CV500-MC221/421
Motion Control Unit

Operation Manual:
Details

Produced June 1995

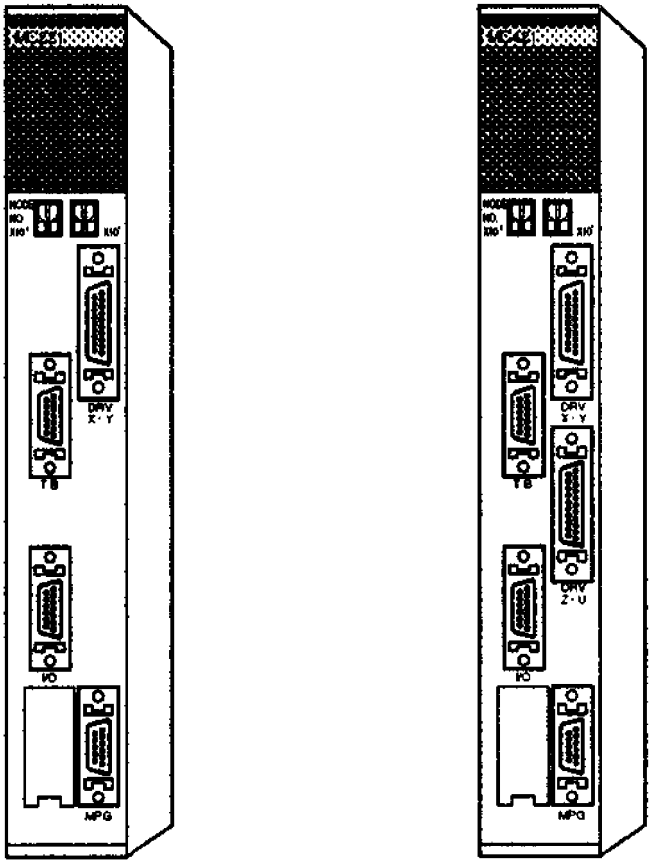


TABLE OF CONTENTS

TABLE OF CONTENTS

About this Manual:

This manual provides details on the features and operation of the CV500-MC221 and CV500-MC421 Motion Control Units and includes the sections described below.

Please read this manual and the other manuals related to the CV500-MC221 and CV500-MC421 Motion Control Units carefully and be sure you understand the information provided before attempting to install and operate the Motion Control Units.

There are four manuals used with the CV500-MC221 and CV500-MC421 Motion Control Units (MC Units). These manuals are listed in the following table. The suffixes have been left off of the catalog numbers. Be sure you are using the most recent version for your area.

Name	Content	Cat. No.
CV500-MC221/MC421 Motion Control Unit Operation Manual: Introduction	Describes the features, applications, and basic operation of the Motion Control Units. Read this manual first before using a Motion Control Unit.	W254
CV500-MC221/MC421 Motion Control Unit Operation Manual: Details	Describes the operation of the Motion Control Units in detail. Read the *Operation Manual: Introduction*, above, before attempting to read this manual.	W255
CVM1-PRS71 Teaching Box Operation Manual	Describes the operation of the Teaching Box connected to a Motion Control Unit.	W257
CV500-ZN3PC1 MC Support Software Operation Manual	Describes creating control programs and setting operating parameters for MC Units using the MC Support Software.	W256

Section 1 shows the main components of the Motion Control Unit, explains its indicators, unit number and DIP switch settings, and describes how to install the Unit.

Section 2 describes how to connect to the I/O, DRV, and MPG connectors and shows the interface circuits for all of the connectors.

Section 3 explains how to connect a personal computer or a Teaching Box.

Section 4 describes the three kinds of data (system parameters, position data, and registers) used in MC Unit functions.

Section 5 describes the functions in the G language.

Section 6 describes the interface area used to exchange information between the PC and MC Unit, such as commands from the PC and status information from the MC Unit.

Section 7 describes how to fix the location of the origin used as a reference point in positioning operations.

Section 8 describes teaching, which reads each axis' current position as position data.

Section 9 provides sample motion control programs written in G language.

Section 10 describes the errors that might occur during operation, their probable causes, and possible remedies.

Section 11 describes the maintenance and inspection necessary to ensure proper operation of the MC Unit.

The **Appendices** provide timing charts that show the operation of control bits and flags when the MC Unit is operated in manual or automatic mode, additional origin search patterns, program coding sheet, parameter settings sheet, and position data coding sheet.

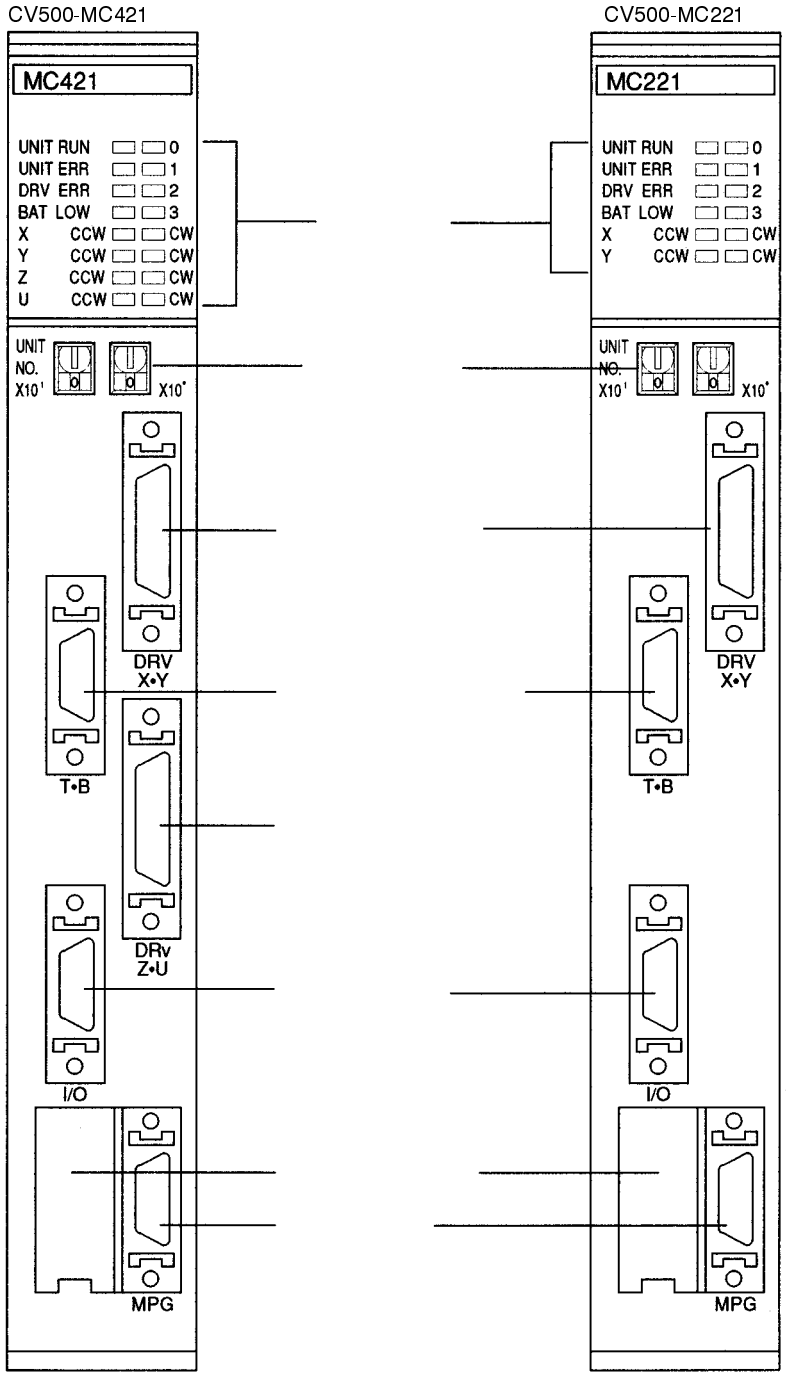
⚠WARNING Failure to read and understand the information provided in this manual may result in personal injury or death, damage to the product, or product failure. Please read each section in its entirety and be sure you understand the information provided in the section and related sections before attempting any of the procedures or operations given.

SECTION 1
Components and Switch Settings

This section shows the main components of the Motion Control Unit, explains its indicators, unit number and DIP switch settings, and describes how to install the Unit.

1-1 Components

The following diagram shows the main components of the CV500-MC221 and CV500-MC421 Motion Control Units (MC Units). The only difference between the Units is that the CV500-MC421 has connectors and indicators for U and Z axes.



Note Refer to *Section 2 Wiring* for details on wiring the MC Unit.

1-2 Indicators

The following table shows the meaning of the indicators on the front of the Unit.

Indicator	Color	Status	Meaning
UNIT RUN	Green	ON	Initialization was completed normally.
		OFF	A power interruption occurred, the Unit was reset from the PC, or the MC Unit is faulty.
UNIT ERR	Red	ON	An error occurred in the MC Unit.
		OFF	The MC Unit is operating normally.
DRV ERR	Red	ON	An error occurred in the servodriver.
		OFF	The servodriver is operating normally.
BAT LOW	Red	ON	The battery voltage has dropped.
		OFF	The battery voltage is normal.
X, Y, Z, or U CCW (X or Y for MC221)	Orange	ON	The motor is rotating counterclockwise.
		OFF	The motor is stopped or rotating counterclockwise.
X, Y, Z, or U CW (X or Y for MC221)	Orange	ON	The motor is rotating clockwise.
		OFF	The motor is stopped or rotating clockwise.

Note Indicators 0 through 3 provide details on the error when the UNIT ERR or DRV ERR indicator is ON. Refer to *10-1 Error Indicators* for details.

1-3 Setting the Unit Number

The unit number setting determines which CIO Area and DM Area words are allocated to the MC Unit. These words are used to transfer data between the PC and MC Unit and are known as the “PC data area interface.”

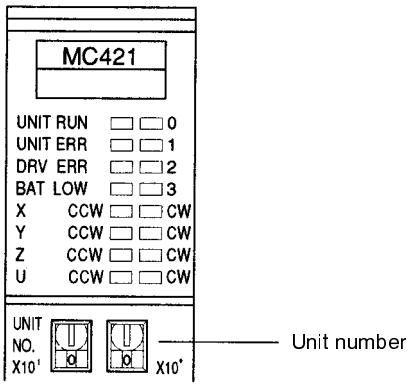
Allocated Words

There are 25 words allocated in the CIO Area and 100 words allocated in the DM Area. The words in the PC data area interface can be calculated from the formulae shown in the following table. Refer to *Section 6 PC Data Area Interface* for more details.

Unit number	CIO words	DM words
n (0 to 15)	CIO 1500 + 25 × n to CIO 1524 + 25 × n	D02000 + 100 × n to D02099 + 100 × n

Unit Number Setting

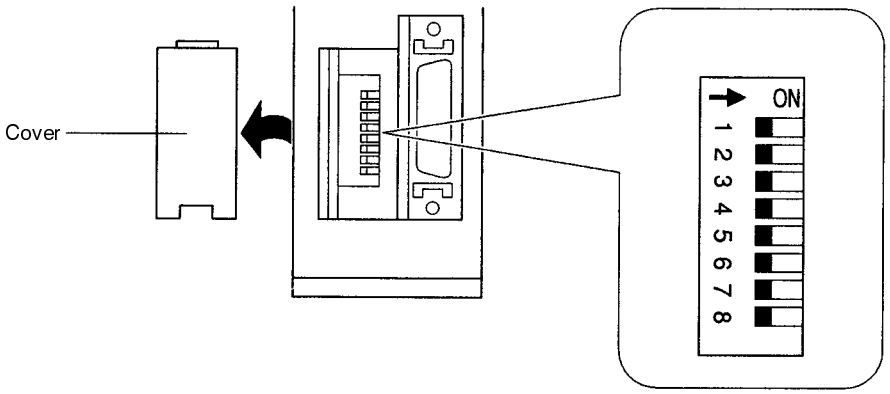
The unit number can be set from 0 to 15. The left switch is the 10s digit (0 or 1) and the right switch is the 1s digit.



Note
1. If the unit number is greater than 15, an error (invalid unit number error) will occur when the power is turned ON.
2. Be sure that none of the CPU Bus Units' unit numbers have been duplicated when connecting more than one CPU Bus Unit to a single PC.

1-4 Setting the DIP Switch

It may be necessary to set the DIP switch when using the Teaching Box and G01 (linear interpolation) function. Remove the cover from the front of the Unit to access the DIP switch, as shown in the following diagram.



DIP Switch Settings

Pin 2 of the DIP switch controls the language displayed on the Teaching Box, as shown in the following table.

Pin	Setting	Usage
1	OFF	Leave this pin set to OFF (the factory setting).
2	OFF	The Teaching Box will display Japanese text when this pin is OFF.
	ON	The Teaching Box will display English text when this pin is ON.
3 (See note)	OFF	If the designated linear interpolation speed exceeds the maximum interpolation feed speed (a value set via the system parameter) of the axis to be moved, linear interpolation is performed at the lowest maximum interpolation feed speed of the axes to be moved.
	ON	If the designated linear interpolation speed converted for each axis exceeds the maximum interpolation feed speed of the axis to be moved, linear interpolation is performed to move either one of the axes at the maximum interpolation feed speed.
4 to 8	---	Leave these pins set to OFF (the factory settings).

Note The ON setting of pin 3 will be effective on Units manufactured on and after July 11, 1995. As for Units manufactured before this date, only the OFF setting will be effective regardless of the ON/OFF setting.

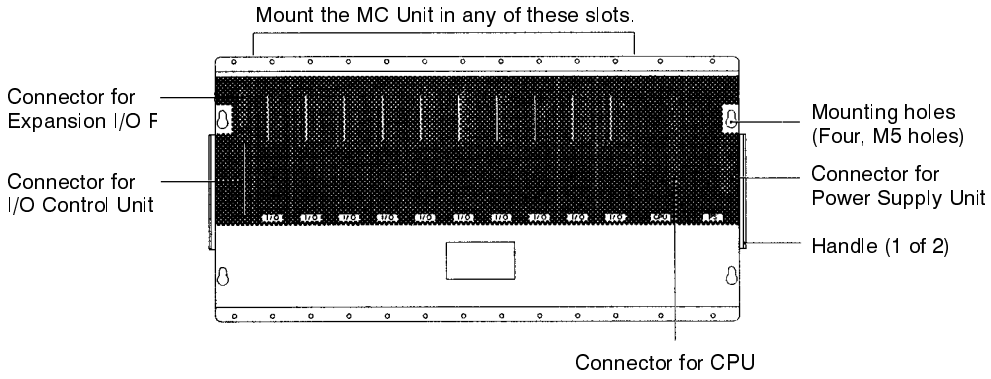
The Teaching Box display language selector (pin 2) is referenced only when the PC is turned ON. Changing the setting of pin 2 won't change the display language if the PC is already ON.

For a detailed description of interpolation operations through pin 3 settings, refer to page NO TAG.

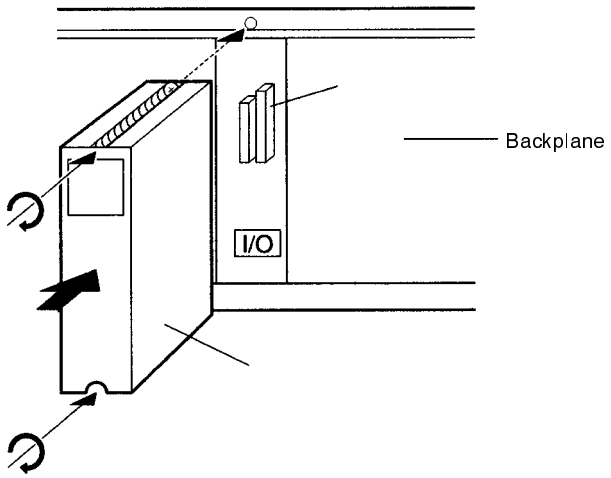
1-5 Installing the MC Unit

This section explains how to mount the MC Unit to a CPU Backplane. Before mounting the Unit, make sure that the CPU Backplane's Power Supply Unit is turned OFF.

The mounting position of the Power Supply Unit and CPU are fixed. The MC Unit can be mounted in any one of the remaining slots. The following example shows a CV500-BC101 Backplane.



Insert the Unit straight ahead, so that the connectors on the back of the Unit line up with those on the Backplane. Once the Unit has been mounted in the proper position, use a standard screwdriver to tighten the screws at the top and the bottom.



SECTION 2
Wiring

This section describes how to connect to the I/O, DRV, and MPG connectors and shows the interface circuits for all of the connectors.

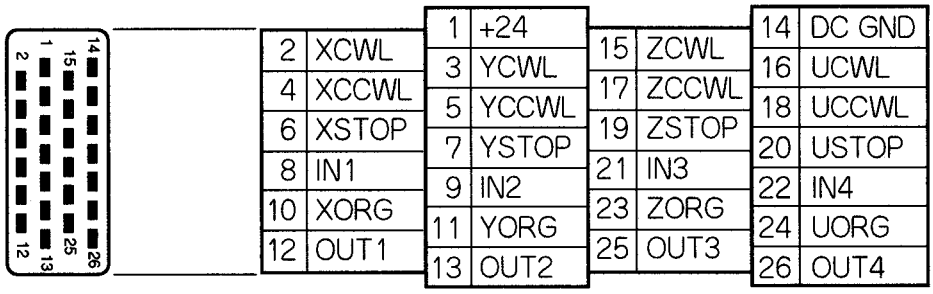
2-1 I/O Connector

The I/O connector is used primarily for wiring to external I/O. There are connections for each axis's CW and CCW limit inputs, stop inputs, and origin proximity inputs, as well as general I/O connections.

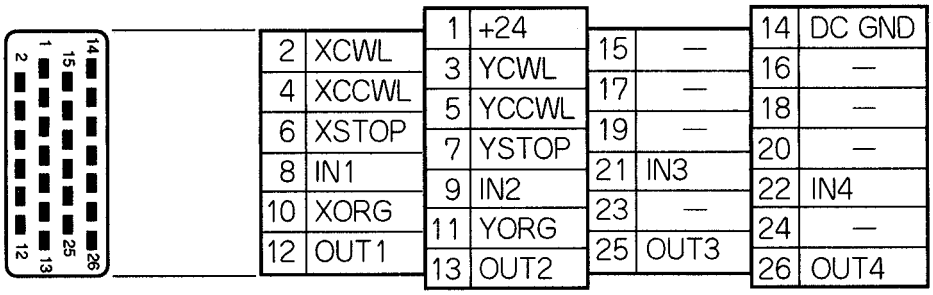
2-1-1 Pin Allocation

A Bellows 26-pin half-pitch plug and case are included with the Unit.

CV500-MC421 Pin Allocation The following diagram shows the pin allocation for the CV500-MC421.



CV500-MC221 Pin Allocation The following diagram shows the pin allocation for the CV500-MC221.



Pin Functions The following table explains the functions of the 26 pins in the I/O connector.

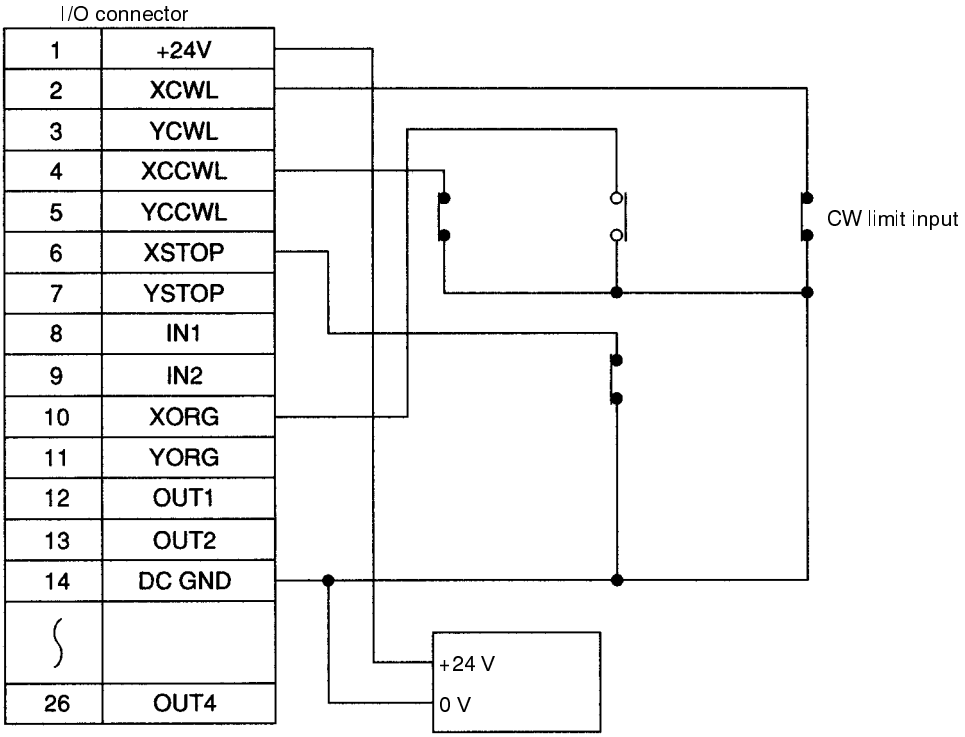
Pin	Symbol[1]	Name	Function
1	+24V	24 VDC input	Connects to the + terminal of the 24-VDC external power supply.
2	XCWL(NC)	X-axis CW limit input	Limits movement of the X-axis in the CW direction.
3	YCWL(NC)	Y-axis CW limit input	Limits movement of the Y-axis in the CW direction.
4	XCCWL(NC)	X-axis CCW limit input	Limits movement of the X-axis in the CCW direction.
5	YCCWL(NC)	Y-axis CCW limit input	Limits movement of the Y-axis in the CCW direction.
6	XSTOP(NC)	X-axis stop input	Invalidates the X-axis's run output and stops it.
7	YSTOP(NC)	Y-axis stop input	Invalidates the Y-axis's run output and stops it.
8	IN1(NO)	General input 1	General input 1
9	IN2(NO)	General input 2	General input 2
10	XORG(NC,NO)[2]	X-axis origin proximity input	Used for the X-axis origin search.
11	YORG(NC,NO)[2]	Y-axis origin proximity input	Used for the Y-axis origin search.
12	OUT1	General output 1	General output 1
13	OUT2	General output 2	General output 2
14	DC GND	24 VDC input ground	Connects to the – terminal (0 V) of the 24-VDC external power supply.
15	ZCWL(NC)[3]	Z-axis CW limit input	Limits movement of the Z-axis in the CW direction.
16	UCWL(NC)[3]	U-axis CW limit input	Limits movement of the U-axis in the CW direction.
17	ZCCWL(NC)[3]	Z-axis CCW limit input	Limits movement of the Z-axis in the CCW direction.
18	UCCWL(NC)[3]	U-axis CCW limit input	Limits movement of the U-axis in the CCW direction.

Pin	Symbol[1]	Name	Function
19	ZSTOP(NC)[3]	Z-axis stop input	Invalidates the Z-axis's run output and stops it.
20	USTOP(NC)[3]	U-axis stop input	Invalidates the U-axis's run output and stops it.
21	IN3(NO)	General input 3	General input 3
22	IN4(NO)	General input 4	General input 4
23	ZORG(NC,NO)[2,3]	Z-axis origin proximity input	Used for the Z-axis origin search.
24	UORG(NC,NO)[2,3]	U-axis origin proximity input	Used for the U-axis origin search.
25	OUT3	General output 3	General output 3
26	OUT4	General output 4	General output 4

Note
1. "NC" stands for normally closed and "NO" stands for normally open. Always short normally closed input terminals which aren't used.
2. Either NC or NO logic can be used. This setting is a mechanical parameter (Origin Proximity Input Logic) set with MC Support Software.
3. Pins 15 through 20, 23, and 24 aren't used in the CV500-MC221.

2-1-2 Connection Example

The following diagram shows an example connection for just the X-axis.



2-2 DRV Connectors

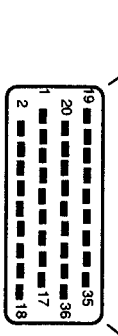
The DRV connectors are used primarily to connect servodrivers. The DRV X-Y connector is for the X and Y axes and the DRV Z-U connector is for the Z and U axes.

2-2-1 Pin Allocation

A Bellows 36-pin half-pitch connector and case are included with the Unit.

DRV X-Y Pin Allocation

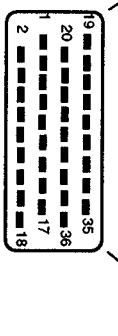
The following diagram shows the pin allocation for the DRV X-Y connector. Pins 6 through 9 and 24 through 27 aren't used.



2	DC GND	1	+24V	20	FDC GND	19	+F24V
4	XRUN	3	XALM	22	YRUN	21	YALM
6	—	5	XALMRS	24	—	23	YALMRS
8	—	7	—	26	—	25	—
10	X-GND	9	—	28	Y-GND	27	—
12	X-$\overline{A}$	11	X-A	30	Y-$\overline{A}$	29	Y-A
14	X-$\overline{B}$	13	X-B	32	Y-$\overline{B}$	31	Y-B
16	X-$\overline{Z}$	15	X-Z	34	Y-$\overline{Z}$	33	Y-Z
18	XAGND	17	XOUT	36	YAGND	35	YOUT

DRV Z-U Pin Allocation

The following diagram shows the pin allocation for the for the DRV Z-U connector. Pins 6 through 9 and 24 through 27 aren't used. (The CV500-MC221 isn't equipped with this connector.)



2	DC GND	1	+24V	20	FDC GND	19	+F24V
4	ZRUN	3	ZALM	22	URUN	21	UALM
6	—	5	ZALMRS	24	—	23	UALMRS
8	—	7	—	26	—	25	—
10	Z-GND	9	—	28	U-GND	27	—
12	Z-$\overline{A}$	11	Z-A	30	U-$\overline{A}$	29	U-A
14	Z-$\overline{B}$	13	Z-B	32	U-$\overline{B}$	31	U-B
16	Z-$\overline{Z}$	15	Z-Z	34	U-$\overline{Z}$	33	U-Z
18	ZAGND	17	ZOUT	36	UAGND	35	UOUT

DRV X-Y Pin Functions The following table explains the functions of the pins in the DRV X-Y connector.

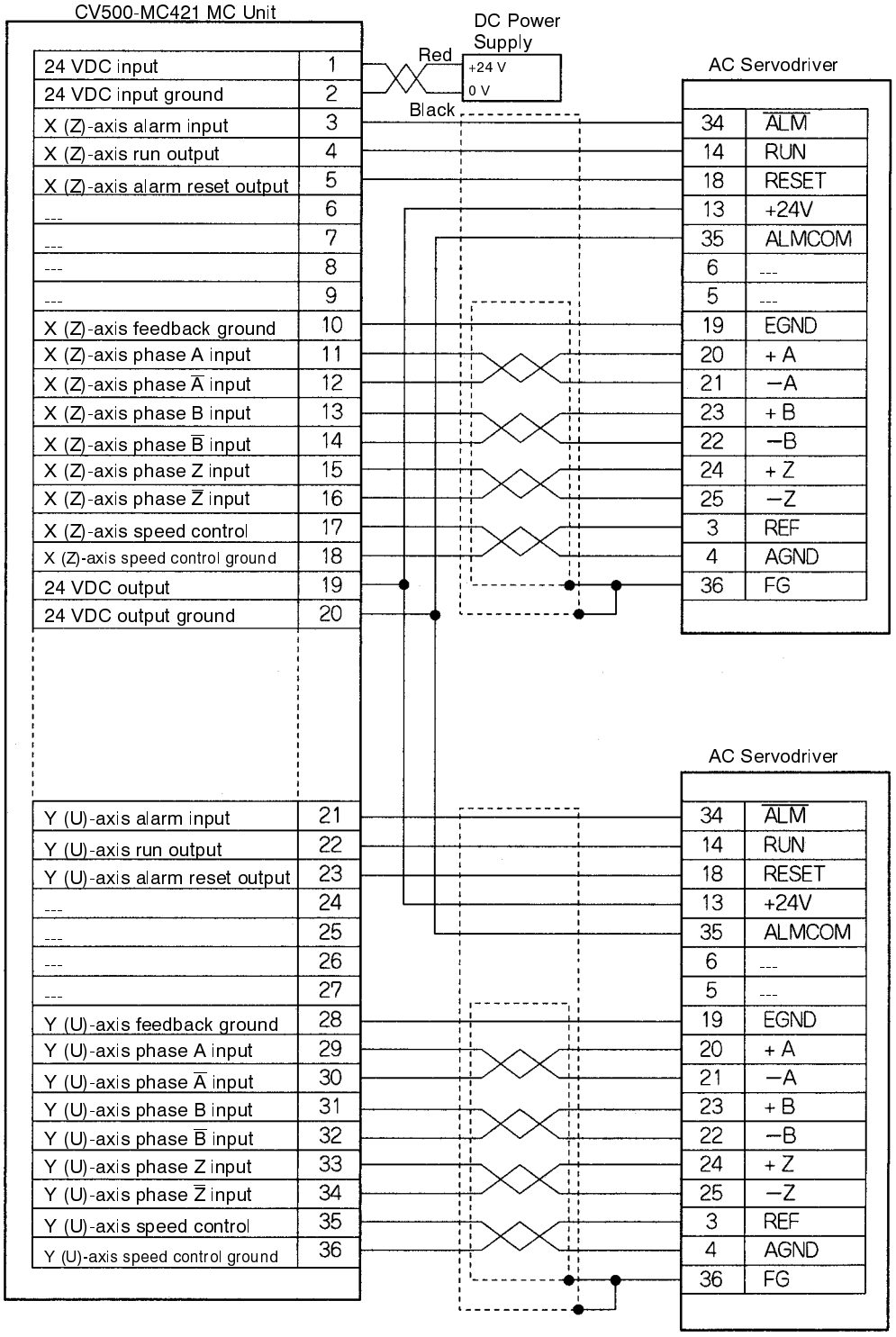
Pin	Symbol	Name	Function
1	+24V	24 VDC input	External power supply's 24-VDC input (for the X-Y axes).
2	DC GND	24 VDC input ground	External power supply's 24-VDC ground (for the X-Y axes).
3	XALM	X-axis alarm input	Driver alarm input for the X-axis
4	XRUN	X-axis run output	Driver run output for the X-axis
5	XALMRS	X-axis alarm reset output	Reset output for the X-axis's driver alarm.
6 to 9	---	---	Not used.
10	X-GND	X-axis feedback ground	Feedback ground for the X-axis
11	X-A	X-axis phase A input	Phase A feedback input for the X-axis
12	X-$\overline{A}$	X-axis phase $\overline{A}$ input	Phase $\overline{A}$ feedback input for the X-axis
13	X-B	X-axis phase B input	Phase B feedback input for the X-axis
14	X-$\overline{B}$	X-axis phase $\overline{B}$ input	Phase $\overline{B}$ feedback input for the X-axis
15	X-Z	X-axis phase Z input	Phase Z feedback input for the X-axis
16	X-$\overline{Z}$	X-axis phase $\overline{Z}$ input	Phase $\overline{Z}$ feedback input for the X-axis
17	XOUT	X-axis speed control	Speed control voltage to the X-axis driver
18	XAGND	X-axis speed control ground	Ground for the X-axis's speed control voltage
19	+F24V	24 VDC output	Supplies 24-VDC input to the driver (for the X-Y axes).
20	FDC GND	24 VDC output ground	Ground for 24-VDC outputs (for the X-Y axes).
21	YALM	Y-axis alarm input	Driver alarm input for the Y-axis
22	YRUN	Y-axis run output	Driver run output for the Y-axis
23	YALMRS	Y-axis alarm reset output	Reset output for the Y-axis's driver alarm.
24 to 27	---	---	Not used.
28	Y-GND	Y-axis feedback ground	Feedback ground for the Y-axis
29	Y-A	Y-axis phase A input	Phase A feedback input for the Y-axis
30	Y-$\overline{A}$	Y-axis phase $\overline{A}$ input	Phase $\overline{A}$ feedback input for the Y-axis
31	Y-B	Y-axis phase B input	Phase B feedback input for the Y-axis
32	Y-$\overline{B}$	Y-axis phase $\overline{B}$ input	Phase $\overline{B}$ feedback input for the Y-axis
33	Y-Z	Y-axis phase Z input	Phase Z feedback input for the Y-axis
34	Y-$\overline{Z}$	Y-axis phase $\overline{Z}$ input	Phase $\overline{Z}$ feedback input for the Y-axis
35	YOUT	Y-axis speed control	Speed control voltage to the Y-axis driver
36	YAGND	Y-axis speed control ground	Ground for the Y-axis's speed control voltage

DRV Z-U Pin Functions The following table explains the functions of the pins in the DRV Z-U connector. This connector is not available on the CV500-MC221.

Pin	Symbol	Name	Function
1	+24V	24 VDC input	External power supply's 24-VDC input (for the Z-U axes).
2	DC GND	24 VDC input ground	External power supply's 24-VDC ground (for the Z-U axes).
3	ZALM	Z-axis alarm input	Driver alarm input for the Z-axis
4	ZRUN	Z-axis run output	Driver run output for the Z-axis
5	ZALMRS	Z-axis alarm reset output	Reset output for the Z-axis's driver alarm.
6 to 9	---	---	Not used.
10	Z-GND	Z-axis feedback ground	Feedback ground for the Z-axis
11	Z-A	Z-axis phase A input	Phase A feedback input for the Z-axis
12	Z-$\overline{A}$	Z-axis phase $\overline{A}$ input	Phase $\overline{A}$ feedback input for the Z-axis
13	Z-B	Z-axis phase B input	Phase B feedback input for the Z-axis
14	Z-$\overline{B}$	Z-axis phase $\overline{B}$ input	Phase $\overline{B}$ feedback input for the Z-axis
15	Z-Z	Z-axis phase Z input	Phase Z feedback input for the Z-axis
16	Z-$\overline{Z}$	Z-axis phase $\overline{Z}$ input	Phase $\overline{Z}$ feedback input for the Z-axis
17	ZOUT	Z-axis speed control	Speed control voltage to the Z-axis driver
18	ZAGND	Z-axis speed control ground	Ground for the Z-axis's speed control voltage
19	+F24V	24 VDC output	Supplies 24-VDC input to the driver (for the Z-U axes).
20	FDC GND	24 VDC output ground	Ground for 24-VDC outputs (for the Z-U axes).
21	UALM	U-axis alarm input	Driver alarm input for the U-axis
22	URUN	U-axis run output	Driver run output for the U-axis
23	UALMRS	U-axis alarm reset output	Reset output for the U-axis's driver alarm.
24 to 27	---	---	Not used.
28	U-GND	U-axis feedback ground	Feedback ground for the U-axis
29	U-A	U-axis phase A input	Phase A feedback input for the U-axis
30	U-$\overline{A}$	U-axis phase $\overline{A}$ input	Phase $\overline{A}$ feedback input for the U-axis
31	U-B	U-axis phase B input	Phase B feedback input for the U-axis
32	U-$\overline{B}$	U-axis phase $\overline{B}$ input	Phase $\overline{B}$ feedback input for the U-axis
33	U-Z	U-axis phase Z input	Phase Z feedback input for the U-axis
34	U-$\overline{Z}$	U-axis phase $\overline{Z}$ input	Phase $\overline{Z}$ feedback input for the U-axis
35	UOUT	U-axis speed control	Speed control voltage to the U-axis driver
36	UAGND	U-axis speed control ground	Ground for the U-axis's speed control voltage

2-2-2 Connection Example

The following diagram shows an example connection for the CV500-MC421 when using an Incremental Encoder.

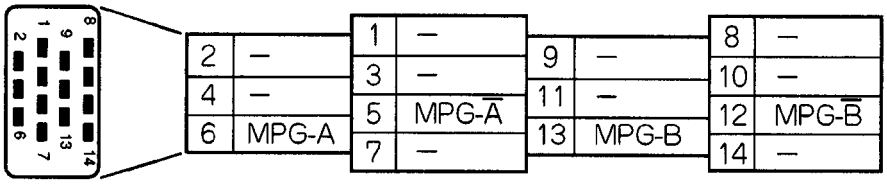


2-3 MPG Connector

The MPG connector is used to connect to the MPG (Manual Pulse Generator). Use a line-driver type of MPG.

2-3-1 Pin Allocation

A Sumitomo-3M 10114-3000VE connector and 10314-52A0-008 case are included with the Unit. The following diagram shows the pin allocation for the MPG connector. Pins 1 through 4, 7 through 11, and 14 aren't used.

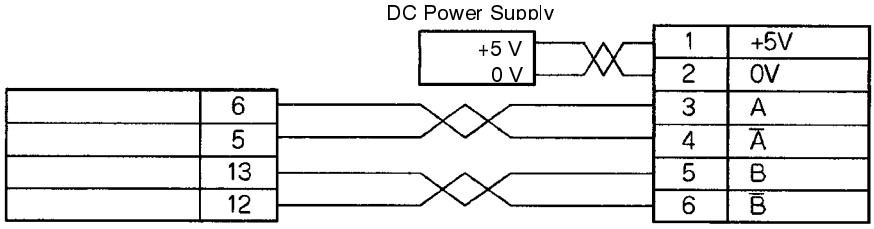


The following table explains the functions of the pins in the MPG connector.

Pin	Symbol	Name	Function
5	MPG-A	MPG phase Ā input	Phase Ā of the MPG input
6	MPG-Ā	MPG phase A input	Phase A of the MPG input
12	MPG-B̄	MPG phase B̄ input	Phase B̄ of the MPG input
13	MPG-B	MPG phase B input	Phase B of the MPG input

2-3-2 Connection Example

The following diagram shows an example connection for the MPG connector.



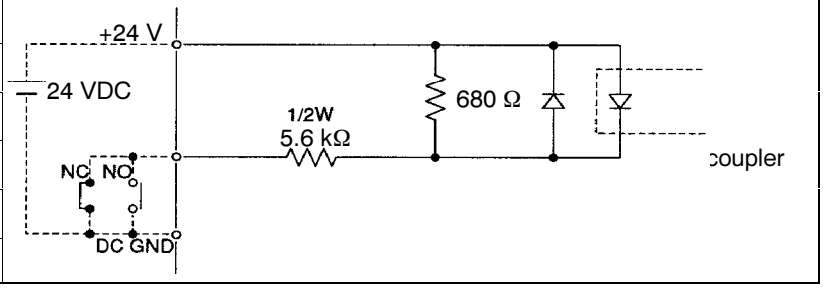
2-4 Interface Circuits

The following tables provide specifications and circuit diagrams for the interface circuits for the I/O, DRV, and MPG connectors. The CV500-MC221 isn't equipped with Z and U axes.

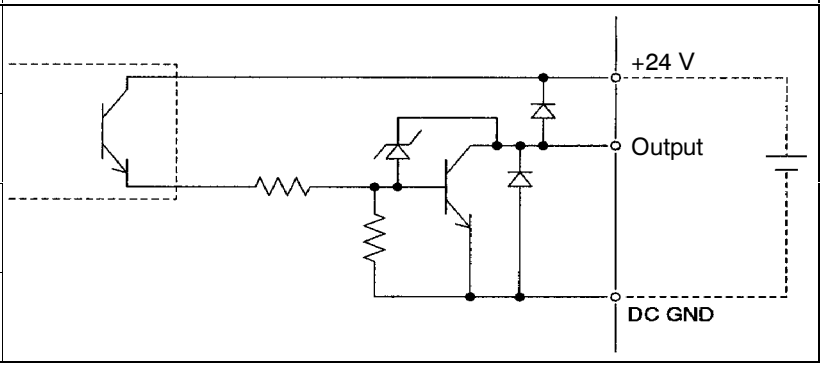
2-4-1 I/O and DRV Connector Circuits

The circuit in the table below is used to interface the following inputs.

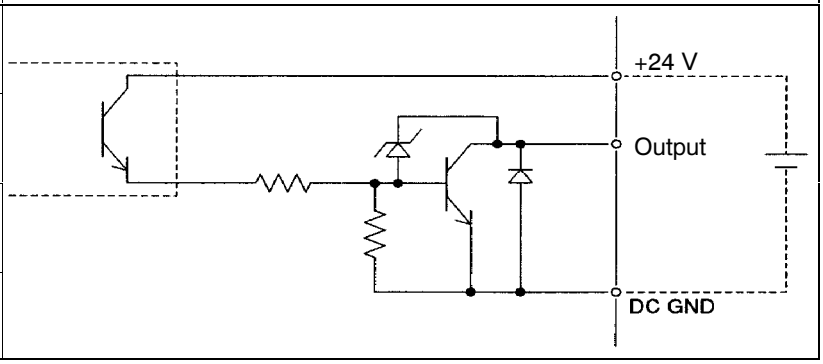
NC inputs: CWL, CCWL, STOP, and ALM (X, Y, Z, and U)
NO inputs: IN (1, 2, 3, and 4)
NC or NO inputs: ORG (X, Y, Z, and U)

Item	Specification	Circuit Configuration
Rated input voltage	24 VDC ± 10%	
Rated input current	4.3 mA	
ON voltage	17.4 V min.	
OFF voltage	3.0 V max.	
ON response time	1 ms max.	
OFF response time	2.5 ms max.	

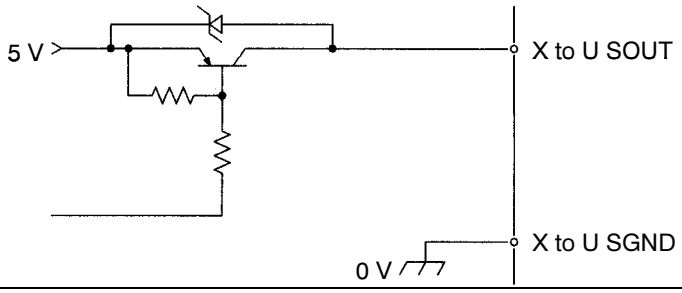
The circuit in the table below is used to interface outputs OUT1 through OUT4.

Item	Specification	Circuit Configuration
Max. switching capacity	50 mA/24 VDC	
Leakage current	0.1 mA max.	
Residual voltage	1.0 V max.	
External supply voltage	24 VDC ± 10%	

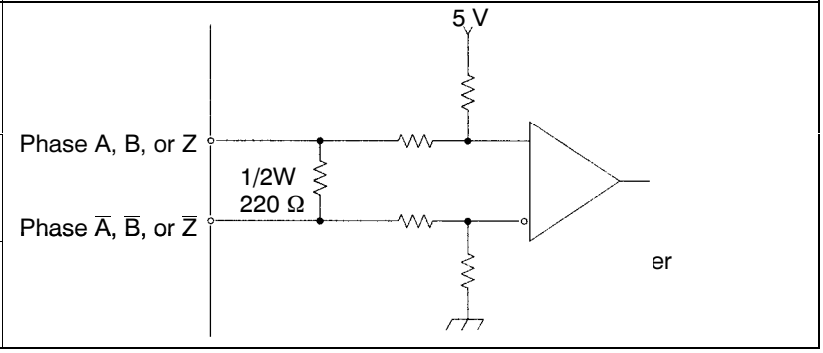
The circuit in the table below is used to interface outputs RUN (X, Y, Z, and U) and ALMRS (X, Y, Z, and U).

Item	Specification	Circuit Configuration
Max. switching capacity	50 mA/24 VDC	
Leakage current	0.1 mA max.	
Residual voltage	1.0 V max.	
External supply voltage	24 VDC ± 10%	

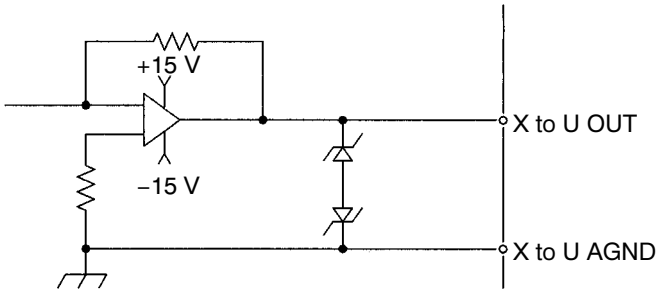
The circuit in the table below is used to interface outputs SOUT (X, Y, Z, and U).

Circuit Configuration


The circuit in the table below is used to interface phase inputs A, $\overline{A}$, B, $\overline{B}$, Z, and $\overline{Z}$ (for the X, Y, Z, and U axes).

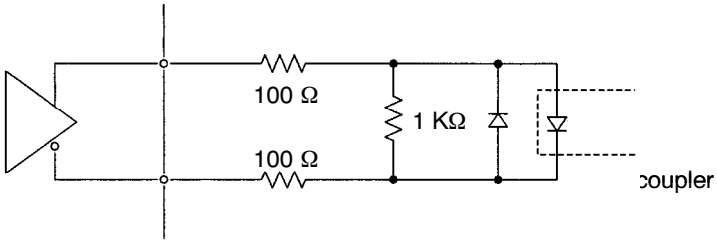
Item	Specification	Circuit Configuration
Signal level	EIA RS-422-A Standards	
Input impedance	220 Ω	
Response frequency	170 kpps max.	

The circuit in the table below is used to interface outputs OUT (X, Y, Z, and U).

Item	Specification	Circuit Configuration
Output voltage	0 to ± 10 V	
Load impedance	10 KΩ min.	

2-4-2 MPG Connector Circuits

The circuit in the table below is used to interface the inputs MPG-A, MPG-$\overline{A}$, MPG-B, and MPG-$\overline{B}$.

Item	Specification	Circuit Configuration
Rated input voltage	5 VDC ± 10%	
Rated input current	5 mA	
ON voltage	3 V min.	
Response frequency	50 kpps max.	

SECTION 3
Connecting Peripheral Devices

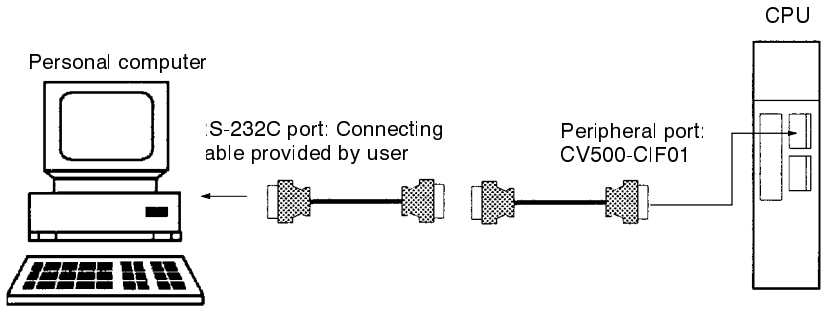
This section explains how to connect a personal computer or a Teaching Box.

3-1 Connecting a Personal Computer

This section explains how to connect a personal computer with MC Support Software to the PC. The personal computer can be connected to the PC's peripheral device connector or host interface connector.

3-1-1 Using the Peripheral Device Connector

Use a CV500-CIF01 Connecting Cable (sold separately) and an RS-232C serial cable to connect the personal computer to the peripheral device connector.



After setting the communications format, "Port," in the MC Support Software, set the communications baud rate. The baud rate is normally set to 9,600 bps. Refer to *17-2 Communications Format* in the *MC Support Software Operation Manual* for details on setting the communications format.

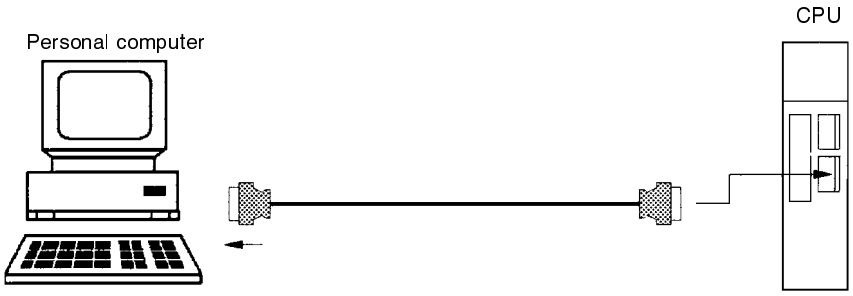
Once the communications baud rate has been set in the computer, set the PC to the same rate with the PC's DIP switch. The baud rate is set with pins 1 and 2, as shown in the following table.

Pin 1 setting	Pin 2 setting	Communications rate
OFF	OFF	50,000 bps
ON	OFF	19,200 bps
OFF	ON	9,600 bps
ON	ON	4,800 bps

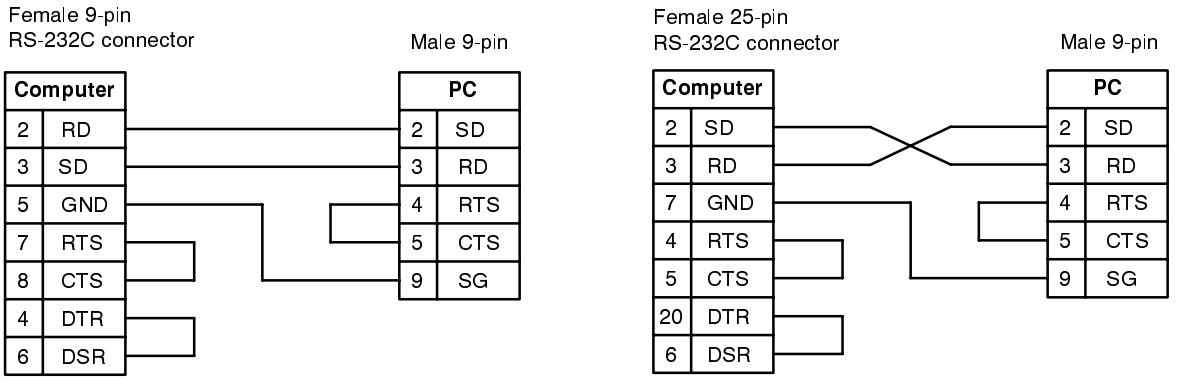
Note Refer to the PC's Operation Manual for more details on the CPU's DIP switch settings.

3-1-2 Using the Host Link Connector

Connect the computer to the PC through either a 9-pin or 25-pin RS-232C port on the computer and the 9-pin RS-232C host interface connector on the PC CPU.



The user must provide a cable that is wired for this connection as shown below. Turn OFF the computer and PC when the connecting the cables. The maximum cable length is 15 meters.



Set the communications format, "Host link," in the MC Support Software. The other protocol settings are normally set as shown in the following table. Refer to *17-2 Communications Format* in the *MC Support Software Operation Manual* for details on setting the communications format.

Parameter	Setting
Communications rate	9,600 bps
Unit number	00
Parity	Even parity
Data length	7 bits
Stop bits	2 bits
Response monitoring time	10 s

Note The PC's settings must match the computer's settings. Refer to the PC's Operation Manual for details on the PC's Host Link settings.

3-2 Connecting the Teaching Box

The procedure for connecting the Teaching Box to the MC Unit is described below.

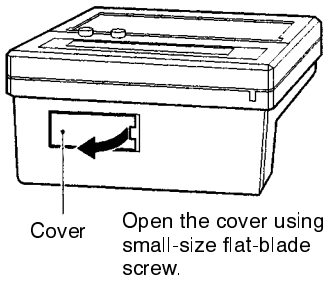
Connecting Cables

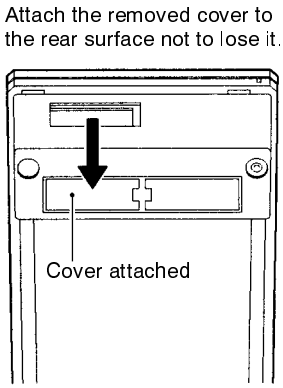
Use the Connecting Cables listed below (separately sold) to connect the Teaching Box to the MC Unit.

Model	Cable length
CV500-CN224	2 m
CV500-CN424	4 m
CV500-CN624	6 m

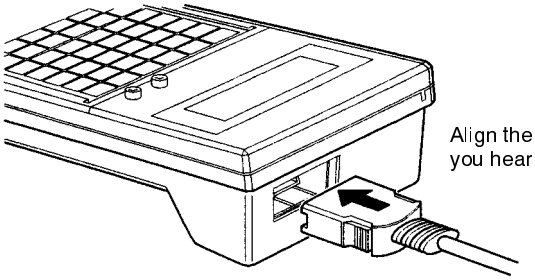
Connection Procedure Use the following procedure to connect the Teaching Box to the MC Unit.

1, 2, 3... 1. Remove the connector cover.

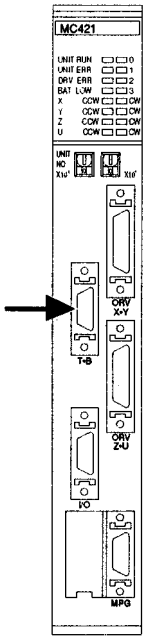




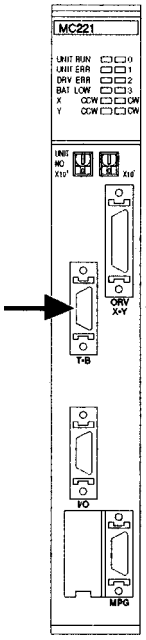
2. Plug the Connecting Cable into the connector.



3. Plug the Connecting Cable into the MC Unit connector marked “T.B.”



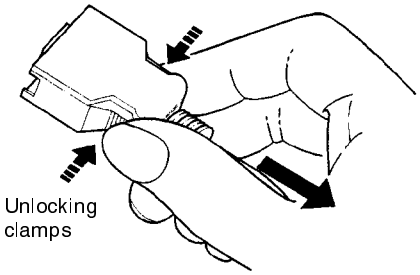
CV500-MC421



CV500-MC221

Removing the Cable

Using your fingers, press in and hold the clamps on both sides of the connector and pull out the connector.



SECTION 4
MC Unit Data

This section describes the three kinds of data (system parameters, position data, and registers) used in MC Unit functions.

4-1 MC Unit Data Configuration

The MC Unit handles three types of data: system parameters, position data, and registers. Programs written in G language aren't treated as data.

Data type	Function
System parameters	These parameters contain system information used in the MC Unit, such as the number of axes used, number of tasks, feed speed, and operating range.
Position data	There are 2,000 addresses that indicate position. In G language, these are identified as A0000 through A1999.
Registers	Registers are used as pointers to position data. There are 32 registers, identified as E00 through E31 in G language.

When the parameters and position data set with the MC Support Software are transferred to the MC Unit, they are stored in the system parameter and position data areas.

The parameters and position data used in the MC Unit can be set very easily with MC Support Software. Refer to the *MC Support Software Operation Manual* for details.

Some of the position data and system parameters (acceleration/deceleration time and interpolation acceleration/deceleration time) can be transferred to the MC Unit using the PC data area interface. Refer to *4-3 Data Transmission and Reception* for details.

4-2 System Parameters

There are three types of system parameters: unit parameters, memory management parameters, and axis parameters. The axis parameters are made up of 5 groups of parameters. The following table describes the function of each parameter.

Parameter name		Function
Unit parameters		These parameters specify information inherent to the Unit, such as the number of axes used and the number of tasks.
Memory management parameter		Specifies the range of the position data for teaching in each task.
Axis parameters (4 axes)	Mechanical parameters	Specifies mechanical and electrical system parameters such as the encoder resolution, pulse rate, and operating range.
	Coordinate system parameters	Specifies the offset values for the reference and workpiece coordinate systems.
	Feed speed parameters	Specifies feed-speed information, such as the maximum feed speed and maximum interpolation feed speed.
	Zone parameters	Specifies zone information.
	Servo parameters	Specifies servo system information such as the position loop gain and in-position.

Note With the CV500-MC221, set parameters for 2 axes only.

4-2-1 Setting Parameters

Set the parameters with the MC Support Software and then transfer them to the MC Unit. Only the acceleration/deceleration time and interpolation acceleration/deceleration time can be transmitted to the MC Unit (or received by the MC Unit) using the PC data area interface. Refer to *4-3 Data Transmission and Reception* for details.

Be sure to turn the MC Unit OFF and ON again after parameters have been transferred. The Unit parameters and mechanical parameters won't be changed unless the MC Unit is turned OFF and ON again.

The memory management, coordinate system, feed speed, zone, and servo parameters will be changed without turning the power OFF and ON. The following table shows when these parameters become effective.

Parameter		Effective
Coordinate system parameters	Reference origin offset	Becomes effective the next time that an origin search is performed.
	Workpiece origin offset	Becomes effective the next time that the program is executed from the beginning.
Feed speed parameters	Maximum feed speed Max. interpolation feed speed High-speed origin search speed Low-speed origin search speed Maximum JOG feed speed	Become effective the next operation.
	Acceleration time Deceleration time	Automatic operation: Become effective the next time that the program is executed from the beginning. Manual operation: Become effective the next operation.
	Interpolation acceleration time Interpolation deceleration time	Become effective the next time that the program is executed from the beginning.
Other parameters		Effective immediately

4-2-2 Parameter Lists

Memory Management Parameters

Parameter name	Function
Position data for task 1 (First and last addresses)	Sets the first and last addresses of the position data used for task 1.
Position data for task 2 (First and last addresses)	Sets the first and last addresses of the position data used for task 2.
Position data for task 3 (First and last addresses)	Sets the first and last addresses of the position data used for task 3. (CV500-MC421 only)
Position data for task 4 (First and last addresses)	Sets the first and last addresses of the position data used for task 4. (CV500-MC421 only)

Mechanical Parameters

Parameter name	Function
Minimum unit setting	Specify the minimum units for the mechanical system.
Display units	Specify the units when monitoring the PV: mm, inches, deg., or pulses.
Rotation direction	Specify whether the motor will rotate in forward or reverse when the control voltage to the servo-motor driver is positive (+).
Stop method	Specify whether the control voltage to the servodriver should drop to 0 V immediately or the accumulated pulses should be output before stopping when a stop input is received.
Encoder ABS/INC	Specify that the encoder used is the INC type (relative values).
Encoder resolution	Set the number of pulses output when the encoder rotates once.
Encoder polarity	Sets whether the motor is made to go forward or reverse when the feedback pulse from the encoder is increased.
Pulse rate	Specifies how much to move the axis per feedback pulse.
Max. motor frequency	Sets the maximum rpm rating for the motor being used.
Stroke limit (–) Stroke limit (+)	Sets the negative and positive stroke limits.
Origin search method	Specifies one of the 3 origin search methods.

Parameter name	Function
Origin search direction	Specifies whether to move in the positive or negative direction when performing an origin search.
Origin deceleration method	Specifies the input method when decelerating near the origin.
Origin proximity input logic	Specifies whether the origin proximity input is normally open or normally closed.
Wiring check	Specifies whether or not to perform a wiring check when power is turned ON.
Wiring check time	Sets the wiring check time.
Wiring check pulses	Sets the number of pulses used in the wiring check.
ABS encoder initial setting	The ABS encoder initial setting and soft reset value are displayed when it is executed. These parameters can't be set with MC Support Software.
ABS encoder soft reset value	

Coordinate System Parameters

Parameter name	Function
Reference origin offset value (X-axis)	Specifies the offset for the X-axis's reference origin.
Workpiece origin offset value (X-axis)	Specifies the offset for the X-axis's workpiece origin.
Reference origin offset value (Y-axis)	Specifies the offset for the Y-axis's reference origin.
Workpiece origin offset value (Y-axis)	Specifies the offset for the Y-axis's workpiece origin.
Reference origin offset value (Z-axis)	Specifies the offset for the Z-axis's reference origin. (CV500-MC421 only)
Workpiece origin offset value (Z-axis)	Specifies the offset for the Z-axis's workpiece origin. (CV500-MC421 only)
Reference origin offset value (U-axis)	Specifies the offset for the U-axis's reference origin. (CV500-MC421 only)
Workpiece origin offset value (U-axis)	Specifies the offset for the U-axis's workpiece origin. (CV500-MC421 only)

Feed Speed Parameters

Parameter name	Function
Maximum feed speed	Sets the maximum feed speed for each axis in PTP operation.
Maximum interpolation feed speed	Sets the maximum feed speed in interpolation operation.
High-speed origin search speed	When an origin search is executed, it is performed at this speed until an origin proximity sensor input is received.
Low-speed origin search speed	An origin search goes from high-speed to low-speed (specified with this parameter) when the origin proximity sensor input is received.
Maximum JOG feed speed	Sets the maximum JOG feed speed.
Acceleration/deceleration curve	Specifies whether the acceleration/deceleration curve is trapezoidal or S-shaped.
Acceleration time Interpolation acceleration time	Sets the time required to accelerate to the specified speed from the start of operation/interpolation operation.
Deceleration time Interpolation deceleration time	Sets the time required to decelerate from the specified speed to zero when stopping operation.
MPG factor	Sets the factor when using an MPG.

Zone Parameters

Parameter name	Function
Zone specification	Specifies whether or not a zone is set.
Zone –/+ direction set values	Sets the upper and lower limits of the zone's range.

Servo Parameters

Parameter name	Function
Accumulated pulses warning value	The error counter alarm flag will go ON if the number of accumulated pulses in the error counter exceeds this set value.
In-position	Set to check the number of accumulated pulses in the error counter.
Position loop gain	Sets the position loop gain.
Position loop FF gain	Sets the position loop FF gain.

4-3 Data Transmission and Reception

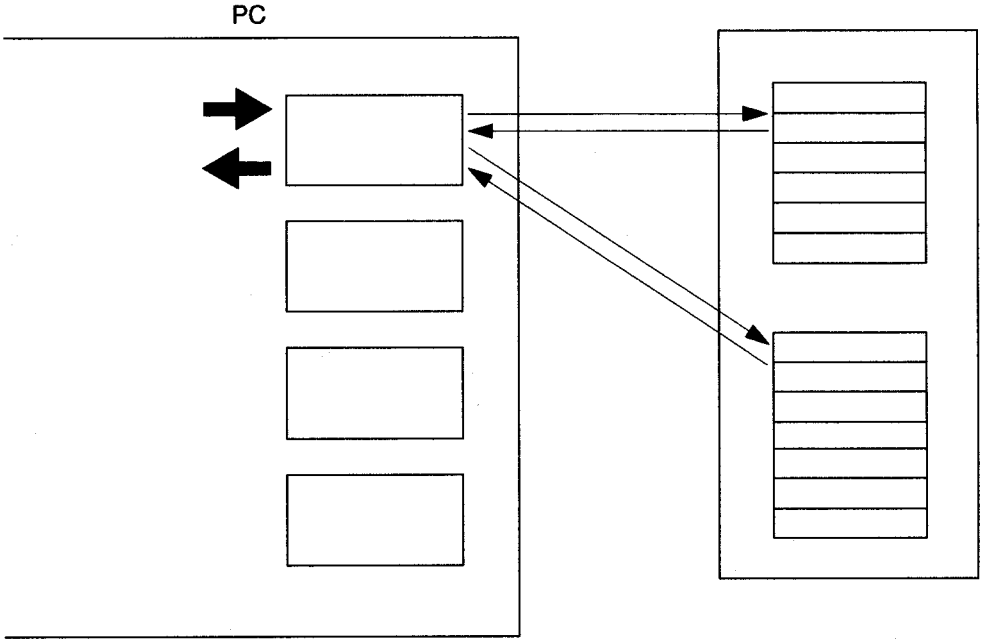
This section explains how to transfer position data and some system parameters between the ladder program and the MC Unit.

Summary

Transmission and reception is performed with four independent words. A single word can be used for transmission or reception, but not both at the same time.

One unit of data can be transferred in each transmission/reception. PC words 1 through 4 are referred to as "transmission (reception) #1" through "transmission (reception) #4" in the MC Units.

This example shows the use of transmission #1 and reception #1. The other words (transmission #2 through transmission #4 and reception #2 through reception #4) are used in the same way.



Transferrable Data

The following table lists the position data and system parameters that can be transferred between the PC and MC Unit.

Position data	System parameters
A0000 through A1999	X, Y, Z, and U-axis acceleration times
	X, Y, Z, and U-axis deceleration times
	X, Y, Z, and U-axis interpolation acceleration times
	X, Y, Z, and U-axis interpolation deceleration times

Note The CV500-MC221 does not have Z and U axes.

4-4 Information Used for Transmission/Reception

This section describes the data settings and flags used in data transmission and reception. Refer to *Section 6 PC Data Area Interface* for details on actual allocation.

4-4-1 Transmission/Reception Bits and Flags

The bits and flags used in data transmission and reception are described below. The bits and flags for transmission #1 and reception #1 are described here, but the other bits and flags operate in the same way. Refer to *Section 6 PC Data Area Interface* for details on actual allocation.

Transmission #1 Control Bit This bit is used to control transmission #1; it is allocated in the CIO Area of the PC data area interface. Data transmission begins when this bit goes ON.

Transmission #1 Completed Flag This flag is turned ON when transmission #1 is completed; it is allocated in the CIO Area of the PC data area interface. This flag goes ON when transmission #1 is completed and goes OFF when the transmission #1 control bit is turned OFF.

Reception #1 Control Bit This bit is used to control reception #1; it is allocated in the CIO Area of the PC data area interface. Data reception begins when this bit goes ON.

Reception #1 Completed Flag This flag is turned ON when reception #1 is completed; it is allocated in the CIO Area of the PC data area interface. This flag goes ON when reception #1 is completed and goes OFF when the reception #1 control bit is turned OFF.

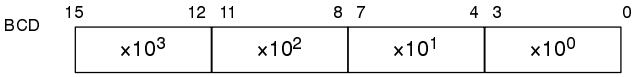
Transfer #1 Error Flag This flag is turned ON when an error occurs during execution of transmission #1 or reception #1 and the data transfer wasn't completed properly; it is allocated in the CIO Area of the PC data area interface.

This flag is turned ON along with the transmission #1 completed flag or reception #1 completed flag; it is turned OFF when the next #1 transfer is completed normally. control bit is turned OFF.

4-4-2 Required Data Settings

The settings described below are required for data transmission and reception. The settings for transmission #1 and reception #1 are described here.

Transfer Address #1 This address specifies the address of the transmission destination or reception source; it is allocated in the DM Area of the PC data area interface. The content of transfer address #1 is 4-digit BCD.



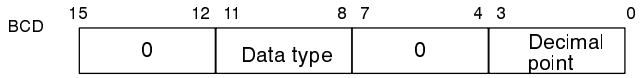
Values of 0000 through 1999 specify position data (A0000 through A1999). System parameters are specified as shown in the following table.

Transfer address #1	Specified system parameter
4106	X-axis acceleration time
4206	Y-axis acceleration time
4306	Z-axis acceleration time
4406	U-axis acceleration time
4107	X-axis deceleration time
4207	Y-axis deceleration time
4307	Z-axis deceleration time
4407	U-axis deceleration time
4108	X-axis interpolation acceleration time
4208	Y-axis interpolation acceleration time
4308	Z-axis interpolation acceleration time
4408	U-axis interpolation acceleration time
4109	X-axis interpolation deceleration time
4209	Y-axis interpolation deceleration time
4309	Z-axis interpolation deceleration time
4409	U-axis interpolation deceleration time

Note The CV500-MC221 does not have Z and U axes.

Transfer #1 Auxiliary Information

Theauxiliary information indicates the type of data to be transmitted/received as well as the location of the decimal point; it is allocated in the DM Area of the PC data area interface.



The data-type digit indicates whether the data is position data or a parameter.

0: Position data
1: Parameter

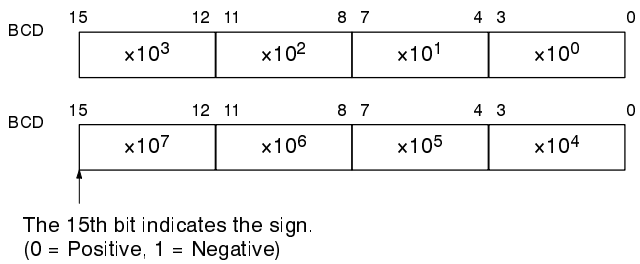
The decimal-point digit indicates the location of the decimal point.

0: There are no digits after the decimal point.
1: There is 1 digit after the decimal point.
2: There are 2 digits after the decimal point.
3: There are 3 digits after the decimal point.
4: There are 4 digits after the decimal point.

Note The decimal point setting is meaningless if the data type is set to parameter (1).

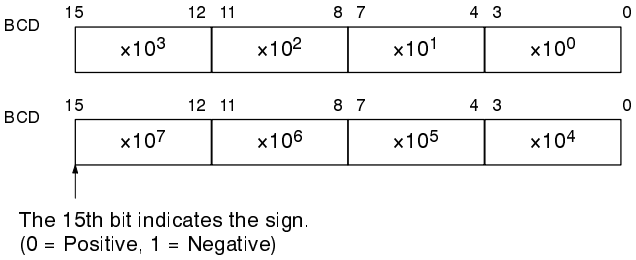
Transmission #1

These are the two words of data that will be transmitted; they are allocated in the DM Area of the PC data area interface.



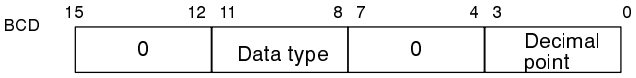
Reception #1

These two words contain the received data; they are allocated in the DM Area of the PC data area interface.



Reception #1 Auxiliary Information

This word contains auxiliary information about the received data; it is allocated in the DM Area of the PC data area interface.



The data-type digit indicates whether the data is position data or a parameter.

0: Position data
1: System parameter

The decimal-point digit indicates the location of the decimal point.

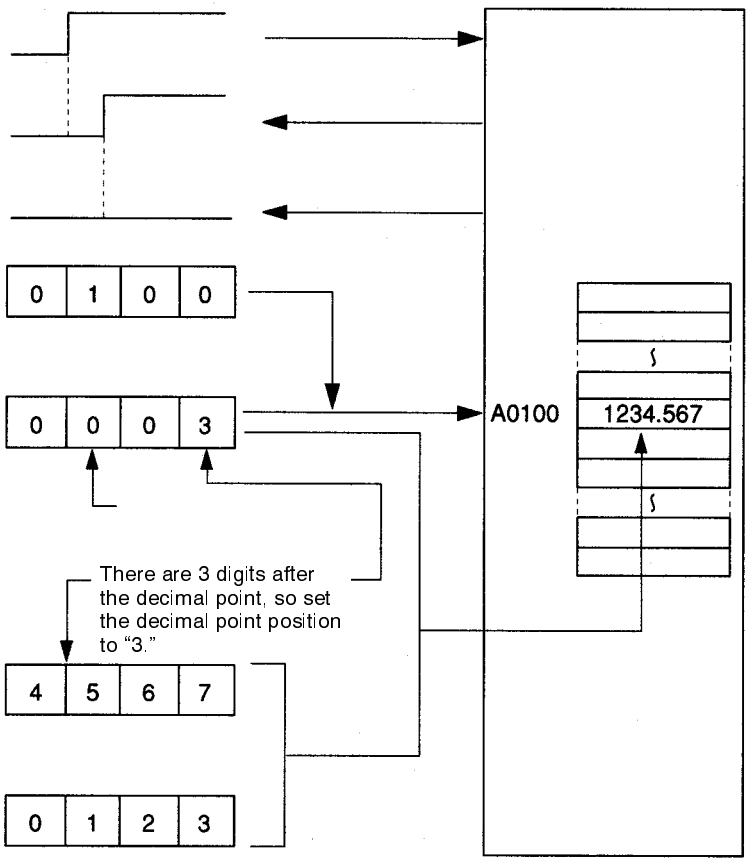
0: There are no digits after the decimal point.
1: There is 1 digit after the decimal point.
2: There are 2 digits after the decimal point.
3: There are 3 digits after the decimal point.
4: There are 4 digits after the decimal point.

4-5 Data Transmission

This section provides two examples of data transmission using transmission #1.

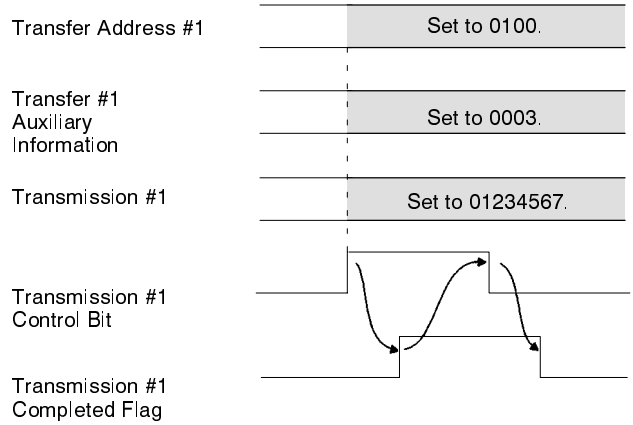
4-5-1 Transmitting Position Data

The following diagram shows how to transmit the value 1234.567 to position data address A0100.



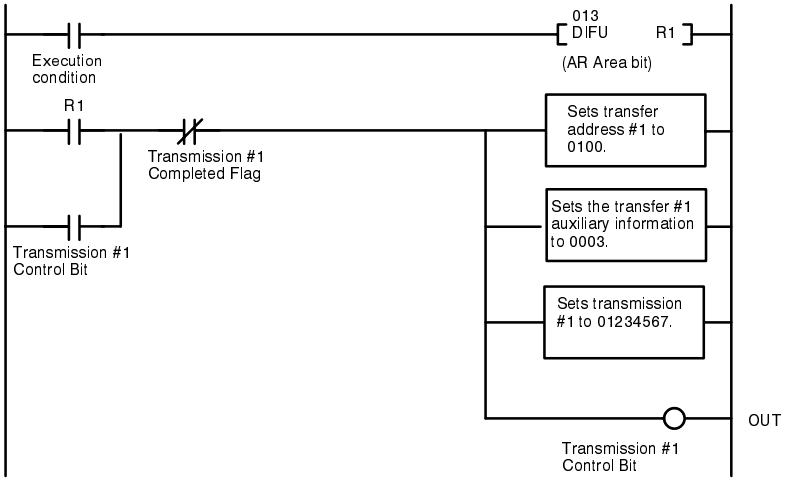
Timing Chart

The following diagram shows the timing of data changes during the transmission of 1234.567 to position data address A0100.



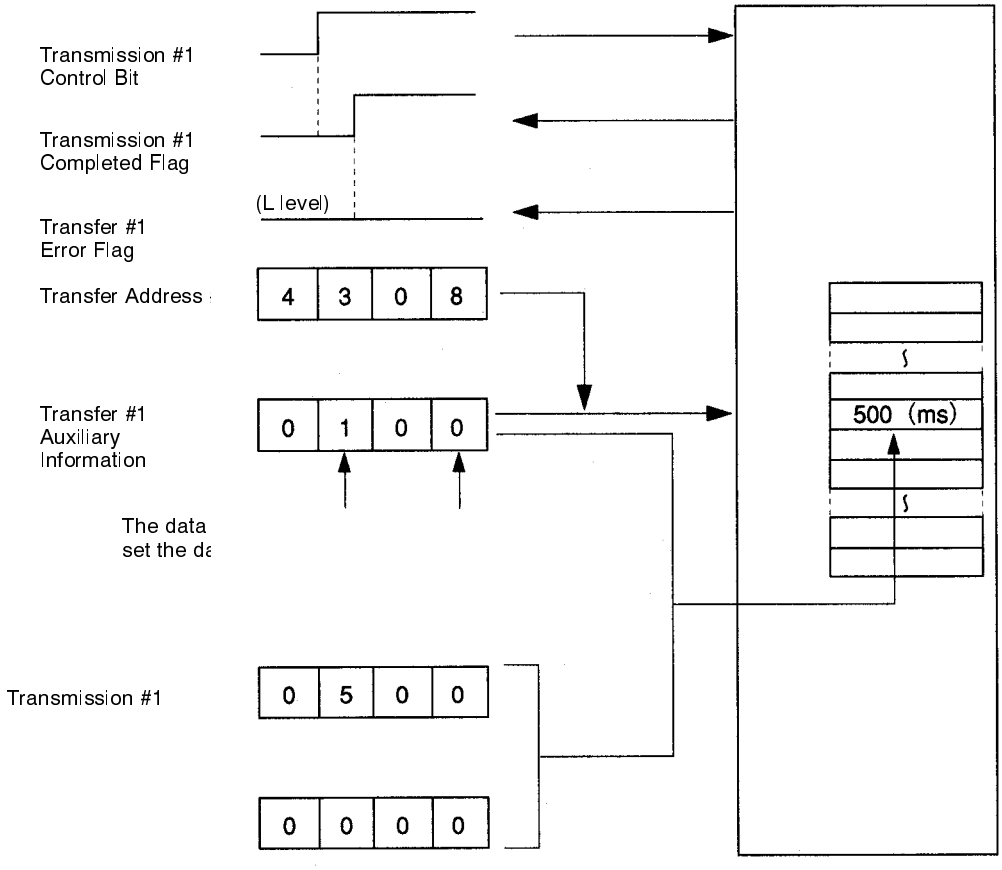
Ladder Program Example

The following diagram shows a ladder program example that transmits 1234.567 to position data address A0100.



4-5-2 Changing a System Parameter

The following diagram shows how to change the Z-axis interpolation acceleration time to 500 ms.

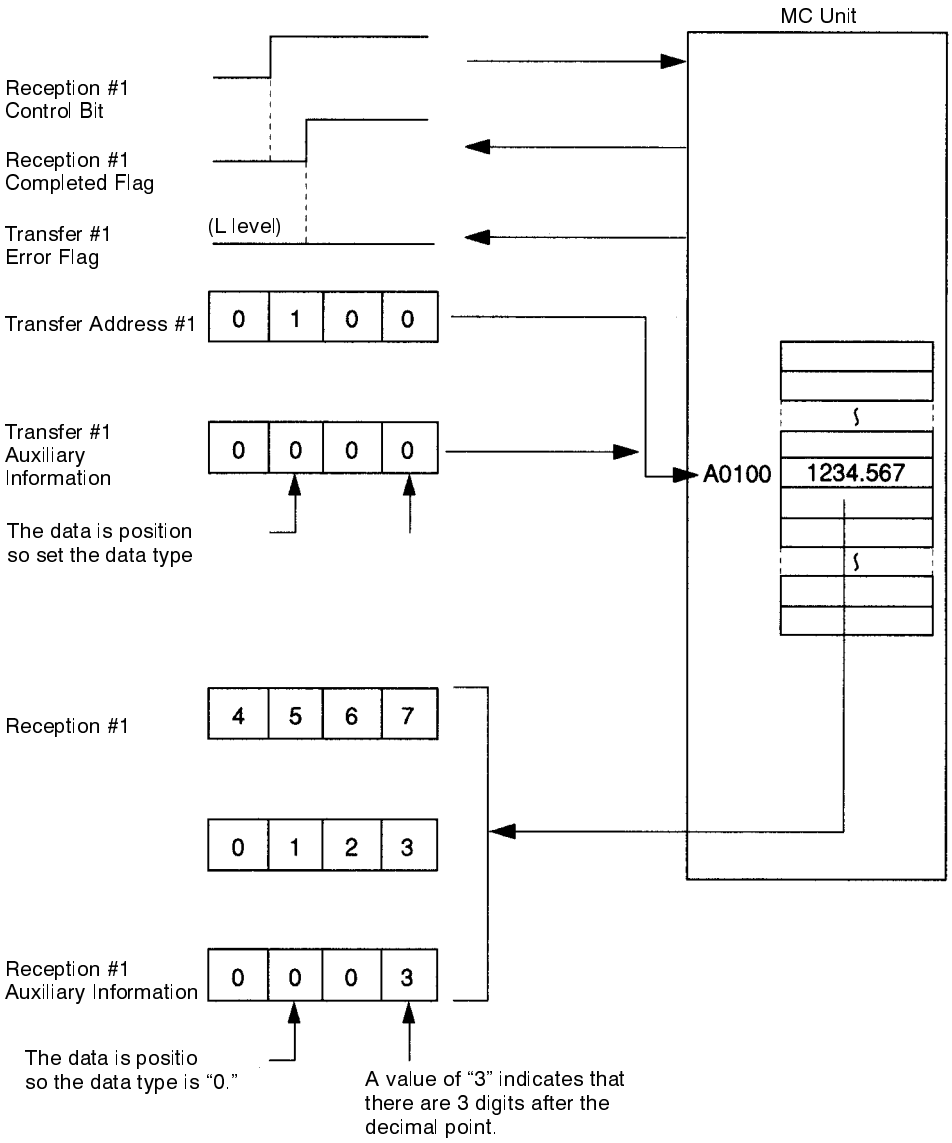


4-6 Data Reception

This section provides two examples of data reception using reception #1.

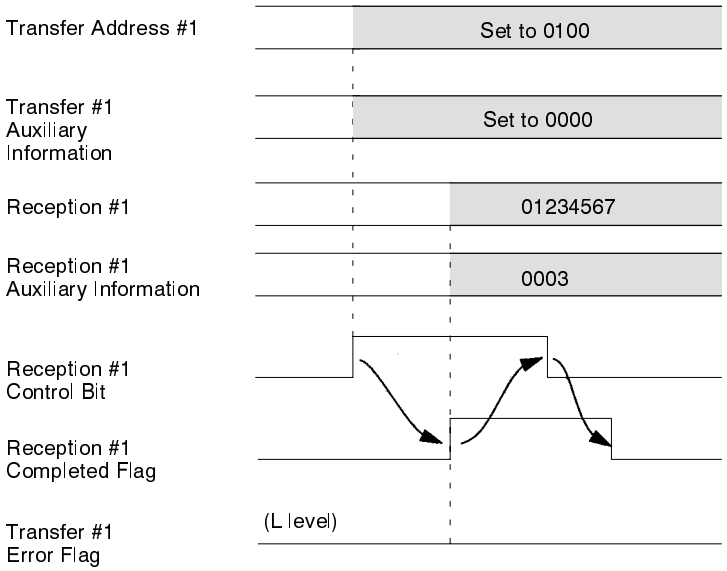
4-6-1 Receiving Position Data

The following diagram shows how to receive the contents of position data address A0100 (1234.567).



