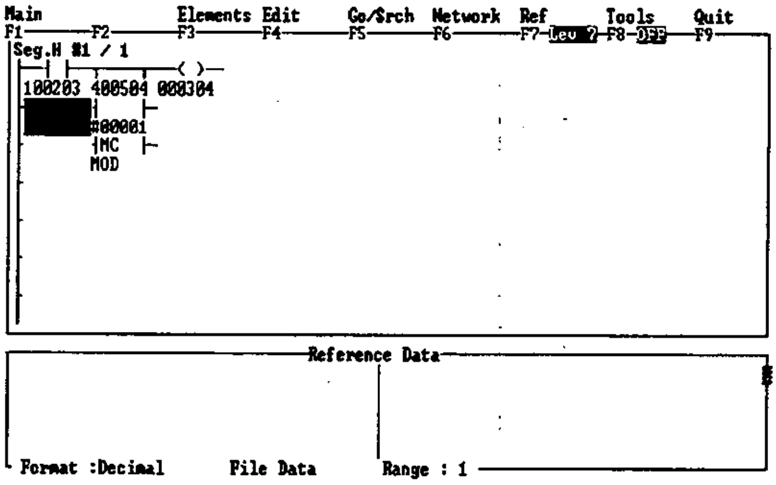


```
Main              Elements Edit     Go/Srch  Network  Ref       Tools    Quit
F1--------F2------F3--------F4------F5-------F6-------F7-Lev ?-F8-OFF---F9---
 Seg.H #1 / 1
 100203 400504 ??????
        #00001
        MC
        MOD

0x Y1x Y2x Q1x Q2x
                               Reference Data

 Format :Decimal       File Data      Range : 1
```

d) Input O304 or 000304 as the reference number of the coil and press the Enter Key. The coil is written and the cursor will move to the beginning of the next line.

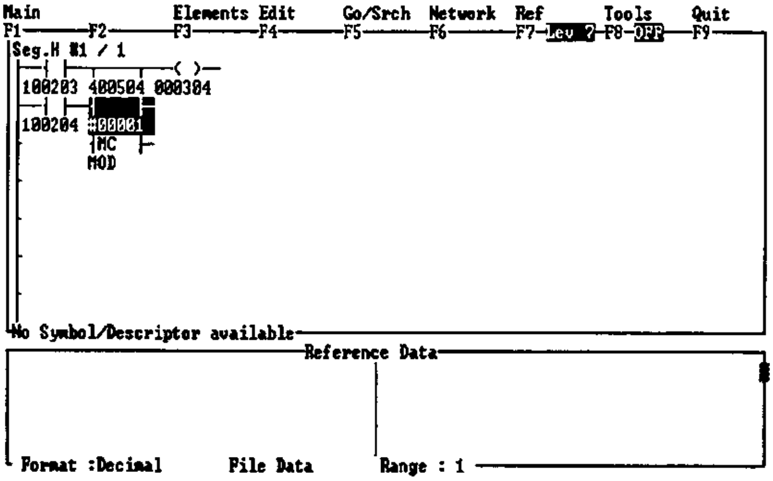


```
Main              Elements Edit     Go/Srch  Network  Ref       Tools    Quit
F1--------F2------F3--------F4------F5-------F6-------F7-Lev ?-F8-OFF---F9---
 Seg.H #1 / 1
 100203 400504 000304
        #00001
        MC
        MOD

                               Reference Data

 Format :Decimal       File Data      Range : 1
```

7) Add the N.O. contact relay for input 2 in the same way the N.O. contact relay for input 1 was added and input I204 or 100204 for the reference number of the relay.
The following screen will appear.

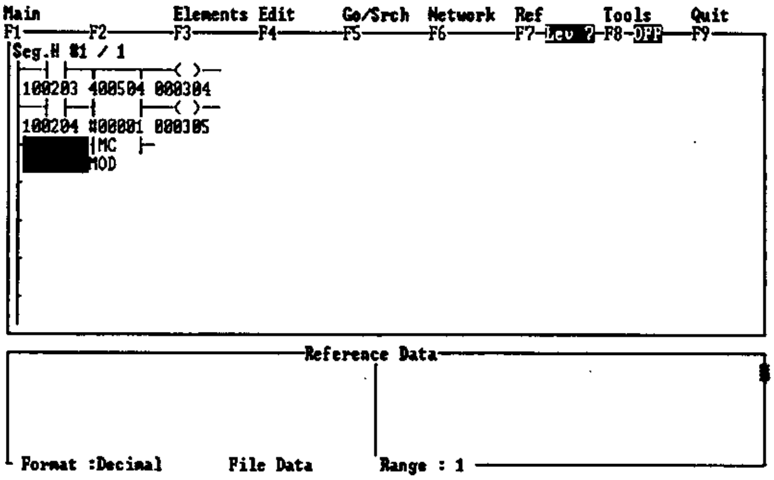


No Symbol/Descriptor available

Format :Decimal

Range : 1

8) After moving the cursor to the next element position on the right, add output 2 coil in the same way the output 1 coil was added and input O305 or 000305 for the reference number of the coil.
The following screen will appear.

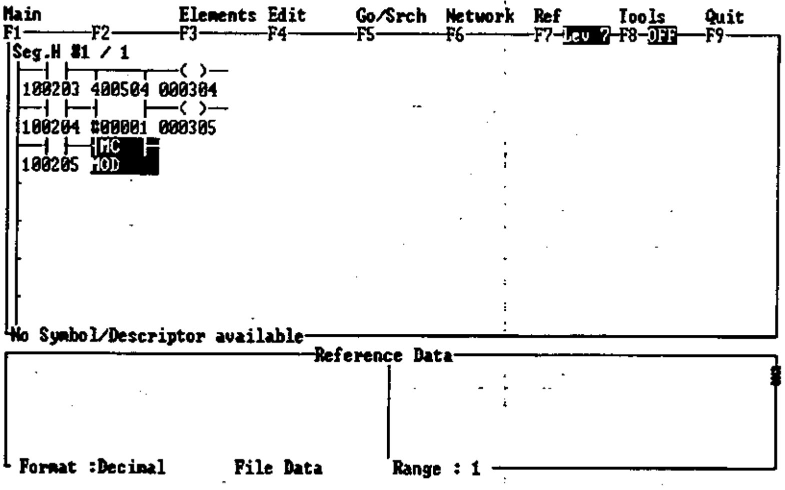


Format :Decimal

Range : 1

9) Add the N.O. contact relay for input 3 in the same way the N.O. contact relay for input 1 was added and input I205 or 100205 for the reference number of the relay.

The following screen will appear and you have completed adding all the necessary elements to the ladder logic diagram.



```
Main                  Elements Edit      Go/Srch  Network  Ref       Tools    Quit
F1-------F2---------F3-------F4---------F5-------F6-------F7-Lev 7-F8-OFF---F9---
 Seg.H #1 / 1
  100203 400504 000304
  100204 $00001 000305
  100205 MC
         MOD
No Symbol/Descriptor available
                               Reference Data
 Format :Decimal      File Data      Range : 1
```

10) The MODE SET (MOD) instruction does not have output 3; output 3 need not be added.

INFO

In the above example, the elements were added one by one horizontally for easy reference. If there is a draft of your motion program, you may write the motion program vertically, in which case you may be able to write the program more quickly than you write the same program horizontally, because the same elements may be input repeatedly.

11) Refer to the following manual for details:
MEMOCON GL120, GL130 MEMOSOFT for DOS User's Manual

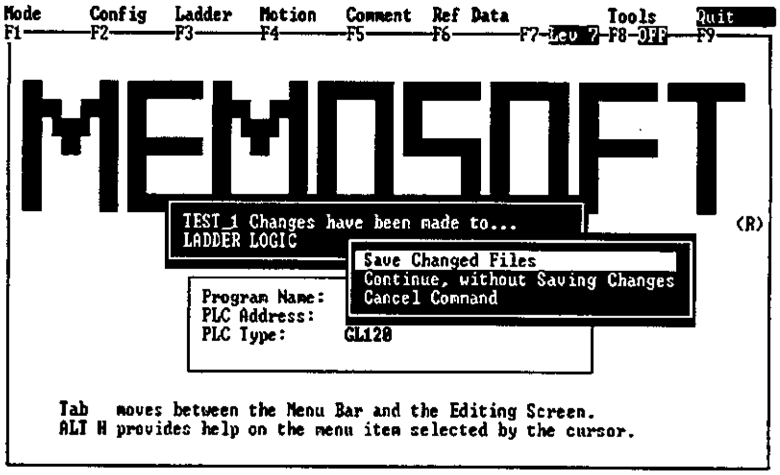
12) Use the following procedure to save the motion program and quit MEMOSOFT.

a) Press the F9 Key.
The segment status screen will appear.

b) Press the F9 Key again.
The initial screen of MEMOSOFT will appear, at which time make sure that TEST1 appears as a program name.

c) Press the F9 Key again.
The quit prompt will appear.

d) Press the Y key and Return Key.
The following screen will appear.



```
TEST_1 Changes have been made to...
LADDER LOGIC

Program Name:
PLC Address:
PLC Type:       GL128

Tab   moves between the Menu Bar and the Editing Screen.
ALT H provides help on the menu item selected by the cursor.
```

e) Make sure that "Save Changed Files," is highlighted and then press the Enter Key. MEMOSOFT will save the motion program and quit.

2.1.3 Ladder Motion Program

1) A set of instructions in a ladder logic program used to execute a motion program and make other designations for the MC20 Module from the CPU Module is called a ladder motion program.

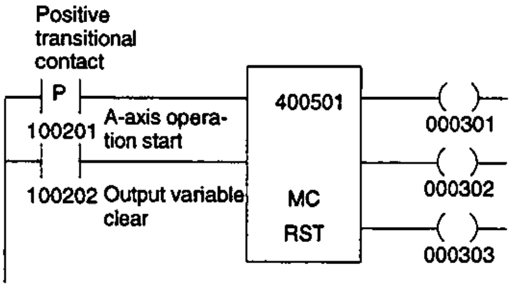
2) The following is an example of a ladder motion program that operates under the following conditions.

a) Axis 1: X; Axis 2: Y; Axis 3: Z; Axis 4: S

b) Position detecting system: Incremental or absolute

3) **Example of Motion Programming in Ladder Logic Program**

a) **MACHINE RESET (RST) Instruction**
This instruction is used to reset the alarm and execution pointer of MC20 Module 1.

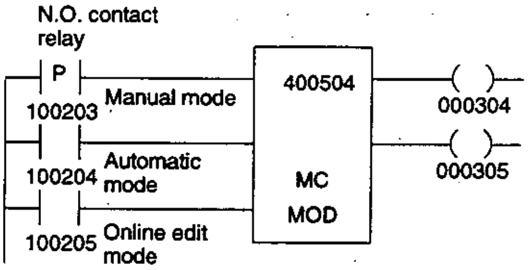


Control Block and the Contents of Its Registers

400501	1	MC20 Module number
400502		Used by system
400503		Status

b) MODE SET (MOD) Instruction

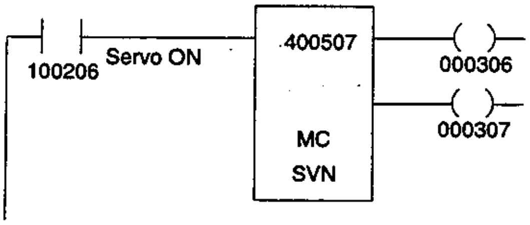
This instruction is used to specify the operation mode of MC20 Module 1.



400504	1	MC20 Module number
400505		Used by system
400506		Status

c) SERVO ON (SVN) Instruction

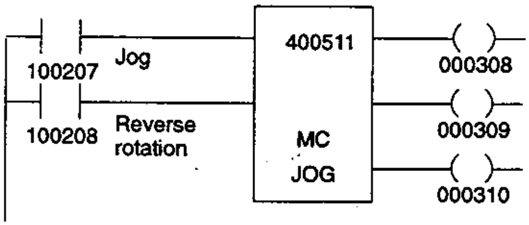
This instruction is used to turn the servomotors of axes 1 to 4 ON.



400507	1	MC20 Module number
400508	15	Controlled axis designation
400509		Used by system
400510		Status

d) JOG (JOG) Instruction 1

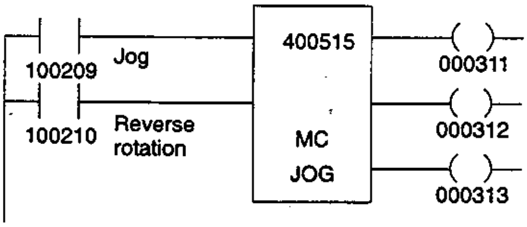
This instruction is used to enable the jogging operation of axis 1.



400511	1	MC20 Module number
400512	1	Controlled axis designation
400513		Used by system
400514		Status

e) JOG (JOG) Instruction 2

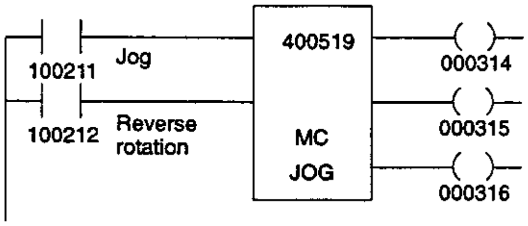
This instruction is used to enable the jogging operation of axis 2.



400515	1	MC20 Module number
400516	2	Controlled axis designation
400517		Used by system
400518		Status

f) JOG (JOG) Instruction 3

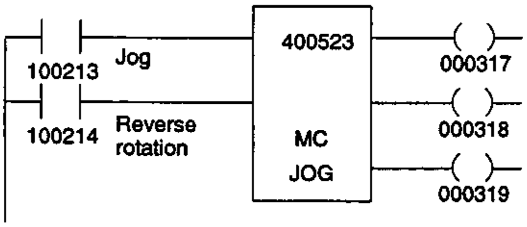
This instruction is used to enable the jogging operation of axis 3.



400519	1	MC20 Module number
400520	4	Controlled axis designation
400521		Used by system
400522		Status

g) JOG (JOG) Instruction 4

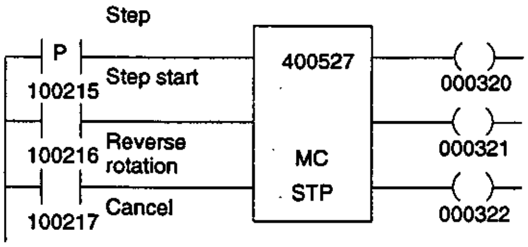
This instruction is used to enable the jogging operation of axis 4.



400523	1	MC20 Module number
400524	8	Controlled axis designation
400525		Used by system
400526		Status

h) STEP (STP) Instruction 1

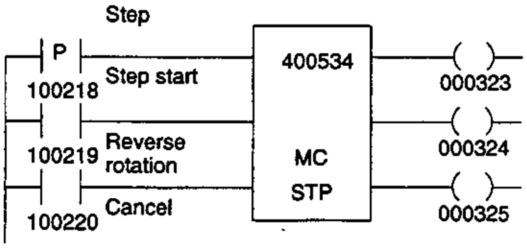
This instruction is used to enable the stepping operation of axis 1.



400527	1	MC20 Module number
400528	1	STEP travel distance L
400529	1000	Controlled axis designation
400530	0	STEP travel distance H
400531	0	Timeout error check time
400532		Used by system
400533		Status

i) STEP (STP) Instruction 2

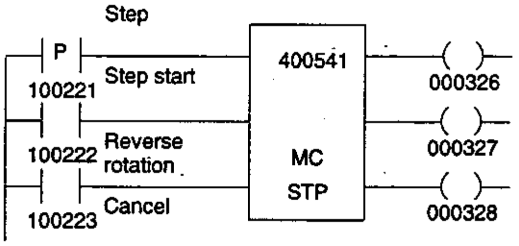
This instruction is used to enable the stepping operation of axis 2.



400534	1	MC20 Module number
400535	2	STEP travel distance L
400536	1000	Controlled axis designation
400537	0	STEP travel distance H
400538	0	Timeout error check time
400539		Used by system
400540		Status

j) **STEP (STP) Instruction 3**

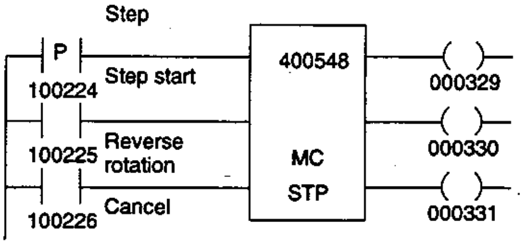
This instruction is used to enable the stepping operation of axis 3.



400541	1	MC20 Module number
400542	4	STEP travel distance L
400543	1000	Controlled axis designation
400544	0	STEP travel distance H
400545	0	Timeout error check time
400546		Used by system
400547		Status

k) **STEP (STP) Instruction 4**

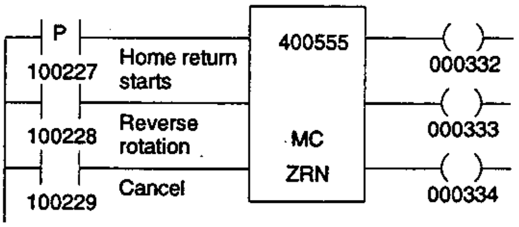
This instruction is used to enable the stepping operation of axis 4.



400548	1	MC20 Module number
400549	8	STEP travel distance L
400550	1000	Controlled axis designation
400551	0	STEP travel distance H
400552	0	Timeout error check time
400553		Used by system
400554		Status

l) **HOME RETURN (ZRN) Instruction 1**

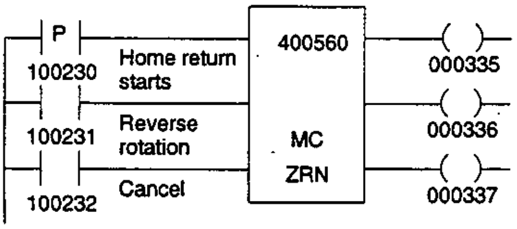
This instruction is used to make axis 1 return to the home position in an incremental or absolute position detecting system.



400555	1	MC20 Module number
400556	1	Controlled axis designation
400557	0	Timeout error check time
400558		Used by system
400559		Status

m) **HOME RETURN (ZRN) Instruction 2**

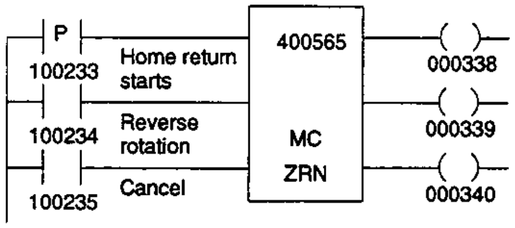
This instruction is used to make axis 2 return to the home position in an incremental or absolute position detecting system.



400560	1	MC20 Module number
400561	2	Controlled axis designation
400562	0	Timeout error check time
400563		Used by system
400564		Status

n) **HOME RETURN (ZRN) Instruction 3**

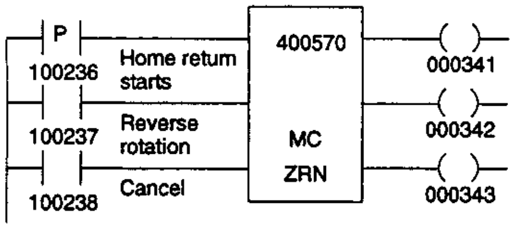
This instruction is used to make axis 3 return to the home position in an incremental or absolute position detecting system.



400565	1	MC20 Module number
400566	4	Controlled axis designation
400567	0	Timeout error check time
400568		Used by system
400569		Status

o) **HOME RETURN (ZRN) Instruction 4**

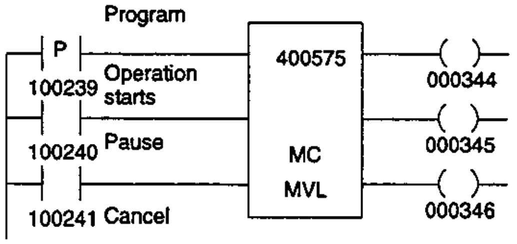
This instruction is used to make axis 4 return to the home position of the incremental or absolute position detecting system.



400570	1	MC20 Module number
400571	8	Controlled axis designation
400572	0	Timeout error check time
400573		Used by system
400574		Status

p) **PROGRAM RUN (MVL) Instruction**

This instruction is used to specify a program and block number and run the program.



400575	1	MC20 Module number
400576	1	Program number
400577	1	Block number
400578	0	Timeout error check time
400579		Used by system
400580		Status

Note (1) Ladder motion instructions should be programmed in high-speed segments if the capacity of the program is not too large and if there is leeway in the scan.

(2) The following ladder motion instructions must be programmed in high-speed segments regardless of the scan time. Each of these instructions requires a high-speed response time. If they are programmed in normal segments, they may not operate properly.

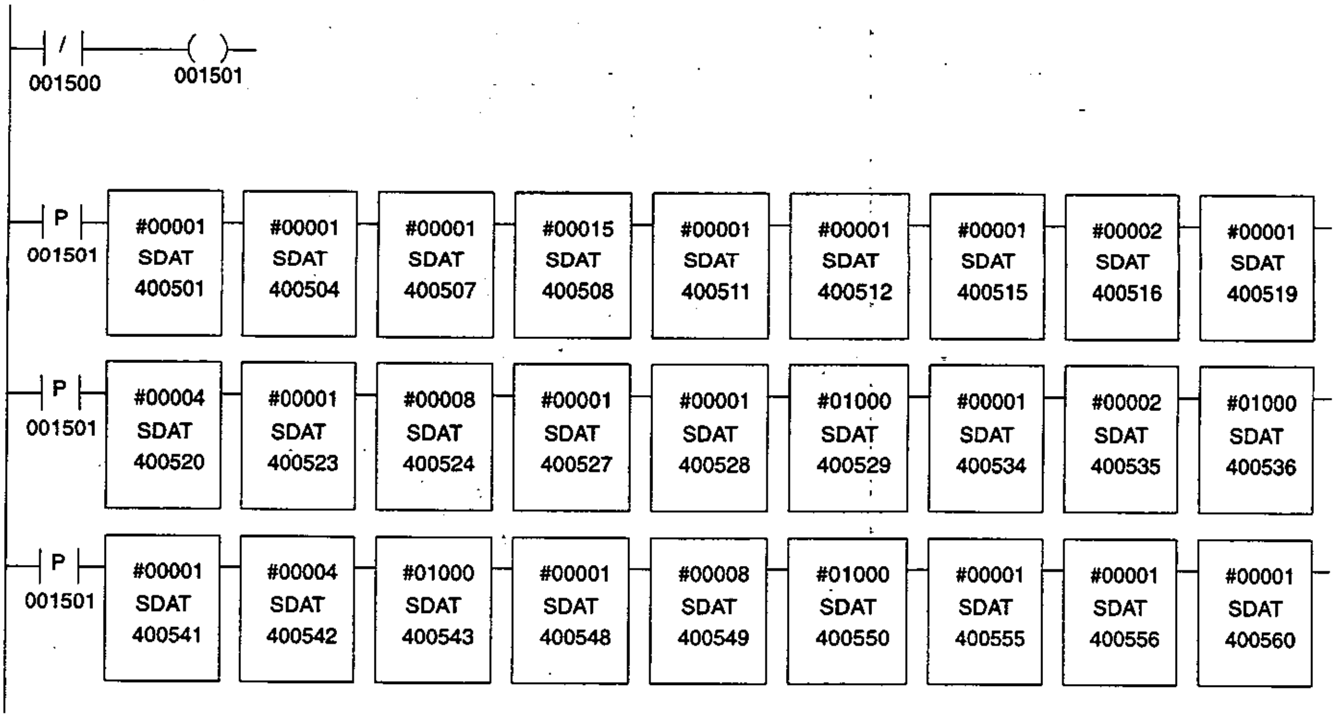
- PROGRAM RUN (MVL)
- JOG (JOG)
- Independent Axis Operations (MVA to MVD)

- STEP (STP)
- HOME RETURN (ZRN)
- EMERGENCY STOP NOTIFICATION (ESP)

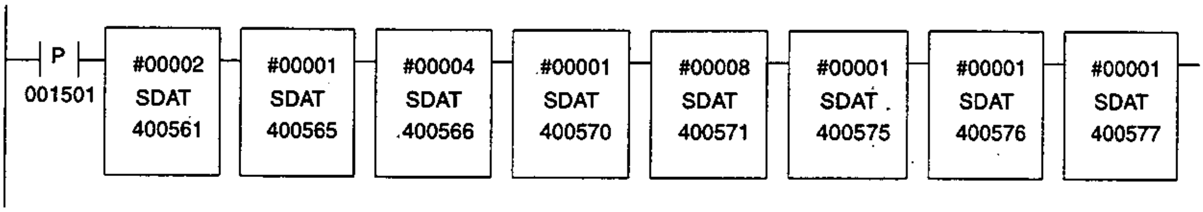
(3) Any other instruction may be programmed in a normal segment to reduce scan time.

4) The following is an example of a ladder logic program segment that sets the values shown for the above example program into the control blocks. This program is written in the normal segment. This program need not be written in the high-speed segment.

Figure 3.2 Example Programing to Set Values in Control Block



Next Screen



SDAT Operation

When input 1 is ON, SDAT will store the source (i.e., the value on the first line) at the destination (i.e., the holding registers specified by the reference on the last line) and output 1 will turn ON.

2.1.4 Coils and Relays for Synchronizing

1) The coils and relays listed below can be used for synchronizing and confirming operations between CPU Module ladder logic program and MC20 Module motion program.

Coil/Relay	Role
MC control coil	A specific signal with a fixed allocation for controlling the MC20 Module from the ladder logic program.
MC control relay	A specific signal with a fixed allocation for transmitting the specific status of the MC20 Module to the CPU Module.
M code relay (M code output)	A signal read by the CPU Module to receive the M code output specified by the SET EXTERNAL OUTPUT (SET) and PASS NOTCH SIGNAL OUTPUT (PNT) commands in the motion program.
MC coil (input variable)	A coil that can be used for general purposes in the ladder logic program. The ON or OFF status of this coil can be read as 1 or 0 for the corresponding input variable in the motion program.
MC relay (output variable)	A relay that can be used for general purposes in the ladder logic program. The corresponding MC relay can be turned ON or OFF by specifying 1 or 0 for the output variable in the motion program.

2) MC control coils and relays are specific signals for the MC20 Module. For details regarding these signals, refer to *Chapter 4 Using MC Control Coils and Relays.*

3) The following table shows the reference numbers corresponding to the various types of coils and relays for synchronizing and confirmation. The program example is for MC20 Module 1.

Table 2.2 Coils and Relays for Synchronizing and Checking

		CPU Module ladder logic program	MC20 Module motion program
Item		1) MC control coil	---
References	MC Module 1	Q10001 to Q10160	---
	MC Module 2	Q20001 to Q20160	---
Program Example		—( )— Q10096: MF96 MFIN coil	---
Item		2) MC control relay	---
References	MC Module 1	P10001 to P10256	---
	MC Module 2	P20001 to P20256	---
Program Example		—| |— P10020: M code read request	---
Item		3) M code relay	3) M code output
References	MC Module 1	M10001 to M10096	M01 to M96
	MC Module 2	M20001 to M20096	M01 to M96
Program Example		—| |— M10096: M96 M code relay	`N101 SET M96 ;`↵ or `N102 PNT M96 ;`↵
Item		4) MC coil	4) Input variable
References	MC Module 1	Y10001 to Y10256	#I1 to #I256
	MC Module 2	Y20001 to Y20256	#I1 to #I256

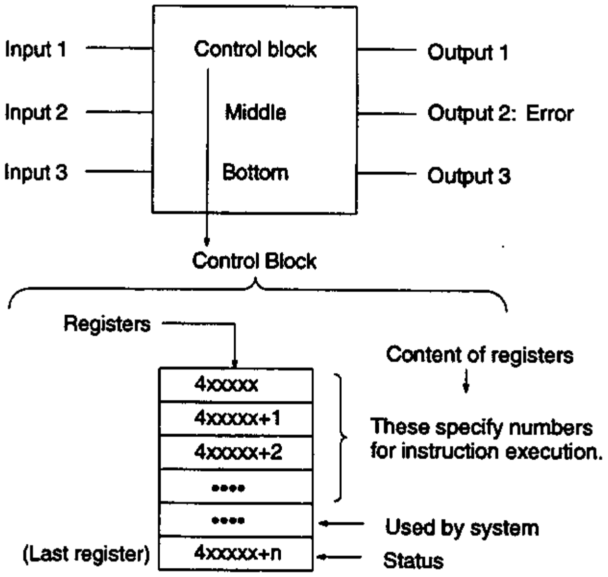
Program Example		—( )— ON → 1
		Y10001
		—( )— OFF → 0
		Y10256
			`N100 IF #I1==1`
			`GOTO 120 ;`↵
			`...`
			`N120 #50=#I256 ;`↵

Item		5) MC relay	5) Output variable
References	MC Module 1	X10001 to X10256	#O1 to #O256
	MC Module 2	X20001 to X20256	#O1 to #O256
Program Example		—| |— ON ← 1 X10001 —| |— OFF ← 0 X10256	N130 #O1=1 ; ↵ ... N140 #O256=0 ; ↵

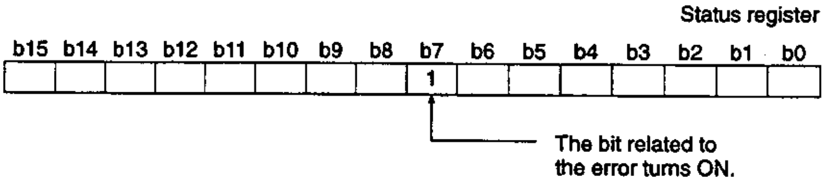
4) MC control coils and relays receive signals from M code outputs and relays. For details, refer to *1.4.3 SET EXTERNAL OUTPUT (SET)*.

2.1.5 Error Status

1) The lowest level of the control block registers (i.e., the top element in a ladder motion instruction) is called "status," and it indicates the error status after the instruction has been executed.



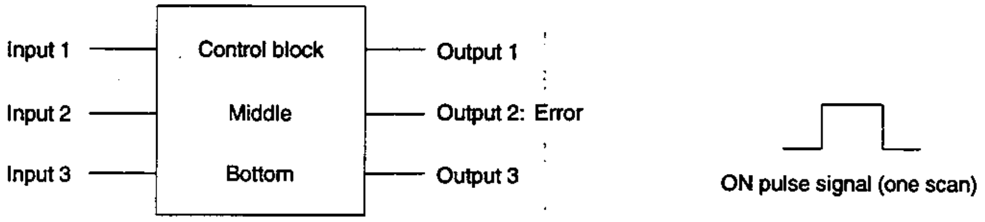
2) If an error occurs when a instruction is executed, the relevant bit in the 16-bit status register will turn ON (i.e., be changed to "1").


Status register
b15 b14 b13 b12 b11 b10 b9 b8 b7 b6 b5 b4 b3 b2 b1 b0
1
The bit related to the error turns ON.

3) The relationships between bit positions and errors are the same for all instructions. These are shown in the following table.

Bit position	Error Name	Error Contents
b0=1	Axis designation error	• The designated axis number is out of range. • An undefined axis name was designated.
b1=1	Data error	• Data was specified outside of the allowable range.
b2=1	No unit	• The designated MC20 Module is not mounted.
b3=1	Busy error	• An attempt was made to execute a instruction that cannot be executed simultaneously with the instruction currently being executed.
b4=1	Instruction prohibited	• The CPU Module determined that the ladder motion instruction cannot be specified for the MC20 Module.
b5=1	Timeout error	• A instruction has been specified for execution by the MC20 Module, but a response is not possible within the time limit.
b6=1	MC not ready	• The MC20 Module is not ready. (MC control relay: MCRD is OFF.)
b7=1	MC busy error	• The MC20 Module is unable to receive the ladder motion instruction.
b8=1	MC error	• The MC20 Module executed an operation that the CPU Module was not expecting.
b9=1	MC interface error	• An error occurred during communications between the CPU Module and the MC20 Module.
b10 to b15	Not used.	————

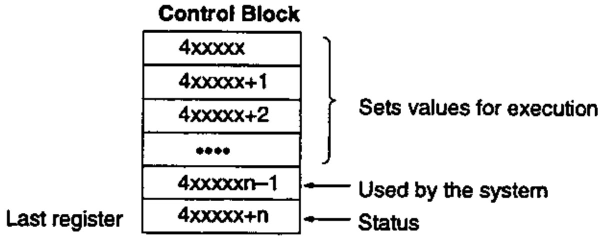
4) When a given ladder motion instruction is executed, its status up to that point is completely reset. If errors subsequently occur with respect to the instruction execution, and one or more bits turn ON in the status register, the instruction execution is forcibly stopped. In that situation, an ON pulse signal is output on output 2 (error) to indicate that an error has occurred.



For the MODE SET (MOD) instruction, however, a pulse signal is not used and the ON status will remain until the cause of the error is cleared.

5) After an error has occurred, the status can be read in the following way. Use the Programming Device to specify the holding register reference number for the status of any instruction for which an error occurs, and check the contents of the various bits by displaying the bits.

Note The register one before the last register in the control block is for system use for the MC20 Module. This register must not be used by the user program.



2.1.6 Application Conditions for Ladder Motion Instructions

Operation Modes and Applicable Instructions

The MC20 Module's operation mode can be changed by executing the MODE SET (MOD) instruction. There are four operation modes. The following table shows which instructions can be executed in each mode.

Table 2.3 Operation Modes and Applicable Instructions

Instruction			Operation Mode			
Symbol		Name	Manual	Automatic	Online Edit	Edit
MOD		MODE SET	○	○	○	○
SVN		SERVO ON	○	○	○	○
MVL		PROGRAM RUN	X	○	X	X
MVA, MVB, MVC, MVD		Independent Axis Operations	○	○	○	○
		Independent Axis Voltage Outputs (See note.)	○	○	○	○
		Independent Axis Ratio Operations (See note.)	○	○	○	○
Program run axes	ZRN	HOME RETURN	○	X	X	X
	JOG	JOG	○	X	○	X
	STP	STEP	○	X	○	X
Independent axes	ZRN	HOME RETURN	○	○	○	○
	JOG	JOG	○	○	○	○
	STP	STEP	○	○	○	○
SMD		SINGLE BLOCK MODE	○	○	○	○
MLK		MACHINE LOCK MODE	○	○	○	○
MRS		MODULE RESET	○	○	○	○
RST		MACHINE RESET	○	○	○	○
ESP		EMERGENCY STOP NOTIFICATION	○	○	○	○
ARS		ALARM RESET	○	○	○	○
MON		MONITOR	○	○	○	○
POS		COORDINATE SETTING	○	○	○	○
PRM		PARAMETER SETTING	○	○	○	○
VAR		H VARIABLE SETTING	○	○	○	○
PTBL		POINT TABLE SETTING	○	○	○	○
ZST		HOME POSITION SETTING	○	○	○	○

Note New step-2 functions.

○: Instructions that can be executed in the specified mode.
X : Instructions that cannot be executed in the specified mode.

Instructions Executable Simultaneously in the Same Scan

There are some ladder motion instructions that can be executed simultaneously, and some that cannot. The instructions that can be executed simultaneously for the same MC20 Module are shown in the following table.

Table 2.4 Instructions Executed Simultaneously in a Scan

Instruction in Progress		Instruction Executed Simultaneously																				
Symbol	Name	MOD	SVN	MVL	INDEPENDENT AXIS OPERATION	VOLTAGE OUTPUT	RATIO OPERATION	ZRN	JOG	STP	SMD	MLK	MRS	RST	ESP	ARS	MON	POS	PRM	VAR	PTBL	ZST
MOD	MODE SET	X	○	X	○	○	○	X	X	X	○	○	○	○	○	○	○	○	○	○	○	○
SVN	SERVO ON	○	Δ	X	X	X	X	X	X	X	○	○	○	○	○	○	○	X	○	○	○	X
MVL	PROGRAM RUN	○	○	X	Δ	Δ	Δ	X	X	X	○	○	○	○	○	○	○	Δ	X	X	X	X
MVA, MVB, MVC, MVD	Independent Axis Operations	○	○	○	Δ	Δ	Δ	Δ	Δ	Δ	○	○	○	○	○	○	○	Δ	X	○	X	X
	Independent Axis Voltage Outputs (See note.)	○	○	○	Δ	Δ	Δ	Δ	Δ	Δ	○	○	○	○	○	○	○	Δ	X	○	X	X
	Independent Axis Ratio Operations (See note.)	○	○	○	Δ	Δ	Δ	Δ	Δ	Δ	○	○	○	○	○	○	○	Δ	X	○	X	X
ZRN	HOME RETURN	○	○	X	Δ	Δ	Δ	Δ	Δ	Δ	○	○	○	○	○	○	○	Δ	X	○	○	X
JOG	JOG	○	○	X	Δ	Δ	Δ	Δ	Δ	Δ	○	○	○	○	○	○	○	Δ	X	○	○	X
STP	STEP	○	○	X	Δ	Δ	Δ	Δ	Δ	Δ	○	○	○	○	○	○	○	Δ	X	○	○	X
SMD	SINGLE BLOCK MODE	○	○	○	○	○	○	○	○	○	X	○	○	○	○	○	○	○	○	○	○	○
MLK	MACHINE LOCK MODE	○	○	○	○	○	○	○	○	○	○	X	○	○	○	○	○	○	○	○	○	X
MRS	MODULE RESET	X	X	X	X	X	X	X	X	X	X	○	X	X	X	X	X	X	X	X	X	X
RST	MACHINE RESET	○	X	X	X	X	X	X	X	X	X	○	X	X	○	○	○	○	X	○	○	X
ESP	EMERGENCY STOP NOTIFICATION	○	X	X	X	X	X	X	X	X	○	○	○	○	X	○	○	X	○	○	○	X
ARS	ALARM RESET	○	○	X	X	X	X	X	X	X	○	○	○	○	○	X	○	○	○	○	○	X
MON	MONITOR	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	X	○	○	○	○	○
POS	COORDINATE SETTING	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	X	X	X	X	X
PRM	PARAMETER SETTING	○	○	X	X	X	X	X	X	X	○	○	○	○	○	○	○	X	X	X	X	X
VAR	H VARIABLE SETTING	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	X	X	X	X	○

Symbol	Name	MOD	SVN	MVL	INDEPENDENT AXIS OPERATION	VOLTAGE OUTPUT	RATIO OPERATION	ZRN	JOG	STP	SMD	MLK	MRS	RST	ESP	ARS	MON	POS	PRM	VAR	PTBL	ZST
PTBL	POINT TABLE SETTING	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	X	X	X	X	○
ZST	HOME POSITION SETTING	○	X	X	X	X	X	X	X	X	○	X	X	X	X	X	○	X	X	○	○	Δ

Note New step-2 functions

○: Instructions that can be executed simultaneously.
Δ : Instructions that can be executed simultaneously if a different axis is specified.
X : Instructions that cannot be executed simultaneously in a scan.

Note There are some combinations that would be meaningless even though it would be technically possible to execute them simultaneously. One example would be to execute SMD while a JOG operation is being carried out. Another would be to execute ARS when there is no alarm during MVL execution.

Other Limiting Conditions

1) Some ladder motion instructions cannot be executed with respect to the same controlled axis in the same MC20 Module while the controlled axis is travelling. If this is attempted, a busy error or an MC busy error will occur.

2) Ladder motion instructions subject to limiting conditions can be written into the ladder logic program, but a busy error will occur when the instructions are executed.

2.2 Instructions for Moving Axes

This section explains how to program ladder motion instructions to move controlled axis. These are basic instructions, so be sure to familiarize yourself with them before proceeding further.

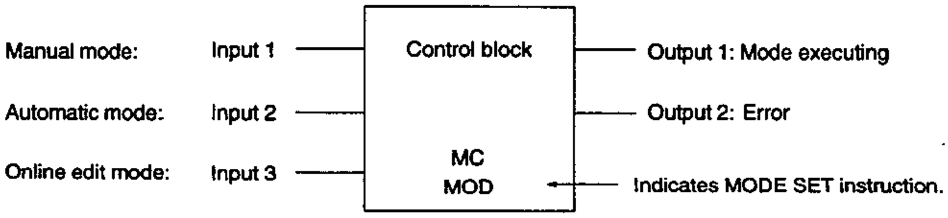
2.2.1 MODE SET (MOD)

Function

MOD switches the operation mode of the MC20 Module.

Structure

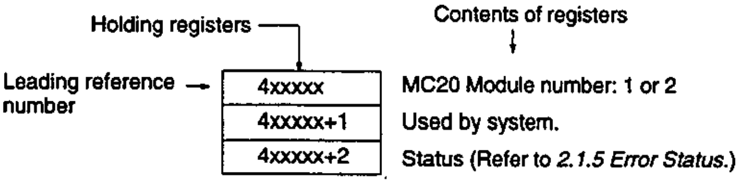
1) The structure of the MOD instruction is shown below.



MOD structure

2) The settings shown below are required for the control block at the top element. The leading reference number is specified for the top element. Refer to *Table 2.5 MOD Structural Elements* for details on the ranges of reference numbers that can be specified.

Control Block



4xxxxx	MC20 Module number: 1 or 2
4xxxxx+1	Used by system.
4xxxxx+2	Status (Refer to *2.1.5 Error Status*.)

Operation

1) The following table shows the basic operation of the inputs and outputs.

