

**C200H-MC221**

# **Motion Control Unit**

## **OPERATION MANUAL**

**OMRON**

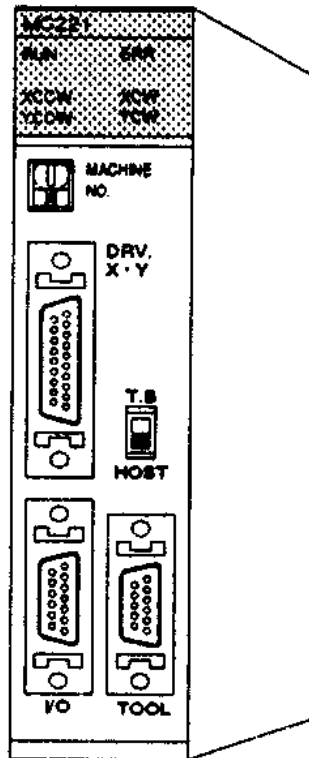


# C200H-MC221

## Motion Control Unit

### Operation Manual: Details

*Revised June 2010*








## **Notice:**

OMRON products are manufactured for use according to proper procedures by a qualified operator and only for the purposes described in this manual.

The following conventions are used to indicate and classify precautions in this manual. Always heed the information provided with them. Failure to heed precautions can result in injury to people or damage to property.

 **DANGER** Indicates an imminently hazardous situation which, if not avoided, will result in death or serious injury.

 **WARNING** Indicates a potentially hazardous situation which, if not avoided, could result in death or serious injury.

 **Caution** Indicates a potentially hazardous situation which, if not avoided, may result in minor or moderate injury, or property damage.

## **OMRON Product References**

All OMRON products are capitalized in this manual. The word “Unit” is also capitalized when it refers to an OMRON product, regardless of whether or not it appears in the proper name of the product.

The abbreviation “Ch,” which appears in some displays and on some OMRON products, often means “word” and is abbreviated “Wd” in documentation in this sense.

The abbreviation “PC” means Programmable Controller and is not used as an abbreviation for anything else.

## **Visual Aids**

The following headings appear in the left column of the manual to help you locate different types of information.

**Note** Indicates information of particular interest for efficient and convenient operation of the product.

**1, 2, 3...** 1. Indicates lists of one sort or another, such as procedures, checklists, etc.

## **© OMRON, 1996**

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, mechanical, electronic, photocopying, recording, or otherwise, without the prior written permission of OMRON.

No patent liability is assumed with respect to the use of the information contained herein. Moreover, because OMRON is constantly striving to improve its high-quality products, the information contained in this manual is subject to change without notice. Every precaution has been taken in the preparation of this manual. Nevertheless, OMRON assumes no responsibility for errors or omissions. Neither is any liability assumed for damages resulting from the use of the information contained in this publication.



# TABLE OF CONTENTS

<b>PRECAUTIONS</b> .....	<b>xvii</b>
1 Intended Audience .....	xviii
2 General Precautions .....	xviii
3 Safety Precautions .....	xviii
4 Operating Environment Precautions .....	xviii
5 Application Precautions .....	xix
<b>SECTION 1</b>	
<b>Components and Switch Settings</b> .....	<b>1</b>
1-1 Components .....	2
1-2 Indicators .....	3
1-3 Setting the Unit Number .....	3
1-4 Setting the DIP Switch .....	4
1-5 Installing the MC Unit .....	5
1-6 Changing the Task Configuration .....	6
<b>SECTION 2</b>	
<b>Wiring</b> .....	<b>7</b>
2-1 I/O Connector .....	8
2-2 DRV Connectors .....	11
2-3 MPG Connector .....	19
2-4 Interface Circuits .....	20
<b>SECTION 3</b>	
<b>Connecting Peripheral Devices</b> .....	<b>23</b>
3-1 Connecting a Personal Computer .....	24
3-2 Connecting the Teaching Box .....	25
<b>SECTION 4</b>	
<b>MC Unit Data</b> .....	<b>29</b>
4-1 MC Unit Data Configuration .....	30
4-2 System Parameters .....	30
4-3 Data Transmission and Reception .....	33
4-4 Details of Data Transmission and Reception Units .....	36
4-5 Details of I/O Transfer and Necessary Information .....	51
4-6 Data Transfer Examples .....	52
4-7 I/O Transfer Timing .....	56
4-8 Details of Intelligent Transfer .....	58
4-9 IOWR Instruction .....	59
4-10 IORD Instruction .....	64
<b>SECTION 5</b>	
<b>G Language</b> .....	<b>69</b>
5-1 List of G Functions .....	70
5-2 G Function Formats .....	71
5-3 List of G Symbols .....	72
5-4 Program Number and Axis Declaration .....	74
5-5 G Functions .....	75
5-6 M-code Outputs .....	107
5-7 Mode Transitions Caused by G Functions .....	112

# TABLE OF CONTENTS

<b>SECTION 6</b>	
<b>PC Data Area Interface</b> .....	<b>113</b>
6-1 Introduction .....	115
6-2 Allocation of the PC Data Area Interface .....	116
6-3 Controlling the MC Unit from the PC .....	118
6-4 Interface Bits .....	121
6-5 Interface Bit Specifics .....	134
<b>SECTION 7</b>	
<b>Fixing the Origin</b> .....	<b>177</b>
7-1 Introduction .....	178
7-2 Sensors Required for an Origin Search .....	178
7-3 Origin Search Parameters .....	179
7-4 Origin Search Patterns .....	180
7-5 Overview of Absolute Encoders .....	186
7-6 Setting Up the Absolute Encoder .....	187
7-7 ABS Soft Reset .....	188
7-8 Absolute Encoder Interface Specifications .....	189
<b>SECTION 8</b>	
<b>Teaching</b> .....	<b>193</b>
8-1 Introduction .....	194
8-2 The Teaching Function .....	194
8-3 Setting the Teaching Address .....	195
8-4 Performing Teaching .....	196
8-5 Errors .....	197
<b>SECTION 9</b>	
<b>Sample Programs</b> .....	<b>199</b>
9-1 Positioning with PTP Control .....	200
9-2 Positioning with Linear Interpolation .....	201
9-3 Positioning with Linear and Circular Interpolation .....	202
9-4 Indirect Addressing with Registers .....	203
9-5 Using the Workpiece Origin Offset .....	204
9-6 Changing the Interpolation Acceleration Time .....	205
9-7 Calculating Position Data .....	206
9-8 Stopping a Program with a General Input .....	208
9-9 Stopping the Program and Substituting Position Data .....	209
9-10 Executing MC Programs from the Ladder Program .....	210
<b>SECTION 10</b>	
<b>Troubleshooting</b> .....	<b>213</b>
10-1 Error Indicators .....	214
10-2 System Errors: Error Codes 0001 to 0017 .....	215
10-3 Task Errors: Error Codes 0020 to 0045 .....	216
10-4 Axis Errors: Error Codes 0060 to 0088 .....	219
<b>SECTION 11</b>	
<b>Maintenance and Inspection</b> .....	<b>223</b>
Routine Inspections .....	224



# TABLE OF CONTENTS

## Appendices

A Control Bit/Flag Timing Charts .....	225
B Additional Origin Search Patterns .....	239
C MC Program Coding Sheet .....	265
D MC Parameter Settings .....	267
E Position Data Coding Sheet .....	271
<b>Glossary .....</b>	<b>273</b>
<b>Index .....</b>	<b>279</b>
<b>Revision History .....</b>	<b>285</b>



## About this Manual:

This manual provides an introduction to the features and operation of the C200H-MC221 Motion Control Unit and includes the sections described below.

Please read this manual and the other manuals related to the C200H-MC221 Motion Control Unit carefully and be sure you understand the information provided before attempting to install and operate the Motion Control Units. **Be sure to read the precautions in the following section.**

There are four manuals used with the C200H-MC221 Motion Control Unit (MC Unit). 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.
C200H-MC221 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.	W314
C200H-MC221 Motion Control Unit Operation Manual: Details	Describes operation of the Motion Control Units in detail. Read the <i>Operation Manual: Introduction</i> , above, before attempting to read this manual.	W315
CVM1-PRO01 Teaching Box Operation Manual (For Motion Control Unit)	Describes the operation of the Teaching Box connected to a Motion Control Unit.	W320
CV500-ZN3AT1-E 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.



## ***Read and Understand this Manual***

Please read and understand this manual before using the product. Please consult your OMRON representative if you have any questions or comments.

## ***Warranty and Limitations of Liability***

### ***WARRANTY***

OMRON's exclusive warranty is that the products are free from defects in materials and workmanship for a period of one year (or other period if specified) from date of sale by OMRON.

OMRON MAKES NO WARRANTY OR REPRESENTATION, EXPRESS OR IMPLIED, REGARDING NON-INFRINGEMENT, MERCHANTABILITY, OR FITNESS FOR PARTICULAR PURPOSE OF THE PRODUCTS. ANY BUYER OR USER ACKNOWLEDGES THAT THE BUYER OR USER ALONE HAS DETERMINED THAT THE PRODUCTS WILL SUITABLY MEET THE REQUIREMENTS OF THEIR INTENDED USE. OMRON DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED.

### ***LIMITATIONS OF LIABILITY***

OMRON SHALL NOT BE RESPONSIBLE FOR SPECIAL, INDIRECT, OR CONSEQUENTIAL DAMAGES, LOSS OF PROFITS OR COMMERCIAL LOSS IN ANY WAY CONNECTED WITH THE PRODUCTS, WHETHER SUCH CLAIM IS BASED ON CONTRACT, WARRANTY, NEGLIGENCE, OR STRICT LIABILITY.

In no event shall the responsibility of OMRON for any act exceed the individual price of the product on which liability is asserted.

IN NO EVENT SHALL OMRON BE RESPONSIBLE FOR WARRANTY, REPAIR, OR OTHER CLAIMS REGARDING THE PRODUCTS UNLESS OMRON'S ANALYSIS CONFIRMS THAT THE PRODUCTS WERE PROPERLY HANDLED, STORED, INSTALLED, AND MAINTAINED AND NOT SUBJECT TO CONTAMINATION, ABUSE, MISUSE, OR INAPPROPRIATE MODIFICATION OR REPAIR.

## ***Application Considerations***

### ***SUITABILITY FOR USE***

OMRON shall not be responsible for conformity with any standards, codes, or regulations that apply to the combination of products in the customer's application or use of the products.

At the customer's request, OMRON will provide applicable third party certification documents identifying ratings and limitations of use that apply to the products. This information by itself is not sufficient for a complete determination of the suitability of the products in combination with the end product, machine, system, or other application or use.

The following are some examples of applications for which particular attention must be given. This is not intended to be an exhaustive list of all possible uses of the products, nor is it intended to imply that the uses listed may be suitable for the products:

- Outdoor use, uses involving potential chemical contamination or electrical interference, or conditions or uses not described in this manual.
- Nuclear energy control systems, combustion systems, railroad systems, aviation systems, medical equipment, amusement machines, vehicles, safety equipment, and installations subject to separate industry or government regulations.
- Systems, machines, and equipment that could present a risk to life or property.

Please know and observe all prohibitions of use applicable to the products.

**NEVER USE THE PRODUCTS FOR AN APPLICATION INVOLVING SERIOUS RISK TO LIFE OR PROPERTY WITHOUT ENSURING THAT THE SYSTEM AS A WHOLE HAS BEEN DESIGNED TO ADDRESS THE RISKS, AND THAT THE OMRON PRODUCTS ARE PROPERLY RATED AND INSTALLED FOR THE INTENDED USE WITHIN THE OVERALL EQUIPMENT OR SYSTEM.**

### ***PROGRAMMABLE PRODUCTS***

OMRON shall not be responsible for the user's programming of a programmable product, or any consequence thereof.

## **Disclaimers**

### ***CHANGE IN SPECIFICATIONS***

Product specifications and accessories may be changed at any time based on improvements and other reasons.

It is our practice to change model numbers when published ratings or features are changed, or when significant construction changes are made. However, some specifications of the products may be changed without any notice. When in doubt, special model numbers may be assigned to fix or establish key specifications for your application on your request. Please consult with your OMRON representative at any time to confirm actual specifications of purchased products.

### ***DIMENSIONS AND WEIGHTS***

Dimensions and weights are nominal and are not to be used for manufacturing purposes, even when tolerances are shown.

### ***PERFORMANCE DATA***

Performance data given in this manual is provided as a guide for the user in determining suitability and does not constitute a warranty. It may represent the result of OMRON's test conditions, and the users must correlate it to actual application requirements. Actual performance is subject to the OMRON Warranty and Limitations of Liability.

### ***ERRORS AND OMISSIONS***

The information in this manual has been carefully checked and is believed to be accurate; however, no responsibility is assumed for clerical, typographical, or proofreading errors, or omissions.





# PRECAUTIONS

This section provides general precautions for using the Motion Control Units (MC Units) and related devices.

**The information contained in this section is important for the safe and reliable application of the Motion Control Unit. You must read this section and understand the information contained before attempting to set up or operate a Motion Control Unit.**

1 Intended Audience .....	xviii
2 General Precautions .....	xviii
3 Safety Precautions .....	xviii
4 Operating Environment Precautions .....	xviii
5 Application Precautions .....	xix

## 1 Intended Audience

This manual is intended for the following personnel, who must also have knowledge of electrical systems (an electrical engineer or the equivalent).

- Personnel in charge of installing FA systems.
- Personnel in charge of designing FA systems.
- Personnel in charge of managing FA systems and facilities.


## 2 General Precautions

The user must operate the product according to the performance specifications described in the operation manuals.


Before using the product under conditions which are not described in the manual or applying the product to nuclear control systems, railroad systems, aviation systems, vehicles, combustion systems, medical equipment, amusement machines, safety equipment, and other systems, machines, and equipment that may have a serious influence on lives and property if used improperly, consult your OMRON representative.


Make sure that the ratings and performance characteristics of the product are sufficient for the systems, machines, and equipment, and be sure to provide the systems, machines, and equipment with double safety mechanisms.

This manual provides information for using the MC Unit. Be sure to read this manual before attempting to use the Unit and keep this manual close at hand for reference during operation.

 **WARNING** It is extremely important that Motion Control Units and related devices be used for the specified purpose and under the specified conditions, especially in applications that can directly or indirectly affect human life. You must consult with your OMRON representative before applying Motion Control Units and related devices to the above mentioned applications.

## 3 Safety Precautions


 **WARNING** Never attempt to disassemble any Units while power is being supplied. Doing so may result in serious electrical shock or electrocution.

 **WARNING** Never touch any of the terminals while power is being supplied. Doing so may result in serious electrical shock or electrocution.


## 4 Operating Environment Precautions

 **Caution** Do not operate the control system in the following places:

- Locations subject to direct sunlight.
- Locations subject to temperatures or humidity outside the range specified in the specifications.
- Locations subject to condensation as the result of severe changes in temperature.
- Locations subject to corrosive or flammable gases.
- Locations subject to dust (especially iron dust) or salts.
- Locations subject to exposure to water, oil, or chemicals.
- Locations subject to shock or vibration.


 **Caution** Take appropriate and sufficient countermeasures when installing systems in the following locations:

- Locations subject to static electricity or other forms of noise.
- Locations subject to strong electromagnetic fields.
- Locations subject to possible exposure to radioactivity.
- Locations close to power supplies.


 **Caution** The operating environment of the MC Unit can have a large effect on the longevity and reliability of the system. Improper operating environments can lead to malfunction, failure, and other unforeseeable problems with the MC Unit. Be sure that the operating environment is within the specified conditions at installation and remains within the specified conditions during the life of the system.

## 5 Application Precautions

Observe the following precautions when using the MC Unit or the PC.

 **WARNING** Failure to abide by the following precautions could lead to serious or possibly fatal injury. Always heed these precautions.

- Always ground the system to 100  $\Omega$  or less when installing the system to protect against electrical shock.
- Always turn OFF the power supply to the PC before attempting any of the following. Not turning OFF the power supply may result in malfunction or electric shock.
  - Mounting or dismounting I/O Units, CPU Units, Memory Cassettes, or any other Units.
  - Assembling the Units.
  - Setting DIP switches or rotary switches.
  - Connecting or wiring the cables.
  - Connecting or disconnecting the connectors.

 **Caution** Failure to abide by the following precautions could lead to faulty operation or the MC Unit or the system or could damage the MC Units or PC Units. Always heed these precautions.

- Fail-safe measures must be taken by the customer to ensure safety in the event of incorrect, missing, or abnormal signals caused by broken signal lines, momentary power interruptions, or other causes.
- Provide external interlock circuits, limit circuits, and other safety circuits in addition to any provided within the PC to ensure safety.
- Provide circuit breakers and other safety measures to provide protection against shorts in external wiring.
- Be sure that all the mounting screws, terminal screws, and cable connector screws are tightened to the torque specified in the relevant manuals. Incorrect tightening torque may result in malfunction.
- Mount the Units only after confirming that connectors are properly connected.
- Always use the power supply voltage specified in the operation manuals. An incorrect voltage may result in malfunction or burning.
- Take appropriate measures to ensure that the specified power with the rated voltage and frequency is supplied. Be particularly careful in places where the power supply is unstable. An incorrect power supply may result in malfunction.

- Be sure that the terminal blocks, Memory Units, expansion cables, and other items with locking devices are properly locked into place. Improper locking may result in malfunction.
- Double-check all the wiring before turning ON the power supply. Incorrect wiring may result in burning.
- Disconnect the functional ground terminal when performing withstand voltage tests. Not disconnecting the functional ground terminal may result in burning.
- Confirm that no adverse effect will occur in the system before changing the present value of any word or any set value in memory. Not doing so may result in an unexpected operation.
- Confirm that set parameters and data operate properly. Not doing so may result in an unexpected operation.
- Do not attempt to take any Units apart, to repair any Units, or to modify any Units in any way.
- Do not pull on the cables or bend the cables beyond their natural limit. Doing either of these may break the cables.
- Do not place objects on top of the cables. Doing so may break the cables.
- Do not hold by the cables when transporting the Units. Doing so may damage the Units.
- Install the Unit properly as specified in the operation manual. Improper installation of the Unit may result in malfunction.

# SECTION 1

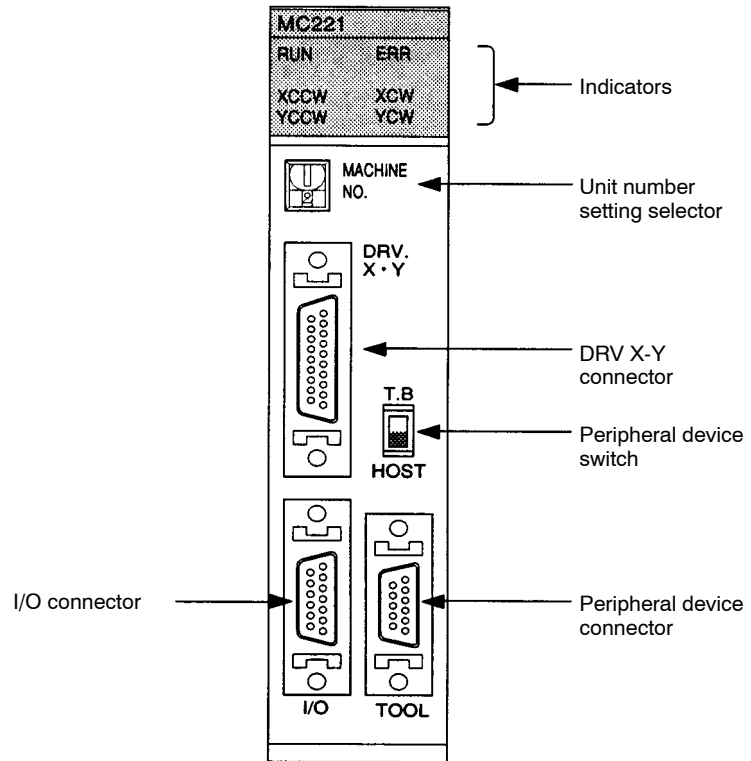
## Components and Switch Settings

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 .....	2
1-2	Indicators .....	3
1-3	Setting the Unit Number .....	3
1-4	Setting the DIP Switch .....	4
1-5	Installing the MC Unit .....	5
1-6	Changing the Task Configuration .....	6
1-6-1	Operation Flow .....	6
1-6-2	Basic Operations .....	6

# 1-1 Components

The following diagram shows the main components of the MC Unit.



**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
RUN	Green	ON	Initialization was completed normally and the connection of the MC Unit and the PC is normal.
		OFF	The MC Unit or PC has an error.
ERR	Red	ON	An error occurred in the MC Unit.
		OFF	The MC Unit is operating normally.
X or Y CCW	Orange	ON	The motor is rotating counterclockwise.
		OFF	The motor is stopped or rotating counterclockwise.
X or Y CW	Orange	ON	The motor is rotating clockwise.
		OFF	The motor is stopped or rotating clockwise.

**Note** A fatal error of the MC Unit will be indicated by the XCCW, YCCW, XCW, or YCW indicator. Refer to *10-1 Error Indicators* for details.

## 1-3 Setting the Unit Number

The unit number setting determines which IR 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

The interface area varies with the type of the PC to which the MC Unit is mounted. Refer to the following examples. Refer to *Section 6 PC Data Area Interface* for more details.

#### C200H/C200HS

IR area: 20 words beginning with word n ( $100 + 10 \times \text{unit no.}$ )  
 DM area: 2 words beginning with word m ( $1,000 + 100 \times \text{unit no.}$ )

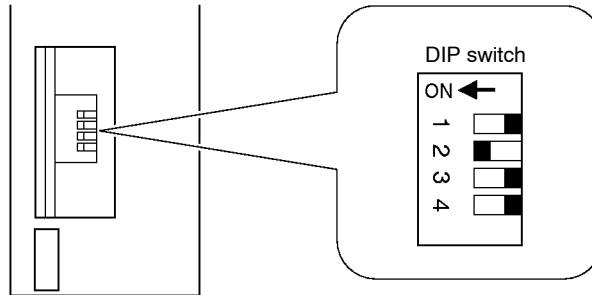
#### C200HX/C200HG/C200HE

IR area: 20 words beginning with word n ( $100 + 10 \times \text{unit no.}$ ) if the unit no. is 8 or less.  
 20 words beginning with word n ( $400 + 10 \times (\text{unit no.} - 10)$ ) if the unit no. is 10 or larger.  
 DM area: 2 words beginning with word m ( $1,000 + 100 \times \text{unit no.}$ )

- Note**
- The unit number can be set from 0 to F (A to F are 10 to 15 respectively).
  - The unit number must be from 0 to 8 if the MC Unit is mounted to the C200H or C200HS.  
 The unit number must be from 0 to 8 or from A to E if the MC Unit is mounted to the C200HX, C200HG, or C200HE.  
 If the unit number is set to 9 or F, an error will occur when the power is turned ON.
  - 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. The DIP switch is on the rear panel of the MC Unit.



### 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	Enables the initial setting and soft reset function of the absolute encoder (the factory setting).
	ON	Disables the initial setting and soft reset function of the absolute encoder (the factory setting).
2	OFF	The Teaching Box will display Japanese text.
	ON	The Teaching Box will display English text (factory setting).
3 to 4	---	Leave these pins set to OFF (factory settings).

The pin 1 setting will be ignored when using an incremental encoder.

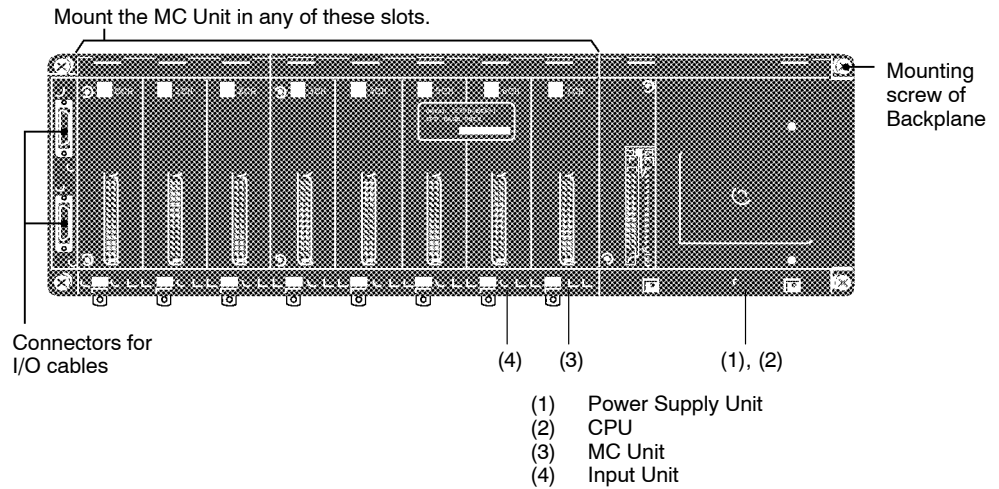
- Note**
1. Use the interface relay for the initial setting and soft reset setting of the absolute encoder. Refer to *Section 6-5 Interface Bit Specifics*.
  2. For setting the absolute encoder from the Teaching Box, refer to *7-6 Setting Up the Absolute Encoder* and *7-7 ABS Soft Reset*.
  3. The Teaching Box display language selector (pin 2) is referenced only when the PC is turned ON.



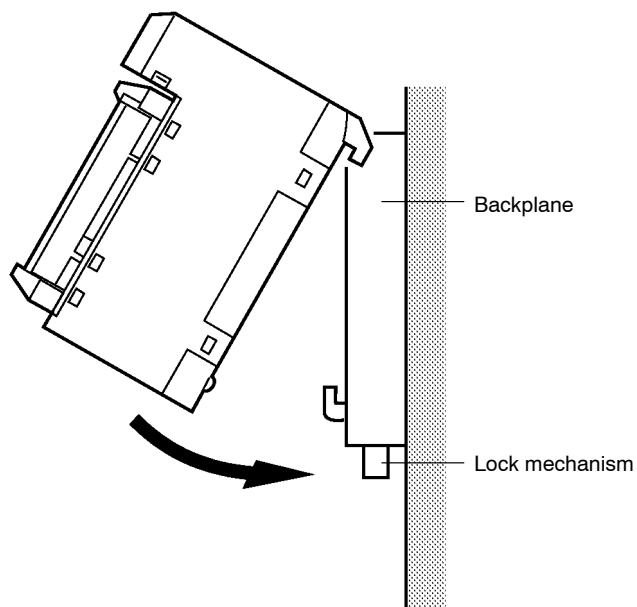
## 1-5 Installing the MC Unit

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 (10/8/5/3). In the following example, the MC Unit is mounted to the slot indicated by (3) and the Input Unit is mounted to the slot indicated by (4).



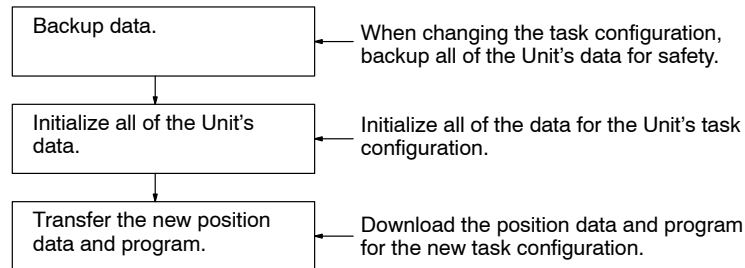
Mount the MC Unit by connecting the connector on the rear panel of the MC Unit to the Backplane.



## 1-6 Changing the Task Configuration

Use the following procedure to change the C200H-MC221's task configuration from task 1 to task 2 or from task 2 to task 1. The Unit is set for task 1 when shipped from OMRON.

### 1-6-1 Operation Flow



### 1-6-2 Basic Operations

#### Data Backup

Use the MC Support Software to backup the Unit's data by transferring the system parameters, program, and position data from the Unit to the personal computer. Refer to the *MC Support Software Operation Manual (W256)* for details on transferring data.

#### Data Initialization

- 1, 2, 3...**
1. Change the task number from 1 to 2 or from 2 to 1 in the MC Support Software's Unit Parameter Edit menu. If there are other parameters that should be changed, change them and save them.
  2. Transfer the parameters created in step 1 (from the Unit to the personal computer). Write "all data" to flash memory.
  3. Turn the Unit's power supply off and on again.
  4. Delete the Unit's entire program with the MC Support Software's MC Program Edit menu. Write "all data" to flash memory.
  5. Turn the Unit's power supply off and on again.

These steps initialize the Unit for the new task configuration.

#### Data Transfer

- 1, 2, 3...**
1. Create the position data and program for the new task configuration and transfer the data and program from the personal computer to the Unit. If the position data and program already exist, just transfer the data and program.
  2. When all of the position data and program have been transferred, write "all data" to flash memory.

These steps complete the Unit's setup.

## 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 .....	8
2-1-1	Pin Allocation .....	8
2-1-2	Attaching a Connector .....	8
2-1-3	Connection Example .....	9
2-2	DRV Connectors .....	11
2-2-1	Pin Allocation .....	11
2-2-2	Attaching a Connector .....	11
2-2-3	Dedicated Driver Cables (Optional) .....	13
2-2-4	Connection Examples .....	15
2-3	MPG Connector .....	19
2-3-1	Wiring Example .....	19
2-3-2	Connection Example .....	19
2-4	Interface Circuits .....	20
2-4-1	I/O and DRV Connector Circuits .....	20

## 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, emergency stop inputs, and origin proximity inputs, as well as general I/O connections. Dedicated cables and terminals can be connected to the I/O connector.

### 2-1-1 Pin Allocation

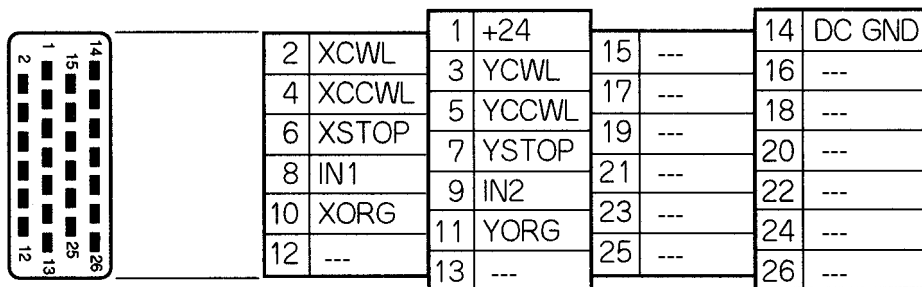
A Bellows 26-pin half-pitch plug and case are included with the Unit.

#### Snap-on Connector

The following connector and case are provided with the Unit.

1 connector: 10126-3000VE (Sumitomo 3M) (Model available in Japan.)

1 case: 10326-42F0-008 or 10326-52F0-008 (Sumitomo 3M) (Models available in Japan.)



#### Pin Functions

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

Pin	Symbol <sup>1</sup>	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 emergency stop input	Invalidates the X-axis's run output and stops it.
7	YSTOP(NC)	Y-axis emergency 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) <sup>2</sup>	X-axis origin proximity input	Used for the X-axis origin search.
11	YORG(NC,NO) <sup>2</sup>	Y-axis origin proximity input	Used for the Y-axis origin search.
14	DC GND	24 VDC input ground	Connects to the – terminal (0 V) of the 24-VDC external power supply.

**Note** 1. “NC” stands for normally closed and “NO” stands for normally open. Always short normally closed input terminals which aren't used.

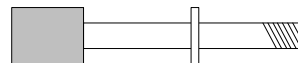
2. Either NC or NO logic can be used. This setting is a mechanical parameter (Origin Proximity Input Logic) set with MC Support Software.

### 2-1-2 Attaching a Connector

Attach a connector in one of the following ways.

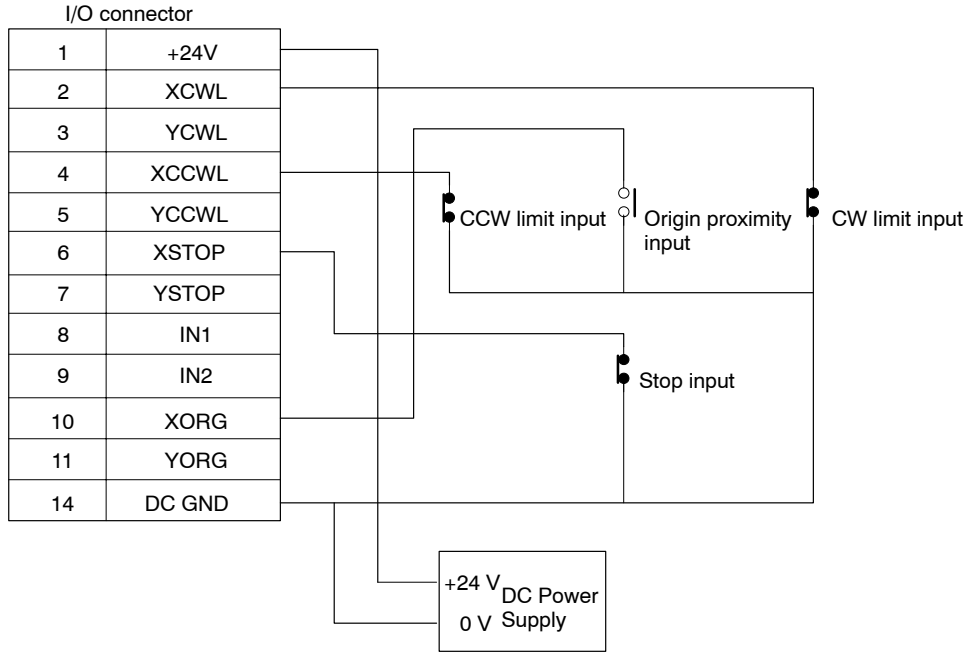
- Use the connector (snap-on type) provided with the Unit.
- When using screws, use anti-electrostatic screws provided with the Unit.

Anti-electrostatic screws (4 screws provided)

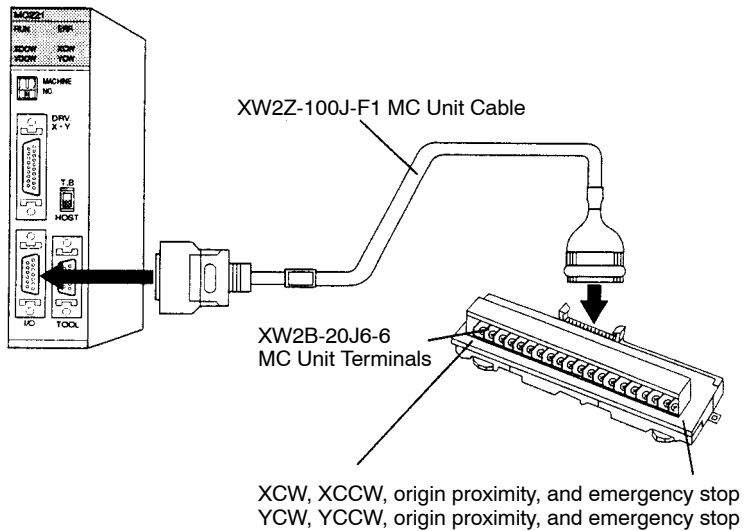


### 2-1-3 Connection Example

The following diagram shows an example connection for just the X-axis. Corresponding wiring is required for the Y-axis.

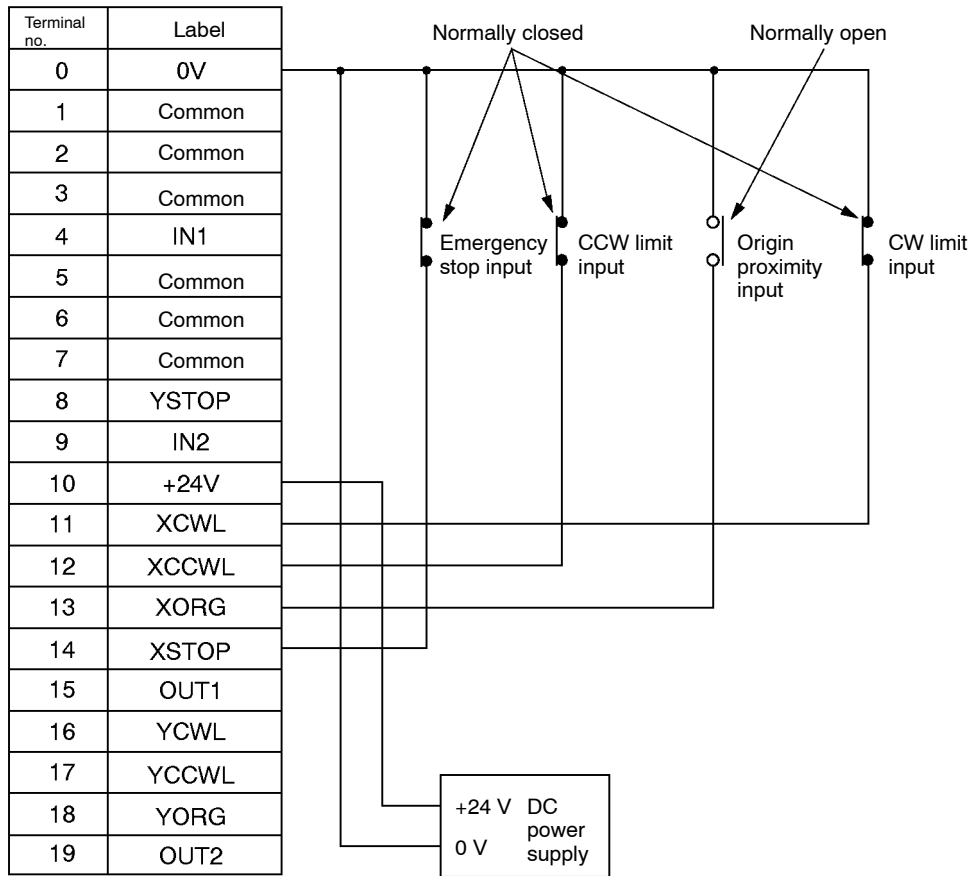


#### Dedicated Cable and Terminals



**Connection Example**

The following diagram shows an example connection for just the X axis. Connect Y-axis wires in the same way.



**Note** OUT1 or OUT2 of the Terminals is not available for the C200H-MC221.

## 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.

**Note** Dedicated driver cables, which are sold separately, are available for OMRON U-, H-, and M-series Servodrivers. Refer to 2-2-3 *Dedicated Driver Cables*.

### 2-2-1 Pin Allocation

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

#### Snap-on Connector

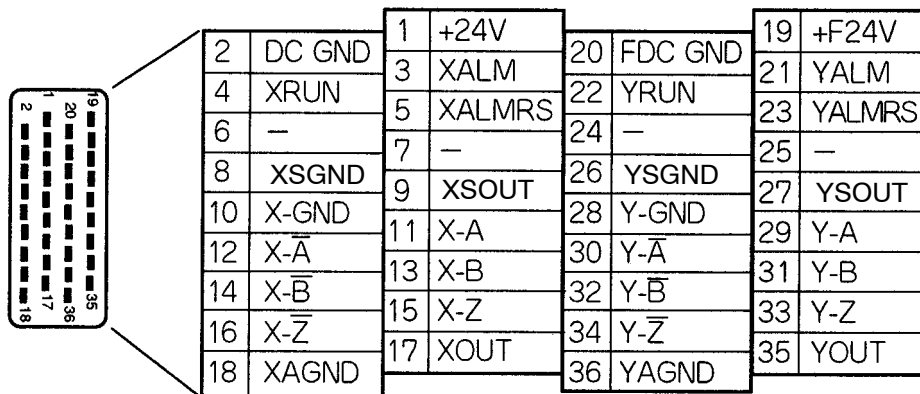
The following connector and case are provided with the Unit.

1 connector: 10136-3000VE (Sumitomo 3M) (Model available in Japan.)

1 case: 10336-42F0-008 or 10336-52F0-008 (Sumitomo 3M) (Models available in Japan.)

#### DRV X-Y Pin Allocation

The following diagram shows the pin allocation for the DRV X-Y connector. Pins 6, 7, 24, and 25 aren't used.

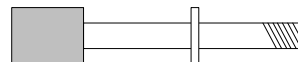


### 2-2-2 Attaching a Connector

Attach a connector in one of the following ways.

- Use the connector (snap-on type) provided with the Unit.
- When using screws, use anti-electrostatic screws provided with the Unit.

Anti-electrostatic screws (4 screws provided)



**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 7	---	---	Not used.
8	XSGND	X-axis SEN signal ground	SEN signal ground for the X-axis
9	XSOUT	X-axis SEN signal output	SEN signal output for the X-axis (absolute encoder driver)
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- $\bar{A}$	X-axis phase $\bar{A}$ input	Phase $\bar{A}$ feedback input for the X-axis
13	X-B	X-axis phase B input	Phase B feedback input for the X-axis
14	X- $\bar{B}$	X-axis phase $\bar{B}$ input	Phase $\bar{B}$ feedback input for the X-axis
15	X-Z	X-axis phase Z input	Phase Z feedback input for the X-axis
16	X- $\bar{Z}$	X-axis phase $\bar{Z}$ input	Phase $\bar{Z}$ 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 25	---	---	Not used.
26	YSGND	Y-axis SEN signal ground	SEN signal ground for the Y-axis
27	YSOUT	Y-axis SEN signal output	SEN signal output for the Y-axis (absolute encoder driver)
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- $\bar{A}$	Y-axis phase $\bar{A}$ input	Phase $\bar{A}$ feedback input for the Y-axis
31	Y-B	Y-axis phase B input	Phase B feedback input for the Y-axis
32	Y- $\bar{B}$	Y-axis phase $\bar{B}$ input	Phase $\bar{B}$ feedback input for the Y-axis
33	Y-Z	Y-axis phase Z input	Phase Z feedback input for the Y-axis
34	Y- $\bar{Z}$	Y-axis phase $\bar{Z}$ input	Phase $\bar{Z}$ 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



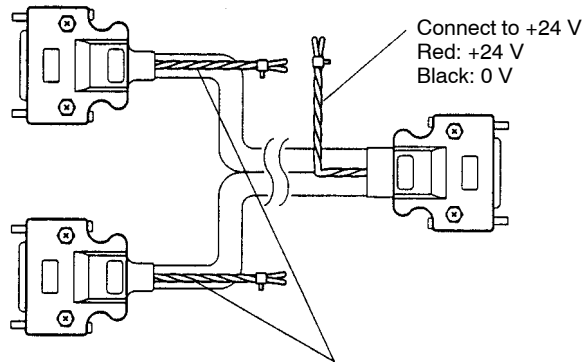
### 2-2-3 Dedicated Driver Cables (Optional)

When using OMRON's U-, H-, or M-series Servodrivers, use Dedicated Driver Cables that are available as options to connect the MC Unit to Servodrivers. These Dedicated Driver Cables will eliminate the need for wiring.

The following Driver Cables are available.

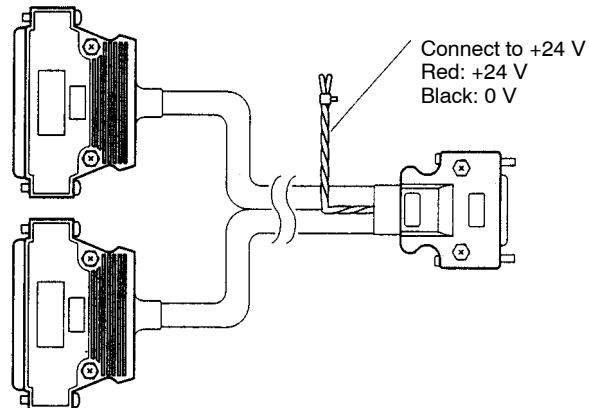
Series	For two axes	For single axis	Length (m)
R88D-U for 30 to 750-W Servodrivers	R88A-CPU001M2	R88A-CPU001M1	1.0
	R88A-CPU002M2	R88A-CPU002M1	2.0
R88D-U for 1 to 5-kW Servodrivers	R88A-CPUB001M2	R88A-CPUB001M1	1.0
	R88A-CPUB002M2	R88A-CPUB002M1	2.0
R88D-H	R88A-CPH001M2	R88A-CPH001M1	1.0
	R88A-CPH002M2	R88A-CPH002M1	2.0
R88D-M	R88A-CPM001M2	R88A-CPM001M1	1.0
	R88A-CPM002M2	R88A-CPM002M1	2.0

#### R88A-CPU001M2/002M2 (For U Series) R88A-CPUB001M2/002M2

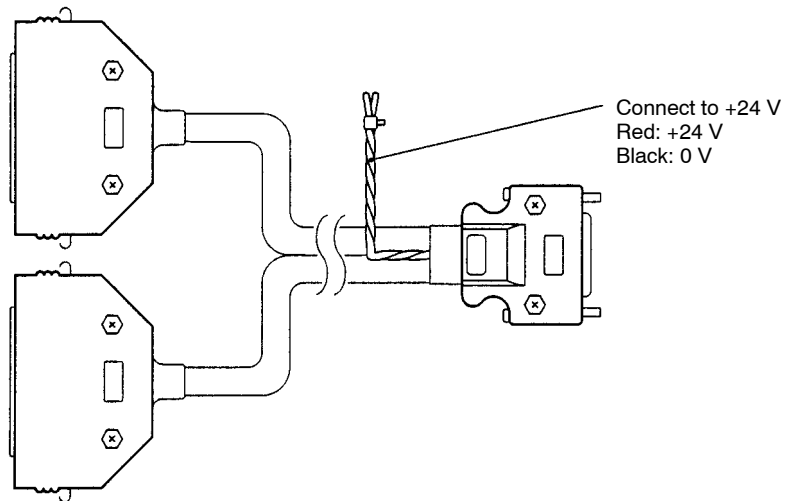


Connect to a battery when using the absolute encoder.  
Red: Positive  
Black: Negative

#### R88A-CPH001M2/002M2 (For H Series)



R88A-CPM001M2/002M2 (For M Series)

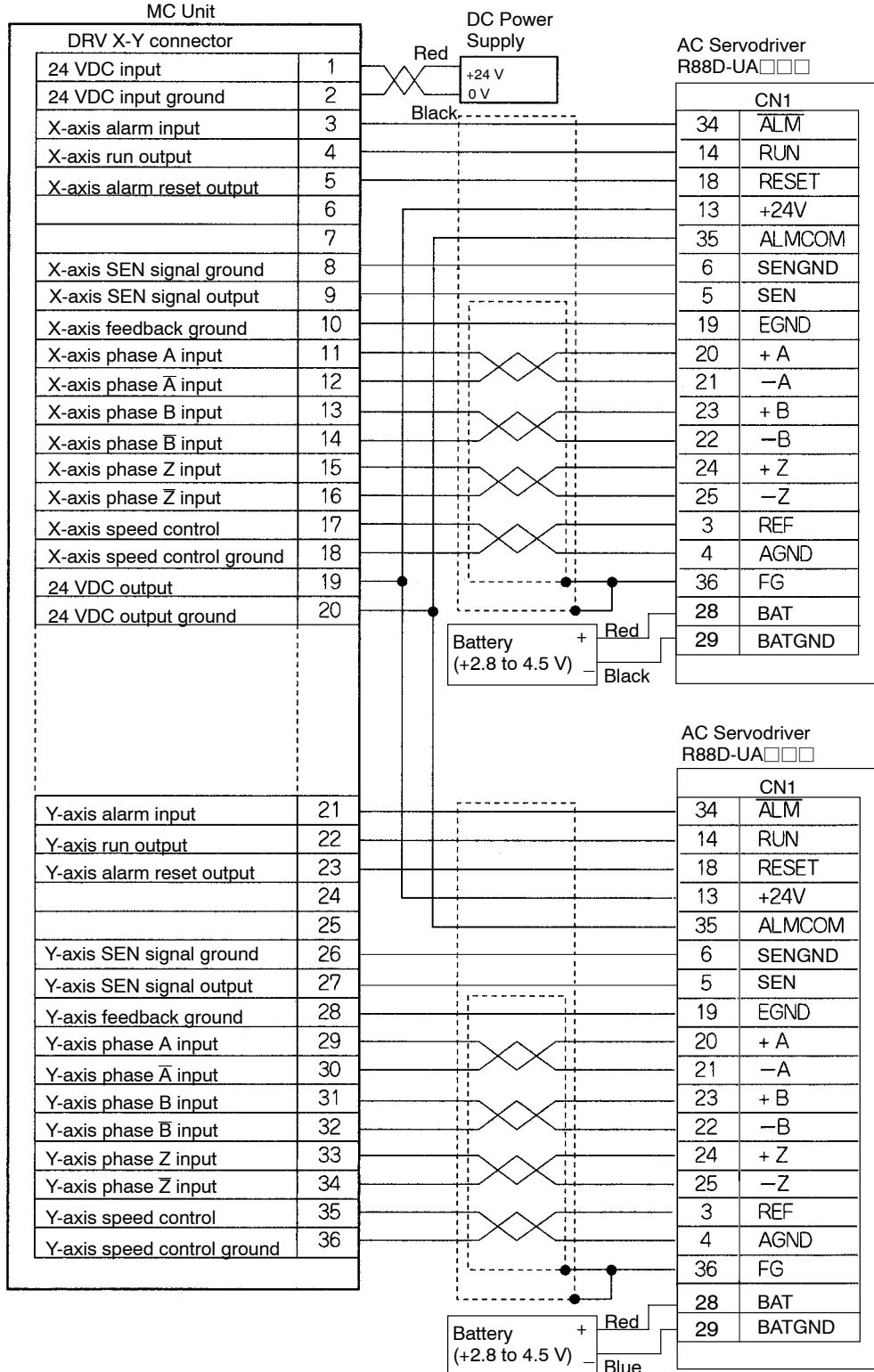


In the case of Cables for single-axis models, only one connector is available for the MC Unit side and only the X-axis signal line is wired.

### 2-2-4 Connection Examples

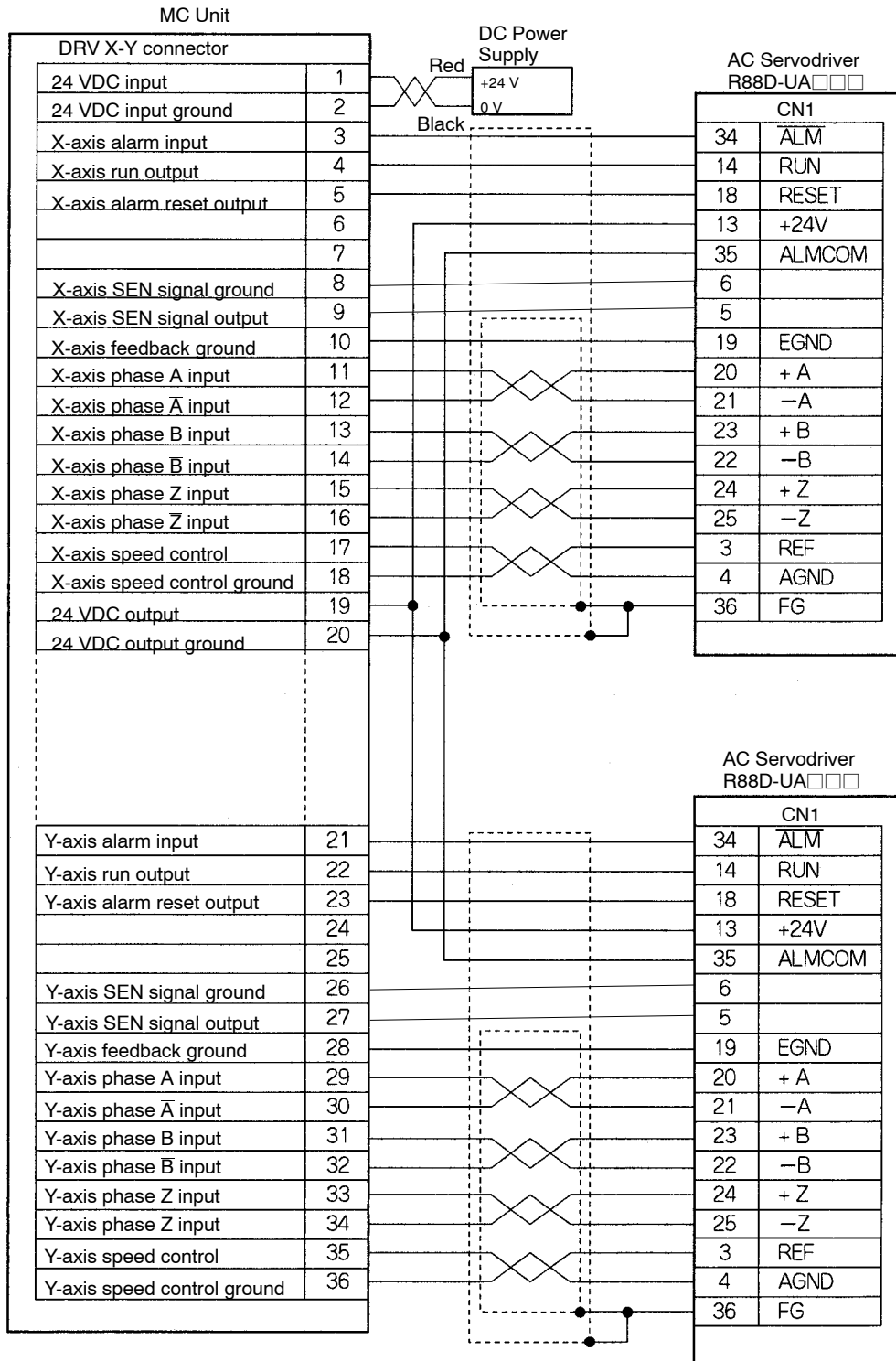
#### Connection to the U-series (30 to 750 W) Models (Using an Absolute Encoder)

Dedicated Driver Cable: R88A-CPU00□M2



Connection to the U-series (30 to 750 W) Models (Using an Incremental Encoder)

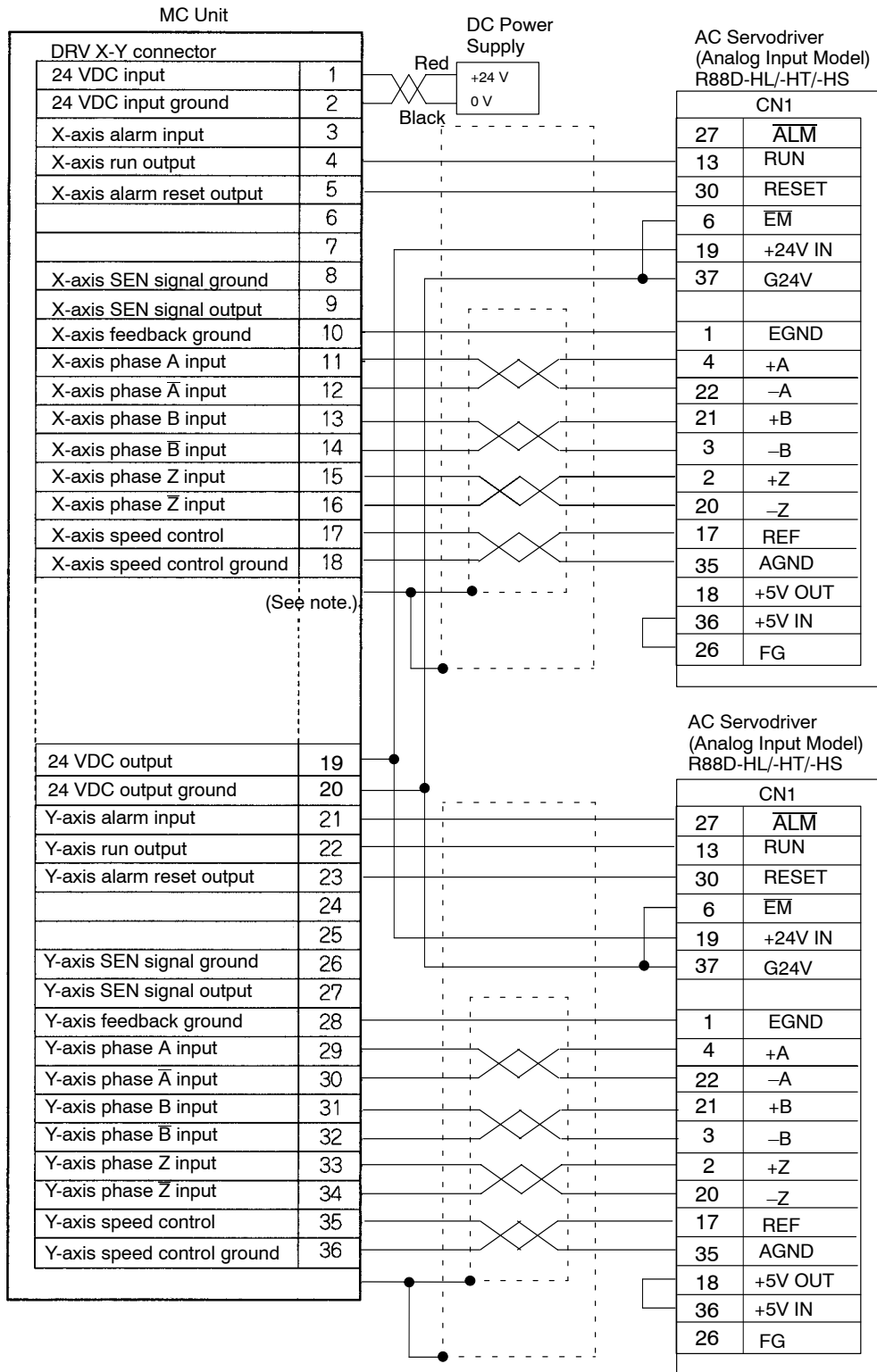
Dedicated Driver Cable: R88A-CPU00□M2



**Note** In the case of the incremental encoder, terminals 5 and 6 on the Servodriver side are connected. However, this can be ignored. Users are not required to connect them.

Connection to the H Series

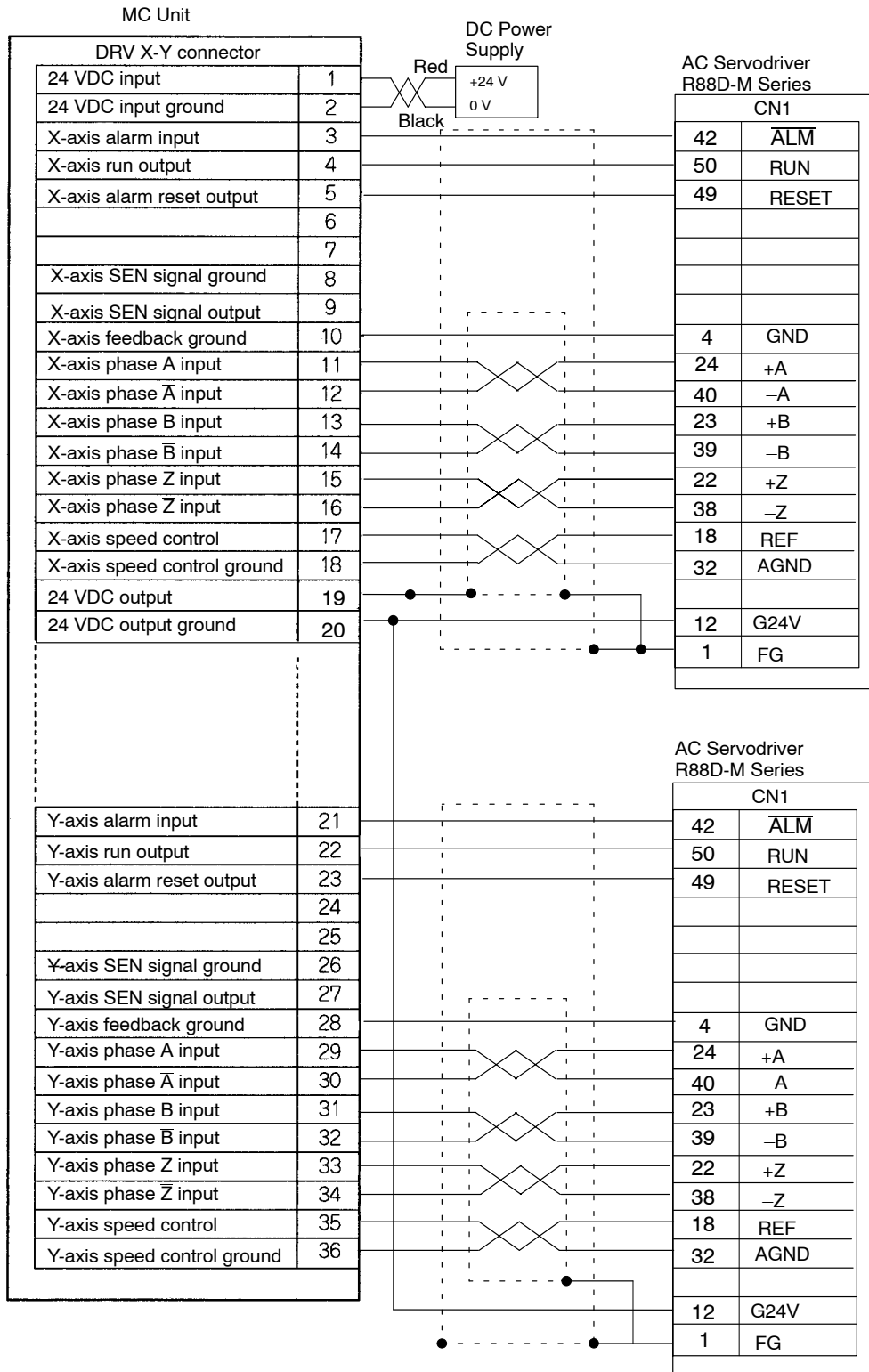
Dedicated Driver Cable: R88A-CPH00□M2



**Note** Ground the shielded line to the connector bracket on the MC Unit side.

Connection to the M Series

Dedicated Driver Cable: R88A-CPM00□M2

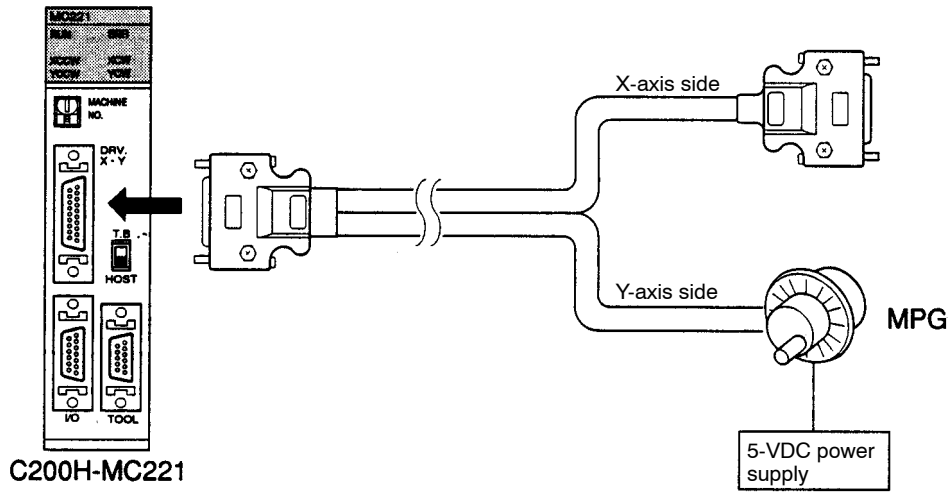


## 2-3 MPG Connector

A manual pulse generator (MPG) can be connected to the Y-axis encoder input if a single axis of the MC Unit is used. Refer to the following wiring example. Use a line-driver type of MPG. (LFG-003-100 manufactured by SAMUTAKU is recommended.)

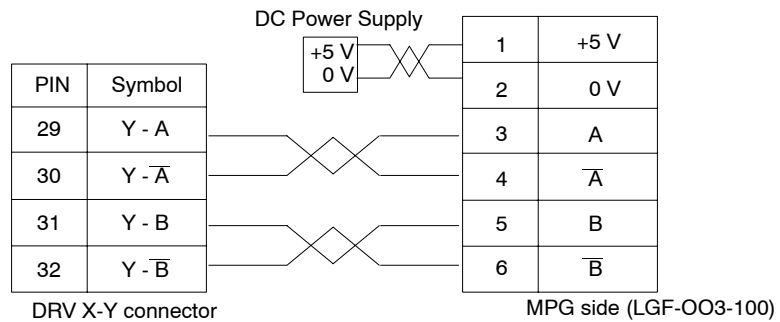
### 2-3-1 Wiring Example

Refer to 2-2 DRV Connectors for dedicated driver cables.



### 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 and DRV connectors.

### 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 and Y)  
 NO inputs: IN (1 and 2)  
 NC or NO inputs: ORG (X and Y)

Item	Specification	Circuit Configuration
Rated input voltage	24 VDC $\pm$ 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 RUN (X and Y) and ALMRS (X and Y).

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 $\pm$ 10%	

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

Circuit Configuration



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

Item	Specification	Circuit Configuration
Signal level	EIA RS-422-A Standards	
Input impedance	220 $\Omega$	
Response frequency	250 kpps max.	

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

Item	Specification	Circuit Configuration
Output voltage	0 to $\pm 10$ V	
Load impedance	10 K $\Omega$ min.	



## SECTION 3

# Connecting Peripheral Devices

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

3-1	Connecting a Personal Computer .....	24
3-1-1	Personal Computer and MC Unit .....	24
3-2	Connecting the Teaching Box .....	25

### 3-1 Connecting a Personal Computer

This section explains how to connect a personal computer with MC Support Software to the PC.

#### 3-1-1 Personal Computer and MC Unit

The MC Unit and personal computer can be connected via the following.

- RS-422
- RS-232C

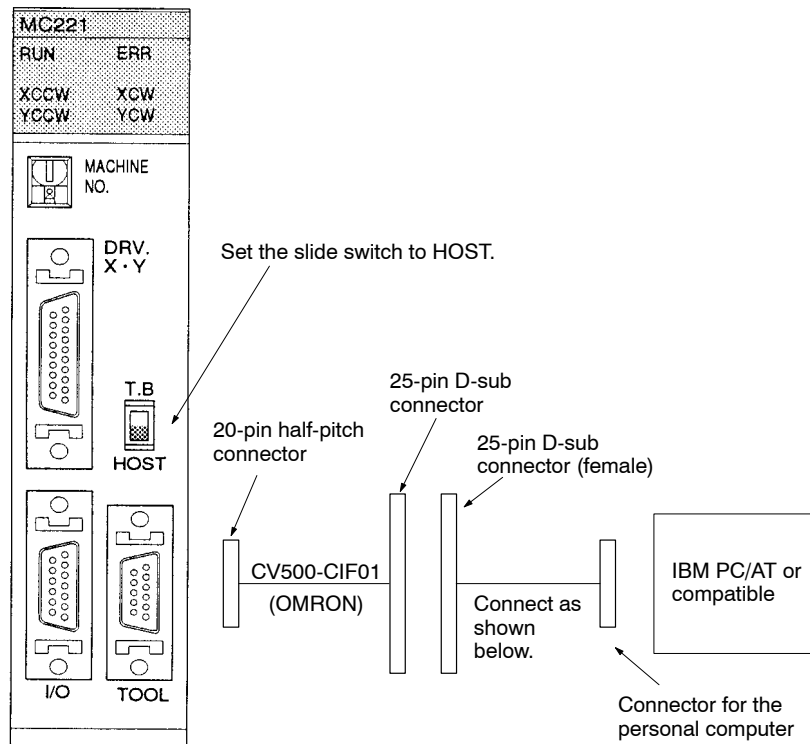
Prepare an RS-232C cable before connecting the MC Unit and personal computer via RS-232C.

The following are the communications conditions of the MC Unit and personal computer via RS-422 or RS-232C.

Communications rate: 9,600 bps  
 Parity: None  
 Data length: 8 bits  
 Stop bits: 2 bits  
 Response monitor time: 10 s

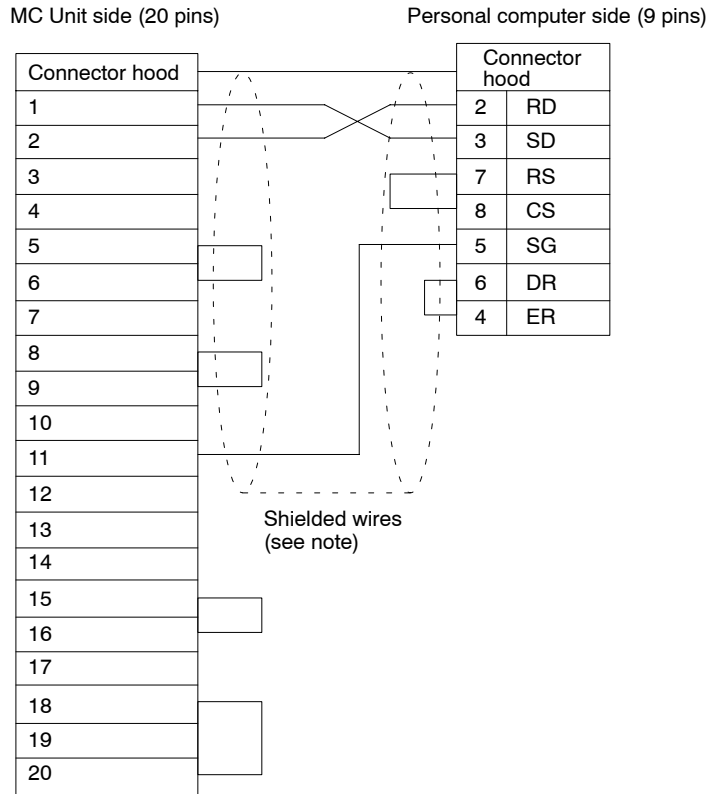
#### RS-422

The following connections are possible.



RS-232C

Wire the following.



**Note** Connect the shielded line to the FG (pin 1) at the computer, and to the connector hood at the MC Unit.

### 3-2 Connecting the Teaching Box

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

#### Connecting Cables

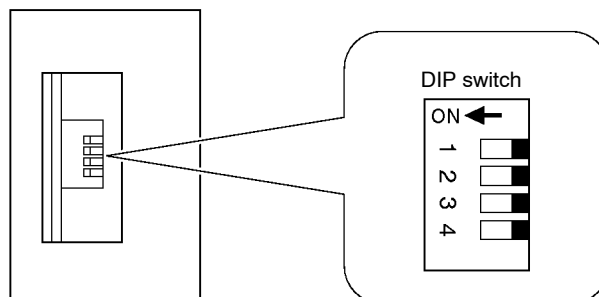
Use the Connecting Cables listed below (separately sold) to connect the Teaching Box to the MC Unit.

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

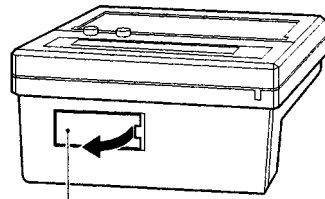
#### Connection Procedure

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

- 1, 2, 3... 1. To display messages in Japanese on the Teaching Box, set pin 2 of the DIP switch on the rear panel of the MC Unit to OFF.

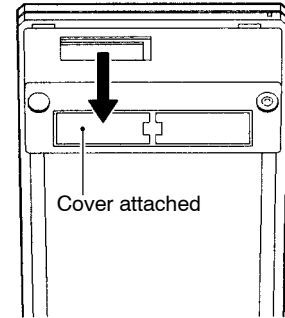


2. Remove the connector cover.

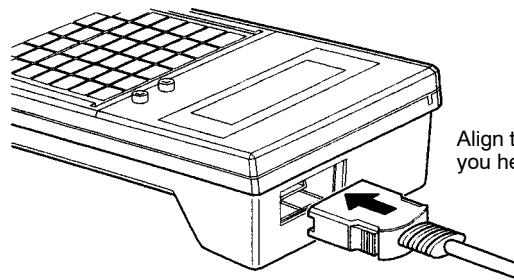


Open the cover using a small-size flat-blade screw.

Attach the removed cover to the rear surface not to lose it.

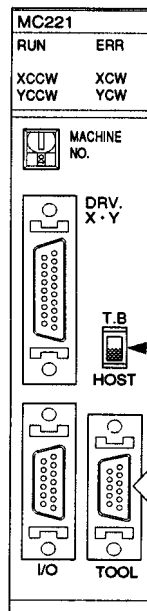


3. Plug the Connecting Cable into the connector.



Align the connector and push it in until you hear a click.

4. Plug the Connecting Cable into the MC Unit connector marked "TOOL."

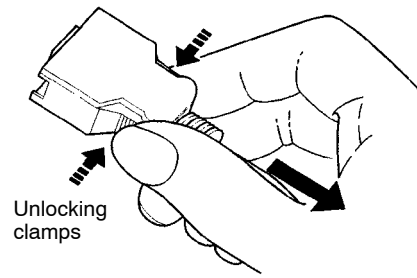


Peripheral device switch

5. Set the peripheral device switch to T.B.

**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 .....	30
4-2	System Parameters .....	30
4-2-1	Setting Parameters .....	31
4-2-2	Parameter Lists .....	31
4-3	Data Transmission and Reception .....	33
4-3-1	Types of Data .....	33
4-3-2	Types of Data Transmission and Reception .....	33
4-3-3	Summary of I/O Transfer .....	34
4-3-4	Summary of Intelligent Transfer .....	35
4-4	Details of Data Transmission and Reception Units .....	36
4-4-1	Data Transmission and Reception Units .....	36
4-4-2	Details of Data .....	37
4-4-3	Details of System Parameters .....	37
4-4-4	Details of Monitoring Data .....	48
4-4-5	Details of Special Data .....	51
4-5	Details of I/O Transfer and Necessary Information .....	51
4-5-1	Instructions and Data .....	51
4-5-2	Common Data for Transmission and Reception .....	51
4-5-3	Transmission Bits and Flags .....	52
4-5-4	Reception Bits and Flags .....	52
4-6	Data Transfer Examples .....	52
4-6-1	Data Transmission .....	52
4-6-2	Data Reception .....	54
4-7	I/O Transfer Timing .....	56
4-8	Details of Intelligent Transfer .....	58
4-9	IOWR Instruction .....	59
4-9-1	Designation .....	59
4-9-2	Flags .....	60
4-9-3	Transfer Example .....	60
4-9-4	Execution Timing of IOWR Instruction .....	62
4-9-5	Transfer Time .....	63
4-10	IORD Instruction .....	64
4-10-1	Designation .....	64
4-10-2	Flags .....	65
4-10-3	Reception Example .....	65
4-10-4	Execution Timing of IORD Instruction .....	67
4-10-5	Transfer Time .....	68

## 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 rate, 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 (mechanical parameters, coordinate system parameters, feed-rate parameters, zone parameters, and servo 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 (2 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-rate parameters	Specifies feed-rate information, such as the maximum feed rate and maximum interpolation feed rate.
	Zone parameters	Specifies zone information.
	Servo parameters	Specifies servo system information such as the position loop gain and in-position.

## 4-2-1 Setting Parameters

Set the parameters with the MC Support Software and then transfer them to the MC Unit.

**Note** 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 rate, 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-rate parameters	Maximum feed rate Max. interpolation feed rate High-speed origin search speed Low-speed origin search speed Maximum JOG feed rate	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.

### 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 (+).
Emergency 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 whether the encoder used is the ABS type (absolute values) or 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.

Parameter name	Function
Stroke limit (-) Stroke limit (+)	Sets the negative and positive stroke limits.
Origin search method	Specifies one of the 3 origin search methods.
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	Decide to whether or not to perform wiring check at the time of servo lock.
Wiring check time	Sets the wiring check time.
Wiring check pulses	Sets the number of pulses used in the wiring check.
ABS encoder initialization value	The ABS encoder initialization value 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.

### Feed-rate Parameters

Parameter name	Function
Maximum feed rate	Sets the maximum feed rate for each axis in PTP operation.
Maximum interpolation feed rate	Sets the maximum feed rate 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 rate	Sets the maximum JOG feed rate.
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.
Backlash compensation	Set the backlash compensation value.

## 4-3 Data Transmission and Reception

This section provides outlines of data transfer between a ladder program and the MC Unit.

### 4-3-1 Types of Data

Four types of data can be transferred.

- Position data
- System parameters
- Monitoring data (for the current position and number of the program being executed)
- Special data (regarding the flash memory)

Position data can be read and written. Monitoring data and part of system parameters can be read only.

The above types of data are allocated to the following addresses of the MC Unit.

Address	Data
0000 to 1999	Position data The addresses correspond to A0000 to A1999.
4000 to 5999	System parameters
6000 to 6099	Monitoring data
6100 to 6101	Special data

The above addresses must be allocated to enable a ladder program to transfer data with the MC Unit.

### 4-3-2 Types of Data Transmission and Reception

The following two types of data transmission and reception are available to a ladder program.

#### I/O Transfer

I/O transfer makes it possible to transfer data of 6,000 word max. by using the PC's I/O refresh data, initial setting, and expansion data areas of the interface area.

Use I/O transfer to transmit or receive a large volume of position data.

#### Intelligent Transfer

The IOWR and IORD instructions of a ladder program make it possible to transfer a maximum of 128-word data.

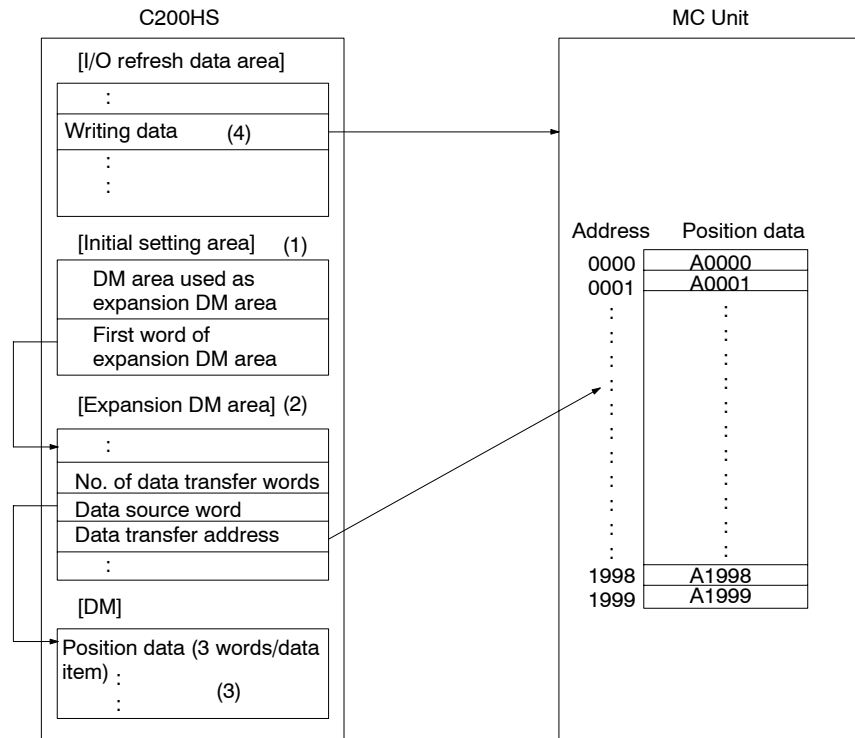
The MC Unit can use these instructions if the MC Unit is mounted to the C200HX, C200HG, or C200HE.

Intelligent transfer can transmit or receive data faster than I/O transfer.

Use intelligent transfer to transmit or receive a small volume of position data at high speed.

### 4-3-3 Summary of I/O Transfer

In the following example, position data is transferred from the C200HS to the MC Unit.



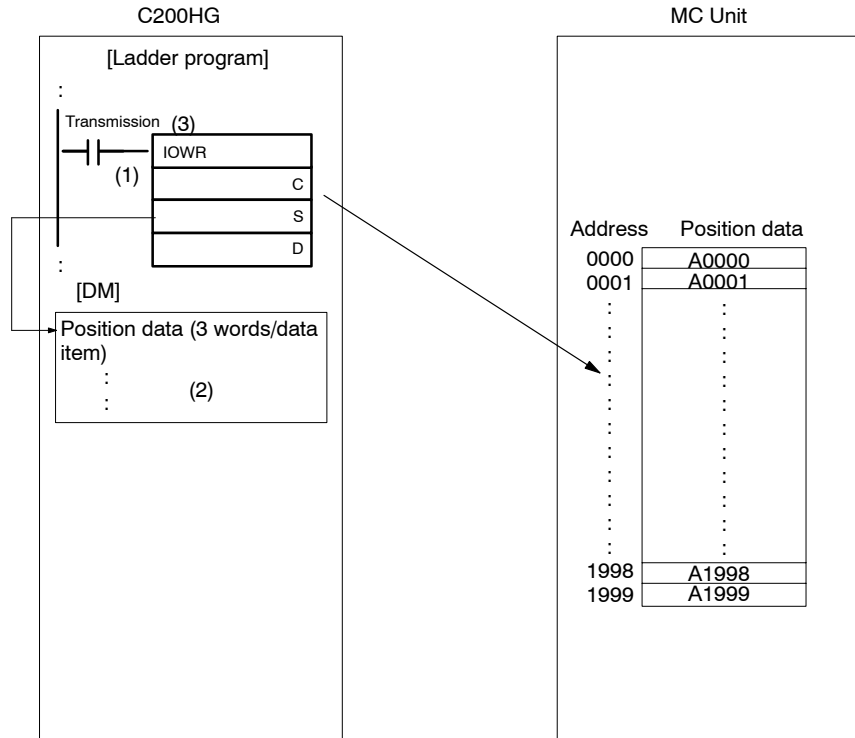
- 1, 2, 3...**
- Set the following data with the initial setting area. The data will be effective when power is supplied to the MC Unit or the MC Unit restarts.
    - Choose a DM area for the expansion data area.
    - Set the first DM word to be allocated to the expansion data area.
  - Set the following data with the expansion data area.
    - The total number of words of the position data to be transmitted (i.e., the number of data transfer words).  
A transmission data item uses three words.
    - The first word (i.e., the data source word) of the DM area that has the position data to be transmitted.
    - The destination address of the position data (i.e., the transfer address of data).
  - Set in the DM area the position data to be transmitted.
  - The position data can be transmitted if the transmission control bit of the I/O refresh area is set to ON.

Position data can be received in the same way.

Other data can be transferred in the same way although a single data item is transferred at a time.

### 4-3-4 Summary of Intelligent Transfer

In the following example, position data is transferred from the C200HG to the MC Unit.



- 1, 2, 3...**
1. Set the following data with the operand of the IOWR instruction.
    - The destination address (C) of the position data.
    - The first word (S) of the DM area that has the position data to be transmitted.
    - The unit number of the destination and the total number of words (D) of the position data to be transmitted.  
A transmission data item uses three words.
  2. Set in the DM area the position data to be transmitted.
  3. Set a bit as a transmission bit. The position data can be transmitted if the transmission bit is set to ON.  
Position data can be received in the same way.  
Other data can be transferred in the same way although a single data item is transferred at a time.

## 4-4 Details of Data Transmission and Reception Units

### 4-4-1 Data Transmission and Reception Units

This section explains units used for data transmission and reception.