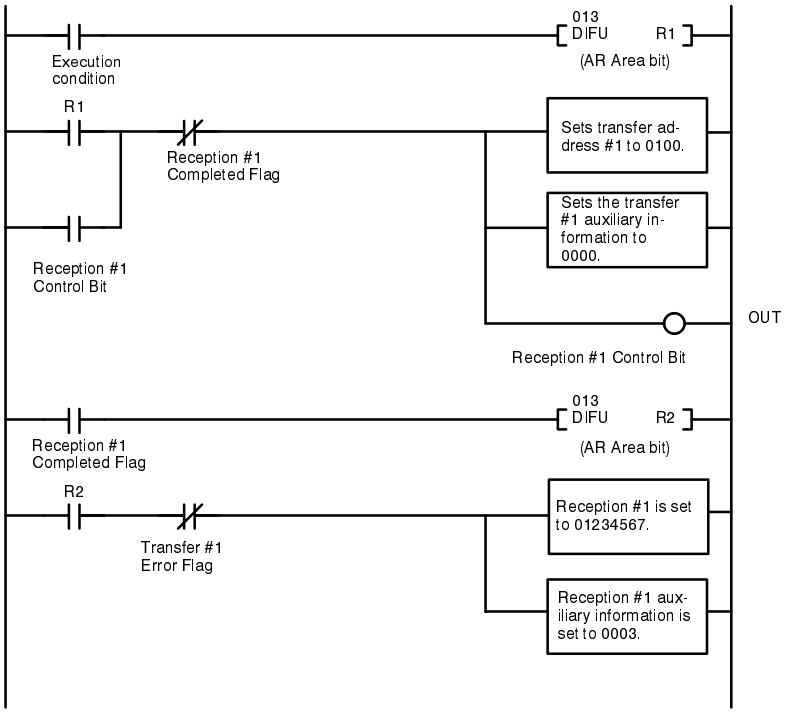
Timing Chart

The following diagram shows the timing of data changes during the reception of 1234.567 to position data address A0100.



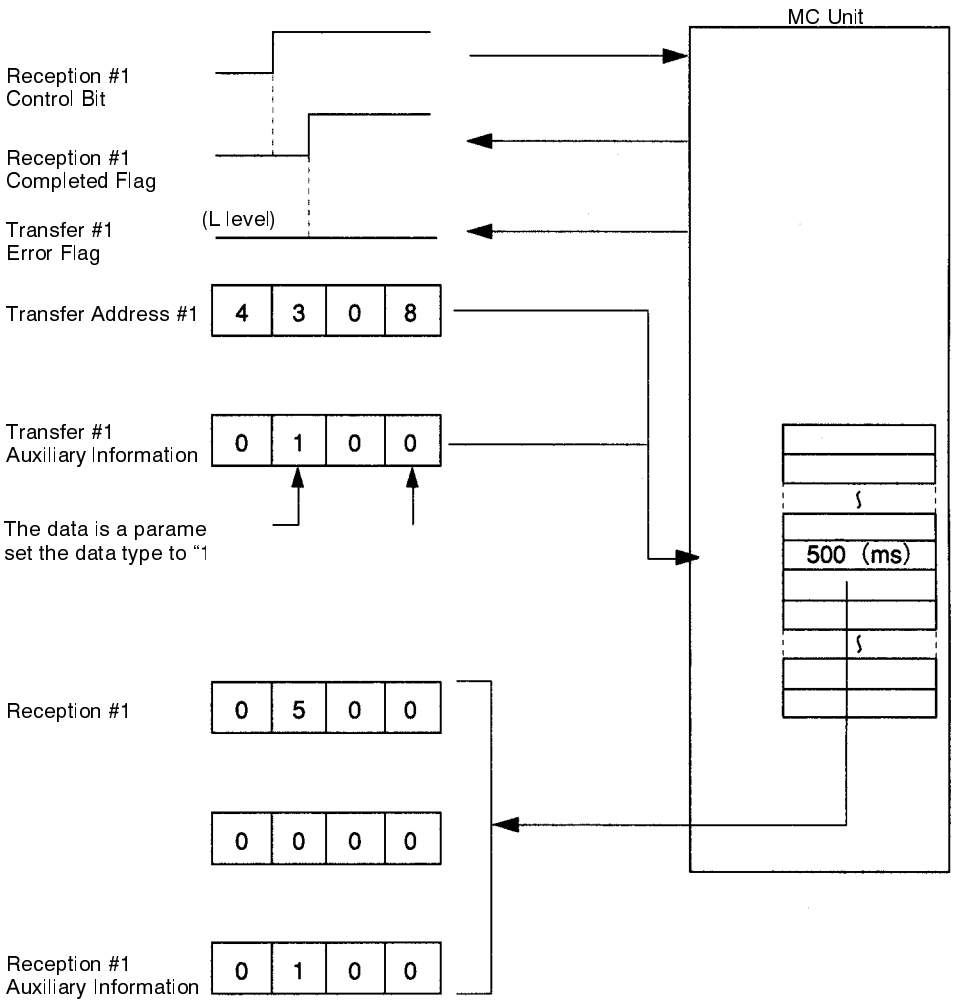
Ladder Program Example

The following diagram shows a ladder program example that transmits 1234.567 to position data address A0100.



4-6-2 Receiving a System Parameter's Contents

The following diagram shows how to receive the contents of the Z-axis interpolation acceleration time (500 ms).



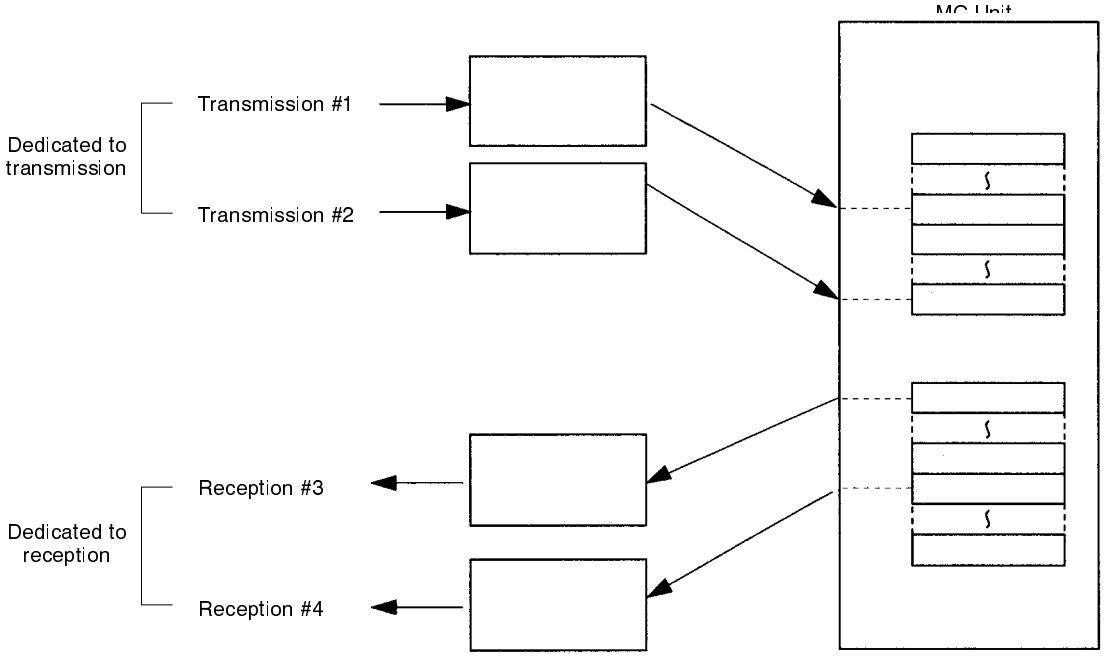
4-7 Precautions when Transferring Data

One word can't be used for transmission and reception at the same time because the same addresses and auxiliary information are used for both transmission and reception.

Dedicating Words

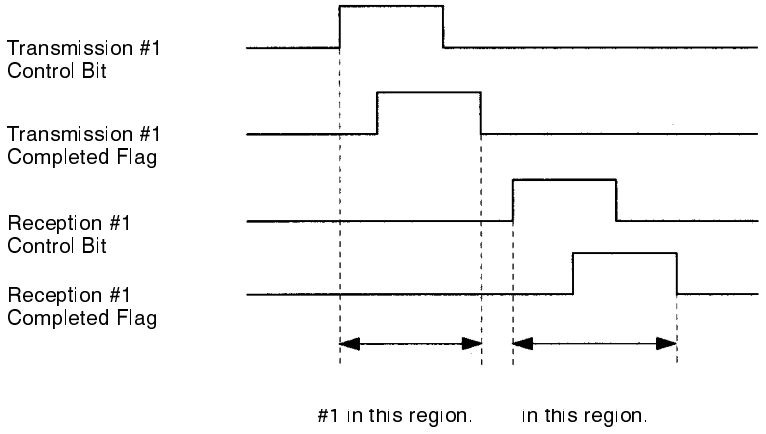
When it is necessary to transmit and receive data, dedicate some word(s) to transmission only, and the other word(s) to reception only, so that there isn't any conflict between transmission and reception.

The following example shows two words dedicated to transmission and two dedicated to reception.

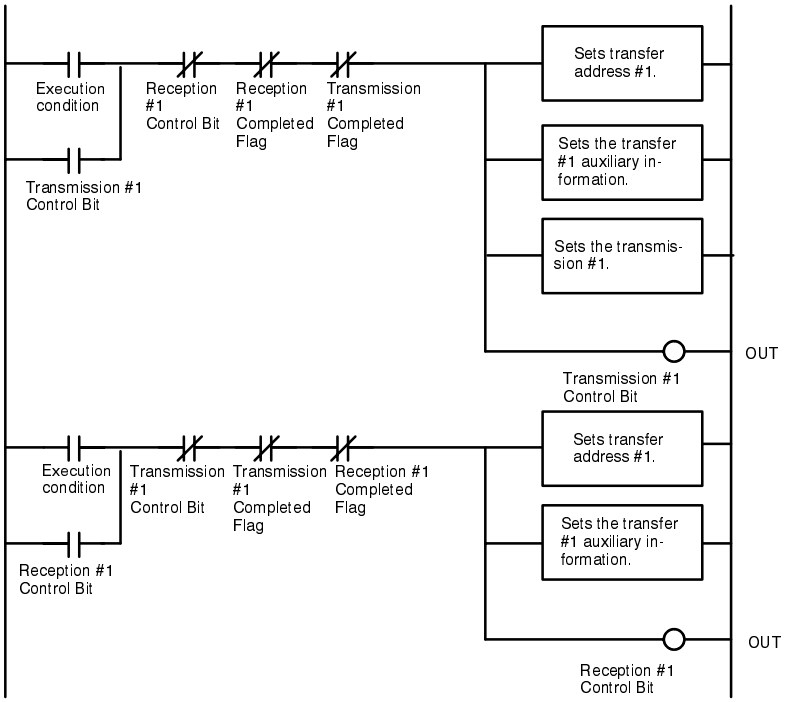


Controlling Timing

When the same word must be used for transmission and reception, use the completion flags to control the timing of the operations and prevent any conflict between transmission and reception.



The following diagram shows an example ladder program that prevents conflicts between transmission and reception.



SECTION 5
G Language

This section describes the functions in the G language. The functions are listed in order by G code.

5-1 List of G Functions

The following table provides summary and brief description of the G-language functions. Refer to the page number in the last column for more details.

Code	Name	Function	Page
G00	POSITIONING	Positions up to 4 axes simultaneously with PTP control at the maximum speed.	45
G01	LINEAR INTERPOLATION	Performs linear interpolation on up to 4 axes simultaneously at the specified interpolation speed.	47
G02	CIRCULAR INTERPOLATION (CLOCKWISE)	Performs 2-axis circular interpolation in the clockwise direction at the specified interpolation speed.	50
G03	CIRCULAR INTERPOLATION (COUNTERCLOCKWISE)	Performs 2-axis circular interpolation in the counterclockwise direction at the specified interpolation speed.	
G04	DWELL TIMER	Waits for the specified length of time.	54
G10	PASS MODE	Performs operations one-by-one in sequence without deceleration to stop.	55
G11	STOP MODE	Performs the next operation after completing positioning.	56
G17	CIRCULAR PLANE SPECIFICATION (X-Y)	Sets the X-Y plane as the plane for circular interpolation.	57
G18	CIRCULAR PLANE SPECIFICATION (X-Z)*	Sets the X-Z plane as the plane for circular interpolation.	
G19	CIRCULAR PLANE SPECIFICATION (Y-Z)*	Sets the Y-Z plane as the plane for circular interpolation.	
G20	CIRCULAR PLANE SPECIFICATION (X-U)*	Sets the X-U plane as the plane for circular interpolation.	
G21	CIRCULAR PLANE SPECIFICATION (Y-U)*	Sets the Y-U plane as the plane for circular interpolation.	
G22	CIRCULAR PLANE SPECIFICATION (Z-U)*	Sets the Z-U plane as the plane for circular interpolation.	
G26	REFERENCE ORIGIN RETURN	Moves to the reference origin.	58
G27	WORKPIECE ORIGIN RETURN	Moves to the workpiece origin.	59
G28	ORIGIN SEARCH	Performs an origin search in the specified axis.	60
G50	SELECT REFERENCE COORDINATE SYSTEM	Specifies the reference coordinate system.	61
G51	SELECT WORKPIECE COORDINATE SYSTEM	Specifies the workpiece coordinate system.	61
G53	CHANGE WORKPIECE ORIGIN OFFSET	Changes the origin of the workpiece coordinate system.	62
G54	CHANGE REFERENCE COORDINATE SYSTEM PV	Changes the present value in the reference coordinate system.	63
G60	ARITHMETIC OPERATIONS	Performs arithmetic operations on numerical values, position data, and registers.	65
G63	SUBSTITUTION	Substitutes numerical values, position data, or registers into other position data or registers.	65
G69	CHANGE PARAMETER	Changes the specified parameter.	66
G70	UNCONDITIONAL JUMP	Unconditionally jumps to the specified block.	67
G71	CONDITIONAL JUMP	Jumps to the specified block when the condition is met.	68
G72	SUBPROGRAM JUMP	Calls the specified subprogram.	69
G73	SUBPROGRAM END	Ends the subprogram.	69
G74	OPTIONAL END	Ends the block currently being executed when the specified optional input is ON.	70
G75	OPTIONAL SKIP	Skips the block after this function when the specified optional input is ON.	71
G76	OPTIONAL PROGRAM PAUSE	Pauses the program when the specified optional input is ON.	72

Code	Name	Function	Page
G79	PROGRAM END	Ends the main program.	74
G90	ABSOLUTE SPECIFICATION	Specifies the use of absolute coordinates in axis operations.	74
G91	INCREMENTAL SPECIFICATION	Specifies the use of relative coordinates in axis operations.	75

Note *CV500-MC421 only.

5-2 G Function Formats

The following table shows the format used for the G-language functions.

Name	Code	Operands
POSITIONING	G00	<Axis movement command ... >_[M<M code>] _[#<optional number>]
LINEAR INTERPOLATION	G01	<Axis movement command ... >_[F<speed command>] _[M<M code>] _[#<optional number>]
CIRCULAR INTERPOLATION (CLOCKWISE)	G02	[<Axis movement command ... >]_<I to H center coordinate ...> _[F<speed command>] _[M<M code>] _[#<optional number>] [<Axis movement command ... >]_<R radius> _[F<speed command>] _[M<M code>] _[#<optional number>]
CIRCULAR INTERPOLATION (COUNTERCLOCKWISE)	G03	[<Axis movement command ... >]_<I to H center coordinate ...> _[F<speed command>] _[M<M code>] _[#<optional number>] [<Axis movement command ... >]_<R radius> _[F<speed command>] _[M<M code>] _[#<optional number>]
DWELL TIMER	G04	□<wait time>
PASS MODE	G10	---
STOP MODE	G11	---
CIRCULAR PLANE SPECIFICATION (X-Y)	G17	---
CIRCULAR PLANE SPECIFICATION (X-Z)[1]	G18	---
CIRCULAR PLANE SPECIFICATION (Y-Z)[1]	G19	---
CIRCULAR PLANE SPECIFICATION (X-U)[1]	G20	---
CIRCULAR PLANE SPECIFICATION (Y-U)[1]	G21	---
CIRCULAR PLANE SPECIFICATION (Z-U)[1]	G22	---
REFERENCE ORIGIN RETURN	G26	<Axis name ... >_[M<M code>]
WORKPIECE ORIGIN RETURN	G27	<Axis name ... >_[M<M code>]
ORIGIN SEARCH	G28	<Axis name ... >_[M<M code>]
SELECT REFERENCE COORDINATE SYSTEM	G50	---
SELECT WORKPIECE COORDINATE SYSTEM	G51	---
CHANGE WORKPIECE ORIGIN OFFSET	G53	<Offset value ... >
CHANGE REFERENCE COORDINATE SYSTEM PV	G54	<Present value ... >

Name	Code	Operands
ARITHMETIC OPERATIONS	G60	<First term = Second term operator third term>
SUBSTITUTION	G63	<First term = Second term>
CHANGE PARAMETER	G69	<#Parameter type/New setting ... >
UNCONDITIONAL JUMP	G70	<Jump destination block number> [/L<number of loops>]
CONDITIONAL JUMP	G71	<Jump destination block number/condition equation>
SUBPROGRAM JUMP	G72	<Subprogram number>
SUBPROGRAM END	G73	---
OPTIONAL END	G74	□<Optional number>
OPTIONAL SKIP	G75	□<Optional number>
OPTIONAL PROGRAM STOP	G76	□<Optional number>
PROGRAM END	G79	---
ABSOLUTE SPECIFICATION	G90	---
INCREMENTAL SPECIFICATION	G91	---

Note 1. CV500-MC421 only.

2. Operand punctuation:

a) < > Content to be input.

b) [ ] Option

c) ... Multiple specification is possible.

d) □ The space is required.

e) _ The space can be left out.

5-3 List of G Symbols

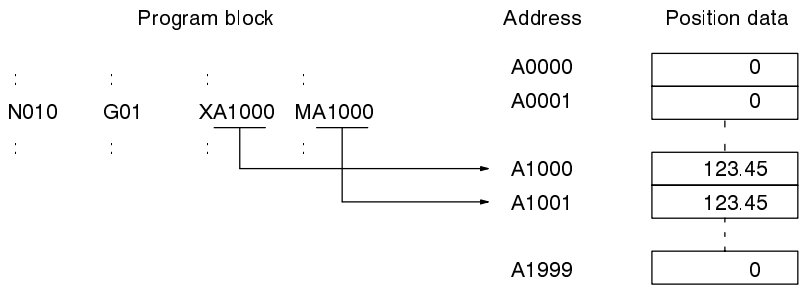
The following table lists the symbols used in G-language programming.

Symbol	Meaning
A	Indicates a position data address. There are 2000 addresses ranging from A0000 to A1999.
E	Indicates an indirect register. There are 32 indirect registers ranging from E00 to E31.
F	Indicates the speed when performing interpolation operations.
G	G function
H*	Arc center coordinate for the U-axis.
I	Arc center coordinate for the X-axis.
J	Arc center coordinate for the Y-axis.
K*	Arc center coordinate for the Z-axis.
L	Number of loops
M	M code
N	Block number
P	Program number
R	Circle radius
U*	U-axis
X	X-axis
Y	Y-axis
Z*	Z-axis
/	Punctuation mark
( )	Indirect specification
#	Optional number, parameter type
*	Comment

Note *CV500-MC421 only.

5-3-1 Specifying Position Data Addresses (A0000 to A1999)

It is possible to use the contents of a position data address for position data or an M code by specifying that address in an argument in an axis movement command or M code. For example, when the following program is executed, the contents of A1000 (123.45) will be used for the X-axis data and the contents of A1001 (123) will be used for the M code.



5-3-2 Specifying Registers (E00 to E31)

Position data can be specified indirectly by specifying a register (E00 to E31) in an axis movement command or M code. The relationship between the register and position data contents is shown below.

Register		Position data	
E00	1000	A1000:	123.35
E01	1001	A1001:	123.35

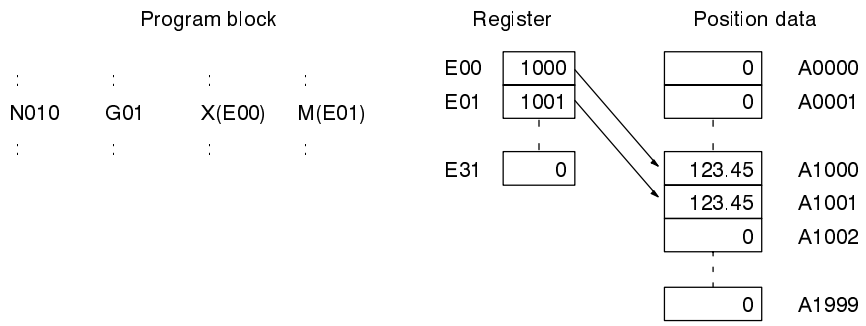
Registers and position data aren't assigned together to each task; they can be used independently for different tasks.

The contents of registers are all cleared to zero only when power is turned ON. Initialize register contents at the beginning of the program by executing an function such as the SUBSTITUTION function (G63). Registers can contain values from 0 through 1999. An error will occur if a value greater than 1999 is input in a register. The possible range of values for position data is –39,999,999 through 39,999,999.

Indirect Addressing of Position Data

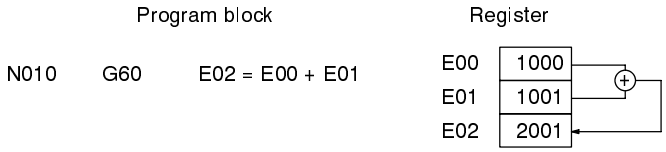
If the register name is in parentheses, i.e. (E00), the content of register will be treated as a position data address and the function will operate on the content of that position data address.

For example, when the following program is executed, the contents of A1000 (123.45) will be used for the X-axis data and the contents of A1001 (123) will be used for the M code.



Direct Addressing of Registers

If the register name isn't in parentheses, i.e. E00, the function will operate directly on the content that register. In the following example, the content of E00 and E01 are added and the result is placed in E02.



5-4 Program Number and Axis Declaration

The program number and axes being used must be declared at the beginning of the program.

Format

<program number_axes>

Parameters

The following table shows the program numbers and axes that can be declared for main programs and subprograms. In the case of the CV500-MC221, only the X and Y axes are available.

Program	Program numbers	Axes
Main program	P000 through P499	X, Y, Z, and U
Subprogram	P500 through P999	X, Y, Z, and U

Usage

There are two kinds of MC Unit programs, main programs and subprograms. Main programs are assigned program numbers P000 through P499 and subprograms are assigned program numbers P500 through P999.

Every program must begin with a declaration of the program number and the axes being used. Main programs end with a PROGRAM END function (G79) and subprograms end with a SUBPROGRAM END function (G73). This declaration is made at the beginning of the program (block number N000); an error will occur if the declaration is made in a later block.

The MC Unit can contain up to 100 programs.

Example Programs

Use the following format for main programs.

```
N000  P100  XYZ      Program number and axis declaration
      :     :
      :     :
N100  G79            PROGRAM END function
```

Use the following format for subprograms.

```
N000  P510  XYZ      Program number and axis declaration
      :     :
      :     :
N100  G73            SUBPROGRAM END function
```

When executing the main program from the first block, make initial settings such as the following.

1, 2, 3...

1. Restore the acceleration/deceleration time, interpolation acceleration/deceleration time, and workpiece origin offset value to the system parameter values.
2. Set the operating mode to pass mode.
3. Set the coordinate system to the reference coordinate system.
4. Set the command method to absolute.

5-5 G Functions

This section provides detailed descriptions of the G functions. The descriptions detail each function's format, parameters, usage, and provide an example program. The function format shows the operands which can be used with the function and indicates the correct syntax using the following punctuation marks.

1, 2, 3...

1. < > These operands must be input.
2. ... Indicates that more than one operand can be specified.
3. [ ] These operands can be added as an option.
4. □ A space must be input.
5. _ The space can be left out.

5-5-1 G00: POSITIONING

Positions up to 4 axes simultaneously with PTP control at high speed.

Format

G00_<Axis movement command ... >_[M<M code>]
_[#<optional number>]

Parameters

The following table shows the possible settings for the parameters.

Parameter		Possible settings
Axis movement command	Axis	X, Y, Z, and U
	Coordinate data	–39,999,999 to +39,999,999 (E00 to E31) A0000 to A1999 + or –
M code*		000 to 999 (E00 to E31) A0000 to A1999
Optional number		00 to 19

Note Refer to *5-6 M-code Outputs* for details on M codes.

Description

Positions up to 4 axes simultaneously with PTP control according to the maximum feed speed, acceleration/deceleration time, and acceleration/deceleration curve settings in the system parameters. The override is also valid with this function.

When just a "+" or "–" is entered for the coordinate data, the axis will be positioned to its + or – stroke limit.

Control is passed on to the next block after all of the specified axes reach their in-position ranges.

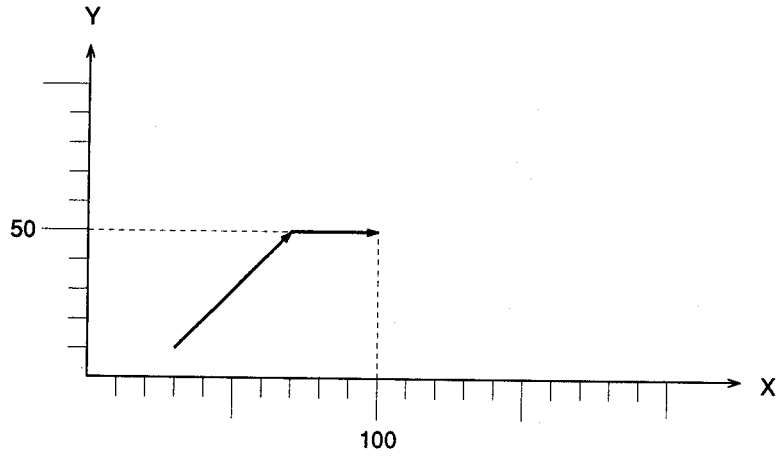
When an optional number is specified, the axis movement command will be carried out when that optional input goes ON. The operation won't be performed until the optional input goes ON. The source of the optional input depends on the optional number specified, as shown below.

0 through 15: Inputs from the PC's data area interface
16 through 19: Inputs from general inputs 1 through 4

Example Programs

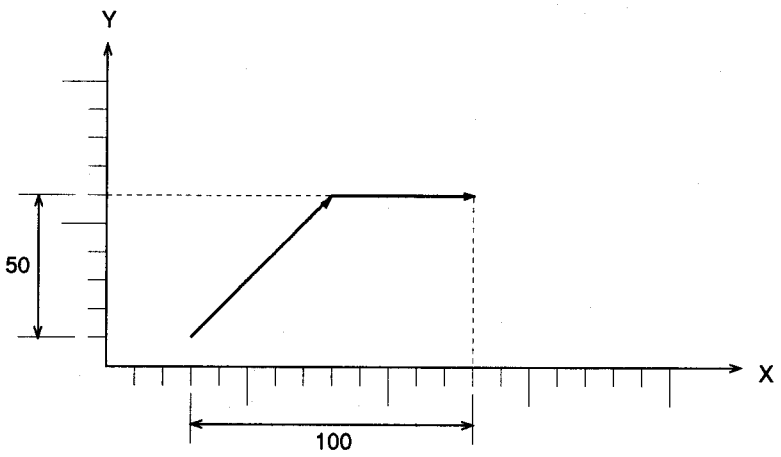
The following program is for absolute operation.

```
      :     :
N010  G90                ABSOLUTE SPECIFICATION
N011  G00   X100  Y50 M001
      :     :
```



The following program is for incremental operation.

```
      :     :
N010  G91                INCREMENTAL SPECIFICATION
N011  G00   X100  Y50 M001
      :     :
```



Note The X and Y axes are operated at the same speed in the above examples.

5-5-2 G01: LINEAR INTERPOLATION

Performs linear interpolation on up to 4 axes simultaneously at the specified interpolation speed.

Format

G01_<Axis movement command ... >_[F<speed command>]
_[M<M code>]
_[#<optional number>]

Parameters

The following table shows the possible settings for the parameters.

Parameter		Possible settings
Axis movement command	Axis	X, Y, Z, and U
	Coordinate data	–39,999,999 to +39,999,999 (E00 to E31) A0000 to A1999
Speed command		0.0001 to 39,999,999 (E00 to E31) A0000 to A1999
M code*		000 to 999 (E00 to E31) A0000 to A1999
Optional number		00 to 19

Note Refer to *5-6 M-code Outputs* for details on M codes.

Description

Performs linear interpolation on up to 4 axes simultaneously at the specified interpolation speed.

The interpolation speed is calculated with the equations shown below. All four axes are shown, but the same calculations apply when three or fewer axes are being used.

$Fx = F \times dLx/L$
$Fy = F \times dLy/L$
$Fz = F \times dLz/L$
$Fu = F \times dLu/L$

Fx through Fu: Interpolation speeds for axes x through u
dLx through dLu: Movement distance for axes x through u
L: Total distance between start and end of linear interpolation

$$\left(L = \sqrt{dLx^2 + dLy^2 + dLz^2 + dLu^2}\right)$$

If the interpolation speed isn't specified, the interpolation will be performed at the last speed used in a G01, G02, or G03 function. An error will occur if the interpolation speed isn't specified and a G01, G02, or G03 function hasn't been executed before.

⚠ Caution When linear interpolation is performed in pass mode and an M code from 0 through 499 or an optional number is specified, the interpolation will be performed in stop mode, not pass mode.

If the same interpolation acceleration/deceleration times and override values aren't set for all of the axes used in a task, the settings for the axis with the highest priority will be used. Axis X has the highest priority, followed by Y, Z, and U.

The override value can be specified from the Teaching Box or PC data area interface. The interpolation acceleration time is always used when operating in Pass mode.

When an optional number is specified, the axis movement command will be carried out when that optional input goes ON. The operation won't be performed until the optional input goes ON. The source of the optional input depends on the optional number specified, as shown below.

0 through 15: Inputs from the PC's data area interface
16 through 19: Inputs from general inputs 1 through 4

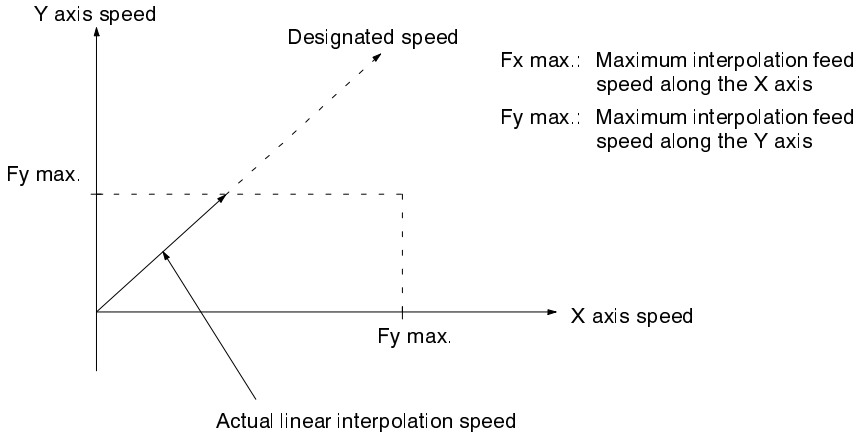
The linear interpolation speed can be controlled in two ways.

The speed can be controlled using pin 3 of the DIP switch on the MC Unit. Normally set pin 3 of the DIP switch to ON. (Factory-set to OFF)

Note The ON setting of pin 3 will be effective on Units manufactured on and after July 11, 1995. As for Units manufactured before this date, only the OFF setting will be effective regardless of the ON/OFF setting.

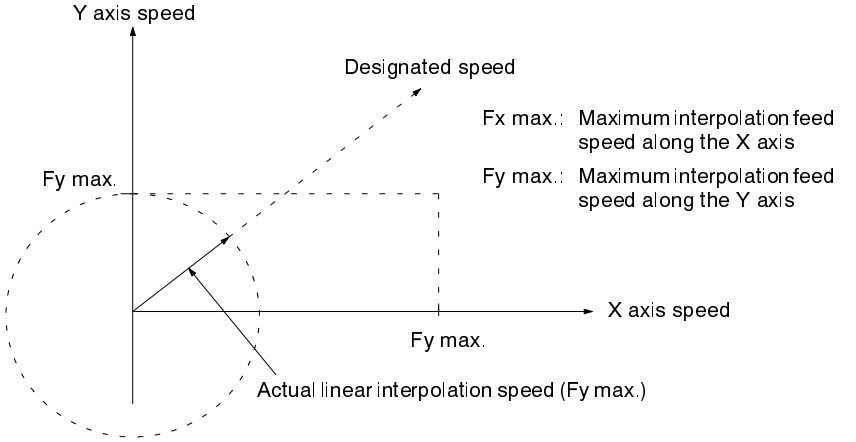
When Pin 3 of the DIP Switch is Set to ON (Normally Set to ON)
If the designated linear interpolation speed converted for each axis exceeds the maximum interpolation feed speed (a value set via the system parameter) of the axis to be moved, linear interpolation is performed to move either one of the axes at the maximum interpolation feed speed.

Example: Linear Interpolation for Two Axes
As the designated speed is higher than Fy max., linear interpolation is performed to move the Y axis at the Fy max. speed.



When Pin 3 of the DIP Switch is Set to OFF
If the designated linear interpolation speed exceeds the maximum interpolation feed speed (a value set via the system parameter) of the axis to be moved, linear interpolation is performed at the lowest maximum interpolation feed speed of the axes to be moved.

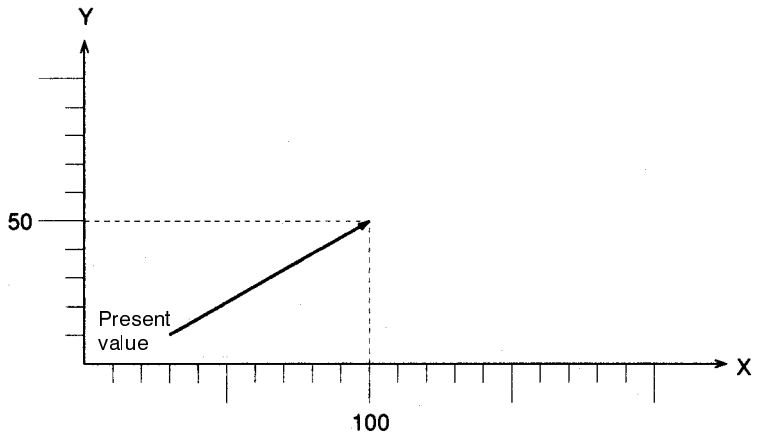
Example: Linear Interpolation for Two Axes
As the designated speed is higher than Fy max., linear interpolation is performed to move the Y axis at Fy max. speed.



Example Programs

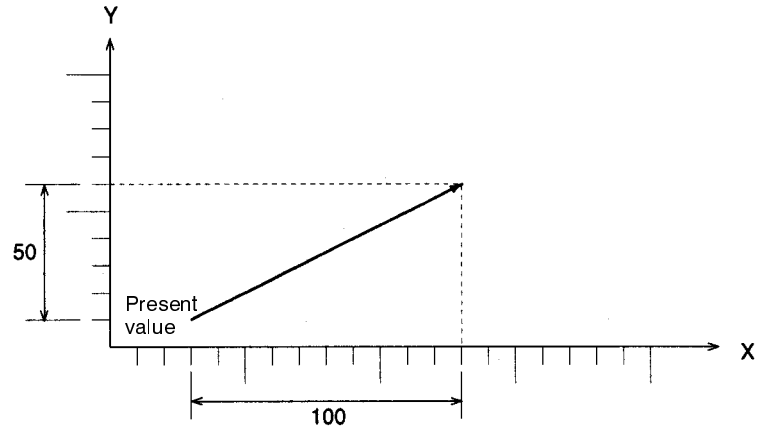
The following program is for absolute operation.

```
      :     :
N010  G90                ABSOLUTE SPECIFICATION
N011  G01   X100  Y50 F300
      :     :
```



The following program is for incremental operation.

```
      :     :
N010  G91                INCREMENTAL SPECIFICATION
N011  G01   X100  Y50 F300
      :     :
```



5-5-3 G02 and G03: CIRCULAR INTERPOLATION

Performs two-axis circular interpolation in the clockwise (G02) or counterclockwise (G03) direction at the specified interpolation speed. With the CV500-MC421, one or two-axis linear interpolation can be added to the two-axis circular interpolation to create helical interpolation.

Format

G02 (G03)_<Axis movement command ... >_<I to H center coordinate ...>
_[F<speed command>]
_[M<M code>]
_[#<optional number>]

G02 (G03)_<Axis movement command ... >_<R radius>
_[F<speed command>]
_[M<M code>]
_[#<optional number>]

Parameters

The following table shows the possible settings for the parameters.

Parameter		Possible settings
Axis movement command	Axis	X, Y, Z, and U
	Coordinate data	–39,999,999 to +39,999,999 (E00 to E31) A0000 to A1999
Center coordinate	Axis	I, J, K, or H
	Radius	R
	Coordinate and radius data	–39,999,999 to +39,999,999 (E00 to E31) A0000 to A1999
Speed command		0.0001 to 39,999,999 (E00 to E31) A0000 to A1999
M code*		000 to 999 (E00 to E31) A0000 to A1999
Optional number		00 to 19

Note Refer to *5-6 M-code Outputs* for details on M codes.

Description

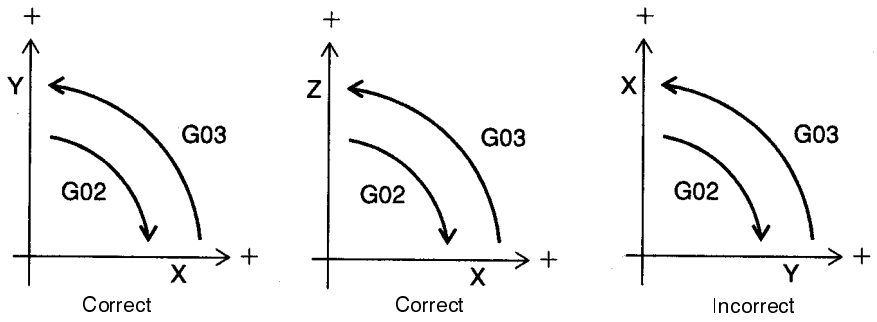
Performs two-axis circular interpolation, two-axis circular interpolation plus one-axis linear interpolation, or two-axis circular interpolation plus two-axis linear interpolation at the specified interpolation speed. Two-axis circular interpolation plus one-axis linear interpolation is called helical interpolation. An axis that isn't controlled by circular interpolation is called an additional axis.

The CIRCULAR PLANE SPECIFICATION functions (G17 through G22) determine the axes in which circular interpolation is performed. The current position is the default coordinate data in the specified circular plane.

Axes outside of the specified circular plane won't be moved unless if they aren't specified.

If the interpolation speed isn't specified, the interpolation will be performed at the last speed used in a G01, G02, or G03 function. An error will occur if the interpolation speed isn't specified and a G01, G02, or G03 function hasn't been executed before. An error will also occur if the G02 or G03 function is executed before the circular plane is specified.

Always make the higher priority axis the horizontal axis in the circular plane. The X-axis has the highest priority, followed by Y, Z, and U. In the following diagram, the two examples on the left are correct, but the one on the right is incorrect.



When an axis outside of the circular plane (an additional axis) is specified, it will be moved by linear interpolation and its interpolation speed can be calculated with the equation shown below.

$Fi = F \times dLi/Lc$

Fi: Interpolation speed for the additional axis
F: Speed command value
dLi: Movement distance for the additional axis
Lc: The circle's circumference

An error will occur if the additional axis' interpolation speed exceeds that axis' maximum interpolation speed.

When specifying the circle's center coordinate, specify the relative distance from the current position (the circle's starting point) and specify one or two axes. An error will occur if there is no specification or if three or four axes are specified.

When specifying the radius, an arc smaller than a semicircle will be drawn with positive values and an arc greater than a semicircle will be drawn with negative values. Either a negative or positive value can be specified for a semicircle.

When specifying a complete circle (360°), the system will operate when either of the following conditions is met in the arc center coordinate specifications.

- When the axis movement command and current position are the same
- When the axis movement command is left out

A complete circle can't be drawn by the radius specification.

The interpolation acceleration/deceleration time and override value setting for the axis in the task with the highest priority will be used. Axis X has the highest priority, followed by Y, Z, and U.

⚠ Caution When circular interpolation is performed in pass mode and an M code from 0 through 499 or an optional number is specified, the interpolation will be performed in stop mode, not pass mode.

The override value can be specified from the Teaching Box or PC data area interface.

When the product of the specified interpolation speed and override value exceeds the maximum interpolation feed speed for the axis being moved by circular interpolation (set in the system parameters), circular interpolation will be performed at the lowest maximum feed speed among the axes being moved. For example, consider the following situation.

```
G02   X100  Y90   I0    J40    F3000
```

X-axis override value:	100.0	[%]
X-axis max. interpolation feed speed:	4000	[mm/s]
Y-axis max. interpolation feed speed:	2000	[mm/s]

Interpolation speed after override calculation = 3000 × 100.0/100.0
= 3000 [mm/s]

The interpolation speed of 3000 [mm/s] exceeds the maximum interpolation feed speed for the Y-axis (2000 [mm/s]), so the interpolation speed is set to 2000 [mm/s].

Note

1. If circular interpolation is performed with 2 axes with different pulse rates, an ellipse will be drawn and circular interpolation might not be possible.
2. When the radius exceeds 9,999,999 pulses, circular interpolation will be performed within the specified circle. Keep the radius under 9,999,999 pulses when performing high-precision circular interpolation.
3. When circular interpolation is performed with the operating mode set to pass mode and the circular plane is changed during operation, circular interpolation will begin after completion of positioning is verified.

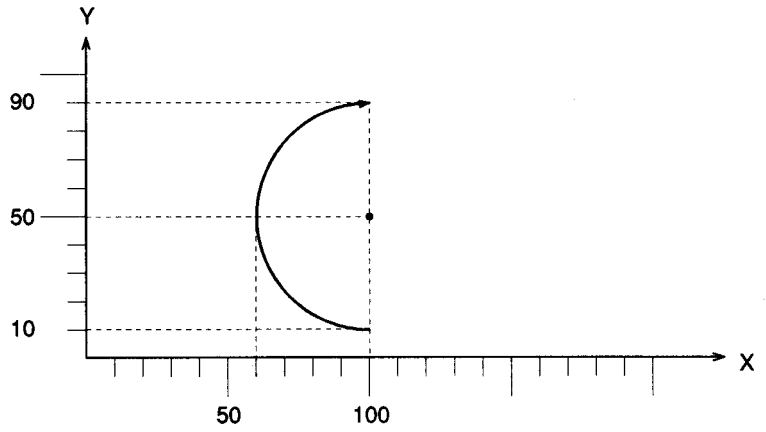
When an optional number is specified, the axis movement command will be carried out when that optional input goes ON. The operation won't be performed until the optional input goes ON. The source of the optional input depends on the optional number specified, as shown below.

0 through 15: Inputs from the PC's data area interface
16 through 19: Inputs from general inputs 1 through 4

Example Programs

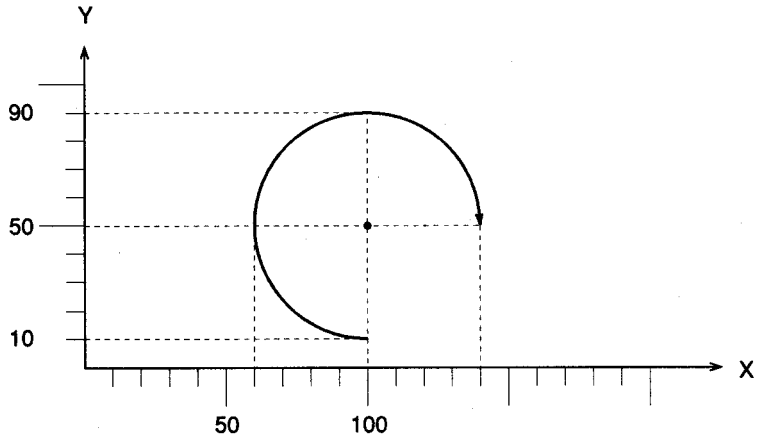
The following program shows circular interpolation with center coordinate specification.

```
      :     :
N010  G90                          ABSOLUTE SPECIFICATION
N011  G17                          Makes X-Y the circular plane.
N012  G02   X100  Y90   I0    J40   F300
      :     :
```



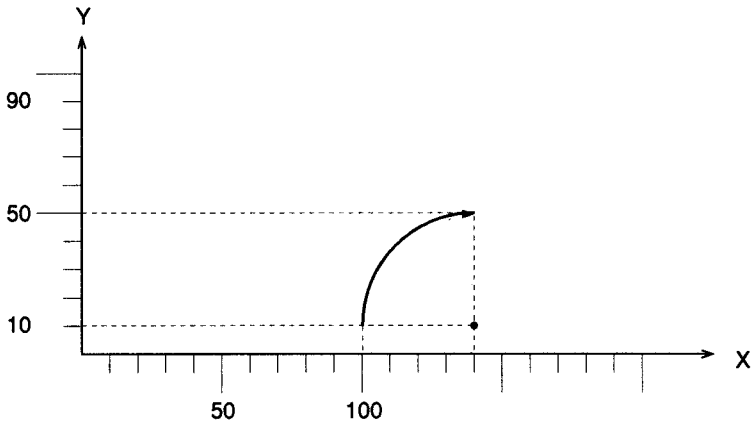
The following program shows circular interpolation with radius specification (R<0). An arc greater than a semicircle will be drawn when R<0.

```
      :     :
N010  G90                          ABSOLUTE SPECIFICATION
N011  G17                          Makes X-Y the circular plane.
N012  G02   X140  Y50   R-40  F300
      :     :
```



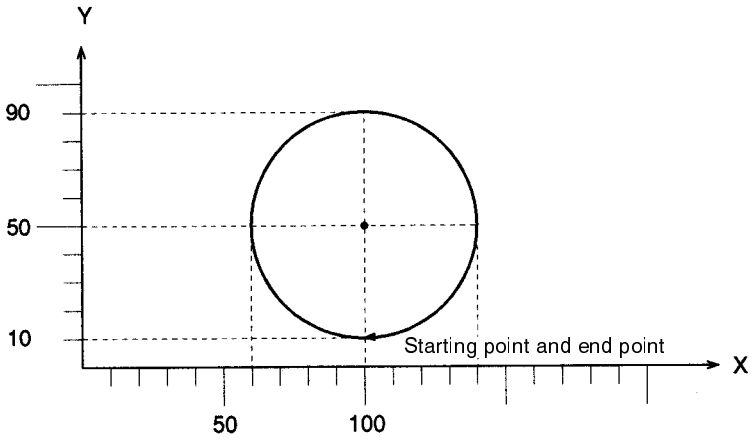
The following program shows circular interpolation with radius specification (R>0). An arc smaller than a semicircle will be drawn when R>0.

```
      :     :
N010  G91                  INCREMENTAL SPECIFICATION
N011  G17                  Makes X-Y the circular plane.
N012  G02   X40   Y40 R40           F300
      :     :
```



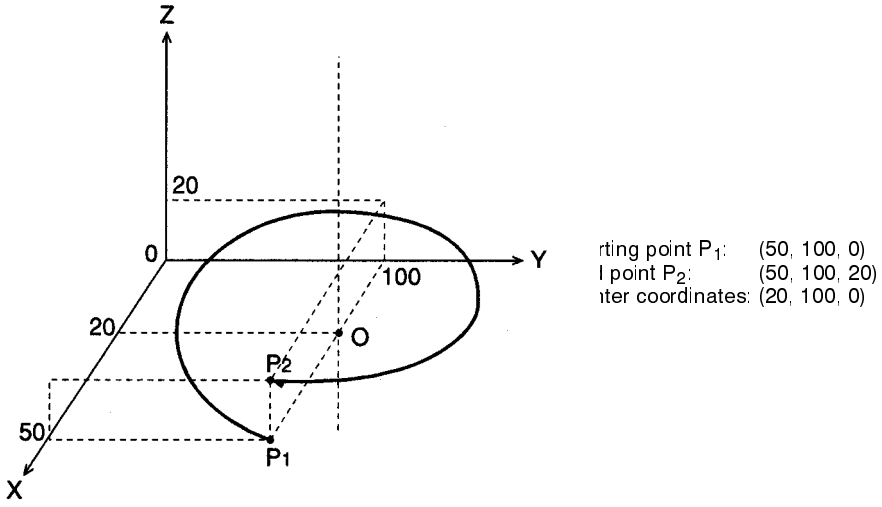
The following program shows circular interpolation of a complete circle.

```
      :     :
N010  G90                          ABSOLUTE SPECIFICATION
N011  G17                          Makes X-Y the circular plane.
N012  G02   X100  Y10   I0    J40   F500
      :     :
```



The following program shows helical interpolation.

```
      :     :
N010  G90                          ABSOLUTE SPECIFICATION
N011  G17                          Makes X-Y the circular plane.
N012  G02   X50   Y100  Z20   I-30  J0    F500
      :     :
```



5-5-4 G04: DWELL TIMER

Waits for the specified length of time.

Format

G04_<wait time>

Parameters

The following table shows the possible settings for the wait time parameter.

Parameter	Possible settings
Wait time	0.001 to 39,999.994 (E00 to E31) A0000 to A1999

Description

This function waits for the specified wait time.

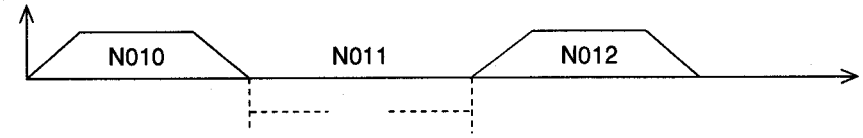
The wait time can be specified in a register or position data address. When the number read from memory includes values below the third decimal place, those values will be rounded off. An error will occur if the result of this rounding is outside of the acceptable range (0.001 to 39,999.994).

Values from 0.001 to 39,999.999 can be input for the wait time, but settings of 39,999.995 to 39,999.999 will cause an error. The accuracy of the timer is about 0.02 s.

⚠ Caution If this function is executed just after a G01, G02, or G03 function, the interpolation will be performed in stop mode, not pass mode, even if pass mode has been selected.

Example Program

```
      :     :
N009  G10
N010  G01   X100  Y100
N011  G04   10
N012  G01   X200  Y200
      :     :
```



Note Program blocks N010 and N012 aren't pass operations.

5-5-5 G10: PASS MODE

Switches the operating mode to pass mode.

Format

G10

Parameters

None

Description

This function switches the operating mode to pass mode. If this function is executed, any subsequent interpolation (linear or circular) function will move on to the next positioning operation without decelerating to a stop. The pass mode is valid until the STOP MODE function (G11) is executed.

The pass operation might not occur if a G01, G02, or G03 function is executed with an extremely short movement.

In pass mode, the next G-code function will be executed without waiting for the completion of a G01, G02, or G03 function that is being executed (pre-execution), so the "executed block number" output to the interface might be different from the actual block number being executed. Also, the subsequent functions will be executed until the next G01, G02, or G03 function in the program.

In the following program, blocks N003 to N005 will be pre-executed while the axis is moving according to the command in block N002. Although he axis is moving according to program block N002, the "executed block number" will be refreshed with block numbers N003 to N006. Program execution will then standby at block N006.

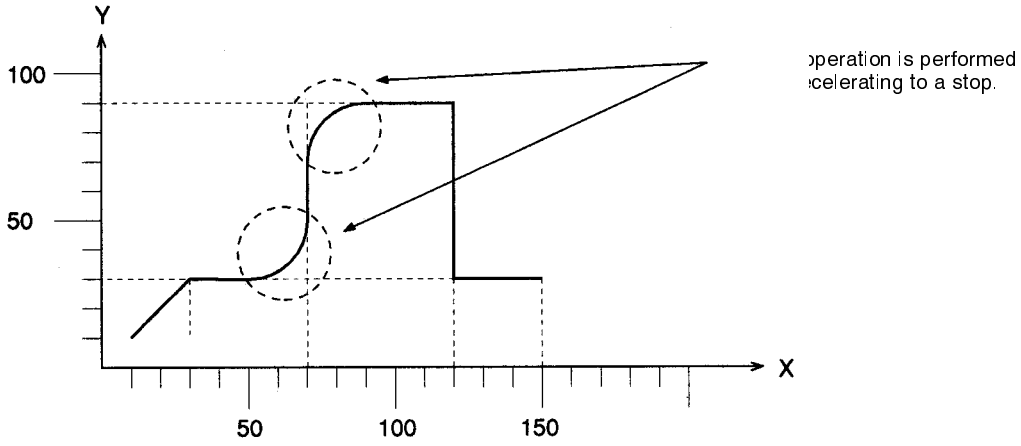
When command processing for N002 is completed, N006 will be executed and the functions after block N006 will be pre-executed.

```
N000  P000  XY
N001  G10
N002  G01   X100  F1000
N003  G63   A0000 = 1
N004  G63   A0001 = 2
N005  G63   A0002 = 3
N006  G01   X200
      :     :
```

⚠ Caution Pre-execution will be stopped and the axis operation will be paused with the following functions:
G00 (PTP operation)
A G01, G02, or G03 function with an M code from M000 to M499
G04 (DWELL TIMER)
G53 (CHANGE WORKPIECE ORIGIN OFFSET)
G54 (CHANGE REFERENCE COORDINATE SYSTEM PV)
G79 (PROGRAM END)
M000 to M499 as independent commands

Example Program

```
      :     :
N010  G00   X30   Y30
N011  G10
N012  G01   X70
N013  G01   Y90
N014  G01   X120
N015  G00   Y30
N016  G00   X150
      :     :
```



5-5-6 G11: STOP MODE

Switches the operating mode to stop mode.

Format G11

Parameters None

Description

This function switches the operating mode to stop mode. If this function is executed, subsequent interpolation (linear or circular) functions will be decelerated to a stop at the end point, an in-position check will be performed, and then the next positioning operation will be performed after completion of the positioning is verified.

The stop mode is valid until one of the following functions is executed:

1, 2, 3...

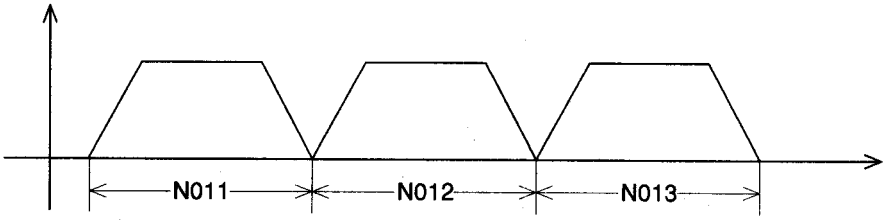
1. The PASS MODE function (G10)
2. The REFERENCE ORIGIN RETURN function (G26)
3. The WORKPIECE ORIGIN RETURN function (G27)
4. The ORIGIN SEARCH function (G28)

A reset will be performed and the operating mode will be switched to pass mode when a REFERENCE ORIGIN RETURN (G26), WORKPIECE ORIGIN RETURN (G27), or ORIGIN SEARCH (G28) function is executed. Refer to *5-7 Mode Transitions Caused by G Functions* for details on modes when resetting.

Example Program

The following program demonstrates positioning in STOP MODE. The Unit moves to the next operation after completion of the previous operation is verified.

```
      :     :
N010  G11
N011  G01   X100
N012  G01   X200
N013  G01   X300
      :     :
```



5-5-7 G17 to G22: CIRCULAR PLANE SPECIFICATION FUNCTIONS

Specifies the plane in which circular interpolation is performed.

Format

G17, G18, G19, G20, G21, or G22

Parameters

None

Description

This function specifies the plane in which circular interpolation is performed, as shown in the following table.

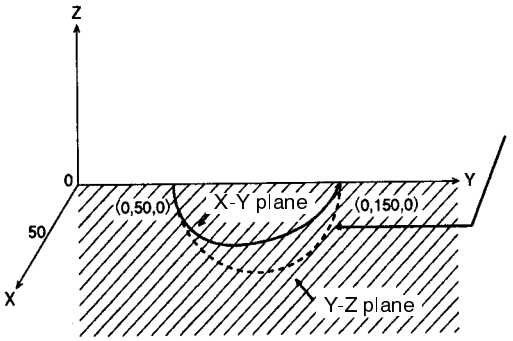
Code	Plane
G17	Specifies the X-Y plane.
G18	Specifies the X-Z plane.
G19	Specifies the Y-Z plane.
G20	Specifies the X-U plane.
G21	Specifies the Y-U plane.
G22	Specifies the Z-U plane.

The specified plane is valid until another one of the CIRCULAR PLANE SPECIFICATION functions is executed. An error will occur if a CIRCULAR INTERPOLATION function is executed before a circular plane has been specified.

Example Program

The following program performs circular interpolation in the X-Y plane.

```
      :     :
N010  G90                    ABSOLUTE SPECIFICATION
N011  G17
N012  G03   X0    Y150  I0    J50   F300
      (G03  X0    Y150  R50   F300)
      :     :
```



peration would be performed
ted line if G19 (Y-Z plane)
ammed in block N011.

5-5-8 G26: REFERENCE ORIGIN RETURN

Moves to the reference origin.

Format

G26_<Axis name ... >_[M<M code>]

Parameters

The following table shows the possible settings for the parameters.

Parameter	Possible settings
Axis name	X, Y, Z, and U
M code*	000 to 999 (E00 to E31) A0000 to A1999

Note Refer to *5-6 M-code Outputs* for details on M codes.

Description

This function moves the specified axes to the reference origin by PTP control. The coordinate system and mode settings will be as follows after the return to the reference origin:

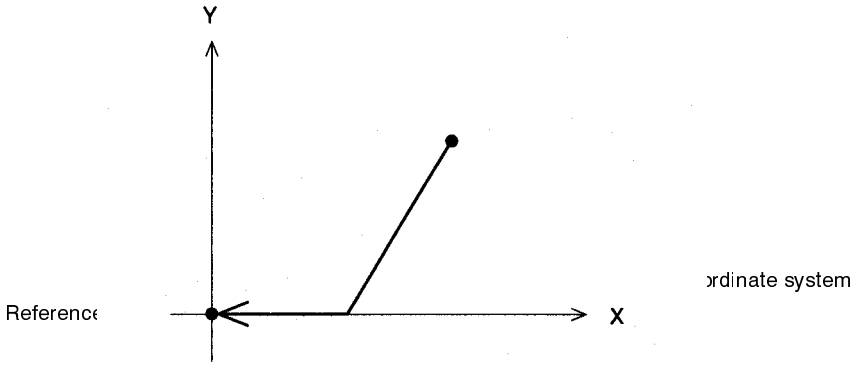
1, 2, 3...

1. The coordinate system is set to the reference coordinate system.
2. The coordinate specification is set to absolute specification.
3. The operating mode is set to pass mode.

The settings won't be switched to the ones above if function G26 is terminated while in progress by the OPTIONAL END function (G74), FORCED BLOCK END, or other function. The function is also considered to be in progress while waiting for the M code reset.

Example Program

```
        :     :
N010  G26   XY    M500
        :     :
```



5-5-9 G27: WORKPIECE ORIGIN RETURN

Moves to the workpiece origin.

Format

G27_<Axis name ... >_[M<M code>]

Parameters

The following table shows the possible settings for the parameters.

Parameter	Possible settings
Axis name	X, Y, Z, and U
M code*	000 to 999 (E00 to E31) A0000 to A1999

Note Refer to *5-6 M-code Outputs* for details on M codes.

Description

This function moves the specified axes to the workpiece origin by PTP control. The coordinate system and mode settings will be as follows after the return to the workpiece origin:

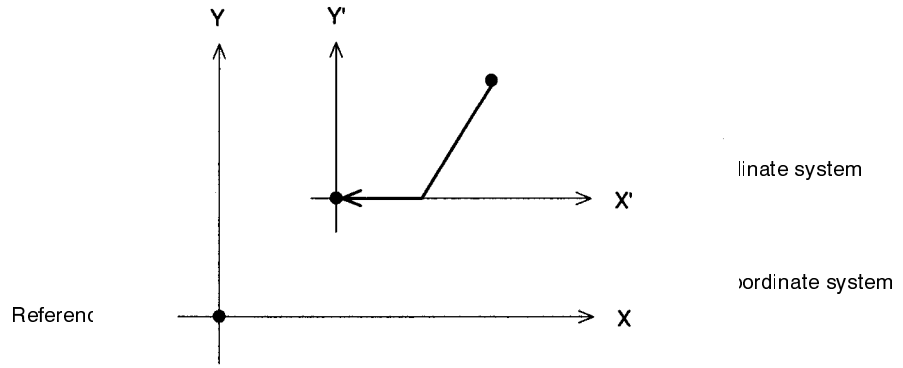
1, 2, 3...

1. The coordinate system is set to the workpiece coordinate system.
2. The coordinate specification is set to absolute specification.
3. The operating mode is set to pass mode.

The settings won't be switched to the ones above if function G27 is terminated while in progress by the OPTIONAL END function (G74), FORCED BLOCK END, or other function. The function is also considered to be in progress while waiting for the M code reset.

Example Program

```
      :     :
N010  G27   XY    M500
      :     :
```



5-5-10 G28: ORIGIN SEARCH

Performs an origin search in the specified axes.

Format

G28_<Axis name ... >_[M<M code>]

Parameters

The following table shows the possible settings for the parameters.

Parameter	Possible settings
Axis name	X, Y, Z, and U
M code[1]	000 to 999 (E00 to E31) A0000 to A1999

Note

1. Refer to *5-6 M-code Outputs* for details on M codes.
2. Refer to *Section 7 Fixing the Origin* for more details on origin searches.

Description

This function performs an origin search in the specified axes.

Axes using Incremental Encoders:

The origin search operation searches for the origin in the specified axes according to the settings of the Origin Search Method, Origin Search Direction, Origin Deceleration Method, and Origin Proximity Input Logic parameters.

For axes in origin mode, the current position is set to zero and fixed as the origin when the origin search method is set. For axes in other modes, the origin is fixed after the origin search operation is executed.

Axes using Absolute Encoders:

The axes are positioned at the mechanical origin by PTP control (the same as a reference origin return). The coordinate system and mode settings will be as follows when the origin search is completed:

1, 2, 3...

1. The coordinate system is set to the reference coordinate system.
2. The coordinate specification is set to absolute specification.
3. The operating mode is set to pass mode.

The settings won't be switched to the ones above if function G28 is terminated while in progress by the OPTIONAL END function (G74), FORCED BLOCK END, or other function. The function is also considered to be in progress while waiting for the M code reset.

Example Program

```
      :     :
N010  G28   XYZ   M500
      :     :
```

5-5-11 G50: SELECT REFERENCE COORDINATE SYSTEM

Sets the coordinate system to the reference coordinate system.

Format

G50

Parameters

None

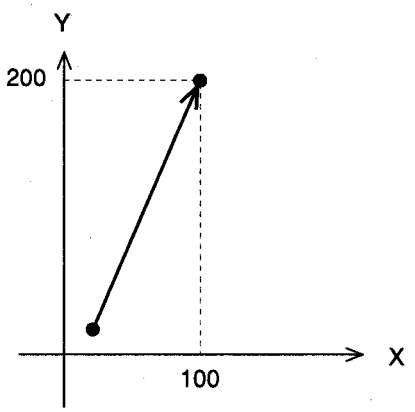
Description

This function sets the coordinate system to the reference coordinate system. After this function is executed, the coordinate data in all axis operations is processed as reference coordinate data. If G50 is executed at the beginning of the program, the coordinate system will be the reference coordinate system.

The reference coordinate system will be in effect after execution of the REFERENCE ORIGIN RETURN function (G26) or the ORIGIN SEARCH function (G28).

Example Program

```
      :     :
N009  G90
N010  G50
N011  G01   X100  Y200
      :     :
```



ɔordinate system

5-5-12 G51: SELECT WORKPIECE COORDINATE SYSTEM

Sets the coordinate system to the workpiece coordinate system.

Format

G51

Parameters

None

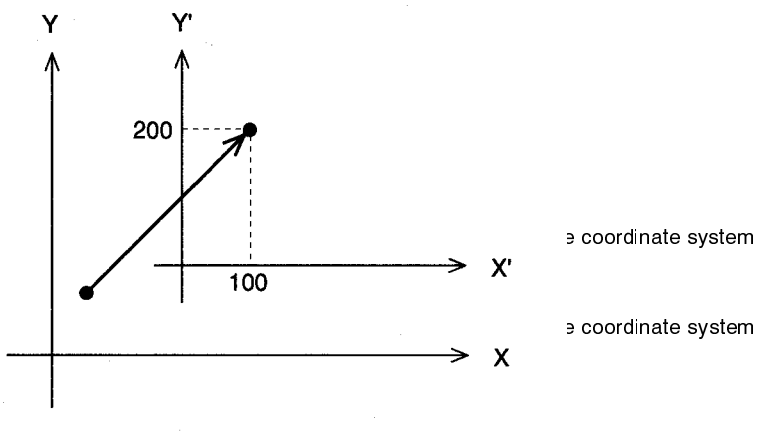
Description

This function sets the coordinate system to the workpiece coordinate system. After this function is executed, the coordinate data in all axis operations is processed as workpiece coordinate data. If G51 is executed at the beginning of the program, the origin of workpiece coordinate system is determined by the workpiece origin offset value (the offset from the reference origin) set in the system parameters.

The workpiece coordinate system will be in effect after execution of the WORKPIECE ORIGIN RETURN function (G27).

Example Program

```
      :     :
N009  G90
N010  G51
N011  G01   X100  Y200
      :     :
```



5-5-13 G53: CHANGE WORKPIECE ORIGIN OFFSET

Changes the origin of the workpiece coordinate system.

Format

G53_<offset value ... >

Parameters

The following table shows the possible settings for the parameters.

Parameter		Possible settings
Offset value	Axis name	X, Y, Z, and U
	Data	–39,999,999 to +39,999,999 (E00 to E31) A0000 to A1999

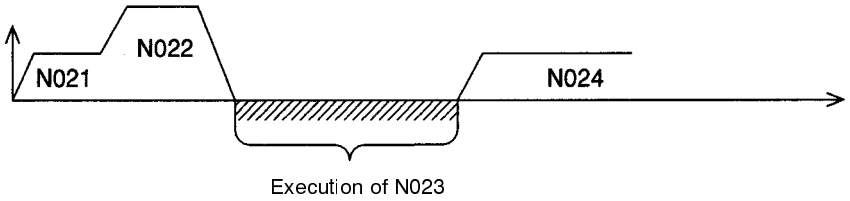
Description

This function changes the offset from the reference coordinate system origin to the workpiece coordinate system origin. The workpiece origin offset value set in the system parameters isn't changed, but the offset values set with this function take precedence.

Unless G53 is executed, the origin of the workpiece coordinate system is determined by the system parameter ("workpiece origin offset" of the coordinate parameter) when executed from the top of the program.

⚠ Caution If this function is executed just after a G01, G02, or G03 function, the interpolation will be performed in stop mode, not pass mode, even if pass mode has been selected. The following example shows this effect.

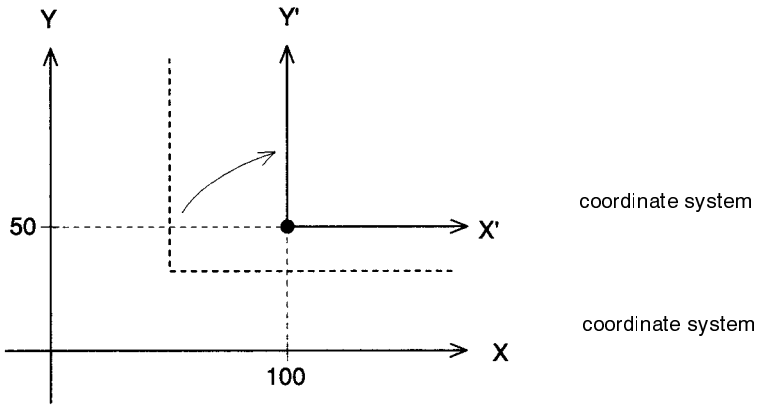
```
      :     :
N020  G10                    Sets pass mode.
N021  G01   X100   Y100   F100
N022  G01   X200
N023  G53   X100             Executed after N022 is completed.
N024  G01   X300
      :     :
```



Note An error will occur if the offset value exceeds the stroke limits set in the system parameters.

Example Program

```
      :     :
N010  G53   X100  Y50
      :     :
```



5-5-14 G54: CHANGE REFERENCE COORDINATE SYSTEM PV

Changes the present value in the reference coordinate system.

Format G54_<present value ... >

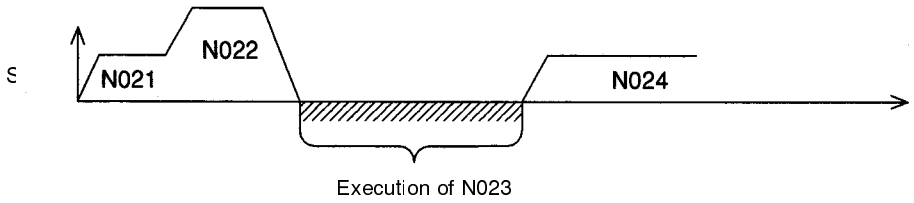
Parameters The following table shows the possible settings for the parameters.

Parameter		Possible settings
Present value	Axis name	X, Y, Z, and U
	Data	–39,999,999 to +39,999,999 (E00 to E31) A0000 to A1999

Description This function changes the present value of the reference coordinate system for Incremental Encoder axes to the specified present value.

⚠ Caution If this function is executed just after a G01, G02, or G03 function, the interpolation will be performed in stop mode, not pass mode, even if pass mode has been selected. The following example shows this effect.

```
      :     :
N020  G10                    Sets pass mode.
N021  G01   X100   Y100   F100
N022  G01   X200
N023  G54   X100             Executed after N022 is completed.
N024  G01   X300
      :     :
```

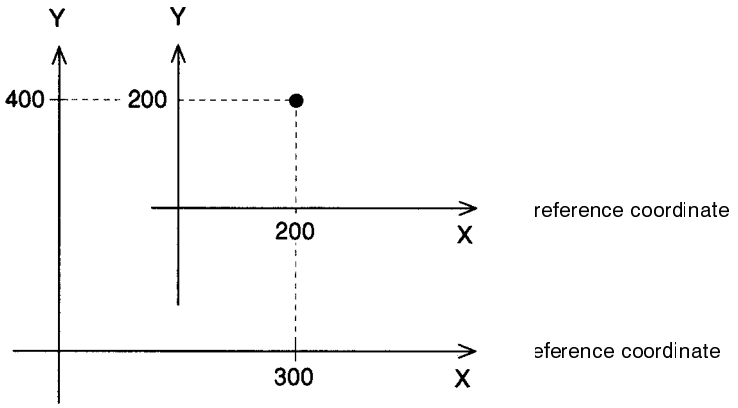


Note
1. An error will occur if the offset value exceeds the stroke limits set in the system parameters.
2. This function can't change the present value for Absolute Encoder axes.

Example Program

The following program shows the present value of the reference coordinate system changed from (300, 400) to (200, 200).

```
      :     :
N010  G54   X200  Y200
      :     :
```



5-5-15 G60: ARITHMETIC OPERATIONS

Performs arithmetic operations on numerical values, position data, and registers.

Format

G60_<first term = second term operator third term>

Parameters

The following table shows the possible settings for the parameters.

Parameter	Possible settings
First term	A0000 to A1999 E00 to E31
Second term	A0000 to A1999 E00 to E31 −39,999,999 to +39,999,999
Operator	+, −, *, or / (addition, subtraction, multiplication, or division)
Third term	A0000 to A1999 E00 to E31 −39,999,999 to +39,999,999

Description

This function performs arithmetic operations on numerical values, position data, or the contents of registers. When the first term is a register, the second and third terms will be integers. (Non-integer values are rounded to the nearest integer.) When the first term is a position data address, the second and third terms will be real numbers and values below the fourth decimal place are rounded off.

The possible values for the result are as follows:

−39,999,999 to −0.0001
0
0.0001 to 39,999,999

An error will occur if the first term's data isn't within the acceptable range. The possible values for the first term are as follows:

When the first term is position data: −39,999,999 to +39,999,999
When the first term is a register: 0000 to 1999

Registers are cleared to zero when the power is turned on, so be sure to initialize the register contents when the program is started.

Example Program

```
      :     :
N010  G60   A0000=A1000−500
      :     :
```

5-5-16 G63: SUBSTITUTION

Substitutes position data and registers.

Format

G63_<first term = second term >

Parameters

The following table shows the possible settings for the parameters.

Parameter	Possible settings
First term	A0000 to A1999 E00 to E31
Second term	A0000 to A1999 E00 to E31 −39,999,999 to +39,999,999 X, Y, Z, or U

Description

This function copies position data, register contents, present values, or numerical values into position data addresses or registers.

When the second term is an axis name, the present position of that axis in the reference coordinate system is copied to the first term. That present position is copied according to the pulse rate and minimum unit setting for that axis specified in the system parameters.

When the first term is a register, the value in the second term will be treated as an integer. Non-integer values are rounded to the nearest integer.

An error will occur if the first term's data isn't within the acceptable range. The possible values for the first term are as follows:

When the first term is position data: −39,999,999 to +39,999,999
When the first term is a register: 0000 to 1999

Registers are cleared to zero when the power is turned on, so be sure to initialize the register contents when the program is started.

Example Program

```
      :     :
N010  G63   A0000=123.45
      :     :
```

5-5-17 G69: CHANGE PARAMETER

Changes the setting of the specified parameter.

Format

G69_<#parameter type/new setting ... >

Parameters

The following table shows the possible settings for the parameters.

Parameter		Possible settings
Parameter type		1, 2, 3, or 4
New setting	Axis name	X, Y, Z, and U
	New setting	0 to 9,999 (E00 to E31) A0000 to A1999

Description

This function changes the setting of the specified parameter. The following table shows the parameters that can be changed and the parameter type values used to identify them.

Parameter type	Parameter
#1	Acceleration time
#2	Deceleration time
#3	Interpolation acceleration time
#4	Interpolation deceleration time

The actual settings in the system parameters aren't changed, but the values set with this function take precedence. The settings units are ms.

When the operating mode is stop mode, the new acceleration/deceleration time will be reflected in the next operation.

When the operating mode is pass mode and the interpolation acceleration/deceleration time is changed during linear or circular interpolation, the new interpolation acceleration/deceleration time will be effective between G01, G02, and G03 functions following the G69 function.

The new parameter setting will be cleared when the first block of the next main program is executed.

If a position data address or register is specified for the new setting, the value will be rounded off to the nearest integer value. An error will occur if the specified data isn't within the range 0 to 9999.

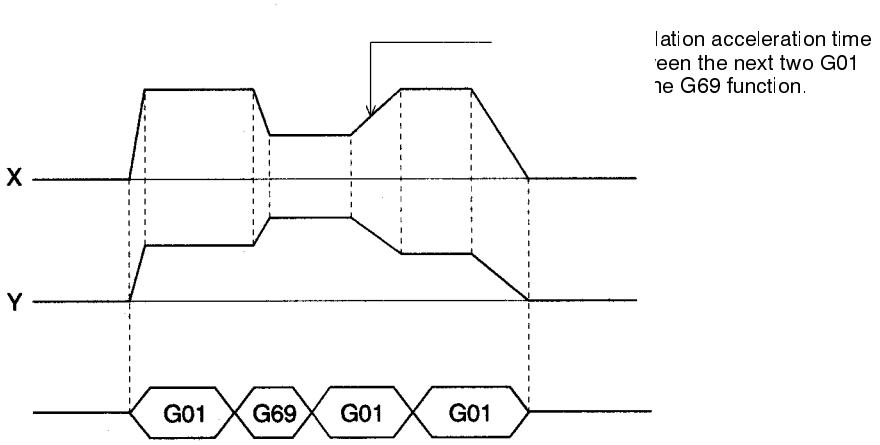
If this function is executed just after a G01, G02, or G03 function, the interpolation will be performed in stop mode, not pass mode, even if pass mode has been selected. The following example shows this effect.

This function can't change the present value for Absolute Encoder axes.

Example Program

The following program shows the effect of changing the interpolation acceleration time during interpolation.

```
      :     :
N009  G01   X200  Y300
N010  G69   #3/X100
N011  G01   X300  Y500
N012  G01   X500  Y800
      :     :
```



5-5-18 G70: UNCONDITIONAL JUMP

Unconditionally jumps to the specified block.

Format

G70_<jump destination block number> [/L<number of loops>]

Parameters

The following table shows the possible settings for the parameters.

Parameter	Possible settings
Destination block number	N000 to N999
Number of loops	1 to 39,999,999 (E00 to E31) A0000 to A1999

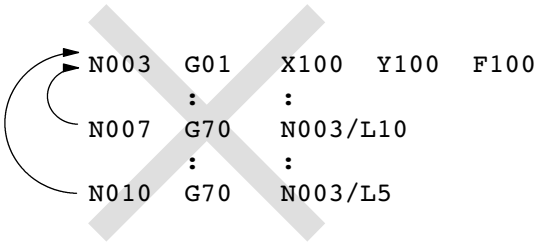
Description

This function unconditionally jumps to the specified block. A fixed number of jumps (a loop) can be created by specifying a number of loops.

When a number of loops is specified, the destination block number must be before the current block number.

If another UNCONDITIONAL JUMP function (G70) or a CONDITIONAL JUMP function (G71) is used to exit a loop before the number of loops has counted down to zero, the number of loops will be cleared when another loop is started.

A second loop function can't be executed while a loop is in progress. The following diagram demonstrates this programming mistake.



Loops can't be nested as shown above, but up to 6 loops can be nested by calling subprograms containing loops. The number of loops in each subprogram is cleared when the subprogram ends. Refer to *5-5-20 G72: Subprogram Jump* for details.

An error will occur if the position data is 0 when the number of loops is set indirectly from position data specified by a register or position data address. Non-integer position data values will be rounded off to the nearest integer value. An error will also occur if the position data is negative.

Example Program

```
      :     :
N020  G00   X100
N030  G71   N100/A1000=1
N040  G70   N020/L100 ..... [1]
      :     :
      :     :
N090  G00   X3200
N100  G70   N090/L50 ..... [2]
      :     :
```

In loop [1], the program will jump to N020 up to 100 times while A1000≠1, so blocks N020 and N030 will be executed up to 101 times.

In loop [2], the program will jump to N090 up to 50 times, so block N090 will be executed up to 51 times.

If A1000=1 on the 20th jump in loop [1], the program would jump to block N100. The remaining value of 80 jumps in the number of loops would be cleared and the number of loops would be set to the new value of 50 by block N100.

5-5-19 G71: CONDITIONAL JUMP

Jumps to the specified block when the condition is met.

Format

G71_<jump destination block number/condition equation>

Parameters

The following table shows the possible settings for the parameters.

Parameter		Possible settings
Destination block number		N000 to N999
Condition equation	First term	A0000 to A1999 E00 to E31
	Operator	=, <, >, or !
	Second term	A0000 to A1999 E00 to E31 –39,999,999 to +39,999,999

Description

This function jumps to the specified block when the given condition is met. The "!" operator is the inequality operator (not equal to). The first and second terms will be compared as real numbers, even if registers are specified.

Example Program

In the following program, A1000 is initialized to 0 in block N009 and then incremented by 1 in block N011 after the positioning operation in block N010. Block N012 causes the program to jump back to N010 and repeat blocks N010 to N012 ten times. The program proceeds when A1000=10.

```
      :     :
N008  G91
N009  G63   A1000=0
N010  G00   X500
N011  G60   A1000=A1000+1
N012  G71   N010/A1000 ! 10
      :     :
```

5-5-20 G72: SUBPROGRAM JUMP

Calls the specified subprogram.

Format

G72_<subprogram number>

Parameters

The following table shows the possible settings for the parameters.