I/O Number	Operation Name	Description	Input Contacts
Input 1	Manual mode	When this input is ON, manual mode is in effect. JOG and STEP operations are enabled.	N.O. contact ─┤ ├─
Input 2	Automatic mode	When this input is ON, automatic mode is in effect. MVL operations are enabled.	N.O. contact ─┤ ├─
Input 3	Online edit mode	When this input is ON, online edit mode is in effect.	N.O. contact ─┤ ├─
Output 1	Mode executing	This output turns ON when the MC20 Module is in manual, automatic, or online edit mode.	
Output 2	Error	This output turns ON when an error occurs. For details, see 2.1.5 *Error Status.*	

Note (a) If inputs 1 through 3 are all OFF, the MC20 Module will be in edit mode.

(b) Use the contacts specified under "Input Contacts" to turn inputs 1 through 3 ON and OFF.

2) The following operations can be executed in online edit mode:

- The Programming Device can be used for programming while teaching positions.
- The Teach Pendant can be used for executing teach mode operations.

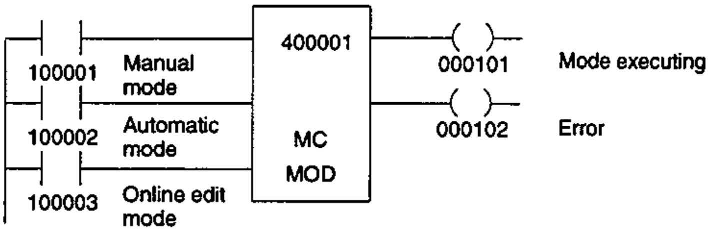
(4) In edit mode, the following files can be transferred from the Programming Device to the MC20 Module:

- Motion program
- Parameters

(5) When any or all of inputs 1 through 3 are turned ON together, the CPU Module transmits the mode signals that are ON to the MC20 Module. The MC20 Module goes into the mode for the mode signal that the MC20 Module recognizes first, and output 1 turns ON and output 2 turns OFF.

◀EXAMPLE▶ In this example, MC20 Module 1 is placed in automatic mode.

Ladder Diagram



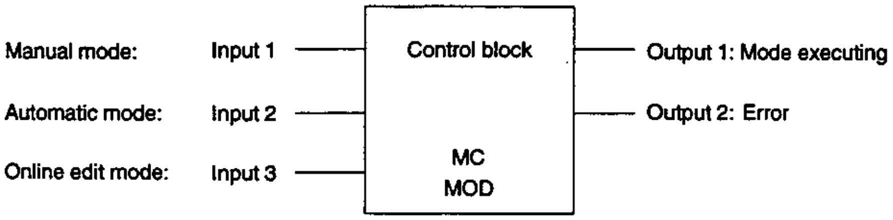
400001	1	MC20 Module no. designation
400002		Used by system.
400003		Status (Refer to 2.1.5 *Error Status.*)

Operation

The MC20 Module number is specified in holding register 400001. When the N.O. input is subsequently turned ON, MC20 Module 1 is switched to automatic mode and the "mode executing" output is turned ON.

Note Only one MOD instruction can be executed with respect to a single MC20 Module. Do not execute more than one instruction at the same time.

SUMMARY The following information summarizes the instruction.



MOD structure

Table 2.5 MOD Structural Elements

Element	Meaning		Possible Settings	
Top	4xxxxx	MC20 Module number: 1 or 2	Holding registers	400001 to 409997 (W00001 to W09997)
	4xxxxx+1	Used by system.	Do not write.	
	4xxxxx+2	Status		
Middle	None		---	
Bottom	Indicates the MOD instruction.		---	

Table 2.6 MOD Operation

Input 1	Input 2	Input 3	Status/Remarks	Output 1	Output 2
ON	OFF	OFF	• Sets manual mode. • MC control relay: MANL (manual mode status) is turned ON.	ON	OFF
OFF	ON	OFF	• Sets automatic mode. • MC control relay: MEML (automatic mode status) is turned ON.	ON	OFF
OFF	OFF	ON	• Sets online edit mode. • MC control relay: OEDL (online edit mode status) is turned ON.	ON	OFF
OFF	OFF	OFF	• No mode is set. • MC20 Module goes into edit mode.	OFF	OFF
ON	ON	OFF	• When any or all of inputs 1 through 3 are turned ON together, the CPU Module transmits the mode signals that are ON to the MC20 Module. • When the above occurs, the MC20 Module goes into the mode for the mode signal that the MC20 Module recognizes first.	ON	OFF
OFF	ON	ON			
ON	OFF	ON			
ON	ON	ON			

Note To verify the mode of the MC20 Module, check the MC control relays PN009 to PN011.

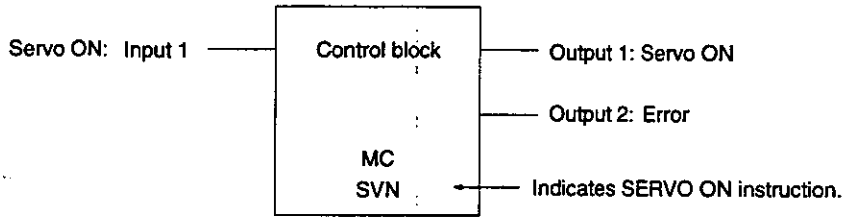
2.2.2 SERVO ON (SVN)

Function

SVN turns the servomotor power ON and OFF to the specified controlled axes.

Structure

1) The structure of the SVN instruction is shown below.



SVN structure

2) The settings shown below are required for the control block in the top element. The leading reference number is specified for the top element. Refer to *Table 2.7 SVN Structural Elements* for details on the ranges of reference numbers that can be specified.

Control Block

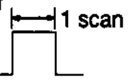
	Holding registers	Contents of registers
Leading reference number →	4xxxxx	MC20 Module number: 1 or 2
	4xxxxx+1	Controlled axis designation: 1, 2, 4, 8, or sum
	4xxxxx+2	Used by system.
	4xxxxx+3	Status (Refer to 2.1.5 *Error Status.*)

3) To designate the controlled axes, specify in holding register 4xxxxx+1 the numbers corresponding to the controlled axes for which the servomotor power is to be turned ON. The numbers are shown in the following table. To designate multiple axes with a single instruction, specify the sum of the desired axes. For example, if the desired axes are 1, 2, and 3, the corresponding numbers would be 1, 2, and 4. Therefore the sum would be 7 (i.e., 1 + 2 + 4 = 7).

Axis Number	Corresponding Number
Axis 1	1
Axis 2	2
Axis 3	4
Axis 4	8

Operation

1) The following table shows the basic operation of the inputs and outputs.

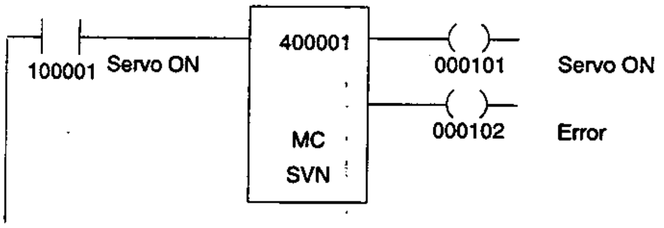
I/O Number	Operation Name	Description	Input Contacts
Input 1	Servo ON	When this input is ON, power is turned ON for the servomotor for the designated controlled axis.	N.O. contact ─┤ ├─
Output 1	Servo ON	This output turns ON when MC control relay SVNx (servo ON) is turned ON.	
Output 2	Error  ON pulse signal	When an error occurs, the ON pulse signal is output and the instruction is terminated. The pulse width is one scan. For details, see 2.1.5 *Error Status.*	

Note Use the contact specified under "Input Contacts" to turn input 1 ON and OFF.

2) If input 1 is turned ON and one second elapses before MC control relay SVNx (axis x servo ON) is turned ON, the ON pulse signal will be given for output 2 and a timeout error will occur.

◀EXAMPLE▶ In this example, servomotor power is turned ON for controlled axes 1, 2, and 3 for MC20 Module 2.

Ladder Diagram



400001	2	MC20 Module no. designation
400002	7	Controlled axis designation (= 1 + 2 + 4)
400003		Used by system.
400004		Status (Refer to *2.1.5 Error Status.*)

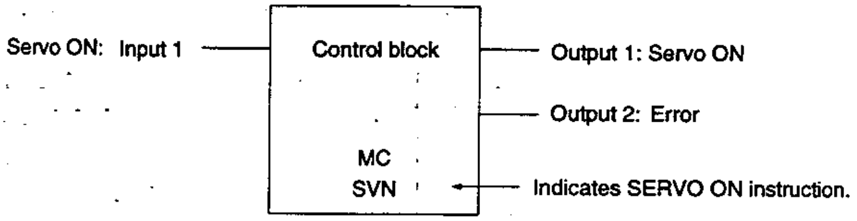
Operation

The MC20 Module "2" is specified in holding register 400001. The axis designation "7" (= 1 + 2 + 4) is set in holding register 400002.

When input 100001 is turned ON, power is supplied to the servomotor for each of axes 1, 2, and 3, and the servo ON output is turned ON.

Note A maximum of four SVN instructions can be executed with respect to a single MC20 Module. Do not execute two or more instructions at the same time.

SUMMARY The following information summarizes the instruction.



SVN structure

Table 2.7 SVN Structural Elements

Element	Meaning		Possible Settings	
Top	4xxxxx	MC20 Module number: 1 or 2	Holding registers	400001 to 409996 (W00001 to W09996)
	4xxxxx+1	Controlled axis designation: 1, 2, 4, 8, or sum (for multiple axis designation)	---	
	4xxxxx+2	Used by system.	Do not write.	
	4xxxxx+3	Status		
Middle	None		---	
Bottom	Indicates the SVN instruction.		---	

Table 2.8 SVN Operation

Input 1	Status/Remarks	Output 1	Output 2
ON	• The servomotor power for the specified controlled axis number will go ON when input 1 is ON. • The MC control relay (SVNx: Axis x Servo ON) for the specified axis number will go ON.	ON	OFF
OFF	• The servomotor power for the specified controlled axis number will go OFF when input 1 is OFF. • The MC control relay (SVNx: Axis x Servo ON) for the specified axis number will go OFF.	OFF	OFF

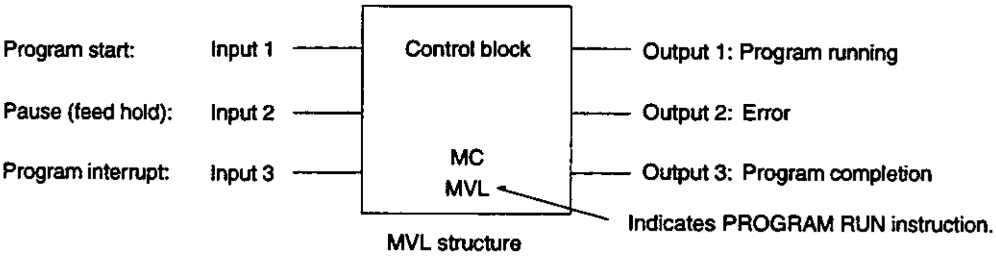
2.2.3 PROGRAM RUN (MVL)

Function

MVL executes the specified motion program (designated by program number), beginning with the specified block number.

Structure

1) The structure of the MVL instruction is shown below.



MVL structure

2) The settings shown below are required for the control block in the top element. The leading reference number is specified for the top element. Refer to *Table 2.9 MVL Structural Elements* for details on the ranges of reference numbers that can be specified.

Control Block

Leading reference number →

Holding registers	Contents of registers
4xxxxx	MC20 Module number: 1 or 2
4xxxxx+1	Program number (O no.): 1 to 99
4xxxxx+2	Block number (B no.): 1 to 999
4xxxxx+3	Timeout error check time: 0 to 20 s (see Note)
4xxxxx+4	Used by system.
4xxxxx+5	Status (Refer to *2.1.5 Error Status.*)

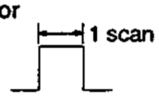
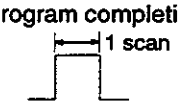
Note The timeout error check time default value is 0, and it is treated as 1 second.

3) The program number (O no.) setting specifies the MC20 Module motion program that is to be run.

4) The block number (B no.) setting specifies the beginning block within the designated program from which the program is to be run.

Operation

1) The following table shows the basic operation of the inputs and outputs.

I/O Number	Operation Name	Description	Input Contacts
Input 1	Program start	The program is reset when the leading edge of the OFF→ON signal for this input is detected. Then, the program is executed beginning from the designated block number.	Positive transitional contact (OFF to ON) ─┤P├─
Input 2	Pause (feed hold)	When this input turns ON, program execution is paused. Even while the program is paused, the status of output 1 remains unchanged (i.e., remains ON). When this input turns OFF, the pause is cleared and program execution is resumed.	N.O. contact ─┤ ├─
Input 3	Program interrupt	When this input turns ON, program execution is interrupted and stopped. At that point the program is reset, but the O no. and the B no. are not changed. When program execution is stopped by means of this input, output 1 is turned OFF and outputs 2 and 3 both remain OFF.	N.O. contact ─┤ ├─
Output 1	Program running	Output 1 remains ON from the point at which input 1 turns ON until program execution is completed (i.e., until MC control relay ENDL is turned ON). It is turned OFF, however, when a program interrupt is executed.	
Output 2	Error  ON pulse signal	When an error occurs, the ON pulse signal is output and the instruction is terminated. Output 3, however, is not turned ON. The pulse width is one scan. For details, see *2.1.5 Error Status.*	
Output 3	Program completion  ON pulse signal	The ON pulse signal is output when the program has been executed normally and completed (i.e., when MC control relay ENDL is turned ON). The pulse width is one scan.	

Note (a) Use the contacts specified under "Input Contacts" to turn inputs 1 through 3 ON and OFF.

(b) The pulse width for ON pulse signal outputs is one scan time.

2) The program can be executed when input 1 turns ON as long as both of the following conditions are satisfied:

- The MC20 Module is in automatic mode (i.e., MC control relay MEML is ON).

- Controlled axis movement has stopped (i.e., MC control relay MDEN is ON) or "single block stopped" is in effect.

3) To restart during **single-block stop status**, turn input 1 (program start) from OFF to ON. Single-block stop status cannot be cleared simply by turning OFF input 2 (pause).

4) When making the "timeout error check time" setting, allow enough time for the moving servo system to be stopped by the "program interrupt," and for the program to be initialized. The timeout error check time can be set in holding register 4xxxxx+3 of the control block. When "program interrupt" is specified by turning input 3 ON, a timeout error will occur if the program initialization has not been completed by the time the timeout error check time elapses.

5) To restart after program execution has been stopped by means of "program interrupt," turn input 1 (program start) from OFF to ON. Program interrupt status cannot be cleared simply by turning OFF input 3 (program interrupt).

6) An override can be set for the rapid traverse speed and maximum interpolation feed speed in 16 steps between 0% and 100% or in increments of 0.1% between 0% and 3276.7%. The override can also be enabled and disabled.

   a) Overrides for the rapid traverse speed and maximum interpolation feed speed are enabled and disable in parameter P0016.

      (1) Override: Enabled for MC control coils (16 steps)

      Ver. B08

      (2) Override: Enabled for MC link registers (in increments of 0.1%)

      (3) Override: Disabled

      The overrides are disabled in the default settings.

   b) When an override is enabled, the percentage can be switched as shown below.

      (1) **MC Control Coils (16 Steps)**
      0, 1, 2, 4, 6, 8, 10, 20, 30, 40, 50, 60, 80, 90, or 100 (%)

      Ver. B08

      (2) **MC Link Registers (0.1% Increments)**
      0.0 to 3276.7 (%)

      When an override is disabled, the speed will be 100% of the set value or reference value.

TERMS

Single-block Stop Status

This term indicates a state in which SINGLE BLOCK MODE (SMD) has been executed and the program has been stopped after the contents of one block have been completely executed while in the single block operation mode.

c) Overrides for the rapid traverse speed and maximum interpolation feed speed are set using the following MC control coils and MC link registers.

Ver. B08

Coil/Register		Use	References	
MC control coils				N is the MC20 Module number (N = 1 or 2)
	16 steps from 0% to 100%	Rapid traverse speed	QN0137 to QN0140	
	16 steps from 0% to 200%	Maximum interpolation feed speed	QN0141 to QN0144	
MC link registers 0.0% to 3276.7% in 0.1% increments		Rapid traverse speed	409909 and 409982	MC Modules 1 and 2 (default allocations)
		Maximum interpolation feed speed	409910 and 409983	

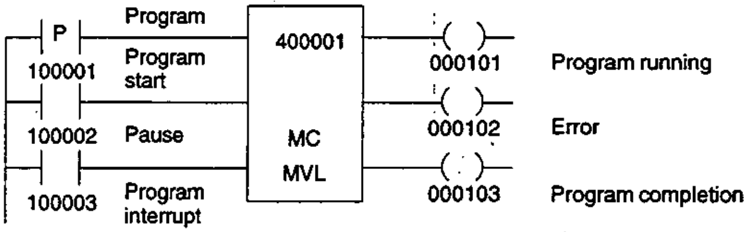
Set the MC link register to 10 times the override percentage when setting an override in 0.1% increments.

Refer to *3.2 MC Control Coil Functions* for details on MC control coils and to *1.5.6 Link Input Variables (#□□□□)* for details on MC link registers.

7) All basic axes will be checked by the MC20 Module when programmed operation is started and during programmed operation to see if the Servo is ON for them. If there is one or more axes with the Servo OFF, an alarm will be generated (A09: Servo Power OFF) and programmed operation will stop. This check is enabled or disabled in P0017 b2 Servo OFF Disable. The MC20 Module is set by default to check Servo ON status.

◀EXAMPLE▶ In this example, the MC20 Module 1 program is executed beginning from block number 10 of program number O05.

Ladder Diagram



400001	1	MC20 Module no. designation
400002	05	Program number (O no.) designation
400003	10	Block number (B no.) designation
400004	3	Timeout error check time (seconds) designation
400005		Used by system.
400006		Status (Refer to *2.1.5 Error Status.*)

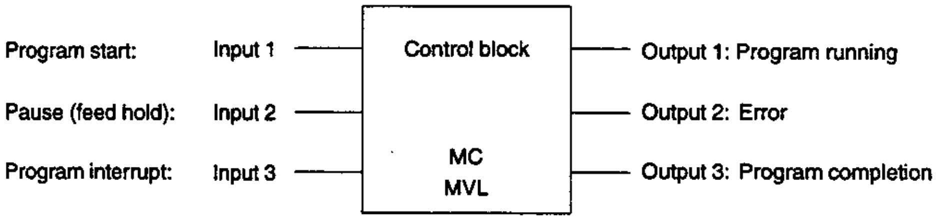
Operation

The settings for holding registers 400001 to 400004 are made as shown above.

When the transitional input 100001 turns from OFF to ON, the program in MC20 Module 1 is executed beginning from block number 10 of program number O05. When program execution begins, output 1 is turned ON; when program execution has been completed, output 1 is turned OFF.

Note Only one MVL instruction can be executed with respect to a single MC20 Module. Do not execute more than one instruction at the same time.

▶ SUMMARY The following information summarizes the instruction.



MVL structure

Table 2.9 MVL Structural Elements

Element	Meaning		Possible Settings	
Top	4xxxxx	MC20 Module number: 1 or 2	Holding registers	400001 to 409994 (W00001 to W09994)
	4xxxxx+1	Program number (O no.): 1 to 99	---	
	4xxxxx+2	Block number (B no.): 1 to 999	---	
	4xxxxx+3	Timeout error check time: 0 to 20 s ("0" is treated as 1 s.)	---	
	4xxxxx+4	Used by system.	Do not write.	
	4xxxxx+5	Status		
Middle	None		---	
Bottom	Indicates the MVL instruction.		---	

Table 2.10 MVL Operation

Input 1	Input 2	Input 3	Status/Remarks	Output 1	Output 2	Output 3
OFF ↓ ON	OFF	OFF	• At the leading edge of input 1, program reset is completed (see Note) and then program execution is started. • MC control relay STL (program running) is turned ON. • The output 3 ON pulse signal is output when the program has been executed normally and completed. MC control relay ENDL (program completion) is turned ON.	ON ↓ OFF	OFF	OFF ↓ ON pulse
ON or OFF	ON	OFF	• When input 2 is turned ON while the program is running, program execution is paused. • MC control relay STL (program running) remains ON and SPL (feed holding) turns ON.	ON	OFF	OFF
ON or OFF	OFF	OFF	• When input 2 is turned OFF while the program is paused, the pause is cleared and program execution is restarted. • MC control relay STL (program running) remains ON and SPL (feed holding) is turned OFF.	ON	OFF	OFF
ON or OFF	OFF	ON	• When input 3 is turned ON while the program is running, program reset is completed (see Note) and then program execution is interrupted and forcibly stopped. • When the program is interrupted, output 1 (program running) is turned OFF. MC control relay STL (program running) is turned OFF and ENDL (program completion) stays OFF. • Program execution can be restarted after the interruption by turning input 1 from OFF to ON.	OFF	OFF	OFF

Note "Program reset is completed" means that the MC relay PRSL (program reset completion) ON pulse signal is output.

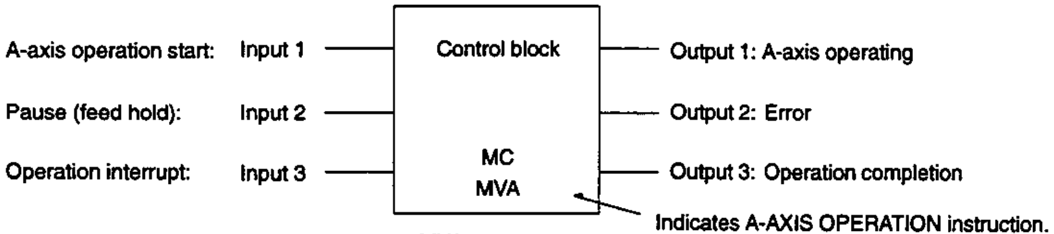
2.2.4 Independent Axis Operations (MVA, MVB, MVC, MVD)

Function

MVA, MVB, MVC, and MVD specify the movements of the four independent axes (A, B, C, and D). The settings are made using independent axis parameters P0001 to P0004. The operations of controlled axes set independently cannot be specified from the MC20 Module motion program.

Structure

1) The structure of the A-AXIS OPERATION (MVA) instruction is shown below.



MVA structure

2) The structures of B-AXIS OPERATION (MVB), C-AXIS OPERATION (MVC), and D-AXIS OPERATION (MVD) are the same as that for A-AXIS OPERATION (MVA) shown above. Only the instruction names are different. The following explanations are given in terms of A-AXIS OPERATION (MVA), but they apply equally to all of the axis operation instructions.

3) The settings shown below are required for the control block in the top element. Use the holding registers for making these settings. The leading reference number is specified for the top element. Refer to *Table 2.11 MVA Structural Elements* for details on the ranges of reference numbers that can be specified.

Control Block

	Data registers	Contents of registers	
Leading reference number →	4xxxxx	MC20 Module number: 1 or 2	
	4xxxxx+1	Instruction type: 0 to 3	
	4xxxxx+2	Target position - L	0 to ±99,999,999 or 1 to 500
	4xxxxx+3	Target position - H	
	4xxxxx+4	Feed speed - L	1 to 240,000
	4xxxxx+5	Feed speed - H	
	4xxxxx+6	Timeout error check time: 0 to 20 s (see Note)	
	4xxxxx+7	Used by system.	
	4xxxxx+8	Status (Refer to *2.1.5 Error Status.*)	

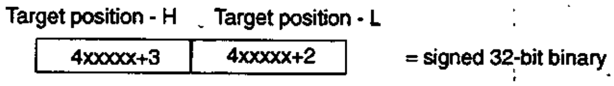
Note The default timeout error check time is 0, and it is treated as 1 s.

4) Instruction types 0 to 3 designate the instruction form for the target position.

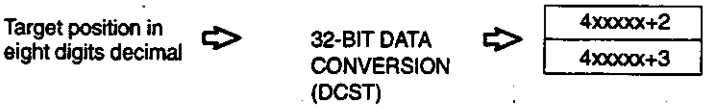
Instruction Type	Instruction Form	Setting Range
0	Target position direct designation, absolute value	0 to ±99,999,999
1	Target position direct designation, incremental value	
2	Point number designation, absolute value	1 to 500
3	Point number designation, incremental value	

Note The "point number" refers to the point number of the point table.

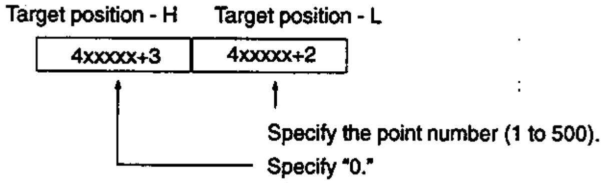
5) When "target position direct designation" is selected, two registers are used to directly specify the target position in **signed 32-bit binary.**



32-BIT DATA CONVERSION (DCST) instruction is convenient when carrying out direct designation of binary numbers.


Target position in eight digits decimal
32-BIT DATA CONVERSION (DCST)
4xxxxx+2
4xxxxx+3

6) When "point number designation" is selected, directly specify the point number (1 to 500) in decimal for target position L. Enter "0" for target position H. A-axis operation will be executed with the specified point number data as the target value.



7) The unit for inputting the value designated as the target position is determined by the MC20 Module's common parameter P0005 (decimal position).

MC20 Module Parameter Setting	Reference Unit	
	Linear Axis	Rotary Axis
P0005 = 1	0.1 mm	0.1°
P0005 = 2	0.01 mm	0.01°
P0005 = 3	0.001 mm	0.001°

TERMS

Signed 32-bit binary

The usage of "binary" here is defined as follows:
Positive value: Normal 32-bit binary.
Negative value: Expressed as two's complement.

8) The unit for inputting the value designated as the target position is determined by the MC20 Module's common parameter P0005 (decimal position).

◀EXAMPLE▶ Linear Axis:

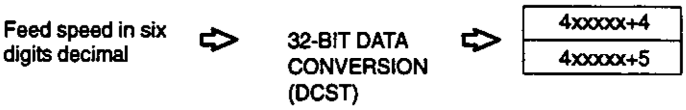
In this example, the target position designation is actually specified in binary but appears in decimal.

Target Position Designation	**A-axis Travel Distance**
(Instruction type = 1)	(P0005 = 3)
-350000	-350.000 mm

9) The feed speed H/L is directly designated in signed 32-bit binary, using two registers.

Feed speed - H	Feed speed - L	
4xxxxx+5	4xxxxx+4	= signed 32-bit binary

32-BIT DATA CONVERSION (DCST) instruction is convenient when carrying out direct designation of binary numbers.


Feed speed in six digits decimal
32-BIT DATA CONVERSION (DCST)
4xxxxx+4
4xxxxx+5

10) The value designated as the feed speed can be expressed in units of either mm/min or deg/min, within a range of F1 to F240000. This has no relation to the MC20 Module's common parameter P0005 (decimal position).

◀EXAMPLE▶ Linear Axis: In this example, the feed speed designation is actually specified in binary but appears in decimal.

Feed speed designation:	2500
A-axis feed speed:	2,500 mm/min

11) An override can be set for the feed speed in 16 steps between 0% and 100% or in increments of 0.1% between 0% and 3276.7%. The overrides can also be enabled and disabled.

a) Overrides for the feed speed are enabled and disable in parameter P0016.

(1) Override: Enabled for MC control coils (16 steps)

Ver. B08

(2) Override: Enabled for MC link registers (in increments of 0.1%)

(3) Override: Disabled

The overrides are disabled in the default settings.

b) When an override is enabled, the percentage can be switched as shown below.

(1) **MC Control Coils (16 Steps)**
0, 1, 2, 4, 6, 8, 10, 20, 30, 40, 50, 60, 80, 90, or 100 (%)

Ver. B08

(2) **MC Link Registers (0.1% Increments)**
0.0 to 3276.7 (%)

When an override is disabled, the speed will be 100% of the set value or reference value.

c) Overrides for the feed speed are set using the following MC control coils and MC link registers.

Ver. B08

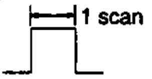
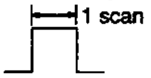
Coil/Register	Use		References	
MC control coils 16 steps from 0% to 100%	Independent axes or manual	Axis 1	QN0145 to QN0148	N is the MC20 Module number (N = 1 or 2)
		Axis 2	QN0149 to QN0152	
		Axis 3	QN0153 to QN0156	
		Axis 4	QN0157 to QN0160	
MC link registers 0.0% to 3276.7% in 0.1% increments	Independent axes or manual	Axis 1	409911 and 409984	MC Modules 1 and 2 (default allocations)
		Axis 2	409912 and 409985	
		Axis 3	409913 and 409986	
		Axis 4	409914 and 409987	

Set the MC link register to 10 times the override percentage when setting an override in 0.1% increments.

Refer to *3.2 MC Control Coil Functions* for details on MC control coils and to *1.5.6 Link Input Variables (#□□□□)* for details on MC link registers.

Operation

1) The following table shows the basic operation of the inputs and outputs.

I/O Number	Operation Name	Description	Input Contacts
Input 1	A-axis operation start	A-axis operation is started when the leading edge of the OFF→ON signal for this input is detected, and positioning to the target position is executed at the specified feed speed.	Positive transitional contact —| P |—
Input 2	Pause (feed hold)	When this input turns ON, A-axis operation is paused. Even while operation is paused, the status of output 1 remains unchanged (i.e., remains ON). When this input turns OFF, the pause is cleared and A-axis operation is resumed.	N.O. contact —| |—
Input 3	Operation interrupt	When this input turns ON, A-axis operation is interrupted and stopped. At this time, the remaining travelling data is cancelled. When A-axis operation is stopped by means of this input, output 1 is turned OFF and outputs 2 and 3 both remain OFF.	N.O. contact —| |—
Output 1	A-axis operating	Output 1 remains ON from the point at which input 1 turns ON until A-axis operation is completed (i.e., until MC control relay ENDL is turned ON). It is turned OFF, however, when an operation interrupt is executed.	
Output 2	Error  ON pulse signal	When an error occurs, the ON pulse signal is output and the instruction is terminated. Output 3, however, is not turned ON. The pulse width is one scan. For details, see *2.1.5 Error Status.*	
Output 3	Operation completion  ON pulse signal	The ON pulse signal is output when the program has been executed normally and completed (i.e., when MC control relay ENDL is turned ON). The pulse width is one scan.	

Note (a) Use the contacts specified under "Input Contacts" to turn inputs 1 through 3 ON and OFF.

(b) The pulse width for ON pulse signal outputs is one scan time.

2) A-axis operation can be executed when input 1 turns ON as long as both of the following conditions are satisfied:

- Input 2, input 3, output 1, output 2, and output 3 are all OFF.

- A-axis is not travelling (i.e., MC control relay MOVA is OFF).

3) When setting the timeout error check time, allow enough time for the moving servo system to be stopped by the operation interrupt, and for the remaining data to be cancelled. The timeout error check time can be set in holding register 4xxxxx+6 of the control block. When "operation interrupt" is specified by input 3 turning ON, a timeout error will occur if the remaining data has not been cancelled by the time the timeout error check time elapses.

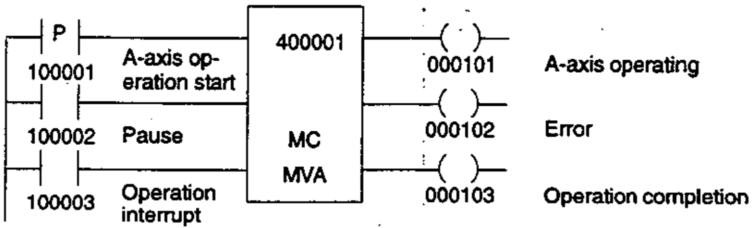
4) To restart after A-axis operation has been stopped by means of the operation interrupt, turn input 1 (A-axis operation start) from OFF to ON. Restarting cannot be done simply by turning input 3 (operation interrupt) OFF.

5) The instruction type, target position, and feed speed values designated by A-AXIS OPERATION will not be affected by executing the operation interrupt.

INFO Use MC control relays MOVA to MOVD (A-axis to D-axis travelling) to check whether axes A through D are travelling. MC control relays MOV1 to MOV4 cannot be used because there is no way of knowing which axes correspond to which independent axes.

◀EXAMPLE▶ In this example, A-axis operation is executed for MC20 Module 1 with a target position of point number 500 (absolute value) and a feed speed of 1,200 mm/min.

Ladder Diagram



400001	1	MC20 Module no. designation
400002	2	Instruction type designation
400003	500	Target position - L
400004	0	Target position - H
400005	Binary	Feed speed - L
400006	1200	Feed speed - H
400007	3	Timeout error check time (seconds) designation
400008		Used by system.
400009		Status (Refer to *2.1.5 Error Status.*)

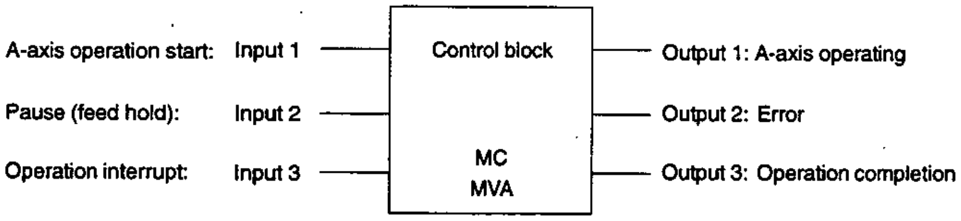
Operation

The settings for holding registers 400001 to 400007 are made as shown above.

When the transitional input 100001 turns from OFF to ON, A-axis positioning for MC20 Module 1 is executed at the specified feed speed, and to the target coordinate position specified by the contents of point number 500 in the point table. When A-axis operation begins, output 1 is turned ON; when the operation has been completed, output 1 is turned OFF.

Note Any or all of the four independent axes (A, B, C, and D) can be operated simultaneously using the AXIS OPERATION instructions. If an attempt is made to execute it at the same time for the same axis, however, a busy error will occur.

SUMMARY The following information summarizes the instruction.


A-axis operation start: Input 1
Pause (feed hold): Input 2
Operation interrupt: Input 3
Control block
MC
MVA
Output 1: A-axis operating
Output 2: Error
Output 3: Operation completion

MVA structure

Table 2.11 MVA Structural Elements

Element	Meaning		Possible Settings	
Top	4xxxxx	MC20 Module number: 1 or 2	Holding registers	400001 to 409991 (W00001 to W09991)
	4xxxxx+1	Instruction type: 0 to 3 0: Target position, absolute value 1: Target position, incremental value 2: Point number, absolute value 3: Point number, incremental value	---	
	4xxxxx+2 4xxxxx+3	Target position - L Target position - H Target position: 0 to ±99,999,999 Point number: 1 to 500	---	
	4xxxxx+4 4xxxxx+5	Feed speed - L Feed speed - H Feed speed: 1 to 240,000 (mm/min or deg/mm)	---	
	4xxxxx+6	Timeout error check time: 0 to 20 s ("0" is treated as 1 s.)	---	
	4xxxxx+7 4xxxxx+8	Used by system. Status	Do not write.	
Middle	None		---	
Bottom	Indicates the MVA instruction.		---	

Table 2.12 MVA Operation

Input 1	Input 2	Input 3	Status/Remarks	Output 1	Output 2	Output 3
OFF ↓ ON	OFF	OFF	• At the leading edge of input 1, A-axis operation starts and positioning to the target position is executed at the specified feed speed. • MC control relay STAL (A-axis operating) is turned ON. • The output 3 ON pulse signal is output when the A-axis operation has been completed. MC control relay ENDA (operation completion) is turned ON.	ON ↓ OFF	OFF	OFF ↓ ON pulse
ON or OFF	ON	OFF	• If input 2 is turned ON while A-axis is being operated, the operation will be paused. • MC control relay STAL (A-axis operating) remains ON and SPAL (A-axis pause) turns ON.	ON	OFF	OFF
ON or OFF	OFF	OFF	• When input 2 is turned OFF while A-axis operation is paused, the pause is cleared and A-axis operation is restarted. • MC control relay STAL (A-axis operating) remains ON and SPAL (A-axis pause) is turned OFF.	ON	OFF	OFF
ON or OFF	OFF	ON	• When input 3 is turned ON while A-axis is being operated, operation is interrupted and forcibly stopped. • When A-axis operation is interrupted, MC control relay STAL (A-axis operating) is turned OFF and ENDA (A-axis operation completion) stays OFF. • A-axis operation can be restarted after the interruption by turning input 1 from OFF to ON.	OFF	OFF	OFF

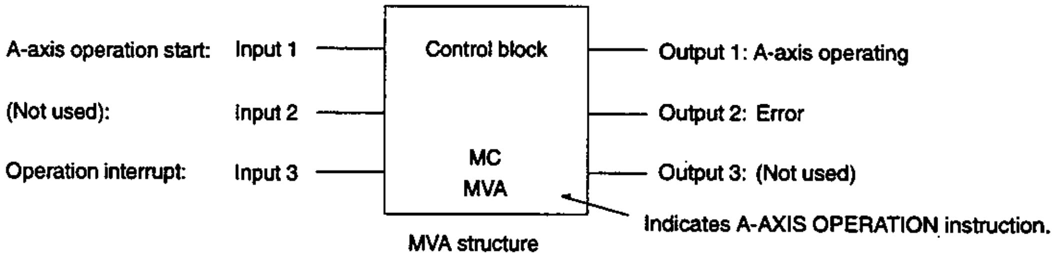
2.2.5 Independent Axis Voltage Outputs (MVA, MVB, MVC, MVD)

Function

The independent axis operation instructions (MVA, MVB, MVC, MVD) specify the target position and feed speed for moving a normal servo axis. For details, refer to the previous section. This section explains the independent axis operation instructions when an independent axis (A, B, C, or D) is set as a voltage output axis. For an overview and the preparations, refer to *1.9.2 VOLTAGE OUTPUT (VCC).*

Structure

1) The structure of the A-AXIS OPERATION (MVA) instruction is shown below.


A-axis operation start: Input 1
(Not used): Input 2
Operation interrupt: Input 3
Control block
MC
MVA
Output 1: A-axis operating
Output 2: Error
Output 3: (Not used)
Indicates A-AXIS OPERATION instruction.

MVA structure

2) The structures of B-AXIS OPERATION (MVB), C-AXIS OPERATION (MVC), and D-AXIS OPERATION (MVD) are the same as that for A-AXIS OPERATION (MVA) shown above. Only the instruction names are different. The following explanations are given in terms of A-AXIS OPERATION (MVA), but they apply equally to all of the axis operation instructions.

3) The settings shown below are required for the control block in the top element. Use the holding registers to make these settings. The leading reference number is specified for the top element. See *Table 2.11 MVA Structural Elements* for details on the ranges of reference numbers that can be specified.

Control Block

Leading reference number → 4xxxxx

Holding registers	Contents of registers (With servo axis)		(With voltage output axis)	
4xxxxx	MC20 Module number: 1 or 2		Same as at left	
4xxxxx+1	Instruction type: 0 to 3		Instruction type: 4	
4xxxxx+2	Target position - L	0 to ±99,999,999 or 1 to 500	Voltage reference value - L	-10,000 to +10,000
4xxxxx+3	Target position - H		Voltage reference value - H	
4xxxxx+4	Feed speed - L	1 to 240,000	Acceleration/deceleration time - L	0 to 99,999
4xxxxx+5	Feed speed - H		Acceleration/deceleration time - H	
4xxxxx+6	Timeout error check time: 0 to 20 s (see Note)		Same as at left	
4xxxxx+7	Used by system.		Same as at left	
4xxxxx+8	Status (Refer to *2.1.5 Error Status.*)		Same as at left	

Note The default timeout error check time is 0, and it is treated as 1 s.

4) The instruction form for holding registers 4xxxxx+2 to 4xxxxx+5 is specified using instruction types 0 to 8.

Instruction Types	Instruction Form	Setting Range
0	Target position direct designation, absolute value	0 to ±99,999,999
1	Target position direct designation, incremental value	
2	Point number designation, absolute value	1 to 500
3	Point number designation, incremental value	
4	Voltage reference value designation (absolute value)	-10,000 to +10,000
5 to 8	Ratio operation designation	1 to 9999 (denominator)

Note Specify "4" as the instruction type for a voltage output axis.

5) Specify "4" as the instruction type, and specify A-axis as a voltage output axis.

6) If the voltage reference – H/L is to be specified, two registers are used to set the voltage reference in signed 32-bit binary.

Voltage reference value - H	Voltage reference value - L	
4xxxxx+3	4xxxxx+2	= Signed 32-bit binary

The 32-BIT DATA CONVERSION (DCST) instruction is convenient when directly designating binary numbers.

8-digit decimal data ⇨ SET WORD DATA (SDAT) instruction
32-BIT DATA CONVERSION (DCST) instruction
SET BIT (SBIT) instruction ⇨

4xxxxx+n	L
4xxxxx+n+1	H

32-bit binary data

For details, see item *6)* in *1.5.6 Link Input Variables (#□□□□).*

7) The setting range and unit for the voltage reference value are shown below. They are not affected by parameter P0005 (decimal point position) of the MC20 Module.

Setting range: -10000 to +10000
Unit: 1 = 0.001 [V] (fixed)

TERMS

Signed 32-bit binary

The usage of "binary" here is defined as follows:
Positive value: Normal 32-bit binary.
Negative value: Two's complement.