Data	Description																														
<p>Position data</p>	<p>Three words are used to transmit or receive a position data item. The following is the configuration of a BCD position data item.</p> <p>Decimal point A = 0 (Indicates 1) = 1 (Indicates 0.1) = 2 (Indicates 0.01) = 3 (Indicates 0.001) = 4 (Indicates 0.0001)</p> <p>3 2 1 0 bit x10<sup>7</sup></p> <p>Sign bit 0: Positive 1: Negative</p> <p>Example 1: The following is the configuration of a position data value of 123.45.</p> <table border="1" style="margin-left: 40px;"> <tr><td>j+0</td><td>0</td><td>0</td><td>0</td><td>2</td></tr> <tr><td>j+1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr> <tr><td>j+2</td><td>0</td><td>0</td><td>0</td><td>1</td></tr> </table> <p>Example 2: The following is the configuration of a position data value of 12345.</p> <table border="1" style="margin-left: 40px;"> <tr><td>j+0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>j+1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr> <tr><td>j+2</td><td>8</td><td>0</td><td>0</td><td>1</td></tr> </table> <p>A position data item uses three words. Therefore, the total words of position data to be transmitted or received are as follows: Total words = 3 words x No. of data items</p>	j+0	0	0	0	2	j+1	2	3	4	5	j+2	0	0	0	1	j+0	0	0	0	0	j+1	2	3	4	5	j+2	8	0	0	1
j+0	0	0	0	2																											
j+1	2	3	4	5																											
j+2	0	0	0	1																											
j+0	0	0	0	0																											
j+1	2	3	4	5																											
j+2	8	0	0	1																											
<p>System parameters Monitoring data Special data</p>	<p>Two words are used to transmit or receive a system parameter, monitoring data item, or special data item. The following is the configuration of a parameter, monitoring data item, or special data item.</p> <p>Data may be BCD or binary.</p> <p>Word (j: Optional)</p> <p>Unlike position data, only one data item can be transmitted or received at a time. Therefore, the total number of words for transmitting or receiving is as follows: Total number of words = 2 (fixed)</p>																														



### 4-4-2 Details of Data

Refer to the following for details of system parameters, monitor data, and special data.

Data		First address	
System parameter	Unit parameter	4000	
	Memory management parameter	4100	
	Mechanical parameter	X-axis	4200
		Y-axis	4225
	Coordinate system parameter	X-axis	4300
		Y-axis	4325
	Feed-rate parameter	X-axis	4400
		Y-axis	4425
	Zone parameter	X-axis	4500
		Y-axis	4525
Servo parameter	X-axis	4600	
	Y-axis	4625	
Monitoring data		6000	
Special data		6100	

### 4-4-3 Details of System Parameters

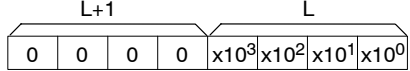
Refer to the following for the abbreviations used in the following tables.

- R: Parameters that can be read only.
- W: Parameters that can be written only.
- R/W: Parameters that can be read and written.

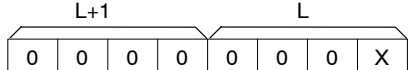
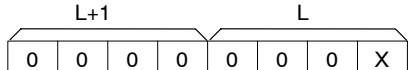
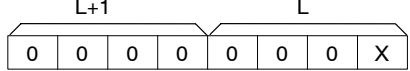
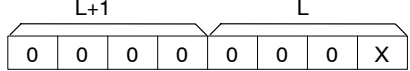
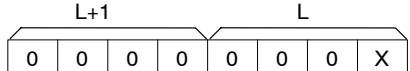
#### Unit Parameters

Axis	Address	R/W	Name	Description
Common	4000	R	Axis configuration	<div style="text-align: center;"> <math>L+1</math>                      <math>L</math>  </div> <p>Specifies the number of axis used with the MC Unit.                      X = 1: A single axis is used.                      X = 2: Two axes are used.</p>
	4001	R	No. of tasks	<div style="text-align: center;"> <math>L+1</math>                      <math>L</math>  </div> <p>Specifies the number of tasks used with the MC Unit.                      X = 1: A single task is used.                      X = 2: Two tasks are used.</p>
	4002	R	Task axis declaration (for task 1)	<div style="text-align: center;"> <math>L+1</math>                      <math>L</math>  </div> <p>Specifies the axis or axes used for task 1.                      X = 0: No axis is used.                      X = 1: X axis is used.                      X = 2: Y axis is used.                      X = 3: X and Y axes are used.</p>
	4003	R	Task axis declaration (for task 2)	<p>Specifies the axis or axes used for task 2.                      There is no difference in data configuration or range between task axis declaration for task 1 and that for task 2.</p>

Memory Management Parameters

Axis	Address	R/W	Name		Description
Common	4100	R	For task 1	First position data item no.	First position data item no. 
	4101			Last position data item no.	
	4102	R	For task 2	First position data item no.	Specifies the first and last numbers of position data items for task 2. There is no difference in data configuration or range between the position data items used for task 1 and those used for task 2.
	4103			Last position data item no.	

Mechanical Parameters

Axis	Address	R/W	Name	Description
X	4200	R	Minimum unit setting	 <p>Specifies the minimum unit (i.e., the decimal position) for the mechanical system.</p> <p>X = 0 (1)                      X = 1 (0.1)                      X = 2 (0.01)                      X = 3 (0.001)                      X = 4 (0.0001)</p>
	4201	R	Display unit	 <p>Specifies the unit displayed with the MC Support Software.</p> <p>X = 0 (mm)                      X = 1 (inch)                      X = 2 (deg)                      X = 3 (Pulse)</p>
	4202	R	Rotation direction	 <p>Specifies the rotation direction of the motor.</p> <p>X = 0: CW with positive voltage.                      X = 1: CCW with positive voltage.</p>
	4203	R	Emergency stop method	 <p>Specifies the method to stop the MC Unit immediately.</p> <p>X = 0: Instantaneous 0 V output.                      X = 1: Pulse accumulation.</p>
	4204	R	Encoder ABS/INC	 <p>Specifies the type of encoder.</p> <p>X = 0: INC encoder                      X = 1: ABS encoder</p>

Axis	Address	R/W	Name		Description
X	4205	R	Encoder resolution		<p>Specifies the resolution of the encoder within a range of 1 to 65,535 ppr.</p>
	4207	R	Encoder polarity		<p>Specifies the polarity of the encoder.                      X = 0: CW with increments.                      X = 1: CCW with decrements.</p>
	4208	R	Pulse rate	Numerator	<p>Numerator</p> <p>Denominator</p> <p>Specifies the numerator and denominator of the pulse rate within a range of 1 to 10,000 to determine the conditions of encoder feedback pulse 1 (i.e., mm/pulse, deg/pulse, inch/pulse, and pulses/pulse) within the following range.  <math>1/100,000 \leq \text{Numerator/Denominator} \leq 1</math></p>
	4209		Denominator		
	4210	R	Max. motor frequency		<p>Specifies the maximum rpm rating for the motor being used within a range of 1 to 32,767 rpm.</p>
4213	R	Stroke limit (-)		<p>Stroke limit (-)</p> <p>Stroke limit (+)</p> <p>Specifies the negative and positive stroke limits within a range of -39,999,999 to 39,999,999. The minimum setting unit is set to 2 (for 0.01), the display unit is set to 0 (for mm), and the negative stroke limit is 3.99 mm (i.e., 399 x 0.01) if the data is 399.</p>	
4214		Stroke limit (+)			

Axis	Address	R/W	Name	Description
X	4215	R	Origin search method	<p>Specifies the method to search the origin.                      X = 0: Origin mode when power is turned ON.                      X = 1: Limit reverse mode                      X = 2: Single direction mode</p>
	4216	R	Origin search direction	<p>Specifies the origin search direction.                      X = 0: Positive direction                      X = 1: Negative direction</p>
	4217	R	Origin deceleration method	<p>Specifies the deceleration method near the origin.                      X = 0: With origin proximity input.                      X = 1: With origin proximity input and limit input.</p>
	4218	R	Origin proximity input logic	<p>Specifies the origin proximity input logic.                      X = 0: Normally open.                      X = 1: Normally closed.</p>
	4219	R	Wiring check	<p>Specifies whether the wiring check function is used.                      X = 0: The wiring check function is not used.                      X = 1: The wiring check function is used.</p>
	4220	R	Wiring check time	<p>Specifies the wiring check time within a range of 0 to 99 (x 10 ms).</p>
	4221	R	Wiring check pulses	<p>Specifies the number of pulses to check the wiring within a range of 0 to 999 pulses.</p>
	4222	R	ABS initialization	<p>Indicates the compensation value of the absolute encoder obtained when the ABS initialization function is executed. The value ranges from -32767 to +32767.</p>
	4223	R	ABS soft reset	<p>Indicates the compensation value of the origin shift obtained when the ABS Soft Reset function is executed. The data configuration and range are the same as for the absolute encoder initial set value.</p>

Axis	Address	R/W	Name	Description
Y	4225	R	Minimum unit setting	Refer to the corresponding X-axis parameters.
	4226	R	Display unit	
	4227	R	Rotation direction	
	4228	R	Emergency stop method	
	4229	R	Encoder ABS/INC	
	4230	R	Encoder resolution	
	4232	R	Encoder polarity	
	4233	R	Numerator pulse rate	
	4234	R	Denominator pulse rate	
	4235	R	Max. motor frequency	
	4238	R	Negative stroke limit	
	4239	R	Positive stroke limit	
	4240	R	Origin search method	
	4241	R	Origin search direction	
	4242	R	Origin deceleration method	
	4243	R	Origin proximity input logic	
	4244	R	Wiring check	
	4245	R	Wiring check time	
	4246	R	Wiring check pulses	
4247	R	ABS initialization		
4248	R	ABS soft reset		

Coordinate System Parameters

Axis	Address	R/W	Name	Description
X	4300	R/W	Reference origin offset value	<p>3 2 1 0 bit x10<sup>7</sup></p> <p>Sign bit 0: Positive 1: Negative</p> <p>Sets the reference origin offset value within a range of -39,999,999 to 39,999,999. The minimum setting unit is set to 2 (for 0.01), the display unit is set to 0 (for mm), and the negative stroke limit is 3.99 mm (i.e., 399 x 0.01) if the data is 399. When this value is changed, it will become valid from the time when the next origin search is executed.</p>
	4301	R/W	Workpiece origin offset value	<p>Sets the offset value of the workpiece origin. There is no difference in data configuration, range, or unit between the workpiece origin offset value and reference origin offset value. When this value is changed, it will become valid from the time when the next program is executed from the first block.</p>
Y	4325	R/W	Reference origin offset value	Refer to the corresponding X-axis parameters.
	4326	R/W	Workpiece origin offset value	

Feed-rate Parameters

Axis	Address	R/W	Name	Description
X	4400	R	Maximum feed rate	<p>Specifies the upper limit of the feed rate with the following units used.                      0: mm/s                      1: inch/s                      2: deg/s                      3: Pulses/s</p> <p>The range varies with the encoder resolution, pulse rate, and display unit.                      The minimum setting unit is set to 2 (for 0.01), the display unit is set to 0 (for mm), the maximum feed rate is 3.99 mm/s (i.e., 399 x 0.01) if the data is 399.</p>
	4401	R	Maximum interpolation feed rate	Specifies the upper limit of the interpolation feed rate. There is no difference in data configuration, range, or unit between the maximum interpolation feed rate and maximum feed rate.
	4402	R	High-speed origin search speed	Specifies the high-speed origin search speed. There is no difference in data configuration, range, or unit between the high-speed origin search speed and maximum feed rate.
	4403	R	Low-speed origin search speed	Specifies the low-speed origin search speed. There is no difference in data configuration, range, or unit between the low-speed origin search speed and maximum feed rate.
	4404	R	Maximum JOG feed rate	Specifies the upper limit of the JOG feed rate. There is no difference in data configuration, range, or unit between the maximum JOG feed rate and maximum feed rate.
	4405	R	Acceleration/deceleration curve	<p>Specifies the type of acceleration/deceleration curve.                      X = 0: Trapezoidal curve                      X = 1: S-shaped curve</p>
	4406	R/W	Acceleration time	<p>Sets the acceleration time to obtain the maximum feed rate within a range of 0 to 9,998 ms.</p>
	4407	R/W	Deceleration time	Sets the deceleration time to reduce the feed rate to zero from the maximum feed rate. There is no difference in data configuration, range, or unit between the deceleration time and maximum feed rate.
	4408	R/W	Interpolation acceleration time	Sets the acceleration time to obtain the maximum interpolation feed rate. There is no difference in data configuration, range, or unit between the deceleration time and interpolation acceleration time.

Axis	Address	R/W	Name	Description								
X	4409	R/W	Interpolation deceleration time	Sets the deceleration time to reduce the interpolation feed rate to zero from the maximum interpolation feed rate. There is no difference in data configuration, range, or unit between the interpolation deceleration time and deceleration time.								
	4410	R/W	MPG factor 1	<div style="text-align: center;"> <math display="block">\overbrace{\quad\quad\quad}^{L+1} \quad \overbrace{\quad\quad\quad}^L</math> <table border="1" style="margin: auto;"> <tr> <td>0</td><td>0</td><td>0</td><td>0</td> <td><math>\times 10^3</math></td><td><math>\times 10^2</math></td><td><math>\times 10^1</math></td><td><math>\times 10^0</math></td> </tr> </table> </div> Sets the factor of an MPG pulse within a range of 1 to 1,000 as set value 1.	0	0	0	0	$\times 10^3$	$\times 10^2$	$\times 10^1$	$\times 10^0$
	0	0	0	0	$\times 10^3$	$\times 10^2$	$\times 10^1$	$\times 10^0$				
	4412	R/W	MPG factor 2	Sets the factor for an MPG pulse as set value 2. There is no difference in data configuration or range between MPG factors 2 and 1.								
	4414	R/W	MPG factor 3	Sets the factor for an MPG pulse as set value 3. There is no difference in data configuration or range between MPG factors 3 and 1.								
4416	R/W	MPG factor 4	Sets the factor for an MPG pulse as set value 4. There is no difference in data configuration or range between MPG factors 4 and 1.									
Y	4425	R	Maximum feed rate	Refer to the corresponding X-axis parameters.								
	4426	R	Maximum interpolation feed rate									
	4427	R	High-speed origin search speed									
	4428	R	Low-speed origin search speed									
	4429	R	Maximum JOG feed rate									
	4430	R	Acceleration/Deceleration curve									
	4431	R/W	Acceleration time									
	4432	R/W	Deceleration time									
	4433	R/W	Interpolation acceleration time									
	4434	R/W	Interpolation deceleration time									

**Note** When the above parameter is changed, it becomes valid from the following timing.

- Acceleration time/deceleration time (in the manual mode): When the next new process is started.
- Acceleration time/deceleration time (in the automatic mode): When the next program is started from the first block.
- Interpolation acceleration time/interpolation deceleration time: When the next program is started from the first block.
- MPG factors: When the next new process is started.

Zone Parameters

When the zone parameter is changed, the changed value will become valid in real time.

Axis	Address	R/W	Name	Description
X	4500	R/W	Zone specification	<p>Validates or invalidates the settings of zones 1 to 8. A zone is valid if a bit corresponding to the zone is set to 1 and invalid if the bit is set to 0.</p>
	4501	R/W	Zone 1	<p>Negative direction set value</p> <p>Zone 1: Positive direction set value</p>
	4502	R/W		<p>Positive direction set value</p> <p>Sets the negative or positive direction range of zone 1.                  The set value is effective if the specification of the zone is valid. The set value can be set within a range of -39,999,999 to 39,999,999 and must satisfy the following condition.                  Negative direction set value ≤ Positive directional set value                  The minimum setting unit is set to 2 (for 0.01), the display unit is set to 0 (for mm), and the negative direction set value is 3.99 mm (i.e., 399 x 0.01) if the negative direction data is 399.</p>



Axis	Address	R/W	Name		Description
X	4503	R/W	Zone 2	Negative direction set value	Sets the negative or positive direction range of zone 2. There is no difference in data configuration, range, or unit between the negative or positive direction set values of zones 2 and 1.
	4504	R/W		Positive direction set value	
	4505	R/W	Zone 3	Negative direction set value	Sets the negative or positive direction range of zone 3. There is no difference in data configuration, range, or unit between the negative or positive direction set values of zones 3 and 1.
	4506	R/W		Positive direction set value	
	4507	R/W	Zone 4	Negative direction set value	Sets the negative or positive direction range of zone 4. There is no difference in data configuration, range, or unit between the negative or positive direction set values of zones 4 and 1.
	4508	R/W		Positive direction set value	
	4509	R/W	Zone 5	Negative direction set value	Sets the negative or positive direction range of zone 5. There is no difference in data configuration, range, or unit between the negative or positive direction set values of zones 5 and 1.
	4510	R/W		Positive direction set value	
	4511	R/W	Zone 6	Negative direction set value	Sets the negative or positive direction range of zone 6. There is no difference in data configuration, range, or unit between the negative or positive direction set values of zones 6 and 1.
	4512	R/W		Positive direction set value	
	4513	R/W	Zone 7	Negative direction set value	Sets the negative or positive direction range of zone 7. There is no difference in data configuration, range, or unit between the negative or positive direction set values of zones 7 and 1.
	4514	R/W		Positive direction set value	
	4515	R/W	Zone 8	Negative direction set value	Sets the negative or positive direction range of zone 8. There is no difference in data configuration, range, or unit between the negative or positive direction set values of zones 8 and 1.
	4516	R/W		Positive direction set value	

Axis	Address	R/W	Name		Description
Y	4525	R/W	Zone specification		Refer to the corresponding X-axis parameters.
	4526	R/W	Zone 1	Negative direction set value	
	4527	R/W		Positive direction set value	
	4528	R/W	Zone 2	Negative direction set value	
	4529	R/W		Positive direction set value	
	4530	R/W	Zone 3	Negative direction set value	
	4531	R/W		Positive direction set value	
	4532	R/W	Zone 4	Negative direction set value	
	4533	R/W		Positive direction set value	
	4534	R/W	Zone 5	Negative direction set value	
	4535	R/W		Positive direction set value	
	4536	R/W	Zone 6	Negative direction set value	
	4537	R/W		Positive direction set value	
	4538	R/W	Zone 7	Negative direction set value	
	4539	R/W		Positive direction set value	
	4540	R/W	Zone 8	Negative direction set value	
	4541	R/W		Positive direction set value	

Servo Parameters

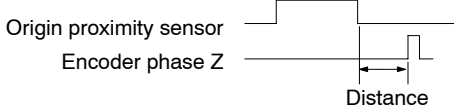
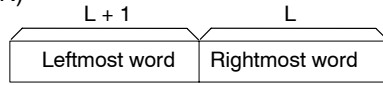
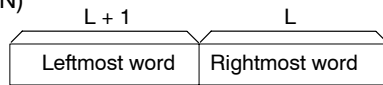
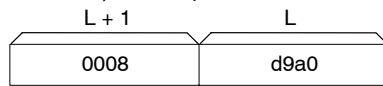
When the servo parameter is changed, the changed value will become valid in real time.

Axis	Address	R/W	Name	Description
X	4600	R/W	Accumulated pulses warning value	<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 2px;"> <span style="margin-right: 5px;">L+1</span> <span style="margin-right: 5px;">0</span> <span style="margin-right: 5px;">0</span> <span style="margin-right: 5px;">0</span> <span style="margin-right: 5px;">x10<sup>4</sup></span> </div> <div style="border: 1px solid black; padding: 2px;"> <span style="margin-right: 5px;">L</span> <span style="margin-right: 5px;">x10<sup>3</sup></span> <span style="margin-right: 5px;">x10<sup>2</sup></span> <span style="margin-right: 5px;">x10<sup>1</sup></span> <span style="margin-right: 5px;">x10<sup>0</sup></span> </div> </div> <p>Sets the accumulated pulses warning value within a range of 0 to 65,000 pulses.</p>
	4601	R/W	In-position	<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 2px;"> <span style="margin-right: 5px;">L+1</span> <span style="margin-right: 5px;">0</span> <span style="margin-right: 5px;">0</span> <span style="margin-right: 5px;">0</span> <span style="margin-right: 5px;">0</span> </div> <div style="border: 1px solid black; padding: 2px;"> <span style="margin-right: 5px;">L</span> <span style="margin-right: 5px;">0</span> <span style="margin-right: 5px;">x10<sup>2</sup></span> <span style="margin-right: 5px;">x10<sup>1</sup></span> <span style="margin-right: 5px;">x10<sup>0</sup></span> </div> </div> <p>Sets to check the in-position within a range of 0 to 999 pulses.</p>
	4602	R/W	Position loop gain	<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 2px;"> <span style="margin-right: 5px;">L+1</span> <span style="margin-right: 5px;">0</span> <span style="margin-right: 5px;">0</span> <span style="margin-right: 5px;">0</span> <span style="margin-right: 5px;">0</span> </div> <div style="border: 1px solid black; padding: 2px;"> <span style="margin-right: 5px;">L</span> <span style="margin-right: 5px;">0</span> <span style="margin-right: 5px;">x10<sup>2</sup></span> <span style="margin-right: 5px;">x10<sup>1</sup></span> <span style="margin-right: 5px;">x10<sup>0</sup></span> </div> </div> <p>Sets the position loop gain within a range of 5/s to 250/s.</p>
	4603	R/W	Position loop FF gain	<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 2px;"> <span style="margin-right: 5px;">L+1</span> <span style="margin-right: 5px;">0</span> <span style="margin-right: 5px;">0</span> <span style="margin-right: 5px;">0</span> <span style="margin-right: 5px;">0</span> </div> <div style="border: 1px solid black; padding: 2px;"> <span style="margin-right: 5px;">L</span> <span style="margin-right: 5px;">0</span> <span style="margin-right: 5px;">x10<sup>2</sup></span> <span style="margin-right: 5px;">x10<sup>1</sup></span> <span style="margin-right: 5px;">x10<sup>0</sup></span> </div> </div> <p>Sets the position loop FF gain within a range of 0% to 100%.</p>
	4604	R/W	Backlash compensation value	<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="border: 1px solid black; padding: 2px;"> <span style="margin-right: 5px;">L+1</span> <span style="margin-right: 5px;">0</span> <span style="margin-right: 5px;">0</span> <span style="margin-right: 5px;">0</span> <span style="margin-right: 5px;">0</span> </div> <div style="border: 1px solid black; padding: 2px;"> <span style="margin-right: 5px;">L</span> <span style="margin-right: 5px;">0</span> <span style="margin-right: 5px;">x10<sup>2</sup></span> <span style="margin-right: 5px;">x10<sup>1</sup></span> <span style="margin-right: 5px;">x10<sup>0</sup></span> </div> </div> <p>Sets the backlash compensation value within a range of 0 to 999 pulses.</p>
Y	4625	R/W	Accumulated pulses warning value	Refer to the corresponding X-axis parameters.
	4626	R/W	In-position	
	4627	R/W	Position loop gain	
	4628	R/W	Position loop FF gain	
	4629	R/W	Backlash compensation value	

4-4-4 Details of Monitoring Data

Axis	Address	R/W	Name	Description																																																																					
Error	6000	R	System error code	<div style="text-align: center;"> <table border="1" style="margin: auto;"> <tr> <td colspan="4" style="text-align: center;">L+1</td> <td colspan="4" style="text-align: center;">L</td> </tr> <tr> <td style="text-align: center;">0</td><td style="text-align: center;">0</td><td style="text-align: center;">0</td><td style="text-align: center;">0</td> <td style="text-align: center;"><math>\times 10^3</math></td><td style="text-align: center;"><math>\times 10^2</math></td><td style="text-align: center;"><math>\times 10^1</math></td><td style="text-align: center;"><math>\times 10^0</math></td> </tr> </table> </div> <p>Outputs system error codes. An error code will be set if a system error results. The address is set to 0 when the system is normal.</p>	L+1				L				0	0	0	0	$\times 10^3$	$\times 10^2$	$\times 10^1$	$\times 10^0$																																																					
	L+1				L																																																																				
	0	0	0	0	$\times 10^3$	$\times 10^2$	$\times 10^1$	$\times 10^0$																																																																	
	6001	R	Task 1 error code	Outputs task 1 error codes. An error code will be set if a task 1 error results. An axis error code will be set but the address will be set to 0 if the axis controlled by task 1 has an error. There is no difference in data configuration between task 1 and system error codes.																																																																					
	6002	R	Task 2 error code	Outputs task 2 error codes. There is no difference in error code condition between tasks 2 and 1 error codes.																																																																					
	6005	R	X-axis error code	Outputs X-axis error codes. An error code will be set if an X-axis error results. There is no difference in data configuration between X-axis and system error codes.																																																																					
6006	R	Y-axis error code	Outputs Y-axis error codes An error code will be set if a Y-axis error results. There is no difference in data configuration between Y-axis and system error codes.																																																																						
6011	R	I/O monitor data	<p>Outputs the ON or OFF status of each I/O signal of the MC Unit.</p> <div style="text-align: center;"> <table border="1" style="margin: auto;"> <tr> <td colspan="4" style="text-align: center;">L+1</td> <td colspan="4" style="text-align: center;">L</td> </tr> <tr> <td colspan="4" style="text-align: center;">Y-axis I/O monitor data</td> <td colspan="4" style="text-align: center;">X-axis I/O monitor data</td> </tr> </table> </div> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Bit</th> <th></th> <th>Name</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td rowspan="16" style="text-align: center; vertical-align: middle;">X-axis</td> <td>General-purpose input 1</td> <td rowspan="16" style="text-align: center; vertical-align: middle;">1: ON 0: OFF</td> </tr> <tr> <td>1 to 7</td> <td>Not used (OFF)</td> </tr> <tr> <td>8</td> <td>CCW limit input</td> </tr> <tr> <td>9</td> <td>CW limit input</td> </tr> <tr> <td>10</td> <td>Origin proximity input</td> </tr> <tr> <td>11</td> <td>Emergency stop input</td> </tr> <tr> <td>12</td> <td>Driver alarm input</td> </tr> <tr> <td>13</td> <td>Run instruction output</td> </tr> <tr> <td>14</td> <td>Driver alarm reset output</td> </tr> <tr> <td>15</td> <td>Sensor ON output</td> <td>1: ON</td> </tr> </tbody> </table> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Bit</th> <th></th> <th>Name</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td rowspan="16" style="text-align: center; vertical-align: middle;">Y-axis</td> <td>General-purpose input 2</td> <td rowspan="16" style="text-align: center; vertical-align: middle;">1: ON 0: OFF</td> </tr> <tr> <td>1 to 7</td> <td>Not used (OFF)</td> </tr> <tr> <td>8</td> <td>CCW limit input</td> </tr> <tr> <td>9</td> <td>CW limit input</td> </tr> <tr> <td>10</td> <td>Origin proximity input</td> </tr> <tr> <td>11</td> <td>Emergency stop input</td> </tr> <tr> <td>12</td> <td>Driver alarm input</td> </tr> <tr> <td>13</td> <td>Run instruction output</td> </tr> <tr> <td>14</td> <td>Driver alarm reset output</td> </tr> <tr> <td>15</td> <td>Sensor ON output</td> <td>1: ON</td> </tr> </tbody> </table>	L+1				L				Y-axis I/O monitor data				X-axis I/O monitor data				Bit		Name	Description	0	X-axis	General-purpose input 1	1: ON 0: OFF	1 to 7	Not used (OFF)	8	CCW limit input	9	CW limit input	10	Origin proximity input	11	Emergency stop input	12	Driver alarm input	13	Run instruction output	14	Driver alarm reset output	15	Sensor ON output	1: ON	Bit		Name	Description	0	Y-axis	General-purpose input 2	1: ON 0: OFF	1 to 7	Not used (OFF)	8	CCW limit input	9	CW limit input	10	Origin proximity input	11	Emergency stop input	12	Driver alarm input	13	Run instruction output	14	Driver alarm reset output	15	Sensor ON output	1: ON
L+1				L																																																																					
Y-axis I/O monitor data				X-axis I/O monitor data																																																																					
Bit		Name	Description																																																																						
0	X-axis	General-purpose input 1	1: ON 0: OFF																																																																						
1 to 7		Not used (OFF)																																																																							
8		CCW limit input																																																																							
9		CW limit input																																																																							
10		Origin proximity input																																																																							
11		Emergency stop input																																																																							
12		Driver alarm input																																																																							
13		Run instruction output																																																																							
14		Driver alarm reset output																																																																							
15		Sensor ON output		1: ON																																																																					
Bit				Name	Description																																																																				
0		Y-axis		General-purpose input 2	1: ON 0: OFF																																																																				
1 to 7				Not used (OFF)																																																																					
8				CCW limit input																																																																					
9				CW limit input																																																																					
10				Origin proximity input																																																																					
11	Emergency stop input																																																																								
12	Driver alarm input																																																																								
13	Run instruction output																																																																								
14	Driver alarm reset output																																																																								
15	Sensor ON output		1: ON																																																																						

Axis	Address	R/W	Name	Description	
Task 1 status	6020	R	Executing program no.	<p>If a program is being executed, the number of the program will be output and if no program is being executed, the number of the last program executed will be output within a range of 0 to 999.</p>	
	6021	R	Executing block no.	<p>If a program is being executed, the block number of the program will be output and if no program is being executed, the block number of the last program executed will be output within a range of 0 to 999.</p>	
	6022	R	Teaching execution address	<p>Outputs the address where the position data is stored at the time of teaching within a range of 0 to 1,999. The address will be refreshed whenever teaching is executed.</p>	
Task 2 status	6024	R	Executing program no.	Refer to the corresponding task 1 data.	
	6025	R	Executing block no.		
	6026	R	Teaching execution address		
Current position	6050	R	Reference coordinate system current position	X	<p>X-axis reference coordinate system current position</p> <p>Y-axis reference coordinate system current position</p> <p>Outputs the reference coordinate system current position within a range of -39,999,999 to 39,999,999. The minimum setting unit is set to 2 (for 0.01), the display unit is set to 0 (for mm), and the reference coordinate system current position is 3.99 mm (i.e., 399 x 0.01) if the data is 399.</p>
	6051	R		Y	
	6054	R	Workpiece coordinate system current position	X	Outputs the workpiece coordinate system current position. There is no difference in data configuration, range, or unit between the work coordinate system current position and reference coordinate system current position.
	6055	R		Y	

Axis	Address	R/W	Name	Description
Current position	6058	R	Reference coordinate system current position with pulses	X Outputs pulses for the workpiece coordinate system current position. There is no difference in data configuration or range between the reference coordinate system current position with pulses and reference coordinate system current position.
	6059	R		Y
	6062	R	Error counter value with pulses	X Outputs pulses for the error counter value within a range of -65,536 to 65,535. There is no difference in data configuration between the deviation counter value with pulses and reference coordinate system current position.
	6063	R		Y
	6066	R	Workpiece origin shift value	X Outputs the shift value between the reference coordinate system origin and workpiece coordinate system origin. There is no difference in data configuration, range, or unit between the workpiece origin shift value and reference coordinate system current position.
	6067	R		Y
	6070	R	Phase-Z margin with pulses	X Outputs pulses for the distance between the origin proximity sensor and encoder phase-Z signal.   There is no difference in data configuration or range between the phase-Z margin with pulses and reference coordinate system current position. This data is refreshed whenever the origin search operation completes. The distance is a slightly dispersed due to the searching speed at the time phase Z is detected. The dispersion can be obtained from the following formula. Dispersion (pulses) = Searching speed when phase Z is detected (pulses/s)/250
	6071	R		Y
6074	R	Reference coordinate system current position (BIN)	X X-axis reference coordinate system current position (BIN)  Y-axis reference coordinate system current position (BIN) 	
6075	R		Y Outputs 32-bit signed binary data pulses for the reference coordinate system current position. The following is the output configuration of the current position if 580,000 pulses (i.e., 8d9a0 in hexadecimal) are output for the current position.  Other current positions can be output within a range of -39,999,999 to 39,999,999 in BCD. This current position is output within a range of -2,147,483,648 to 2,147,483,647 pulses as binary data.	

### 4-4-5 Details of Special Data

Only intelligent transfer can be used to transmit or receive special data.

Axis	Address	R/W	Name	Description
Common	6100	R/W	Writing position data to the flash memory	<div style="text-align: center;"> </div> <p>Writes position data to the flash memory.                      X = 0: No data has been written or data has been written.                      X = 1: Data is being written or data has been written.                      Set this area to 1 to write position data to the flash memory. The area will be set to 0 when the data has been written.</p>
	6101	R/W	Writing parameters to the flash memory	<p>Writes parameters to the flash memory.                      There is no difference in usage between parameter flash memory write and position data flash memory write.</p>

## 4-5 Details of I/O Transfer and Necessary Information

I/O transfer makes it possible to transmit or receive data of 6,000 words max. or 2,000 position data items at a time.

A total of 6,000 words is used to transfer 2,000 position data items (i.e., 2,000 x 3).

Use I/O transfer if it is necessary to transfer a large number of data items.

### 4-5-1 Instructions and Data

The following information is necessary for the transmission and reception of position data using I/O transfer. Refer to *Section 6 PC Data Area Interface* for details.

### 4-5-2 Common Data for Transmission and Reception

#### Initial Setting Area

I/O transfer uses the expansion data area of the PC data area interface. Therefore, the following settings are necessary in the initial setting area.

- To set the expansion data area in the DM area or EM area (available to the C200HX and C200HG only).
- To determine the first word of the DM area or EM area to which the expansion data area is allocated.

#### Expansion Data Area

The expansion data area must contain the following data.

- Data used for transmission:
  - No. of data transfer words
  - Data source word
  - Data destination address
- Data used for reception:
  - No. of data transfer words
  - Data source address
  - Data destination word

The data items used for transmission or reception can be referred to when the transmission or reception control bit is turned ON.

The data writing source word and data reading destination word can be designated within the ranges shown in the following table.

PC model	Expansion data area
C200H	DM 0000 to DM 0899
C200HS	DM 0000 to DM 0999 DM 2000 to DM 5999
C200HX, C200HG, and C200HE	EM area (if the EM area is unavailable) DM 0000 to DM 0999 DM 2600 to DM 5999 (The upper limit depends on the memory size.) EM area (if the EM area is available) DM 0000 to DM 0999 DM 2600 to DM 5999 (The upper limit depends on the memory size.) EM 0000 to EM 6143

**Note** The DM area or EM area to be used can be specified in the initial setting area. Refer to 6-4-1 Initial Setting Area.

### 4-5-3 Transmission Bits and Flags

The following bits and flags are used for transmission.

- Transmission Control Bit** This bit is used to control transmission. It is allocated in the I/O refresh area of the PC data area interface. Data transmission begins when this bit goes ON.
- Transmission Completed Flag** This flag is turned ON when transmission is completed. It is allocated in the I/O refresh area of the PC data area interface. This flag goes ON when transmission is completed and goes OFF when the Transmission Control Bit is turned OFF.
- Transmission Error Flag** This flag is turned ON if data transmission has not been completed properly due to an error during data transmission.

### 4-5-4 Reception Bits and Flags

The following bits and flags are used for reception.

- Reception Control Bit** This bit is used to control reception. It is allocated in the I/O refresh area of the PC data area interface. Data reception begins when this bit goes ON.
- Reception Completed Flag** This flag is turned ON when reception is completed. It is allocated in the I/O refresh area of the PC data area interface. This flag goes ON when reception is completed and goes OFF when the Reception Control Bit is turned OFF.
- Reception Error Flag** This flag is turned ON if data reception has not been completed properly due to an error during data reception.

## 4-6 Data Transfer Examples

### 4-6-1 Data Transmission

This section provides examples of data transmission using the C200HS with the MC Unit, the unit number of which is set to 0.

**Example 1** The following data items are transmitted to position data address A0456 to A0460.

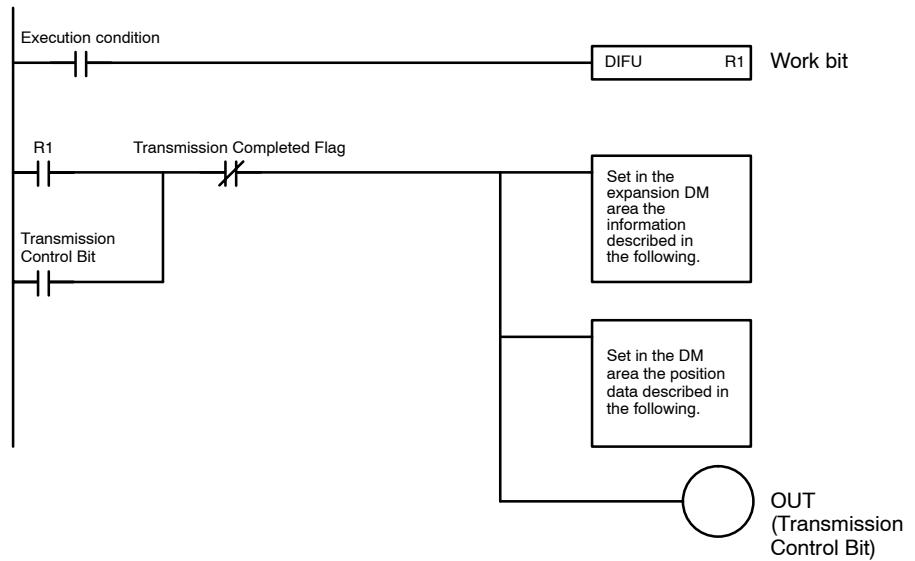
Position data	Position data address
345678.90 .....	A0456
123456.78 .....	A0457
-387654.32 .....	A0458
123.456 .....	A0459
54321.00 .....	A0460

- 1, 2, 3... 1. Set the data described below in the initial setting area with the Programming Console.



2. Turn the PC OFF and ON.
3. Write and execute the following diagram to transfer the position data.

**Ladder Diagram Example**



**Settings**

Initial setting area

D0100	000D	← Set an expansion DM area in the DM area.
D0101	0500	← Set the first word of the expansion DM area to 0500.

Set in the expansion data area the following information used for data transmission.

D0500	0015	← Set a total of 15 words to be transmitted (i.e., 3 words x 5 data items).
D0501	0100	← Set the first word (100) of the position data.
D0502	0456	← Set the first address (A0456) of the MC Unit for the position data.
:	:	
:	:	
:	:	
D0522	:	

**After Transmission**

DM

D0100	0002	} ← Means 345678.90
D0101	7890	
D0102	3456	
D0103	0002	} ← Means 123456.78
D0104	5678	
D0105	1234	
D0106	0002	} ← Means -387654.32
D0107	5432	
D0108	B876	
D0109	0003	} ← Means 123.456
D0110	3456	
D0111	0012	
D0112	0002	} ← Means 54321.00
D0113	2100	
D0114	0543	

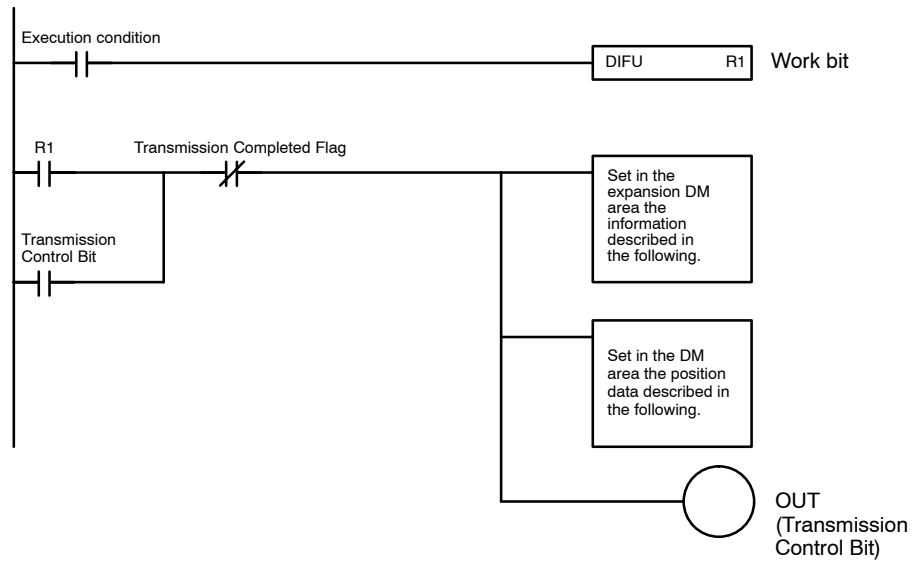
**Example 2**

The acceleration time of the X axis is changed to 500 ms.

- 1, 2, 3...**
1. Set the data described below in the initial setting area with the Programming Console.
  2. Turn the PC OFF and ON.

3. Write and execute the following diagram to change the acceleration time.

**Ladder Diagram Example**



**Settings**

Initial setting area

D0100	000D	← Set an expansion DM area in the DM area.
D0101	0500	← Set the first word of the expansion DM area to 0500.

Set in the expansion data area the following information used for data transmission.

D0500	0002	← Set a total of two words to be transmitted (i.e., 2 words x 1 data item).
D0501	0100	← Set the first word (100) of the acceleration time.
D0502	4406	← Set the address (4406) of the MC Unit for the acceleration time.
:	:	
:	:	
:	:	
D0522	:	

**Note:** A parameter can change a single data item at a time. In other words, a total of two words is transmitted at a time.

**After Transmission**

DM

D0100	0500	← Means 500 ms.
D0101	0000	

**4-6-2 Data Reception**

Refer to the following for examples of data reception using the C200HS with the MC Unit, the unit number of which is set to 0.

**Example 1**

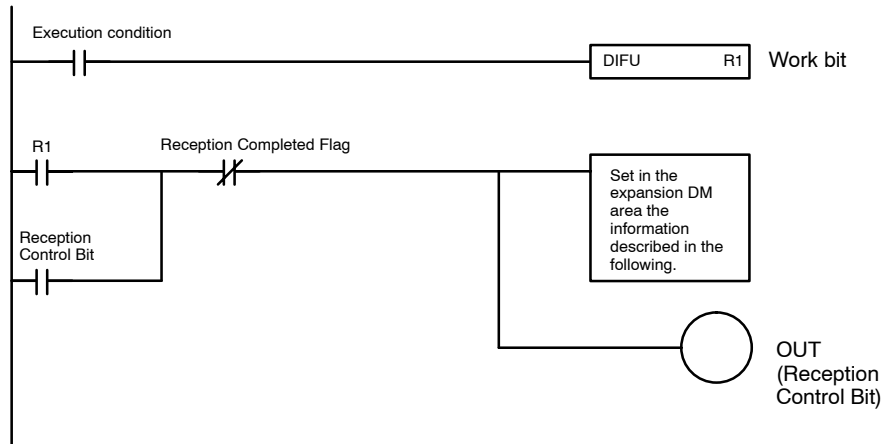
Data items A0456 to A0460 are received at DM 0100 to DM 0114.

Position data	Position data address
A0456 = 45678.90 . . . . .	DM 0100 to DM 0102
A0457 = 123456.78 . . . . .	DM 0103 to DM 0105
A0458 = -387654.32 . . . . .	DM 0106 to DM 0108
A0459 = 123.456 . . . . .	DM 0109 to DM 0111
A0460 = 54321.00 . . . . .	DM 0112 to DM 0114

- 1, 2, 3...**
1. Set the data described below in the initial setting area with the Programming Console.
  2. Turn the PC OFF and ON.

3. Write and execute the following diagram to receive the position data.

**Ladder Diagram Example**



**Settings**

Initial setting area

D0100	000D	← Set an expansion DM area in the DM area.
D0101	0500	← Set the first word of the expansion DM area to 0500.

Set in the expansion data area the information described below used for data reception.

D0500	:	
:	:	
D0503	0015	← Set a total of 15 words to be received (i.e., 3 words x 5 data items).
D0504	0456	← Set the first address (A0456) of the position data.
D0505	0100	← Set the first word (100) of the DM area to store the position data.
:	:	
D0522	:	

**After Reception DM**

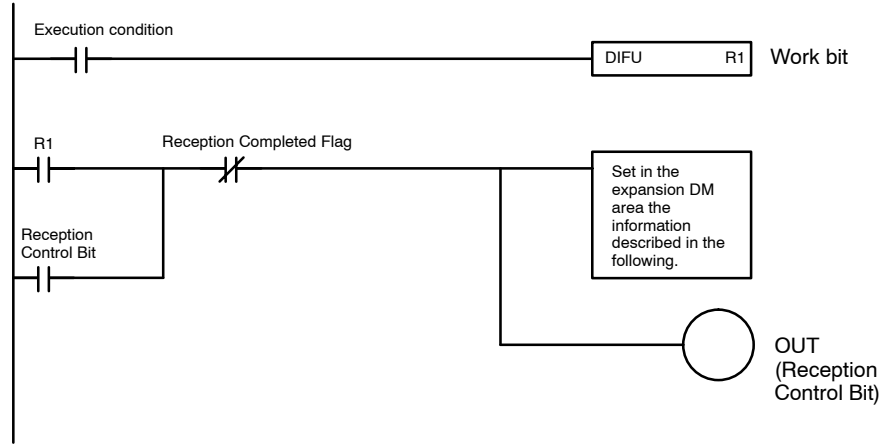
D0100	0002	← Means 345678.90
D0101	7890	
D0102	3456	
D0103	0002	← Means 123456.78
D0104	5678	
D0105	1234	
D0106	0002	← Means -387654.32
D0107	5432	
D0108	B876	
D0109	0003	← Means 123.456
D0110	3456	
D0111	0012	
D0112	0002	← Means 54321.00
D0113	2100	
D0114	0543	

**Example 2**

The acceleration time (500 ms) of the X axis is received at DM 0100 to DM 0101.

- 1, 2, 3... 1. Set the data described below in the initial setting area with the Programming Console.
2. Turn the PC OFF and ON.
3. Write and execute the following diagram to receive the acceleration time.

Ladder Diagram Example



Settings

Initial setting area

D0100 000D ← Set an expansion DM area in the DM area.  
 D0101 0500 ← Set the first word of the expansion DM area to 0500.

Set in the expansion data area the following information used for data reception.

D0500	:
:	:
D0503	0002
D0504	4406
D0505	0100
:	:
D0522	:

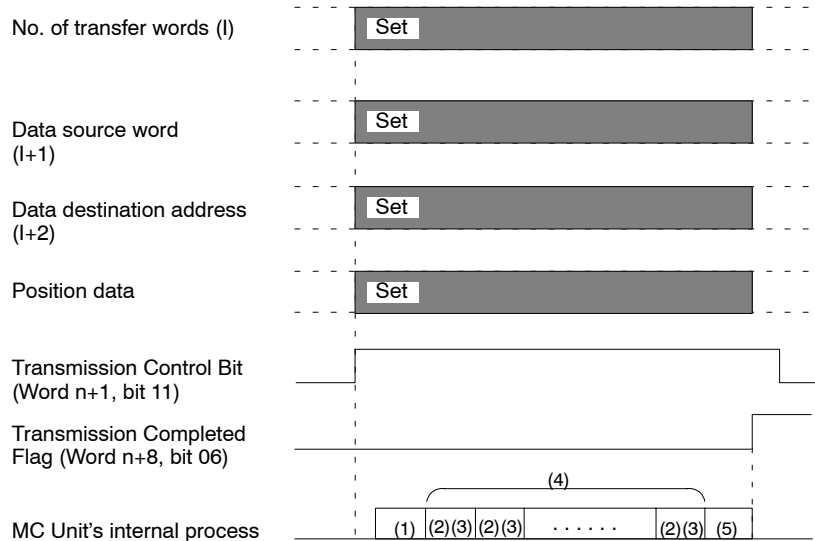
← Set a total of two words to be received.  
 ← Set the first address (4406) of the acceleration time.  
 ← Set the first word (100) of the DM area to store the acceleration time.

DM

D0100 0500 ← Means 500 ms.  
 D0101 0000

## 4-7 I/O Transfer Timing

This section provides information on the transfer timing and transfer time of a large number of position data items.



Set the number of transfer words, data source word, and data destination address in the expansion data area and turn the transmission control bit (word n+1, bit 11) ON for I/O transfer.

The MC Unit performs the following and turns the Transmission Completed Flag (word n+8, bit 6) ON.

- 1, 2, 3...**
1. The MC Unit obtains parameters (the number of transfer words, data source word, and data destination address) to turn the transmission control bit ON. It takes a few cycles to obtain the parameters.
  2. The MC Unit obtains transmission data from the PC. Transmission data of 20 words max. can be obtained at a time. If the number of words of the transmission data exceeds 20, the remaining transmission data is obtained in sequence in blocks of 20 words. It takes a few cycles to obtain the 20-word transmission data.
  3. The MC Unit checks whether the transmission data is correct after receiving the first block of data. It takes a few milliseconds to check this.
  4. The MC Unit repeats steps 2 and 3 described above until the MC Unit finishes checking all the transmission data.
  5. The MC Unit turns the Transmission Completed Flag ON after data transmission. It takes a few scans to turn the Transmission Completed Flag ON.

The calculation of transmission time is complicated. Refer to the following example of transmission time with the C200HS as a reference value.

Cycle time: 4.2 ms

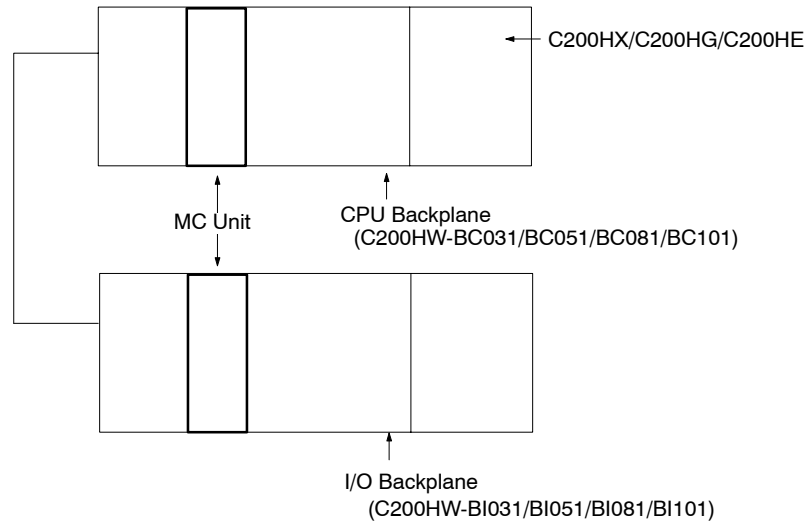
Status of MC Unit: 1 task with 2 axes that are servo-locked.

No. of position data items	Transmission time (s) (Between the transmission control bit turned ON and the next transmission control bit turned ON)
1	0.04
10	0.06
100	0.18
500	0.89
1000	1.6

### 4-8 Details of Intelligent Transfer

Intelligent transfer is available to the MC Unit if the MC Unit is mounted to the C200HX, C200HG, or C200HE. Intelligent transfer makes it possible to transmit or receive data of 128 words max. or a maximum of 42 position data items with one instruction.

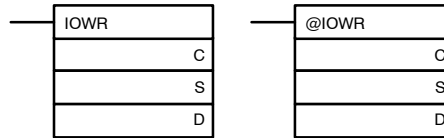
As shown in the following diagram, however, an MC Unit each must be mounted to the CPU Backplane and I/O Backplane for intelligent transfer. The Remote I/O Unit is not available for intelligent transfer.



## 4-9 IOWR Instruction

Use the IOWR instruction to transmit data from the PC to the MC Unit. Refer to the following for the format of the IOWR instruction.

### 4-9-1 Designation



Operand	Description																		
C	<div style="border: 1px solid black; padding: 2px; display: inline-block;">MC Unit address (BCD)</div> Designates the address of the MC Unit that will receive the data.																		
S	<div style="border: 1px solid black; padding: 2px; display: inline-block;">No. of first source word</div> Designates the first word of the PC that has the first block of data. Refer to the following table to designate first data source words. Refer to the <i>C200HX/C200HG/C200HE PCs Operation Manual</i> for these allocations in detail. <table border="1" style="margin-left: 20px; margin-top: 10px;"> <thead> <tr> <th>Area</th> <th>Range</th> </tr> </thead> <tbody> <tr> <td>IR area 1</td> <td>000 to 250</td> </tr> <tr> <td>IR area 2</td> <td>256 to 511</td> </tr> <tr> <td>HR area</td> <td>HR 00 to HR 99</td> </tr> <tr> <td>AR area</td> <td>AR 00 to AR 27</td> </tr> <tr> <td>LR area</td> <td>LR 00 to LR 63</td> </tr> <tr> <td>Timer/Counter</td> <td>T/C 000 to T/C 511</td> </tr> <tr> <td>Data memory</td> <td>DM 0000 to DM 6143</td> </tr> <tr> <td>Indirect DM addressing</td> <td>*DM 0000 to *DM 6655</td> </tr> </tbody> </table>	Area	Range	IR area 1	000 to 250	IR area 2	256 to 511	HR area	HR 00 to HR 99	AR area	AR 00 to AR 27	LR area	LR 00 to LR 63	Timer/Counter	T/C 000 to T/C 511	Data memory	DM 0000 to DM 6143	Indirect DM addressing	*DM 0000 to *DM 6655
Area	Range																		
IR area 1	000 to 250																		
IR area 2	256 to 511																		
HR area	HR 00 to HR 99																		
AR area	AR 00 to AR 27																		
LR area	LR 00 to LR 63																		
Timer/Counter	T/C 000 to T/C 511																		
Data memory	DM 0000 to DM 6143																		
Indirect DM addressing	*DM 0000 to *DM 6655																		
D	<div style="display: flex; align-items: center;"> <div style="border: 1px solid black; padding: 2px; margin-right: 5px;">X</div> <div style="border: 1px solid black; padding: 2px; flex-grow: 1;">No. of data transfer words (BCD)</div> </div> <div style="margin-left: 20px; margin-top: 5px;">                     ↑                      Destination unit no. (0 to 8 or A to E)                 </div> Designates the destination unit number and the number of transfer data words.  The number of data transfer words varies as follows: Three words are required for the transmission of a position data item. Therefore, 12 words are required to transmit four position data items (i.e., 3 words x 4 position data items).  Two words are required for the transmission of a data item other than a position data item.																		

## 4-9-2 Flags

Value	ON	OFF
ER (25503)	<p>The number of data transfer words is 0, 128 or over, or not BCD.</p> <p>The indirect DM is set to 6656 or over or not BCD.</p> <p>The destination unit number is other than 0 to F or in the SYSMAC BUS.</p> <p>The instruction was not completed normally.</p> <p>A Unit that has an error was specified.</p>	None of the conditions on the left exists.
CY (25504)	---	---
> (25505)	---	---
= (25506)	<p>Data transmission was normal.</p> <p>One of the following error has occurred, in which case bit 14 of word n+8 turns ON.</p> <p>The position data is not within the acceptable range or BCD.</p> <p>Data other than position data was transmitted in the wrong format or not within the acceptable range.</p>	<p>The I/O transfer required with a previous cycle has not been completed.</p> <p>The previous intelligent transfer has not been completed.</p> <p>The address of the specified MC Unit is not within the acceptable range or BCD.</p> <p>The data was transmitted to an address that can be read only.</p> <p>The number of data transfer words was one of the following.</p> <p>The number of position data transmission words was not a multiple of three.</p> <p>The number of data transmission words other than position data transmission words was not two.</p> <p>The value of the MC Unit's address plus the number of transfer words divided by three exceeded the value of the address of the position data.</p>
< (25507)	---	---
OF (25404)	---	---
UF (25405)	---	---
N (25402)	---	---

## 4-9-3 Transfer Example

Refer to the following for examples of data transmission when using the C200HG with the MC Unit mounted, the unit number of which is set to 0. In these examples, the operand of the IOWR instruction and the transmission data are normal.

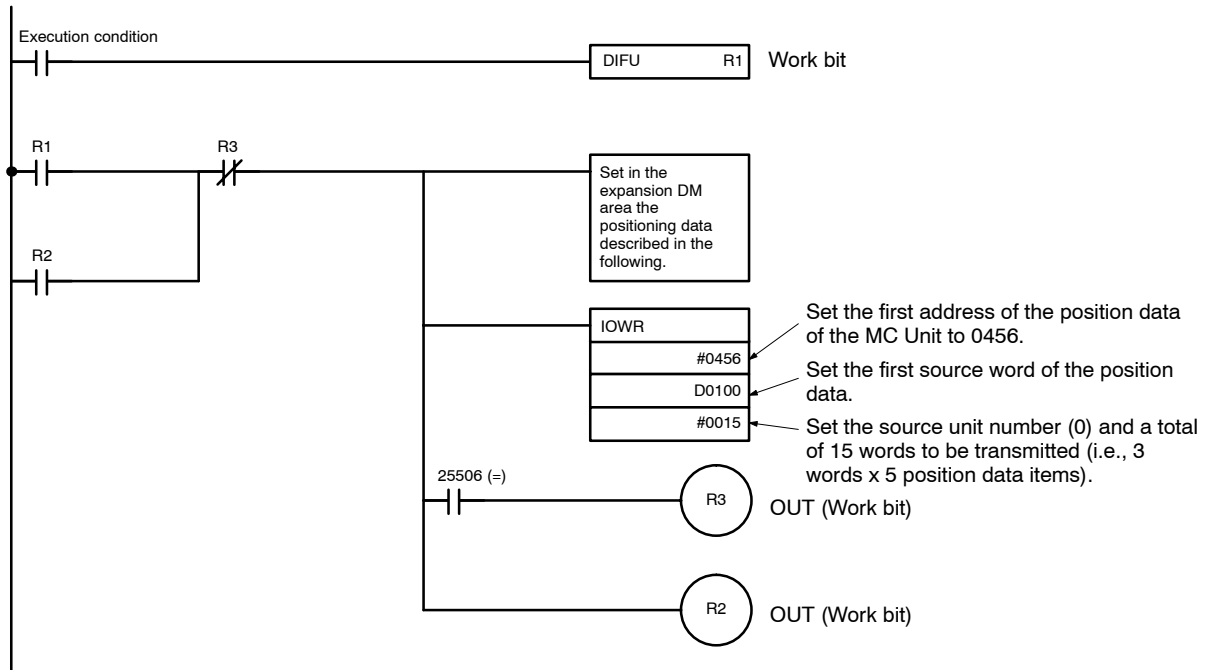
At the time of debugging, write a ladder program to refer to the Error Flag (word n+8 bit 14).



**Example 1**

The following data items are transmitted to position data address A0456 to A0460.

Position data	Position data address
345678.90 .....	A0456
123456.78 .....	A0457
-387654.32 .....	A0458
123.456 .....	A0459
54321.00 .....	A0460

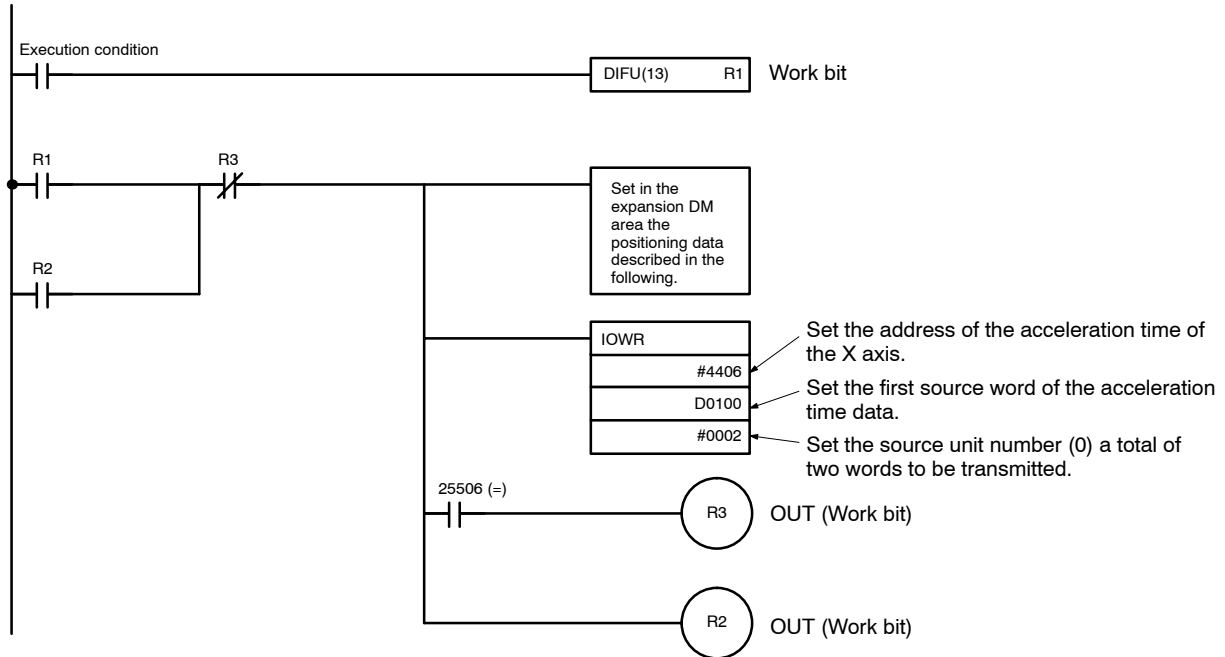


**Settings**  
DM

D0100	0002	← Means 345678.90
D0101	7890	
D0102	3456	
D0103	0002	← Means 123456.78
D0104	5678	
D0105	1234	
D0106	0002	← Means -387654.32
D0107	5432	
D0108	B876	
D0109	0003	← Means 123.456
D0110	3456	
D0111	0012	
D0112	0002	← Means 54321.00
D0113	2100	
D0114	0543	

**Example 2**

The acceleration time of the X axis is changed to 500 ms.

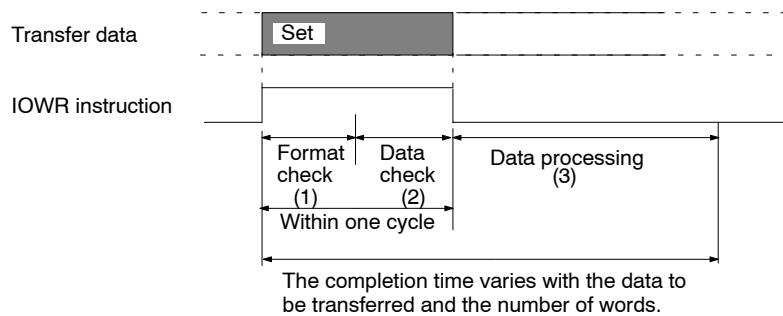


**Settings**  
DM

D0100	0500	← Means 500 ms.
D0101	0000	

**4-9-4 Execution Timing of IOWR Instruction**

Refer to the following for the execution timing and processing method of the IOWR instruction.



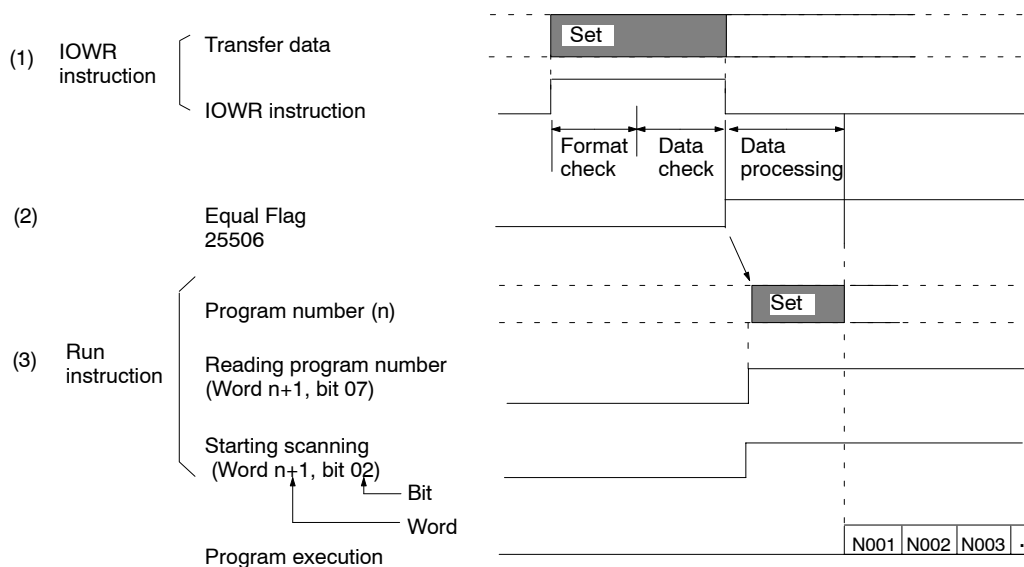
- 1, 2, 3...**
1. The PC performs a format check to confirm whether the designation of the operand of the IOWR instruction is correct. ER Flag 25503 will be turned OFF if the designation is correct. ER Flag 25503 will be turned ON and the IOWR instruction will not be executed if the designation is wrong.
  2. The MC Unit performs a data check to confirm if the MC Unit can process the data designated with the operand. Flag 25506 will be turned ON if the MC Unit can process the data. Flag 25506 will be turned OFF and the IOWR instruction will not be executed if the MC Unit cannot process the data.
  3. The range of the data transferred from the PC is checked. The data will be transferred to the specified address if the data is within the acceptable range. The data will not be transferred, bit 14 of word n+8 will be turned ON, and the corresponding error number will be output to word n+9 or word l+12 of the expansion data area if the data is not within the acceptable range.

**Note** The programmed run instruction will be executed after the data is processed if the programmed run instruction is given while the data is being processed in the previous step 3. Refer to the following example.

**Example**

The program run instruction is executed after position data is transferred.

- 1, 2, 3...**
1. Position data is transmitted with the IOWR instruction.
  2. Equal Flag 25506 will be turned ON when the transmission of the position data is completed.
  3. The run instruction is given. The instruction will not be executed until the MC Unit completes processing the position data.
  4. The MC Units will execute the run instruction after the MC Units completes processing the position data.



**4-9-5 Transfer Time**

The calculation of transmission time with the IOWR instruction is complicated and varies with the PC model, ladder program, and the status of the MC Unit. Refer to the following example of transmission time with the C200HX as a reference value.

Cycle time: 2.5 ms

Status of MC Unit: 1 task with 2 axes that are servo-locked.

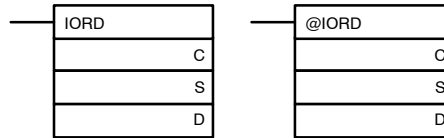
No. of position data items	Transmission time (ms)
1	1.7
6	3.2
20	8.5
42	16

**Note** The IOWR instruction transfers all data while the IOWR instruction is being executed. In the above example, the ladder cycle time will increase according to the above table. Adjust the time to monitor the cycle time if necessary.

## 4-10 IORD Instruction

Use the IORD instruction for the PC to receive data from the MC Unit. Refer to the following for the format of the IORD instruction.

### 4-10-1 Designation



Operand	Description																		
C	<div style="border: 1px solid black; display: inline-block; padding: 2px;">MC Unit address (BCD)</div> Designates the address of the MC Unit that will receive the data.																		
S	<table border="1" style="display: inline-table; margin-right: 10px;"> <tr> <td style="text-align: center; width: 20px;">X</td> <td style="padding: 2px;">No. of data transfer words (BCD)</td> </tr> </table> <div style="margin-left: 20px;">                         ↑                          Destination unit no. (0 to 8 or A to E)                     </div> Designates the destination unit number and a total of data transfer words.  The number of data transfer words varies as follows:  Three words are required for the transmission of a position data item. Therefore, 12 words are required to transmit four position data items (i.e., 3 words x 4 position data items).  Two words are required for the transmission of a data item other than a position data item.	X	No. of data transfer words (BCD)																
X	No. of data transfer words (BCD)																		
D	<div style="border: 1px solid black; display: inline-block; padding: 2px;">No. of first source word</div> Refer to the following table to designate first data source words of the PC to store the reception data.  Refer to the <i>C200HX/C200HG/C200HE PCs Operation Manual</i> for these words in detail. <table border="1" style="margin-top: 10px; width: 100%;"> <thead> <tr> <th style="text-align: center;">Area</th> <th style="text-align: center;">Range</th> </tr> </thead> <tbody> <tr> <td>Bit area 1</td> <td>000 to 250</td> </tr> <tr> <td>Bit area 2</td> <td>256 to 511</td> </tr> <tr> <td>HR area</td> <td>HR 00 to HR 99</td> </tr> <tr> <td>AR area</td> <td>AR 00 to AR 27</td> </tr> <tr> <td>LR area</td> <td>LR 00 to LR 63</td> </tr> <tr> <td>Timer/Counter</td> <td>T/C 000 to T/C 511</td> </tr> <tr> <td>Data memory</td> <td>DM 0000 to DM 6143</td> </tr> <tr> <td>Indirect DM addressing</td> <td>*DM 0000 to *DM 6655</td> </tr> </tbody> </table>	Area	Range	Bit area 1	000 to 250	Bit area 2	256 to 511	HR area	HR 00 to HR 99	AR area	AR 00 to AR 27	LR area	LR 00 to LR 63	Timer/Counter	T/C 000 to T/C 511	Data memory	DM 0000 to DM 6143	Indirect DM addressing	*DM 0000 to *DM 6655
Area	Range																		
Bit area 1	000 to 250																		
Bit area 2	256 to 511																		
HR area	HR 00 to HR 99																		
AR area	AR 00 to AR 27																		
LR area	LR 00 to LR 63																		
Timer/Counter	T/C 000 to T/C 511																		
Data memory	DM 0000 to DM 6143																		
Indirect DM addressing	*DM 0000 to *DM 6655																		

### 4-10-2 Flags

Value	ON	OFF
ER (25503)	<p>The number of data transfer words is 0, 128 or over, or not BCD.</p> <p>The indirect DM is set to 6656 or over or not BCD.</p> <p>The source unit number is other than 0 to F or in the SYSMAC BUS.</p> <p>The instruction was not completed normally.</p> <p>A Unit that has an error was specified.</p>	None of the conditions on the left exists.
CY (25504)	---	---
> (25505)	---	---
= (25506)	<p>Data reception was normal.</p> <p>One of the following error has occurred, in which case bit 14 of word n+8 turns ON.</p> <p>The position data is valid.</p>	<p>The I/O transfer required with a previous cycle has not been completed.</p> <p>The previous intelligent transfer has not been completed.</p> <p>The address of the specified MC Unit is not within the acceptable range or BCD.</p> <p>The number of data transfer words was one of the following.</p> <p>The number of position data transmission words was not a multiple of three.</p> <p>The number of data transmission words other than position data transmission words was not two.</p> <p>The value of the MC Unit's address plus the number of transfer words divided by three exceeded the value of the address of the position data.</p>
< (25507)	---	---
OF (25404)	---	---
UF (25405)	---	---
N (25402)	---	---

### 4-10-3 Reception Example

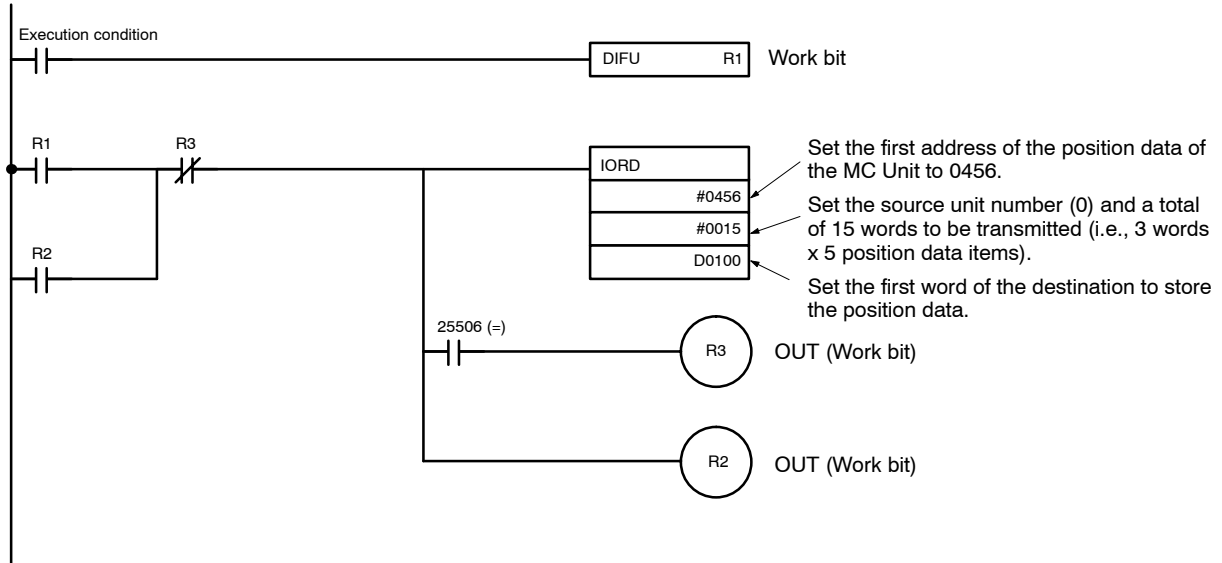
Refer to the following for examples of data reception using the C200HG with the MC Unit mounted, the unit number of which is set to 0. In these examples, the operand of the IORD instruction and the transmission data are normal.

At the time of debugging, write a ladder program to refer to the Error Flag (word n+8, bit 14).

**Example 1**

Data items A0456 to A0460 are received at DM 0100 to DM 0114.

Position data	Position data address
A0456 = 45678.90 . . . . .	DM 0100 to DM 0102
A0457 = 123456.78 . . . . .	DM 0103 to DM 0105
A0458 = -387654.32 . . . . .	DM 0106 to DM 0108
A0459 = 123.456 . . . . .	DM 0109 to DM 0111
A0460 = 54321.00 . . . . .	DM 0112 to DM 0114



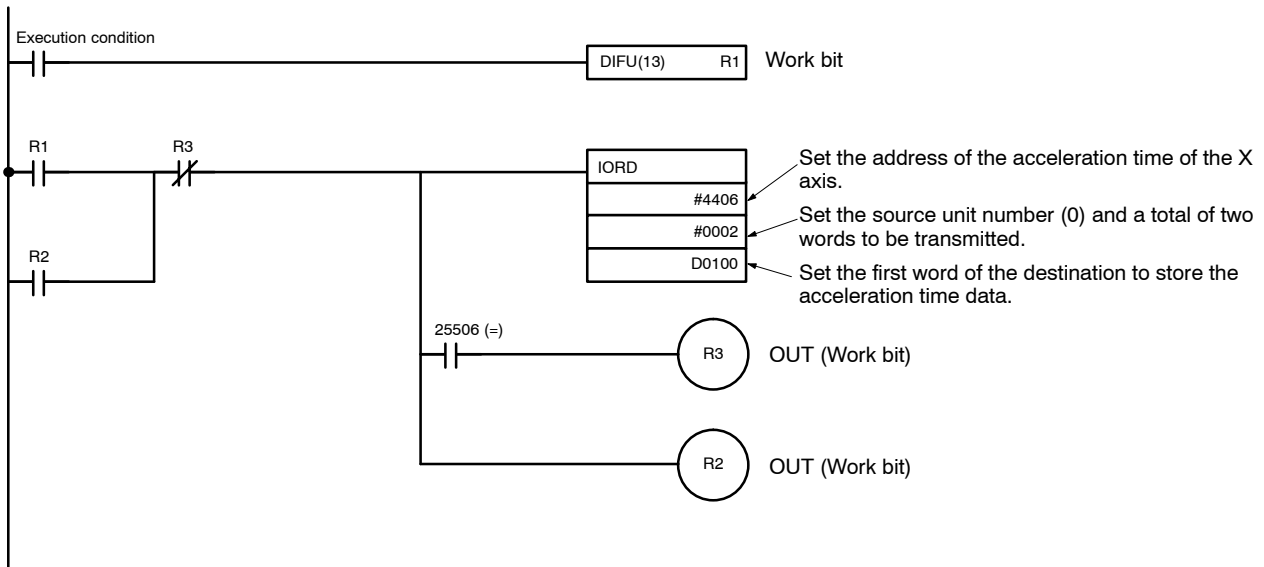
**Settings**

DM

D0100	0002	← Means 345678.90.
D0101	7890	
D0102	3456	
D0103	0002	← Means 123456.78.
D0104	5678	
D0105	1234	
D0106	0002	← Means -387654.32.
D0107	5432	
D0108	B876	
D0109	0003	← Means 123.456.
D0110	3456	
D0111	0012	
D0112	0002	← Means 54321.00.
D0113	2100	
D0114	0543	

**Example 2**

The acceleration time (500 ms) of the X axis is received at DM 0100 to DM 0101.

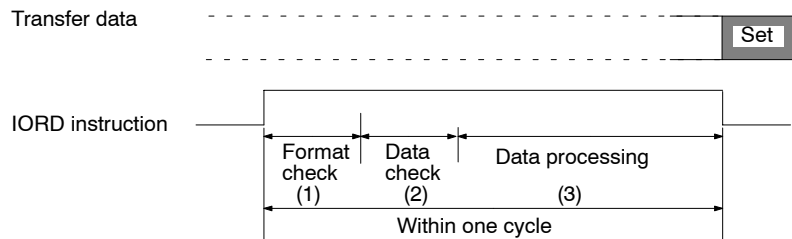


**After Reception**  
DM

D0100	0500	← Means 500 ms.
D0101	0000	

**4-10-4 Execution Timing of IORD Instruction**

Refer to the following for the execution timing and processing method of the IORD instruction.



- 1, 2, 3...**
1. The PC performs a format check to confirm whether the designation of the operand of the IORD instruction is correct. ER Flag 25503 will be turned OFF if the designation is correct. ER Flag 25503 will be turned ON and the IOWR instruction will not be executed if the designation is wrong.
  2. The MC Unit performs a data check to confirm if the MC Unit can process the data designated with the operand. The Equal Flag 25506 will be turned ON if the MC Unit can process the data. The Equal Flag 25506 will be turned OFF and the IOWR instruction will not be executed if the MC Unit cannot process the data.
  3. The MC Unit transfers the data to the PC. The Equal Flag 25506 will be turned ON when the data transmission is completed.

### 4-10-5 Transfer Time

The calculation of transmission time with the IORD instruction is complicated and varies with the PC model, ladder program, and the status of the MC Unit. Refer to the following example of transmission time with the C200HX as a reference value.

Cycle time: 2.5 ms

Status of MC Unit: 1 task with 2 axes that are servo-locked.

No. of position data items	Transmission time (ms)
1	1.2
6	1.9
20	4.2
42	7.3

**Note** The IORD instruction transfers all data while the IORD instruction is being executed. In the above example, the ladder scan time will increase according to the above table.

Adjust the time to monitor the cycle time if necessary.



# 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	70
5-2	G Function Formats	71
5-3	List of G Symbols	72
5-3-1	Specifying Position Data Addresses (A0000 to A1999)	73
5-3-2	Specifying Registers (E00 to E31)	73
5-4	Program Number and Axis Declaration	74
5-5	G Functions	75
5-5-1	G00: POSITIONING	75
5-5-2	G01: LINEAR INTERPOLATION	77
5-5-3	G02 and G03: CIRCULAR INTERPOLATION	78
5-5-4	G04: DWELL TIMER	82
5-5-5	G10: PASS MODE	83
5-5-6	G11: STOP MODE	84
5-5-7	G17: CIRCULAR PLANE SPECIFICATION FUNCTIONS	85
5-5-8	G26: REFERENCE ORIGIN RETURN	85
5-5-9	G27: WORKPIECE ORIGIN RETURN	86
5-5-10	G28: ORIGIN SEARCH	87
5-5-11	G29: ORIGIN UNDEFINED	88
5-5-12	G30: SPEED CONTROL AXIS FEEDING	88
5-5-13	G31: INTERRUPT INCHING	89
5-5-14	G50: SELECT REFERENCE COORDINATE SYSTEM	93
5-5-15	G51: SELECT WORKPIECE COORDINATE SYSTEM	93
5-5-16	G53: CHANGE WORKPIECE ORIGIN OFFSET	94
5-5-17	G54: CHANGE REFERENCE COORDINATE SYSTEM PV	95
5-5-18	G60: ARITHMETIC OPERATIONS	96
5-5-19	G63: SUBSTITUTION	97
5-5-20	G69: CHANGE PARAMETER	98
5-5-21	G70: UNCONDITIONAL JUMP	99
5-5-22	G71: CONDITIONAL JUMP	100
5-5-23	G72: SUBPROGRAM JUMP	101
5-5-24	G73: SUBPROGRAM END	101
5-5-25	G74: OPTIONAL END	102
5-5-26	G75: OPTIONAL SKIP	103
5-5-27	G76: OPTIONAL PROGRAM PAUSE	104
5-5-28	G79: PROGRAM END	105
5-5-29	G90: ABSOLUTE SPECIFICATION	105
5-5-30	G91: INCREMENTAL SPECIFICATION	106
5-6	M-code Outputs	107
5-6-1	Introduction	107
5-6-2	M-code Data	108
5-6-3	M-code Examples	108
5-6-4	M Code Resets	109
5-6-5	Effect of Mode Changes on M Code Resets	109
5-6-6	Clearing M000 to M499 with Forced Block End or G74	109
5-6-7	Clearing M500 to M999 with Forced Block End or G74	111
5-7	Mode Transitions Caused by G Functions	112

## 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 2 axes simultaneously with PTP control at the maximum speed.	75
G01	LINEAR INTERPOLATION	Performs linear interpolation on up to 2 axes simultaneously at the specified interpolation speed.	77
G02	CIRCULAR INTERPOLATION (CLOCKWISE)	Performs 2-axis circular interpolation in the clockwise direction at the specified interpolation speed.	78
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.	82
G10	PASS MODE	Performs operations one-by-one in sequence without deceleration to stop.	83
G11	STOP MODE	Performs the next operation after completing positioning.	84
G17	CIRCULAR PLANE SPECIFICATION (X-Y)	Sets the X-Y plane as the plane for circular interpolation.	85
G26	REFERENCE ORIGIN RETURN	Moves to the reference origin.	85
G27	WORKPIECE ORIGIN RETURN	Moves to the workpiece origin.	86
G28	ORIGIN SEARCH	Performs an origin search in the specified axis.	87
G29	ORIGIN UNDEFINED	Makes the origin undefined.	88
G30	SPEED CONTROL	Performs the speed control of a maximum of two axis.	88
G31	INTERRUPT INCHING	Performs the inching of a single axis for a certain distance after an interrupt input.	89
G50	SELECT REFERENCE COORDINATE SYSTEM	Specifies the reference coordinate system.	93
G51	SELECT WORKPIECE COORDINATE SYSTEM	Specifies the workpiece coordinate system.	93
G53	CHANGE WORKPIECE ORIGIN OFFSET	Changes the origin of the workpiece coordinate system.	94
G54	CHANGE REFERENCE COORDINATE SYSTEM PV	Changes the present value in the reference coordinate system.	95
G60	ARITHMETIC OPERATIONS	Performs arithmetic operations on numerical values, position data, and registers.	96
G63	SUBSTITUTION	Substitutes numerical values, position data, or registers into other position data or registers.	97
G69	CHANGE PARAMETER	Changes the specified parameter.	98
G70	UNCONDITIONAL JUMP	Unconditionally jumps to the specified block.	99
G71	CONDITIONAL JUMP	Jumps to the specified block when the condition is met.	100
G72	SUBPROGRAM JUMP	Calls the specified subprogram.	101
G73	SUBPROGRAM END	Ends the subprogram.	101
G74	OPTIONAL END	Ends the block currently being executed when the specified optional input is ON.	102
G75	OPTIONAL SKIP	Skips the block after this function when the specified optional input is ON.	103
G76	OPTIONAL PROGRAM PAUSE	Pauses the program when the specified optional input is ON.	104
G79	PROGRAM END	Ends the main program.	105

Code	Name	Function	Page
G90	ABSOLUTE SPECIFICATION	Specifies the use of absolute coordinates in axis operations.	105
G91	INCREMENTAL SPECIFICATION	Specifies the use of relative coordinates in axis operations.	106

## 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 J 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 J 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	---
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>]
ORIGIN UNDEFINED	G29	<Axis name ...>
SPEED CONTROL	G30	<Axis movement command>
INTERRUPT INCHING	G31	<Axis movement command> _[F <Speed command 1>] _ [F <Speed command 2>] _ [<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 ... >
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>

Name	Code	Operands
SUBPROGRAM END	G73	---
OPTIONAL END	G74	<input type="checkbox"/> <Optional number>
OPTIONAL SKIP	G75	<input type="checkbox"/> <Optional number>
OPTIONAL PROGRAM STOP	G76	<input type="checkbox"/> <Optional number>
PROGRAM END	G79	---
ABSOLUTE SPECIFICATION	G90	---
INCREMENTAL SPECIFICATION	G91	---

**Note** 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.

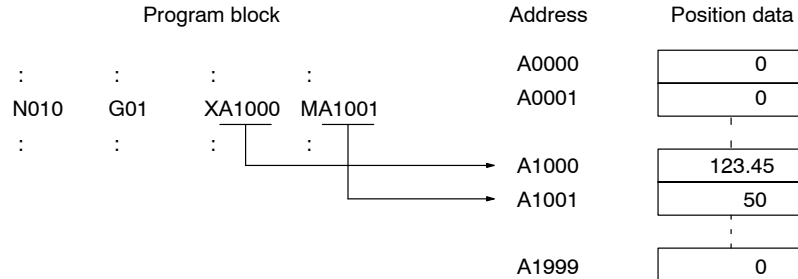
## 5-3 List of G Symbols

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
I	Arc center coordinate for the X-axis.
J	Arc center coordinate for the Y-axis.
L	Number of loops
M	M code
N	Block number
P	Program number
R	Circle radius
X	X-axis
Y	Y-axis
/	Punctuation mark
()	Indirect specification
#	Optional number, parameter type
*	Comment

### 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 (50) 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:	50

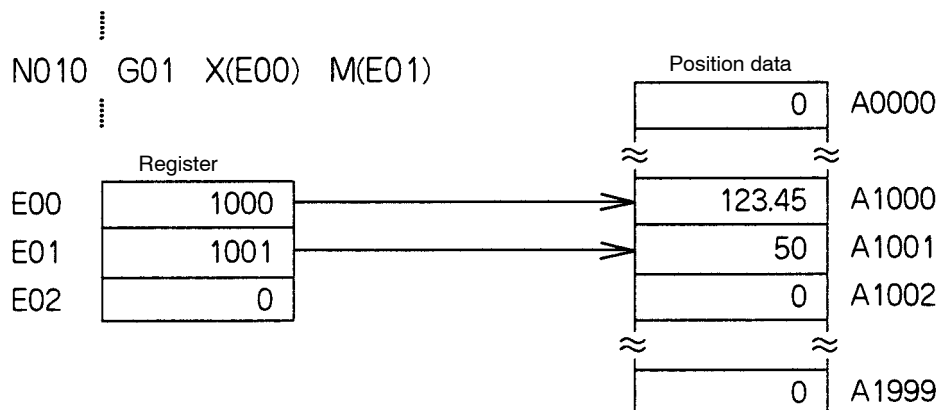
Registers and position data aren't assigned together to each task; they can be used independently for different tasks.