Parameter	Possible settings
Subprogram number	P500 to P999

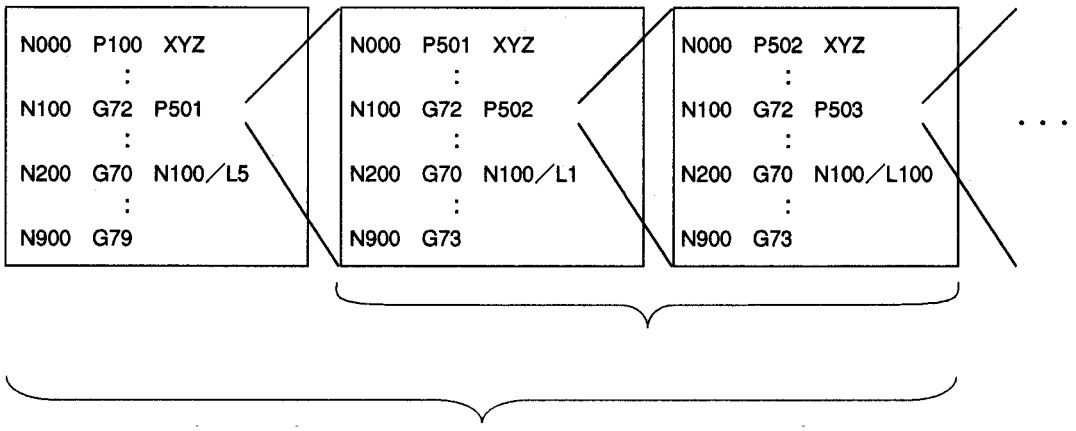
Description

This function calls and executes the specified subprogram. Subprograms can be nested 5 times. An error will occur if a sixth subprogram is called.

Example Program

```
      :     :
N010  G72   P500
      :     :
```

Up to 6 loops can be created by calling subprograms, as shown in the following diagram.



Counting the main program, up to 6 loops can be created with G70.

5-5-21 G73: SUBPROGRAM END

Ends the subprogram.

Format

G73

Parameters

None

Description

This function ends the subprogram and returns control to the block after the one that called the subprogram. This function must be programmed at the end of every subprogram.

Example Program

```
N000  P510  XYZ
      :     :
      :     :
      :     :
N100  G73
```

5-5-22 G74: OPTIONAL END

Ends the block currently being executed when the specified optional input goes ON.

Format

G74□<optional number>

Note The □ symbol indicates a space that must be included.

Parameters

The following table shows the possible settings for the parameters.

Parameter	Possible settings
Optional number	00 to 19

Description

If the specified input is received while the block after the one with this function is being executed, execution of that block will be interrupted and the program will proceed to execute the following block.

If an axis operation is being executed, the operation will be decelerated to a stop before proceeding to the next block. If a DWELL TIMER function (G04) is being executed, the function will be interrupted and the remaining time cancelled.

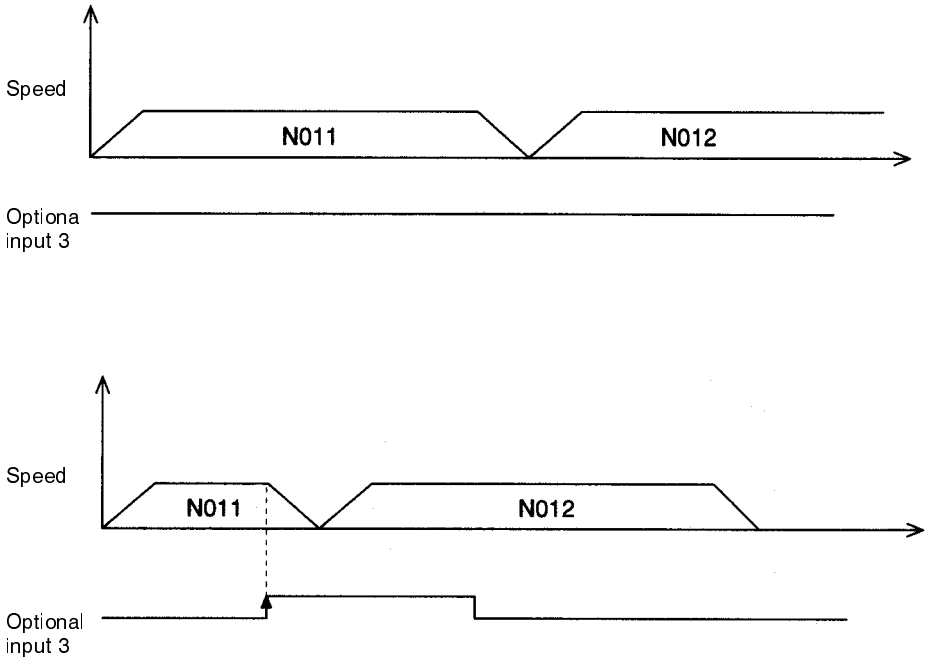
The source of the optional input depends on the optional number specified, as shown below.

0 through 15: Inputs from the PC's data area interface
16 through 19: Inputs from general inputs 1 through 4

This function is ineffective if the following block contains a SUBPROGRAM END function (73) or PROGRAM END function (79). If the specified optional input is ON already, this function will operate just like the OPTIONAL SKIP function (G75).

Example Program

```
      :     :
N009  G91
N010  G74   3
N011  G01   X100
N012  G01   X100
      :     :
```



5-5-23 G75: OPTIONAL SKIP

Skips the next block when the specified optional input is ON.

Format

G75□<optional number>

Note The □ symbol indicates a space that must be included.

Parameters

The following table shows the possible settings for the parameters.

Parameter	Possible settings
Optional number	00 to 19

Description

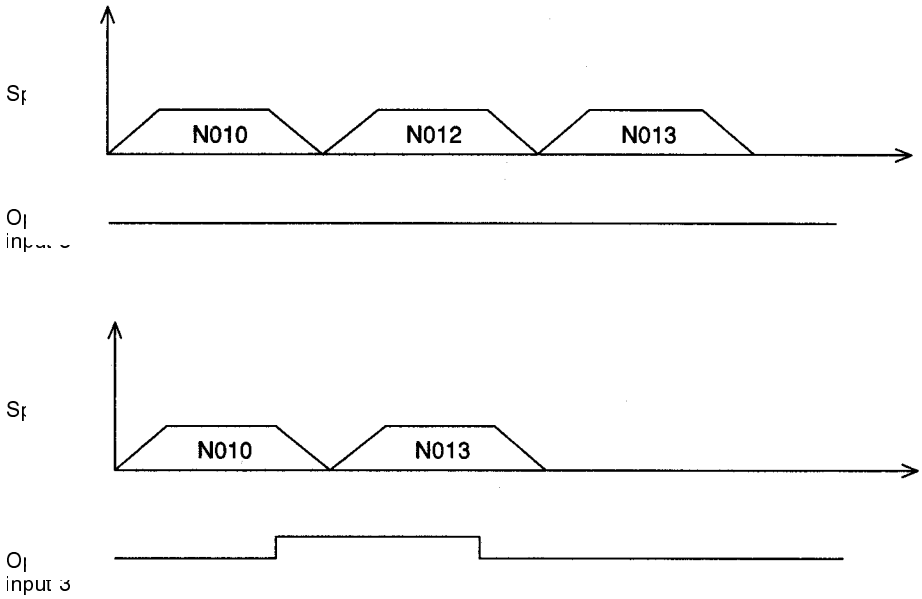
If the specified input is ON when this function is executed, the next block will be skipped. The source of the optional input depends on the optional number specified, as shown below.

0 through 15: Inputs from the PC's data area interface
16 through 19: Inputs from general inputs 1 through 4

Program execution will continue normally if the optional input comes ON while the next block is being executed. The OPTIONAL SKIP function is ineffective if the following block contains a SUBPROGRAM END function (73) or PROGRAM END function (79).

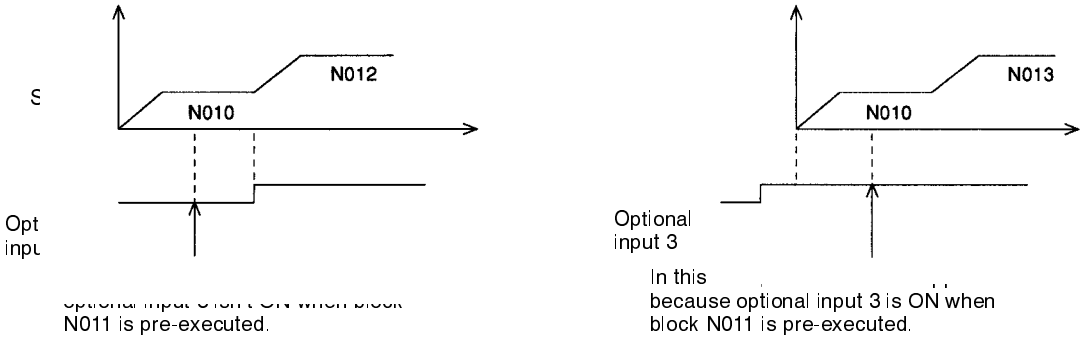
Example Program

```
        :     :
N008  G11
N009  G91
N010  G01   X50
N011  G75   3
N012  G01   X50
N013  G01   X50
        :     :
```



When block N008 contains the pass mode function (G10), blocks N011 and N012 are pre-executed while N010 is being executed, so block N012 won't be skipped if optional input 3 is turned ON after execution of block N010.

To ensure that block N012 will be skipped, make sure that optional input 3 is ON before block N010 is executed.



N011 is pre-executed.

because optional input 3 is ON when block N011 is pre-executed.

5-5-24 G76: OPTIONAL PROGRAM PAUSE

Pauses the program when the specified optional input is ON.

Format G76□<optional number>

Note The □ symbol indicates a space that must be included.

Parameters The following table shows the possible settings for the parameters.

Parameter	Possible settings
Optional number	00 to 19

Description

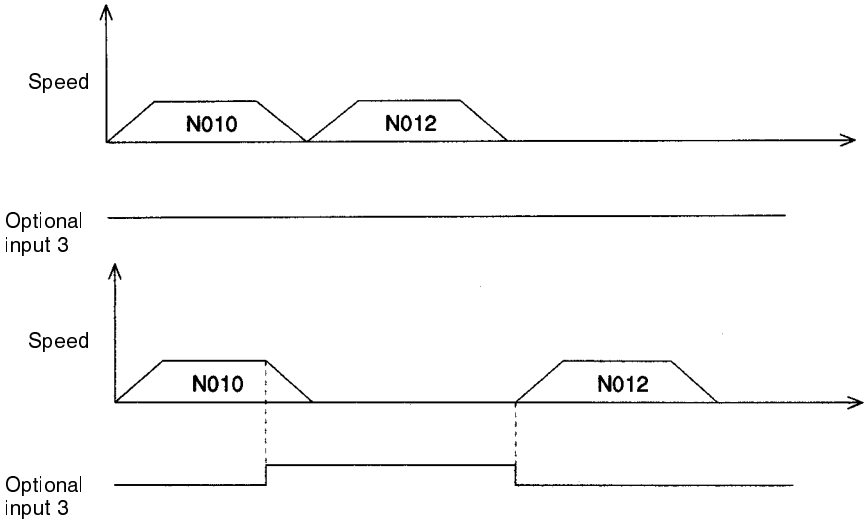
If the specified input is ON when this function is executed, program execution will be paused. Program execution will continue from the next block when the optional input is reset. The source of the optional input depends on the optional number specified, as shown below.

0 through 15: Inputs from the PC's data area interface
16 through 19: Inputs from general inputs 1 through 4

Program execution will continue normally if the optional input comes ON while the next block is being executed.

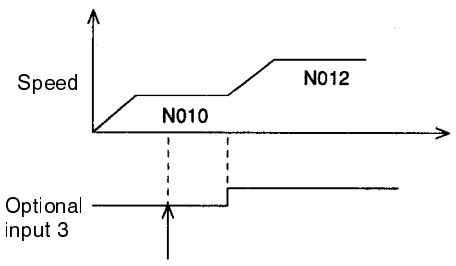
Example Program

```
      :     :
N008  G11
N009  G91
N010  G01   X100
N011  G76   3
N012  G01   X100
      :     :
```

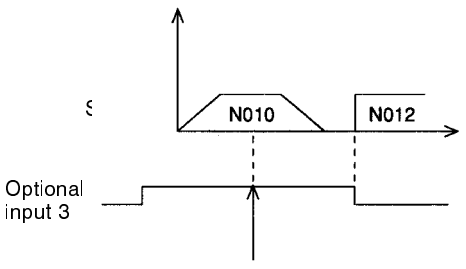


When block N008 contains the PASS MODE function (G10), blocks N011 and N012 are pre-executed while N010 is being executed, so the program won't be paused if optional input 3 is turned ON after execution of block N010.

To ensure that the program will be paused, make sure that optional input 3 is ON before block N010 is executed.



The program isn't paused because optional input 3 isn't ON when block N011 is pre-executed.



In this case, the program is paused because optional input 3 is ON when block N011 is pre-executed.

5-5-25 G79: PROGRAM END

Ends the main program.

Format

G79

Parameters

None

Description

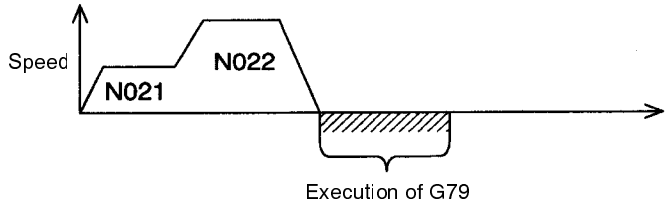
This function ends the main program and must be included at the end of the main program.

When G79 is executed and an axis is in operation, the Unit will wait for the axis to be positioned before executing G79. M codes M500 to M999 will be forcibly cleared if they are being output when G79 is executed.

Example Program

The following example shows how G79 is executed after axis operation is completed in pass mode.

```
        :      :
N020    G10                          Sets pass mode.
N021    G01    X100    Y100    F100
N022    G01    X200
N023    G79    X100                 Executed after N022 is completed.
        :      :
```



5-5-26 G90: ABSOLUTE SPECIFICATION

Specifies the use of absolute coordinates in axis operations.

Format

G90

Parameters

None

Description

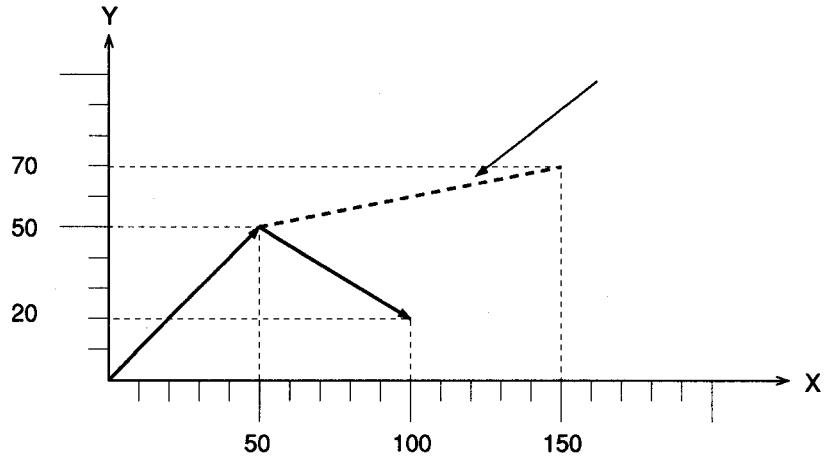
This function specifies that each axis' absolute coordinate system is to be used when positioning axis operations. After this function is executed, the coordinate data in axis movement commands is treated as absolute coordinate data.

In addition to G90, the absolute coordinate system is put into effect when a REFERENCE ORIGIN RETURN (G26), WORKPIECE ORIGIN RETURN (G27), or ORIGIN SEARCH (G28) function is executed.

Example Program

The following example shows the different axis operations with absolute and incremental specifications.

```
      :     :
N010  G90
N011  G01   X50   Y50
N012  G01   X100  Y20
      :     :
```



5-5-27 G91: INCREMENTAL SPECIFICATION

Specifies the use of relative coordinates in axis operations.

Format

G91

Parameters

None

Description

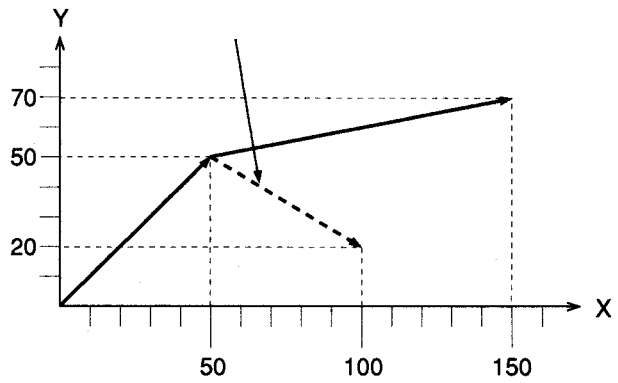
This function specifies that positioning axis operations are performed relative to the current position. After this function is executed, the coordinate data in axis movement commands is treated as the distance to be moved from the current position.

Example Program

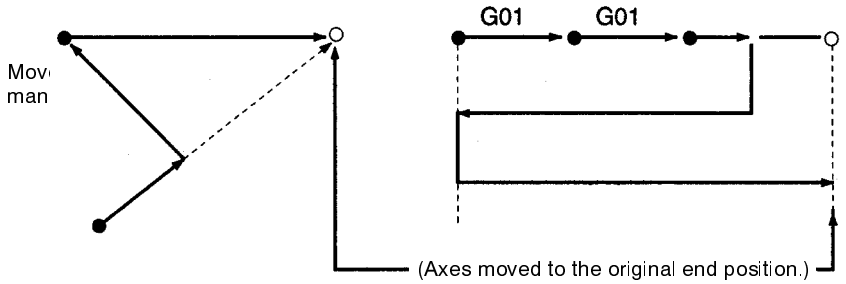
The following example shows the different axis operations with absolute and incremental specifications.

```
      :     :
N010  G91
N011  G01   X50   Y50
N012  G01   X100  Y20
      :     :
```

The axis operation would be performed



If positioning with incremental specification is interrupted by a pause command, the axes will be moved to the original end position after the operation is restarted. Also, if the axes have been moved or an origin search was performed after the operation was interrupted, the axes will still be moved to the original end position.



5-6 M-code Outputs

5-6-1 Introduction

M codes are information used to interlock with external devices in positioning operation processes. External devices are devices connected directly to the PC or the MC Unit's general outputs.

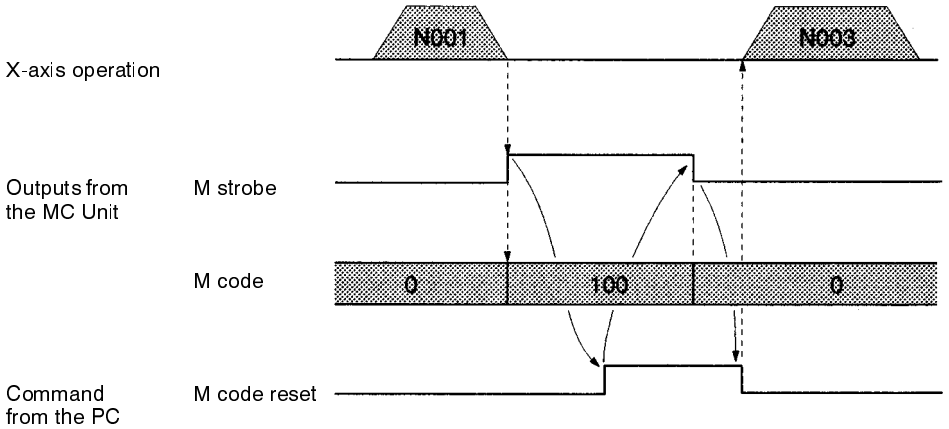
Example Program

The following program shows a practical example of M code usage.

```
N000  P100  X ..........  Declares the program number.
N001  G01   X100  F100.  Moves to 100 mm on the X-axis by linear
                         interpolation.
N002  M100 ..............  Outputs M code 100 to the PC and waits
                         for OK to perform the next operation.
                         Proceeds to next block when the M-code
                         reset is received from the PC.
N003  G01   X0 ..........  Moves to 0 mm on the X-axis by linear
                         interpolation.
N002  G79 ...............  Declares the end of the program.
```

Timing Chart

The following diagram shows the timing of the example program's execution. The M strobe, M code, and M code reset are provided in the PC data area interface.



Explanation

The following list provides a running explanation of the program's execution.

1, 2, 3...

1. In block N001, the X-axis is moved to 100 mm by linear interpolation.
2. In block N002, M code 100 is output after the X-axis positioning is completed; the M strobe is turned ON at the same time to indicate that the M code has been set.

3. In the PC, the processes corresponding to M code 100 are performed when the M strobe goes ON. The PC turns ON the M code reset when these processes are completed.
4. In the MC Unit, the M code is cleared to 0 and the M strobe is turned OFF when the PC turns ON the M code reset.
5. When the M code reset from the PC goes OFF, block N003 is executed, moving the X-axis to 0 mm.
6. The program ends in block N004.

5-6-2 M-code Data

There are some M codes that interlock as explained on the previous page, and others that don't interlock. Also, there are M codes that can handle general inputs as M code reset signals. These differences are explained below.

When an M code is specified in a register or position data address, the specified value will be rounded to the nearest integer value. An error will occur if the result isn't an acceptable M code value (0 to 999).

M Codes 0 to 499

M codes 0 to 499 are used to interlock with external devices. Execution of the program is paused when there isn't an M code reset input. These M codes can be divided into two groups, described below.

M Codes 0 to 495

M codes 0 to 495 use an M code reset signal from the PC. These M codes are output to the PC only.

M Codes 496 to 499

M codes 496 to 499 use general inputs in the MC Unit as M code reset signals. These M codes are output to the PC and cause the MC Unit's corresponding general outputs to go ON.

Use these M codes when interlocking with high-speed external devices. The following table shows the general inputs and outputs that correspond to M codes 495 to 499.

M code	M code reset signal	Output
496	General input 1	General output 1
497	General input 2	General output 2
498	General input 3	General output 3
499	General input 4	General output 4

Note These M codes can't be cleared to 0 by an M code reset from the PC.

M Codes 500 to 999

Use M codes 500 to 999 when it isn't necessary to interlock. These M codes are just output and the program is executed without waiting for an M code reset input. These M codes can be divided into two groups, described below.

M Codes 500 to 995

These M codes are output to the PC only.

M Codes 996 to 999

M codes 996 to 999 are output to the PC and cause the MC Unit's corresponding general outputs to go ON. The following table shows the general outputs that correspond to M codes 995 to 999.

M code	Output
996	General output 1
997	General output 2
998	General output 3
999	General output 4

These M codes are cleared when the PROGRAM END function (G79) is executed.

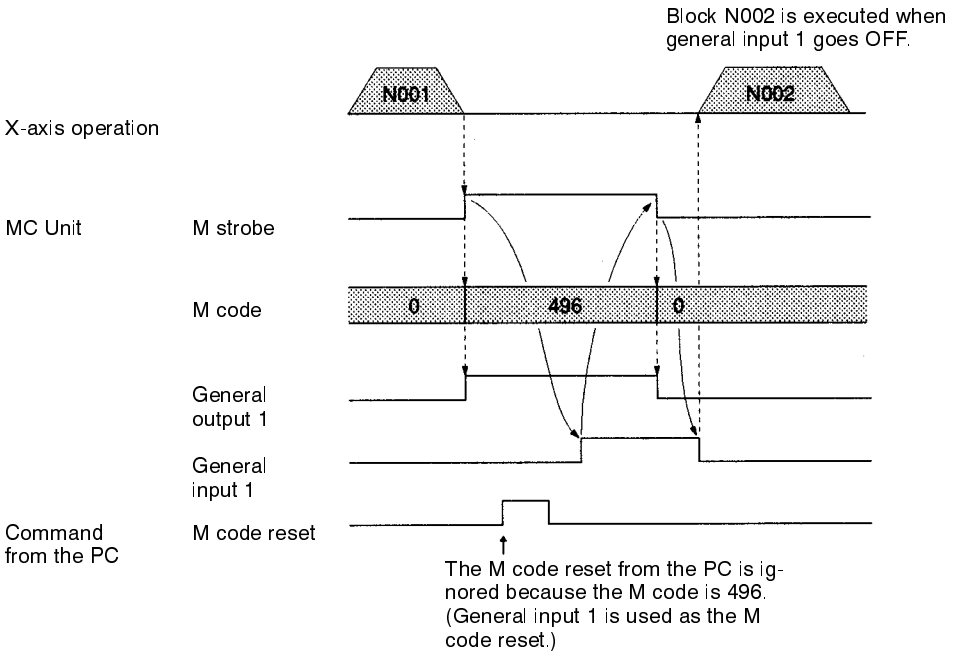
5-6-3 M-code Examples

M codes can be used independently or in conjunction with "G codes" such as G01 and G02 which execute axis operations.

Example 1

The following program outputs M code 496. M code 496 interlocks through a general input.

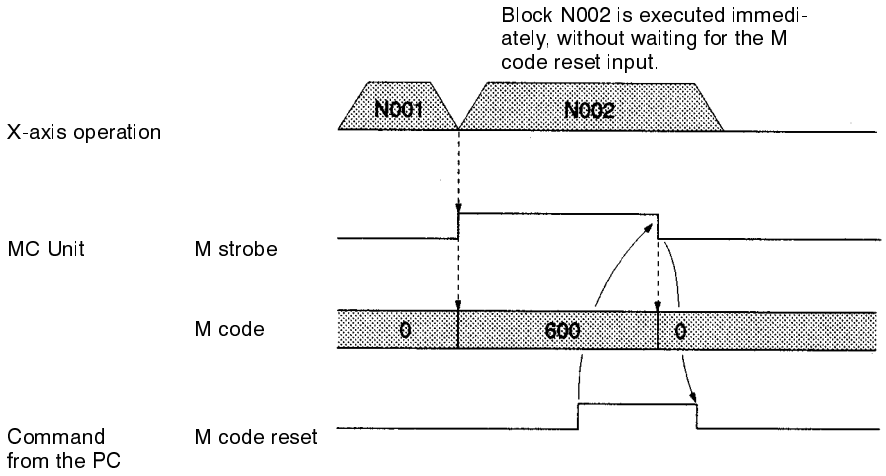
```
N000  P000  X
N001  G01   X100  F100  M496
N002  G01   X200  F100
N003  G79
```



Example 2

The following program just outputs M code 600. (M codes over 499 don't interlock.)

```
N000  P000  X
N001  G01   X100  F100  M600
N002  G01   X200  F100
N003  G79
```

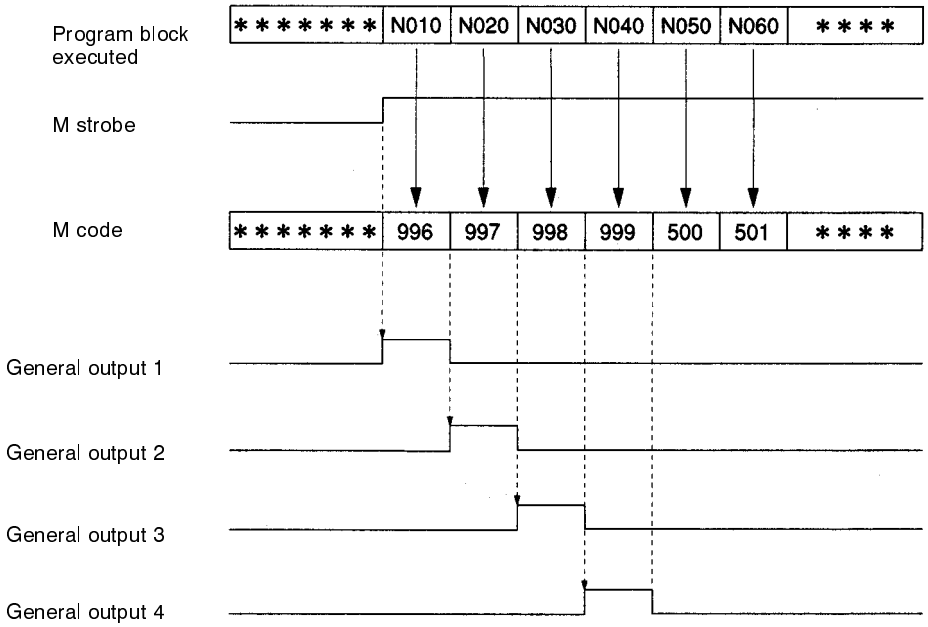


Example 3

The following program section turns general ON and OFF.

```
:     :
N010  M996
N020  M997
N030  M998
N040  M999
N050  M500
N060  M501
:     :
```

The previous M code is overwritten when a new M code is output. If the M code being output is 996 to 999, the corresponding general output will be turned ON, as shown in the following diagram.



5-6-4 M Code Resets

M Code Resets from the PC

Depending on the status of the MC Unit, M code resets from the PC might or might not be able to reset the M code.

Mode		Explanation
Manual mode		Valid for M500 to M999
Automatic mode	Program not being executed	Valid for M500 to M999
	Program being executed	Valid for all M codes except M496 to M499. For M codes M496 to M499, the reset is performed by the corresponding general input signal.

M Code Resets from General Inputs

M code resets from general inputs also depend on the status of the MC Unit.

Mode		Explanation
Manual mode		Not valid for resetting any M codes
Automatic mode	Program not being executed	Not valid for resetting any M codes
	Program being executed	Valid for M codes M496 to M499

5-6-5 Effect of Mode Changes on M Code Resets

The status of M codes isn't affected by switching from manual to automatic mode or vice-versa.

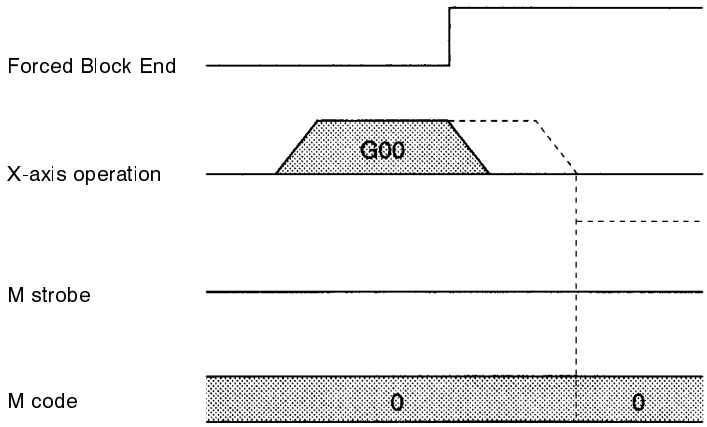
5-6-6 Clearing M000 to M499 with Forced Block End or G74

The M strobe and M code outputs for M codes 000 to 499 will be cleared if the program block that outputs the M code is cleared with the Forced Block End signal (PC data area interface bit) or the OPTIONAL END function (G74).

Example 1

The M strobe and M code won't be output if a Forced Block End signal is received before they are output.

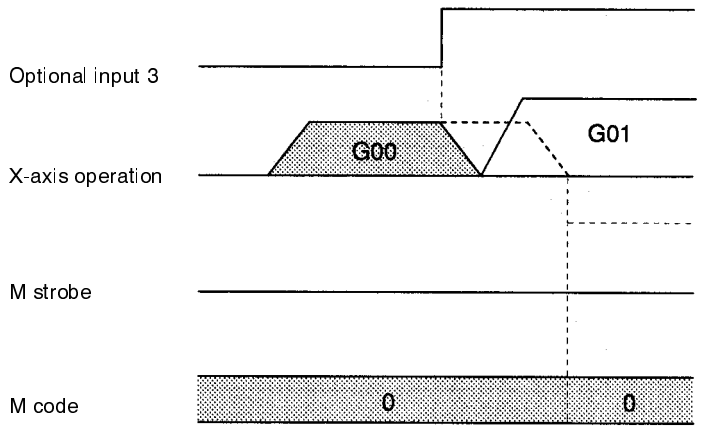
```
G00    X500   M100
```



Example 2

The M strobe and M code won't be output if the OPTIONAL END function's optional input is received before they are output.

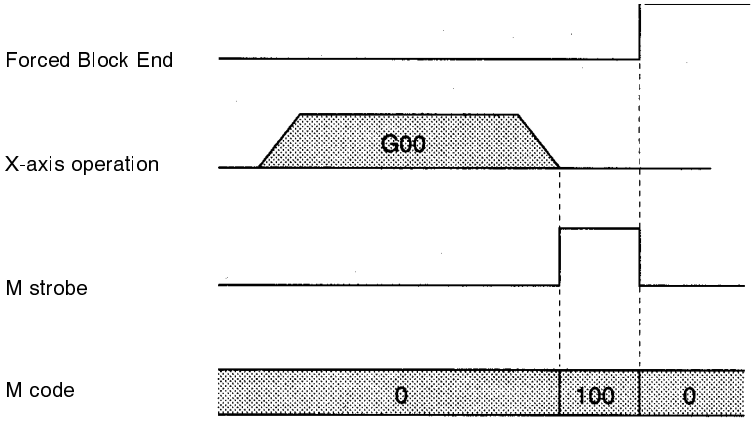
```
G74    3
G00    X500   M100
G01    X1000
```



Example 3

The M strobe and M code will be cleared if a Forced Block End signal is received while they are being output.

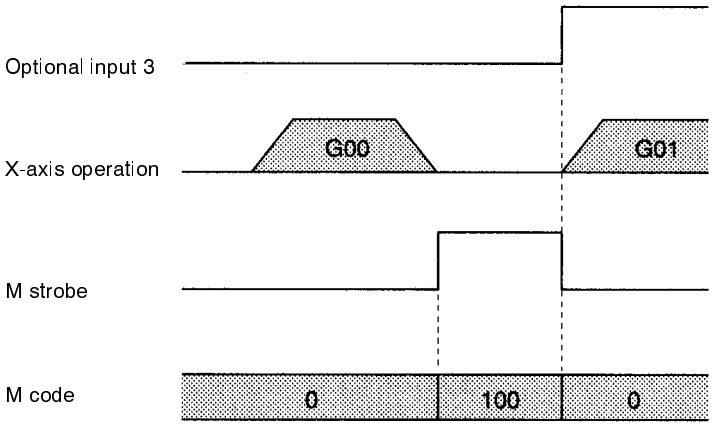
```
G00    X500   M100
```



Example 4

The M strobe and M code will be cleared if the OPTIONAL END function's optional input is received while they are being output.

```
G74    3
G00    X500   M100
G01    X1000
```



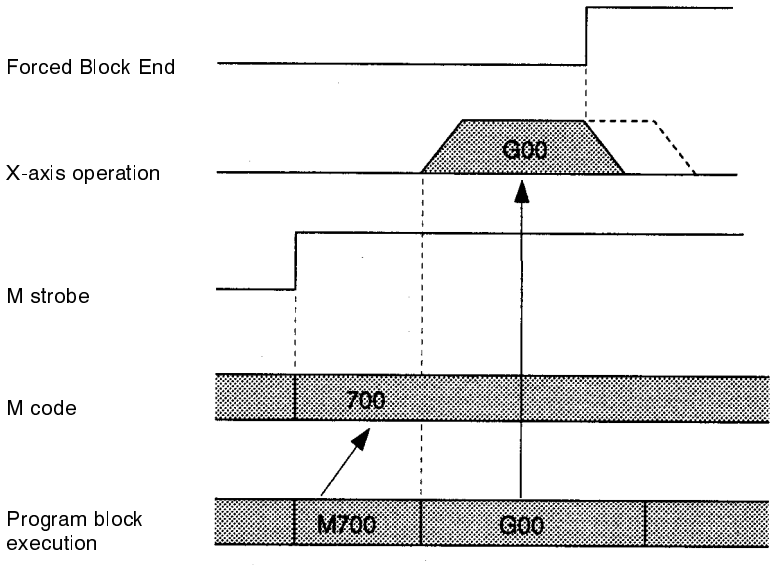
5-6-7 Clearing M500 to M999 with Forced Block End or G74

The M strobe and M code outputs for M codes 500 to 999 won't be output cleared if the program block that outputs the M code is cleared with the Forced Block End signal (PC data area interface bit) or the OPTIONAL END function (G74), but previous M codes won't be cleared.

Example 5

The M strobe and M code won't be output if a Forced Block End signal is received before they are output, but a previous M code and M strobe won't be cleared.

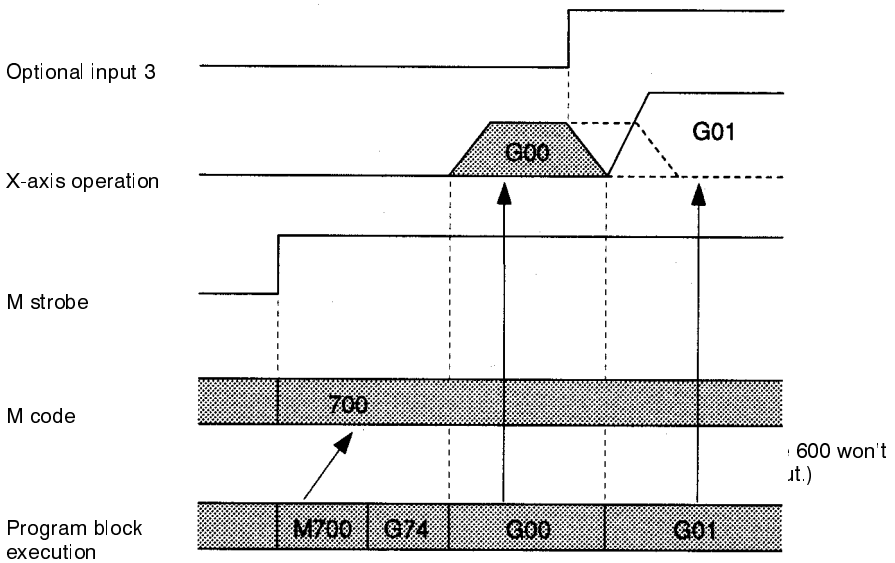
```
M700
G00    X500   M600
```



Example 6

The M strobe and M code won't be output if the Optional End function's optional input is received before they are output, but a previous M code and M strobe won't be cleared.

```
M700
G74    3
G00    X500   M600
G01    X1000
```



5-7 Mode Transitions Caused by G Functions

Execution of some G-code functions will change the MC Unit's modes and coordinate settings. The following table shows the operations and functions that can change the MC Unit's status.

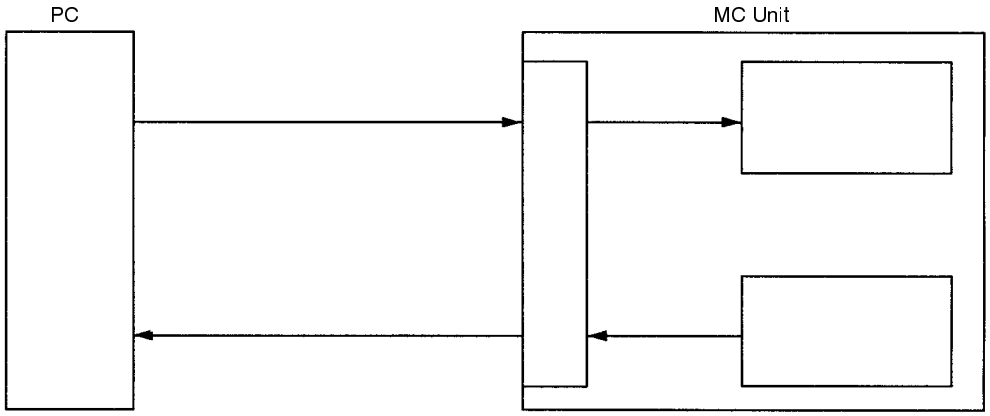
Operation	G code	Status after the operation		
		Coordinate system	Command mode	Operating Mode
Reset	---	Reference	Absolute	Pass mode
Execution of the main program	---	Reference	Absolute	Pass mode
ORIGIN SEARCH	G28	Reference	Absolute	Pass mode
REFERENCE ORIGIN RETURN	G26	Reference	Absolute	Pass mode
WORKPIECE ORIGIN RETURN	G27	Workpiece	Absolute	Pass mode

SECTION 6
PC Data Area Interface

This section describes the interface area used to exchange information between the PC and MC Unit, such as commands from the PC and status information from the MC Unit.

6-1 Introduction

The area used to exchange data (such as commands from the PC, tasks from the MC Unit, and axis status information) between the PC and MC Unit is known as the PC data area interface.



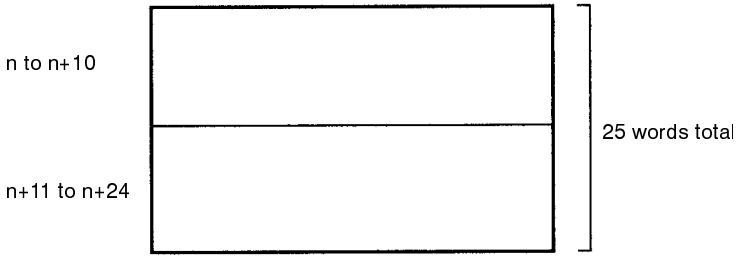
In order to operate the PC data area interface from the PC, allocate CIO and DM words and create a ladder program to process the data. Refer to *6-3 Controlling the MC Unit from the PC* for details on creating a ladder program to control the MC Unit.

6-2 Allocation of the PC Data Area Interface

The PC data area interface used by the MC Unit is composed of 25 CIO Area words and 100 DM Area words. The same number of words are allocated to CV500-MC221 and CV500-MC421 MC Units.

6-2-1 CIO Area Words

The 25 CIO Area words are divided into 11 output words (PC → MC Unit) and 14 input words (PC ← MC Unit).



The unit number set on the MC Unit determines which 25 CIO Area words are allocated to the Unit: n = 1500 + (25 × unit number). Refer to *1-3 Setting the Unit Number* for details.

CIO Area Word Assignment

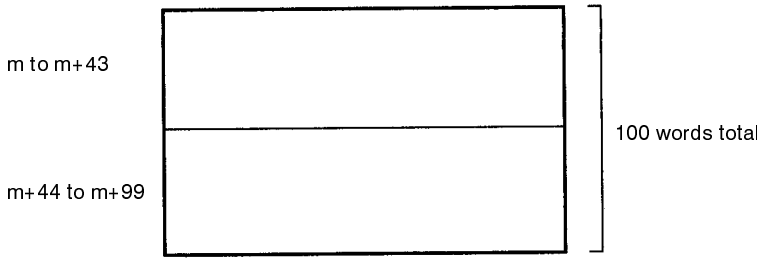
The following diagram shows the specific functions performed by the CIO Area words. CIO words are processed in bit units, so they are known as interface bits. Refer to *6-4 Interface Bits* for details on the use of these bits. (Words with asterisks are not used in the CV500-MC221.)

I/O usage	Word	Function
Output area (PC → MC Unit)	n	Control bits for all tasks
	n+1	Optional inputs for all tasks
	n+2	Control bits for task 1
	n+3	Control bits for task 2
	n+4	*Control bits for task 3
	n+5	*Control bits for task 4
	n+6	Not used
	n+7	X-axis control bits
	n+8	Y-axis control bits
	n+9	*Z-axis control bits
	n+10	*U-axis control bits
Input area (MC Unit → PC)	n+11	Status flags for all tasks
	n+12	General I/O for all tasks
	n+13	Task 1 status flags
	n+14	Task 2 status flags
	n+15	*Task 3 status flags
	n+16	*Task 4 status flags
	n+17	X-axis status flags
	n+18	
	n+19	Y-axis status flags
	n+20	
	n+21	*Z-axis status flags (See note.)
	n+22	
	n+23	*U-axis status flags (See note.)
	n+24	

Note The following flags can be used in the CV500-MC221: Transmission Data Completed, Reception Data Completed, and Transfer Data Error.

6-2-2 DM Area Words

The 100 DM Area words are divided into 44 output words (PC → MC Unit) and 56 input words (PC ← MC Unit).



The unit number set on the MC Unit determines which 100 DM Area words are allocated to the Unit: m = 2000 + (100 × unit number). Refer to *1-3 Setting the Unit Number* for details.

DM Area Word Usage

The following diagram shows the specific functions performed by the DM Area words. DM words are processed in word units only. Refer to *6-5 DM Word Usage* for details on the use of these words. (Words with asterisks are not used in the CV500-MC221.)

I/O Usage	Words	Function
Output area (PC → MC Unit)	m to m+3	Transfer Address #1 Transfer #1 Auxiliary Information Transmission Data #1
	m+4 to m+7	Transfer Address #2 Transfer #2 Auxiliary Information Transmission Data #2
	m+8 to m+11	Transfer Address #3 Transfer #3 Auxiliary Information Transmission Data #3
	m+12 to m+15	Transfer Address #4 Transfer #4 Auxiliary Information Transmission Data #4
	m+16 to m+18	Task 1 command words
	m+19 to m+21	Task 2 command words
	m+22 to m+24	*Task 3 command words
	m+25 to m+27	*Task 4 command words
	m+28 to m+30	Not used
	m+31 to m+33	X-axis command words
	m+34 to m+36	Y-axis command words
	m+37 to m+39	*Z-axis command words
	m+40 to m+42	*U-axis command words
	m+43	Not used
Input area (MC Unit → PC)	m+44 to m+65	System version number System error code Task error code Axis error code Reception #1 to #4 Auxiliary Information Reception Data #1 to #4
	m+66 to m+70	Task 1 status words
	m+71 to m+75	Task 2 status words
	m+76 to m+80	*Task 3 status words
	m+81 to m+85	*Task 4 status words
	m+86	Not used
	m+87 to m+89	X-axis status words
	m+90 to m+92	Y-axis status words
	m+93 to m+95	Z-axis status words
	m+96 to m+98	U-axis status words
	m+99	Not used

6-3 Controlling the MC Unit from the PC

This section describes the PC's ladder programs that control the MC Unit in manual mode or automatic mode.

Manual and Automatic Modes There are two ways to control the MC Unit: manual mode or automatic mode. Automatic mode uses the G-language program in the MC Unit.

In manual mode, the following functions can be used to control each axis. These functions are called commands here. Refer to *6-6 Interface Bit Specifics* for details on the control bits that execute each of these commands.

Deceleration Stop
Origin Search
Reference Origin Return
Jogging
Current Position Preset
Enable MPG
Servo Lock
Servo Free

6-3-1 Manual Mode

Each command is executed on the up-differentiation (OFF → ON transition) of its control bit. The following shows the commands that have a function when their control bits are OFF.

Command	ON function	OFF function
Deceleration Stop	Stops other commands.	---
Origin Search	Continues operation.	Stops operation.
Reference Origin Return		
Jogging		
Enable MPG		

Executing the Deceleration Stop command has no effect when no other commands are being executed, but no other commands can be received while the Deceleration Stop command is left ON.

The Origin Search, Reference Origin Return, Jogging, and Enable MPG commands are executed on up-differentiation and continue operation while the control bit is ON. To interrupt operation, turn the control bit OFF or execute the Deceleration Stop command.

Operation of the Current Position Preset, Set ABS Defaults, ABS Soft Reset, Servo Lock, and Servo Free commands can't be stopped. These commands are processed to completion on the up-differentiation of the corresponding control bits.

Command Priority When more than one command is executed at the same time, the commands will be processed in the following order.

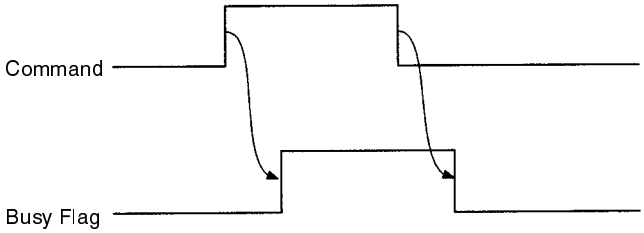
1, 2, 3...

1. Deceleration Stop
2. Origin Search
3. Reference Origin Return
4. Jogging
5. Current Position Preset
6. Set ABS Defaults
7. ABS Soft Reset
8. Enable MPG
9. Servo Lock
10. Servo Free

Deceleration Stop has the highest priority and Servo Free has the lowest.

Busy Flag

The Busy Flag is turned ON when a command is being executed, and turned OFF when the command is completed. Deceleration Stop is the only command that can be received while this flag is ON.



It is recommended to write a sequence program that will prevent other commands from being executed while the commands that can't be interrupted (Current Position Preset, Set ABS Defaults, ABS Soft Reset, Servo Lock, and Servo Free) are being executed.

⚠ **Caution** After executing a command, do not execute another command until the first command is completed and the Busy Flag is turned OFF.

6-3-2 Automatic Mode

The MC program is executed in automatic mode. Use the following procedure to operate in automatic mode. This example is for task 1.

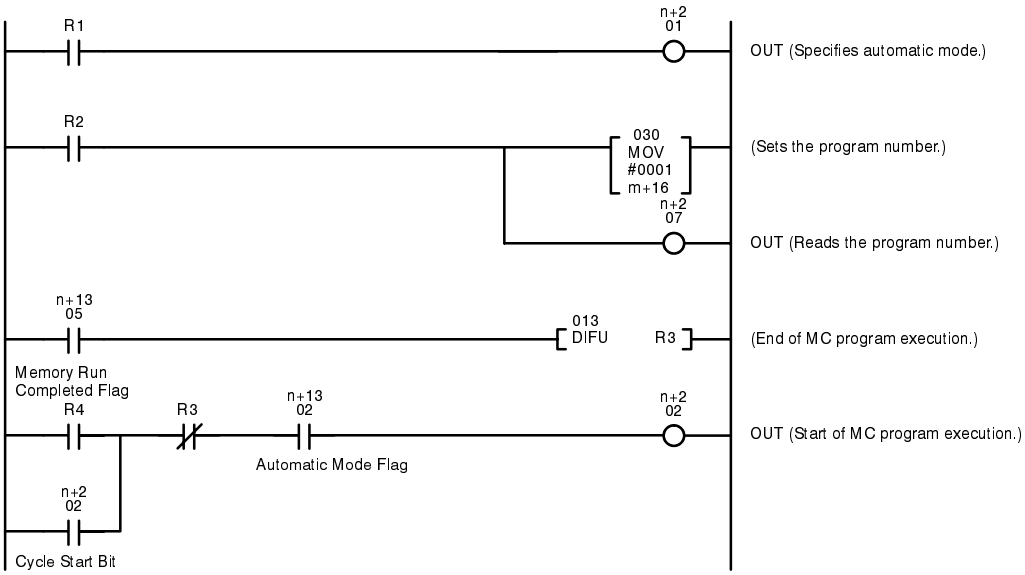
1, 2, 3...

1. Set the MC Unit to automatic mode. Turn ON bit 01 of word n+2.
2. Set the program number of the program you want to execute. (Word m+16)
3. Turn ON the Program Number Read bit. Turn ON bit 07 of word n+2.
4. The specified program will be executed when the Cycle Start bit (bit 02 of word n+2) is turned ON.

Refer to *6-6 Interface Bit Specifics* for details on the Program Read, Cycle Start, and Automatic/Manual Mode control bits. Refer to *6-5 DM Word Usage* for details on specifying the program number.

6-3-3 Sample Ladder Program

The following sample program incorporates the procedure to execute the MC program in automatic mode. This program is the minimum required to execute the MC program. Add the program elements needed for your actual application.



Use Auxiliary Area bits for inputs R1, R2, R3, and R4. The first words in the CIO Area (n) and DM Area (m) can be calculated from the following equations. Refer to *1-3 Setting the Unit Number* for details on unit number settings.

n = 1500 + (25 × unit number)
m = 2000 + (100 × unit number)

In the sample program, program number P001 is executed in task 1. Change the task and program number if necessary.

Program Flow

Turn ON input R1 to specify automatic mode.

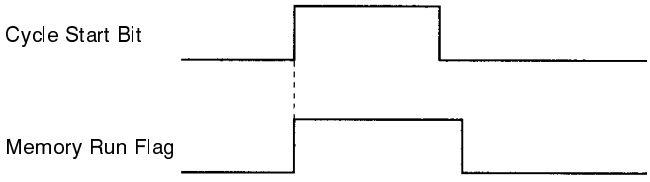
Turn ON input R2 in the following situations:

- The first time that the program is executed after the power is turned ON.
- When executing the program from the beginning.
- When the program number has been changed.

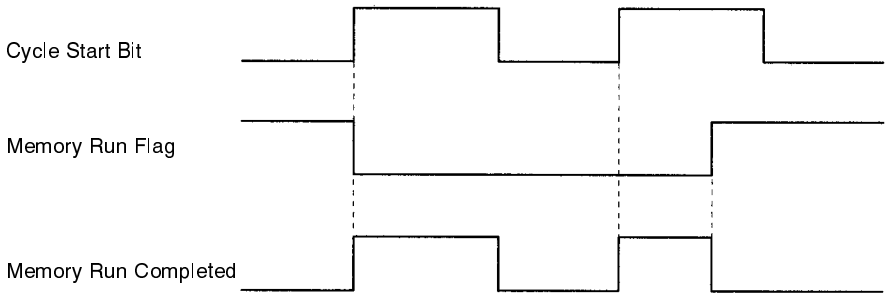
Execution of the MC program will begin when R4 goes ON.

Memory Run Flag

The Memory Run Flag is equivalent to the manual mode's Busy Flag. This flag is ON while the MC program is being executed; it is OFF when program execution is completed or stopped.



Memory Run Completed Flag The Memory Run Completed Flag goes ON when the MC program is completed. The Memory Run Flag will go OFF when this bit is turned ON, even if the Cycle Start Bit remains ON.



6-4 Interface Bits

This section lists the bits and flags in the CIO Area of the PC data area interface. Refer to *6-6 Interface Bit Specifics* for more details.

The "signal" column indicates whether the bit or flag operates when ON, OFF, or at up or down-differentiation. Up-differentiation is indicated by an up arrow (↑) and down-differentiation is indicated by a down arrow (↓).

6-4-1 Output Words (PC → MC Unit)

Word n: Control Bits for All Tasks

Bit(s)	Name	Signal	Function	Page
00	System Error Reset Bit	↑	Resets system errors.	106
01	Transmission #1 Control Bit	↑	Starts transmission #1.	107
02	Reception #1 Control Bit	↑	Starts reception #1.	108
03	Transmission #2 Control Bit	↑	Starts transmission #2.	107
04	Reception #2 Control Bit	↑	Starts reception #2.	108
05	Transmission #3 Control Bit	↑	Starts transmission #3.	107
06	Reception #3 Control Bit	↑	Starts reception #3.	108
07	Transmission #4 Control Bit	↑	Starts transmission #4.	107
08	Reception #4 Control Bit	↑	Starts reception #4.	108
09 to 15	Not used	---	---	---

Word n+1: Optional Inputs for All Tasks

Bit	Name	Signal	Function	Page
00	Optional input 0	0 or 1	0: Optional input 0 is OFF. 1: Optional input 0 is ON.	109
01	Optional input 1	0 or 1	0: Optional input 1 is OFF. 1: Optional input 1 is ON.	
02	Optional input 2	0 or 1	0: Optional input 2 is OFF. 1: Optional input 2 is ON.	
03	Optional input 3	0 or 1	0: Optional input 3 is OFF. 1: Optional input 3 is ON.	
04	Optional input 4	0 or 1	0: Optional input 4 is OFF. 1: Optional input 4 is ON.	
05	Optional input 5	0 or 1	0: Optional input 5 is OFF. 1: Optional input 5 is ON.	
06	Optional input 6	0 or 1	0: Optional input 6 is OFF. 1: Optional input 6 is ON.	

Bit	Name	Signal	Function	Page
07	Optional input 7	0 or 1	0: Optional input 7 is OFF. 1: Optional input 7 is ON.	109
08	Optional input 8	0 or 1	0: Optional input 8 is OFF. 1: Optional input 8 is ON.	
09	Optional input 9	0 or 1	0: Optional input 9 is OFF. 1: Optional input 9 is ON.	
10	Optional input 10	0 or 1	0: Optional input 10 is OFF. 1: Optional input 10 is ON.	
11	Optional input 11	0 or 1	0: Optional input 11 is OFF. 1: Optional input 11 is ON.	
12	Optional input 12	0 or 1	0: Optional input 12 is OFF. 1: Optional input 12 is ON.	
13	Optional input 13	0 or 1	0: Optional input 13 is OFF. 1: Optional input 13 is ON.	
14	Optional input 14	0 or 1	0: Optional input 14 is OFF. 1: Optional input 14 is ON.	
15	Optional input 15	0 or 1	0: Optional input 15 is OFF. 1: Optional input 15 is ON.	

Word n+2: Control Bits for Task 1

Bit(s)	Name	Signal	Function	Page
00	Task Error Reset Bit	↑	Resets task errors.	109
01	Automatic/Manual Mode	↑	Any axes operating in manual mode will decelerate to a stop.	110
		1	Automatic mode	
		↓	When the MC program is being executed, it will be stopped and any operating axes will decelerate to a stop.	
		0	Manual mode	
02	Cycle Start Bit	↑	Starts MC program execution.	111
		1	Continues MC program execution.	
		0	Stops MC program execution.	
03	Single Block	1	Executes a single block.	113
04	Forced Block End	↑	Forces an end to the block.	114
		1	Prevents cycle start.	
05	Pause	↑	Pauses execution.	115
		1	Prevents cycle start.	
06	M Code Reset Bit	↑	Resets the M code.	116
		1	M code reset standby. (Prevents M code output.)	
		↓	Clears M code reset standby.	
07	Program Number Read Bit	1	Reads the program number.	113
08 to 11	Not used	---	---	---
12	Teaching Address Set Bit	↑	Sets the address when teaching.	117
13	Teaching Bit	↑	Starts teaching.	118
14	Teaching Type Bit	1	Current position (feedback value)	118
		0	Target position	
15	Not used	---	---	---

Word n+3: Control Bits for Task 2

Bit(s)	Function
00 to 15	The functions of task 2's control bits are the same as task 1's (word n+2).

Word n+4: Control Bits for Task 3

Bit(s)	Function
00 to 15	The functions of task 3's control bits are the same as task 1's (word n+2).

Word n+5: Control Bits for Task 4

Bit(s)	Function
00 to 15	The functions of task 4's control bits are the same as task 1's (word n+2).

Word n+6: Not used

Bit(s)	Function
00 to 15	Not used

Word n+7: X-Axis Control Bits

Bit	Name	Signal	Function	Page
00	Deceleration Stop	↑	Deceleration Stop	119
		1	Prevents other manual commands.	
01	Origin Search	↑	Starts the origin search.	120
		1	Continues the origin search.	
		↓	Stops the origin search.	
02	Reference Origin Return	↑	Starts moving the axis to the reference origin.	122
		1	Continues movement.	
		↓	Stops movement.	
03	Jogging	↑	Starts jog operation.	122
		1	Continues jog operation.	
		↓	Stops jog operation.	
04	Not used	---	---	---
05	Current Position Preset	↑	Presets the current position.	123
06	Reserved	---	---	---
07	Reserved	---	---	---
08	Enable MPG	↑	Starts MPG operation.	124
		1	Continues MPG operation.	
		↓	Stops MPG operation.	
09	Servo Lock	↑	Starts servo lock.	126
10	Servo Free	↑	Clears the servo lock status.	126
11	Driver Alarm Reset	1	Turns ON the driver alarm reset output.	127
		0	Turns OFF the driver alarm reset output.	
12	Override Setting	1	Enables the override setting.	128
		0	Disables the override setting.	
13	Jog direction	1	Specifies negative direction for jogging.	123
		0	Specifies positive direction for jogging.	
14 and 15	MPG Ratio Specifiers	Bits 14, 15:	0, 0 sets MPG ratio (1) 1, 0 sets MPG ratio (2) 0, 1 sets MPG ratio (3) 1, 1 sets MPG ratio (4)	125

Word n+8: Y-Axis Control Bits

Bit(s)	Function
00 to 15	The Y-axis' control bit functions are the same as the X-axis' (word n+7).

Word n+9: Z-Axis Control Bits

Bit(s)	Function
00 to 15	The Z-axis' control bit functions are the same as the X-axis' (word n+7).

Word n+10: U-Axis Control Bits

Bit(s)	Function
00 to 15	The U-axis' control bit functions are the same as the X-axis' (word n+7).

6-4-2 Input Words (MC Unit → PC)

Word n+11: Status Flags for All Tasks

Bit(s)	Name	Signal	Meaning	Page
00	System Error Flag	1	Indicates a system error occurred.	129
		0	Normal	
01	System Error Reset Completed	↑	System error reset has been completed.	129
		↓	Off when the system error reset signal goes OFF.	
02 to 06	Not used	---	---	---
07	Battery Alarm Flag	1	The battery voltage has dropped below the rated value.	130
		0	Normal	
08 to 12	Not used	---	---	---
13	Teaching Box Enabled	1	The Teaching Box is in Enabled mode.	130
		0	Teaching Box not in Enabled mode.	
14	Teaching Box Protected	1	The Teaching Box is in Protect mode.	130
		0	Teaching Box not in Protect mode.	
15	Teaching Box Connected	1	The Teaching Box is connected.	131
		0	The Teaching Box isn't connected.	

Word n+12: General I/O for All Tasks

The general inputs and outputs indicate the status of I/O signals in the MC Unit itself. The status of these I/O signals can be displayed on-screen by selecting "Display MC I/O status" from the MC Monitoring menu; they can also be monitored from the Teaching Box. Refer to *13-6 Displaying MC I/O Status* in the *MC Support Software Operation Manual* for more details.

Bit(s)	Name	Signal	Meaning	Page
00	General input 1	0 or 1	0: General input 1 is OFF. 1: General input 1 is ON.	131
01	General input 2	0 or 1	0: General input 2 is OFF. 1: General input 2 is ON.	
02	General input 3	0 or 1	0: General input 3 is OFF. 1: General input 3 is ON.	
03	General input 4	0 or 1	0: General input 4 is OFF. 1: General input 4 is ON.	
04	General output 1	0 or 1	0: General output 1 is OFF. 1: General output 1 is ON.	131
05	General output 2	0 or 1	0: General output 2 is OFF. 1: General output 2 is ON.	
06	General output 3	0 or 1	0: General output 3 is OFF. 1: General output 3 is ON.	
07	General output 4	0 or 1	0: General output 4 is OFF. 1: General output 4 is ON.	
08 to 15	Not used	---	---	---

Word n+13: Task 1 Status Flags

The status of the Memory Run and Memory Run Completed Flags can be displayed on-screen by selecting "Display MC I/O status" from the MC Monitoring menu. Refer to *13-6 Displaying MC I/O Status* in the *MC Support Software Operation Manual* for more details.

Bit	Name	Signal	Meaning	Page
00	Task Error Flag	1	A task error occurred.	131
		0	Normal	
01	Task Error Reset Completed	↑	Task error reset processing has been completed.	132
		↓	The Task Error Reset Bit is OFF.	
02	Automatic Mode Flag	1	The Unit is in automatic mode.	132

Bit	Name	Signal	Meaning	Page
03	Manual Mode Flag	1	The Unit is in manual mode.	132
04	Memory Run Flag	1	The MC program is being executed.	133
05	Memory Run Completed Flag	1	MC program execution has been completed. (G79 was executed.)	133
06	Optional Input Standby	1	Waiting for optional input.	134
07	M Code Reset Standby	1	Waiting for M code reset.	134
08	M Strobe	↑	An M code (0 to 999) is being output.	134
		↓	M code reset	
09	Cycle Start Received	↑	The Cycle Start Signal was received.	135
		↓	The Cycle Start Signal is OFF.	
10	Not used	---	---	---
11	Not used	---	---	---
12	Teaching Address Setting Completed	↑	The teaching address has been set.	136
		↓	Teaching address set command is OFF.	
13	Teaching Completed	↑	Teaching has been completed.	136
		↓	Teaching command is OFF.	
14	Teaching Error	↑	A teaching error occurred.	137
		↓	OFF when the next teaching command is completed normally or teaching address setting is completed.	
15	Teaching Address Over	↑	A position data address was too high.	137
		↓	OFF when the next teaching command is completed normally or teaching address setting is completed.	

Word n+14: Task 2 Status Flags

Bit(s)	Function
00 to 15	Task 2's status flags are the same as task 1's (word n+13).

Word n+15: Task 3 Status Flags

Bit(s)	Function
00 to 15	Task 3's status flags are the same as task 1's (word n+13).

Word n+16: Task 4 Status Flags

Bit(s)	Function
00 to 15	Task 4's status flags are the same as task 1's (word n+13).

Word n+17: X-axis Status Flags 1

Bit	Name	Signal	Meaning	Page
00	Reference Origin	1	Within the in-position range of the reference origin.	138
		0	Outside the in-position range of the reference origin.	
01	Workpiece Origin	1	Within the in-position range of the workpiece origin.	138
		0	Outside the in-position range of the workpiece origin.	
02	MPG Operation	1	MPG operation being executed.	139
03	Busy Flag	1	Manual command being processed.	139
04	Transmission #1 Completed	↑	Data #1 transmission has been completed.	140
		↓	Transmission #1 Control Bit is OFF.	

Bit	Name	Signal	Meaning	Page
05	Reception #1 Completed	↑	Data #1 reception has been completed.	141
		↓	Reception #1 Control Bit is OFF.	
06	Transfer #1 Error	↑	An error occurred during transmission #1 or reception #1.	141
		↓	OFF when the next transmission #1 or reception #1 is completed normally.	
07	Override Data Alarm	↑	The override value was incorrect in the override command.	142
		↓	The override value was correct in the override command.	
08	Zone 1 Flag	1	Within zone 1.	142
		0	Outside of zone 1.	
09	Zone 2 Flag	1	Within zone 2.	
		0	Outside of zone 2.	
10	Zone 3 Flag	1	Within zone 3.	
		0	Outside of zone 3.	
11	Zone 4 Flag	1	Within zone 4.	
		0	Outside of zone 4.	
12	Zone 5 Flag	1	Within zone 5.	
		0	Outside of zone 5.	
13	Zone 6 Flag	1	Within zone 6.	
		0	Outside of zone 6.	
14	Zone 7 Flag	1	Within zone 7.	
		0	Outside of zone 7.	
15	Zone 8 Flag	1	Within zone 8.	
		0	Outside of zone 8.	

Word n+18: X-axis Status Flags 2

The status of the CCW Limit Input, CW Limit Input, Emergency Stop Input, Alarm Input, Run Command Output, and Alarm Reset Output Flags can be displayed on-screen by selecting “Display MC I/O status” from the MC Monitoring menu. Refer to *13-5 Displaying MC I/O Status* in the *MC Support Software Operation Manual* for more details.

Bit(s)	Name	Signal	Meaning	Page
00	Servo Lock ON	1	Servo lock status	142
		0	Servo free status	
01	No Origin Flag	1	No origin (The reference origin isn’t fixed.)	143
		0	The reference origin is fixed.	
02	Axis Operating	1	There is an axis movement command.	144
		0	There isn’t an axis movement command.	
03	Positioning Completed	1	Within the in-position range	144
		0	Outside the in-position range	
04 to 06	Not used	---	---	---
07	Error Counter Alarm	1	The number of accumulated pulses in the error counter exceeds the error counter warning value.	145
08	CCW Limit Input	1	The CCW limit input is ON.	145
		0	The CCW limit input is OFF.	

Bit(s)	Name	Signal	Meaning	Page
09	CW Limit Input	1	The CW limit input is ON.	145
		0	The CW limit input is OFF.	
10	Origin Proximity Input	1	The origin proximity input is ON.	145
		0	The origin proximity input is OFF.	
11	Emergency Stop Input	1	The emergency stop input is ON.	146
		0	The emergency stop input is OFF.	
12	Alarm Input	1	The driver alarm input is ON.	146
		0	The driver alarm input is OFF.	
13	Run Command Output	1	The run command output is ON.	146
		0	The run command output is OFF.	
14	Alarm Reset Output	1	The driver alarm reset output is ON.	146
		0	The driver alarm reset output is OFF.	
15	Reserved	---	---	---

Word n+19: Y-axis Status Flags 1

Bit(s)	Name	Signal	Meaning
00 to 03	The Y-axis' status flags are the same as the X-axis' (word n+17).		
04	Transmission #2 Completed	↑	Transmission #2 has been completed.
		↓	Transmission #2 Control Bit is OFF.
05	Reception #2 Completed	↑	Reception #2 has been completed.
		↓	Reception #2 Control Bit is OFF.
06	Transfer #2 Error	↑	An error occurred during transmission #2 or reception #2.
		↓	OFF when the next transmission #2 or reception #2 is completed normally.
07 to 15	The Y-axis' status flags are the same as the X-axis' (word n+17).		

Word n+20: Y-axis Status Flags 2

Bit(s)	Name	Signal	Meaning
00 to 15	The Y-axis' status flags are the same as the X-axis' (word n+18).		

Word n+21: Z-axis Status Flags 1

Bit(s)	Name	Signal	Meaning
00 to 03	The Z-axis' status flags are the same as the X-axis' (word n+17).		
04	Transmission #3 Completed	↑	Transmission #3 has been completed.
		↓	Transmission #3 Control Bit is OFF.
05	Reception #3 Completed	↑	Reception #3 has been completed.
		↓	Reception #3 Control Bit is OFF.
06	Transfer #3 Error	↑	An error occurred during transmission #3 or reception #3.
		↓	OFF when the next transmission #3 or reception #3 is completed normally.
07 to 15	The Z-axis' status flags are the same as the X-axis' (word n+17).		

Word n+22: Z-axis Status Flags 2

Bit(s)	Name	Signal	Meaning
00 to 15	The Z-axis' status flags are the same as the X-axis' (word n+18).		

Word n+23: U-axis Status Flags 1

Bit(s)	Name	Signal	Meaning
00 to 03	The U-axis' status flags are the same as the X-axis' (word n+17).		
04	Transmission #4 Completed	↑	Transmission #4 has been completed.
		↓	Transmission #4 Control Bit is OFF.
05	Reception #4 Completed	↑	Reception #4 has been completed.
		↓	Reception #4 Control Bit is OFF.
06	Transfer #4 Error	↑	An error occurred during transmission #4 or reception #4.
		↓	OFF when the next transmission #4 or reception #4 is completed normally.
07 to 15	The U-axis' status flags are the same as the X-axis' (word n+17).		

Word n+24: U-axis Status Flags 2

Bit(s)	Name	Signal	Meaning
00 to 15	The U-axis' status flags are the same as the X-axis' (word n+18).		

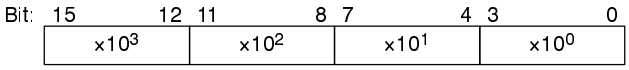
6-5 DM Word Usage

This section describes the functions performed by the words in the DM Area of the PC data area interface.

6-5-1 Output Words (PC → MC Unit)

Word m: Transfer Address #1

This word specifies the destination (source) address used when transmission (reception) is executed with the Transmission #1 Control Bit (Reception #1 Control Bit). The address is stored in BCD as shown in the following diagram.



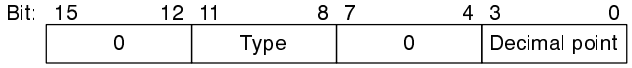
If the transmission/reception data is position data, specify a value from 0000 to 1999. These values correspond to position data addresses A0000 to A1999.

If the transmission/reception data is a parameter (acceleration/deceleration times), specify one of the following addresses:

4106: The X-axis' acceleration time
4206: The Y-axis' acceleration time
4306: The Z-axis' acceleration time
4406: The U-axis' acceleration time
4107: The X-axis' deceleration time
4207: The Y-axis' deceleration time
4307: The Z-axis' deceleration time
4407: The U-axis' deceleration time
4108: The X-axis' interpolation acceleration time
4208: The Y-axis' interpolation acceleration time
4308: The Z-axis' interpolation acceleration time
4408: The U-axis' interpolation acceleration time
4109: The X-axis' interpolation deceleration time
4209: The Y-axis' interpolation deceleration time
4309: The Z-axis' interpolation deceleration time
4409: The U-axis' interpolation deceleration time

Word m+1: Transfer #1 Auxiliary Information

This word indicates what type of data is being transferred and the position of the decimal point, as shown in the following diagram.



The type digit (0 or 1) indicates the type of data being transferred.

0: Position data
1: Parameter data (an acceleration or deceleration time)

The decimal point digit (0 to 4) indicates the position of the decimal point.

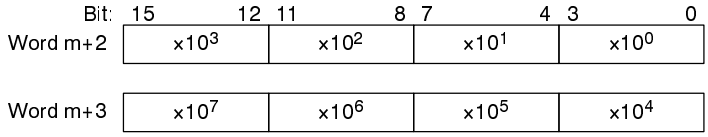
0: No decimal point
1: One digit after the decimal point
2: Two digits after the decimal point
3: Three digits after the decimal point
4: Four digits after the decimal point

Words m+2 and m+3: Transmission Data #1

These words contain the data that will be transmitted.

When transmitting acceleration or deceleration time data, set word m+3 to 0000 and set the time in word m+2. The time can be 0000 to 9999 ms.

Use the following format when transmitting position data. Bit 15 of word m+3 indicates the sign (0=positive, 1=negative).



Word m+4: Transfer Address #2

Transfer Address #2 has the same format as Transfer Address #1 (word m).

Word m+5: Transfer #2 Auxiliary Information

The auxiliary information for transfer #2 has the same format as auxiliary information for transfer #1 (word m+1).

Words m+6 and m+7: Transmission Data #2

Transmission Data #2 has the same format as Transmission Data #1 (words m+2 and m+3).

Word m+8: Transfer Address #3

Transfer Address #3 has the same format as Transfer Address #1 (word m).

Word m+9: Transfer #3 Auxiliary Information

The auxiliary information for transfer #3 has the same format as auxiliary information for transfer #1 (word m+1).

Words m+10 and m+11: Transmission Data #3

Transmission Data #3 has the same format as Transmission Data #1 (words m+2 and m+3).

Word m+12: Transfer Address #4

Transfer Address #4 has the same format as Transfer Address #1 (word m).

Word m+13: Transfer #4 Auxiliary Information

The auxiliary information for transfer #4 has the same format as auxiliary information for transfer #1 (word m+1).

Words m+14 and m+15: Transmission Data #4

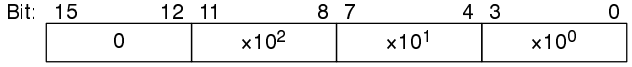
Transmission Data #4 has the same format as Transmission Data #1 (words m+2 and m+3).

Words m+16, 17, and 18: Task 1 Command Words

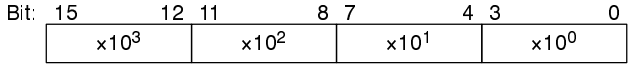
Words m+16, m+17, and m+18 contain the program number, teaching start address, and current position type for task 1.

Word m+16: Program Number

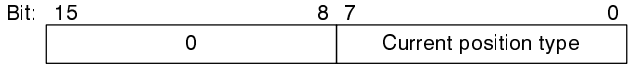
This word specifies the program number of the program that will be executed in automatic mode. If the Program Number Read Bit is ON when the Cycle Start Bit is turned ON, the program number stored here will be read and the corresponding program will be executed.



Word m+17: Teaching Start Address
The contents of this word will be used as the teaching address when the teaching address is set.



Word m+18: Current Position Type
This word indicates the type of current position data output to DM.



The current position type can have the following values:

0: Current position in the reference coordinate system
1: Current position in the workpiece coordinate system
2: Current position in the reference coordinate system (pulses)
3: Current position in the workpiece coordinate system (pulses)
4: Shift from the workpiece origin

Words m+19, 20, and 21: Task 2 Command Words

Words m+19, m+20, and m+21 have the same format and function as words m+16, m+17, and m+18 (task 1 command words).

Words m+22, 23, and 24: Task 3 Command Words

Words m+22, m+23, and m+24 have the same format and function as words m+16, m+17, and m+18 (task 1 command words).

Words m+25, 26, and 27: Task 4 Command Words

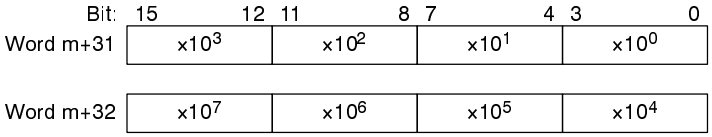
Words m+25, m+26, and m+27 have the same format and function as words m+16, m+17, and m+18 (task 1 command words).

Words m+31, 32, and 33: X-axis Command Words

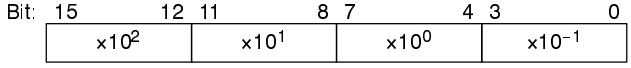
Words m+31 and m+32 contain the X-axis' current position preset value. Word m+33 contains the X-axis' override data.

Words m+31 and m+32: X-axis' Current Position Preset Value
When the current position preset command is executed in automatic mode, the value stored in these words will be set as the present value.

The value is in BCD and the position of the decimal point is determined by the Minimum Setting Unit parameter (a machine parameter). Bit 15 of word m+32 indicates the sign (0=positive, 1=negative).



Word m+33: X-axis' Override Data
This word contains the override data used in axis operations. The value is BCD and the minimum unit is 0.1%.



Words m+34, 35, and 36: Y-axis Command Words

Words m+34, m+35, and m+36 have the same format and function as words m+31, m+32, and m+33 (the X-axis' command words).

Words m+37, 38, and 39: Z-axis Command Words

Words m+37, m+38, and m+39 have the same format and function as words m+31, m+32, and m+33 (the X-axis' command words).

Words m+40, 41, and 42: U-axis Command Words

Words m+40, m+41, and m+42 have the same format and function as words m+31, m+32, and m+33 (the X-axis' command words).

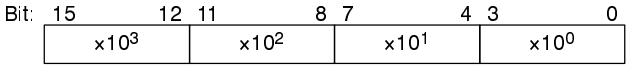
Word m+43: Not used

Word m+43 isn't used.

6-5-2 Input Words (MC Unit → PC)

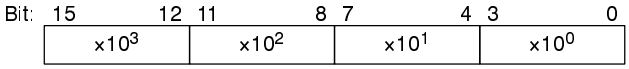
Word m+44: System Version Number

The system version number is output in 4-digit BCD.



Word m+45: System Error Code

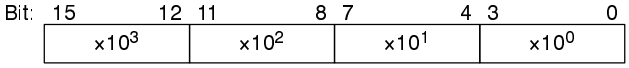
The system error code is output in 4-digit BCD. The error code will be set in this word when the System Error Flag is turned ON.



Word m+46: Task 1 Error Code

The task 1 error code is output in 4-digit BCD. The error code will be set in this word when the Task Error Flag is turned ON.

When an axis error occurs and a task error doesn't occur in the task associated with that axis, the Task Error Flag will be turned ON but this flag will contain 0000 (indicating normal operation).



Word m+47: Task 2 Error Code

The task 2 error code in word m+47 has the same format and function as the task 1 error code in word m+46.

Word m+48: Task 3 Error Code

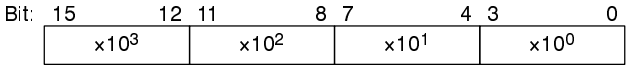
The task 3 error code in word m+48 has the same format and function as the task 1 error code in word m+46.

Word m+49: Task 4 Error Code

The task 4 error code in word m+49 has the same format and function as the task 1 error code in word m+46.

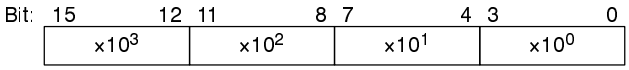
Word m+50: X-axis Error Code

The X-axis error code is output in 4-digit BCD. The error code will be set in this word when an error occurs in the X axis. (0000 indicates normal operation.)



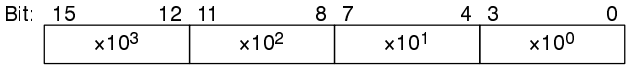
Word m+51: Y-axis Error Code

The Y-axis error code is output in 4-digit BCD. The error code will be set in this word when an error occurs in the X axis. (0000 indicates normal operation.)



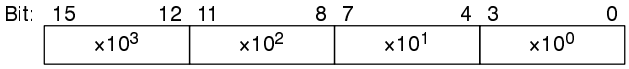
Word m+52: Z-axis Error Code

The Z-axis error code is output in 4-digit BCD. The error code will be set in this word when an error occurs in the X axis. (0000 indicates normal operation.)



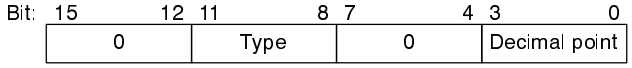
Word m+53: U-axis Error Code

The U-axis error code is output in 4-digit BCD. The error code will be set in this word when an error occurs in the X axis. (0000 indicates normal operation.)



Word m+54: Reception #1 Auxiliary Information

This word indicates what type of data is being received and the position of the decimal point, as shown in the following diagram.



The type digit (0 or 1) indicates the type of data being received.

0: Position data
1: Parameter data (an acceleration or deceleration time)

The decimal point digit (0 to 4) indicates the position of the decimal point.

0: No decimal point
1: One digit after the decimal point
2: Two digits after the decimal point
3: Three digits after the decimal point
4: Four digits after the decimal point

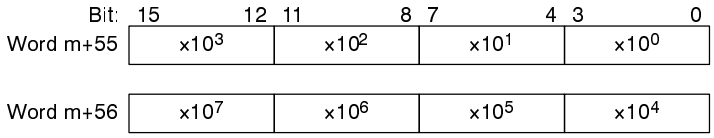
Word m+54 will contain "0100" if the received data is an acceleration or deceleration time.