◀EXAMPLE▶ Linear Axis: In this example, the voltage reference value designation is specified in binary but appears in decimal.

Voltage reference designation (instruction type= 4)		Target voltage reference value
-4500	=	-4.500

8) Specify the voltage reference as an absolute value. The default voltage reference is 0 V.

9) The voltage reference is valid only while operating with the MVA instruction. If the operation is interrupted by the MVA instruction, the voltage reference automatically becomes 0.

10) If the acceleration/deceleration time – H/L is to be specified, two registers are used to set the linear acceleration/deceleration time in signed 32-bit binary.

Acceleration/ deceleration time - H	Acceleration/ deceleration time - L	
4xxxxx+5	4xxxxx+4	= Signed 32-bit binary

The 32-BIT DATA CONVERSION (DCST) instruction is convenient when directly designating binary numbers.

8-digit decimal data ⇨ SET WORD DATA (SDAT) instruction
32-BIT DATA CONVERSION (DCST) instruction
SET BIT (SBIT) instruction ⇨

32-bit binary data	
4xxxxx+n	L
4xxxxx+n+1	H

For details, refer to item *6)* in *1.5.6 Link Input Variables.*

11) The setting range and unit for the acceleration/deceleration time are shown below. They are not affected by parameter P0005 (decimal point position) of the MC20 Module.

Setting range: 0 to 99999
Unit: 1 = 0.001 [V] (fixed)

◀EXAMPLE▶ Linear Axis: In this example, the acceleration/deceleration time designation is specified in binary but appears in decimal.

Acceleration/deceleration time designation		A-axis linear acceleration/deceleration time
2500	=	2.500 s

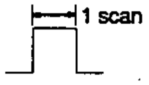
12) An explanation of acceleration/deceleration time is provided below.

- The acceleration/deceleration time is the time taken for linear acceleration/deceleration from a voltage reference value of 0 until the target voltage reference value is reached.

- The acceleration/deceleration time in the MVA instruction is not modal. If the operation is interrupted by the MVA instruction, it is reset to 0 s.

Operation

1) The following table shows the basic operation of the inputs and outputs.

I/O Number	Operation Name	Description	Input Contacts
Input 1	A-axis operation start	A-axis operation is started when the leading edge (OFF→ON) of this input is detected. The specified voltage reference value is output at the specified acceleration/deceleration time.	Positive transitional contacts —| P |—
Input 2	---	Not used.	—
Input 3	Operation interrupt	When this input turns ON, A-axis operation is interrupted and stopped. At this time, the voltage reference value is reset to 0 V. When A-axis operation is stopped by this input, output 1 is turned OFF and output 2 (error) remains OFF.	N.O. contacts —| |—
Output 1	A-axis operating	Output 1 remains ON from the point at which input 1 turns ON until A-axis operation is completed (i.e., until MC control relay ENDL is turned ON). It is turned OFF, however, if operation is interrupted.	
Output 2	Error  ON pulse signal	When an error occurs, an ON pulse signal is output and the instruction is ended. The pulse width is one scan. For details, see *2.1.5 Error Status.*	
Output 3	---	Not used.	

Note (a) Use the contacts specified under "Input Contacts" to turn ON and OFF inputs 1 through 3.

(b) The pulse width for ON pulse signal outputs is one scan time.

2) A-axis operation can be executed when input 1 turns ON as long as both of the following conditions are satisfied:

- Input 3, output 1, and output 2 are all OFF.

- The A-axis is not moving (i.e., MC control relay MOVA is OFF).

3) When setting the timeout error check time, allow enough time for the moving servo system to be stopped by the operation interrupt, and for the remaining data to be cancelled. The timeout error check time can be set in holding register 4xxxxx+6 of the control block. When input 3 is turned ON to interrupt operation, a timeout error will occur if the remaining data has not been cancelled by the time the timeout error check time elapses.

4) To restart after A-axis operation has been interrupted, turn input 1 (A-axis operation start) from OFF to ON. Restarting cannot be done simply by turning input 3 (operation interrupt) OFF.

5) An override can be set for the voltage reference in 16 steps between 0% and 100% or in increments of 0.1% between 0% and 3276.7%. The overrides can also be enabled and disabled.

a) Overrides for the voltage references are enabled and disable in parameter P0016.

(1) Override: Enabled for MC control coils (16 steps)

Ver. B08

(2) Override: Enabled for MC link registers (in increments of 0.1%)

(3) Override: Disabled

The overrides are disabled in the default settings.

b) When an override is enabled, the percentage can be switched as shown below.

(1) **MC Control Coils (16 Steps)**
0, 1, 2, 4, 6, 8, 10, 20, 30, 40, 50, 60, 80, 90, or 100 (%)

Ver. B08

(2) **MC Link Registers (0.1% Increments)**
0.0 to 3276.7 (%)

When an override is disabled, the speed will be 100% of the set value or reference value.

c) Overrides for voltage references are set using the following MC control coils and MC link registers.

Ver. B08

Coil/Register	Use		References	
MC control coils 16 steps from 0% to 100%	Independent axes or manual	Axis 1	QN0145 to QN0148	N is the MC20 Module number (N = 1 or 2)
		Axis 2	QN0149 to QN0152	
		Axis 3	QN0153 to QN0156	
		Axis 4	QN0157 to QN0160	
MC link registers 0.0% to 3276.7% in 0.1% increments	Independent axes or manual	Axis 1	409911 and 409984	MC Modules 1 and 2 (default allocations)
		Axis 2	409912 and 409985	
		Axis 3	409913 and 409986	
		Axis 4	409914 and 409987	

Set the MC link register to 10 times the override percentage when setting an override in 0.1% increments.

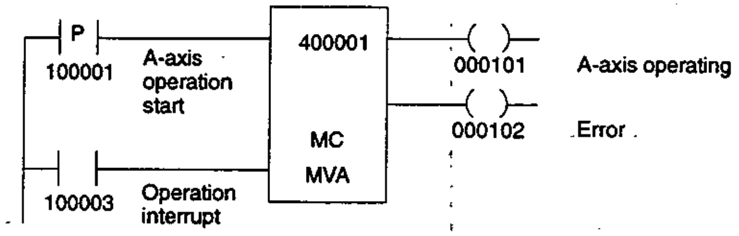
Refer to *3.2 MC Control Coil Functions* for details on MC control coils and to *1.5.6 Link Input Variables (#□□□□)* for details on MC link registers.

INFO

Use MC control relays MOVA to MOVD (A-axis to D-axis travelling) to check whether axes A through D are moving. MC control relays MOV1 to MOV4 cannot be used because there is no way of knowing which axes correspond to which independent axes.

◀**EXAMPLE**▶ In this example, a voltage reference value of 0 V to 6 V is output to the A-axis in a linear acceleration/deceleration time of 1.5 s for MC20 Module number 1.

Ladder Diagram



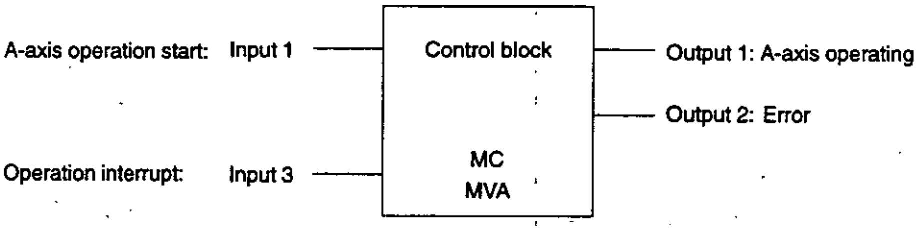
400001	1	MC20 Module no. designation
400002	4	Instruction type designation
400003	Binary	Voltage reference value - L
400004	6000	Voltage reference value - H
400005	Binary	Acceleration/deceleration time - L
400006	1500	Acceleration/deceleration time - H
400007	3	Timeout error check time (seconds) designation
400008		Used by system.
400009		Status (Refer to *2.1.5 Error Status.*)

Operation

The settings for holding registers 400001 to 400007 are made as shown above. When transitional contact input 100001 turns from OFF to ON, a voltage reference value of 0 V to +6 V is output in a linear acceleration/deceleration time of 1.5 seconds from the A-axis of MC20 Module 1. When A-axis operation begins, output 1 is turned ON. Output 1 will turn OFF if operation is interrupted.

Note Any or all of the four independent axes (A, B, C, and D) can be operated simultaneously using the AXIS OPERATION (MVA, MVB, MVC, MVD) instructions. If an attempt is made to execute an more than one instruction at the same time for the same axis, however, a busy error will occur.

➧ **SUMMARY** The following information summarizes the instruction.


A-axis operation start: Input 1
Operation interrupt: Input 3
Control block
MC
MVA
Output 1: A-axis operating
Output 2: Error

MVA structure

Table 2.13 MVA Structural Elements

Element	Meaning	Possible Settings	
Top	4xxxxx MC20 Module number: 1 or 2	Holding registers	400001 to 409991 (W00001 to W09991)
	4xxxxx+1 Instruction type: 0 to 3 0: Target position designation, absolute value 1: Target position designation, incremental value 2: Point number designation, absolute value 3: Point number designation, incremental value 4: Voltage reference value designation (absolute value) 5 to 8: Specify the ratio operation	---	
	4xxxxx+2 Voltage reference value - L 4xxxxx+3 Voltage reference value - H Setting range: -10,000 to +10,000 Unit: 1 = 0.001 V	---	
	4xxxxx+4 Acceleration/deceleration time - L 4xxxxx+5 Acceleration/deceleration time - H Setting range: 0 to +99,999 Unit: 1= 0.001 [s]	---	
	4xxxxx+6 Timeout error check time: 0 to 20 s ("0" is treated as 1 s.)	---	
	4xxxxx+7 Used by system. 4xxxxx+8 Status	Do not write.	
Middle	None	---	
Bottom	Indicates the MVA instruction.	---	

Table 2.14 MVA Operation

Input 1	Input 2	Input 3	Status/Remarks	Output 1	Output 2	Output 3
OFF ↓ ON	Not used	OFF	• A-axis operation starts at the leading edge of input 1. The specified voltage reference value is output at the specified acceleration/deceleration time. • MC control relay STAL (A-axis operating) is turned ON. • When A-axis operation has been completed, MC control relay ENDAL (A-axis operation completion) turns ON.	ON ↓ OFF	OFF	Not used
ON or OFF	Not used	ON	• When input 3 is turned ON during A-axis operation, operation is interrupted and forcibly stopped. • When A-axis operation is interrupted, MC control relay STAL (A-axis operating) is turned OFF and ENDA (A-axis operation completion) remains OFF. • A-axis operation can be restarted after the interruption by turning input 1 from OFF to ON.	OFF	OFF	Not used

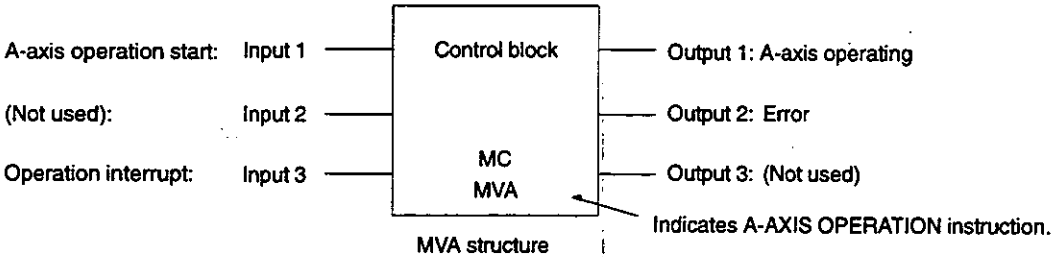
2.2.6 Independent Axis Ratio Operations (MVA, MVB, MVC, MVD)

Function

The independent axis operation instructions (MVA, MVB, MVC, MVD) specify the target position and feed speed for moving a normal servo axis. For details, see *2.2.4 Independent Axis Operations (MVA, MVB, MVC, MVD)* in this chapter. The independent axis operation instructions can be used when independent axes (A, B, C, or D) are specified as slave axes for ratio operation. For an overview and the preparations, refer to *1.10.2 RATIO OPERATION (PGS).*

Structure

1) The structure of the A-AXIS OPERATION (MVA) instruction is shown below.



MVA structure

2) The structures of B-AXIS OPERATION (MVB), C-AXIS OPERATION (MVC), and D-AXIS OPERATION (MVD) are the same as that for A-AXIS OPERATION (MVA) shown above. Only the instruction names are different. The following explanations are given in terms of A-AXIS OPERATION (MVA), but they apply equally to all of the axis operation instructions.

3) The settings shown below are required for the control block in the top element. Use the holding registers to make these settings. The leading reference number is specified for the top element. See *Table 2.11 MVA Structural Elements* for details on the ranges of reference numbers that can be specified.

Control Block

Leading reference number →

Holding registers	Contents of registers (With servo axis)		(With voltage output axis)	
4xxxxx	MC20 Module number: 1 or 2		Same as at left	
4xxxxx+1	Instruction type: 0 to 3		Instruction type: 4	
4xxxxx+2	Target position - L	0 to ±99,999,999 or 1 to 500	Ratio denominator - L	1 to 9,999
4xxxxx+3	Target position - H		Ratio denominator - H	
4xxxxx+4	Feed speed - L	1 to 240,000	Ratio numerator - L	±1 to ±9,999
4xxxxx+5	Feed speed - H		Ratio numerator - H	
4xxxxx+6	Timeout error check time: 0 to 20 s (see Note)		Same as at left	
4xxxxx+7	Used by system.		Same as at left	
4xxxxx+8	Status (See *2.1.5 Error Status.*)		Same as at left	

Note The default timeout error check time is 0, and it is treated as 1 s.

4) As shown in the following table, the instruction method for holding registers 4xxxxx+2 to 4xxxxx+5 is specified using instruction types (0 to 8).

Instruction Type	Instruction Form	Setting Range
0	Target position direct designation, absolute value	0 to ±99,999,999
1	Target position direct designation, incremental value	
2	Point number designation, absolute value	1 to 500
3	Point number designation, incremental value	
4	Voltage reference value designation (absolute value designation)	-10,000 to +10,000
5	Ratio operation designation, master axis = axis 1	1 to 9,999 (denominator)
6	Ratio operation designation, master axis = axis 2	1 to 9,999 (denominator)
7	Ratio operation designation, master axis = axis 3	1 to 9,999 (denominator)
8	Ratio operation designation, master axis = axis 4	1 to 9,999 (denominator)

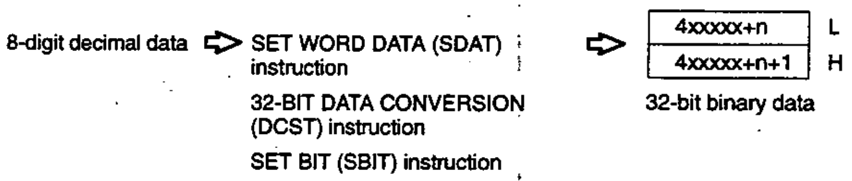
Note When performing ratio operation, select any one of 5 through 8 as the instruction type.

5) Setting instruction type 5 through 8 specifies the ratio operation master axis and at the same time specifies the A-axis as the slave axis.

6) If the ratio denominator - H/L is to be specified, two registers are used to set the ratio denominator in signed 32-bit binary.

Ratio denominator - H	Ratio denominator - L	
4xxxxx+3	4xxxxx+2	= Signed 32-bit binary

The 32-BIT DATA CONVERSION (DCST) instruction is convenient when directly designating binary numbers.


8-digit decimal data
SET WORD DATA (SDAT) instruction
32-BIT DATA CONVERSION (DCST) instruction
SET BIT (SBIT) instruction
4xxxxx+n
L
4xxxxx+n+1
H
32-bit binary data

For details, see item *6)* in *1.5.6 Link Input Variables (#□□□□).*

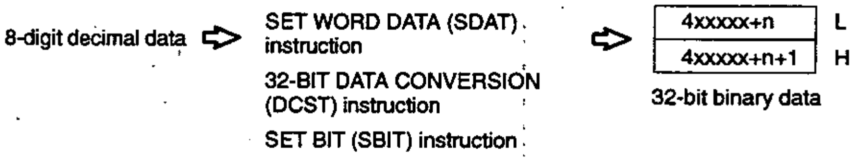
7) The setting range for the ratio denominator is as follows:

Setting range: 1 to 9,999

8) If the ratio numerator – H/L is to be specified, two registers are used to set the ratio numerator in signed 32-bit binary.

Ratio numerator - L	Ratio numerator - H	
4xxxxx+5	4xxxxx+4	= signed 32-bit binary

The 32-BIT DATA CONVERSION (DCST) instruction is convenient when directly designating binary numbers.


8-digit decimal data
SET WORD DATA (SDAT) instruction
32-BIT DATA CONVERSION (DCST) instruction
SET BIT (SBIT) instruction
4xxxxx+n
L
4xxxxx+n+1
H
32-bit binary data

For details, see item *6)* in *1.5.6 Link Input Variables (#□□□□).*

9) The setting range for the ratio numerator is as follows:

Setting range: ±1 to ±9,999

TERMS

Signed 32-bit binary

The usage of "binary" here is defined as follows:
Positive value: Normal 32-bit binary.
Negative value: Expressed as two's complement.

Note (1) In the following cases, an MC busy error will occur in the MVA instruction, and the MVA instruction's output 2 (error) will turn ON.

(a) An unused axis was specified as the master axis.

(b) The slave axis was specified as the master axis.
For example, when the A-axis is axis 1, the MVA instruction has been executed as instruction type 5.

(c) Another ladder motion instruction (such as JOG or STEP) has been executed for the slave axis (A-axis).

(d) Zero (0) has been specified in the ratio numerator or denominator, so ratio operation cannot been executed.

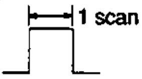
(2) In the following cases, an MC busy error will occur in the COORDINATE SETTING (POS) instruction and the POS instruction's output 2 (error) will turn ON.

(a) A COORDINATE SETTING (POS) ladder motion instruction (such as JOG or STEP) was executed for the slave axis (A-axis).

(3) If the final slave axis speed exceeds the value set in parameter PA201 (maximum feed speed), the following alarm will be generated.
Alarm: A11 (speed trailing disabled)

Operation

1) The following table shows the basic operation of the inputs and outputs.

I/O Number	Operation Name	Description	Input Contacts
Input 1	A-axis operation start	A-axis operation is started when the leading edge (OFF→ON) of this input is detected. The slave axis (A-axis) operates at a travel distance equivalent to the travel distance of the specified master axis multiplied by the ratio (numerator/denominator).	Positive transitional contact —| P |—
Input 2	---	Not used.	---
Input 3	Operation interrupt	When this input turns ON, A-axis operation is interrupted and stopped. The ratio operation is cancelled. When A-axis operation is stopped by means of this input, output 1 is turned OFF while output 2 (error) remains in OFF.	N.O. contact —| |—
Output 1	A-axis operating	Output 1 remains ON from the point at which input 1 turns ON until A-axis operation is completed (i.e., until MC control relay ENDAL is turned ON). It is turned OFF, however, when operation is interrupted.	
Output 2	Error  ON pulse signal	When an error occurs, the ON pulse signal is output and the instruction is terminated. The pulse width is one scan. For details, see *2.1.5 Error Status.*	
Output 3	---	Not used.	

Note (a) Use the contacts specified under "Input Contacts" to turn ON and OFF inputs 1 through 3.

(b) The pulse width for ON pulse signal outputs is one scan time.

2) A-axis operation can be executed when input 1 turns ON as long as both of the following conditions are satisfied:

- Input 3, output 1, and output 2 are all OFF.

- A-axis is not moving (i.e., MC control relay MOVA is OFF).

3) When setting the timeout error check time, allow enough time for the moving servo system to be stopped by the operation interrupt, and for the remaining data to be cancelled. The timeout error check time can be set in holding register 4xxxxx+6 of the control block. When input 3 is turned ON to interrupt operation, a timeout error will occur if the remaining data has not been cancelled by the time the timeout error check time elapses.

4) To restart after A-axis operation has been interrupted, turn input 1 (A-axis operation start) from OFF to ON. Restarting cannot be done simply by turning OFF input 3 (operation interrupt).

5) Overrides can be applied independently to the ratio operation master axis and the slave axis respectively.

INFO

1) The ratio cannot be changed while a ratio operation is being executed using the A-AXIS OPERATION (MVA) instruction. To change the ratio, forcibly end the MVA instruction, reset the ratio, and re-execute the MVA instruction. If the data in the holding registers in which the ratio is specified is to be forcibly changed while the MVA instruction is executing, the ratio operation will be forcibly ended by the MVA instruction.

2) In the following cases, the ratio operation for all axes will be cancelled.

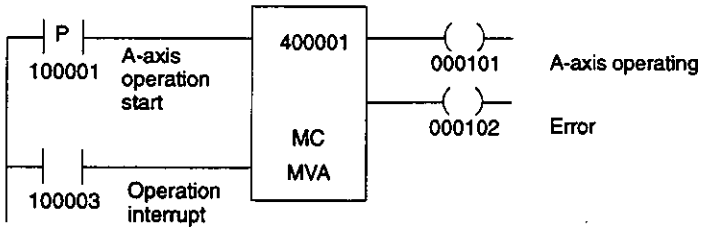
a) A alarm has been generated in the MC20 Module.

b) The MODULE RESET (MRS) instruction or the MACHINE RESET (RST) instruction has been executed.

3) Use MC control relays MOVA to MOVD (A-axis to D-axis travelling) to check whether axes A through D are moving. MC control relays MOV1 to MOV4 cannot be used because there is no way of knowing which axes correspond to which independent axes.

◀EXAMPLE▶ A ratio operation in which axis 1 is the master axis, axis 4 is the slave axis (A-axis), and the ratio operation is performed at 100/200 for MC20 Module No.1.

Ladder Diagram



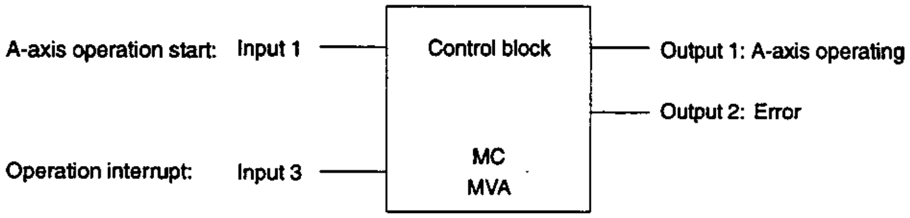
400001	1	MC20 Module no. designation
400002	5	Instruction type designation
400003	Binary	Ratio denominator - L
400004	200	Ratio denominator - H
400005	Binary	Ratio numerator - L
400006	100	Ratio numerator - H
400007	3	Timeout error check time (seconds) designation
400008		Used by system.
400009		Status (See *2.1.5 Error Status.*)

Operation

The settings for holding registers 400001 to 400007 are made as shown in the previous diagram. When transitional input 100001 goes from OFF to ON, axis 1 is made the master axis and a ratio operation with a ratio of 1/2 is performed for the A-axis of MC20 Module 1. When A-axis operation starts, output 1 is turned ON. When an input 3 operation interrupt is executed, output 1 is turned OFF, and the ratio operation is cancelled.

Note Any or all of the four independent axes (A, B, C, and D) can be operated simultaneously using the axis operation instructions. If an attempt is made to execute more than one instruction at the same time for the same axis, however, a busy error will occur.

SUMMARY The following information summarizes the instruction.


A-axis operation start: Input 1
Operation interrupt: Input 3
Control block
MC
MVA
Output 1: A-axis operating
Output 2: Error

MVA structure

Table 2.15 MVA Structural Elements

Element	Meaning	Possible Settings	
Top	4xxxxx MC20 Module number: 1 or 2	Holding registers	400001 to 409991 (W00001 to W09991)
	4xxxxx+1 Instruction type: 0 to 3 0: Target position designation, absolute value 1: Target position designation, incremental value 2: Point number designation, absolute value 3: Point number designation, incremental value 4: Voltage reference value designation (absolute value) 5: Ratio operation designation, master axis = axis 1 6: Ratio operation designation, master axis = axis 2 7: Ratio operation designation, master axis = axis 3 8: Ratio operation designation, master axis = axis 4	---	
	4xxxxx+2 Ratio denominator - L 4xxxxx+3 Ratio denominator - H Setting range: 1 to 9,999	---	
	4xxxxx+4 Ratio numerator - L 4xxxxx+5 Ratio numerator - H Setting range: ±1 to ±9,999	---	
	4xxxxx+6 Timeout error check time:0 to 20 s ("0" is treated as 1 s.)	---	
	4xxxxx+7 Used by system. 4xxxxx+8 Status	Do not write.	
Middle	None	---	
Bottom	Indicates the MVA instruction.	---	

Table 2.16 MVA Operation

Input 1	Input 2	Input 3	Status/Remarks	Output 1	Output 2	Output 3
OFF ↓ ON	Not used	OFF	• A-axis operation starts when input 1 is turned ON. The A-axis is operated for a travel distance calculated by multiplying the reference travel distance of the specified master axis by the ratio (numerator/denominator). • MC control relay STAL (A-axis operating) is turned ON. • When A-axis operation has been completed, MC control relay ENDAL (A-axis operation completion) turns ON.	ON ↓ OFF	OFF	Not used
ON or OFF	Not used	ON	• When input 3 is turned ON during A-axis operation, operation is interrupted and forcibly stopped. Ratio operation is cancelled. • When A-axis operation is interrupted, MC control relay STAL (A-axis operating) is turned OFF and ENDAL (A-axis operation completion) remains OFF. • A-axis operation can be restarted after the interruption by turning input 1 from OFF to ON.	OFF	OFF	Not used

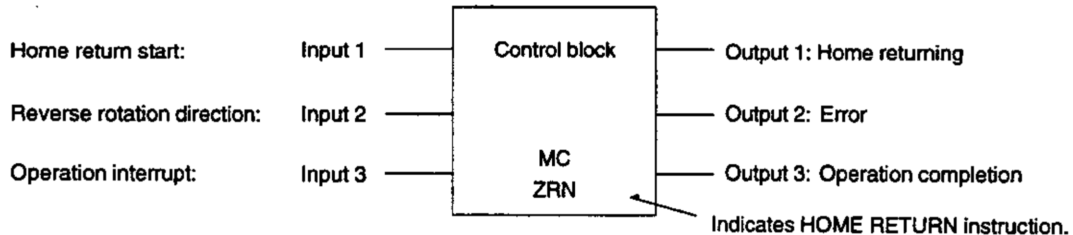
2.2.7 HOME RETURN (ZRN)

Function

ZRN returns the specified controlled axis to its home position in an incremental or absolute system.

Structure

1) The structure of the HOME RETURN (ZRN) instruction is shown below.



ZRN structure

2) The settings shown below are required for the control block in the top element. Use the holding registers for making these settings. The leading reference number is specified for

the top element. Refer to *Table 2.17 ZRN Structural Elements* for details on the ranges of reference numbers that can be specified.

Control Block

	Holding registers	Contents of registers
Leading reference number →	4xxxxx	MC20 Module number: 1 or 2
	4xxxxx+1	Control axis designation: 1, 2, 4, or 8
	4xxxxx+2	Timeout error check time: 0 to 20 s (see Note)
	4xxxxx+3	Used by system.
	4xxxxx+4	Status (See *2.1.5 Error Status.*)

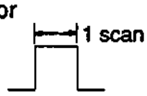
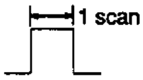
Note The timeout error check time default value is 0, and it is treated as 1 second.

3) To designate the controlled axis, specify in holding register 4xxxxx+1 the number corresponding to the axis which is to be returned to its home position. The numbers are shown in the following table. For example, specify "4" to designate axis 3. Multiple axes cannot be designated for a single instruction.

Axis Number	Corresponding Number
Axis 1	1
Axis 2	2
Axis 3	4
Axis 4	8

Operation

1) The following table shows the basic operation of the inputs and outputs.

I/O Number	Operation Name	Description	Input Contacts
Input 1	Home return start	Home return is started for the specified axis when the leading edge of the OFF→ON signal for this input is detected.	Positive transitional contact ─┤P├─
Input 2	Reverse rotation direction	If this input is OFF when the home position return is started, then the return to the home position will be executed in the forward rotation direction. If this input is ON, then the return will be in the reverse rotation direction.	N.O. contact ─┤ ├─
Input 3	Operation interrupt	When this input turns ON, the home return operation is interrupted and forcibly stopped. When home return is stopped by means of this input, output 1 is turned OFF and outputs 2 and 3 both remain OFF.	N.O. contact ─┤ ├─
Output 1	Home returning	Output 1 remains ON from the point at which input 1 turns ON until the return to the home position is completed (i.e., until MC control relay ZPTx is turned ON). It is turned OFF, however, when an operation interrupt is executed.	
Output 2	Error  ON pulse signal	When an error occurs, the ON pulse signal is output and the instruction is terminated. Output 3, however, is not turned ON. The pulse width is one scan. For details, see *2.1.5 Error Status.*	
Output 3	Operation completion  ON pulse signal	The ON pulse signal is output when the home return operation has been executed normally and completed (i.e., when MC control relay ZPTx is turned ON). The pulse width is one scan.	

Note (a) Use the contacts specified under "Input Contacts" to turn inputs 1 through 3 ON and OFF.

(b) The pulse width for ON pulse signal outputs is one scan time.

2) HOME RETURN can be executed when input 1 turns ON as long as all of the following conditions are satisfied:

- Input 3, output 1, output 2, and output 3 are all OFF.

- The designated axis is not travelling (i.e., MC control relay MOVx is OFF).

- The operation mode is set to manual mode (i.e., MC control relay MANL is ON).

3) When making the "timeout error check time" setting, allow enough time for the moving servo system to be stopped by the "operation interrupt," and for the remaining data to be cancelled. The timeout error check time can be set in holding register 4xxxxx+2 of the

control block. When "operation interrupt" is specified by turning input 3 ON, a timeout error will occur if the designated axis has not finished travelling by the time the timeout error check time elapses.

4) To restart after operation has been stopped by means of "operation interrupt," turn input 1 (home return start) from OFF to ON. Operation interrupt status cannot be cleared simply by turning input 3 (operation interrupt) OFF.

5) The controlled axis designation and other values designated by HOME RETURN will not be affected by executing "operation interrupt."

6) The MC20 Module's PA302 (home return direction) parameter is given priority in determining the "home return start" direction for HOME RETURN (ZRN). If the direction designations given by input 2 (reverse rotation direction) and parameter PA302 do not match, HOME RETURN (ZRN) will not be executed and a busy error will occur.

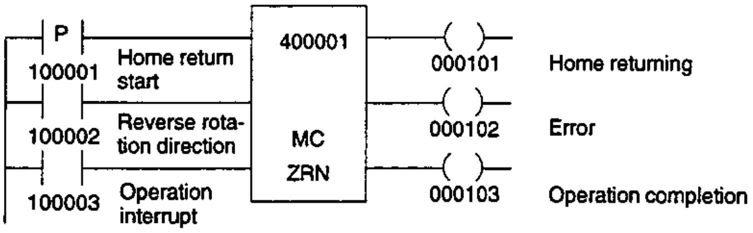
7) The home return operation carried out by HOME RETURN (ZRN) is the same as that of the ZRN command described in *1.2.5 HOME RETURN (ZRN)*. That is, with the exception of parameter PA302 (home return direction), the home return operation is carried out according to the MC20 Module home return parameters.

Position Detection System	Type of Operation	Home Position Return Operation Description
PA402 = 0 or 2 Incremental position detecting system	PA301 = 0 Home return operation 1	A three-step deceleration method based on a deceleration limit switch and a C-phase pulse.
	PA301 = 1 Home return operation 2	A two-step deceleration method based on a home position limit switch.
	PA301 = 2 Home return operation 3	A three-step deceleration method based on a deceleration limit switch and a home position limit switch.
	PA301 = 3 Home return operation 4	A two-step deceleration method based on a C-phase pulse.
PA402 = 3 Absolute position detecting system	---	Carries out positioning to a machine coordinate system home position.

Note HOME RETURN (ZRN) can be used simultaneously for as many as four different controlled axes. If an attempt is made to execute it at the same time for the same axis, however, a busy error will occur.

◀EXAMPLE▶ In this example, HOME RETURN is executed for axis 2 of MC20 Module 2. The operation is started in the reverse rotation direction. Assume that the PA302 (home return direction) parameter setting is 1 (i.e., negative direction).

Ladder Diagram



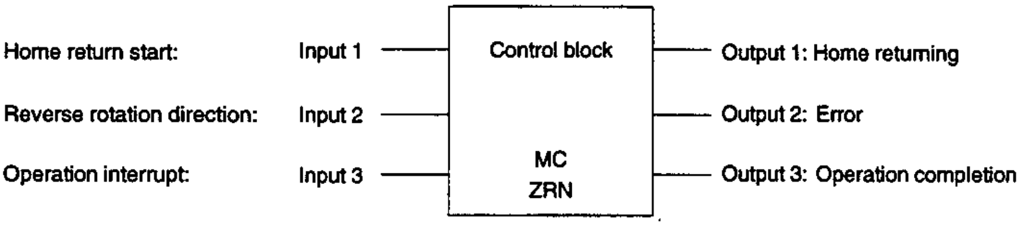
400001	2	MC20 Module no. designation
400002	2	Controlled axis designation
400003	3	Timeout error check time (seconds) designation
400004		Used by system.
400005		Status (See *2.1.5 Error Status.*)

Operation

The settings for holding registers 400001 to 400003 are made as shown above.

When input 2 (10002) is turned ON and the transitional input 100001 is turned from OFF to ON, the home return operation is started, in the reverse rotation direction, for axis 2 of MC20 Module 2. When the operation begins, output 1 is turned ON; when the operation has been completed, output 1 is turned OFF.

SUMMARY The following information summarizes the instruction.



ZRN structure

Table 2.17 ZRN Structural Elements

Element	Meaning		Possible Settings	
Top	4xxxxx	MC20 Module number: 1 or 2	Holding registers	400001 to 409995 (W00001 to W09995)
	4xxxxx+1	Controlled axis designation: 1, 2, 4, or 8	---	
	4xxxxx+2	Timeout error check time: 0 to 20 s ("0" is treated as 1 s.)	---	
	4xxxxx+3	Used by system.	Do not write.	
	4xxxxx+4	Status		
Middle	None		---	
Bottom	Indicates the ZRN instruction.		---	

Table 2.18 ZRN Operation

Input 1	Input 2	Input 3	Status/Remarks	Output 1	Output 2	Output 3
OFF ↓ ON	OFF	OFF	• At the leading edge of input 1, the home return operation is started in the forward rotation direction. • MC control relays ZRNxL (Axis x home returning) and MOVx (Axis x travelling) are turned ON. • The input 3 ON pulse signal is output when the home return operation has been completed. MC control relay ZPTx (Axis x home position) is turned ON.	ON ↓ OFF	OFF	OFF ↓ ON pulse
OFF ↓ ON	ON	OFF	• At the leading edge of input 1, the home return operation is started in the reverse rotation direction. • MC control relays ZRNxL (Axis x home returning) and MOVx (Axis x travelling) are turned ON. • The output 3 ON pulse signal is output when the home return operation has been completed. MC control relay ZPTx (Axis x home position) is turned ON.	ON ↓ OFF	OFF	OFF ↓ ON pulse
ON or OFF	OFF	ON	• When input 3 is turned ON during the home return operation, the operation is interrupted and forcibly stopped. • When the home return operation is interrupted, MC control relay ZRNxL (Axis x home returning) and MOVx (Axis x travelling) are turned OFF and ZPTx (Axis x home position) stays OFF. • The home return operation can be restarted after the interruption by turning input 1 from OFF to ON.	OFF	OFF	OFF

Note The MC20 Module's PA302 (home return direction) parameter is given priority in determining the "home return start" direction for HOME RETURN (ZRN). If the direction designations given by input 2 (reverse rotation direction) and parameter PA302 do not match, HOME RETURN (ZRN) will not be executed and a busy error will occur.

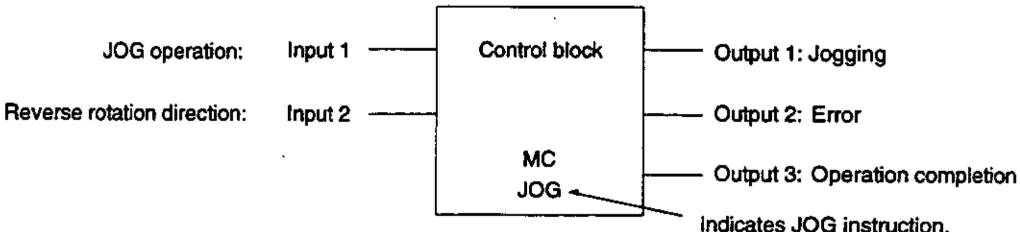
2.2.8 JOG (JOG)

Function

JOG moves the specified MC20 Module controlled axis in the specified direction at rapid traverse speed while the switch is ON.

Structure

1) The structure of the JOG instruction is shown below.


JOG operation:
Input 1
Reverse rotation direction:
Input 2
Control block
MC
JOG
Output 1: Jogging
Output 2: Error
Output 3: Operation completion
Indicates JOG instruction.

JOG structure

2) The settings shown below are required for the controlled block in the top element. Use the holding registers for making these settings. The leading reference number is specified for the top element. See *Table 2.19 JOG Structural Elements* for details on the ranges of reference numbers that can be specified.

Control Block

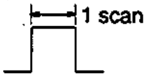
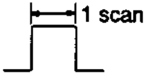
	Holding registers	Contents of registers
Leading reference number →	4xxxxx	MC20 Module number: 1 or 2
	4xxxxx+1	Controlled axis designation: 1, 2, 4, or 8
	4xxxxx+2	Used by system.
	4xxxxx+3	Status (See *2.1.5 Error Status.*)

3) To designate the controlled axis, specify in holding register 4xxxxx+1 the number corresponding to the axis for which the JOG operation is to be executed. The numbers are shown in the following table. For example, specify "8" to designate axis 4. Multiple axes cannot be designated for a single instruction.

Axis Number	Corresponding Number
Axis 1	1
Axis 2	2
Axis 3	4
Axis 4	8

Operation

1) The following table shows the basic operation of the inputs and outputs.

I/O Number	Operation Name	Description	Input Contacts
Input 1	JOG operation	While this input is ON, the specified controlled axis moves at rapid traverse speed.	N.O. contact ─┤ ├─
Input 2	Reverse rotation direction	If this input is OFF when the home position return is started, then the return to the home position will be executed in the forward rotation direction. If this input is ON, then the return will be in the reverse rotation direction.	N.O. contact ─┤ ├─
Output 1	Jogging	Output 1 remains ON from the point at which input 1 turns ON until the next time input 1 turns OFF.	
Output 2	Error  ON pulse signal	When an error occurs, the ON pulse signal is output and the instruction is terminated. Output 3, however, is not turned ON. The pulse width is one scan. For details, see *2.1.5 Error Status.*	
Output 3	Operation completion  ON pulse signal	The ON pulse signal is output when input 1 turns OFF and the JOG operation is completed (i.e., when MC control relay JOGx is turned OFF). The pulse width is one scan.	

Note (a) Use the contacts specified under "Input Contacts" to turn inputs 1 through 3 ON and OFF.

(b) The pulse width for ON pulse signal outputs is one scan time.

2) The JOG operation can be executed when input 1 turns ON as long as all of the following conditions are satisfied:

- Outputs 1, 2, and 3 are all OFF.

- The designated axis is not travelling (i.e., MC control relay MOVx is OFF).

- The operation mode is set to manual mode or online edit mode (i.e., MC control relay MANL or OEDL is ON).

3) An override can be set for the jog speed in 16 steps between 0% and 100% or in increments of 0.1% between 0% and 3276.7%. The overrides can also be enabled and disabled.

a) Overrides for the jog speed are enabled and disable in parameter P0016.

(1) Override: Enabled for MC control coils (16 steps)

(2) Override: Enabled for MC link registers (in increments of 0.1%)

(3) Override: Disabled

The overrides are disabled in the default settings.

b) When an override is enabled, the percentage can be switched as shown below.

(1) **MC Control Coils (16 Steps)**
0, 1, 2, 4, 6, 8, 10, 20, 30, 40, 50, 60, 80, 90, or 100 (%)

Ver.
B08

(2) **MC Link Registers (0.1% Increments)**
0.0 to 3276.7 (%)

When an override is disabled, the speed will be 100% of the set value or reference value.

c) Overrides for the jog speed are set using the following MC control coils and MC link registers.

Ver.
B08

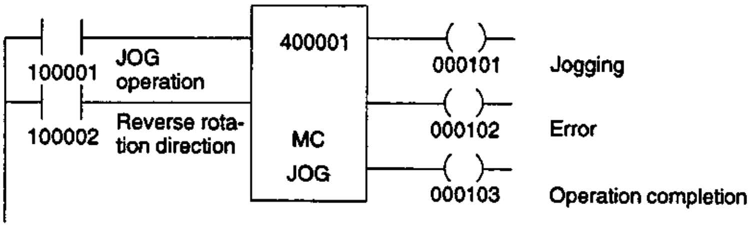
Coil/Register	Use		References	
MC control coils 16 steps from 0% to 100%	Independent axes or manual	Axis 1	QN0145 to QN0148	N is the MC20 Module number (N = 1 or 2)
		Axis 2	QN0149 to QN0152	
		Axis 3	QN0153 to QN0156	
		Axis 4	QN0157 to QN0160	
MC link registers 0.0% to 3276.7% in 0.1% increments	Independent axes or manual	Axis 1	409911 and 409984	MC Modules 1 and 2 (default allocations)
		Axis 2	409912 and 409985	
		Axis 3	409913 and 409986	
		Axis 4	409914 and 409987	

Set the MC link register to 10 times the override percentage when setting an override in 0.1% increments.