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

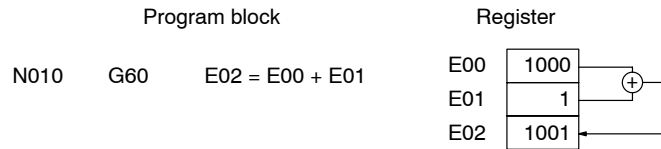
If the register name is in parentheses, i.e. (E00), the content of register will be treated as a position data address.

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 (50) 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.

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

**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 XY           Program number and axis declaration
      :               :
      :               :
N100 G79              PROGRAM END function
    
```

Use the following format for subprograms.

```

N000 P510 XY          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. □ The space is required.
  5. \_ The space can be left out.

### 5-5-1 G00: POSITIONING

Positions up to 2 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 and Y
	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 06

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

#### Description

Positions up to 2 axes simultaneously with PTP control according to the maximum feed rate, acceleration/deceleration time, and acceleration/deceleration curve settings in the system parameters. The override is also valid with this function.

The override is referred to only once at the start of G00. The feed rate cannot be changed during operation.

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.

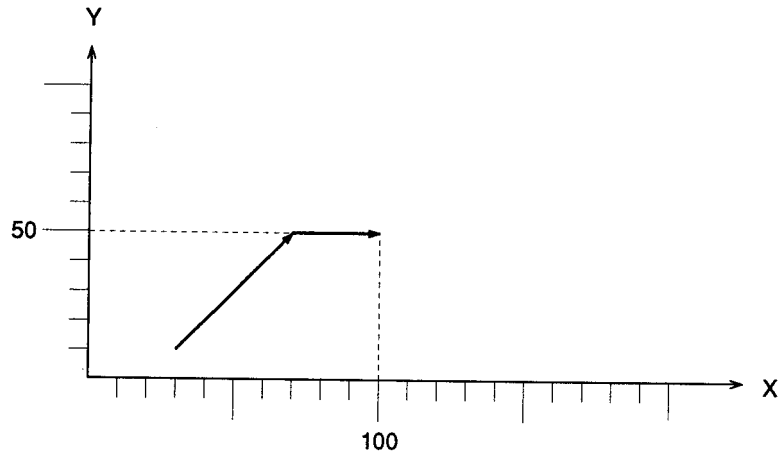
- 0 through 4: Inputs from the PC's data area interface
- 5 and 6: Inputs from general inputs 1 and 2

Example Programs

The following program is for absolute operation.

```

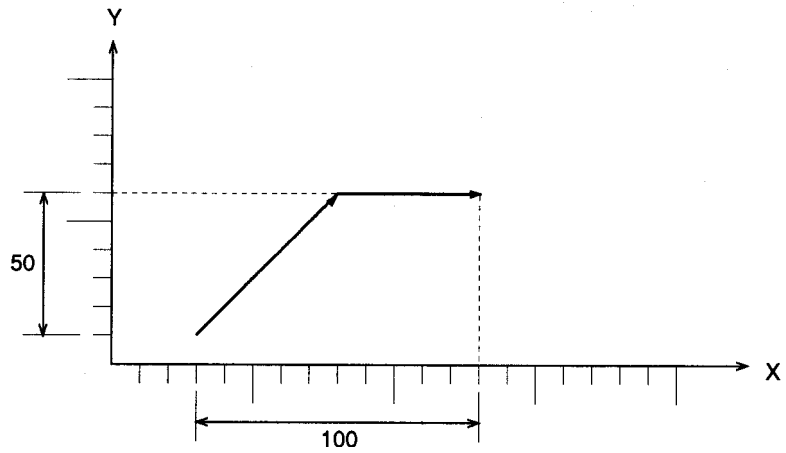
:      :
N010  G90          ABSOLUTE SPECIFICATION
N011  G00  X100  Y50  M001
:      :
    
```



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 2 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 and Y
	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 06

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

### Description

Performs linear interpolation on up to 2 axes simultaneously at the specified interpolation speed.

The interpolation speed is calculated with the equations shown below. Two axes are shown, but the same calculations apply when a single axis is being used.

$$F_x = F \times dL_x / L$$

$$F_y = F \times dL_y / L$$

$F_x$  and  $F_y$ : Interpolation speeds for axes x and y

$dL_x$  and  $dL_y$ : Movement distance for axes x and y

L: Total distance between start and end of linear interpolation

$$\left( L = \sqrt{dL_x^2 + dL_y^2} \right)$$

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 the 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.

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

The override is referred to only once at the start of G01. The feed rate cannot be changed during operation.

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 4: Inputs from the PC's data area interface

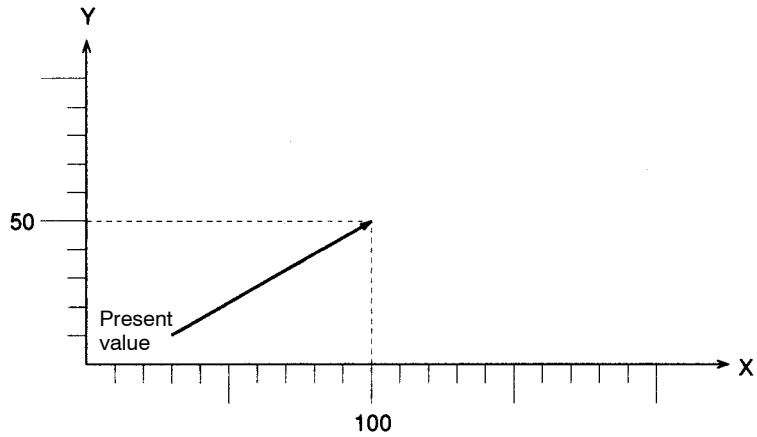
5 and 6: Inputs from general inputs 1 and 2

Example Programs

The following program is for absolute operation.

```

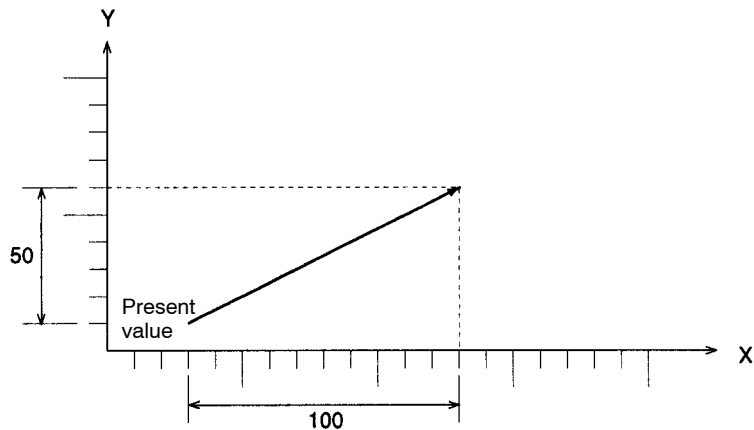
:      :
N010  G90          ABSOLUTE SPECIFICATION
N011  G01  X100  Y50  F300
:      :
    
```



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.

Format

```

G02 (G03) _<Axis movement command ... > _<I to J 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 and Y
	Coordinate data	–39,999,999 to +39,999,999 (E00 to E31) A0000 to A1999
Center coordinate	Axis	I or J
	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(0) to 999 (E00 to E31) A0000 to A1999
Optional number		00 to 06

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

## Description

Performs two-axis circular interpolation at the specified interpolation speed.

The current position is the default coordinate data in the specified circular plane.

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.

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.

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 G02 and G03 functions use the interpolation acceleration time, interpolation deceleration time, and override value of the X axis.

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

The override is referred to only once at the start of G02 and G03. The feed rate cannot be changed during operation.

**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.

When the product of the specified interpolation speed and override value exceeds the maximum interpolation feed rate for the axis being moved by circular interpolation (set in the system parameters), circular interpolation will be performed at the lowest maximum feed rate 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 rate: 4000 [mm/s]
Y-axis max. interpolation feed rate: 2000 [mm/s]
```

$$\begin{aligned} \text{Interpolation speed after override calculation} &= 3000 \times 100.0/100.0 \\ &= 3000 \text{ [mm/s]} \end{aligned}$$

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

**Note** 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.

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.

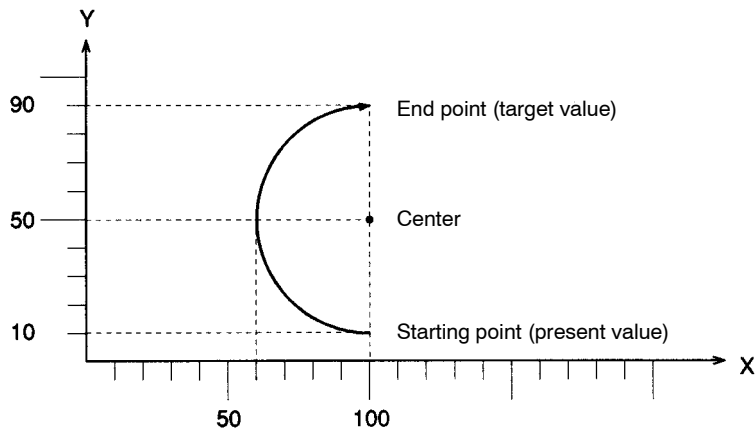
- 0 through 4: Inputs from the PC's data area interface
- 5 and 6: Inputs from general inputs 1 and 2

**Example Programs**

The following program shows circular interpolation with center coordinate specification.

```

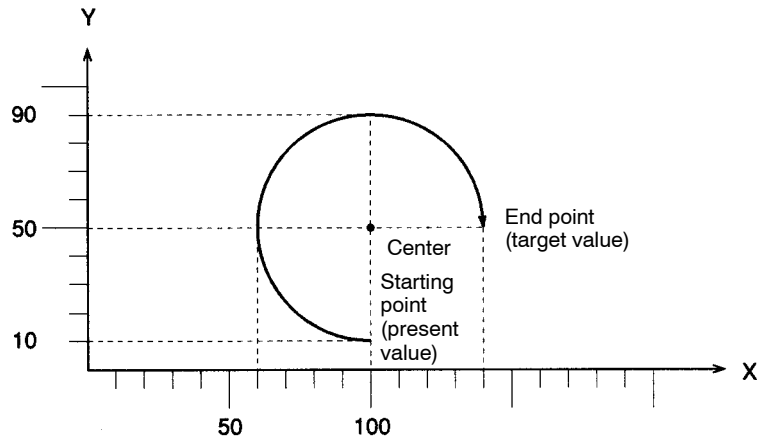
:      :
N010 G90 ABSOLUTE SPECIFICATION
N011 G17 Makes X-Y the circular plane.
:      : (This need not be described.)
N012 G02 X100 Y90 I0 J40 F300
:      :
```



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

```

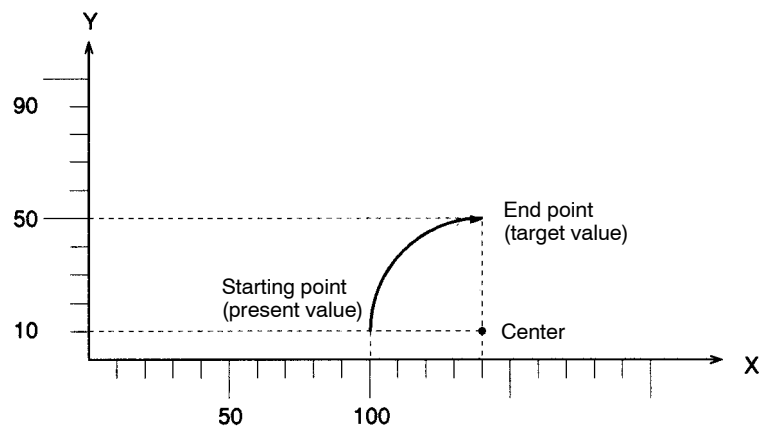
:      :
N010  G90          ABSOLUTE SPECIFICATION
N011  G17          Makes X-Y the circular plane.
                        (This need not be described.)
N012  G02  X140  Y50  R-40  F300
:      :
    
```



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

```

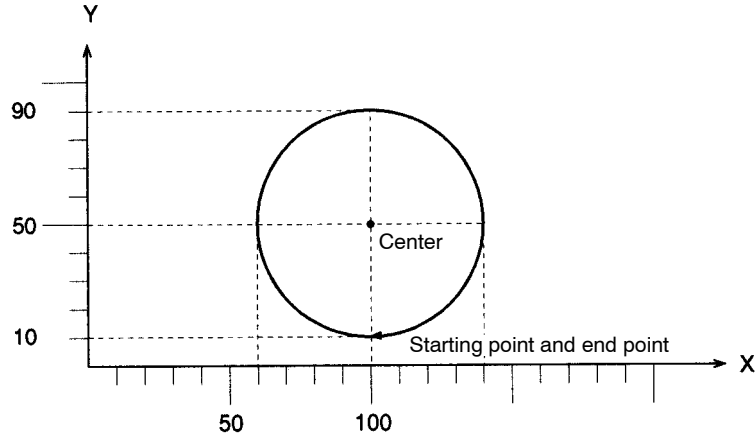
:      :
N010  G91          INCREMENTAL SPECIFICATION
N011  G17          Makes X-Y the circular plane.
                        (This need not be described.)
N012  G02  X40  Y40  R40  F300
:      :
    
```



The following program shows circular interpolation of a complete circle.

```

:      :
N010  G90          ABSOLUTE SPECIFICATION
N011  G17          Makes X-Y the circular plane.
                        (This need not be described.)
N012  G02  X100  Y10  I0  J40  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. 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.



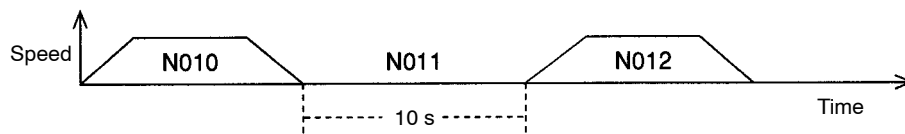
**Caution**

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. The subsequent functions will be executed until the next G01, G02, or G03 function in the program.


In the following program, blocks N003 to N006 will be pre-executed while the axis is moving according to the command in block N002. Although the 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
      :

```

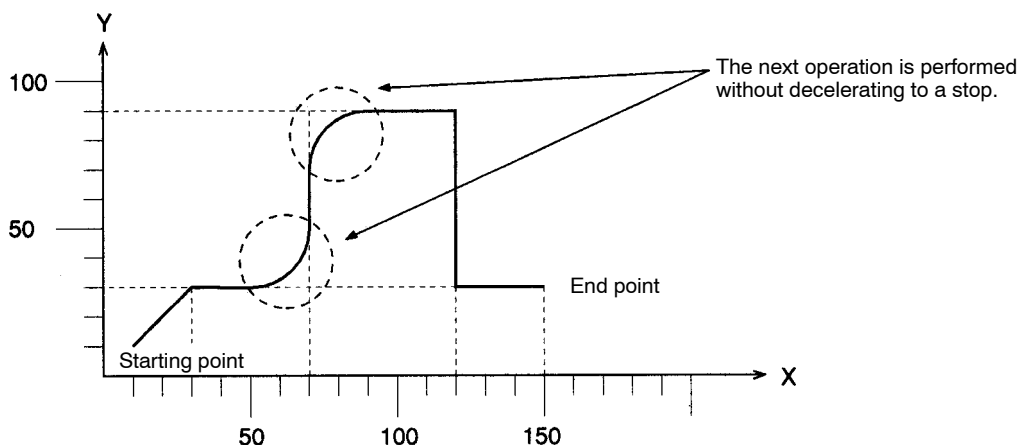
-  **Caution** Pre-execution will be stopped and the axis operation will be paused with the following functions:
- G00 (PTP operation)
  - A G01, G02, or G03 function with an M code from M000 to M499
  - G04 (DWEELL TIMER)
  - G29 (ORIGIN UNDEFINED)
  - G30 (SPEED CONTROL)
  - G31 (INTERRUPT INCHING)
  - 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.

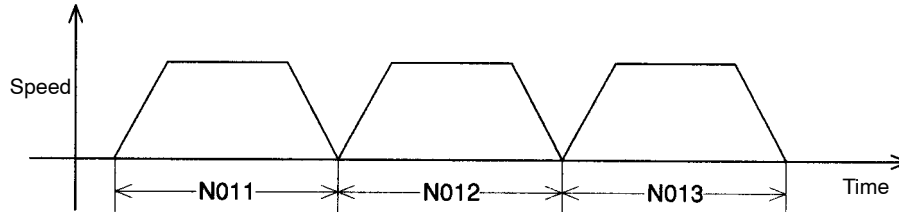


**Example Program**

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: CIRCULAR PLANE SPECIFICATION FUNCTIONS**

Specifies the plane in which circular interpolation is performed.

**Format**

G17

**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.

This function need not be described with the MC Unit.

Circular planes are treated as X-Y plane.

This function makes it possible to transplant programs used with the CV500-MC421 or CV500-MC221.

**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
    
```

**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 and Y
M code*	000 to 999 (E00 to E31) A0000 to A1999

**Note** Refer to 5-6 *M-code Outputs* for details on M codes.  
Refer to *Section 7 Fixing the Origin* for ORIGIN SEARCH in detail.

**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.

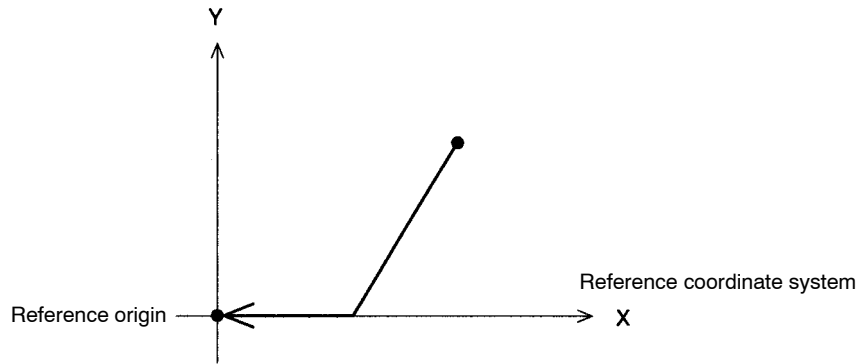
The settings won't be switched to the ones above if function G26 is terminated while in progress by the OPTIONAL END function (G74), FORCED BLOCK END, or other function. The function is also considered to be in progress while waiting for the M code reset.

The override is referred to only once at the start of G26. The feed rate cannot be changed during operation.

**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 and Y
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.

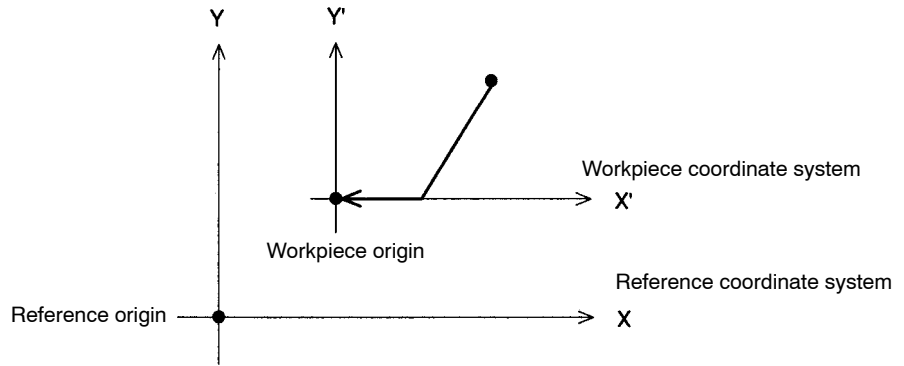
The settings won't be switched to the ones above if function G27 is terminated while in progress by the OPTIONAL END function (G74), FORCED BLOCK END, or other function. The function is also considered to be in progress while waiting for the M code reset.

The override is referred to only once at the start of G27. The feed rate cannot be changed during operation.

**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 and Y
M code <sup>1</sup>	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 will be set to the machine origin by PTP control in the same manner as for the reference origin return.

**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  XY   M500
:      :
    
```

### 5-5-11 G29: ORIGIN UNDEFINED

Makes the origins of axes undefined.

**Format**

G29\_ <Axis name ... >

**Parameters**

Axis: X and Y

**Description**

Makes the origins of specified axes undefined.

This function is usually used for the following applications.

To define the origins of axes when the system is turned on and make the origins undefined for continuous axis feeding.

**Example Program**

```

N000 P000 XY
N001 G00 X0 Y0 (Moves the axis to the origin of the system)
N002 G29 XY (Makes the origin undefined)
N003 G30 X+ Y+ (Makes the axes feed continuously)
      :      :

```

### 5-5-12 G30: SPEED CONTROL AXIS FEEDING

Performs the feed rate control of a maximum of two axes.

**Format**

G30\_ <axis movement command ... >

**Parameters**

The following table shows the possible settings for the parameters.

Axis movement command	
Axis	Coordinate data
X	+
Y	-

**Description**

The axis is fed in the positive direction if the coordinate data is set to positive.

The axis is fed in the negative direction if the coordinate data is set to negative.

The feed rate of each axis can be set individually.

X-axis feed rate = X-axis maximum feed rate x X-axis override value/100

Y-axis feed rate = Y-axis maximum feed rate x Y-axis override value/100

Fast feed rate: Set with system parameters.

Override value: Set with PC or Teaching Box within a range of 0.1% to 100.0%.

Example: The following X-axis feed rate will be obtained if the X-axis maximum feed rate is 10,000 mm/s and the X-axis override value is 50.0%.

X-axis feed rate = 10,000 x 50.0 /100 = 5,000 mm/s.

The acceleration time or deceleration time varies with the override value. Refer to the following.

X-axis acceleration time = X-axis acceleration time\* x override value/100.

X-axis deceleration time = X-axis deceleration time\* x override value/100

Y-axis acceleration time = Y-axis acceleration time\* x override value/100

Y-axis deceleration time = Y-axis deceleration time\* x override value /100

\*These values are set with the system parameters.

The feed rate can be changed without any time lag if the override value is changed and the feed rate operation depends on whether or not the origin is defined.

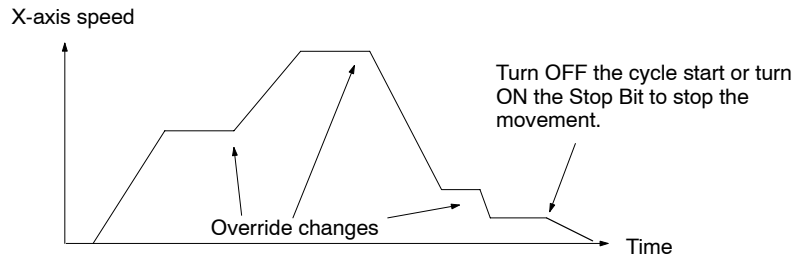
With defined origin: The stroke limit is valid and the current position is always refreshed.

With undefined origin: The stroke limit is invalid and the current position will be set to 0 when the operation starts.

**Example Program**

```

:      :
N010  G30  X+
:      :
    
```



**Note** The previous G01, G02, and G03 functions will operate in stop mode instead of pass mode if this function is executed.

**5-5-13 G31: INTERRUPT INCHING**

Performs the inching of the specified single axis for a certain distance for positioning after a general input turns ON.

**Format**

```

G31_ <axis movement command>_ <F<speed command 1>>_ [F<speed command 2>]_ [<M code>]
    
```

**Parameters**

The following table shows the possible settings for the parameters.

Parameter		Possible settings
Axis movement command	Axis	X and Y
	Coordinate data	-39,999,999 to +39,999,999 (E00 to E31) A0000 to A1999
Speed command 1 Speed command 2		-39,999,999 to +39,999,999 (E00 to E31) A0000 to A1999
M code		000 to 999 (E00 to E31) A0000 to A1999

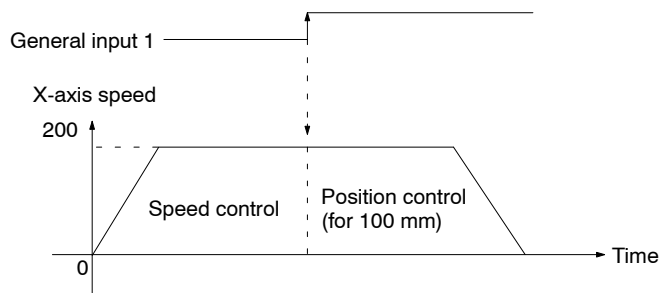
**Note** The value of speed command 1 must be larger than that of the speed command 2.

**Description**

The specified axis will be moved for the specified distance for positioning after general input 1 or 2 is turned ON.

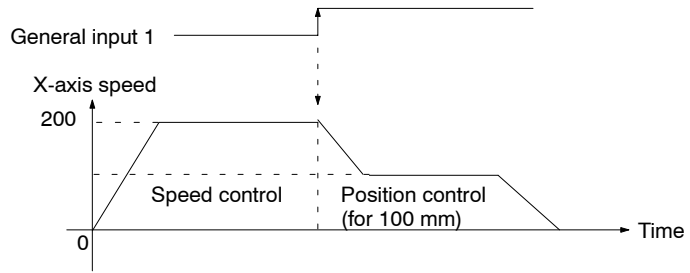
Refer to the following examples.

Example 1: Speed command 2 is not specified with the following format.  
G31 X100 F200 (mm)



The speed control of the X axis is performed and the X axis moves at 200 mm/s until general input 1 is turned ON. The X axis will move for 100 mm for positioning from the position where general input 1 was turned ON.

Example 2: Speed command 2 is specified with the following format.  
 G31 X100 F200 F100 (mm)



The speed control of the X axis is performed and the X axis moves at 200 mm/s until general input 1 is turned ON. The speed of the X axis will decelerate to the speed specified by speed command 2 (i.e., 100 mm/s) and the X axis will move for 100 mm for positioning from the position where general input 1 was turned ON.

There is no difference in acceleration/deceleration time between the G31 and G00 functions. The acceleration/deceleration time usually set automatically is proportional to the ratio of the speed command to the maximum feed rate. Refer to the following *Operation Examples*.

Refer to the following for the relationship between interrupt inching axes and general inputs.

Axis	General input
X	Rising edge of general input 1
Y	Rising edge of general input 2

The feed rate operation depends on whether the origin is defined or not.

With defined origin: The soft limit is valid and the current position is always refreshed.

With undefined origin: The soft limit is invalid and the current position will be set to 0 when the operation starts or a general input is turned ON.

**Note** The previous G01, G02, and G03 function will operate in stop mode instead of pass mode if this function is executed.

An error will result if the G31 function is executed if the value of speed command 2 is the same as or larger than that of speed command 1. Make sure that the value of the speed command 1 is larger than of speed command 2.

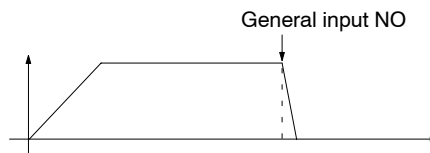
The value of a speed command must be the same as or less than the maximum feed rate. If the value is larger than the maximum feed rate, the axis will move at maximum feed rate and the override will be ignored.

**Operation Examples**

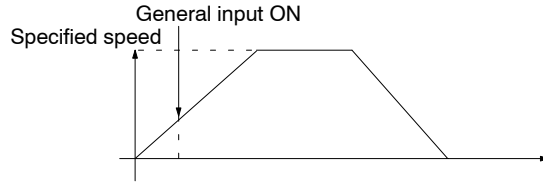
The following positioning curves are obtained according to the speed and movement of the axis if a general input is turned ON.

**Without Speed Command 2**

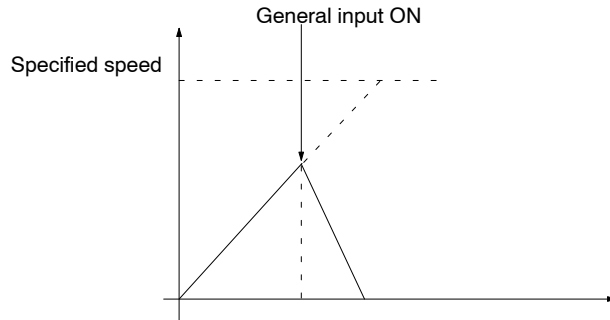
- 1, 2, 3... 1. A general input is turned ON while the axis is moving at constant speed, but the deceleration time is extremely short because the speed is high and the movement is small.



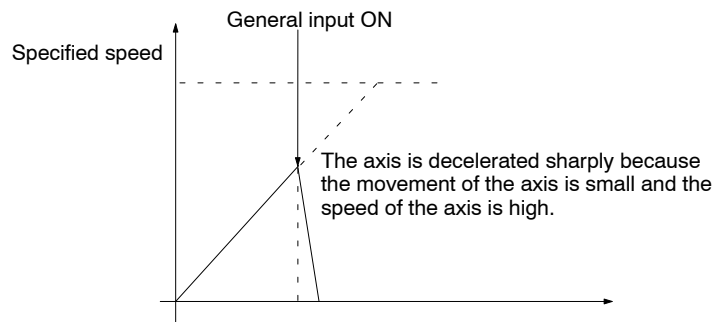
2. A general input is turned ON while the axis is being accelerated. (1)



3. A general input is turned ON while the axis is being accelerated. (2)

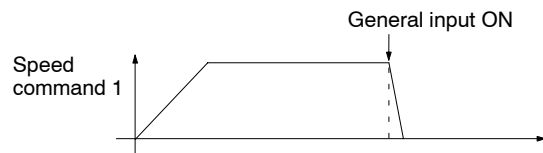


4. A general input is turned ON while the axis is being accelerated. (3)

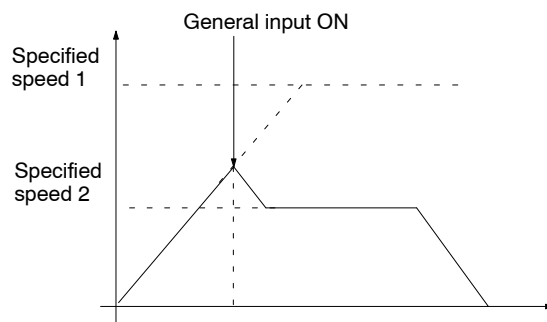


**With Speed Command 2**

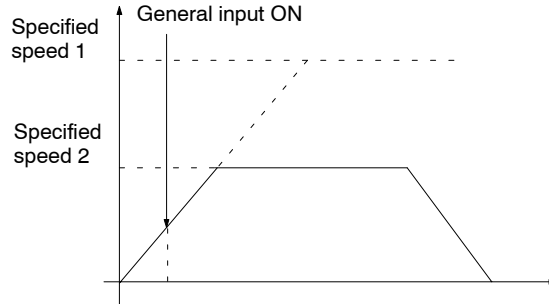
- 1, 2, 3... 1. A general input is turned ON while the axis is moving at constant speed, but the deceleration time is extremely short and the speed is not decelerated to what was specified with speed command 2 because the speed of the axis is high and the movement of the axis is small.



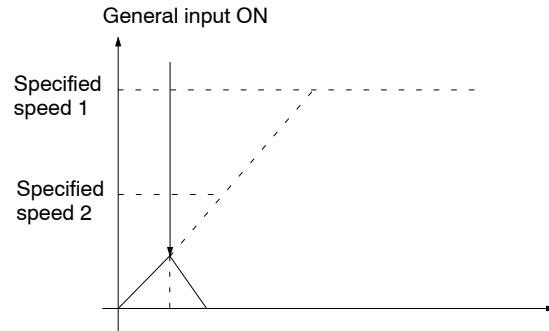
2. A general input is turned ON while the axis is being accelerated. (1)



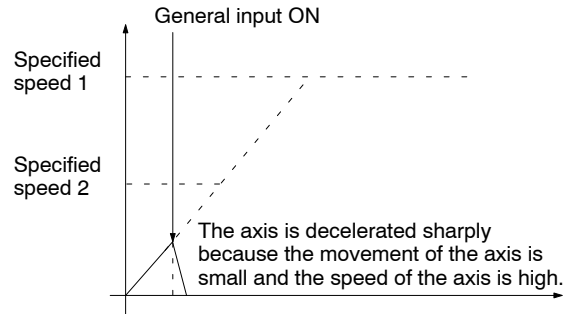
3. A general input is turned ON while the axis is being accelerated. (2)



4. A general input is turned ON while the axis is being accelerated. (3)



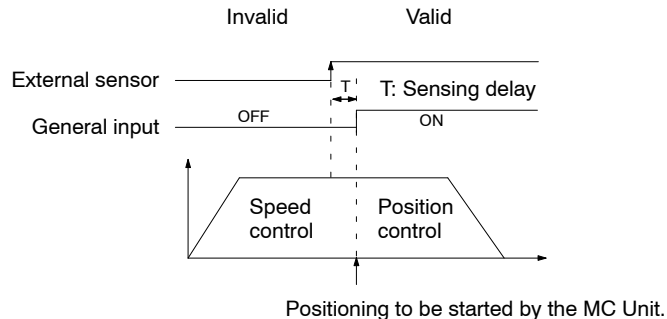
5. A general input is turned ON when the axis is being accelerated. (4)



**Supplementary Information**

When positioning using INTERRUPT INCHING, positions that are determined after external sensors become valid will vary depending on various conditions such as an ambient temperature. This is caused by a delay in sensing by external sensors and the general input circuit of the MC Unit.

Shown below is a description of the MC Unit's sensing delay assuming there is no sensing delay due to external sensors.



- The external sensor changes from invalid to valid.



- The status of this external sensor is taken in as the general input. There is a maximum delay of 1 ms in this general input circuit. Therefore, the MC Unit starts positioning control 1 ms max. after the external sensor becomes valid.
- Accordingly, the actual position to be determined by the MC Unit after the external sensor becomes valid can be obtained according to the following formula.

Actual movement amount [pulse] = Specified movement amount [pulse] + Feed rate [pps] x Sensing delay [s]

- Actual movement amount refers to the amount from the point where the external sensor becomes valid to the point where the machine stops movement.
- Specified movement amount refers to the amount specified by G31.
- Feed rate refers to the one specified by G31.
- Sensing delay is 0.001 s max.

### 5-5-14 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.

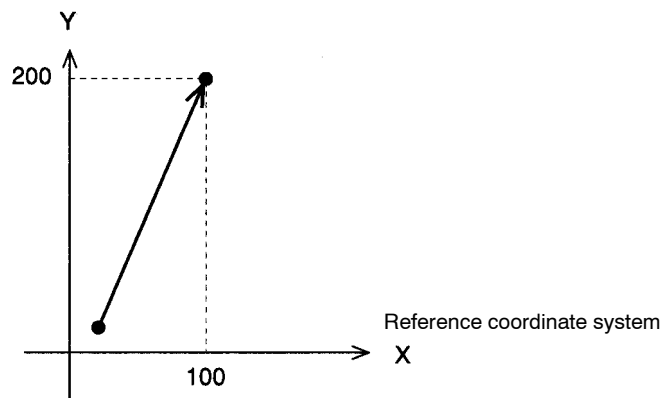
The reference coordinate system will be in effect after execution of the REFERENCE ORIGIN RETURN function (G26) or the ORIGIN SEARCH function (G28).

#### Example Program

```

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

```



### 5-5-15 G51: SELECT WORKPIECE COORDINATE SYSTEM

Sets the coordinate system to the workpiece coordinate system.

**Format** G51

**Parameters** None

**Description**

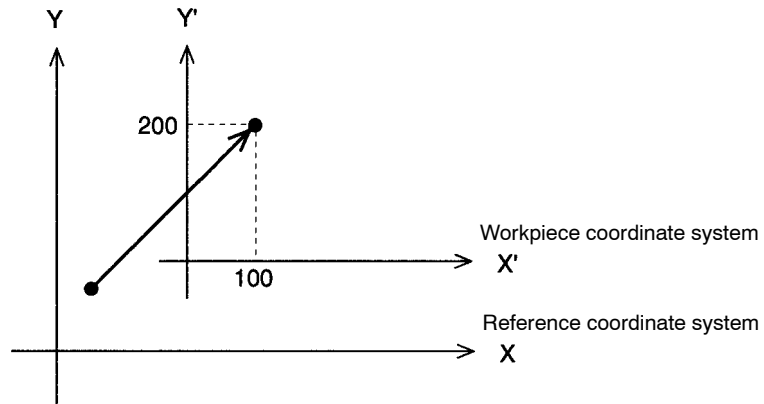
This function sets the coordinate system to the workpiece coordinate system. After this function is executed, the coordinate data in all axis operations is processed as workpiece coordinate data. If G51 is executed at the beginning of the program, the origin of workpiece coordinate system is determined 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
:      :
    
```



**5-5-16 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 and Y
	Data	-39,999,999 to +39,999,999 (E00 to E31) A0000 to A1999

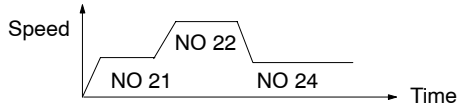
**Description**

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.

```

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

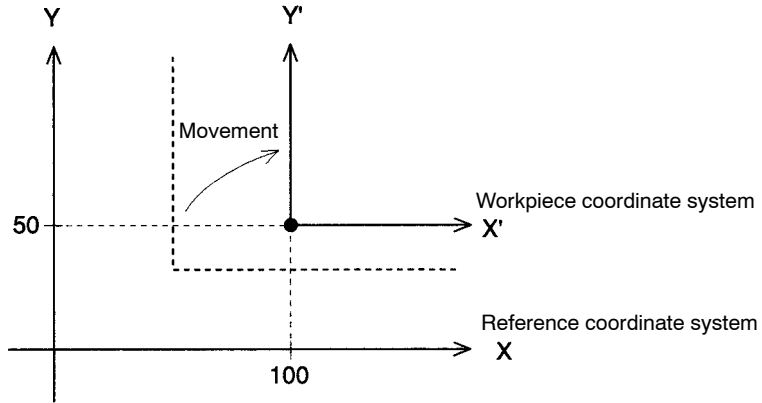


**Note** An error will occur if the offset value exceeds the stroke limits set in the system parameters.

**Example Program**

```

:      :
N010  G53  X100  Y50
:      :
    
```



**5-5-17 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 and Y
	Data	-39,999,999 to +39,999,999 (E00 to E31) A0000 to A1999

**Description**

This function changes the present value of the specified reference coordinate system to a specified value. For the absolute encoder axis, if the servo lock is executed after the servo free is executed, the position will be preset to the absolute position registered in the absolute encoder.



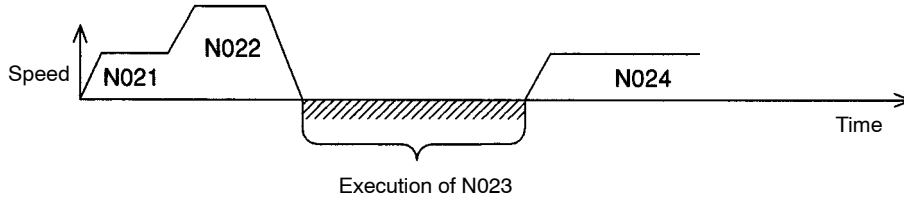
**Caution**

The present position presetting for the absolute encoder axis applies only to the present position registered in the MC Unit. The present position registered in the absolute encoder will not be preset. Therefore, the position will be automatically updated to the present position registered in the absolute encoder when the next time the servo lock is executed.

**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
:      :
    
```



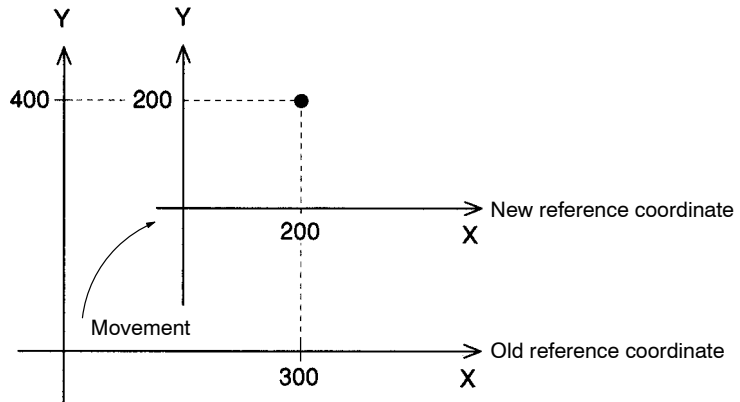
**Note** An error will occur if the offset value exceeds the stroke limits set in the system parameters.

**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
:      :
    
```



**5-5-18 G60: ARITHMETIC OPERATIONS**

Performs arithmetic operations on 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 will be 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
:      :
```

**5-5-19 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 or Y

**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 A1000=123.45
:      :
```

**5-5-20 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 and Y
	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.

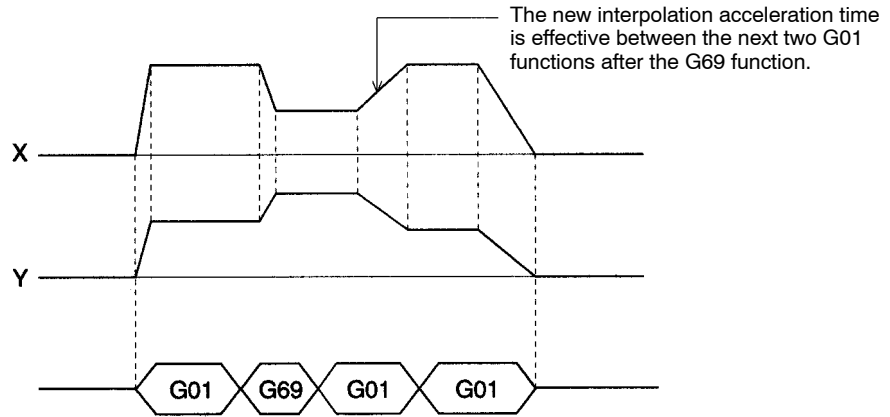
This function can't change the present value for absolute encoder axes.

**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
:      :
    
```



**5-5-21 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

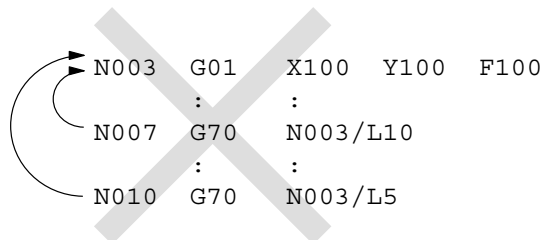
**Description**

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 5-5-23 G72: Subprogram Jump for details.

When the number of loops is set indirectly from position data with register specification or position data number specification, the number of loops will be 0 if the value of the position data is 0. 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.

**5-5-22 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
:      :

```



### 5-5-23 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 A0000 to A1999

**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.

If the subprogram number is set to the number of position data, the subprogram in the position data will be called and executed.

An error will result if the position data does not satisfy the following condition.

$$500 \leq \text{Position data} \leq 999$$

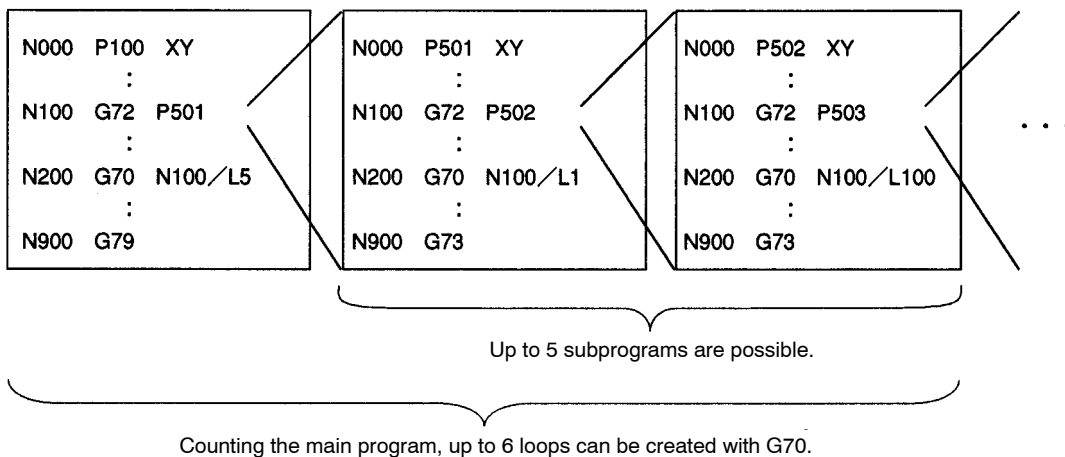
**Example Program**

```

:      :
N010  G72  P500
:      :

```

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



### 5-5-24 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  XY
:      :
:      :
:      :
N100  G73

```

### 5-5-25 G74: OPTIONAL END

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

**Format** G74□<optional number>

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

Parameter	Possible settings
Optional number	00 to 06

**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.

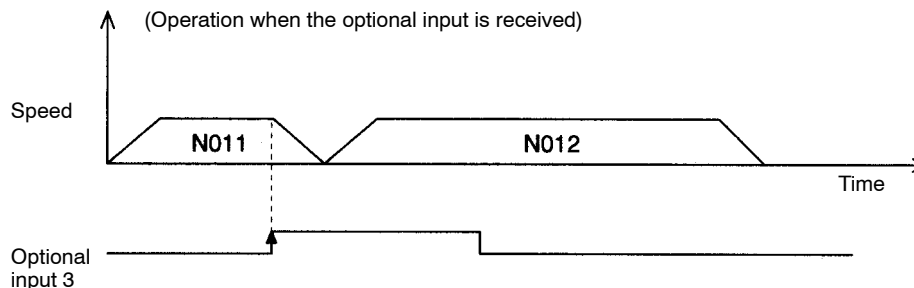
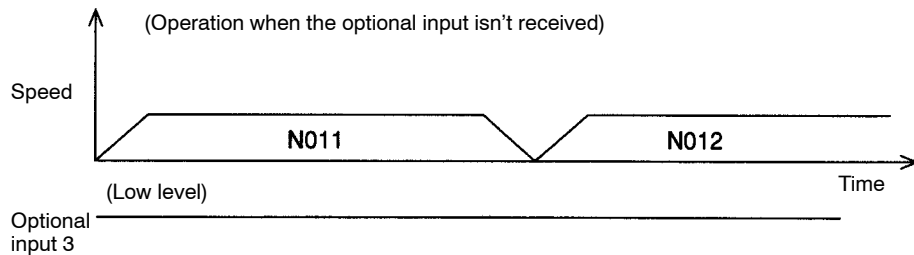
- 0 through 4: Inputs from the PC's data area interface
- 5 and 6: Inputs from general inputs 1 and 2

This function is ineffective if the following block contains a SUBPROGRAM END function (G73) or PROGRAM END function (G79). 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
:      :
    
```



### 5-5-26 G75: OPTIONAL SKIP

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

**Format** G75□<optional number>

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

Parameter	Possible settings
Optional number	00 to 06

**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.

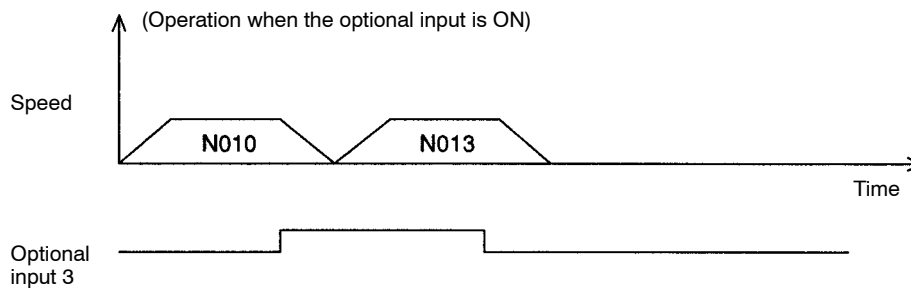
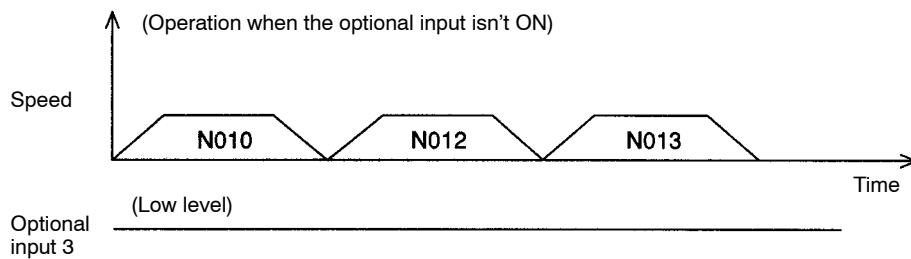
- 0 through 4: Inputs from the PC's data area interface
- 5 and 6: Inputs from general inputs 1 and 2

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 (G73) or PROGRAM END function (G79).

**Example Program**

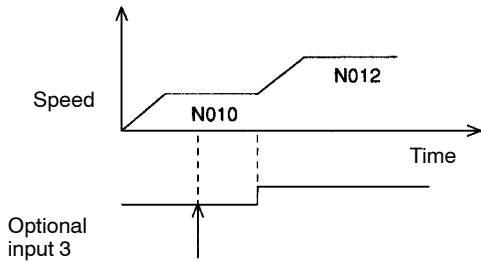
```

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

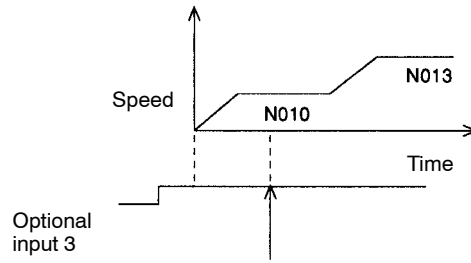


When block N008 contains the pass mode function (G10), blocks N011 and N012 are pre-executed while N010 is being executed, so block N012 won't be skipped if optional input 3 is turned ON after execution of block N010.

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



Block N012 isn't skipped because optional input 3 isn't ON when block N011 is pre-executed.



In this case, block N012 is skipped because optional input 3 is ON when block N011 is pre-executed.

### 5-5-27 G76: OPTIONAL PROGRAM PAUSE

Pauses the program when the specified optional input is ON.

**Format**

G76□<optional number>

**Parameters**

The following table shows the possible settings for the parameters.

Parameter	Possible settings
Optional number	00 to 06

**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.

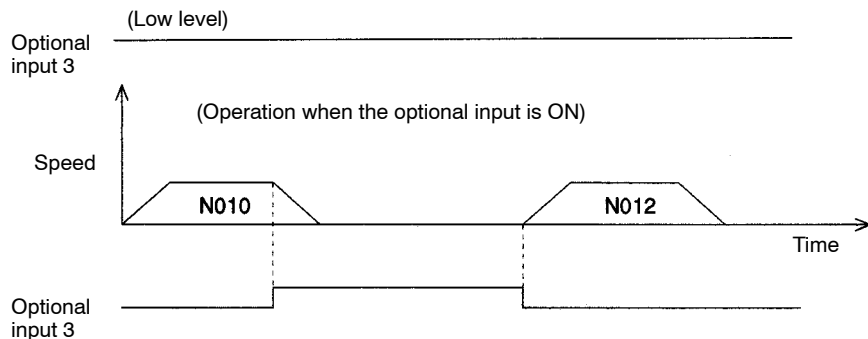
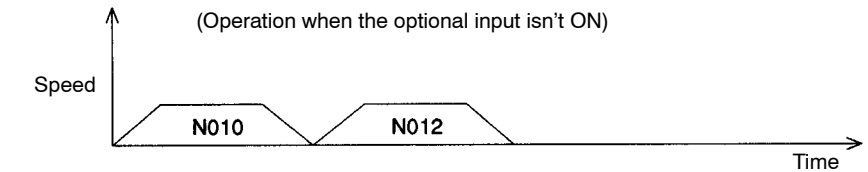
- 0 through 4: Inputs from the PC's data area interface
- 5 and 6: Inputs from general inputs 1 and 2

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

**Example Program**

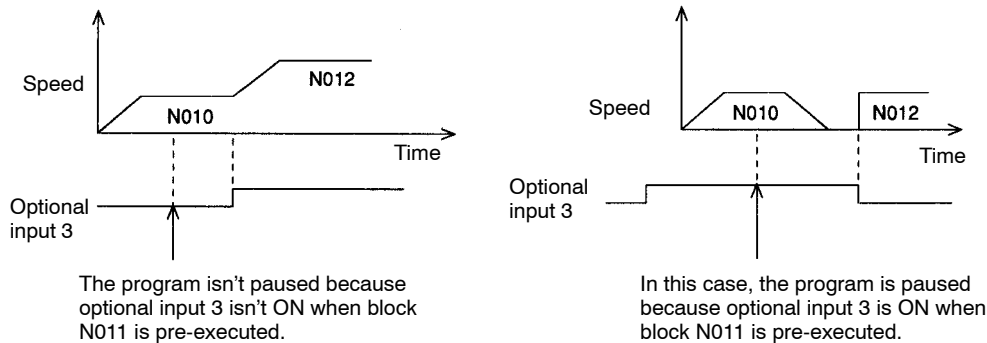
```

:      :
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.



### 5-5-28 G79: PROGRAM END

Ends the main program.

**Format** G79

**Parameters** None

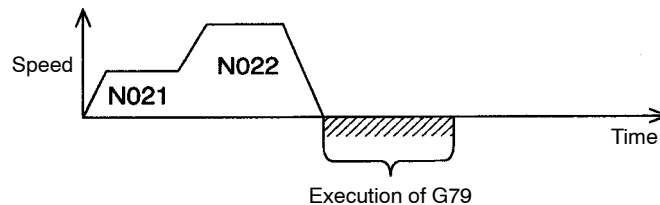
**Description** This function ends the main program and must be included at the end of the main program.

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           Executed after N022 is completed.
:      :
    
```



### 5-5-29 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.

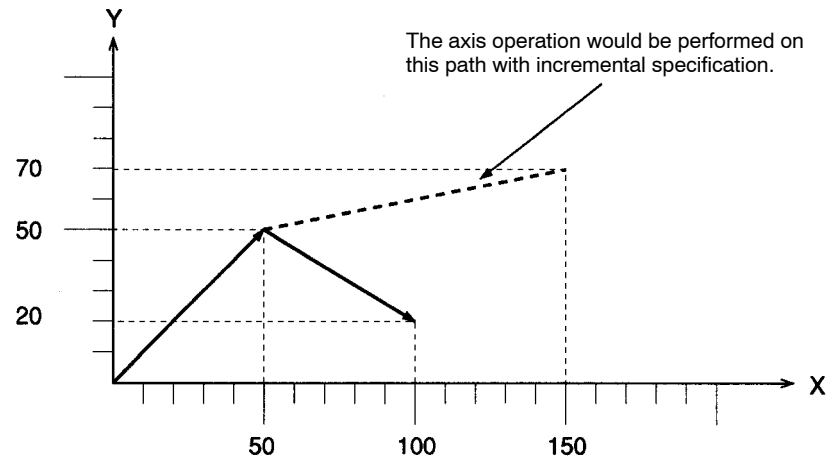
### Example Program

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

```

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

```



### 5-5-30 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.

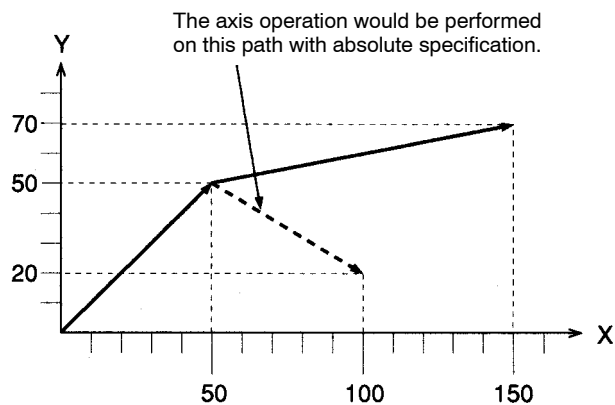
### Example Program

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

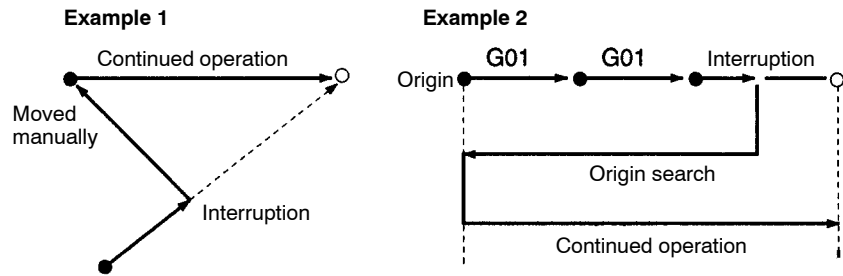
```

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

```



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 consist of information used to interlock with external devices in positioning operation processes. External devices are devices connected to the PC.

#### Example Program

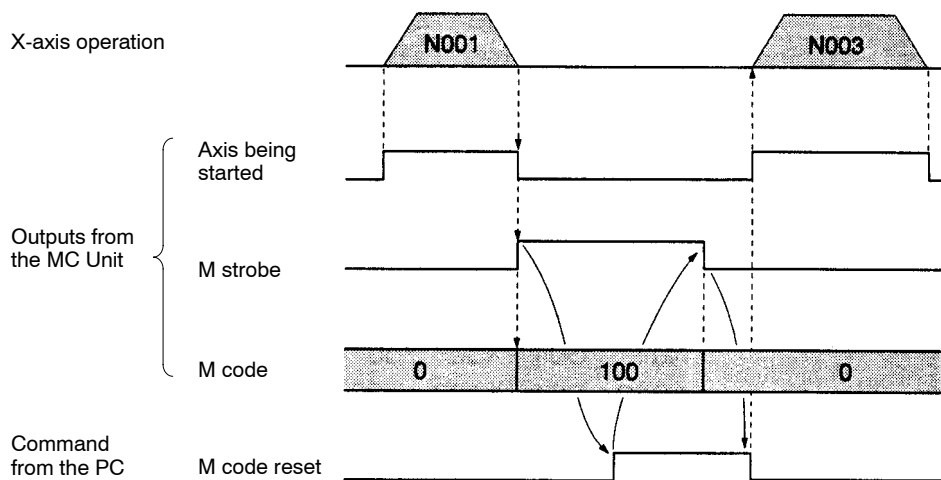
The following program shows a practical example of M code usage.

```

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.
N004 G79 ..... Declares the end of the program.
    
```

#### Timing Chart

The following diagram shows the timing of the example program's execution. The axis being started, 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. 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.

#### 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 are cleared when the PROGRAM END function (G79) is executed.

### 5-6-3 M-code Examples

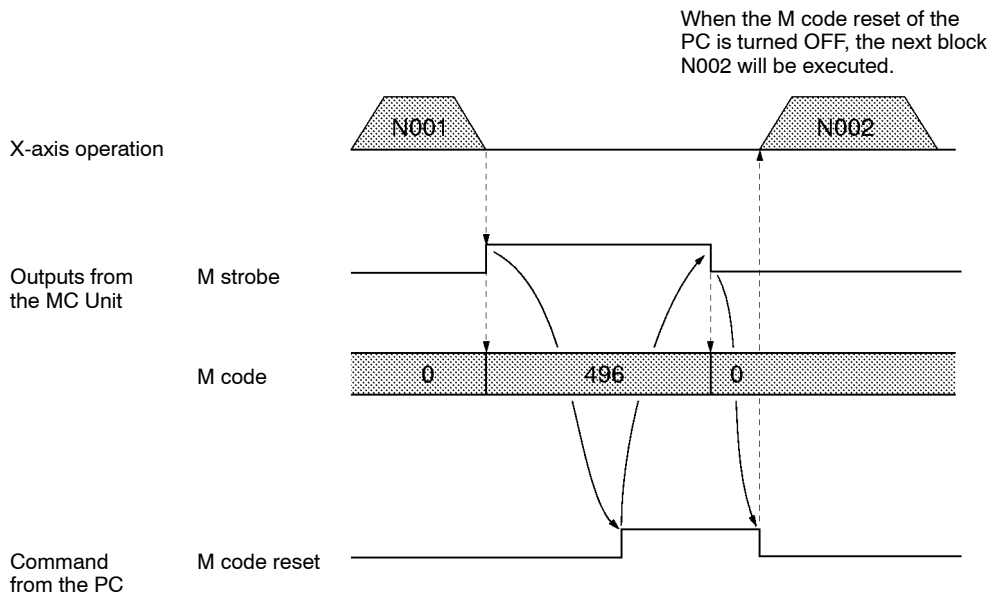
M codes can be used independently or in conjunction with "G codes" such as G00 and G01 which execute axis operations.

#### Example 1

The following program outputs M code 496.

```

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



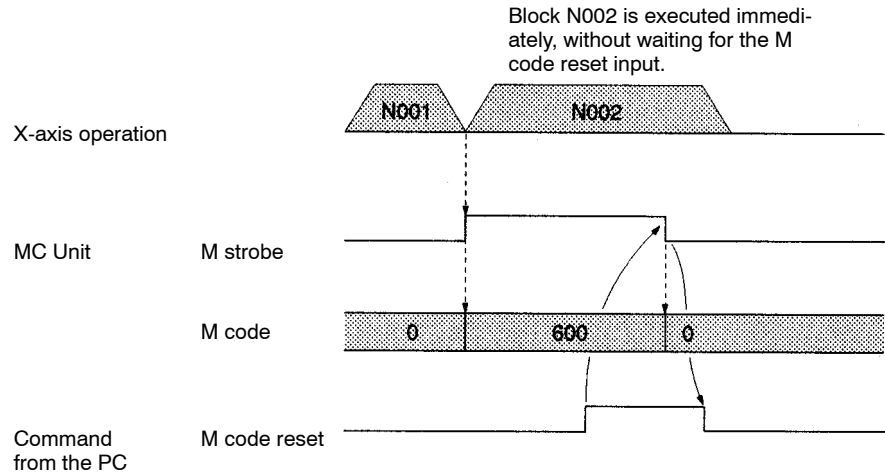


**Example 2**

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
    
```



**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 M codes M000 to M999.

**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.

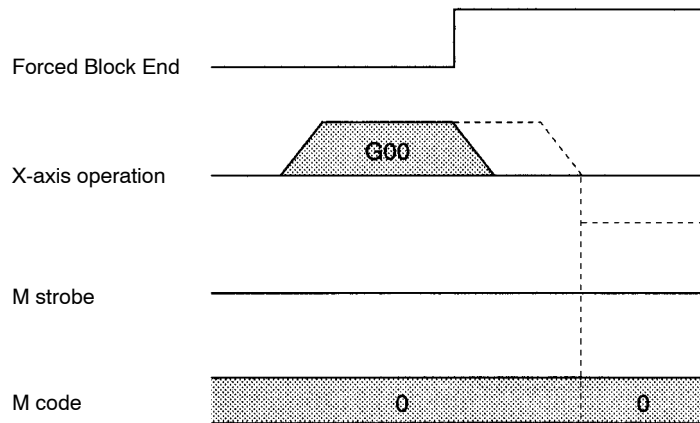
**5-6-6 Clearing M000 to M499 with Forced Block End or G74**

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.

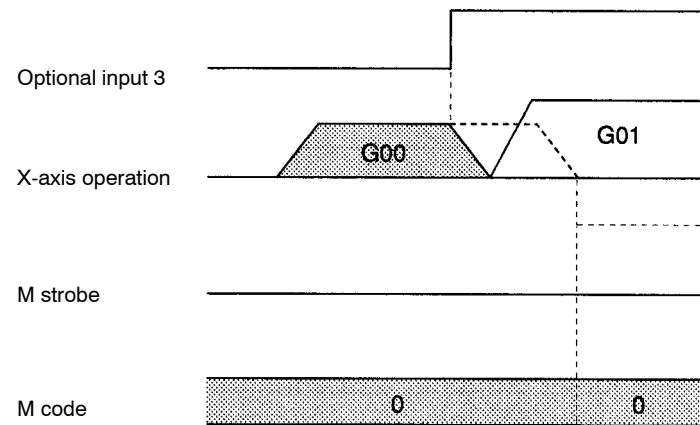
```
G00 X500 M100
```



**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.

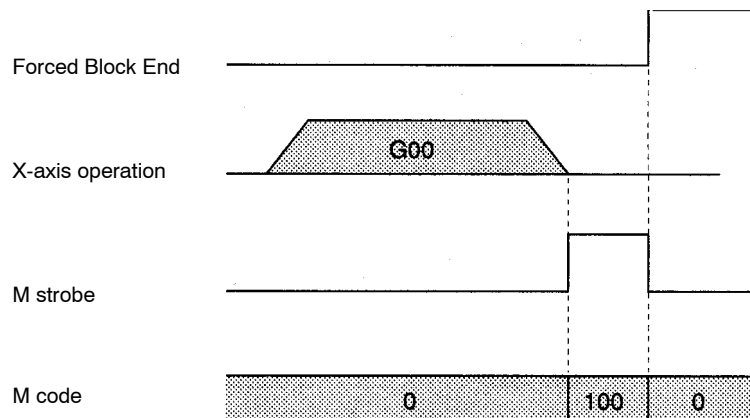
```
G74 3
G00 X500 M100
G01 X1000
```



**Example 3**

When the output has been completed and the system is set to wait for the M code reset, the M strobe will be turned OFF and the M code will be cleared to release the M code reset.

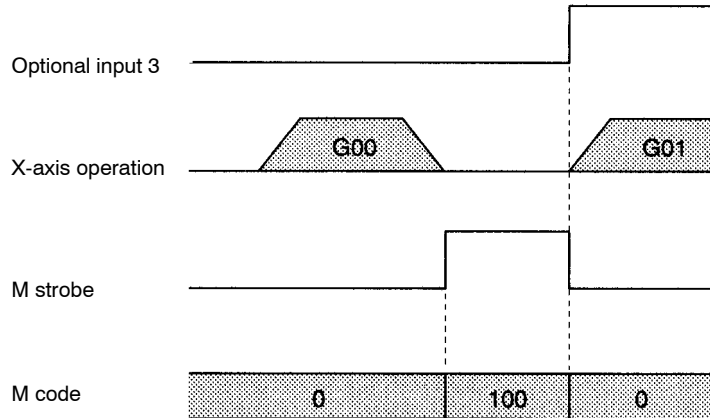
```
G00 X500 M100
```



**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
```



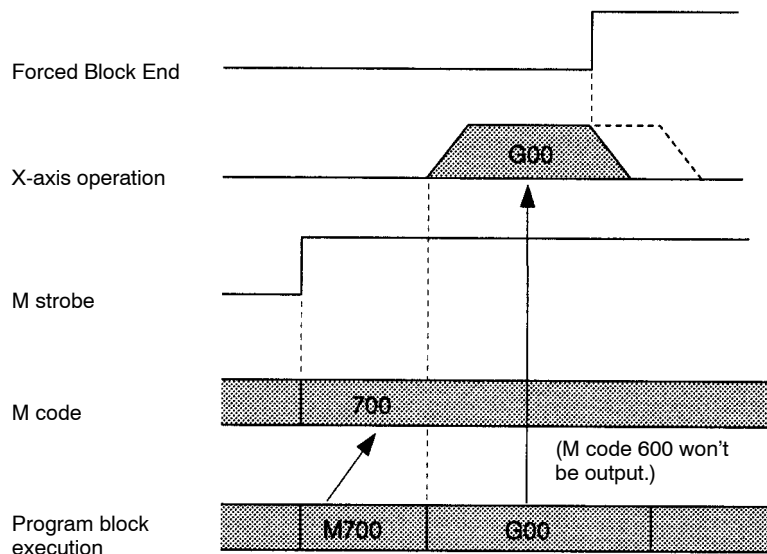
**5-6-7 Clearing M500 to M999 with Forced Block End or G74**

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.

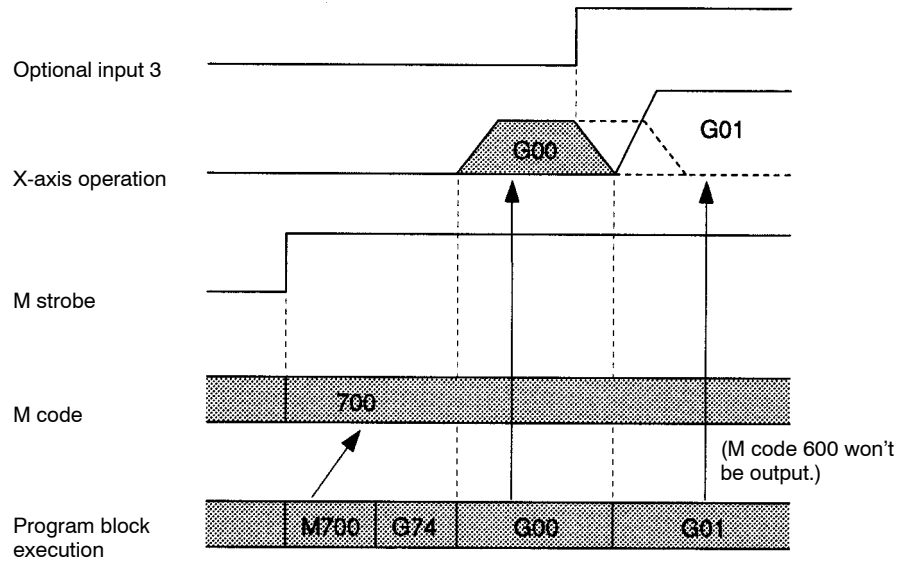
```
M700
G00 X500 M600
```



**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

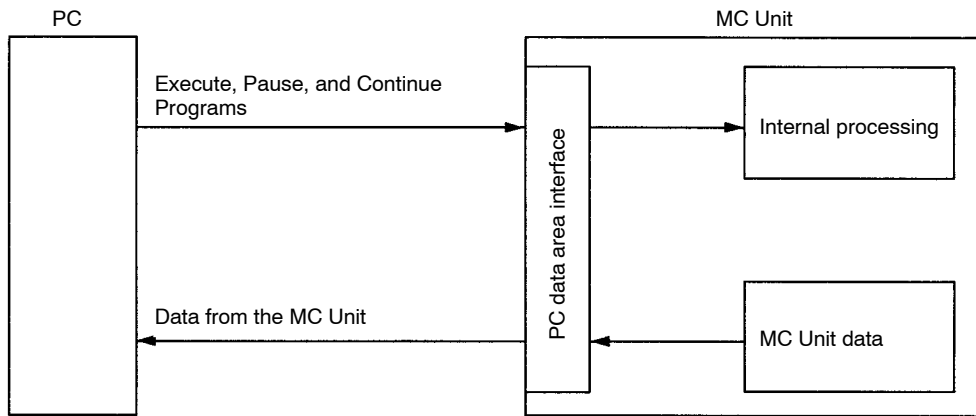
This section describes the interface area used to exchange information between the PC and MC Unit, such as commands from the PC and status information from the MC Unit.

6-1	Introduction .....	115
6-2	Allocation of the PC Data Area Interface .....	116
6-2-1	C200H/C200HS PC Interface .....	116
6-2-2	C200HX/C200HG/C200HE PC Interface .....	117
6-3	Controlling the MC Unit from the PC .....	118
6-3-1	Manual Mode .....	118
6-3-2	Automatic Mode .....	119
6-3-3	Sample Ladder Program .....	120
6-4	Interface Bits .....	121
6-4-1	Initial Setting Area .....	121
6-4-2	I/O Refresh Area .....	123
6-4-3	Expansion Data Area .....	132
6-5	Interface Bit Specifics .....	134
6-5-1	Word n+1 Bit 01: Automatic/Manual Mode Bit (Task 1) .....	134
6-5-2	Word n+1 Bit 02: Cycle Start Bit (Task 1) .....	135
6-5-3	Word n+1 Bit 03: Single Block Bit (Task 1) .....	137
6-5-4	Word n+1 Bit 04: Forced Block End Bit (Task 1) .....	138
6-5-5	Word n+1 Bit 05: Pause Bit (Task 1) .....	139
6-5-6	Word n+1 Bit 06: M Code Reset Bit (Task 1) .....	140
6-5-7	Word n+1 Bit 07: Program Number Read Bit (Task 1) .....	142
6-5-8	Word n+1 Bit 08: Teaching Address Set Bit (Task 1) .....	142
6-5-9	Word n+1 Bit 09: Teaching Bit (Task 1) .....	142
6-5-10	Word n+1 Bit 10: Teaching Type Bit (Task 1) .....	143
6-5-11	Word n+1 Bit 11: Transmission Control Bit (All Tasks) .....	144
6-5-12	Word n+1 Bit 12: Reception Control Bit (All Tasks) .....	145
6-5-13	Word n+1 Bit 13: Flash Memory Write Bit (All Tasks) .....	145
6-5-14	Word n+1 Bit 14: Error Reset Bit (All Tasks) .....	146
6-5-15	Word n+1 Bit 15: Expansion Data Read Bit (All Tasks) .....	147
6-5-16	Word n+3 Bits 11 to 15: Optional Inputs .....	147
6-5-17	Word n+5 Bit 00: Deceleration Stop Bit (X-axis) .....	148
6-5-18	Word n+5 Bit 01: Origin Search Bit (X-axis) .....	149
6-5-19	Word n+5 Bit 02: Reference Origin Return Bit (X-axis) .....	150
6-5-20	Word n+5 Bit 03: Jogging Bit (X-axis) .....	151
6-5-21	Word n+5 Bit 13: Jog Direction Bit (X-axis) .....	152
6-5-22	Word n+5 Bit 05: Current Position Preset Bit (X-axis) .....	152
6-5-23	Word n+5 Bit 06: ABS Initialization Bit (X-axis) .....	153
6-5-24	Word n+5 Bit 07: ABS Soft Reset Bit (X-axis) .....	154
6-5-25	Word n+5 Bit 08: Enable MPG Bit (X-axis) .....	155
6-5-26	Word n+5 Bits 14 and 15: MPG Ratio Specifiers 1 and 2 (X-axis) .....	156
6-5-27	Word n+5 Bit 09: Servo Lock Bit (X-axis) .....	156
6-5-28	Word n+5 Bit 10: Servo Free Bit (X-axis) .....	157
6-5-29	Word n+5 Bit 11: Driver Alarm Reset Bit (X-axis) .....	157
6-5-30	Word n+5 Bit 12: Override Setting Bit (X-axis) .....	158
6-5-31	Word n+8 Bit 06: Transmission Completion Flag .....	159
6-5-32	Word n+8 Bit 07: Data Transmission Error Flag .....	159
6-5-33	Word n+8 Bit 08: Reception Completion Flag .....	160

6-5-34	Word n+8 Bit 09: Data Reception Error Flag .....	161
6-5-35	Word n+8 Bit 10: Flash Memory Write Completion Flag .....	162
6-5-36	Word n+8 Bit 11: Teaching Box Enabled Flag .....	162
6-5-37	Word n+8 Bit 12: Teaching Box Protected Flag .....	162
6-5-38	Word n+8 Bit 13: Initial Setting Alarm Flag .....	162
6-5-39	Word n+8 Bit 14: Error Flag .....	163
6-5-40	Word n+8 Bit 15: Expansion Data Read Completion Flag .....	163
6-5-41	Word n+11 Bit 00: Task Error Flag (Task 1) .....	164
6-5-42	Word n+11 Bit 02: Automatic Mode Flag (Task 1) .....	164
6-5-43	Word n+11 Bit 03: Manual Mode Flag (Task 1) .....	165
6-5-44	Word n+11 Bit 04: Memory Run Flag (Task 1) .....	165
6-5-45	Word n+11 Bit 05: Memory Run Completed Flag (Task 1) .....	165
6-5-46	Word n+11 Bit 06: Optional Input Standby Flag (Task 1) .....	166
6-5-47	Word n+11 Bit 07: M Code Reset Standby Flag (Task 1) .....	166
6-5-48	Word n+11 Bit 08: M Strobe Flag (Task 1) .....	167
6-5-49	Word n+11 Bit 09: Cycle Start Received Flag (Task 1) .....	167
6-5-50	Word n+11 Bit 12: Teaching Address Setting Completed (Task 1) .....	168
6-5-51	Word n+11 Bit 13: Teaching Completed Flag (Task 1) .....	168
6-5-52	Word n+11 Bit 14: Teaching Error Flag (Task 1) .....	169
6-5-53	Word n+11 Bit 15: Teaching Address Over Flag (Task 1) .....	169
6-5-54	Word n+16 Bit 00: Reference Origin Flag (X-axis) .....	170
6-5-55	Word n+16 Bit 01: Busy Flag (X-axis) .....	170
6-5-56	Word n+16 Bit 02: Servo Lock ON Flag (X-axis) .....	171
6-5-57	Word n+16 Bit 03: No Origin Flag (X-axis) .....	171
6-5-58	Word n+16 Bit 04: Axis Operating Flag (X-axis) .....	172
6-5-59	Word n+16 Bit 05: Positioning Completed Flag (X-axis) .....	173
6-5-60	Word n+16 Bit 06: Error Counter Alarm Flag (X-axis) .....	173
6-5-61	Word n+16 Bit 07: Alarm Input (X-axis) .....	174
6-5-62	Word n+16 Bits 08 to 15: Zone Flags (X-axis) .....	174
6-5-63	Word n+19 Bit 15: MPG Operation Flag (X-axis) .....	174

## 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.

### PC Cycle Time

The cycle time will be extended as shown in the following table for each Unit mounted to the CPU Backplane. Take this into consideration when designing the system.

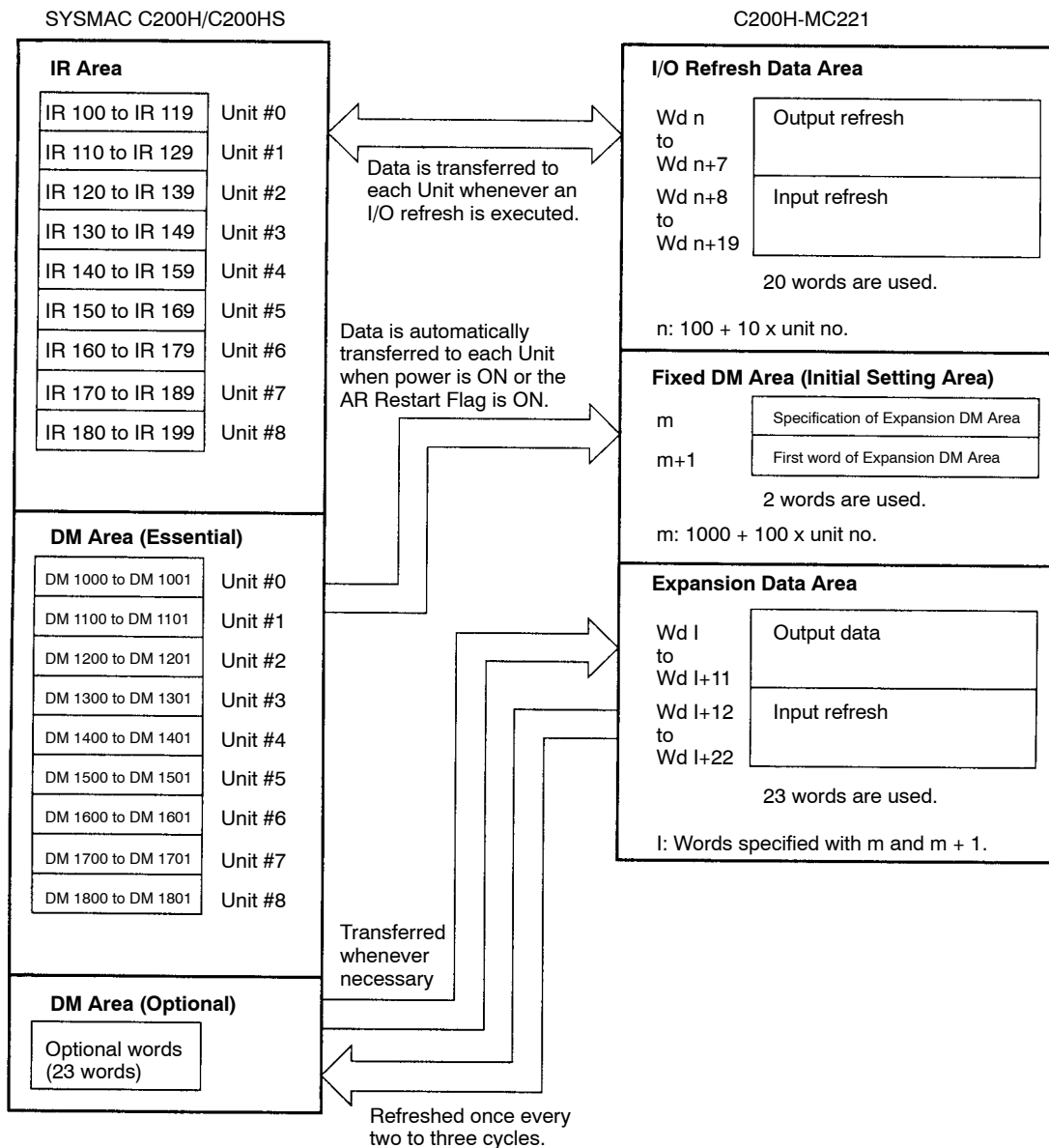
PC model	C200H	C200HS	C200HX/HG/HE
Case 1	Approx. 4 ms	Approx. 3 ms	Approx. 2 ms
Case 2	Approx. 5 ms	Approx. 4 ms	Approx. 3 ms

Case 1: When refreshing only the I/O refresh data area (normal case).