Words m+55 and m+56: Reception Data #1

The data received in response to the Reception #1 Control Bit is set in these words. The Reception #1 Completed Flag (bit 5 of word n+17) will be turned ON when the data has been set in these words.

When the received data is acceleration or deceleration time data, word m+55 will be set to 0000 and the time (0000 to 9999 ms) will be set in word m+55.

The following format is used for position data. Bit 15 of word m+56 indicates the sign (0=positive, 1=negative).



Word m+57: Reception #2 Auxiliary Information

The auxiliary information for Reception Data #2 has the same format as auxiliary information for Reception Data #1 (word m+54).

Words m+58 and m+59: Reception Data #2

Reception Data #2 has the same format as Reception Data #1 (words m+55 and m+56).

Word m+60: Reception #3 Auxiliary Information

The auxiliary information for Reception Data #3 has the same format as auxiliary information for Reception Data #1 (word m+54).

Words m+61 and m+62: Reception Data #3

Reception Data #3 has the same format as Reception Data #1 (words m+55 and m+56).

Word m+63: Reception #4 Auxiliary Information

The auxiliary information for Reception Data #4 has the same format as auxiliary information for Reception Data #4 (word m+54).

Words m+64 and m+65: Reception Data #4

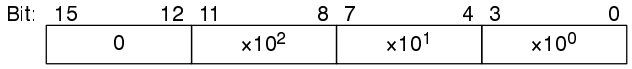
Reception Data #4 has the same format as Reception Data #1 (words m+55 and m+56).

Words m+66 to m+70: Task 1 Status Words

Words m+66 through m+70 contain the current program number, current block number, M code, and teaching address for task 1.

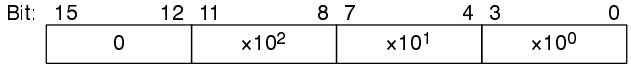
Word m+66: Current Program Number

When an MC program is being executed, the program number of the program being executed is output to this word in BCD. If a program isn't being executed, this word will contain the program number of the last program that was executed. This word is reset to 0000 when the power is turned ON or when there has been a change from the MC Support Software.



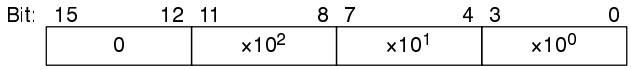
Word m+67: Current Block Number

When an MC program is being executed, the block number being executed is output to this word in BCD. If a program isn't being executed, this word will contain the block number of the last block that was executed. (This word is reset to 0000 when the power is turned ON or when there has been a change from the MC Support Software.)



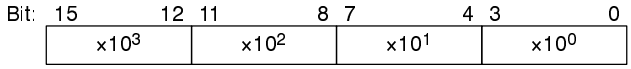
Word m+68: M code

The M code is output to this word in BCD. The M code is valid when the M strobe has been turned ON.



Word m+69: Teaching Address

When teaching is performed, the address containing the position data (the teaching address) is output to this word.



Word m+70: Not Used

Words m+71 to m+75: Task 2 Status Words

The task 2 status words have the same format and function as the task 1 status words (m+66 through m+70).

Words m+76 to m+80: Task 3 Status Words

The task 3 status words have the same format and function as the task 1 status words (m+66 through m+70).

Words m+81 to m+85: Task 4 Status Words

The task 4 status words have the same format and function as the task 1 status words (m+66 through m+70).

Word m+86

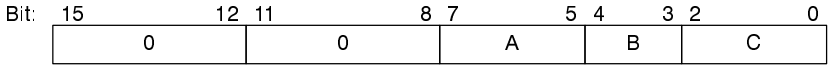
Word m+86 isn't used.

Words m+87 to m+89: X-axis Status Words

Words m+88 and m+89 contain the X-axis' current position and m+87 contains the auxiliary information for the current position data.

Word m+86: X-axis Current Position (Auxiliary Information)

The contents of this word indicate the decimal point position, units, and current position type, as shown below:



A: The value of bits 5, 6, and 7 specifies the decimal point position.

- 0: No decimal point
- 1: One digit after the decimal point
- 2: Two digits after the decimal point
- 3: Three digits after the decimal point
- 4: Four digits after the decimal point

B: The value of bits 3 and 4 specifies the units.

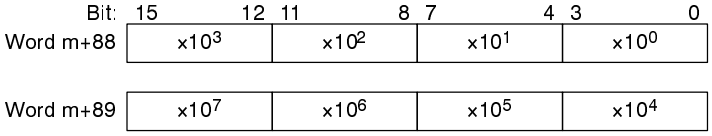
- 0: mm
- 1: Inches
- 2: Degrees
- 3: Pulses

C: The value of bits 0, 1, and 2 specifies the current position type.

- 0: Current position in the reference coordinate system
- 1: Current position in the workpiece coordinate system
- 2: Current position in the reference coordinate system (pulses)
- 3: Current position in the workpiece coordinate system (pulses)
- 4: Shift from the workpiece origin

Words m+88 and m+89: X-axis Current Position

The X-axis' current position data is output to words m+88 and m+89, as shown in the following diagram. Bit 15 of word m+89 indicates the sign (0=positive, 1=negative).



Words m+90 to m+92: Y-axis Status Words

The Y-axis status words have the same format and function as the X-axis status words (m+87 through m+89).

Words m+93 to m+95: Z-axis Status Words

The Z-axis status words have the same format and function as the X-axis status words (m+87 through m+89).

Words m+96 to m+98: U-axis Status Words

The U-axis status words have the same format and function as the X-axis status words (m+87 through m+89).

Word m+99

Word m+99 isn't used.

6-6 Interface Bit Specifics

This section provides details on the functions of the interface bits in the CIO Area. The timing chart and ladder program examples use the bits and words for task 1 and the X-axis.

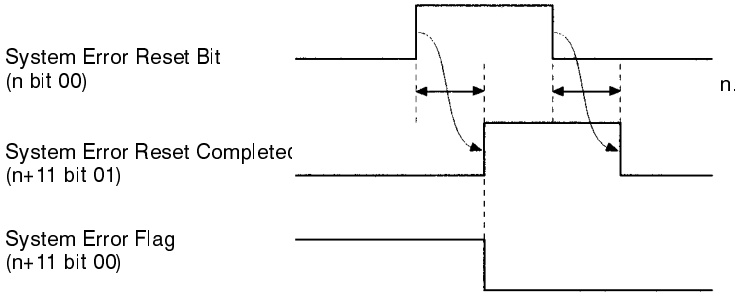
6-6-1 Word n Bit 00: System Error Reset Bit

System errors can be reset by turning this bit ON. Keep this bit ON until the System Error Reset Completed Flag is turned ON.

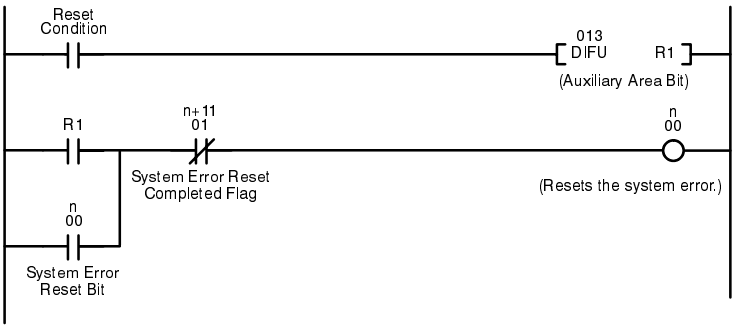
Signal

System errors are reset when this bit goes ON (up-differentiation).

Timing Chart



Program Example



6-6-2 Word n Bit 01: Transmission #1 Control Bit

This bit is used to initiate transmission of Transmission Data #1 to the MC Unit. Turn this bit ON after setting the Transfer Address #1, Transfer #1 Auxiliary Information, and Transmission Data #1 (DM words m through m+3).

The user can determine whether the transmission has been completed by checking the status of the Transmission #1 Completed Flag. Keep the Transmission #1 Control Bit ON until the Transmission Data #1 Completed Flag is turned ON.

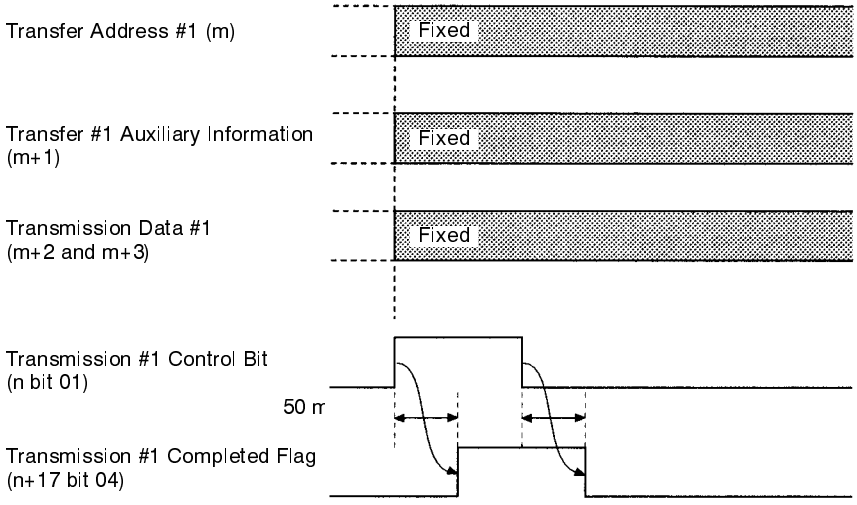
Data transmission and data reception can't be performed at the same time. Data transmission has higher priority.

The Transmission #2, #3, and #4 Control Bits (word n bits 03, 05, and 07) operate in the same way.

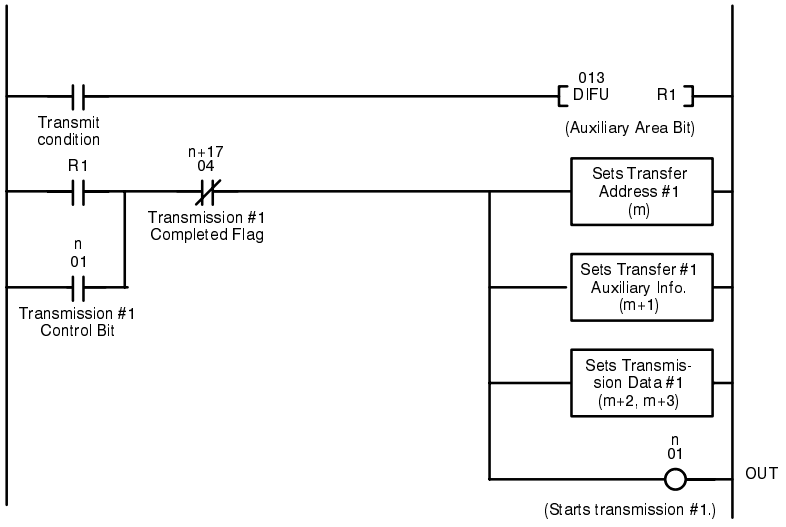
Signal

Transmission starts when this bit goes ON (up-differentiation).

Timing Chart



Program Example



6-6-3 Word n Bit 02: Reception #1 Control Bit

This bit is used to initiate reception of Reception Data #1 from the MC Unit. Turn this bit ON after setting Transfer Address #1 and the Reception #1 Auxiliary Information (DM words m and m+54).

The user can determine whether the reception has been completed by checking the status of the Reception #1 Completed Flag. Keep the Reception #1 Control Bit ON until the Reception Data #1 Completed Flag is turned ON. When the reception is completed normally, the received data will be stored in words m+54 through m+56.

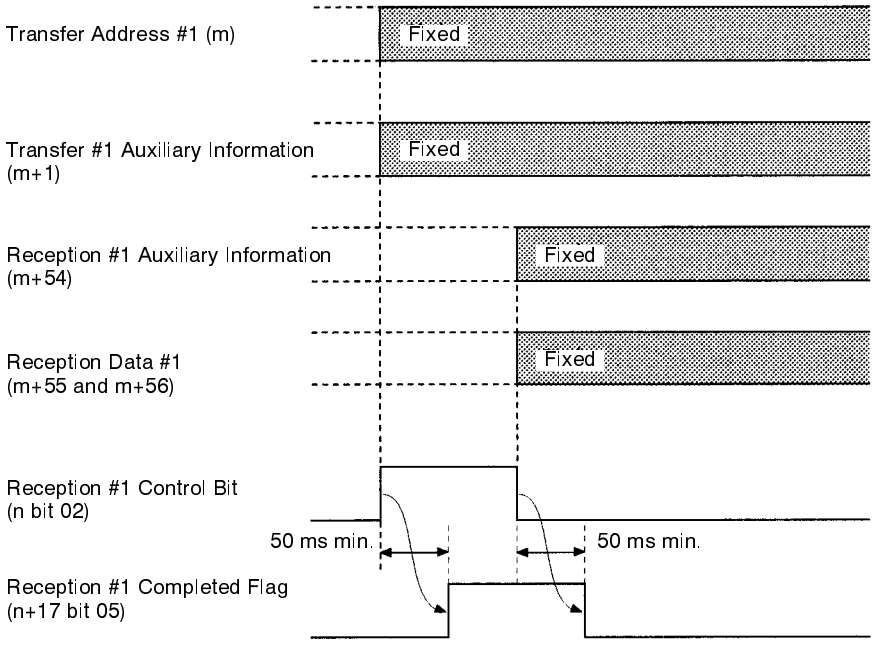
Data transmission and data reception can't be performed at the same time. Data transmission has higher priority.

The Reception #2, #3, and #4 Control Bits (word n bits 04, 06, and 08) operate in the same way.

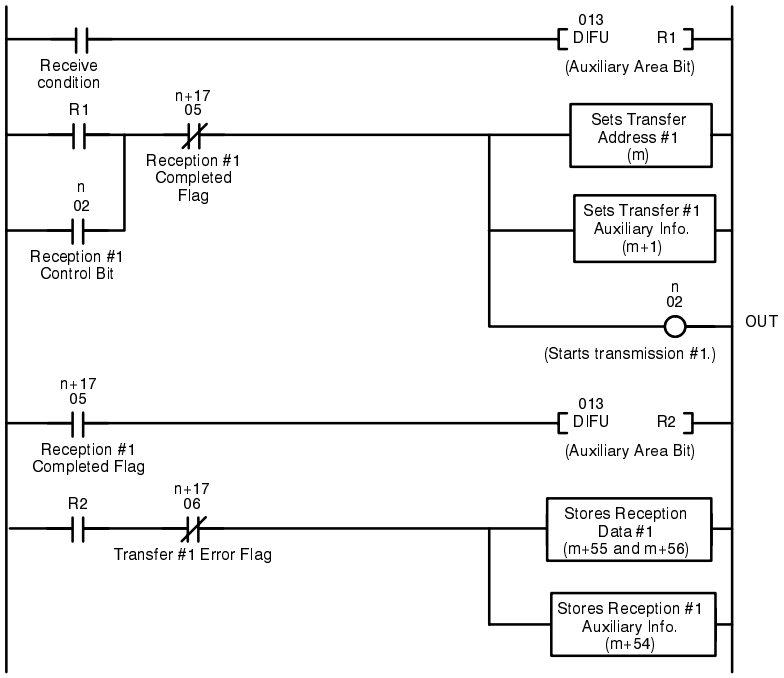
Signal

Reception starts when this bit goes ON (up-differentiation).

Timing Chart



Program Example

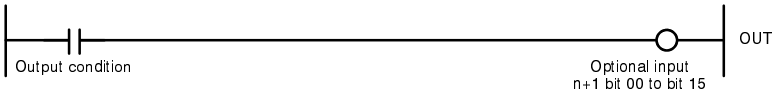


6-6-4 Word n+1 Bits 00 to 15: Optional Inputs

These bits can be used as conditions in MC programs when executing the OPTIONAL END (G74), OPTIONAL SKIP (G75), or OPTIONAL PROGRAM STOP (G76) functions or using optional numbers in functions G00, G01, G02, or G03. There are 16 optional inputs available (bits 00 to 15).

Signal Optional inputs have a value of 1 when the input is ON, 0 when the input is OFF.

Program Example



6-6-5 Word n+2 Bit 00: Task Error Reset Bit (Task 1)

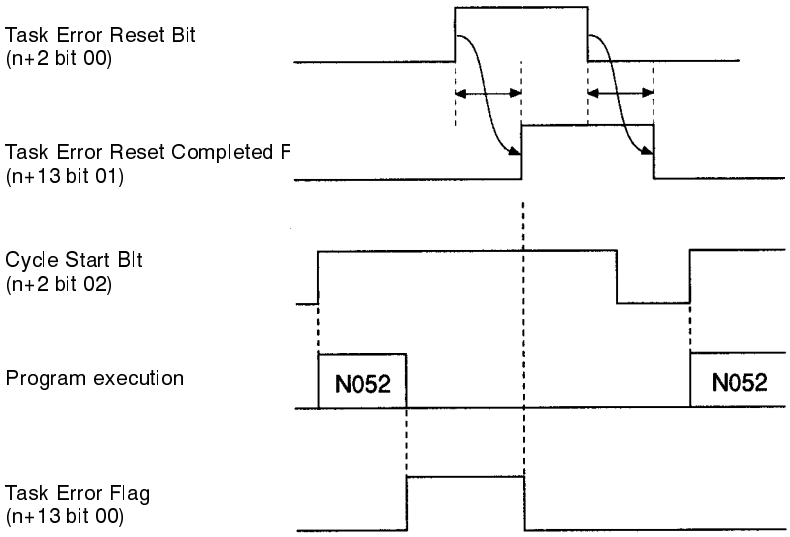
This bit is used to clear errors that occur in each task and axis. Normally, this bit is turned ON after eliminating the cause of the error. If the reset bit is turned ON without eliminating the cause of the error, the error will just occur again the next time the program is executed.

Program execution can be continued after the error is reset. Since the current program number and block number are retained, the program will be executed from the block where the error occurred when operation is restarted with the Cycle Start Bit. Keep this bit ON until the Task Error Reset Completed Flag is turned ON.

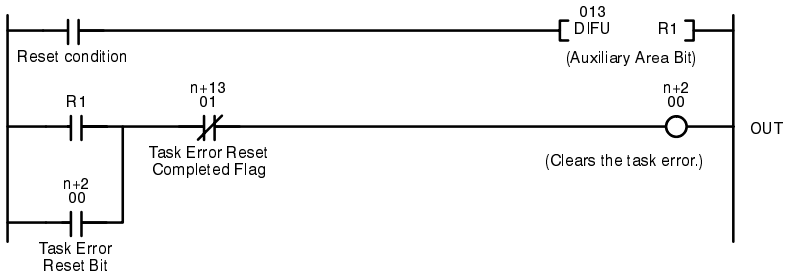
The task error bits for tasks 2, 3, and 4 (bit 00 in words n+3, n+4, and n+5) operate in the same way.

Signal The error is cleared when this bit goes ON (up-differentiation).

Timing Chart



Program Example



6-6-6 Word n+2 Bit 01: Automatic/Manual Mode Bit (Task 1)

This bit determines the mode for task 1. Turn this bit ON to specify automatic mode, OFF to specify manual mode. In manual mode, commands such as manual origin search, manual origin return, and jogging are possible. In automatic mode, the MC program can be executed.

When the mode is switched from manual to automatic, any operating axes will decelerate to a stop. When the mode is switched from automatic to manual, the program will be stopped if it is operating. The program will be restarted from the block where it was stopped if the mode is switched back to automatic and the program is restarted.

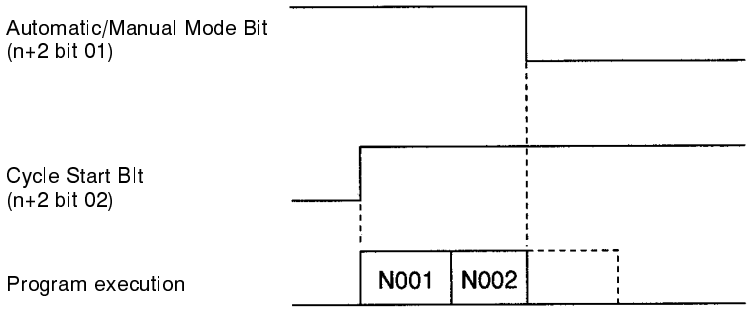
The automatic/manual mode bits for tasks 2, 3, and 4 (bit 01 in words n+3, n+4, and n+5) operate in the same way.

Signal

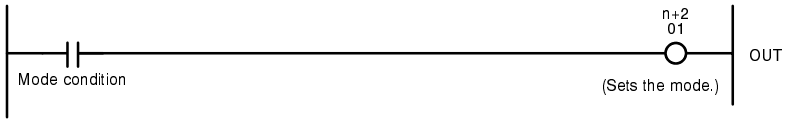
The automatic/manual mode bit settings have the following functions.

Signal	Function
↑ (OFF-to-ON)	If any axes are operating in manual mode, they will be decelerated to a stop.
ON	Automatic mode
↓ (ON-to-OFF)	If the MC program is being executed, it will be stopped and any operating axes will be decelerated to a stop.
OFF	Manual mode

Timing Chart



Program Example



6-6-7 Word n+2 Bit 02: Cycle Start Bit (Task 1)

This bit starts MC program execution if the mode is set to automatic mode. Turn this bit ON to start the program from the beginning or restart a program that has been interrupted from the block where it was interrupted.

If an error has been reset, the program will continue when the Cycle Start Bit is turned ON.

The Cycle Start Bit is referenced at the start of each block and execution continues if it is ON. If the Cycle Start Bit is OFF, the program will be stopped. In Pass operation, the next block will be executed even if the Cycle Start Bit is OFF but the program will be stopped after the next block is completed.

The Cycle Start Bits for tasks 2, 3, and 4 (bit 02 in words n+3, n+4, and n+5) operate in the same way.

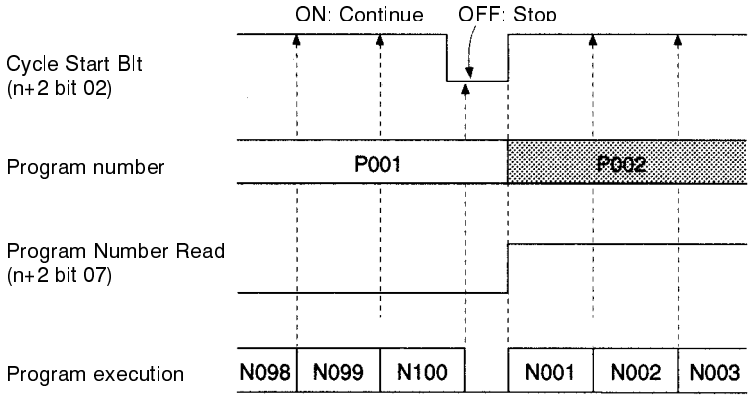
Signal

The Cycle Start Bit settings have the following functions.

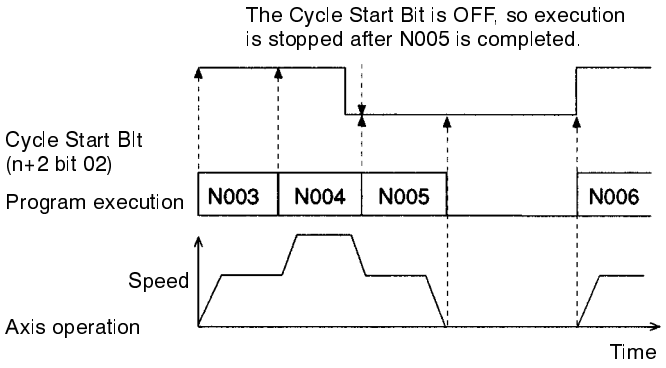
Signal	Function
↑ (OFF-to-ON)	Starts MC program execution.
ON	Continues execution. (Referenced at the start of each block.)
OFF	Stops execution. (Referenced at the start of each block.)

Timing Chart

Normal Operation:

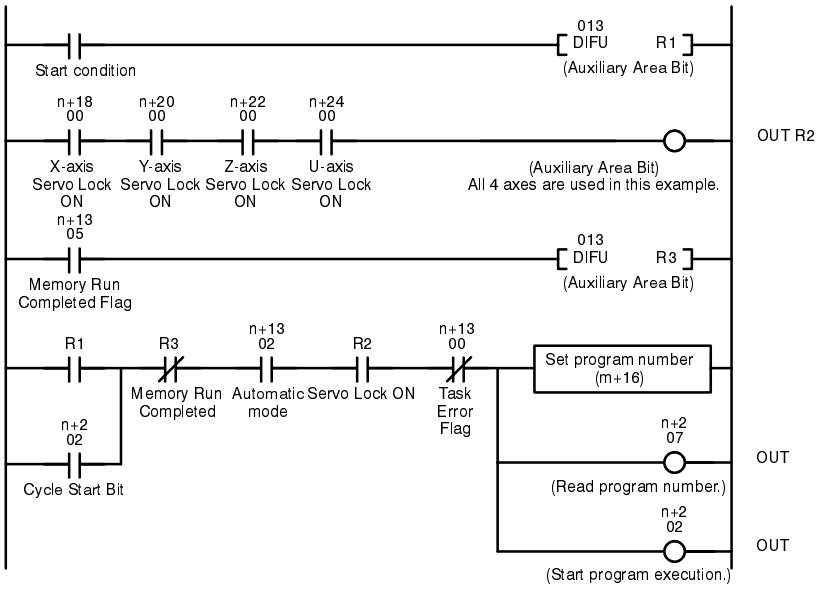


Pass Operation:




```
N003  G01   X100  F100
N004  G01   X300  F200
N005  G01   X400  F100
```

Program Example



Note When executing a positioning instruction from the MC program, all of the axes being used must be servo-locked (Servo Lock ON Flags ON), so be sure to add a condition like R2 in the program example above.

6-6-8 Word n+2 Bit 07: Program Number Read Bit (Task 1)

This bit specifies the number of the MC program that will be executed; it is valid only when the mode is set to automatic mode. The status of this bit is referenced when the Cycle Start Bit goes ON. If this bit is ON when the Cycle Start Bit goes ON, the program number set in DM word m+16 will be read and the specified program will be executed from the beginning.

When any MC programs in the same task have been edited, added, or deleted, always make sure that this bit is ON the next time that the Cycle Start Bit is turned ON. An error will occur if the Program Number Read Bit is OFF.

Signal

The Program Number Read Bit settings have the following functions.

Signal	Function
ON	Indicates to read the program number from DM word m+16.

Timing Chart

Refer to the timing chart in *6-6-7 Cycle Start Bit (Task 1)*.

Program Example

Refer to the program example in *6-6-7 Cycle Start Bit (Task 1)*.

6-6-9 Word n+2 Bit 03: Single Block Bit (Task 1)

This bit is used to execute a single block of the program; it is valid only when the mode is set to automatic mode. The Single Block Bit and Cycle Start Bit are used together.

- Single block execution can be performed by turning ON the Single Block Bit and then turning the Cycle Start BIt ON and OFF.
- When the program is being executed the status of the Cycle Start Bit is checked at the start of each block; if it is ON, the next block will be executed and program execution will be paused.

To switch from single block execution to normal program execution, turn OFF the Single Block Bit and then turn ON the Cycle Start Bit. The program will be executed continuously starting from the next block.

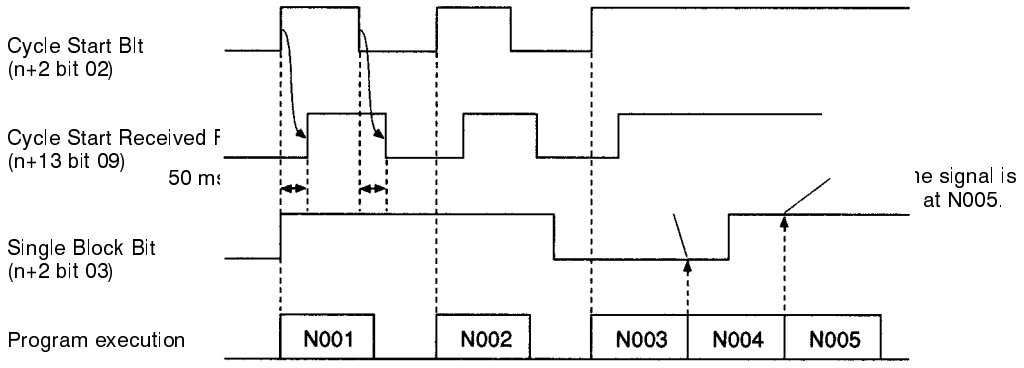
When the Single Block Bit is turned ON during Pass operation, the next block's positioning operation will be completed before program execution is paused.

The Single Block Bits for tasks 2, 3, and 4 (bit 03 in words n+3, n+4, and n+5) operate in the same way.

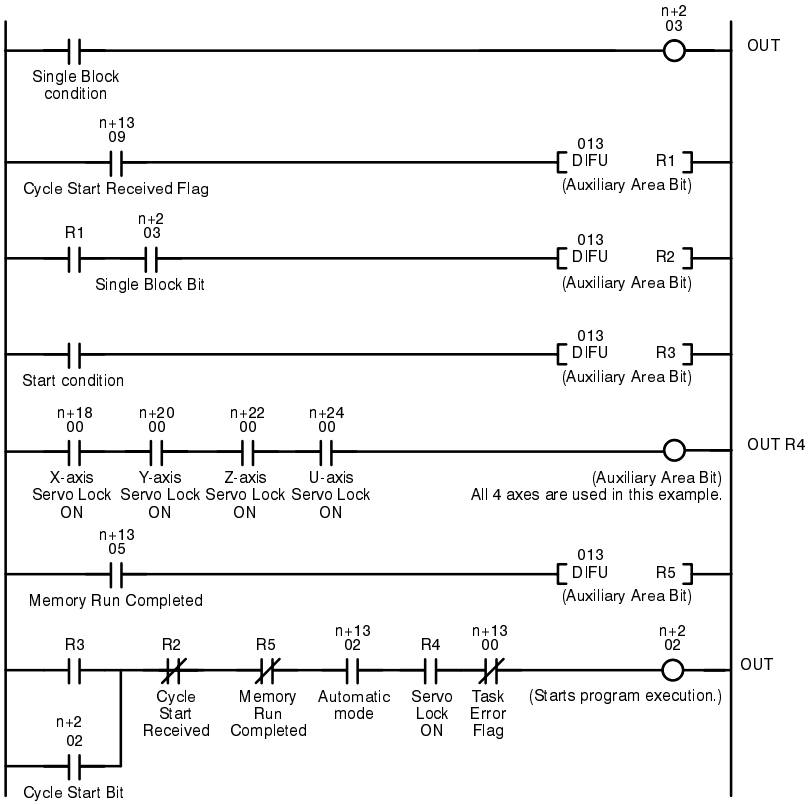
Signal

The status of the Single Block BIt is checked when the Cycle Start Bit is turned ON and at the beginning of each block. When the bit is ON, program execution is paused after the next block is executed.

Timing Chart



Program Example



Note When executing a positioning instruction from the MC program, all of the axes being used must be servo-locked (Servo Lock ON Flags ON), so be sure to add a condition like R4 in the program example above.

6-6-10 Word n+2 Bit 04: Forced Block End Bit (Task 1)

This bit is used to force termination of the block being executed; it is valid only when the mode is set to automatic mode. When a block is terminated, the remainder of the block is canceled. Program execution will restart from the next block when restarted with the Cycle Start BIt.

Any axes that are operating will be decelerated to a stop. If the block being executed is waiting for an M code reset, the M code will be cleared to 0, the M strobe will be turned OFF, and then program execution will be stopped.

Operation of the Cycle Start Bit is disabled as long as the Forced Block End Bit is ON. (The program won't be executed when the Cycle Start Bit is turned ON if the Forced Block End Bit is ON.)

The Forced Block End Bit has a higher priority than Pause Bit (n+2 bit 05). If both of these bits go ON at the same time, the forced block end command will be executed. Keep the Forced Block End Bit ON until the Memory Run Flag goes OFF.

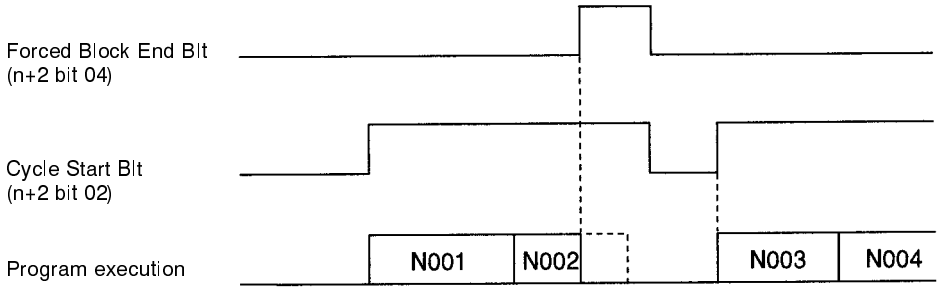
The Forced Block End Bits for tasks 2, 3, and 4 (bit 04 in words n+3, n+4, and n+5) operate in the same way.

Signal

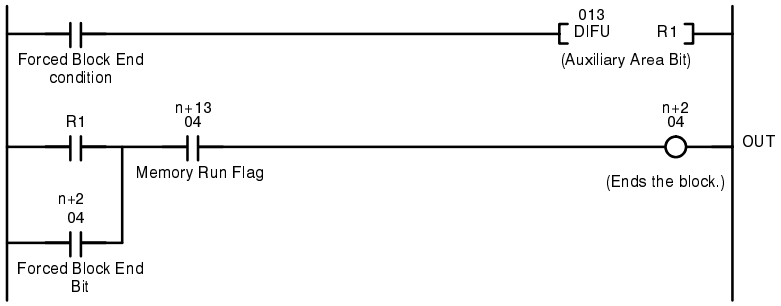
The Forced Block End Bit settings have the following functions.

Signal	Function
↑ (OFF-to-ON)	Terminates the block being executed.
ON	As long as the bit remains ON, the program won't be executed when the Cycle Start Bit goes ON.

Timing Chart



Program Example



6-6-11 Word n+2 Bit 05: Pause Bit (Task 1)

This bit is used to temporarily pause program execution; it is valid only when the mode is set to automatic mode.

When this bit is turned ON, the block being executed will be interrupted and any axes that are operating will be decelerated to a stop. Keep the Pause Bit ON until the Memory Run Flag goes OFF. Program execution can be restarted from the interrupted block by turning ON the Cycle Start BIt.

Operation of the Cycle Start Bit is disabled as long as the Pause Bit is ON. (The program won't be executed when the Cycle Start Bit is turned ON if the Pause Bit is ON.)

The Pause Bit has a lower priority than the Forced Block End Bit (n+2 bit 04). If both of these bits go ON at the same time, the forced block end command will be executed.

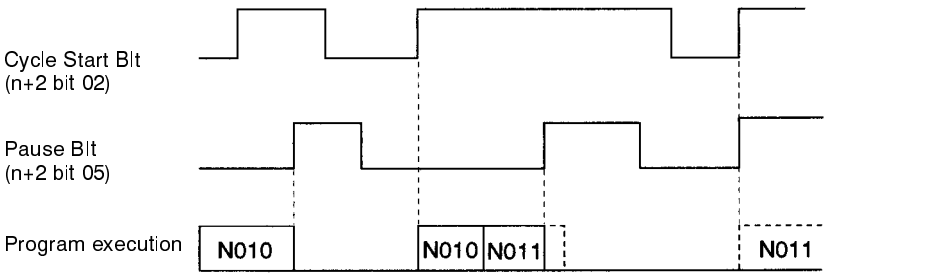
The Pause Bits for tasks 2, 3, and 4 (bit 05 in words n+3, n+4, and n+5) operate in the same way.

Signal

The Pause Bit settings have the following functions.

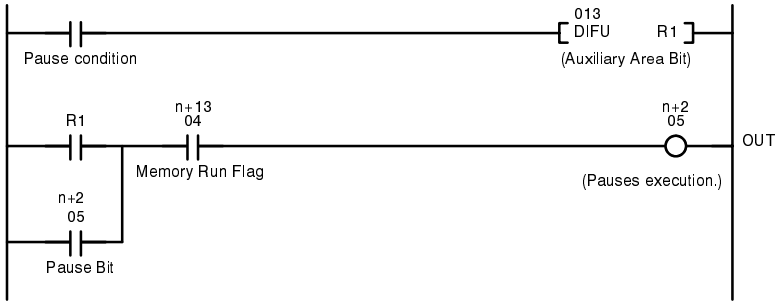
Signal	Function
↑ (OFF-to-ON)	Pauses execution of the block being executed.
ON	As long as the bit remains ON, the program won't be executed when the Cycle Start Bit goes ON.

Timing Chart



The Cycle Start Bit has no effect because the Pause Bit is ON.

Program Example



6-6-12 Word n+2 Bit 06: M Code Reset Bit (Task 1)

This bit resets the M code output.

If M code 0 to 495 is output in the program, that block will standby for an M code reset. The progress through the blocks will be interrupted. In this case, the M Code Reset Bit can be turned ON to clear the M code to 0 and turn OFF the M strobe. Program execution will restart when this bit is turned OFF. (Keep this bit ON until the M strobe goes OFF.)

If the M Code Reset BIt is ON when a block is executed, that block's M code won't be output. When M codes 500 to 999 are being output and this bit is turned ON, the M code will be reset whether the mode is automatic or manual.

M codes 496 to 499 can't be reset by the M Code Reset Bit; reset these M codes with general inputs.

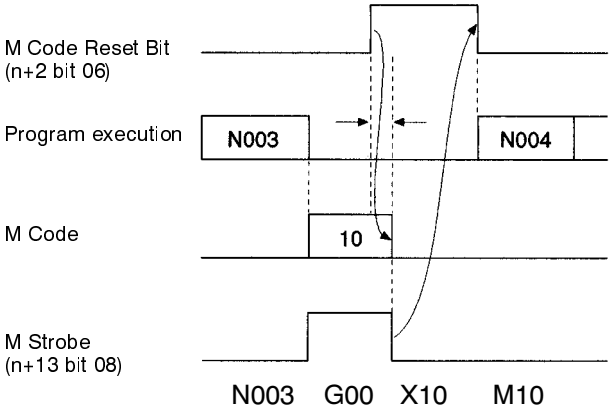
The M Code Reset Bits for tasks 2, 3, and 4 (bit 06 in words n+3, n+4, and n+5) operate in the same way.

Signal

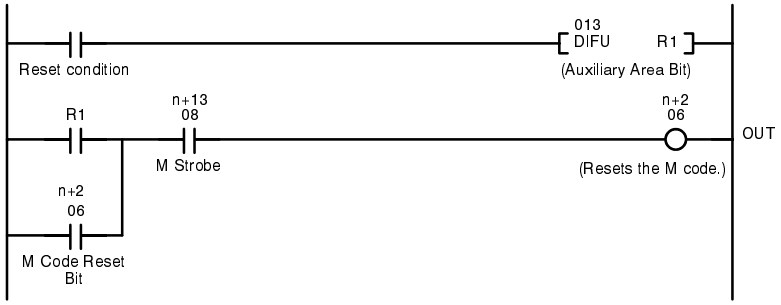
The M Code Reset Bit settings have the following functions.

Signal	Function
↑ (OFF-to-ON)	If there is an M code reset standby (M000 to M495), the M code will be reset. M codes M500 to M999 will be reset.
ON	M code reset standby (when there is an M code reset standby) Prevents M code output.
↓ (ON-to-OFF)	Clears M code reset standby.

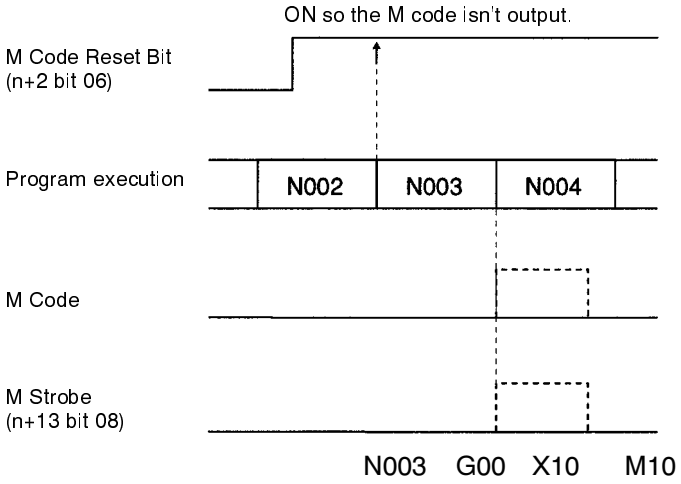
Timing Chart 1



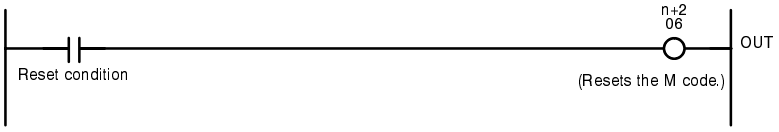
Program Example 1



Timing Chart 2



Program Example 2



6-6-13 Word n+2 Bit 12: Teaching Address Set Bit (Task 1)

This bit specifies the leading address of the position data addresses containing the current position for teaching. The Teaching Address Set Bits for tasks 2, 3, and 4 (bit 12 in words n+3, n+4, and n+5) operate in the same way.

When this bit is turned ON, the teaching start address in DM word m+17 is set as the teaching address. Immediately after power is turned ON the teaching address is 0.

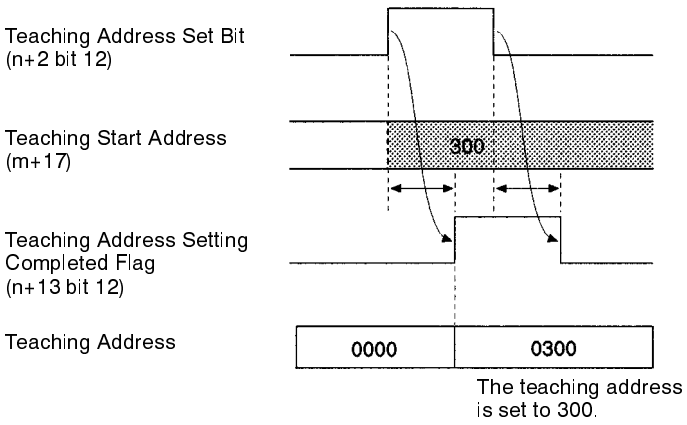
This bit is effective when it goes from OFF to ON (up-differentiation), but keep the bit ON until the Teaching Address Setting Completed Flag goes ON.

Signal

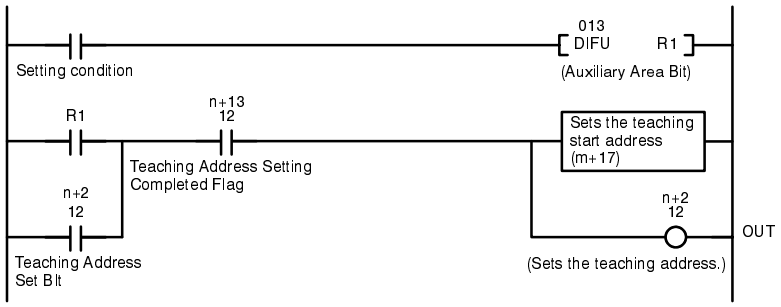
The Teaching Address Set Bit settings have the following functions.

Signal	Function
↑ (OFF-to-ON)	The teaching start address in DM word m+17 is transferred to the MC Unit and set as the teaching address.

Timing Chart



Program Example



6-6-14 Word n+2 Bit 13: Teaching Bit (Task 1)

This bit is used to read the reference coordinate system position into the position data address. The Teaching Bits for tasks 2, 3, and 4 (bit 13 in words n+3, n+4, and n+5) operate in the same way.

When this bit is turned ON, the current position is read in order as position data then the axes that are being managed in the task and the teaching addresses that are output to DM are incremented. Keep this bit ON until the Teaching Completed Flag goes ON.

Teaching can be performed in automatic or manual mode, even if a task error has occurred.

Signal

The current value is read to DM when the Teaching Bit goes from OFF to ON (up-differentiation).

6-6-15 Word n+2 Bit 14: Teaching Type Bit (Task 1)

This bit determines which position is used when teaching is performed. There are two positions that can be specified:

- Target Position:
  The current position of the command
- Current Position (Feedback Value):
  The actual current position produced from the encoder

Determine which position is appropriate for your application. The status of this bit is referenced when the Teaching Bit goes from OFF to ON.

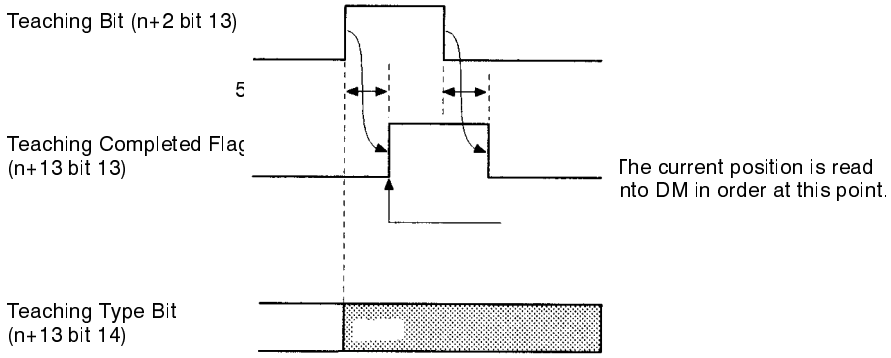
The Teaching Type Bits for tasks 2, 3, and 4 (bit 14 in words n+3, n+4, and n+5) operate in the same way.

Signal

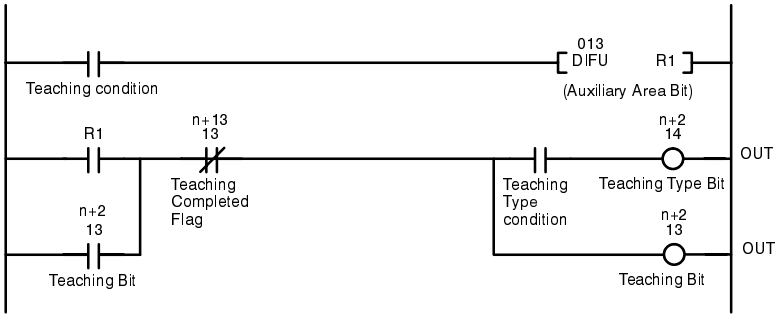
The Teaching Type Bit settings have the following functions.

Signal	Function
1	Specifies the current position (feedback value).
0	Specifies the target position.

Timing Chart

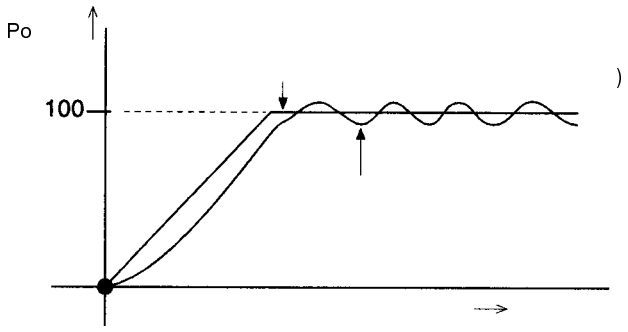


Program Example



Example Operation

In this example, the X-axis is moved to X100, so the X target position (the command's current position) is 100. Actually, a servo is used and a varying deviation from 100 develops. The actual current position from the encoder reflects this deviation.



6-6-16 Word n+7 Bit 00: Deceleration Stop Bit (X-axis)

This bit is used to stop the X-axis' operation when one of the following manual commands is being executed: Origin Search, Origin Return, Jogging, or MPG Operation. The deceleration stop signal can be received even if the Busy Flag is ON.

Keep this bit ON until both the axis operation and MPG operation have gone OFF. The next manual command won't be received until the first manual command goes OFF, even after deceleration stop is completed with this bit.

The Deceleration Stop Bit doesn't have any effect if it is turned ON while the axis is stopped, but other manual commands can't be received as long as the bit is ON. This disabling function can be used when interlocking and is valid with manual commands that don't involve axis operations.

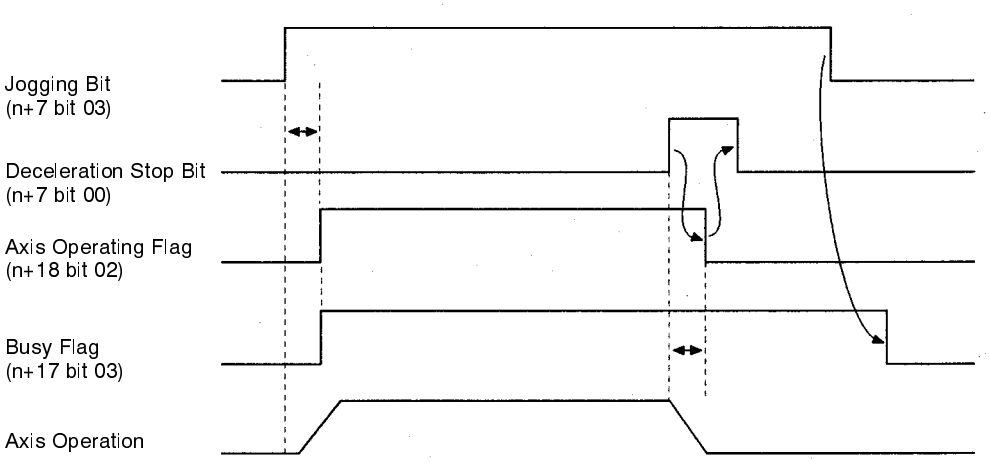
The Deceleration Stop Bits for the Y, Z, and U axes (bit 00 in words n+8, n+9, and n+10) operate in the same way.

Signal

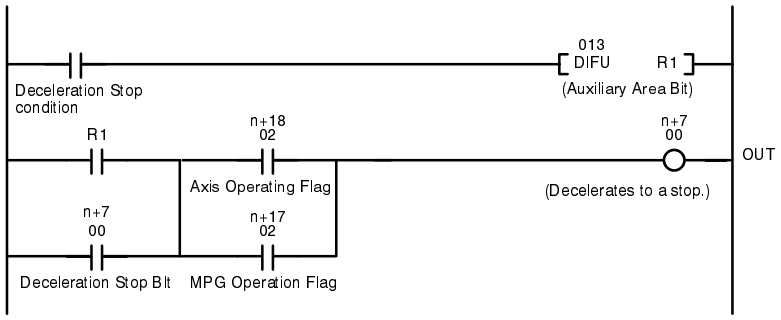
The Deceleration Stop Bit settings have the following functions.

Signal	Function
↑ (OFF-to-ON)	Starts decelerating manual operations to a stop.
1	Prevents reception of other manual commands.

Timing Chart



Program Example



6-6-17 Word n+7 Bit 01: Origin Search Bit (X-axis)

This bit is used to perform a manual origin search. The origin search begins when the Origin Search Bit goes from OFF to ON (up-differentiation). Overrides are ignored in manual origin searches.

If a reference origin offset is specified in the system parameters, the axis will be moved that distance from the mechanical origin by PTP control after the mechanical origin is detected. At this point, overrides are valid using the maximum high-speed feed rate, acceleration time, and deceleration time. The position after the movement becomes the origin of the reference coordinate system.

This bit is ignored when the X-axis' Busy Flag (n+17 bit 03) is ON. The origin search operation will be stopped if the Origin Search Bit is turned OFF while the search is in progress. Keep this bit ON until the X-axis' No Origin Flag (n+18 bit 01) is turned OFF.

When the "Origin Search Method" machine parameter has been set to "Set origin at power ON" with the MC Support Software, that position will be the origin for manual origin searches.

An Origin Return Operation will be performed for axes set for absolute operation.

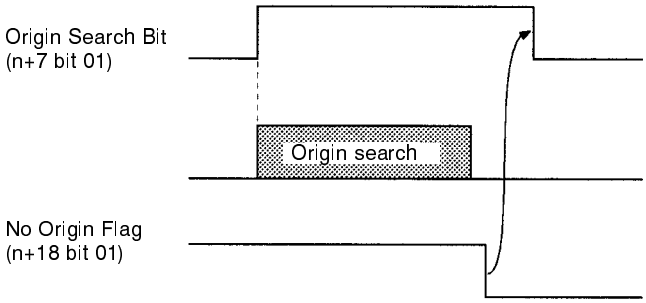
The Origin Search Bits for the Y, Z, and U axes (bit 01 in words n+8, n+9, and n+10) operate in the same way.

Signal

The Origin Search Bit settings have the following functions.

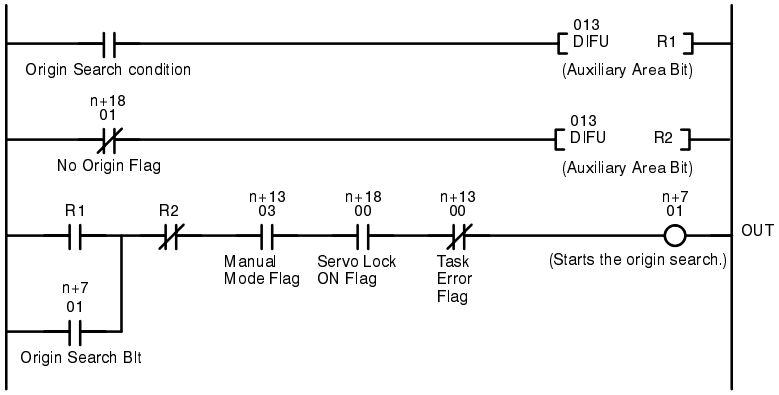
Signal	Function
↑ (OFF-to-ON)	Starts the origin search.
↓ (ON-to-OFF)	Stops the origin search if it is in progress.

Timing Chart

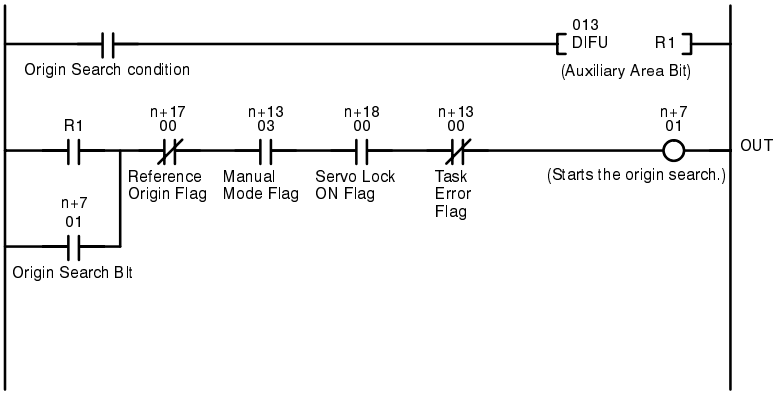


Program Examples

The following example is for incremental operation.



The following example is for absolute operation.



6-6-18 Word n+7 Bit 02: Reference Origin Return Bit (X-axis)

This bit is used to move the X-axis to the origin of the reference coordinate system. The positioning begins when the Reference Origin Return Bit goes from OFF to ON (up-differentiation). The maximum high-speed feed rate, acceleration time, and deceleration time are used during the operation.

An override (0.1% to 100.0%) can be applied with the manual origin return. An error will occur if this bit is turned ON but the mechanical origin hasn't been fixed.

This bit is ignored if the Busy Flag is ON. Keep this bit ON until the Reference Origin Flag is turned ON.

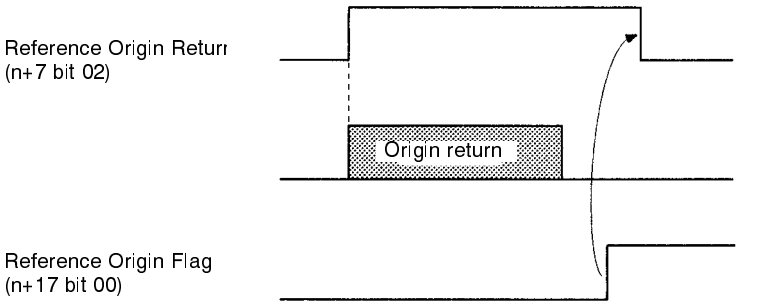
The Reference Origin Return Bits for the Y, Z, and U axes (bit 02 in words n+8, n+9, and n+10) operate in the same way.

Signal

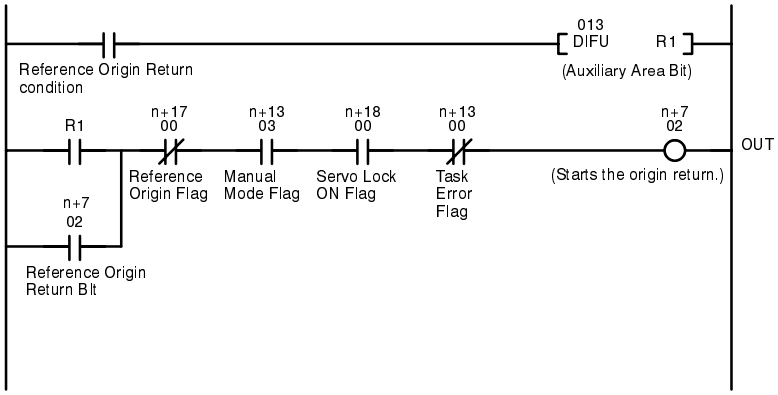
The Reference Origin Return Bit settings have the following functions.

Signal	Function
↑ (OFF-to-ON)	Starts the manual origin return.
↓ (ON-to-OFF)	Stops the manual origin return if it is in progress.

Timing Chart



Program Example



6-6-19 Word n+7 Bit 03: Jogging Bit (X-axis)

This bit is used to start the jog operation. The axis is moved in the specified jog direction when this bit is turned ON and decelerates to a stop when this bit is turned OFF.

Jogging is performed with PTP control using the maximum high-speed feed rate, acceleration time, and deceleration time. An override (0.1% to 100.0%) can be applied with jogging. Operation depends on whether the mechanical origin has been fixed or not.

The Jogging Bits for the Y, Z, and U axes (bit 03 in words n+8, n+9, and n+10) operate in the same way.

- When the mechanical origin has been fixed:
  a) Jogging can be performed within the software limits.
  b) The axis is decelerated to a stop at the software limit.
- When the mechanical origin hasn't been fixed, jogging can be performed without any restrictions.

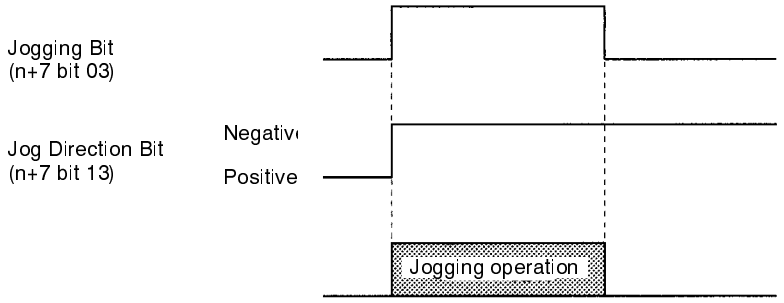
This bit is ignored if the Busy Flag is ON.

Signal

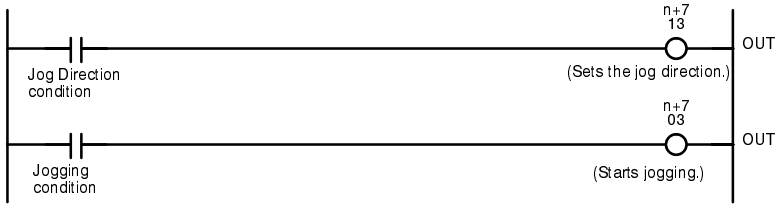
The Jogging Bit settings have the following functions.

Signal	Function
↑ (OFF-to-ON)	Starts jogging.
↓ (ON-to-OFF)	Stops the jogging operation.

Timing Chart



Program Example



6-6-20 Word n+7 Bit 13: Jog Direction Bit (X-axis)

This bit determines which direction the axis will be moved in jog operation; it is used together with the Jogging Bit. The status of the Jog Direction Bit is read when the Jogging Bit is turned ON and has the following meaning.

ON: Negative direction (The direction that reduces the present value.)
OFF: Positive direction (The direction that increases the present value.)

The Jog Direction Bits for the Y, Z, and U axes (bit 13 in words n+8, n+9, and n+10) operate in the same way.

Signal

The Jog Direction Bit settings have the following functions.

Signal	Function
1	Sets the jog direction to negative.
0	Sets the jog direction to positive.

6-6-21 Word n+7 Bit 05: Current Position Preset Bit (X-axis)

This bit is used to set the current position to a new preset value. The preset value must be set in advance in DM words m+31 and m+32 (the X-axis' current position preset value).

The reference origin is automatically fixed after presetting the current position, so it isn't necessary to perform an origin search. Keep this bit ON until the Busy Flag goes ON. This bit is ignored if the Busy Flag is ON.

The current position preset operation can be performed in the servo-lock or servo-free status. This operation won't be interrupted while in progress even if the deceleration stop operation is executed.

The software limits are checked when this command is executed, and the current position preset won't be performed if there is an error.

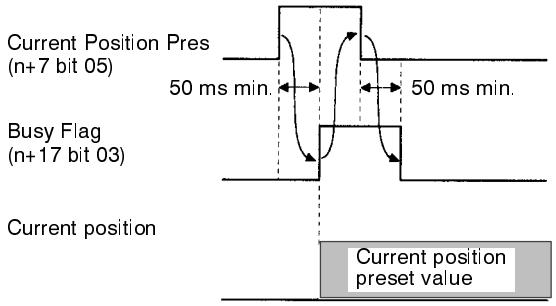
This operation has no effect on axes set for absolute operation.

The Current Position Preset Bits for the Y, Z, and U axes (bit 05 in words n+8, n+9, and n+10) operate in the same way.

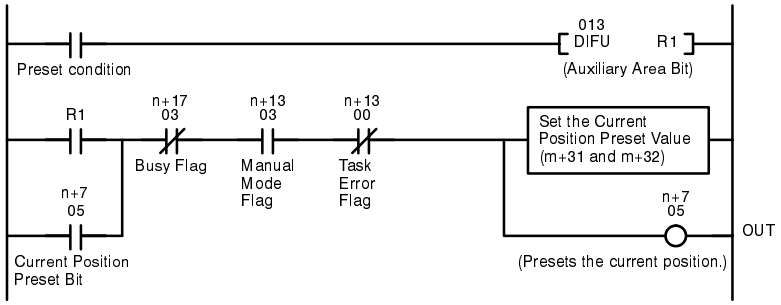
Signal

The current position preset operation is started when the Current Position Preset Bit goes from OFF to ON (up-differentiation).

Timing Chart



Program Example



6-6-22 Word n+7 Bit 08: Enable MPG Bit (X-axis)

This bit is used to start MPG operation; it is valid only when the mode is set to manual mode. As long as this bit is ON, the command will reference the MPG Ratio Specifiers (n+7 bits 14 and 15), read the input pulses from the MPG, multiply them by the MPG ratio, and perform MPG operation.

MPG operation will be stopped when this bit goes OFF. This bit is ignored if the Busy Flag is ON.

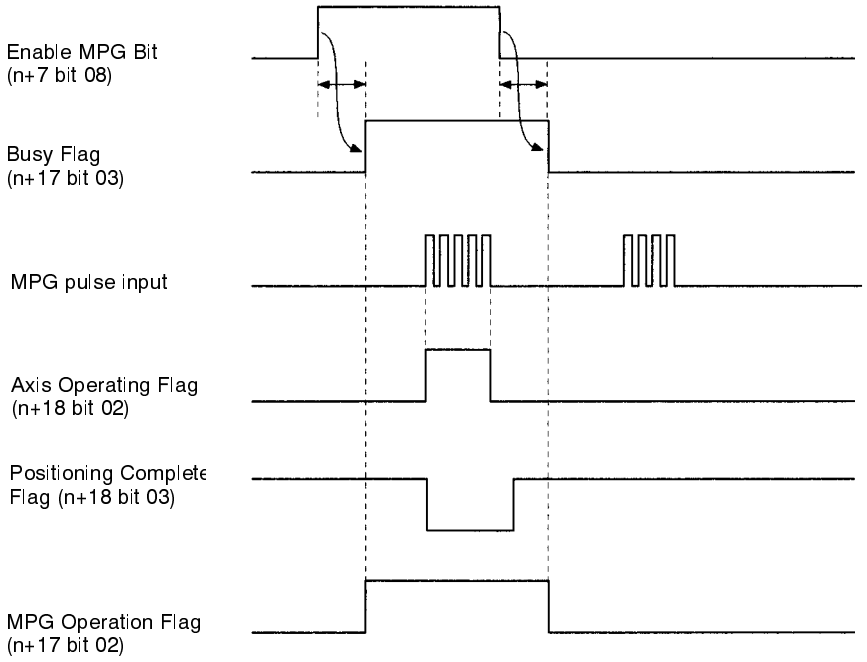
The Enable MPG Bits for the Y, Z, and U axes (bit 08 in words n+8, n+9, and n+10) operate in the same way.

Signal

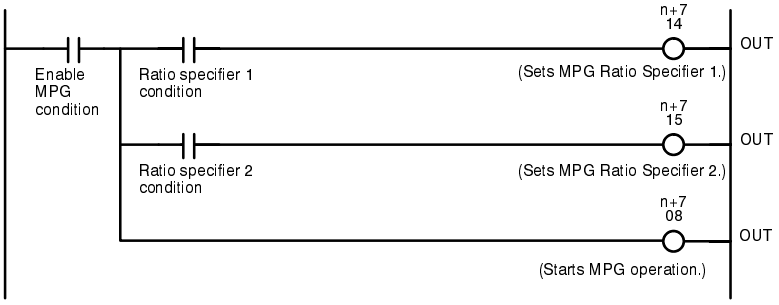
The Enable MPG Bit settings have the following functions.

Signal	Function
↑ (OFF-to-ON)	Starts MPG operation.
↓ (ON-to-OFF)	Stops MPG operation.

Timing Chart



Program Example



6-6-23 Word n+7 Bits 14 and 15: MPG Ratio Specifiers (X-axis)

These bits are used to specify which MPG ratio used in MPG operation; they are valid only when the mode is set to manual mode. The status of these bits is checked continuously during MPG operation.

Bit 14	Bit 15	MPG Ratio Selected
OFF	OFF	MPG Ratio (1) in the Feedrate Parameters.
ON	OFF	MPG Ratio (2) in the Feedrate Parameters.
OFF	ON	MPG Ratio (3) in the Feedrate Parameters.
ON	ON	MPG Ratio (4) in the Feedrate Parameters.

The MPG Ratio Specifiers for the Y, Z, and U axes (bits 14 and 15 in words n+8, n+9, and n+10) operate in the same way.

Note Refer to *8-3-5 MPG Ratios* in the *MC Support Software Operation Manual* for details on the MPG ratio.

6-6-24 Word n+7 Bit 09: Servo Lock Bit (X-axis)

This bit is used to apply the servo lock; it is valid only when the mode is set to manual mode. The processes listed below are performed when this bit goes ON.

1, 2, 3...

1. The present value is read from axes set for absolute operation.
2. A wiring check is performed the first time the servo lock command is executed after power is turned ON. If the wiring check is normal, wiring checks won't be performed when subsequent servo lock commands are executed.
3. A position loop is formed.
4. The Run Command Output to the servodriver is turned ON. This forms a speed loop.

This command won't be interrupted while in progress even if the deceleration stop command is executed.

Keep this bit ON until the Busy Flag goes ON. This bit is ignored if the Busy Flag is ON.

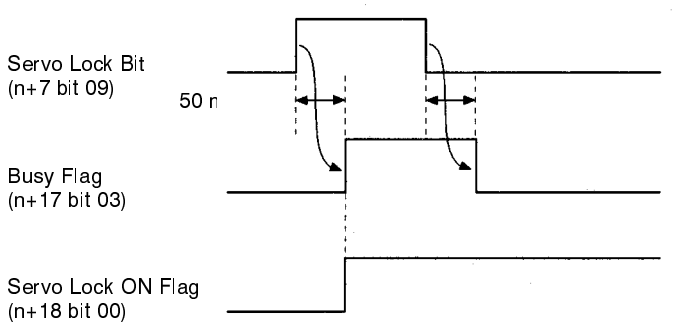
The Servo Lock Bits for the Y, Z, and U axes (bit 09 in words n+8, n+9, and n+10) operate in the same way.

Signal

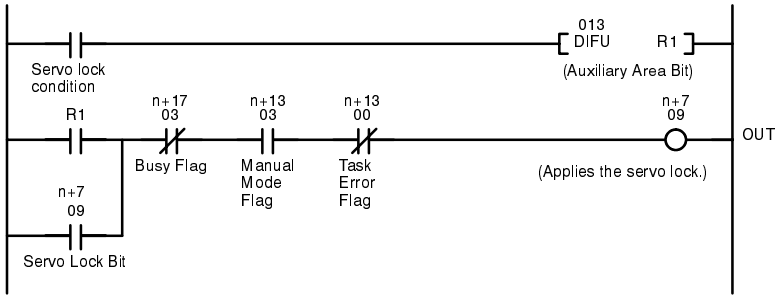
The Servo Lock BIt settings have the following functions.

Signal	Function
↑ (OFF-to-ON)	Starts the servo lock.

Timing Chart



Program Example



6-6-25 Word n+7 Bit 10: Servo Free Bit (X-axis)

When this bit is turned ON, the Run Command Output to the servodriver is cleared and the position loop is cleared. This command won't be interrupted while in progress even if the deceleration stop command is executed.

Keep this bit ON until the Busy Flag goes ON. This bit is ignored if the Busy Flag is ON.

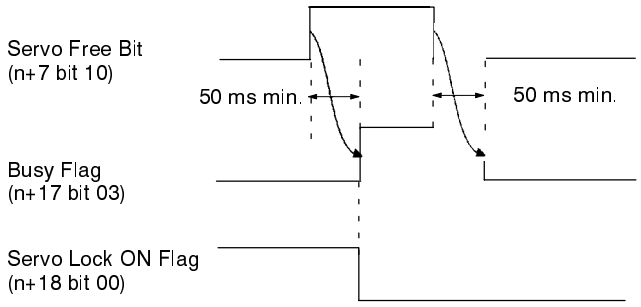
The Servo Free Bits for the Y, Z, and U axes (bit 10 in words n+8, n+9, and n+10) operate in the same way.

Signal

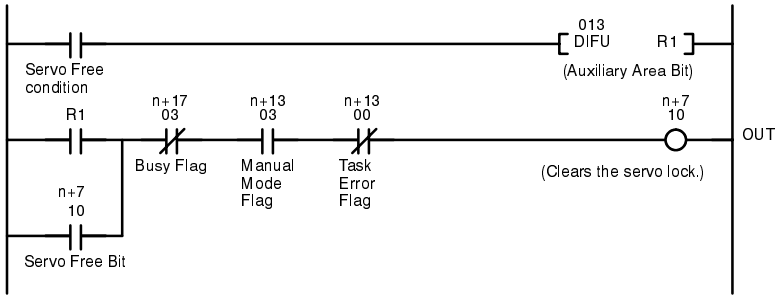
The Servo Free Bit settings have the following functions.

Signal	Function
↑ (OFF-to-ON)	Clears the servo lock status.

Timing Chart



Program Example



6-6-26 Word n+7 Bit 11: Driver Alarm Reset Bit (X-axis)

This bit is used to control the Driver Alarm Reset Output, an output to the servo-driver; it is usually OFF. When this bit is turned ON, the Driver Alarm Reset Output goes ON as well. Keep this bit ON until the Driver Alarm Input goes OFF.

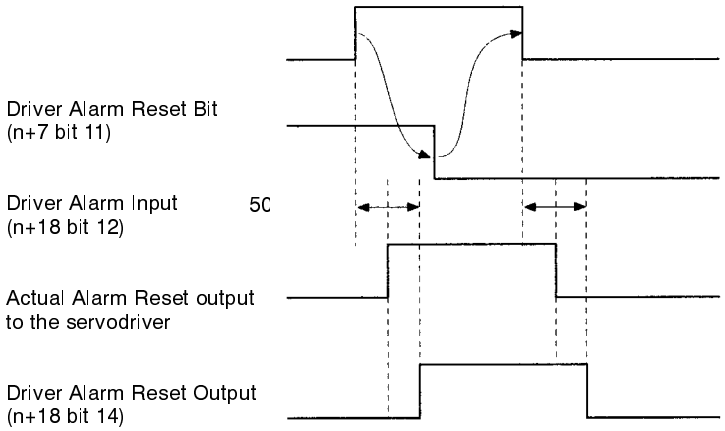
The Driver Alarm Reset Bits for the Y, Z, and U axes (bit 11 in words n+8, n+9, and n+10) operate in the same way.

Signal

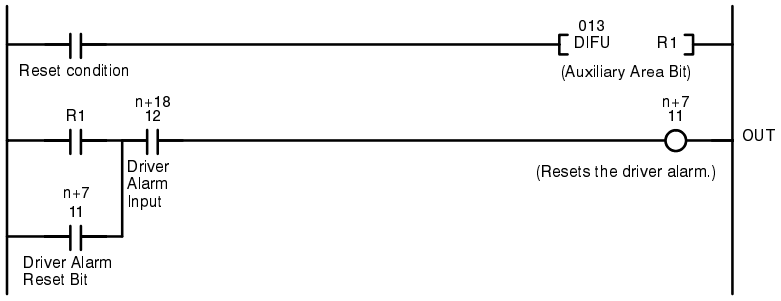
The Driver Alarm Reset Bit settings have the following functions.

Signal	Function
1 (ON)	Turns ON the Driver Alarm Reset Output.
0 (OFF)	Turns OFF the Driver Alarm Reset Output.

Timing Chart



Program Example



6-6-27 Word n+7 Bit 12: Override Setting Bit (X-axis)

This bit is used to change the feed rate for the X-axis. When this bit is turned ON, the Override Data in DM word m+33 becomes valid and is read. The override can be applied in the following operations.

Mode	Operation	Setting Range
Automatic	G00 (POSITIONING)	0.1% to 100.0%
	G01 (LINEAR INTERPOLATION)	0.1% to 199.9%
	G02, G03 (CIRCULAR INTERPOLATION)	0.1% to 199.9%
	G26 (REFERENCE ORIGIN RETURN)	0.1% to 100.0%
	G27 (WORKPIECE ORIGIN RETURN)	0.1% to 100.0%
	G28 (ORIGIN SEARCH)	0.1% to 100.0%
Manual	Jogging	0.1% to 100.0%
	Manual origin return	0.1% to 100.0%

The override setting will be set to 100.0% automatically if the setting is between 100.0% and 199.9% but the permitted setting range is 0.1% to 100.0%. If the override setting is outside of the maximum range (0.1% to 199.9%), the Override Data Alarm BIt (n+17 bit 07) will be turned ON and the override setting data will be invalid.

When the power is turned ON and the override setting isn't made with this bit, the override will be set to 100.0% automatically.

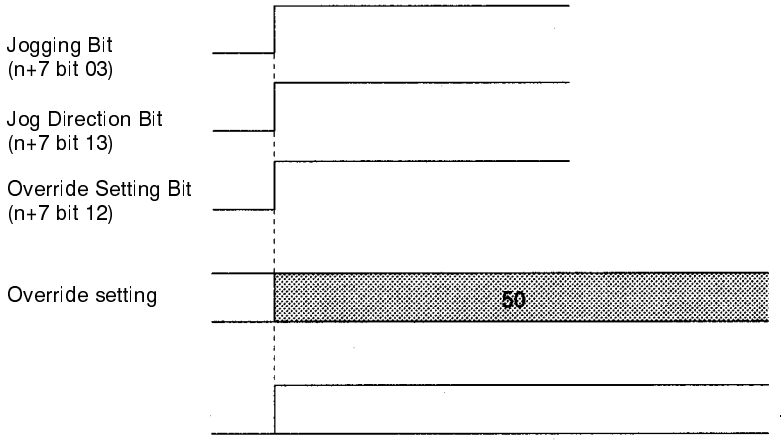
The Override Setting Bits for the Y, Z, and U axes (bit 12 in words n+8, n+9, and n+10) operate in the same way.

Signal

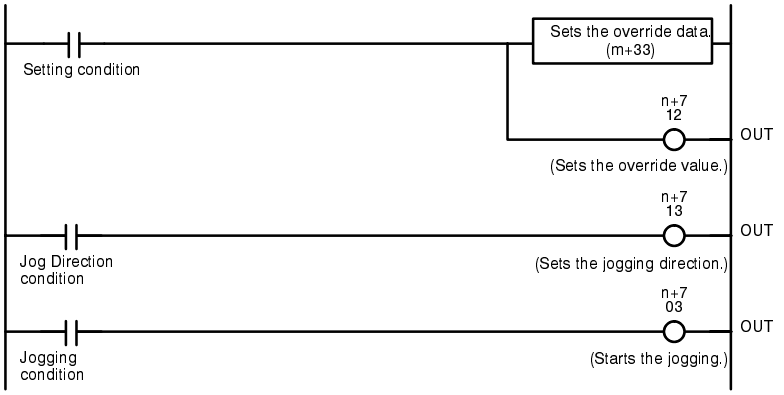
The Driver Alarm Reset Bit settings have the following functions.

Signal	Function
1 (ON)	The override data is valid.
0 (OFF)	The override data is invalid.

Timing Chart



Program Example



6-6-28 Word n+11 Bit 00: System Error Flag

This flag is turned ON when a system error has occurred. The flag can be reset with the System Error Reset Bit (n bit 00).

Signal

The System Error Flag settings have the following meanings.

Signal	Meaning
↑ (OFF-to-ON)	Goes ON when a system error has occurred.
↓ (ON-to-OFF)	Goes OFF when reset with the System Error Reset Bit.

6-6-29 Word n+11 Bit 01: System Error Reset Completed Flag

This flag is turned ON when system error reset processing has been completed. Use this bit to determine when the System Error Reset BIt can be turned OFF. The System Error Reset BIt must be kept ON until this flag is turned ON.

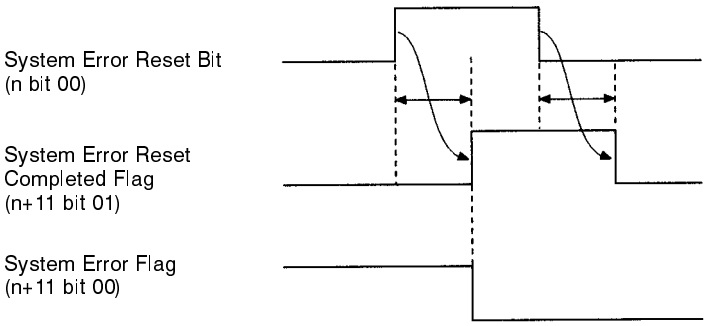
This flag won't be turned OFF until the System Error Reset Bit is turned OFF, even if error resetting is completed.

Signal

The System Error Reset Completed Flag settings have the following meanings.

Signal	Meaning
↑ (OFF-to-ON)	Goes ON when system error reset processing is completed.
↓ (ON-to-OFF)	Goes OFF when the System Error Reset Bit goes OFF.

Timing Chart



6-6-30 Word n+11 Bit 07: Battery Alarm Flag

This flag is turned ON when the voltage of the memory backup battery drops below the rated value. The battery must be replaced when this flag is turned ON. Refer to *11-1 Battery Replacement* for details.

It is recommended to use this flag as a condition for an output to an error indicator.

Signal

The Battery Alarm Flag settings have the following meanings.

Signal	Meaning
1	ON when battery voltage is below the rated value.
0	OFF when the battery voltage is normal.

6-6-31 Word n+11 Bit 13: Teaching Box Enabled Flag

This flag indicates whether the Teaching Box is in enabled mode. Use this flag as a condition to interlock the ladder program in this mode.

When this flag is ON, the only commands that can be received from the PC are Deceleration Stop, Forced Block End, Pause, Optional Input, and M Code Reset.

Refer to the Teaching Box Operation Manual for more details on the Teaching Box's enabled mode.

Signal

The Teaching Box Enabled Flag settings have the following meanings.

Signal	Meaning
1	ON when the Teaching Box is in enabled mode.
0	OFF when the Teaching Box isn't in enabled mode.

6-6-32 Word n+11 Bit 14: Teaching Box Protected Flag

This flag indicates whether the Teaching Box is in protect mode. Use this flag as a condition to interlock the ladder program in this mode.

No operations can be performed from the PC when this flag is ON.

Refer to the *Teaching Box Operation Manual* for more details on the Teaching Box's protect mode.

Signal

The Teaching Box Protected Flag settings have the following meanings.

Signal	Meaning
1	ON when the Teaching Box is in protect mode.
0	OFF when the Teaching Box isn't in protect mode.

6-6-33 Word n+11 Bit 15: Teaching Box Connected Flag

This flag indicates whether the Teaching Box is connected. Use this flag as a condition when interlocking the ladder program.

Signal

The Teaching Box Connected Flag settings have the following meanings.

Signal	Meaning
1	ON when the Teaching Box is connected.
0	OFF when the Teaching Box isn't connected.

6-6-34 Word n+12 Bits 00 to 03: General Inputs 1 to 4

These bits indicate the status of general inputs 1 through 4. The corresponding bit will be ON when the general input is ON, OFF when the general input is OFF.