Refer to *3.2 MC Control Coil Functions* for details on MC control coils and to *1.5.6 Link Input Variables (#□□□□)* for details on MC link registers.

◀EXAMPLE▶ In this example, JOG is executed for axis 4 of MC20 Module 2. The operation is started in the reverse rotation direction.

Ladder Diagram



400001	1	MC20 Module no. designation
400002	8	Controlled axis designation
400003		Used by system.
400004		Status (See *2.1.5 Error Status*.)

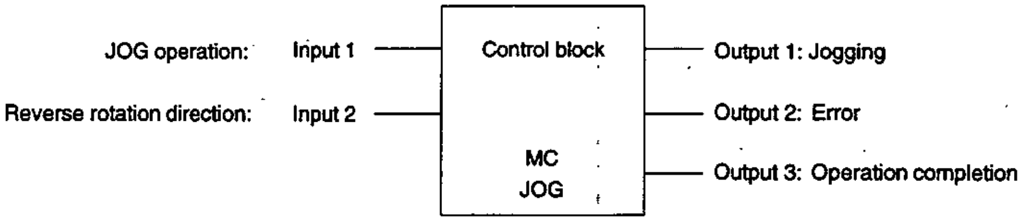
Operation

The settings for holding registers 400001 and 400002 are made as shown above.

After input 2 (10002) is turned ON, axis 4 of MC20 Module 1 moves in the reverse rotation direction while the transitional input 100001 is ON. While the JOG operation continues, output 1 is turned ON; when input 1 turns OFF, the operation is completed and output 1 is turned OFF.

Note The JOG instruction can be used simultaneously for as many as four different control axes. If an attempt is made to execute it at the same time for the same axis, however, a busy error will occur.

SUMMARY The following information summarizes the instruction.



JOG structure

Table 2.19 JOG Structural Elements

Element	Meaning		Possible Settings	
Top	4xxxxx	MC20 Module number: 1 or 2	Holding registers	400001 to 409996 (W00001 to W09996)
	4xxxxx+1	Controlled axis designation: 1, 2, 4, or 8	---	
	4xxxxx+4	Used by system.	Do not write.	
	4xxxxx+5	Status		
Middle	None		---	
Bottom	Indicates the JOG instruction.		---	

Table 2.20 JOG Operation

Input 1	Input 2	Input 3	Status/Remarks	Output 1	Output 2	Output 3
ON	OFF	None	• The specified axis is moved in the forward rotation direction (= forward JOG). • MC control relays JOGxL (Axis x jogging) and MOVx (Axis x travelling) are tumed ON.	ON	OFF	OFF
OFF	OFF	None	• The above forward JOG operation is completed, and the output 3 (operation completion) ON pulse signal is output.	OFF	OFF	ON pulse
ON	ON	None	• The specified axis is moved in the reverse rotation direction (= reverse JOG operation). • MC control relays JOGxL (Axis x jogging) and MOVx (Axis x travelling) are tumed ON.	ON	OFF	OFF
OFF	ON	None	• The above reverse JOG operation is completed, and the output 3 (operation completion) ON pulse signal is output.	OFF	OFF	ON pulse

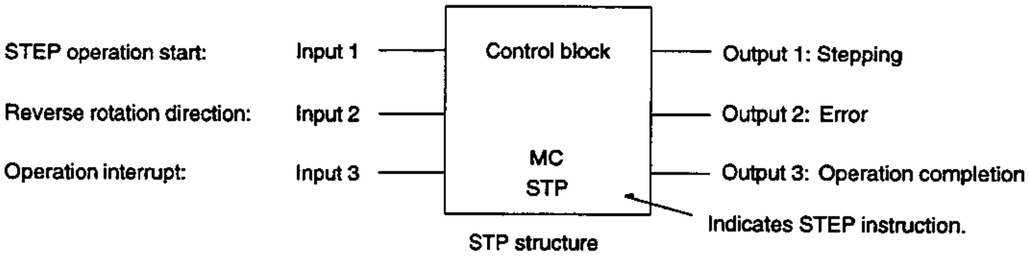
2.2.9 STEP (STP)

Function

STP moves the specified MC20 Module controlled axis in the specified direction at rapid traverse speed for the specified travel distance only.

Structure

1) The structure of the STP instruction is shown below.



STP structure

2) The settings shown below are required for the control block in the top element. The leading reference number is specified for the top element. See *Table 2.21 STP Structural Elements* for details on the ranges of reference numbers that can be specified.

Control Block

Holding registers	Contents of registers
4xxxxx (Leading reference number)	MC20 Module number: 1 or 2
4xxxxx+1	Controlled axis designation: 1, 2, 4, or 8
4xxxxx+2	STEP travelling distance - L (1 to 99,999,999)
4xxxxx+3	STEP travelling distance - H (1 to 99,999,999)
4xxxxx+4	Timeout error check time: 0 to 20 s (see Note)
4xxxxx+5	Used by system.
4xxxxx+6	Status (See *2.1.5 Error Status.*)

Note The timeout error check time default value is 0, and it is treated as 1 second.

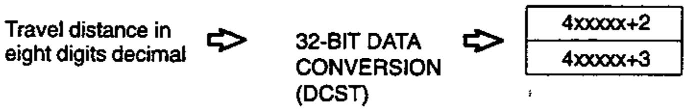
3) To designate the controlled axis, specify in holding register 4xxxxx+1 the number corresponding to the axis for which the STEP operation is to be executed. The numbers are shown in the following table. For example, specify "4" to designate axis 3. Multiple axes cannot be designated for a single instruction.

Axis Number	Corresponding Number
Axis 1	1
Axis 2	2
Axis 3	4
Axis 4	8

4) The STEP travel distance H/L (incremental value) is directly designated in **32-bit binary**, using two registers.

STEP travel distance - H	STEP travel distance - L	
4xxxxx+3	4xxxxx+2	= signed 32-bit binary

32-BIT DATA CONVERSION (DCST) instruction is convenient when carrying out direct designation of binary numbers.


Travel distance in eight digits decimal
32-BIT DATA CONVERSION (DCST)
4xxxxx+2
4xxxxx+3

32-bit Binary

The usage of "binary" here is defined as follows:
Positive value: Normal 32-bit binary.
Negative value: Not used, because the designation is incremental.

5) The unit for inputting the value designated as the STEP travel distance is determined by the MC20 Module's common parameter P0005 (decimal position).

MC20 Module Parameter Setting	Reference Unit	
	Linear Axis	Rotary Axis
P0005 = 1	0.1 mm	0.1°
P0005 = 2	0.01 mm	0.01°
P0005 = 3	0.001 mm	0.001°

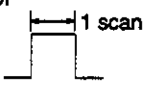
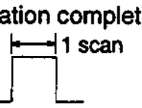
◀EXAMPLE▶ Linear Axis:

In this example, the STEP travel distance designation is actually specified in binary but appears in decimal.

STEP travel distance designation:	50000
Actual travel distance (when P0005 = 3):	50.000 mm

Operation

1) The following table shows the basic operation of the inputs and outputs.

I/O Number	Operation Name	Description	Input Contacts
Input 1	STEP operation start	STEP operation is started for the specified axis when the leading edge of the OFF→ON signal for this input is detected. The specified axis travels only for the specified distance.	Positive transitional contact —| P |—
Input 2	Reverse rotation direction	If this input is OFF when input 1 turns ON, then the STEP operation will be executed in the forward rotation direction. If this input is ON, then the operation will be executed in the reverse rotation direction.	N.O. contact —| |—
Input 3	Operation interrupt	When this input turns ON, the STEP operation is interrupted and forcibly stopped, and the remaining data is cancelled. When STEP is stopped by means of this input, output 1 is turned OFF and outputs 2 and 3 both remain OFF.	N.O. contact —| |—
Output 1	Stepping	Output 1 remains ON from the point at which input 1 turns ON until the STEP operation is completed (i.e., until MC control relay MOVx is turned OFF). It is turned OFF, however, when an operation interrupt is executed.	
Output 2	Error  ON pulse signal	When an error occurs, the ON pulse signal is output and the instruction is terminated. Output 3, however, is not turned ON. The pulse width is one scan. For details, see *2.1.5 Error Status.*	
Output 3	Operation completion  ON pulse signal	The ON pulse signal is output when the STEP operation has been completed (i.e., when MC control relay STPxL is turned OFF). The pulse width is one scan.	

Note (a) Use the contacts specified under "Input Contacts" to turn inputs 1 through 3 ON and OFF.

(b) The pulse width for ON pulse signal outputs is one scan time.

2) STEP can be executed when input 1 turns ON as long as all of the following conditions are satisfied:

- Input 3, output 1, output 2, and output 3 are all OFF.
- The designated axis is not travelling (i.e., MC control relay MOVx is OFF).
- The operation mode is set to manual mode (i.e., MC control relay MANL is ON).

3) When making the "timeout error check time" setting, allow enough time for the moving servo system to be stopped by the "operation interrupt," and for the remaining data to be cancelled. The timeout error check time can be set in holding register 4xxxxx+4 of the control block. When "operation interrupt" is specified by turning input 3 ON, a timeout error will occur if the designated axis has not finished travelling by the time the timeout error check time elapses.

4) To restart after operation has been stopped by means of "operation interrupt," turn input 1 (STEP operation start) from OFF to ON. Operation interrupt status cannot be cleared simply by turning input 3 (operation interrupt) OFF.

5) The STEP travel distance designation and other values designated by STEP will not be affected by executing "operation interrupt."

6) An override can be set for the step operation speeds in 16 steps between 0% and 100% or in increments of 0.1% between 0% and 3276.7%. The overrides can also be enabled and disabled.

a) Overrides for the step operation speeds are enabled and disable in parameter P0016.

(1) Override: Enabled for MC control coils (16 steps)

Ver. B08

(2) Override: Enabled for MC link registers (in increments of 0.1%)

(3) Override: Disabled

The overrides are disabled in the default settings.

b) When an override is enabled, the percentage can be switched as shown below.

(1) **MC Control Coils (16 Steps)**
0, 1, 2, 4, 6, 8, 10, 20, 30, 40, 50, 60, 80, 90, or 100 (%)

Ver. B08

(2) **MC Link Registers (0.1% Increments)**
0.0 to 3276.7 (%)

When an override is disabled, the speed will be 100% of the set value or reference value.

c) Overrides for the step operation speeds are set using the following MC control coils and MC link registers.

Ver. B08

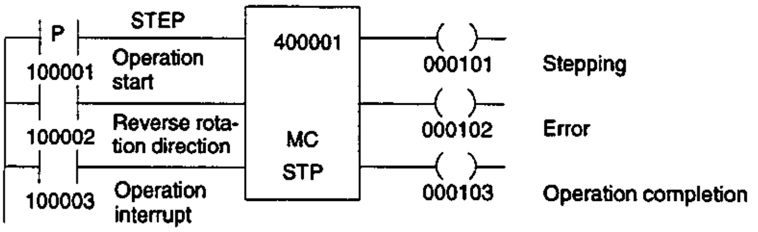
Coil/Register	Use	References		
MC control coils 16 steps from 0% to 100%	Independent axes or manual	Axis 1	QN0145 to QN0148	N is the MC20 Module number (N = 1 or 2)
		Axis 2	QN0149 to QN0152	
		Axis 3	QN0153 to QN0156	
		Axis 4	QN0157 to QN0160	
MC link registers 0.0% to 3276.7% in 0.1% increments	Independent axes or manual	Axis 1	409911 and 409984	MC Modules 1 and 2 (default allocations)
		Axis 2	409912 and 409985	
		Axis 3	409913 and 409986	
		Axis 4	409914 and 409987	

Set the MC link register to 10 times the override percentage when setting an override in 0.1% increments.

Refer to *3.2 MC Control Coil Functions* for details on MC control coils and to *1.5.6 Link Input Variables (#□□□□)* for details on MC link registers.

◀EXAMPLE▶ In this example, the STEP operation is executed for axis 3 of MC20 Module 2. The axis moves in the reverse rotation direction, for a travelling distance of 40 mm (incremental value) only.

Ladder Diagram



400001	2	MC20 Module number: 1 or 2
400002	4	Controlled axis designation: 1, 2, 4, or 8
400003	Binary	STEP travelling distance - L
400004	40000	STEP travelling distance - H
400005	3	Timeout error check time: 0 to 20 s (see Note)
400006		Used by system.
400007		Status (See *2.1.5 Error Status.*)

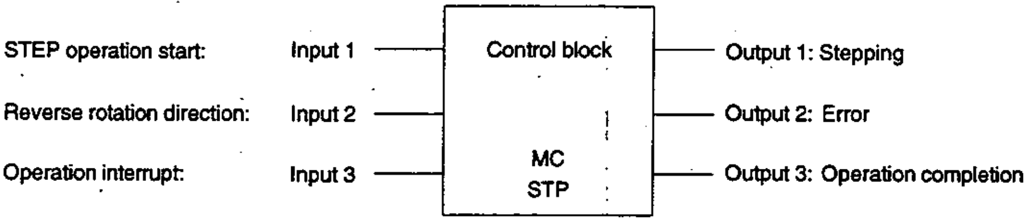
Operation

The settings for holding registers 400001 to 400005 are made as shown above.

After input 2 (10002) is turned ON, axis 3 of MC20 Module 2 is moved in the reverse rotation direction for 40.000 mm whenever the transitional input 100001 is turned from OFF to ON (as long as parameter P0005 = 3). When the operation begins, output 1 is turned ON; when the operation has been completed, output 1 is turned OFF.

Note The STEP instruction can be used simultaneously for as many as four different controlled axes. If an attempt is made to execute it at the same time for the same axis, however, a busy error will occur.

SUMMARY The following information summarizes the instruction.



STP structure

Table 2.21 STP Structural Elements

Element	Meaning		Possible Settings	
Top	4xxxxx	MC20 Module number: 1 or 2	Holding registers	400001 to 409993 (W00001 to W09993)
	4xxxxx+1	Controlled axis designation: 1, 2, 4, or 8	---	
	4xxxxx+2	STEP travelling distance - L	---	
	4xxxxx+3	STEP travelling distance - H 1 to 99,999,999		
	4xxxxx+4	Timeout error check time: 0 to 20 s ("0" is treated as 1 s.)	---	
	4xxxxx+5	Used by system.	Do not write.	
	4xxxxx+6	Status		
Middle	None		---	
Bottom	Indicates the STP instruction.		---	

Table 2.22 STP Operation

Input 1	Input 2	Input 3	Status/Remarks	Output 1	Output 2	Output 3
OFF ↓ ON	OFF	OFF	• At the leading edge of input 1, the STEP operation is executed in the forward rotation direction for the specified travelling distance (= forward STEP operation).	ON	OFF	OFF
			• MC control relays STPxL (Axis x stepping) and MOVx (Axis x travelling) are turned ON.	↓		↓
			• The output 3 ON pulse signal is output when the forward STEP operation has been completed.	OFF		ON pulse
OFF ↓ ON	ON	OFF	• At the leading edge of input 1, the STEP operation is executed in the reverse rotation direction (= reverse STEP operation).	ON	OFF	OFF
			• MC control relays STPxL (Axis x stepping) and MOVx (Axis x travelling) are turned ON.	↓		↓
			• The output 3 ON pulse signal is output when the reverse STEP operation has been completed.	OFF		ON pulse
ON or OFF	OFF	ON	• When input 3 is turned ON during the STEP operation, the operation is interrupted and forcibly stopped.	OFF	OFF	OFF
			• When the STEP operation is interrupted, MC control relay STPxL (Axis x stepping) and MOVx (Axis x travelling) are turned OFF.			
			• The STEP operation can be restarted after the interruption by turning input 1 from OFF to ON.			

2.3 Control Instructions

This section explains the instructions that control the MC20 Module by carrying out operations such as designating modes and resetting status. These are basic instructions, so be sure to familiarize yourself with them before proceeding further.

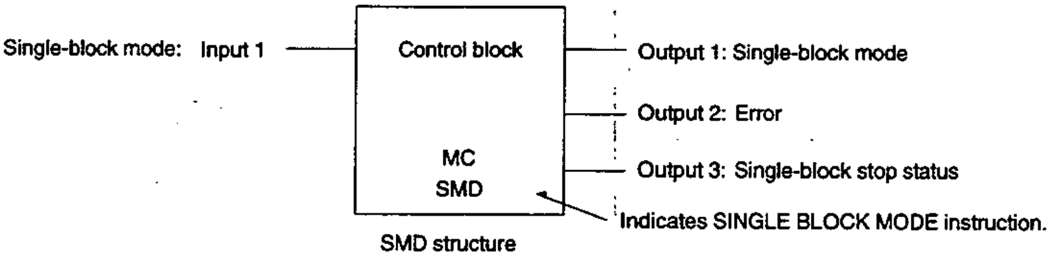
2.3.1 SINGLE BLOCK MODE (SMD)

Function

When SMD is executed, it switches to the single-block stop status when one block has finished.

Structure

1) The structure of the SMD instruction is shown below.


Single-block mode: Input 1
Control block
MC
SMD
Output 1: Single-block mode
Output 2: Error
Output 3: Single-block stop status
Indicates SINGLE BLOCK MODE instruction.

SMD structure

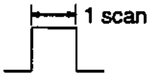
2) The settings shown below are required for the control block in the top element. The leading reference number is specified for the top element. Refer to *Table 2.23 SMD Structural Elements* for details on the ranges of reference numbers that can be specified.

Control Block

Holding registers	Contents of registers
4xxxxx (Leading reference number)	MC20 Module number: 1 or 2
4xxxxx+1	Used by system.
4xxxxx+2	Status (Refer to *2.1.5 Error Status.*)

Operation

1) The following table shows the basic operation of the inputs and outputs.

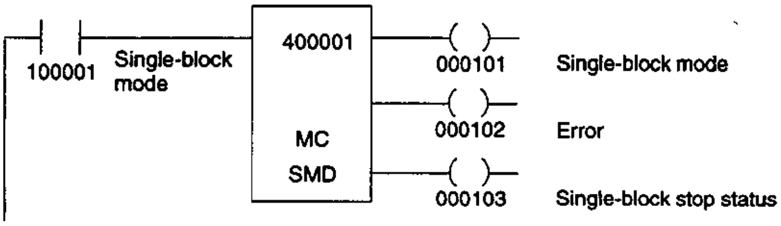
I/O Number	Operation Name	Description	Input Contacts
Input 1	Single-block mode	When this input is ON, the MC20 Module is in single block operation mode. While in this mode, program operation is paused after each block is executed.	N.O. contact ─┤ ├─
Output 1	Single-block mode	This output turns ON when MC control relay SBKL (single-block mode) is turned ON.	
Output 2	Error  ON pulse signal	When an error occurs, the ON pulse signal is output and the instruction is terminated. Output 3, however, is not turned ON. The pulse width is one scan. For details, refer to *2.1.5 Error Status.*	
Output 3	Single-block stop status	Program execution is paused and this output turns ON when program execution of each block is completed while in the single-block operation mode. To clear the single-block stop status and restart program execution, turn the PROGRAM RUN (MVL) input 1 from OFF to ON.	

Note Use the contact specified under "Input Contacts" to turn input 1 ON and OFF.

2) If input 1 is turned ON and one second elapses before MC control relay SBKL (single-block mode) is turned ON, the ON pulse signal will be given for output 2 and a timeout error will occur.

◀EXAMPLE▶ In this example, servodriver power is turned on for control axes 1, 2, and 3 for MC20 Module 2.

Ladder Diagram



400001	1	MC20 Module no. designation
400002		Used by system.
400003		Status (Refer to *2.1.5 Error Status.*)

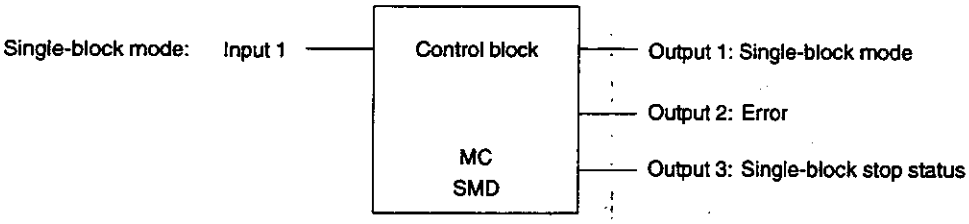
Operation

The MC20 Module number "1" is specified in holding register 400001.

When N.O. input 100001 is turned ON, MC20 Module 1 is placed in single-block operation mode.

Note Only one SINGLE BLOCK MODE (SMD) instruction can be executed with respect to a single MC20 Module.

➧ SUMMARY The following information summarizes the instruction.


Single-block mode:
Input 1
Control block
MC
SMD
Output 1: Single-block mode
Output 2: Error
Output 3: Single-block stop status

SMD structure

Table 2.23 SMD Structural Elements

Element	Meaning		Possible Settings	
Top	4xxxxx	MC20 Module number: 1 or 2	Holding registers	400001 to 409997 (W00001 to W09997)
	4xxxxx+1	Used by system.	Do not write.	
	4xxxxx+2	Status		
Middle	None		——	
Bottom	Indicates the SMD instruction.		——	

Table 2.24 SMD Operation

Input 1	Status/Remarks	Output 1	Output 2	Output 3
ON	● When input 1 is ON, the specified MC20 Module goes into the single-block operation mode, and MC control relay SBKL (single-block mode) turns ON. ● When execution of a block is completed during program execution, single-block stop status goes into effect. MC control relay SBKS (single-block stop status) turns ON. ● To clear the single-block stop status and restart program execution, turn the PROGRAM RUN (MVL) input 1 from OFF to ON.	ON	OFF	OFF ↓ ON
OFF	● When input 1 is OFF, the MC20 Module goes into the normal continuous operation mode.	OFF	OFF	OFF

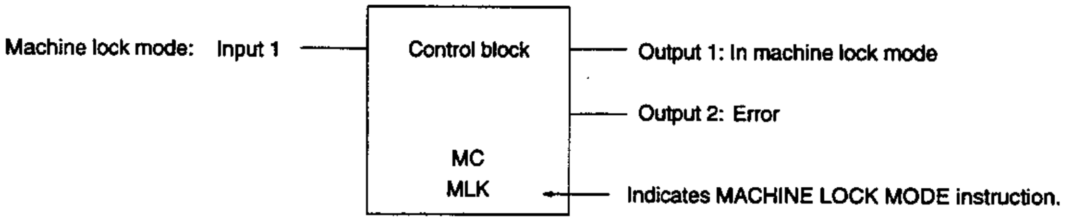
2.3.2 MACHINE LOCK MODE (MLK)

Function

When PROGRAM RUN (MVL) is executed, the current position display is changed according to the program. MACHINE LOCK MODE (MLK), however, changes the mode so that the actual controlled axis is locked and does not move.

Structure

1) The structure of the MLK instruction is shown below.


Machine lock mode: Input 1
Control block
Output 1: In machine lock mode
Output 2: Error
MC
MLK
Indicates MACHINE LOCK MODE instruction.

MLK structure

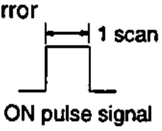
2) The settings shown below are required for the control block in the top element. The leading reference number is specified for the top element. Refer to *Table 2.25 MLK Structural Elements* for details on the ranges of reference numbers that can be specified.

Control Block

Holding registers	Contents of registers
4xxxxx (Leading reference number)	MC20 Module number: 1 or 2
4xxxxx+1	Used by system.
4xxxxx+2	Status (Refer to *2.1.5 Error Status.*)

Operation

1) The following table shows the basic operation of the inputs and outputs.

I/O Number	Operation Name	Description	Input Contacts
Input 1	Machine lock mode	When this input is ON, the MC20 Module is in machine lock operation mode. While in this mode, the current position display changes according to the program but the controlled axis is locked and does not move.	N.O. contact —| |—
Output 1	In machine lock mode	This output turns ON when input 1 is turned ON and MC control relay MLK (machine lock mode) is turned ON.	
Output 2	Error 	When an error occurs, the ON pulse signal is output and the instruction is terminated. The pulse width is one scan. For details, refer to *2.1.5 Error Status.*	

Note (a) Use the contact specified under "Input Contacts" to turn input 1 ON and OFF.

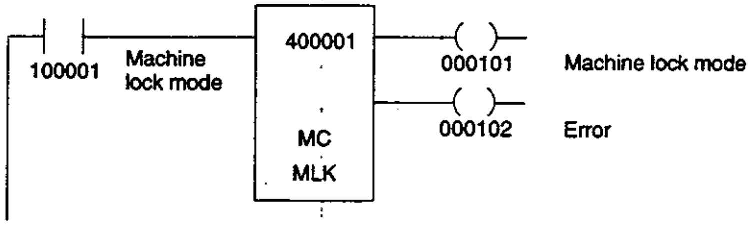
(b) The pulse width of the ON pulse signal is one scan.

2) If input 1 is turned ON and one second elapses before MC control relay MLK (machine lock mode) is turned ON, the ON pulse signal will be given for output 2 and a timeout error will occur.

3) When a travelling axis is stopped by means of switching to the machine lock mode, MC control relay MLKxS (axis x machine lock stop) is turned ON. When the machine lock mode is cleared and program execution begins, this MC control relay(s) is turned OFF.

◀**EXAMPLE**▶ In this example, machine lock mode is specified for MC20 Module 2.

Ladder Diagram



400001	2	MC20 Module no. designation
400002		Used by system.
400003		Status (Refer to *2.1.5 Error Status.*)

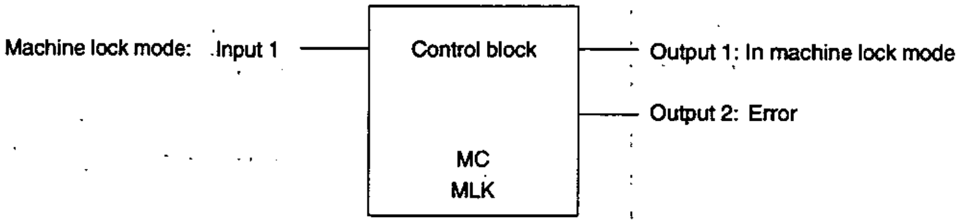
Operation

The MC20 Module number "2" is specified in holding register 400001.

When N.O. input 100001 is turned ON, MC20 Module 2 is placed in machine lock mode. Even when the program is running, the controlled axis is locked and does not move.

Note Only one MACHINE LOCK MODE (MLK) instruction can be executed with respect to a single MC20 Module.

➧ **SUMMARY** The following information summarizes the instruction.



MLK structure

Table 2.25 MLK Structural Elements

Element	Meaning		Possible Settings	
Top	4xxxxx	MC20 Module number: 1 or 2	Holding registers	400001 to 409997 (W00001 to W09997)
	4xxxxx+1	Used by system.	Do not write.	
	4xxxxx+2	Status		
Middle	None		——	
Bottom	Indicates the MLK instruction.		——	

Table 2.26 MLK Operation

Input 1	Status/Remarks	Output 1	Output 2	Output 3
ON	● When input 1 is ON, the specified MC20 Module goes into the machine lock operation mode, and MC control relay MLK (machine lock mode) turns ON. ● When a travelling axis is stopped by means of switching to the machine lock mode, MC control relay MLKxS (axis x machine lock stop) is turned ON. ● To clear the "machine lock stop" first turn OFF input 1 and then turn the PROGRAM RUN (MVL) input 1 from OFF to ON.	ON	OFF	None
OFF	● When input 1 is OFF, the machine lock mode is cleared.	OFF	OFF	None

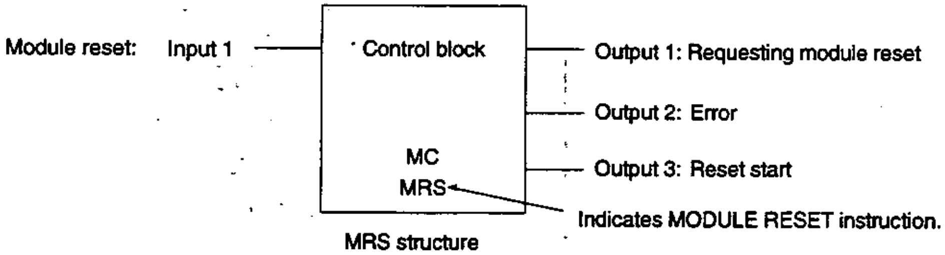
2.3.3 MODULE RESET (MRS)

Function

MRS initializes the internal status of the MC20 Module. That is, it resets all alarms of the MC20 Module and **validates the parameters renewed with the Programming Device.**

Structure

1) The structure of the MRS instruction is shown below.



MRS structure

2) The settings shown below are required for the control block in the top element. The leading reference number is specified for the top element. Refer to *Table 2.27 MRS Structural Elements* for details on the ranges of reference numbers that can be specified.

Control Block

Holding registers

Leading reference number →	Contents of registers
4xxxxx	MC20 Module number: 1 or 2
4xxxxx+1	Used by system.
4xxxxx+2	Status (Refer to *2.1.5 Error Status.*)

3) The actual module reset operation is executed as the MC20 Module receives the MODULE RESET (MRS) output 3 (reset start) signal. For that reason, execute the next instruction with respect to the MC20 Module after at least five seconds have elapsed since the output 3 ON pulse signal was output.

TERMS

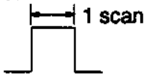
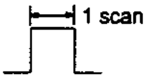
Validates parameters renewed with the Programming Device

For safety, MC20 Module parameters are not put into effect immediately after being changed by the Programming Device. To change the parameters to the new set values and put them into effect, execute either the MODULE RESET (MRS) or MACHINE RESET (RST) instruction. There are some parameters, however, that are not changed by the MACHINE RESET (RST) instruction. The only ones that are changed are indicated by “Reset” in the “When in Effect” column in *Appendix B Parameters.* All parameters can be changed to the last set values by turning the MC20 Module OFF and ON or by executing the MODULE RESET (MRS) instruction.

Note If the next instruction is executed less than five seconds after the reset is started by the output 3 signal, an "MC not ready" error will occur and that instruction will be terminated. Be sure to execute the next instruction until at least five seconds have elapsed.

Operation

1) The following table shows the basic operation of the inputs and outputs.

I/O Number	Operation Name	Description	Input Contacts
Input 1	Module reset	Module reset is requested when the leading edge of the OFF→ON signal for this input is detected. That is, initialization of the MC20 Module's internal status is specified.	Positive transitional contact —| P |—
Output 1	Requesting module reset	This output is turned ON by the leading edge of the input 1 signal, and it is turned OFF upon completion of the module reset request to the MC20 Module.	
Output 2	Error  ON pulse signal	When an error occurs, the ON pulse signal is output and the instruction is terminated. Output 3, however, is not turned ON. The pulse width is one scan. For details, refer to *2.1.5 Error Status.*	
Output 3	Reset start  ON pulse signal	The output 3 ON pulse signal is output when the module reset request to the MC20 Module has been completed properly. The pulse width is one scan.	

Note (a) Use the contact specified under "Input Contacts" to turn input 1 ON and OFF.

(b) The pulse width of the ON pulse signal is one scan.

2) The following items are initialized at the specified MC20 Module when module reset is executed by turning input 1 ON.

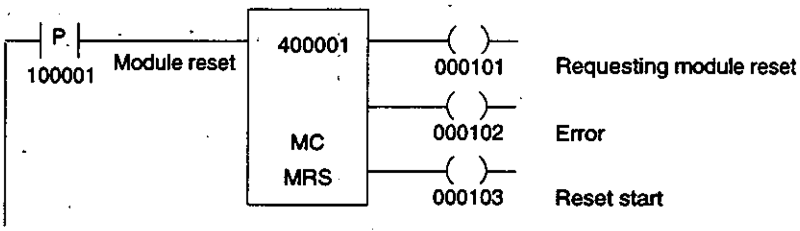
- Travelling controlled axes are decelerated and stopped.
- Any remaining travelling data is deleted.
- Motion program execution pointers are deleted.
- In an incremental position detecting system, the machine coordinate system's zero point is made equivalent to the current position.
- All alarms are reset and all alarm histories are cleared.
- Parameters are changed to the values renewed with the Programming Device.
- The following data is cleared: (1) common variables, (2) MC coils, and (3) matrix setting instruction values.
- After the above items are executed, the MC20 Module is restarted from the beginning.

The following data is retained.

- System variables
- H variables
- Point table values
- Home positions of absolute detecting systems

◀EXAMPLE▶ In this example, module reset is executed for MC20 Module 1.

Ladder Diagram



400001	1	MC20 Module no. designation
400002		Used by system.
400003		Status (Refer to *2.1.5 Error Status.*)

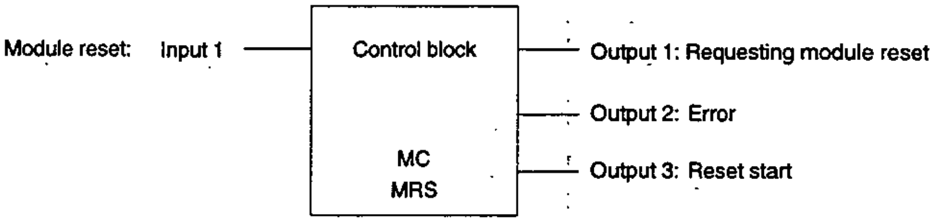
Operation

The MC20 Module number "1" is specified in holding register 400001.

When input 1 (100001) is turned ON, module reset is executed and MC20 Module 1 is returned to the status that was in effect immediately after it was powered up.

Note Only one MODULE RESET (MRS) instruction can be executed with respect to a single MC20 Module.

SUMMARY The following information summarizes the instruction.



MRS structure

Table 2.27 MRS Structural Elements

Element	Meaning		Possible Settings	
Top	4xxxxx	MC20 Module number: 1 or 2	Holding registers	400001 to 409997 (W00001 to W09997)
	4xxxxx+1	Used by system.	Do not write.	
	4xxxxx+2	Status		
Middle	None		——	
Bottom	Indicates the MRS instruction.		——	

Table 2.28 MRS Operation

Input 1	Status/Remarks	Output 1	Output 2	Output 3
OFF ↓ ON	• Module reset is requested at the leading edge of input 1. That is, initialization of the MC20 Module's internal status is specified.	ON ↓	OFF	OFF ↓
	• The output 3 ON pulse signal is output when the above module reset request has been completed.	OFF		ON pulse
	• When the MC20 Module receives this ON pulse output, it executes the module reset within five seconds.			

Note Approximately 5 s is required to complete resetting the MC20 Module. Wait at least 5 s after executing MODULE RESET (MRS) before executing the next instruction in the program.

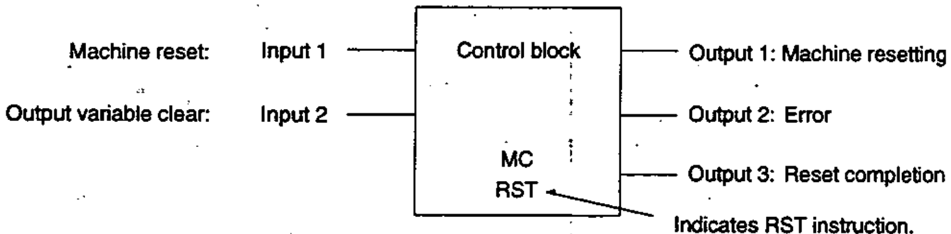
2.3.4 MACHINE RESET (RST)

Function

RST is used to partly initialize the status of the MC20 Module. RST resets some alarms of the MC20 Module and **changes parameters to the values renewed with the PARAMETER SETTING (PRM) instruction.**

Structure

1) The structure of the RST instruction is shown below.



RST structure

2) The settings shown below are required for the control block in the top element. The leading reference number is specified for the top element. Refer to *Table 2.29 RST Structural Elements* for details on the ranges of reference numbers that can be specified.

Control Block

	Holding registers	Contents of registers
Leading reference number →	4xxxxx	MC20 Module number: 1 or 2
	4xxxxx+1	Used by system.
	4xxxxx+2	Status (Refer to *2.1.5 Error Status.*)

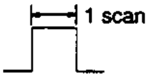
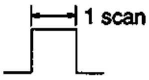
Changes parameters to values renewed with PARAMETER SETTING (PRM) instruction

For safety, MC20 Module parameters are not put into effect immediately after being changed with the PARAMETER SETTING (PRM) instruction. To change the parameters to the new set values and put them into effect, execute the MACHINE RESET (RST) instruction. If MODULE RESET (MRS) instruction is executed or the MC20 Module is turned OFF and ON after the PARAMETER SETTING (PRM) instruction is executed, the new set values will be canceled and the previous parameters set with the Programming Device will be valid.

To change the parameters to the values set with the Programming Device, execute MODULE RESET (MRS) or MACHINE RESET (RST) instruction. There are some parameters, however, that are not changed by the MACHINE RESET (RST) instruction. The only ones that are changed are indicated by “Reset” in the “When in Effect” column in *Appendix B Parameters.* All parameters can be changed to the new set values by turning the MC20 Module OFF and ON or by executing the MODULE RESET (MRS) instruction.

Operation

1) The following table shows the basic operation of the inputs and outputs.

I/O Number	Operation Name	Description	Input Contacts
Input 1	Machine reset	Module reset is executed when the leading edge of the OFF→ON signal for this input is detected. That is, MC20 Module alarm status is reset and parameters are changed to the last set values.	Positive transitional contact —| P |—
Input 2	Output variable clear	If this input is OFF when input 1 is ON, output variables are not cleared. If this input is ON, then output variables are not cleared.	N.O. contact —| |—
Output 1	Machine resetting	This output is turned ON by the leading edge of the input 1 signal, and is turned OFF when machine reset execution is completed. That is, it is turned OFF when MC control relay RSTL (machine resetting) turns OFF.	
Output 2	Error  ON pulse signal	When an error occurs, the ON pulse signal is output and the instruction is terminated. Output 3, however, is not turned ON. The pulse width is one scan. For details, refer to *2.1.5 Error Status.*	
Output 3	Reset completion  ON pulse signal	The ON pulse signal is output when the machine reset operation is completed properly. The pulse width is one scan.	

Note (a) Use the contact specified under "Input Contacts" to turn input 1 ON and OFF.

(b) The pulse width of the ON pulse signal is one scan.

2) The following items are executed at the specified MC20 Module when the machine reset operation is executed by turning input 1 ON.

- Travelling controlled axes are decelerated and stopped.
- Any remaining travelling data is deleted.
- Motion program execution pointers are deleted.
- Part of the alarms are reset. (See note on the next page.)
- Alarm histories are cleared.
- Parameters are changed to the last set values.
- Parameters renewed with the PARAMETER SETTING (PRM) instruction are validated.
- The following data is cleared: (1) MC relays, and (2) M code relays.

The following data, however, is retained.

- Common variables
- System variables
- H variables
- Point table values
- Matrix setting instruction values
- Machine coordinate system's zero point in an incremental position detecting system
- Home position for each axis specified in an absolute position detecting system

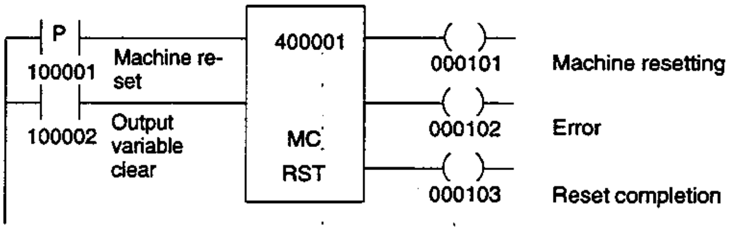
Note The alarm reset is executed, but the following important alarms are not reset.

Code	Message	Code	Message
79	Parameter destruction	A14	Absolute encoder alarm
80	Axis name duplicated designation	A15	Absolute encoder communication error
82	Illegal parameter	A18	Absolute encoder battery alarm
84	E^{2}PROM error		

The status of these alarms can only be reset by powering up or by executing MODULE RESET (MRS). The Servopack alarm status, however, can be reset by means of the alarm reset output from the MC20 Module.

◀EXAMPLE▶ In this example, the machine reset operation is executed for MC20 Module 2, and the output variables are cleared.

Ladder Diagram



400001	2	MC20 Module no. designation
400002		Used by system.
400003		Status (Refer to *2.1.5 Error Status.*)

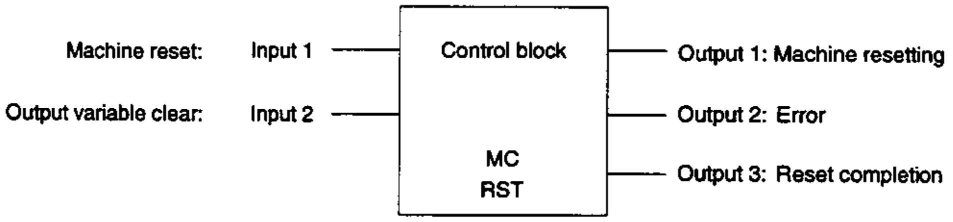
Operation

The MC20 Module number "2" is specified in holding register 400001.

After input 2 (10002) is turned ON, the machine reset operation is executed when input 1 (100001) is ON. The alarm status is reset and the parameters are changed to the last set values. In addition, the output variables are cleared.