Case 2: When refreshing the expansion data area as well.

## 6-2 Allocation of the PC Data Area Interface

### 6-2-1 C200H/C200HS PC Interface



#### Explanation

Each MC Unit occupies 20 words within a range of 100 to 199 words of the Special I/O Unit area as an I/O refresh data area.

Each MC Unit occupies two words as a fixed data area (initial setting area) within a range of DM 1000 to DM 1999.

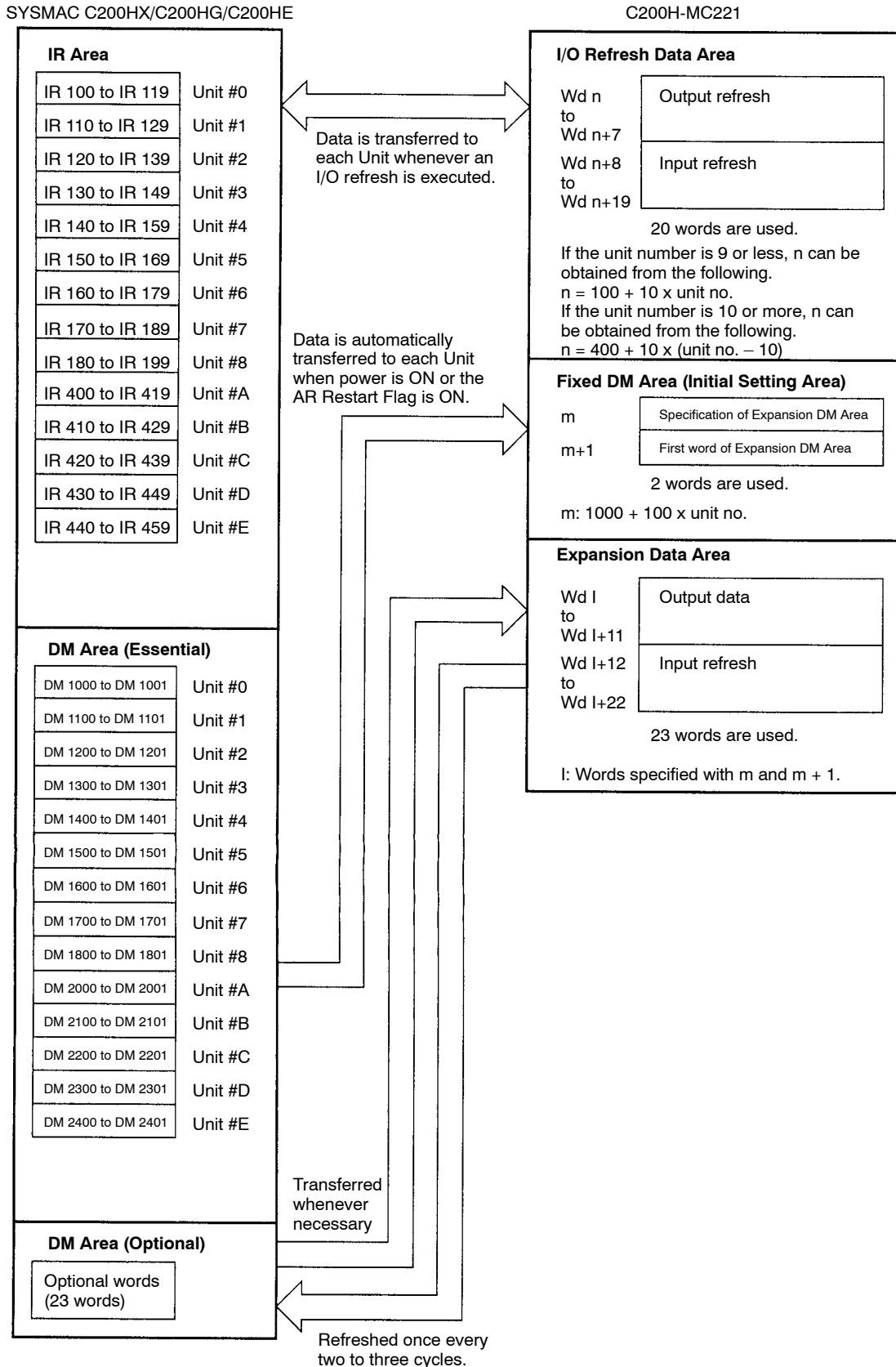
If an expansion DM area is made effective by designating the above initial setting area, the area will occupy 23 words except from DM 1000 to DM 1999. The first word of the expansion DM area is specified in the initial setting area.

- Note**
1. Each unit number must be unique. Do not set unit 9.
  2. The DM memory must be within a range of DM 7000 to DM 7999 for the C200HS in linear mode.

Refer to 1-3 *Setting the Unit Number* for details.



### 6-2-2 C200HX/C200HG/C200HE PC Interface



Each MC Unit uses 20 words from IR 100 to IR 199 and IR 400 to IR 459 of the Special I/O Unit area as an I/O refresh data area.

Each MC Unit uses two words as a Fixed DM Area (i.e., an initial setting area) between DM 1000 and DM 2599.

If an expansion DM area is effective, the area occupies 23 words except from DM 1000 to DM 2599. The first word of the expansion DM area is specified in the initial setting area.

**Note** Each unit number must be unique. Do not set unit #9 and F.

## 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 *6-5 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
- ABS Initialization
- ABS Soft Reset
- Enable MPG
- Servo Lock
- Servo Free

Each command operates independently on each axis. Either the automatic mode or manual mode can be selected by turning ON or OFF the automatic/manual mode bit.

### 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, ABS Initialization, 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**

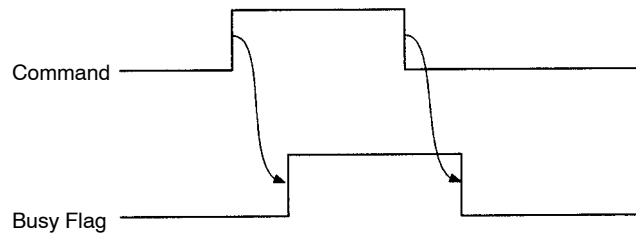
When more than one command is executed at the same time, the commands will be processed in the following order.

- 1, 2, 3...**
1. Deceleration Stop
  2. Origin Search
  3. Reference Origin Return
  4. Jogging
  5. Current Position Preset
  6. ABS Initialization
  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, ABS initialization, 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**

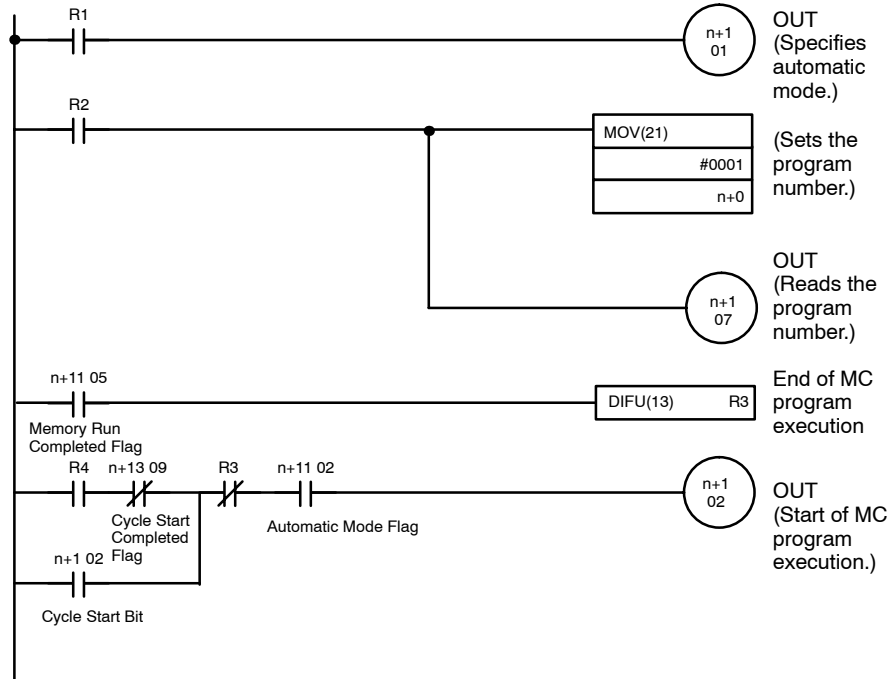
The MC program is executed in automatic mode. Use the following procedure to operate in automatic mode. This example is for task 1.

- 1, 2, 3...**
1. Set the MC Unit to automatic mode. Turn ON bit 01 of word n+1.
  2. Set the program number of the program you want to execute. (Word n+0)
  3. Turn ON the Program Number Read bit. Turn ON bit 07 of word n+1.
  4. The specified program will be executed when the Cycle Start bit (bit 02 of word n+1) is turned ON.

Refer to 6-5 *Interface Bit Specifics* for the reception control bit, program number read bit, cycle start bit, and automatic/manual mode bit.

### 6-3-3 Sample Ladder Program

The following sample program incorporates the procedure to execute the MC program. 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.

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.

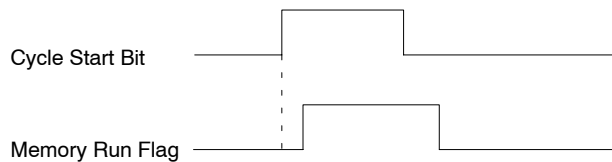
Turn ON input R2 in the following situations:

- 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.

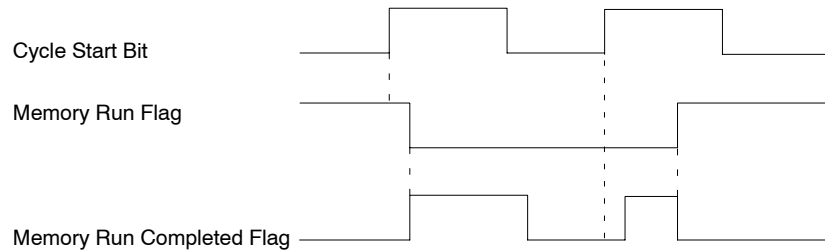
Execution of the MC program will begin when R4 goes ON.

#### 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 IR area of the PC data area interface. Refer to *6-5 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 (↓).

The following interface areas are explained below.

- Initial setting area
- I/O refresh area
- Expansion data area

### 6-4-1 Initial Setting Area

The expansion data area is specified in the initial setting area.

If the expansion data area is not specified or wrongly specified, initial setting alarm bit 13 of word  $n+8$  of the I/O refresh area will be turned ON.

The expansion data area cannot be used while this bit is turned ON. The MC Unit can be, however, in JOG or programmed operation without using the expansion data area. Refer to the following for the expansion data area.

Output

Word	Bit	Description										
m	00 to 15	<p>Specification of expansion data area (16-bit binary)</p> <div style="border: 1px solid black; width: 150px; height: 20px; margin: 5px 0;"></div> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 20%;">A</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>0000</td> <td>The expansion data area is not used.</td> </tr> <tr> <td>000D</td> <td>Sets the expansion data area in the DM area.</td> </tr> <tr> <td>000E</td> <td>Sets the expansion data area in the EM area.</td> </tr> </tbody> </table> <p>Specifies whether to use the expansion data area and sets the expansion data area in the DM or EM area.</p> <p>The EM area can be set with the C200HX or C200HG and bank 0 will be used. Refer to the <i>C200HX/C200HG/C200HE PCs Operation Manual</i> for the EM area and bank 0.</p>	A	Meaning	0000	The expansion data area is not used.	000D	Sets the expansion data area in the DM area.	000E	Sets the expansion data area in the EM area.		
A	Meaning											
0000	The expansion data area is not used.											
000D	Sets the expansion data area in the DM area.											
000E	Sets the expansion data area in the EM area.											
m+1	00 to 15	<p>Specification of first word of expansion data area (4-digit BCD)</p> <div style="border: 1px solid black; width: 150px; height: 20px; margin: 5px 0; display: flex; justify-content: space-around;"> <span>x10<sup>3</sup></span> <span>x10<sup>2</sup></span> <span>x10<sup>1</sup></span> <span>x10<sup>0</sup></span> </div> <p>Specifies the first word of the expansion data area.</p> <p>Example: Set the word to 100 for D0100.</p> <p>The expansion data area varies with the PC model to which the MC Unit is mounted. Refer to the following table.</p> <p>If the expansion data area is allocated incorrectly, the initial setting alarm bit 13 of word n+8 will be turned ON.</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 30%;">PC model</th> <th>Expansion data area</th> </tr> </thead> <tbody> <tr> <td>C200H</td> <td>DM 0000 to DM 0899</td> </tr> <tr> <td>C200HS C200HE(-Z )</td> <td>DM 0000 to DM 0899 DM 2000 to DM 5999</td> </tr> <tr> <td>C200HX/ C200HG (-Z)</td> <td>[Models conforming to Unit No. 10) DM 0000 to DM 0999 DM 2000 to DM 5999 EM 0000 to EM 6143</td> </tr> <tr> <td></td> <td>[Models conforming to Unit No. 16) DM 0000 to DM 0999 DM 2600 to DM 5999 EM 0000 to EM 6143</td> </tr> </tbody> </table>	PC model	Expansion data area	C200H	DM 0000 to DM 0899	C200HS C200HE(-Z )	DM 0000 to DM 0899 DM 2000 to DM 5999	C200HX/ C200HG (-Z)	[Models conforming to Unit No. 10) DM 0000 to DM 0999 DM 2000 to DM 5999 EM 0000 to EM 6143		[Models conforming to Unit No. 16) DM 0000 to DM 0999 DM 2600 to DM 5999 EM 0000 to EM 6143
PC model	Expansion data area											
C200H	DM 0000 to DM 0899											
C200HS C200HE(-Z )	DM 0000 to DM 0899 DM 2000 to DM 5999											
C200HX/ C200HG (-Z)	[Models conforming to Unit No. 10) DM 0000 to DM 0999 DM 2000 to DM 5999 EM 0000 to EM 6143											
	[Models conforming to Unit No. 16) DM 0000 to DM 0999 DM 2600 to DM 5999 EM 0000 to EM 6143											

## 6-4-2 I/O Refresh Area

Refer to the following table for the I/O refresh area.

## IR Bits

## Output

Word	Bit	Description	Page				
n	00 to 15	Task 1 program numbers 0000 to 0999 (4-digit BCD) <table border="1" style="margin-left: 20px;"> <tr> <td>0</td> <td>x10<sup>2</sup></td> <td>x10<sup>1</sup></td> <td>x10<sup>0</sup></td> </tr> </table> Specifies program numbers executed in automatic mode. If the Control Reception Bit is turned ON, a program number is read at the rising edge of the cycle start to execute the program.	0	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>	135
0	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>				
n+1	Control Bits for Task 1	00	Not used	---			
		01	Automatic/Manual Mode	↑	Any axes operating in manual mode will decelerate to a stop.	134	
				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.	135	
				1	Continues MC program execution.		
				0	Stops MC program execution.		
		03	Single Block	1	Executes a single block.	137	
		04	Forced Block End	↑	Forces an end to the block.	138	
	1			Prevents cycle start.			
	05	Pause	↑	Pauses execution.	139		
			1	Prevents cycle start.			
	06	M Code Reset Bit	↑	Resets the M code.	140		
			1	M code reset standby. (Prevents M code output.)			
			↓	Clears M code reset standby.			
	07	Program Number Read Bit	1	Reads the program number.	142		
	08	Teaching Address Set Bit	↑	Sets the address when teaching.	142		
	09	Teaching Bit	↑	Starts teaching.	142		
	10	Teaching Type Bit	1	Current position (feedback value)	143		
0			Target position				
All tasks common 1	11	Transmission Control Bit	↑	Starts transmission.	144		
	12	Reception Control Bit	↑	Starts reception.	145		
	13	Flash Memory Write Bit	↑	Starts writing data to the flash memory.	145		
	14	Error Reset Bit	↑	Resets an error.	146		
	15	Expansion Data Read Bit	1	Refreshes the expansion data area.	147		
0			Leaves the expansion area without refreshing.				
n+2	00 to 15	Task 2 program numbers 0000 to 0999 (4-digit BCD) <table border="1" style="margin-left: 20px;"> <tr> <td>0</td> <td>x10<sup>2</sup></td> <td>x10<sup>1</sup></td> <td>x10<sup>0</sup></td> </tr> </table> Specifies program numbers executed in automatic mode. If the Control Reception Bit is turned ON, a program number is read at the rising edge of the cycle start to execute the program.	0	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>	135
0	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>				

Word	Bit	Description		Page				
n+3	Control Bits for Task 2	00	Not used		---			
		01	Automatic/Manual Mode	↑	Any axes operating in manual mode will decelerate to a stop.	134		
				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.	135		
				1	Continues MC program execution.			
				0	Stops MC program execution.			
		03	Single Block	1	Executes a single block.	137		
		04	Forced Block End	↑	Forces an end to the block.	138		
	1			Prevents cycle start.				
	05	Pause	↑	Pauses execution.	139			
			1	Prevents cycle start.				
	06	M Code Reset Bit	↑	Resets the M code.	140			
			1	M code reset standby. (Prevents M code output.)				
			↓	Clears M code reset standby.				
	07	Program Number Read Bit	1	Reads the program number.	142			
	08	Teaching Address Set Bit	↑	Sets the address when teaching.	142			
	09	Teaching Bit	↑	Starts teaching.	142			
	10	Teaching Type Bit	1	Current position (feedback value)	143			
0			Target position					
All task common 2	11	Optional input 0	1	Optional input 0 is ON.	147			
			0	Optional input 0 is OFF.				
	12	Optional input 1	1	Optional input 1 is ON.				
			0	Optional input 1 is OFF.				
	13	Optional input 2	1	Optional input 2 is ON.				
			0	Optional input 2 is OFF.				
	14	Optional input 3	1	Optional input 3 is ON.				
			0	Optional input 3 is OFF.				
	15	Optional input 4	1	Optional input 4 is ON.				
			0	Optional input 4 is OFF.				
n+4	00 to 15	X-axis override 0000 to 1999 (4-digit BCD) <table border="1" style="margin: 5px auto; border-collapse: collapse;"> <tr> <td style="padding: 2px 5px;"><math>\times 10^2</math></td> <td style="padding: 2px 5px;"><math>\times 10^1</math></td> <td style="padding: 2px 5px;"><math>\times 10^0</math></td> <td style="padding: 2px 5px;"><math>\times 10^{-1}</math></td> </tr> </table> With 0.1% increments		$\times 10^2$	$\times 10^1$	$\times 10^0$	$\times 10^{-1}$	158
$\times 10^2$	$\times 10^1$	$\times 10^0$	$\times 10^{-1}$					
		Specifies the override value of the axis in operation. The override value is used while this setting is turned ON.						



Word	Bit	Description		Page				
n+5	X-axis Control Bits	00	Deceleration Stop	↑	Deceleration Stop	148		
					1		Prevents other manual commands.	
		01	Origin Search		↑	Starts the origin search.	149	
					1	Continues the origin search.		
					↓	Stops the origin search.		
		02	Reference Origin Return		↑	Starts moving the axis to the reference origin.	150	
					1	Continues movement.		
					↓	Stops movement.		
		03	Jogging		↑	Starts jog operation.	151	
					1	Continues jog operation.		
					↓	Stops jog operation.		
		04	Not used				---	
		05	Current Position Preset	↑	Presets the current position.	152		
		06	ABS Initialization	↑	Executes ABS Initialization	153		
		07	ABS Soft Reset	↑	Executes ABS Soft Reset	154		
		08	Enable MPG		↑	Starts MPG operation.	155	
					1	Continues MPG operation.		
					↓	Stops MPG operation.		
		09	Servo Lock	↑	Starts servo lock.	156		
		10	Servo Free	↑	Clears the servo lock status.	157		
11	Driver Alarm Reset		1	Turns ON the driver alarm reset output.	157			
			0	Turns OFF the driver alarm reset output.				
12	Override Setting		1	Enables the override setting.	158			
			0	Disables the override setting.				
13	Jog Direction		1	Specifies negative direction for jogging.	152			
			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)		156				
n+6	00 to 15	Y-axis override 0000 to 1999 (4-digit BCD) <table border="1" style="display: inline-table; vertical-align: middle;"> <tr> <td>x10<sup>2</sup></td> <td>x10<sup>1</sup></td> <td>x10<sup>0</sup></td> <td>x10<sup>-1</sup></td> </tr> </table> With 0.1% increments Specifies the override value of the axis in operation. The override value is used while this setting is turned ON.		x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>	x10 <sup>-1</sup>	158
x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>	x10 <sup>-1</sup>					

Word	Bit	Description		Page		
n+7	Y-axis Control Bits	00	Deceleration Stop	↑	Deceleration Stop	148
				1	Prevents other manual commands.	
		01	Origin Search	↑	Starts the origin search.	149
				1	Continues the origin search.	
				↓	Stops the origin search.	
		02	Reference Origin Return	↑	Starts moving the axis to the reference origin.	150
				1	Continues movement.	
				↓	Stops movement.	
		03	Jogging	↑	Starts jog operation.	151
				1	Continues jog operation.	
				↓	Stops jog operation.	
		04	Not used			---
		05	Current Position Preset	↑	Presets the current position.	152
		06	ABS Initialization	↑	Executes ABS Initialization	153
		07	ABS Soft Reset	↑	Executes ABS Soft Reset	154
		08	Not used			---
		09	Servo Lock	↑	Starts servo lock.	156
		10	Servo Free	↑	Clears the servo lock status.	157
		11	Driver Alarm Reset	1	Turns ON the driver alarm reset output.	157
				0	Turns OFF the driver alarm reset output.	
12	Override Setting	1	Enables the override setting.	158		
		0	Disables the override setting.			
13	Jog direction	1	Specifies negative direction for jogging.	152		
		0	Specifies positive direction for jogging.			
14	Not used			---		
15	Not used			---		

## Input

Word	Bit	Description	Page																																																						
n+8	00 to 04	<p>Error Type Data When an error results, an error output indicating the error type will be turned ON and will remain valid until the error is corrected.</p> <table border="1"> <thead> <tr> <th colspan="5">Bits</th> <th rowspan="2">Error</th> <th rowspan="2">Remarks</th> </tr> <tr> <th>04</th> <th>03</th> <th>02</th> <th>01</th> <th>00</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>Normal</td> <td>OFF while an error is resulting.</td> </tr> <tr> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>1</td> <td>System error</td> <td>ON while an error is resulting.</td> </tr> <tr> <td>0</td> <td>0</td> <td>0</td> <td>1</td> <td>0</td> <td>Task 1 error</td> <td></td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>0</td> <td>0</td> <td>Task 2 error</td> <td></td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>0</td> <td>0</td> <td>X-axis error</td> <td></td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>Y-axis error</td> <td></td> </tr> </tbody> </table>	Bits					Error	Remarks	04	03	02	01	00	0	0	0	0	0	Normal	OFF while an error is resulting.	0	0	0	0	1	System error	ON while an error is resulting.	0	0	0	1	0	Task 1 error		0	0	1	0	0	Task 2 error		0	1	0	0	0	X-axis error		1	0	0	0	0	Y-axis error		163
Bits					Error	Remarks																																																			
04	03	02	01	00																																																					
0	0	0	0	0	Normal	OFF while an error is resulting.																																																			
0	0	0	0	1	System error	ON while an error is resulting.																																																			
0	0	0	1	0	Task 1 error																																																				
0	0	1	0	0	Task 2 error																																																				
0	1	0	0	0	X-axis error																																																				
1	0	0	0	0	Y-axis error																																																				
System status	05	Not used	---																																																						
	06	Completion of Data Transmission	↑	Data has been transmitted.	159																																																				
			↓	No data is being transmitted.																																																					
	07	Data Transmission Error	↑	An error resulted during data transmission.	159																																																				
			↓	Data has been retransmitted normally.																																																					
	08	Completion of Data Reception	↑	Data has been received.	160																																																				
			↓	No data is being received.																																																					
	09	Data Reception Error	↑	An error resulted during data reception.	161																																																				
			↓	Retransmitted data has been received normally.																																																					
	10	Completion of Flash Memory Write	↑	Data has been written to the flash memory.	162																																																				
			↓	No data is being written to the flash memory.																																																					
	11	Teaching Box in Enable Mode	1	The Teaching Box is in enable mode.	162																																																				
			0	The Teaching Box is not in enable mode.																																																					
	12	Teaching Box in Protect Mode	1	The Teaching Box is in protect mode.	162																																																				
			0	The Teaching Box is not in protect mode.																																																					
13	Initial Setting Alarm	1	Initial setting is incorrect.	162																																																					
		0	Initial setting is correct.																																																						
14	Error	1	An error has resulted.	163																																																					
		0	No error has resulted.																																																						
15	Completion of Expansion Data Read	1	The expansion data area is being refreshed.	163																																																					
		0	The expansion data area is left without being refreshed.																																																						
n+9	00 to 15	<p>Error code (4-digit BCD)</p> <table border="1"> <tr> <td>x10<sup>3</sup></td> <td>x10<sup>2</sup></td> <td>x10<sup>1</sup></td> <td>x10<sup>0</sup></td> <td>Normal: 0000</td> </tr> </table> <p>The above is the error code format of the MC Unit. An error code is valid while an error bit is turned ON. If an error results, check the error type data to find the type of error, such as a system, task 1, task 2, X-axis, or Y-axis error.</p>	x10 <sup>3</sup>	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>	Normal: 0000	163																																																	
x10 <sup>3</sup>	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>	Normal: 0000																																																					
n+10	00 to 15	<p>Task 1 M code 0000 to 0999 (4-digit BCD)</p> <table border="1"> <tr> <td>0</td> <td>x10<sup>2</sup></td> <td>x10<sup>1</sup></td> <td>x10<sup>0</sup></td> </tr> </table> <p>The M code is output, which is valid when the M strobe is turned ON.</p>	0	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>	167																																																		
0	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>																																																						

Word	Bit	Description		Page				
n+11	Task 1 Status Flags	00	Task Error Flag	1	A task error occurred.	164		
					0		Normal	
		01	Not used			---		
		02	Automatic Mode Flag	1	The Unit is in automatic mode.	164		
				0	Mode other than the above.			
		03	Manual Mode Flag	1	The Unit is in manual mode.	165		
				0	Mode other than the above.			
		04	Memory Run Flag	1	The MC program is being executed.	165		
				0	Mode other than the above.			
		05	Memory Run Completed Flag	1	MC program execution has been completed. (G79 was executed.)	165		
				0	Mode other than the above.			
		06	Optional Input Standby	1	Waiting for optional input.	166		
				0	Mode other than the above.			
		07	M Code Reset Standby	1	Waiting for M code reset.	166		
				0	Mode other than the above.			
		08	M Strobe	↑	An M code (0 to 999) is being output.	167		
				↓	M code reset			
		09	Cycle Start Received	↑	The Cycle Start Signal was received.	167		
				↓	The Cycle Start Signal is OFF.			
		10	Not used			---		
11	Not used			---				
12	Teaching Address Setting Completed	↑	The teaching address has been set.	168				
		↓	Teaching address set command is OFF.					
13	Teaching Completed	↑	Teaching has been completed.	168				
		↓	Teaching command is OFF.					
14	Teaching Error	↑	A teaching error occurred.	169				
		↓	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.	169				
		↓	OFF when the next teaching command is completed normally or teaching address setting is completed.					
n+12	00 to 15	Task 2 M code 0000 to 0999 (4-digit BCD)		167				
		<table border="1" style="display: inline-table; vertical-align: middle;"> <tr> <td style="padding: 2px 10px;">0</td> <td style="padding: 2px 10px;"><math>\times 10^2</math></td> <td style="padding: 2px 10px;"><math>\times 10^1</math></td> <td style="padding: 2px 10px;"><math>\times 10^0</math></td> </tr> </table>		0	$\times 10^2$	$\times 10^1$	$\times 10^0$	
0	$\times 10^2$	$\times 10^1$	$\times 10^0$					
		The M code is output, which is valid when the M strobe is turned ON.						

Word	Bit	Description		Page		
n+13	Task 2 Status Flags	00	Task Error Flag	1	A task error occurred.	164
					0	
		01	Not used			---
		02	Automatic Mode Flag	1	The Unit is in automatic mode.	164
				0	Mode other than the above.	
		03	Manual Mode Flag	1	The Unit is in manual mode.	165
				0	Mode other than the above.	
		04	Memory Run Flag	1	The MC program is being executed.	165
				0	Mode other than the above.	
		05	Memory Run Completed Flag	1	MC program execution has been completed. (G79 was executed.)	165
				0	Mode other than the above.	
		06	Optional Input Standby	1	Waiting for optional input.	166
				0	Mode other than the above.	
		07	M Code Reset Standby	1	Waiting for M code reset.	166
				0	Mode other than the above.	
		08	M Strobe	↑	An M code (0 to 999) is being output.	167
				↓	M code reset	
		09	Cycle Start Received	↑	The Cycle Start Signal was received.	167
				↓	The Cycle Start Signal is OFF.	
		10	Not used			---
11	Not used			---		
12	Teaching Address Setting Completed	↑	The teaching address has been set.	168		
		↓	Teaching address set command is OFF.			
13	Teaching Completed	↑	Teaching has been completed.	168		
		↓	Teaching command is OFF.			
14	Teaching Error	↑	A teaching error occurred.	169		
		↓	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.	169		
		↓	OFF when the next teaching command is completed normally or teaching address setting is completed.			
n+14 n+15	00 to 15	X-axis current position $-39,999,999$ to $39,999,999$ (8-digit BCD) 		---		
		Remarks The current position of the reference coordinate system is output with a decimal point set with the system parameters.				

Word	Bit	Description		Page		
n+16	X-axis Status Flags	00	Reference Origin	1	Within the in-position range of the reference origin.	170
				0	Outside the in-position range of the reference origin.	
		01	Busy Flag	1	A manual command is being executed.	170
				0	Other than the above.	
		02	Servo Lock ON	1	Servo lock status	171
				0	Servo free status	
		03	No Origin Flag	1	No origin (The reference origin isn't fixed.)	171
				0	The reference origin is fixed.	
		04	Axis Operating	1	There is an axis movement command.	172
				0	There isn't an axis movement command.	
		05	Positioning Completed	1	Within the in-position range	173
				0	Outside the in-position range	
		06	Error Counter Alarm	1	The number of accumulated pulses in the error counter exceeds the error counter warning value.	173
				0	Other than the above.	
		07	Alarm Input	1	The driver alarm input is ON.	174
				0	The driver alarm input is OFF.	
		08	Zone 1 Flag	1	Within zone 1.	174
				0	Outside of zone 1.	
		09	Zone 2 Flag	1	Within zone 2.	174
				0	Outside of zone 2.	
		10	Zone 3 Flag	1	Within zone 3.	174
				0	Outside of zone 3.	
		11	Zone 4 Flag	1	Within zone 4.	174
				0	Outside of zone 4.	
		12	Zone 5 Flag	1	Within zone 5.	174
				0	Outside of zone 5.	
		13	Zone 6 Flag	1	Within zone 6.	174
				0	Outside of zone 6.	
14	Zone 7 Flag	1	Within zone 7.	174		
		0	Outside of zone 7.			
15	Zone 8 Flag	1	Within zone 8.	174		
		0	Outside of zone 8.			
n+17 n+18	00 to 15	Y-axis current position $-39,999,999$ to $39,999,999$ (8-digit BCD) 			---	
		Remarks The current position of the reference coordinate system is output with a decimal point set with the system parameters.				

Word	Bit		Description		Page	
n+19	Y-axis Status Flags	00	Reference Origin	1	Within the in-position range of the reference origin.	170
				0	Outside the in-position range of the reference origin.	
		01	Busy Flag	1	A manual command is being executed.	170
				0	Other than the above.	
		02	Servo Lock ON	1	Servo lock status	171
				0	Servo free status	
		03	No Origin Flag	1	No origin (The reference origin isn't fixed.)	171
				0	The reference origin is fixed.	
		04	Axis Operating	1	There is an axis movement command.	172
				0	There isn't an axis movement command.	
		05	Positioning Completed	1	Within the in-position range	173
				0	Outside the in-position range	
		06	Error Counter Alarm	1	The number of accumulated pulses in the error counter exceeds the error counter warning value.	173
				0	Other than the above.	
		07	Alarm Input	1	The driver alarm input is ON.	174
				0	The driver alarm input is OFF.	
		08	Zone 1 Flag	1	Within zone 1.	174
				0	Outside of zone 1.	
		09	Zone 2 Flag	1	Within zone 2.	174
				0	Outside of zone 2.	
		10	Zone 3 Flag	1	Within zone 3.	174
				0	Outside of zone 3.	
		11	Zone 4 Flag	1	Within zone 4.	174
				0	Outside of zone 4.	
		12	Zone 5 Flag	1	Within zone 5.	174
				0	Outside of zone 5.	
		13	Zone 6 Flag	1	Within zone 6.	174
				0	Outside of zone 6.	
14	Zone 7 Flag	1	Within zone 7.	174		
		0	Outside of zone 7.			
15	Zone 8 Flag/MPG valid	Zone 8 Flag	1	Within zone 8.	174	
			0	Outside of zone 8.		
		MPG in operation	1	MPG operation is being performed.		
			0	Other than the above.		

### 6-4-3 Expansion Data Area

Refer to the following for the expansion data area.

#### Expansion Data Area Allocation

Output

Word	Bit	Description	Page																														
I	00 to 15	No. of data transmission words (4-digit BCD) <div style="border: 1px solid black; padding: 2px; display: inline-block;"> <table style="border-collapse: collapse;"> <tr> <td style="border: 1px solid black; padding: 2px;">x10<sup>3</sup></td> <td style="border: 1px solid black; padding: 2px;">x10<sup>2</sup></td> <td style="border: 1px solid black; padding: 2px;">x10<sup>1</sup></td> <td style="border: 1px solid black; padding: 2px;">x10<sup>0</sup></td> </tr> </table> </div>	x10 <sup>3</sup>	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>	144																										
x10 <sup>3</sup>	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>																														
I+1	00 to 15	Transmission data source word (4-digit BCD) <div style="border: 1px solid black; padding: 2px; display: inline-block;"> <table style="border-collapse: collapse;"> <tr> <td style="border: 1px solid black; padding: 2px;">x10<sup>3</sup></td> <td style="border: 1px solid black; padding: 2px;">x10<sup>2</sup></td> <td style="border: 1px solid black; padding: 2px;">x10<sup>1</sup></td> <td style="border: 1px solid black; padding: 2px;">x10<sup>0</sup></td> </tr> </table> </div>	x10 <sup>3</sup>	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>																											
x10 <sup>3</sup>	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>																														
I+2	00 to 15	Transmission data destination address (4-digit BCD) <div style="border: 1px solid black; padding: 2px; display: inline-block;"> <table style="border-collapse: collapse;"> <tr> <td style="border: 1px solid black; padding: 2px;">x10<sup>3</sup></td> <td style="border: 1px solid black; padding: 2px;">x10<sup>2</sup></td> <td style="border: 1px solid black; padding: 2px;">x10<sup>1</sup></td> <td style="border: 1px solid black; padding: 2px;">x10<sup>0</sup></td> </tr> </table> </div>	x10 <sup>3</sup>	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>																											
x10 <sup>3</sup>	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>																														
I+3	00 to 15	No. of data reception words (4-digit BCD) <div style="border: 1px solid black; padding: 2px; display: inline-block;"> <table style="border-collapse: collapse;"> <tr> <td style="border: 1px solid black; padding: 2px;">x10<sup>3</sup></td> <td style="border: 1px solid black; padding: 2px;">x10<sup>2</sup></td> <td style="border: 1px solid black; padding: 2px;">x10<sup>1</sup></td> <td style="border: 1px solid black; padding: 2px;">x10<sup>0</sup></td> </tr> </table> </div>	x10 <sup>3</sup>	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>	145																										
x10 <sup>3</sup>	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>																														
I+4	00 to 15	Reception data source address (4-digit BCD) <div style="border: 1px solid black; padding: 2px; display: inline-block;"> <table style="border-collapse: collapse;"> <tr> <td style="border: 1px solid black; padding: 2px;">x10<sup>3</sup></td> <td style="border: 1px solid black; padding: 2px;">x10<sup>2</sup></td> <td style="border: 1px solid black; padding: 2px;">x10<sup>1</sup></td> <td style="border: 1px solid black; padding: 2px;">x10<sup>0</sup></td> </tr> </table> </div>	x10 <sup>3</sup>	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>																											
x10 <sup>3</sup>	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>																														
I+5	00 to 15	Reception data destination word (4-digit BCD) <div style="border: 1px solid black; padding: 2px; display: inline-block;"> <table style="border-collapse: collapse;"> <tr> <td style="border: 1px solid black; padding: 2px;">x10<sup>3</sup></td> <td style="border: 1px solid black; padding: 2px;">x10<sup>2</sup></td> <td style="border: 1px solid black; padding: 2px;">x10<sup>1</sup></td> <td style="border: 1px solid black; padding: 2px;">x10<sup>0</sup></td> </tr> </table> </div>	x10 <sup>3</sup>	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>																											
x10 <sup>3</sup>	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>																														
I+6 I+7	00 to 15	<p>X-axis current position preset value (8-digit BCD)</p> <div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center; width: 50%;">I+7</td> <td style="text-align: center; width: 50%;">I+6</td> </tr> <tr> <td style="text-align: center;">x10<sup>7</sup> x10<sup>6</sup> x10<sup>5</sup> x10<sup>4</sup></td> <td style="text-align: center;">x10<sup>3</sup> x10<sup>2</sup> x10<sup>1</sup> x10<sup>0</sup></td> </tr> </table> </div> <div style="margin-bottom: 10px;"> <table style="margin-left: 20px;"> <tr> <td style="text-align: center;">3 2 1 0 bit</td> </tr> <tr> <td style="text-align: center;">x10<sup>7</sup></td> </tr> </table> </div> <p>Sign bit 0: Positive 1: Negative</p> <p>When using the current position preset function at bit 05 of word n+5, specify the preset value in this area. The data to be set depends on the minimum unit set with the MC Support Software. In the following example, the X axis is set to 100.</p> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">Minimum unit</th> <th style="text-align: left;">I+7</th> <th style="text-align: left;">I+6</th> <th style="text-align: left;">Remarks</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>0000</td> <td>0100</td> <td>Indicates 100</td> </tr> <tr> <td>0.1</td> <td>0000</td> <td>1000</td> <td>Indicates 100.0</td> </tr> <tr> <td>0.01</td> <td>0001</td> <td>0000</td> <td>Indicates 100.00</td> </tr> <tr> <td>0.001</td> <td>0010</td> <td>0000</td> <td>Indicates 100.000</td> </tr> <tr> <td>0.0001</td> <td>0100</td> <td>0000</td> <td>Indicates 100.0000</td> </tr> </tbody> </table>	I+7	I+6	x10 <sup>7</sup> x10 <sup>6</sup> x10 <sup>5</sup> x10 <sup>4</sup>	x10 <sup>3</sup> x10 <sup>2</sup> x10 <sup>1</sup> x10 <sup>0</sup>	3 2 1 0 bit	x10 <sup>7</sup>	Minimum unit	I+7	I+6	Remarks	1	0000	0100	Indicates 100	0.1	0000	1000	Indicates 100.0	0.01	0001	0000	Indicates 100.00	0.001	0010	0000	Indicates 100.000	0.0001	0100	0000	Indicates 100.0000	152
I+7	I+6																																
x10 <sup>7</sup> x10 <sup>6</sup> x10 <sup>5</sup> x10 <sup>4</sup>	x10 <sup>3</sup> x10 <sup>2</sup> x10 <sup>1</sup> x10 <sup>0</sup>																																
3 2 1 0 bit																																	
x10 <sup>7</sup>																																	
Minimum unit	I+7	I+6	Remarks																														
1	0000	0100	Indicates 100																														
0.1	0000	1000	Indicates 100.0																														
0.01	0001	0000	Indicates 100.00																														
0.001	0010	0000	Indicates 100.000																														
0.0001	0100	0000	Indicates 100.0000																														
I+8 I+9	00 to 15	<p>Y-axis current position preset value (8-digit BCD)</p> <div style="border: 1px solid black; padding: 5px; margin-bottom: 10px;"> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center; width: 50%;">I+9</td> <td style="text-align: center; width: 50%;">I+8</td> </tr> <tr> <td style="text-align: center;">x10<sup>7</sup> x10<sup>6</sup> x10<sup>5</sup> x10<sup>4</sup></td> <td style="text-align: center;">x10<sup>3</sup> x10<sup>2</sup> x10<sup>1</sup> x10<sup>0</sup></td> </tr> </table> </div> <div style="margin-bottom: 10px;"> <table style="margin-left: 20px;"> <tr> <td style="text-align: center;">3 2 1 0 bit</td> </tr> <tr> <td style="text-align: center;">x10<sup>7</sup></td> </tr> </table> </div> <p>Sign bit 0: Positive 1: Negative</p> <p>When using the current position preset function at bit 05 of word n+7, specify the preset value in this area. Refer to the X-axis current position preset value.</p>	I+9	I+8	x10 <sup>7</sup> x10 <sup>6</sup> x10 <sup>5</sup> x10 <sup>4</sup>	x10 <sup>3</sup> x10 <sup>2</sup> x10 <sup>1</sup> x10 <sup>0</sup>	3 2 1 0 bit	x10 <sup>7</sup>	152																								
I+9	I+8																																
x10 <sup>7</sup> x10 <sup>6</sup> x10 <sup>5</sup> x10 <sup>4</sup>	x10 <sup>3</sup> x10 <sup>2</sup> x10 <sup>1</sup> x10 <sup>0</sup>																																
3 2 1 0 bit																																	
x10 <sup>7</sup>																																	



Word	Bit	Description	Page
I+10	00 to 15	Task 1 teaching start address 0000 to 1999 (4-digit BCD) <div style="border: 1px solid black; display: inline-block; padding: 2px;"> <math>x10^3</math> <math>x10^2</math> <math>x10^1</math> <math>x10^0</math> </div> When using the teaching address set function at bit 08 of word n+1, specify the teaching start address in this area.	142
I+11	00 to 15	Task 2 teaching start address 0000 to 1999 (4-digit BCD) <div style="border: 1px solid black; display: inline-block; padding: 2px;"> <math>x10^3</math> <math>x10^2</math> <math>x10^1</math> <math>x10^0</math> </div> When using the teaching address set function at bit 08 of word n+3, specify the teaching start address in this area.	142
I+12	00 to 15	System error code (4-digit BCD) <div style="border: 1px solid black; display: inline-block; padding: 2px;"> <math>x10^3</math> <math>x10^2</math> <math>x10^1</math> <math>x10^0</math> </div> Normal: 0000 When a system error results, the corresponding error code will be output.	---
I+13	00 to 15	Task 1 error code (4-digit BCD) <div style="border: 1px solid black; display: inline-block; padding: 2px;"> <math>x10^3</math> <math>x10^2</math> <math>x10^1</math> <math>x10^0</math> </div> Normal: 0000 When a task 1 error results, the corresponding error code will be output.	---
I+14	00 to 15	Task 2 error code (4-digit BCD) <div style="border: 1px solid black; display: inline-block; padding: 2px;"> <math>x10^3</math> <math>x10^2</math> <math>x10^1</math> <math>x10^0</math> </div> Normal: 0000 When a task 2 error results, the corresponding error code will be output.	---
I+15	00 to 15	X-axis error code (4-digit BCD) <div style="border: 1px solid black; display: inline-block; padding: 2px;"> <math>x10^3</math> <math>x10^2</math> <math>x10^1</math> <math>x10^0</math> </div> Normal: 0000 When an X-axis error results, the corresponding error code will be output.	---
I+16	00 to 15	Y-axis error code (4-digit BCD) <div style="border: 1px solid black; display: inline-block; padding: 2px;"> <math>x10^3</math> <math>x10^2</math> <math>x10^1</math> <math>x10^0</math> </div> Normal: 0000 When a Y-axis error results, the corresponding error code will be output.	---
I+17	00 to 15	Task 1 execution program no. 0000 to 0999 (4-digit BCD) <div style="border: 1px solid black; display: inline-block; padding: 2px;">           0 <math>x10^2</math> <math>x10^1</math> <math>x10^0</math> </div> The number of the program being executed or on hold with task 1 will be output.	---
I+18	00 to 15	Task 1 execution block no. 0000 to 0999 (4-digit BCD) <div style="border: 1px solid black; display: inline-block; padding: 2px;">           0 <math>x10^2</math> <math>x10^1</math> <math>x10^0</math> </div> The number of the block being executed or on hold with task 1 will be output.	---

Word	Bit	Description	Page				
I+19	00 to 15	Task 1 teaching execution address 0000 to 1999 (4-digit BCD) <table border="1" style="margin-left: 40px;"> <tr> <td>x10<sup>3</sup></td> <td>x10<sup>2</sup></td> <td>x10<sup>1</sup></td> <td>x10<sup>0</sup></td> </tr> </table> When the teaching address function at bit 09 of word n+1 is executed with task 1, the first address of the position data will be output. This address is refreshed whenever teaching is executed.	x10 <sup>3</sup>	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>	142
x10 <sup>3</sup>	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>				
I+20	00 to 15	Task 2 execution program no. 0000 to 0999 (4-digit BCD) <table border="1" style="margin-left: 40px;"> <tr> <td>0</td> <td>x10<sup>2</sup></td> <td>x10<sup>1</sup></td> <td>x10<sup>0</sup></td> </tr> </table> The number of the program being executed or on hold with task 2 will be output.	0	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>	---
0	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>				
I+21	00 to 15	Task 2 execution block no. 0000 to 0999 (4-digit BCD) <table border="1" style="margin-left: 40px;"> <tr> <td>0</td> <td>x10<sup>2</sup></td> <td>x10<sup>1</sup></td> <td>x10<sup>0</sup></td> </tr> </table> The number of the block being executed or on hold with task 2 will be output.	0	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>	---
0	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>				
I+22	00 to 15	Task 2 teaching execution address 0000 to 1999 (4-digit BCD) <table border="1" style="margin-left: 40px;"> <tr> <td>x10<sup>3</sup></td> <td>x10<sup>2</sup></td> <td>x10<sup>1</sup></td> <td>x10<sup>0</sup></td> </tr> </table> When the teaching address function at bit 09 of word n+3 is executed with task 2, the first address of the position data will be output. This address is refreshed whenever teaching is executed.	x10 <sup>3</sup>	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>	142
x10 <sup>3</sup>	x10 <sup>2</sup>	x10 <sup>1</sup>	x10 <sup>0</sup>				

## 6-5 Interface Bit Specifics

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

### 6-5-1 Word n+1 Bit 01: Automatic/Manual Mode Bit (Task 1)

Bits 00 to 10 of word n+3 are the control bits of task 2. The control bits of task 1 are explained below. There is no difference in function between the control bits of tasks 1 and 2 although their words are different.

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.

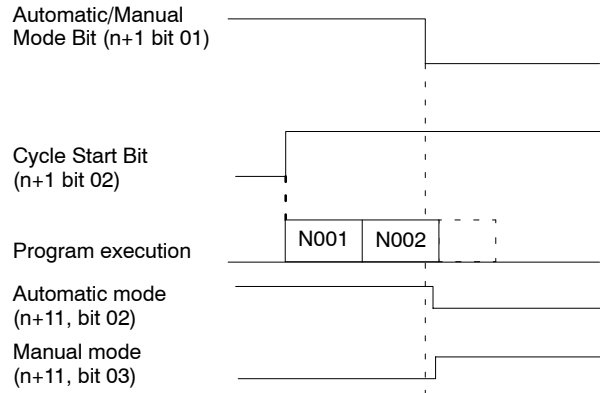
When the mode is switched from manual to automatic, any operating axes will decelerate to a stop. When the mode is switched from automatic to manual, the program will be stopped if it is operating. The program will be restarted from the block where it was stopped if the mode is switched back to automatic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

**Signal**

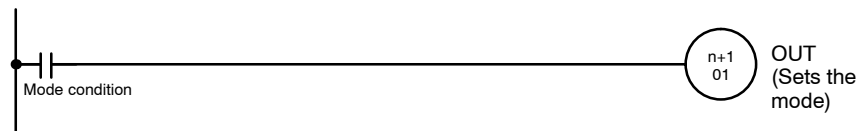
The automatic/manual mode bit settings have the following functions.

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**



**6-5-2 Word n+1 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 task 2 (bit 02 in word n+3) operate in the same way.

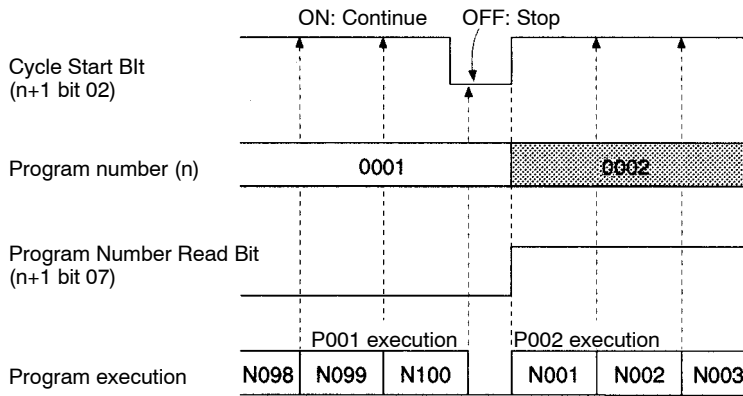
**Signal**

The Cycle Start Bit settings have the following functions.

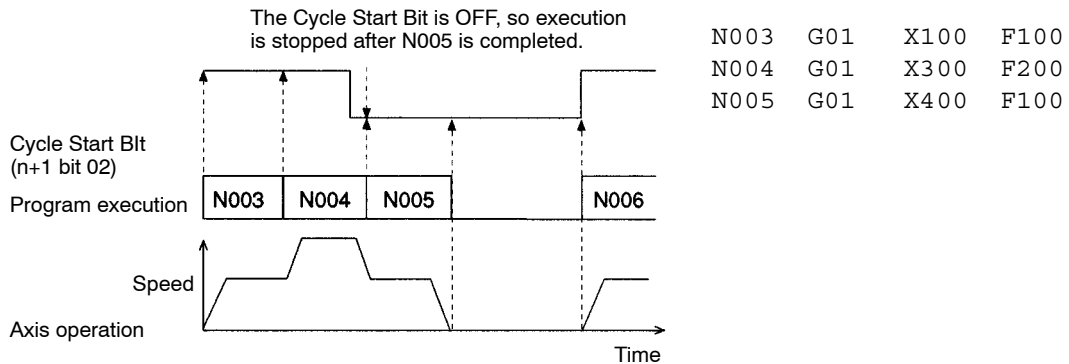
Signal	Function
↑ (OFF-to-ON)	Starts MC program execution.
ON	Continues execution. (Referenced at the start of each block.)
OFF	Stops execution. (Referenced at the start of each block.)

Timing Chart

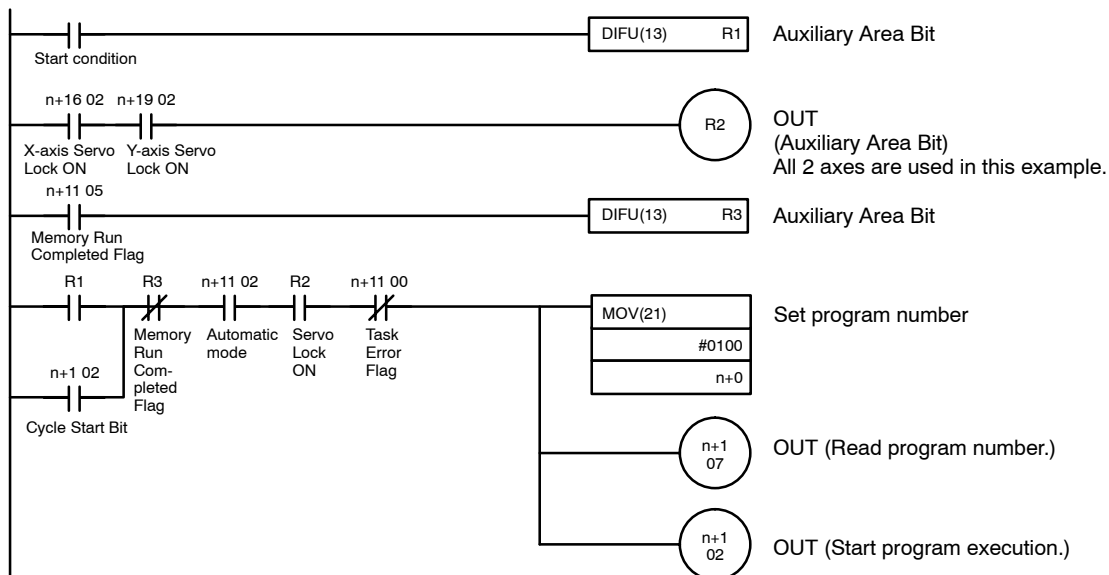
Normal Operation:



Pass Operation:



Program Example



**Note** 1. 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.

2. Programs can be continued even after the following operations.
- Before continuing the program operation, be sure to confirm that existing conditions will not cause any problem. If continuous program operation causes a problem, do not continue running the program; instead create a ladder program will execute the program from the beginning.
- Program operation running under the automatic mode was stopped intentionally or came to a stop by itself.  
Examples: Cycle start was turned OFF.  
An error occurred.
  - Manual command was executed under the manual mode.  
Examples: Origin search was executed.  
Jog was executed.  
Servo lock was executed.
  - Reset to the automatic mode and cycle start turned ON (for executing continuous operation).

For example, program operation was temporarily suspended while moving to X10000 and has stopped at the X500 position.

Reset to manual mode and execute the origin search. After the origin search is complete, reset to the automatic mode to continue program operation. The program will continue the move operation to the X10000 position.

### 6-5-3 Word n+1 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.

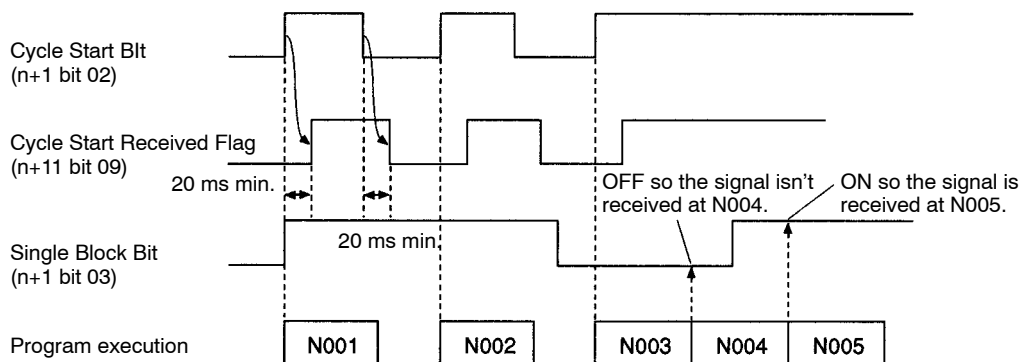
When the Single Block Bit is turned ON during Pass operation, the next block's positioning operation will be completed before program execution is paused.

The Single Block Bits for task 2 (bit 03 in word n+3) operate in the same way.

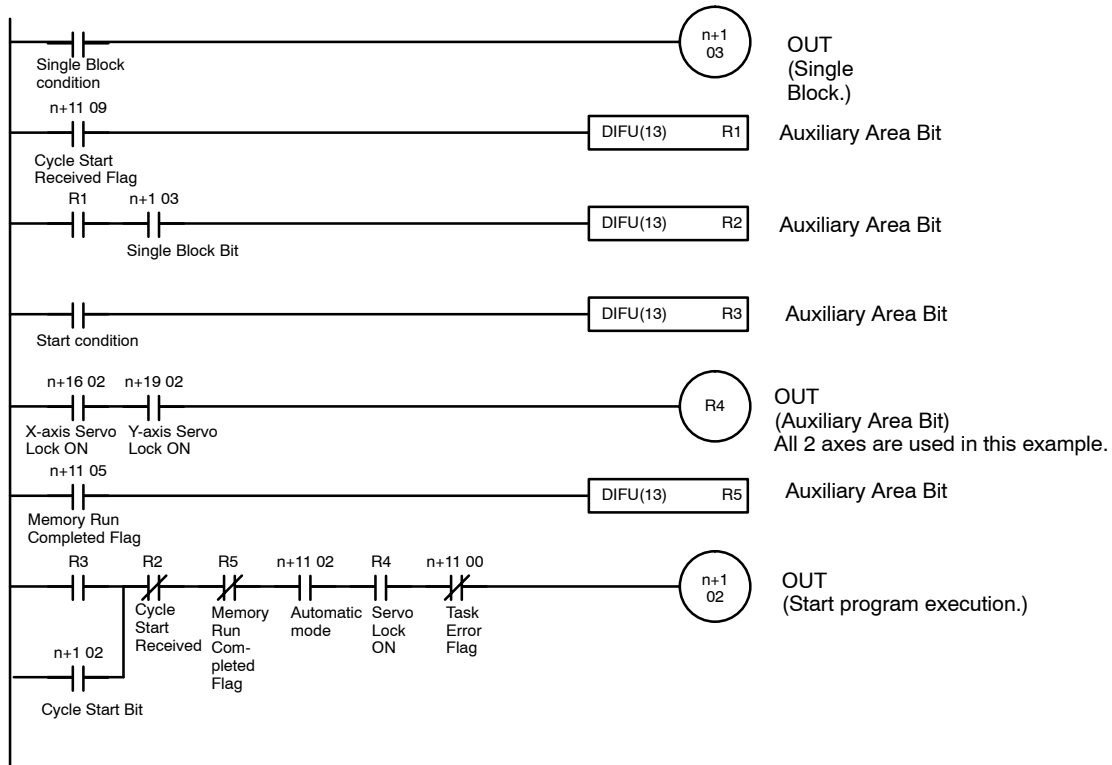
#### Signal

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.

6-5-4 Word n+1 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+1 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.

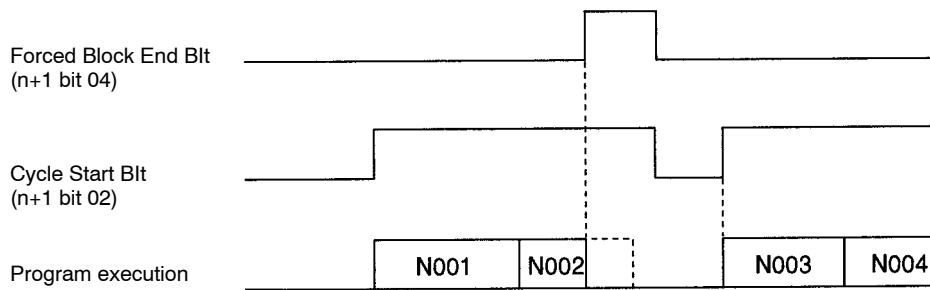
The Forced Block End Bits for task 2 (bit 04 in word n+3) operate in the same way.

Signal

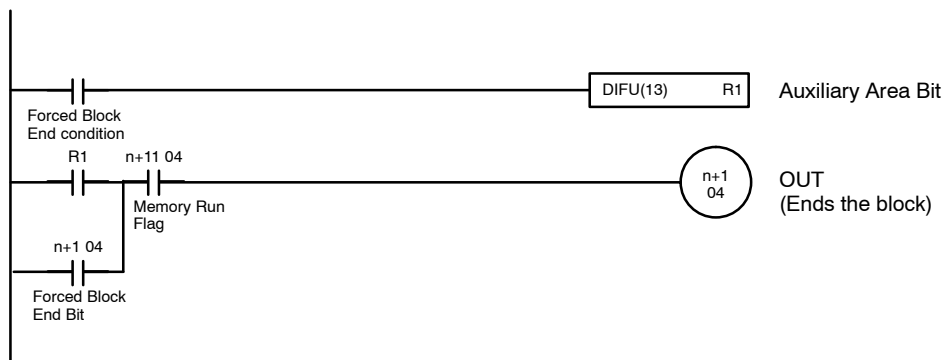
The Forced Block End Bit settings have the following functions.

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



6-5-5 Word n+1 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+1 bit 04). If both of these bits go ON at the same time, the forced block end command will be executed.

Keep this bit ON until the memory operation is turned OFF.

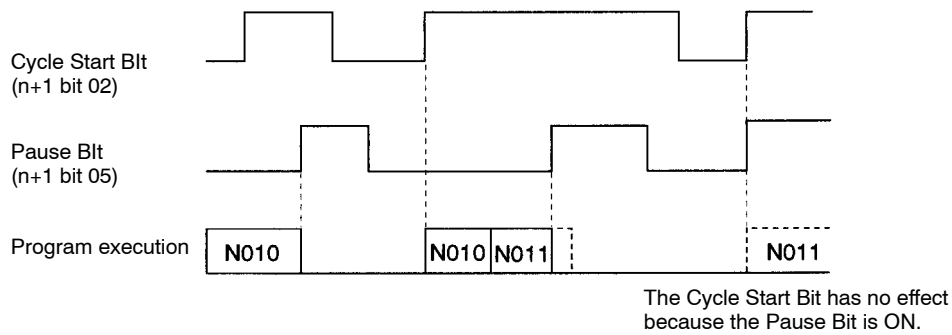
The Pause Bits for tasks 2 (bit 05 in word n+3) operate in the same way.

Signal

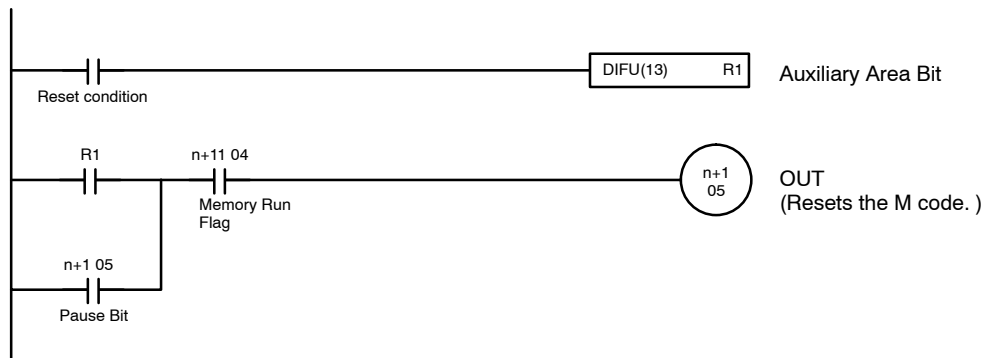
The Pause Bit settings have the following functions.

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



Program Example



6-5-6 Word n+1 Bit 06: M Code Reset Bit (Task 1)

This bit resets the M code output.

If M code 0 to 499 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.

The M Code Reset Bits for task 2 (bit 06 in word n+3) operate in the same way.

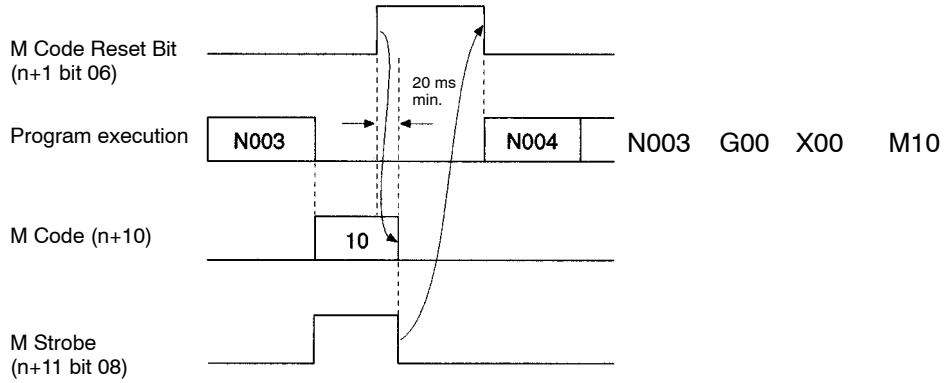
Signal

The M Code Reset Bit settings have the following functions.

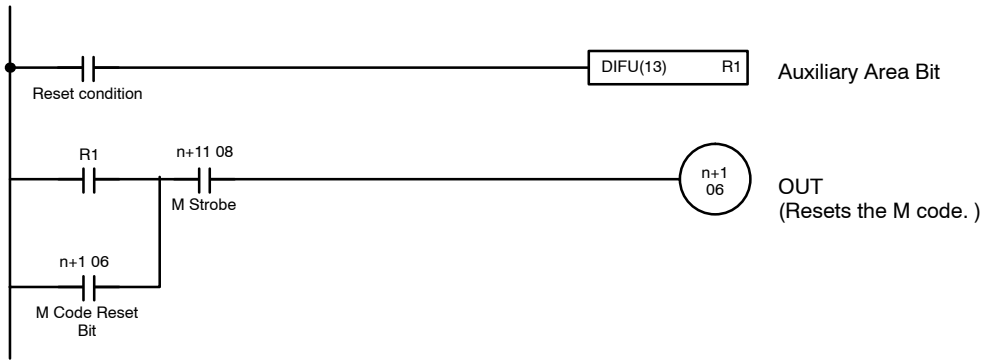
Signal	Function
↑ (OFF-to-ON)	If there is an M code reset standby (M000 to M499), the M code will be reset. M codes M500 to M999 will be reset.
ON	M code reset standby (when there is an M code reset standby) Prevents M code output.
↓ (ON-to-OFF)	Clears M code reset standby.



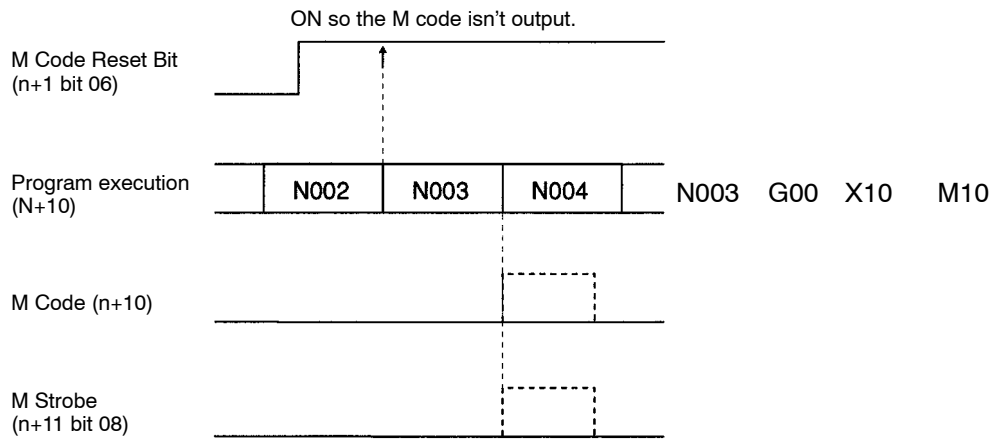
Timing Chart 1



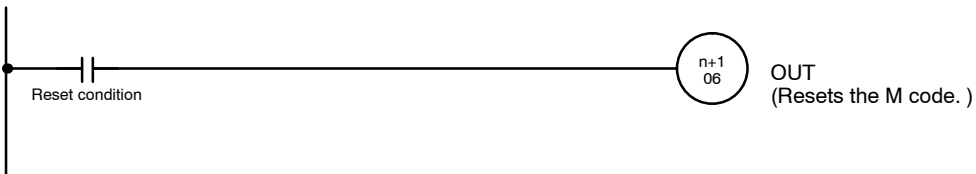
Program Example 1



Timing Chart 2



Program Example 2



### 6-5-7 Word n+1 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 word n 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 word n.

**Note** Refer to the Cycle Start Bit (n+1 bit 02).

### 6-5-8 Word n+1 Bit 08: 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 task 2 (bit 08 in word n+3) operate in the same way.

At the rising edge of this bit, the teaching start address set in the expansion data area will be set as the first address for teaching.

This bit is available when the expansion data area is valid. 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.

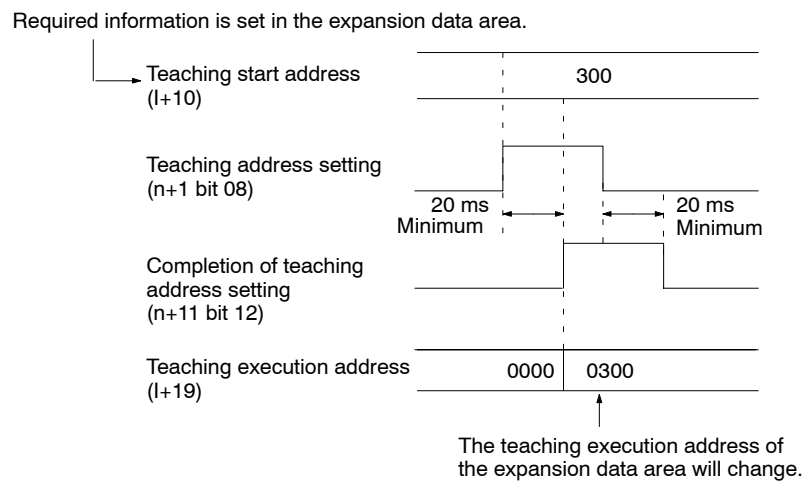
**Signal**

The Teaching Address Set Bit settings have the following functions.

Signal	Function
↑ (OFF-to-ON)	The teaching start address set in the expansion data area is transferred to the MC Unit and set as a teaching execution address.

**Timing Chart**

Example: The teaching start address of task 1 is changed to 300.



### 6-5-9 Word n+1 Bit 09: Teaching Bit (Task 1)

This bit is used to read the reference coordinate system position into the position data address. The Teaching Bits for task 2 (bit 09 in word n+3) 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 position data storage area in the MC Unit when the Teaching Bit goes from OFF to ON (up-differentiation).

**Note** Refer to the Teaching Type Bit (n+1 bit 10).

**6-5-10 Word n+1 Bit 10: 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.

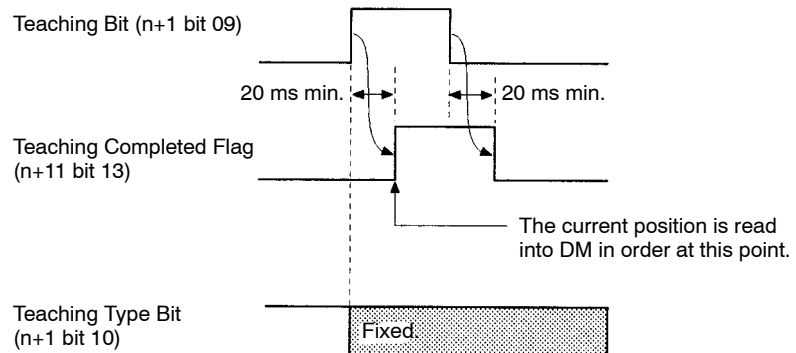
The Teaching Type Bits for task 2 (bit 10 in word n+3) operate in the same way.

**Signal**

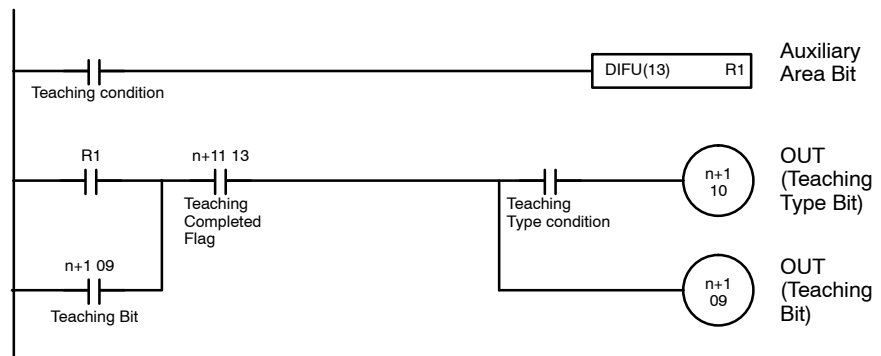
The Teaching Type Bit settings have the following functions.

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

**Timing Chart**

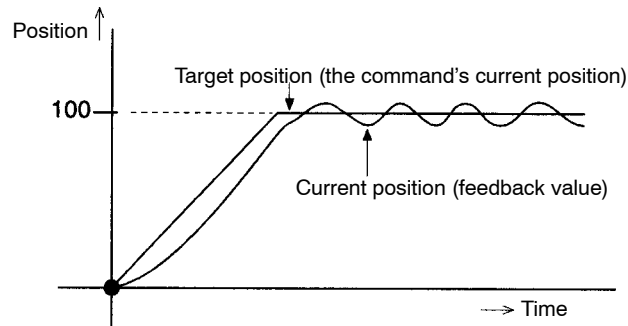


**Program Example**



**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.



**6-5-11 Word n+1 Bit 11: Transmission Control Bit (All Tasks)**

This bit is used to write data to the MC Unit.

The completion of data writing to the MC Unit can be confirmed with the Transmission Completed Flag.

This bit must be on hold until the Transmission Completed Flag is turned ON.

This bit is available when the expansion data area is valid.

Data cannot be written to the MC Unit, read from the MC Unit, and written to the flash memory at the same time. If they are instructed together, the instructions are executed in the following order.

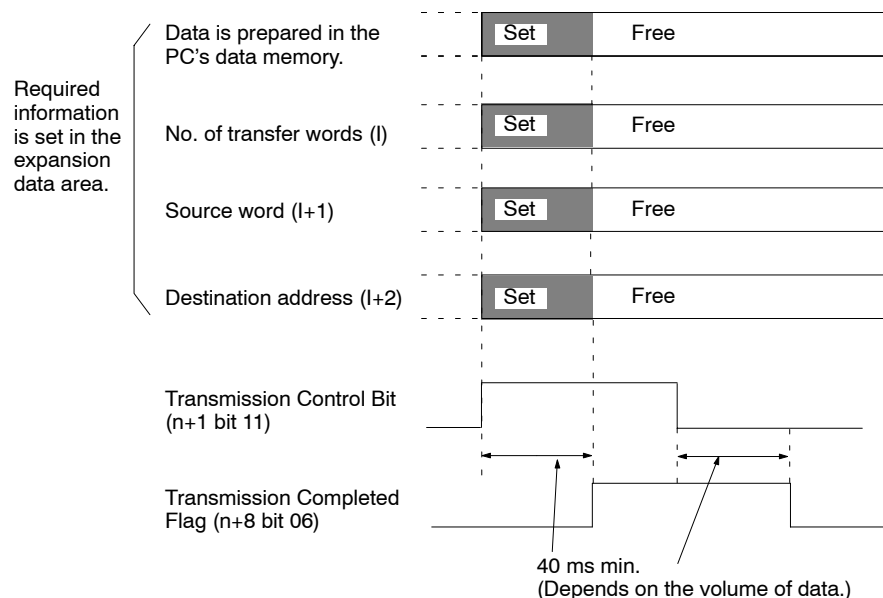
Priority: Data written to the MC Unit, data read from the MC Unit, and data written to the flash memory.

**Signal**

The data transmission settings have the following functions.

Signal	Function
↑	Writes Data

**Timing Chart**



**Note** Refer to Section 4 MC Unit Data.

### 6-5-12 Word n+1 Bit 12: Reception Control Bit (All Tasks)

This bit is used to read data from the MC Unit.

The completion of data reading from the MC Unit can be confirmed with the Reception Completed Flag.

This bit must be on hold until the Reception Completed Flag is turned ON.

This bit is available when the expansion data area is valid.

Data cannot be written to the MC Unit, read from the MC Unit, and written to the flash memory at the same time. If they are instructed together, the instructions are executed in the following order.

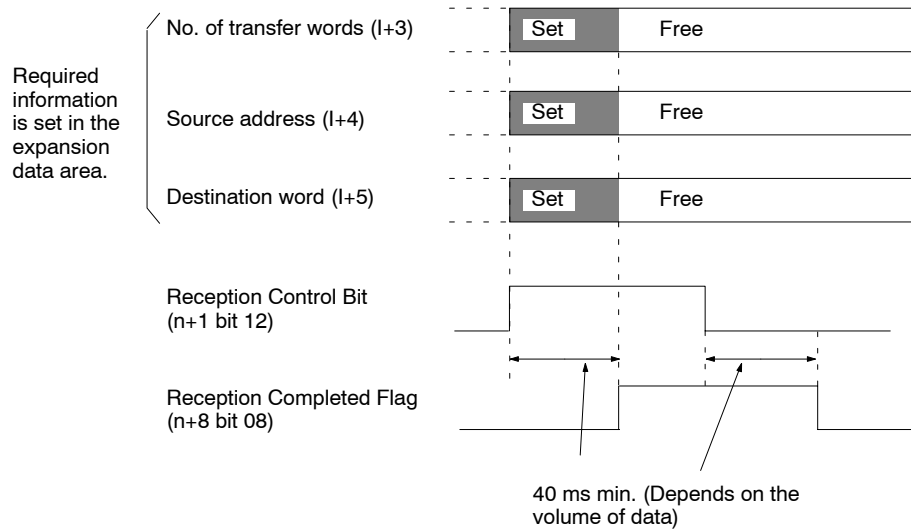
Priority: Data written to the MC Unit, data read from the MC Unit, and data written to the flash memory.

**Signal**

The Reception Control Bit settings have the following functions.

Signal	Function
↑	Reads Data

**Timing Chart**



**Note** Refer to *Section 4 MC Unit Data*.

### 6-5-13 Word n+1 Bit 13: Flash Memory Write Bit (All Tasks)

The position data of the MC Unit will be stored in the flash memory when this bit is turned ON.

This bit must be on hold until the Flash Memory Write Completed Flag is turned ON.

Data cannot be written to the MC Unit, read from the MC Unit, and written to the flash memory at the same time. If they are instructed together, the instructions are executed in the following order.

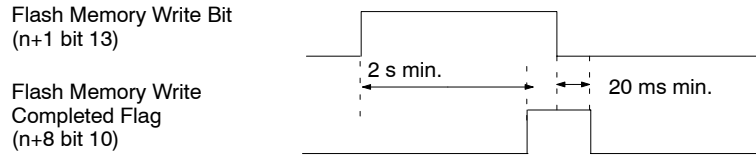
Priority: Data written to the MC Unit, data read from the MC Unit, and data written to the flash memory.

**Signal**

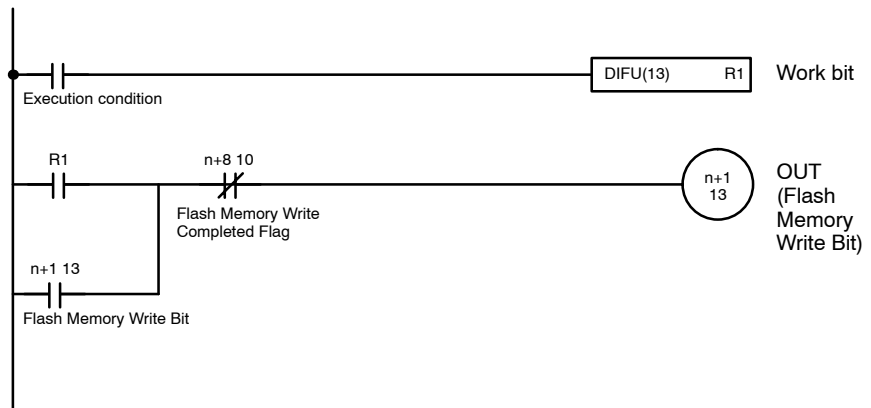
The Flash Memory Write settings have the following functions.

Signal	Function
↑	Writes Data to Flash Memory

Timing Chart



Program Example



6-5-14 Word n+1 Bit 14: Error Reset Bit (All Tasks)

The error of the MC Unit will be reset when this bit is turned ON.

This bit must be on hold until the Error Flag is turned OFF.

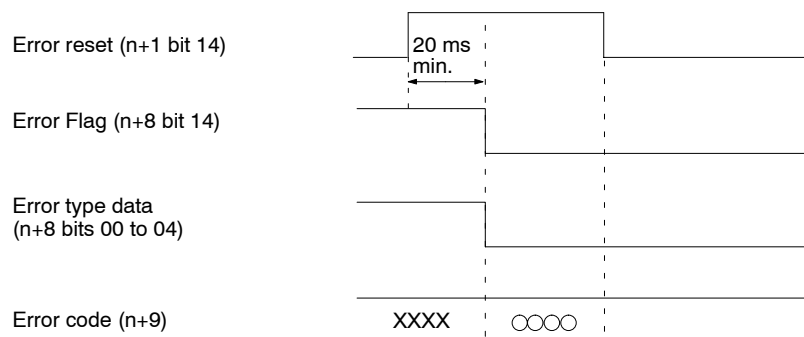
When the Error Flag is turned OFF, error type data will be turned OFF as well.

Signal

The Error Reset settings have the following functions.

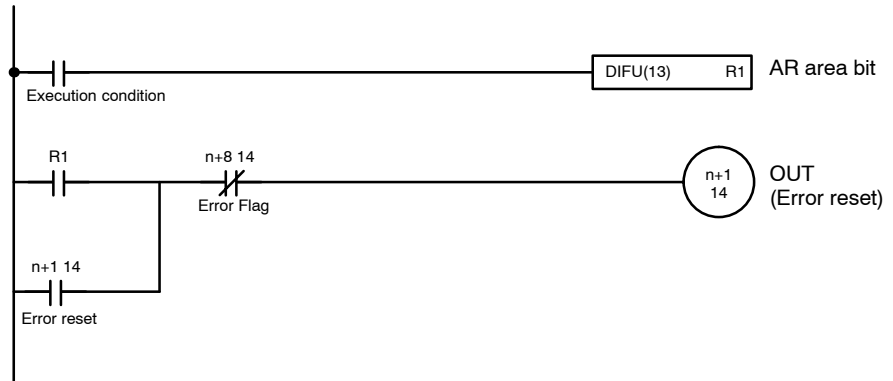
Signal	Function
↑	Resets MC Unit error.

Timing Chart



Another error during this period will not turn the bit ON until the Error Reset Bit is turned OFF.