Signal

The status of these bits have the following meanings.

Signal	Meaning
1	ON when the corresponding general input is ON.
0	OFF when the corresponding general input is OFF.

6-6-35 Word n+12 Bits 04 to 07: General Outputs 1 to 4

These bits indicate the status of general outputs 1 through 4. The corresponding bit will be ON when the general output is ON, OFF when the general output is OFF.

Signal

The status of these bits have the following meanings.

Signal	Meaning
1	ON when the corresponding general output is ON.
0	OFF when the corresponding general output is OFF.

6-6-36 Word n+13 Bit 00: Task Error Flag (Task 1)

This flag is turned ON when a task error has occurred in task 1. The flag can be reset with task 1's Task Error Reset Bit (n+2 bit 00). If the cause of the error isn't corrected, the flag will be turned ON again the next time the task is executed.

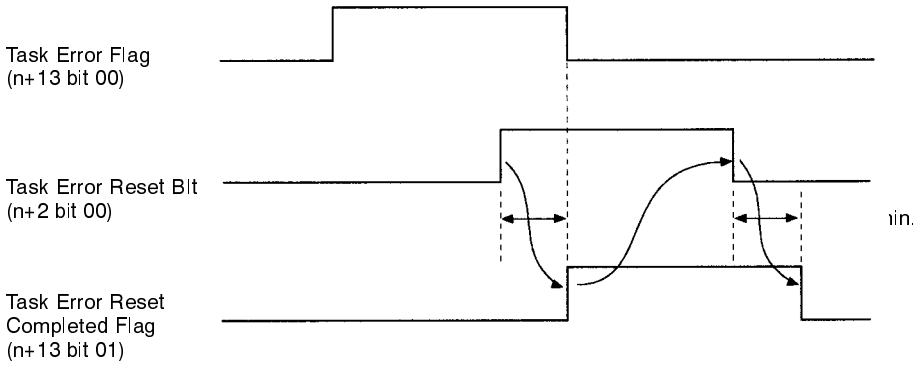
The Task Error Flags for tasks 2, 3, and 4 (bit 00 in words n+14, n+15, and n+16) operate in the same way.

Signal

The Task Error Flag settings have the following meanings.

Signal	Meaning
↑ (OFF-to-ON)	Goes ON when a task error has occurred.
↓ (ON-to-OFF)	Goes OFF when reset with the Task Error Reset Bit.
0	Indicates normal operation.

Timing Chart



6-6-37 Word n+13 Bit 01: Task Error Reset Completed Flag (Task 1)

This flag is turned ON when task error reset processing has been completed. Use this bit to determine when the Task Error Reset BIt can be turned OFF. Turn the Task Error Reset BIt OFF when this flag is turned ON.

The Task Error Reset Completed Flags for tasks 2, 3, and 4 (bit 01 in words n+14, n+15, and n+16) operate in the same way.

Signal

The Task Error Reset Completed Flag settings have the following meanings.

Signal	Meaning
↑ (OFF-to-ON)	Goes ON when task error reset processing is completed.
↓ (ON-to-OFF)	Goes OFF when the Task Error Reset Bit goes OFF.

6-6-38 Word n+13 Bit 02: Automatic Mode Flag (Task 1)

This flag indicates that the task is in automatic mode. Always make sure that this flag is ON when starting MC program execution (with the Cycle Start Bit). It will be ignored if this flag is OFF.

Use the four flags listed below to determine the operating status of a task in automatic mode. When the program is executed from the Teaching Box the flags will change just as they do in automatic mode. These flags are all turned OFF when the mode is switched from automatic to manual.

- Memory Run Flag (n+13 bit 04)
- Memory Run Completed Flag (n+13 bit 05)
- Optional Input Standby Flag (n+13 bit 06)
- M Code Reset Standby Flag (n+13 bit 07)

This flag won't be turned ON during system initialization, even if an automatic mode command is received from the PC. The flag will be turned ON when initialization is completed and the Unit is able to received commands in automatic mode. Initialization takes about 130 ms.

The Automatic Mode Flags for tasks 2, 3, and 4 (bit 02 in words n+14, n+15, and n+16) operate in the same way.

Signal

The Automatic Mode Flag settings have the following meanings.

Signal	Meaning
1	ON when the task is in automatic mode.
0	OFF when the task is in automatic mode.

6-6-39 Word n+13 Bit 03: Manual Mode Flag (Task 1)

This flag indicates that the task is in manual mode. Always make sure that this flag is ON when executing commands that are valid only in manual mode. Those commands will be ignored if this flag is OFF.

This flag won't be turned ON during system initialization, even if an manual mode command is received from the PC. The flag will be turned ON when initialization is completed and the Unit is able to received commands in manual mode. Initialization takes about 130 ms.

The Manual Mode Flags for tasks 2, 3, and 4 (bit 03 in words n+14, n+15, and n+16) operate in the same way.

Signal

The Manual Mode Flag settings have the following meanings.

Signal	Meaning
1	ON when the task is in manual mode.
0	OFF when the task is in manual mode.

6-6-40 Word n+13 Bit 04: Memory Run Flag (Task 1)

This flag indicates whether the MC program is being executed in automatic mode. This flag is also ON when the M code is being reset. The flag will remain ON when decelerating to a stop; it will be turned OFF when the axes are fully stopped.

Use the status of the Memory Run Flag to determine how long to keep ON control bits such as the Pause Bit or Forced Block End Bit when executing those commands. This flag can also be used as an interlock condition, checking whether the MC program is being executed or not.

The Memory Run Flags for tasks 2, 3, and 4 (bit 04 in words n+14, n+15, and n+16) operate in the same way.

Signal

The Memory Run Flag settings have the following meanings.

Signal	Meaning
1	ON when the MC program is being executed.
0	OFF when the MC program isn't being executed.

6-6-41 Word n+13 Bit 05: Memory Run Completed Flag (Task 1)

This flag is turned ON when a G79 function (PROGRAM END) is executed. Use the Memory Run Completed Flag to determine when to turn OFF the Cycle Start Bit. The Memory Run Flag will be turned OFF when this flag is turned ON.

The Memory Run Completed Flag will be turned OFF again the next time the MC program is executed. This bit will be turned OFF when the mode is switched from automatic to manual, but it will be turned ON again when the mode is switched back to automatic.

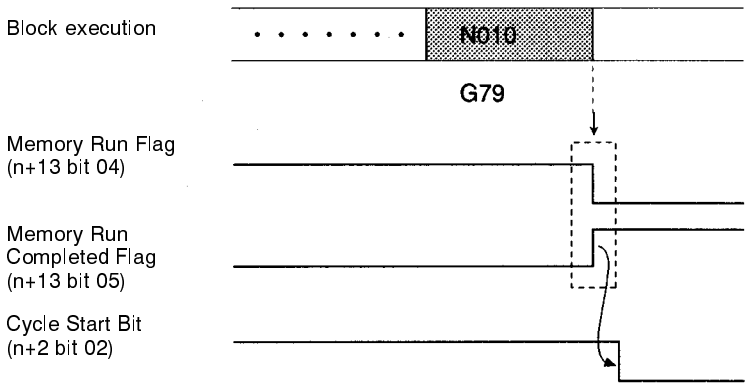
The Memory Run Completed Flags for tasks 2, 3, and 4 (bit 05 in words n+14, n+15, and n+16) operate in the same way.

Signal

The Memory Run Completed Flag settings have the following meanings.

Signal	Meaning
1	ON when the MC program has ended (when G79 was executed).
↓ (ON-to-OFF)	OFF when the MC program is executed or the mode is switched from automatic to manual.

Timing Chart



6-6-42 Word n+13 Bit 06: Optional Input Standby Flag (Task 1)

This flag indicates whether a G-code function that waits for an optional input is being executed. The Unit will be in a standby status when this flag is ON, so turn the optional input OFF or ON using a condition from the ladder program. The following standby states are possible:

- The OPTIONAL STOP function (G76) stops MC program execution when the specified optional input goes ON and waits for the input to go OFF.
- If an optional input is added as a parameter in an INTERPOLATION function (G00, G01, or G02), the function will be executed when the specified optional input goes ON.

The Optional Input Standby Flags for tasks 2, 3, and 4 (bit 06 in words n+14, n+15, and n+16) operate in the same way.

Signal

The Optional Input Standby Flag settings have the following meanings.

Signal	Meaning
1	ON when waiting for an optional input.
0	OFF when not waiting for an optional input.

6-6-43 Word n+13 Bit 07: M Code Reset Standby Flag (Task 1)

This flag indicates whether MC program execution has been stopped to wait for an M code reset. When this flag is ON, MC program execution won't continue until the M Code Reset Bit is turned ON and then OFF again. The standby status occurs when an M code from 0 to 499 is output.

The Memory Run Flag will go ON when this flag is ON.

It is convenient to use this flag when determining whether an M code stands by for an M code reset (M codes 0 to 499) or not (M codes 500 to 999).

The M Code Reset Standby Flags for tasks 2, 3, and 4 (bit 07 in words n+14, n+15, and n+16) operate in the same way.

Signal

The M Code Reset Standby Flag settings have the following meanings.

Signal	Meaning
1	ON when waiting for an M code reset.
0	OFF when not waiting for an M code reset.

6-6-44 Word n+13 Bit 08: M Strobe Flag (Task 1)

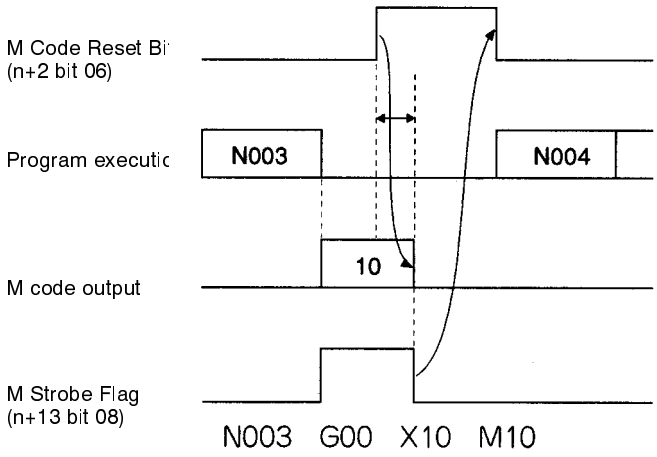
This flag indicates whether an M code is being output. The M Strobe Flags for tasks 2, 3, and 4 (bit 08 in words n+14, n+15, and n+16) operate in the same way.

Signal

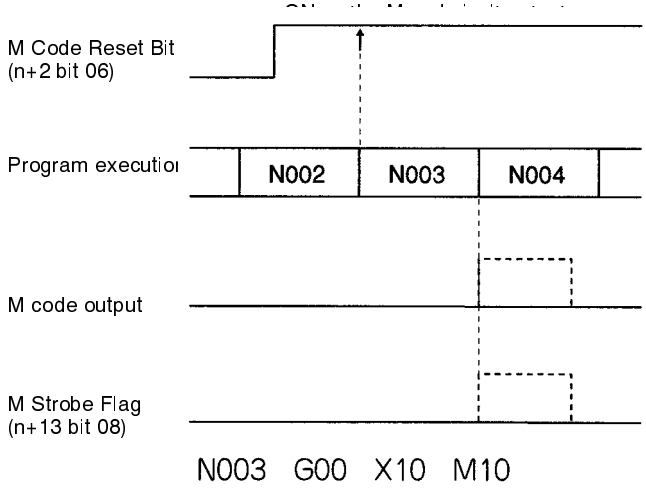
The M Strobe Flag settings have the following meanings.

Signal	Meaning
1	ON when there is an M code output (0 to 999).
0	OFF when there isn't an M code output.

Timing Chart 1



Timing Chart 2



6-6-45 Word n+13 Bit 09: Cycle Start Received Flag (Task 1)

This flag is turned ON when the up-differentiation of the Cycle Start Bit is received. Use this flag to control Cycle Start Bit timing when executing one block at a time (single block execution).

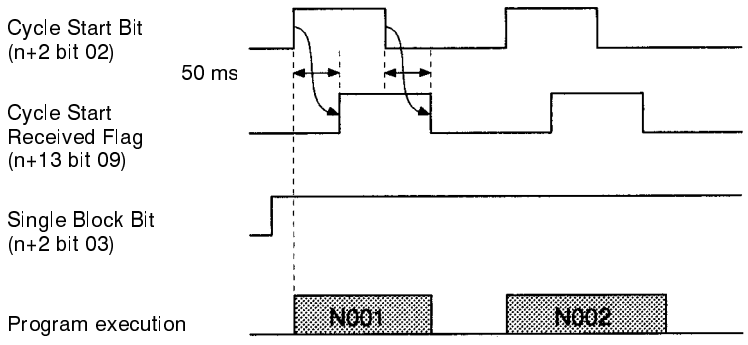
The Cycle Start Received Flags for tasks 2, 3, and 4 (bit 09 in words n+14, n+15, and n+16) operate in the same way.

Signal

The Cycle Start Received Flag settings have the following meanings.

Signal	Meaning
1	ON when a valid Cycle Start Bit signal has been received.
0	OFF when the Cycle Start Bit signal is OFF.

Timing Chart



6-6-46 Word n+13 Bit 12: Teaching Address Setting Completed (Task 1)

This flag is turned ON when the Teaching Address Set Bit's processing has been completed. Use this flag together with the Teaching Address Set Bit.

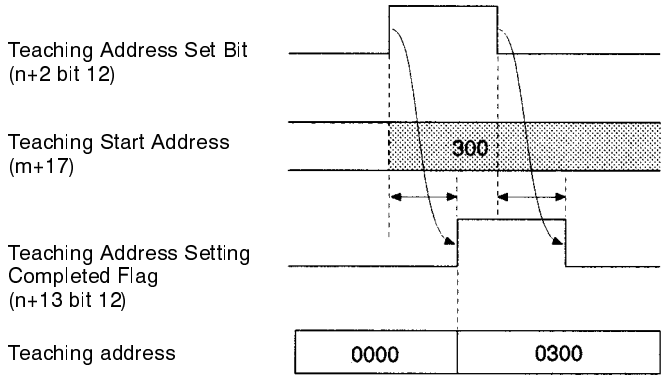
The Teaching Address Setting Completed Flags for tasks 2, 3, and 4 (bit 12 in words n+14, n+15, and n+16) operate in the same way.

Signal

The Teaching Address Setting Completed Flag settings have the following meanings.

Signal	Meaning
↑ (OFF-to-ON)	Goes ON when the teaching address setting is completed.
↓ (ON-to-OFF)	Goes OFF when the Teaching Address Set Bit goes OFF.

Timing Chart



6-6-47 Word n+13 Bit 13: Teaching Completed Flag (Task 1)

This flag is turned ON when the Teaching Bit's processing has been completed. Use this flag together with the Teaching Bit.

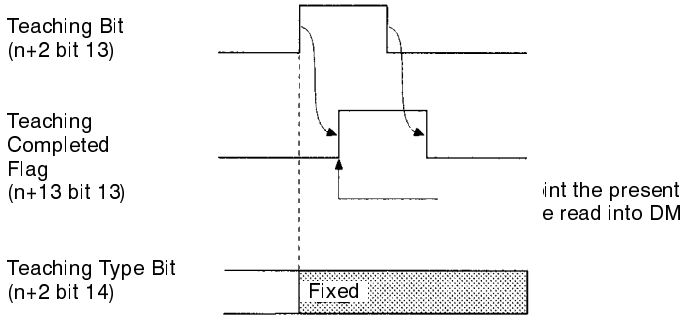
The Teaching Completed Flags for tasks 2, 3, and 4 (bit 13 in words n+14, n+15, and n+16) operate in the same way.

Signal

The Teaching Completed Flag settings have the following meanings.

Signal	Meaning
↑ (OFF-to-ON)	Goes ON when teaching is completed.
↓ (ON-to-OFF)	Goes OFF when the Teaching Bit goes OFF.

Timing Chart



6-6-48 Word n+13 Bit 14: Teaching Error Flag (Task 1)

This flag is turned ON when a teaching error occurs. When an error has occurred, the Teaching Completed Flag and Teaching Address Setting Completed Flag will be turned ON at the same time.

For example, errors will occur in the following cases:

- Teaching Bit is turned ON, but the origin isn't fixed.
- Teaching address isn't within the acceptable range (0 to 1999).

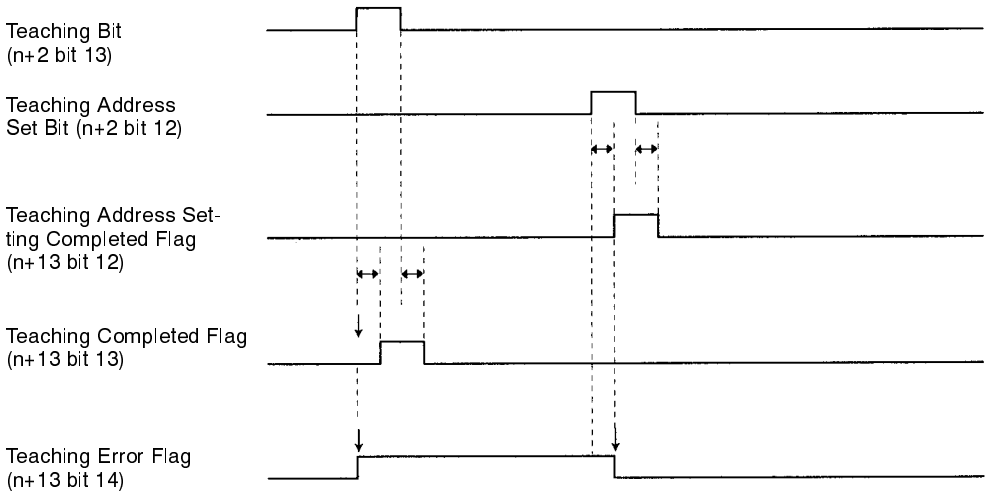
The Teaching Error Flags for tasks 2, 3, and 4 (bit 14 in words n+14, n+15, and n+16) operate in the same way.

Signal

The Teaching Error Flag settings have the following meanings.

Signal	Meaning
↑ (OFF-to-ON)	Goes ON when a teaching error occurs.
↓ (ON-to-OFF)	Goes OFF the next time that teaching is completed normally or teaching address setting is completed normally.

Timing Chart



6-6-49 Word n+13 Bit 15: Teaching Address Over Flag (Task 1)

This flag is turned ON when the Teaching Bit is turned ON, but the position data address set in the Memory Parameter Edit menu (a submenu of the Edit MC Parameters menu) isn't within the acceptable range. The Teaching Completed Flag will be turned ON at the same time.

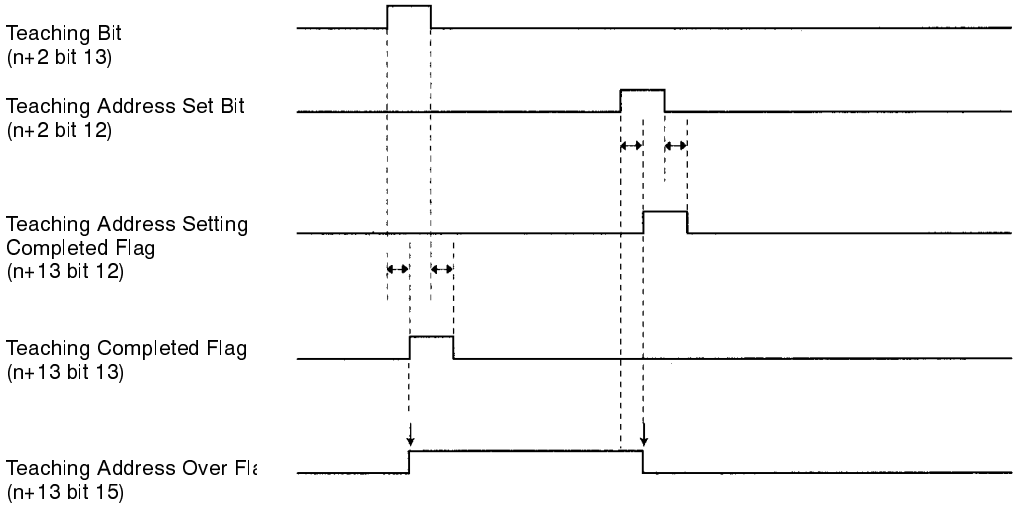
The Teaching Address Over Flags for tasks 2, 3, and 4 (bit 15 in words n+14, n+15, and n+16) operate in the same way.

Signal

The Teaching Address Over Flag settings have the following meanings.

Signal	Meaning
↑ (OFF-to-ON)	Goes ON when the position data address exceeds the range.
↓ (ON-to-OFF)	Goes OFF when teaching address setting is completed normally or teaching is completed normally.

Timing Chart



6-6-50 Word n+17 Bit 00: Reference Origin Flag (X-axis)

This flag is turned ON when the axis reaches or passes the reference origin and is within the in position range specified in the system parameters.

The Reference Origin Flags for the Y, Z, and U axes (bit 00 in words n+19, n+21, and n+23) operate in the same way.

Signal

The Reference Origin Flag settings have the following meanings.

Signal	Meaning
1	ON when the axis is within the reference origin's in position range.
0	OFF when the axis isn't within the reference origin's in position range.

6-6-51 Word n+17 Bit 01: Workpiece Origin Flag (X-axis)

This flag is turned ON when the axis reaches or passes the workpiece origin and is within the in position range specified in the system parameters.

The Workpiece Origin Flags for the Y, Z, and U axes (bit 01 in words n+19, n+21, and n+23) operate in the same way.

Signal

The Workpiece Origin Flag settings have the following meanings.

Signal	Meaning
1	ON when the axis is within the workpiece origin's in position range.
0	OFF when the axis isn't within the workpiece origin's in position range.

6-6-52 Word n+17 Bit 02: MPG Operation Flag (X-axis)

This flag is turned ON when the Enable MPG Bit is turned ON and MPG operation is being performed. The following commands can't be performed while this flag is ON.

Origin Search
Reference Origin Return
Jogging
Servo Lock
Servo Free
Current Position Preset

This flag is turned ON at the same time as the Busy Flag, and is turned OFF when the Enable MPG Bit is turned OFF.

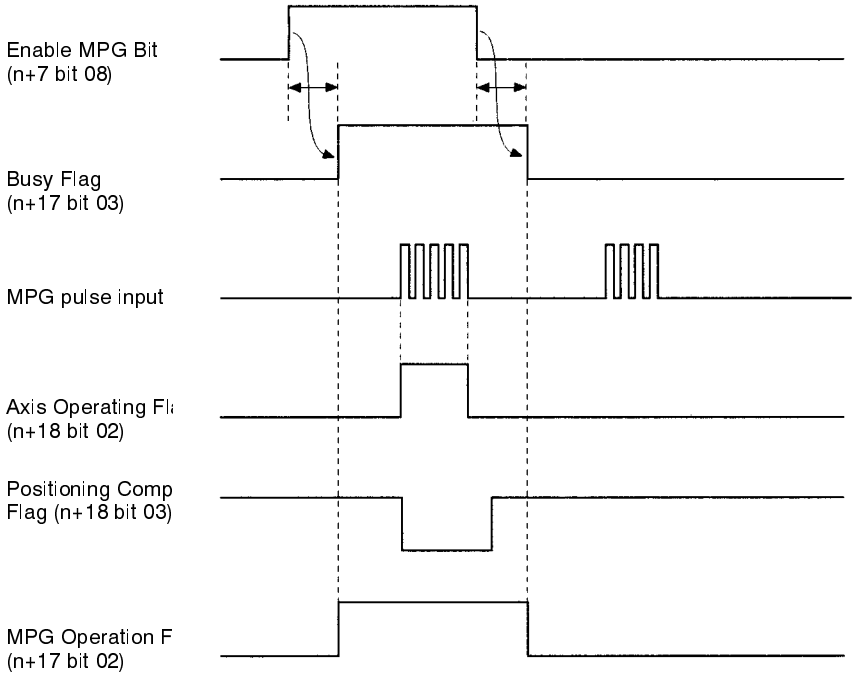
The MPG Operation Flags for the Y, Z, and U axes (bit 02 in words n+19, n+21, and n+23) operate in the same way.

Signal

The MPG Operation Flag settings have the following meanings.

Signal	Meaning
1	ON when MPG operation is being performed.
0	OFF when MPG operation isn't being performed.

Timing Chart



6-6-53 Word n+17 Bit 03: Busy Flag (X-axis)

This flag is turned ON when a command is being executed in manual mode. The Busy Flag will be turned ON when one of the following commands are being executed.

Origin Search
Reference Origin Return
Jogging
Servo Lock
Servo Free
Current Position Preset
MPG Operation

The Busy Flag will be turned OFF when the bit controlling the command is turned OFF. Another command can't be received while the Busy Flag is ON.

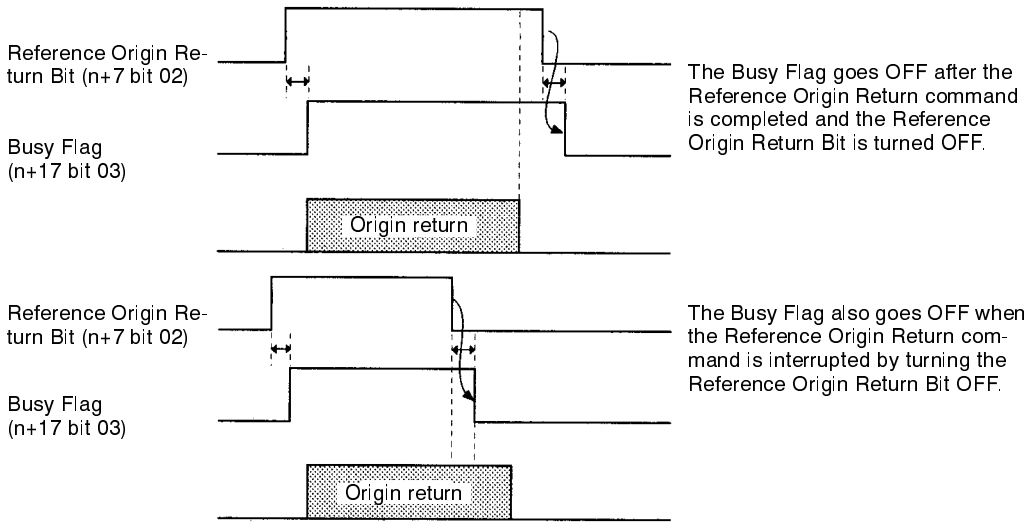
The Busy Flags for the Y, Z, and U axes (bit 03 in words n+19, n+21, and n+23) operate in the same way.

Signal

The Busy Flag settings have the following meanings.

Signal	Meaning
↑ (OFF-to-ON)	Goes ON when a command is executed in manual mode.
1	Stays ON while the manual mode command is being processed.
↓ (ON-to-OFF)	Goes OFF when the command's control bit is turned OFF.

Timing Chart



6-6-54 Word n+17 Bit 04: Transmission #1 Completed Flag

This flag indicates whether transmission #1 has been completed. Use this flag to determine when to turn OFF the Transmission #1 Control Bit. This flag won't be turned OFF until the Transmission #1 Control Bit is turned OFF, even if the transmission has been completed. The Reception #1 Control Bit's signal can't be received while this flag is ON.

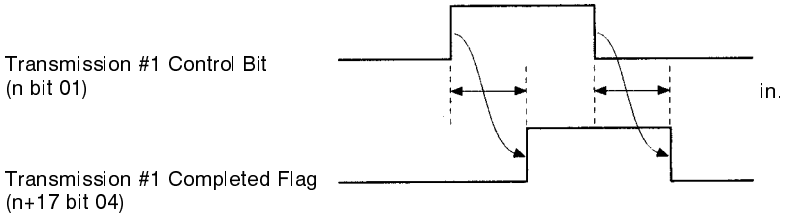
The Transmission #2, #3, and #4 Completed Flags (bit 04 in words n+19, n+21, and n+23) operate in the same way.

Signal

The Transmission #1 Completed Flag settings have the following meanings.

Signal	Meaning
↑ (OFF-to-ON)	Goes ON when transmission #1 processing is completed.
↓ (ON-to-OFF)	Goes OFF when the Transmission #1 Control Bit is turned OFF.

Timing Chart



6-6-55 Word n+17 Bit 05: Reception #1 Completed Flag

This flag indicates whether reception #1 from the MC Unit has been completed. Use this flag to determine when to turn OFF the Reception #1 Control Bit. This flag won't be turned OFF until the Reception #1 Control Bit is turned OFF, even if the reception has been completed. The Transmission #1 Control Bit's signal can't be received while this flag is ON.

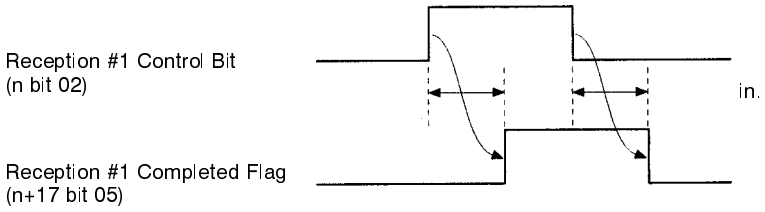
The Reception #2, #3, and #4 Completed Flags (bit 05 in words n+19, n+21, and n+23) operate in the same way.

Signal

The Reception #1 Completed Flag settings have the following meanings.

Signal	Meaning
↑ (OFF-to-ON)	Goes ON when reception #1 processing is completed.
↓ (ON-to-OFF)	Goes OFF when the Reception #1 Control Bit is turned OFF.

Timing Chart



6-6-56 Word n+17 Bit 06: Transfer #1 Error Flag

This flag is turned ON when an error occurred during execution of transmission #1 or reception #1, so the command couldn't be completed normally. This flag is turned ON at the same time as the Transmission/Reception #1 Completed Flag; it is turned OFF the next time the Transmission/Reception #1 Completed Flag is turned ON after a normal data transfer.

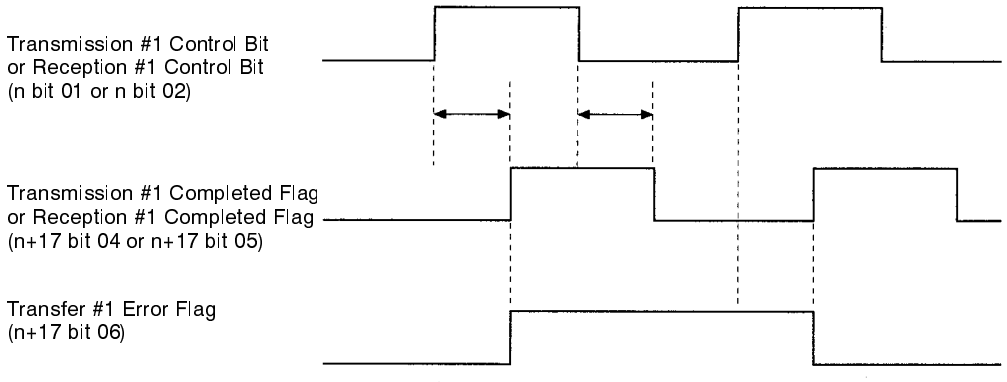
The Transfer #2, #3, and #4 Error Flags (bit 06 in words n+19, n+21, and n+23) operate in the same way.

Signal

The Transfer #1 Error Flag settings have the following meanings.

Signal	Meaning
↑ (OFF-to-ON)	Goes ON when an error occurs in transmission/reception #1.
↓ (ON-to-OFF)	Goes OFF the next time transmission/reception #1 is completed normally.

Timing Chart



6-6-57 Word n+17 Bit 07: Override Data Alarm Flag (X-axis)

When the override setting command is executed by turning ON the Override Setting Bit (n+7 bit 12), the override data set in DM word m+33 is check and this flag is turned ON if the data is incorrect. An error will occur if the data isn't BCD or isn't within the acceptable range (0.1 to 199.9). The flag will be turned OFF when the override setting command is executed and the override data is correct.

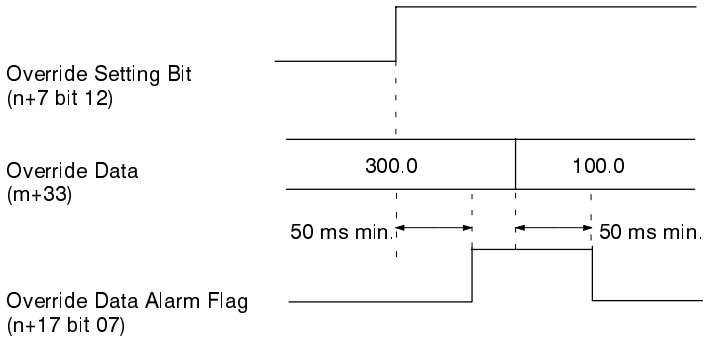
The Override Data Alarm Flags for the Y, Z, and U axes (bit 07 in words n+19, n+21, and n+23) operate in the same way.

Signal

The Override Data Alarm Flag settings have the following meanings.

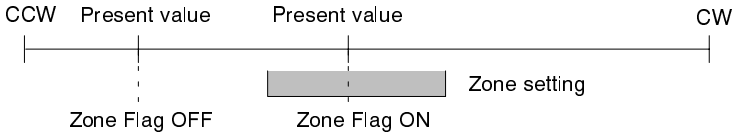
Signal	Meaning
↑ (OFF-to-ON)	Goes ON when the override setting command is executed and the override data is incorrect.
↓ (ON-to-OFF)	Goes OFF when the override setting command is executed and the override data is correct.

Timing Chart



6-6-58 Word n+17 Bits 08 to 15: Zone Flags (X-axis)

Word n+17 bits 08 through 15 are the flags for zones 1 through 8. These flags are turned ON when the axis is within the zone specified in the zone parameters.



The Zone Flags for the Y, Z, and U axes (bits 08 to 15 in words n+19, n+21, and n+23) operate in the same way.

Signal

The Zone Flag settings have the following meanings.

Signal	Meaning
1	ON when axis is within the zone's range.
0	OFF when axis is outside of the zone's range.

6-6-59 Word n+18 Bit 00: Servo Lock ON Flag (X-axis)

This flag is turned ON when the servo lock is ON. The servo lock is a status that outputs a Run command to the servodriver and forms a position loop. Execute positioning functions when this flag is ON; an error will occur if they are executed when this flag is OFF.

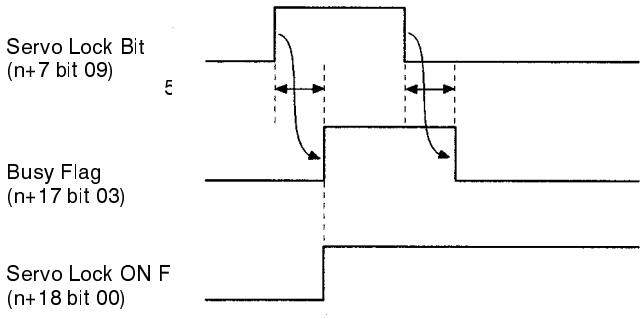
The Servo Lock ON Flags for the Y, Z, and U axes (bit 00 in words n+20, n+22, and n+24) operate in the same way.

Signal

The Servo Lock ON Flag settings have the following meanings.

Signal	Meaning
1	ON when the servo lock is ON.
0	OFF when the servo lock is OFF.

Timing Chart



6-6-60 Word n+18 Bit 01: No Origin Flag (X-axis)

This flag is turned ON when the reference origin isn't fixed. In automatic mode, execute positioning functions when this flag is OFF. In manual mode, execute origin returns when this flag is OFF. An error will occur if these operations are executed when this flag is ON.

The No Origin Flag will be turned ON in the following cases:

- The power is turned ON and the machine parameter's origin search method isn't set to "set origin at power on."
- An origin search operation is started.
- An encoder wiring disconnect error has occurred.

The origin will be fixed and the No Origin Flag will be turned OFF in the following cases:

- The power is turned ON and the machine parameter's origin search method is set to "set origin at power on."
- An origin search operation is completed normally.
- A current position preset command is completed.
- A G54 function (CHANGE REFERENCE COORDINATE SYSTEM PV) is executed.

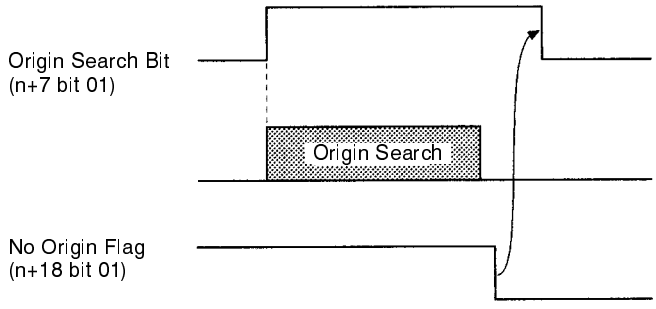
The No Origin Flags for the Y, Z, and U axes (bit 01 in words n+20, n+22, and n+24) operate in the same way.

Signal

The No Origin Flag settings have the following meanings.

Signal	Meaning
1	ON when the location of the reference origin is fixed.
0	OFF when the location of the reference origin isn't fixed.

Timing Chart



6-6-61 Word n+18 Bit 02: Axis Operating Flag (X-axis)

This flag is turned ON when axis is operating. An operating axis is an axis that is being given a command value to move the axis. For example, when the axis is being operated with a trapezoidal curve, the axis is operating from the start of the acceleration command to the end of the deceleration command. Use this flag to determine when the axis has stopped after executing a deceleration stop command.

The Axis Operating Flags for the Y, Z, and U axes (bit 02 in words n+20, n+22, and n+24) operate in the same way.

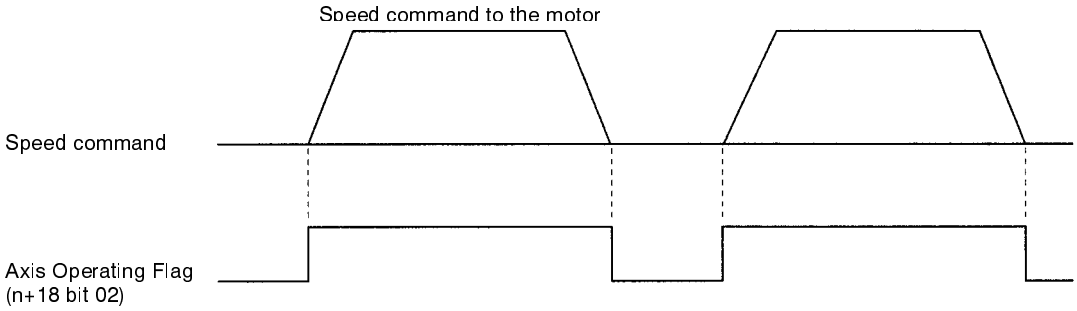
Signal

The Axis Operating Flag settings have the following meanings.

Signal	Meaning
1	ON when an axis movement command is being executed.
0	OFF when an axis movement command isn't being executed.

Timing Chart

The following example shows a trapezoidal speed output to the motor.



6-6-62 Word n+18 Bit 03: Positioning Completed Flag (X-axis)

This flag is turned ON when the axis reaches its target position in a positioning operation and the axis is within the in position range specified in the system parameters. Use this flag as a condition to check when the target position has been reached and the next function can be executed.

Once the axis enters the in position range of the target position and the Positioning Completed Flag is turned ON, the flag will remain ON until the next positioning operation is performed, even if the axis leaves the in position range.

The Positioning Completed Flag will be turned ON if the positioning operation is interrupted before the target position is reached but the axis is within the in position range.

In position checks are performed regularly during MPG operation and this flag will be turned ON when the axis is within the in position range.

When a positioning operation is started, the Positioning Completed Flag will be turned OFF at the same time that the Axis Operating Flag is turned ON.

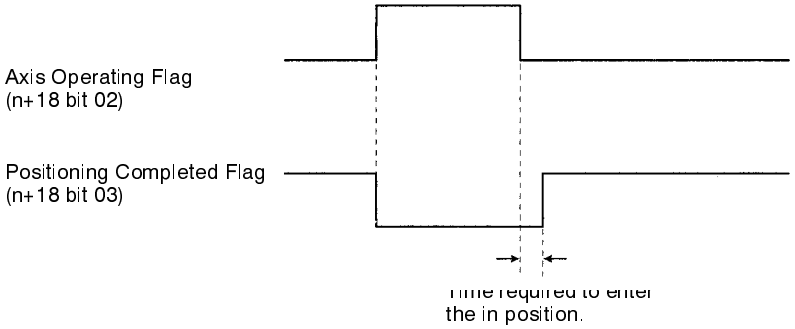
The Positioning Completed Flags for the Y, Z, and U axes (bit 03 in words n+20, n+22, and n+24) operate in the same way.

Signal

The Positioning Completed Flag settings have the following meanings.

Signal	Meaning
1	ON when the axis is within the in position range.
0	OFF when the axis isn't within the in position range.

Timing Chart



6-6-63 Word n+18 Bit 07: Error Counter Alarm Flag (X-axis)

This flag is turned ON when the count in the error counter exceed the "error warn count" specified in the Servo Parameters. This flag will be turned ON as a warning, but the positioning operation will continue as normal.

The Error Counter Alarm Flags for the Y, Z, and U axes (bit 07 in words n+20, n+22, and n+24) operate in the same way.

Signal

The Error Counter Alarm Flag settings have the following meanings.

Signal	Meaning
1	ON when the error counter exceeds the error warn count.
0	OFF when the error counter is less than the error warn count.

6-6-64 Word n+18 Bit 08: CCW Limit Input (X-axis)

This flag is turned ON when the CCW limit input is ON; it is turned OFF when the CCW limit input is OFF. The CCW Limit Inputs for the Y, Z, and U axes (bit 08 in words n+20, n+22, and n+24) operate in the same way.

Signal

The CCW Limit Input settings have the following meanings.

Signal	Meaning
1	ON when the CCW limit input is ON.
0	OFF when the CCW limit input is OFF.

6-6-65 Word n+18 Bit 09: CW Limit Input (X-axis)

This flag is turned ON when the CW limit input is ON; it is turned OFF when the CW limit input is OFF. The CW Limit Inputs for the Y, Z, and U axes (bit 09 in words n+20, n+22, and n+24) operate in the same way.

Signal

The CW Limit Input settings have the following meanings.

Signal	Meaning
1	ON when the CW limit input is ON.
0	OFF when the CW limit input is OFF.

6-6-66 Word n+18 Bit 10: Origin Proximity Input (X-axis)

This flag is turned ON when the origin proximity input is ON; it is turned OFF when the origin proximity input is OFF. The origin proximity inputs for the Y, Z, and U axes (bit 10 in words n+20, n+22, and n+24) operate in the same way.

Signal

The Origin Proximity Input settings have the following meanings.

Signal	Meaning
1	ON when the origin proximity input is ON.
0	OFF when the origin proximity input is OFF.

6-6-67 Word n+18 Bit 11: Emergency Stop Input (X-axis)

This flag is turned ON when the emergency stop input is ON; it is turned OFF when the emergency stop input is OFF. The emergency stop inputs for the Y, Z, and U axes (bit 11 in words n+20, n+22, and n+24) operate in the same way.

Signal The Emergency Stop Input settings have the following meanings.

Signal	Meaning
1	ON when the emergency stop input is ON.
0	OFF when the emergency stop input is OFF.

6-6-68 Word n+18 Bit 12: Alarm Input (X-axis)

This flag is turned ON when the driver alarm input is ON; it is turned OFF when the alarm input is OFF. The alarm inputs for the Y, Z, and U axes (bit 12 in words n+20, n+22, and n+24) operate in the same way.

Signal The Alarm Input settings have the following meanings.

Signal	Meaning
1	ON when the driver alarm input is ON.
0	OFF when the driver alarm input is OFF.

6-6-69 Word n+18 Bit 13: Run Command Output (X-axis)

This flag is turned ON when the Run command output is being output; it is turned OFF when the Run command output isn't being output. The Run command outputs for the Y, Z, and U axes (bit 13 in words n+20, n+22, and n+24) operate in the same way.

Signal The Run Command Output settings have the following meanings.

Signal	Meaning
1	ON when the run command output is being output.
0	OFF when the run command output isn't being output.

6-6-70 Word n+18 Bit 14: Alarm Reset Output (X-axis)

This flag is turned ON when the driver alarm reset output is being output; it is turned OFF when the alarm reset output isn't being output. The alarm reset outputs for the Y, Z, and U axes (bit 14 in words n+20, n+22, and n+24) operate in the same way.

Signal The Alarm Reset Output settings have the following meanings.

Signal	Meaning
1	ON when the driver alarm reset output is being output.
0	OFF when the driver alarm reset output isn't being output.

SECTION 7
Fixing the Origin

This section describes how to fix the location of the origin used as a reference point in positioning operations.

7-1 Introduction

In motion control systems using incremental encoders, the location of the origin must be fixed after the power is turned ON. There are two ways to fix the origin in MC Units.

Operation	Description
Current position preset	The origin is fixed by setting the position where the motor is stopped to an arbitrary value.
Origin search	The origin is fixed by moving the motor and using the limit switches, origin proximity switches, and encoder's Z-phase signals. If the Machine Parameter's Origin Search Method parameter is set to "Set origin at power ON," the Unit will automatically fix the position when power is turned ON as the origin, without moving the motor.

The current position preset operation can be performed by turning ON the Current Position Preset Bit (n+7 bit 05) or executing function G54 (CHANGE REFERENCE COORDINATE SYSTEM PV).

The origin search operation can be performed by turning ON the Origin Search Bit (n+7 bit 01) or executing function G28 (ORIGIN SEARCH). An origin search can also be started from the Teaching Box.

7-2 Sensors Required for an Origin Search

The following sensors and conditions are required in order to perform an origin search.

Sensor	Description
CW and CCW limit switches	Normally closed inputs are required.
Origin proximity switch	The logic of this sensor can be changed in the Machine Parameters, so either normally closed or normally open inputs can be used.
Encoder's Z-phase signal	The MC Unit detects the signal when it changes from Low to High.

CW and CCW Limit Switches These switches indicate that the workpiece's maximum range has been reached when the motor rotates clockwise (CW) or counterclockwise (CCW).

Origin Proximity Switch This switch indicates that the axis is near the origin.

Encoder's Z-phase Signal This signal is output when the encoder has rotated one revolution. The first Z-phase position after the origin proximity switch input changes from ON to OFF is the actual origin.

7-3 Origin Search Parameters

Seven parameters are used in the origin search operation. These parameters are set with the MC Support Software.

Origin Search Method
Origin Search Direction
Origin Deceleration Method
Origin Proximity Logic
Origin Search High-speed Feedrate
Origin Search Low-speed Feedrate
Reference Origin Offset

Origin Search Method

There are three possible settings for the Origin Search Method.

Set Origin at Power ON Mode
The origin is automatically set to the motor's position when power is turned ON.

Reverse Mode
The direction is reversed if a limit switch input is received during the origin search. The origin will be fixed when the Z-phase signal goes ON after the origin proximity switch goes from ON to OFF (normally open input).

When there isn't an origin proximity switch, a limit switch can be used instead.

1 Direction Mode
The origin search is performed in just one direction. An error will occur if a limit switch input is received before the origin proximity switch input.

When there isn't an origin proximity switch, a limit switch can be used instead.

Setting the Parameters

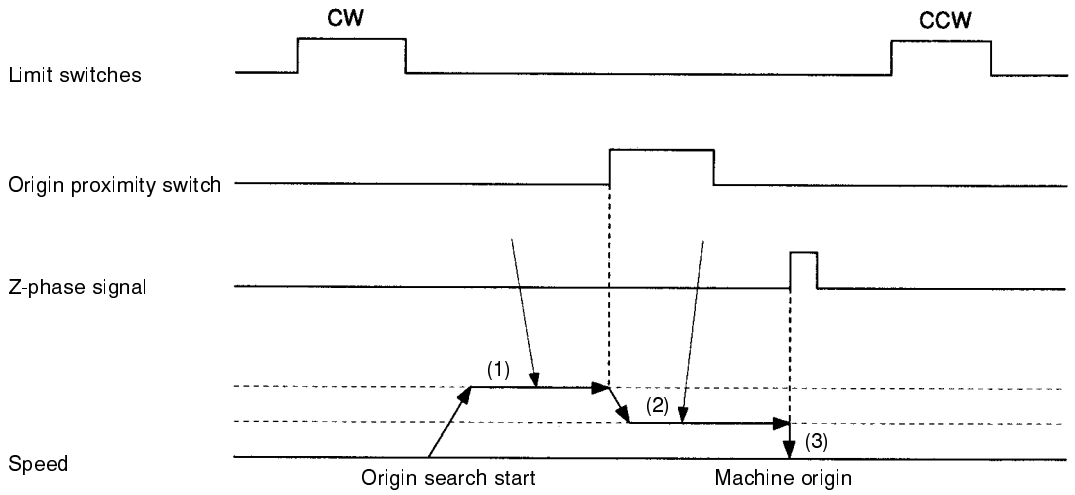
Refer to the *MC Support Software Operation Manual* for details on setting these parameters. The Origin Search Method, Origin Search Direction, Origin Deceleration Method, and Origin Proximity Logic parameters are in the Machine Parameters Edit menu. The Origin Search High-speed Feedrate and Origin Search Low-speed Feedrate parameters are in the Feedrate Parameters Edit menu. The Reference Origin Offset parameter is in the Coordinate Parameters Edit menu.

7-4 Origin Search Patterns

This section provides basic examples of origin search patterns with the Origin Search Method set to Reverse mode or One Direction mode. Operation will vary depending on the position of the workpiece when the origin search is executed.

7-4-1 Basic Origin Search Pattern

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW. Operation will vary and errors might occur with different limit switch, origin proximity switch, and Z-phase input timing. Refer to *Appendix B Additional Origin Search Patterns* for more examples.



Description

There are three steps involved in this origin search.

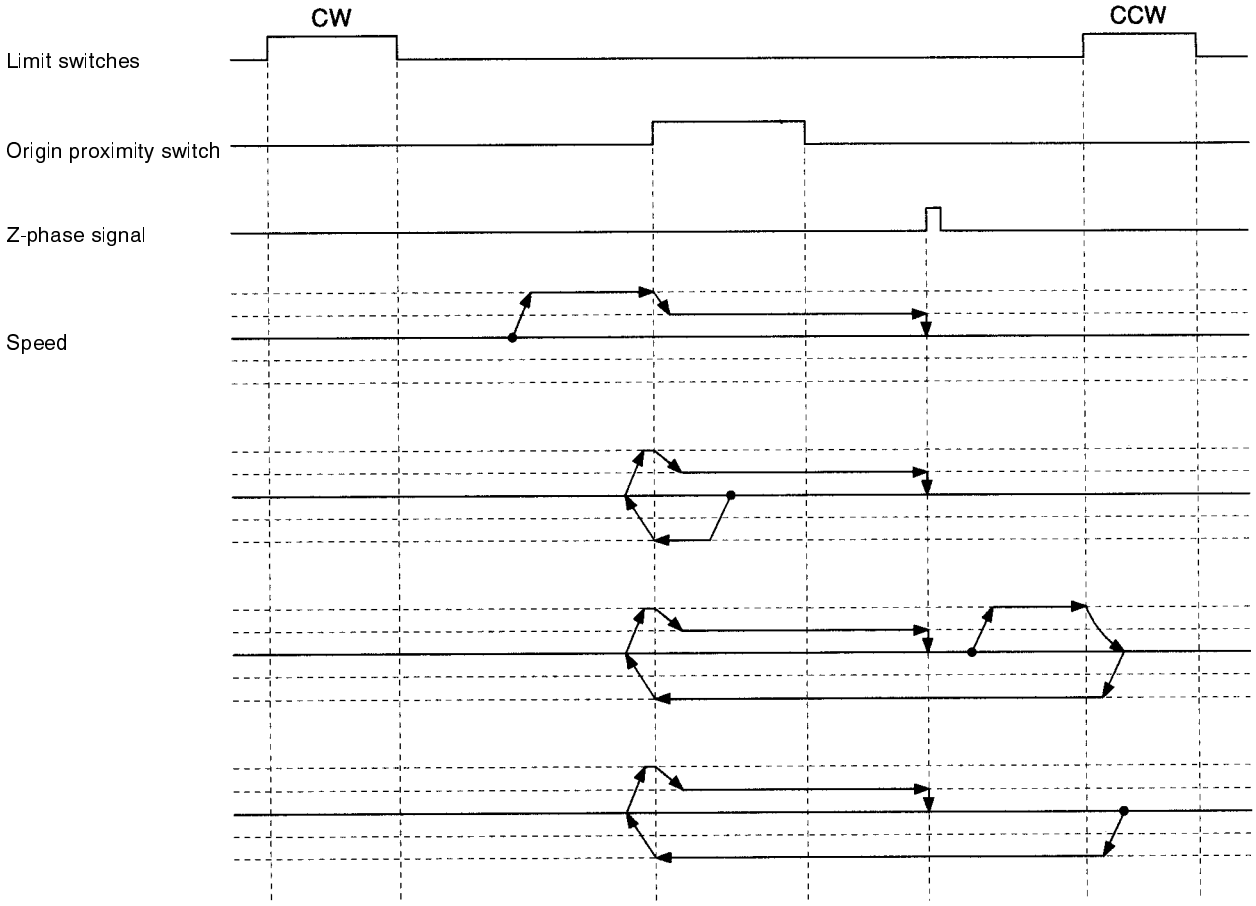
1, 2, 3...

1. When the origin search is executed, the axis is moved in the specified direction at the origin search high-speed feed rate. The acceleration time is used at this point.
2. When the origin proximity switch input is received, the search speed is reduced to the origin search low-speed feed rate. The deceleration time is used at this point.
3. After the origin proximity switch input goes OFF, the machine origin is fixed at the position where the Z-phase signal goes ON. The control voltage drops to zero at this point.

   The axis is then moved by the Reference Origin Offset amount with PTP control and the resulting position is fixed as the reference origin.

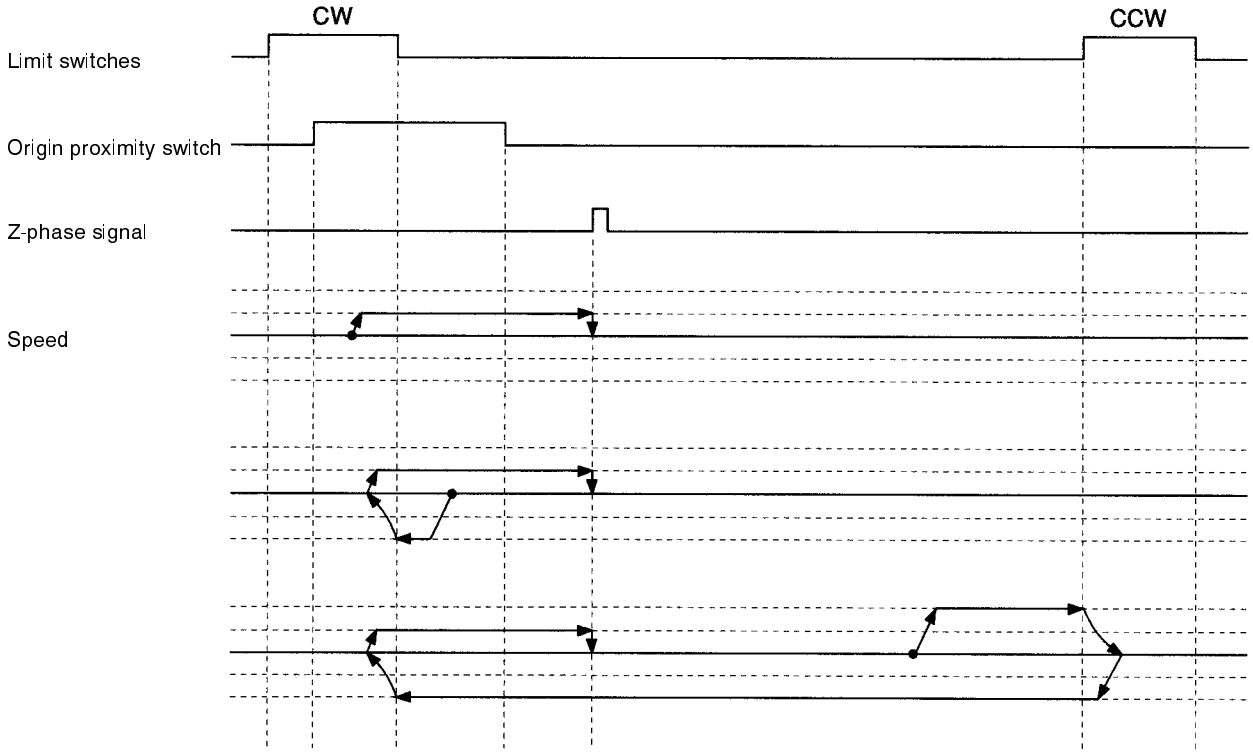
7-4-2 Reverse-mode Origin Searches 1

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW.



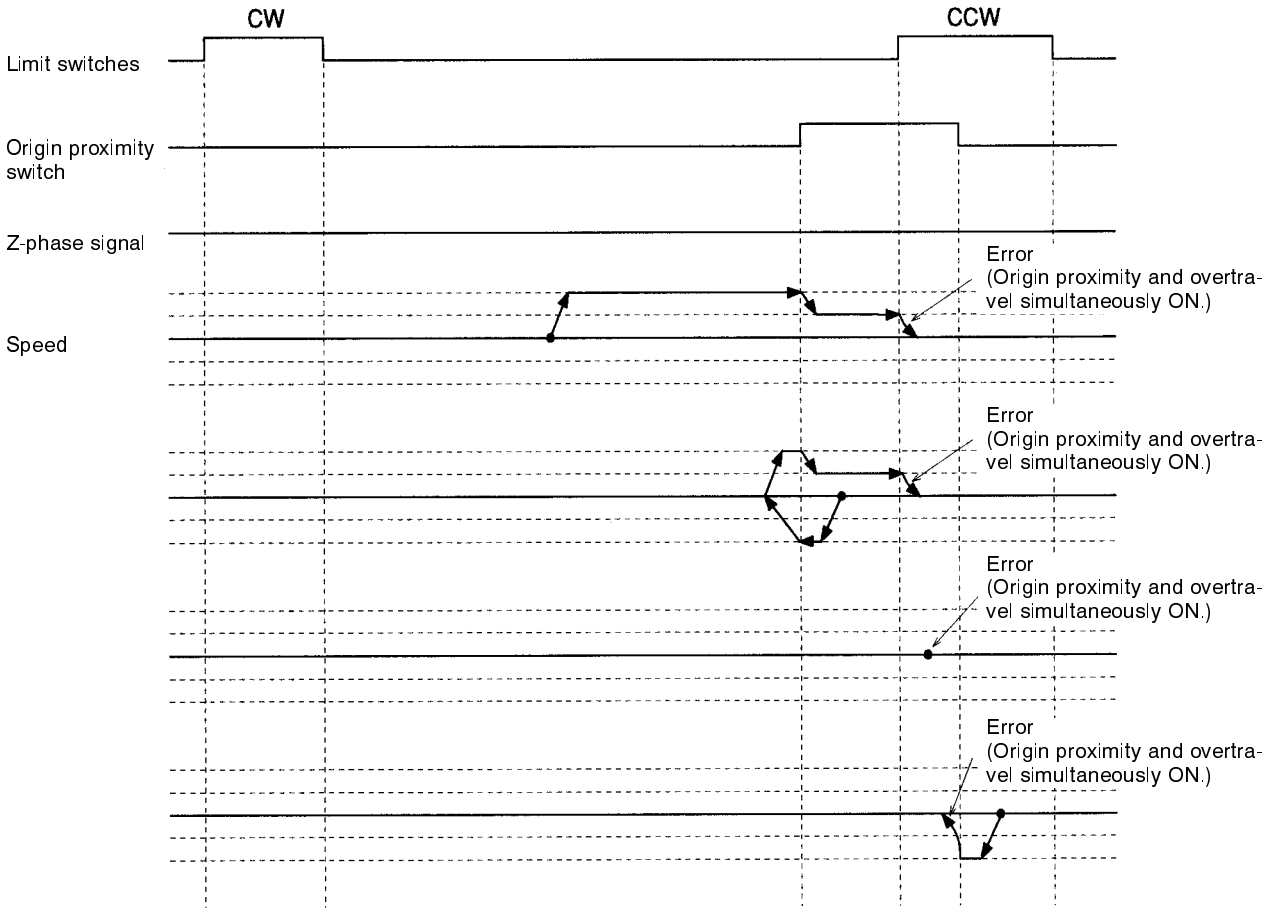
7-4-3 Reverse-mode Origin Searches 2

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW.



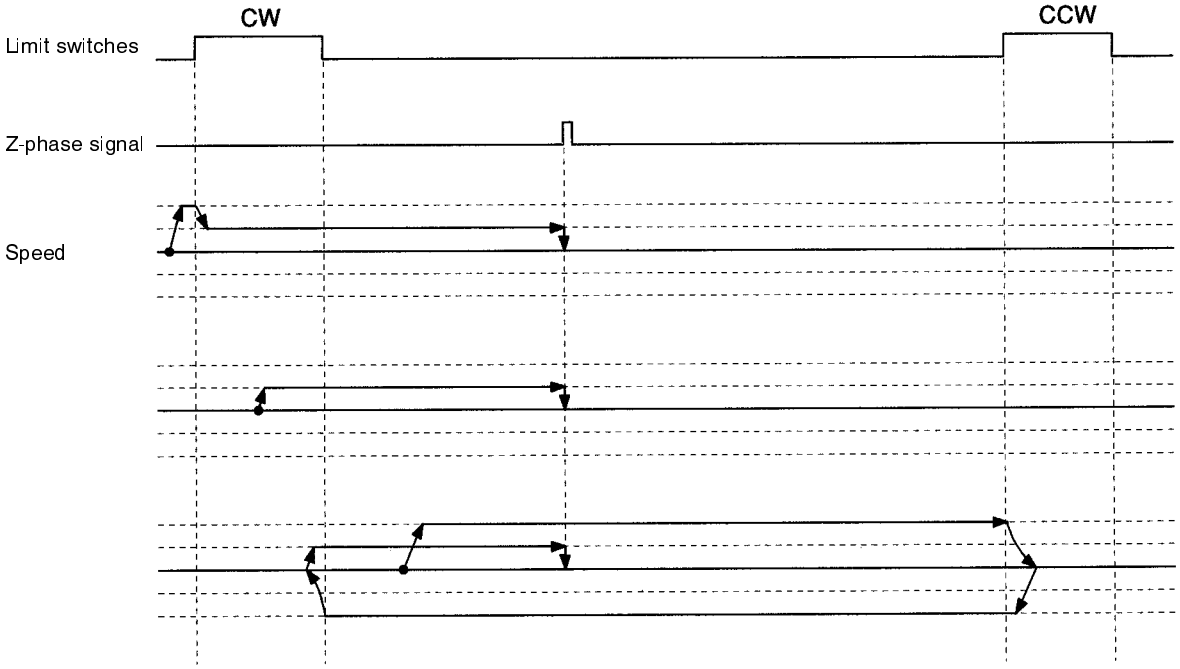
7-4-4 Reverse-mode Origin Searches 3

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW. There is no Z-phase input, so all of these searches result in errors.



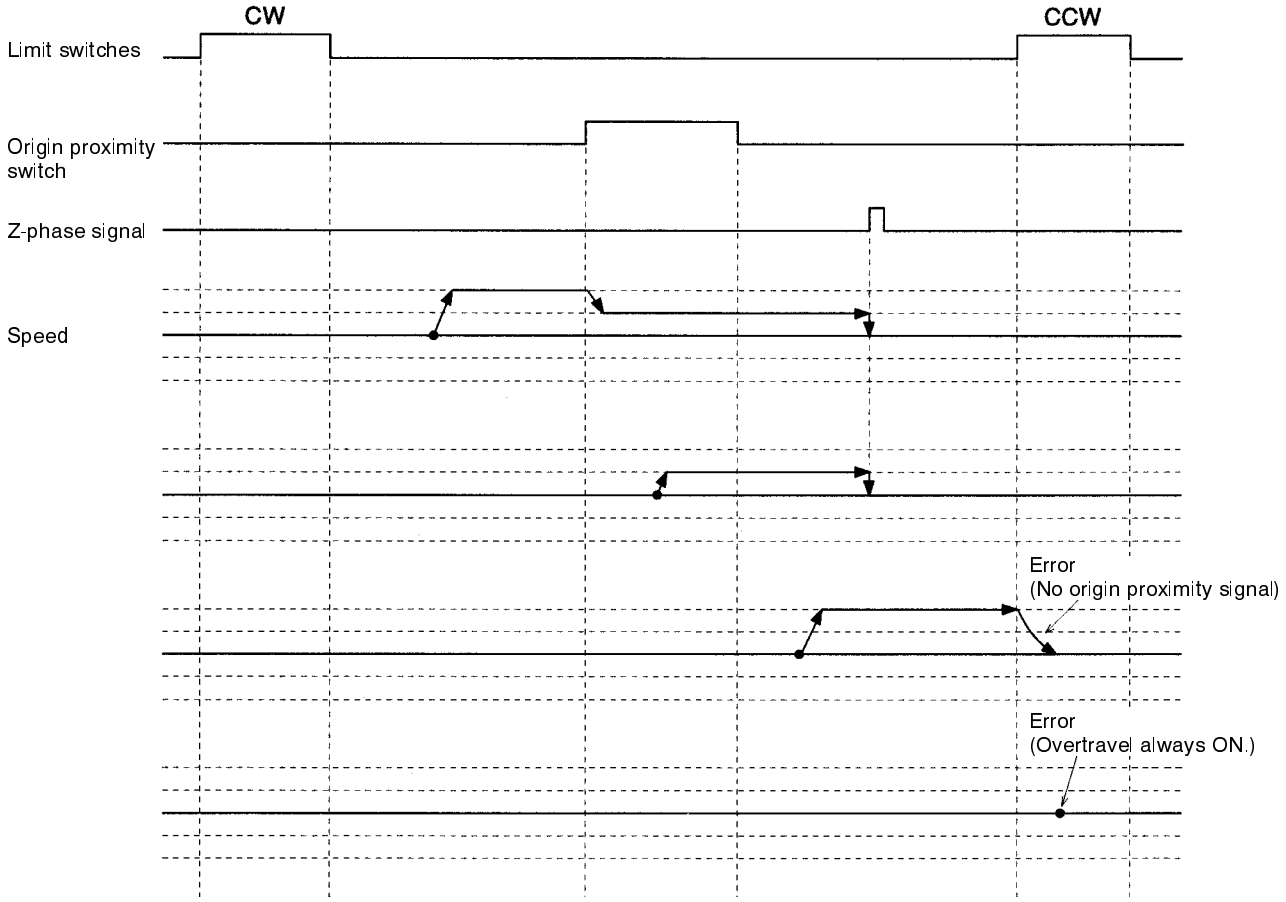
7-4-5 Reverse-mode Origin Searches 4

These origin searches are performed without an origin proximity switch and the initial search direction set to CCW.



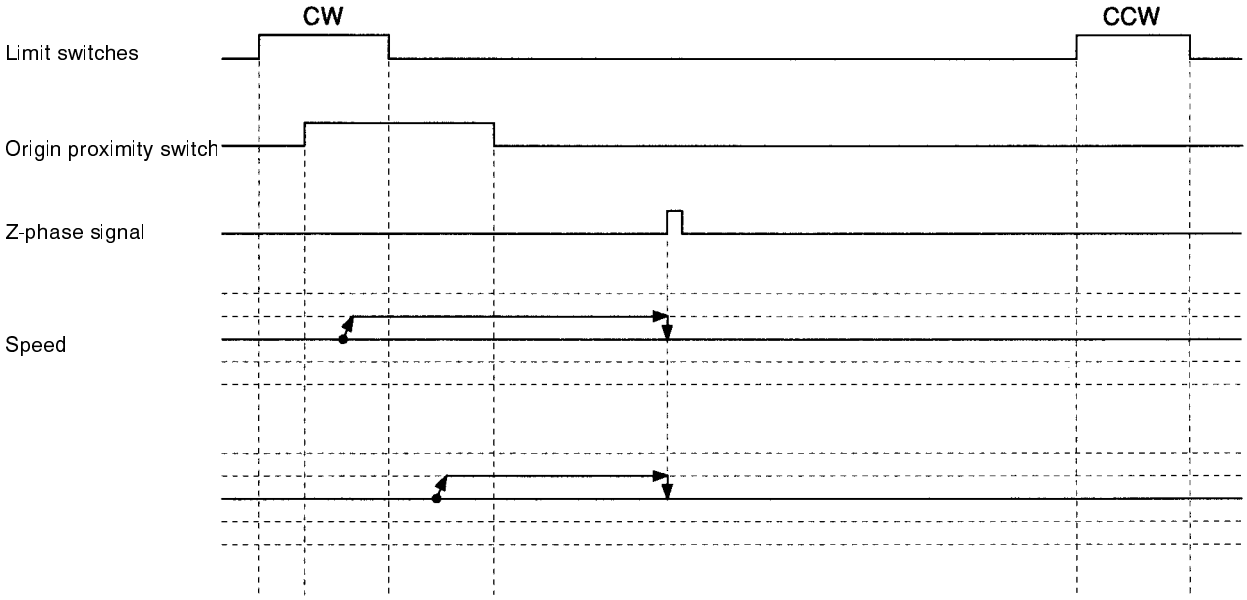
7-4-6 One Direction-mode Origin Searches 1

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW.



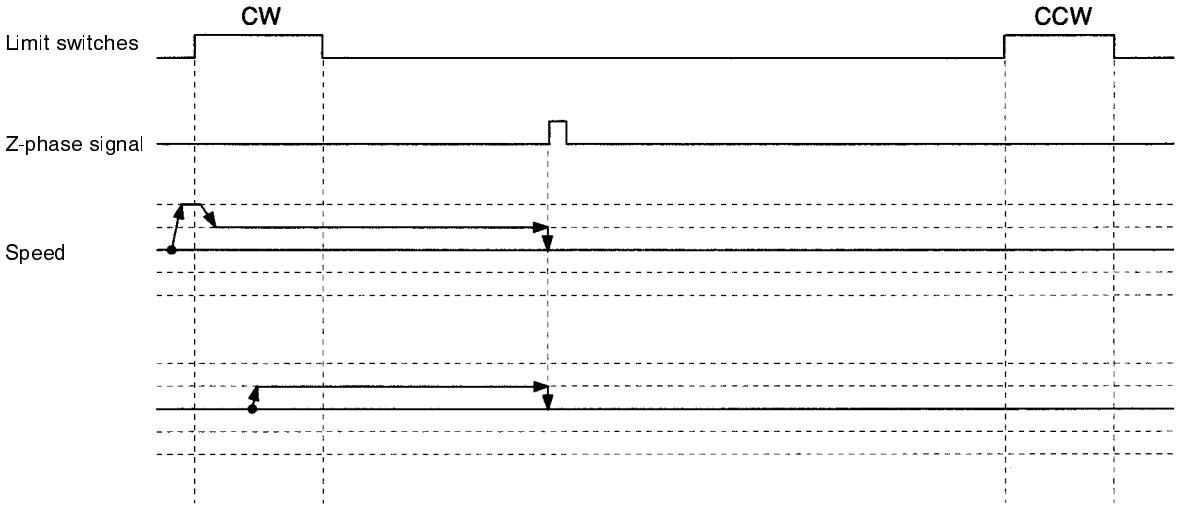
7-4-7 One Direction-mode Origin Searches 2

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW.



7-4-8 One Direction-mode Origin Searches 3

These origin searches are performed without an origin proximity switch and the initial search direction set to CCW.

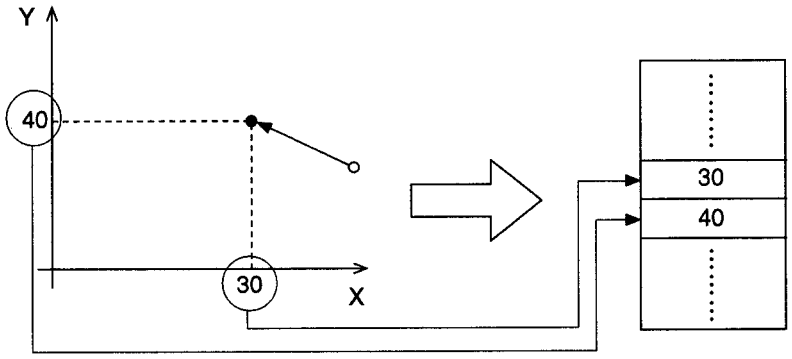


SECTION 8
Teaching

This section describes teaching, which reads each axis' current position as position data.

8-1 Introduction

After moving the axes to a desired position, that position can be read to position data addresses as position data. This process is known as teaching.



Teaching is performed separately in each task. When the teaching command is executed, the current position data for all of the axes in the task (the axes declared in the Unit Parameter Edit menu) will be read and stored in the specified data addresses in order (X, Y, Z, and U).

Position Format

Teaching always reads the current position in the reference coordinate system. The current position can be read in either of two formats.

- The target position
- The feedback value

Refer to *6-6 Interface Bit Specifics* for more details on these formats.

Teaching Method

Teaching can be performed in two ways.

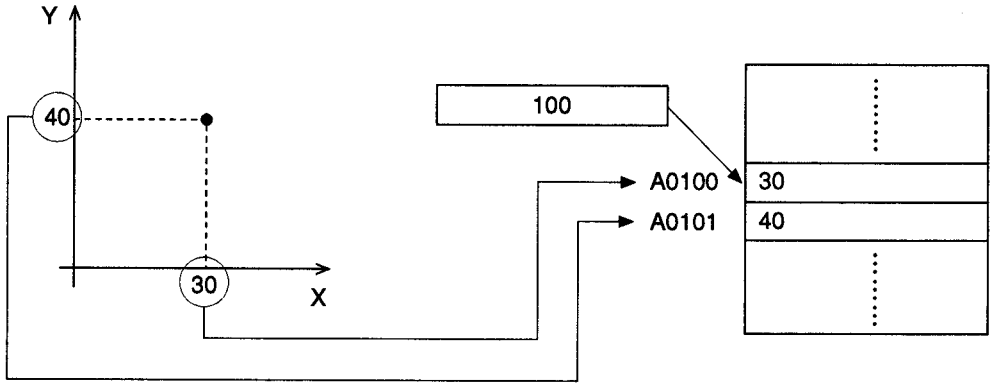
- Teaching can be performed from the ladder program. Execute teaching through the PC data area interface.
- Teaching can be performed from the Teaching Box.

8-2 The Teaching Function

This example describes the teaching function using task 1 with axes X and Y.

Teaching Address

When performing teaching, it is necessary to specify the position data address where the current position data will be stored. The teaching address is used to specify this position data address. The teaching address is initialized to 0 when the power is turned ON.



In the diagram above, the teaching address is 100, which indicates position data address A0100. Since the current position in task 1 is (30, 40), the X-axis' current position of 30 will be stored in position data address A0100 and the Y-axis' current position of 40 will be stored in position data address A0101.

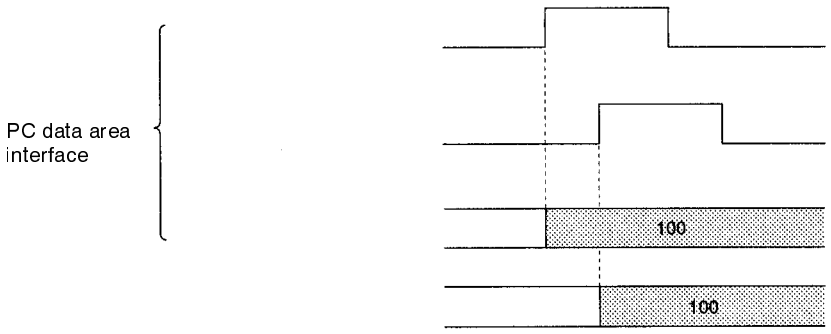
8-3 Setting the Teaching Address

In order to perform teaching with the desired position data address, the teaching address must be set to the corresponding value. This example shows how to set the teaching address from the ladder program.

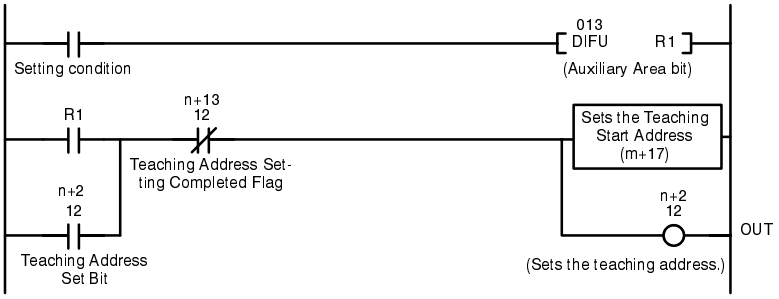
Set the desired value in the Teaching Start Address (DM word m+17). Next, turn ON the Teaching Address Set Bit for task 1 (n+2 bit 12); keep this bit ON until the Teaching Address Setting Completed Flag (n+13 bit 12) is turned ON.

The timing chart and ladder program for this operation are shown below. Refer to *6-6 Interface Bit Specifics* for more details.

Timing Chart



Example Program



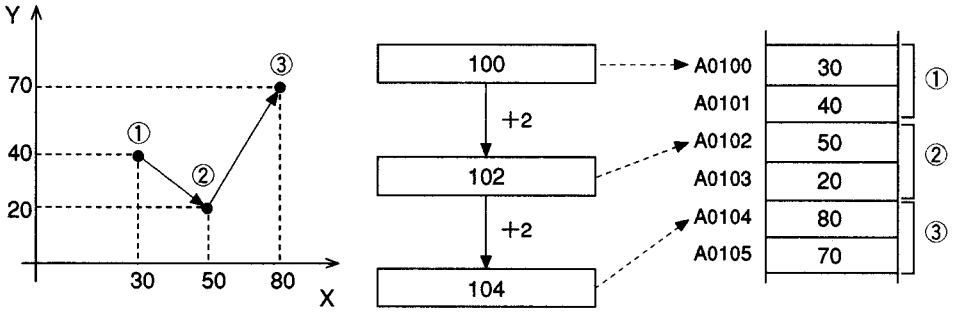
Using the Teaching Box

Press "TEACH" when performing teaching from the Teaching Box. The teaching address can be changed by using the Numeric Keys to change the position data address displayed in the lower-left corner of the screen. Refer to the *Teaching Box Operation Manual* for details.

8-4 Performing Teaching

Teaching can be performed once the teaching address has been set. When teaching is performed, the X-axis' current position of 30 is stored in position data address A0100 and the Y-axis' current position of 40 is stored in position data address A0101.

The teaching address is then automatically incremented by 2, to 102. Since the teaching address is incremented automatically, the position data can be stored sequentially without resetting the teaching address.



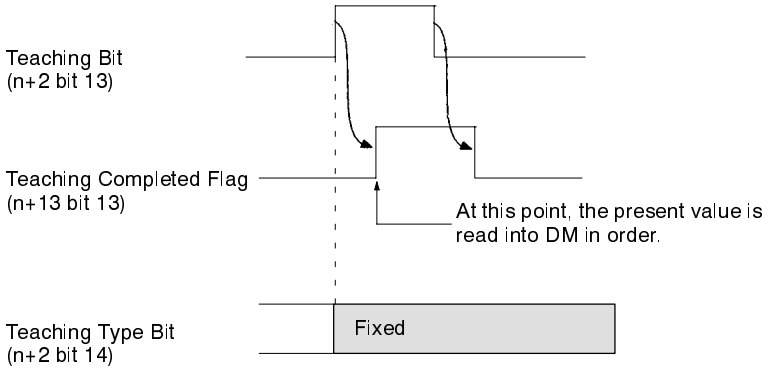
In the example above, teaching is performed three times at points 1, 2, and 3. The X, Y current position data is stored and the teaching address is automatically incremented to the next open position data address.

Performing Teaching from the Ladder Program

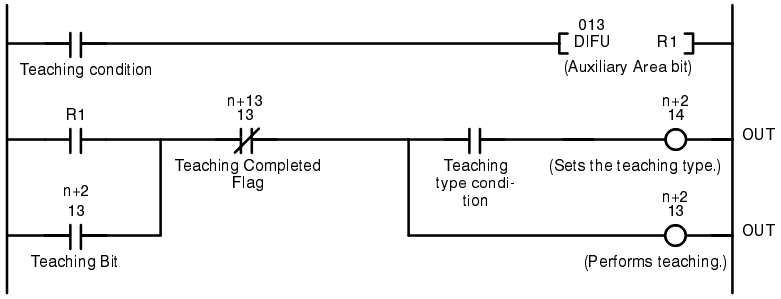
Turn ON the Teaching Bit for task 1 (n+2 bit 13) and keep this bit ON until the Teaching Completed Flag (n+13 bit 13) is turned ON.

The timing chart and ladder program for this operation are shown below. Refer to *6-6 Interface Bit Specifics* for more details.

Timing Chart



Example Program



Using the Teaching Box

After pressing “TEACH” to set the position data address, press “WRITE” and “YES.” Refer to the *Teaching Box Operation Manual* for details.