Note Only one MACHINE RESET (RST) instruction can be executed with respect to a single MC20 Module. Do not execute more than one instruction at the same time.

SUMMARY The following information summarizes the instruction.



RST structure

Table 2.29 RST Structural Elements

Element	Meaning		Possible Settings	
Top	4xxxxx	MC20 Module number: 1 or 2	Holding registers	400001 to 409997 (W00001 to W09997)
	4xxxxx+1	Used by system.	Do not write.	
	4xxxxx+2	Status		
Middle	None		———	
Bottom	Indicates the RST instruction.		———	

Table 2.30 RST Operation

Input 1	Input 2	Input 3	Status/Remarks	Output 1	Output 2	Output 3
OFF ↓ ON	OFF	None	• Machine reset is executed at the leading edge of input 1. Output variables, however, are not cleared. • During the machine reset operation, MC control relay RSTL (machine resetting) is ON. • When the machine reset operation has been properly completed, the output 3 (reset completion) pulse signal is output.	ON ↓ OFF	OFF	OFF ↓ ON pulse
OFF ↓ ON	ON	None	• Machine reset is executed at the leading edge of input 1, and output variables are cleared. • During the machine reset operation, MC control relay RSTL (machine resetting) is ON. • When the machine reset operation has been properly completed, the output 3 (reset completion) pulse signal is output.	ON ↓ OFF	OFF	OFF ↓ ON pulse

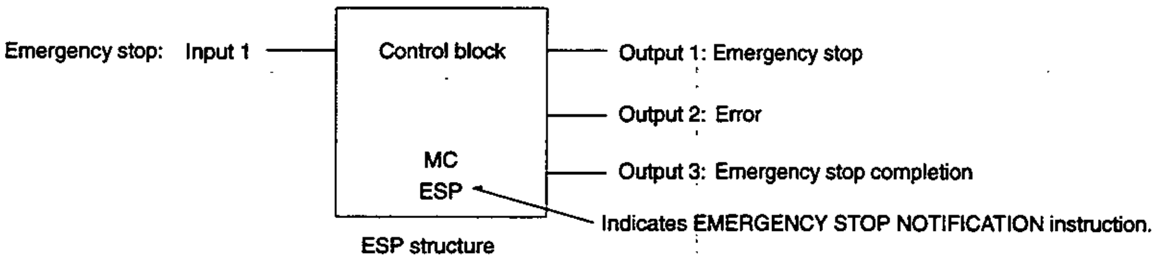
2.3.5 EMERGENCY STOP NOTIFICATION (ESP)

Function

ESP notifies the MC20 Module that the emergency stop button for the relevant system has been pressed.

Structure

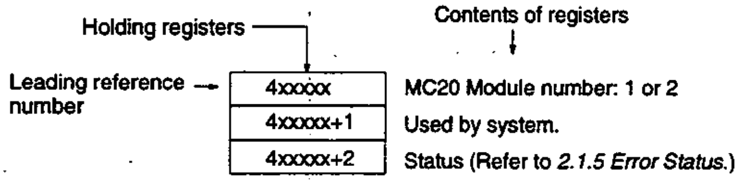
1) The structure of the ESP instruction is shown below.


Emergency stop: Input 1
Control block
MC
ESP
Output 1: Emergency stop
Output 2: Error
Output 3: Emergency stop completion
Indicates EMERGENCY STOP NOTIFICATION instruction.

ESP structure

2) The settings shown below are required for the control block in the top element. The leading reference number is specified for the top element. Refer to *Table 2.31 ESP Structural Elements* for details on the ranges of reference numbers that can be specified.

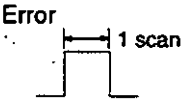
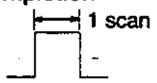
Control Block



Holding registers	Contents of registers
4xxxxx	MC20 Module number: 1 or 2
4xxxxx+1	Used by system.
4xxxxx+2	Status (Refer to *2.1.5 Error Status.*)

Operation

1) The following table shows the basic operation of the inputs and outputs.

I/O Number	Operation Name	Description	Input Contacts
Input 1	Emergency stop	This input is connected to the contact that turns ON when the emergency stop button is pressed for the relevant system. When this input is ON, the MC20 Module is notified that the emergency stop is in effect.	N.O. contact —| |—
Output 1	Emergency stop	This output turns ON when MC control relay ESPL (emergency stop) is turned ON by the turning ON of input 1.	
Output 2	Error  ON pulse signal	When an error occurs, the ON pulse signal is output and the instruction is terminated. Output 3, however, is not turned ON. The pulse width is one scan. For details, refer to *2.1.5 Error Status.*	
Output 3	Emergency stop completion  ON pulse signal	The output 3 pulse signal is output when MC control relay ESPL (emergency stop) is turned OFF by the turning OFF of input 1. The pulse width is one scan.	

Note (a) Use the contact specified under "Input Contacts" to turn input 1 ON and OFF.

(b) The pulse width of the ON pulse signal is one scan.

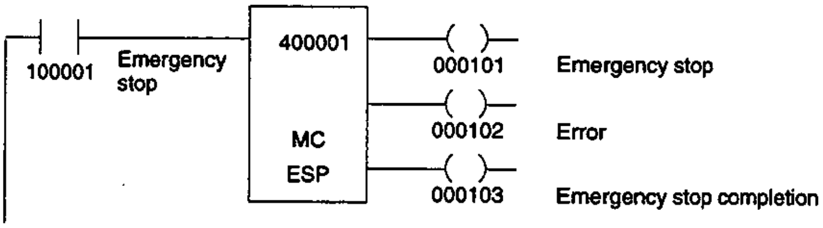
2) The following items are executed at the specified MC20 Module when the emergency stop notification is executed by turning ON input 1.

- Travelling controlled axes are decelerated and stopped.
- Any remaining travelling data is deleted.
- Motion program execution pointers are deleted.
- The servo is turned OFF.

All MC20 Module internal status and data other than the above is retained.

◀EXAMPLE▶ In this example, MC20 Module 1 is notified that an emergency stop is in effect.

Ladder Diagram



400001	1	MC20 Module no. designation
400002		Used by system.
400003		Status (Refer to *2.1.5 Error Status.*)

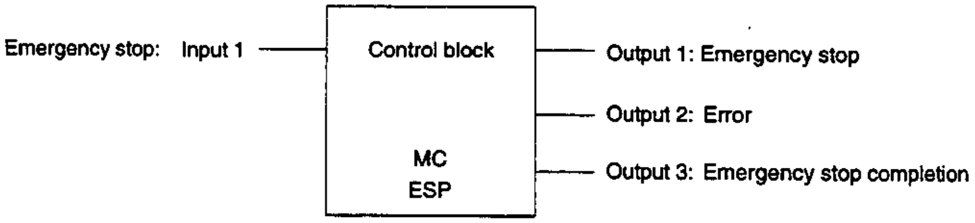
Operation

The MC20 Module number "1" is specified in holding register 400001.

When input 1 (100001) is turned ON, MC20 Module 1 is notified that an emergency stop is in effect. Any travelling controlled axes are decelerated and stopped.

Note Only one EMERGENCY STOP NOTIFICATION (ESP) instruction can be executed with respect to a single MC20 Module. Do not execute two or more instructions at the same time.

➧ SUMMARY The following information summarizes the instruction.



• ESP structure

Table 2.31 ESP Structural Elements

Element	Meaning		Possible Settings	
Top	4xxxxx	MC20 Module number: 1 or 2	Holding registers	400001 to 409997 (W00001 to W09997)
	4xxxxx+1	Used by system.	Do not write.	
	4xxxxx+2	Status		
Middle	None		——	
Bottom	Indicates the ESP instruction.		——	

Table 2.32 ESP Operation

Input 1	Status/Remarks	Output 1	Output 2	Output 3
ON	• When input 1 is ON, the specified MC20 Module is notified that an emergency stop is in effect. MC control relay ESPL (emergency stop) is turned ON.	ON	OFF	OFF
OFF	• When input 1 is OFF, the specified MC20 Module is notified that the emergency stop is finished. MC control relay ESPL (emergency stop) is turned OFF. • The output 3 pulse signal is output as the emergency stop completion signal.	OFF	OFF	ON pulse

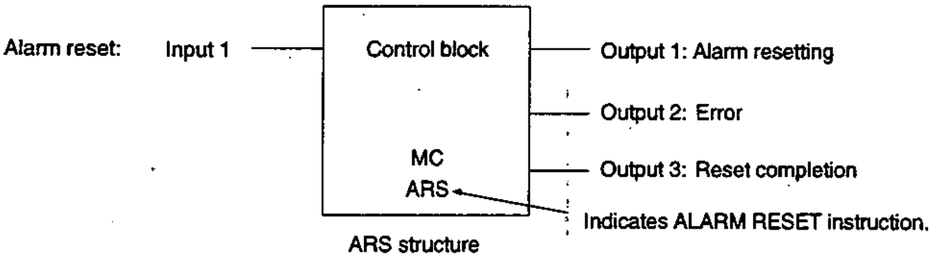
2.3.6 ALARM RESET (ARS)

Function

ARS resets the MC20 Module's alarm status and clears the alarm history.

Structure

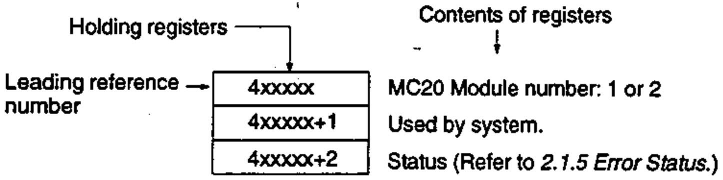
1) The structure of the ARS instruction is shown below.


Alarm reset:
Input 1
Control block
MC
ARS
Output 1: Alarm resetting
Output 2: Error
Output 3: Reset completion
Indicates ALARM RESET instruction.

ARS structure

2) The settings shown below are required for the control block in the top element. The leading reference number is specified for the top element. Refer to *Table 2.33 ARS Structural Elements* for details on the ranges of reference numbers that can be specified.

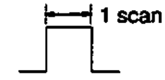
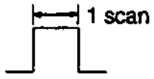
Control Block



Holding registers	Contents of registers
4xxxxx	MC20 Module number: 1 or 2
4xxxxx+1	Used by system.
4xxxxx+2	Status (Refer to *2.1.5 Error Status.*)

Operation

1) The following table shows the basic operation of the inputs and outputs.

I/O Number	Operation Name	Description	Input Contacts
Input 1	Alarm reset	The alarm reset is executed when the leading edge of the OFF→ON signal for this input is detected. That is, the alarm status is reset and the alarm history is cleared.	Positive transitional contact —| P |—
Output 1	Alarm resetting	This output is turned ON by the leading edge of the input 1 signal, and it is turned OFF when the alarm reset operation is completed, i.e., when the MC control relay ARSL (alarm reset completion) ON pulse signal is output.	
Output 2	Error  ON pulse signal	When an error occurs, the ON pulse signal is output and the instruction is terminated. Output 3, however, is not turned ON. The pulse width is one scan. For details, refer to *2.1.5 Error Status.*	
Output 3	Reset completion  ON pulse signal	The output 3 ON pulse signal is output when the alarm reset operation has been completed properly. The pulse width is one scan.	

Note (a) Use the contact specified under "Input Contacts" to turn input 1 ON and OFF.

(b) The pulse width of the ON pulse signal is one scan.

2) The following items are reset/cleared at the specified MC20 Module when alarm reset is executed by turning ON input 1.

- The alarm status is reset.

- The **alarm history data** is cleared.

All MC20 Module internal status and data other than the above is retained.

Note Even when alarm reset is executed, the following important alarms are not reset.

Code	Message	Code	Message
79	Parameter destruction	A14	Absolute encoder alarm
80	Axis name duplicated designation	A15	Absolute encoder communication error
82	Illegal parameter	A18	Absolute encoder battery alarm
84	E^{2}PROM error		

TERMS

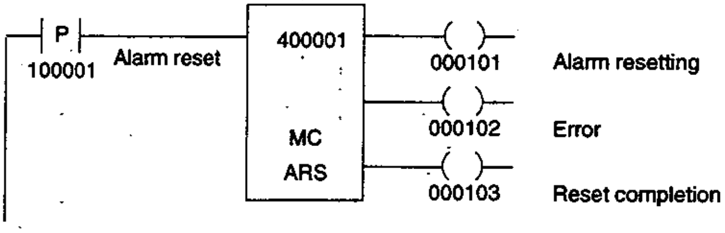
Alarm History Data

The MC20 Module maintains an error history in which a maximum of nine errors that have occurred can be stored at one time. This alarm history data can be displayed on the Programming Device, after executing the MONITOR (MON) instruction in which monitor number 0010 is designated.

The status of these alarms can only be reset by powering up or by executing MODULE RESET (MRS). The Servopack alarm status, however, can be reset by means of the alarm reset output from the MC20 Module.

◀**EXAMPLE**▶ In this example, module reset is executed for MC20 Module 1.

Ladder Diagram



400001	1	MC20 Module no. designation
400002		Used by system.
400003		Status (Refer to *2.1.5 Error Status.*)

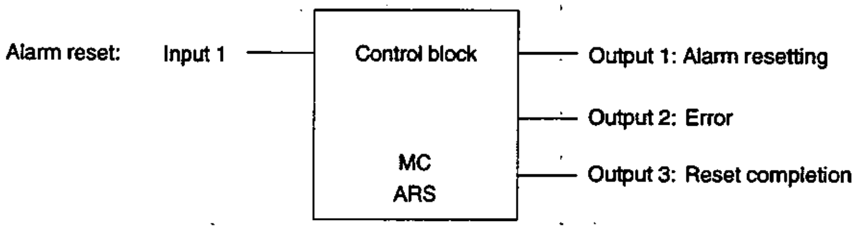
Operation

The MC20 Module number "1" is specified in holding register 400001.

When input 1 (100001) is turned ON, the alarm reset is executed. The alarm status of the MC20 Module 1 is reset and the alarm history is cleared.

Note Only one ALARM RESET (ARS) instruction can be executed with respect to a single MC20 Module. Do not execute two or more instructions at the same time.

➧ **SUMMARY** The following information summarizes the instruction.



ARS structure

Table 2.33 ARS Structural Elements

Element	Meaning		Possible Settings	
Top	4xxxxx	MC20 Module number: 1 or 2	Holding registers	400001 to 409997 (W00001 to W09997)
	4xxxxx+1	Used by system.	Do not write.	
	4xxxxx+2	Status		
Middle	None		——	
Bottom	Indicates the ARS instruction.		——	

Table 2.34 ARS Operation

Input 1	Status/Remarks	Output 1	Output 2	Output 3
OFF ↓ ON	• Alarm reset is executed at the leading edge of input 1. That is, the alarm status is reset and the alarm history is cleared.	ON ↓	OFF	OFF ↓
	• The output 3 ON pulse signal is output when the above alarm reset operation has been completed. The MC control relay ARSL (alarm reset completion) ON pulse signal is also output.	OFF		ON pulse

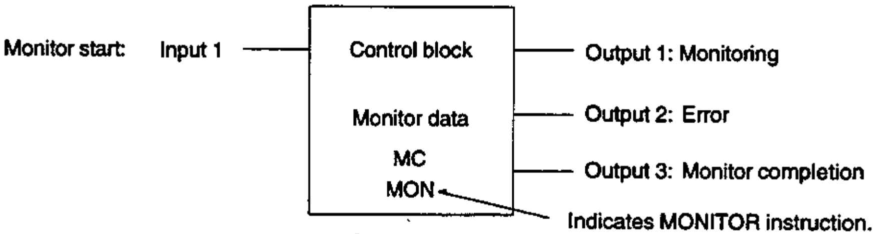
2.3.7 MONITOR (MON)

Function

MON monitors all kinds of status and data for the specified MC20 Module controlled axis.

Structure

1) The structure of the MON instruction is shown below.



MON structure

2) The settings shown below are required for the control block in the top element. The leading reference number is specified for the top element. Refer to *Table 2.35 MON Structural Elements* for details on the ranges of reference numbers that can be specified.

Control Block

Holding registers	Contents of registers
4xxxxx (Leading reference number)	MC20 Module number: 1 or 2
4xxxxx+1	Monitor number: 0001 to 4603
4xxxxx+2	Controlled axis designation: 1, 2, 4, or 8
4xxxxx+3	Used by system.
4xxxxx+4	Status (Refer to *2.1.5 Error Status.*)

3) Specify a monitor number to designate the object that is to be monitored. The relationships between the monitor numbers and monitored objects are shown in Table 2.35.

Table 2.35 Monitor Numbers and Monitored Objects

Monitor Number	N =	Monitored Object
000N	1	Current position
	2	Following error
	3	Current speed
	4	Instruction speed
0010	—	Current alarm and alarm history (9 items)
002N	1 to 6	MC20 Module external input status
0031	—	MC20 Module external output status
01NN	0 to 15	Common parameters
0200	—	O and B numbers being executed
0300	—	H variables (H1 to H8)
1NNN	1 to 500	Point table data
2NNN	1 to 199	Common variables
3NNN	001 to 008	System variables
	101 to 116	Link input variables
	201 to 216	Link output variables
4NNN	101 to 603	Individual axis parameters (NNN: Individual axis parameter 3-digit numbers from PA onward)

4) When monitor no. 4NNN (individual axis parameters) is specified, specify in holding register 4xxxxx+2 the number corresponding to the controlled axis which is to be monitored. For example, specify "4" to monitor the individual axis parameters for axis 3. Multiple axes cannot be designated for a single instruction. The numbers corresponding to the four controlled axes are shown below.

Axis Number	Corresponding Number
Axis 1	1
Axis 2	2
Axis 3	4
Axis 4	8

This axis designation will be ignored for monitoring objects other than monitor no. 4NNN (individual axis parameters). n this case, in principle, set each corresponding value to zero.

5) As shown in the following illustration, the results of the monitoring operation are stored in the holding registers in the middle element of the instruction. Specify the leading reference number for the results. In the following illustration, this is represented by 4xxxxx+5.

Although the number of holding registers varies with the monitor number (i.e., the object of monitoring), 16 holding registers are always used for monitor data. The capacity of a single holding register is 16 bits in binary.

Monitor Data

Leading reference number: 4xxxxx+5

Holding registers	Contents of registers		
4xxxxx+5	Monitor data 1		Monitor data 1 – L
4xxxxx+6	Monitor data 2		Monitor data 1 – H
4xxxxx+7	• • •		• • •
4xxxxx+...	• • •	or	• • •
4xxxxx+14	Monitor data 9		Monitor data 5 – L
4xxxxx+15	Monitor data 10		Monitor data 5 – H
4xxxxx+16			• • •
4xxxxx+...			• • •
4xxxxx+...			Monitor data 8 – L
4xxxxx+20			Monitor data 8 – H
	(16-bit data)		(32-bit data)

Results of monitoring operation (maximum of 16 registers)

INFO

It is possible to specify 4yyyyy instead of 4xxxxx+5 as the leading reference number. To avoid confusion, the use of continuous reference numbers is recommended.

6) The format of the monitor data stored in the holding registers is as follows:

a) Binary 16 bits: Monitor number, O number, B number, etc.

b) Binary 32 bits (signed): Current position, variables, parameters, etc.

Note As used here, "binary 32 bits (signed)" means the following:

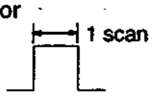
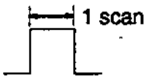
Positive value:	Normal 32-bit binary.
Negative value:	Expressed as two's complement.

7) The Programming Device can be used for viewing the monitor data stored by the MONITOR instruction. Specify the reference number of the holding registers used for the designated monitor data, and display the data using either the decimal display mode or the double decimal display mode.

8) The summary at the end of this section provides a monitor data register map which shows the storage locations for the various monitor numbers.

Operation

The following table shows the basic operation of the inputs and outputs.

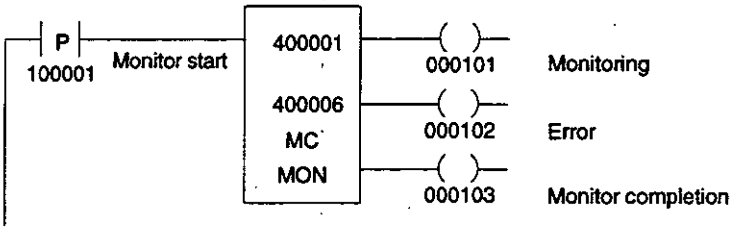
I/O Number	Operation Name	Description	Input Contacts
Input 1	Monitor start	Monitoring is executed when the leading edge of the OFF→ON signal for this input is detected. That is, the monitor data corresponding to the specified monitor number is stored in the specified data registers.	Positive transitional contact —| P |—
Output 1	Monitoring	This output is turned ON by the leading edge of the input 1 signal, and it is turned OFF when the monitoring operation is completed.	
Output 2	Error  ON pulse signal	When an error occurs, the ON pulse signal is output and the instruction is terminated. Output 3, however, is not turned ON. The pulse width is one scan. For details, refer to *2.1.5 Error Status.*	
Output 3	Monitor completion  ON pulse signal	The output 3 ON pulse signal is output when the monitoring operation has been completed properly. The pulse width is one scan.	

Note (a) Use the contact specified under "Input Contacts" to turn input 1 ON and OFF.

(b) The pulse width of the ON pulse signal is one scan.

◀EXAMPLE▶ In this example, the individual axis parameter PA202 (rapid traverse speed) set value is monitored for the controlled axis 3 of the MC20 Module 1.

Ladder Diagram



Control block	400001	1	MC20 Module no. designation
	400002	4202	Monitor number designation
	400003	4	Controlled axis designation (= axis 3)
	400004		Used by system.
	400005		Status (Refer to *2.1.5 Error Status.*)
Monitor data	400006		Controlled axis number
	400007		Rapid traverse speed – L
	400008		Rapid traverse speed – H

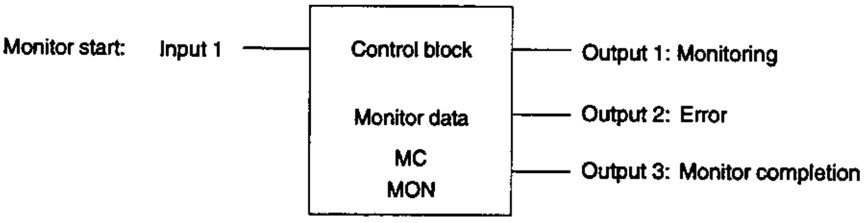
Operation

The above settings are made in control block holding registers 400001 to 400003, and 400006 is designated as the leading reference number for monitor data.

When input 1 (100001) is turned from OFF to ON, the axis 3 for the MC20 Module 1 "rapid traverse speed" parameter values are stored in holding registers 400007 and 400008.

Note Multiple MONITOR (MON) instructions can be written into a ladder logic program for a single MC20 Module. It is not possible, however, to execute two or more instructions at the same time.

SUMMARY The following information summarizes the instruction.



MON structure

Table 2.36 MON Structural Elements

Element	Meaning		Possible Settings	
Top	4xxxxx	MC20 Module number: 1 or 2	Holding registers	400001 to 409979 (W00001 to W09979)
	4xxxxx+1	Monitor number: 0001 to 4GNN (4603)	———	
	4xxxxx+2	Controlled axis designation: 1, 2, 4, or 8	———	
	4xxxxx+3	Used by system.	Do not write.	
	4xxxxx+4	Status		
Middle	4xxxxx+5	Monitor data 1 or 1–L	Holding registers	400006 to 409984 (W00006 to W09984)
			16 registered used.	
	4xxxxx+6	Monitor data 2 or 1–H	16 registered used.	
	4xxxxx+...	Monitor data 3 or 2–L	16 registered used.	
	4xxxxx+...	. . .	16 registered used.	
	4xxxxx+19	Monitor data 8–L	16 registered used.	
	4xxxxx+20	Monitor data 8–H	16 registered used.	
Bottom	Indicates the MON instruction.		———	

Table 2.37 MON Operation

Input 1	Status/Remarks	Output 1	Output 2	Output 3
OFF ↓ ON	• Monitoring is executed at the leading edge of input 1. That is, the monitor data corresponding to the specified monitor number is stored in the specified holding registers.	ON ↓	OFF	OFF ↓
	• The output 3 (monitor completion) ON pulse signal is output when the monitoring operation has been properly completed.	OFF		ON pulse

Monitor Numbers and Monitored Objects

Monitor Number	N =	Monitored Object
000N	1	Current position
	2	Following error
	3	Current speed
	4	Instruction speed
0010	—	Current alarm and alarm history (9 items)
002N	1 to 6	MC20 Module external input status
0031	—	MC20 Module external output status
01NN	0 to 15	Common parameters
0200	—	O and B numbers being executed
0300	—	H variables (H1 to H8)
1NNN	1 to 500	Point table data
2NNN	1 to 199	Common variables
3NNN	001 to 008	System variables
	101 to 116	Link input variables
	201 to 216	Link output variables
4NNN	101 to 603	Individual axis parameters (NNN: Individual axis parameter 3-digit numbers from PA onward)

Monitor Data Register Map (Storage Locations)

Monitor number: 0001
Monitored object: Current position

4xxxxx+5	Axis 1 current position – L
4xxxxx+6	Axis 1 current position – H
4xxxxx+7	Axis 2 current position – L
4xxxxx+8	Axis 2 current position – H
4xxxxx+9	Axis 3 current position – L
4xxxxx+10	Axis 3 current position – H
4xxxxx+11	Axis 4 current position – L
4xxxxx+12	Axis 4 current position – H

[Specified unit]

Monitor number: 0002
Monitored object: Following error

4xxxxx+5	Axis 1 following error – L
4xxxxx+6	Axis 1 following error – H
4xxxxx+7	Axis 2 following error – L
4xxxxx+8	Axis 2 following error – H
4xxxxx+9	Axis 3 following error – L
4xxxxx+10	Axis 3 following error – H
4xxxxx+11	Axis 4 following error – L
4xxxxx+12	Axis 4 following error – H

[Pulses]

Monitor number: 0003
Monitored object: Current speed

4xxxxx+5	Axis 1 current speed – L
4xxxxx+6	Axis 1 current speed – H
4xxxxx+7	Axis 2 current speed – L
4xxxxx+8	Axis 2 current speed – H
4xxxxx+9	Axis 3 current speed – L
4xxxxx+10	Axis 3 current speed – H
4xxxxx+11	Axis 4 current speed – L
4xxxxx+12	Axis 4 current speed – H

[r/min]

Monitor number: 0004
Monitored object: Instruction speed

4xxxxx+5	Axis 1 instruction speed – L
4xxxxx+6	Axis 1 instruction speed – H
4xxxxx+7	Axis 2 instruction speed – L
4xxxxx+8	Axis 2 instruction speed – H
4xxxxx+9	Axis 3 instruction speed – L
4xxxxx+10	Axis 3 instruction speed – H
4xxxxx+11	Axis 4 instruction speed – L
4xxxxx+12	Axis 4 instruction speed – H

[mm/min]

Monitor number: 0010
Monitored object: Current alarm and alarm history (9 items)

4xxxxx+5	Current alarm
4xxxxx+6	Alarm history 1
4xxxxx+7	Alarm history 2
4xxxxx+8	Alarm history 3
4xxxxx+9	Alarm history 4
4xxxxx+10	Alarm history 5
4xxxxx+11	Alarm history 6
4xxxxx+12	Alarm history 7
4xxxxx+13	Alarm history 8
4xxxxx+14	Alarm history 9

Monitor number: 01NN
Monitored object: Common parameters

4xxxxx+5	Parameter value – L
4xxxxx+6	Parameter value – H

Monitor number: 0200
Monitored object: O and B numbers being executed

4xxxxx+5	O number (program number)
4xxxxx+6	B number (block number)

Monitor number: 1NNN
Monitored object: Point table data

4xxxxx+5	Point number
4xxxxx+6	Axis 1 point data – L
4xxxxx+7	Axis 1 point data – H
4xxxxx+8	Axis 2 point data – L
4xxxxx+9	Axis 2 point data – H
4xxxxx+10	Axis 3 point data – L
4xxxxx+11	Axis 3 point data – H
4xxxxx+12	Axis 4 point data – L
4xxxxx+13	Axis 4 point data – H

Monitor number: 3NNN
Monitored object: System variables

4xxxxx+5	System variable value – L
4xxxxx+6	System variable value – H

Monitor number: 002N
Monitored object: MC20 Module external input status

4xxxxx+5	Input status (see Note 2 on the next page)

Monitor number: 003N
Monitored object: MC20 Module external output status

4xxxxx+5	Output status (see Note 3 on the next page)

Monitor number: 0300
Monitored object: H variables (H1 to H8)

4xxxxx+5	H 1–L
4xxxxx+6	H 1–H
4xxxxx+7	H 2–L
4xxxxx+8	H 2–H
4xxxxx+...	• • •
4xxxxx+...	• • •
4xxxxx+...	• • •
4xxxxx+17	H 7–L
4xxxxx+18	H 7–H
4xxxxx+19	H 8–L
4xxxxx+20	H 8–H

Monitor number: 2NNN
Monitored object: Common variables

4xxxxx+5	Common variable value – L
4xxxxx+6	Common variable value – H

Monitor number: 3NNN
Monitored object: Link input variables

4xxxxx+5	Link input variable value – L
4xxxxx+6	Link input variable value – H

Monitor number: **3NNN**
Monitored object: **Link output variables**

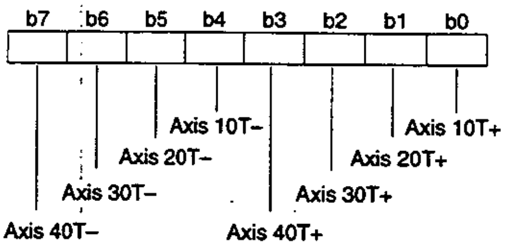
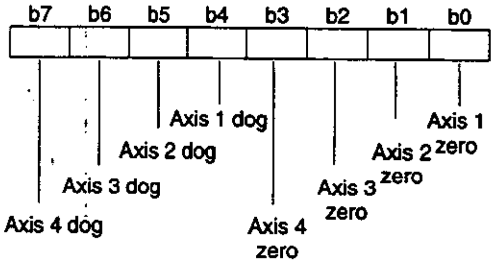
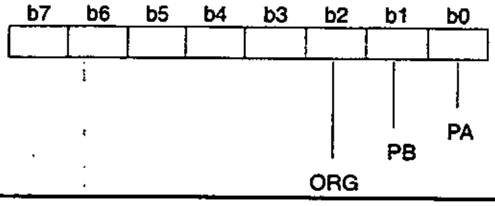
4xxxxx+5	Link output variable value – L
4xxxxx+6	Link output variable value – H

Monitor number: **4NNN**
Monitored object: **Individual axis parameters**

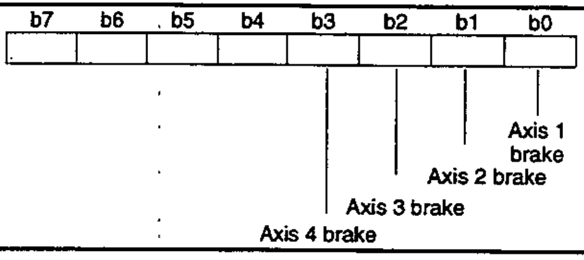
4xxxxx+5	Axis number
4xxxxx+6	Parameter value – L
4xxxxx+7	Parameter value – H

Note (1) The contents of any registers up to 4xxxxx+5 to 4xxxxx+20 without effective data will be reset to "0."

(2) The meaning of the external input status is shown below.

Monitor number	Name	1 = ON, 0 = OFF
0021	Overtravel signal	 b7 b6 b5 b4 b3 b2 b1 b0 Axis 10T– Axis 10T+ Axis 20T– Axis 20T+ Axis 30T– Axis 30T+ Axis 40T– Axis 40T+
0022	Zero, deceleration limit switch signal	 b7 b6 b5 b4 b3 b2 b1 b0 Axis 1 dog Axis 1 zero Axis 2 dog Axis 2 zero Axis 3 dog Axis 3 zero Axis 4 dog Axis 4 zero
0023 0024 0025 0026	Axis 1 encoder pulse Axis 2 encoder pulse Axis 3 encoder pulse Axis 4 encoder pulse	

(3) The meaning of the external output status is shown below.

Monitor number	Name	1 = ON, 0 = OFF
0031	Brake output signal	 b7 b6 b5 b4 b3 b2 b1 b0 Axis 1 brake Axis 2 brake Axis 3 brake Axis 4 brake

2.4 Data Setting Instructions

This section explains the ladder motion instructions that set and change CPU Module parameters, variables, constants, etc., from the ladder logic program. These instructions are used for relatively advanced applications.

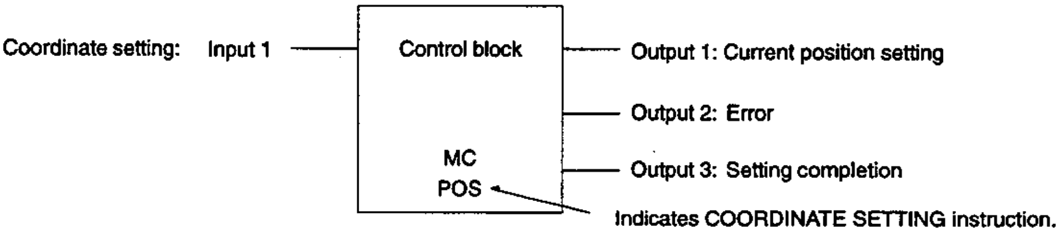
2.4.1 COORDINATE SETTING (POS)

Function

POS changes the current position to the coordinate values for the specified axes, and creates a new coordinate system. It has the same effect as the motion program's CURRENT POSITION SETTING (POS) instruction. The new coordinate system set by COORDINATE SETTING (POS) is called a "work coordinate system."

Structure

1) The structure of the POS instruction is shown below.


Coordinate setting:
Input 1
Control block
MC
POS
Output 1: Current position setting
Output 2: Error
Output 3: Setting completion
Indicates COORDINATE SETTING instruction.

POS structure

2) The settings shown below are required for the control block in the top element. The leading reference number is specified for the top element. Refer to *Table 2.38 POS Structural Elements* for details on the ranges of reference numbers that can be specified.

Control Block

	Holding registers	Contents of registers	
Leading reference number →	4xxxxx	MC20 Module number: 1 or 2	
	4xxxxx+1	Controlled axis designation: 1, 2, 4, 8, or sum	
	4xxxxx+2	Axis 1 coordinate value – L	0 to ±99,999,999
	4xxxxx+3	Axis 1 coordinate value – H	
	4xxxxx+4	Axis 2 coordinate value – L	0 to ±99,999,999
	4xxxxx+5	Axis 2 coordinate value – H	
	4xxxxx+6	Axis 3 coordinate value – L	0 to ±99,999,999
	4xxxxx+7	Axis 3 coordinate value – H	
	4xxxxx+8	Axis 4 coordinate value – L	0 to ±99,999,999
	4xxxxx+9	Axis 4 coordinate value – H	
	4xxxxx+10	Used by system.	
	4xxxxx+11	Status (Refer to *2.1.5 Error Status.*)	

3) To designate the controlled axes, specify in holding register 4xxxxx+1 the numbers corresponding to the axes that are to have the current position coordinates set. The numbers are shown in the following table. To designate multiple axes with a single instruction, specify the sum of the desired axes. For example, if the desired axes are 1, 2, 3, and 4, the corresponding numbers would be 1, 2, 4, and 8. Therefore the sum would be 15 (i.e., 1 + 2 + 4 + 8 = 15).

Axis Number	Corresponding Number
Axis 1	1
Axis 2	2
Axis 3	4
Axis 4	8

4) The axis x coordinate value is directly designated in **signed 32-bit binary**, using two registers.

Axis 1 coordinate value – H	Axis 1 coordinate value – L	
4xxxxx+3	4xxxxx+2	= signed 32-bit binary

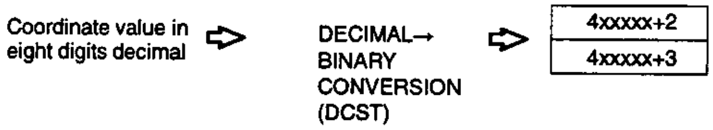
TERMS

Signed 32-bit Binary

The usage of "binary" here is defined as follows:

Positive value: Normal 32-bit binary.
Negative value: Expressed as two's complement.

DECIMAL→BINARY CONVERSION (DCST) instruction is convenient when carrying out direct designation of binary numbers.


Coordinate value in eight digits decimal
DECIMAL→ BINARY CONVERSION (DCST)
4xxxxx+2
4xxxxx+3

5) The unit for inputting the value designated as the coordinate value is determined by the MC20 Module's common parameter P0005 (decimal position).

MC20 Module Parameter Setting	Reference Unit	
	Linear Axis	Rotary Axis
P0005 = 1	0.1 mm	0.1°
P0005 = 2	0.01 mm	0.01°
P0005 = 3	0.001 mm	0.001°

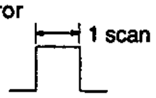
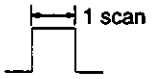
◀EXAMPLE▶ Linear Axis:

In this example, the coordinate value designation is actually specified in binary but appears in decimal.

Coordinate value designation:	–455000
Actual coordinate setting (when P0005 = 3):	–455.000 mm

Operation

The following table shows the basic operation of the inputs and outputs.

I/O Number	Operation Name	Description	Input Contacts
Input 1	Coordinate setting	The coordinates are set when the leading edge of the OFF→ON signal for this input is detected. That is, the current position is changed, for the specified controlled axis only, to the specified coordinates.	Positive transitional contact —| P |—
Output 1	Current position setting	This output is turned ON by the leading edge of the input 1 signal, and it is turned OFF when the coordinate setting operation is completed.	
Output 2	Error  1 scan ON pulse signal	When an error occurs, the ON pulse signal is output and the instruction is terminated. Output 3, however, is not turned ON. The pulse width is one scan. For details, refer to *2.1.5 Error Status*.	
Output 3	Setting completion  1 scan ON pulse signal	The output 3 ON pulse signal is output when the coordinate setting operation has been completed properly. The pulse width is one scan.	

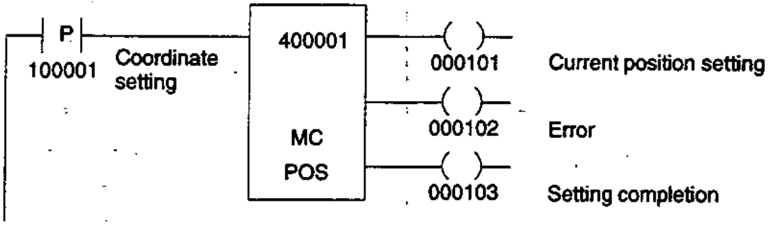
Note (a) Use the contact specified under "Input Contacts" to turn input 1 ON and OFF.

(b) The pulse width of the ON pulse signal is one scan.

⚠ Caution The COORDINATE SETTING (POS) command is used to create a new workpiece coordinate system. Before executing this command, be absolutely sure to check the workpiece coordinate system to make sure that the designation of the workpiece coordinate system is correct. Failure to do this may result in damage to equipment, serious personal injury, or even death.

◀EXAMPLE▶ In this example, MC20 Module 1's current position is set to the following coordinates: X = 100 mm; Y = 200 mm; Z = 300 mm; and S = 400 mm.

Ladder Diagram



400001	1	MC20 Module number: 1 or 2
400002	15	Controlled axis designation: Axes 1, 2, 3, and 4
400003	Binary	Axis 1 coordinate value – L
400004	100000	Axis 1 coordinate value – H
400005	Binary	Axis 2 coordinate value – L
400006	200000	Axis 2 coordinate value – H
400007	Binary	Axis 3 coordinate value – L
400008	300000	Axis 3 coordinate value – H
400009	Binary	Axis 4 coordinate value – L
400010	400000	Axis 4 coordinate value – H
400011		Used by system.
400012		Status (Refer to *2.1.5 Error Status*.)

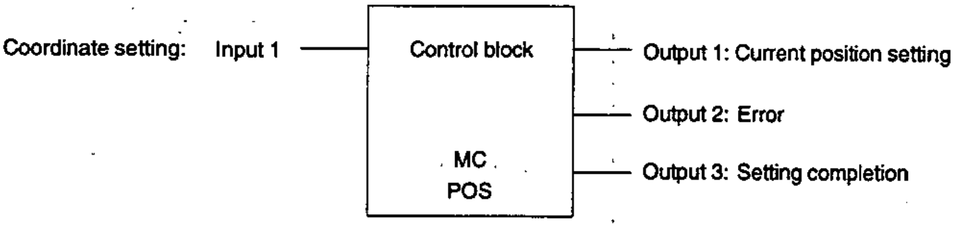
Operation

The above settings are made in control block holding registers 400001 to 400010.

When input 1 (100001) is turned ON, the current position is set to the specified coordinate value for axes 1, 2, 3, and 4 of MC20 Module 1.

Note Multiple COORDINATE SETTING (POS) instructions can be written into a ladder logic program for a single MC20 Module. It is not possible, however, to execute two or more instructions at the same time.

➧ SUMMARY The following information summarizes the instruction.