Program Example



6-5-15 Word n+1 Bit 15: Expansion Data Read Bit (All Tasks)

The IN refresh of the expansion data area will be refreshed when this bit and Expansion Data Read Completed Flag are ON.

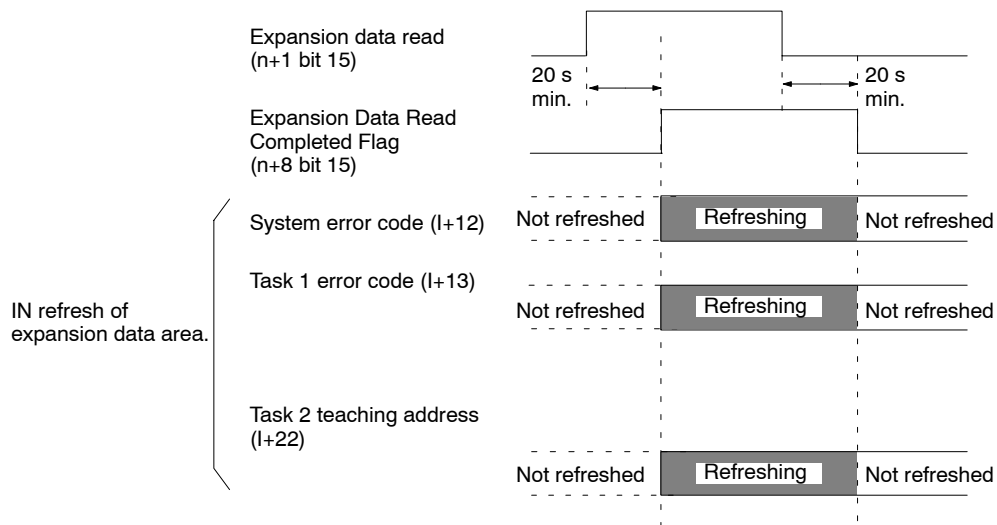
The IN refresh of the expansion data area will not be refreshed when this bit is turned OFF.

Signal

The Expansion Data Read settings have the following functions.

Signal	Function
0	Keeps present expansion data.
1	Refreshes expansion data.

Timing Chart



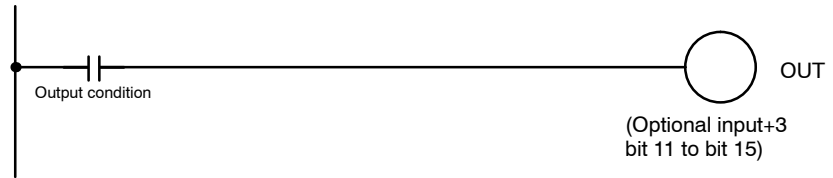
6-5-16 Word n+3 Bits 11 to 15: Optional Inputs

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 5 optional inputs available (bits 00 to 04).

Signal

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

Program Example



6-5-17 Word n+5 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.

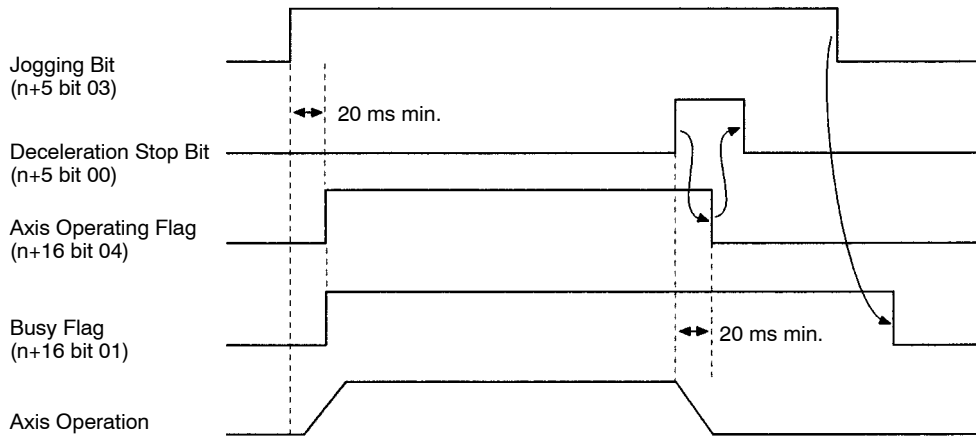
The Deceleration Stop Bits for the Y axes (bit 00 in word n+7) operate in the same way.

Signal

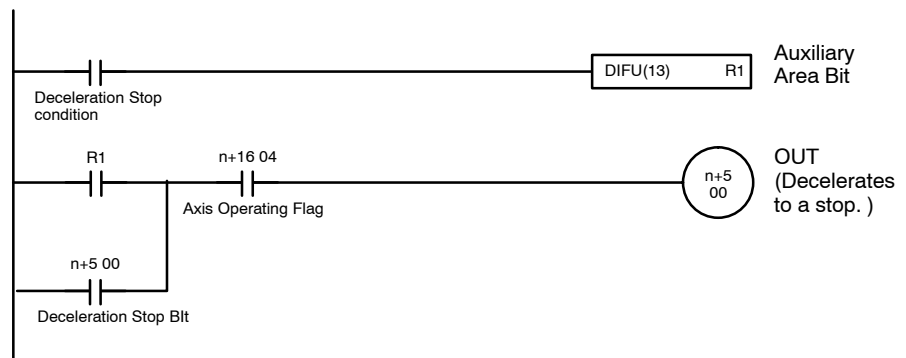
The Deceleration Stop Bit settings have the following functions.

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

Timing Chart



Program Example





### 6-5-18 Word n+5 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+16 bit 01) 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+16 bit 03) 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.

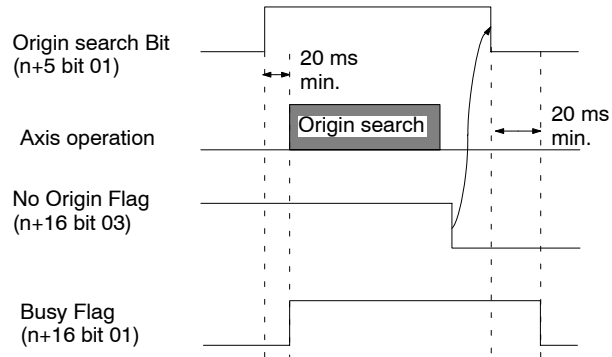
An Origin Return Operation will be performed for axes set for absolute operation. The Origin Search Bits for the Y axes (bit 01 in word n+7) operate in the same way.

**Signal**

The Origin Search Bit settings have the following functions.

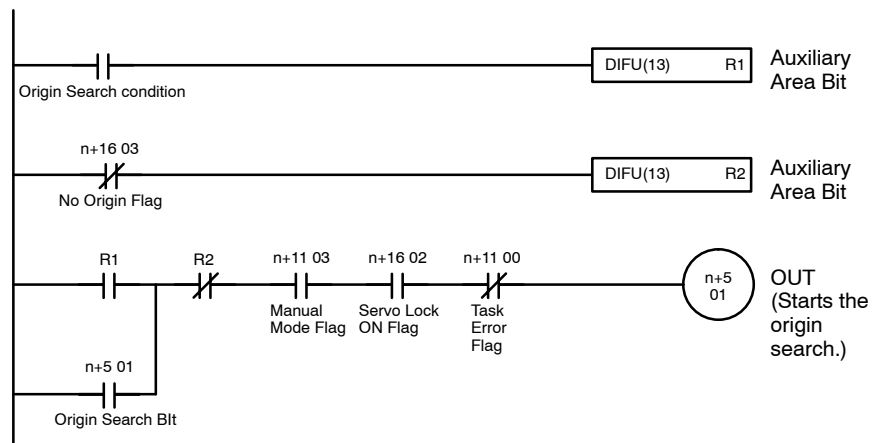
Signal	Function
↑ (OFF-to-ON)	Starts the origin search.
↓ (ON-to-OFF)	Stops the origin search if it is in progress.

**Timing Chart**



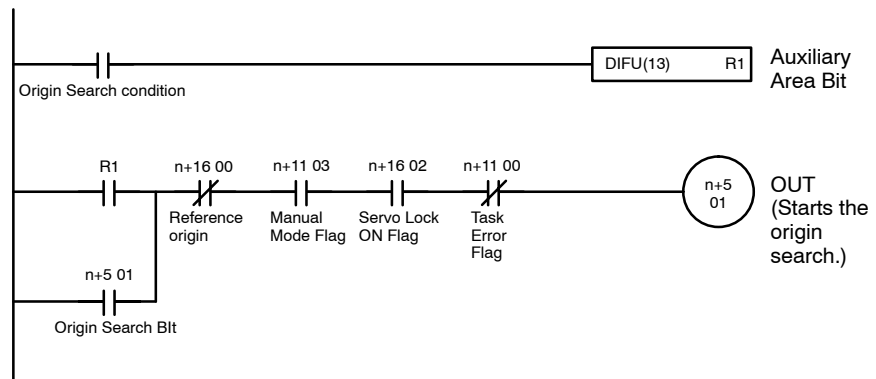
**Program Example 1**

The following example is for incremental operation.



Program Example 2

The following example is for absolute operation.



6-5-19 Word n+5 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 using PTP control. The positioning will begin 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.

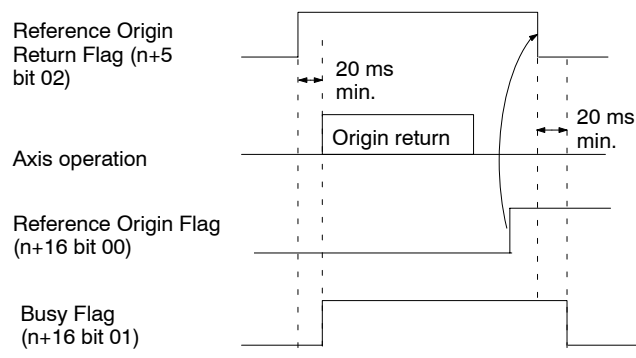
The Reference Origin Return Bits for the Y axes (bit 02 in word n+7) operate in the same way.

Signal

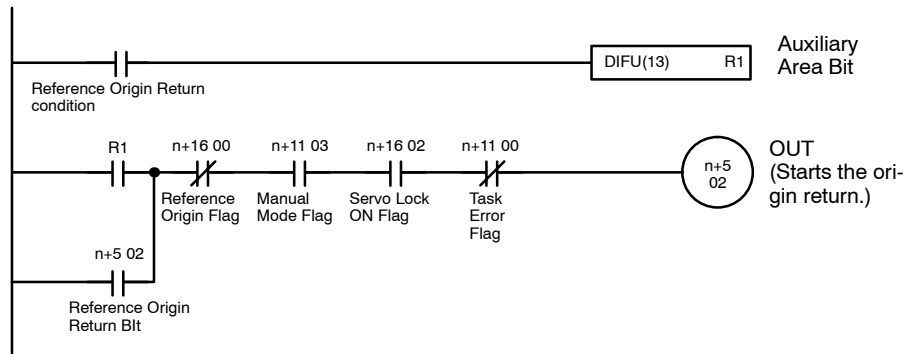
The Reference Origin Return Bit settings have the following functions.

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

Timing Chart



Program Example



6-5-20 Word n+5 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 Bit for the Y axis (bit 03 in word n+7) operate in the same way.

- When the mechanical origin has been fixed:
  - a) Jogging can be performed within the software limits.
  - b) The axis will consume accumulated pulses and stop within the software limits.
- When the mechanical origin hasn't been fixed, jogging can be performed without any restrictions.

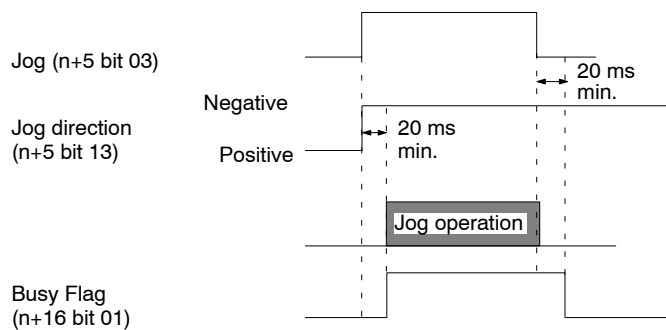
This bit is ignored if the Busy Flag is ON.

Signal

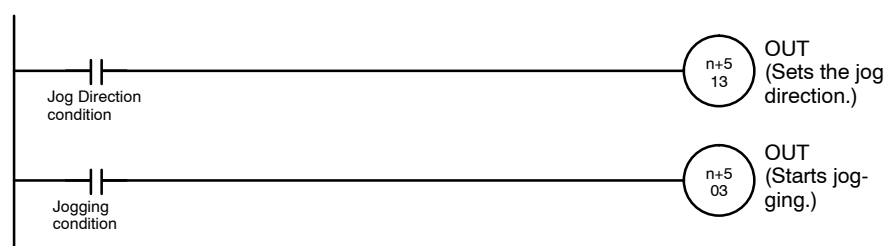
The Jogging Bit settings have the following functions.

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

Timing Chart



Program Example



### 6-5-21 Word n+5 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 axes (bit 13 in word n+7) 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.

**Note** Refer to 6-5-20 Word n+5 Bit 03: Jogging Bit (X-axis).

### 6-5-22 Word n+5 Bit 05: Current Position Preset Bit (X-axis)

This bit is used to set a new current position.

The new current position must be preset in the expansion data area.

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 available when the expansion data area is valid. 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.

The Current Position Preset Bits for the Y axes (bit 05 in word n+7) operate in the same way.



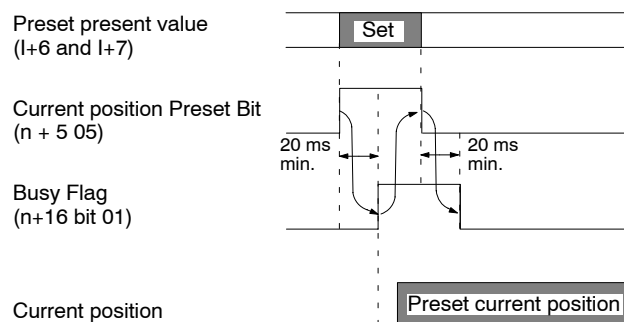
#### Caution

The present position presetting for the absolute encoder axis applies only to the present position registered in the MC Unit. The present position registered in the absolute encoder will not be preset. Therefore, the position will be automatically updated to the present position registered in the absolute encoder when the servo lock is executed next time.

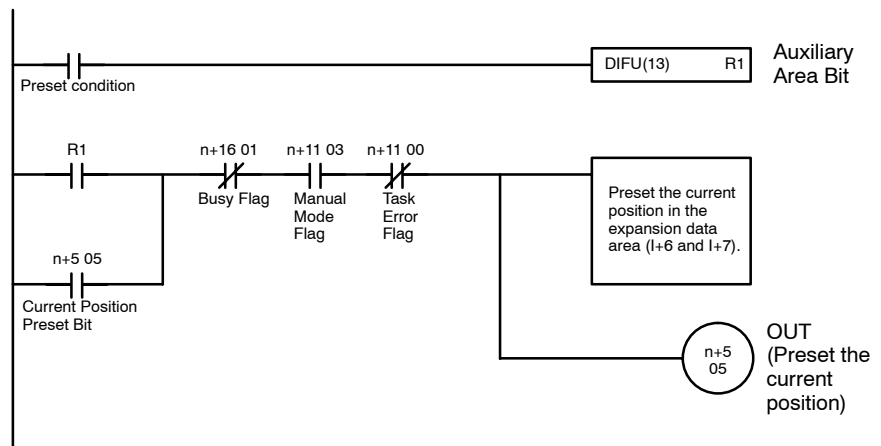
#### Signal

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

#### Timing Chart



Program Example



6-5-23 Word n+5 Bit 06: ABS Initialization Bit (X-axis)

This bit reads absolute data from the absolute encoder and sets the current value to zero. It will be enabled when turned ON, but the following restrictions apply.

- It must not be used within one second after turning ON servodriver control power supply or main circuit power supply.
- It cannot be used while a run command is being output to the servodriver. (An error will occur.)

This command is normally used to correct the absolute position after the absolute encoder has been replaced or set up. Once this bit has been turned ON, processing will continue until completion even if the bit is turned OFF before completion. Keep this bit ON, however, until the Busy Flag turns ON.

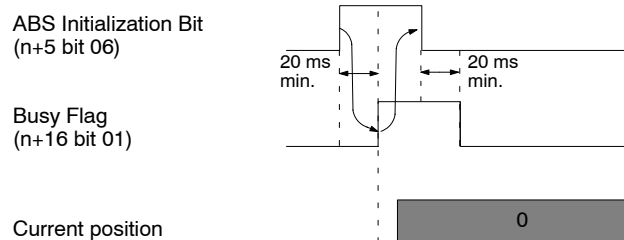
This command is ignored when the Busy Flag is ON. An error will occur if it is executed during a servo lock. Processing will not be interrupted during execution of the command even if a deceleration stop command is executed. This command has no effect on incremental operation.

Do not execute this command while system parameters are being downloaded. Doing so may result in incorrect initial settings of the absolute encoder. Refer to 7-6 *Setting Up the Absolute Encoder* for details.

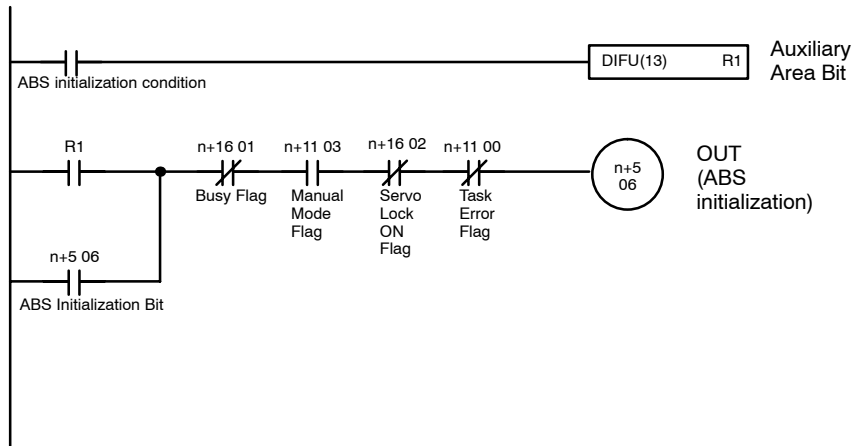
Signal

Reads absolute data and sets the current value to 0 when this bit goes from OFF to ON (up-differentiation).

Timing Chart



Program Example



6-5-24 Word n+5 Bit 07: ABS Soft Reset Bit (X-axis)

This bit corrects absolute data through software and sets the current value to zero. It is used to perform fine adjustment of absolute data after the absolute encoder has been initialized. This bit will be enabled when turned ON.

Keep this bit ON until the Busy Flag turns ON. This bit will be ignored when the Busy Flag is ON.

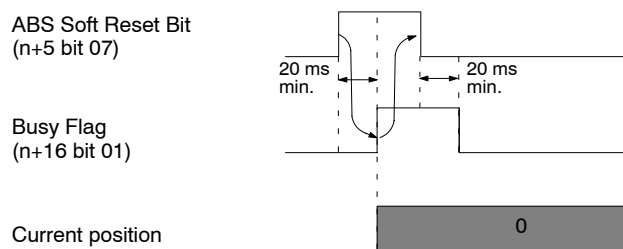
This command can be executed regardless of whether the servo lock is applied or not. Processing will not be interrupted during execution of the command even if a deceleration stop command is executed.

Do not execute this command while system parameters are being downloaded. Doing so may result in incorrect ABS Soft Reset values. Refer to 7-7 *ABS Soft Reset* for details.

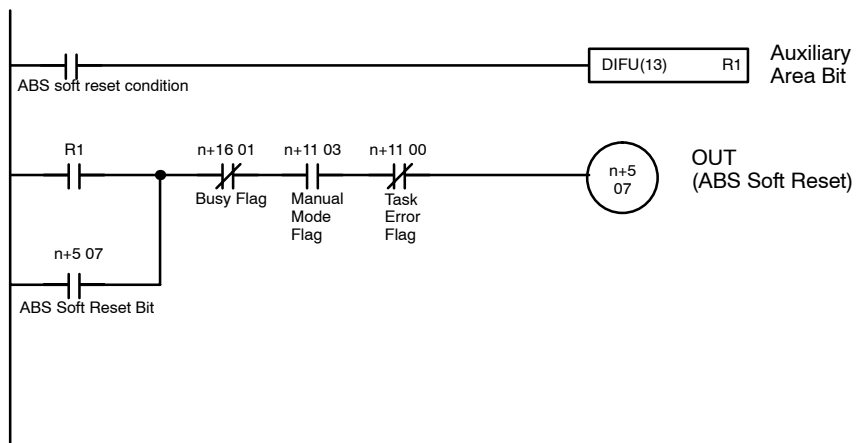
Signal

Executes ABS Soft Reset when this bit goes from OFF to ON (up-differentiation).

Timing Chart



Program Example



### 6-5-25 Word n+5 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+5 bits 14 and 15), read the input pulses from the MPG, multiply them by the MPG ratio, and perform MPG operation.

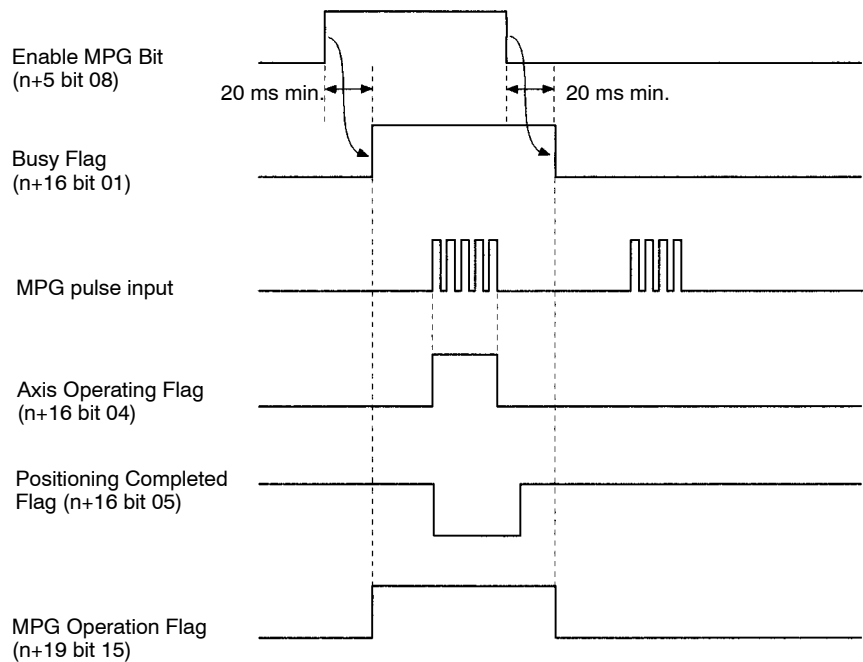
MPG operation will be stopped when this bit goes OFF. This bit is ignored if the Busy Flag is ON. This function is available to the X axis only.

#### Signal

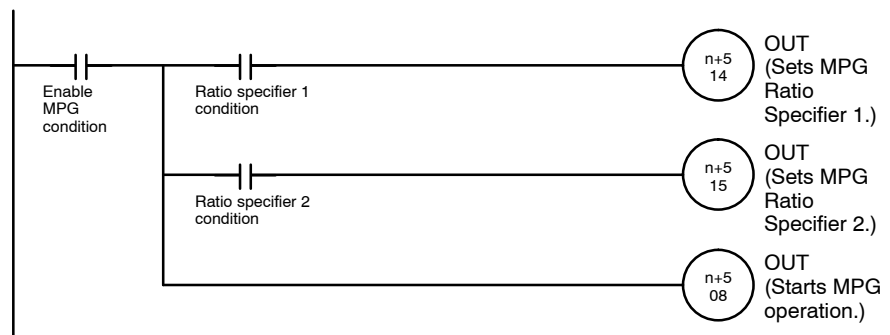
The Enable MPG Bit settings have the following functions.

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

#### Timing Chart



#### Program Example



### 6-5-26 Word n+5 Bits 14 and 15: MPG Ratio Specifiers 1 and 2 (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.

These bits are used in the following combination.

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.

**Note** Refer to 8-3-5 MPG Ratios in the MC Support Software Operation Manual for details on the MPG ratio and to 6-5-25 Word n+5 Bit 08: Enable MPG Bit (X-axis).

### 6-5-27 Word n+5 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. This bit is ignored if the Busy Flag is ON.

Keep this bit ON until the Busy Flag goes ON. Turn OFF this bit when the Busy Flag turns ON.

The Servo Lock Bits for the Y axes (bit 09 in word n+7) operate in the same way.

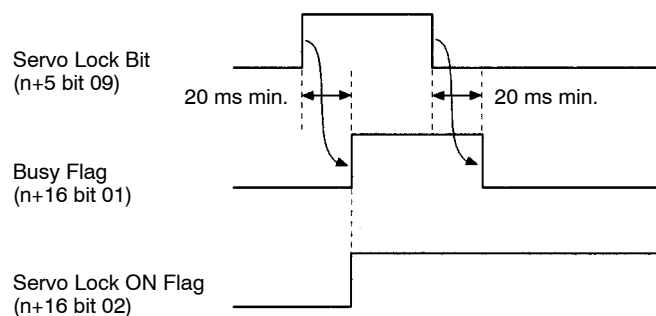
Turn OFF this bit when the Busy Flag turns ON.

#### Signal

The Servo Lock Bit settings have the following functions.

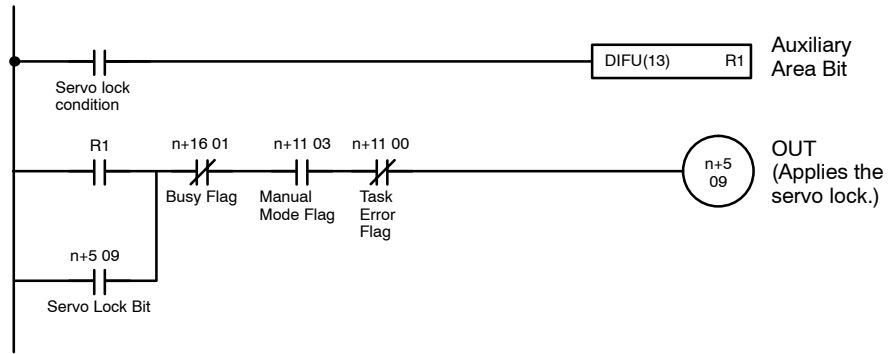
Signal	Function
↑ (OFF-to-ON)	Starts the servo lock.

#### Timing Chart





Program Example



6-5-28 Word n+5 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. This bit is ignored if the Busy Flag is ON

Keep this bit ON until the Busy Flag goes ON. Turn OFF this bit when the Busy Flag turns ON.

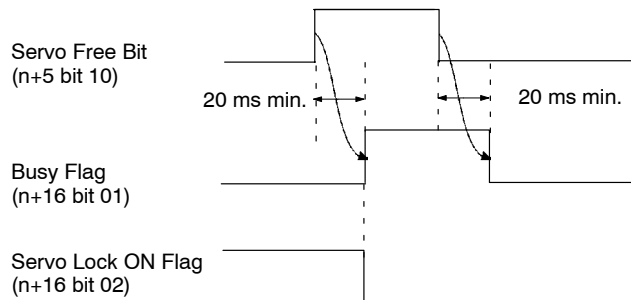
The Servo Free Bits for the Y axes (bit 10 in word n+7) operate in the same way. Turn OFF this bit when the Busy Flag turns ON.

Signal

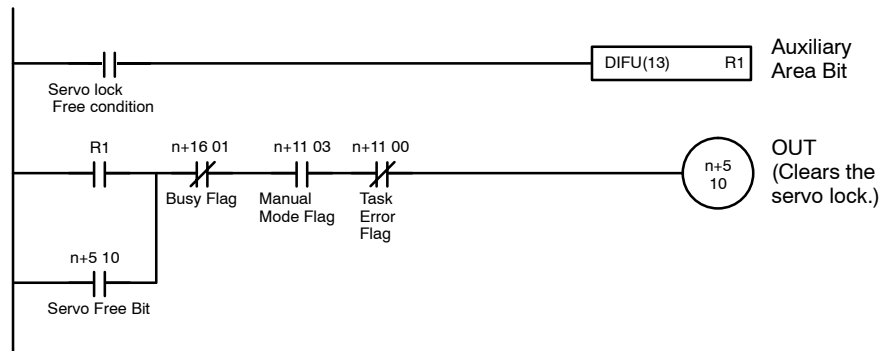
The Servo Free Bit settings have the following functions.

Signal	Function
↑ (OFF-to-ON)	Clears the servo lock status.

Timing Chart



Program Example



6-5-29 Word n+5 Bit 11: Driver Alarm Reset Bit (X-axis)

This bit is used to control the Driver Alarm Reset Output, an output to the servodriver; 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.

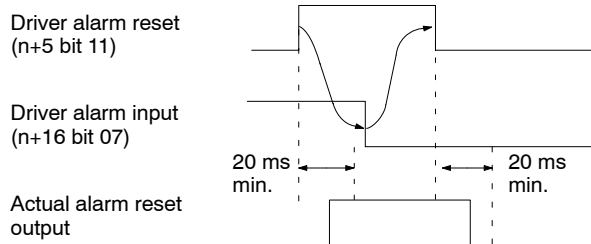
The Driver Alarm Reset Bits for the Y axes (bit 11 in word n+7) operate in the same way.

**Signal**

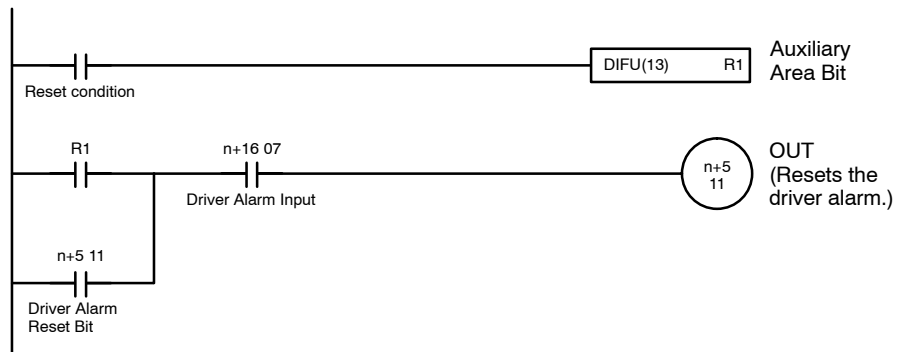
The Driver Alarm Reset Bit settings have the following functions.

Signal	Function
1 (ON)	Turns ON the Driver Alarm Reset Output.
0 (OFF)	Turns OFF the Driver Alarm Reset Output.

**Timing Chart**



**Program Example**



**6-5-30 Word n+5 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 n+4 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) (When moving from the machine origin to the reference origin)	0.1% to 100.0%
	G30 (SPEED CONTROL)	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 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.

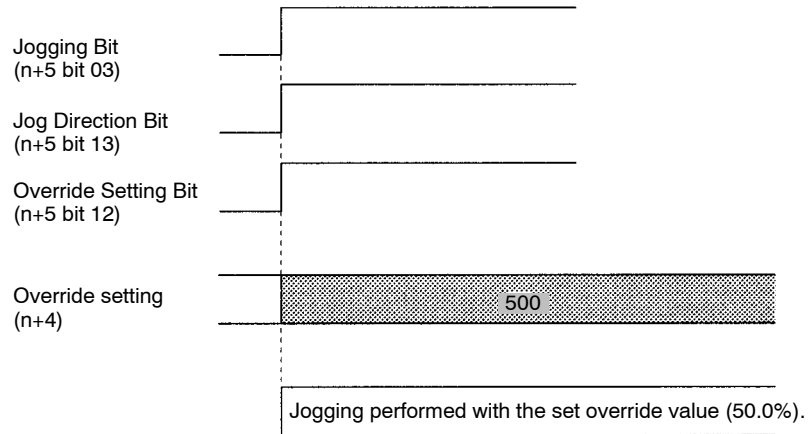
The Override Setting Bits for the Y axes (bit 12 in word n+7) operate in the same way.

**Signal**

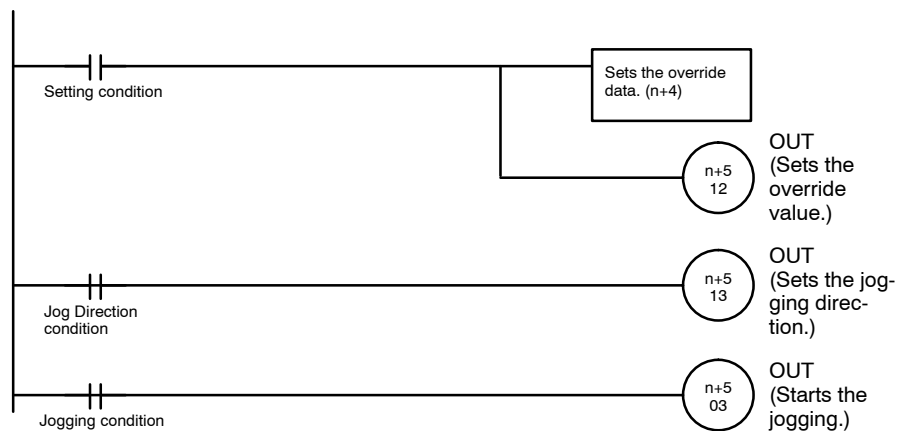
The Driver Alarm Reset Bit settings have the following functions.

Signal	Function
1 (ON)	The override data is valid.
0 (OFF)	The override data is invalid.

**Timing Chart**



**Program Example**



**6-5-31 Word n+8 Bit 06: Transmission Completion Flag**

This flag will be turned ON regardless of whether data has been transmitted normally or not.

This flag is used for the timing to turn the Transmission Control Bit ON.

This flag will not turn OFF until the Transmission Control Bit is turned OFF.

No data can be transmitted while this flag is ON.

**Signal**

The Transmission Completion Flag settings have the following meanings.

Signal	Function
↑	Data has been transmitted.
↓	No data is being written.

**Note** Refer to 6-5-11 Word n+1 Bit 11: Transmission Control Bit.

**6-5-32 Word n+8 Bit 07: Data Transmission Error Flag**

This flag will be turned ON when one of the following errors results during data transmission.

- The number of transfer words, source word, or destination address is not in BCD.

- The source word exceeds the upper limit of the data memory.
- The destination address is not in the acceptable range.
- The number of transfer words is not within 3 to 6,000 in three-word blocks for position data and two-word blocks for other data.
- The source word plus the number of transfer words exceed the upper limit of the data memory.
- The type of the data is one of the following:
  - Position data not in BCD.
  - Position data with a decimal position not to the right of the rightmost digit, between the rightmost and second-rightmost digit, between the second-rightmost and third-rightmost digit, between the third-rightmost and fourth-rightmost digit, or between the fourth-rightmost and fifth-rightmost digit.
  - Position data not within a range of -39,999,999 to 39,999,999.
  - A parameter not in BCD.
  - A parameter not within the acceptable range.

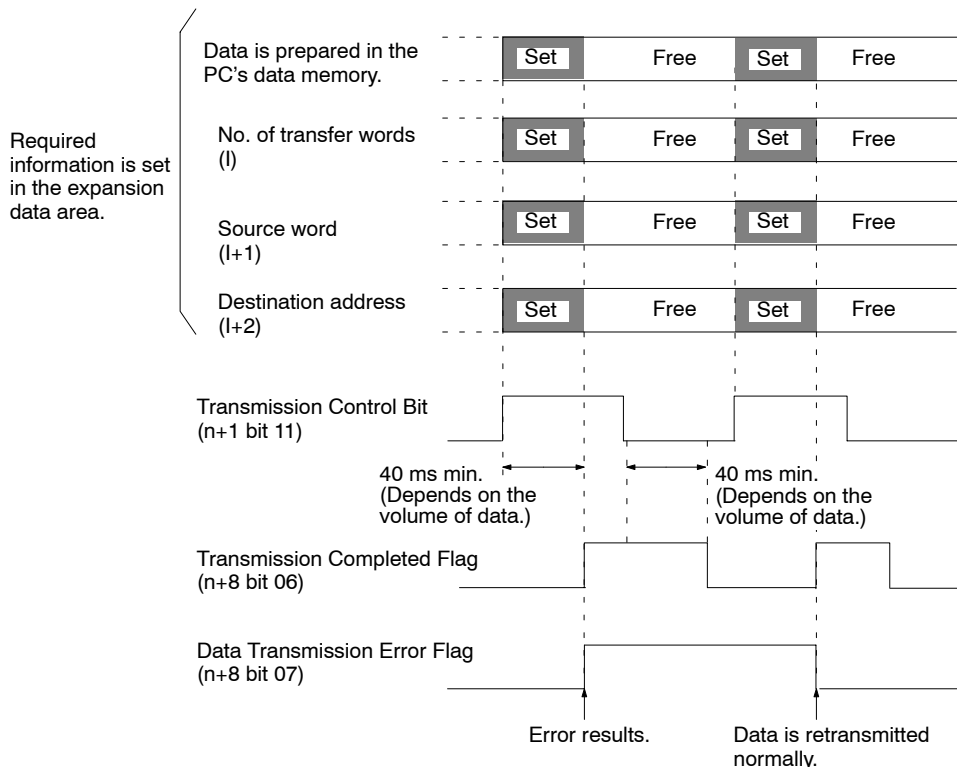
This flag will be turned OFF when data is transmitted normally.

**Signal**

The Data Transmission Error Flag settings have the following meanings.

Signal	Function
↑	An error resulted during data transmission.
↓	Data was retransmitted normally.

**Timing Chart**



**6-5-33 Word n+8 Bit 08: Reception Completion Flag**

This flag will be turned ON regardless of whether data has been received normally or not.

This flag is used for the timing to turn the Reception Control Bit ON.

This flag will not turn OFF until the Reception Control Bit is turned OFF.

No data can be received when this flag is ON.

**Signal**

The Reception Completion Flag settings have the following meanings.

Signal	Function
↑	Data has been received.
↓	No data is being received.

**Note** Refer to 6-5-12 Word n+1 Bit 12: Reception Control Bit.

**6-5-34 Word n+8 Bit 09: Data Reception Error Flag**

This flag will be turned ON when one of the following errors results during data reception.

- The number of transfer words, source address, or destination word is not BCD.
- The source address is not in the acceptable range.
- The destination word exceeds the upper limit of the data memory.
- The number of transfer words is not within 3 to 6,000 in three-word blocks for position data and two-word blocks for other data.
- The destination word plus the number of transfer words exceed the upper limit of the data memory.
- The destination word plus the number of transfer words are not within the acceptable range.

Example: If the Reception Control Bit is turned ON to read 600-word position data beginning with position data number 1,900, the last position data number exceeds the upper limit 1,999 (i.e.,  $1,900 + 600/3 = 2,100$ )

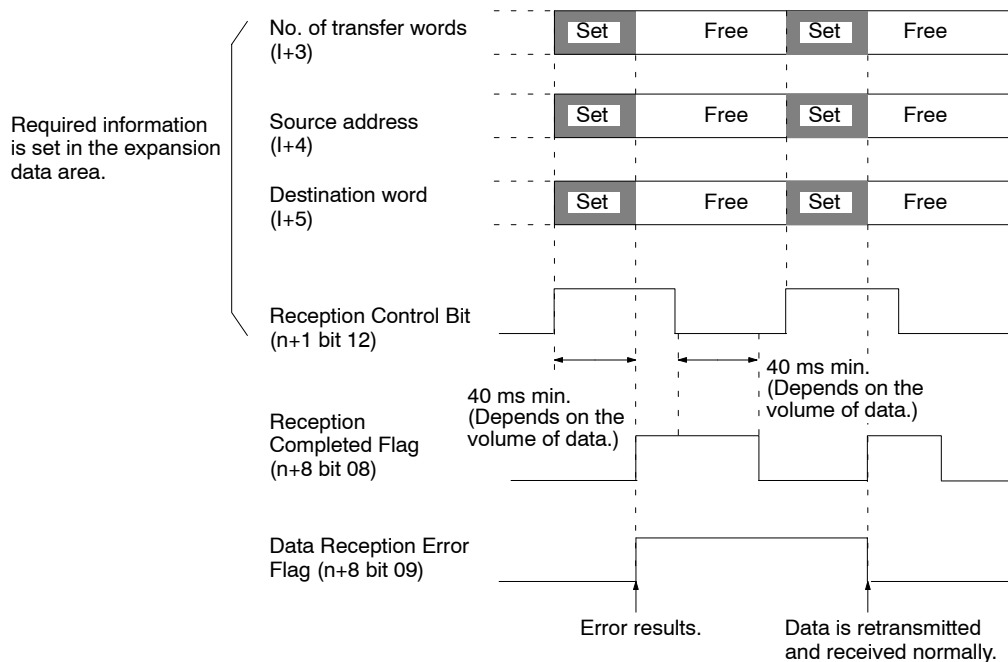
This flag will be turned OFF when data is received normally.

**Signal**

The Data Reception Error Flag settings have the following meanings.

Signal	Function
↑	An error resulted during data reception.
↓	Data was retransmitted and received normally.

**Timing Chart**



**6-5-35 Word n+8 Bit 10: Flash Memory Write Completion Flag**

This flag is turned ON when data is written to the flash memory normally.  
 This flag is used for the timing to turn the Flash Memory Write Bit ON.  
 This bit will not turn OFF until the Flash Memory Write Bit is turned OFF.  
 No data can be written when this flag is ON.

**Signal**

The Flash Memory Write Completion Flag settings have the following meanings.

Signal	Function
↑	Data has been written.
↓	No data is being written.

**Note** Refer to 6-5-13 Word n+1 Bit 13: Flash Memory Write Bit.

**6-5-36 Word n+8 Bit 11: 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.

**6-5-37 Word n+8 Bit 12: 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.

**6-5-38 Word n+8 Bit 13: Initial Setting Alarm Flag**

This flag will be turned ON under one of the following conditions.

- The initial setting area is not within the following areas.

PC model	Expansion data area
C200H	DM 0000 to DM 0899
C200HS	DM 0000 to DM 0999 DM 2000 to DM 5999
C200HX, C200HG, and C200HE	Model without EM area DM 0000 to DM 0999 DM 2600 to DM 5999 (The upper limit depends on the memory size.)  Model with EM area DM 0000 to DM 0999 DM 2600 to DM 5999 (The upper limit depends on the memory size.) EM 0000 to EM 6143

- The expansion data area is not set to 0, D, or E.
- The first word of the expansion data area is not BCD or exceeds the upper limit of the data memory.

The MC Unit can, however, operate without using the expansion data area even if this flag is ON.

**Signal**

The Initial Setting Alarm Flag settings have the following meanings.

Signal	Meaning
1	The initial setting data is incorrect.
0	The initial setting data is correct.

**6-5-39 Word n+8 Bit 14: Error Flag**

This flag is turned ON when an error results.

Check the error type data and error code to find the type of error.

If more than one error results, the error type data and error code indicate the error detected first and the error code is output to the | +12 on | +16 expansion information area.

To check all the errors, refer to the expansion data.

This flag is turned OFF with an error reset input.

This flag is turned ON again when another error results.

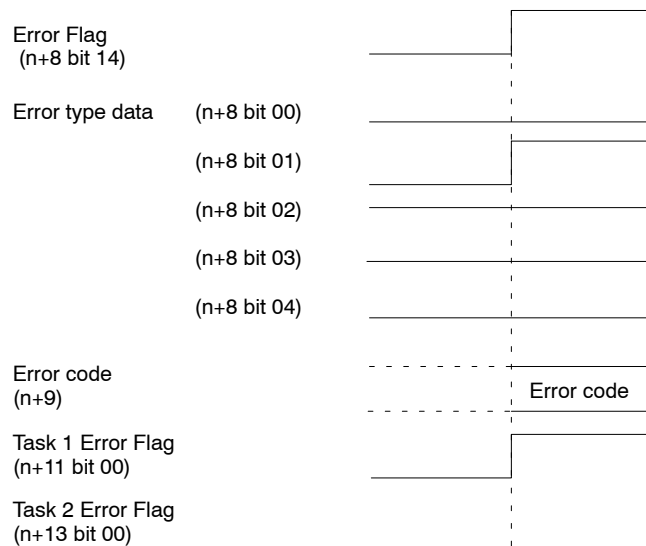
**Signal**

The Error Flag settings have the following meanings.

Signal	Meaning
1	An error resulted.
0	No error resulted.

**Timing Chart**

Example: In this example, a task 1 error resulted.

**6-5-40 Word n+8 Bit 15: Expansion Data Read Completion Flag**

This flag is used to check whether the IN refresh in the expansion data area is refreshed.

The IN refresh is always refreshed when this flag is turned ON.

The IN refresh is not refreshed when this flag is turned OFF.

**Signal** The Expansion Data Read Completion Flag settings have the following meanings.

Signal	Meaning
1	The expansion data is not refreshed.
0	The expansion data is refreshed.

**Note** Refer to 6-5-15 Word  $n+1$  Bit 15: Expansion Data Read Bit.

### 6-5-41 Word $n+11$ Bit 00: Task Error Flag (Task 1)

This bit is turned ON when one of the following error results.

- A program error.
- An error of the axis controlled by the task.

If the cause of the error isn't corrected, the flag will be turned ON again the next time the task is executed. The Error Flag (word  $n+8$  bit 14) will be turned ON when this bit is turned ON.

Word  $n+13$  bit 00 is used by task 2. The status of task 1 is explained below. There is no difference in function between the statuses of tasks 1 and 2 although their words are different.

**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.

**Note** Refer to 6-5-39 Word  $n+8$  Bit 14: Error Flag.

### 6-5-42 Word $n+11$ 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 ( $n+11$  bit 04)
- Memory Run Completed Flag ( $n+11$  bit 05)
- Optional Input Standby Flag ( $n+11$  bit 06)
- M Code Reset Standby Flag ( $n+11$  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 task 2 (bit 02 in word  $n+13$ ) 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.

**Note** Refer to 6-5-1 Word  $n+1$  Bit 01: Automatic/Manual Mode Bit.



**6-5-43 Word n+11 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 task 2 (bit 03 in word n+13) 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.

**Note** Refer to 6-5-1 Word n+1 Bit 01: Automatic/Manual Mode Bit.

**6-5-44 Word n+11 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 task 2 (bit 04 in word n+13) 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.

**Note** Refer to 6-5-45 Word n+11 Bit 05: Memory Run Completed Flag.

**6-5-45 Word n+11 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.

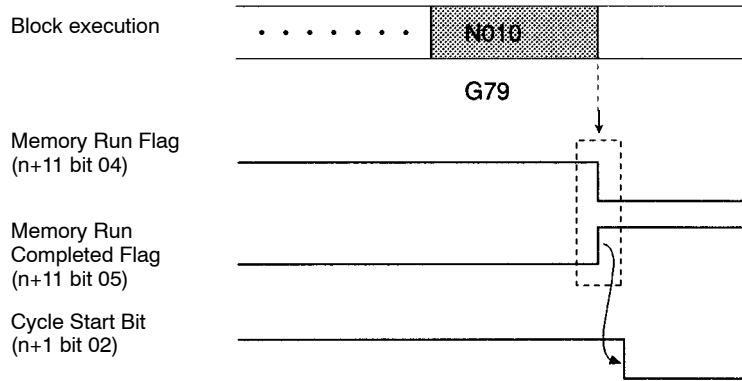
The Memory Run Completed Flags for task 2 (bit 05 in word n+13) operate in the same way.

**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



**6-5-46 Word n+11 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 task 2 (bit 06 in word n+13) 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.

**6-5-47 Word n+11 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 task 2 (bit 07 in word n+13) 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.

### 6-5-48 Word n+11 Bit 08: M Strobe Flag (Task 1)

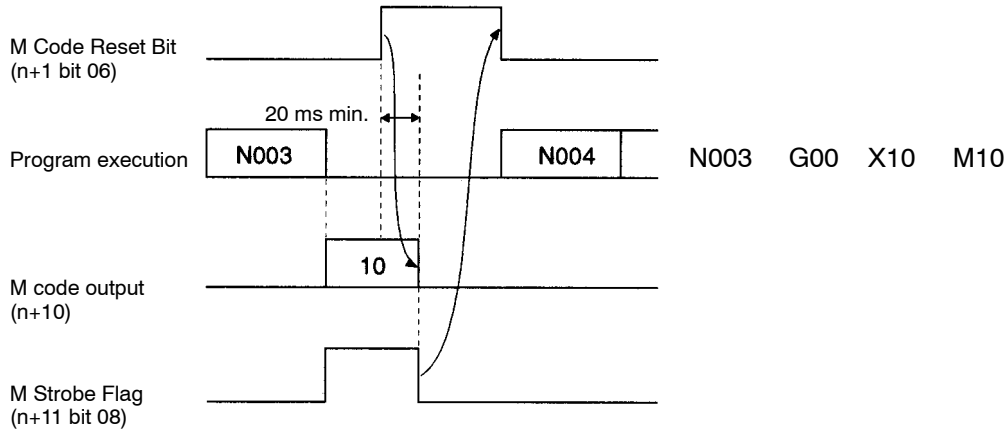
This flag indicates whether an M code is being output. The M Strobe Flags for task 2 (bit 08 in word n+13) operate in the same way.

**Signal**

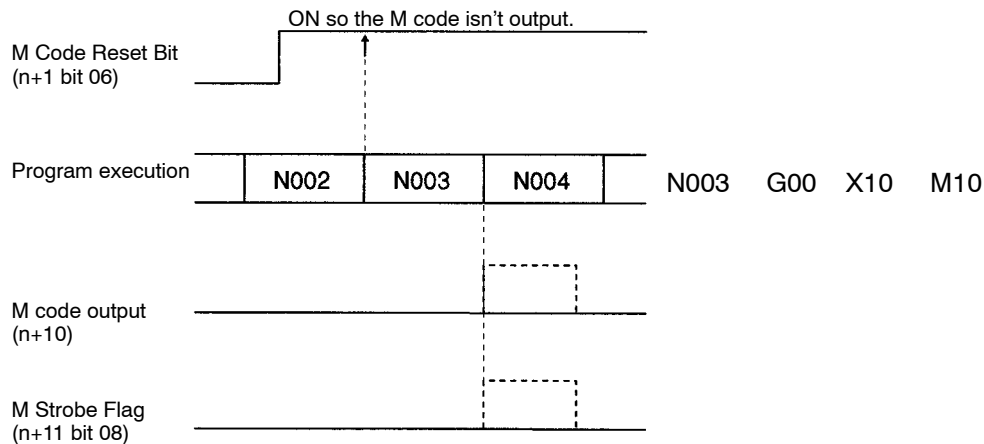
The M Strobe Flag settings have the following meanings.

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**



### 6-5-49 Word n+11 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).

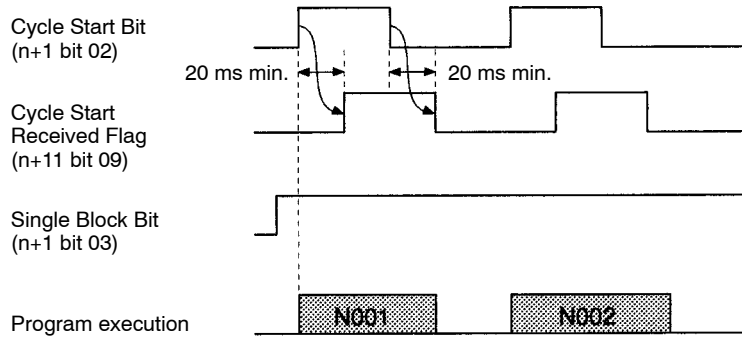
The Cycle Start Received Flags for task 2 (bit 09 in word n+13) operate in the same way.

**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



**6-5-50 Word n+11 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.

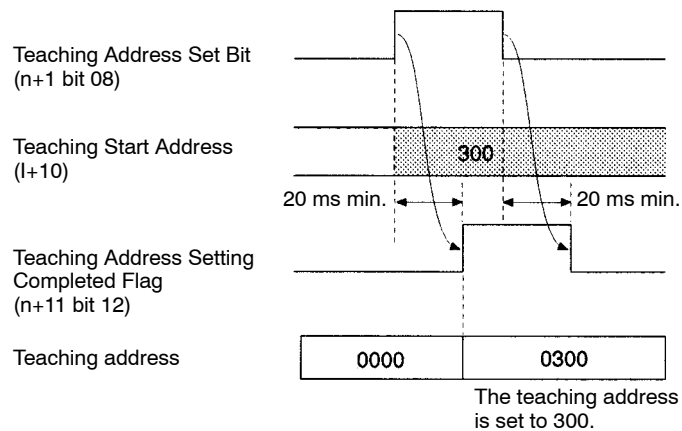
The Teaching Address Setting Completed Flags for task 2 (bit 12 in word n+13) operate in the same way.

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



**6-5-51 Word n+11 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.

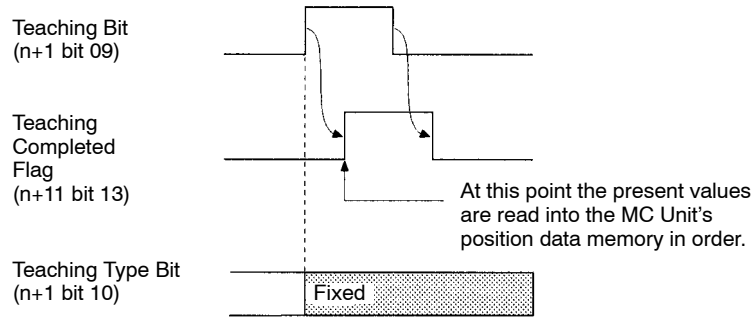
The Teaching Completed Flags for task 2 (bit 13 in word n+13) operate in the same way.

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



6-5-52 Word n+11 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).

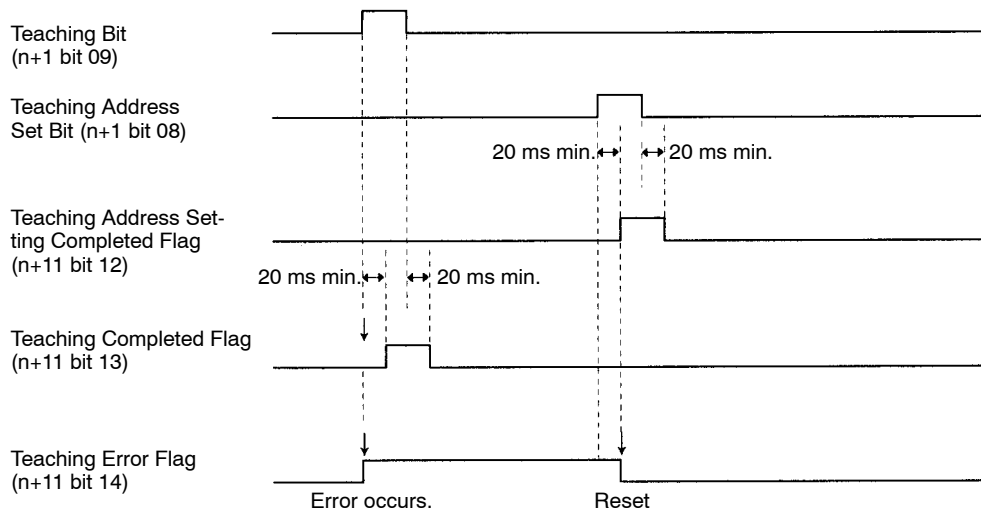
The Teaching Error Flags for task 2 (bit 14 in words n+13) operate in the same way.

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



6-5-53 Word n+11 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.

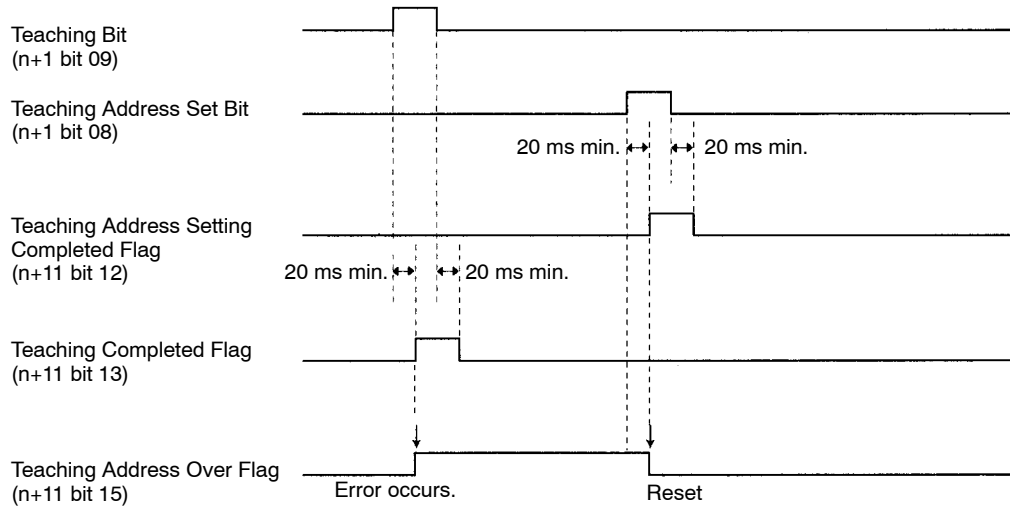
The Teaching Address Over Flags for task 2 (bit 15 in word n+13) operate in the same way.

**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**



**6-5-54 Word n+16 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 axes (bit 00 in word n+19) 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.

**6-5-55 Word n+16 Bit 01: 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
- ABS Initialization
- ABS Soft Reset
- 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.

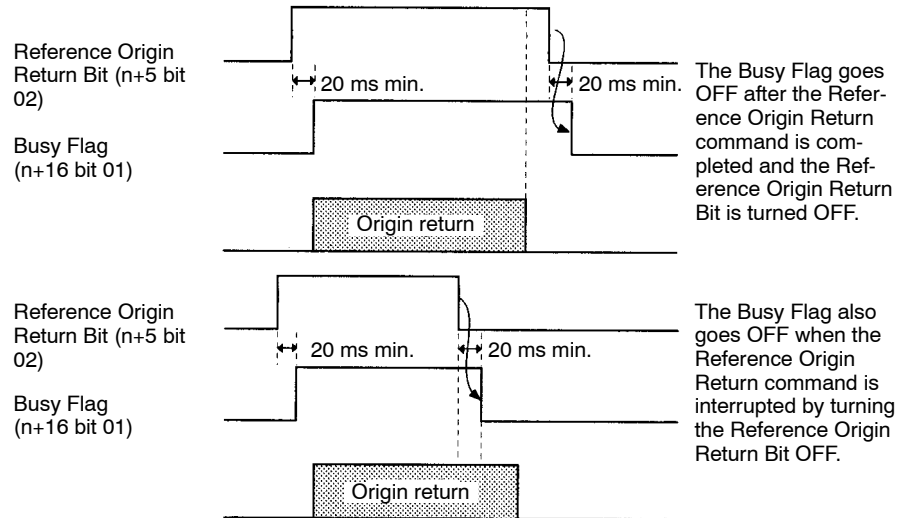
The Busy Flags for the Y axes (bit 01 in word n+19) operate in the same way.

**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**



**6-5-56 Word n+16 Bit 02: 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.

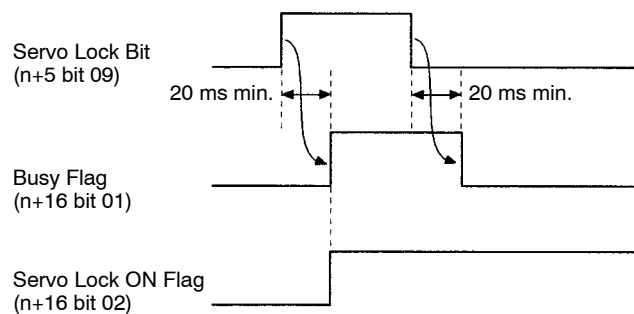
The Servo Lock ON Flags for the Y axes (bit 02 in word n+19) operate in the same way.

**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**



**6-5-57 Word n+16 Bit 03: No Origin Flag (X-axis)**

This flag is turned ON when the reference origin isn't fixed. Execute positioning functions in automatic mode while this flag is OFF. Make sure that this flag is OFF when executing origin returns, otherwise, an error will occur.

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.”
- Data from the absolute encoder was not read properly for the absolute specification axis.
- A wiring error, feedback pulse error, or current position counter overflow resulted when the MC Unit is in origin search operation.
- G29 (ORIGIN UNDEFINED) was executed.

The origin will be fixed and the No Origin Flag will be turned OFF in the following cases:

**Absolute Specification Axis**

- Data was read properly from the absolute encoder.

**Incremental Specification Axis**

- 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.

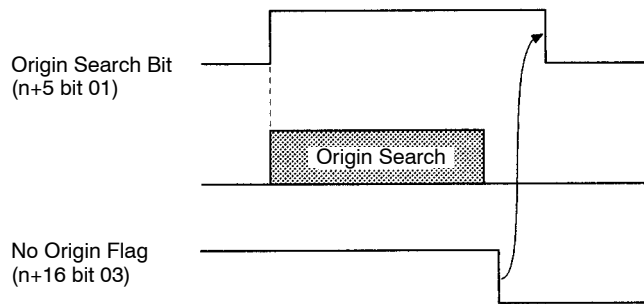
The No Origin Flags for the Y axes (bit 03 in words n+19) operate in the same way.

**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**



**6-5-58 Word n+16 Bit 04: 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 axes (bit 04 in words n+19) operate in the same way.

**Signal**

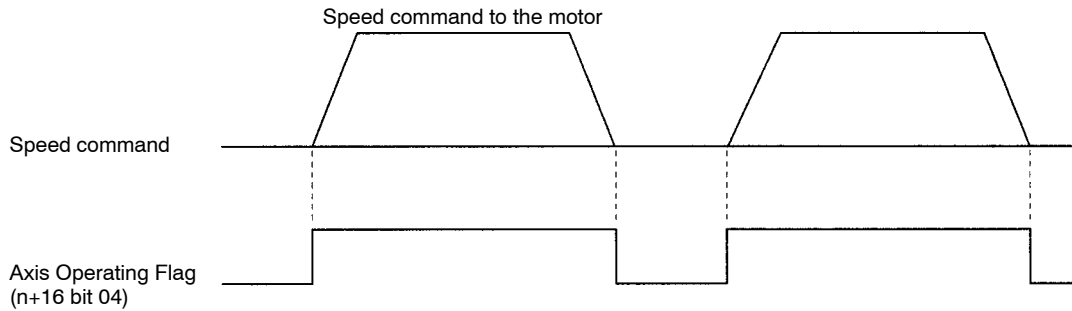
The Axis Operating Flag settings have the following meanings.

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.



**6-5-59 Word n+16 Bit 05: 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.

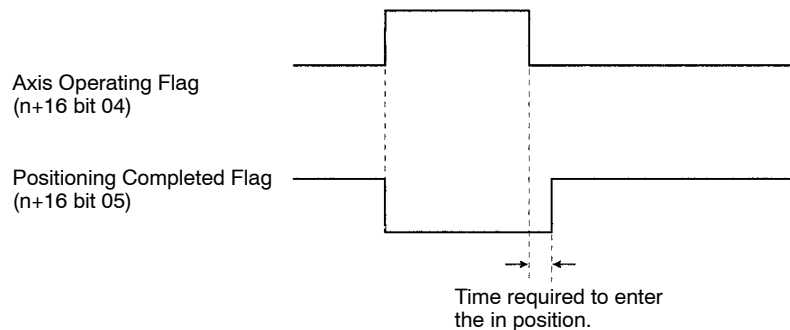
The Positioning Completed Flags for the Y axes (bit 05 in words n+19) operate in the same way.

**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**



**6-5-60 Word n+16 Bit 06: Error Counter Alarm Flag (X-axis)**

This flag is turned ON when the count in the error counter exceeded 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 axes (bit 06 in words n+19) 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.

**6-5-61 Word n+16 Bit 07: 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 axes (bit 07 in words n+19) operate in the same way.

**Signal**

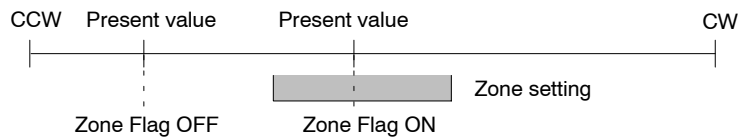
The Alarm Input settings have the following meanings.

Signal	Meaning
1	ON when the driver alarm input is ON.
0	OFF when the driver alarm input is OFF.

**Note** Refer to 6-5-29 Word n+5 Bit 11: Driver Alarm Reset Bit.

**6-5-62 Word n+16 Bits 08 to 15: Zone Flags (X-axis)**

Word n+16 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 axes (bits 08 to 15 in word n+19) 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.

**6-5-63 Word n+19 Bit 15: MPG Operation Flag (X-axis)**

This flag is used only if Y axis is used for an MPG. (Refer to *Section 2-3 MPG Connector*.) This flag will be in zone 8 like word n+16 if no MPG is connected. 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
- ABS initialization
- ABS Soft Reset

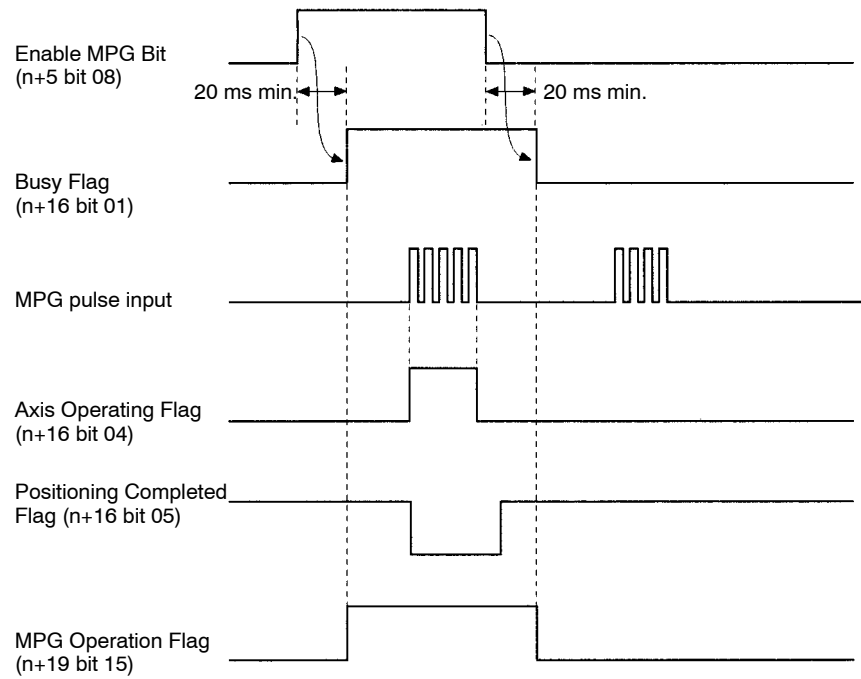
This flag is turned ON at the same time as the Busy Flag, and is turned OFF when the Enable MPG Bit is turned OFF.

**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





# 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 .....	178
7-2	Sensors Required for an Origin Search .....	178
7-3	Origin Search Parameters .....	179
7-4	Origin Search Patterns .....	180
7-4-1	Basic Origin Search Pattern .....	180
7-4-2	Reverse-mode Origin Searches 1 .....	181
7-4-3	Reverse-mode Origin Searches 2 .....	182
7-4-4	Reverse-mode Origin Searches 3 .....	183
7-4-5	Reverse-mode Origin Searches 4 .....	184
7-4-6	One Direction-mode Origin Searches 1 .....	185
7-4-7	One Direction-mode Origin Searches 2 .....	185
7-4-8	One Direction-mode Origin Searches 3 .....	186
7-5	Overview of Absolute Encoders .....	186
7-6	Setting Up the Absolute Encoder .....	187
7-6-1	Purpose .....	187
7-6-2	Setup Procedure .....	187
7-7	ABS Soft Reset .....	188
7-8	Absolute Encoder Interface Specifications .....	189

## 7-1 Introduction

This section will describe methods of origin search for motion control systems using either incremental encoders or absolute encoders.

### Using Incremental Encoders

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 (X axis: n+5 bit 05; Y axis: 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 (X axis: n+5 bit 01; Y axis: n+7 bit 01) or executing function G28 (ORIGIN SEARCH). An origin search can also be started from the Teaching Box.

Refer to *Sections 7-2 to 7-4* for determining the origin.

### Using Absolute Encoders

The absolute encoders retain absolute data by using the backup batteries when the power to the MC Unit or Servodrivers are turned OFF.

When the power to the MC Unit is turned ON again and if the system is set to the servo lock state, absolute data will be read from the absolute encoder to determine the present position. No origin search is required.

### Absolute Encoder Setup

When an absolute encoder is installed for the first time, when the battery is replaced, or when the encoder cable between the Servodriver and motor is removed for a long time, the absolute data must be defined. This is called "ABS Initialization."

When making fine adjustments on the present value after the ABS Initialization, execute the "ABS Soft Reset."

For further details, refer to *Sections 7-5 to 7-8*.

The ABS Initialization and ABS Soft Reset can be executed from the interface area (bits 06 and 07 of n+5 words) or from the Teaching Box. For executing from the interface area, refer to *Section 6 PC Data Area Interface*. For executing from the Teaching Box, refer to the *Teaching Box Operation Manual*.

## 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.

<b>CW and CCW Limit Switches</b>	These switches indicate that the workpiece's maximum range has been reached when the motor rotates clockwise (CW) or counterclockwise (CCW).
<b>Origin Proximity Switch</b>	This switch indicates that the axis is near the origin.
<b>Encoder's Z-phase Signal</b>	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

The reference origin offset value is valid only in the origin search operation. The value is not valid for the axes using the absolute encoder.

### 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

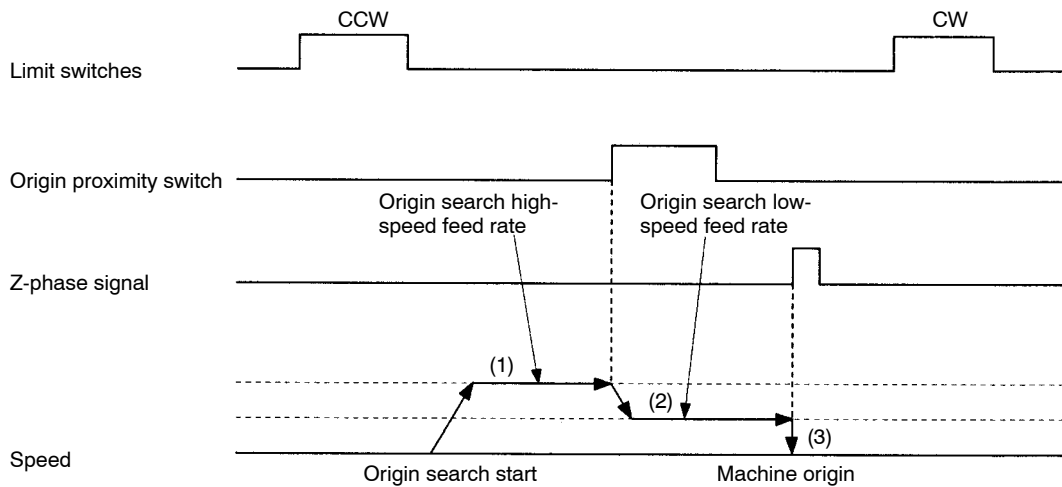
Refer to the *MC Support Software Operation Manual* for details on setting these parameters. The Origin Search Method, Origin Search Direction, Origin Deceleration Method, and Origin Proximity Logic parameters are in the Machine Parameters Edit menu. The Origin Search High-speed Feedrate and Origin Search 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 CW. 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.

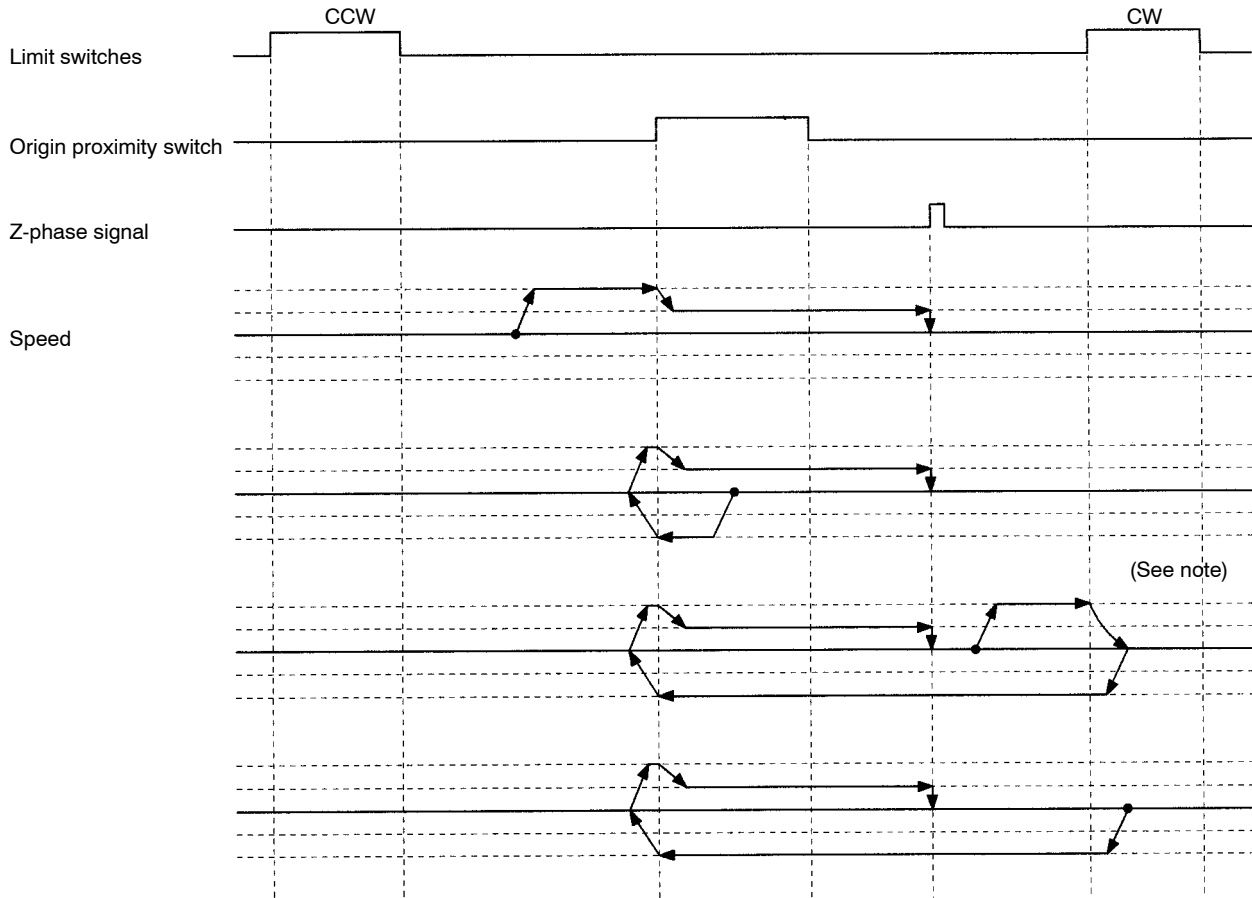
- 1, 2, 3...** 1. When the origin search is executed, the axis is moved in the specified direction at the origin search high-speed feed rate.
2. When the origin proximity switch input is received, the search speed is reduced to the origin search low-speed feed rate.
3. After the origin proximity switch 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.

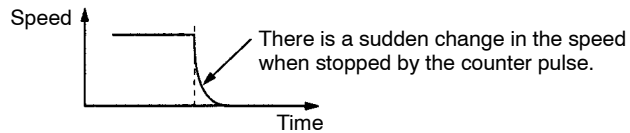


### 7-4-2 Reverse-mode Origin Searches 1

These origin searches are performed with an origin proximity switch and the initial search direction set to CW.

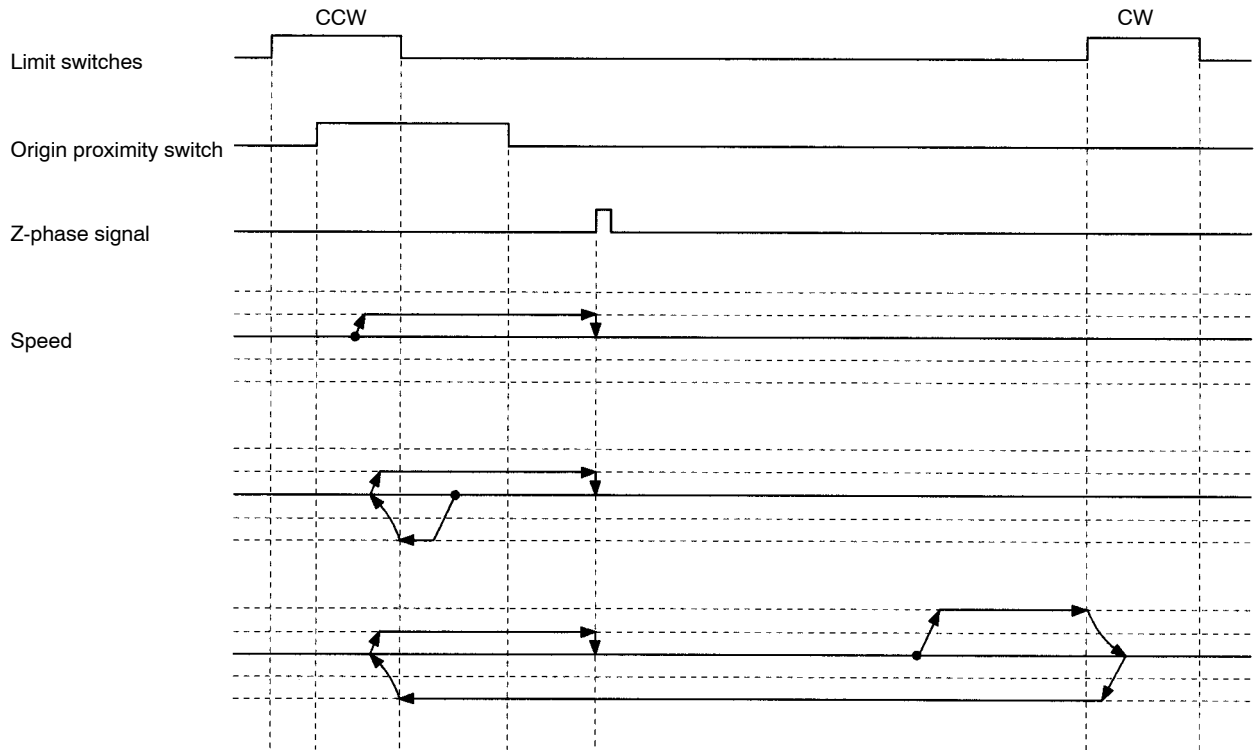


**Note** When the direction is reversed by a CCW limit switch or CW limit switch during the origin search operation, reverse operation will start after the movement is stopped by the counter pulse. Also, it does not decelerate to a stop.



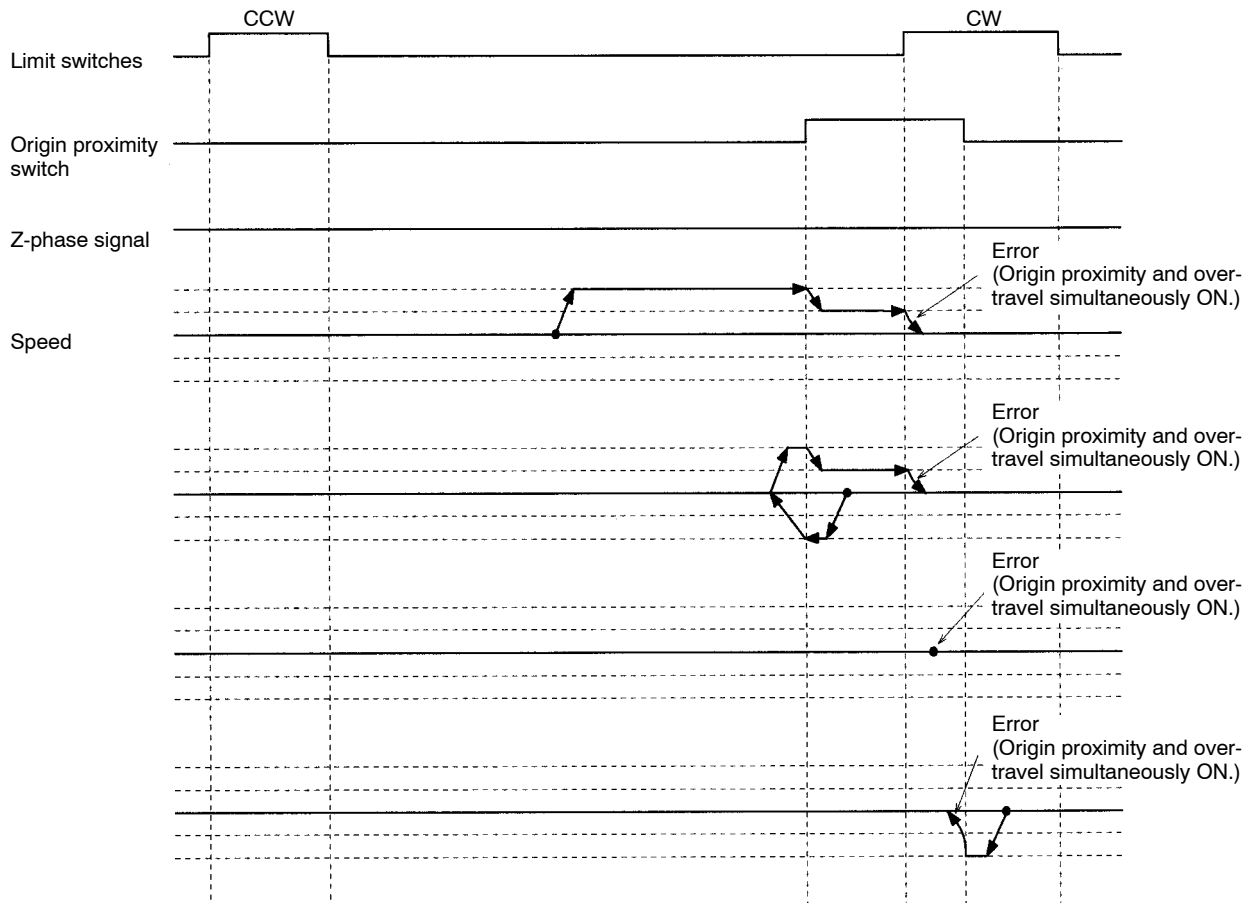
### 7-4-3 Reverse-mode Origin Searches 2

These origin searches are performed with an origin proximity switch and the initial search direction set to CW.