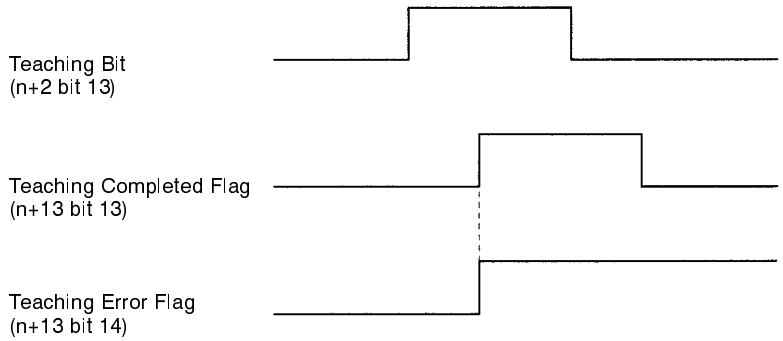
8-5 Errors

8-5-1 Teaching Errors

An error will occur and the Teaching Error Flag (n+13 bit 14) will be turned ON in the following cases when performing teaching.

- Teaching is executed, but the origin hasn’t been fixed. If even one of the axes in the task hasn’t been fixed, the current position data for all of the axes won’t be stored in the position data addresses and the teaching address won’t be incremented.
- The teaching address isn’t within the acceptable range (0 to 1999). The teaching address won’t be incremented if the value is incorrect.

Timing Chart



Refer to *6-6 Interface Bit Specifics* for more details.

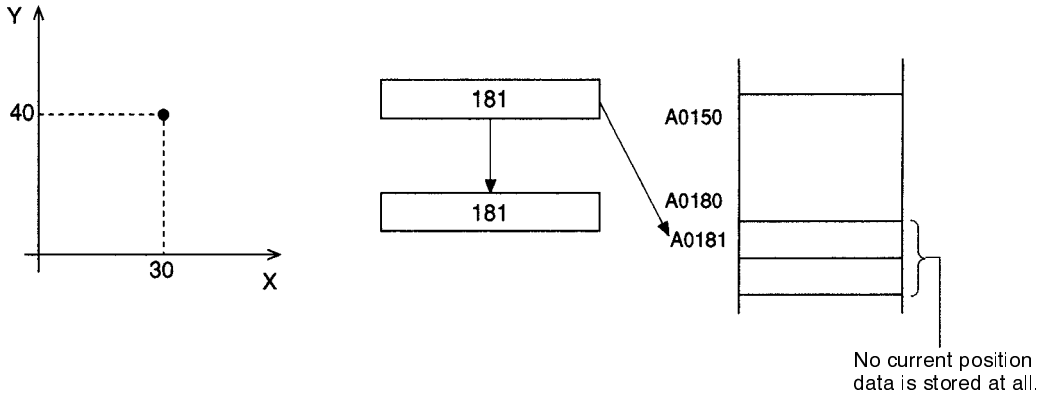
8-5-2 Teaching Address Over Errors

A “teaching address over” error will occur when teaching is performed with a teaching address that isn’t within the position data address range set in the Memory Parameter Edit menu (a submenu of the MC Parameter Edit menu). If a “teaching address over” error occurs, the current position won’t be stored in the position data addresses and the teaching address won’t be incremented.

In the following examples the start address for task 1 has been set to 150 and the end address has been set to 180, so a “teaching address over” error will occur when teaching is performed with a teaching address less than 150 or greater than 180.

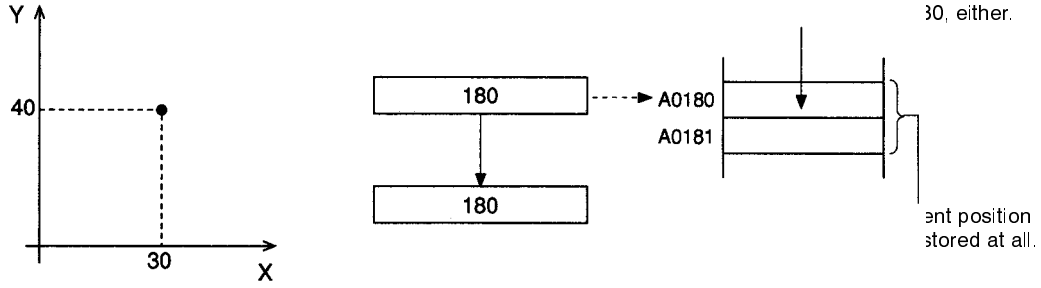
Example 1

In this example, the teaching address is already greater than 180 when teaching is performed.



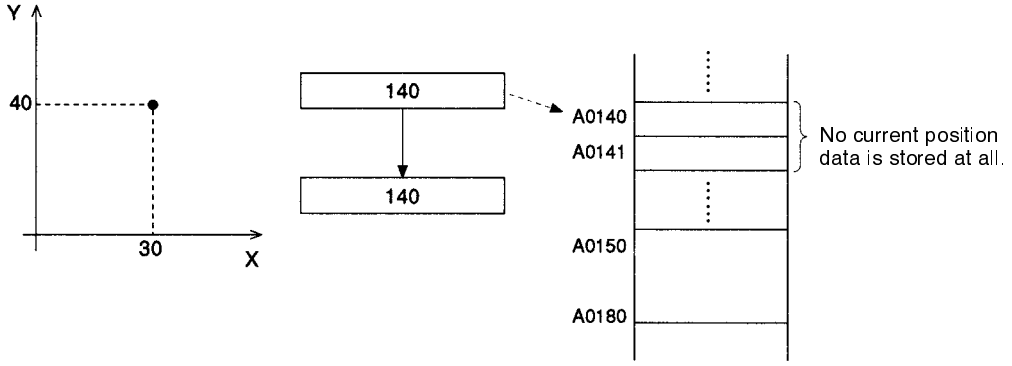
Example 2

In this example, the teaching address isn’t greater than 180 when teaching is performed, but it is greater than 180 when the data is to be stored.



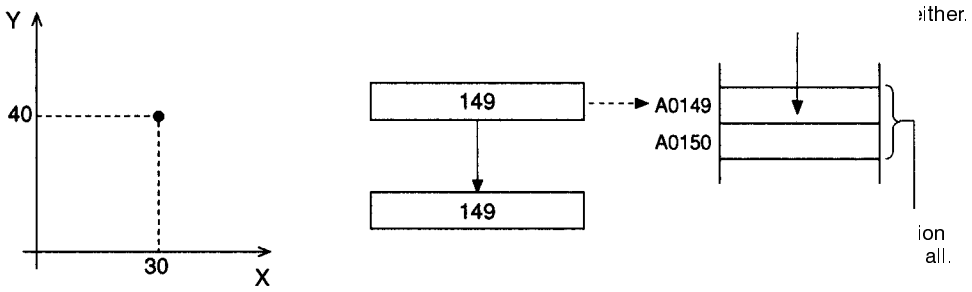
Example 3

In this example, the teaching address is less than 150 when teaching is performed.

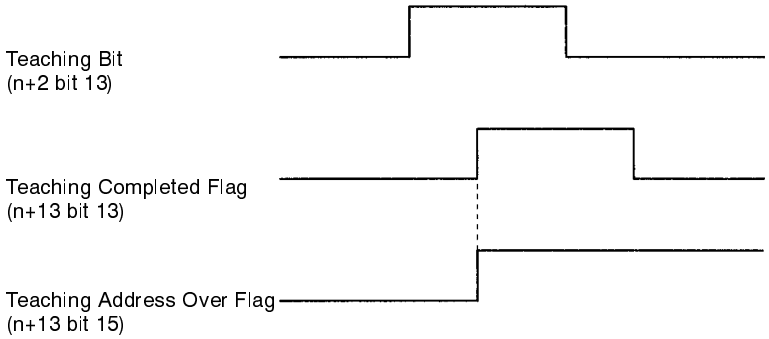


Example 4

In this example, the teaching address is just one less than the start address (150) when teaching is performed.



Timing Chart



SECTION 9
Sample Programs

This section provides sample motion control programs written in G language. Refer to *9-11 Executing MC Programs from the Ladder Program* for details on executing these programs from the PC's ladder program.

9-1 Positioning with PTP Control

This program performs the following 3 positioning operations and then returns to the reference origin. An M code is output when positioning is completed.

1, 2, 3...

1. X100 Y50 Z20 (Output M code 20.)
2. X250 Y150 Z−100
3. X300 Y200 Z−200 (Output M code 700.)
4. Return to reference origin.

Sample Program

The following sample program performs the positioning operations listed above.

```
N000  P001  XYZ
N001  G00   X100  Y50   Z20   M20
N002  G00   X250  Y150  Z-100
N003  G04   2
N004  G00   X300  Y200  Z-200 M700
N005  G26   XYZ
N006  G79
```

Explanation

Block N000 declares the program number and axes being used.

N001
Positions the axes to (X100, Y50, Z20) by PTP control. Absolute positioning (default) is used, because nothing is specified.

When positioning is completed, M code 20 is output and the Unit waits for a reset (M code reset standby).

N002
When the M code reset is received, block N002 positions the axes to (X250, Y150, Z−100) by PTP control.

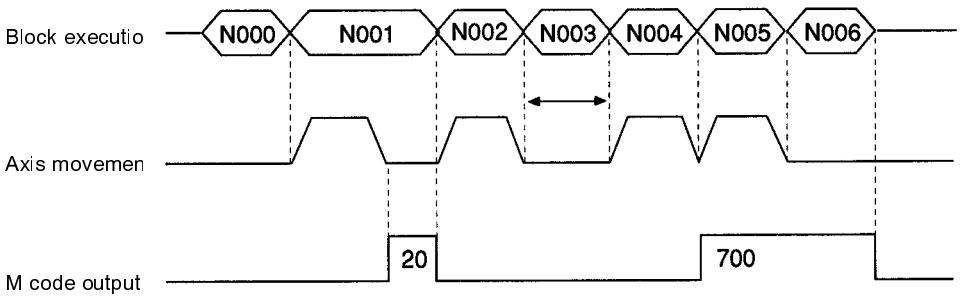
N003
Waits for 2 seconds.

N004
Positions the axes to (X300, Y200, Z−200) by PTP control. When positioning is completed, M code 700 is output and the next block is executed. There is no M code reset standby with M codes 500 through 999.

N005
Returns the X, Y, and Z axes to the reference origin.

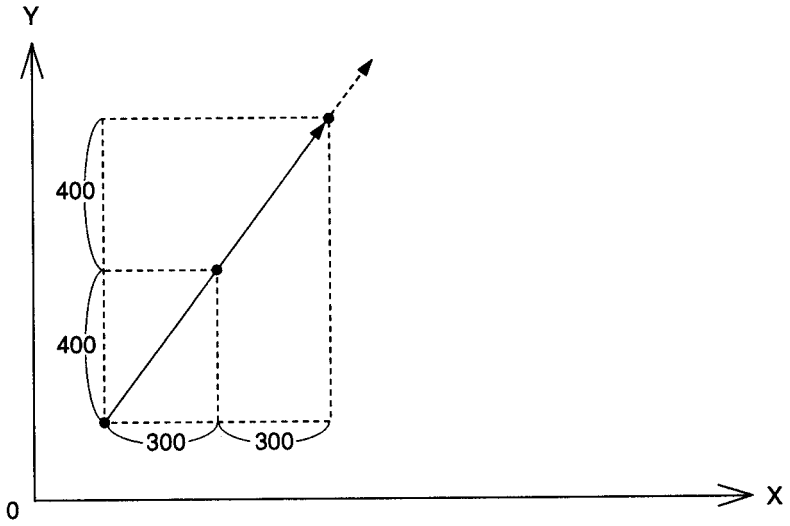
N006
Ends the program. When M code 700 is being output, it is cleared by the PROGRAM END function.

Timing Chart



9-2 Positioning with Linear Interpolation

When the optional input goes ON, this program uses linear interpolation to move the X-axis by 300 and the Y-axis by 400 from the current position. This positioning operation will be repeated (up to 21 times) until position data address A1000 contains 1.



Sample Program

```
N000  P002  XY
N001  G91
N002  G01   X300  Y400  F50   #16
N003  G71   N005/A1000=1
N004  G70   N002/L20
N005  G79
```

Explanation

Block N000 declares the program number and axes being used.

N001
Specifies incremental positioning.

N002
Waits until optional input 16 (general input 1) goes ON. When it goes ON, the X-axis is moved by 300 and the Y-axis by 400 with linear interpolation at a speed of 50.

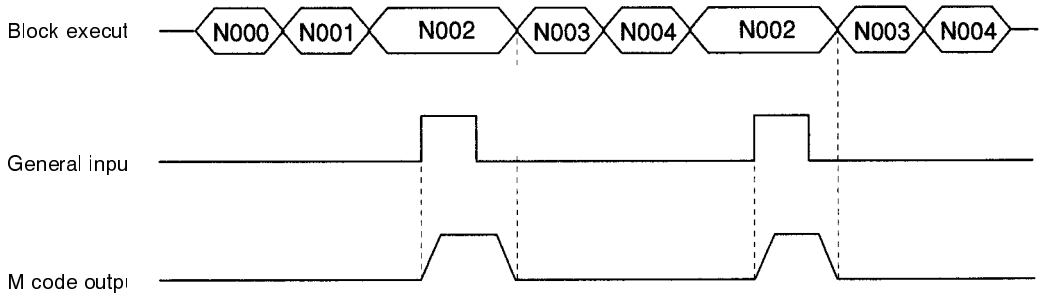
N003
Checks the content of A1000 and ends the program if it is 1.

N004
Jumps to block N002. Block N004 will jump to N002 20 times (performing 21 positioning operations).

N005
Ends the program.

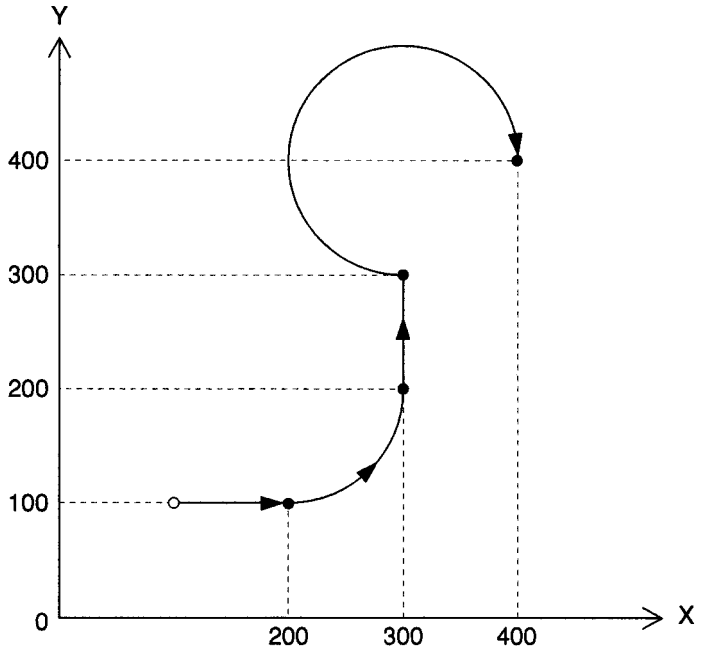
Note The operations above are performed in Stop mode not Pass mode because an optional input is specified.

Timing Chart



9-3 Positioning with Linear and Circular Interpolation

This program uses linear and circular interpolation to move the X and Y axes in the pattern shown in the following diagram.



Sample Program

```
N000  P003  XY
N001  G17
N002  G01   X200  Y100  F50
N003  G03   X300  Y200  R100
N004  G01   Y300
N005  G02   X400  Y400  R-100
N006  G79
```

Explanation

Block N000 declares the program number and axes being used.

N001
Sets the X-Y plane as the plane for circular interpolation.

N002
Moves the axes to (X200, Y100) by linear interpolation.

N003
Moves the axes to (X300, Y200) by circular interpolation in the counter-clockwise direction with a radius of 100. A positive value is specified for the radius, so an arc smaller than a semicircle (1/4 circle) is drawn.

N004
Moves the axes to (X300, Y300) by linear interpolation.

N005
Moves the axes to (X400, Y400) by circular interpolation in the clockwise direction with a radius of 100. A negative value is specified for the radius, so an arc greater than a semicircle (3/4 circle) is drawn.

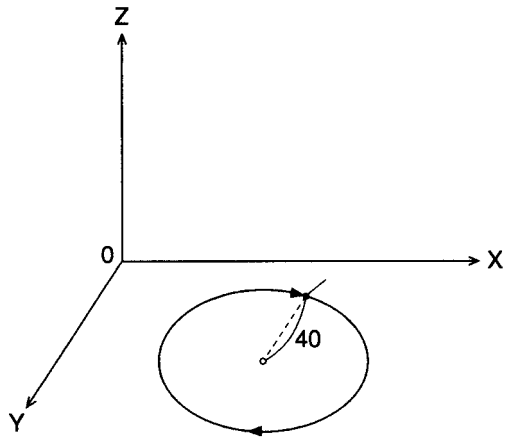
N006
Ends the program.

Note This series of operations is performed in Pass mode.

9-4 Positioning with Circular and Helical Interpolation

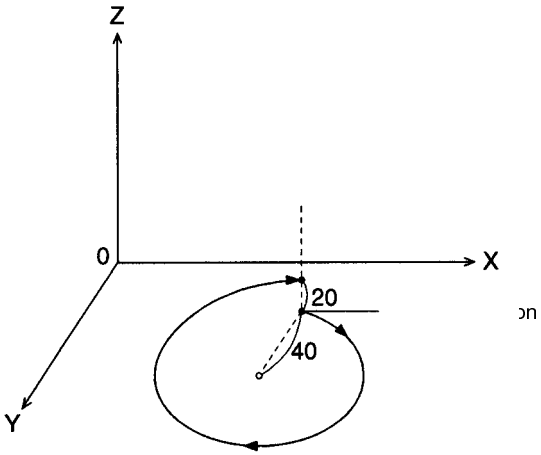
This program positions the axes with circular interpolation and then helical interpolation, as shown in the following diagrams.

Circular Interpolation



This positioning operation uses circular interpolation to draw a counter-clockwise circle with a radius of 40 around a center point that is Y40 from the current position.

Helical Interpolation



This operation performs the same circular interpolation described previously, while moving 20 in the Z-direction at the same time.

Sample Program

```
N000  P004  XYZ
N001  G91
N002  G11
N003  G17
N004  G03   X0    Y0    I0    J40   F500
N005  G03   X0    Y0    Z20   I0    J40
N006  G79
```

Explanation

Block N000 declares the program number and axes being used.

N001
Specifies incremental positioning.

N002
Specifies Stop mode.

N003
Sets the X-Y plane as the plane for circular interpolation.

N004
A counter-clockwise circle with a radius of 40 is drawn around a point (X0, Y40) from the current position, returning the axes to the original current position. The tangential speed is set to 500.

N005
Helical interpolation is performed by moving the Z axis 20 from its current position while a counter-clockwise circle with a radius of 40 is drawn in the X-Y plane around a point (X0, Y40) from the current position.

N006
Ends the program.

9-5 Indirect Addressing with Registers

This program uses registers to indirectly specify position data stored in position data addresses and uses this data in positioning operations.

Sample Program

```
N000  P005  XYZU
N001  G11
N002  G63   E1=100
N003  G63   E2=101
N004  G63   E3=102
N005  G63   E4=103
N006  G01   X(E1) Y(E2) Z(E3) U(E4) F100
N007  G60   E1=E1+4
N008  G60   E2=E2+4
N009  G60   E3=E3+4
N010  G60   E4=E4+4
N011  G75   3
N012  G70   N006/L3
N013  G79
```

Explanation

Block N000 declares the program number and axes being used.

N001
Specifies Stop mode.

N002 through N005
Set the initial position data addresses in the registers.

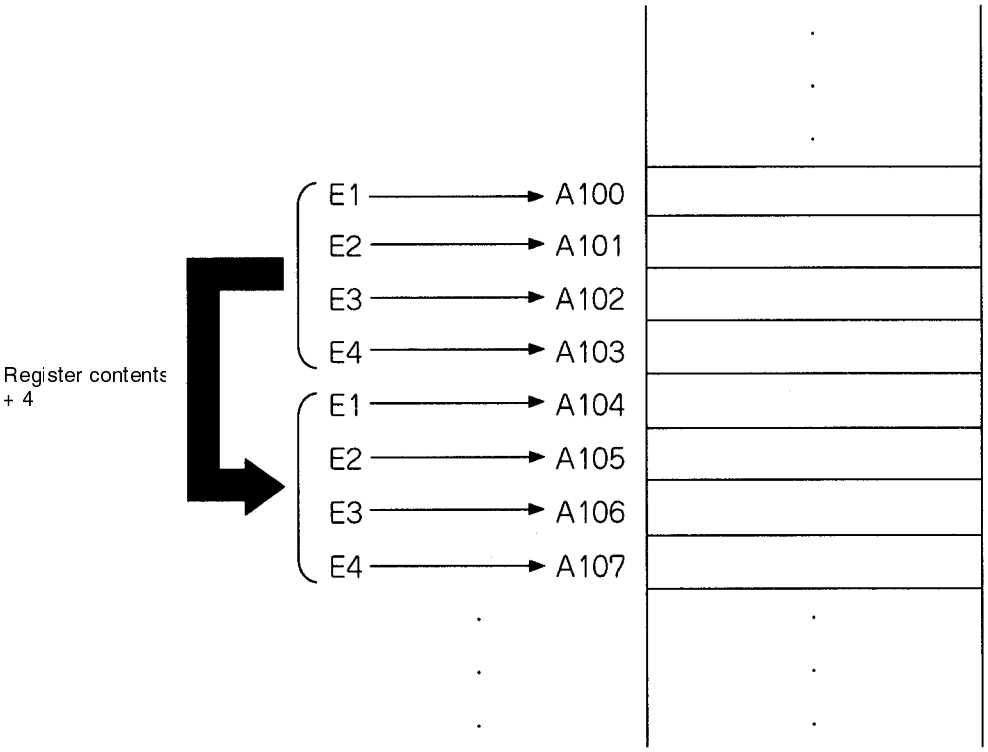
N006
Reads the position data from the addresses specified in the registers and moves the axes to this point by linear interpolation.

N007 through N010
Increment the contents of the registers by 4. (Specifying the next group of four position data addresses.)

N011
If optional input 3 is ON, the next block will be skipped and the program will end.

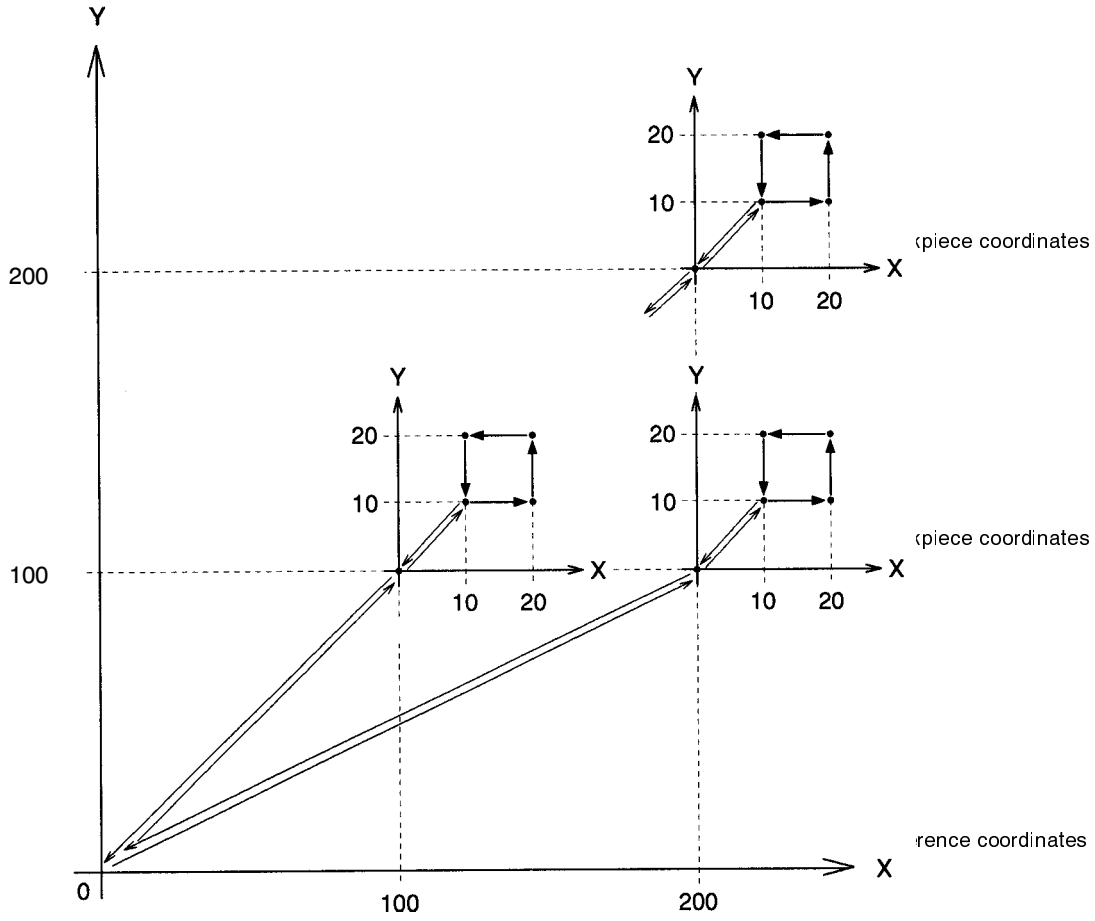
N012
Jumps to block N006 and repeats the positioning operation 3 times.

N013
Ends the program.



9-6 Using the Workpiece Origin Offset

A positioning pattern can be performed any number of times by repeatedly changing the workpiece origin offset and calling the subprogram. This method is convenient when using the workpiece coordinate system, especially with absolute positioning.



Sample Program

```
N000  P006  XY
N001  G53   X100  Y100
N002  G72   P800
N003  G53   X200
N004  G72   P800
N005  G53   Y200
N006  G72   P800
N007  G79
```

Sample Subprogram

```
N000  P800  XY
N001  G27   XY    M10
N002  G01   X10   Y10   F10
N003  G01   X20
N004  G01   Y20
N005  G01   X10
N006  G01   Y10
N007  G27   XY    M20
N008  G26   XY    M30
N009  G73
```

Explanation

Block N000 declares the program number and axes being used.

P006: N001
Sets the workpiece origin offset to (X100, Y100).

P006: N002
Calls subprogram P800.

P800: N001
Returns to the workpiece origin. The workpiece coordinate system is selected at this time. M code M10 is output and the Unit waits for the M code reset.

P800: N002 through N006
The subprogram's series of absolute positioning operations is performed by linear interpolation in Pass mode.

P800: N007
Returns to the workpiece origin. M code M20 is output and the Unit waits for the M code reset.

P800: N008
When the reset is received, block N008 returns to the reference origin. M code M30 is output and the Unit waits for the M code reset.

P800: N009
When the reset is received, block N009 returns to main program P006.

P006: N003
Changes the workpiece origin offset to X200.

P006: N004
Calls subprogram P800 and repeats the same series of operations.

9-7 Changing the Interpolation Acceleration Time

This program changes the setting of the interpolation acceleration time parameter and performs linear interpolation.

Sample Program

```
N000  P007  XY
N001  G69   #3/X500
N002  G01   X1000 Y500  F300
N003  G01   X2000 Y-1000
N004  M20
N005  G69   #3/X100
N006  G01   X3000 Y1000
N007  G01   X4000 Y1500
N008  G79
```

Explanation

Block N000 declares the program number and axes being used.

N001
Changes the setting of the X-axis' interpolation acceleration time parameter to 500 ms, which becomes the X-axis acceleration time used for interpolation in this task.

N002
Moves to (X1000, Y500) by linear interpolation with an acceleration time of 500 ms and speed of 300.

N003
Moves to (X2000, Y−1000) in Pass mode with an acceleration time of 500 ms.

N004
Once the pass operation is completed, N004 outputs M code M20 and waits for the M code reset.

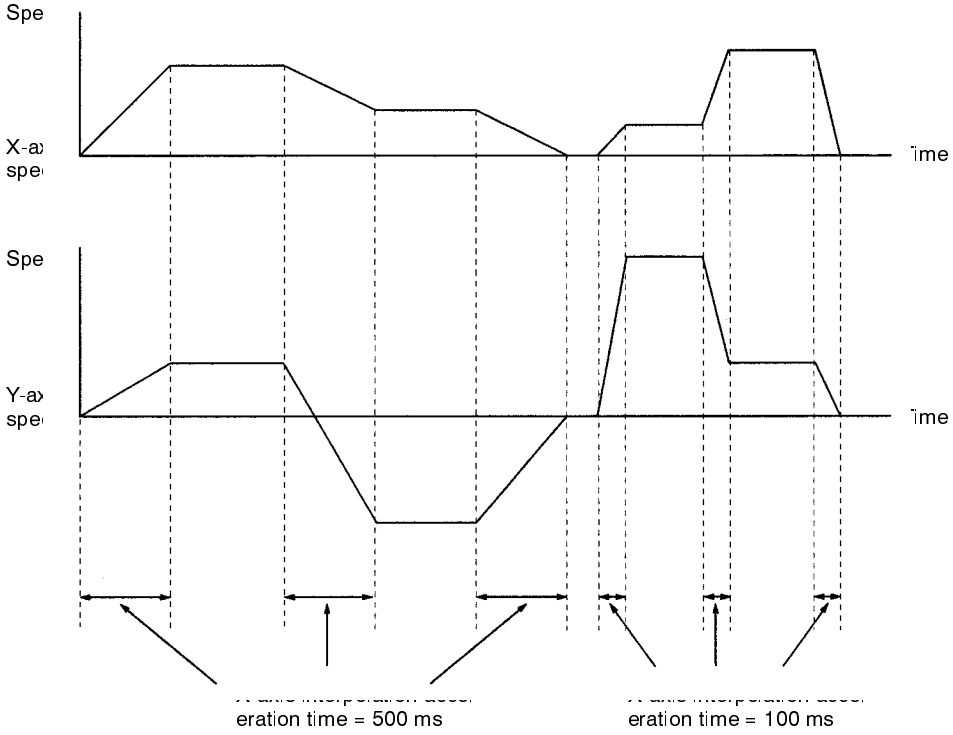
N005
Changes the setting of the interpolation acceleration time to 100 ms.

N006
Moves to (X3000, Y1000) by linear interpolation with an acceleration time of 100 ms.

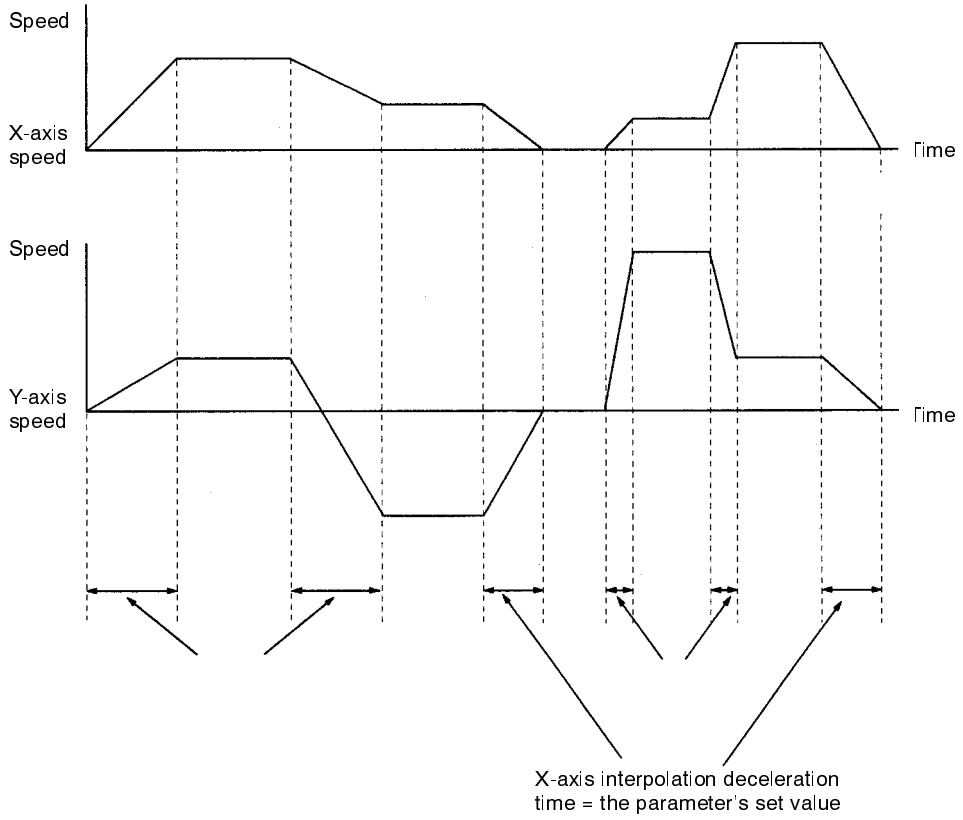
N007
Moves to (X4000, Y1500) in Pass mode with an acceleration time of 100 ms.

N008
Ends the program.

Timing Charts

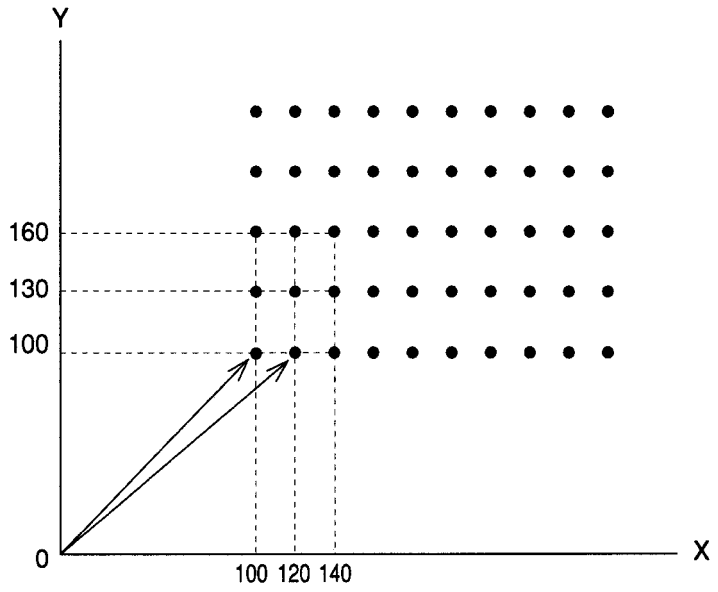


The following operations would be performed if one G11 (STOP MODE) function were inserted between blocks N002 and N003, another between blocks N006 and N007, and a G10 (PASS MODE) function were inserted between blocks N005 and N006.



9-8 Calculating Position Data

After initializing the position data, this program moves the axes 10 times while incrementing the X-axis' position data by 20. Next, the Y-axis' position data is incremented by 30 and the X-axis positioning operation is repeated. In all the process is performed 5 times, as shown in the following diagram.



Sample Program 1

```
N000  P008  XY
N001  G11
N002  G63   E00=0
N003  G63   A1000=100
N004  G63   A1001=100
N005  G01   XA1000 YA1001 F20 M10
N006  G01   X0    Y0    F50
N007  G60   A1000=A1000+20
N008  G70   N005/L10
N009  G63   A1000=100
N010  G60   A1001=A1001+30
N011  G60   E00=E00+1
N012  G71   N005/E0 ! 6
N013  G79
```

Explanation

Block N000 declares the program number and axes being used.

N001
Specifies Stop mode.

N002
Initializes register E0 to 0.

N003
Substitutes the X-axis' initial position data into address A1000.

N004
Substitutes the Y-axis' initial position data into address A1001.

N005
Moves the axes by linear interpolation to the X position indicated in A1000 and the Y position indicated in A1001.

N006
Returns to the origin.

N007
Increments the X-axis' position data by 20.

N008
Jumps to block N004 and repeats the above process 10 times.

N009
Initializes the X-axis' position data to its original value.

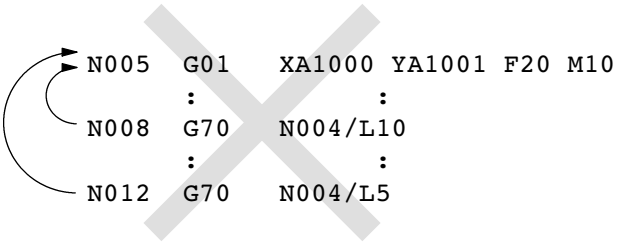
N010
Increments the Y-axis' position data by 30.

N011
Uses register E0 as a loop counter with an initial value of 0. The content of E0 is incremented by 1.

N012
Jumps to N004 as long as the content of E0 isn't 6. When E0=6, N012 proceeds to the next block and ends the program.

N013
Ends the program.

Note Register E0 is used as a loop counter in block N012 because loops can't be nested as shown below.



The loops in the example above won't operate properly. A subprogram can also be used for nesting, as shown in the following sample program.

Sample Program 2

```
N000  P008  XY
N001  G11
N002  G63   A1000=100
N003  G63   A1001=100
N004  G72   P700
N005  G63   A1000=100
N006  G60   A1001=A1001+30
N007  G70   N004/L5
N008  G79
```

Subprogram

```
N000  P700  XY
N001  G01   XA1000 YA1001 F20 M10
N002  G01   X0    Y0    F50
N003  G60   A1000=A1000+20
N004  G70   N001/L10
N005  G73
```

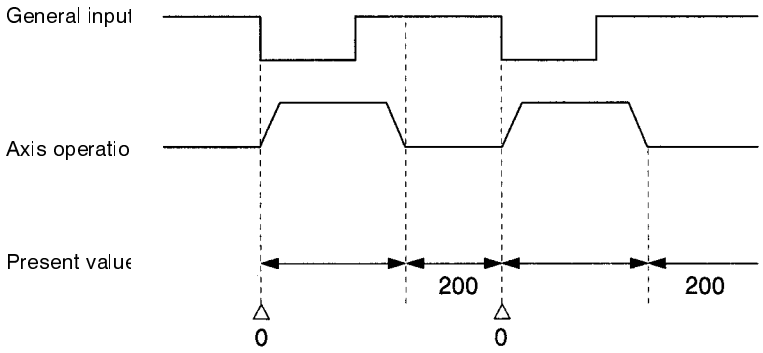
Explanation

Blocks N004 through N007 in sample program 1 have been converted to subprogram format in subprogram P700, which is called from step N004 in main program P008. Main program P008 and subprogram P700 have independent loops, so they will operate properly.

Function G73 (SUBPROGRAM END) must be included at the end of subprogram P700.

9-9 Stopping a Program with a General Input

This program shows how to stop MC program execution when general input 1 goes ON.



Sample Program

```
N000  P009  X
N001  G11
N002  G76   16
N003  G54   X=0
N004  G01   X200  F100
N005  G70   N001
N006  G79
```

Explanation

Block N000 declares the program number and axes being used.

N001
Specifies Stop mode.

N002
Stops execution when general input 1 goes ON. (Proceeds to the next block when general input 1 is OFF.)

N003
Presets the X-axis' present value to 0.

N004
Absolute positioning of the X axis.

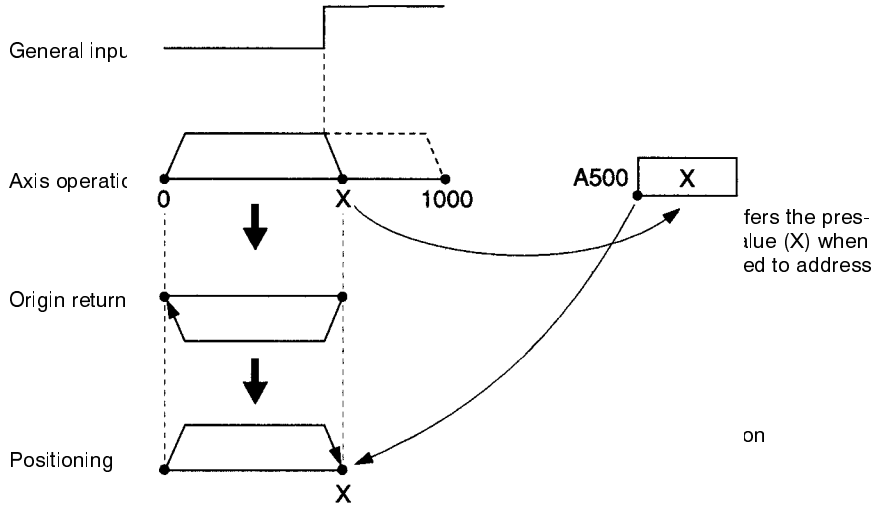
N005
Unconditionally jumps to block N001, repeating blocks N001 through N004. The process can be repeated indefinitely without an overflow because the present value is preset to 0 each time.

N006
Ends the program.

9-10 Stopping the Program and Substituting Position Data

This program moves the X axis to X1000 at a speed of 100 by linear interpolation. The movement will be decelerated to a stop by function G74 (OPTIONAL END) if general input 2 goes ON before the positioning operation is completed.

The X position when the movement was stopped will be stored in address A500 and that position data will be used for later positioning operations. This process is useful for applications in which the position where the operation was stopped will be used for positioning rather than the original target position.



Sample Program

```
N000  P010  X
N001  G11
N002  G74   17
N003  G01   X1000 F100
N004  G63   A500=X
N005  G01   X0
N006  G01   XA500
N007  G70   N005
N008  G79
```

Explanation

Block N000 declares the program number and axes being used.

N001
Specifies Stop mode.

N002
Execution of the next block is stopped when general input 2 goes ON.

N003
Moves the X axis with a target position of X1000. The positioning operation will be decelerated to a stop if general input 2 goes ON before positioning is completed.

N004
The stopping position is stored in address A500.

N005
Returns to the origin.

N006
Positions the X axis using the position data stored in address A500 in block N004.

N007
Jumps to N005 and repeats the positioning operation.

N008
Ends the program.

9-11 Executing MC Programs from the Ladder Program

This section describes ladder programs that will execute G-language MC programs. Use the following procedure to execute MC programs.

1, 2, 3...

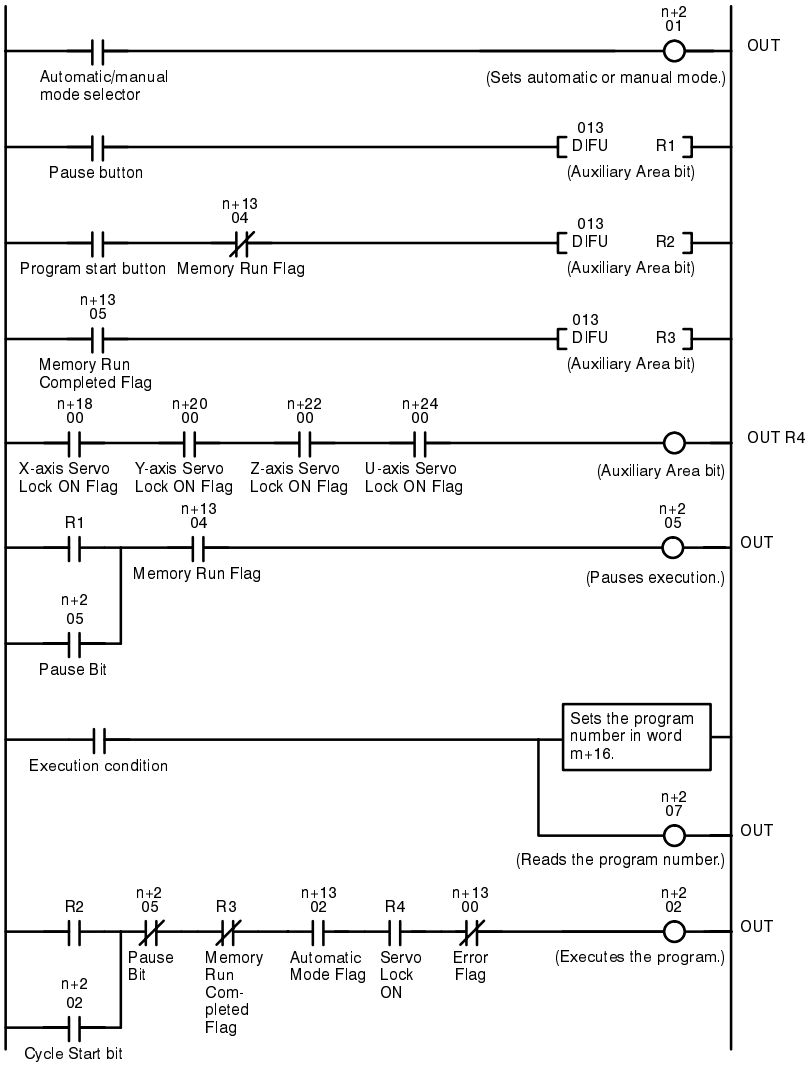
1. Set the MC Unit to automatic mode. (Turn ON bit 01 of CIO word n+2.)
2. Set the program number of the desired MC program. (DM word m+16)
3. Read the program number from DM word m+16 to the MC Unit. (Turn ON bit 07 of CIO word n+2.)
4. Turn ON the Cycle Start Bit to execute the MC program specified in DM word m+16. (Turn ON bit 02 of CIO word n+2.)

This procedure executes task 1. The following table shows the equivalent CIO Area control bits and DM words for tasks 2, 3, and 4.

Bit/word	Task 1	Task 2	Task 3	Task 4
Automatic/Manual Mode Bit	n+2: bit 01	n+3: bit 01	n+4: bit 01	n+5: bit 01
Program number	DM m+16	DM m+19	DM m+22	DM m+25
Program Number Read Bit	n+2: bit 07	n+3: bit 07	n+4: bit 07	n+5: bit 07
Cycle Start Bit	n+2: bit 02	n+3: bit 02	n+4: bit 02	n+5: bit 02

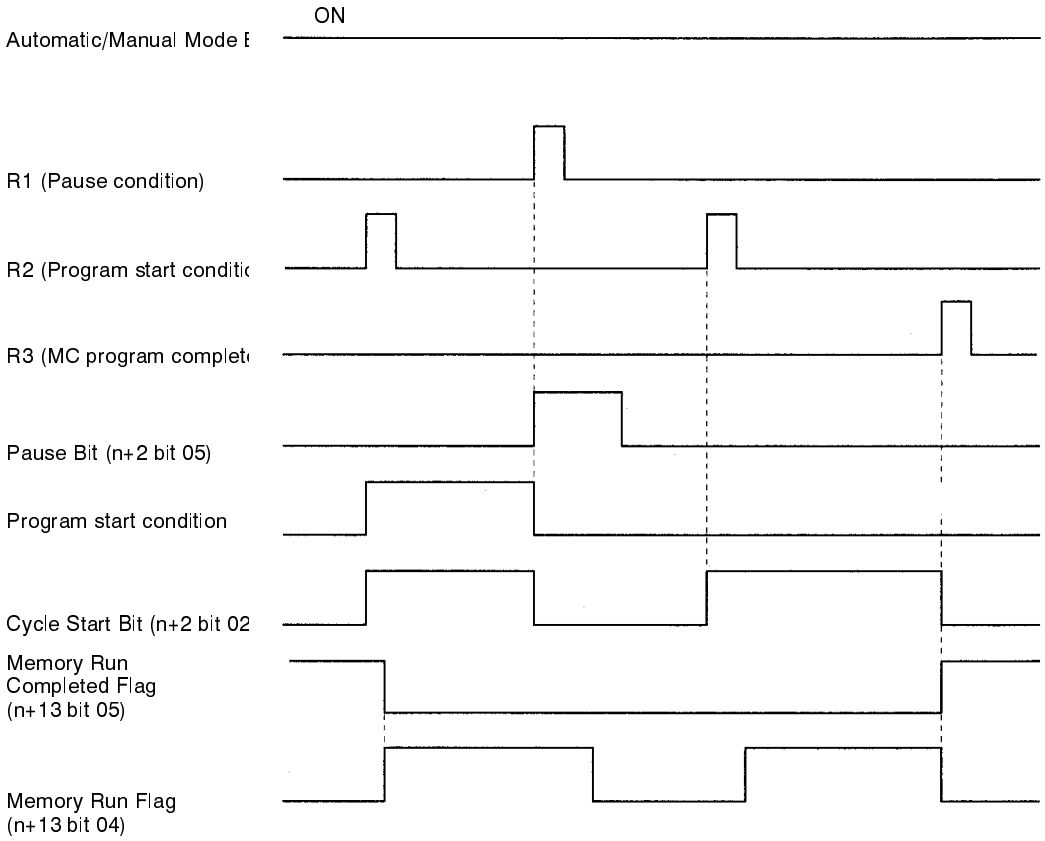
Sample Ladder Program

The following ladder program executes an MC program. (Task 1)



Note When executing a positioning operation in the MC program, all of the axes being used must be servo-locked (Servo Lock ON flags ON), so be sure to add a condition like R4 in the sample program above.

Timing Chart



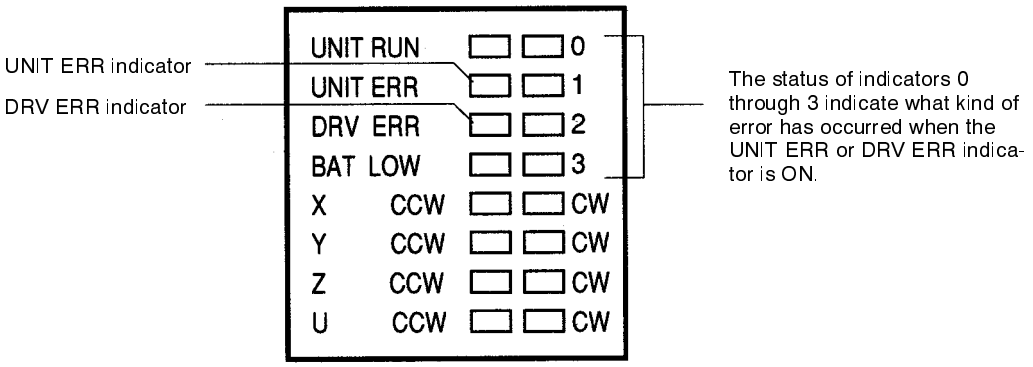
SECTION 10
Troubleshooting

This section describes the errors that might occur during operation, their probable causes, and possible remedies.

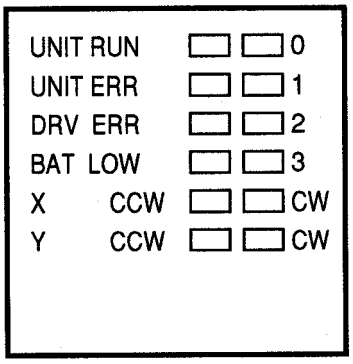
10-1 Error Indicators

The error indicators are located on the front of MC Unit, as shown in the following diagram.

CV500-MC421



CV500-MC221



One or more of indicators 0 through 3 might go ON at the same time as the UNIT ERR or DRV ERR indicator.

UNIT ERR Indicator The table below shows the errors indicated by different combinations of the UNIT ERR indicator and indicators 0 through 3.

DRV ERR Indicator When the DRV ERR indicator goes ON, indicators 0 through 3 correspond to axes X through U.

Indicator Status							Error	Error Processing
UNIT RUN	**UNIT ERR**	**DRV ERR**	**0**	**1**	**2**	**3**		
OFF	OFF	OFF	OFF	OFF	OFF	OFF	Watchdog timer timeout error	The MC Unit's watchdog timer has timed out. Replace the MC Unit.
ON	OFF	OFF	OFF	OFF	OFF	OFF	No error	---
ON	ON	OFF	OFF	OFF	OFF	OFF	MC Unit error	An error has occurred in the MC Unit itself. The servodriver is OK. Details on the error can be determined by checking the error code output to the Teaching Box or PC data area interface. (See note.)
OFF			OFF	OFF	OFF	ON	Unit number duplication error	The same unit number has been set on another Unit. Check the unit number and make sure that it is unique.
			OFF	OFF	ON	OFF	Unit number error	The unit number has been set too high (16 to 99). Check the unit number and make sure that it is 00 to 15.
			OFF	OFF	ON	ON	I/O table not registered	Register the I/O table.
			OFF	ON	OFF	OFF	Other CPU error	Can't communicate with the PC. Change the unit number temporarily and then change it back again. If this doesn't solve the problem, the MC Unit or PC may be faulty.
			OFF	ON	OFF	ON	System file read error	
			OFF	ON	ON	OFF	System file setting error	
			ON	OFF	OFF	OFF	PC watchdog timer error	An error has occurred in the PC. Check the PC.
			ON	OFF	OFF	ON	EEPROM error	An error has occurred in EEPROM, the MC Unit, or in the MC Unit's RAM. Replace the MC Unit.
			ON	ON	ON	ON	RAM error	
ON or OFF			ON	OFF	ON	OFF	CPU error	
ON	OFF	ON	X	Y	Z	U	Driver alarm	An error has occurred in the servodriver indicated by indicators 0 through 3. Either the servodriver's power is OFF or an error has occurred. The error can be ignored if the indicated axis isn't being used. The MC Unit can start as long as the driver alarm hasn't occurred in an axis being used.
ON	ON	ON	OFF	OFF	OFF	OFF	MC Unit error	An error has occurred in the MC Unit itself. The servodriver is OK. Details on the error can be determined by checking the error code output to the Teaching Box or PC data area interface. (See note.)

Note Refer to *10-2 System Errors: Error Codes 0001 to 0013*, *10-3 Task Errors: Error Codes 0020 to 0044*, or *10-4 Axis Errors: Error Codes 0060 to 0088* for details on the error code.

10-2 System Errors: Error Codes 0001 to 0013

When a system error occurs, the System Error Flag (CIO word n+11 bit 00) will be turned ON and the error code will be output to DM word m+45. (The error message in parentheses will be displayed at the Teaching Box.)

Code	Error	Error Processing
0001	System parameters destroyed (SYS PARA CORRUPT)	The system parameters have been destroyed. (See note 1.) Download the system parameters from the MC Support Software again, and then turn the power OFF and ON. The system parameters can be destroyed by any of the following: The memory data might have been destroyed by noise. The backup battery voltage might have dropped. The PC (MC Unit) might have been turned OFF while the system parameters were being downloaded. This error can't be cleared by an error reset.
0002	Wrong number of tasks (TASK NOT CONSIST)	The number of tasks registered in the system parameters doesn't match the number register in the program. (See note 1.) Either delete all programs using the MC Support Software or download system parameters with the number of tasks that matches the number registered in the program. The error is cleared automatically when all programs are deleted. Download the program(s) again. When downloading the system parameters again, turn the Unit's power OFF and then ON again. This error can be caused by any of the following: The system parameters that were downloaded had a different number of tasks registered. The memory data might have been destroyed by noise. The backup battery voltage might have dropped. The PC (MC Unit) might have been turned OFF while the system parameters were being downloaded. This error can't be cleared by an error reset.
0003	Task 1 program destroyed (TASK 1 PROG CORR)	Task 1's program has been destroyed. Delete all task 1 programs and the error will be cleared automatically, then download the programs again. The programs can be destroyed by any of the following: The memory data might have been destroyed by noise. The backup battery voltage might have dropped. The PC (MC Unit) might have been turned OFF while the system parameters were being downloaded. The PC (MC Unit) might have been turned OFF while the programs were being downloaded. This error can't be cleared by an error reset.
0004	Task 2 program destroyed (TASK 2 PROG CORR)	Task 2's program has been destroyed. See the error processing described for error code 0003.
0005	Task 3 program destroyed (TASK 3 PROG CORR)	Task 3's program has been destroyed. See the error processing described for error code 0003.
0006	Task 4 program destroyed (TASK 4 PROG CORR)	Task 4's program has been destroyed. See the error processing described for error code 0003.

Code	Error	Error Processing
0007	CPU Bus stopped (CPU BUS STOP ERR)	Cyclic processing (the periodic refreshing process) couldn't be executed with the PC itself. (See note 2.) This error can be caused by either of the following: The PC has too heavy of a servicing load because servicing time for other CPU Bus Units increased (due to online editing from a host computer, for example). CPU Bus Service to the MC Unit has been disabled by turning ON its CPU Service Disable Bit in word A015. If the PC's servicing load is too heavy, lengthen the maximum cycle time (R in the PC Setup). If CPU Bus service has been disabled, turn OFF the MC Unit's CPU Service Disable Bit in word A015. This error can be cleared by an error reset, but will occur again if the cause isn't corrected.
0008	Parity error (PARITY ERROR)	A hardware error has been detected in communications with the PC. (See note 2.) This error can be caused by a problem in the MC Unit or the PC. Try removing the MC Unit from the Backplane and then reinstalling it. If this doesn't solve the problem, replace the MC Unit or PC. This error can be cleared by an error reset, but will occur again if the cause isn't corrected.
0009	PC stopped (PC STOP ERROR)	A fatal or non-fatal error has occurred in the PC. (See note 2.) Refer to the PC's Operation Manual for details on fatal and non-fatal errors. The PC stopped error can be cleared by an error reset, but will occur again if the cause isn't corrected.
0010	Deceleration stop (DECEL STOP)	The Teaching Box's deceleration stop key was pressed. Reset the error after checking for any unsafe conditions. This error can be cleared by an error reset, but will occur again if the cause isn't corrected.
0011	Setup error (SYSTEM SET ERROR)	System parameters with new Unit or Machine parameters were downloaded. The error can be cleared by an error reset, but the Unit will operate with the old parameters unless the power is turned OFF and then ON again.
0012	Illegal data (DATA ERROR)	Destroyed position data was received. The error can be cleared by an error reset, but transfer the position data again. The position data can be destroyed by either of the following: The memory data might have been destroyed by noise. The backup battery voltage might have dropped.
0013	Error log destroyed (ERR HISTORY CORR)	The error history data was destroyed. The error history will be initialized automatically when the data has been destroyed. Either reset the error or turn the power OFF and then ON again. The error history data can be destroyed by either of the following: The memory data might have been destroyed by noise. The backup battery voltage might have dropped.

Note 1. Checked when power is turned ON.
2. Checked regularly.

10-3 Task Errors: Error Codes 0020 to 0044

When a system error occurs in task 1, 2, 3, or 4, the error code will be output to DM word m+46, m+47, m+48, or m+49, respectively. In this table, the term "program" refers to the G-language MC program. The error message in parentheses will be displayed at the Teaching Box.

Code	Error	Error Processing
0020	Program number error (PROGRAM No. ERR)	The specified program number isn't BCD or is outside of the acceptable range. Specify a 4-digit BCD program number from 0000 to 0999.
0021	Program not found (NO PROGRAM)	The specified main program or subprogram doesn't exist. If a SUBPROGRAM JUMP function (G72) was executed, the specified subprogram doesn't exist. Check whether the specified program has been created or the specified program number was incorrect.

Code	Error	Error Processing
0022	Program axis declaration error (AXIS SPEC ERR)	The axes used in the program are set in the parameters. Be sure to set the axes being used in the parameters. This setting is made with the MC Support Software.
0023	Illegal G code (INV G CODE COMND)	An unused G code was used in the program. Somehow the program has been destroyed. Download the program again. The program could have been destroyed by any of the following: The memory data might have been destroyed by noise. The backup battery voltage might have dropped. The PC (MC Unit) might have been turned OFF while the program was being downloaded.
0024	Program out of range (PROG RANGE OVER)	Attempted to execute the next block of the program, but no block existed. Somehow the program has been destroyed. Download the program again. The program could have been destroyed by any of the following: The memory data might have been destroyed by noise. The backup battery voltage might have dropped. The PC (MC Unit) might have been turned OFF while the program was being downloaded.
0025	M code out of range (M CODE RANGE OVR)	Attempted to output an M code from a position data address or register, but the content wasn't within the acceptable range for M codes (000 to 999). Make sure that the M code is from 000 to 999.
0026	Interpolation rate not set (INTER SP NOT SET)	The interpolation speed wasn't specified when an interpolation function (G01, G02, or G03) was executed, and the interpolation speed hadn't been specified since program execution was started. Specify the interpolation speed in the program.
0027	Arc interpolation command value error (CIR ARC INT ERR)	A circular interpolation function (G01, G02, or G03) was executed, but an arc can't be drawn because of one of the following reasons. The arc angle was too small because the start point and end point were extremely close and the radius was too large. With radius specification, the start point and end point are the same or the distance between the start point and end point is small compared to the radius. With center specification, the distance between the center and the start point or end point is zero. With center specification, the difference is too great between the radius of the center and the start point and the radius of the center and the end point.
0028	Arc interpolation plane specification error (CIR ARC PLNE ERR)	A circular interpolation function (G01, G02, or G03) was executed, but the circular plane wasn't set or was set incorrectly. An axis specified in a circular interpolation function wasn't set with the circular plane specification function. The circular plane specification function specified an axis that isn't used in the task.
0029	Overflow error (OVERFLOW ERROR)	An overflow occurred when the axis command value was converted with the minimum setting value or pulse rate in a G00, G01, G02, or G03 function. Correct the program or the position data. Check the setting for the minimum setting unit parameter in the system parameters.
0030	Divide-by-zero error (ZERO DIVIDE ERR)	A division by zero occurred when a G60 function (ARITHMETIC OPERATIONS) was executed. Correct the program or the position data. Check the setting for the minimum setting unit parameter in the system parameters.
0031	Position data out of range (POSIT DATA OVER)	The position data exceeded the acceptable range when a G60 function (ARITHMETIC OPERATIONS) was executed or the specified axis' origin wasn't fixed when a current position was substituted in a G63 function (SUBSTITUTION). If the error occurred in a G60 function, correct the program or position data. If the error occurred in a G63 function, correct the program or fix the location of the specified axis' origin by executing an origin search.

Code	Error	Error Processing
0032	Register out of range (REGIST RANGE OVR)	The register contents exceeded the acceptable range when a G60 function (ARITHMETIC OPERATIONS) or a G63 function (SUBSTITUTION) was executed. Correct the program or the data.
0033	Parameter out of range (PARA RANGE OVER)	The parameter setting exceeded the acceptable range when a G69 function (CHANGE PARAMETER) was executed. Correct the program or the data.
0034	Nesting error (NESTING ERROR)	There were six or more nesting levels when a G72 function (SUBPROGRAM JUMP) was executed. Correct the program so that there are five or fewer nesting levels.
0035	No return destination (NO RETURN DESTIN)	A subprogram's program number was read to start execution, so there was no return destination when the SUBPROGRAM END function (G73) was executed. Always start the main program first and then call the subprogram from the main program.
0036	Program number not set (PROG No. NOT SET)	When the MC program was started from the ladder program, the Cycle Start Bit was turned ON without turning ON the Program Number Read Bit. Turn ON the Cycle Start Bit after turning ON the Program Number Read Bit. The Program Number Read Bit must be turned ON when the Cycle Start Bit is turned ON after using the MC Support Software to add, edit, or delete any MC programs in the task.
0037	Number out of range (NUM RANGE OVER)	A numerical value exceeded the acceptable range during execution of G04 (time value), G53 (offset value), G54 (preset value), G69 (parameter value), or G70 (number of loops) function. An overflow occurred when the present value preset from the PC, the offset value, or preset value was converted with the minimum setting unit from the system parameters. Correct the program, data, or preset command value from the ladder program. Check the minimum setting unit in the system parameters when the error occurs in a present value preset from the PC, function G53, or function G54.
0038	Negative interpolation rate (INTER SP – DESIG)	A zero or negative value in a position data address or register was specified for the interpolation speed for linear or circular interpolation function. Correct the program or data so that a positive value is used for the interpolation speed.
0039	Illegal intermediate code (INTER CODE ERROR)	Somehow the program has been destroyed. Download the program again. The intermediate code error could have been caused by any of the following: The memory data might have been destroyed by noise. The backup battery voltage might have dropped.
0040	Operating time overflow (RUN TIME OVERRUN)	Attempted to execute a G00, G01, G02, or G03 positioning function, but the movement to the specified target would take too long at the specified speed. Correct the program, data, or parameters to increase the speed or reduce the distance to be moved.
0041	Position counter overflow (MOVE DIST OVER)	Can't move to the target specified in a G01 function because it is too far. Correct the program or data to reduce the distance to be moved.
0042	Position counter overflow during arc movement (CIR ARC DIST OVER)	Can't move the amount specified in a circular interpolation function because it is too far. Either the distance between the center and start/end point is too great, the arc length is too great, or the distance between the start and end points is too great. Correct the program or data to reduce the distance to be moved.
0043	Arc cannot be completed (CIR ARC CONT ERR)	Attempted to continue circular interpolation, but couldn't because an axis error occurred during circular interpolation, the circular interpolation was decelerated to a stop and then another positioning operation such as jogging was performed, or the circular interpolation was decelerated to a stop and then an axis error occurred. When this error occurs, circular interpolation can't be continued, so restart the program from the beginning. Circular interpolation can be continued when the interpolation is decelerated to a stop and no errors or other positioning operations occur while the interpolation is stopped.
0044	Additional axis speed overflow (SUP AXIS SP OVER)	The interpolation feed rate specified for the supplemental axis in circular interpolation exceeds the maximum interpolation feed rate set in the system parameters. Correct the program or data to reduce the supplemental axis' interpolation feed rate or increase the setting for the maximum interpolation feed rate parameter in the system parameters.

10-4 Axis Errors: Error Codes 0060 to 0088

When a error occurs in the X, Y, Z, or U axis, the error code will be output to DM word m+50, m+51, m+52, or m+53, respectively. (The error message in parentheses will be displayed at the Teaching Box.)

Code	Error	Error Processing
0060	Driver alarm encountered (DRV ALARM INPUT)	The driver alarm input went ON or an attempt was made to operate the axis or execute a CHANGE REFERENCE COORDINATE SYSTEM PV function (G54) with the driver alarm input ON after resetting an error. Check the status of the servodriver and correct the error if one is found. Turn the servodriver ON if it's OFF.
0061	Error counter overflow (ERR CNT OVERFLOW)	The number of pulses in the error counter exceeded the upper limit (65,535 pulses). This error could be caused by a disconnected/broken encoder cable, electrical noise, improperly adjusted servo system, or improperly adjusted/faulty machine system.
0062	Clockwise overtravel error (CW OVER TRAVEL)	The CW limit switch came ON during axis operation or an attempt was made to operate the axis with the CW limit switch ON after resetting an error. Move the axis in the CCW direction with manual operation.
0063	Counterclockwise overtravel error (CCW OVER TRAVEL)	The CCW limit switch came ON during axis operation or an attempt was made to operate the axis with the CCW limit switch ON after resetting an error. Move the axis in the CW direction with manual operation.
0064	Both direction overtravel error (CW, CCW OVER TR)	Both limit switches came ON during axis operation or an attempt was made to operate the axis with both limit switches ON after resetting an error. Check whether the limit switches have been wired incorrectly.
0065	No origin signal (NO ORIGIN SIGNAL)	During an origin search, the origin proximity switch went from ON to OFF, but then a limit switch went ON before the Z-phase input. This error could be caused by a fault Z-phase input in the encoder, a disconnected or broken encoder cable, or placement the origin proximity switch too close to a limit switch. Refer *Appendix B Additional Origin Search Patterns* for details on how errors occur during origin searches.
0066	No origin proximity signal (NO ORIG PROX SIG)	A limit switch went ON before the origin proximity switch during a one direction-mode origin search, or both limit switches went ON before the origin proximity switch during a reverse-mode origin search. Check the installation of the origin proximity switch and the wiring of the limit switches. Also check the location of the axes when the origin search was started. Refer to *Appendix B Additional Origin Search Patterns* for details on how errors occur during origin searches.
0067	Clockwise software limit (CW SOFT LIM OVR)	The command value exceeded the CW software limit in an axis operation, the absolute position exceeded the CW software limit when an absolute encoder's position was read, the command value exceeded the CW software limit in a G53 or G54 function, or the command value exceeded the CW software limit when the present value preset was executed from the PC. Check whether the software limit in the system parameters is appropriate. (An error might occur if circular interpolation is performed very close to the software limit.) Increase the software limits if they aren't appropriate. If they are appropriate, correct the program, data, or command value from the PC.
0068	Counterclockwise software limit (CCW SFT LIM OVR)	The command value exceeded the CCW software limit in an axis operation, the absolute position exceeded the CCW software limit when an absolute encoder's position was read, the command value exceeded the CCW software limit in a G53 or G54 function, or the command value exceeded the CCW software limit when the present value preset was executed from the PC. Check whether the software limit in the system parameters is appropriate. (An error might occur if circular interpolation is performed very close to the software limit.) Increase the software limits if they aren't appropriate. If they are appropriate, correct the program, data, or command value from the PC.
0069	Origin point not confirmed (ORIG NOT ESTABLI)	Attempted to execute an axis movement command even though the location of the origin hadn't been fixed. When using an incremental encoder, perform an origin search. When using an absolute encoder, perform a servo lock and fix the origin. If the servo lock is already ON, perform a servo free operation and then perform a servo lock.

Code	Error	Error Processing
0070	Servo lock error (SERVO LOCK ERROR)	Attempted to execute a G-code command or manual command, but the axis wasn't in servo lock status. Set the axis in servo lock.
0071	PV preset error (PV PRESET ERROR)	The preset value wasn't BCD or was out-of-range when a present value preset command was received from the PC. Correct the program or data so that the preset value is BCD and within the acceptable range.
0072	Emergency stop input (STOP INPUT)	The emergency stop input came ON or an attempt was made to operate the axis with the emergency stop input ON after resetting an error. Clear the emergency stop input.
0077	PV counter overflow (PRESPOS CNT OVER)	An overflow (2,147,483,647 pulses) occurred in the current position counter. This error could be caused by a faulty encoder, a disconnected/broken encoder cable, the axis moving beyond the counter's range, or electrical noise. Try executing the command again after correcting the problem.
0082	Origin proximity & overtravel simultaneously ON (OR PRX AND OT ON)	During an origin search, the origin proximity switch and the limit switch in the direction of the search are both ON at the same time. Change the mounting positions of the origin proximity switch and limit switch. Refer to *Appendix B Additional Origin Search Patterns* for details on how errors occur during origin searches.
0083	Overtravel always ON (OVER TRAVEL ON)	The limit switch in the direction of the search was already ON when a one direction-mode origin search was executed. Check the limit switch in the initial origin search direction. Also check the location of the axis when the origin search was started. Refer to *Appendix B Additional Origin Search Patterns* for details on how errors occur during origin searches.
0084	Origin proximity reverse error (OR PRX REVROT ER)	The reverse-mode origin search can't be performed because the limit switch in the direction of the search came ON while the search was reversing because of the origin proximity input. This is the same as when the origin proximity input is also used as a limit switch. Check the limit switch in the initial origin search direction as well as the mounting positions of the origin proximity switch and limit switch. Refer to *Reverse-mode Origin Searches 12* in *Appendix B* for details on this error.
0085	Overtravel reverse error (OT REV ROTAT ERR)	The reverse-mode origin search can't be performed because the other limit switch or the origin proximity switch came ON while the search was reversing because of a limit switch input. Check the limit switch in the initial origin search direction as well as the mounting positions of the origin proximity switch and limit switch. Refer to *Reverse-mode Origin Searches 12* in *Appendix B* for details on this error.
0086	Wiring error (WIRING ERROR)	A wiring error was detected during the wiring check automatically performed with the servo lock. Check whether the number of pulses and check time in the wiring check are correct. If they are correct, check whether the encoder's A-phase/B-phase wiring is correct.
0087	MPG ratio overflow (MPG FACTOR OVER)	The MPG's ratio (factor) is too great, so the command value can't be generated. Reduce the MPG ratio or the frequency of the pulses from the MPG. (Turn the MPG a little slower.)
0088	Abnormal feedback pulses (FEEDBK PULSE ERR)	The feedback pulses exceeded 195,000 pps or a task error occurred while the absolute encoder's data was being read and the read was cancelled. The excessive feedback pulse frequency could be caused by a broken encoder cable, encoder error, or noise.

10-5 Errors Output to the Error History

The following table lists errors that are output to the error history.

FAL No.	Error	Comments
001	PC watchdog timer error	Indicated by the MC Unit's indicators. See *10-1 Error Indicators*.
002	PC servicing error (cyclic)	System error with error code 0007
003	PC RAM error (parity)	System error with error code 0008
005	Unit number error	Indicated by the MC Unit's indicators. See *10-1 Error Indicators*.
006	Other CPU error	
008	System file setting error	
009	System file read error	
300	PC stopped error	System error with error code 0009
301	System parameters destroyed	System error with error code 0001
302	Program destroyed	System errors with error code 0003 through 0006 correspond to errors in tasks 1 through 4.
303	Illegal G code	Task error with error code 0023 (Output for task 1, 2, 3, or 4.)
304	Program out of range	Task error with error code 0024 (Output for task 1, 2, 3, or 4.)
305	No origin signal	Axis error with error code 0065 (Output for axis X, Y, U, or Z.)
306	Error counter overflow	Axis error with error code 0061 (Output for axis X, Y, U, or Z.)

SECTION 11
Maintenance and Inspection

This section describes the maintenance and inspection necessary to ensure proper operation of the MC Unit.

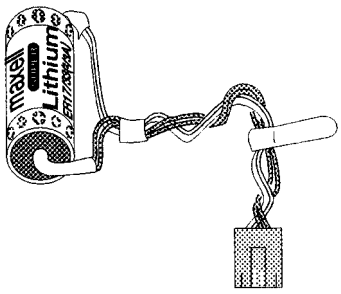
11-1 Battery Replacement

The backup battery's expected lifetime is 5 years at 25°C (77°F), shorter at higher temperatures. The MC Unit's BAT LOW indicator will light when the battery voltage drops below its rated value. Replace the battery when this indicator lights.

The Battery Alarm Flag will be turned ON when the BAT LOW indicator goes on. Refer to *6-6-30 Word n+11 Bit 07: Battery Alarm Flag* for details.

Replacement Battery

The replacement battery model is C500-BAT08.

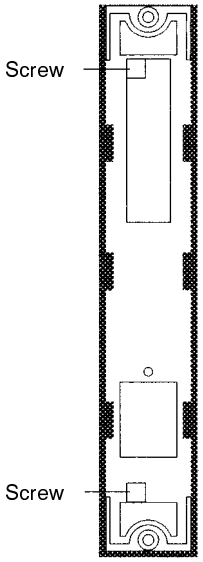


Procedure

Use the following procedure to replace the backup battery.

1, 2, 3...

1. Turn OFF the power. (If the power is already OFF, turn it ON for one minute and then turn it OFF.)
2. Remove the MC Unit from the Backplane.
3. Unscrew the two screws on the back of the Unit and remove the back cover.



4. Unplug the battery connector and install the new battery.
   Replace the battery within 5 minutes and be sure not to touch any electronic components within the Unit.
5. Replace the Unit's back cover and screws.

⚠ **Caution** Dispose of the battery properly.

11-2 Routine Inspections

In order for your MC Unit to continue operating at optimum condition, periodic inspections are necessary. The main components of the Unit are semiconductors and have a long service life, but depending on the operating environment, there may be more or less deterioration of these and other parts. A standard inspection schedule is once every six months to one year. More frequent inspections may be advisable depending on the operating environment. Try to maintain the inspection schedule once it has been set.

Inspection Points

Check to be sure that the power supply, ambient temperature, humidity, etc. are within the specifications. Be sure that these are no loose screws and that all battery and cable connections are secure. Clean any dust or dirt that has accumulated.

Item	Inspection points	Specification
Power Supply	Measure the voltage variations at the power supply terminal block.	100 to 120 VAC (85 to 132 VAC max.) 200 to 240 VAC (170 to 264 VAC max.) 24 VDC (20.4 to 26.4 VDC max.)
I/O Power Supply	Measure the voltage variations at the I/O power supply terminal block.	Within I/O specifications.
Ambient conditions	Ambient temperature	0°C to 55°C
	Ambient humidity	10% to 90% RH
	Dust/Dirt accumulation	None
System condition	Each Unit's installation	No looseness
	Cable connections	
	Wiring terminal screws	
	Damage to external wiring and cables	None
Ground	Check each Unit/machine's ground.	The ground should be connected.
Insulation	Check that each terminal is insulated.	No shorts or damage insulation
Battery	Check the battery.	Expected life: 5 years (at 25°C)

Required Tools

The following tools are recommended when performing an inspection.

Standard and phillips-head screwdrivers
Voltmeter
Alcohol and a clean towel
Oscilloscope
Humidity gauge
Thermometer

Precautions

Be sure to turn OFF the power when performing an inspection or replacing a Unit. After replacing a Unit make sure the new Unit is operating properly.

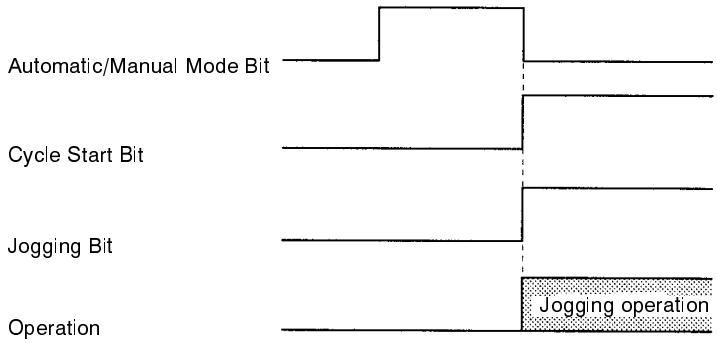
If a contact isn't functioning properly, try cleaning the contact with a clean cloth an alcohol. Remove any lint from the towel before re-installing the Unit.

Appendix A
Control Bit/Flag Timing Charts

This appendix provides timing charts that show the operation of control bits and flags when the MC Unit is operated in manual or automatic mode. The following table lists the timing charts.

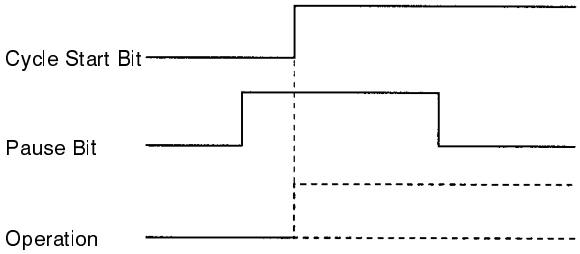
Timing Charts

1. Automatic/Manual Mode Bit, Cycle Start Bit, and Jogging Bit



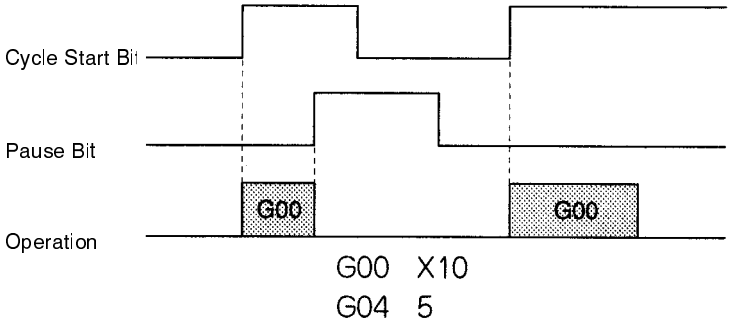
The Jogging Bit signal is received at the same time that manual mode is set. The Cycle Start Bit signal is ignored.

2. Cycle Start Bit and Pause Bit



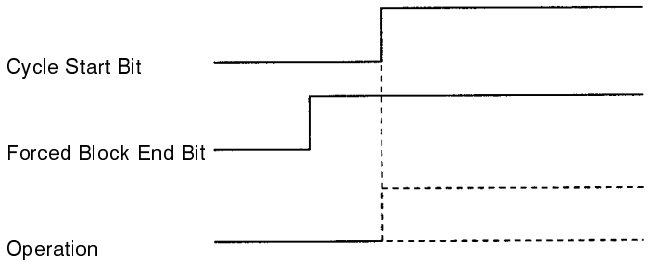
Even though the Cycle Start Bit is turned ON, MC program execution isn't started because the Pause Bit is ON. Program execution will begin if the Pause Bit is turned OFF and the Cycle Start Bit is turned OFF and then ON again.

3. Cycle Start Bit and Pause Bit



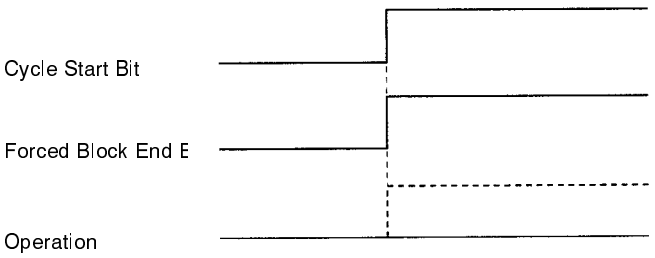
MC program execution is paused by turning ON the Pause Bit. Program execution will continue when the Pause Bit is turned OFF and the Cycle Start Bit is turned ON again.