Coordinate setting: Input 1
Control block
MC
POS
Output 1: Current position setting
Output 2: Error
Output 3: Setting completion

POS structure

Table 2.38 POS Structural Elements

Element	Meaning		Possible Settings	
Top	4xxxxx	MC20 Module number: 1 or 2	Holding registers	400001 to 409988 (W00001 to W09988)
	4xxxxx+1	Axis designation: 1, 2, 4, 8, or sum (multiple axis designation)	——	
	4xxxxx+2	Axis 1 coordinate value – L	——	
	4xxxxx+3	Axis 1 coordinate value – H		
	Coordinate value: 0 to ±99,999,999			
	4xxxxx+4	Axis 2 coordinate value – L	——	
	4xxxxx+5	Axis 2 coordinate value – H		
	Coordinate value: 0 to ±99,999,999			
	4xxxxx+6	Axis 3 coordinate value – L	——	
	4xxxxx+7	Axis 3 coordinate value – H		
	Coordinate value: 0 to ±99,999,999			
	4xxxxx+8	Axis 4 coordinate value – L	——	
	4xxxxx+9	Axis 4 coordinate value – H		
	Coordinate value: 0 to ±99,999,999			
	4xxxxx+10	Used by system.	Do not write.	
	4xxxxx+11	Status		
Middle	None		——	
Bottom	Indicates the POS instruction.		——	

Table 2.39 POS Operation

Input 1	Status/Remarks	Output 1	Output 2	Output 3
OFF ↓ ON	• The coordinates are set at the leading edge of input 1. That is, the current position is changed, for the specified controlled axis only, to the specified coordinate value. • The output 3 (setting completion) ON pulse signal is output when the coordinate setting has been properly completed.	ON ↓ OFF	OFF	OFF ↓ ON pulse

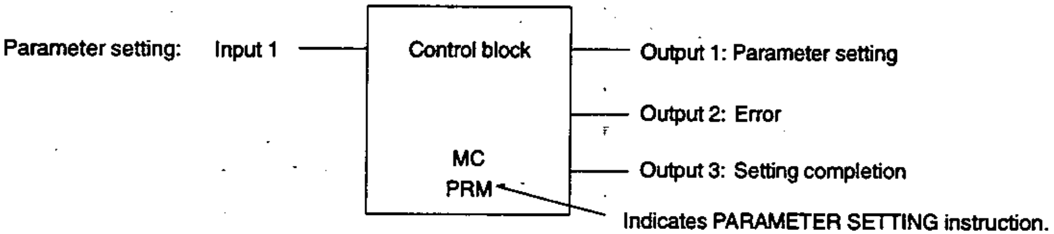
2.4.2 PARAMETER SETTING (PRM)

Function

PRM changes the set values for specified parameter. The changed settings, however, are not actually put into effect until after the MACHINE RESET (RST) instruction has been executed.

Structure

1) The structure of the PRM instruction is shown below.


Parameter setting:
Input 1
Control block
MC
PRM
Output 1: Parameter setting
Output 2: Error
Output 3: Setting completion
Indicates PARAMETER SETTING instruction.

PRM structure

2) The settings shown below are required for the control block in the top element. The leading reference number is specified for the top element. Refer to *Table 2.40 PRM Structural Elements* for details on the ranges of reference numbers that can be specified.

Control Block

Holding registers	Contents of registers	
4xxxxx (Leading reference number)	MC20 Module number: 1 or 2	
4xxxxx+1	Parameter number: 0006 to 4603	
4xxxxx+2	Parameter set value – L	0 to ±99,999,999
4xxxxx+3	Parameter set value – H	
4xxxxx+4	Used by system.	
4xxxxx+5	Status (Refer to *2.1.5 Error Status.*)	

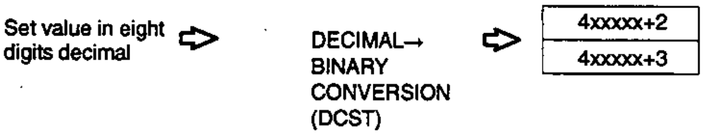
3) Enter a four-digit number (i.e., the four "n" digits in Pnnnn) for the parameter number. For example, to change the individual axis parameter PA509 (stored stroke limit (–) axis-4) value, set "4509" as the parameter number. (The "A" in the individual axis parameter indicates the axis number.)

4) Some parameters cannot be changed by the PARAMETER SETTING (PRM) instruction. The parameters that cannot be changed are marked by "X" in the column in the *Appendix 2 Parameters* column that indicates whether the parameters can be changed from the ladder logic program.

5) The parameter set value is directly designated in **signed 32-bit binary**, using two registers.

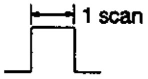
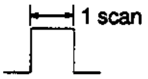
Parameter set value – H	Parameter set value – L	
4xxxxx+3	4xxxxx+2	= signed 32-bit binary

DECIMAL→BINARY CONVERSION (DCST) instruction is convenient when carrying out direct designation of binary numbers.


Set value in eight digits decimal
DECIMAL→ BINARY CONVERSION (DCST)
4xxxxx+2
4xxxxx+3

Operation

1) The following table shows the basic operation of the inputs and outputs.

I/O Number	Operation Name	Description	Input Contacts
Input 1	Parameter setting	The parameter is set when the leading edge of the OFF→ON signal for this input is detected. That is, the parameter set value specified by the parameter number is changed to the specified parameter set value.	Positive transitional contact —| P |—
Output 1	Parameter setting	This output is turned ON by the leading edge of the input 1 signal, and it is turned OFF when the parameter setting operation is completed.	
Output 2	Error  ON pulse signal	When an error occurs, the ON pulse signal is output and the instruction is terminated. Output 3, however, is not turned ON. The pulse width is one scan. For details, refer to *2.1.5 Error Status.*	
Output 3	Setting completion  ON pulse signal	The output 3 ON pulse signal is output when the parameter setting operation has been completed properly. The pulse width is one scan.	

Note (a) Use the contact specified under "Input Contacts" to turn input 1 ON and OFF.

(b) The pulse width of the ON pulse signal is one scan.

IMPORTANT

2) MC20 Module parameters are not put into effect immediately after being changed with the PARAMETER SETTING (PRM) instruction. To change the parameters to the new set values and put them into effect, execute the MACHINE RESET (RST) instruction. If the MODULE RESET (MRS) instruction is executed or the MC20 Module is turned OFF and ON after the PARAMETER SETTING (PRM) instruction is executed, the new set values will be invalid.

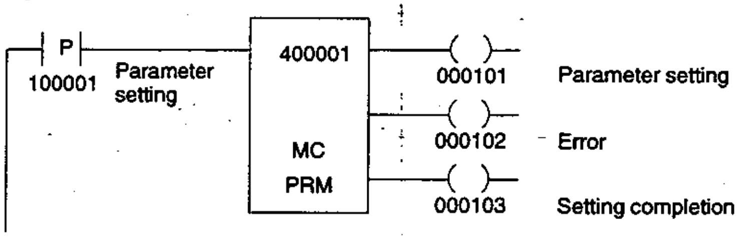
◀EXAMPLE▶ In this example, MC20 Module 1's individual axis parameter PA201 (maximum feed speed) is set to 12 m/min for axis 3.

TERMS

Signed 32-bit Binary

The usage of "binary" here is defined as follows:
Positive value: Normal 32-bit binary.
Negative value: Expressed as two's complement.

Ladder Diagram



400001	1	MC20 Module number: 1 or 2
400002	3201	Parameter number (201, axis 3)
400003	Binary	Axis 1 coordinate value – L
400004	12000	Axis 1 coordinate value – H
400005		Used by system.
400006		Status (Refer to *2.1.5 Error Status.*)

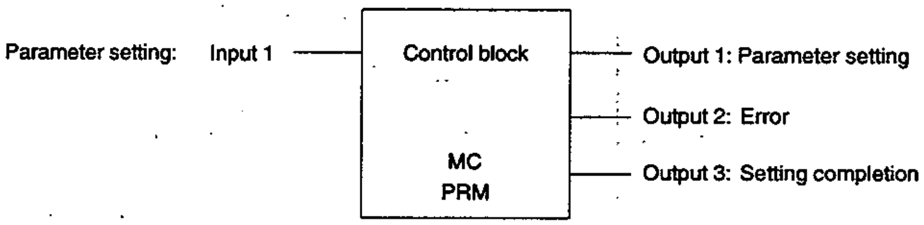
Operation

The above settings are made in control block holding registers 400001 to 400004.

When input 1 (100001) is turned ON, the maximum feed speed for axis 3 of MC20 Module 1 is changed to 12,000 mm/min.

Note Multiple PARAMETER SETTING (PRM) instructions can be written into a ladder logic program for a single MC20 Module. It is not possible, however, to execute two or more instructions at the same time.

➧ SUMMARY The following information summarizes the instruction.



PRM structure

Table 2.40 PRM Structural Elements

Element	Meaning	Possible Settings	
Top	4xxxxx MC20 Module number: 1 or 2	Holding registers	400001 to 409994 (W00001 to W09994)
	4xxxxx+1 Parameter number: 0006 to 4603	——	
	4xxxxx+2 Parameter set value – L 4xxxxx+3 Parameter set value – H Set value: 0 to ±99,999,999	——	
	4xxxxx+5 Used by system. 4xxxxx+6 Status	Do not write.	
Middle	None	——	
Bottom	Indicates the PRM instruction.	——	

Table 2.41 PRM Operation

Input 1	Status/Remarks	Output 1	Output 2	Output 3
OFF ↓ ON	• The parameter is set at the leading edge of input 1. That is, the parameter set value specified by the parameter number is changed to the specified parameter set value. • The output 3 (setting completion) ON pulse signal is output when the parameter setting has been properly completed.	ON ↓ OFF	OFF	OFF ↓ ON pulse

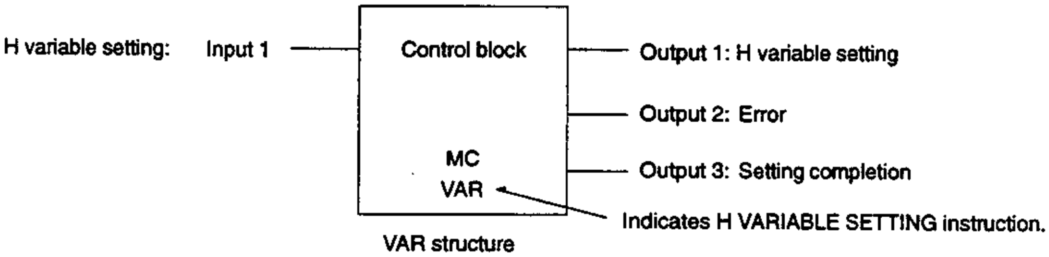
2.4.3 H VARIABLE SETTING (VAR)

Function

VAR sets the H variable values (H1 to H8). H variables are variables that can be used in MC20 Module motion program.

Structure

1) The structure of the VAR instruction is shown below.


H variable setting:
Input 1
Control block
MC
VAR
Output 1: H variable setting
Output 2: Error
Output 3: Setting completion
Indicates H VARIABLE SETTING instruction.

VAR structure

2) The settings shown below are required for the control block in the top element. The leading reference number is specified for the top element. Refer to *Table 2.42 VAR Structural Elements* for details on the ranges of reference numbers that can be specified.

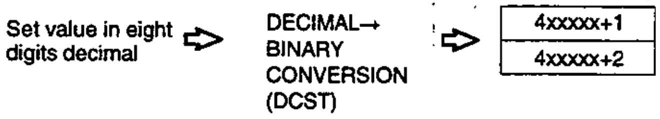
Control Block

	Holding registers	Contents of registers	
Leading reference number	4xxxxx	MC20 Module number: 1 or 2	
	4xxxxx+1	H1 set value – L	0 to ±99,999,999
	4xxxxx+2	H1 set value – H	
	4xxxxx+3	H2 set value – L	0 to ±99,999,999
	4xxxxx+4	H2 set value – H	
	4xxxxx+5	H3 set value – L	0 to ±99,999,999
	4xxxxx+6	H3 set value – H	
	4xxxxx+7	H4 set value – L	0 to ±99,999,999
	4xxxxx+8	H4 set value – H	
	4xxxxx+9	H5 set value – L	0 to ±99,999,999
	4xxxxx+10	H5 set value – H	
	4xxxxx+11	H6 set value – L	0 to ±99,999,999
	4xxxxx+12	H6 set value – H	
	4xxxxx+13	H7 set value – L	0 to ±99,999,999
	4xxxxx+14	H7 set value – H	
	4xxxxx+15	H8 set value – L	0 to ±99,999,999
	4xxxxx+16	H8 set value – H	
	4xxxxx+17	Used by system.	
	4xxxxx+18	Status (Refer to *2.1.5 Error Status.*)	

3) The Hn set value is directly designated in **signed 32-bit binary,** using two registers.

H1 set value – H	H1 set value – L	
4xxxxx+2	4xxxxx+1	= signed 32-bit binary

DECIMAL→BINARY CONVERSION (DCST) instruction is convenient when carrying out direct designation of binary numbers.


Set value in eight digits decimal
DECIMAL→ BINARY CONVERSION (DCST)
4xxxxx+1
4xxxxx+2

TERMS

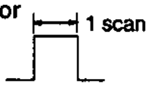
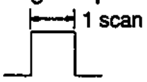
Signed 32-bit Binary

The usage of "binary" here is defined as follows:

Positive value: Normal 32-bit binary.
Negative value: Expressed as two's complement.

Operation

1) The following table shows the basic operation of the inputs and outputs.

I/O Number	Operation Name	Description	Input Contacts
Input 1	H variable setting	The H variable settings are executed when the leading edge of the OFF→ON signal for this input is detected. That is, the specified Hn set values are set for the H variables (H1 to H8).	Positive transitional contact —┤P├—
Output 1	H variable setting	This output is turned ON by the leading edge of the input 1 signal, and it is turned OFF when the H variable setting operation is completed.	
Output 2	Error  ON pulse signal	When an error occurs, the ON pulse signal is output and the instruction is terminated. Output 3, however, is not turned ON. The pulse width is one scan. For details, refer to *2.1.5 Error Status.*	
Output 3	Setting completion  ON pulse signal	The output 3 ON pulse signal is output when the H variable setting operation has been completed properly. The pulse width is one scan.	

Note (a) Use the contact specified under "Input Contacts" to turn input 1 ON and OFF.

(b) The pulse width of the ON pulse signal is one scan.

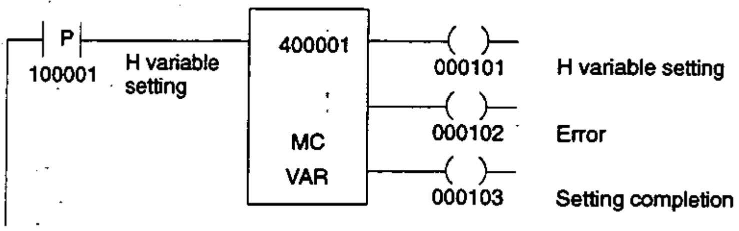
2) Values that are set or changed by H VARIABLE (VAR) go into effect immediately.

3) H variable values set by H VARIABLE (VAR) are retained, without being cleared, even when the following instructions are executed.

- MODULE RESET (MRS)
- MACHINE RESET (RST)
- EMERGENCY STOP NOTIFICATION (ESP)
- ALARM RESET (ARS)

◀EXAMPLE▶ In this example, MC20 Module 1's H variables are set to the following values: H1 = 2000; H2 = 4000; H3 = 6000; and H4 to H8 = 0.

Ladder Diagram



400001	1	MC Module number: 1 or 2
400002	Binary	H1 set value – L
400003	2000	H1 set value – H
400004	Binary	H2 set value – L
400005	4000	H2 set value – H
400006	Binary	H3 set value – L
400007	6000	H3 set value – H
400008	Binary	H4 set value – L
400009	0	H4 set value – H
4000••	• • •	• • •
4000••	• • •	• • •
400016	Binary	H8 set value – L
400017	0	H8 set value – H
400018		Used by system.
400019		Status (Refer to *2.1.5 Error Status.*)

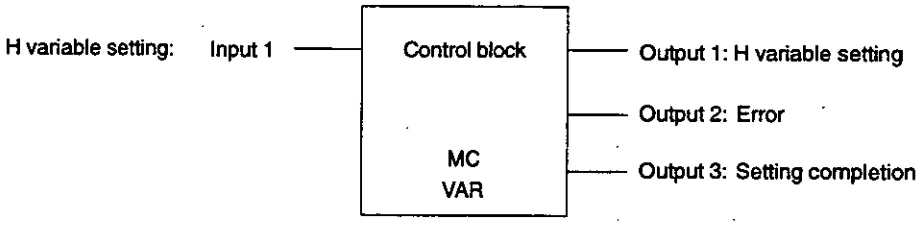
Operation

The above settings are made in control block holding registers 400001 to 400017.

When input 1 (100001) is turned ON, the values shown above are set for H variables H1, H2, and H3 for MC20 Module 1. H4 through H8 are set to "0."

Note Multiple H VARIABLE SETTING (VAR) instructions can be written into a ladder logic program for a single MC20 Module. It is not possible, however, to execute two or more instructions at the same time.

➧ **SUMMARY** The following information summarizes the instruction.



VAR structure

Table 2.42 VAR Structural Elements

Element	Meaning		Possible Settings	
Top	4xxxxx	MC20 Module number: 1 or 2	Holding registers	400001 to 409981 (W00001 to W09981)
	4xxxxx+1	H1 set value – L	——	
	4xxxxx+2	H1 set value – H		
	Coordinate value: 0 to ±99,999,999			
	4xxxxx+3	H2 set value – L	——	
	4xxxxx+4	H2 set value – H		
	Coordinate value: 0 to ±99,999,999			
	4xxxxx+5	H3 set value – L	——	
	4xxxxx+6	H3 set value – H		
	Coordinate value: 0 to ±99,999,999			
	····· ·····		——	
	4xxxxx+15	H8 set value – L	——	
	4xxxxx+16	H8 set value – H		
	Coordinate value: 0 to ±99,999,999			
	4xxxxx+17	Used by system.	Do not write.	
	4xxxxx+18	Status		
Middle	None		——	
Bottom	Indicates the VAR instruction.		——	

Table 2.43 VAR Operation

Input 1	Status/Remarks	Output 1	Output 2	Output 3
OFF ↓ ON	● The H variable settings are executed at the leading edge of input 1. That is, the specified Hn set values are set for the H variables (H1 to H8). ● The output 3 (setting completion) ON pulse signal is output when the H variable setting has been properly completed.	ON ↓ OFF	OFF	OFF ↓ ON pulse

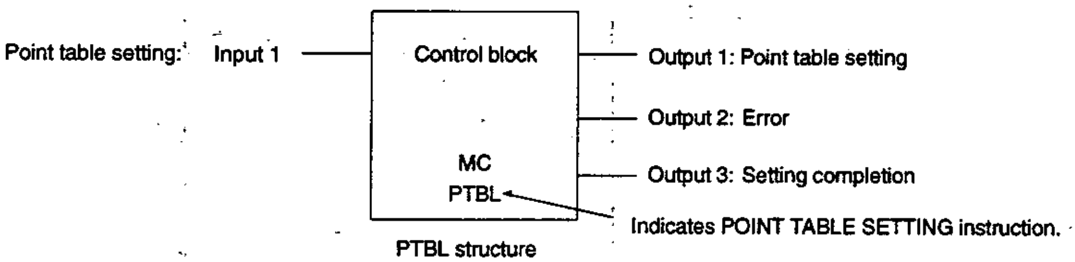
2.4.4 POINT TABLE SETTING (PTBL)

Function

PTBL designates one set of point table data for a specified point number. Point table data can be set by the #E□□□ format like a variable in an MC20 Module motion program.

Structure

1) The structure of the PTBL instruction is shown below.


Point table setting: Input 1
Control block
MC
PTBL
Output 1: Point table setting
Output 2: Error
Output 3: Setting completion
Indicates POINT TABLE SETTING instruction.

PTBL structure

2) The settings shown below are required for the control block at the top element. The leading reference number is specified for the top element. Refer to *Table 2.44 PTBL Structural Elements* for details on the ranges of reference numbers that can be specified.

Control Block

Leading reference number →

Holding registers	Contents of registers
4xxxxx	MC20 Module number: 1 or 2
4xxxxx+1	First register address: 1 to maximum register number – 7
4xxxxx+2	Point number: 1 to 500
4xxxxx+3	Used by system.
4xxxxx+4	Status (Refer to *2.1.5 Error Status.*)

3) For the first register address, specify the leading address of the eight holding registers where the data that is to be set to the point table is stored. Omit the "4" from the "4yyyyy" when specifying the leading address.

Control Block

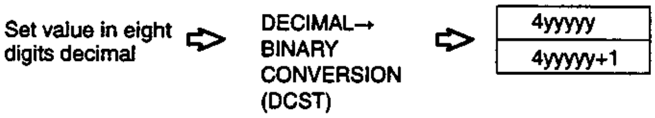
Leading reference number →

Data registers	Contents of registers	
4yyyyy	Axis 1 set value – L	0 to ±99,999,999
4yyyyy+1	Axis 1 set value – H	
4yyyyy+2	Axis 2 set value – L	0 to ±99,999,999
4yyyyy+3	Axis 2 set value – H	
4yyyyy+4	Axis 3 set value – L	0 to ±99,999,999
4yyyyy+5	Axis 3 set value – H	
4yyyyy+6	Axis 4 set value – L	0 to ±99,999,999
4yyyyy+7	Axis 4 set value – H	

4) The axis n set value is directly designated in **signed 32-bit binary**, using two registers.

Axis 1 set value – H	Axis 1 set value – L	
4yyyyy+1	4yyyyy	= signed 32-bit binary

DECIMAL→BINARY CONVERSION (DCST) instruction is convenient when carrying out direct designation of binary numbers.

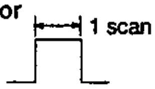
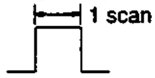

Set value in eight digits decimal
DECIMAL→ BINARY CONVERSION (DCST)
4yyyyy
4yyyyy+1

5) Specify a number from 1 to 500 as the point number for the point table. Set, as the axis data for that point number, the values stored in the eight holding registers beginning with the first register address explained above.

Note POINT TABLE SETTING (PTBL) designates to the specified point number the data for axes 1 to 4. This data is handled as a single set of data. It is not possible to designate separate data for individual axes.

Operation

1) The following table shows the basic operation of the inputs and outputs.

I/O Number	Operation Name	Description	Input Contacts
Input 1	Point table setting	The point table settings are executed when the leading edge of the OFF→ON signal for this input is detected. That is, the complete set of data beginning from the first register address is designated to the specified point number.	Positive transitional contact —| P |—
Output 1	Point table setting	This output is turned ON by the leading edge of the input 1 signal, and it is turned OFF when the point table setting operation is completed.	
Output 2	Error  ON pulse signal	When an error occurs, the ON pulse signal is output and the instruction is terminated. Output 3, however, is not turned ON. The pulse width is one scan. For details, refer to *2.1.5 Error Status.*	
Output 3	Setting completion  ON pulse signal	The output 3 ON pulse signal is output when the point table setting operation has been completed properly. The pulse width is one scan.	

TERMS

Signed 32-bit Binary

The usage of "binary" here is defined as follows:
Positive value: Normal 32-bit binary.
Negative value: Expressed as two's complement.

Note (a) Use the contact specified under "Input Contacts" to turn input 1 ON and OFF.

(b) The pulse width of the ON pulse signal is one scan.

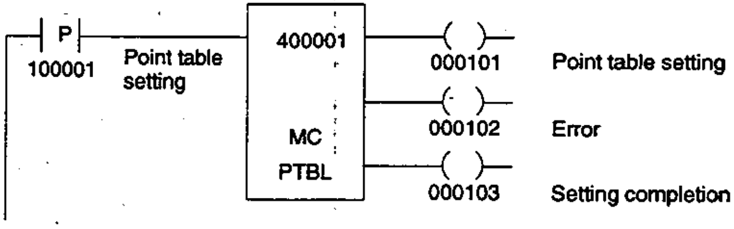
2) Values that are set or changed by POINT TABLE SETTING (PTBL) go into effect immediately.

3) Point table values set by POINT TABLE SETTING (PTBL) are retained, without being cleared, even when the following instructions are executed.

- MODULE RESET (MRS)
- MACHINE RESET (RST)
- EMERGENCY STOP NOTIFICATION (ESP)
- ALARM RESET (ARS)

◀**EXAMPLE**▶ In this example, the following data is set for point number 500 of the MC20 Module 1's point table: Axis 1 = 35000; axis 2 = 45000; axis 3 = 55000; axis 4 = 0.

Ladder Diagram



Control block	400001	1	MC20 Module number designation
	400002	101	First register address designation
	400003	500	Point number designation
	400004		Used by system.
	400005		Status (Refer to *2.1.5 Error Status.*)

Point table data	400101	Binary	Axis 1 set value – L
	400102	35000	Axis 1 set value – H
	400103	Binary	Axis 2 set value – L
	400104	45000	Axis 2 set value – H
	400105	Binary	Axis 3 set value – L
	400106	55000	Axis 3 set value – H
	400107	Binary	Axis 4 set value – L
	400108	0	Axis 4 set value – H

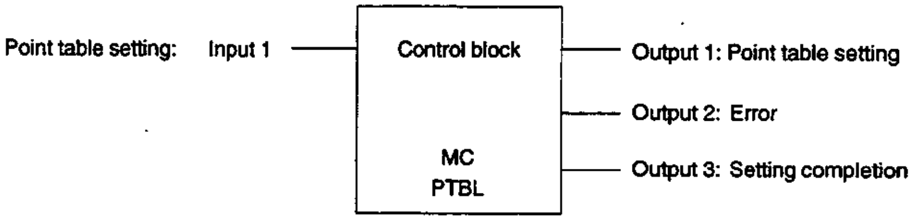
Operation

The above settings are made in control block holding registers 400101 to 400108.

When input 1 (100001) is turned ON, the values shown for axis n above are set for point number 500 of the MC20 Module's point table.

Note Multiple POINT TABLE SETTING (PTBL) instructions can be written into a ladder logic program for a single MC20 Module. It is not possible, however, to execute two or more instructions at the same time.

SUMMARY The following information summarizes the instruction.



PTBL structure

Table 2.44 PTBL Structural Elements

Element	Meaning		Possible Settings	
Top	4xxxxx	MC20 Module number: 1 or 2	Holding registers	400001 to 409995 (W00001 to W09995)
	4xxxxx+1	First register address: 1 to (maximum register address – 7) (The values in the eight holding registers beginning with this address are set in the point table.)	——	
	4xxxxx+2	Point number: 1 to 500	——	
	4xxxxx+3	Used by system.	Do not write.	
	4xxxxx+4	Status		
Middle	None		——	
Bottom	Indicates the PTBL instruction.		——	

Table 2.45 PTBL Operation

Input 1	Status/Remarks	Output 1	Output 2	Output 3
OFF ↓ ON	• The point table settings are executed at the leading edge of input 1. That is, the complete set of data beginning from the first register address is designated to the specified point number. • The output 3 (setting completion) ON pulse signal is output when the point table setting has been properly completed.	ON ↓ OFF	OFF	OFF ↓ ON pulse

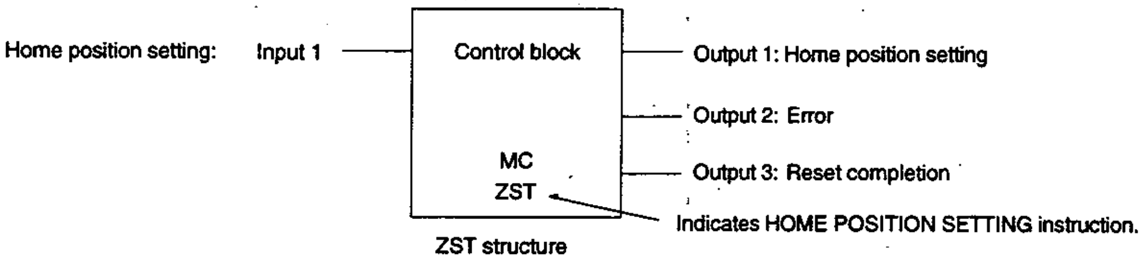
2.4.5 HOME POSITION SETTING (ZST)

Function

ZST sets the home position in an absolute position detecting system.

Structure

1) The structure of the ZST instruction is shown below.


Home position setting:
Input 1
Control block
MC
ZST
Output 1: Home position setting
Output 2: Error
Output 3: Reset completion
Indicates HOME POSITION SETTING instruction.

ZST structure

2) The settings shown below are required for the control block in the top element. The leading reference number is specified for the top element. Refer to *Table 2.46 ZST Structural Elements* for details on the ranges of reference numbers that can be specified.

Control Block

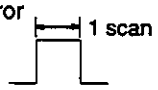
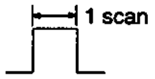
Holding registers	Contents of registers
4xxxxx (Leading reference number)	MC20 Module number: 1 or 2
4xxxxx+1	Controlled axis designation: 1, 2, 4, 8 or sum
4xxxxx+2	Used by system.
4xxxxx+3	Status (Refer to *2.1.5 Error Status.*)

3) To designate the controlled axes, specify in holding register 4xxxxx+1 the numbers corresponding to the axes that are to have the home position set. The numbers are shown in the following table. To designate multiple axes with a single instruction, specify the sum of the desired axes. For example, if the desired axes are 1, 2, 3, and 4, the corresponding numbers would be 1, 2, 4, and 8. Therefore the sum would be 15 (i.e., 1 + 2 + 4 + 8 = 15). The home position will not be set for any axes that are not designated here.

Axis Number	Corresponding Number
Axis 1	1
Axis 2	2
Axis 3	4
Axis 4	8

Operation

1) The following table shows the basic operation of the inputs and outputs.

I/O Number	Operation Name	Description	Input Contacts
Input 1	Home position setting	The home position is set when the leading edge of the OFF→ON signal for this input is detected. That is, the machine coordinate zero point is set for the absolute position detecting system of the specified MC20 Module.	Positive transitional contact ─┤P├─
Output 1	Home position setting	This output is turned ON by the leading edge of the input 1 signal, and it is turned OFF when the home position setting operation is completed, i.e., when MC control relay ZREx (axis x home position setting completion) is turned ON.	
Output 2	Error  ON pulse signal	When an error occurs, the ON pulse signal is output and the instruction is terminated. Output 3, however, is not turned ON. The pulse width is one scan. For details, refer to *2.1.5 Error Status.*	
Output 3	Reset completion  ON pulse signal	The output 3 ON pulse signal is output when the home position setting operation has been completed properly. The pulse width is one scan.	

Note (a) Use the contact specified under "Input Contacts" to turn input 1 ON and OFF.

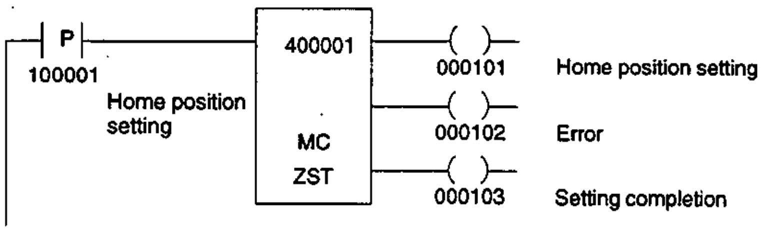
(b) The pulse width of the ON pulse signal is one scan.

2) MC control relay ZREx (axis x home position setting completion), which turns ON when the home position setting operation is finished, turns OFF again the next time HOME POSITION SETTING (ZST) is started to execute.

⚠**Caution** This command is used to set the machine coordinate home position of an absolute positioning system. Before executing this command, be absolutely sure to check the machine coordinate home position to make sure that the designation of the machine coordinate home position is correct. Failure to do this may result in damage to equipment, serious personal injury, or even death.

◀EXAMPLE▶ In this example, the home position is set for axes 1, 2, 3, and 4 of MC20 Module 1.

Ladder Diagram



400001	1	MC20 Module number: 1 or 2
400002	15	Controlled axis designation: 1 + 2 + 4 + 8
400003		Used by system.
400004		Status (Refer to *2.1.5 Error Status.*)

Operation

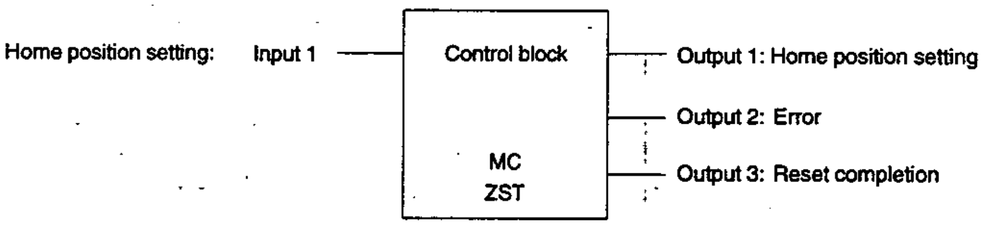
The above settings are made in holding registers 400001 and 400002.

When input 1 (100001) is turned ON, the home position is set for all four axes of the MC20 Module. That is, the machine coordinate zero point is set for the absolute position detecting system.

Note (a) Multiple HOME POSITION SETTING (ZST) instructions can be written into a ladder logic program for a single MC20 Module. It is not possible, however, to execute two or more instructions at the same time.

(b) An MC busy error will occur if HOME POSITION SETTING (ZST) is executed for an MC Module for which parameter PA402 is set to the incremental system.

SUMMARY The following information summarizes the instruction.



ZST structure

Table 2.46 ZST Structural Elements

Element	Meaning		Possible Settings	
Top	4xxxxx	MC20 Module number: 1 or 2	Holding registers	400001 to 409996 (W00001 to W09996)
	4xxxxx+1	Axis designation: 1, 2, 4, 8, or sum (multiple axis designation)	——	
	4xxxxx+2	Used by system.	Do not write.	
	4xxxxx+3	Status		
Middle	None		——	
Bottom	Indicates the ZST instruction.		——	

Table 2.47 ZST Operation

Input 1	Status/Remarks	Output 1	Output 2	Output 3
OFF ↓ ON	• The home position set at the leading edge of input 1. That is, the machine coordinate zero point is set for the absolute position detecting system of the specified MC20 Module. • MC control relay ZMDL (home position setting mode status) remains ON during the home position setting operation. • The output 3 (reset completion) ON pulse signal is output when the home position setting has been properly completed, i.e., when MC control relay ZREx (axis x home position setting completion) is turned ON.	ON ↓ OFF	OFF	OFF ↓ ON pulse

Note MC control relay ZREx (axis x home position setting completion), which turns ON when the home position setting operation is finished, turns OFF again the next time HOME POSITION SETTING (ZST) is started to execute.

Using MC Control Coils and Relays

3

This chapter explains how to use the specific coils and relays that transmit signals between the MC20 Module and the CPU Module. Basic signals are explained in this chapter, so be sure to read it before attempting operation.

3.1 Outline

This section provides a general outline of the contents of this chapter.

1) MC control coils are specific signals with fixed allocations that serve to control the MC Module from the CPU Module.

2) MC control coil reference numbers are assigned as follows:

MC20 Module 1:	Q10001 to Q10160
MC20 Module 2:	Q20001 to Q20160

3) A chart of MC control coils is provided in item *1)* of *3.2 MC Control Coil Functions.*

4) The names and functions of all MC control coils are shown in item *2)* of *3.2 MC Control Coil Functions.*

5) MC control relays are specific signals with fixed allocations that transmit the MC20 Module's status to the CPU Module.

6) MC control relay reference numbers are assigned as follows:

MC20 Module 1:	P10001 to P01256
MC20 Module 2:	P20001 to P20256

7) A chart of MC control relays is provided in item *1)* of *3.3 MC Control Relay Functions.*

8) The names and functions of all MC control relays are shown in item *2)* of *3.3 MC Control Relay Functions.*

3.2 MC Control Coil Functions

1) The following chart shows the MC control coils and their reference numbers.

MF01: MF IN coil
to
MF96:

QN0008	QN0007	QN0006	QN0005	QN0004	QN0003	QN0002	QN0001
MF08	MF07	MF06	MF05	MF04	MF03	MF02	MF01
QN0016	QN0015	QN0014	QN0013	QN0012	QN0011	QN0010	QN0009
MF16	MF15	MF14	MF13	M12	MF11	MF10	MF09
QN0024	QN0023	QN0022	QN0021	QN0020	QN0019	QN0018	QN0017
MF24	0F23	MF22	0F21	MF20	MF19	MF18	MF17
QN0032	QN0031	QN0030	QN0029	QN0028	QN0027	QN0026	QN0025
MF32	MF31	MF30	MF29	MF28	MF27	MF26	MF25
QN0040	QN0039	QN0038	QN0037	QN0036	QN0035	QN0034	QN0033
MF40	MF39	MF38	MF37	MF36	MF35	MF34	MF33
QN0048	QN0047	QN0046	QN0045	QN0044	QN0043	QN0042	QN0041
MF48	MF47	MF46	MF45	MF44	MF43	MF42	MF41
QN0056	QN0055	QN0054	QN0053	QN0052	QN0051	QN0050	QN0049
MF56	MF55	MF54	MF53	MF52	MF51	MF50	MF49
QN0064	QN0063	QN0062	QN0061	QN0060	QN0059	QN0058	QN0057
MF64	MF63	MF62	MF61	MF60	MF589	MG58	MF57
QN0072	QN0071	QN0070	QN0069	QN0068	QN0067	QN0066	QN0065
MF72	MF71	MF70	MF69	MF68	MF67	MF66	MF65
QN0080	QN0079	QN0078	QN0077	QN0076	QN0075	QN0074	QN0073
MF80	MF79	MF78	MF77	MF76	MF75	MF74	MF73
QN0088	QN0087	QN0086	QN0085	QN0084	QN0083	QN0082	QN0081
MF88	MF87	MF86	MF85	MF84	MF83	MF82	MF81
QN0096	QN0095	QN0094	QN0093	QN0092	QN0091	QN0090	QN0089
MF96	MF95	MF94	MF93	MF92	MF91	MF90	MF89

Note To obtain the actual reference numbers, replace N with 1 for MC20 Module 1 and replace N with 2 for MC20 Module 2.

Ver. B08

ALMDISx: Axis alarm disable function effective for axis x

PCNx: PCON command for axis x

FBREQ: Feedback monitor
VELREQ: Speed monitor
ERRREQ: Deviation monitor

QN0104	QN0103	QN0102	QN0101	QN0100	QN0099	QN0098	QN0097
				ALMDIS4	ALMDIS3	ALMDIS2	ALMDIS1

QN0112	QN0111	QN0110	QN0109	QN0108	QN0107	QN0106	QN0105
ERRREQ	VELREQ	FBREQ		PCN4	PCN3	PCN2	PCN1

Ver. B08

Not used

QN0120	QN0119	QN0118	QN0117	QN0116	QN0115	QN0114	QN0113

QN0128	QN0127	QN0126	QN0125	QN0124	QN0123	QN0122	QN0121

QN0136	QN0135	QN0134	QN0133	QN0132	QN0131	QN0130	QN0129

ROV0 to ROV3: Rapid traverse speed override

FOV0 to FOV3: Interpolation speed override

OV10 to OV43: Override for independent axis and manual operation

QN0144	QN0143	QN0142	QN0141	QN0140	QN0139	QN0138	QN0137
FOV3	FOV2	FOV1	FOV0	ROV3	ROV2	ROV1	ROV0

QN0152	QN0151	QN0150	QN0149	QN0148	QN0147	QN0146	QN0145
OV23	OV22	OV21	OV20	OV13	OV12	OV11	OV10

QN0160	QN0159	QN0158	QN0157	QN0156	QN0155	QN0154	QN0153
OV43	OV42	OV41	OV40	OV33	OV32	OV31	OV30

Note To obtain the actual reference numbers, replace N with 1 for MC20 Module 1 and replace N with 2 for MC20 Module 2.

2) The following table shows the names and functions of the MC control coils. The following abbreviations are used for the modes: O: On-line edit; A: Automatic; M: Manual; E: Edit. In place of the “N” in the reference number, enter “1” for MC20 Module 1 or “2” for MC Module 2. “O” indicates valid modes; “X” indicated invalid modes.