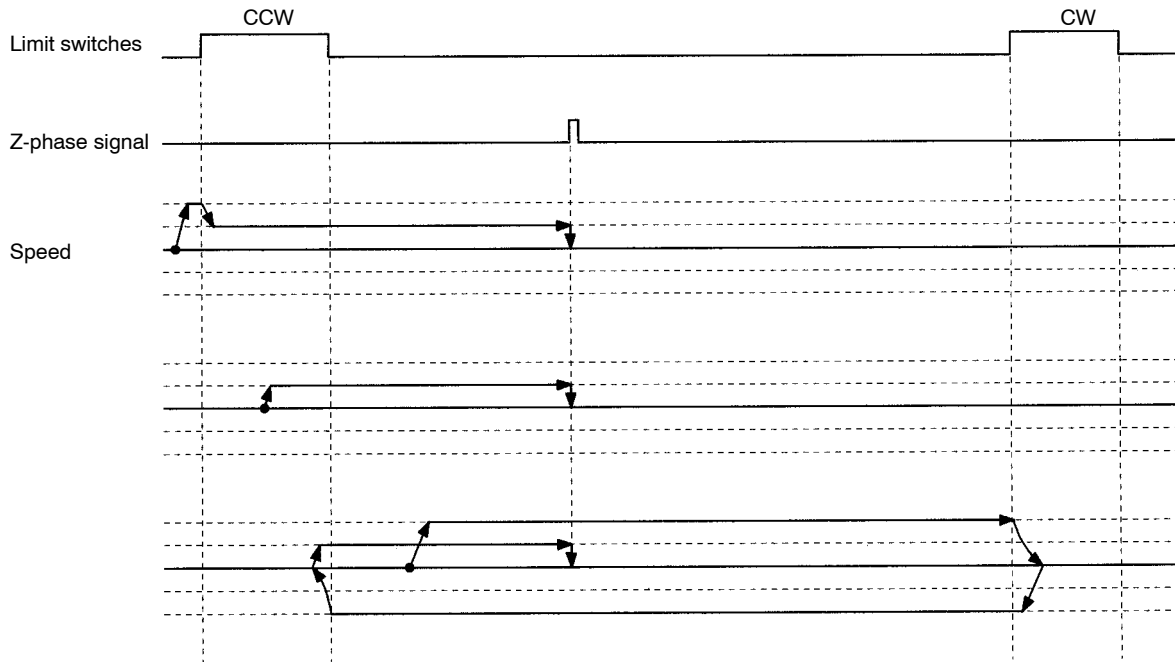
### 7-4-4 Reverse-mode Origin Searches 3

These origin searches are performed with an origin proximity switch and the initial search direction set to CW. There is no Z-phase input, so all of these searches result in errors.



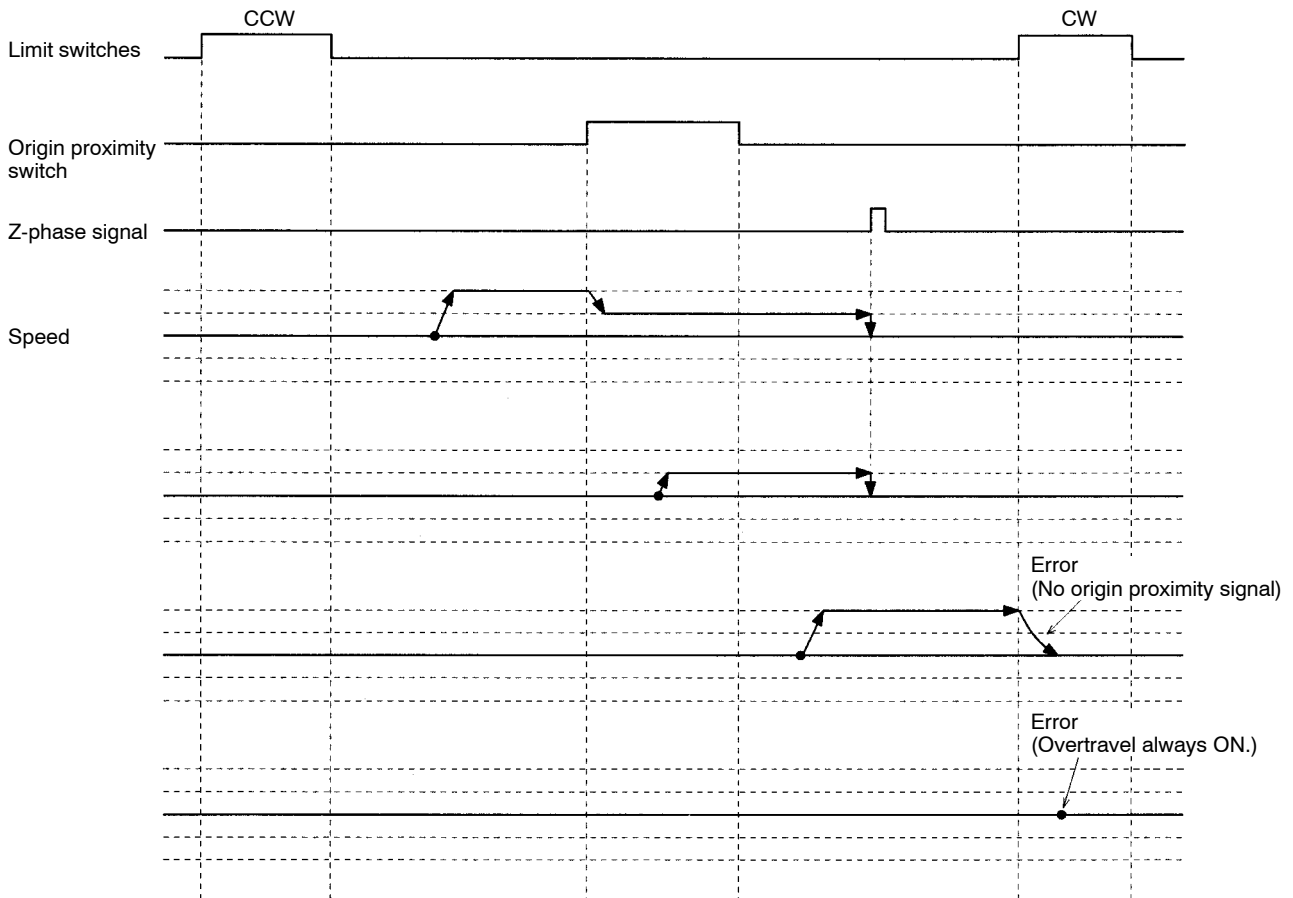
### 7-4-5 Reverse-mode Origin Searches 4

These origin searches are performed without an origin proximity switch and the initial search direction set to CW.



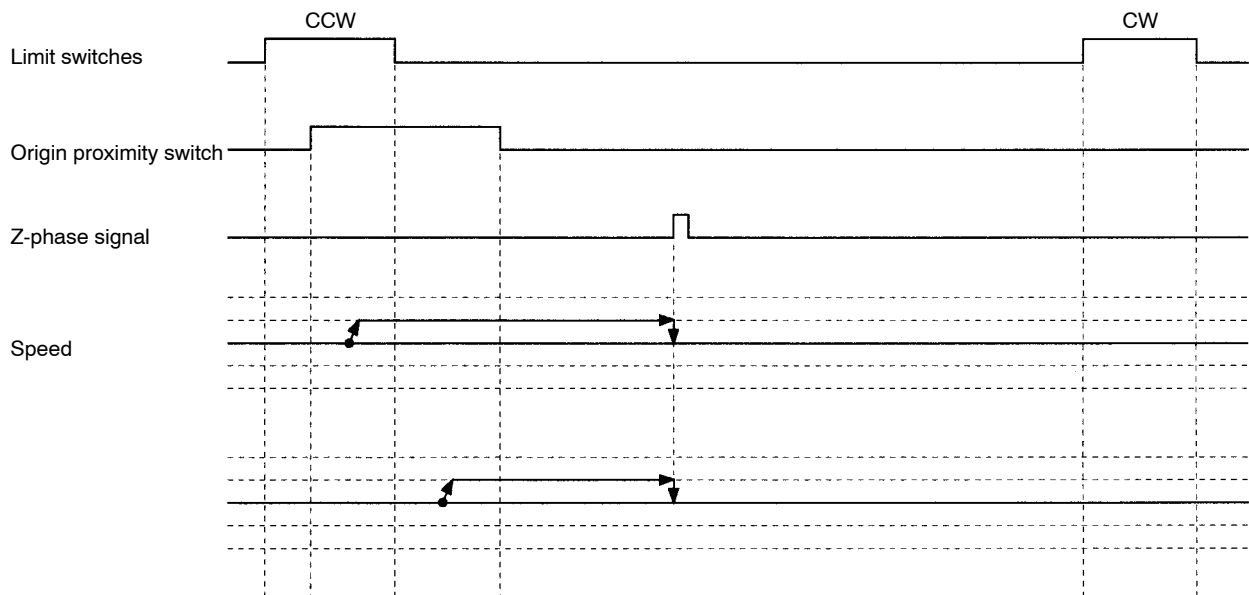
### 7-4-6 One Direction-mode Origin Searches 1

These origin searches are performed with an origin proximity switch and the initial search direction set to CW.



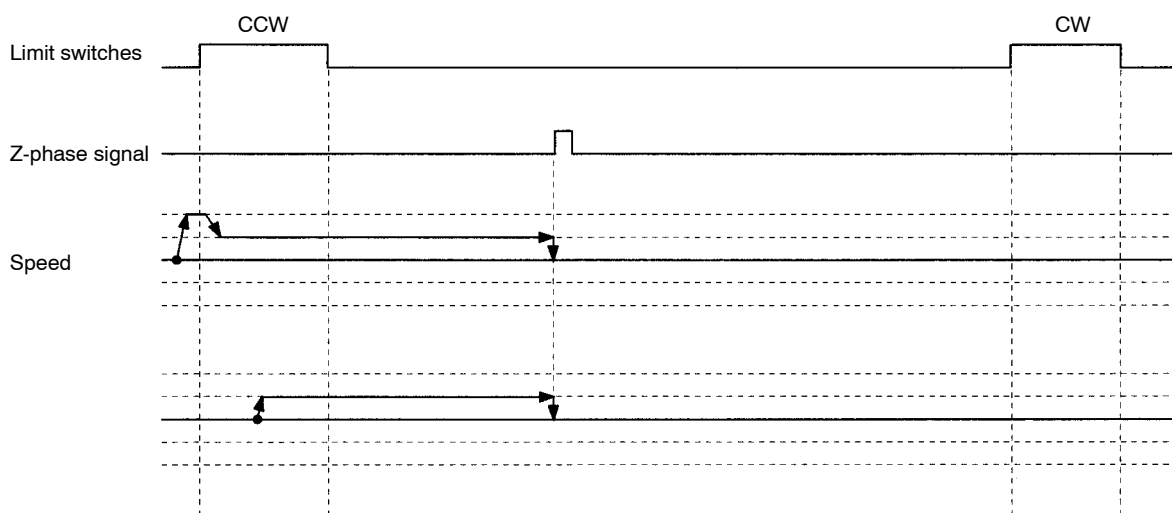
### 7-4-7 One Direction-mode Origin Searches 2

These origin searches are performed with an origin proximity switch and the initial search direction set to CW.



### 7-4-8 One Direction-mode Origin Searches 3

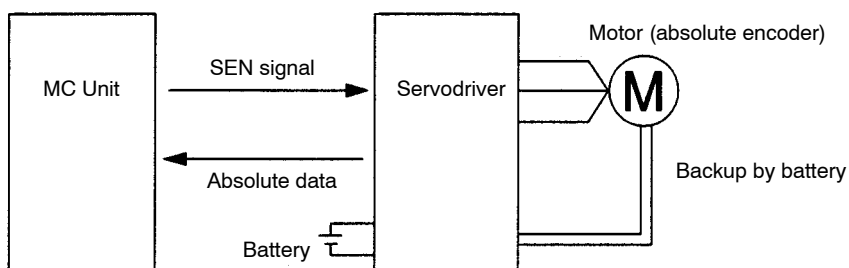
These origin searches are performed without an origin proximity switch and the initial search direction set to CW.



## 7-5 Overview of Absolute Encoders

### Overview

The absolute encoder retains absolute data by using the backup battery when the MC Unit is turned OFF. When the MC Unit is turned ON again, the absolute data can be read from the absolute encoder to determine the current position. No origin search is required.



The MC Unit receives data from the absolute encoder by changing the SEN signal level from low to high.

**Note** For details on absolute encoder interfaces, refer to *7-8 Absolute Encoder Interface Specifications*.

The MC Unit receives absolute data by changing the SEN signal state from OFF to ON when the Run Command Output to the Servodriver is changed from OFF to ON.

### Applicable Absolute Encoders

The following Servomotors are provided with absolute encoders that can be connected to the MC Unit.

- OMRON U Series: R88M-U□□□□□S
- R88M-U□□□□□T

## 7-6 Setting Up the Absolute Encoder

### 7-6-1 Purpose

An absolute encoder that can be connected to the MC Unit must be set up after the revolution count data of the encoder has been cleared to zero or after the encoder has been disconnected from the battery over a long period of time (for two days or more for OMRON U-series Absolute Encoders). At the same time, the MC Unit must be initialized.

The reason for this is because the present value of the axis in the MC Unit must be set to zero when the revolution count data of the absolute encoder has been cleared to zero.

### 7-6-2 Setup Procedure

The absolute encoder setup and MC Unit initialization procedures are described as follows.

#### Absolute Encoder Setup

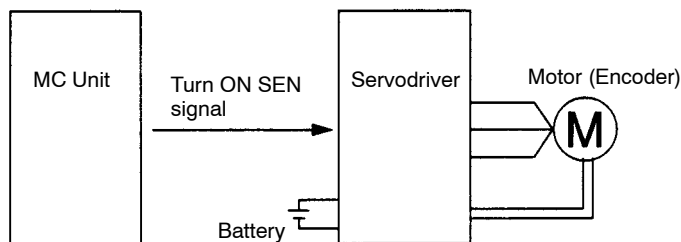
The following setup procedure is for OMRON U-series Absolute Encoders.

**1, 2, 3...**

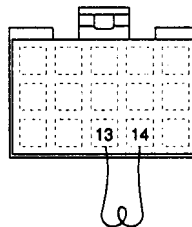
1. Connect the battery and wire the Servodriver and Absolute Encoder.
2. Turn ON the Servodriver SEN signal input.

With the MC Unit connected to the Servodriver, turn ON the MC Unit and the Servodriver. The SEN signal from the MC Unit will turn ON.

The Absolute Encoder will receive a 5-V power supply. Leave the power ON for at least 3 minutes to fully charge the backup capacitor. Do not issue servo lock commands from the MC Unit during this operation.



3. Turn OFF the Servodriver and MC Unit, then disconnect the encoder connector.
4. Short-circuit the encoder connector terminals 13 and 14 for one or two seconds.



5. Reconnect the encoder connector.
6. Turn ON the Servodriver and MC Unit. Then, initialize the MC Unit.

#### Initializing the MC Unit

**1, 2, 3...**

1. Check that the pin 1 of the DIP switch on the rear of the MC Unit is set to OFF. If this pin is set to ON, an error will occur when the MC Unit is initialized.
2. Release the servo lock. If the servo lock is applied, an error will occur when the MC Unit is initialized.

- Execute initialization.

Initialization can be executed either from the data area interface using a ladder program or by using the Teaching Box.

#### Using a Ladder Program

Turn ON the ABS Initialization Bit in manual mode when the Busy Flag is OFF. Leave this bit ON until the Busy Flag turns ON.

#### Using the Teaching Box

Select **5. ABS INIT SET** from the extension mode menu, and execute processing.

- Note**
- Execute initialization when absolute data is within the range of  $\pm 32767$  pulses. Otherwise, an error will occur.
  - Refer to *6-5 Interface Bit Specifics* for details on a timing chart and ladder program example of the ABS Initialization Bit. Refer to the *Teaching Box Operation Manual* for details on Teaching Box operation.

ABS Initialization will reset the current position to zero and store the absolute data in the machine parameter area as the Absolute Encoder initial setting.

- Note** After ABS Initialization is completed, back up the system parameters using the MC Support Software.

Use the flash memory write function of the MC Support Software or Teaching Box to write the settings to the flash memory.

The Absolute Encoder initial settings can be displayed using the MC Support Software, but they cannot be changed.

Do not execute ABS Initialization while system parameters are being downloaded. Otherwise, the Absolute Encoder initial settings will not be stored properly.

Do not execute ABS Initialization within one second after the main power supply or control power supply to the Servodriver is turned ON.

## 7-7 ABS Soft Reset

ABS Soft Reset is used to perform fine adjustment of the origin position in servo lock status after ABS Initialization has been executed.

Unlike ABS Initialization, the Absolute Encoder does not have to be set up.

### Procedure

- 1, 2, 3...**
  - Check that the pin 1 of the DIP switch on the rear of the MC Unit is set to OFF. If this pin is set to ON, an error will occur when ABS Soft Reset is executed.
  - Execute ABS Soft Reset.  
As is the case with ABS Initialization, processing can be executed either from the data area interface by using a ladder program or by using the Teaching Box.

#### Using a Ladder Program

Turn ON the ABS Soft Reset Bit in manual mode when the Busy Flag is OFF. Leave this bit ON until the Busy Flag turns ON.

#### Using the Teaching Box

Select **6. ABS SOFT RSET** from the extension mode menu, and execute processing.

- Note**
- Execute ABS Soft Reset when the current position is within the range of  $\pm 32767$  pulses. Otherwise, an error will occur.
  - Refer to *6-5 Interface Bit Specifics* for details on a timing chart and ladder program example of the ABS Soft Reset Bit. Refer to the *Teaching Box Operation Manual* for details on Teaching Box operation.

ABS Soft Reset can be executed regardless whether the servo lock is applied or released.



ABS Soft Reset will reset the current position to zero and store the Absolute Encoder Soft Reset value (correction value for zero reset) in the machine parameter area.

**Note** After ABS Soft Reset is completed, back up the system parameters using the MC Support Software.

The ABS Soft Reset values can be displayed using the MC Support Software, but they cannot be changed.

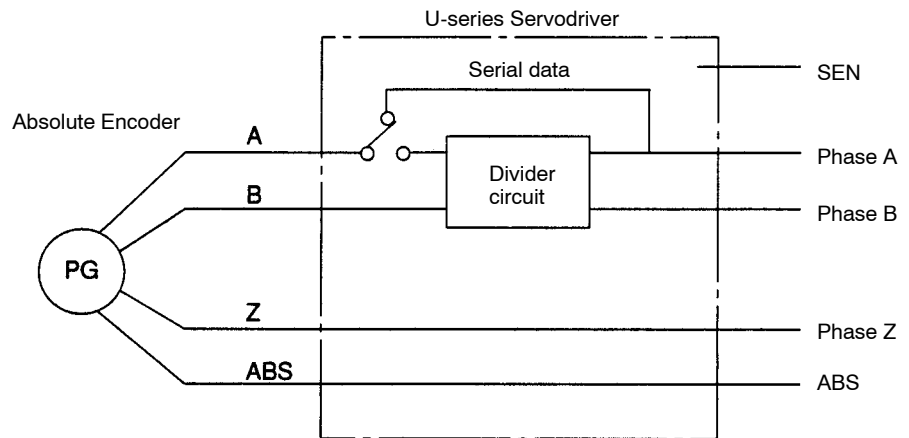
Do not perform ABS Soft Reset while system parameters are being downloaded. Otherwise, the Absolute Encoder Software Reset values will not be stored properly.

## 7-8 Absolute Encoder Interface Specifications

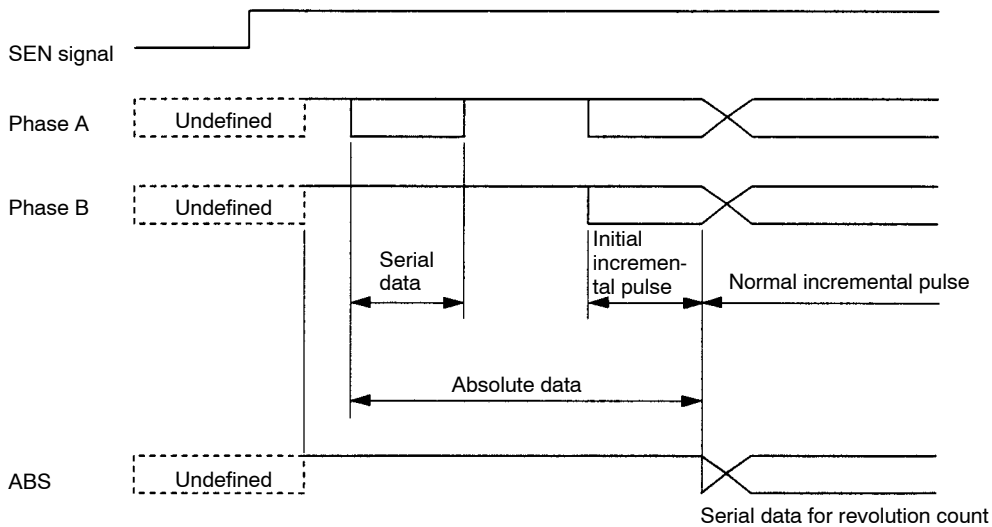
This section describes the interface specifications used for OMRON U-series Servodrivers with absolute encoder specifications.

### 12-bit Absolute Encoder Output Specifications

A 12-bit absolute encoder outputs data from phase A, phase B, phase Z, and ABS. It has the following system configuration.



- 1, 2, 3...**
1. Absolute data is output as serial data from phase A by changing the SEN signal level from low to high.
  2. Initial incremental pulses are output from phase A and phase B (90° differential two-phase pulses).
  3. Output operation (90° differential two-phase pulses) similar to that of ordinary incremental encoders is performed.
  4. The number of revolutions is output as serial data from ABS. The MC Unit does not receive ABS output.

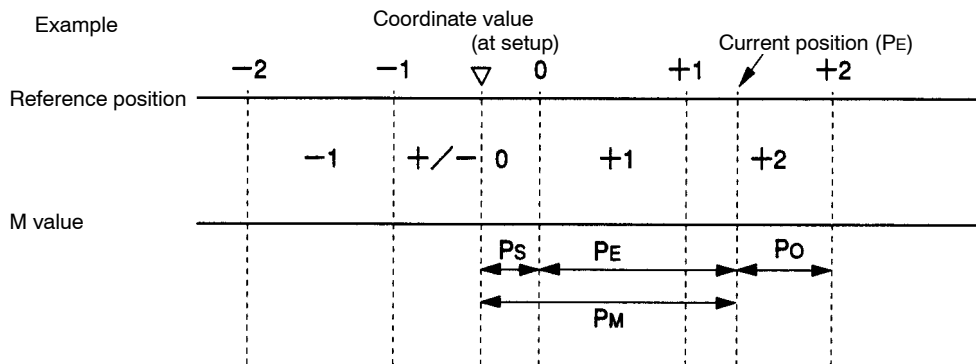


**Contents of Absolute Data**

- **Serial Data:** Indicates how many revolutions the motor shaft has rotated from the reference position (specified at setup).
- **Initial Incremental Pulse Count:** Indicates the number of pulses generated when the motor rotates from the origin position to the current position of the motor shaft at a maximum speed of about 4,900 r/min.

The current position (PE) can be determined using the following formula, where M is serial data (number of motor revolutions), PO is the number of initial incremental pulses counted, and R is the number of output pulses per motor shaft revolution that is set for the divider circuit.

$$PE = M \times R + PO$$



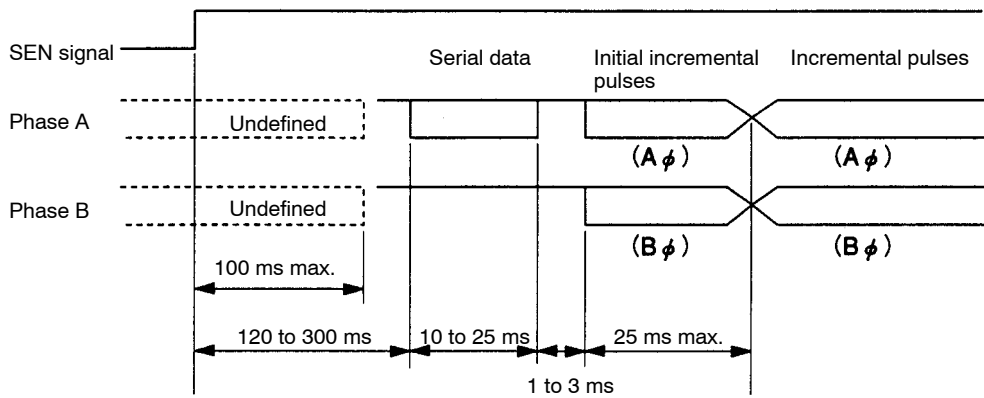
- PE: Present value (servo) read by encoder
- M: Number of revolutions (servo)
- R: Number of pulses per encoder revolution (4,096 pulses)
- PO: Number of initial incremental pulses (servo) read by encoder (normally, negative value)
- PS: Number of initial incremental pulses read at setup (normally, a negative value, which will be then stored in the MC Unit)
- PM: Present value of the system

$$PE = M \times R + PO$$

$$PM = PE - PS$$

**Absolute Data Reception Procedure**

Absolute data is processed in the following sequence.



- 1, 2, 3...**
1. The SEN signal level is set to high.
  2. After 100 ms, the system enters serial data reception wait status.
  3. The system receives 8 bytes of serial data.
  4. The system enters normal incremental operation status about 50 ms after the last serial data is received.

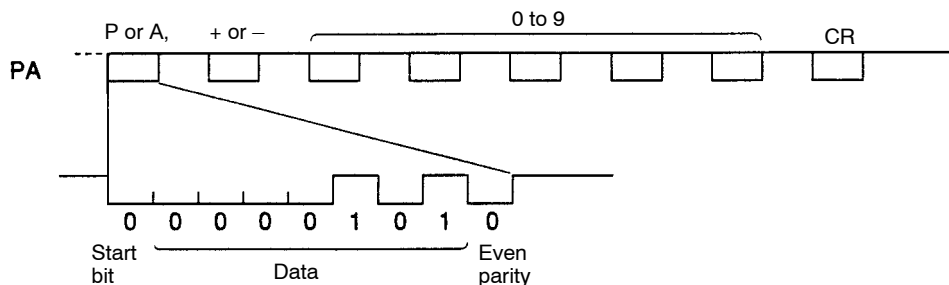
**Phase-A Serial Data Specifications for Absolute Data**

The following table shows phase-A serial data specifications for absolute data.

**Phase-A Serial Data Specifications**

Data transmission method	Asynchronous
Baud rate	9,600 bps
Start bit	1 bit
Stop bit	1 bit
Parity	Even
Character code	ASCII 7 bits
Data format	8 characters (P or A) (+ or -) (0 to 9) x 5 digits (CR)

**Serial Data**



During normal operation, the number of revolutions (5 digits) from the reference position (specified at setup) is output as serial data in the above format.

Zero revolutions are output as "P+00000(CR)" or "P-00000(CR)." The number of revolutions exceeding ±99999 will not be output correctly. In other words, after exceeding ±99999, the number of revolutions will return to 000000.



# SECTION 8

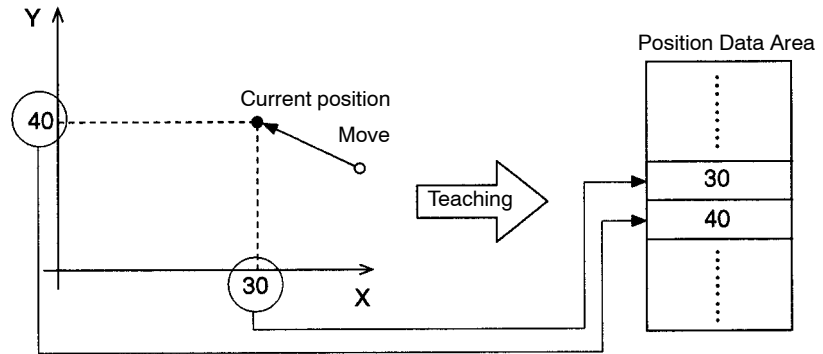
## Teaching

This section describes teaching, which reads each axis' current position as position data.

8-1	Introduction .....	194
8-2	The Teaching Function .....	194
8-3	Setting the Teaching Address .....	195
8-4	Performing Teaching .....	196
8-5	Errors .....	197
8-5-1	Teaching Errors .....	197
8-5-2	Teaching Address Over Errors .....	197

## 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 and Y).

### 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 6-5 Interface Bit Specifics for more details on these formats.

### Teaching Method

Teaching can be performed in two ways.

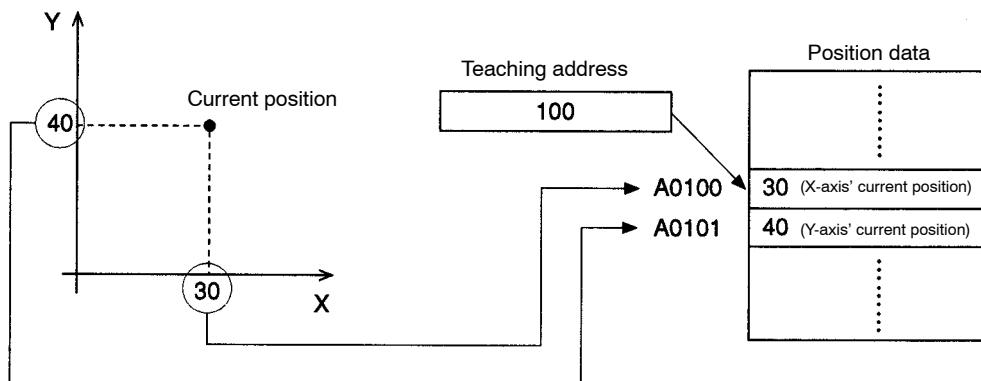
- Teaching can be performed from the ladder program. Execute teaching through the PC data area interface.
- Teaching can be performed from the Teaching Box.

## 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.

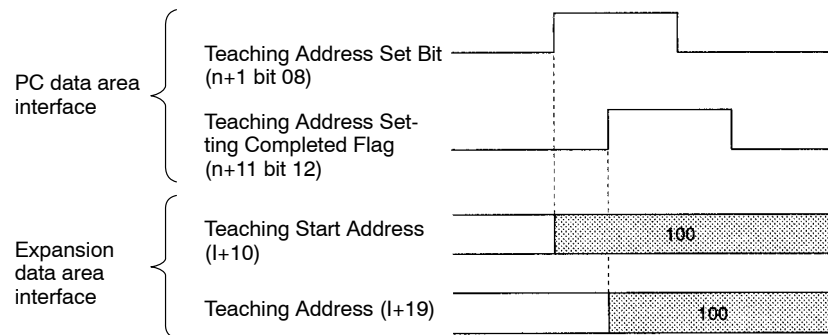
### 8-3 Setting the Teaching Address

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. The teaching address is set in the expansion data area. The following example shows how to set the teaching address.

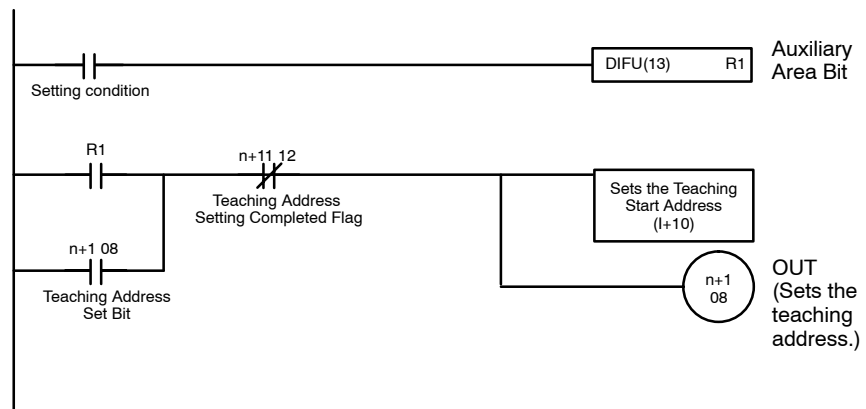
Set the desired value in the Teaching Start Address (DM word I+10). Next, turn ON the Teaching Address Set Bit for task 1 (n+1 bit 08); keep this bit ON until the Teaching Address Setting Completed Flag (n+11 bit 12) is turned ON.

The timing chart and ladder program for this operation are shown below. Refer to 6-5 *Interface Bit Specifics* for more details.

#### Timing Chart



#### Example Program



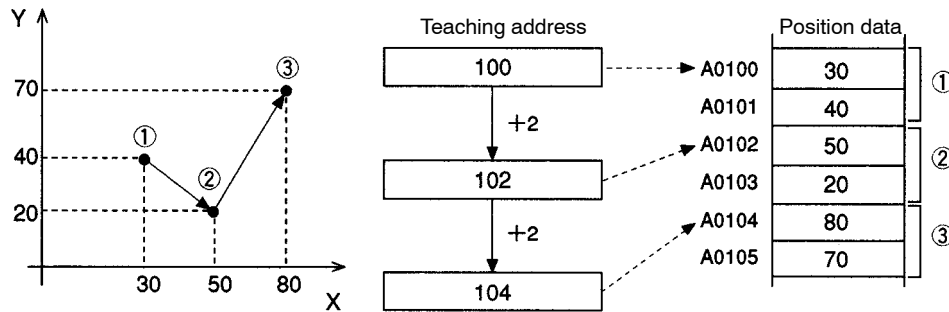
#### Using the Teaching Box

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.



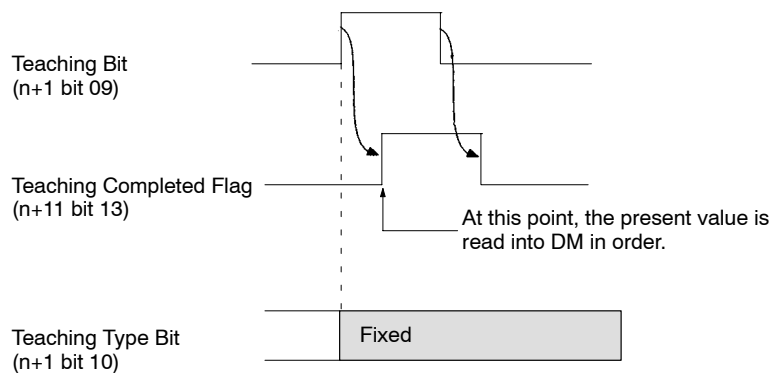
In the example above, teaching is performed three times at points 1, 2, and 3. The X, Y current position data is stored and the teaching address is automatically incremented to the next open position data address.

### Performing Teaching from the Ladder Program

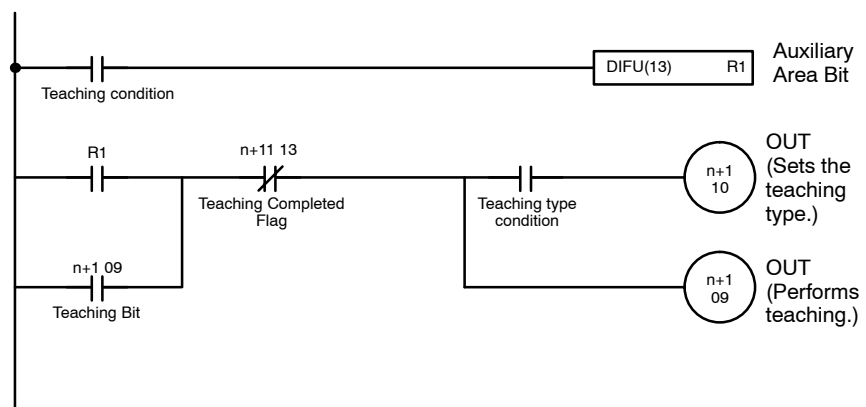
Turn ON the Teaching Bit for task 1 (n+1 bit 09) and keep this bit ON until the Teaching Completed Flag (n+11 bit 13) is turned ON.

The timing chart and ladder program for this operation are shown below. Refer to 6-5 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.

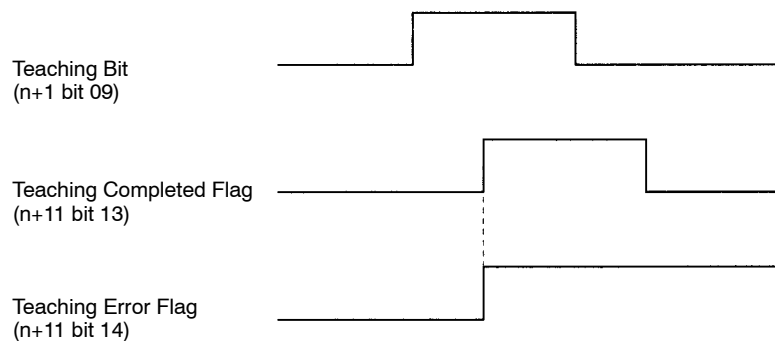
## 8-5 Errors

### 8-5-1 Teaching Errors

An error will occur and the Teaching Error Flag (n+11 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 6-5 Interface Bit Specifics for more details.

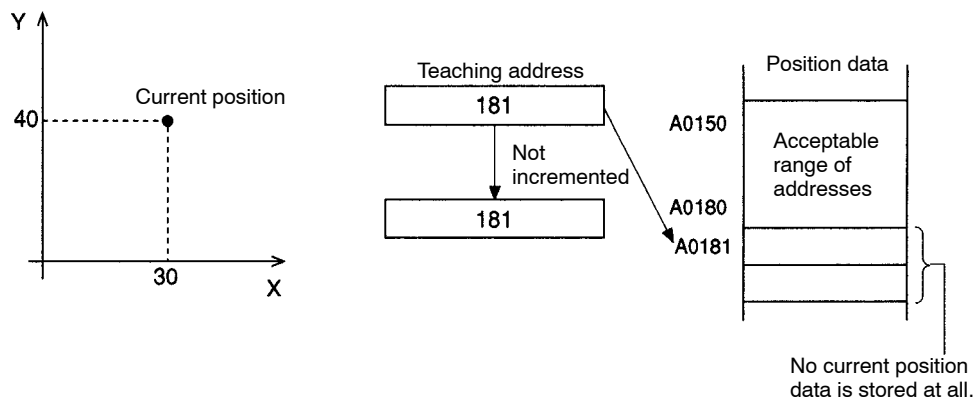
### 8-5-2 Teaching Address Over Errors

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.

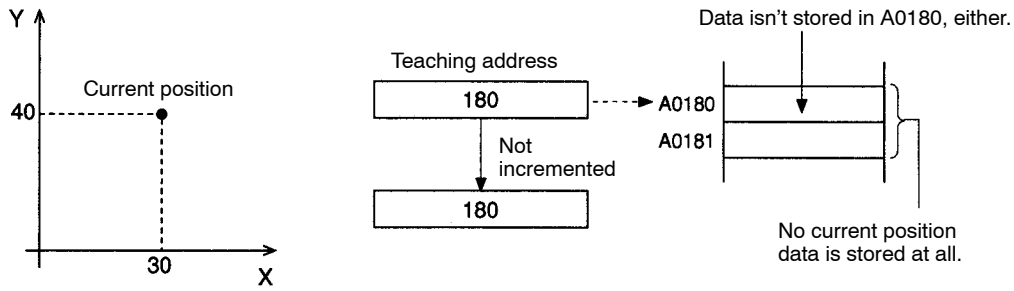
**Example 1**

In this example, the teaching address is already greater than 180 when teaching is performed.



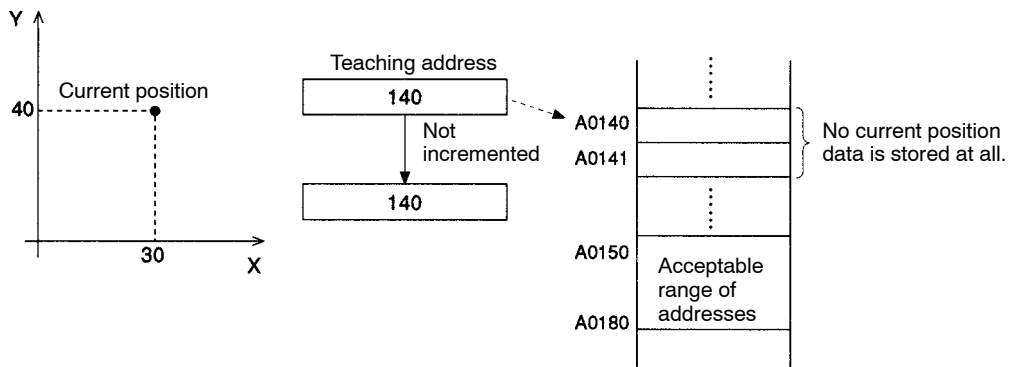
**Example 2**

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.



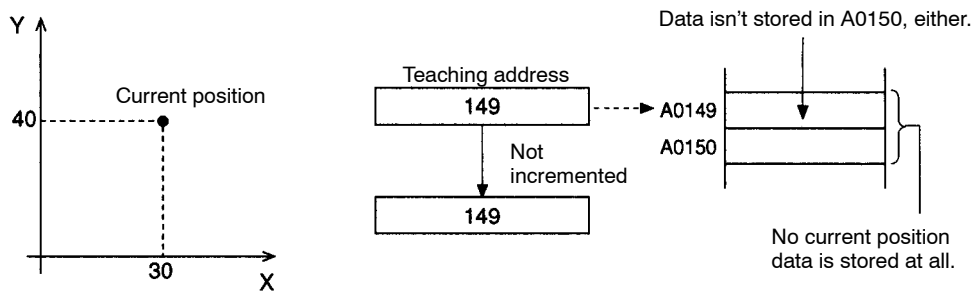
**Example 3**

In this example, the teaching address is less than 150 when teaching is performed.

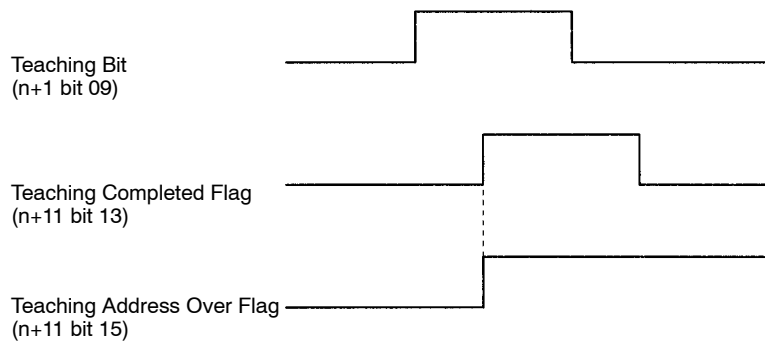


**Example 4**

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 *9-10 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 .....	200
9-2	Positioning with Linear Interpolation .....	201
9-3	Positioning with Linear and Circular Interpolation .....	202
9-4	Indirect Addressing with Registers .....	203
9-5	Using the Workpiece Origin Offset .....	204
9-6	Changing the Interpolation Acceleration Time .....	205
9-7	Calculating Position Data .....	206
9-8	Stopping a Program with a General Input .....	208
9-9	Stopping the Program and Substituting Position Data .....	209
9-10	Executing MC Programs from the Ladder Program .....	210

## 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 (Output M code 20.)
  2. X250 Y150
  3. X300 Y200 (Output M code 700.)
  4. Return to reference origin.

### Sample Program

The following sample program performs the positioning operations listed above.

```

N000 P001 XY
N001 G00 X100 Y50 M20
N002 G00 X250 Y150
N003 G04 2
N004 G00 X300 Y200 M700
N005 G26 XY
N006 G79

```

### Explanation

Block N000 declares the program number and axes being used.

#### N001

Positions the axes to (X100, Y50) 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) by PTP control.

#### N003

Waits for 2 seconds.

#### N004

Positions the axes to (X300, Y200) by PTP control. When positioning is completed, M code 700 is output and the next block is executed without waiting for the reset.

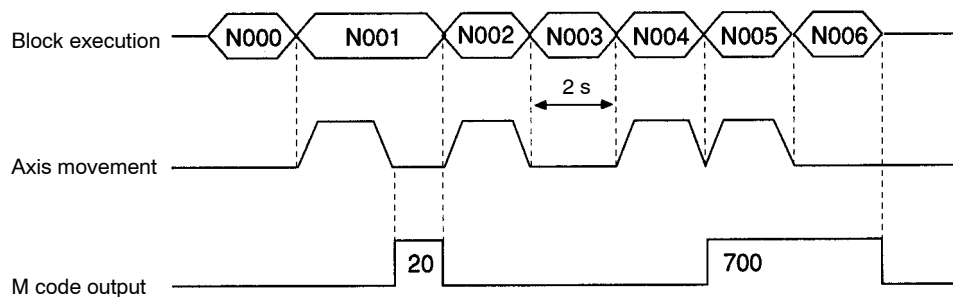
#### N005

Returns the X and Y axes to the reference origin.

#### N006

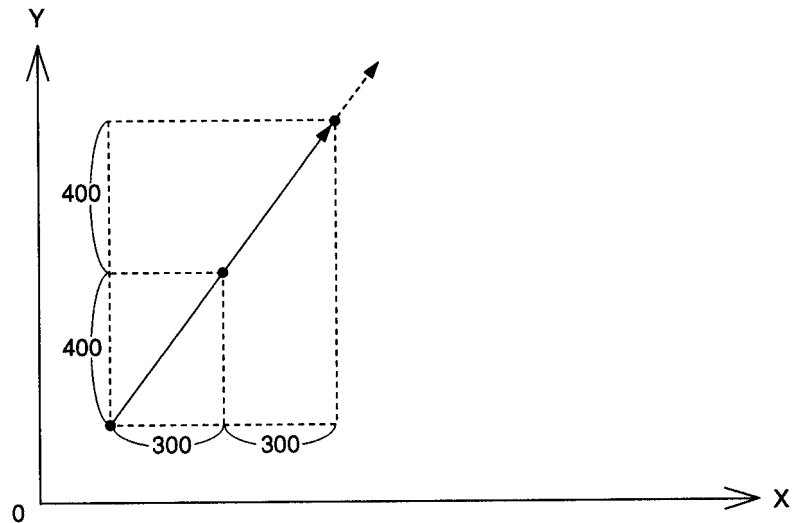
Ends the program. When M code 700 is being output, it is cleared by the PROGRAM END function.

### 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 #5
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 5 (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

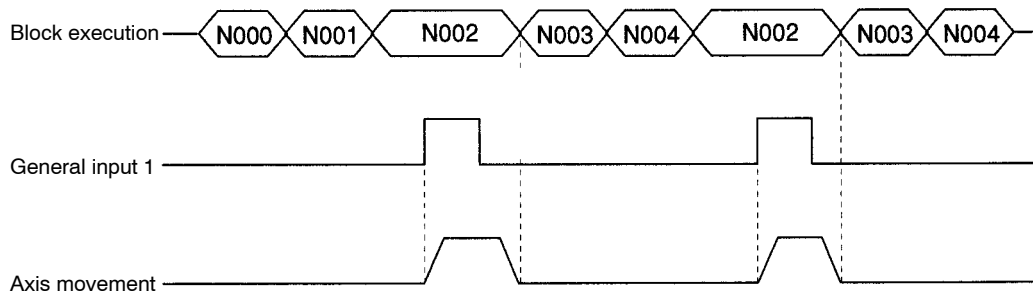
Jumps to block N002. Block N004 will jump to N002 20 times (performing 21 positioning operations).

#### N005

Ends the program.

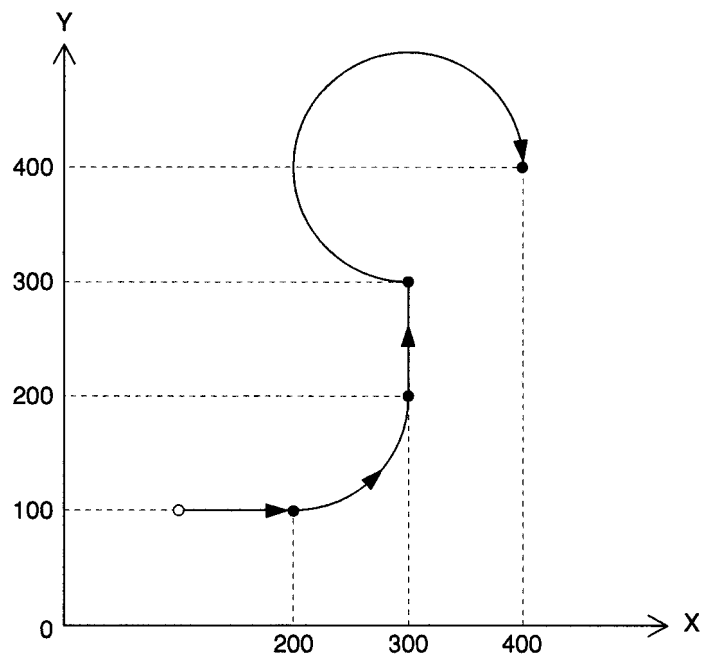
**Note** The operations above are performed in Stop mode not Pass mode because an optional input is specified.

## 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 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 XY
N001 G11
N002 G63 E1=100
N003 G63 E2=101
N004 G01 X(E1) Y(E2) F100
N005 G60 E1=E1+2
N006 G60 E2=E2+2
N007 G75 3
N008 G70 N004/L3
N009 G79
```

**Explanation**

Block N000 declares the program number and axes being used.

**N001**

Specifies Stop mode.

**N002 through N003**

Set the initial position data addresses in the registers.

**N004**

Reads the position data from the addresses specified in the registers and moves the axes to this point by linear interpolation.

**N005 through N006**

Updates the contents of the registers.

**N007**

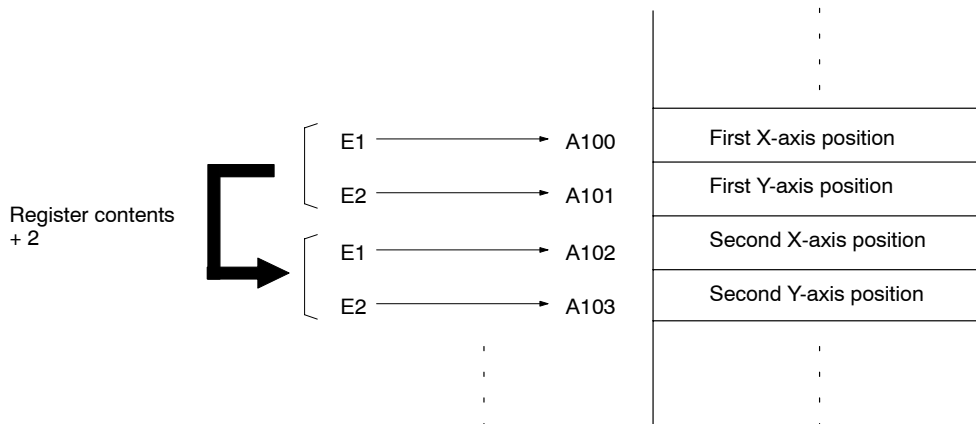
If optional input 3 is ON, the next block will be skipped and the program will end.

**N008**

Jumps to block N004 and repeats the positioning operation 3 times.

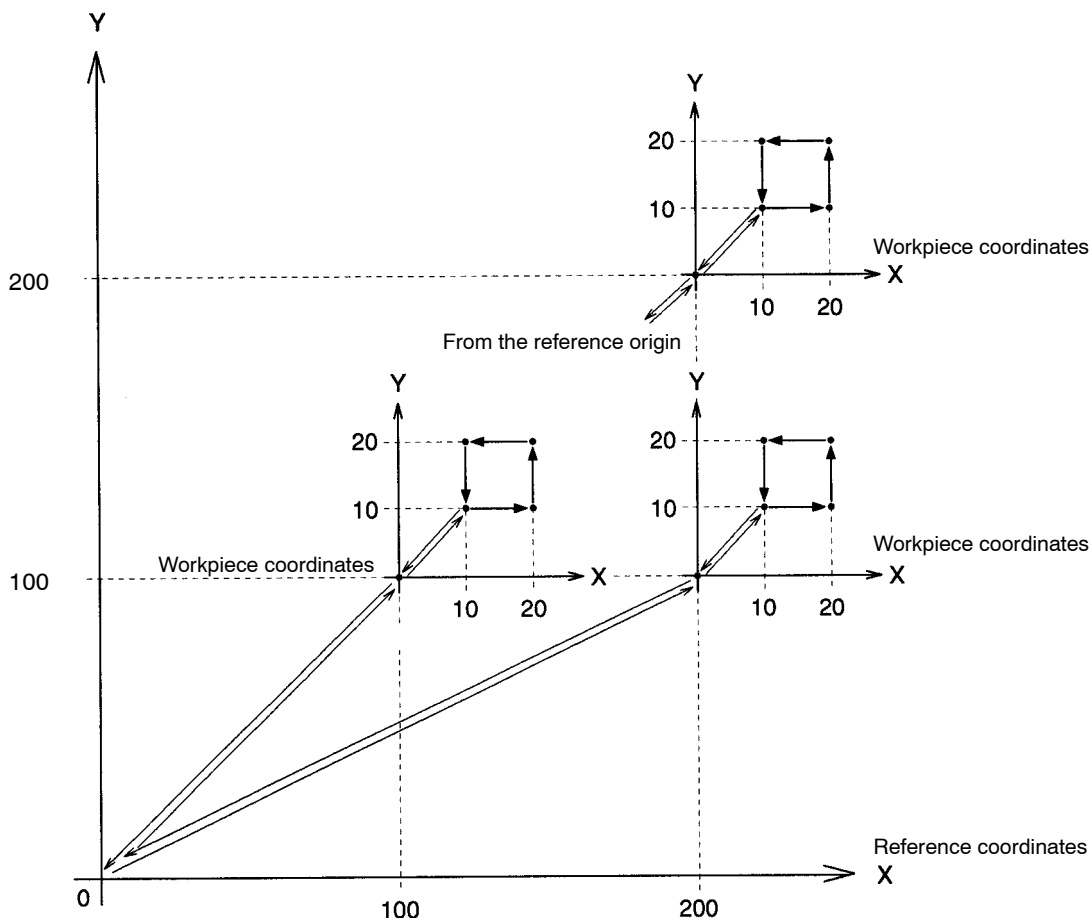
**N013**

Ends the program.



### 9-5 Using the Workpiece Origin Offset

Positioning for the same pattern can be performed any number of times by repeatedly changing the workpiece origin offset and calling the subprogram. Using the coordinate system is useful particularly for the absolute operation.



**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.

## 9-6 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**

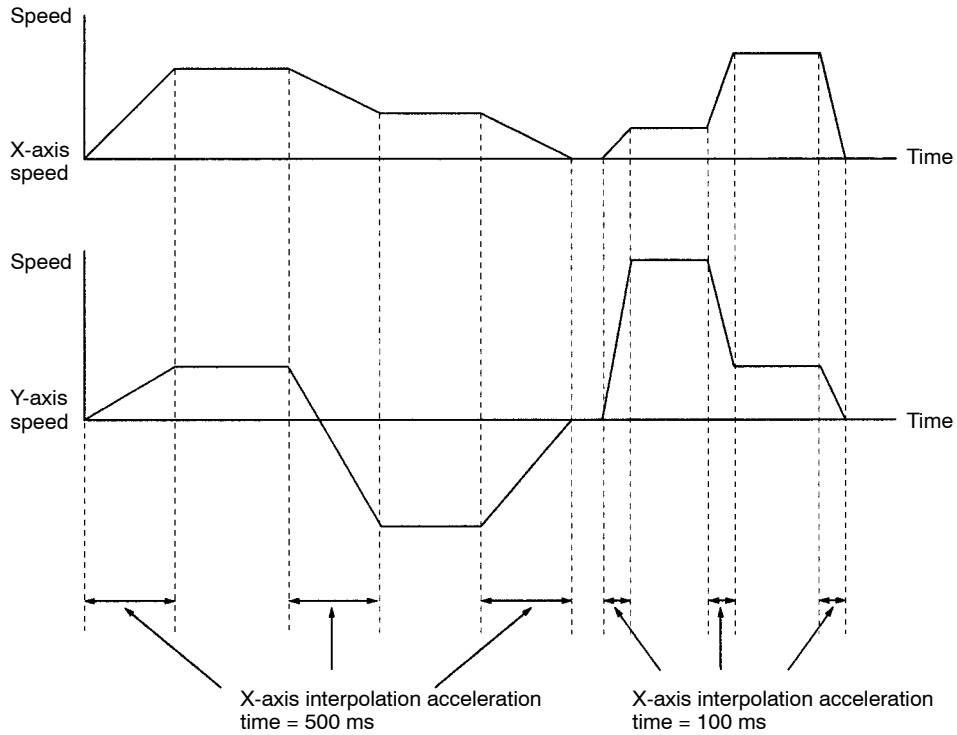
Moves to (X3000, Y1000) by linear interpolation with an acceleration time of 100 ms.

**N007**

Moves to (X4000, Y1500) in Pass mode with an acceleration time of 100 ms.

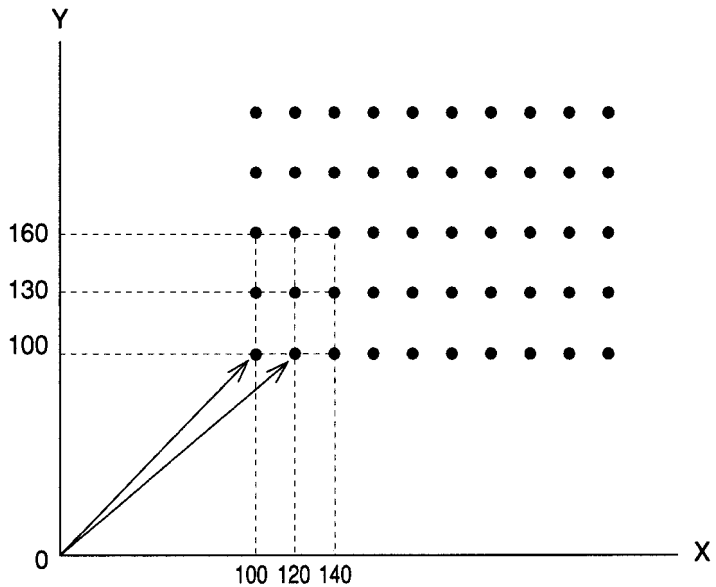
**N008**  
Ends the program.

**Timing Charts**



**9-7 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/L9
N009 G63 A1000=100
N010 G60 A1001=A1001+30
N011 G60 E00=E00+1
N012 G71 N005/E0 ! 5
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 9 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.

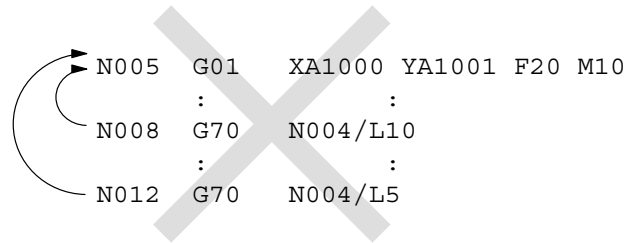
**N012**

Jumps to N004 as long as the content of E0 isn't 5. When E0=5, 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/L4
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/L9
N005 G73
```

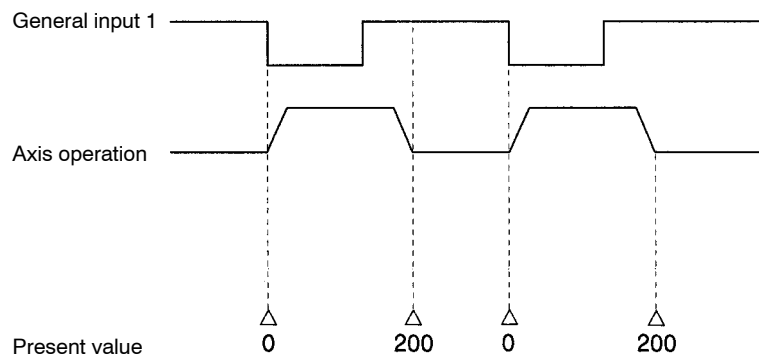
**Explanation**

Blocks N005 through N008 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.

## 9-8 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 5
N003 G54 X0
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.

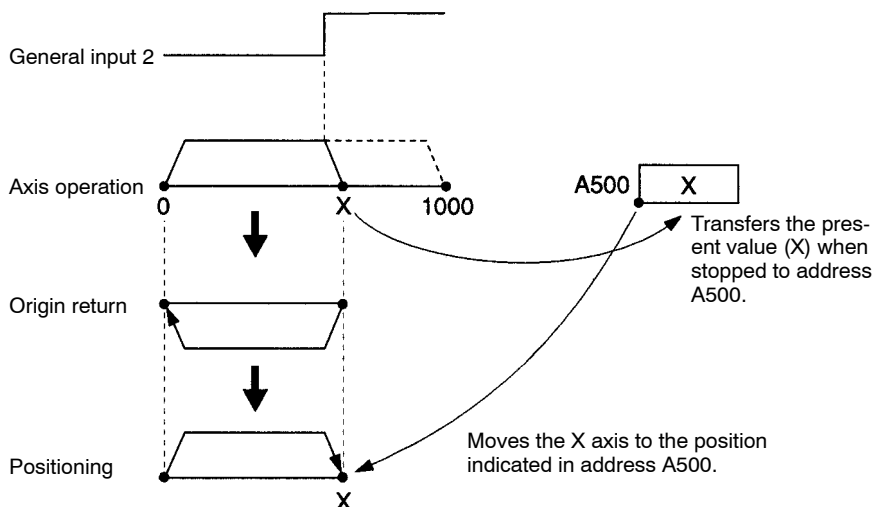
**N006**

Ends the program.

## 9-9 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 6
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.

## 9-10 Executing MC Programs from the Ladder Program

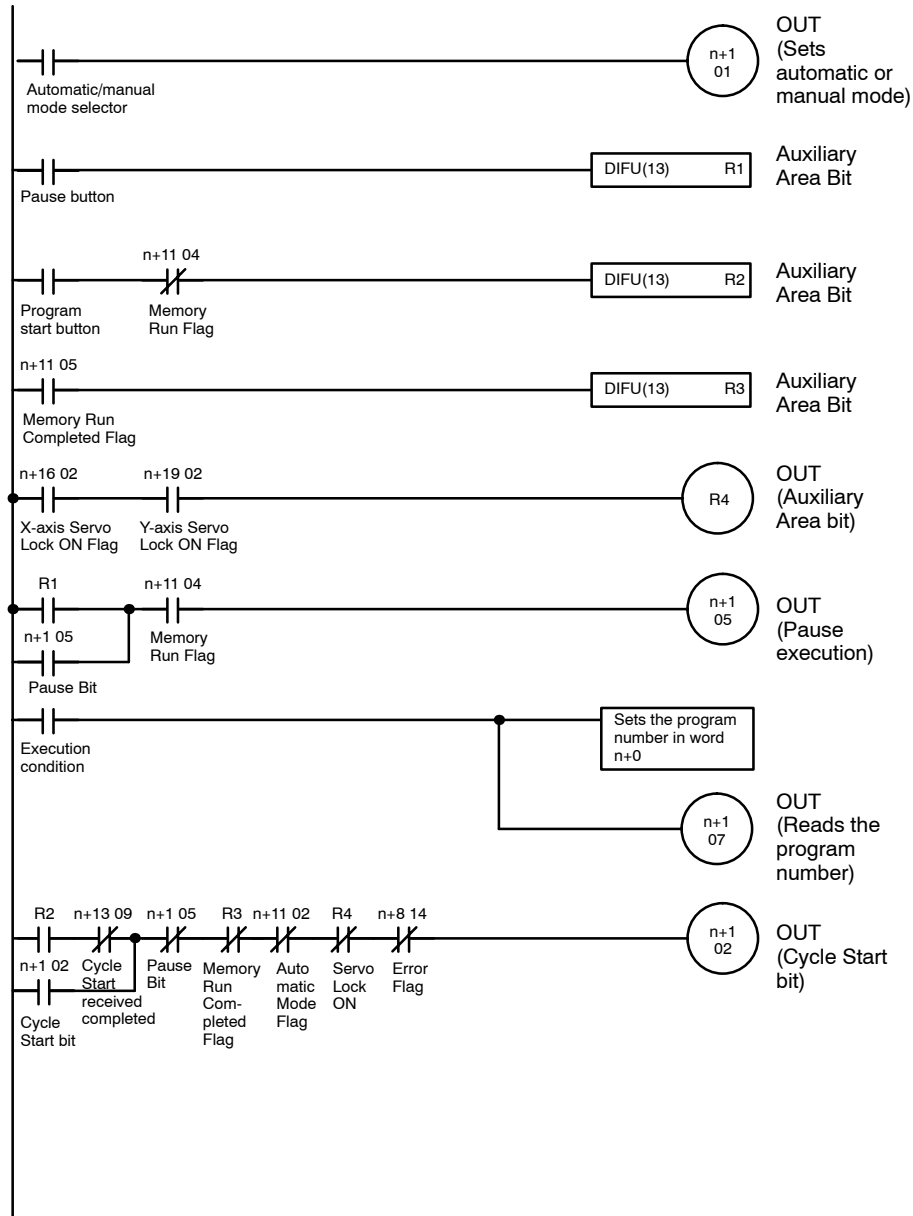
This section describes ladder programs that will execute G-language MC programs. Use the following procedure to execute MC programs.

- 1, 2, 3...**
1. Set the MC Unit to automatic mode. (Turn ON bit 01 of IR word n+1.)
  2. Set the program number of the desired MC program. (IR word n+0)
  3. Read the program number from IR word n+0 to the MC Unit. (Turn ON bit 07 of IR word n+1.)
  4. Turn ON the Cycle Start Bit to execute the MC program specified in IR word n+0. (Turn ON bit 02 of IR word n+1.)

This procedure executes task 1. The following table shows the equivalent IR Area control bits and IR words for task 2.

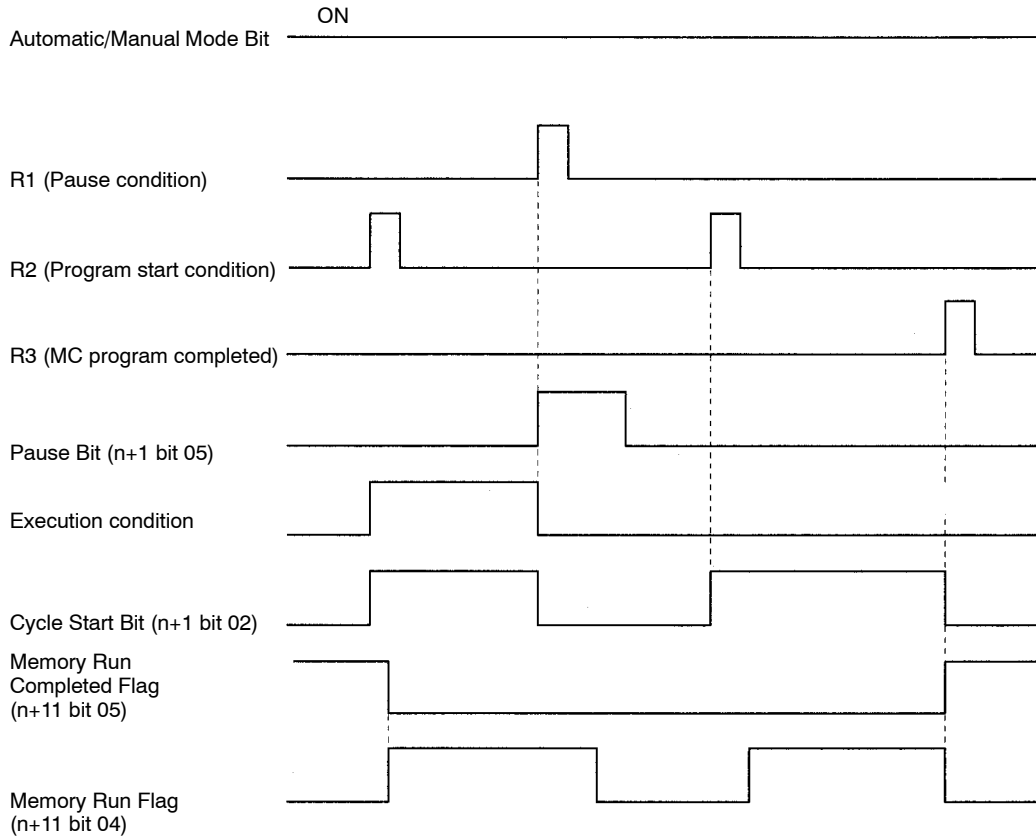
Bit/word	Task 2
Automatic/Manual Mode Bit	n+3: bit 01
Program number	n+2
Program Number Read Bit	n+3: bit 07
Cycle Start Bit	n+3: bit 02

**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

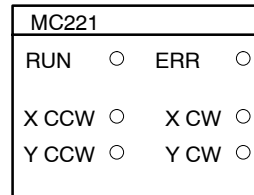
## Troubleshooting

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

10-1	Error Indicators .....	214
10-2	System Errors: Error Codes 0001 to 0017 .....	215
10-3	Task Errors: Error Codes 0020 to 0045 .....	216
10-4	Axis Errors: Error Codes 0060 to 0088 .....	219

# 10-1 Error Indicators

The error indicators are located on the front of MC Unit, as shown in the following diagram.



RUN	ERR	XCCW	XCW	YCCW	YCW	Error	Error Processing
OFF	OFF	OFF	OFF	OFF	OFF	Watchdog timer timeout error I/O table verification error	The MC Unit's watchdog timer has timed out. Replace the MC Unit. Create an I/O table and try again.
ON	OFF	ON/OFF	ON/OFF	ON/OFF	ON/OFF	No error	---
ON	ON	OFF	OFF	OFF	OFF	MC Unit error	An error has occurred in the MC Unit itself. Check details of the error with the PC, Teaching Box, or MC Support Software.
ON	ON	ON	OFF	OFF	OFF	EEPROM error	An error has occurred in EEPROM, the MC Unit, or in the MC Unit's RAM. Replace the MC Unit. Replace the MC Unit.
ON	ON	OFF	ON	OFF	OFF	RAM error	
ON	ON	OFF	OFF	ON	OFF	CPU error	

**Note** For error codes larger than 1000 that might occur while using the MC Support Software, refer to *Section 18 Error Processing* of the *MC Support Software Operation Manual (W256)*.

Refer to *Sections 10-2 to 10-4* for details about the error codes. The first error code will be output to the n+9 words of the work area.

## 10-2 System Errors: Error Codes 0001 to 0017

An error code will be output to word I+12 of the expansion data area when a system error results. When a system error occurs, the Error Flag of the interface bit (word n+8 bit 14) will be turned ON and an error code will be output in 4-digit BCD to word I+12. Refer to the following table for system errors in detail. (The error message in parentheses will be displayed on the MC Support Software and the Teaching Box.)

Code	Error	Error Processing
0001	System parameters destroyed (SYS PARA CORRUPT)	<p>The system parameters have been destroyed. (See note.) Download the system parameters from the MC Support Software again and write them to the flash memory, and then turn the power OFF and ON. The system parameters can be destroyed by any of the following:</p> <ul style="list-style-type: none"> <li>The memory data might have been destroyed by noise.</li> <li>The PC (MC Unit) might have been turned OFF while data was being written to the flash memory.</li> </ul> <p>This error can't be cleared by an error reset.</p>
0002	Wrong number of tasks (TASK NOT CONSIST)	<p>The number of tasks registered in the system parameters doesn't match the number register in the program. (See note.) 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 and write the program(s) to the flash memory. 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:</p> <ul style="list-style-type: none"> <li>The system parameters that were downloaded had a different number of tasks registered.</li> <li>The memory data might have been destroyed by noise.</li> <li>The PC (MC Unit) might have been turned OFF while data was being written to the flash memory.</li> </ul> <p>This error can't be cleared by an error reset.</p>
0003	Task 1 program destroyed (TASK 1 PROG CORR)	<p>Task 1's program has been destroyed. (See note.) Delete all task 1 programs using the MC Support Software and the error will be cleared automatically, then download the programs again and write them to the flash memory. The programs can be destroyed by any of the following:</p> <ul style="list-style-type: none"> <li>The memory data might have been destroyed by noise.</li> <li>The PC (MC Unit) might have been turned OFF while data was being written to the flash memory.</li> </ul> <p>This error can't be cleared by an error reset.</p>
0004	Task 2 program destroyed (TASK 2 PROG CORR)	<p>Task 2's program has been destroyed (see note). See the error processing described for error code 0003.</p>
0010	Deceleration stop (DECEL STOP)	<p>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.</p>
0011	Setup error (SYSTEM SET ERROR)	<p>System parameters with new Unit or Machine parameters were downloaded. Write the system parameters to the flash memory with the MC Support Software or Teaching Box and then turn the power OFF and ON. 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.</p>
0012	Illegal data (DATA ERROR)	<p>Position data was destroyed when an attempt was made to receive it using the data receive instruction relay or destroyed position data was received with the IORD instruction. The error can be cleared by an error reset, but transfer the position data again. The position data can be destroyed by the following:</p> <ul style="list-style-type: none"> <li>The memory data might have been destroyed by noise.</li> </ul>

Code	Error	Error Processing
0014	IORD syntax error (IORD SYNTAX ERR)	One of the following errors occurred when the IORD instruction was executed with the C200HX, C200HG, or C200HE. The MC address is not within the acceptable range. The MC address is not BCD. The number of transfer data items is not within the acceptable range. The number of transfer data items is not BCD. The number of transfer data items is not in 4-byte units. Set the right data and executed the IORD instruction again.
0015	IOWR syntax error (IOWR SYNTAX ERR)	One of the following errors occurred when the IOWR instruction was executed with the C200HX, C200HG, or C200HE. The MC address is not within the acceptable range. The MC address is not BCD. The number of transfer data items is not within the acceptable range. Example: When rewriting a 4-byte parameter, 8 or 12 bytes were specified. The number of transfer data items is not in 4-byte units. Set the right data and executed the IOWR instruction again.
0016	IOWR data error (IOWR DATA ERR)	One of the following errors occurred in data to be written to the MC Unit when the IOWR instruction was executed with the C200HX, C200HG, or C200HE. The data is not BCD. The data is not within the acceptable range. Set the right data and executed the IOWR instruction again.
0017	Flash memory error (FLASH MEMORY ERR)	Data was not saved in the flash memory. Save the data in the flash memory again. The error can be cleared when the data is written normally. Replace the MC Unit if the error persists.

**Note** Checked when power is turned ON.

## 10-3 Task Errors: Error Codes 0020 to 0045

An error code will be output to words I+13 to I+14 of the expansion data area when a task 1 or 2 error results. When a system error occurs in task 1 or 2 the error code will be output to word I+13 or I+14 respectively. In this table, the term “program” refers to the G-language MC program. The error message in parentheses will be displayed on the MC Support Software and 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. The position data specified with G72 is not within a range of 500 to 999. Specify a 4-digit BCD program number from 0000 to 0999. Set the position data specified with G72 within a range of 500 to 999.
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.
0022	Program axis declaration error (AXIS SPEC ERR)	The axes used in the program are not 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 and write it to the flash memory, and then turn the power OFF and ON. The program could have been destroyed by any of the following: The memory data might have been destroyed by noise. The PC (MC Unit) might have been turned OFF while data was being written to the flash memory.

Code	Error	Error Processing
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 and write it to the flash memory, and then turn the power OFF and ON. The program could have been destroyed by any of the following:  The memory data might have been destroyed by noise.  The PC (MC Unit) might have been turned OFF while data was being written to the flash memory.
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 (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 long 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 long between the radius of the center and the start point and the radius of the center and the end point.  Correct the program or the position data.
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, G03, or G31 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.
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.
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.

Code	Error	Error Processing
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 offset value or preset value was converted with the minimum setting unit from the system parameters. Correct the program or the position data. Check the setting for the minimum setting unit parameter in the system parameters when the error occurs in a present value preset from the G53 or 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 interpolation, circular interpolation, or interrupt inching. Correct the program or the position 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 and write it to the flash memory, and then turn the power OFF and ON. The intermediate code error could have been caused by any of the following:  The memory data might have been destroyed by noise.  The PC (MC Unit) might have been turned OFF while data was being written to the flash memory.
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, position 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 position 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.
0045	Second speed over (SECOND SP OVER)	The value of speed command 2 was the same as or larger than the value of speed command 1 when G30 was executed. Set so that the value of speed command 1 is larger than the value of speed command 2.

## 10-4 Axis Errors: Error Codes 0060 to 0088

An error code will be output to words I+15 to I+16 of the expansion data area when an X or Y-axis error results. When a error occurs in the X or Y axis, the error code will be output to word I+15 or I+16 respectively. (The error message in parentheses will be displayed on the MC Support Software and the Teaching Box.)

Code	Error	Error Processing
0060	Driver alarm encountered (DRV ALARM INPUT)	The driver alarm input was turned ON or an attempt was made to operate the axis or execute a PRESENT VALUE CHANGE function (G54) or ORIGIN UNDEFINED function (G29) 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 <i>Appendix B Additional Origin Search Patterns</i> 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 <i>Appendix B Additional Origin Search Patterns</i> 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, position 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, position data, or command value from the PC.

Code	Error	Error Processing
0069	Origin point not confirmed (ORIG NOT ESTABL)	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.
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 preset value so that it will be 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.
0073	ABS encoder error (ABS I/F ERR)	No absolute value data was sent from the Absolute Encoder. Communications error occurred while receiving data. Possible causes are as follows; Encoder cable was disconnected or broken. Encoder is faulty. MC Unit is faulty. Check the encoder cable and try again. Remove any possible causes of noise. If the same error occurs again, either the encoder or MC Unit is faulty. Replace the encoder or MC Unit.
0074	ABS data error (ABS DATA ERROR)	Absolute value data was properly (without any communications error) sent from the absolute encoder, but the data was illegal. Possible causes are as follows; Encoder cable was disconnected or broken. Encoder is faulty. MC Unit is faulty. Check the encoder cable and try again. Remove any possible causes of noise. If the same error occurs again, either the encoder or MC Unit is faulty. Replace the encoder or MC Unit.
0075	ABS set prohibit error (ABS SET PROHIBIT)	Set pin 1 of the DIP switch on the MC Unit to OFF.
0076	ABS initialization invalid error (ABS INI SET INVL)	Set to the servo free state.
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.
0078	ABS rotation counter error (ABS ROTA CNT ERR)	An attempt was made to set the current position from the absolute encoder, but the current position couldn't be defined due to the following reasons; Rotation of the absolute encoder was beyond $\pm 99999$ . Rotation of the absolute encoder was within $\pm 99998$ , but the value with the compensation value added exceeded the counter value. If the driver setup has not been completed yet, complete the setup and try again. If the driver setup has already been completed, shift the axis to decrease the rotation of the absolute encoder and try again.
0079	ABS encoder error (ABS ERR)	Power was not supplied to the absolute encoder or the absolute encoder was faulty. Connect a battery to the absolute encoder. If the battery has already been connected, the absolute encoder may be faulty. Replace the absolute encoder.
0080	ABS initialization range over (ABS INT SET OVR)	The ABS data exceeded the range from $-32767$ to $+32767$ pulses. If the driver setup has not been completed yet, complete the setup and try again. If the driver setup has already been completed, shift the axis so that the ABS data falls within $\pm 32767$ pulses and try again.



Code	Error	Error Processing
0081	ABS soft reset range over (ABS SFT RSET OVR)	The current position data exceeded the range from -32767 to +32767 pulses. Shift the current position to a position within $\pm 32767$ pulses and try again.
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 <i>Appendix B Additional Origin Search Patterns</i> 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 <i>Appendix B Additional Origin Search Patterns</i> 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 <i>Reverse-mode Origin Searches 12</i> in <i>Appendix B</i> 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 <i>Reverse-mode Origin Searches 12</i> in <i>Appendix B</i> 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 275,000 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. Take appropriate countermeasures and try again. Remove the causes that have caused the task error.



# SECTION 11

## Maintenance and Inspection

This section describes the maintenance and inspection necessary to ensure proper operation of the MC Unit.

Routine Inspections .....	224
---------------------------	-----

# 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 (Permissible voltage variation: 85 to 132 VAC max.) 200 to 240 VAC (Permissible voltage variation: 170 to 264 VAC max.) 24 VDC (Permissible voltage variation: 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 (in-panel) temperature	0°C to 55°C
	Ambient (in-panel) humidity	10% to 90% RH (no condensation)
	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

## Required Tools

The following tools are recommended when performing an inspection.

- Standard and Phillips-head screwdrivers
- Voltmeter
- Alcohol and a clean towel
- Synchroscope
- 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.

Before resuming operations after MC Unit replacement, be sure to transfer the programs, position data, and system parameters that are required for starting operations again and carry out flash write procedures.

# Appendix A

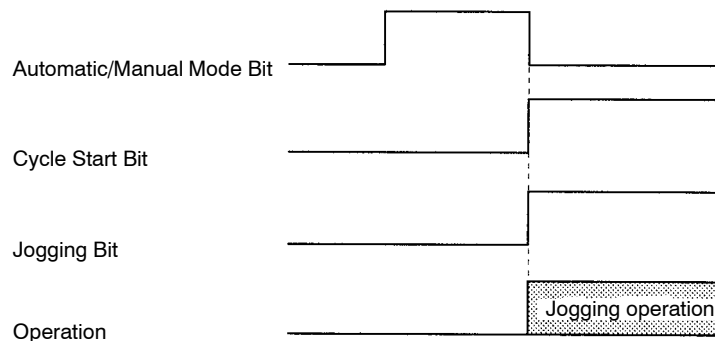
## Control Bit/Flag Timing Charts

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 Chart(s)	Control Bit(s) and Flag(s)	Page
1	Automatic/Manual Mode Bit, Cycle Start Bit, and Jogging Bit	225
2 and 3	Cycle Start Bit and Pause Bit	226
4 to 9	Cycle Start Bit and Forced Block End Bit	226
10 to 12	Cycle Start Bit, Forced Block End Bit, and Pause Bit	228
13 and 14	Automatic/Manual Mode Bit, Cycle Start Bit, and Pause Bit	229
15 and 16	Automatic/Manual Mode Bit, Cycle Start Bit, and Forced Block End Bit	229
17	Cycle Start Bit, Single Block Bit, and Forced Block End Bit	230
18 and 19	Cycle Start Bit and M Code Reset Bit	230
20 and 21	Cycle Start Bit, Forced Block End Bit, M Code Reset Bit, and M code output	231
22	Automatic/Manual Mode Bit and M code output	231
23 and 24	Cycle Start Bit, Pause Bit, M Code Reset Bit, and M code output	232
25	Cycle Start Bit, Forced Block End Bit, and M code output	232
26	Cycle Start Bit, Forced Block End Bit, Task Error Flag, and Task Error Reset Bit	233
27	Cycle Start Bit, Task Error Flag, and Task Error Reset Bit	233
28	Automatic/Manual Mode Bit, Cycle Start Bit, Jogging Bit, and Manual Mode Flag	233
29	Origin Search Bit, Busy Flag, and Automatic/Manual Mode Bit	234
30 and 31	Cycle Start Bit, Forced Block End Bit, and optional input	234
32 to 34	Cycle Start Bit and optional input	235
35 and 36	Forced Block End Bit and optional input	235
37 and 38	Forced Block End Bit, M Code Reset Bit, and M code output	236
39	Pause Bit and optional input	236
40	Pause Bit, M Code Reset Bit, and M code output	237
41	Optional input, M Code Reset Bit, and M code output	237

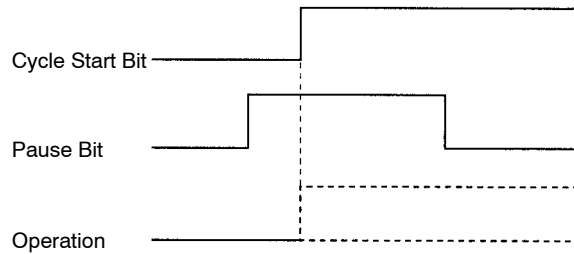
## Timing Charts

### 1. Automatic/Manual Mode Bit, Cycle Start Bit, and Jogging Bit



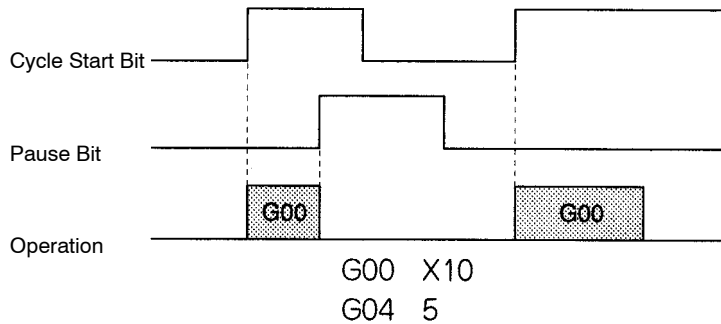
The Jogging Bit signal is received at the same time that manual mode is set. The Cycle Start Bit signal is ignored.