4. Cycle Start Bit and Forced Block End Bit



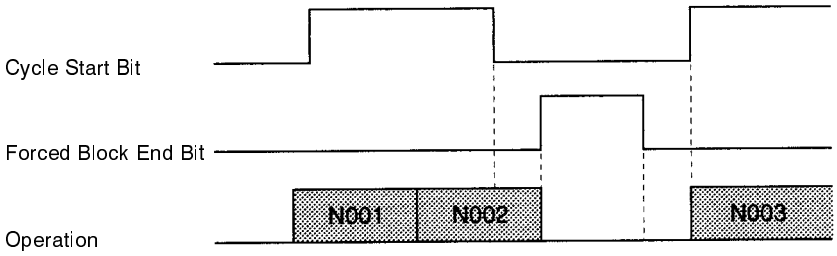
Even though the Cycle Start Bit is turned ON, MC program execution isn't started because the Forced Block End Bit is ON. Operation will begin if the Forced Block End Bit is turned OFF and the Cycle Start Bit is turned OFF and then ON again.

5. Cycle Start Bit and Forced Block End Bit



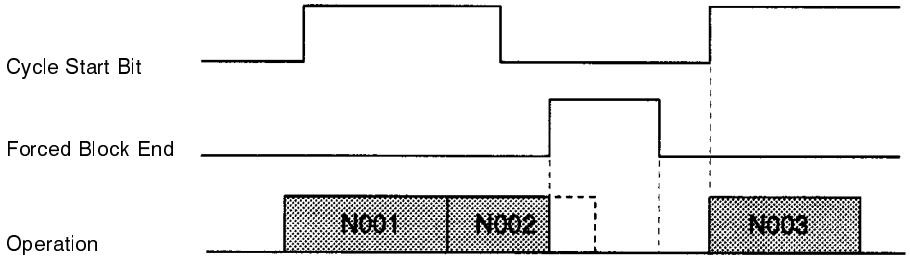
When the Cycle Start Bit and Forced Block End Bit are turned ON at the same time, the Forced Block End Bit takes precedence, so the program isn't executed. Operation will begin if the Forced Block End Bit is turned OFF and the Cycle Start Bit is turned OFF and then ON again.

6. Cycle Start Bit and Forced Block End Bit



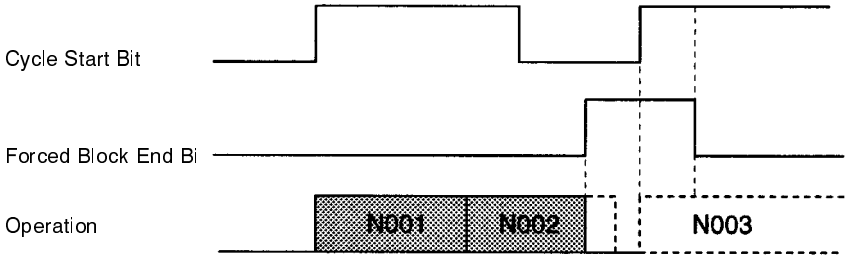
The Forced Block End Bit is turned ON at the end of block N002, but the down-differentiation of the Cycle Start Bit takes precedence, so the Forced Block End Bit is ignored. Program execution is restarted when the Cycle Start Bit is turned ON again.

7. Cycle Start Bit and Forced Block End Bit



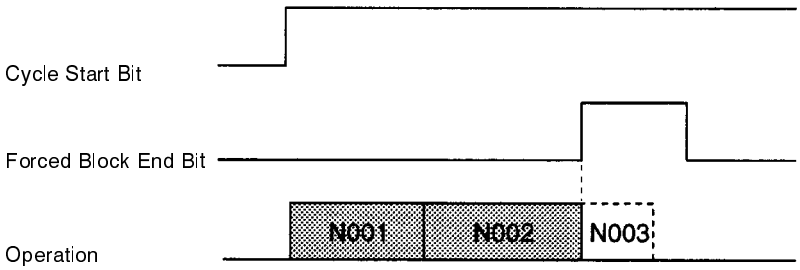
The Forced Block End Bit is turned ON to cancel execution of block N002 and stop program execution. Program execution is restarted when the Cycle Start Bit is turned ON again.

8. Cycle Start Bit and Forced Block End Bit



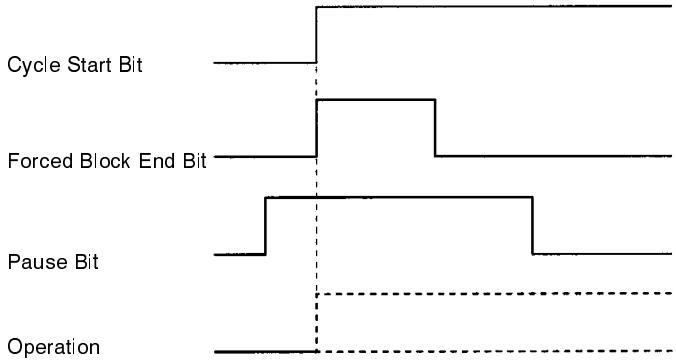
The Forced Block End Bit is turned ON to cancel execution of block N002 and stop program execution. Even though the Cycle Start Bit is turned ON again, the Forced Block End Bit is still ON, so program execution isn't restarted.

9. Cycle Start Bit and Forced Block End Bit



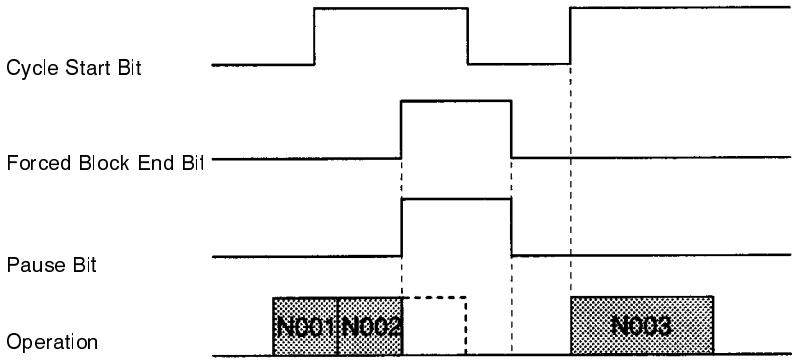
The Forced Block End Bit is turned ON between blocks N002 and N003, so execution of block N003 is cancelled and program execution is stopped.

10. Cycle Start Bit, Forced Block End Bit, and Pause Bit



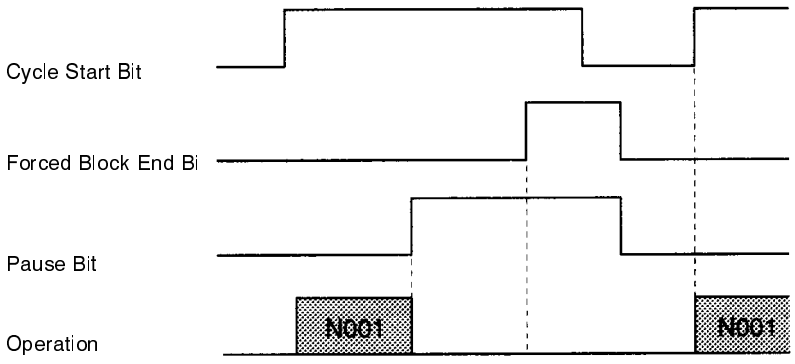
The Pause Bit is ON before the Cycle Start Bit is turned ON, so MC program execution isn't started. The Forced Block End Bit is turned ON at the same time as the Cycle Start Bit, but the Pause Bit is already ON and the Forced Block End Bit has no effect.

11. Cycle Start Bit, Forced Block End Bit, and Pause Bit



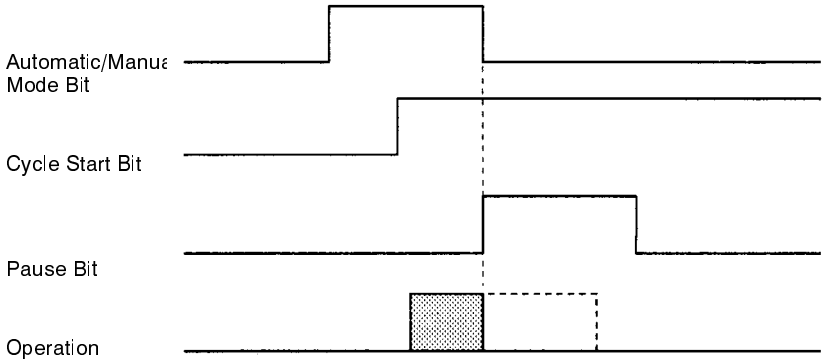
The Forced Block End Bit and Pause Bit are turned ON at the same time, but the Forced Block End Bit takes precedence, so execution of block N002 is cancelled.

12. Cycle Start Bit, Forced Block End Bit, and Pause Bit



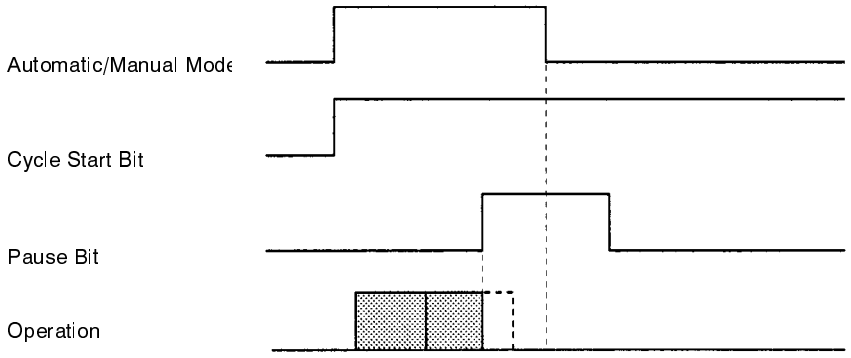
Program execution is paused with the Pause Bit. The Forced Block End Bit signal is ignored. Program execution will continue when the Cycle Start Bit is turned ON again.

13. Automatic/Manual Mode Bit, Cycle Start Bit, and Pause Bit



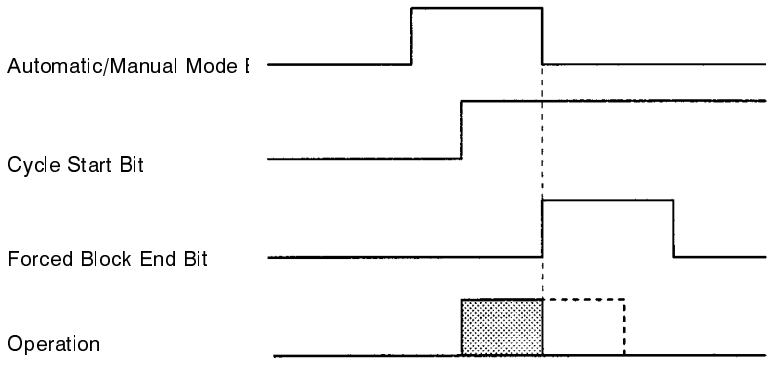
Program execution is stopped when the Automatic/Manual Mode Bit is turned ON. The Unit is in manual mode, so the Pause Bit is ignored.

14. Automatic/Manual Mode Bit, Cycle Start Bit, and Pause Bit



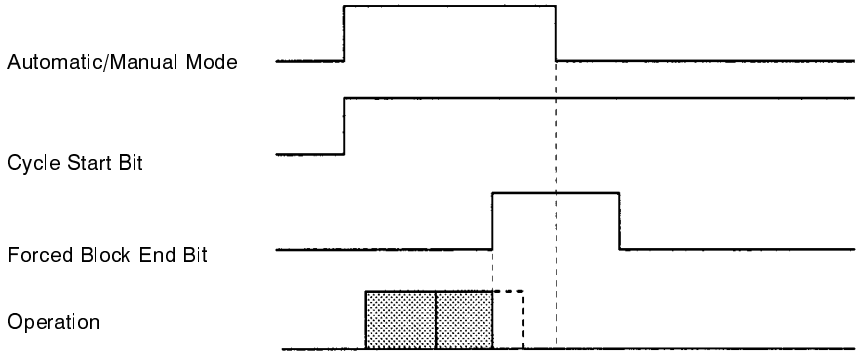
After program execution is paused with the Pause Bit, the Unit is switched to manual mode by turning OFF the Automatic/Manual Mode Bit.

15. Automatic/Manual Mode Bit, Cycle Start Bit, and Forced Block End Bit



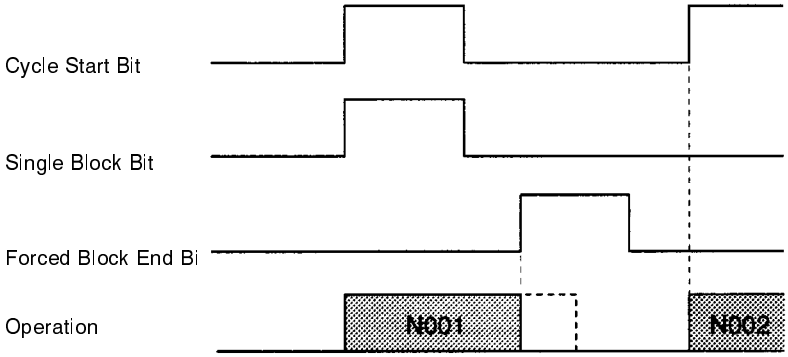
Program execution is stopped by turning OFF the Automatic/Manual Mode Bit. The Forced Block End Bit is ignored.

16. Automatic/Manual Mode Bit, Cycle Start Bit, and Forced Block End Bit



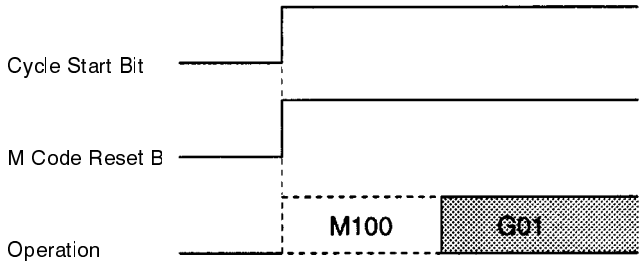
Program execution is stopped by the Forced Block End Bit. The Unit is switched to manual mode when the Automatic/Manual Mode Bit is turned OFF.

17. Cycle Start Bit, Single Block Bit, and Forced Block End Bit



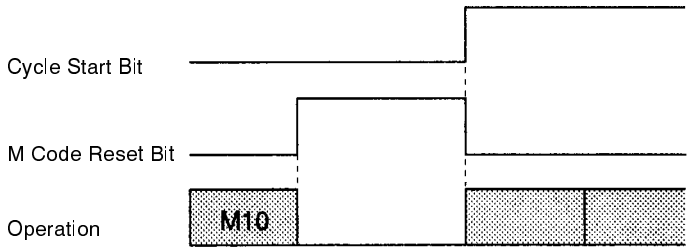
Program execution is started with the Single Block Bit ON, but execution of block N001 is stopped by the Forced Block End Bit. The next block is executed when the Cycle Start Bit is turned ON again.

18. Cycle Start Bit and M Code Reset Bit



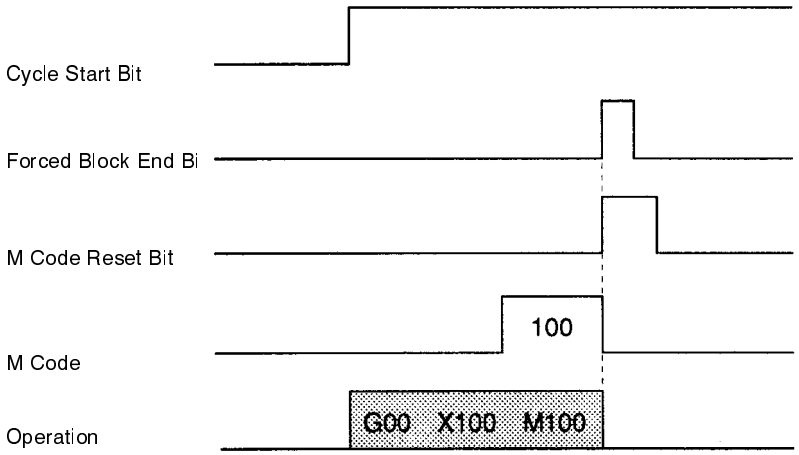
Program execution is started by the Cycle Start Bit. The next operation (G01) is executed without outputting M code 100 because the M Code Reset Bit is ON.

19. Cycle Start Bit and M Code Reset Bit



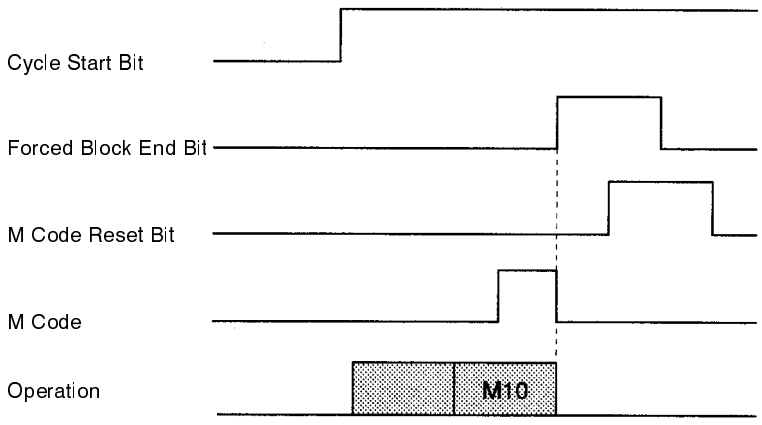
The Cycle Start Bit is invalid while the Unit is waiting for an M code reset. When the M Code Reset BIt is turned OFF, the standby status is cleared and the status of the Cycle Start Bit is checked. The Cycle Start Bit is ON, so program execution is restarted.

20. Cycle Start Bit, Forced Block End Bit, M Code Reset Bit, and M Code Output



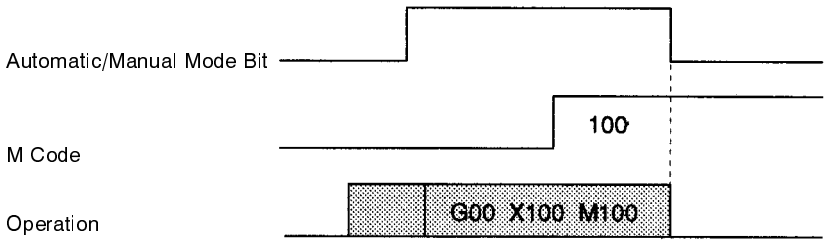
The Forced Block End Bit takes precedence and stops program execution when the Forced Block End Bit and the M Code Reset BIt are turned ON at the same time. The M code is cleared when the program is stopped. The program was already stopped, so program execution isn't restarted when the M Code Reset Bit is turned OFF.

21. Cycle Start Bit, Forced Block End Bit, M Code Reset Bit, and M Code Output



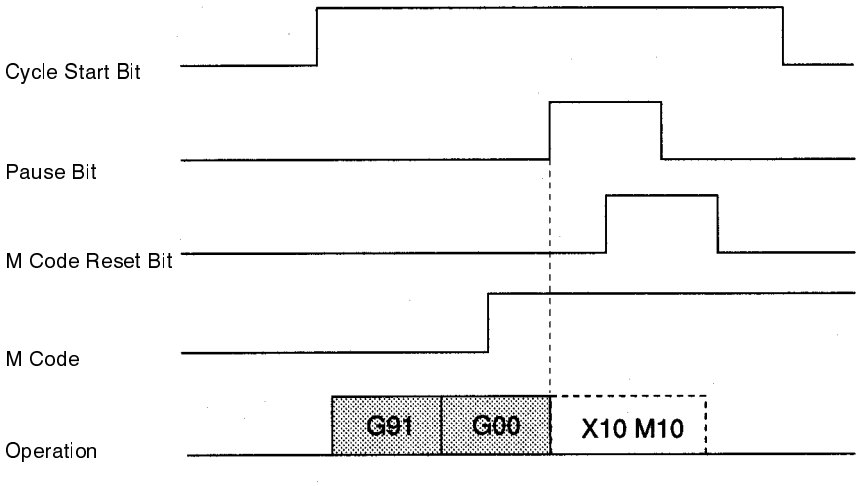
The program is stopped and the M code is cleared when the Forced Block End Bit goes ON.

22. Automatic/Manual Mode Bit and M Code Output



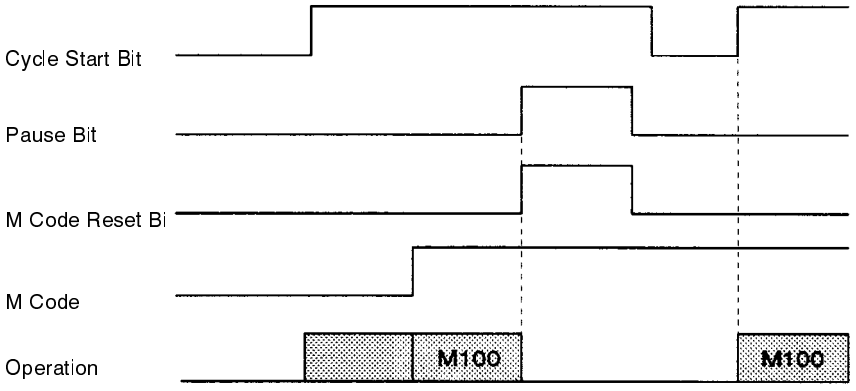
The M code isn't cleared when the Unit is switched manual mode.

23. Cycle Start Bit, Pause Bit, M Code Reset Bit, and M Code Output



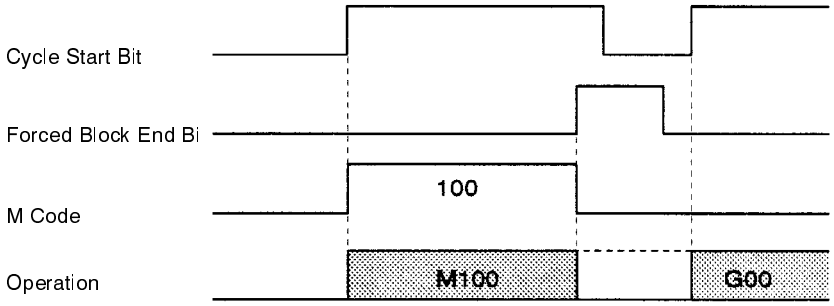
The program is paused when the Pause Bit is turned ON, but the M code isn't cleared. M code M10 isn't cleared by the M Code Reset Bit because program execution is paused.

24. Cycle Start Bit, Pause Bit, M Code Reset Bit, and M Code Output



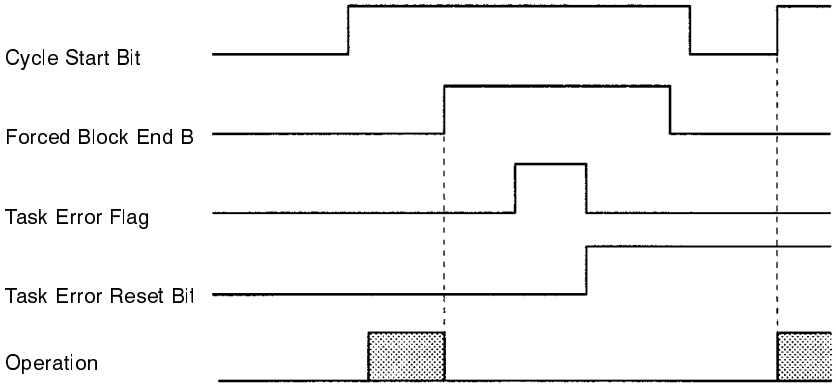
If the Pause Bit and M Code Reset Bit are turned ON at the same time, the Pause Bit becomes valid. Program execution is paused, but the M code isn't cleared. When the program is restarted, the M code will be output because it was ON originally.

25. Cycle Start Bit, Forced Block End Bit, and M Code Output



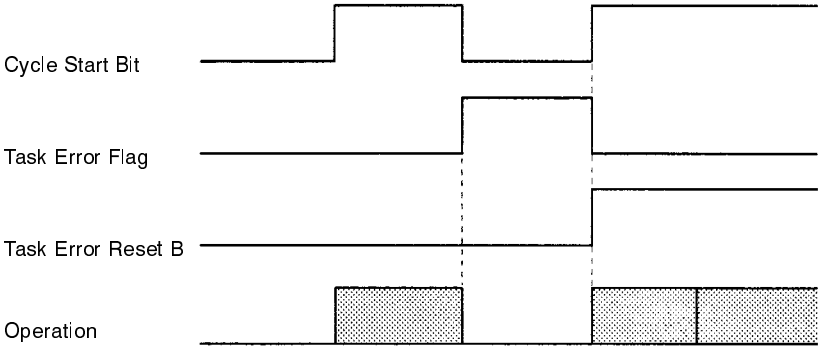
The Forced Block End Bit is turned ON while the Unit is standing by for an M code reset, cancelling the block.

26. Cycle Start Bit, Forced Block End Bit, Task Error Flag, and Task Error Reset Bit



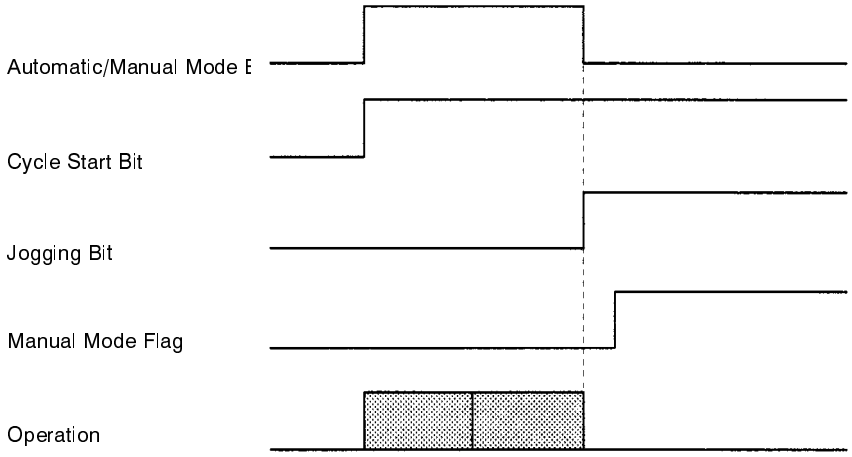
Operation is stopped with the Forced Block End Bit and then a task error occurs. Correct the cause of the error, clear the error by turning ON the Task Error Reset Bit, and restart the program by toggling the Cycle Start Bit.

27. Cycle Start Bit, Task Error Flag, and Task Error Reset Bit



After a task error occurs, the Task Error Reset Bit and Cycle Start Bit are turned ON at the same time, clearing the error and restarting operation simultaneously.

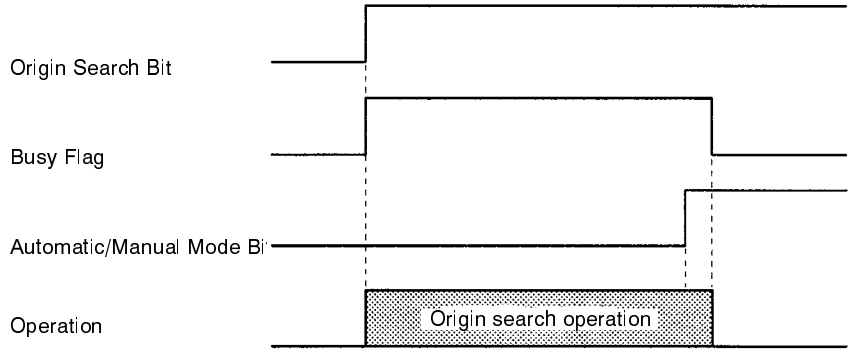
28. Automatic/Manual Mode Bit, Cycle Start Bit, Jogging Bit, and Manual Mode Flag



The Automatic/Manual Mode Bit is turned OFF while the program is being executed, switching the Unit to manual mode. Switching to manual mode causes the program to stop and axes to decelerate to a stop, but the Manual Mode Flag is OFF while the axes are decelerating so the Jogging Bit signal is invalid.

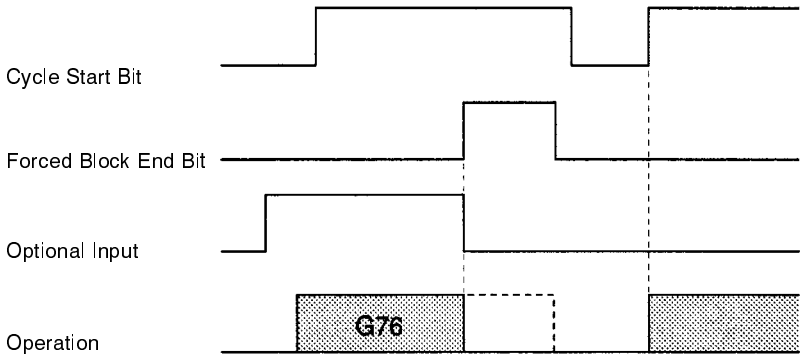
The Manual Mode Flag will be turned ON when positioning is completed after decelerating to a stop. The Jogging Bit signal can be received once the Manual Mode Flag is ON.

29. Origin Search Bit, Busy Flag, and Automatic/Manual Mode Bit



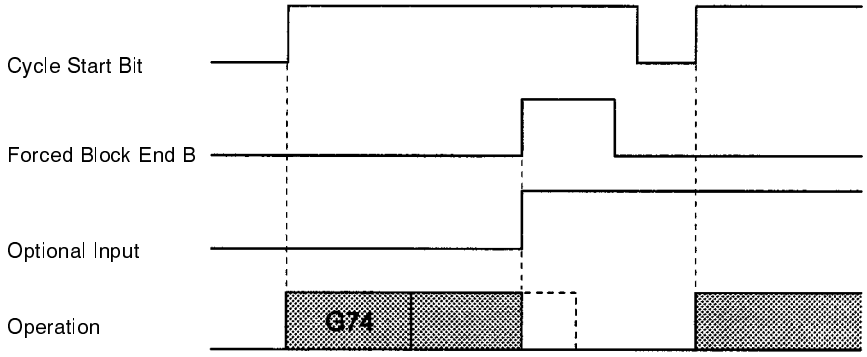
The manual mode origin search is stopped when the Automatic/Manual Mode Bit is turned ON. All axes in the task are stopped and the Unit enters automatic mode. At this point, the Busy Flags for all of the axes will be OFF, even though the Origin Search Bit remains ON.

30. Cycle Start Bit, Forced Block End Bit, and Optional Input



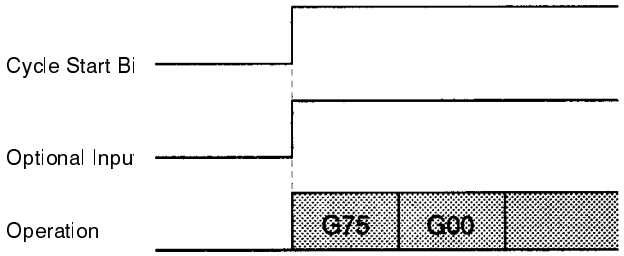
While the OPTIONAL PROGRAM STOP function (G76) is being executed, the block is cancelled by turning ON the Forced Block End Bit. The optional input is turned OFF at the same time that the Forced Block End Bit is turned ON, so in the end the input has no effect.

31. Cycle Start Bit, Forced Block End Bit, and Optional Input



While the block after the OPTIONAL END function (G74) is being executed, the block is cancelled by turning ON the Forced Block End Bit. The optional input and Forced Block End Bit are turned ON at the same time, but the Forced Block End Bit takes precedence.

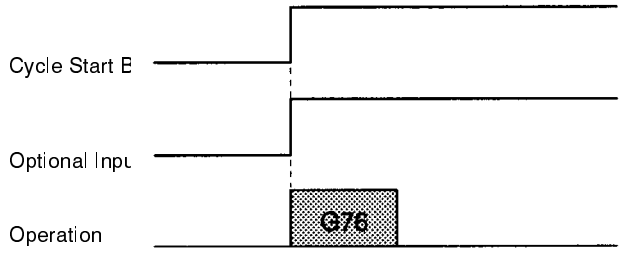
32. Cycle Start Bit and Optional Input




```
G75  1
G04  10
G00  X100
```

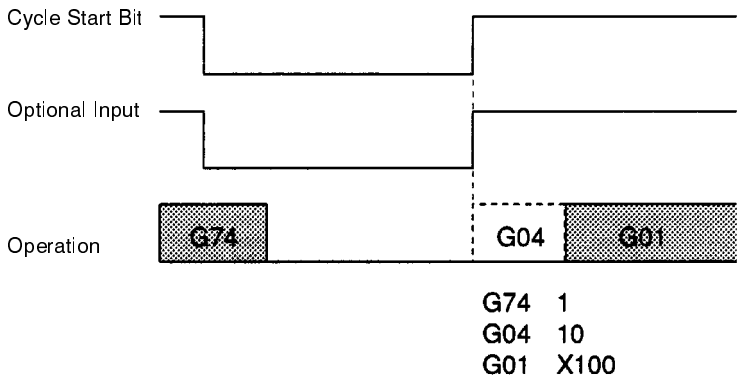
Program execution starts when the Cycle Start Bit is turned ON. The next block after the OPTIONAL SKIP function (G75) is skipped because the optional input is ON.

33. Cycle Start Bit and Optional Input



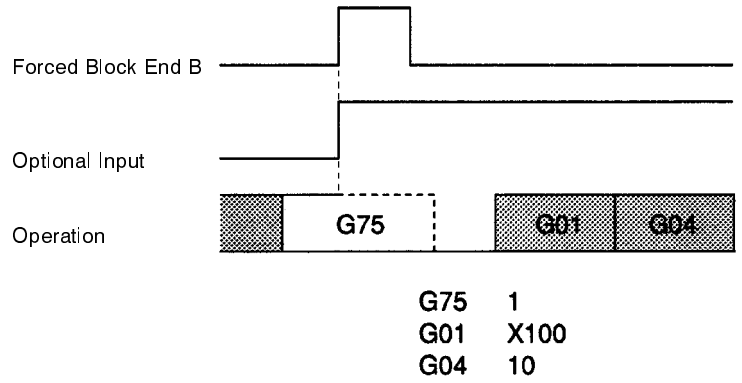
Program execution starts when the Cycle Start Bit is turned ON. The status of the optional input is checked when the OPTIONAL PROGRAM STOP function (G76) is executed, and the program is stopped because the optional input is ON.

34. Cycle Start Bit and Optional Input



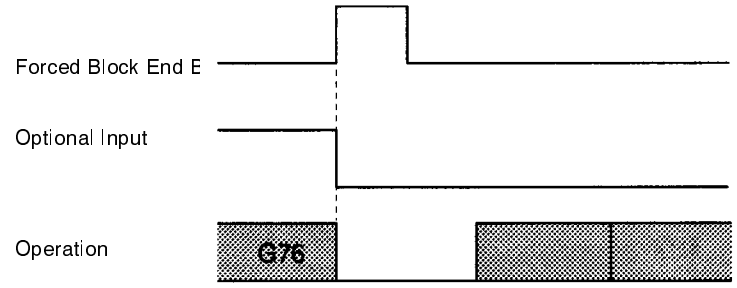
Program execution is restarted when the Cycle Start Bit is turned ON. The status of the optional input is read due to the execution of the OPTIONAL END function (G74) and G04 is cancelled.

35. Forced Block End Bit and Optional Input



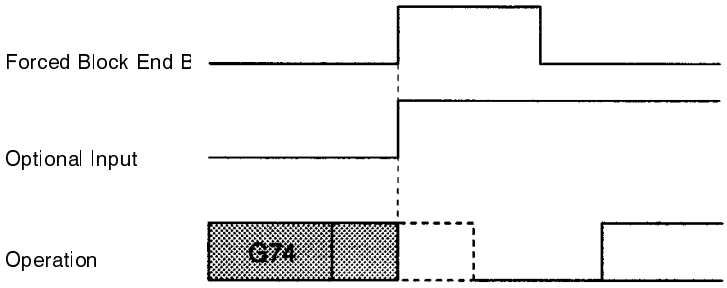
The Forced Block End Bit takes precedence if it's turned ON at the same time as the optional input, so the OPTIONAL SKIP function (G75) is cancelled and invalid. The optional input is ON when the program is restarted, but G01 isn't skipped.

36. Forced Block End Bit and Optional Input



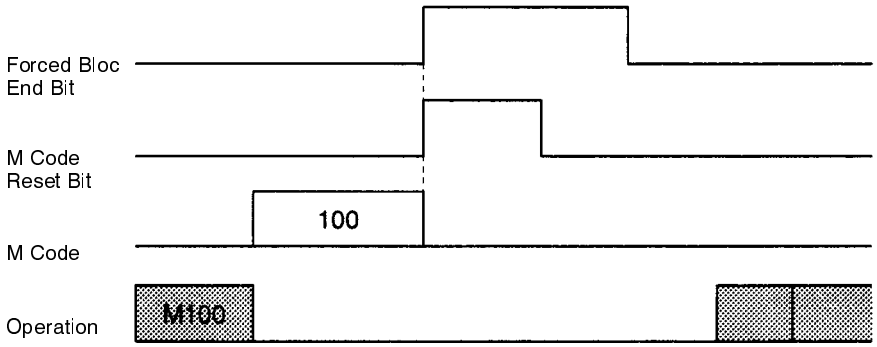
The OPTIONAL PROGRAM STOP function (G76) is cancelled by the Forced Block End Bit. The optional input is turned OFF at the same time, but this is ignored and program execution is stopped.

37. Forced Block End Bit and Optional Input



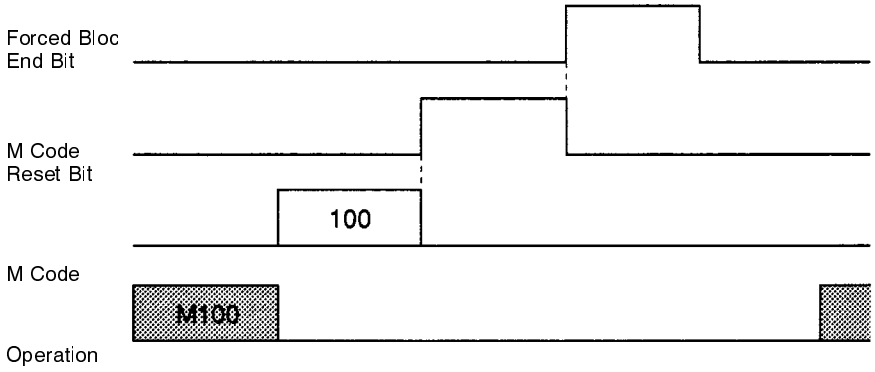
The Forced Block End Bit takes precedence if it's turned ON at the same time as the optional input in the block after an OPTIONAL END function (G74). The Unit stands by for restarting after program execution is stopped.

38. Forced Block End Bit, M Code Reset Bit, and M Code Output



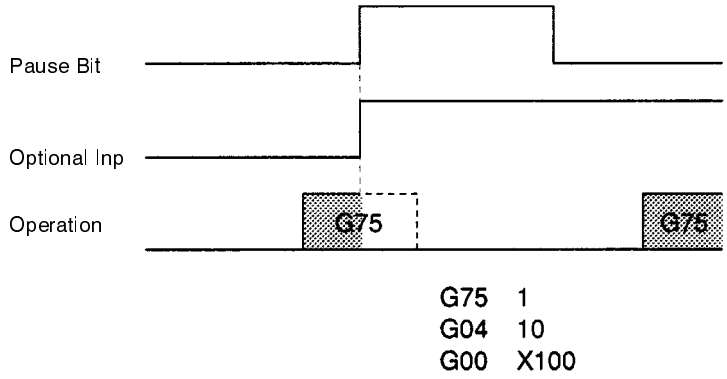
Turning ON the Forced Block End Bit clears the M code and stops program execution. The M Code Reset Bit signal is ignored.

39. Forced Block End Bit, M Code Reset Bit, and M Code Output



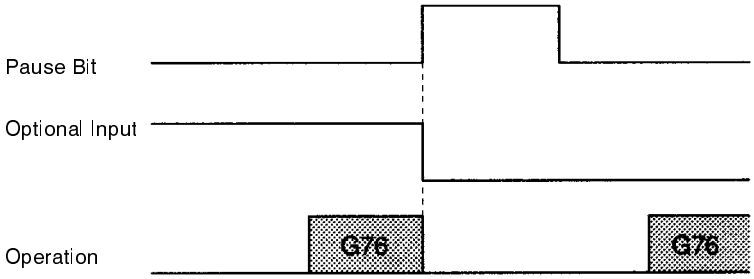
Turning ON the Forced Block End Bit cancels the M100 block waiting for an M code reset. The down-differentiation of the M Code Reset Bit is ignored.

40. Pause Bit and Optional Input



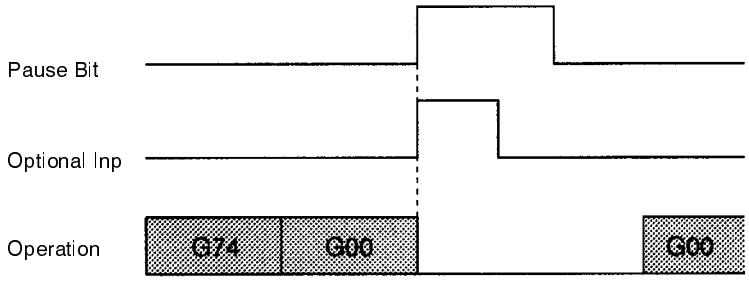
The execution of the OPTIONAL SKIP function (G75) is paused before it is completed by turning ON the Pause Bit. The optional input is ignored.

41. Pause Bit and Optional Input



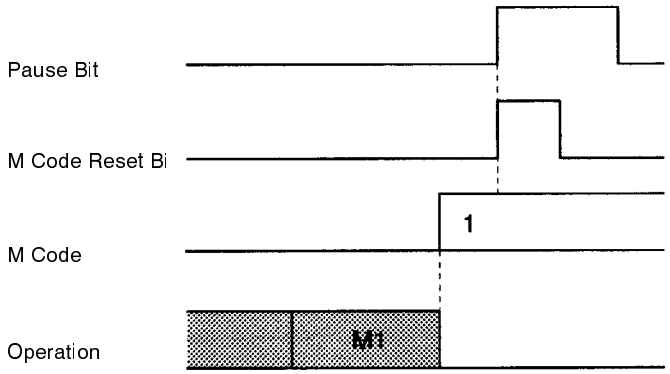
The execution of the OPTIONAL PROGRAM STOP function (G76) is paused before it is completed by turning ON the Pause Bit. The optional input is ignored even if it goes OFF at the same time. Function G76 is executed again when the program is restarted.

42. Pause Bit and Optional Input



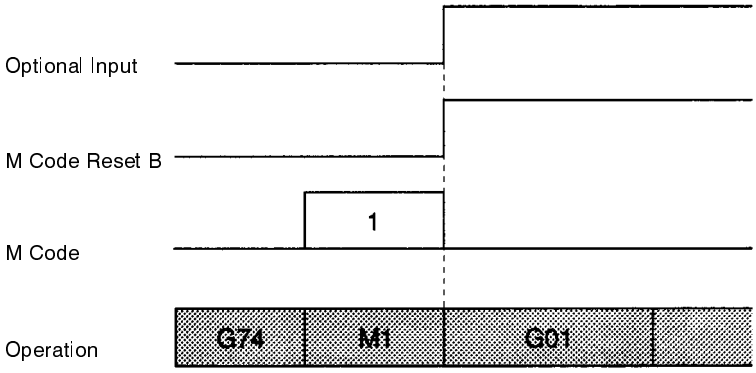
The G00 positioning operation is stopped by turning ON the Pause Bit in the block after the OPTIONAL END function (G74). The optional input is ignored even if it goes OFF at the same time. (The program isn't ended.) Execution of function G00 is continued when the program is restarted.

43. Pause Bit, M Code Reset Bit, and M Code Output



The program is stopped by the Pause Bit. The M Code Reset Bit is ignored, so the M code isn't cleared.

44. Optional Input, M Code Reset Bit, and M Code Output



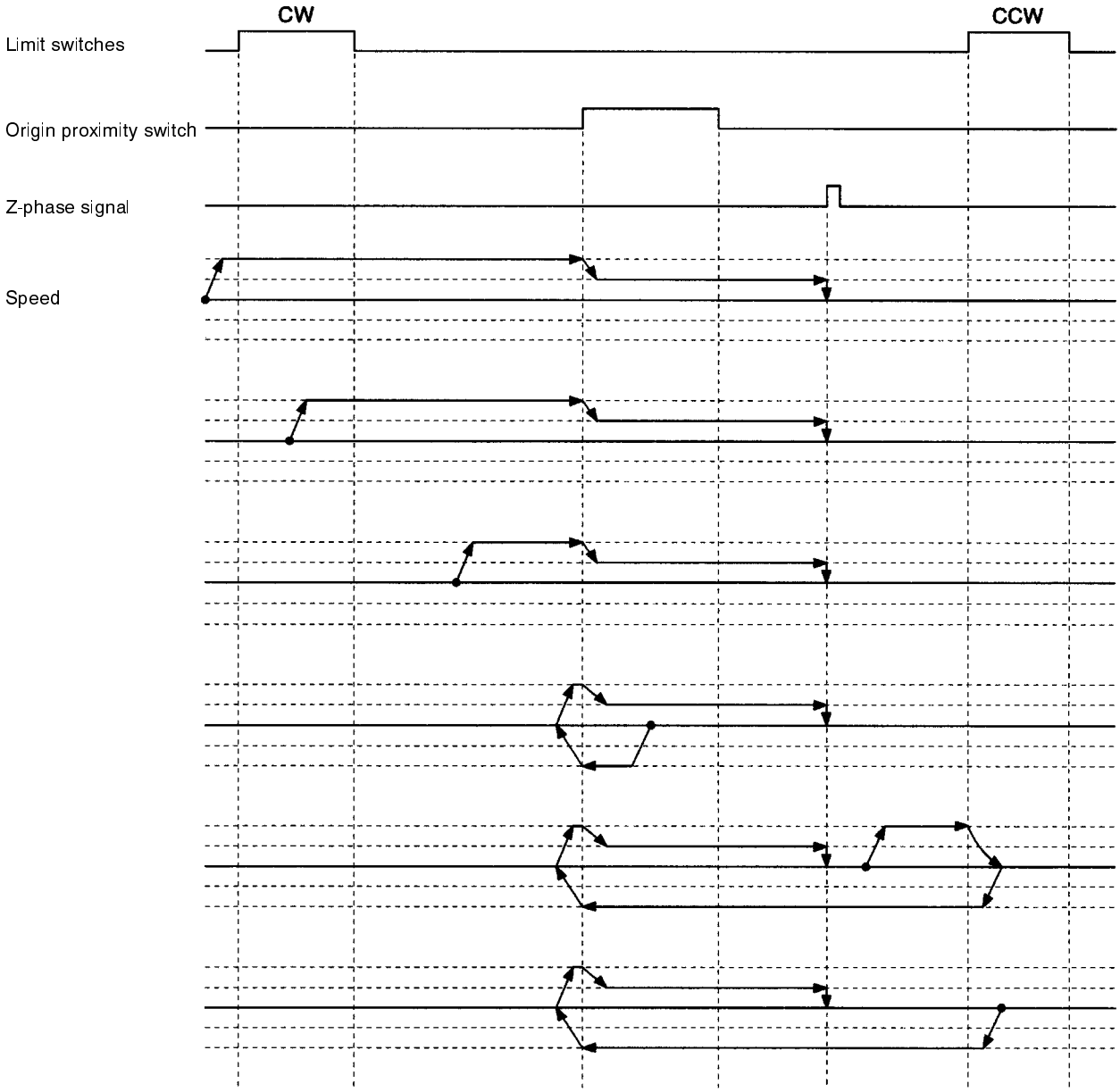
The M code command in the block after the OPTIONAL END function (G74) is stopped by the optional input. The M code is cleared and the next block is executed immediately. The M Code Reset Bit is ignored.

Appendix B
Additional Origin Search Patterns

This appendix provides additional examples of origin search patterns with the Origin Search Method set to Reverse mode or One Direction mode. Operation will vary depending on the position of the workpiece when the origin search is executed.

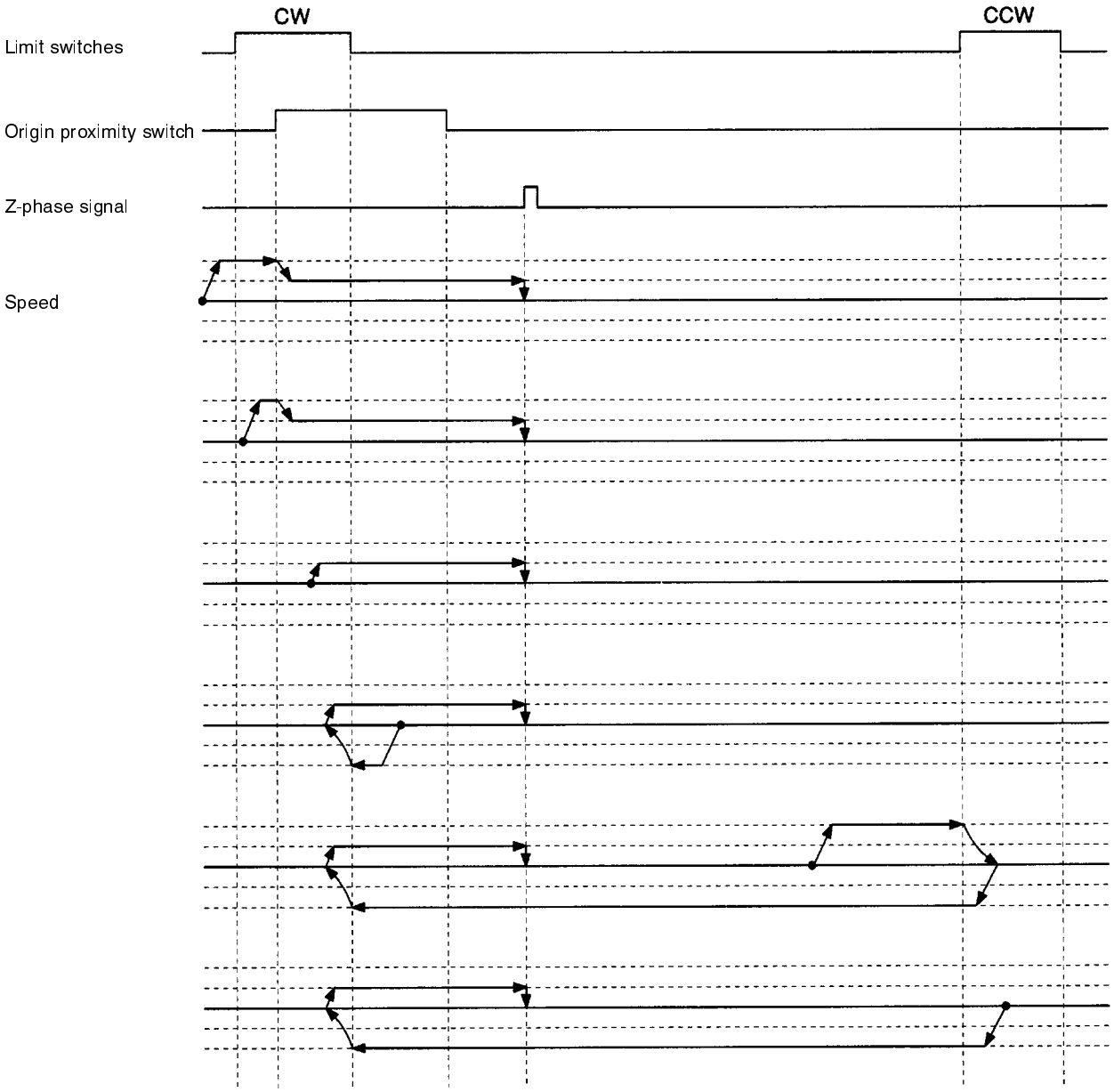
Reverse-mode Origin Searches 1

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW.



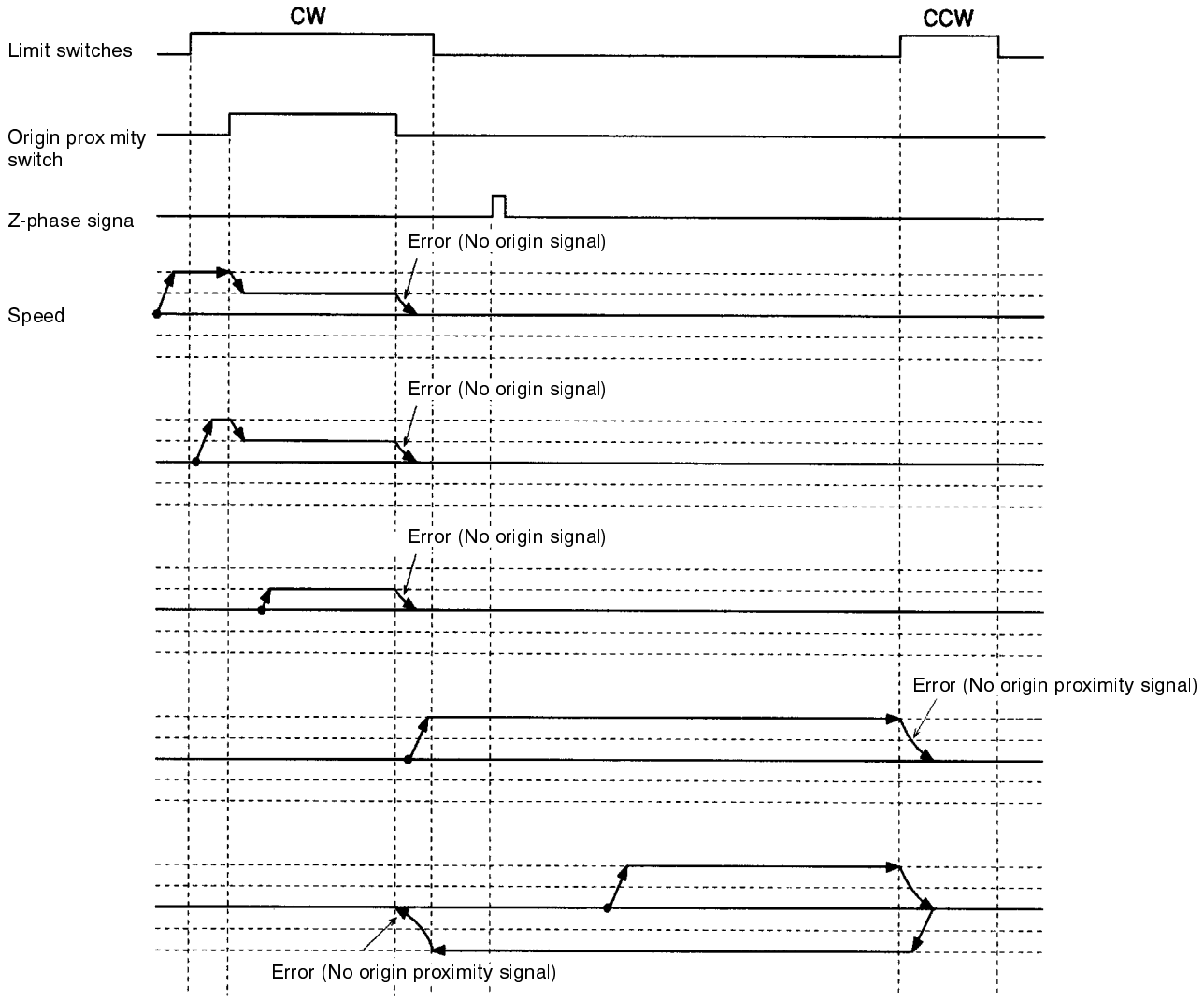
Reverse-mode Origin Searches 2

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW.



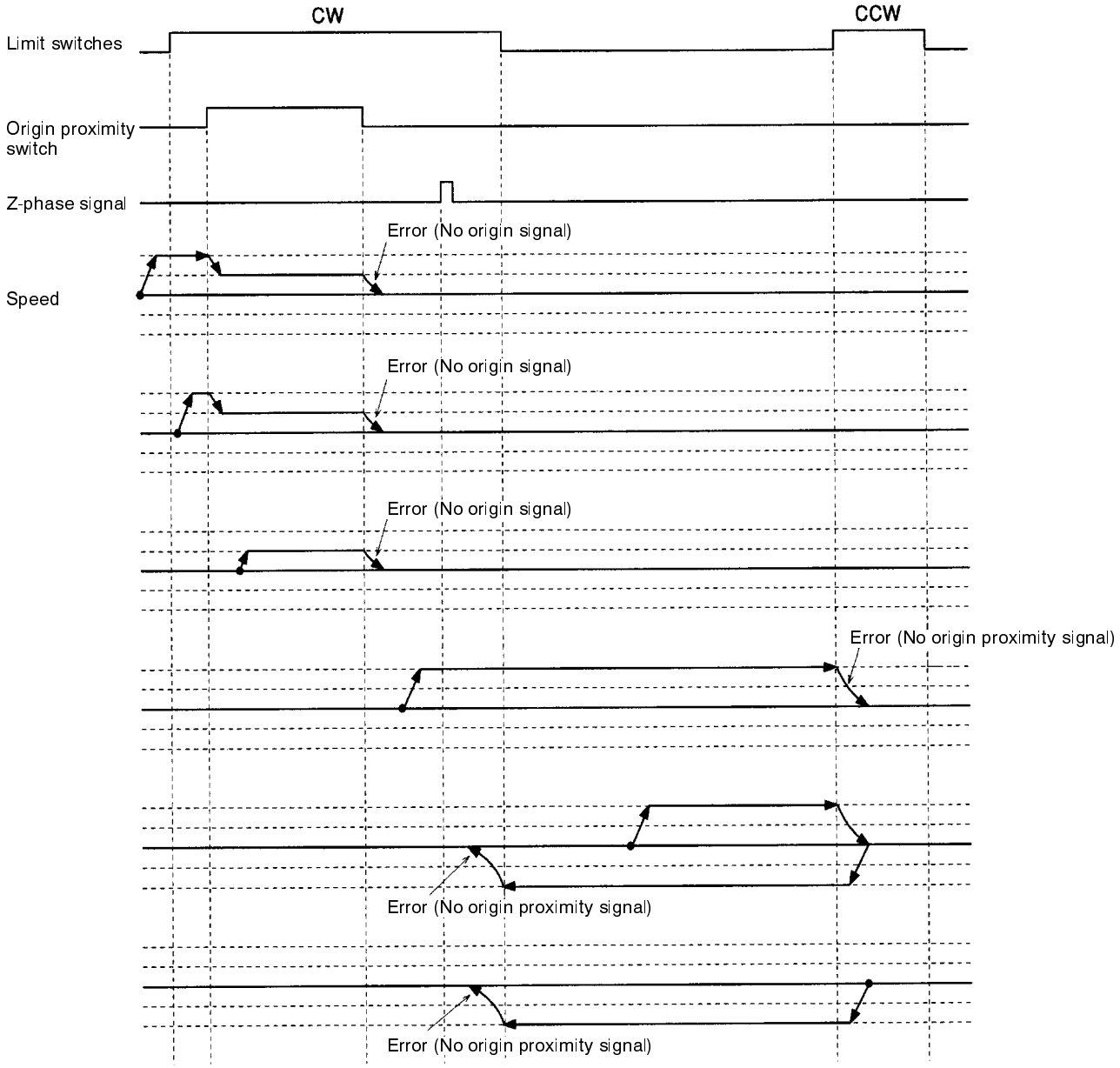
Reverse-mode Origin Searches 3

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW.



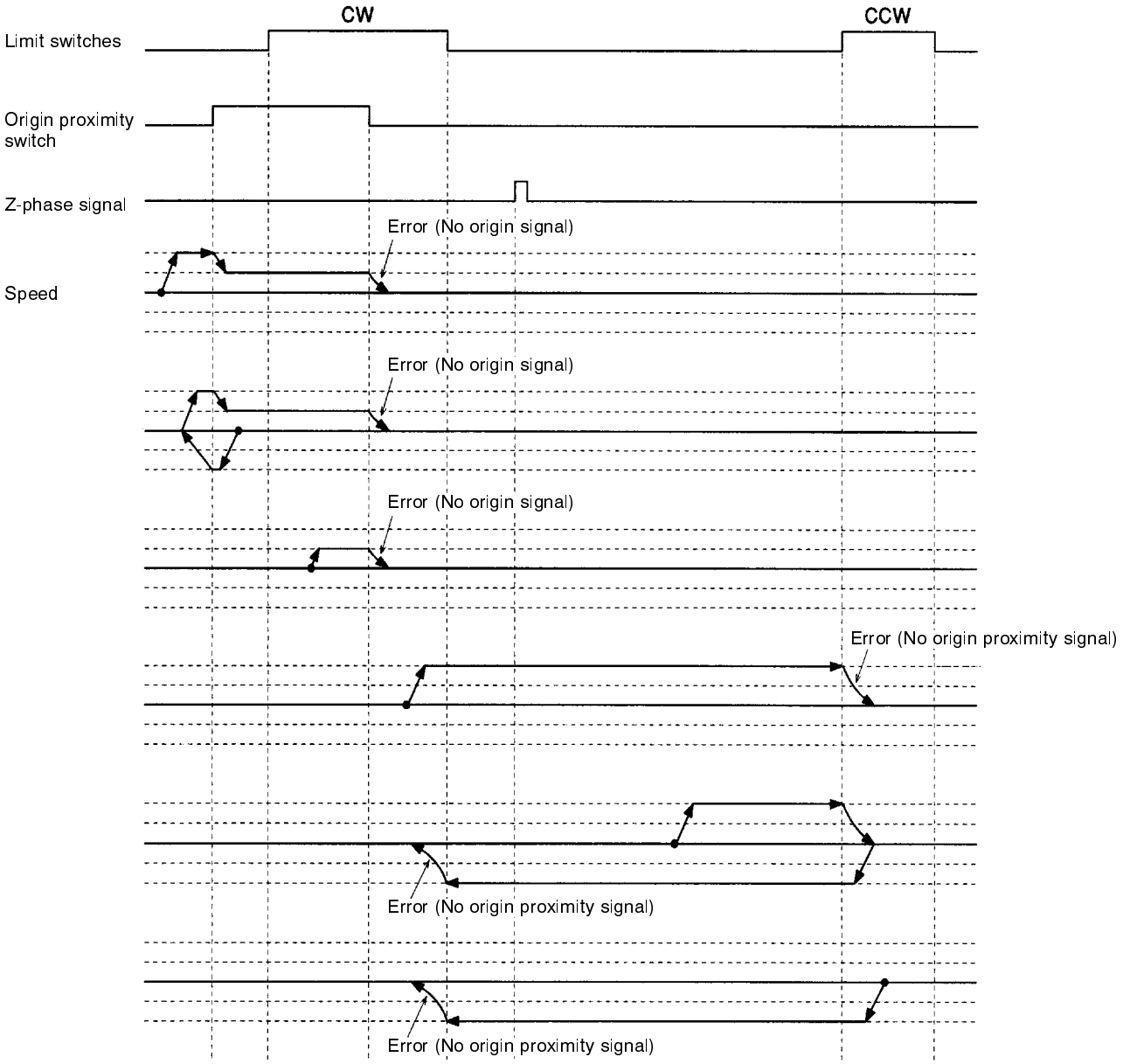
Reverse-mode Origin Searches 4

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW.



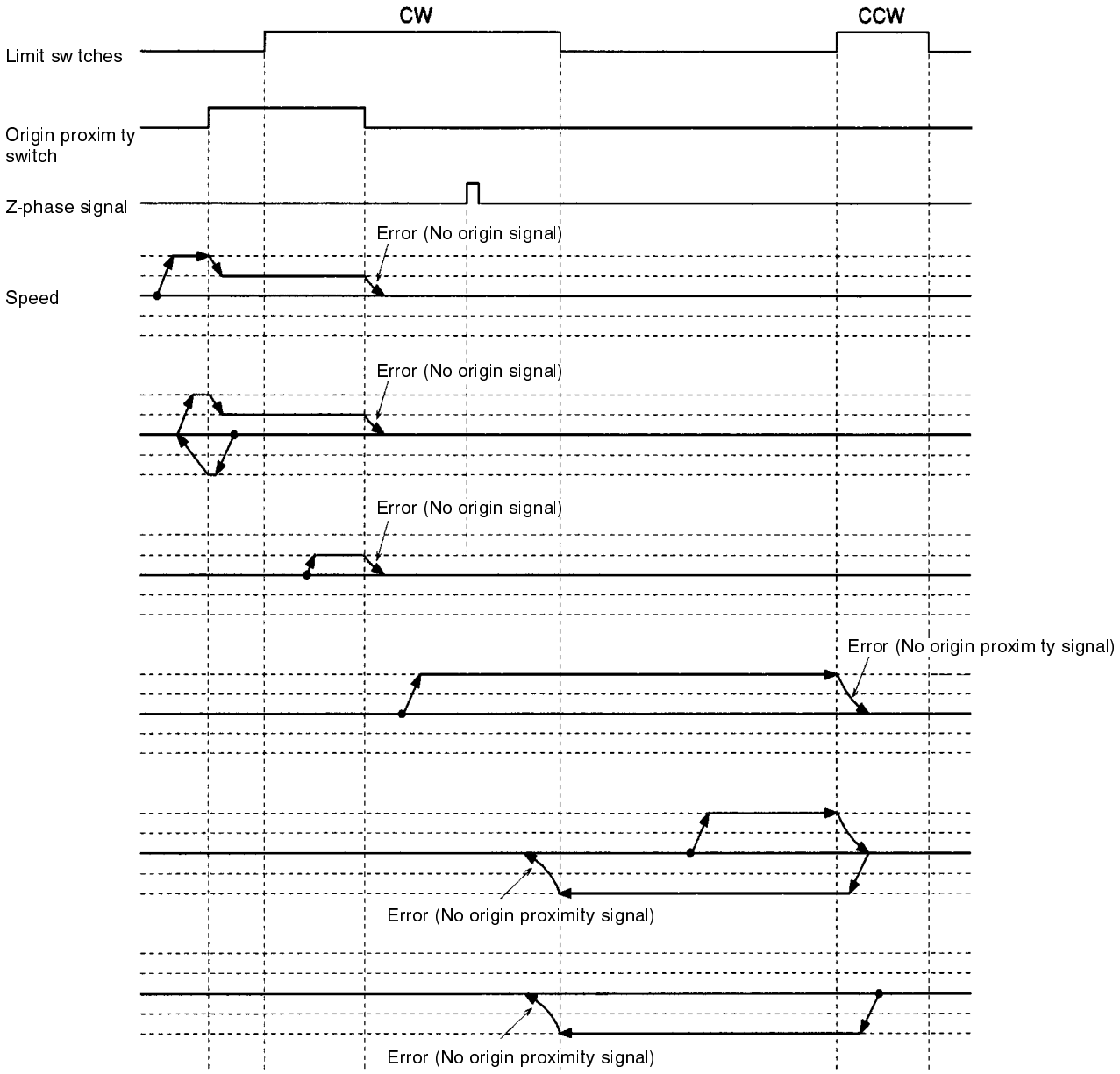
Reverse-mode Origin Searches 5

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW.



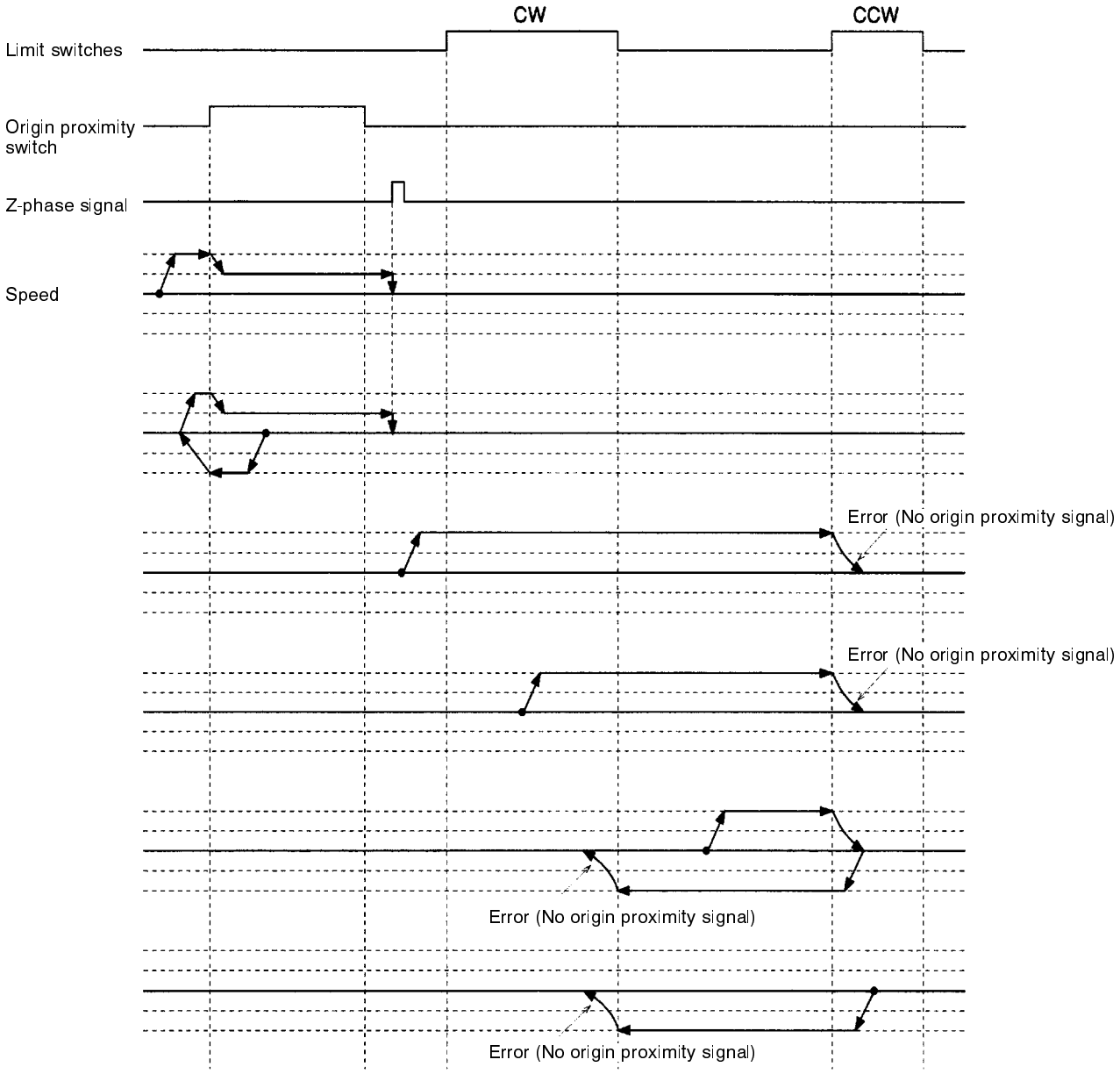
Reverse-mode Origin Searches 6

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW.



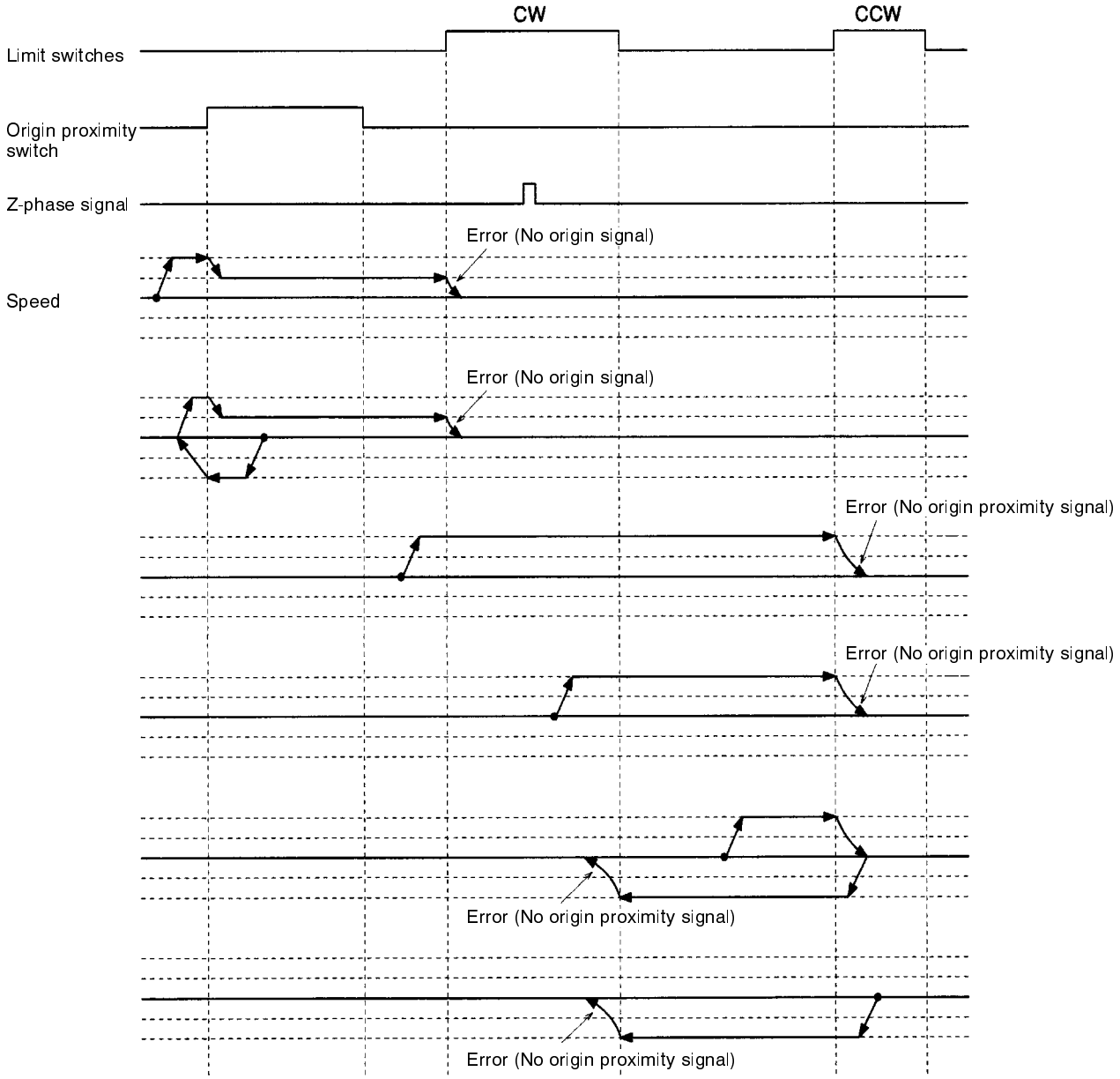
Reverse-mode Origin Searches 7

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW.



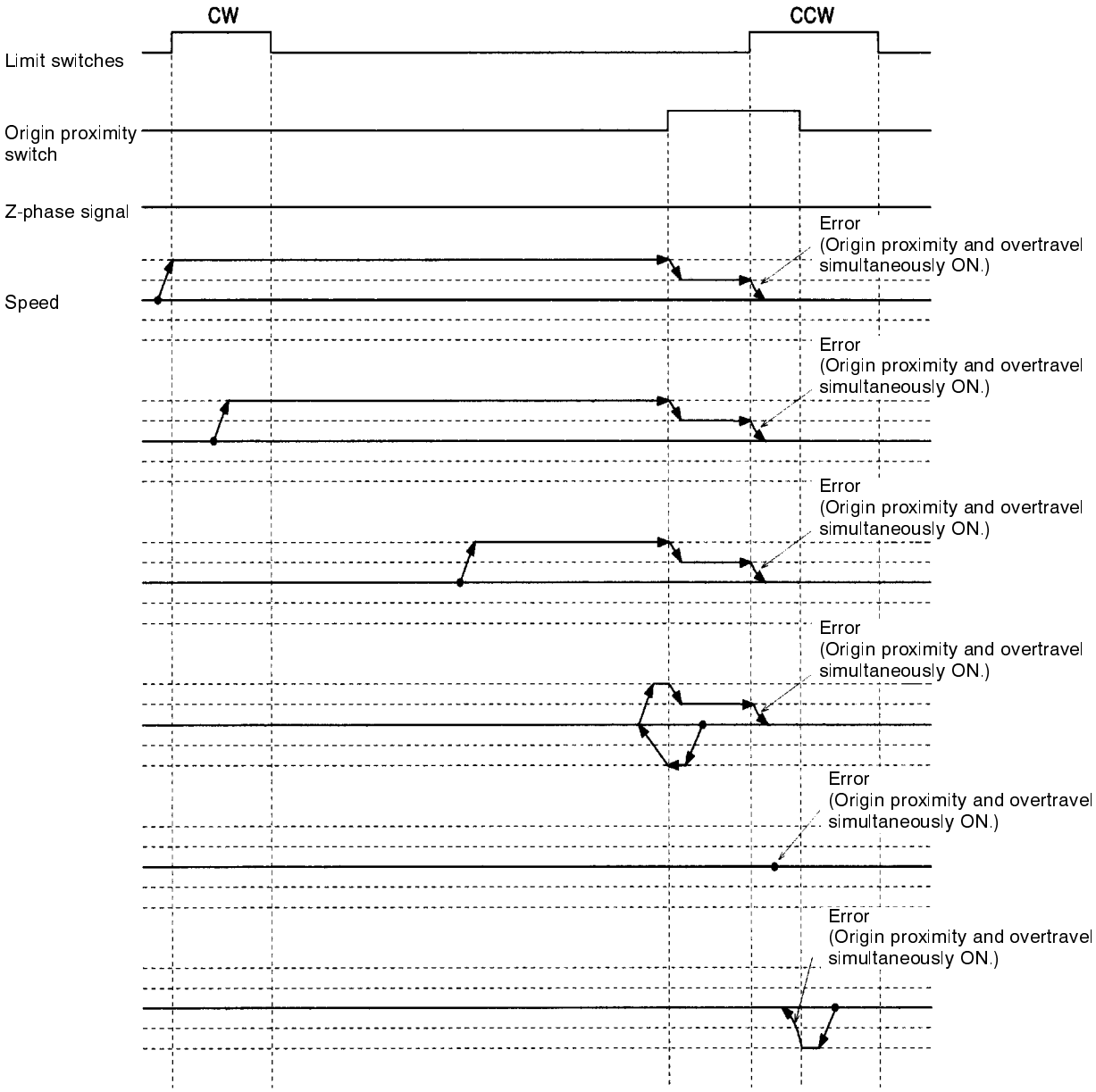
Reverse-mode Origin Searches 8

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW.



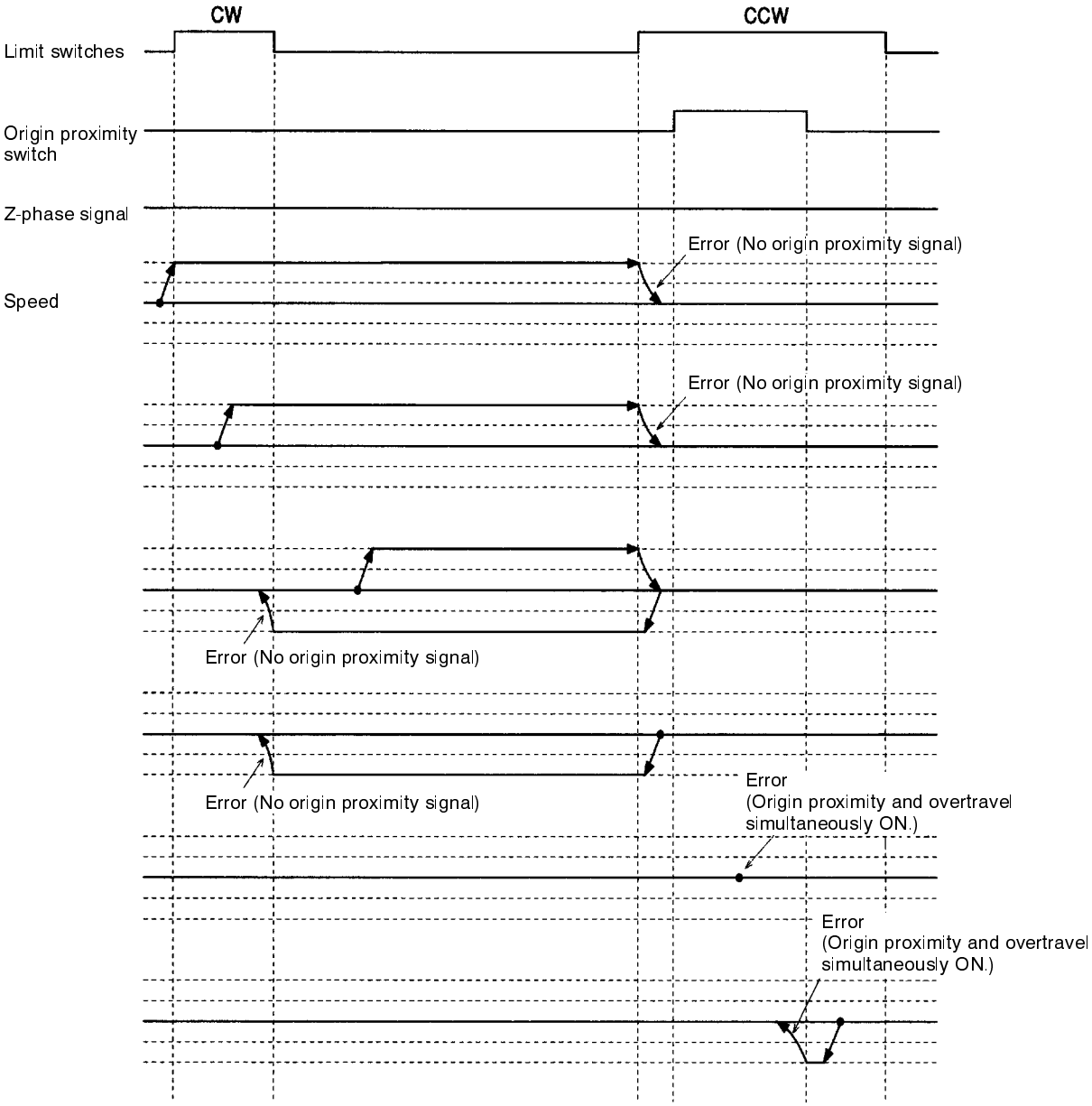
Reverse-mode Origin Searches 9

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW, but no Z-phase signal.