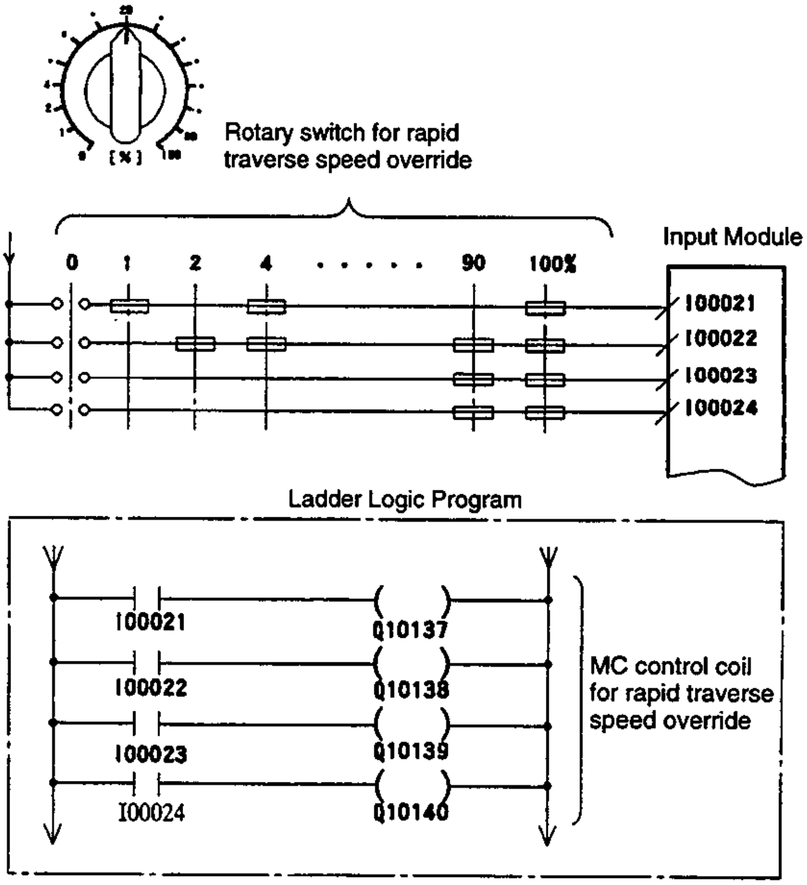
<table>
<tr><th rowspan="2">Reference</th><th rowspan="2">Signal</th><th rowspan="2">Name</th><th colspan="4">Mode</th><th rowspan="2">Function and Timing</th></tr>
<tr><th>O</th><th>A</th><th>M</th><th>E</th></tr>
<tr><td>QN0001 to QN0096</td><td>MF01 to MF96</td><td>MFIN coil</td><td>O</td><td>O</td><td>O</td><td>O</td><td>Execute work in the ladder program according to the ON timing of the M code relay numbers specified by the PASS NOTCH SIGNAL OUTPUT (PNT) or SET EXTERNAL OUTPUT (SET) command. After the work is completed, the MFIN coil with the number corresponding to the M code relay number is turned ON, then the M code relay is turned OFF and the motion program in the MC20 Module proceeds to the next block.</td></tr>
<tr><td>QN0097</td><td>ALMDIS-1</td><td>Axis 1 alarm disable</td><td rowspan="4">O</td><td rowspan="4">O</td><td rowspan="4">O</td><td rowspan="4">O</td><td rowspan="4">If one of the coils is turned ON when bit 1 of parameter P0017 (axis alarm disable usage), the Servo can be turned ON and axis movements will be possible even when an axis alarm occurs for another axis.</td></tr>
<tr><td>QN0098</td><td>ALMDIS-2</td><td>Axis 2 alarm disable</td></tr>
<tr><td>QN0099</td><td>ALMDIS-3</td><td>Axis 3 alarm disable</td></tr>
<tr><td>QN0100</td><td>ALMDIS-4</td><td>Axis 4 alarm disable</td></tr>
<tr><td>QN0105</td><td>PCN1</td><td>Axis 1 PCON command</td><td rowspan="4">X</td><td rowspan="4">X</td><td rowspan="4">O</td><td rowspan="4">O</td><td rowspan="4">These commands turn ON and OFF the PCON□ signal output from SVn of the axis servo connectors. 0 = OFF, 1 = ON.
These commands are effective at the following times:
The control axis is a basic axis in editing or manual mode.
The control axis is an independent axis.</td></tr>
<tr><td>QN0106</td><td>PCN2</td><td>Axis 2 PCON command</td></tr>
<tr><td>QN0107</td><td>PCN3</td><td>Axis 3 PCON command</td></tr>
<tr><td>QN0108</td><td>PCN4</td><td>Axis 4 PCON command</td></tr>
<tr><td>QN0110</td><td>FBREQ</td><td>Feedback monitor</td><td rowspan="3">O</td><td rowspan="3">O</td><td rowspan="3">O</td><td rowspan="3">O</td><td rowspan="3">These coils are used to set the type of current position that the MC20 Module sends to the CPU Module.
<table>
<tr><th colspan="3">QNO</th><th rowspan="2">Current Position</th><th rowspan="2">Unit</th></tr>
<tr><th>112</th><th>111</th><th>110</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>Reference current position</td><td>Reference unit</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>Feedback current position</td><td>Reference unit</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>Reference speed</td><td>mm/min or deg/min</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>Feedback speed</td><td>mm/min or deg/min</td></tr>
<tr><td>1</td><td>0 or 1</td><td>0 or 1</td><td>Deviation</td><td>pulses</td></tr>
</table></td></tr>
<tr><td>QN0111</td><td>VELREQ</td><td>Speed monitor</td></tr>
<tr><td>QN0112</td><td>ERRREQ</td><td>Deviation monitor</td></tr>
</table>

<table>
<tr><th rowspan="2">Reference</th><th rowspan="2">Signal</th><th rowspan="2">Name</th><th colspan="4">Mode</th><th rowspan="2">Function and Timing</th></tr>
<tr><th>O</th><th>A</th><th>M</th><th>E</th></tr>
<tr><td>QN0137
QN0138
QN0139
QN0140</td><td>ROV0
ROV1
ROV2
ROV3</td><td>Rapid traverse speed overrides</td><td>O</td><td>O</td><td>X</td><td>X</td><td>These coils are used to change the rapid traverse speed override (%) in the program operation mode when PROGRAM RUN (MVL) is executed. The change can be made while the axes are travelling.
<table>
<tr><th colspan="4">QN0</th><th rowspan="2">Override (%)</th><th colspan="4">QN0</th><th rowspan="2">Override (%)</th></tr>
<tr><th>140</th><th>139</th><th>138</th><th>137</th><th>140</th><th>139</th><th>138</th><th>137</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>30</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>40</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>2</td><td>1</td><td>0</td><td>1</td><td>0</td><td>50</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td><td>4</td><td>1</td><td>0</td><td>1</td><td>1</td><td>60</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>6</td><td>1</td><td>1</td><td>0</td><td>0</td><td>70</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>8</td><td>1</td><td>1</td><td>0</td><td>1</td><td>80</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>10</td><td>1</td><td>1</td><td>1</td><td>0</td><td>90</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td><td>20</td><td>1</td><td>1</td><td>1</td><td>1</td><td>100</td></tr>
</table></td></tr>
<tr><td>QN0141
QN0142
QN0143
QN0144</td><td>FOV0
FOV1
FOV2
FOV3</td><td>Interpolation speed override</td><td>O</td><td>O</td><td>X</td><td>X</td><td>These coils are used to change the interpolation speed override (%) in the program operation mode when PROGRAM RUN (MVL) is executed. The change can be made while the axes are travelling.
<table>
<tr><th colspan="4">QN0</th><th rowspan="2">Override (%)</th><th colspan="4">QN0</th><th rowspan="2">Override (%)</th></tr>
<tr><th>144</th><th>143</th><th>142</th><th>141</th><th>144</th><th>143</th><th>142</th><th>141</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>60</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>80</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>2</td><td>1</td><td>0</td><td>1</td><td>0</td><td>100</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>1</td><td>4</td><td>1</td><td>0</td><td>1</td><td>1</td><td>120</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>8</td><td>1</td><td>1</td><td>0</td><td>0</td><td>140</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>10</td><td>1</td><td>1</td><td>0</td><td>1</td><td>160</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>20</td><td>1</td><td>1</td><td>1</td><td>0</td><td>180</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>1</td><td>40</td><td>1</td><td>1</td><td>1</td><td>1</td><td>200</td></tr>
</table></td></tr>
</table>

Reference	Signal	Name	Mode O	Mode A	Mode M	Mode E	Function and Timing
QN0145	OV10	Axis 1 independent axis and manual override	O	O	O	O	These coils are used to change the traverse speed override (%) during independent axis operation or JOG/STEP operation in manual mode. The change can be made while the axes are travelling. The following data applies to axis 1. Axes 2 through 4 are the same except for different reference numbers.
QN0146	OV11						
QN0147	OV12						
QN0148	OV13						
QN0149	OV20	Axis 2 independent axis and manual override					
QN0150	OV21						
QN0151	OV22						
QN0152	OV23						
QN0153	OV30	Axis 3 independent axis and manual override					
QN0154	OV31						
QN0155	OV32						
QN0156	OV33						
QN0157	OV40	Axis 4 independent axis and manual override					
QN0158	OV41						
QN0159	OV42						
QN0160	OV43						

QN0				Override (%)	QN0				Override (%)
148	147	146	145		148	147	146	145	
0	0	0	0	0	1	0	0	0	30
0	0	0	1	1	1	0	0	1	40
0	0	1	0	2	1	0	1	0	50
0	0	1	1	4	1	0	1	1	60
0	1	0	0	6	1	1	0	0	70
0	1	0	1	8	1	1	0	1	80
0	1	1	0	10	1	1	1	0	90
0	1	1	1	20	1	1	1	1	100

The following illustration is an example using MC control coils ZN0137 to QN0140 for rapid traverse speed override.



3.3 MC Control Relay Functions

1) The following chart shows the MC control relays and their reference numbers.

MCRD: MC20 Module ready
ALRM: MC20 Module alarm

ESPL: Emergency stop

PN0008	PN0007	PN0006	PN0005	PN0004	PN0003	PN0002	PN0001
ESPL						ALRM	MCRD

MANL: Manual mode
MEML: Automatic mode
OEDL: On-line edit mode
TBXL: Teach Pendant mode

RSTL: Machine resetting
PRSL: Program reset completion
ARSL: Alarm reset completion

PN0016	PN0015	PN0014	PN0013	PN0012	PN0011	PN0010	PN0009
ARSL	PRSL	RSTL		TBXL	OEDL	MEML	MANL

STL: Program running
SPL: Feed holding
ENDL: Program completion
MFIR: M code sampling
MLKL: Machine lock mode
SBKL: Single-block mode

SBKS: Single-block stop
MDEN: Travelling completion

PN0024	PN0023	PN0022	PN0021	PN0020	PN0019	PN0018	PN0017
MDEN	SBKS	SBKL	MLKL	MFIR	ENDL	SPL	STL

SVN$_x$: Axis x servo ON status
MLK$_x$S: Axis x machine lock stop

PN0032	PN0031	PN0030	PN0029	PN0028	PN0027	PN0026	PN0025
MLK4S	MLK3S	MLK2S	MLK1S	SVN4	SVN3	SVN2	SVN1

MOV$_x$: Axis x travelling
MOV$_A$: A-axis travelling

PN0040	PN0039	PN0038	PN0037	PN0036	PN0035	PN0034	PN0033
MOVD	MOVC	MOVB	MOVA	MOV4	MOV3	MOV2	MOV1

JOG$_x$L: Axis x jogging
STP$_x$L: Axis x stepping

PN0048	PN0047	PN0046	PN0045	PN0044	PN0043	PN0042	PN0041
STP4L	STP3L	STP2L	STP1L	JOG4L	JOG3L	JOG2L	JOG1L

ZRN$_x$L: Axis x home position returning

PN0056	PN0055	PN0054	PN0053	PN0052	PN0051	PN0050	PN0049
				ZRN4L	ZRN3L	ZRN2L	ZRN1L

x = Axis number 1, 2, 3, or 4
A = Independent axis number A, B, C, or D

Note To obtain the actual reference numbers, replace N with 1 for MC20 Module 1 and replace N with 2 for MC20 Module 2.

A Independent axis number A, B, C, or D

ST$_A$L: A-axis operating
SP$_A$L: A-axis holding
END$_A$L: A-axis operation completion
ENBL$_A$: A-axis effective

PN0064	PN0063	PN0062	PN0061	PN0060	PN0059	PN0058	PN0057
ENBLB	ENDBL	SPBL	STBL	ENBLA	ENDAL	SPAL	STAL

PN0072	PN0071	PN0070	PN0069	PN0068	PN0067	PN0066	PN0065
ENBLD	ENDDL	SPDL	STDL	ENBLC	ENDCL	SPCL	STCL

ZPT$_x$: Axis x home position
ALM$_x$: Axis x alarm

PN0080	PN0079	PN0078	PN0077	PN0076	PN0075	PN0074	PN0073
ALM4	ALM3	ALM2	ALM1	ZPT4	ZPT3	ZPT2	ZPT1

BRK$_x$: Axis x brake signal

PN0088	PN0087	PN0086	PN0085	PN0084	PN0083	PN0082	PN0081
				BRK4	BRK3	BRK2	BRK1

PCNLx: Axis x PCON output status

Ver. B08

FBMON: Feedback monitoring
VELMON: Speed monitoring
ERRMON: Deviation monitoring

PN0096	PN0095	PN0094	PN0093	PN0092	PN0091	PN0090	PN0089
ERRMON	VELMON	FBMON		PCNL4	PCNL3	PCNL2	PCNL1

Direct input status
POT$_x$: Axis x overtravel (+)
NOT$_x$: Axis x overtravel (–)
DEC$_x$: Axis x deceleration dog
ZER$_x$: Axis x home position pulse

PN0104	PN0103	PN0102	PN0101	PN0100	PN0099	PN0098	PN0097
ZER2	DEC2	NOT2	POT2	ZER1	DEC1	NOT1	POT1

PN0112	PN0111	PN0110	PN0109	PN0108	PN0107	PN0106	PN0105
ZER4	DEC4	NOT4	POT4	ZER3	DEC3	NOT3	POT3

Home position setting status
ZRE$_x$: Axis x home position setting completion
ZMDL: Home position setting mode status

PN0120	PN0119	PN0118	PN0117	PN0116	PN0115	PN0114	PN0113
ZMDL				ZRE4	ZRE3	ZRE2	ZRE1

ERST: External reset
MBLD: Multi-block operating
EXPx: Axis x external input signal status

PN0128	PN0127	PN0126	PN0125	PN0124	PN0123	PN0122	PN0121
EXP4	EXP3	EXP2	EXP1			MBDL	ERST

Not used.

PN0136	PN0135	PN0134	PN0133	PN0132	PN0131	PN0130	PN0129

~

PN0200	PN0199	PN0198	PN0197	PN0196	PN0195	PN0194	PN0193

Used by system.
(Other use prohibited.)

PN0208	PN0207	PN0206	PN0205	PN0204	PN0203	PN0202	PN0201

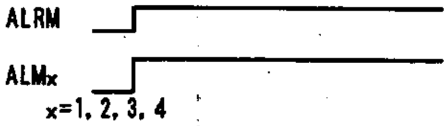
~

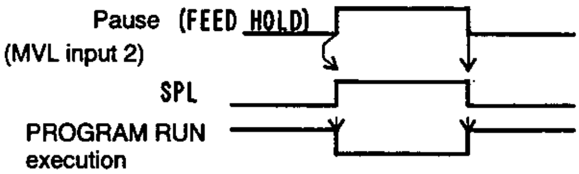
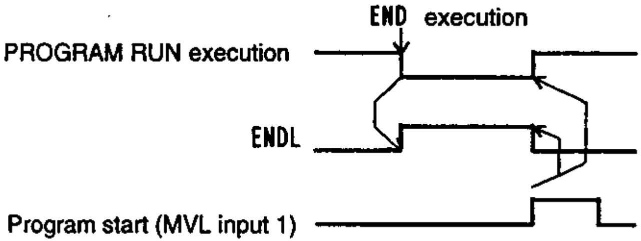
PN0256	PN0255	PN0254	PN0253	PN0252	PN0251	PN0250	PN0249

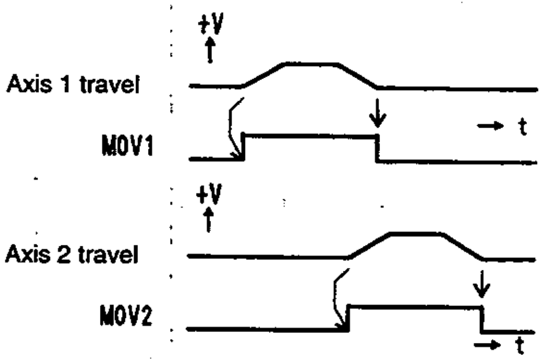
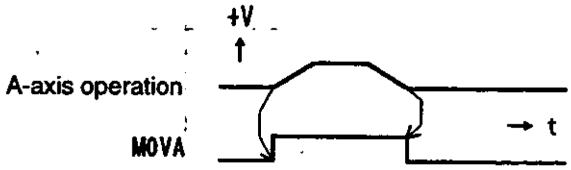
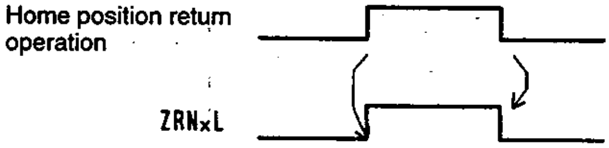
x = Axis number 1, 2, 3, or 4
A = Independent axis number A, B, C, or D

Note To obtain the actual reference numbers, replace N with 1 for MC20 Module 1 and replace N with 2 for MC20 Module 2.

2) The following table shows the names and functions of the MC control relays. The following abbreviations are used for the modes: O: On-line edit; A: Automatic; M: Manual; E: Edit. In place of the "N" in the reference number, enter either "1" for MC20 Module 1 or "2" for MC20 Module 2. "O" indicates valid modes; "X" indicated invalid modes.

Reference	Signal	Name	Mode				Function and Timing
			O	A	M	E	
PN0001	MCRD	MC20 Module ready	O	O	O	O	Indicates that the MC20 Module is ready. It is the same as the system information "MC ready."
PN0002	ALRM	MC20 Module alarm	O	O	O	O	Indicates that the MC Module is in alarm status. When an alarm occurs, the alarm code is set and then the alarm output is turned ON. When the operation is executed to clear all of the alarms in the history, the code is set to "0" and the alarm output is turned OFF.
PN0077	ALM1	Axis 1 alarm	O	O	O	O	 ALMx is turned ON when an alarm occurs at its corresponding axis. When only ALRM is ON, it indicates that a common alarm has occurred.
PN0087	ALM2	Axis 2 alarm					
PN0079	ALM3	Axis 3 alarm					
PN0080	ALM4	Axis 4 alarm					
PN0008	ESPL	Emergency stop	O	O	O	O	Indicates that EMERGENCY STOP NOTIFICATION (ESP) has been executed, and that an emergency stop is in effect.
PN0009	MANL	Manual mode	O	O	O	O	Indicates the MC20 Module's current operation mode.
PN0010	MEML	Automatic mode					
PN0011	OEDL	On-line edit mode					
PN0012	TBXL	Teach Pendant mode					
PN0014	RSTL	Machine resetting	O	O	O	O	Indicates that MACHINE RESET (RST) has been executed and that the MC20 Module is being reset. The following operations are executed. • The alarm and alarm history are cleared. • The MC coils are reset. • Changes the parameters to the last set values.
PN0015	PRSL	Program reset completion	X	O	X	X	Indicates that the initialization for program running has been completed. The program can be run from the O number and B number specified by PROGRAM RUN (MVL).
PN0016	ARSL	Alarm reset completion	O	O	O	O	Indicates that the MC20 Module's alarm reset has been completed.

Reference	Signal	Name	Mode				Function and Timing
			O	A	M	E	
PN0017	STL	Program running	O	O	X	X	Indicates that the program is running. The ON and OFF conditions are as follows: (a) ON conditions • PROGRAM RUN (MVL) is being executed. • A block commanded from the programming device is being executed. (b) OFF conditions • PROGRAM END (END) is executed. • The execution of a block is completed. • An alarm occurs. • MACHINE RESET (RST) is executed.
PN0018	SPL	Feed holding	X	O	X	X	Indicates that feed hold is in effect during program running.  Pause (FEED HOLD) (MVL input 2) SPL PROGRAM RUN execution
PN0019	ENDL	Program completion	O	O	X	X	Turns ON when PROGRAM END (END) is executed while the program is running, and turns OFF upon restarting.  END execution PROGRAM RUN execution ENDL Program start (MVL input 1)
PN0020	MFIR	M code sampling	O	O	X	X	This signal requests that the CPU Module read the M code output. The M code is output by means of either SET EXTERNAL OUTPUT (SET) or PASS NOTCH SIGNAL OUTPUT (PNT).
PN0021	MLKL	Machine lock mode	O	O	O	O	Indicates that the machine lock mode is in effect.
		Machine lock stop:	O	O	O	O	MLKxL turns ON when an axis that is being operated is stopped by means of switching to the machine lock mode. It turns OFF when the machine lock mode is cleared and program run is restarted.
PN0029	MLK1S	Axis 1					
PN0030	MLK2S	Axis 2					
PN0031	MLK3S	Axis 3					
PN0032	MLK4S	Axis 4					
PN0022	SBKL	Single-block mode	X	O	X	X	Indicates that the mode which stops program running after completion of every block is in effect.
PN0023	SBKS	Single-block stop	X	O	X	X	Indicates that the program is stopped in single-block mode. This relay also turns ON when PROGRAM STOP (STP) is executed while the program is running.

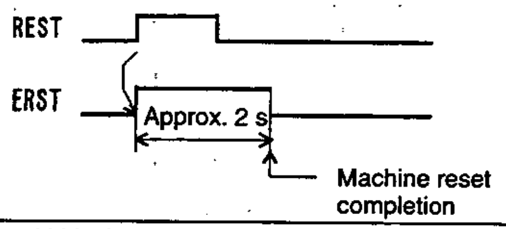
Reference	Signal	Name	Mode				Function and Timing
			O	A	M	E	
PN0024	MDEN	Travelling completion	O	O	X	X	Indicates that the pulse distribution of move command has been completed while the program is running.
PN0025	SVN1	Axis 1 servo ON status	O	O	O	O	Indicates that the controlled axis is in servo ON status.
PN0026	SVN2	Axis 2 servo ON status					
PN0027	SVN3	Axis 3 servo ON status					
PN0028	SVN4	Axis 4 servo ON status					
PN0033	MOV1	Axis 1 travelling	O	O	O	X	Indicates that the controlled axis is travelling. This signal is in effect regardless of program execution, manual operation, or mode.  +V Axis 1 travel MOV1 t +V Axis 2 travel MOV2 t
PN0034	MOV2	Axis 2 travelling					
PN0035	MOV3	Axis 3 travelling					
PN0036	MOV4	Axis 4 travelling					
PN0037	MOVA	Axis A travelling	O	O	O	O	Indicates that any of controlled axes A to D are travelling. 
PN0038	MOVB	Axis B travelling					
PN0039	MOVC	Axis C travelling					
PN0040	MOVD	Axis D travelling					
PN0041	JOG1L	Axis 1 jogging	X	X	O	X	Indicates that the respective axis is jogging.
PN0042	JOG2L	Axis 2 jogging					
PN0043	JOG3L	Axis 3 jogging					
PN0044	JOG4L	Axis 4 jogging					
PN0045	STP1L	Axis 1 stepping	X	X	O	X	Indicates that the respective axis is stepping.
PN0046	STP2L	Axis 2 stepping					
PN0047	STP3L	Axis 3 stepping					
PN0048	STP4L	Axis 4 stepping					
PN0049	ZRN1L	Axis 1 home position returning	O	O	O	X	Indicates that the home position return operation is being executed. This signal turns ON when either of the two HOME RETURN commands is executed (i.e., either the MC20 Module motion command or the CPU Module ladder motion command). Example: Operation based on ladder motion command 
PN0050	ZRN2L	Axis 2 home position returning					
PN0051	ZRN3L	Axis 3 home position returning					
PN0052	ZRN4L	Axis 4 home position returning					

Reference	Signal	Name	Mode				Function and Timing
			O	A	M	E	
PN0057	STAL	A-axis operating	O	O	O	O	**MC Control Relays for Independent Axes** ● A to D-axis operating: Indicates that the respective axis operation command is being executed. (This signal remains ON even while operation is paused.) ● A to D-axis holding: Indicates that operation of the respective axis is temporarily stopped. ● A to D-axis operation completion: Indicates that operation of the respective axis has been completed. ● A to D-axis effective: Indicates that the independent axis is in effect as a controlled axis. (This depends on a parameter setting.)
PN0058	SPAL	A-axis holding					
PN0059	ENDAL	A-axis operation completion					
PN0060	ENBLA	A-axis effective					
PN0061	STBL	B-axis operating					
PN0062	SPBL	B-axis holding					
PN0063	ENDBL	B-axis operation completion					
PN0064	ENBLB	B-axis effective					
PN0065	STCL	C-axis operating					
PN0066	SPCL	C-axis holding					
PN0067	ENDCL	C-axis operation completion					
PN0068	ENBLC	C-axis effective					
PN0069	STDL	D-axis operating					
PN0070	SPDL	D-axis holding					
PN0071	ENDDL	D-axis operation completion					
PN0072	ENBLD	D-axis effective					
PN0073	ZPT1	Axis 1 home position	O	O	O	O	Indicates that the current position for the respective axis is at the home position (i.e., the machine coordinate zero point ±PA307 range). In an incremental position detecting system, however, this signal is not output until the first HOME RETURN (ZRN) command is completed after the power has been turned ON.
PN0074	ZPT2	Axis 2 home position					
PN0075	ZPT3	Axis 3 home position					
PN0076	ZPT4	Axis 4 home position					
PN0081	BRK1	Axis 1 brake signal	O	O	O	O	Indicates the ON/OFF status of the brake output. 0: OFF 1: ON
PN0082	BRK2	Axis 2 brake signal					
PN0083	BRK3	Axis 3 brake signal					
PN0084	BRK4	Axis 4 brake signal					
PN0089	PCNL1	Axis 1 PCON output status	O	O	O	O	Indicates the PCON□ signal output status from SVn of the axis servo connectors. 0 = OFF, 1 = ON. The final output status is indicated regardless of whether the control axis is a basic axis or independent axis and regardless of the operating mode.
PN0090	PCNL2	Axis 2 PCON output status					
PN0091	PCNL3	Axis 3 PCON output status					
PN0092	PCNL4	Axis 4 PCON output status					

Ver. B08

<table>
<tr><th rowspan="2">Reference</th><th rowspan="2">Signal</th><th rowspan="2">Name</th><th colspan="4">Mode</th><th colspan="5" rowspan="2">Function and Timing</th></tr>
<tr><th>O</th><th>A</th><th>M</th><th>E</th></tr>
<tr><td rowspan="8">PN0094
PN0095
PN0095</td><td rowspan="8">FBMON
VELMON
ERRMON</td><td rowspan="8">Feedback monitoring
Speed monitoring
Deviation monitoring</td><td rowspan="8">O</td><td rowspan="8">O</td><td rowspan="8">O</td><td rowspan="8">O</td><td colspan="5">Indicate the type of current position that the MC20 Module sends to the CPU Module.</td></tr>
<tr><td colspan="3">PN0</td><td rowspan="2">Current Position</td><td rowspan="2">Unit</td></tr>
<tr><td>096</td><td>095</td><td>094</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>Reference current position</td><td>Reference unit</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>Feedback current position</td><td>Reference unit</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>Reference speed</td><td>mm/min or deg/min</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>Feedback speed</td><td>mm/min or deg/min</td></tr>
<tr><td>1</td><td>0 or 1</td><td>0 or 1</td><td>Deviation</td><td>pulses</td></tr>
</table>

Reference	Signal	Name	Mode				Function and Timing
			O	A	M	E	
PN0097	POT1	Axis 1 overtravel (+)	O	O	O	O	Indicates the ON/OFF status of the MC20 Module's direct input.
PN0098	NOT1	Axis 1 overtravel (–)					
PN0099	DEC1	Axis 1 deceleration dog					
PN0100	ZER1	Axis 1 home position pulse					
PN0101	POT2	Axis 2 overtravel (+)					
PN0102	NOT2	Axis 2 overtravel (–)					
PN0103	DEC2	Axis 2 deceleration dog					
PN0104	ZER2	Axis 2 home position pulse					
PN0105	POT3	Axis 3 overtravel (+)					
PN0106	NOT3	Axis 3 overtravel (–)					
PN0107	DEC3	Axis 3 deceleration dog					
PN0108	ZER3	Axis 3 home position pulse					
PN0109	POT4	Axis 4 overtravel (+)					
PN0110	NOT4	Axis 4 overtravel (–)					
PN0111	DEC4	Axis 4 deceleration dog					
PN0112	ZER4	Axis 4 home position pulse					
PN0113	ZRE1	Axis 1 home position setting completion	O	O	O	O	This signal turns ON when the home position setting is completed in an absolute position detecting system.
PN0114	ZRE2	Axis 2 home position setting completion					
PN0115	ZRE3	Axis 3 home position setting completion					
PN0116	ZRE4	Axis 4 home position setting completion					
PN0120	ZMDL	Home position setting mode	O	O	O	O	Indicates the status in which the above home position setting can be executed.

Reference	Signal	Name	Mode				Function and Timing
			O	A	M	E	
PN0121	ERST	External reset	O	O	O	O	This signal is output in synchronously with the MACHINE RESET (RST) command to reset external devices. 
PN0122	MBDL	Multi-block operating	O	X	X	X	Indicates multi-block operation from a Teaching Pendant.
PN0125	EXP1	Axis 1 external input status	O	O	O	O	Indicates the status of the external input signals EXPn. 0 = OFF, 1 = ON The final status is indicated regardless of whether the control axis is a basic axis or independent axis and regardless of the operating mode.
PN0126	EXP2	Axis 2 external input status					
PN0127	EXP3	Axis 3 external input status					
PN0128	EXP4	Axis 4 external input status					

Appendix A

Instruction and Command Lists

Appendix A-1 provides a table of the MC20 Module motion commands. These commands are explained in detail in Sections 2-2 through 2-8.

Appendix A-2 shows of the characters that can be designated for each MC20 Module motion command.

Appendix A-3 shows the MC20 Module motion commands that can be used together in the same block.

Appendix A-4 provides a table of the CPU Module ladder motion instructions These instructions are explained in detail in Sections 3-2 through 3-4.

Appendix A-5 shows the modes in which each CPU Module ladder motion instruction can be executed.

Appendix A-6 shows which CPU Module ladder motion instructions can be executed simultaneously.

A.1 MC20 Module Motion Commands

These commands are classified as follows: M1 through M6 are for modal groups 1 through 6, and NM is for the non-modal group of commands.

Command	Name	Class	Command Format	Function/Meaning	Page
MOV	POSITIONING	NM	MOV X–Y–Z–S–; ⏎ Target position	Executes simultaneous positioning for maximum four axes at rapid traverse speed.	1-34
MVS	LINEAR INTERPOLATION	NM	MVS X–Y–Z–S–F–T–; ⏎ Target position	Executes linear travel at tangential velocity F for maximum four axes simultaneously.	1-37
MCW MCC	CIRCULAR INTERPOLATION CW CIRCULAR INTERPOLATION CCW	NM	MCW X–Y–R–F–; ⏎ MCC X–Y–I–J–F–T–; ⏎	Execute circular travel at tangential velocity F for two axes simultaneously, following radius R or center-point coordinates I and J.	1-40
MCW MCC	HELICAL INTERPOLATION CW HELICAL INTERPOLATION CCW	NM	MCW X–Y–R–ZF–F–; ⏎ MCC X–Y–I–J–Z–F–T–; ⏎	Move three axes simultaneously in a combination of circular interpolation and linear interpolation outside of the circular interpolation plane. Speed designation F becomes the circular interpolation tangential velocity.	1-46
PXY PYZ PZX PXS PZS PYS	PLANE XY PLANE YZ PLANE ZX PLANE XS PLANE ZS PLANE YS	M1	MCW PXY X–Y–R–Z–F–; ⏎ MCC PXY X–Y–I–J–ZF–F–; ⏎	Designate the plane in which the circular interpolation is to be executed. Also valid for designating circular interpolation plane in the helical interpolation command.	1-46
		M6	PST PXY X–Y–I–J–U–V–; ⏎	Also used to designate the PALLET SET (PST) matrix plane for use in PALLET MOVE (PMV).	1-46
ZRN	HOME RETURN	NM	ZRN X–Y–Z–S–; ⏎ Intermediate position	Returns each axis to its home position after positioning to the intermediate position. For the first time after power-up, positioning will return directly to the home position without travelling to the intermediate position.	1-49
PMV	PALLET MOVE	NM	PMV P–C–; ⏎ Grid point number Pallet number	Positions at rapid traverse speed to the position of the grid point number for the specified pallet number. The grid point data must be saved in advance by means of PALLET SET (PST).	1-54

Command	Name	Class	Command Format	Function/Meaning	Page
PST	PALLET SET	M5	PST PXY P– X–Y– $\underbrace{\text{I–J–}}_{\text{Number of grid points}}$ $\underbrace{\text{U–V–}}_{\text{Grid point pitch}}$;⏎	Saves in memory, for the specified plane, the pallet number and its grid point data. This must be done before PALLET MOVE (PMV) is executed.	1-54
SKP	SKIP	NM	SKP X–Y–Z–S–F–T–⏎	When the SKIP signal turns ON while the axes are travelling, skips the remaining movement and operation proceeds to the next block. The position at which the SKIP signal turned ON is saved.	1-57
ABS	ABSOLUTE PROGRAMMING MODE	M2	ABS;⏎	Treats all subsequent coordinate words as absolute values.	1-61
INC	INCREMENTAL PROGRAMMING MODE	M2	INC;⏎	Treats all subsequent coordinate words as incremental values.	1-63
POS	CURRENT POSITION SET	NM	POS $\underbrace{\text{X–Y–Z–S–}}_{\text{Desired coordinate values}}$;⏎	Changes the current position to the desired coordinate values. Subsequent move commands utilize the new coordinate system.	2-59
MVM	MOVE ON MACHINE COORDINATE	NM	MVM MVS X–Y–Z–S–F–;⏎	Goes to the target position on machine coordinate system. The coordinate system set automatically at the completion of the home position return is called a machine coordinate system. This coordinate system is not affected by CURRENT POSITION SET (POS).	1-66
TIM	DWELL TIME	NM	TIM P–;⏎	Operation is paused for the amount of time specified by P, and then proceeds to the next block.	1-68
STP	PROGRAM STOP	NM	STP;⏎	Stops the motion program until it is restarted by means of the start operation.	1-69
END	PROGRAM END	NM	END;⏎	Ends the motion program.	1-69
PFN	IN-POSITION CHECK	NM	MVS X–F– PFN;⏎ or PFN;⏎	Proceeds to the next block after the positioning which is commanded in the same or previous block enters the positioning completion range (parameter setting).	1-70
INP	SECOND IN-POSITION RANGE SETTING	M3	INP $\underbrace{\text{X–Y–Z–S–}}_{\text{Second positioning completion range}}$;⏎	Proceeds to the next block after subsequently commanded interpolations enter second positioning completion range.	1-73

Command	Name	Class	Command Format	Function/Meaning	Page
SET	SET EXTERNAL OUTPUT	NM	SET M□□ ;↵ M code (01 to 96)	Proceeds to the next block after the M code is output and the MFIN signal from the CPU Module is returned.	1-74
PNT	PASS NOTCH SIGNAL OUTPUT	M4	PNT X–Y–Z–S–M□□; ↵ Transit point for each axis MOV X–Y–Z–S–;↵	Outputs the specified M code (01 to 96) when the transit points for all axes have been passed during movement in subsequent blocks.	1-77
SNG	IGNORE SINGLE-BLOCK SIGNAL	NM	SNG MOV X–;↵ SNG MVS Z–F–;↵	A block with the command will be operated continuously even in the single-block operation mode.	1-79
IOW	I/O WAIT	NM	IOW (front output variable = O) (condition I/O variable = Δ) (back output variable C = □);↵	After the front output variable "O" has been output, outputs the back output variable "□" when the condition I/O variable reaches the "Δ" state.	1-80
GSB	SUB-PROGRAM CALL	NM	GSB P□□ L□□;↵	Executes the program number specified by P as a sub-program for the number of times specified by L.	1-82
RET	SUB-PROGRAM END	NM	RET;↵	Designates the end of the sub-program.	1-83
PCN *2	PCON SIGNAL OUTPUT	NM	PCN X–Y–Z–S–; ↵	Turns ON or OFF the PCON□ signal output for the specified axis.	1-113
VCC *2	VOLTAGE OUTPUT	NM	VCC X–Y–Z–S–T–; ↵	Outputs the analog voltage as the output for the specified axis.	1-116
EXM *2	EXTERNAL POSITIONING	NM	EXM X–I–U–Y–J–V–; ↵	When the external positioning signal turns ON, moves only the external positioning travel distance and is completed.	1-134
PGS *2	RATIO OPERATION	NM	PGS X–I–Y–J–Z–K– MS□; ↵	Operates a slave axis for the travel distance that is calculated by multiplying the travel distance of the master axis by a ratio.	1-139
PGR *2	RATIO OPERATION CANCEL	NM	PGR;↵ or PGR X0 Y0; ↵	Cancels the ratio operations for all axes. Cancels the ratio operations for the specified axis or axes.	1-139
TSS *2	TRAILING SYNCHRONOUS OPERATION	NM	TSS a–P– MS□;↵ X, Y, Z or Z	After synchronizing the trailing axis with the master axis, switches to trailing synchronous mode.	1-151

Command	Name	Class	Command Format	Function/Meaning	Page
TSR *2	TRAILING SYNCHRONOUS OPERATION CANCEL	NM	TSR; ↵	Cancels the trailing synchronous operation.	1-151

*1: M1 to M6: Commands for modal groups 1 to 6
NM: Commands for non–modal groups

*2: New step-2 functions.

Command	Name	Command Format	Function/Meaning	Reference
	Variables			
#□□□	Common variable	#1 to #199	• Used for general purposes.	1-85
#I□□□	Input variable	#I1 to #I256		1-85
#O□□□	Output variable	#O1 to #O256	• Reads MC coil status.	1-86
#□□□□	System variable	#1001 to #1018		1-87
			• Outputs MC relay.	
#□□□□	Link input variable	#1101 to #1116	• Reads system variables such as the current position and saved SKIP positions.	1-93
#□□□□	Link output variable	#1201 to #1216		1-97
H□	H variable	H1 to H8		1-99
			• Reads MC link register value.	
			• Transmits values to MC link register.	
			• Used for positions and speeds.	
	Arithmetic Commands			
=	DEFINE	#i = 100, #j = #i	Numeric range: 0 to ±99,999,999 Integers only. Digits to the right of the decimal point are discarded. Calculations are carried out from left to right with no order of priority.	1-100
+	ADD	#i = #j + #k		1-100
–	SUBTRACT	#i = #j – #k		1-100
*	MULTIPLY	#i = #j * #k		1-101
/	DIVIDE	#i = #j / #k		1-101
	COMBINE	#i = #j – #k / #m		1-101
	Control Commands			
IF GOTO	BRANCH	IF <condition> GOTO n; ↵	• When the condition is realized, the program jumps to block n.	1-102
WHILE DO DEND	REPEAT	WHILE <condition> DO m; ↵ . . . DEND m; ↵	• While the condition is in effect, the block from DO m to DEND m is repeated.	1-103
#E□□□	POINT TABLE POSITION	MOV –#E–; ↵ MOV X#E–Y#E–F–; ↵	A point table for storing position data for the four axes is created. It is then possible to move to a given position by specifying a point number with the motion command.	1-106

A.2 Characters Usable in Motion Commands

These commands are classified as follows: M1 through M6 are for modal groups 1 through 6, and NM is for the non-modal group of commands. (O: Characters that can be used; Blank: Characters that cannot be used.)

Command	Name	Class	Command Overlap	Characters that Can Be Designated			
				N X Y Z S	R I J K L	XF YF ZF SF	F P C L M T
MOV	POSITIONING	NM	Yes	OOOOO			
MVS	LINEAR INTERPOLATION			OOOOO			O O
MCW	CIRCULAR INTERPOLATION CW			OOOOO	OOOOO		O O
MCC	CIRCULAR INTERPOLATION CCW			OOOOO	OOOOO		O O
MCW	HELICAL INTERPOLATION CW			OOOOO	OOOOO	O O O O	O O
MCC	HELICAL INTERPOLATION CCW			OOOOO	OOOOO	O O O O	O O
PXY PYZ PZX PXS PZS PYS	PLANE XY PLANE YZ PLANE ZX PLANE XS PLANE ZS PLANE YS	M1 M6 (Note 3)	Yes Yes	O O O O O O			
ZRN	HOME RETURN	NM	No	OOOOO			
PMV	PALLET MOVE	NM	No	O			OO
PST	PALLET SET	M5	Yes	OOOOO	OOOO	(U, V, W, T)	O
SKP	SKIP	NM	No	OOOOO			O O
ABS	ABSOLUTE PROGRAMMING MODE	M2	Yes	O			
INC	INCREMENTAL PROGRAMMING MODE			O			
POS	CURRENT POSITION SET	NM	No	OOOOO			
MVM	MOVE ON MACHINE COORDINATES	NM	Yes	O			
TIM	DWELL TIME	NM	No	O			O
STP	PROGRAM STOP	NM	No	O			
END	PROGRAM END	NM	No	O			
PFN	IN-POSITION CHECK	NM	Yes	O			
INP	SECOND IN-POSITION RANGE SETTING	M3	No	O			
SET	SET EXTERNAL OUTPUT	NM	Yes	O			O
PNT	PASS NOTCH SIGNAL OUTPUT	M4	Yes	OOOOO			O
SNG	IGNORE SINGLE-BLOCK SIGNAL	NM	Yes	O			
IOW	I/O WAIT	NM	No	O			
GSB	SUB-PROGRAM CALL	NM	No	O			O O
RET	SUB-PROGRAM END	NM	No	O			
PCN	PCON SIGNAL OUTPUT	NM	No	OOOOO			
VCC	VOLTAGE OUTPUT	NM	No	OOOOO			O
EXM	EXTERNAL POSITIONING	NM	No	OOOOO	OOOO	(U, V, W, T)	
PGS	RATIO OPERATION	NM	No	OOOOO	OOOO	(MS□)	
PGR	RATIO OPERATION CANCEL	NM	No	OOOOO			

Command	Name	Class	Command Overlap	Characters that Can Be Designated			
				N X Y Z S	R I J K L	XF YF ZF SF	F P C L M T
TSS	TRAILING SYNCHRONOUS OPERATION	NM	No	O O O O O		(MS□)	O
TSR	TRAILING SYNCHRONOUS OPERATION CANCEL	NM	No	O			

Note (1) Commands marked with "Yes" in the "Command Overlap" column can be designated in the same block with other commands. Commands in the same group, e.g., MOV to MCC, PXY to PYS, and ABS to INC) cannot be used together in the same block. For details, refer to *A.3 Compatibility of MC20 Module Commands.*

(2) A numeral (0 to 9), H variable (H1 to H8) and common variable (#1 to #199) can be added after the designated character.