**2. Cycle Start Bit and Pause Bit**



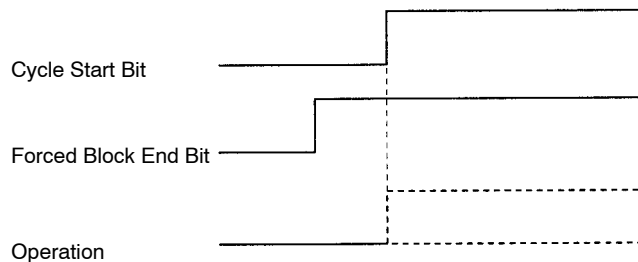
Even though the Cycle Start Bit is turned ON, MC program execution isn't started because the Pause Bit is ON. Program execution will begin if the Pause Bit is turned OFF and the Cycle Start Bit is turned OFF and then ON again.

**3. Cycle Start Bit and Pause Bit**



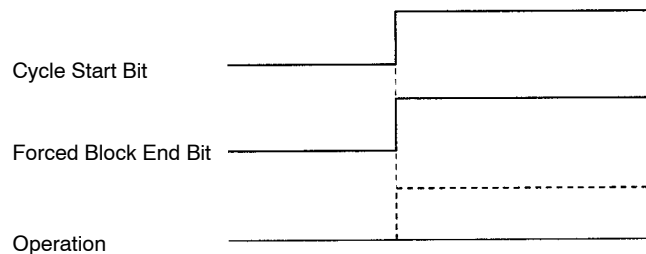
MC program execution is paused by turning ON the Pause Bit. Program execution will continue when the Pause Bit is turned OFF and the Cycle Start Bit is turned ON again.

**4. Cycle Start Bit and Forced Block End Bit**



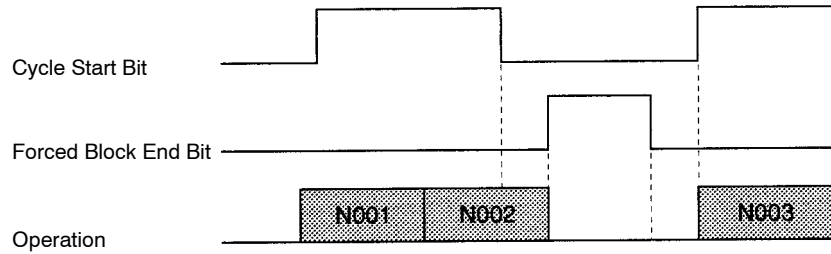
Even though the Cycle Start Bit is turned ON, MC program execution isn't started because the Forced Block End Bit is ON. Operation will begin if the Forced Block End Bit is turned OFF and the Cycle Start Bit is turned OFF and then ON again.

**5. Cycle Start Bit and Forced Block End Bit**



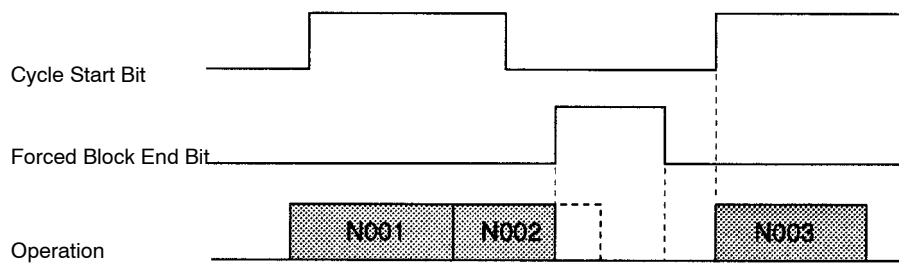
When the Cycle Start Bit and Forced Block End Bit are turned ON at the same time, the Forced Block End Bit takes precedence, so the program isn't executed. Operation will begin if the Forced Block End Bit is turned OFF and the Cycle Start Bit is turned OFF and then ON again.

**6. Cycle Start Bit and Forced Block End Bit**



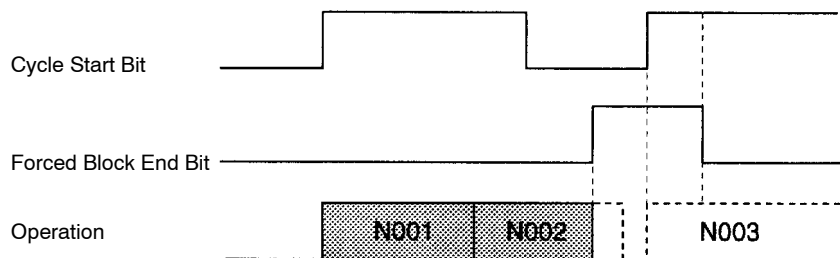
The Forced Block End Bit is turned ON at the end of block N002, but the down-differentiation of the Cycle Start Bit takes precedence, so the Forced Block End Bit is ignored. Program execution is restarted when the Cycle Start Bit is turned ON again.

**7. Cycle Start Bit and Forced Block End Bit**



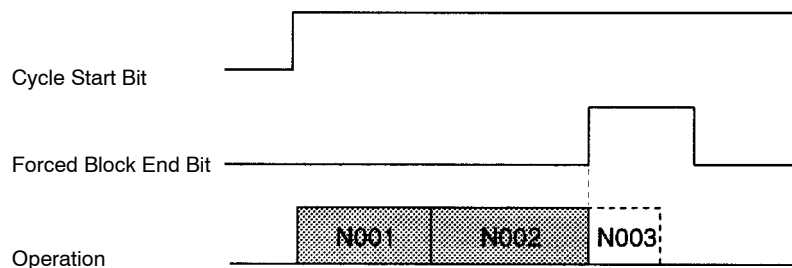
The Forced Block End Bit is turned ON to cancel execution of block N002 and stop program execution. Program execution is restarted when the Cycle Start Bit is turned ON again.

**8. Cycle Start Bit and Forced Block End Bit**



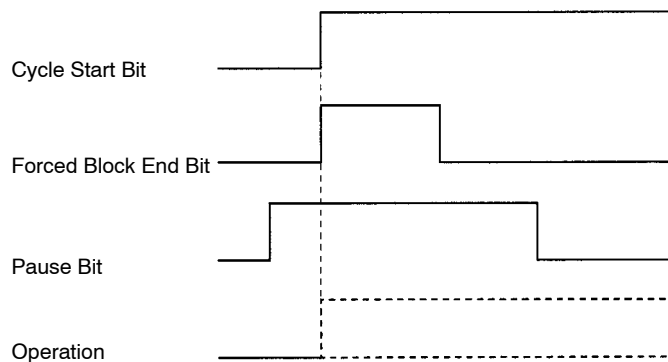
The Forced Block End Bit is turned ON to cancel execution of block N002 and stop program execution. Even though the Cycle Start Bit is turned ON again, the Forced Block End Bit is still ON, so program execution isn't restarted.

**9. Cycle Start Bit and Forced Block End Bit**



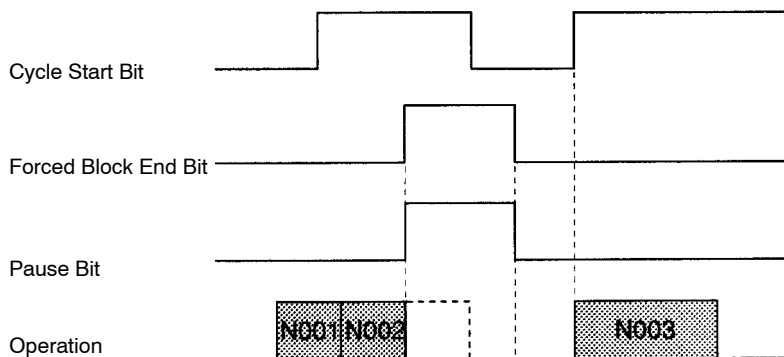
The Forced Block End Bit is turned ON between blocks N002 and N003, so execution of block N003 is cancelled and program execution is stopped.

**10. Cycle Start Bit, Forced Block End Bit, and Pause Bit**



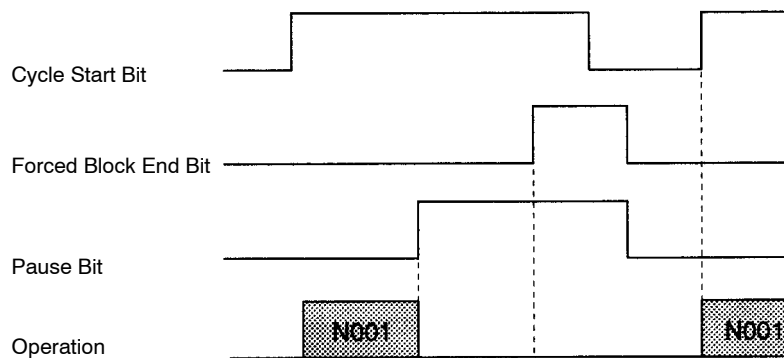
The Pause Bit is ON before the Cycle Start Bit is turned ON, so MC program execution isn't started. The Forced Block End Bit is turned ON at the same time as the Cycle Start Bit, but the Pause Bit is already ON and the Forced Block End Bit has no effect.

**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.

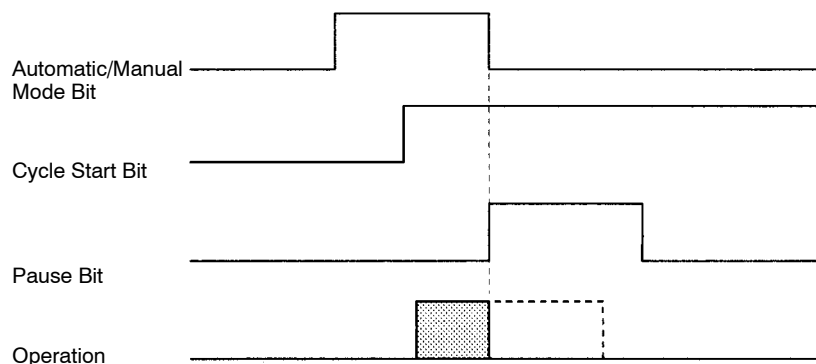
**12. Cycle Start Bit, Forced Block End Bit, and Pause Bit**



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.

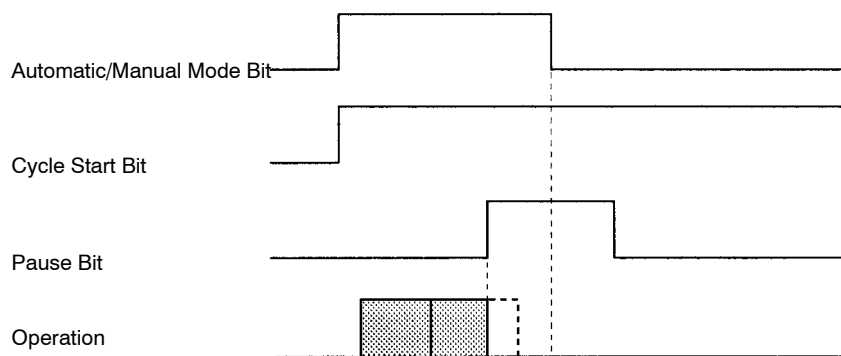


**13. Automatic/Manual Mode Bit, Cycle Start Bit, and Pause Bit**



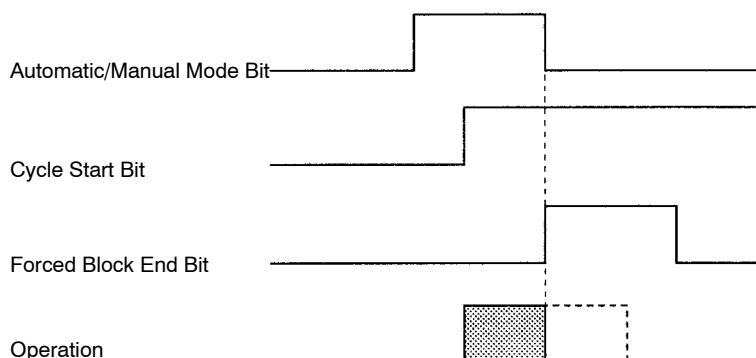
Program execution is stopped when the Automatic/Manual Mode Bit is turned OFF. The Unit is in manual mode, so the Pause Bit is ignored.

**14. Automatic/Manual Mode Bit, Cycle Start Bit, and Pause Bit**



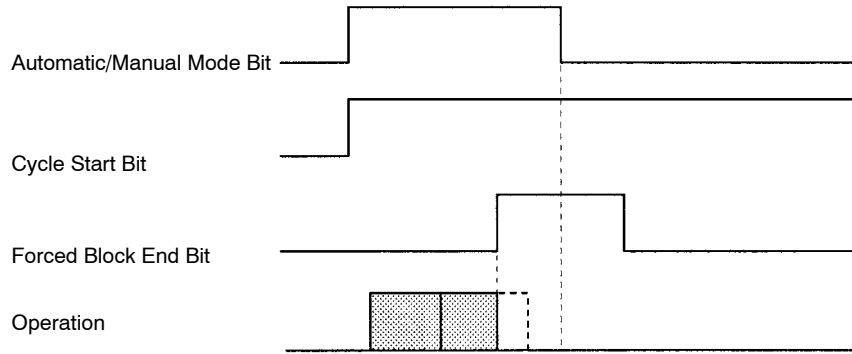
After program execution is paused with the Pause Bit, the Unit is switched to manual mode by turning OFF the Automatic/Manual Mode Bit.

**15. Automatic/Manual Mode Bit, Cycle Start Bit, and Forced Block End Bit**



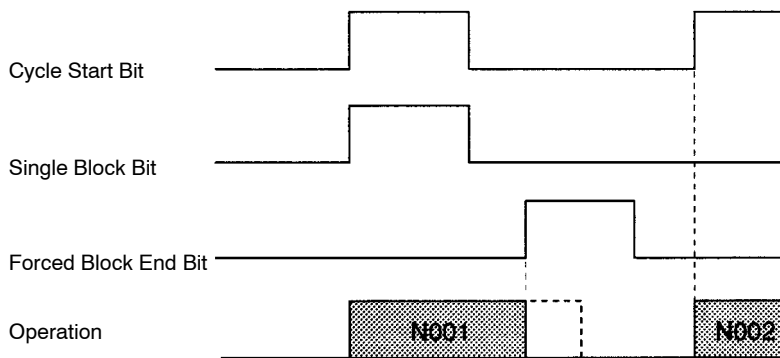
Program execution is stopped by turning OFF the Automatic/Manual Mode Bit. The Forced Block End Bit is ignored.

**16. Automatic/Manual Mode Bit, Cycle Start Bit, and Forced Block End Bit**



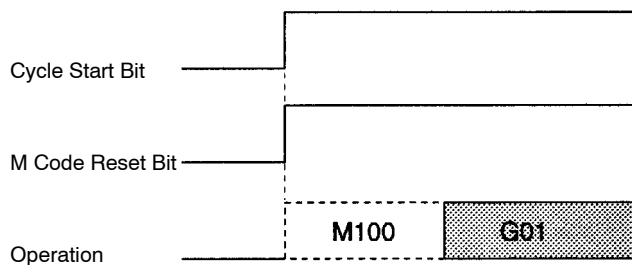
Program execution is stopped by the Forced Block End Bit. The Unit is switched to manual mode when the Automatic/Manual Mode Bit is turned OFF.

**17. Cycle Start Bit, Single Block Bit, and Forced Block End Bit**



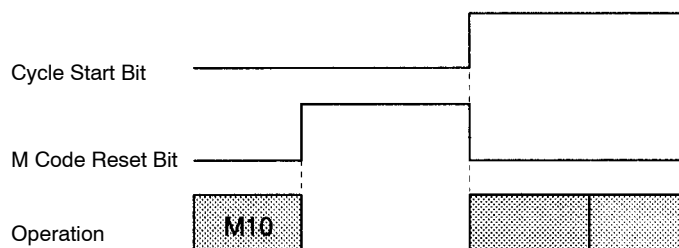
Program execution is started with the Single Block Bit ON, but execution of block N001 is stopped by the Forced Block End Bit. The next block is executed when the Cycle Start Bit is turned ON again.

**18. Cycle Start Bit and M Code Reset Bit**



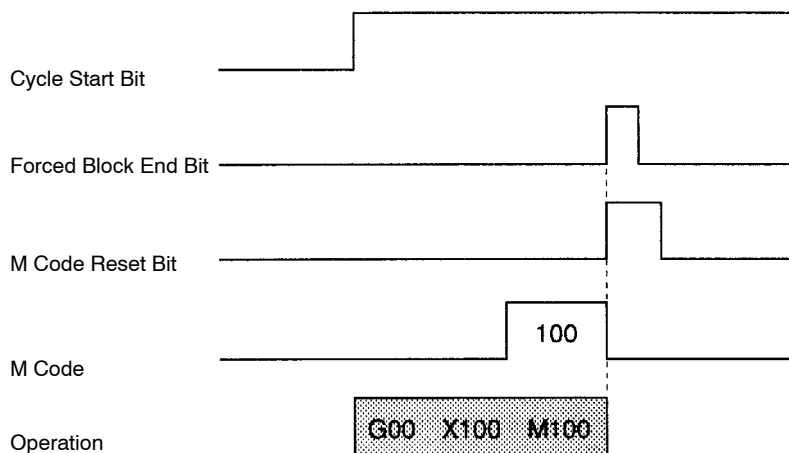
Program execution is started by the Cycle Start Bit. The next operation (G01) is executed without outputting M code 100 because the M Code Reset Bit is ON.

**19. Cycle Start Bit and M Code Reset Bit**



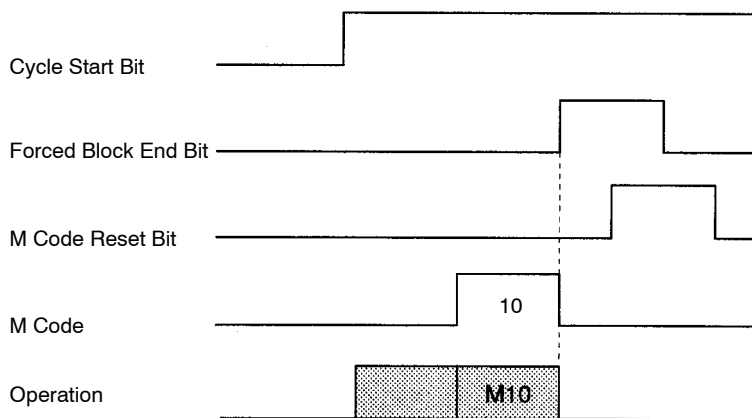
The Cycle Start Bit is invalid while the Unit is waiting for an M code reset. When the M Code Reset Bit is turned OFF, the standby status is cleared and the status of the Cycle Start Bit is checked. The Cycle Start Bit is ON, so program execution is restarted.

**20. Cycle Start Bit, Forced Block End Bit, M Code Reset Bit, and M Code Output**



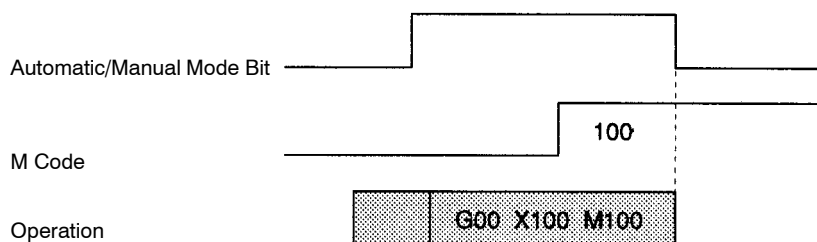
The Forced Block End Bit takes precedence and stops program execution when the Forced Block End Bit and the M Code Reset Bit are turned ON at the same time. The M code is cleared when the program is stopped. The program was already stopped, so program execution isn't restarted when the M Code Reset Bit is turned OFF.

**21. Cycle Start Bit, Forced Block End Bit, M Code Reset Bit, and M Code Output**



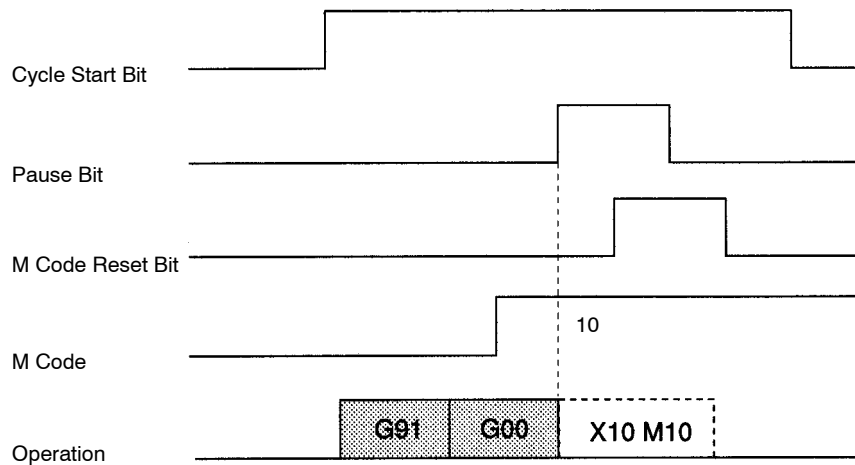
The program is stopped and the M code is cleared when the Forced Block End Bit goes ON.

**22. Automatic/Manual Mode Bit and M Code Output**



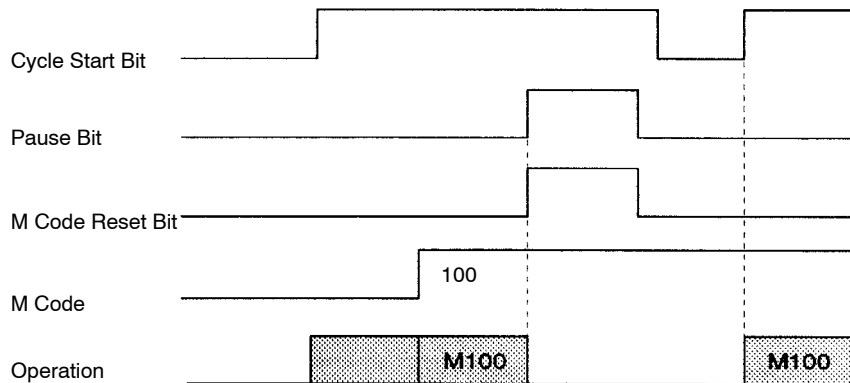
The M code isn't cleared when the Unit is switched manual mode.

**23. Cycle Start Bit, Pause Bit, M Code Reset Bit, and M Code Output**



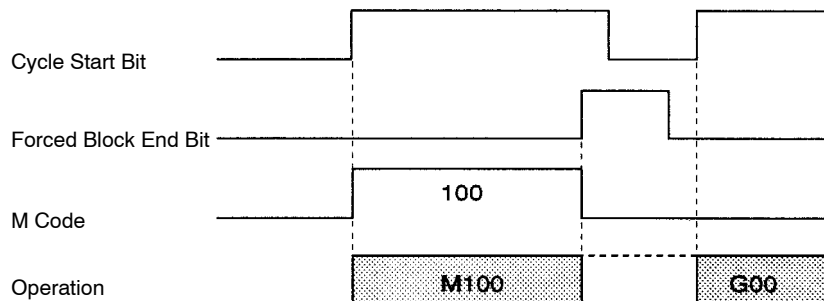
The program is paused when the Pause Bit is turned ON, but the M code isn't cleared. M code M10 isn't cleared by the M Code Reset Bit because program execution is paused.

**24. Cycle Start Bit, Pause Bit, M Code Reset Bit, and M Code Output**



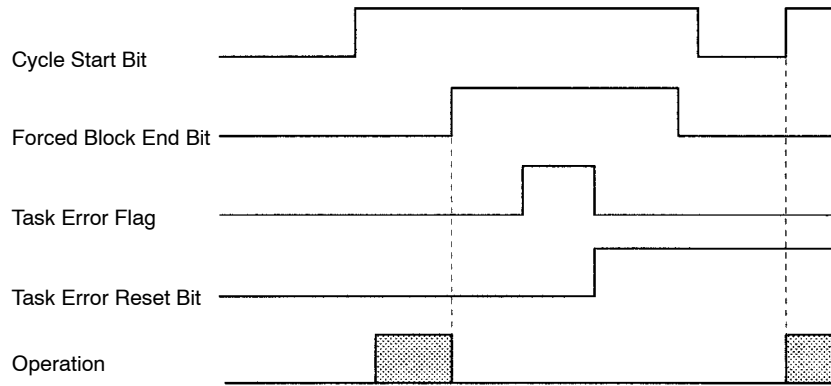
If the Pause Bit and M Code Reset Bit are turned ON at the same time, the Pause Bit becomes valid. Program execution is paused, but the M code isn't cleared. When the program is restarted, the M code will be output because it was ON originally.

**25. Cycle Start Bit, Forced Block End Bit, and M Code Output**



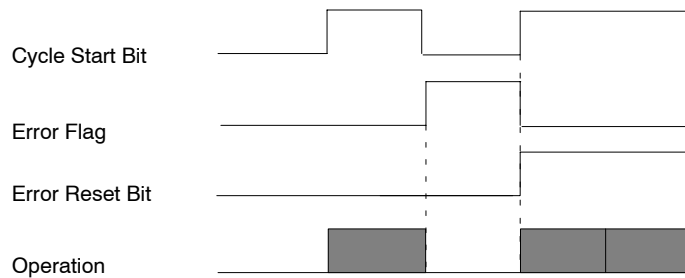
The Forced Block End Bit is turned ON while the Unit is standing by for an M code reset, cancelling the block.

**26. Cycle Start Bit, Forced Block End Bit, Task Error Flag, and Task Error Reset Bit**



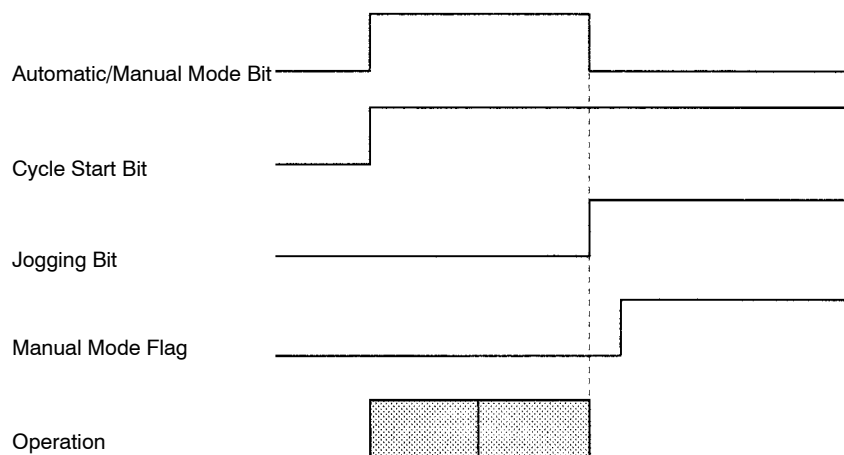
Operation is stopped with the Forced Block End Bit and then a task error occurs. Correct the cause of the error, clear the error by turning ON the Task Error Reset Bit, and restart the program by toggling the Cycle Start Bit.

**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.

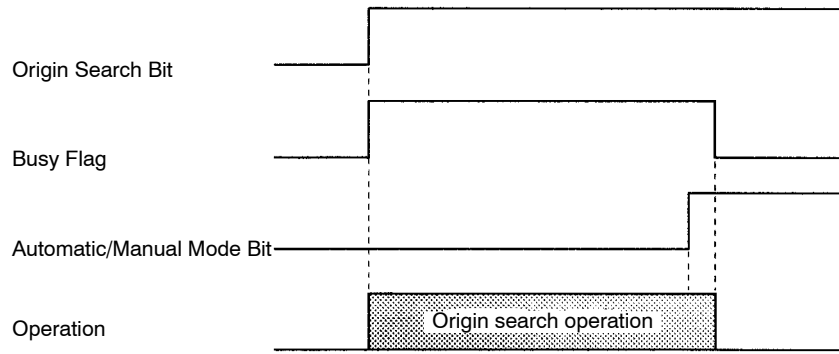
**28. Automatic/Manual Mode Bit, Cycle Start Bit, Jogging Bit, and Manual Mode Flag**



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.

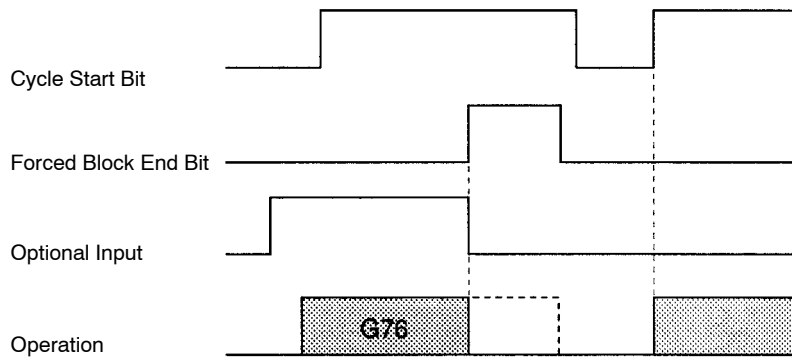
The Manual Mode Flag will be turned ON when positioning is completed after decelerating to a stop. The Jogging Bit signal can be received once the Manual Mode Flag is ON.

**29. Origin Search Bit, Busy Flag, and Automatic/Manual Mode Bit**



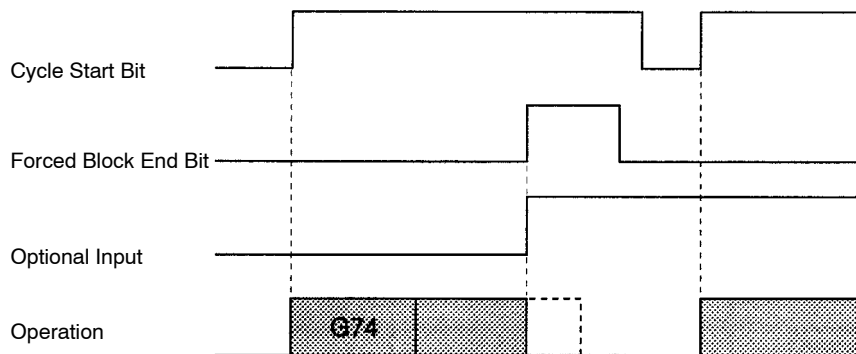
The manual mode origin search is stopped when the Automatic/Manual Mode Bit is turned ON. All axes in the task are stopped and the Unit enters automatic mode. At this point, the Busy Flags for all of the axes will be OFF, even though the Origin Search Bit remains ON.

**30. Cycle Start Bit, Forced Block End Bit, and Optional Input**



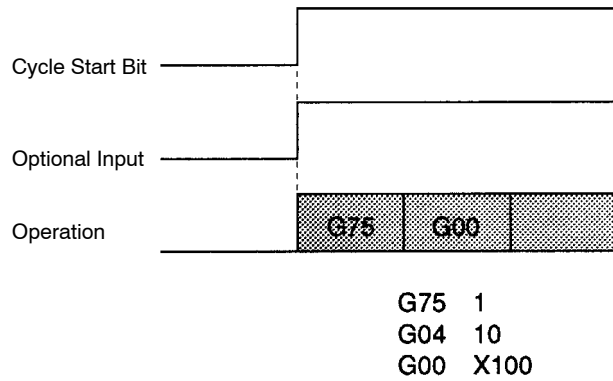
While the OPTIONAL PROGRAM STOP function (G76) is being executed, the block is cancelled by turning ON the Forced Block End Bit. The optional input is turned OFF at the same time that the Forced Block End Bit is turned ON, so in the end the input has no effect.

**31. Cycle Start Bit, Forced Block End Bit, and Optional Input**



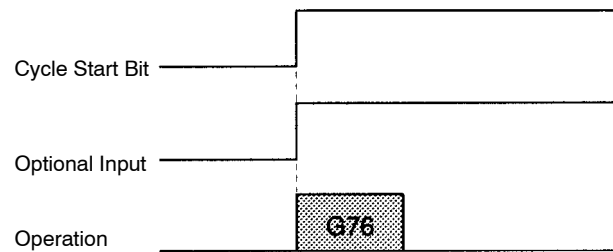
While the block after the OPTIONAL END function (G74) is being executed, the block is cancelled by turning ON the Forced Block End Bit. The optional input and Forced Block End Bit are turned ON at the same time, but the Forced Block End Bit takes precedence.

**32. Cycle Start Bit and Optional Input**



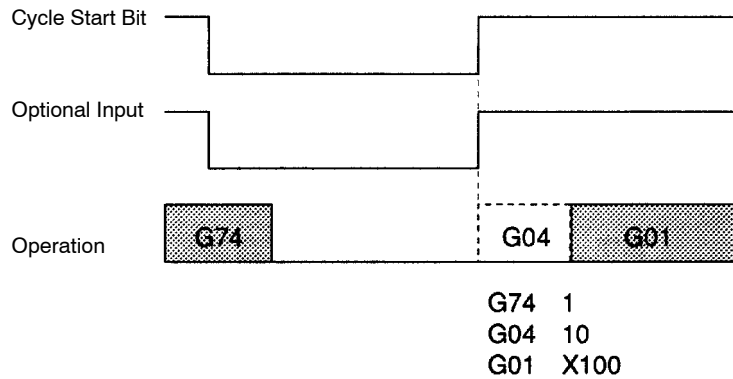
Program execution starts when the Cycle Start Bit is turned ON. The next block after the OPTIONAL SKIP function (G75) is skipped because the optional input is ON.

**33. Cycle Start Bit and Optional Input**



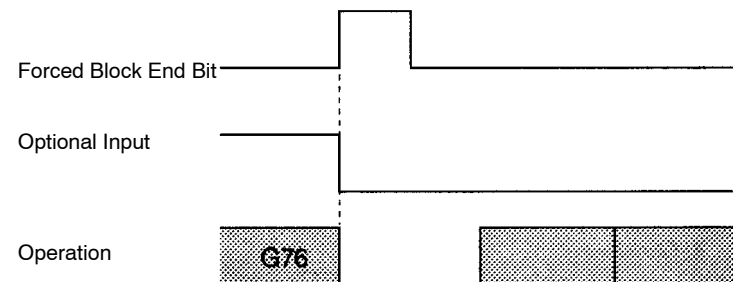
Program execution starts when the Cycle Start Bit is turned ON. The status of the optional input is checked when the OPTIONAL PROGRAM STOP function (G76) is executed, and the program is stopped because the optional input is ON.

**34. Cycle Start Bit and Optional Input**



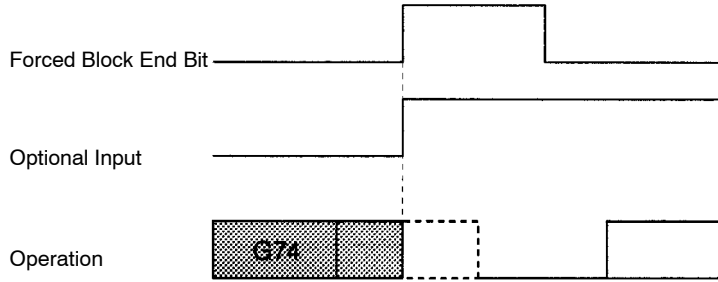
Program execution is restarted when the Cycle Start Bit is turned ON. The status of the optional input is read due to the execution of the OPTIONAL END function (G74) and G04 is cancelled.

**35. Forced Block End Bit and Optional Input**



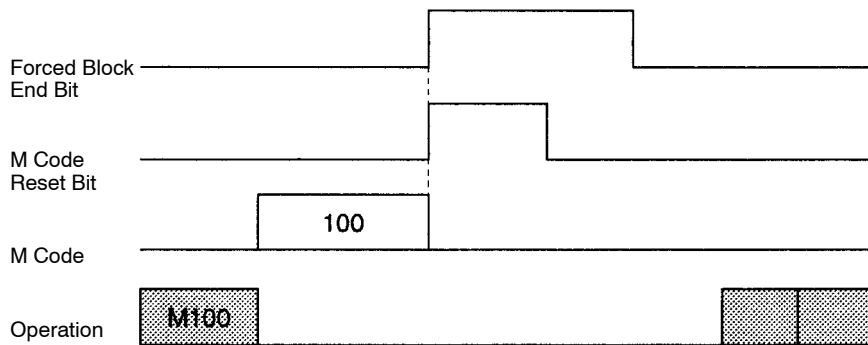
The OPTIONAL PROGRAM STOP function (G76) is cancelled by the Forced Block End Bit. The optional input is turned OFF at the same time, but this is ignored and program execution is stopped.

**36. Forced Block End Bit and Optional Input**



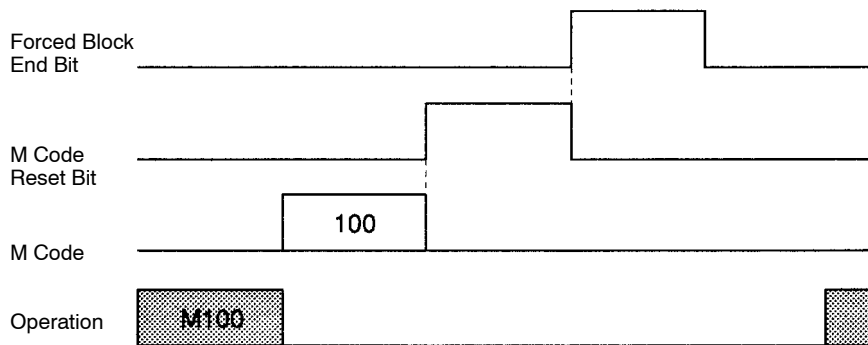
The Forced Block End Bit takes precedence if it's turned ON at the same time as the optional input in the block after an OPTIONAL END function (G74). The Unit stands by for restarting after program execution is stopped.

**37. Forced Block End Bit, M Code Reset Bit, and M Code Output**



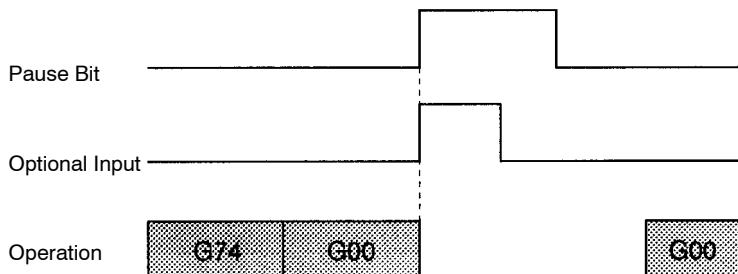
Turning ON the Forced Block End Bit clears the M code and stops program execution. The M Code Reset Bit signal is ignored.

**38. 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.

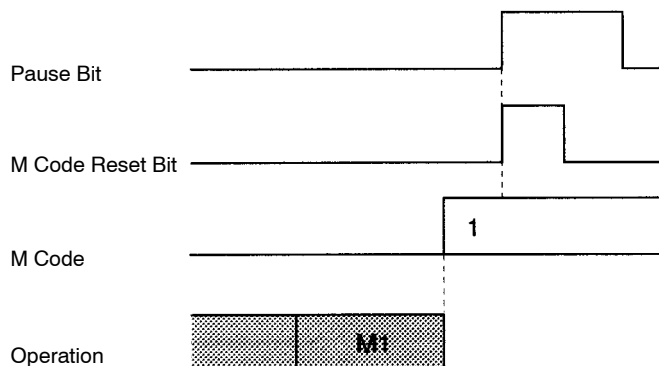
**39. Pause Bit and Optional Input**





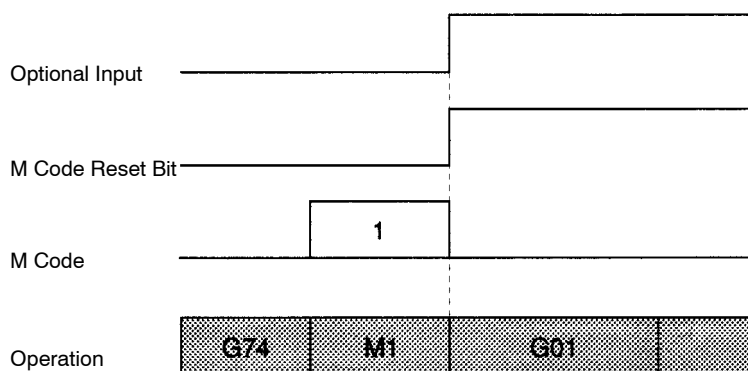
The G00 positioning operation is stopped by turning ON the Pause Bit in the block after the OPTIONAL END function (G74). The optional input is ignored even if it goes OFF at the same time. (The program isn't ended.) Execution of function G00 is continued when the program is restarted.

#### 40. 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.

#### 41. 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.



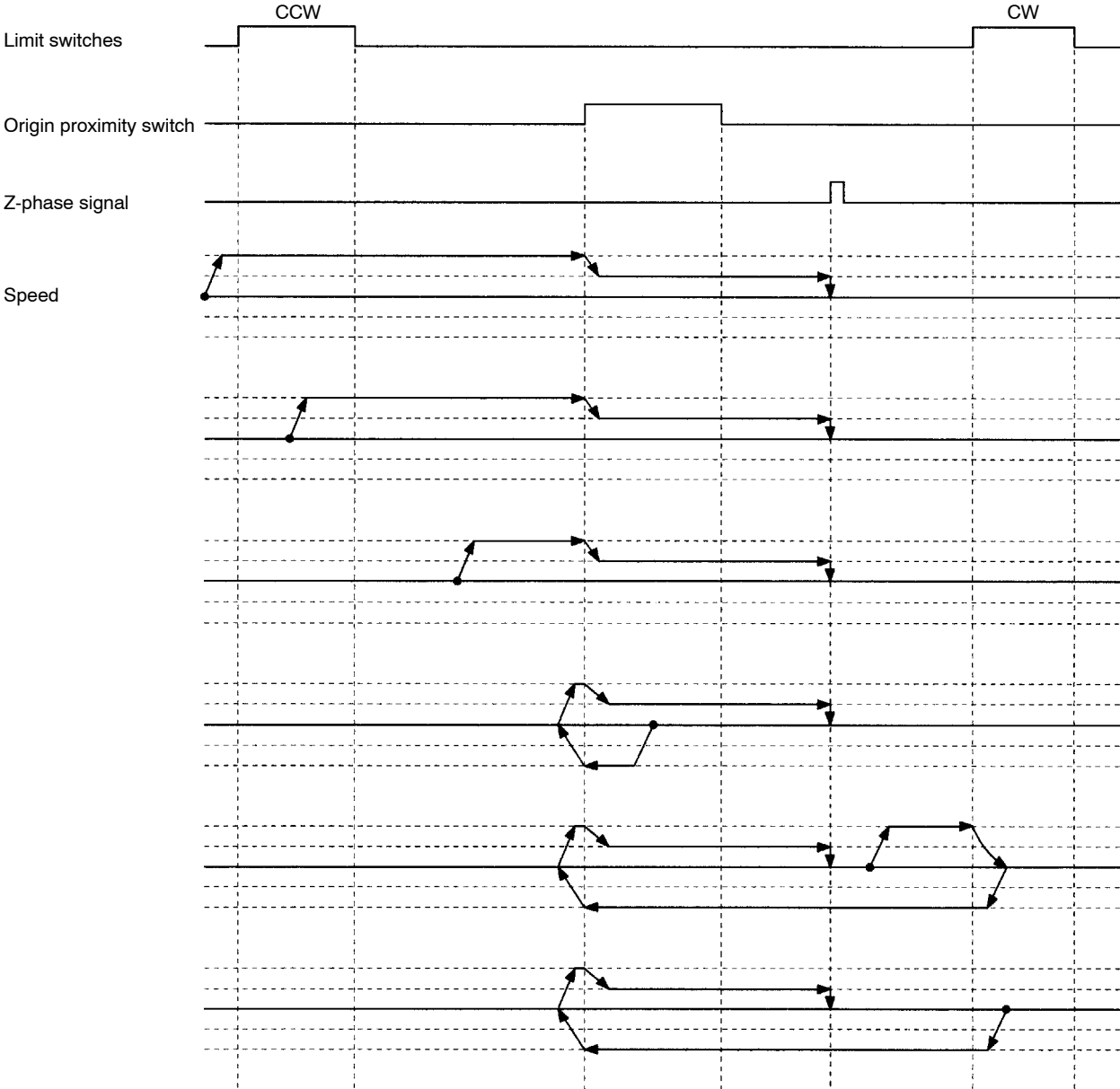
# Appendix B

## 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.

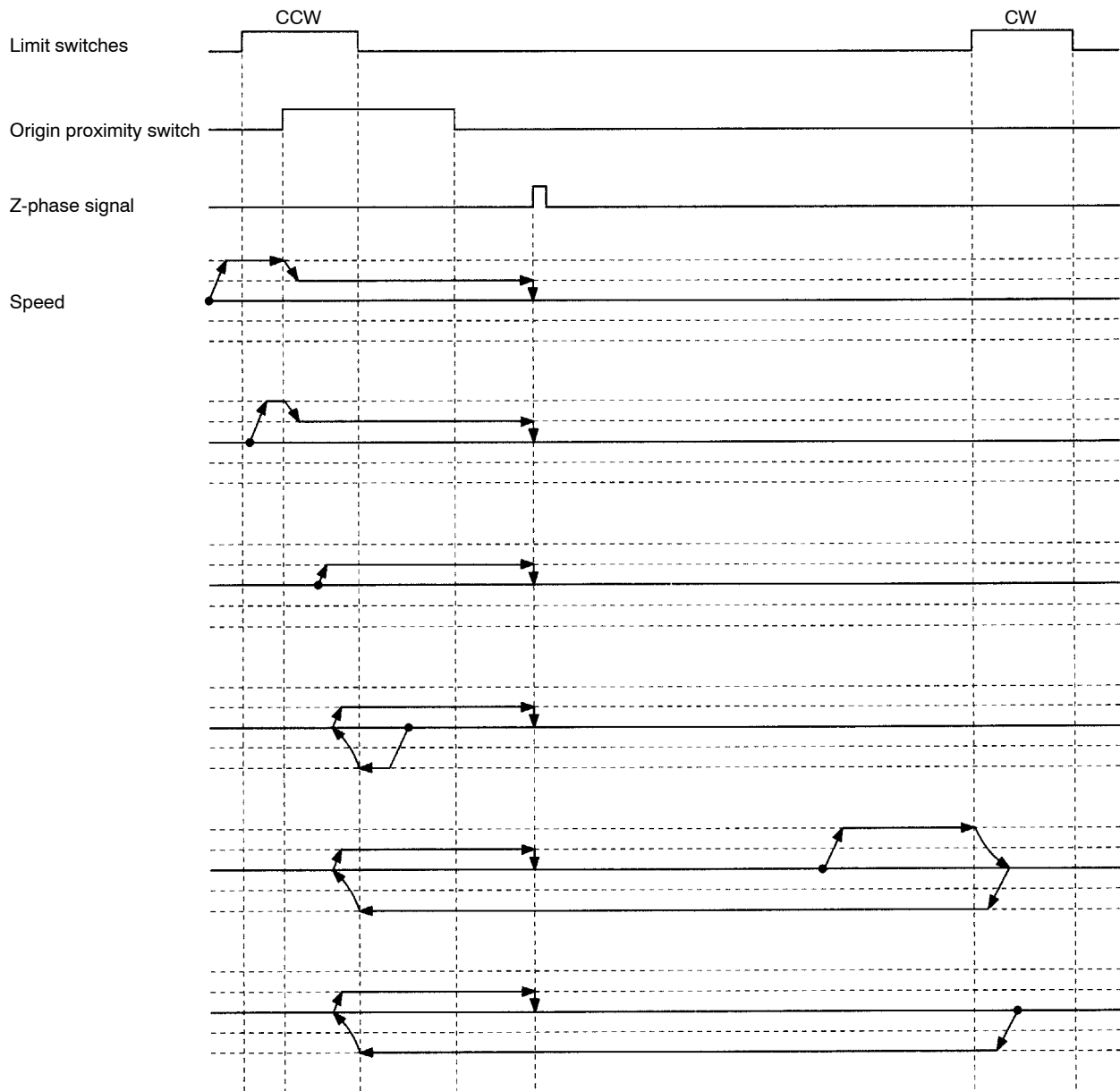
### Reverse-mode Origin Searches 1

These origin searches are performed with an origin proximity switch and the initial search direction set to CW.



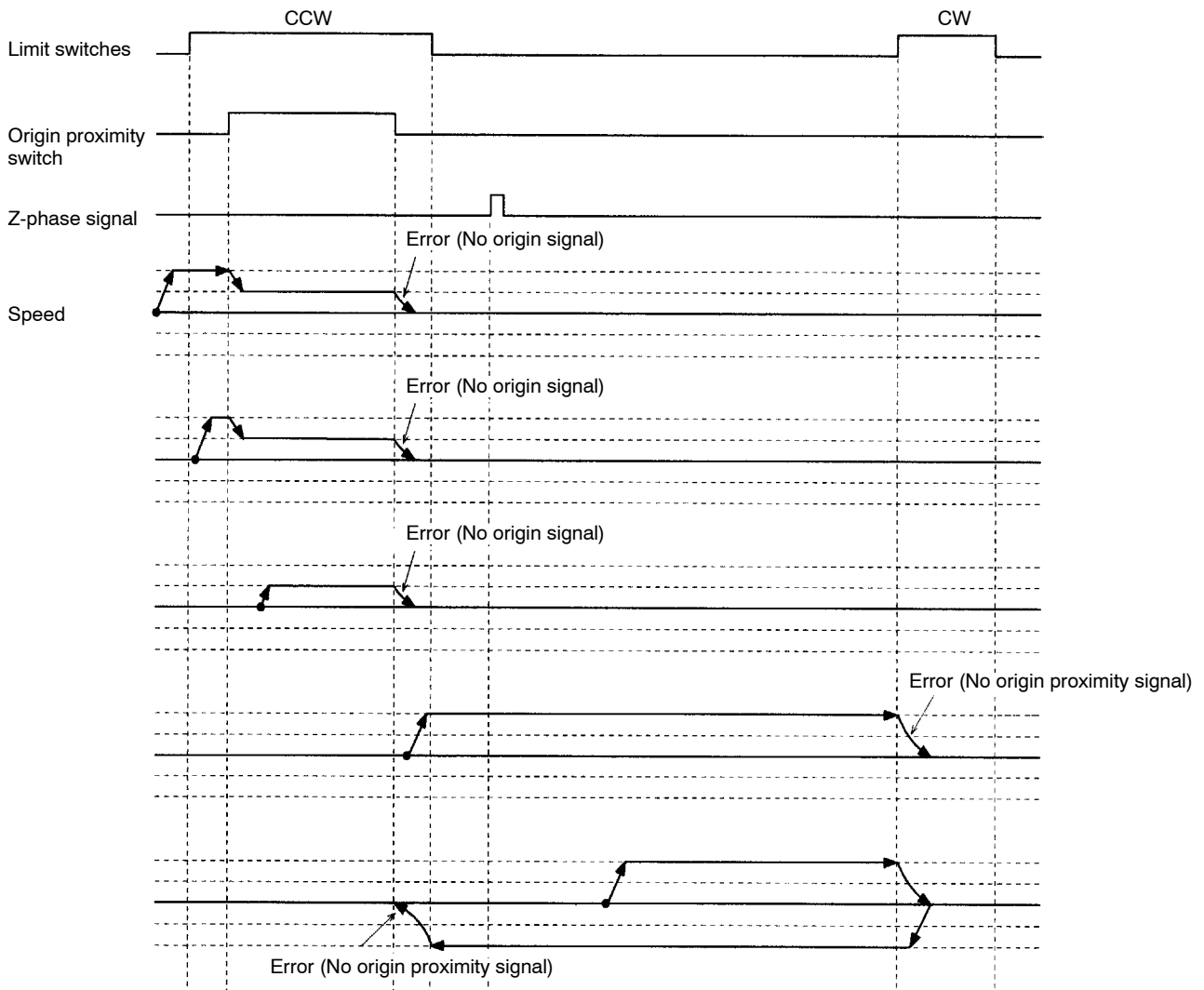
## Reverse-mode Origin Searches 2

These origin searches are performed with an origin proximity switch and the initial search direction set to CW.



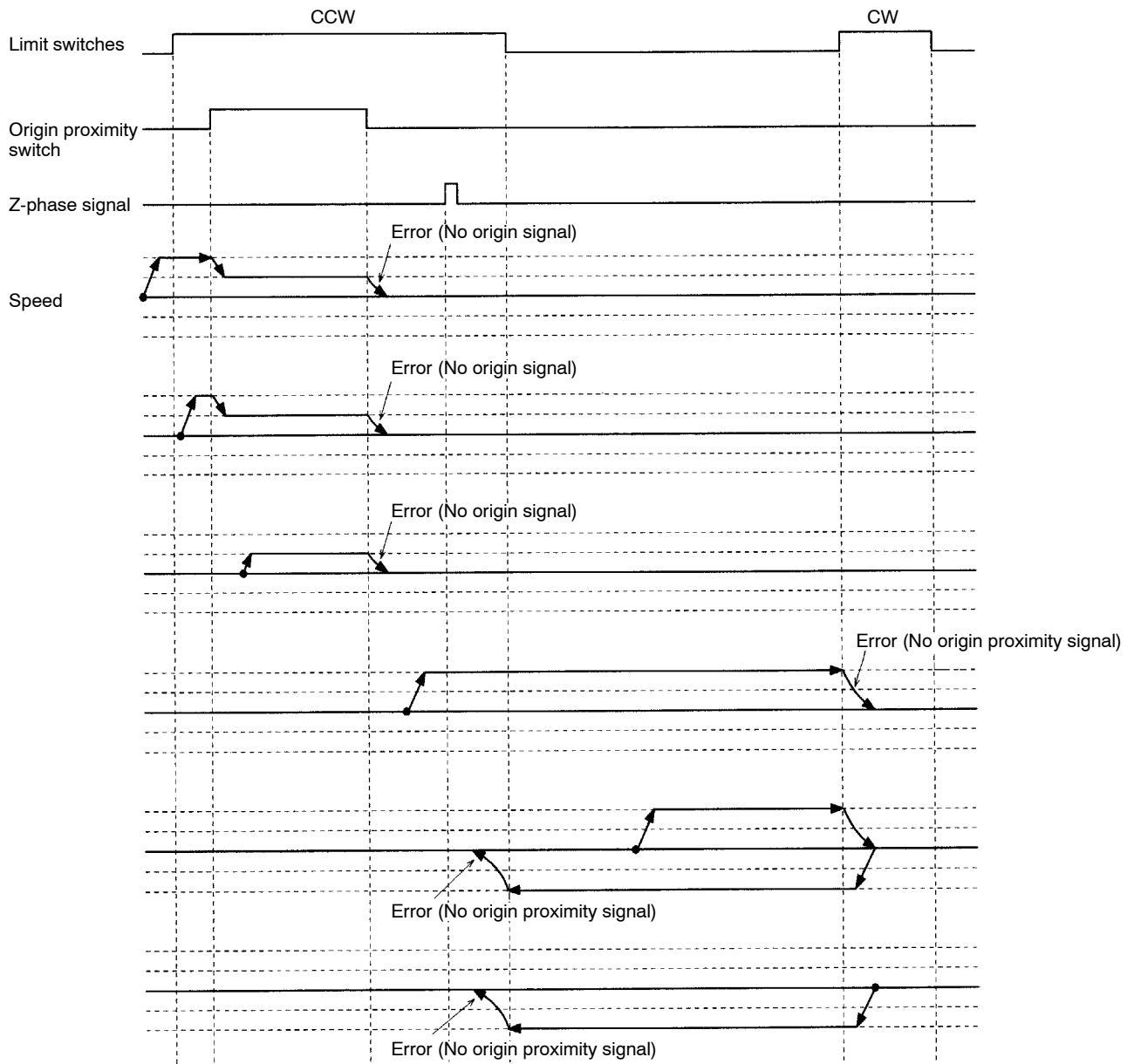
### Reverse-mode Origin Searches 3

These origin searches are performed with an origin proximity switch and the initial search direction set to CW.



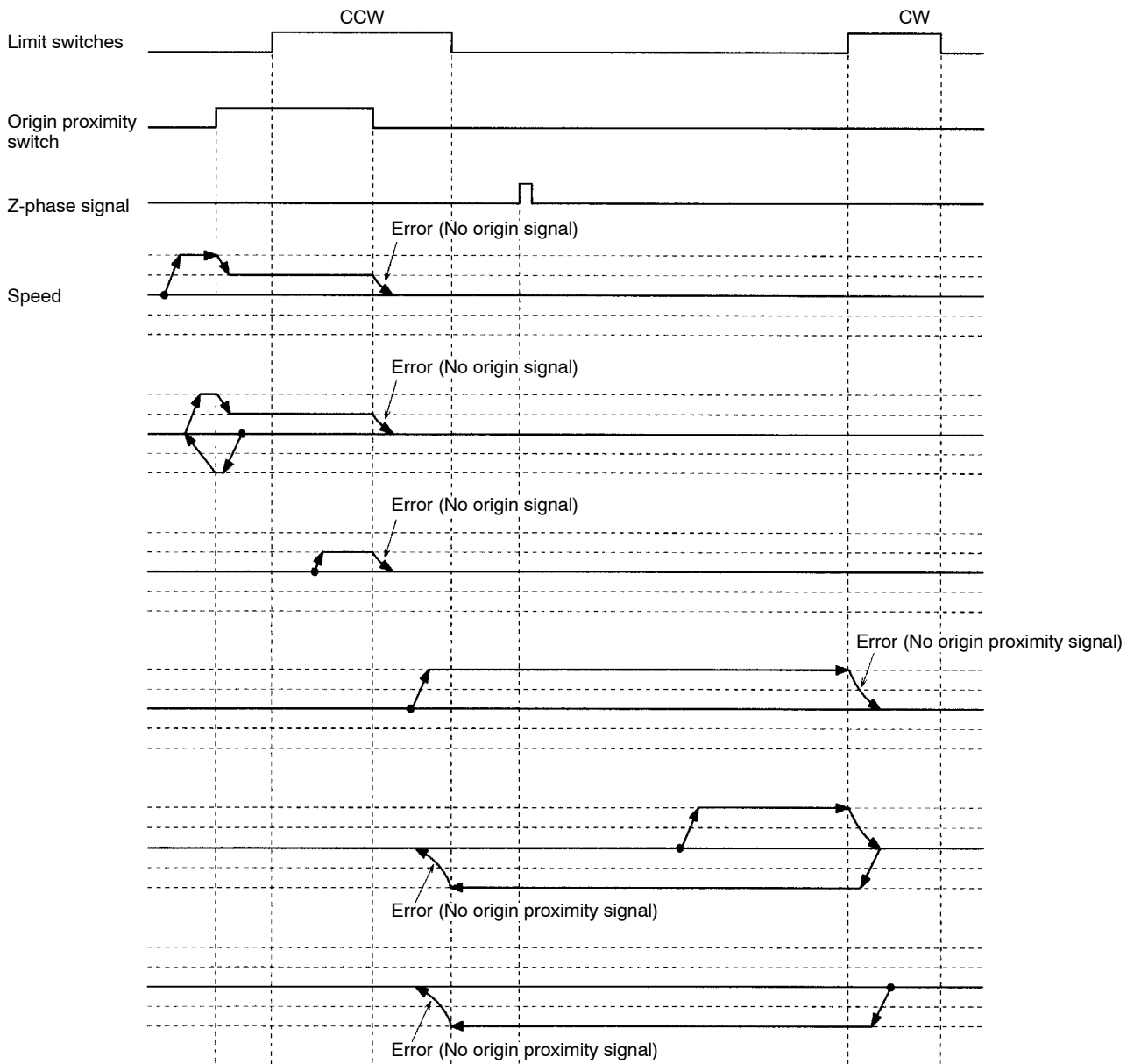
# Reverse-mode Origin Searches 4

These origin searches are performed with an origin proximity switch and the initial search direction set to CW.



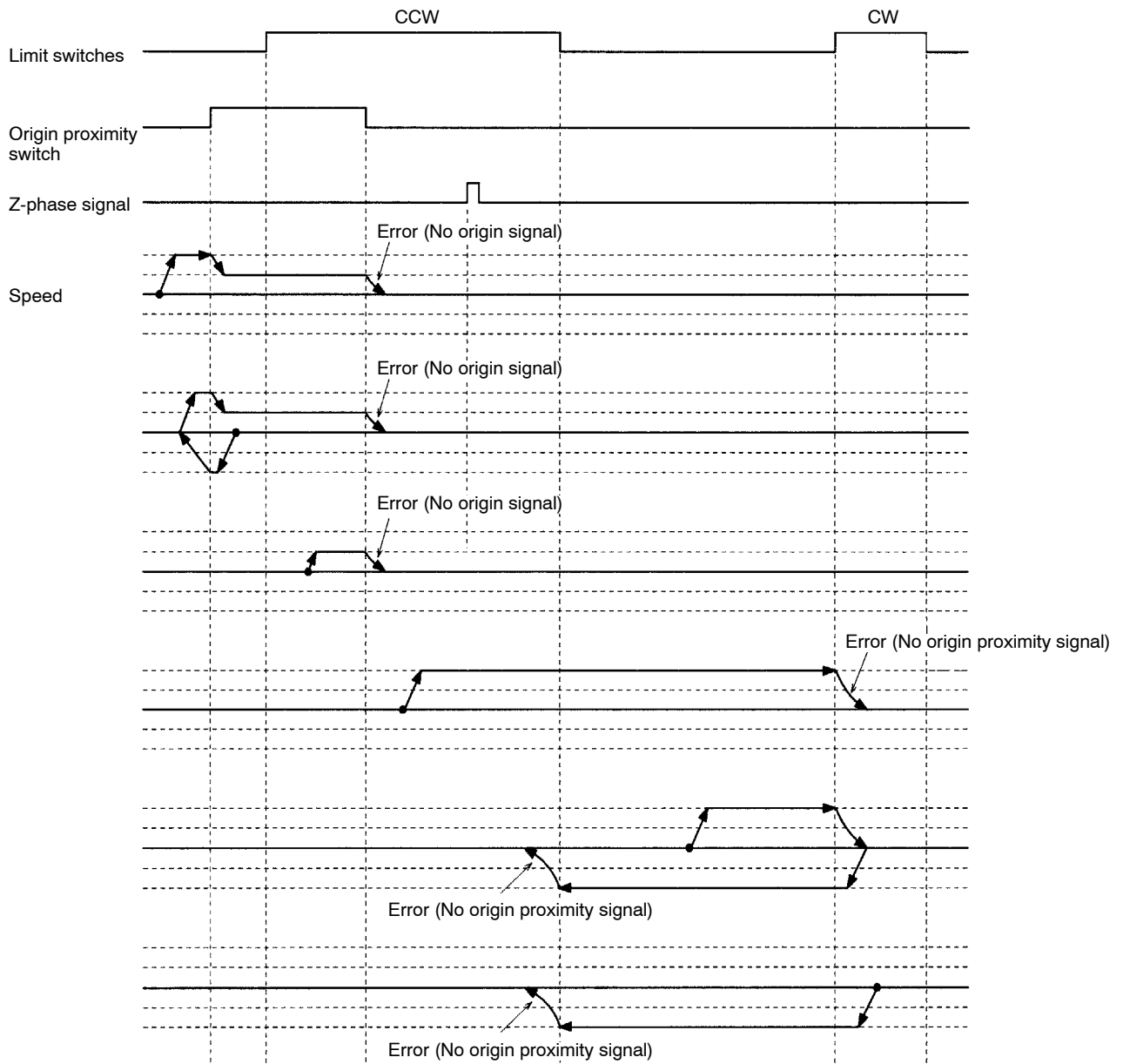
# Reverse-mode Origin Searches 5

These origin searches are performed with an origin proximity switch and the initial search direction set to CW.



# Reverse-mode Origin Searches 6

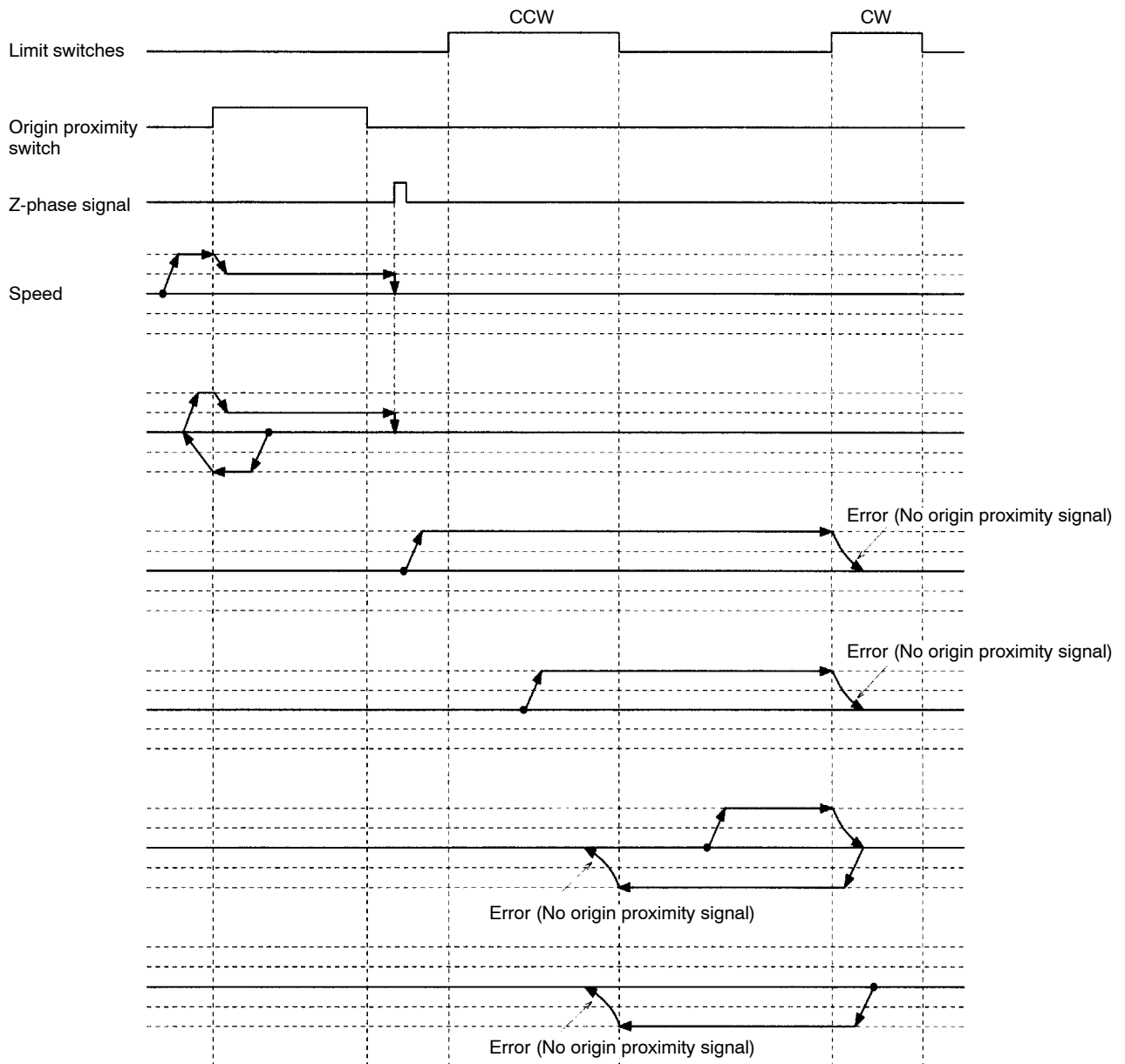
These origin searches are performed with an origin proximity switch and the initial search direction set to CW.





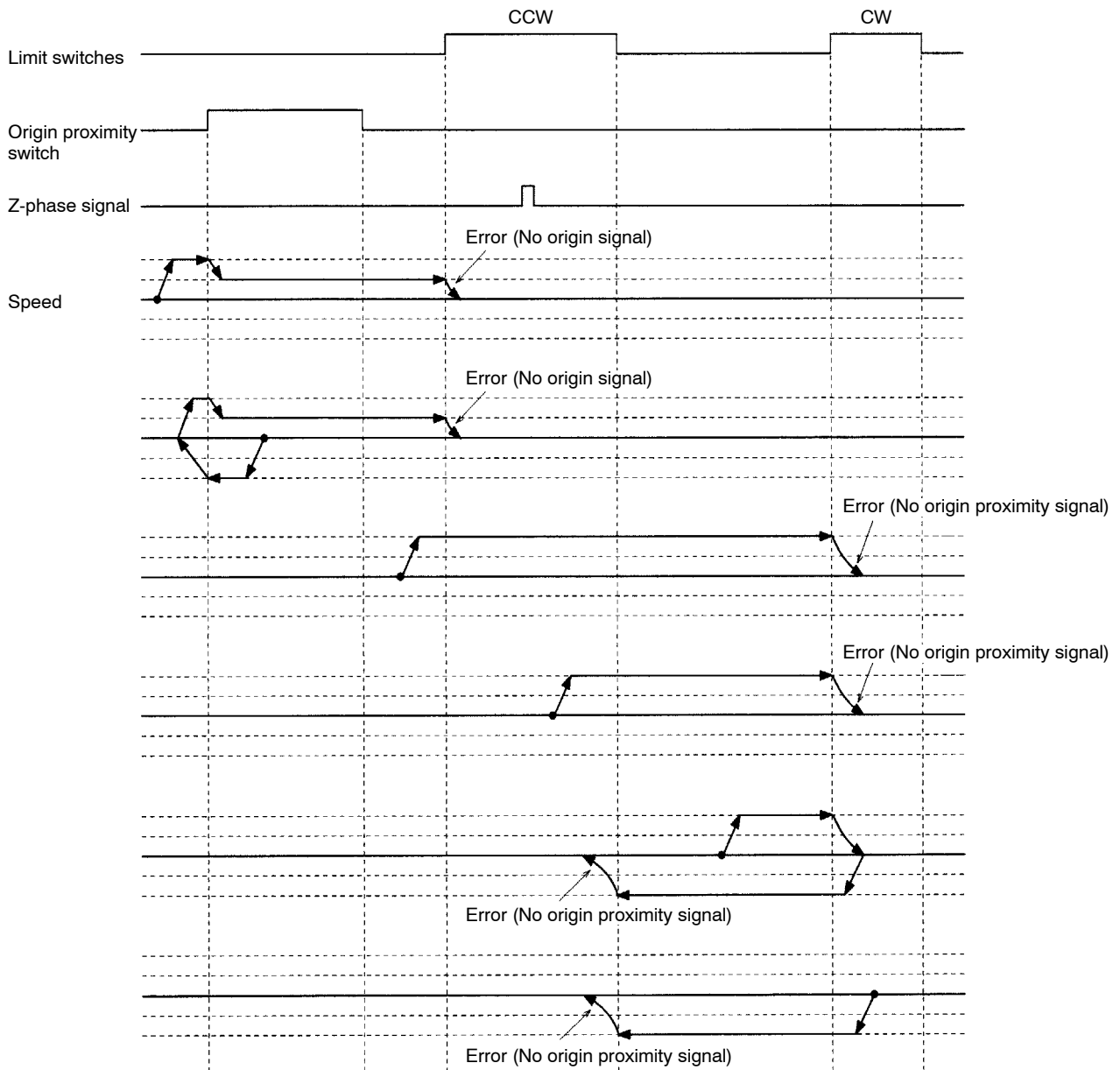
# Reverse-mode Origin Searches 7

These origin searches are performed with an origin proximity switch and the initial search direction set to CW.



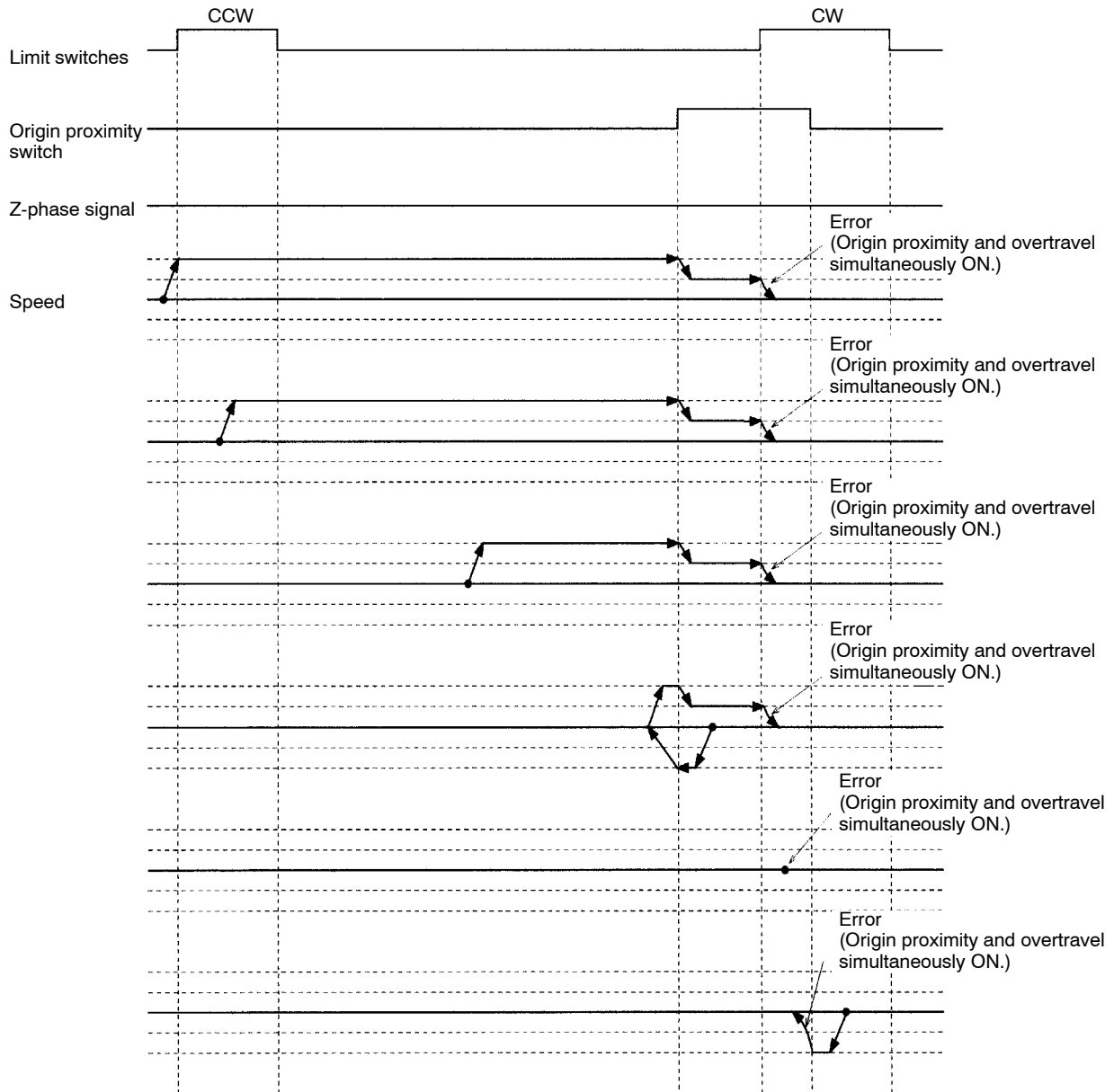
# Reverse-mode Origin Searches 8

These origin searches are performed with an origin proximity switch and the initial search direction set to CW.



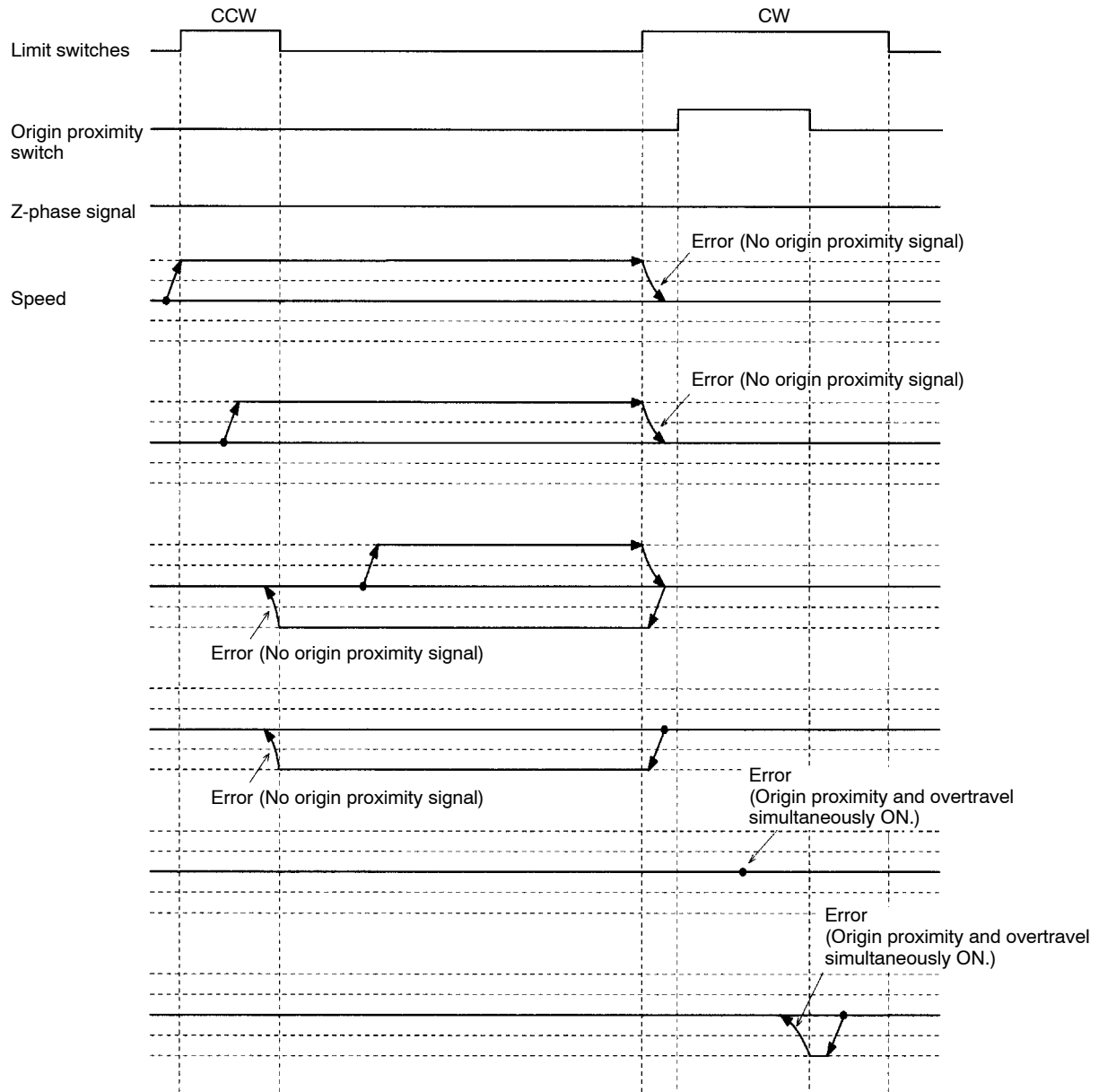
## Reverse-mode Origin Searches 9

These origin searches are performed with an origin proximity switch and the initial search direction set to CW, but no Z-phase signal.



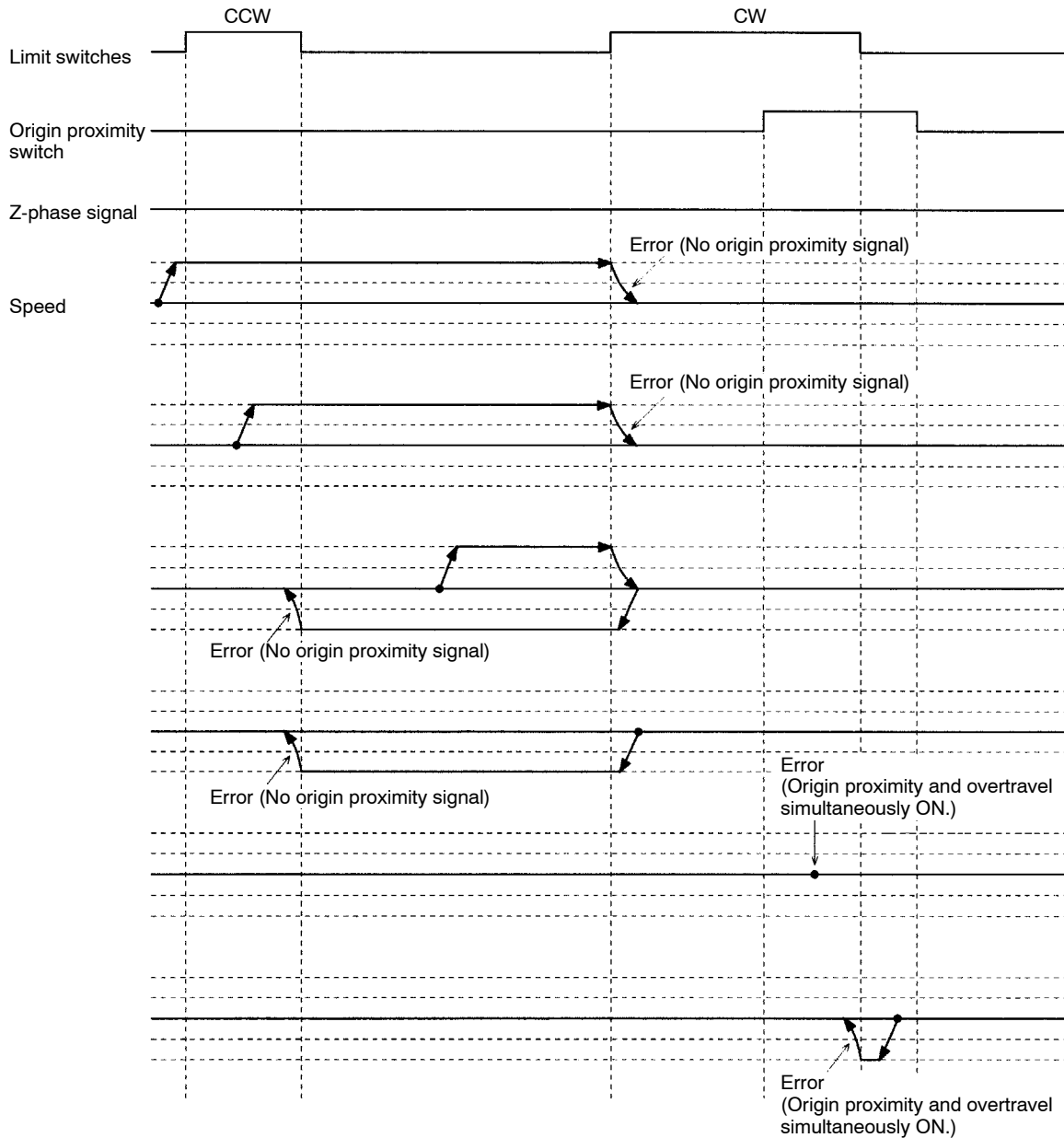
# Reverse-mode Origin Searches 10

These origin searches are performed with an origin proximity switch and the initial search direction set to CW, but no Z-phase signal.