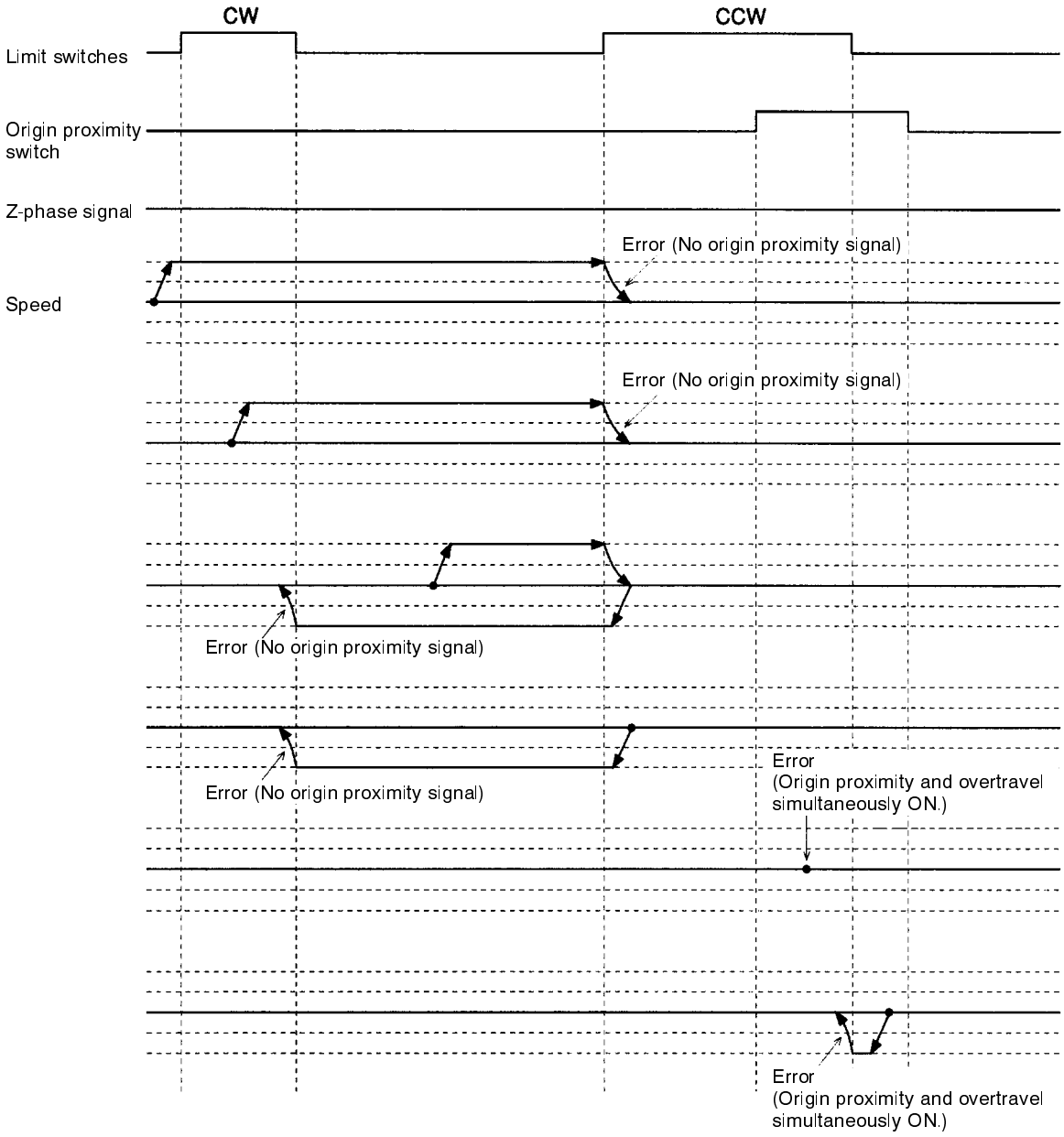
Reverse-mode Origin Searches 10

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW, but no Z-phase signal.



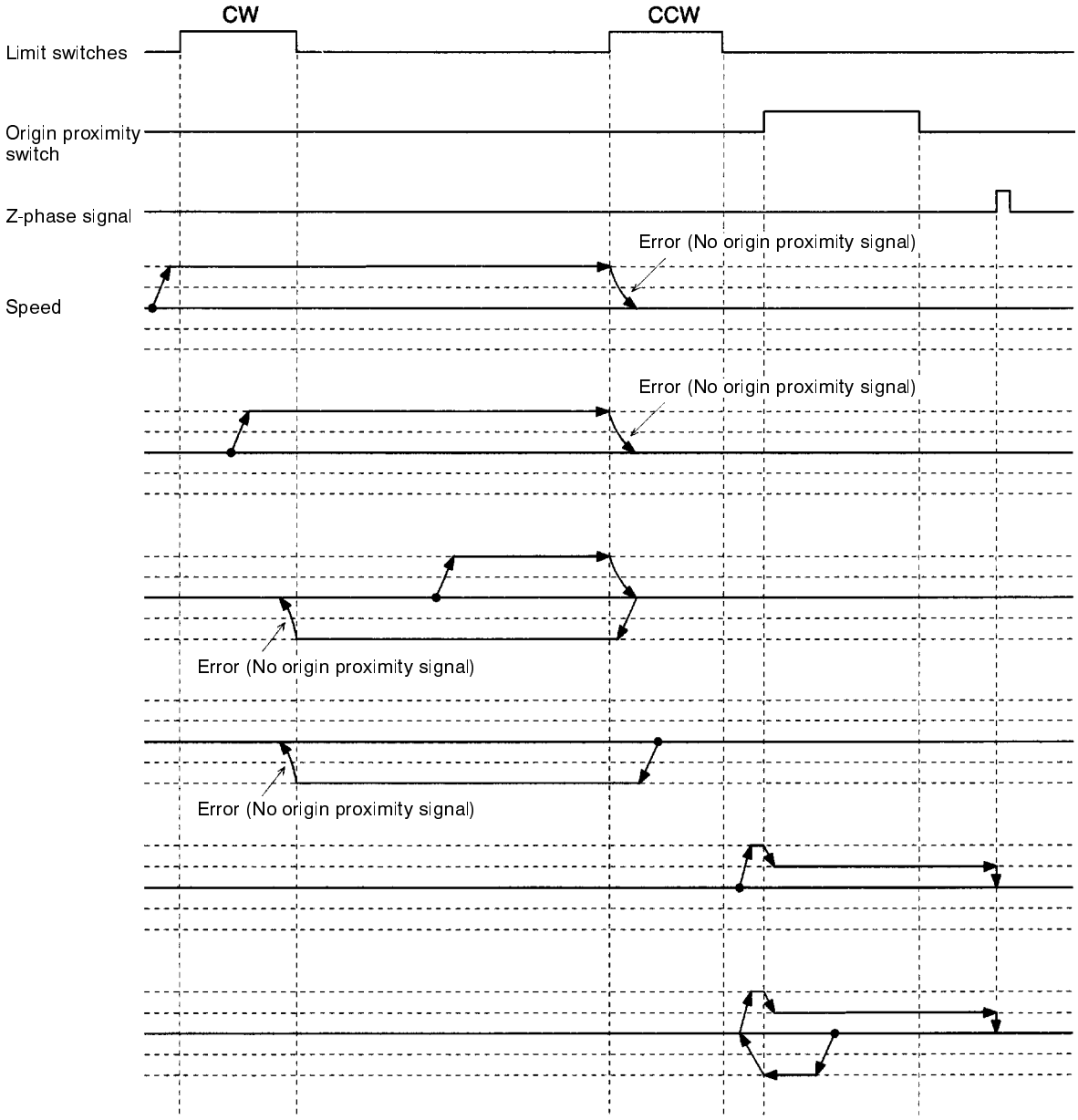
Reverse-mode Origin Searches 11

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW, but no Z-phase signal.

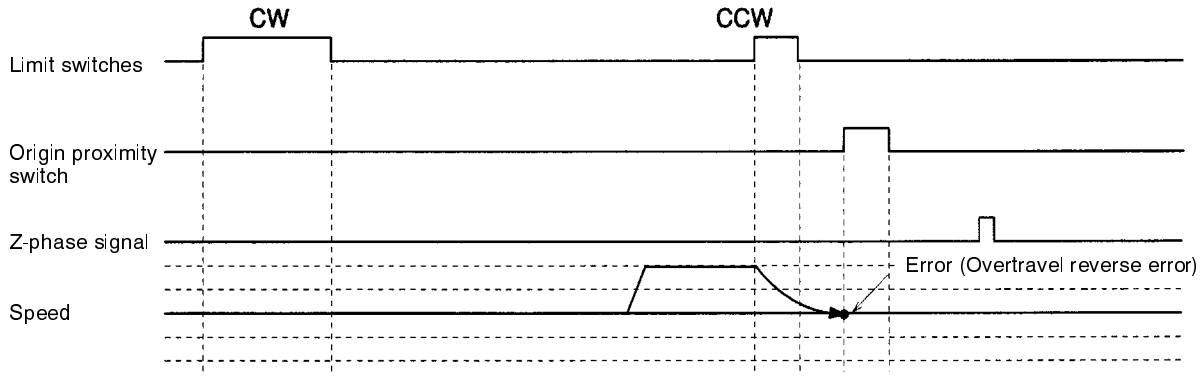


Reverse-mode Origin Searches 12

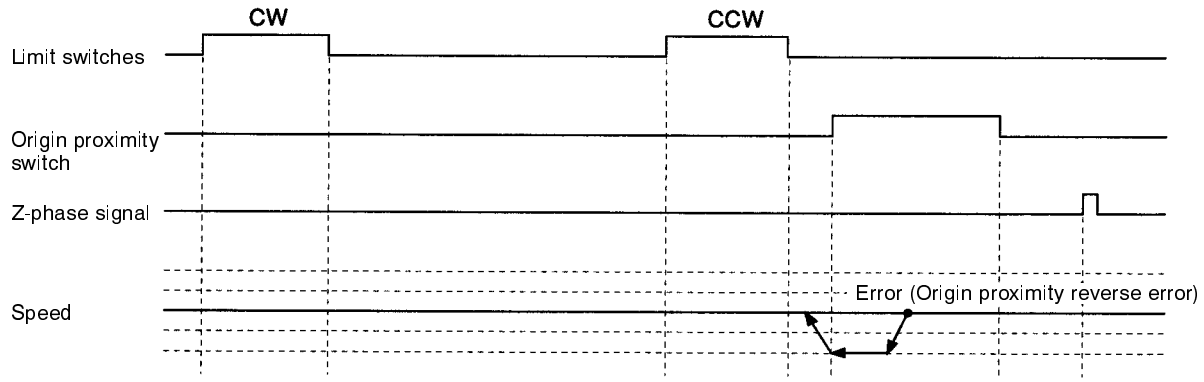
These origin searches are performed with an origin proximity switch and the initial search direction set to CCW.



The following error can occur if the CCW limit switch is ON for a short time and the CCW limit switch signal is close to the origin proximity signal.

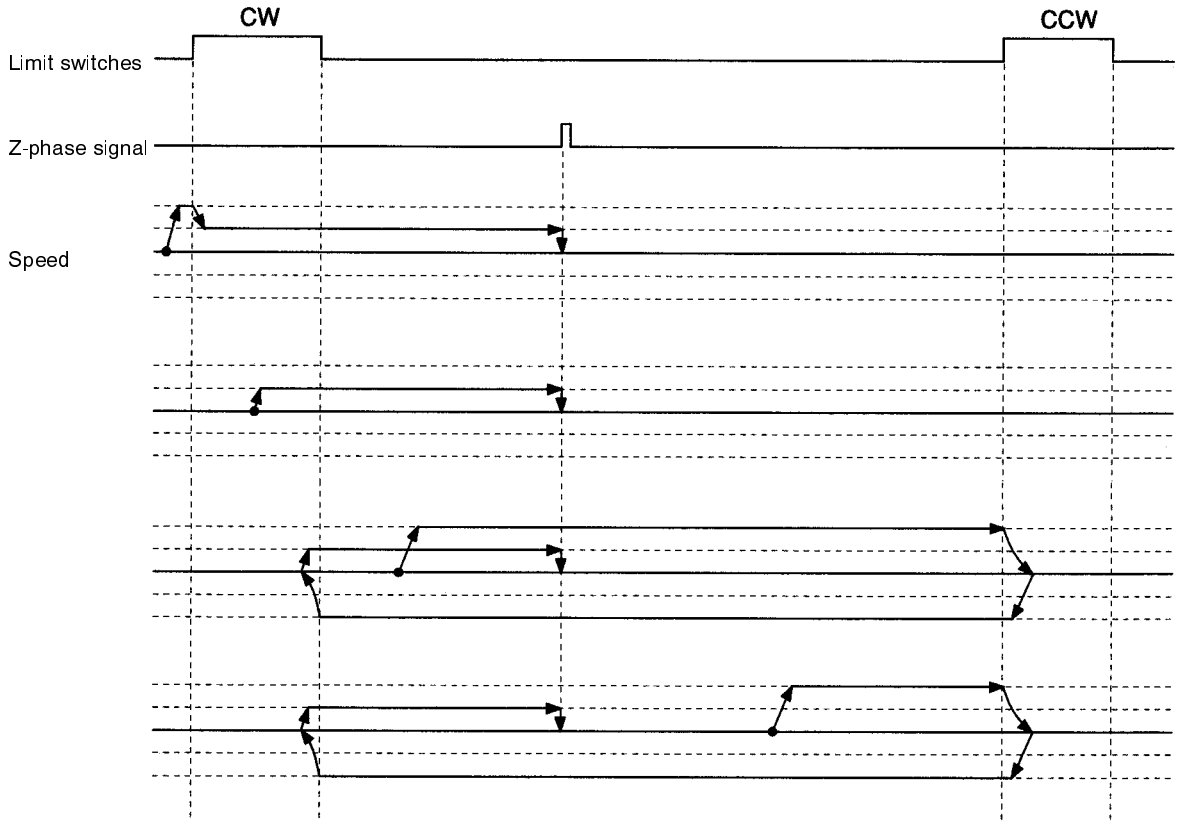


The following error can occur if the CCW limit switch signal is too close to the origin proximity signal.



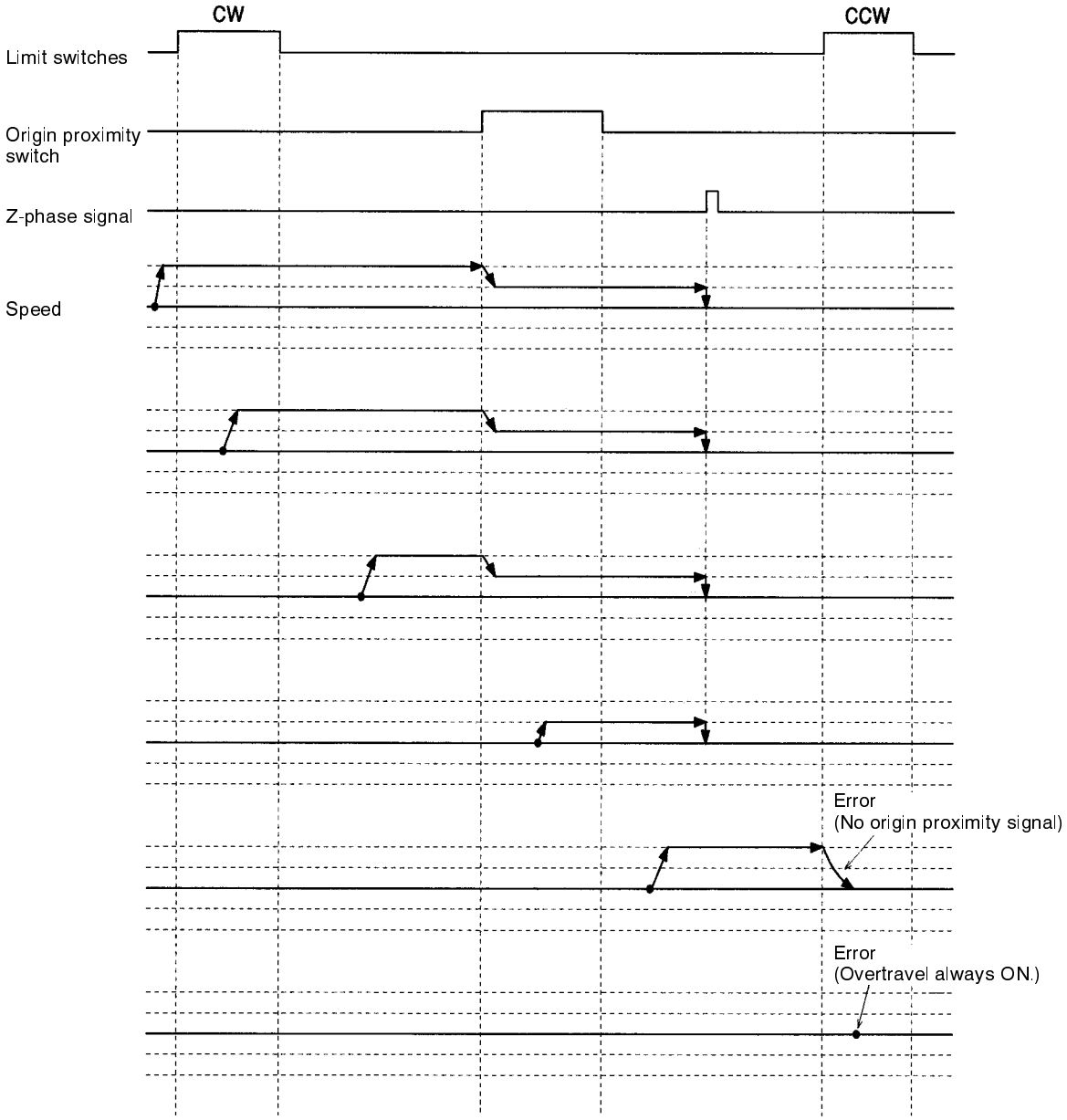
Reverse-mode Origin Searches 13

These origin searches are performed without an origin proximity switch and the initial search direction set to CCW.



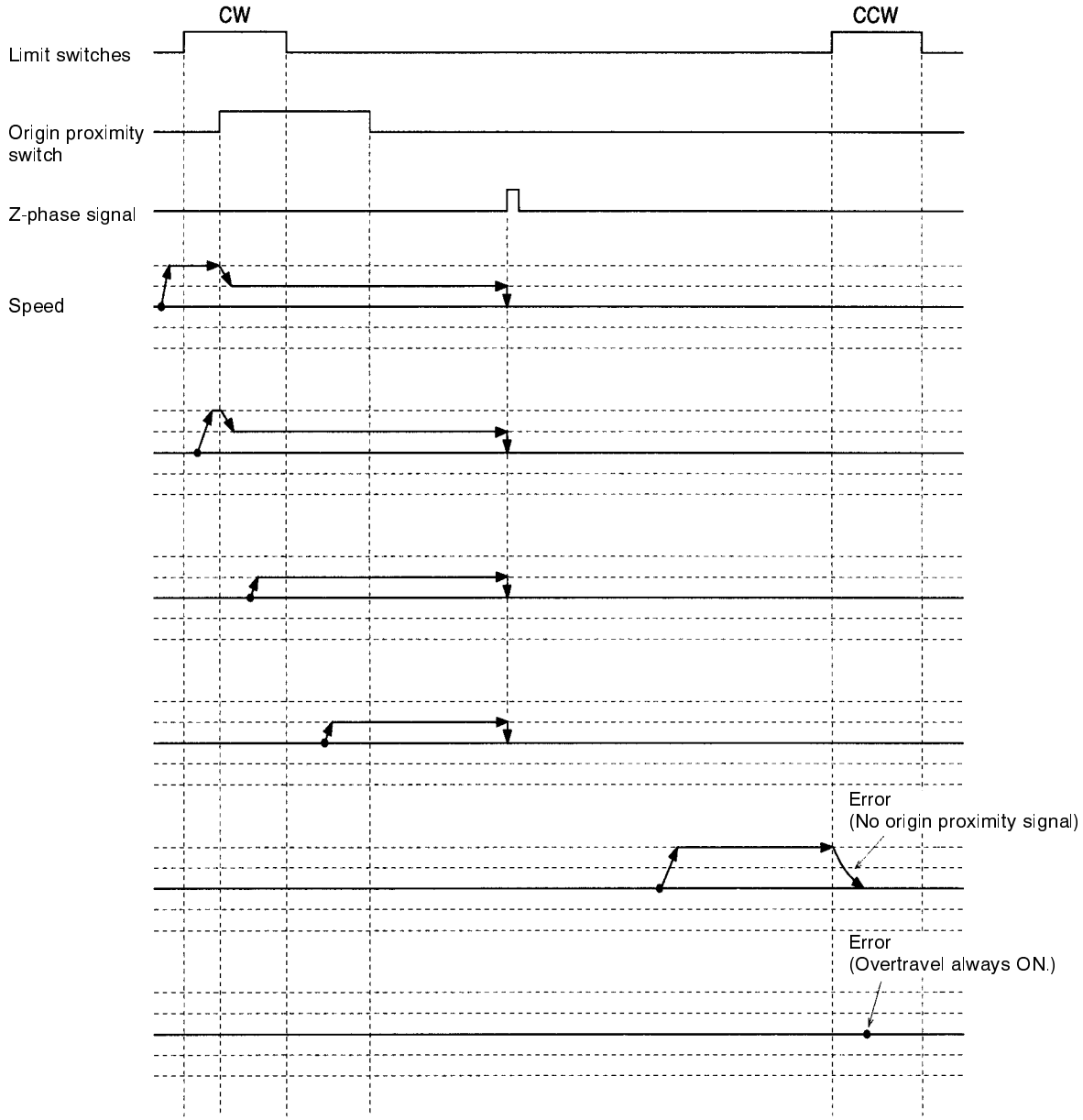
One Direction-mode Origin Searches 1

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW.



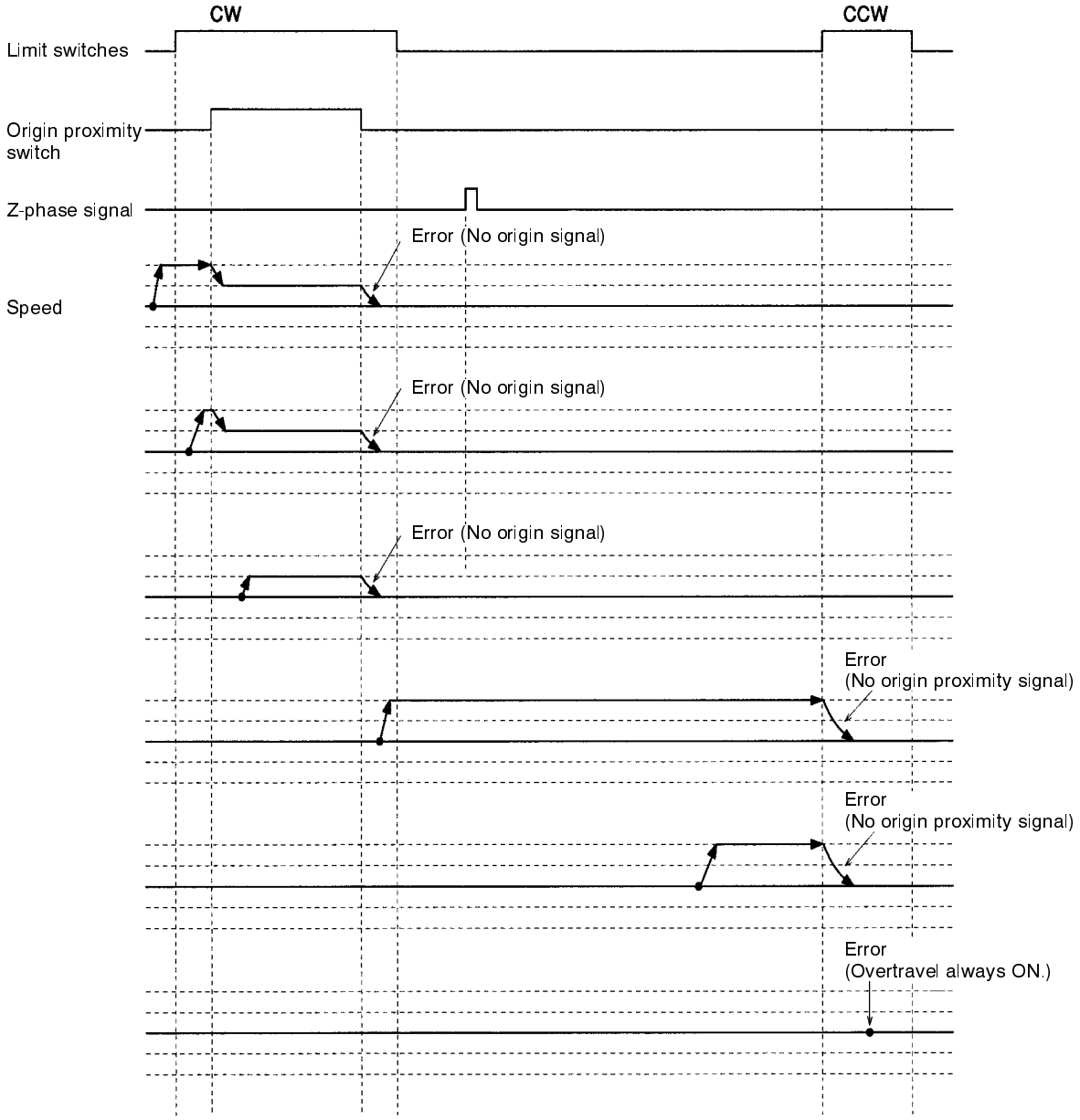
One Direction-mode Origin Searches 2

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW.



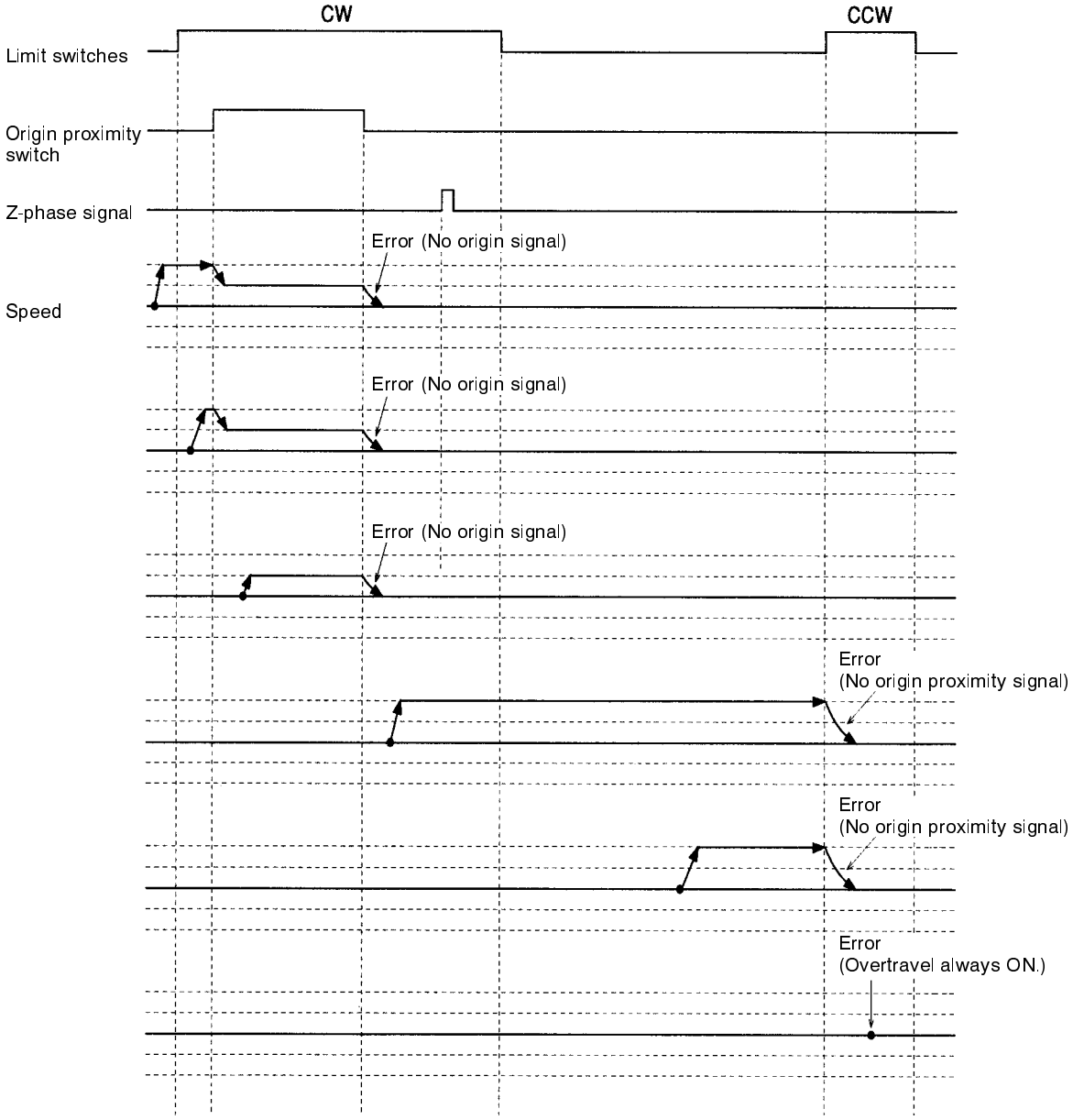
One Direction-mode Origin Searches 3

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW.



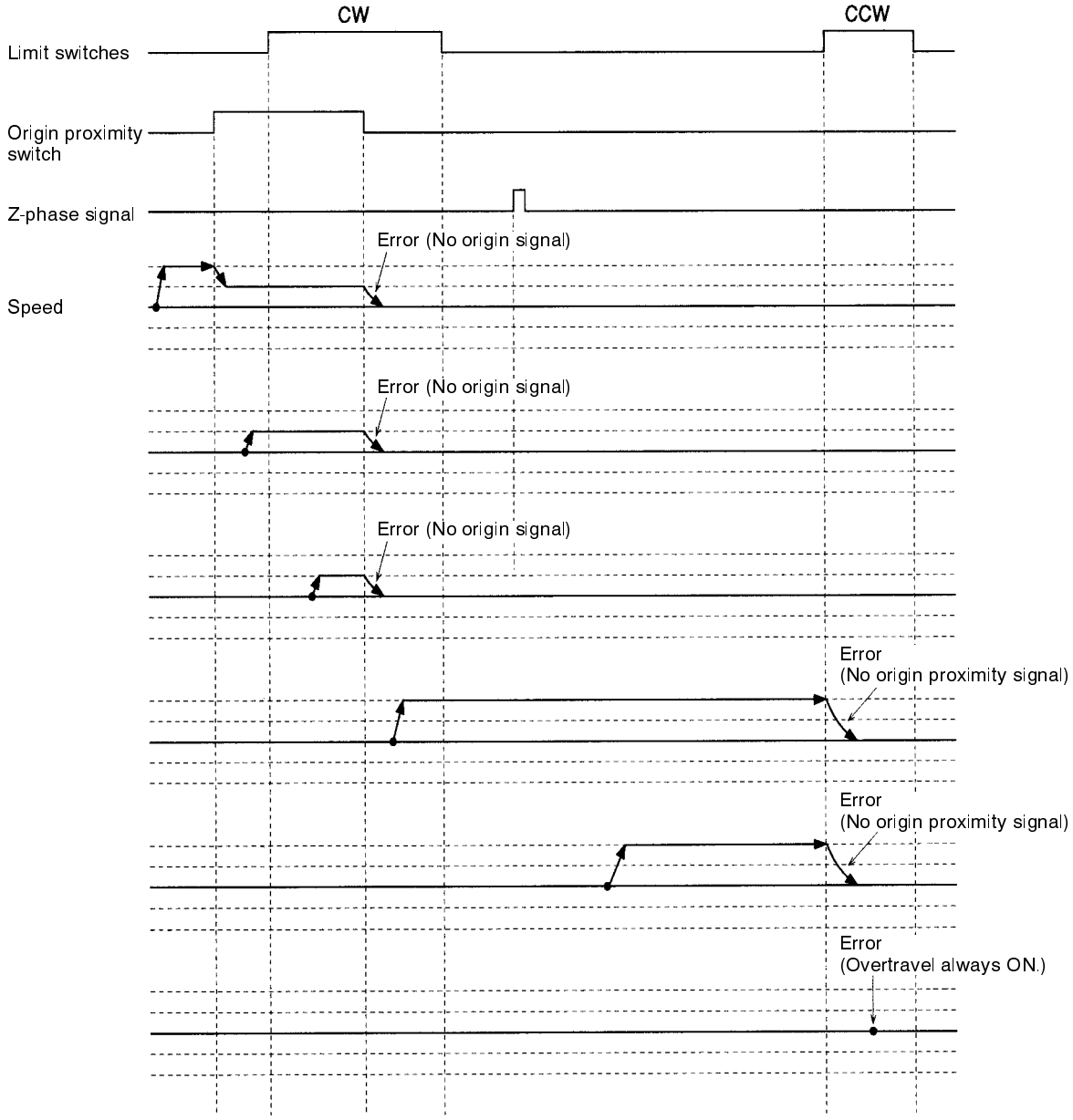
One Direction-mode Origin Searches 4

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW.



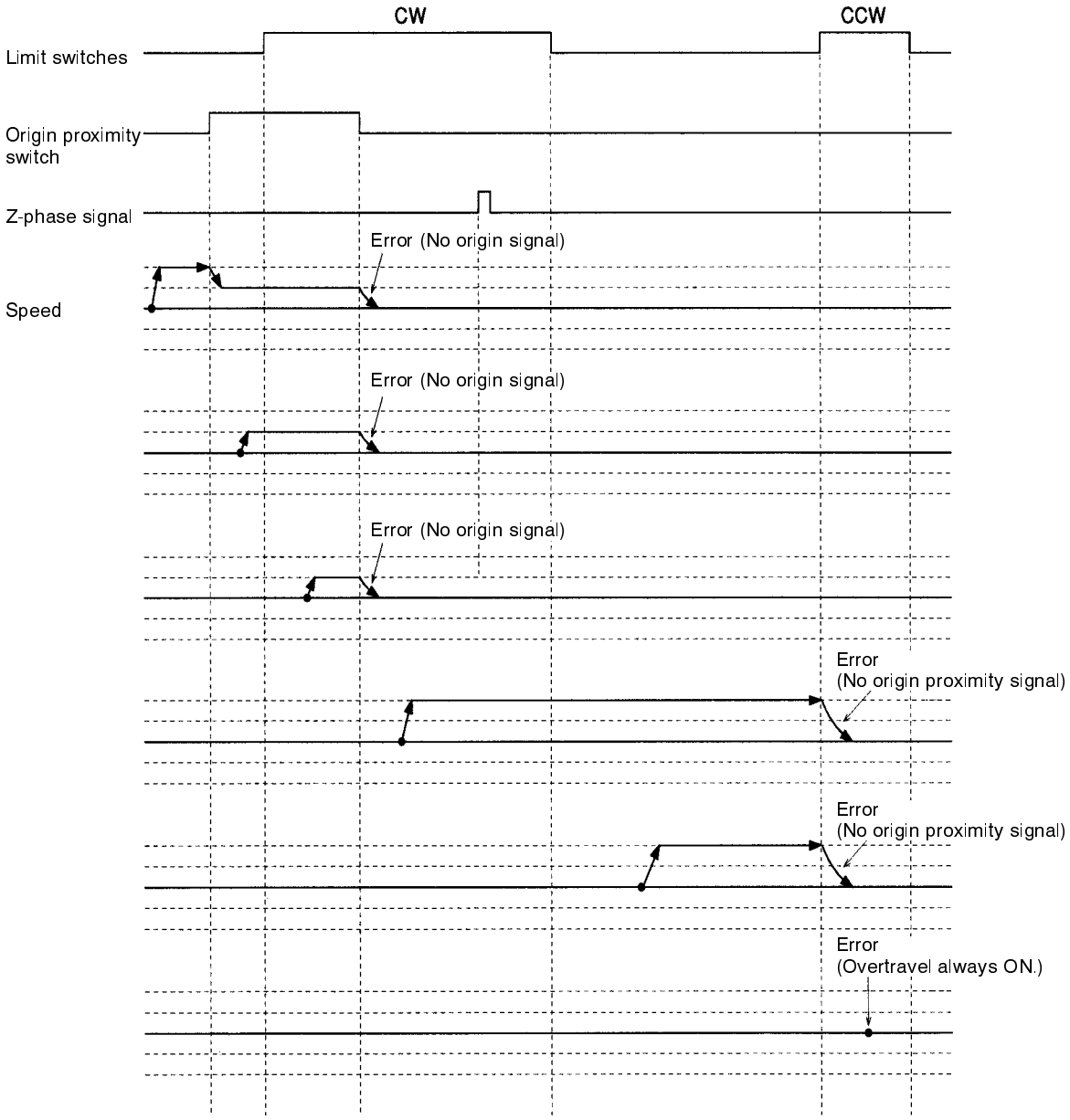
One Direction-mode Origin Searches 5

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW.



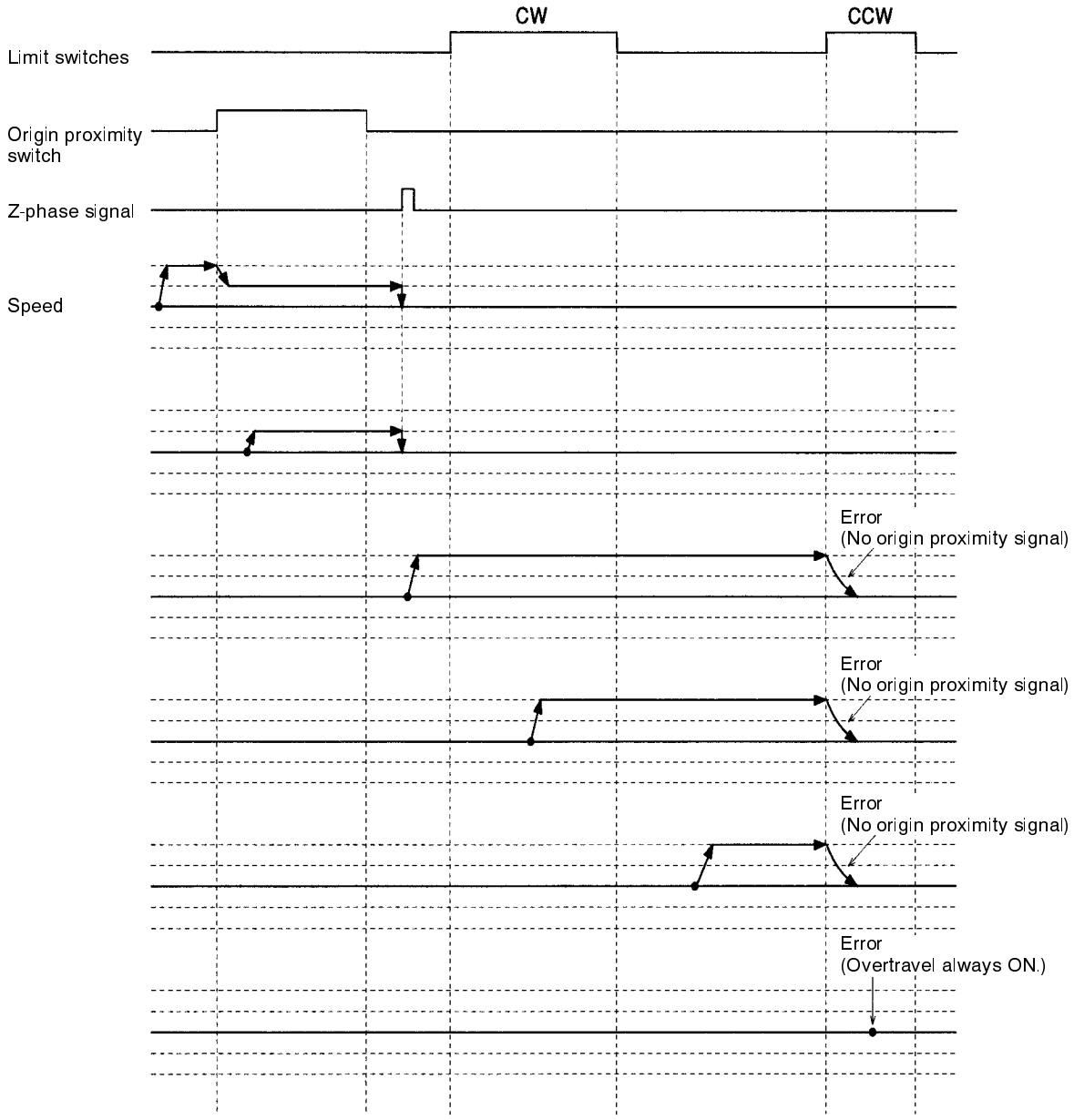
One Direction-mode Origin Searches 6

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW.



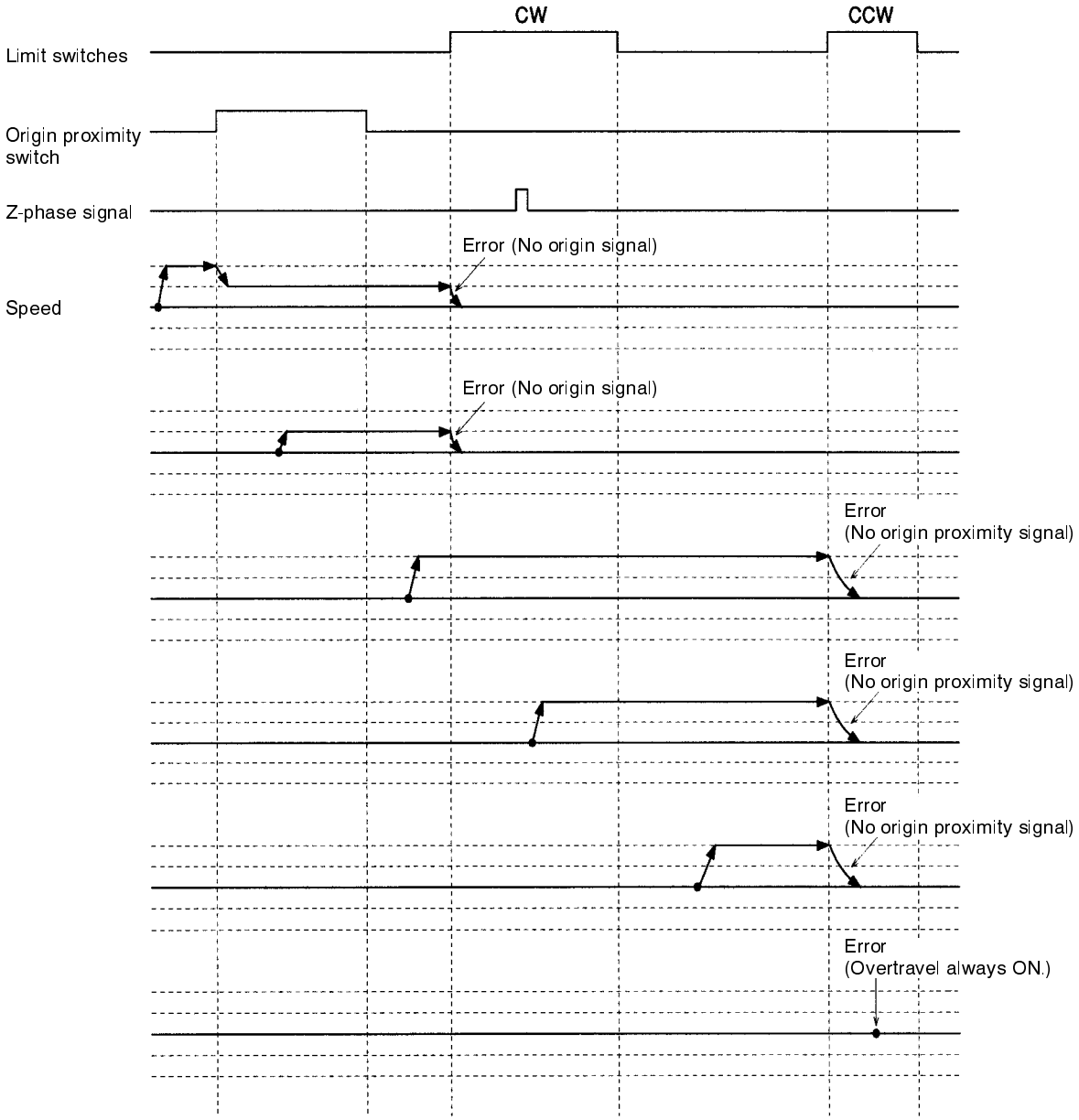
One Direction-mode Origin Searches 7

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW.



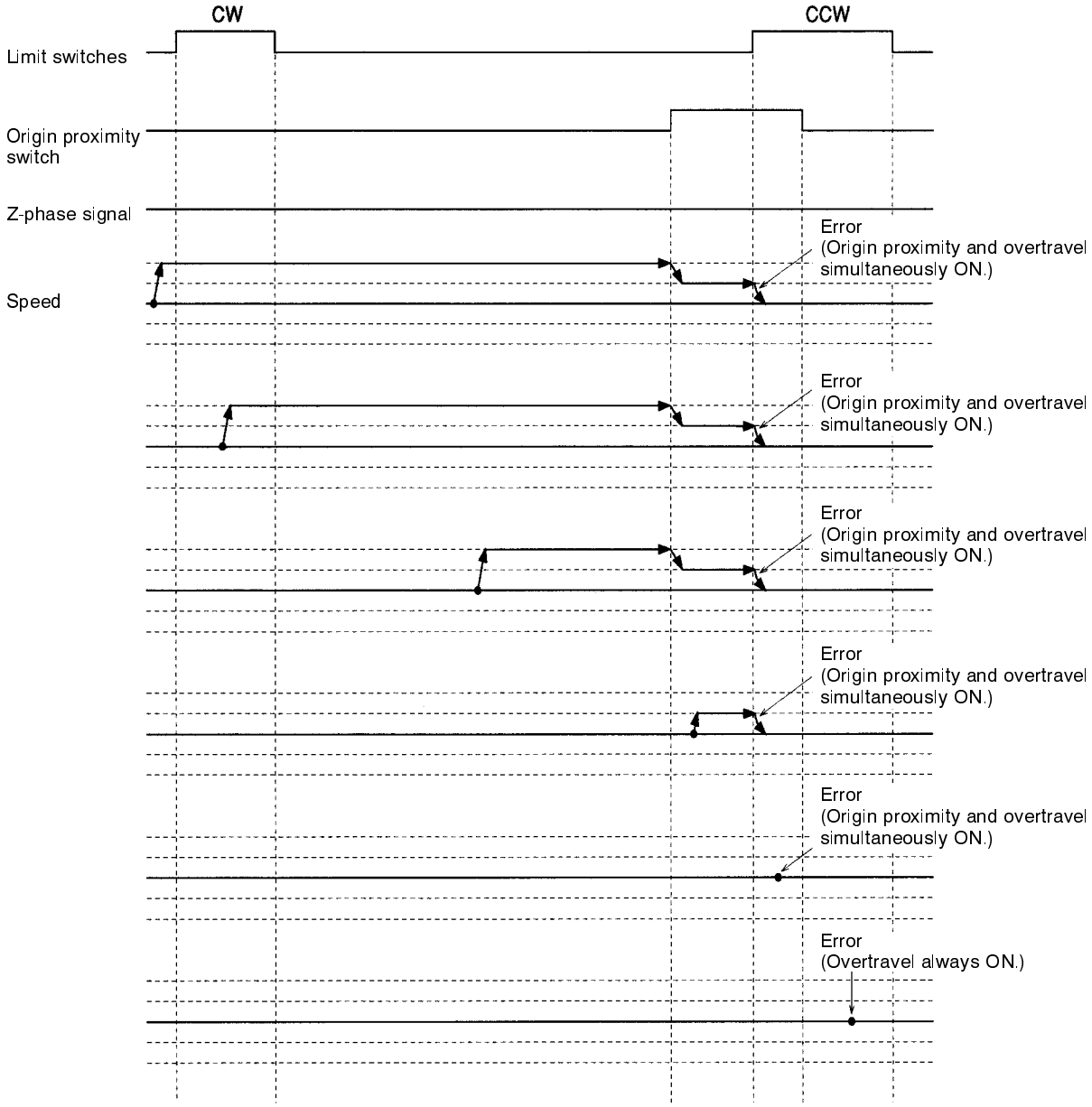
One Direction-mode Origin Searches 8

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW.



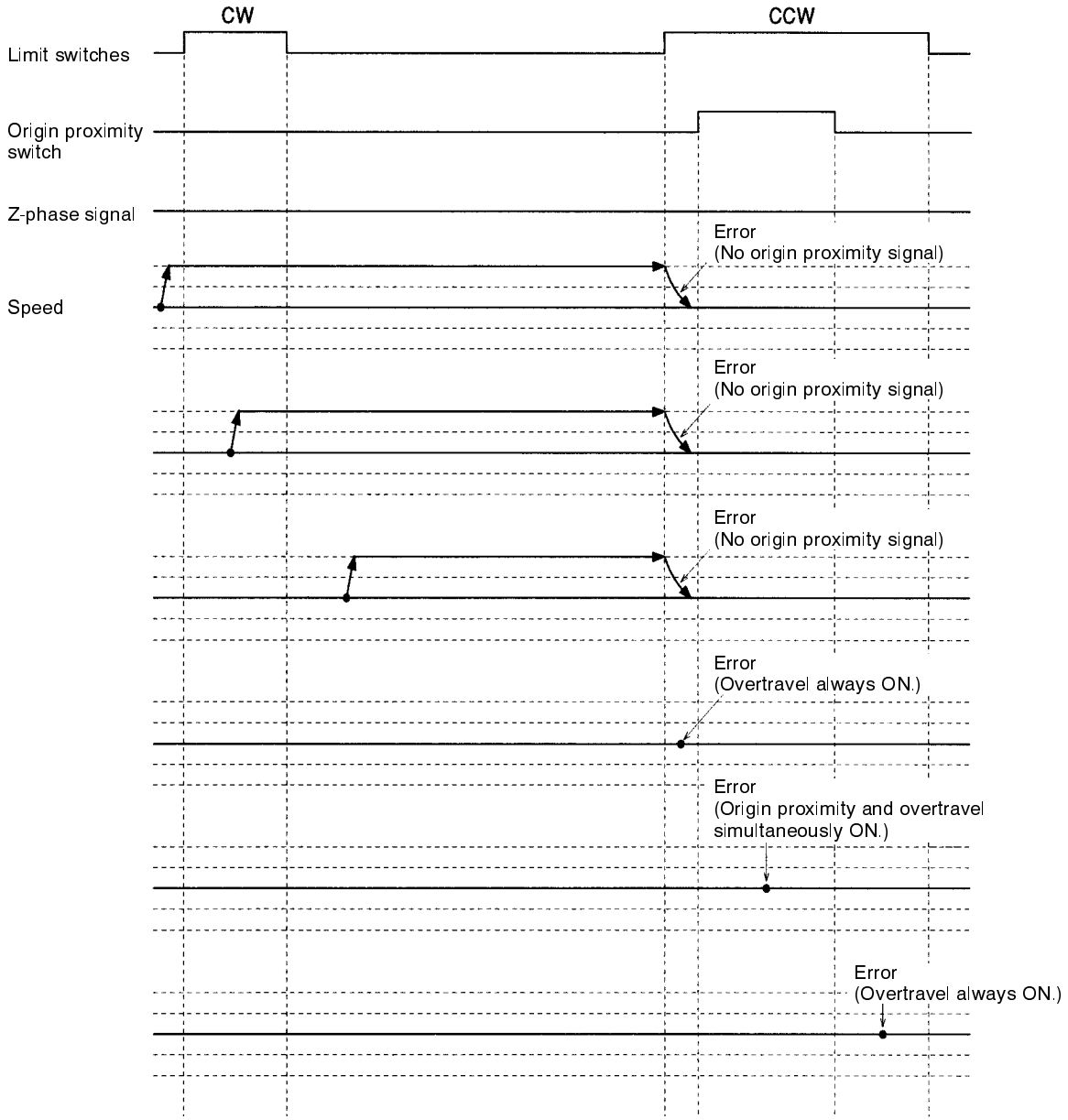
One Direction-mode Origin Searches 9

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW.



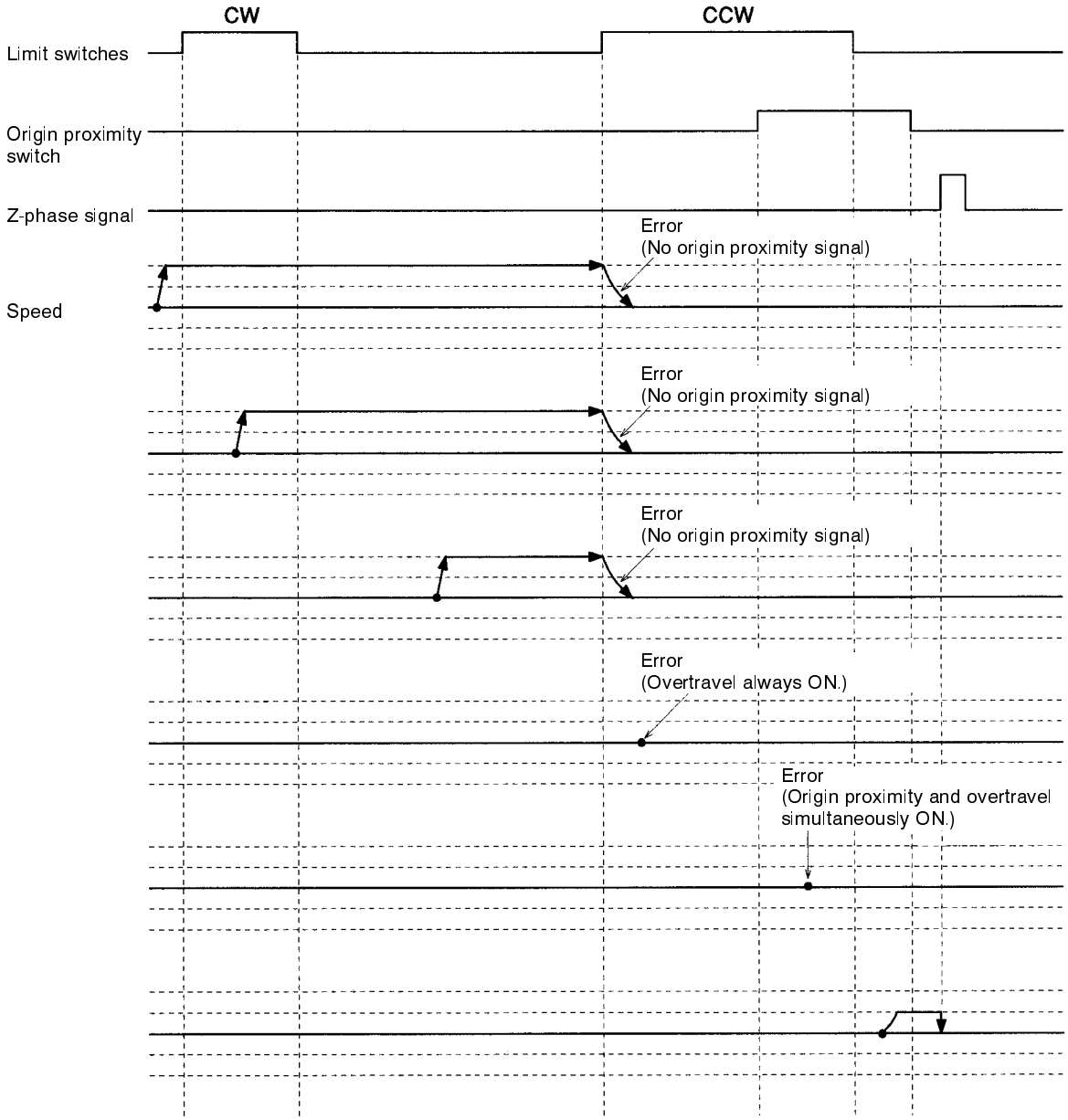
One Direction-mode Origin Searches 10

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW, but no Z-phase signal.



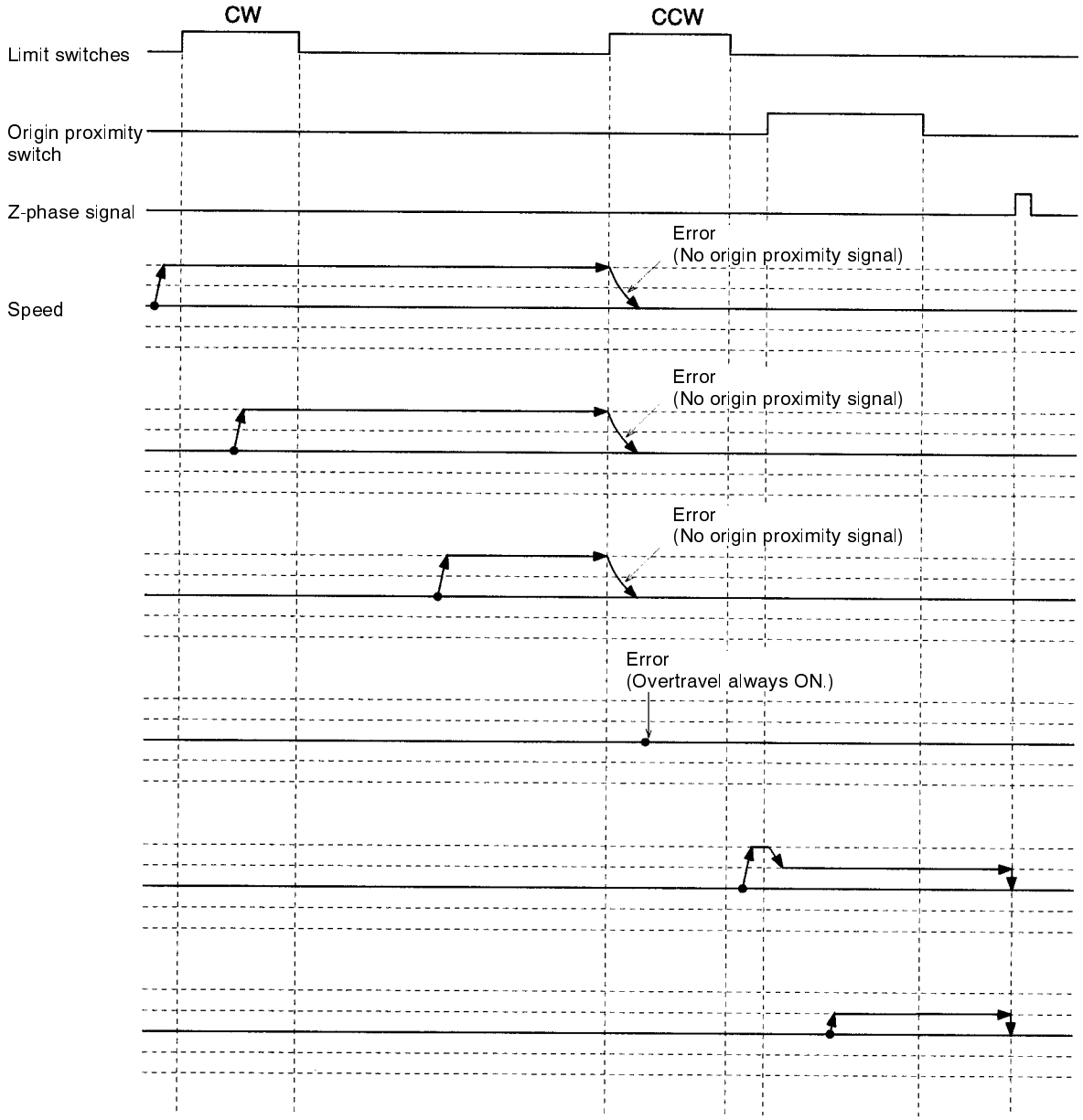
One Direction-mode Origin Searches 11

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW.



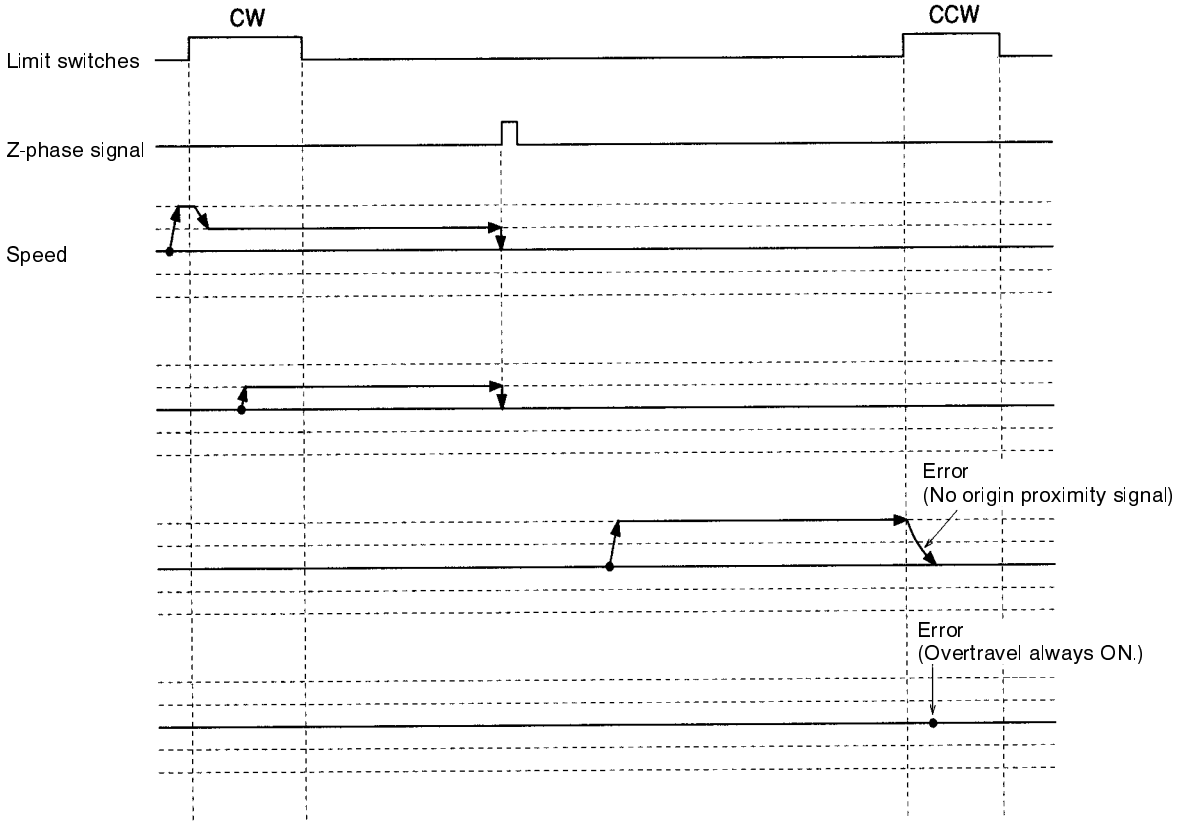
One Direction-mode Origin Searches 12

These origin searches are performed with an origin proximity switch and the initial search direction set to CCW.



One Direction-mode Origin Searches 13

These origin searches are performed without an origin proximity switch and the initial search direction set to CCW.



Appendix C
MC Program Coding Sheet

The following page can be copied for use in coding MC programs. When coding programs, be sure to specify all G codes and operands. These will be necessary when inputting programs.

Programmer: **Program Number:** **Date:** **Page:**

Block No.	G Code	Operands	Comments
N 0			
N 1			
N 2			
N 3			
N 4			
N 5			
N 6			
N 7			
N 8			
N 9			
N 0			
N 1			
N 2			
N 3			
N 4			
N 5			
N 6			
N 7			
N 8			
N 9			
N 0			
N 1			
N 2			
N 3			
N 4			
N 5			
N 6			
N 7			
N 8			
N 9			
N 0			
N 1			
N 2			
N 3			
N 4			
N 5			
N 6			
N 7			
N 8			
N 9			
N 0			
N 1			
N 2			
N 3			
N 4			
N 5			
N 6			
N 7			
N 8			
N 9			

Appendix D
MC Parameter Settings

Programmer: **Program Number:** **Date:** **Page:**

Unit Parameters

Parameter	Setting	Setting Range
Number of axes		1 to 4
Number of tasks		1 to 4
Task 1 axes		Any combination of X, Y, Z, and U
Task 2 axes		
Task 3 axes		
Task 4 axes		

Memory Parameters

Parameter	Setting		Setting Range
	Start address	End address	
Task 1 position data			0000 to 1999
Task 2 position data			
Task 3 position data			
Task 4 position data			

Machine Parameters

Parameter	Setting	Setting Range
Minimum setting unit		1, 0.1, 0.01, 0.001, or 0.0001
Display unit		mm, inches, degrees, or pulses
Rotate direction		---
Emergency stop method		---
Encoder resolution		1 to 65,535 ppr
Encoder polarity		---
Pulse rate		1 to 100,000
Maximum motor speed		1 to 32,767 rpm
Negative software limit		–39,999,999 to +39,999,999
Positive software limit		
Origin search method		---
Origin search direction		---
Origin decel. method		---
Origin proximity logic		N.O. or N.C.
Wiring check ON/OFF		---
Wiring check time		0 to 99 (×10 ms)
Wiring check pulses		0 to 999 pulses

Coordinate Parameters

Parameter	Setting			
	X axis	Y axis	Z axis	U axis
Reference origin offset				
Workpiece origin offset				

Programmer: **Program Number:** **Date:** **Page:**

Feedrate Parameters

Parameter	Setting				Setting Range
	X axis	Y axis	Z axis	U axis	
Max. high-speed feed rate					1 to 39,999,999
Max. interpolation feed rate					
Origin search high speed					
Origin search low speed					
Max. jog feed rate					
Accel./Decel. curve					---
Acceleration time					0 to 9,999 ms
Deceleration time					
Interpolation accel. time					
Interpolation decel. time					
MPG ratio 1					1 to 1,000
MPG ratio 2					
MPG ratio 3					
MPG ratio 4					

Zone Parameters

Parameter		Setting
Zone 1 specification		
Zone 1	Negative SV	
	Positive SV	
Zone 2 specification		
Zone 2	Negative SV	
	Positive SV	
Zone 3 specification		
Zone 3	Negative SV	
	Positive SV	
Zone 4 specification		
Zone 4	Negative SV	
	Positive SV	
Zone 5 specification		
Zone 5	Negative SV	
	Positive SV	
Zone 6 specification		
Zone 6	Negative SV	
	Positive SV	
Zone 7 specification		
Zone 7	Negative SV	
	Positive SV	
Zone 8 specification		
Zone 8	Negative SV	
	Positive SV	

Servo Parameters

Parameter	Setting				Setting Range
	X axis	Y axis	Z axis	U axis	
Error counter warning					0 to 65,000 pulses
In position					0 to 999 pulses
Position loop gain					5 to 150 (1/s)
Position loop FF gain					0 to 100 (%)

Appendix E
Position Data Coding Sheet

The following page can be copied to record the data stored in position data addresses.

Programmer: **Program Number:** **Date:** **Page:**

Address	Data	Comments
A 00		
A 01		
A 02		
A 03		
A 04		
A 05		
A 06		
A 07		
A 08		
A 09		
A 10		
A 11		
A 12		
A 13		
A 14		
A 15		
A 16		
A 17		
A 18		
A 19		
A 20		
A 21		
A 22		
A 23		
A 24		
A 25		
A 26		
A 27		
A 28		
A 29		
A 30		
A 31		
A 32		
A 33		
A 34		
A 35		
A 36		
A 37		
A 38		
A 39		
A 40		
A 41		
A 42		
A 43		
A 44		
A 45		
A 46		
A 47		
A 48		
A 49		
A 50		

Address	Data	Comments
A 51		
A 52		
A 53		
A 54		
A 55		
A 56		
A 57		
A 58		
A 59		
A 60		
A 61		
A 62		
A 63		
A 64		
A 65		
A 66		
A 67		
A 68		
A 69		
A 70		
A 71		
A 72		
A 73		
A 74		
A 75		
A 76		
A 77		
A 78		
A 79		
A 80		
A 81		
A 82		
A 83		
A 84		
A 85		
A 86		
A 87		
A 88		
A 89		
A 90		
A 91		
A 92		
A 93		
A 94		
A 95		
A 96		
A 97		
A 98		
A 99		

Glossary

absolute position — A position given in respect to the origin rather than in respect to the present position.

acceleration/deceleration curve — Curves which determine the rate of acceleration to the maximum feed rate and the rate of deceleration from the maximum feed rate.

auxiliary bit — A bit in the Auxiliary Area.

Backplane — A base to which Units are mounted to form a Rack. Backplanes provide a series of connectors for these Units along with buses to connect them to the CPU and other Units and wiring to connect them to the Power Supply Unit. Backplanes also provide connectors used to connect them to other Backplanes.

basic instruction — A fundamental instruction used in a ladder diagram. See *advanced instruction*.

baud rate — The data transmission speed between two devices in a system measured in bits per second.

BCD — Short for binary-coded decimal.

binary — A number system where all numbers are expressed in base 2, i.e., numbers are written using only 0's and 1's. Each group of four binary bits is equivalent to one hexadecimal digit. Binary data in memory is thus often expressed in hexadecimal for convenience.

bit — The smallest piece of information that can be represented on a computer. A bit has the value of either zero or one, corresponding to the electrical signals ON and OFF. A bit represents one binary digit. Some bits at particular addresses are allocated to special purposes, such as holding the status of input from external devices, while other bits are available for general use in programming.

block number — Numbers used to distinguish blocks in MC programs. Block numbers are roughly equivalent to program line numbers.

bus — A communications path used to pass data between any of the Units connected to it.

channel — See *word*.

CIO Area — A memory area used to control I/O and to store and manipulate data. CIO Area addresses do not require prefixes.

comment block — A program block that contains comments input by the programmer. Comment blocks and program blocks share the same block numbers, but comment blocks begin with an asterisk rather than an "N."

control bit — A bit in a memory area that is set either through the program or via a Programming Device to achieve a specific purpose, e.g., a Restart Bit is turned ON and OFF to restart a Unit.

counter — A dedicated group of digits or words in memory used to count the number of times a specific process has occurred, or a location in memory accessed through a TC bit and used to count the number of times the status of a bit or an execution condition has changed from OFF to ON.

CPU The name of the Unit in a PC that contains the main CPU and other main PC components. See also *central processing unit*.

CPU Backplane A Backplane used to create a CPU Rack.

CPU Bus Unit A special Unit used with CV-series PCs that mounts to the CPU bus. This connection to the CPU bus enables special data links, data transfers, and processing.

CPU Bus Unit Area A part of the CIO Area allocated to CPU Bus Units. The use of the words and bits in this area is determined by the Unit to which they are allocated.

CPU Rack The main Rack in a building-block PC, the CPU Rack contains the CPU, a Power Supply, and other Units. The CPU Rack, along with the Expansion CPU Rack, provides both an I/O bus and a CPU bus.

CV-mode A form of communications useable only with CV-series PCs. See *C-mode*.

CV-series PC Any of the following PCs: CV500, CV1000, CV2000, or CVM1

CVSS See *CV Support Software*.

CW and CCW Abbreviations for clockwise (CW) and counterclockwise (CCW). CW and CCW are defined for a motor shaft in reference to a viewer facing the shaft on the end of the motor from which the shaft extends from the motor for connection.

CW/CCW limits Limits on the CW and CCW sides of the origin which can be internally set to restrict rotation of the shaft.

data area An area in the PC's memory that is designed to hold a specific type of data.

decimal A number system where numbers are expressed to the base 10. In a PC all data is ultimately stored in binary form, four binary bits are often used to represent one decimal digit, via a system called binary-coded decimal.

dwell time A setting that specifies the period of time during which positioning will stop before execution of the next positioning action.

error counter A counter used to ensure positioning accuracy when positioning via pulse trains. The error counter receives a target position as a specific number of pulses in a pulse train from the Motion Control Unit and outputs analog speed voltages to drive a servomotor accordingly. The specified number of pulses in the error counter is counted down by feedback from an encoder measuring actual motor shaft movement, causing voltage output to stop when the number of pulses equals zero, i.e., when the target position has been reached.

feedback The return of a portion of the output of a circuit or device to its input. It is used in servocontrol systems to help bring actual values closer to target values.

flag A dedicated bit in memory that is set by the system to indicate some type of operating status. Some flags, such as the carry flag, can also be set by the operator or via the program.

gain The increase in signal power produced by an amplifier.

G language A programming language used widely in position control. Program functions are entered simply by entering a "G," a 2-digit numerical code, and adding any needed parameters.

hunting The tendency, in servosystems, to overcompensate when the system's momentum carries it past the target position.

IBM PC/AT or compatible A computer that has similar architecture to, that is logically compatible with, and that can run software designed for an IBM PC/AT computer.

inching Manual feeding wherein positioning is executed one pulse at a time.

incremental position A position given in respect to the present position, rather than in respect to the origin.

initial position The present position when a start command is executed.

in position The range within which the system is determined to be at the target position.

input The signal coming from an external device into the PC. The term input is often used abstractly or collectively to refer to incoming signals.

interpolation The mathematical calculation of missing values based pm known values. The Motion Control Unit uses interpolation when positioning along two or more axes simultaneously. There are three types of interpolation possible: linear, circular, and helical (a combination of linear and circular).

interface An interface is the conceptual boundary between systems or devices and usually involves changes in the way the communicated data is represented. Interface devices such as NSBs perform operations like changing the coding, format, or speed of the data.

least-significant (bit/word) See *rightmost (bit/word)*.

leftmost (bit/word) The highest numbered bits of a group of bits, generally of an entire word, or the highest numbered words of a group of words. These bits/words are often called most-significant bits/words.

linear interpolation Dual-axis, linear positioning from the present position to a point designated as the interpolation end point based on specified points.

load The processes of copying data either from an external device or from a storage area to an active portion of the system such as a display buffer. Also, an output device connected to the PC is called a load.

local In network communications, the node or device from which communications are being viewed. See *remote*.

LSS Abbreviation for Ladder Support Software.

M code An abbreviation for machine code. The user can set various M codes for various positions so that each M code will be output when the workpiece passes its respective position.

MC program A G-language program that controls the MC Unit's operation.

megabyte A unit of storage equal to one million bytes.

most-significant (bit/word) See *leftmost (bit/word).*

MS-DOS An operating system in common use on smaller computers.

NC contacts Normally-closed contacts. A pair of contacts on a relay that open when the relay is energized.

negative software limit The lower limit on the number of pulses set as a software parameter.

nesting Programming one loop within another loop, programming a call to a subroutine within another subroutine, or programming an IF–ELSE programming section within another IF–ELSE section.

NO contacts Normally-open contacts. A pair of contacts on a relay that close when the relay is energized.

OFF The status of an input or output when a signal is said not to be present. The OFF state is generally represented by a low voltage or by non-conductivity, but can be defined as the opposite of either.

offline The state in which a Programming Device is not functionally connected to the CPU, although it may be connected physically.

offset A positive or negative value added to a base value such as an address to specify a desired value.

ON The status of an input or output when a signal is said to be present. The ON state is generally represented by a high voltage or by conductivity, but can be defined as the opposite of either.

online The state in which a Programming Device is functionally connected to the CPU so that CPU data and programs can be monitored or accessed.

online edit An edit to a program made from a peripheral device connected to and currently online with a PC in PROGRAM or MONITOR mode. In MONITOR mode, this means that the program is changed while it is actually being executed.

origin proximity input A signal input to indicate that the axis is near the origin.

origin search An operation used to automatically move the axes to the origin or to define the origin.

output The signal sent from the PC to an external device. The term output is often used abstractly or collectively to refer to outgoing signals.

parameters Data which determines limits and other conditions under which an operation will be carried out.

PC An acronym for Programmable Controller.

PC Setup A group of operating parameters set in the PC from a Programming Device to control PC operation.

positive software limit The upper limit on the number of pulses set as a software parameter.

present value The current value registered in a device at any instant during its operation. Present value is abbreviated as PV. The use of this term is generally restricted to timers and counters.

program block A unit of programming in MC programs roughly equivalent to program lines.

Programmable Controller A computerized device that can accept inputs from external devices and generate outputs to external devices according to a program held in memory. Pro-

grammable Controllers are used to automate control of external devices. Although single-unit Programmable Controllers are available, building-block Programmable Controllers are constructed from separate components. Such Programmable Controllers are formed only when enough of these separate components are assembled to form a functional assembly, i.e., there is no one individual Unit called a PC.

Programming Device A Peripheral Device used to input a program into a PC or to alter or monitor a program already held in the PC. There are dedicated programming devices, such as Programming Consoles, and there are non-dedicated devices, such as a host computer.

pulses Discrete signals sent at a certain rate. The Motion Control Unit outputs pulses, each of which designates a certain amount of movement. Such pulses are converted to an equivalent control voltage in actual positioning.

pulse rate The distance moved the motor shaft divided by the number of pulses required for that movement.

pulse train A series of pulses output together.

remote In network communications, the node or device with which communications are taking place. See *local*.

retrieve The processes of copying data either from an external device or from a storage area to an active portion of the system such as a display buffer. Also, an output device connected to the PC is called a load.

rightmost (bit/word) The lowest numbered bits of a group of bits, generally of an entire word, or the lowest numbered words of a group of words. These bits/words are often called least-significant bits/words.

RUN mode The operating mode used by the PC for normal control operations.

servicing The process whereby the PC provides data to or receives data from external devices or remote I/O Units, or otherwise handles data transactions for Link Systems.

servolock An operation whereby a rotary encoder is used to maintain the position of a motor while it is stopped. Whenever the motor axis moves, the rotary encoder sends a feedback pulse to an error counter, causing a rotation voltage to be generated in the reverse direction so that the motor rotates back to its original position.

software error An error that originates in a software program.

sub-program A group of instructions that are executed independently of the main program.

target position A parameter for a positioning action that designates what position is to be reached at the completion of the action.

teaching Automatically writing the present position into memory, via the Teaching Box, as the target position for the designated positioning action.

transfer The process of moving data from one location to another within the PC, or between the PC and external devices. When data is transferred, generally a copy of the data is sent to the destination, i.e., the content of the source of the transfer is not changed.

unit address A number used to control network communications in FINS protocol. Unit addresses are computed for Units in various ways, e.g., 10 hex is added to the unit number to determine the unit address for a CPU Bus Unit.

unit number A number assigned to some Link Units, Special I/O Units, and CPU Bus Units to facilitate identification when assigning words or other operating parameters.

uploading The process of transferring a program or data from a lower-level or slave computer to a higher-level or host computer. If a Programming Devices is involved, the Programming Device is considered the host computer.

watchdog timer A timer within the system that ensures that the scan time stays within specified limits. When limits are reached, either warnings are given or PC operation is stopped depending on the particular limit that is reached.

WDT See *watchdog timer*.

wiring check A check performed automatically at startup to detect wiring problems such as reversed polarity or disconnections.

word A unit of data storage in memory that consists of 16 bits. All data areas consists of words. Some data areas can be accessed only by words; others, by either words or bits.

work bit A bit that can be used for data calculation or other manipulation in programming, i.e., a 'work space' in memory. Also see *work word*.

write-protect A state in which the contents of a storage device can be read but cannot be altered.

zone A range of positions or values which can be defined so that flags are turned ON whenever the present position is within the range.

Index

A

B

C

D

E

F

G

H

I

J

L

M

N

O

P

R

S

T

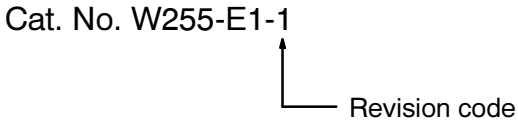
U

W

Z

Revision History

A manual revision code appears as a suffix to the catalog number on the front cover of the manual.



The following table outlines the changes made to the manual during each revision. Page numbers refer to the previous version.

Revision code	Date	Revised content
1	June 1995	Original production