(3) The plane designation for CIRCULAR INTERPOLATION is in modal group M1.
The matrix plane designation for PALLET MOVE is in modal group M6.

A.3 Compatibility of MC20 Module Commands

The following table shows which commands can be used together in the same block. An "O" indicates that the two commands can be used together and an "X" indicates that they cannot.

Command	Compatible Commands																																		
	MOV	MVS	MCW	MCC	PXY	PYZ	PZX	PXS	PYS	PZS	ZRN	PMV	PST	SKP	ABS	INC	POS	MVM	TIM	STP	END	PFN	INP	SET	PNT	SNG	IOW	GSB	RET	=+−*/	IF	GOTO	WHILE	DO	DEND
MOV	X	X	X	X	O	O	O	O	O	O	X	X	X	X	O	O	X	O	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
MVS	X	X	X	X	O	O	O	O	O	O	X	X	X	X	O	O	X	O	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
MCW (Note 1)	X	X	X	X	O	O	O	O	O	O	X	X	X	X	O	O	X	O	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
MCC (Note 1)	X	X	X	X	O	O	O	O	O	O	X	X	X	X	O	O	X	O	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
PXY	O	O	O	O	X	X	X	X	X	X	X	O	O	X	O	O	X	O	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
PYZ	O	O	O	O	X	X	X	X	X	X	X	O	O	X	O	O	X	O	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
PZX	O	O	O	O	X	X	X	X	X	X	X	O	O	X	O	O	X	O	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
PXS	O	O	O	O	X	X	X	X	X	X	X	O	O	X	O	O	X	O	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
PYS	O	O	O	O	X	X	X	X	X	X	X	O	O	X	O	O	X	O	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
PZS	O	O	O	O	X	X	X	X	X	X	X	O	O	X	O	O	X	O	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
ZRN	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
PMV	X	X	X	X	O	O	O	O	O	O	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
PST	X	X	X	X	O	O	O	O	O	O	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
SKP	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
ABS	O	O	O	O	O	O	O	O	O	O	O	O	O	O	X	X	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O
INC	O	O	O	O	O	O	O	O	O	O	O	O	O	O	X	X	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O
POS	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
MVM	O	O	O	O	O	O	O	O	O	O	X	X	X	X	O	O	X	X	X	X	X	O	X	X	O	O	X	X	X	X	X	X	X	X	X
TIM	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
STP	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
END	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
PFN	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	X	O	O	O	O	O	O	O	O	O	O	O	O	O
INP	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
SET	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
PNT	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	O	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
SNG	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	O	X	O	O	O	O	O	O	O	O	O
IOW	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
GSB	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
RET	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
PCN *2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
VCC *2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
EXM *2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
PGS *2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
PGR *2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	X	X	X	X	O	X	X	X	X	X	X	X	X	X
TSS *2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	X	X	X	X	O	X	X	X	X	X	X	X	X	X
TSR *2	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
=, +, X, ∗, /	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X
IF	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	O	X	X	X
GOTO	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	O	X	X	X	X
WHILE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	O	X
DO	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	O	X	X
DEND	X	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O	X	X	X	X	X	O	X	X	X	O	X	X	X	X	X	X	X	X	X

*2: New step-2 functions.

Note

1) The circular and helical interpolation versions of MCW and MCC are compatible with the same commands.

2) Refer to *A-1 MC20 Module Motion Commands* for a list of command names.

A.4 CPU Module Motion Instructions

Instruction	Name	Function/Meaning	Page
MOD	MODE SET	Switches the MC20 Module operation mode.	2-31
SVN	SERVO ON	Turns the servomotor power ON and OFF.	2-34
MVL	PROGRAM RUN	Specifies the program and block numbers, and runs the program.	2-37
MVA, MVB, MVC, MVD	Independent Axis Operations	Normally operates independent axes A to D.	2-43
	Independent Axis Voltage Outputs (See note.)	Outputs the analog voltage (voltage output) from independent axes A to D.	2-51
	Independent Axis Ratio Operations (See note.)	Performs ratio operations with independent axes A to D as slave axes.	2-58
ZRN	HOME RETURN	Returns to home position in an incremental or absolute position detecting system.	2-65
JOG	JOG	Executes jogging.	2-71
STP	STEP	Executes stepping.	2-75
SMD	SINGLE BLOCK MODE	Switches to single block operation mode.	2-82
MLK	MACHINE LOCK MODE	Causes the controlled axes not to move but the current position data changes according to program running.	2-85
MRS	MODULE RESET	Initializes the MC20 Module and resets all alarm status.	2-88
RST	MACHINE RESET	Resets part of the alarm status and enables the parameters changed with the PARAMETER SETTING (PRM) instruction.	2-92
ESP	EMERGENCY STOP NOTIFICATION	Notifies the MC20 Module that the emergency stop button has been pressed.	2-96
ARS	ALARM RESET	Resets the MC20 Module alarm.	2-98
MON	MONITOR	Monitors all internal data, including alarms, parameters, and program numbers.	2-101
POS	COORDINATE SETTING	Changes current position data.	2-109
PRM	PARAMETER SETTING	Sets data for parameters.	2-114
VAR	H VARIABLE SETTING	Sets data for H1 to H8 variables.	2-117
PTBL	POINT TABLE SETTING	Sets data for point table.	2-122
ZST	HOME POSITION SETTING	Sets the home position for absolute position detecting system.	2-126

Note New step-2 functions.

A.5 Operation Modes and Applicable Instructions

Instruction			Operation Mode			
Symbol		Name	Manual	Automatic	Online Edit	Edit
MOD		MODE SET	○	○	○	○
SVN		SERVO ON	○	○	○	○
MVL		PROGRAM RUN	X	○	X	X
MVA, MVB, MVC, MVD		Independent Axis Operations	○	○	○	○
		Independent Axis Voltage Outputs (See note.)	○	○	○	○
		Independent Axis Ratio Operations (See note.)	○	○	○	○
Program run axes	ZRN	HOME RETURN	○	X	X	X
	JOG	JOG	○	X	○	X
	STP	STEP	○	X	○	X
Independent axes	ZRN	HOME RETURN	○	○	○	○
	JOG	JOG	○	○	○	○
	STP	STEP	○	○	○	○
SMD		SINGLE BLOCK MODE	○	○	○	○
MLK		MACHINE LOCK MODE	○	○	○	○
MRS		MODULE RESET	○	○	○	○
RST		MACHINE RESET	○	○	○	○
ESP		EMERGENCY STOP NOTIFICATION	○	○	○	○
ARS		ALARM RESET	○	○	○	○
MON		MONITOR	○	○	○	○
POS		COORDINATE SETTING	○	○	○	○
PRM		PARAMETER SETTING	○	○	○	○
VAR		H VARIABLE SETTING	○	○	○	○
PTBL		POINT TABLE SETTING	○	○	○	○
ZST		HOME POSITION SETTING	○	○	○	○

Note New step-2 functions.

○: Instructions that can be executed in that mode.
X : Instructions that cannot be executed in that mode.

A.6 Instructions Executed Simultaneously in a Scan

Instruction in Progress		Instruction Executed Simultaneously																				
Symbol	Name	MOD	SVN	MVL	INDEPENDENT AXIS OPERATION	VOLTAGE OUTPUT	RATIO OPERATION	ZRN	JOG	STP	SMD	MLK	MRS	RST	ESP	ARS	MON	POS	PRM	VAR	PTBL	ZST
MOD	MODE SET	X	○	X	○	○	○	X	X	X	○	○	○	○	○	○	○	○	○	○	○	○
SVN	SERVO ON	○	Δ	X	X	X	X	X	X	X	○	○	○	○	○	○	○	X	○	○	○	X
MVL	PROGRAM RUN	○	○	X	Δ	Δ	Δ	X	X	X	○	○	○	○	○	○	○	Δ	X	X	X	X
MVA, MVB, MVC, MVD	Independent Axis Operations	○	○	○	Δ	Δ	Δ	Δ	Δ	Δ	○	○	○	○	○	○	○	Δ	X	○	X	X
	Independent Axis Voltage Outputs (See note.)	○	○	○	Δ	Δ	Δ	Δ	Δ	Δ	○	○	○	○	○	○	○	Δ	X	○	X	X
	Independent Axis Ratio Operations (See Note.)	○	○	○	Δ	Δ	Δ	Δ	Δ	Δ	○	○	○	○	○	○	○	Δ	X	○	X	X
ZRN	HOME RETURN	○	○	X	Δ	Δ	Δ	Δ	Δ	Δ	○	○	○	○	○	○	○	Δ	X	○	○	X
JOG	JOG	○	○	X	Δ	Δ	Δ	Δ	Δ	Δ	○	○	○	○	○	○	○	Δ	X	○	○	X
STP	STEP	○	○	X	Δ	Δ	Δ	Δ	Δ	Δ	○	○	○	○	○	○	○	Δ	X	○	○	X
SMD	SINGLE BLOCK MODE	○	○	○	○	○	○	○	○	○	X	○	○	○	○	○	○	○	○	○	○	○
MLK	MACHINE LOCK MODE	○	○	○	○	○	○	○	○	○	○	X	○	○	○	○	○	○	○	○	○	X
MRS	MODULE RESET	X	X	X	X	X	X	X	X	X	X	○	X	X	X	X	X	X	X	X	X	X
RST	MACHINE RESET	○	X	X	X	X	X	X	X	X	X	○	X	X	○	○	○	○	X	○	○	X
ESP	EMERGENCY STOP NOTIFICATION	○	X	X	X	X	X	X	X	X	○	○	○	○	X	○	○	X	○	○	○	X
ARS	ALARM RESET	○	○	X	X	X	X	X	X	X	○	○	○	○	○	X	○	○	○	○	○	X
MON	MONITOR	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	X	○	○	○	○	○
POS	COORDINATE SETTING	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	X	X	X	X	X
PRM	PARAMETER SETTING	○	○	X	X	X	X	X	X	X	○	○	○	○	○	○	○	X	X	X	X	X
VAR	H VARIABLE SETTING	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	X	X	X	X	○
PTBL	POINT TABLE SETTING	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	X	X	X	X	○
ZST	HOME POSITION SETTING	○	X	X	X	X	X	X	X	X	○	X	X	X	X	X	○	X	X	○	○	Δ

Note New step-2 functions.

○ : Instructions that can be executed simultaneously.

Δ : Instructions that can be executed simultaneously if a different axis is specified.

X : Instructions that cannot be executed simultaneously in a scan.

Appendix B

Parameter Lists

- The specific settings required to operate the MC20 Module are known as parameters. Set the parameters to the optimal values based on the mechanical specifications and the characteristics of the servodrivers being used.

- There are 4 types of parameters, shown in the following table.

Type	Meaning
A	Parameters that must be set for standard usage.
B	Parameters set when required.
C	Parameters normally left unchanged.
D	Parameters not set by the user. (Do not change the default settings.)

- The "default values" are the parameter settings made at the factory.

- All new parameter settings are not effective immediately after they have been changed with the Programming Device. The "Effective" column in the following tables indicates when the parameter is effective.

  Reset
  New parameter settings for these parameters are effective after the MACHINE RESET (RST) instruction has been executed.

  Power-up
  New parameter settings for these parameters (and all other parameters) are effective after the power is turned ON or the MODULE RESET (MRS) instruction is executed.

- MC20 Module parameters are not put into effect immediately after being changed with the PARAMETER SETTING (PRM) instruction. To change the parameters to the new set values and put them into effect, execute the MACHINE RESET (RST) instruction.

B.1 Parameters Common to All Control Axes

No.	Name[1]	Setting Range[1]	Units	Related Parameters	Read/ Write[2]	Effective[4]	Default Value	Type
P0000	ID Code	Alphanumeric characters (4)	ASCII code	---	No	Immediate	0	B
P0001	Axis 1 name specification	Axis X: b0=1 Axis Y: b1=1	Note 3.	---	No	Power-up	X	C
P0002	Axis 2 name specification	Axis Z: b2=1 Axis S: b3=1		---	No	Power-up	Y	C
P0003	Axis 3 name specification	Axis A: b4=1 Axis B: b5=1		---	No	Power-up	Z	C
P0004	Axis 4 name specification	Axis C: b6=1 Axis D: b7=1		---	No	Power-up	S	C
P0005	Decimal point position	1, 2, or 3	---	---	No	Reset	3	B
P0006	Max. interpolation feed speed setting	1 to 240,000	mm/min. (deg./ min.)	---	Yes	Reset	24,000	A
P0007	Time constant of linear A/D for interpolation (1)	1 to 10,000	ms	---	Yes	Reset	100	A
P0008	Time constant of linear A/D for interpolation (2)	1 to 10,000	ms	---	Yes	Reset	100	B
P0009	Linear A/D constant switch speed for interpolation	0 to 240,000	mm/min. (deg./ min.)	---	Yes	Reset	24,000	B
P0010	Deceleration constant of the asymmetric A/D for interpolation	1 to 10,000	ms	---	Yes	Reset	100	B
P0011	Time constant of the exponential A/D for interpolation	2 to 1,000	ms	---	Yes	Reset	100	B
P0012	Bias speed of the exponential A/D for interpolation	0 to 240,000	mm/min. (deg./ min.)	---	Yes	Reset	0	B
P0013	Time constant of moving average A/D for interpolation	2 to 1,000	ms	---	Yes	Reset	100	B
P0014	A/D type setting for interpolation (MVS, MCW, etc.)	0: None 1: Single step linear A/D 2: Double step linear A/D 3: Asymmetric linear A/D		---	No	Reset	1	B
P0015	Filter selection for interpolation (MVS, MCW, etc.)	0: None 1: Exponential A/D 2: Exponential A/D with bias 3: Moving average A/D 4: S-curve A/D		---	No	Reset	0	B

No.	Name[1]	Setting Range[1]	Units	Related Parameters	Read/ Write[2]	Effective[4]	Default Value	Type
P0016	Override enabled/disabled bn = 0: Disabled bn = 1: Enabled	Overrides Using MC Control Coils: 16 Steps		202	○	Reset	0	B
		b0: Axis 1 b1: Axis 2 b2: Axis 3 b3: Axis 4	For independent axis and manual operation					
		b8: Rapid traverse b9: Interpolation feed	For program run					
	Ver. B08	Overrides Using MC Link Registers: 0.1% Increments						
		b11: Axis 1 b12: Axis 2 b13: Axis 3 b14: Axis 4	For independent axis and manual operation					
		b15: Rapid traverse b16: Interpolation feed	For program run					
B08 P0017	Function selections 4	b0: Manual operation simultaneous outputs b1: Axis alarm disable b2: Servo OFF disable			×	Power-up	0	B

Note 1) "A/D" stands for acceleration/deceleration speed.

2) "Yes" indicates parameters can be changed from the ladder logic program.

3) Set axis X, Y, Z, S, A, B, C, or D in the Programming Device.
Set "–" if the axis is not used.

4) Effective modes are indicated as follows:

Power-up

Parameter settings not effective immediately after they have been made with the Programming Device or Teach Pendant but effective only after the MODULE RESET (MRS) instruction is executed.

Reset

Parameter settings not effective immediately after they have been made with the Programming Device or Teach Pendant but effective only after the MC20 Module is turned off and on, the MACHINE RESET (RST) instruction is executed, or the MODULE RESET (MRS) instruction is executed.

B.2 Individual Axis Parameters (Positioning)

No.[1]	Name	Setting Range	Units	Related Parameters	Read/ Write[2]	Effective	Default Value	Type
PA101	Position loop gain	0 to 200	s^{-1}	---	Yes	Reset	30	A
PA102	Feed forward gain	0 to 200	%	---	Yes	Reset	0	A
PA103	Positioning completion range	0 to 100,000	Note 3.	PA104	Yes	Reset	10	A
PA104	Positioning completion check time	0 to 100,000	ms	PA103	Yes	Reset	100,000	A
PA105	Following error margin	0 to 200	%	PA101	Yes	Reset	200	A

Note 1) "A" in the parameter number stands for the axis number (1 through 4).

2) "Yes" in the Read/Write column indicates the parameter can be changed from the ladder logic program (Ladder or Teach Pendant.

3) Reference unit.

B.3 Individual Axis Parameters (Speed, Acceleration)

No.[1]	Name[2]	Setting Range[2]	Units	Related Parameters	Read/ Write[3]	Effective	Default Value	Type
PA201	Maximum feed speed	1 to 240,000	mm/min. (deg./min.)	---	Yes	Reset	24,000	A
PA202	Rapid traverse speed	1 to 240,000	mm/min. (deg./min.)	PA201, PA204, PA205, PA206	Yes	Reset	24,000	A
PA203	Not used.	---	---	---	---	---	0	---
PA204	Linear A/D constant (1)	1 to 10,000	ms	PA201	Yes	Reset	100	A
PA205	Linear A/D constant (2)	1 to 10,000	ms	PA201, PA202, PA204, PA206	Yes	Reset	100	B
PA206	Linear A/D constant switch speed	0 to 240,000	mm/min. (deg./min.)	PA201, PA202, PA204, PA205	Yes	Reset	24,000	B
PA207	Deceleration constant for asymmetric A/D	1 to 10,000	ms	PA201, PA213 to 216	Yes	Reset	100	B
PA208	Not used.	---	---	---	---	---	0	---
PA209	Time constant for exponential A/D	2 to 1,000	ms	PA210, PA217	Yes	Reset	100	B
PA210	Bias speed for exponential A/D	0 to 240,000	mm/min. (deg./min.)	PA209, PA217	Yes	Reset	0	B
PA211	Time constant for moving average A/D	2 to 1,000	ms	PA217	Yes	Reset	100	B
PA212	Not used.	---	---	---	---	---	---	---
PA213	A/D type for positioning (MOV/STEP)	0: None 1: Single step linear A/D 2: Double step linear A/D 3: Asymmetric linear A/D		PA204 to 207	Yes	Reset	1	B
PA214	A/D type for JOG operation (JOG)							
PA215	A/D type for independent axis operation (MVA to MVD)							
PA216	A/D type for HOME RETURN (ZRN)	Fixed at single step linear A/D		---	---	---	---	---
PA217	Filter selection (See note 4.)	0: None 1: Exponential A/D 2: Exponential A/D with bias 3: Moving average A/D 4: S-curve A/D.		PA209 PA210 PA211	Yes	Reset	0	B

Note 1) "A" in the parameter number stands for the axis number (1 through 4).

2) "A/D" stands for acceleration/deceleration speed.

3) "Yes" in the "Read/Write" column indicates that the parameter can be changed from the ladder logic program (Ladder or Teach Pendant).

4) The filter for PA213 to PA216.

B.4 Individual Axis Parameters (Home Position Return)

No.[1]	Name	Setting Range	Units	Related Parameters	Read/ Write[2]	Effective	Default Value	Type
PA301	Home position return mode	0: DEC+C 1: ZERO 2: DEC+ZERO 3: C	---	PA302 to PA308	No	Power-up	0	B
PA302	Home position return direction	0: Positive 1: Negative	---	---	No	Power-up	0	B
PA303	Home position returning feed speed	1 to 240,000	mm/min. (deg./min.)	---	Yes	Reset	10,000	B
PA304	Home position returning approach speed	1 to 240,000	mm/min. (deg./min.)	---	Yes	Reset	1,000	B
PA305	Home position returning creep speed	1 to 240,000	mm/min. (deg./min.)	---	Yes	Reset	500	B
PA306	Home position returning final travelling distance	0 to 99999999	Note 3.	---	Yes	Reset	0	B
PA307	Home position output width	0 to 32,767	Pulses	---	Yes	Reset	100	B
PA308	Home position pulse polarity selection	0: Positive transition 1: Negative transition		---	No	Power-up	0	C
PA309	Not used	---						
PA310	Deceleration limit switch inversion	0: Disabled 1: Enabled		---	No	Power-up	0	C

Note

1) "A" in the parameter number stands for the axis number (1 through 4).

2) "Yes" in the "Read/Write" column indicates the parameter can be changed from the ladder logic program (Ladder or Teach Pendant).

3) Reference unit.

B.5 Individual Axis Parameters (Absolute Detecting System)

No.[1]	Name	Setting Range	Units	Related Parameters	Read/ Write[2]	Effective	Default Value	Type
PA401	Absolute encoder allowable error	0 to 1,000,000	Pulses	PA402	No	Power-up	40,960	A
PA402	Absolute detecting system selection	See note 3.		PA401, PA403, PA404	No	Power-up	0	C
PA403	Reference offset 1 at home position setting (home position shift distance)	0 to 99,999,999	Note 4.	PA402, PA404	No	Power-up	0	B
PA404	Reference offset 2 at home position setting (home position shift adjustment)	0 to ±99,999,999	Note 4.	PA402, PA403	No	Power-up	0	B

Note

1) "A" in the parameter number stands for the axis number (1 through 4).

2) These parameters cannot be changed from the ladder logic program (Ladder or T-BOX).

3) The first bit (b0) determines the detecting system and the second bit (b1) determines the encoder to be used.

Parameter Setting	Bit Values	Encoder	Detecting System
0	b0=0, b1=0	Incremental	Incremental
2	b0=0, b1=1	Absolute	Incremental
3	b0=1, b1=1	Absolute	Absolute

4) Reference unit.

B.6 Individual Axis Parameters (Machine System and Peripheral Equipment)

No.[1]	Name	Setting Range	Units	Related Parameters	Read/ Write[2]	Effective	Default Value	Type
PA501	Number of encoder pulses	32,768 max.	Pulses/ rev.	PA502	No	Power-up	2,048	C
PA502	Encoder pulse signal selection	1: AB phase x 1 2: AB phase x 2 4: AB phase x 4		PA501	No	Power-up	4	B
PA503	One machine rotation/ reference unit	1 to 1,500,000	Note 3.	---	No	Power-up	10,000	A
PA504	Gear ratio (motor rev.)	1 to 10,000,000	Revs.	---	No	Power-up	1	A
PA505	Gear ratio (load rev.)	1 to 10,000,000	Revs.	---	No	Power-up	1	A
PA506	Mode settings:				No	Power-up		
	b0: Motor revolution direction	0: Forward, 1: Reverse		---			0	B
	b1: Finite/Infinite length	0: Finite length, 1: Infinite length		510-b0			0	B
	b2: Linear/Rotary axis	0: Linear axis, 1: Rotary axis		---			0	C
	b3: Axis type	0: Servo, 1: Voltage output		506-b4			0	C
	b4: Counter	0: Do not use, 1: Use		506-b3			0	B
PA507	Backlash compensation	0 to 32,768	Pulses	PA510 (b1)	Yes	Reset	0	B
PA508	Stored stroke limit (+)	–99999999 to +99999999	Note 3.	PA510 (b0)	Yes	Reset	+99999999	B
PA509	Stored stroke limit (–)	–99999999 to +99999999	Note 3.	PA510 (b0)	Yes	Reset	–99999999	B
PA510	b0: Stored stroke limit	0: Not used 1: Used		PA508, PA509	No	Power-up	0	B
	b1: Backlash offset usage	0: Not used 1: Used		PA507	No	Power-up	0	B
PA511	Servomotor rated revolution	100 to 4,500	r/min.	---	No	Reset	3,000	B
PA512	Speed instruction D/A output	1 to 10	V	---	No	Reset	6	B
PA513	Servomotor maximum revolution	0 to 10,000	r/min.	---	No	Reset	4,000	B
PA514	Constant of automatic 0 tune	1,000 to 9,999	ms	---	---	---	1,000	D

Note

1) "A" in the parameter number stands for the axis number (1 through 4).

2) "Yes" in the "Read/Write" column indicates the parameter can be changed from the ladder logic program (Ladder or Teach Pendant).

3) Reference unit.

B.7 Individual Axis Parameters (Servo External I/O)

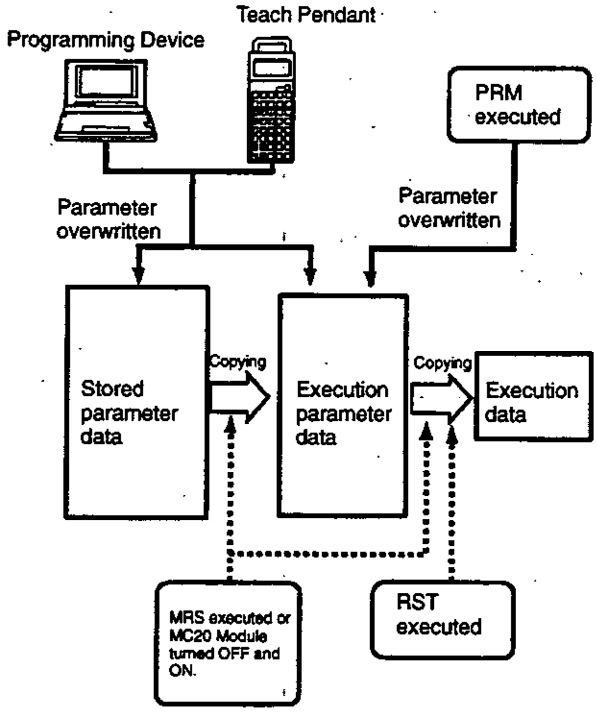
No.[1]	Name	Setting Range	Units	Related Parameters	Read/ Write[2]	Effective	Default Value	Type
PA601	b0: Overtravel input signal (OT)	0: Not used 1: Used		---	No	Power-up	1	B
	b2: Brake control output signal (BRK)	0: Not used 1: Used		PA602, PA603			0	B
PA602	Brake time (Tb) (= servo OFF)	8 to 1,000	ms	PA601 (b2), PA603	Yes	Reset	8	B
PA603	Brake ON motor speed	1 to 10,000	r/min	PA601 (b2), PA602	Yes	Reset	1	B

Note 1) "A" in the parameter number stands for the axis number (1 through 4).

2) "Yes" in the "Read/Write" column indicates the parameter can be changed from the ladder logic program (Ladder or Teach Pendant).

B.8 Setting Parameters

1) The following illustration shows how parameters are overwritten with the Programming Device, Teach Pendant, or PARAMETER SETTING (PRM) instruction.



- PRM: PARAMETER SETTING instruction
- MRS: MODULE RESET instruction
- RST: MACHINE RESET instruction

Refer to *2.4 Data Setting Instructions* for details on the above instructions.

2) The Programming Device and Teach Pendant overwrite both the stored and execution parameter data. Then if the MODULE RESET (MRS) instruction is executed or the MC20 Module is turned OFF and ON, the contents of the stored parameter data are copied as the execution parameter data and the execution data is created according to the contents of the execution parameter data. At this point, the overwritten parameters will be valid. If the MACHINE RESET (RST) instruction is executed after the parameters are overwritten, the overwritten parameters will be also valid.

3) If the PARAMETER SETTING (PRM) instruction is executed, only the execution parameter data will be overwritten. Then if the MACHINE RESET (RST) instruction is executed, the execution data will be created according to the contents of the execution parameter data. At this point, the overwritten parameters will be valid.

Note If the MODULE RESET (MRS) instruction is executed or the MC20 Module is turned OFF and ON after the PARAMETER SETTING (PRM) instruction is executed, the overwritten parameters will be invalid and the previous parameters will be valid.

Appendix C

Alarm Display Lists

The tables in this appendix show the alarm displays and their likely causes and remedies. Message displays do not appear on a Teach Pendant, as shown in the following table.

Device	Alarm Code Display	Message Display
Programming Device	Yes	Yes
Teach Pendant	Yes	No

The last column in each table indicates the rank of the error (A, B, or C).

Error Rank	Description
A	Servo OFF
B	Decelerating to stop.
C	Program run stops.

C.1 Common Alarms

Code	Message	Likely Cause(s)	Remedy	Rank
001	Program capacity exceeded	There is not enough space area for the program.	Delete unneeded programs.	C
002	1-block character exceeded	The number of characters in 1 block has exceeded 128. There is not a ";" within 1 block.	Correct the program. (Reduce the number of characters.)	C
003	No program number	Cannot find the specified program.	Load the program or correct it.	C
004	Address error	There is not any data after the symbol. There is not a symbol before the data. The address designation is improper.	Correct the program.	C
005	Numerical value designation error	A minus sign, zero, or decimal point has been used incorrectly. The decimal point placement is incorrect.	Correct the program. Check the decimal point position parameter. (P0005)	C
006	Character error	There is an unusable character in a significant information area.	Correct the program.	C
007	Data digit number error	The input data's digits are incorrect.	Correct the program. (Data digit number)	C
008	Command error	An unusable command has been used.	Correct the program.	C
009	Multi-command error	Commands that cannot be executed at the same time have been executed in 1 block.	Correct the program.	C
010	F designation undefined	There is not an interpolation feed speed: F designation during interpolation operation.	Correct the program.	C
011	No radius designation for circular interpolation	The radius has been set to "0" in a circular interpolation command.	Correct the program. (Either R or I and J)	C
012	Not used.	---	---	---
013	Not used.	---	---	---
014	Pass notch signal output command error	A position greater than the maximum programmable value was designated with an "infinite length" axis.	Check the function setting parameters. Correct the program.	C
015	Not used.	---	---	---
016	Plane designation error	The interpolation plane designation is missing or incorrect. The plane designation is missing or incorrect in a PST command.	Correct the program.	C
017	H variable error	A designated value in H1 through H8 is incorrect.	Correct the program.	C
018	No sub-program number	P□□ is not specified in the GSB command's block.	Correct the program.	C
019	Not used.	---	---	---
020	Not used.	---	---	---
021	Sub-program nesting error	Five or more levels were nested in a sub-program.	Correct the program so that there are 4 or less nesting levels.	C
022	Program end error	There is not an END command at the end of the program. There is not a RET command at the end of a sub-program.	Add an END command at the end of the program. Add a RET command at the end of the sub-program.	C
023	Not used.	---	---	---

Code	Message	Likely Cause(s)	Remedy	Rank
024	Axis undefined	The axis you want to use is not valid. The servo is not ON for the axis you want to use.	Correct the program. Check the system setting parameters. Check the servo ON circuit. Check the servo alarm.	C
025	Zero division	A division by zero was performed.	Correct the program. Check related parameters.	C
026	Arithmetic overflow	An overflow occurred in an arithmetic.	Correct the program. Check related parameters.	C
027	Branch command error	There is not a destination in a branch command.	Correct the program.	C
028	Repeat command error	There is not an end command (DEND) for the repeat command. The repeatable range is duplicated. There are more than 3 repeatable nesting levels. The repeat number designation is incorrect.	Correct the program.	C
029	Matrix setting command error	The set value of the matrix setting command is outside the proper range. The grid point setting is outside the proper range.	Correct the program.	C
030	Point table setting error	The point table setting is outside the proper range.	Correct the program.	C
031	Not used.	---	---	---
032	Not used.	---	---	---
033	M code designation error	The M code designation is incorrect.	Correct the program.	C
034	Operation block "=" error	The location of the "=" in an arithmetic expression is incorrect.	Correct the program.	C
035	Variable designation error	A common, input, H, or point table variable designation is incorrect.	Correct the program.	C
036	Operator designation error	An operator designation in an arithmetic expression is incorrect.	Correct the program.	C
037	Waiting for input signal command error	The specification of a waiting for input signal command (IOW) is incorrect.	Correct the program.	C
038	Comparison operator designation error	The comparison operator designation in a conditional expression is incorrect.	Correct the program.	C
039	F designation value exceeded	The command value of the interpolation feed speed: F designation exceeds the maximum.	Correct the program.	C
040	Number of terms in expression exceeded	There are more than 11 terms in the expression.	Correct the program.	C
041	Duplicated axis designation	A moving axis has been designated to move with another instruction.	Correct the program.	C
042	ABSO designation range exceeded	An axis with infinite length designation has been designated to move to a position exceeding the allowable range with the MC20 Module in absolute mode.	Correct the program.	C
043 to 078	Not used.	---	---	---

Code	Message	Likely Cause(s)	Remedy	Rank
079	Parameter destruction	The backup battery has been removed. There is an error in the power system. There is a problem with the MC20 Module.	Check the CPU Module's battery. Check the power system. Set the parameters and program again. Contact your Yaskawa representative if the problem recurs.	A
080	Axis name duplicated designation	An axis name has been duplicated.	Correct the parameters.	A
081	Emergency stop	Emergency stop	Clear the emergency stop. Check the emergency stop input.	A
082	Illegal parameter	There is a problem with the group of parameters that have been set. For example, the parameter combination might be incorrect.	Set the parameters again.	---
083	Not used.	---	---	---
084	E^{2}PROM error	There is an error in the gain adjustment or zero adjustment written in EEPROM.	Contact your Yaskawa representative.	A
085 to 090	Not used.	---	---	---

C.2 Individual Axis Alarms

"A" in the code stands for the axis number (1 through 4).

Code	Message	Likely Cause(s)	Remedy	Rank
A01	Servo Amp abnormal	Servo Amp error.	Check for Servo Amp errors. Contact your Yaskawa representative if the problem recurs even after the Servo Amp is reset.	A
A02	Overtravel (+)	The overtravel (+) signal went ON. Operating error or program error. Parameter setting error.	Check the overtravel limit switch, bring it back in the opposite direction after resetting. Check parameters related to overtravel alarm detection. Check the overtravel input signal.	B
A03	Overtravel (–)	The overtravel (–) signal went ON. Operating error or program error Parameter setting error.		
A04	Excessively following error	The following error of the servo system is too large.	Check connections between the MC20 Module, Servo Amps, and motor. Check the settings of the system setting parameters and parameters related to the servo. Check other factors such as the mechanical load.	A
A05	Stored stroke limit (+)	An attempt was made to move beyond the possible movement range of the stored stroke limit. Operating error or program error. Parameter setting error.	After checking the program and operation, reset and bring it back in the opposite direction. Check parameters related to the stored stroke limit.	B
A06	Stored stroke limit (–)			
A07	Positioning error	Positioning cannot be performed correctly.	Check parameters related to the servo. Check connections between the Servo Amps and motors. Check other factors such as the mechanical load.	A
A08	Servo ON/OFF time-out	Servo Amp error. MC20 Module faulty.	Check for Servo Amp errors. Contact your Yaskawa representative if the problem recurs even after the Servo Amp is reset.	A
A09	Servo power OFF	Movement of the controlled axis was ordered but the servo power was not ON.	Reset the alarm and then turn ON the servo power.	A
A10	Encoder disconnected	Encoder's wiring faulty or disconnected. Encoder or Servo Amp faulty. MC20 Module faulty.	Check the encoder's wiring. Contact your Yaskawa representative.	A

Code	Message	Likely Cause(s)	Remedy	Rank
A11	Speed follow-up disabled	During follow-up synchronous operation, master axis parameter PA201 (rapid traverse speed) exceeded follow-up axis parameter PA201 (maximum feed speed). In a ratio operation, master axis parameter PA201 (rapid traverse speed) exceeded slave axis parameter PA201 (maximum feed speed). In a ratio operation, the slave axis speed exceeded parameter PA201 (maximum feed speed).	Lower the setting of master axis parameter PA201 (rapid traverse speed). Lower the setting of master axis parameter PA201 (rapid traverse speed). Lower the ratio setting.	B
A12	Absolute detecting system axis move error	In the system that absolute encoder is used; The axis travelled while power was OFF. Parameter setting error. Absolute encoder error.	Reset after checking the mechanical position and display position. Check the system setting parameters. Check the absolute encoder's wiring. Initialize the absolute encoder.	A
A13	Not used.	---	---	---
A14	Absolute encoder alarm	In the system that absolute encoder is used; Absolute encoder alarm.	Check the alarm information with the Servopack's digital operator. Initialize the absolute encoder.	A
A15	Absolute encoder communication error	In the system that absolute encoder is used; Absolute encoder communication error.	Check the wiring between the MC20 Module and Servo Amp. Check the "SEN" signal. Check the 24 VDC power.	A
A16	Home position setting incomplete	In the system that absolute encoder is used; The "home position setting" has not been made.	Perform the "home position setting" operation.	B
A17	Home position set during travelling	In the system that absolute encoder is used; Positioning was not completed when the home position was set. The home position was set while the axis was moving.	Check the positioning completion range parameter. Perform the "home position setting" operation after movement is completed.	B
A18	Absolute encoder battery alarm	In the system that absolute encoder is used; Battery alarm from absolute encoder.	Check the battery. Check the wiring between the MC20 Module and Servo Amps. Check the wiring between the Servo Amp and motor.	A
A19	Not used.	---	---	---
A20	Not used.	---	---	---

INDEX

Motion Module MC20

SOFTWARE USER'S MANUAL

IRUMA BUSINESS CENTER
480, Kamifujisawa, Iruma, Saitama 358-8555, Japan
Phone 81-42-962-5696 Fax 81-42-962-6138

YASKAWA ELECTRIC AMERICA, INC.
2121 Norman Drive South, Waukegan, IL 60085, U.S.A.
Phone 1-847-887-7000 Fax 1-847-887-7370

MOTOMAN INC. HEADQUARTERS
805 Liberty Lane West Carrollton, OH 45449, U.S.A.
Phone 1-937-847-6200 Fax 1-937-847-6277

YASKAWA ELÉTRICO DO BRASIL COMÉRCIO LTD.A.
Avenida Fagundes Filho, 620 Bairro Saude-Sao Pàulo-SP, Brazil CEP: 04304-000
Phone 55-11-5071-2552 Fax 55-11-5581-8795

YASKAWA ELECTRIC EUROPE GmbH
Am Kronberger Hang 2, 65824 Schwalbach, Germany
Phone 49-6196-569-300 Fax 49-6196-569-398

Motoman Robotics Europe AB
Box 504 S38525 Torsås, Sweden
Phone 46-486-48800 Fax 46-486-41410

Motoman Robotec GmbH
Kammerfeldstraβe1, 85391 Allershausen, Germany
Phone 49-8166-90-100 Fax 49-8166-90-103

YASKAWA ELECTRIC UK LTD.
1 Hunt Hill Orchardton Woods Cumbernauld, G68 9LF, United Kingdom
Phone 44-1236-735000 Fax 44-1236-458182

YASKAWA ELECTRIC KOREA CORPORATION
Kfpa Bldg #1201, 35-4 Youido-dong, Yeongdungpo-Ku, Seoul 150-010, Korea
Phone 82-2-784-7844 Fax 82-2-784-8495

YASKAWA ELECTRIC (SINGAPORE) PTE. LTD.
151 Lorong Chuan, #04-01, New Tech Park Singapore 556741, Singapore
Phone 65-6282-3003 Fax 65-6289-3003

YASKAWA ELECTRIC (SHANGHAI) CO., LTD.
4F No.18 Aona Road, Waigaoqiao Free Trade Zone, Pudong New Area, Shanghai 200131, China
Phone 86-21-5866-3470 Fax 86-21-5866-3869

YATEC ENGINEERING CORPORATION
4F., No.49 Wu Kong 6 Rd, Wu-Ku Industrial Park, Taipei, Taiwan
Phone 886-2-2298-3676 Fax 886-2-2298-3677

YASKAWA ELECTRIC (HK) COMPANY LIMITED
Rm. 2909-10, Hong Kong Plaza, 186-191 Connaught Road West, Hong Kong
Phone 852-2803-2385 Fax 852-2547-5773

BEIJING OFFICE
Room No. 301 Office Building of Beijing International Club, 21
Jianguomenwai Avenue, Beijing 100020, China
Phone 86-10-6532-1850 Fax 86-10-6532-1851

TAIPEI OFFICE
9F, 16, Nanking E. Rd., Sec. 3, Taipei, Taiwan
Phone 886-2-2502-5003 Fax 886-2-2505-1280

SHANGHAI YASKAWA-TONGJI M & E CO., LTD.
27 Hui He Road Shanghai China 200437
Phone 86-21-6553-6060 Fax 86-21-5588-1190

BEIJING YASKAWA BEIKE AUTOMATION ENGINEERING CO., LTD.
30 Xue Yuan Road, Haidian, Beijing P.R. China Post Code: 100083
Phone 86-10-6233-2782 Fax 86-10-6232-1536

SHOUGANG MOTOMAN ROBOT CO., LTD.
7, Yongchang-North Street, Beijing Economic Technological Investment & Development Area,
Beijing 100076, P.R. China
Phone 86-10-6788-0551 Fax 86-10-6788-2878

YASKAWA ELECTRIC CORPORATION

YASKAWA

In the event that the end user of this product is to be the military and said product is to be employed in any weapons systems or the manufacture thereof, the export will fall under the relevant regulations as stipulated in the Foreign Exchange and Foreign Trade Regulations. Therefore, be sure to follow all procedures and submit all relevant documentation according to any and all rules, regulations and laws that may apply.

Specifications are subject to change without notice
for ongoing product modifications and improvements.

MANUAL NO. SIEZ-C825-20.52C
© Printed in Japan September 2002 96-3 ◇7
02-5①
95-81192, 95-C82-031