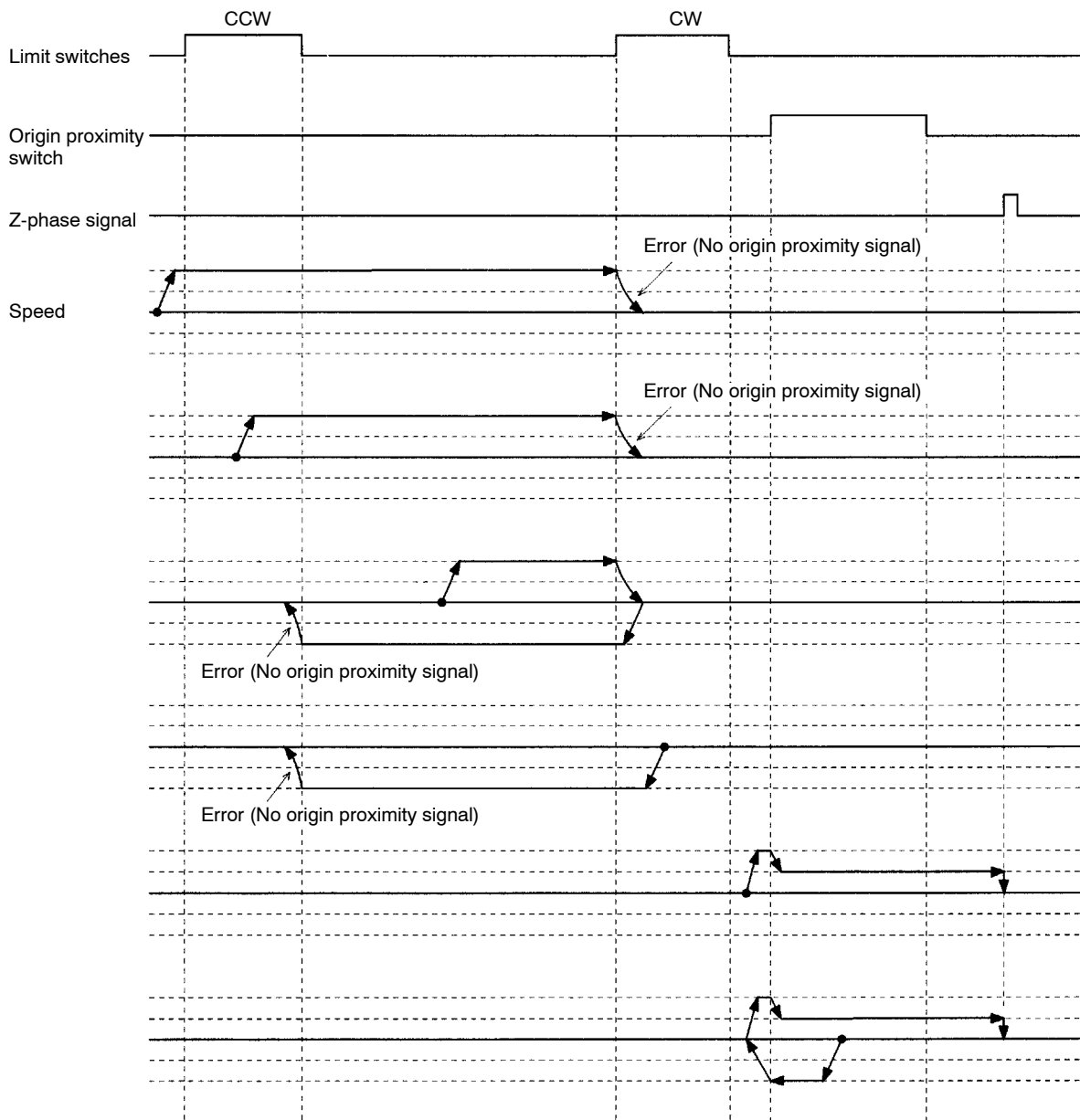
## Reverse-mode Origin Searches 11

These origin searches are performed with an origin proximity switch and the initial search direction set to CW, but no Z-phase signal.

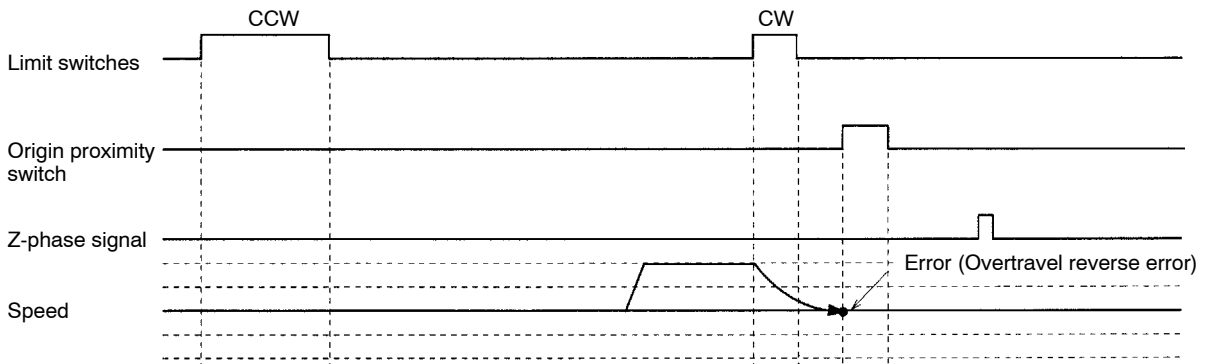


## Reverse-mode Origin Searches 12

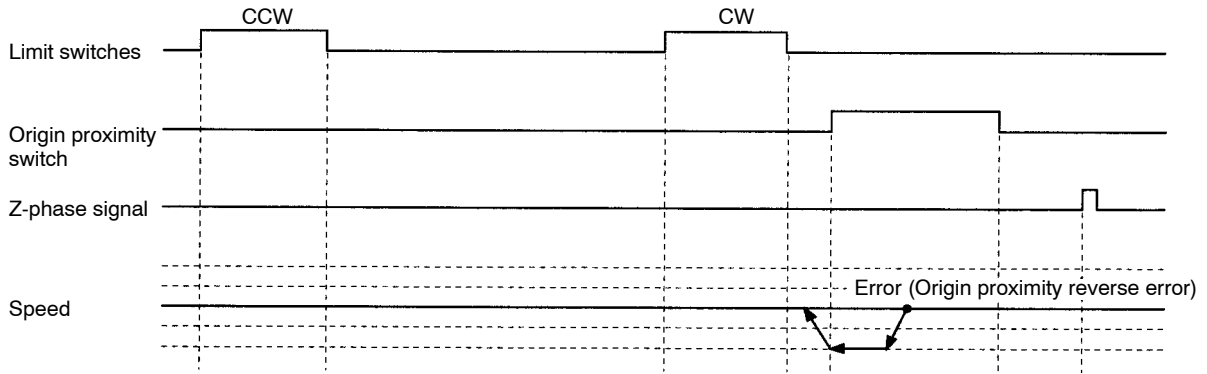
These origin searches are performed with an origin proximity switch and the initial search direction set to CW.



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.

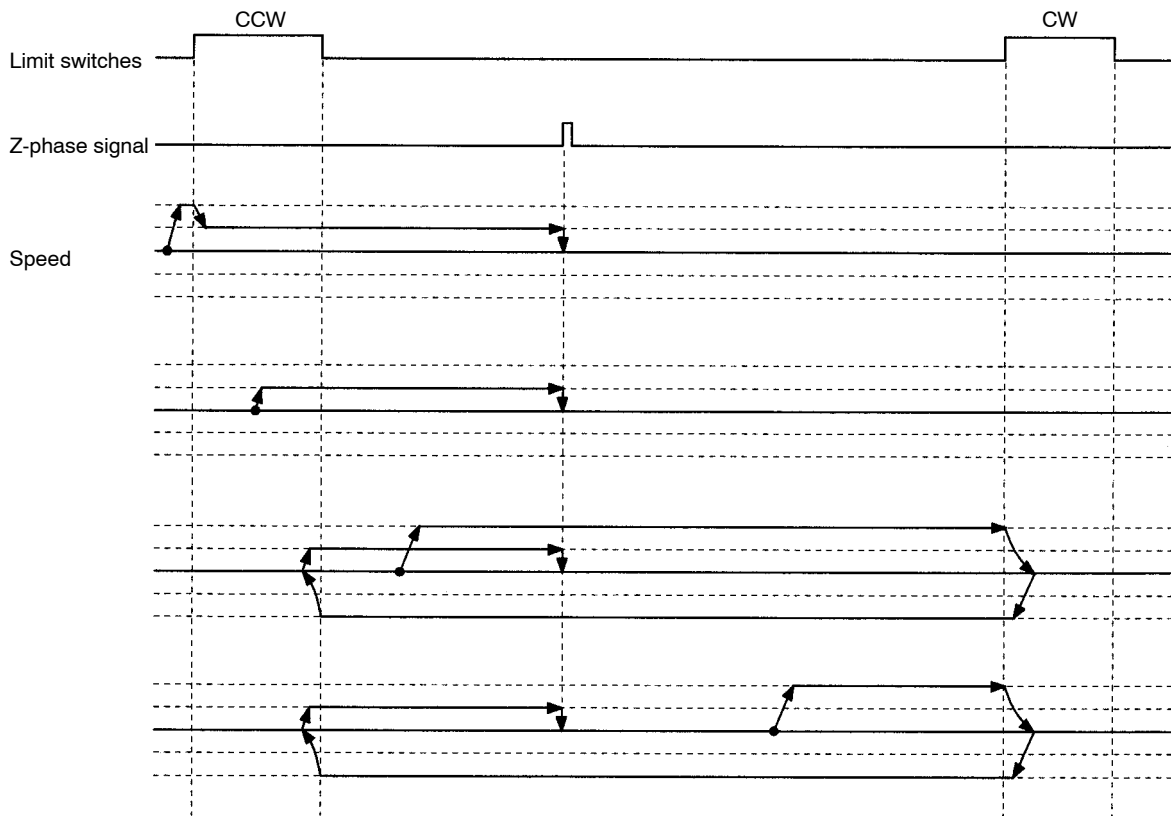


The following error can occur if the CCW limit switch signal is too close to the origin proximity signal.



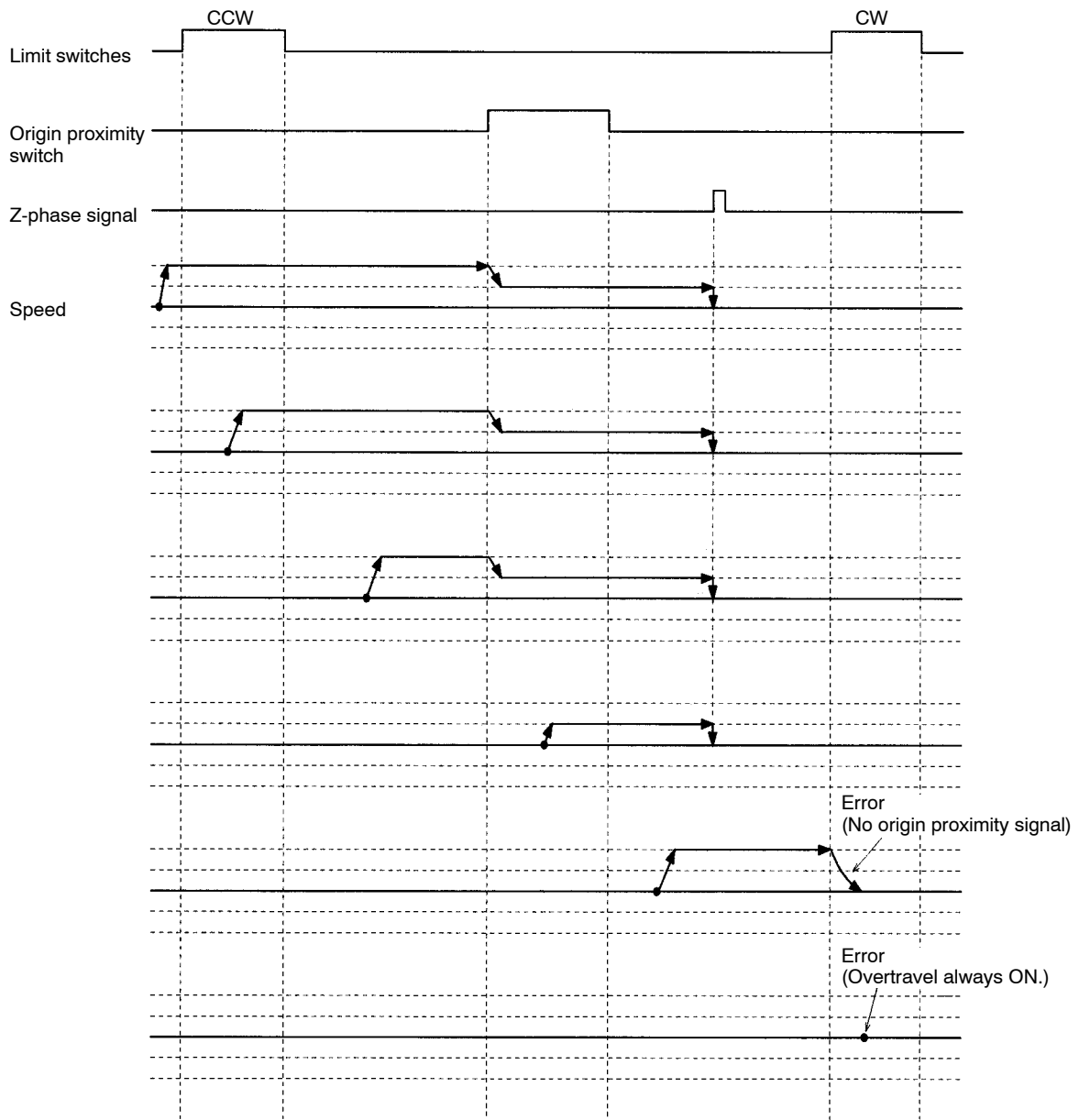
### Reverse-mode Origin Searches 13

These origin searches are performed without an origin proximity switch and the initial search direction set to CW.



# One Direction-mode Origin Searches 1

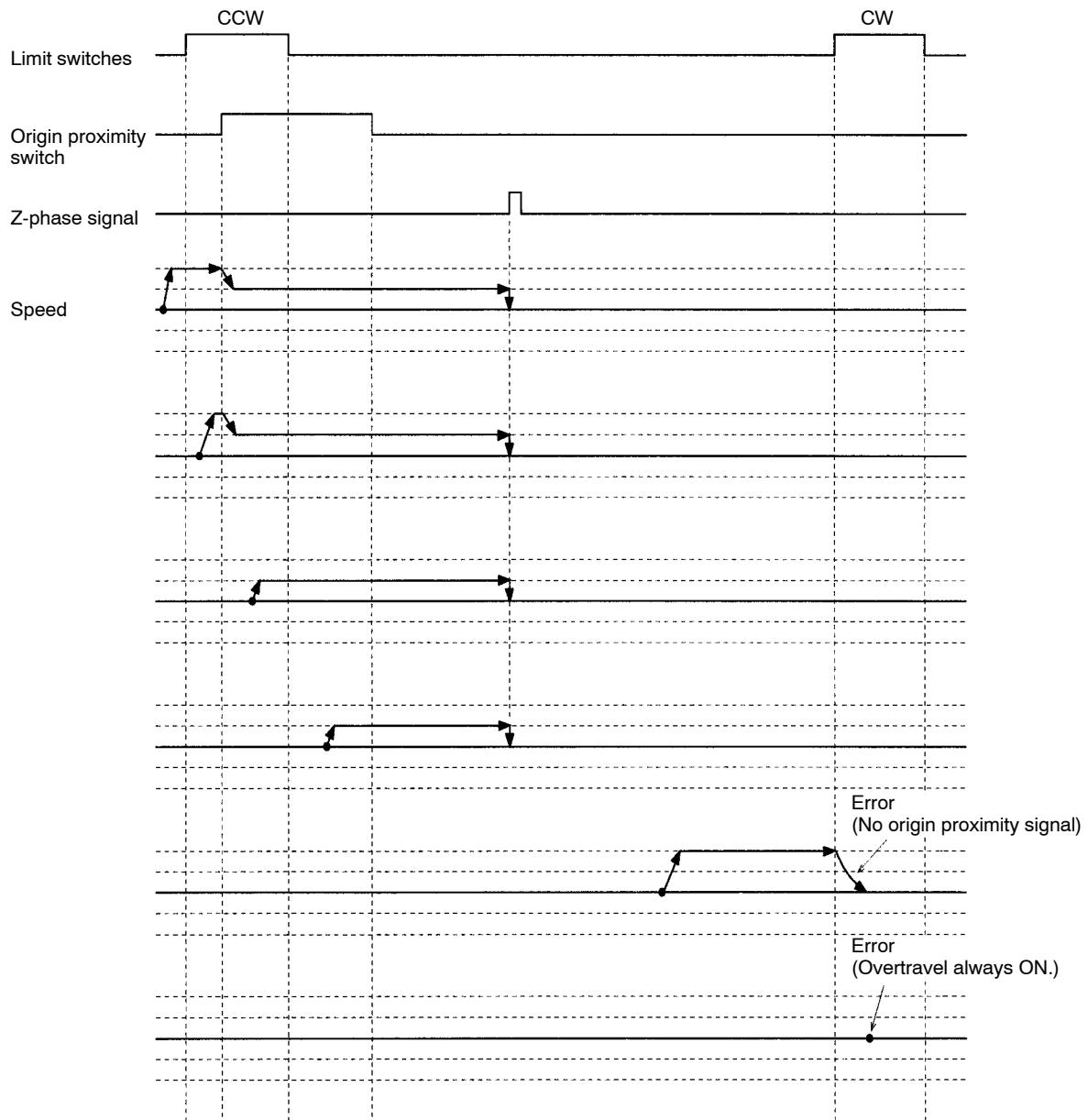
These origin searches are performed with an origin proximity switch and the initial search direction set to CW.





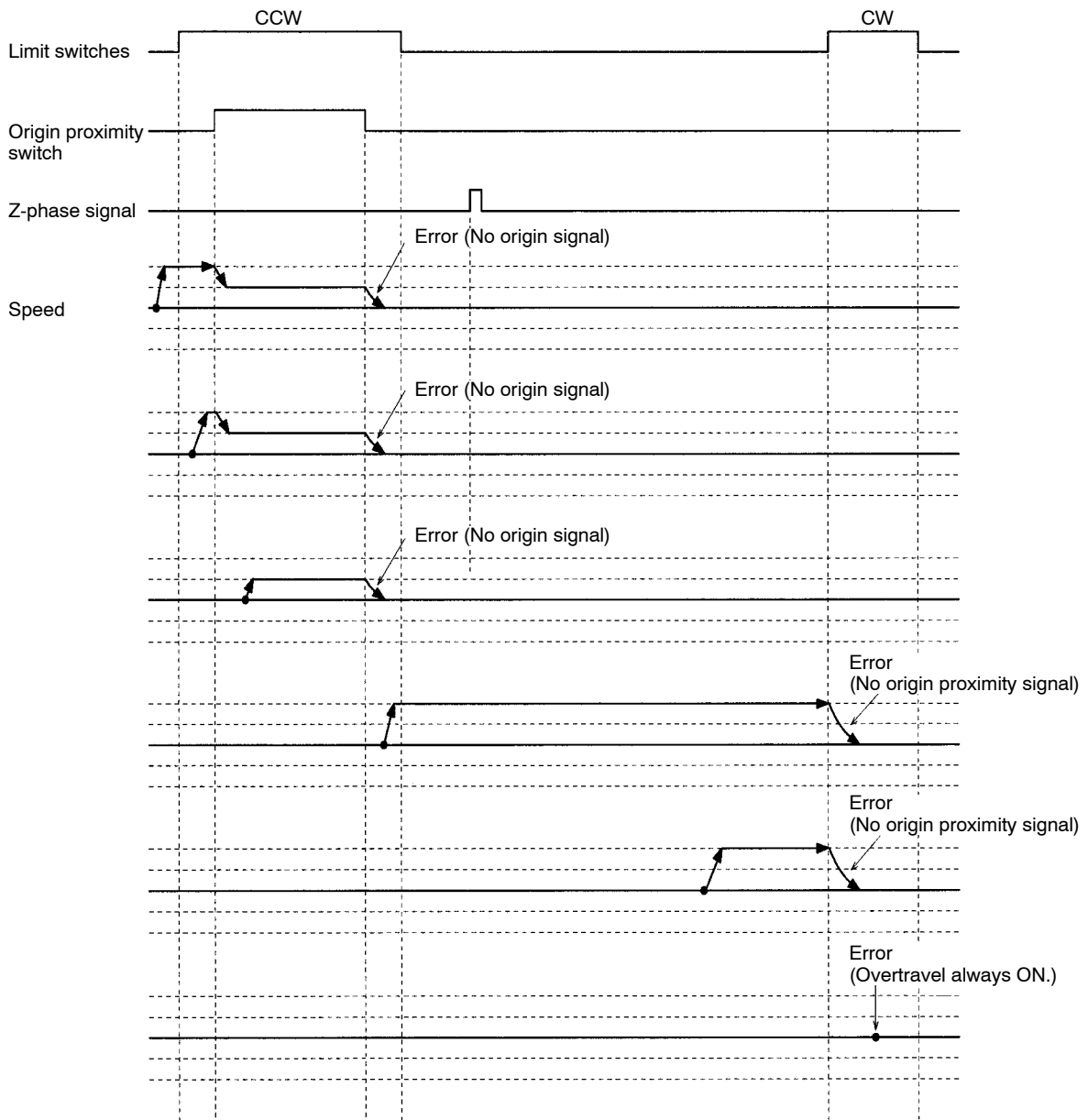
# One Direction-mode Origin Searches 2

These origin searches are performed with an origin proximity switch and the initial search direction set to CW.



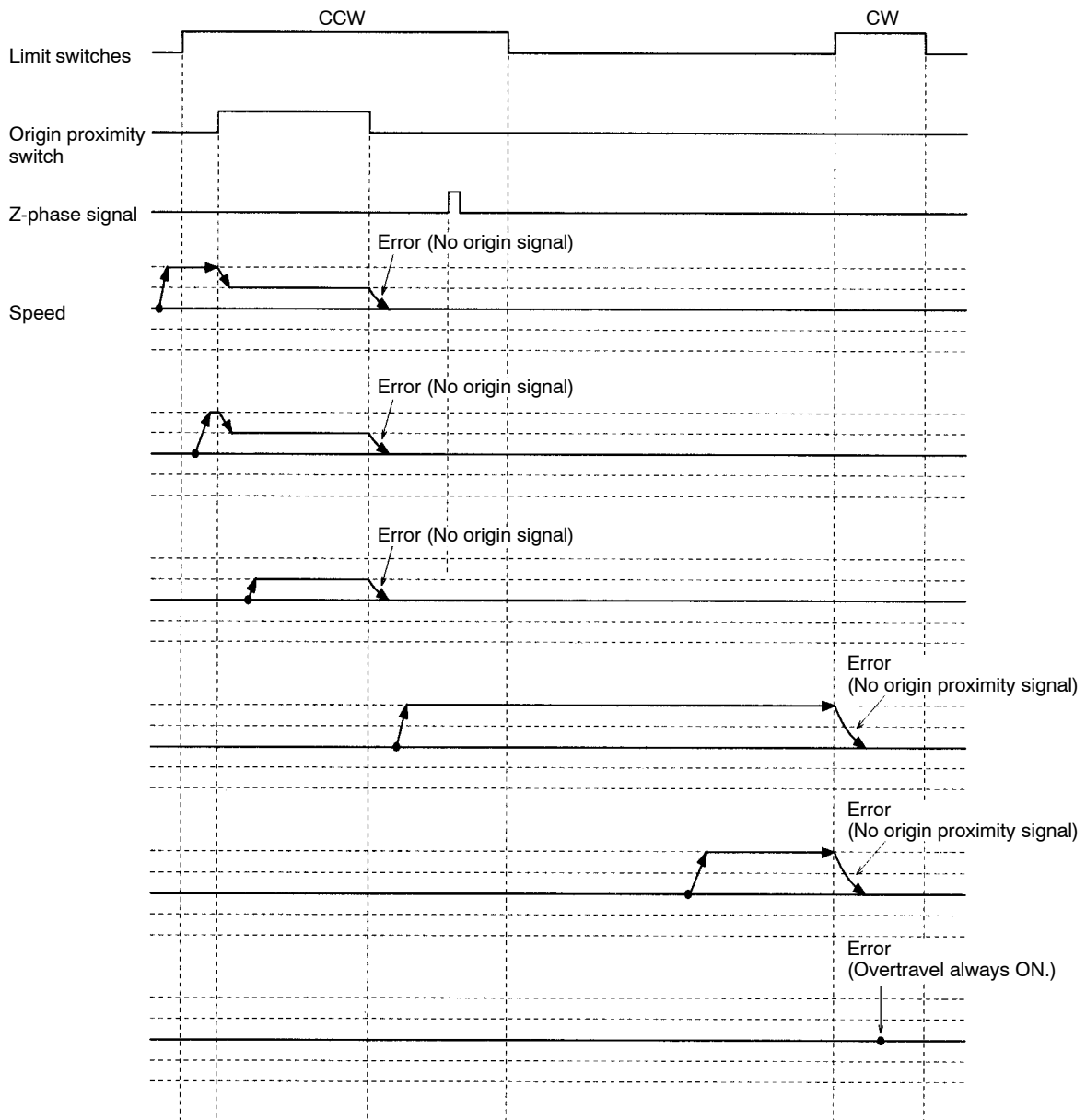
### One Direction-mode Origin Searches 3

These origin searches are performed with an origin proximity switch and the initial search direction set to CW.



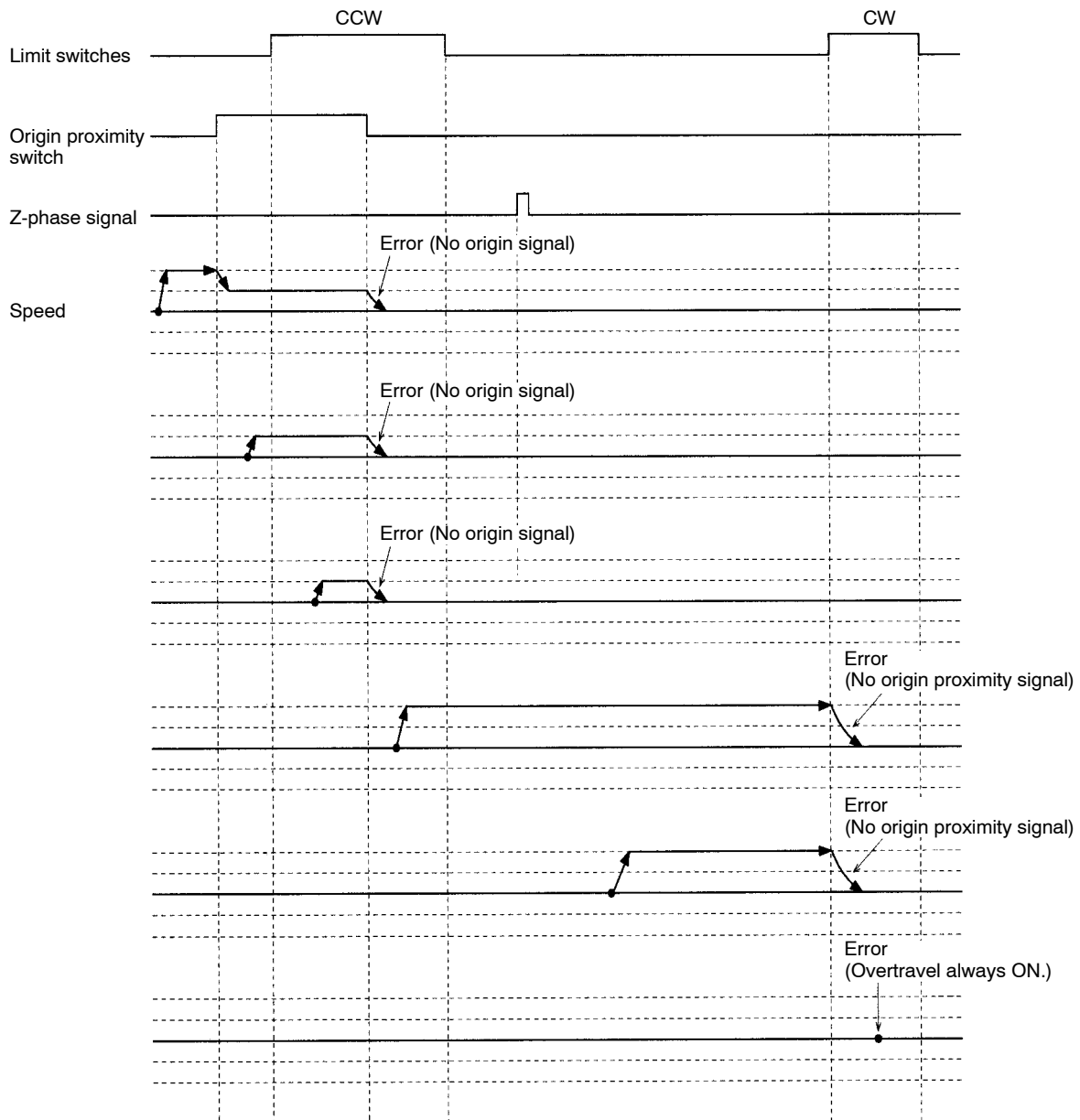
# One Direction-mode Origin Searches 4

These origin searches are performed with an origin proximity switch and the initial search direction set to CW.



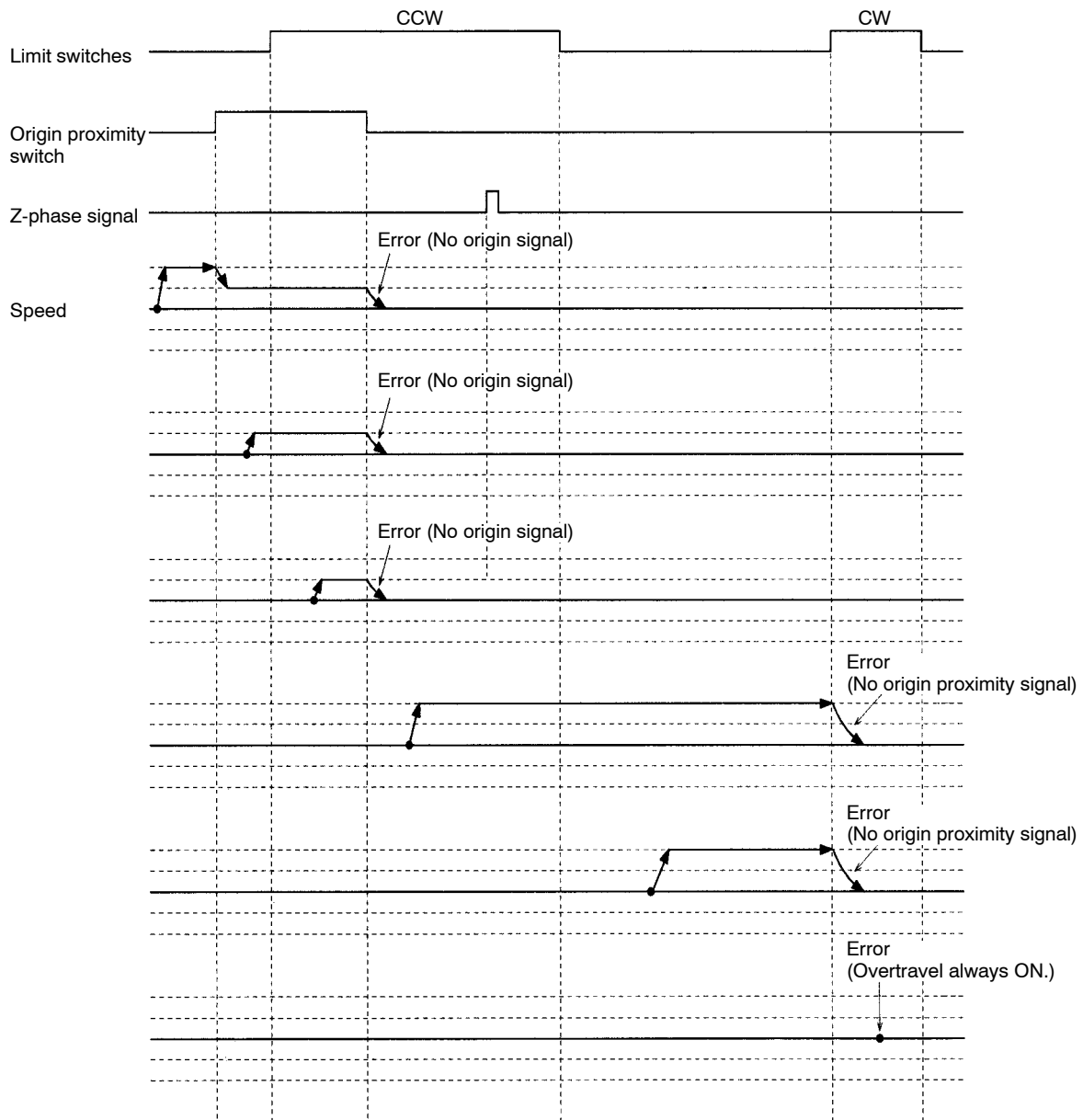
# One Direction-mode Origin Searches 5

These origin searches are performed with an origin proximity switch and the initial search direction set to CW.



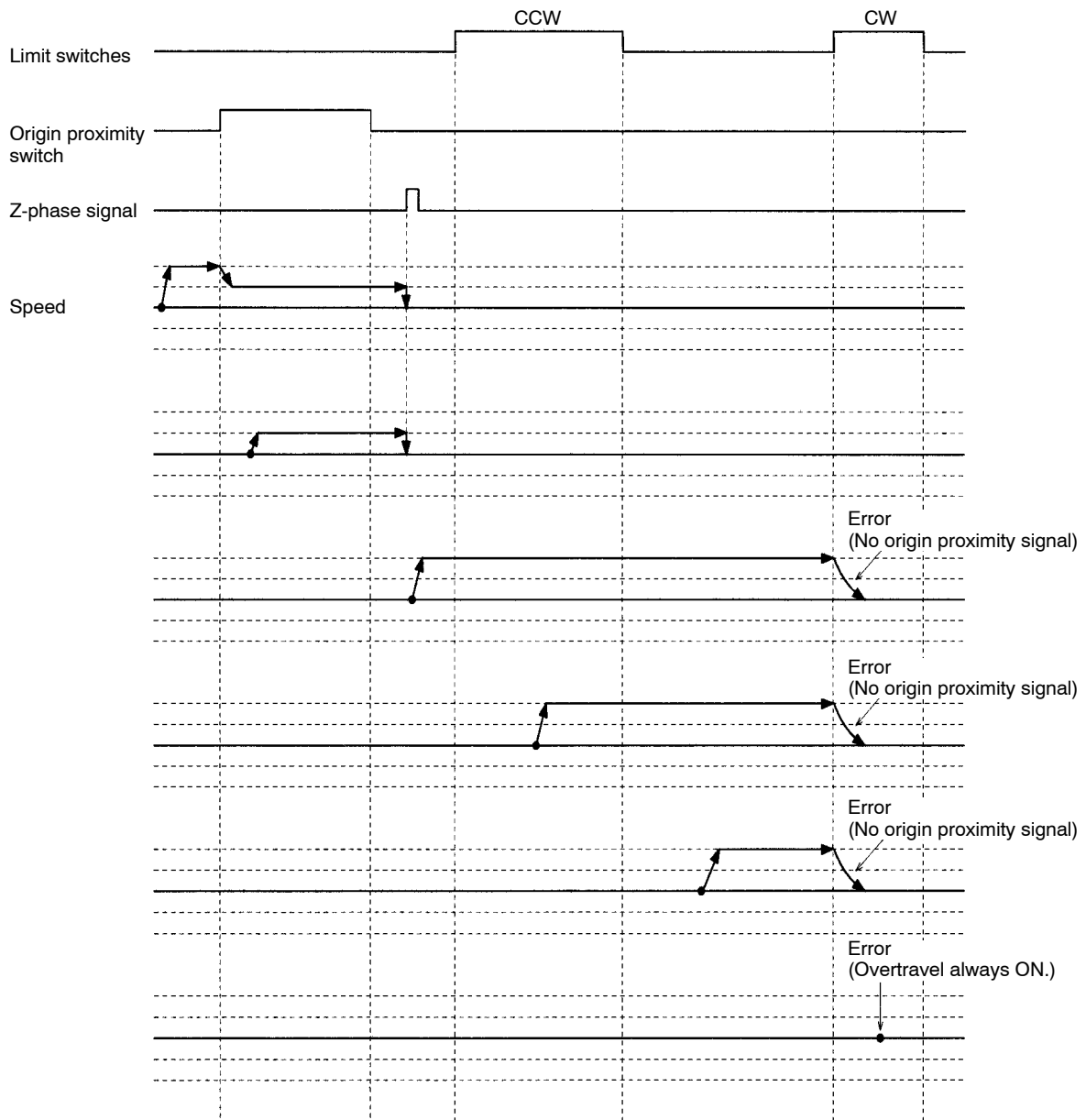
# One Direction-mode Origin Searches 6

These origin searches are performed with an origin proximity switch and the initial search direction set to CW.



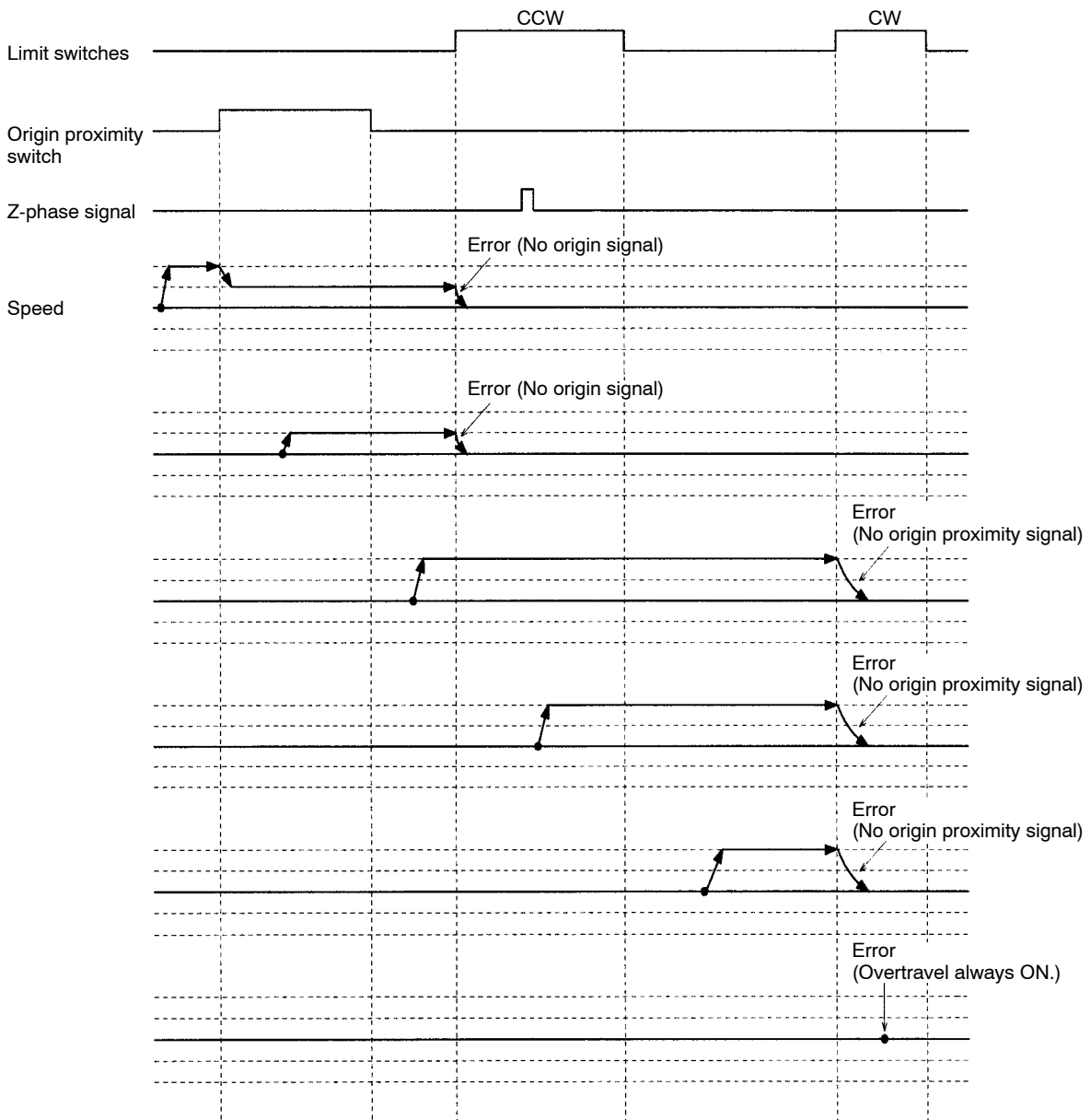
# One Direction-mode Origin Searches 7

These origin searches are performed with an origin proximity switch and the initial search direction set to CW.



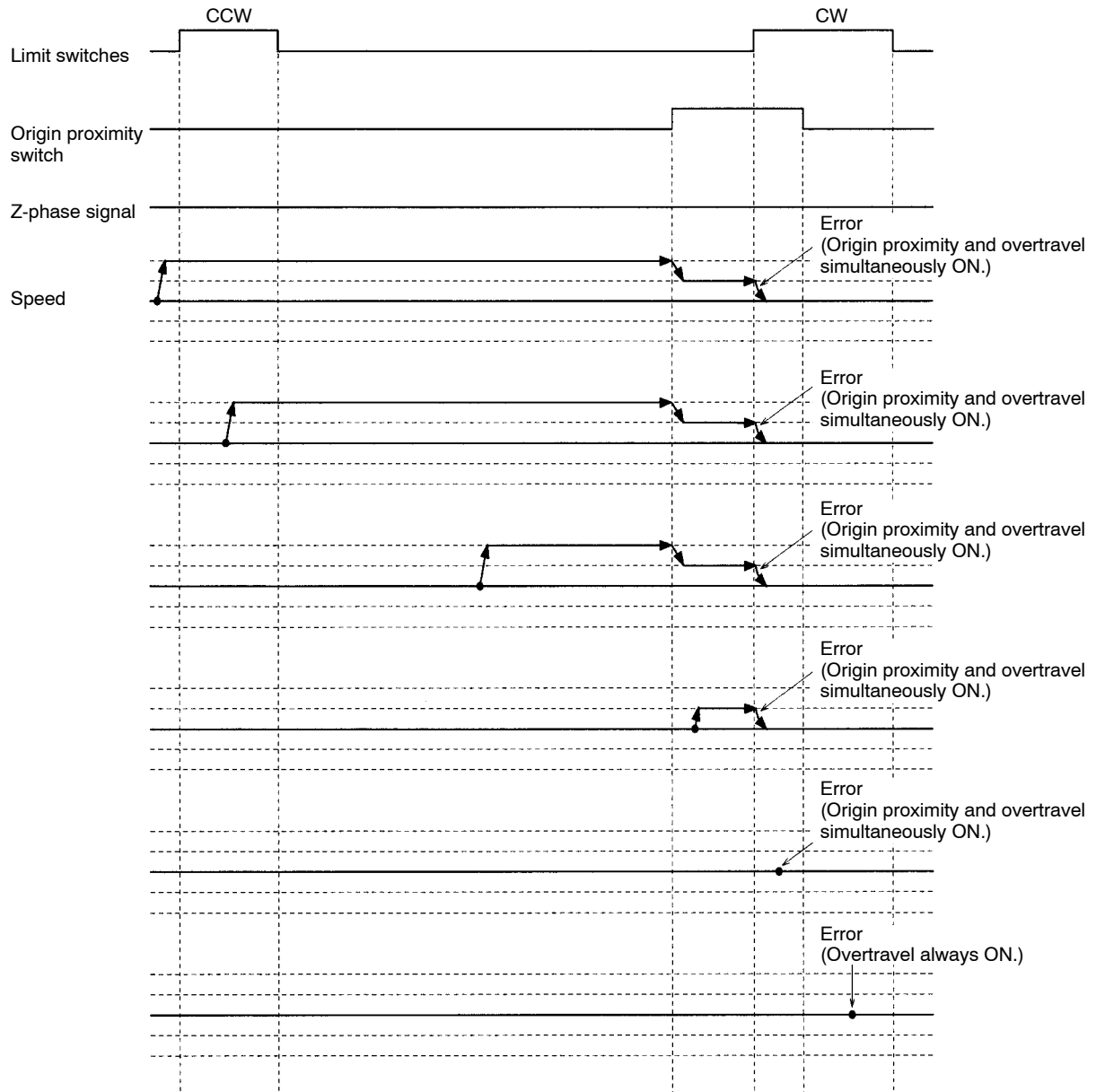
# One Direction-mode Origin Searches 8

These origin searches are performed with an origin proximity switch and the initial search direction set to CW.



## One Direction-mode Origin Searches 9

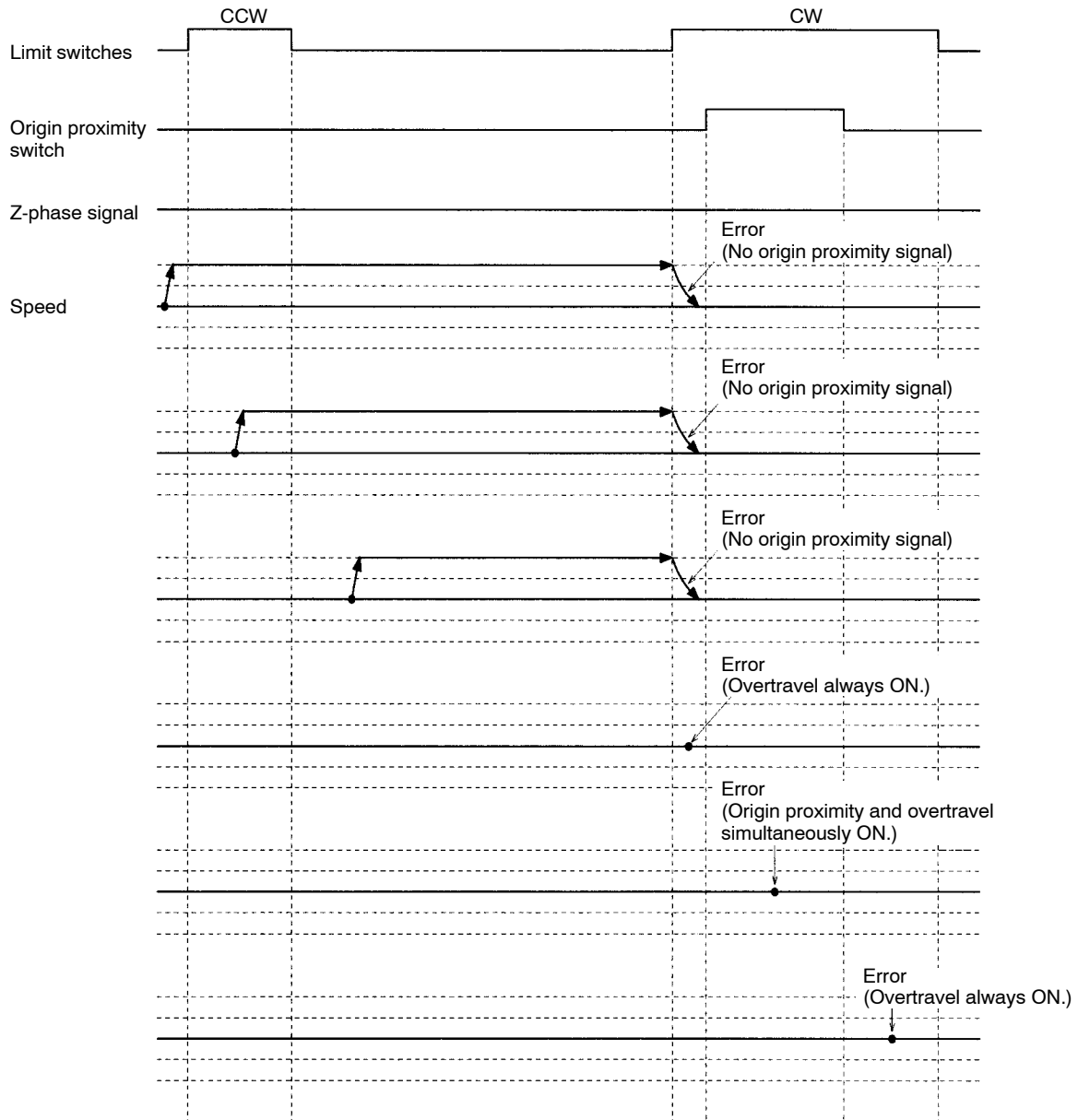
These origin searches are performed with an origin proximity switch and the initial search direction set to CW. No input for the Z phase.





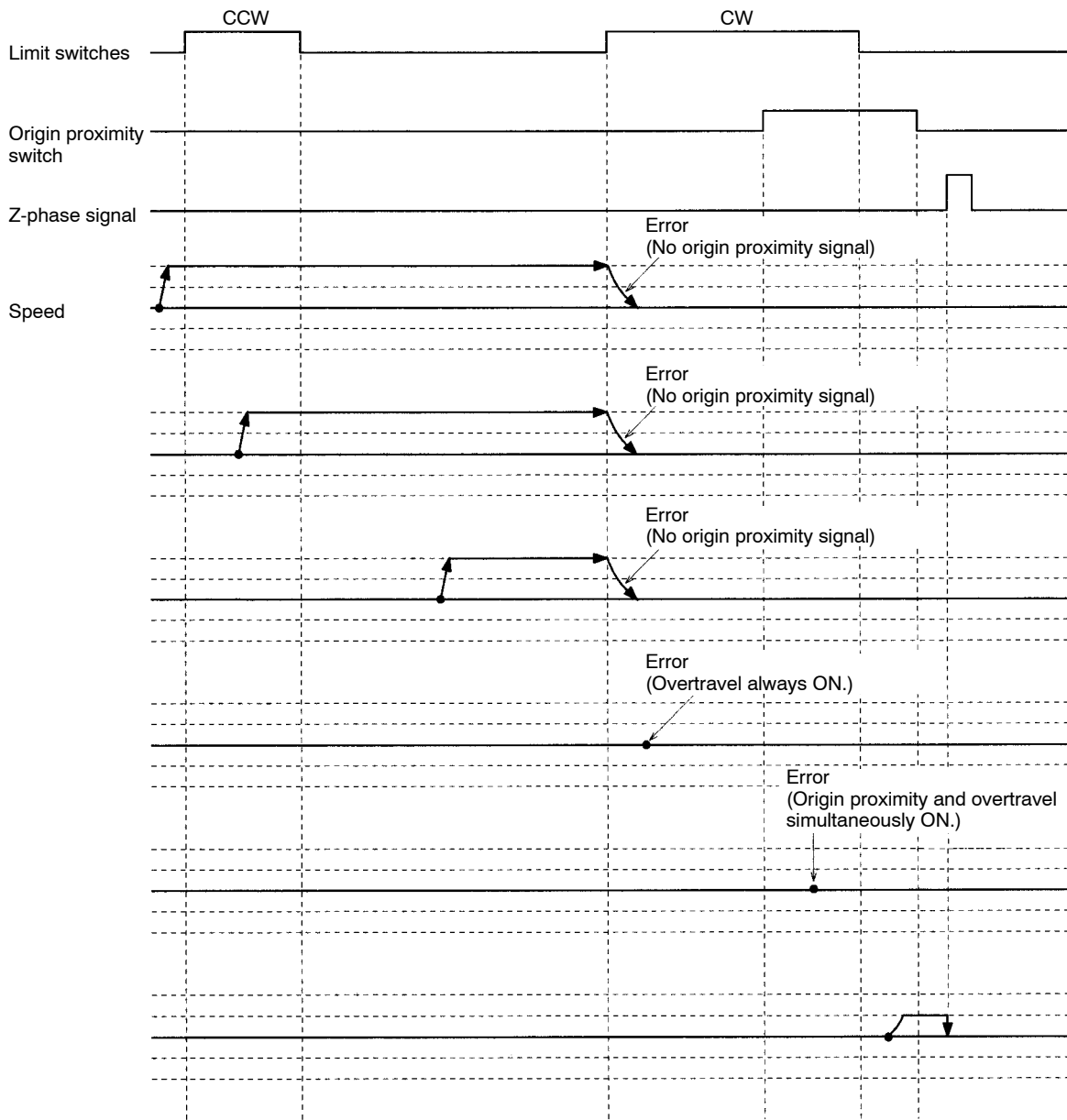
# One Direction-mode Origin Searches 10

These origin searches are performed with an origin proximity switch and the initial search direction set to CW, but no Z-phase signal.



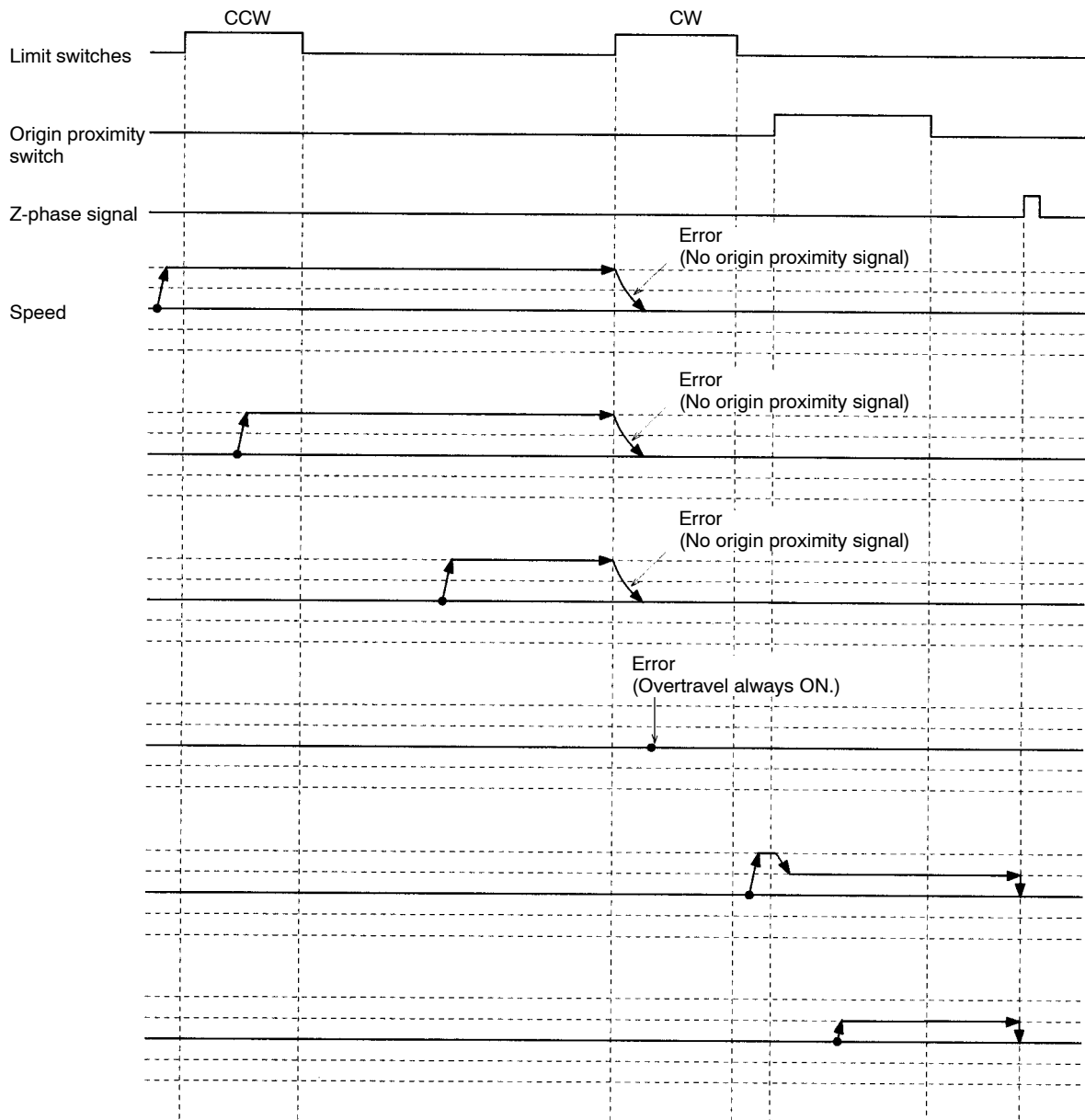
# One Direction-mode Origin Searches 11

These origin searches are performed with an origin proximity switch and the initial search direction set to CW.



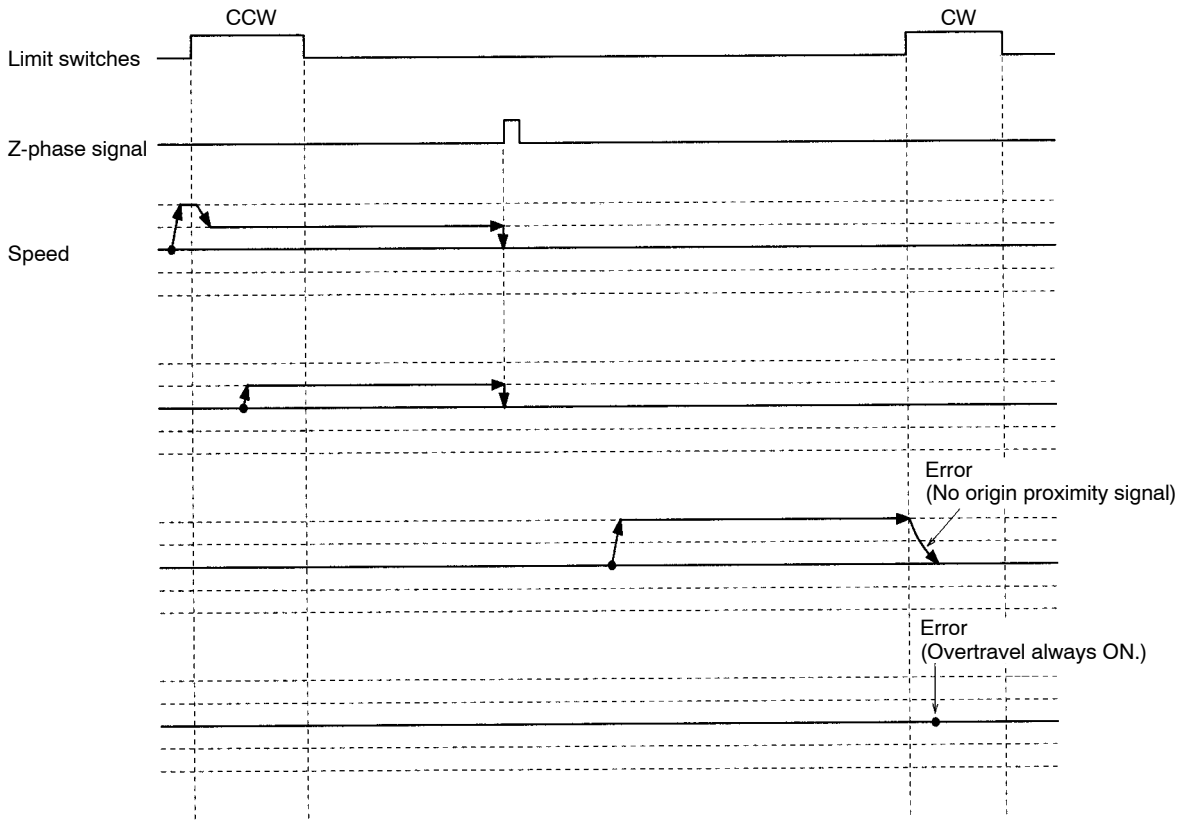
# One Direction-mode Origin Searches 12

These origin searches are performed with an origin proximity switch and the initial search direction set to CW.



### One Direction-mode Origin Searches 13

These origin searches are performed without an origin proximity switch and the initial search direction set to CW.



# **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 2
Number of tasks		1 to 2
Task 1 axes		Any combination of X and Y
Task 2 axes		

### Memory Parameters

Parameter	Setting		Setting Range
	Start address	End address	
Task 1 position data			0000 to 1999
Task 2 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		Forward by + voltage/reverse by + voltage
Emergency stop method		Immediately sets the voltage output to 0 V or stops by the counter pulse.
Encoder type		Incremental encoder or absolute encoder
Encoder resolution		1 to 65,535 ppr
Encoder polarity		Forward or reverse by an increase in the encoder.
Pulse rate		1 to 100,000
Maximum motor speed		1 to 32,767 r/min
Negative software limit		-39,999,999 to +39,999,999
Positive software limit		
Origin search method		Origin mode, limit reverse mode, or one direction mode
Origin search direction		Positive or negative direction
Origin decel. method		Using origin proximity input or both origin proximity input and limit input.
Origin proximity logic		N.O. or N.C.
Wiring check ON/OFF		ON or OFF
Wiring check time		0 to 99 (×10 ms)
Wiring check pulses		0 to 999 pulses

### Coordinate Parameters

Parameter	Setting	
	X axis	Y axis
Reference origin offset		
Workpiece origin offset		



Programmer:

Program Number:

Date:

Page:

### Feedrate Parameters

Parameter	Setting		Setting Range
	X axis	Y 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			Trapezoidal or S-shaped
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	
Error counter warning			0 to 65,000 pulses
In position			0 to 999 pulses
Position loop gain			5 to 250 (1/s)
Position loop FF gain			0 to 100 (%)
Backlash compensation value			0 to 999 pulses

# **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

<b>absolute position</b>	A position given in respect to the origin rather than in respect to the present position.
<b>acceleration/deceleration curve</b>	Curves which determine the rate of acceleration to the maximum feed rate and the rate of deceleration from the maximum feed rate.
<b>auxiliary bit</b>	A bit in the Auxiliary Area.
<b>Backplane</b>	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.
<b>basic instruction</b>	A fundamental instruction used in a ladder diagram. See <i>advanced instruction</i> .
<b>baud rate</b>	The data transmission speed between two devices in a system measured in bits per second.
<b>BCD</b>	Short for binary-coded decimal.
<b>binary</b>	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.
<b>bit</b>	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.
<b>block number</b>	Numbers used to distinguish blocks in MC programs. Block numbers are roughly equivalent to program line numbers.
<b>bus</b>	A communications path used to pass data between any of the Units connected to it.
<b>channel</b>	See <i>word</i> .
<b>comment block</b>	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."
<b>control bit</b>	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.
<b>counter</b>	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.
<b>CPU</b>	The name of the Unit in a PC that contains the main CPU and other main PC components. See also <i>central processing unit</i> .

<b>CPU Backplane</b>	A Backplane used to create a CPU Rack.
<b>CPU Rack</b>	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.
<b>CW and CCW</b>	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.
<b>CW/CCW limits</b>	Limits on the CW and CCW sides of the origin which can be internally set to restrict rotation of the shaft.
<b>data area</b>	An area in the PC's memory that is designed to hold a specific type of data.
<b>decimal</b>	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.
<b>DM Area</b>	A data area used to hold only word data. Words in the DM area cannot be accessed bit by bit.
<b>dwel time</b>	A setting that specifies the period of time during which positioning will stop before execution of the next positioning action.
<b>error counter</b>	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.
<b>feedback</b>	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.
<b>flag</b>	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.
<b>gain</b>	The increase in signal power produced by an amplifier.
<b>G language</b>	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.
<b>hunting</b>	The tendency, in servosystems, to overcompensate when the system's momentum carries it past the target position.
<b>IBM PC/AT or compatible</b>	A computer that has similar architecture to, that is logically compatible with, and that can run software designed for an IBM PC/AT computer.
<b>inching</b>	Manual feeding wherein positioning is executed one pulse at a time.
<b>incremental position</b>	A position given in respect to the present position, rather than in respect to the origin.
<b>initial position</b>	The present position when a start command is executed.

<b>in position</b>	The range within which the system is determined to be at the target position.
<b>input</b>	The signal coming from an external device into the PC. The term input is often used abstractly or collectively to refer to incoming signals.
<b>interpolation</b>	The mathematical calculation of missing values based on 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).
<b>interface</b>	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.
<b>IR Area</b>	A data area whose principal function is to hold the status of inputs coming into the system and that of outputs that are to be set out of the system. Bits and words in the IR that are used this way are called I/O bits and I/O words. The remaining bits in the IR area are work bits.
<b>least-significant (bit/word)</b>	See <i>rightmost (bit/word)</i> .
<b>leftmost (bit/word)</b>	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.
<b>linear interpolation</b>	Dual-axis, linear positioning from the present position to a point designated as the interpolation end point based on specified points.
<b>load</b>	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.
<b>local</b>	In network communications, the node or device from which communications are being viewed. See <i>remote</i> .
<b>LSS</b>	Abbreviation for Ladder Support Software.
<b>M code</b>	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.
<b>MC program</b>	A G-language program that controls the MC Unit's operation.
<b>megabyte</b>	A unit of storage equal to one million bytes.
<b>most-significant (bit/word)</b>	See <i>leftmost (bit/word)</i> .
<b>MS-DOS</b>	An operating system in common use on smaller computers.
<b>NC contacts</b>	Normally-closed contacts. A pair of contacts on a relay that open when the relay is energized.
<b>negative software limit</b>	The lower limit on the number of pulses set as a software parameter.
<b>nesting</b>	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.

<b>NO contacts</b>	Normally-open contacts. A pair of contacts on a relay that close when the relay is energized.
<b>OFF</b>	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.
<b>offline</b>	The state in which a Programming Device is not functionally connected to the CPU, although it may be connected physically.
<b>offset</b>	A positive or negative value added to a base value such as an address to specify a desired value.
<b>ON</b>	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.
<b>online</b>	The state in which a Programming Device is functionally connected to the CPU so that CPU data and programs can be monitored or accessed.
<b>online edit</b>	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.
<b>origin proximity input</b>	A signal input to indicate that the axis is near the origin.
<b>origin search</b>	An operation used to automatically move the axes to the origin or to define the origin.
<b>output</b>	The signal sent from the PC to an external device. The term output is often used abstractly or collectively to refer to outgoing signals.
<b>parameters</b>	Data which determines limits and other conditions under which an operation will be carried out.
<b>PC</b>	An acronym for Programmable Controller.
<b>PC Setup</b>	A group of operating parameters set in the PC from a Programming Device to control PC operation.
<b>positive software limit</b>	The upper limit on the number of pulses set as a software parameter.
<b>present value</b>	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.
<b>program block</b>	A unit of programming in MC programs roughly equivalent to program lines.
<b>Programmable Controller</b>	A computerized device that can accept inputs from external devices and generate outputs to external devices according to a program held in memory. Programmable 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.
<b>Programming Device</b>	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.
<b>pulses</b>	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.
<b>pulse rate</b>	The distance moved the motor shaft divided by the number of pulses required for that movement.
<b>pulse train</b>	A series of pulses output together.
<b>remote</b>	In network communications, the node or device with which communications are taking place. See <i>local</i> .
<b>retrieve</b>	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.
<b>rightmost (bit/word)</b>	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.
<b>RUN mode</b>	The operating mode used by the PC for normal control operations.
<b>servicing</b>	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.
<b>servolock</b>	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.
<b>software error</b>	An error that originates in a software program.
<b>SSS</b>	Abbreviation for SYSMAC Support Software.
<b>sub-program</b>	A group of instructions that are executed independently of the main program.
<b>target position</b>	A parameter for a positioning action that designates what position is to be reached at the completion of the action.
<b>teaching</b>	Automatically writing the present position into memory, via the Teaching Box, as the target position for the designated positioning action.
<b>transfer</b>	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.
<b>uploading</b>	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 Device is involved, the Programming Device is considered the host computer.
<b>watchdog timer</b>	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.

---

## *Glossary*

---

<b>WDT</b>	See <i>watchdog timer</i> .
<b>wiring check</b>	A check performed automatically at startup to detect wiring problems such as reversed polarity or disconnections.
<b>word</b>	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.
<b>work bit</b>	A bit that can be used for data calculation or other manipulation in programming, i.e., a 'work space' in memory. Also see <i>work word</i> .
<b>write-protect</b>	A state in which the contents of a storage device can be read but cannot be altered.
<b>zone</b>	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

- ABS Initialization Bit, 153
- ABS Soft Reset, 188
- ABS Soft Reset Bit, 154
- absolute coordinates, 105
- ABSOLUTE SPECIFICATION, G90, 105
- addresses
  - position data, specifying, 73
  - teaching, 195
- Alarm Input, 174
- allocations
  - DRV connector pins, 11
  - words, 3
- ARITHMETIC OPERATIONS, G60, 96
- automatic mode, 119
- Automatic Mode Flag, 164
- Automatic/Manual Mode Bit, 134
- axes
  - errors, 219
  - program, 74
- Axis Operating Flag, 172
- blocks
  - conditional jump, 100
  - end, 102
  - jump, 99
  - skip, 103
- Busy Flag, 119, 170

## C

- cables
  - driver, 13
  - Teaching Box, 25
  - removal, 27
- CHANGE PARAMETER, G69, 98
- CHANGE REFERENCE COORDINATE SYSTEM PV, G54, 95
- CHANGE WORKPIECE ORIGIN OFFSET, G53, 94
- CIRCULAR INTERPOLATION, G02 and G03, 78
- circular interpolation, 78, 85
  - plane, 85
  - sample program, 202
- CIRCULAR PLANE SPECIFICATION FUNCTIONS, G17, 85
- components, 2
- CONDITIONAL JUMP, G71, 100

- connections
  - examples
    - encoders, 15
    - X-axis, 9
  - personal computer, 24
  - Teaching Box, 25
- connectors
  - attaching, 8, 11
  - DRV
    - interface circuits, 20
    - pin allocations, 11
    - pin functions, 12
  - I/O
    - interface circuits, 20
    - pin functions, 8
  - MPG, pin functions, 19
  - wiring, 8
- coordinate system, 93
  - reference coordinate system setting, 93
  - workpiece coordinate system setting, 93
- coordinate system parameters, 41
- coordinates
  - absolute, 105
  - relative, 106
- Current Position Preset Bit, 152
- Cycle Start Bit, 135
- Cycle Start Received Flag, 167

## D

- Data Reception Error Flag, 161
- Data Transmission Error Flag, 159
- Deceleration Stop Bit, 148
- DIP switch settings
  - English, 4
  - Japanese, 4
- Driver Alarm Reset Bit, 157
- Driver Cables, 13
- DWELL TIMER, G04, 82

## E

- Enable MPG Bit, 155
- encoders
  - absolute, 187
    - ABS Soft Reset, 188
    - interface specifications, 189
  - connection examples, 15
  - current position preset, 178
  - origin search, 178
  - Z-phase signal, 179
- Error Counter Alarm Flag, 173

Error Flag, 163  
Error Reset Bit, 146  
errors, 214  
  axes, 219  
  indicators, 214  
  system, 215  
  tasks, 216  
  teaching, 197  
    address over, 197

Expansion Data Read Bit, 147  
Expansion Data Read Completion Flag, 163

## F

feed-rate parameters, 42  
flags  
  IORD, 65  
  IOWR, 60  
Flash Memory Write Bit, 145  
Flash Memory Write Completion Flag, 162  
Forced Block End  
  M codes 000 to 499, 109  
  M codes 500 to 999, 111  
Forced Block End Bit, 138  
formats, G functions, 71

## G

G functions, 75  
  formats, 71  
  G00: POSITIONING, 75  
  G01: LINEAR INTERPOLATION, 77  
  G02 and G03: CIRCULAR INTERPOLATION, 78  
  G04: DWELL TIMER, 82  
  G10: PASS MODE, 83  
  G11: STOP MODE, 84  
  G17: CIRCULAR PLANE SPECIFICATION FUNCTIONS, 85  
  G26: REFERENCE ORIGIN RETURN, 85  
  G27: WORKPIECE ORIGIN RETURN, 86  
  G28: ORIGIN SEARCH, 87  
  G29: ORIGIN UNDEFINED, 88  
  G30: Speed Control Axis Feeding, 88  
  G31: INTERRUPT INCHING, 89  
  G50: SELECT REFERENCE COORDINATE SYSTEM, 93  
  G51: SELECT WORKPIECE COORDINATE SYSTEM, 93  
  G53: CHANGE WORKPIECE ORIGIN OFFSET, 94  
  G54: CHANGE REFERENCE COORDINATE SYSTEM PV, 95  
  G60: ARITHMETIC OPERATIONS, 96  
  G63: SUBSTITUTION, 97  
  G69: CHANGE PARAMETER, 98  
  G70: UNCONDITIONAL JUMP, 99  
  G71: CONDITIONAL JUMP, 100  
  G72: SUBPROGRAM JUMP, 101  
  G73: SUBPROGRAM END, 101  
  G74: OPTIONAL END, 102

  G75: OPTIONAL SKIP, 103  
  G76: OPTIONAL PROGRAM PAUSE, 104  
  G79: PROGRAM END, 105  
  G90: ABSOLUTE SPECIFICATION, 105  
  G91: INCREMENTAL SPECIFICATION, 106  
  list, 70  
  mode transitions, 112  
G symbols, list, 72  
general input, stopping, program, 208

## I

I/O, connectors, 8  
I/O transfer, 33, 34  
  timing, 56  
INCREMENTAL SPECIFICATION, G91, 106  
indicators, 3, 214  
  error, 214  
Initial Setting Alarm Flag, 162  
inspection, 224  
  precautions, 224  
  tools, 224  
intelligent transfer, 33, 35, 58  
interface bits, 121  
  details, 134  
  Word n+1 Bit 01: Automatic/Manual Mode Bit, 134  
  Word n+1 Bit 02: Cycle Start Bit, 135  
  Word n+1 Bit 03: Single Block Bit, 137  
  Word n+1 Bit 04: Forced Block End Bit, 138  
  Word n+1 Bit 05: Pause Bit, 139  
  Word n+1 Bit 06: M Code Reset Bit, 140  
  Word n+1 Bit 07: Program Number Read Bit, 142  
  Word n+1 Bit 08: Teaching Address Set Bit, 142  
  Word n+1 Bit 09: Teaching Bit, 142  
  Word n+1 Bit 10: Teaching Type Bit, 143  
  Word n+1 Bit 11: Transmission Control Bit, 144  
  Word n+1 Bit 12: Reception Control Bit, 145  
  Word n+1 Bit 13: Flash Memory Write Bit, 145  
  Word n+1 Bit 14: Error Reset Bit, 146  
  Word n+1 Bit 15: Expansion Data Read Bit, 147  
  Word n+11 Bit 00: Task Error Flag (Task 1), 164  
  Word n+11 Bit 02: Automatic Mode Flag, 164  
  Word n+11 Bit 03: Manual Mode Flag, 165  
  Word n+11 Bit 04: Memory Run Flag, 165  
  Word n+11 Bit 05: Memory Run Completed Flag, 165  
  Word n+11 Bit 06: Optional Input Standby Flag, 166  
  Word n+11 Bit 07: M Code Reset Standby Flag, 166  
  Word n+11 Bit 08: M Strobe Flag, 167  
  Word n+11 Bit 09: Cycle Start Received Flag, 167  
  Word n+11 Bit 12: Teaching Address Setting Completed, 168  
  Word n+11 Bit 13: Teaching Completed Flag, 168  
  Word n+11 Bit 14: Teaching Error Flag, 169  
  Word n+11 Bit 15: Teaching Address Over Flag, 169  
  Word n+16 Bit 00: Reference Origin Flag, 170  
  Word n+16 Bit 01: Busy Flag, 170  
  Word n+16 Bit 02: Servo Lock ON Flag, 171  
  Word n+16 Bit 03: No Origin Flag, 171  
  Word n+16 Bit 04: Axis Operating Flag, 172

Word n+16 Bit 05: Positioning Completed Flag, 173  
Word n+16 Bit 06: Error Counter Alarm Flag, 173  
Word n+16 Bit 07: Alarm Input, 174  
Word n+16 Bits 08 to 15: Zone Flags, 174  
Word n+19 Bit 15: MPG Operation Flag, 174  
Word n+3 Bits 11 to 15: Optional Inputs, 147  
Word n+5 Bit 00: Deceleration Stop Bit, 148  
Word n+5 Bit 01: Origin Search Bit, 149  
Word n+5 Bit 02: Reference Origin Return Bit, 150  
Word n+5 Bit 03: Jogging Bit, 151  
Word n+5 Bit 05: Current Position Preset Bit, 152  
Word n+5 Bit 06: ABS Initialization Bit, 153  
Word n+5 Bit 07: ABS Soft Reset Bit, 154  
Word n+5 Bit 08: Enable MPG Bit, 155  
Word n+5 Bit 09: Servo Lock Bit, 156  
Word n+5 Bit 10: Servo Free Bit, 157  
Word n+5 Bit 11: Driver Alarm Reset Bit, 157  
Word n+5 Bit 12: Override Setting Bit, 158  
Word n+5 Bit 13: Jog Direction Bit, 152  
Word n+5 Bits 14 and 15: MPG Ratio Specifiers, 156  
Word n+8 Bit 06: Transmission Completion Flag, 159  
Word n+8 Bit 07: Data Transmission Error Flag, 159  
Word n+8 Bit 08: Reception Completion Flag, 160  
Word n+8 Bit 09: Data Reception Error Flag, 161  
Word n+8 Bit 10: Flash Memory Write Completion Flag, 162  
Word n+8 Bit 11: Teaching Box Enabled Flag, 162  
Word n+8 Bit 12: Teaching Box Protected Flag, 162  
Word n+8 Bit 13: Initial Setting Alarm Flag, 162  
Word n+8 Bit 14: Error Flag, 163  
Word n+8 Bit 15: Expansion Data Read Completion Flag, 163

interface circuits

DRV connectors, 20  
I/O connectors, 20

interlocks, M code, 107, 108

interpolation acceleration time, 205

INTERRUPT INCHING, G31, 89

IORD, 64

execution timing, 67  
flags, 65

IOWR, 59

execution timing, 62  
flags, 60

## J—L

Jog Direction Bit, 152

Jogging Bit, 151

ladder programs

automatic mode, 120  
executing G-language programs, 210  
teaching, 196

limit switches, 179

LINEAR INTERPOLATION, G01, 77

linear interpolation, 77  
sample program, 201

## M—N

M Code Reset Bit, 140

M Code Reset Standby Flag, 166

M codes, 107, 108

0 to 499, 108  
000 to 499, clearing, 109  
500 to 999, 108  
clearing, 111  
examples, 108  
resets, 109  
timing chart, 107

M Strobe Flag, 167

manual mode, 118

Manual Mode Flag, 165

MC Support Software, 24

origin search parameters, 179  
setting parameters, 31  
task configuration, 6

mechanical parameters, 38

memory management parameters, 38

Memory Run Completed Flag, 121, 165

Memory Run Flag, 120, 165

modes

automatic, 119  
manual, 118

monitoring data, 36, 48

mounting, 5

MPG Operation Flag, 174

MPG Ratio Specifiers, 156

No Origin Flag, 171

numerical values, arithmetic operations, 96

## O

OPTIONAL END, G74, 102

Optional Input Standby Flag, 166

Optional Inputs, 147

OPTIONAL PROGRAM PAUSE, G76, 104

OPTIONAL SKIP, G75, 103

origin, workpiece coordinate system, 94

origin proximity switch, 179

ORIGIN SEARCH, G28, 87

origin search, 179, 180

parameters, 179  
patterns, 180  
basic, 180  
one direction mode, 185, 252  
reverse mode, 181, 239

sensors, 178

settings, 179

one direction mode, 179

reverse mode, 179

set origin at power ON mode, 179

Origin Search Bit, 149  
ORIGIN UNDEFINED, G29, 88  
Override Setting Bit, 158

## P

parameters, 30, 31, 205  
  axis, 30  
  changing settings, 98  
  coordinate system, 41  
    list, 32  
  feed rate, 42  
    list, 32  
  interpolation acceleration time, 205  
  mechanical, 38  
    list, 31  
  memory management, 38  
    list, 31  
  origin search, 179  
  servo, 47  
    list, 32  
  setting, 31  
  system, 30, 36, 37  
  Unit, 37  
  zone, 44  
    list, 32  
PASS MODE, G10, 83  
pass mode, 83  
Pause Bit, 139  
PC data area interface, 3  
personal computers, connections, 24  
pins  
  DRV connectors, allocations, 11  
  functions  
    DRV connectors, 12  
    I/O connectors, 8  
    MPG connectors, 19  
  I/O connectors, allocations, 8  
position data, 36, 96, 194  
  address, specifying, 73  
  arithmetic operations, 96  
  calculating, 206  
  specifying using register, 73  
    direct addressing, 74  
    indirect addressing, 73  
  substitution, 97  
  teaching, 194  
POSITIONING, G00, 75  
Positioning Completed Flag, 173  
precautions  
  during inspection, 224  
  general, xiii  
present value, reference coordinate system, 95  
PROGRAM END, G79, 105  
program number, 74  
Program Number Read Bit, 142

Programmable Controller. *See* PCs  
programs, 208  
  axes, 74  
  end, 105  
  executing from ladder programs, 210  
  format, 74  
  parameters, 74  
  pause, 104  
  program number, 74  
  stopping  
    general input, 208  
    substituting position data, 209  
PTP control, 75  
  sample program, 200

## R

reception, 54  
  data, examples, 54  
Reception Completed Flag, 52  
Reception Completion Flag, 160  
Reception Control Bit, 52, 145  
Reception Error Flag, 52  
reference coordinate system, changing present value, 95  
reference origin, 85  
Reference Origin Flag, 170  
REFERENCE ORIGIN RETURN, G26, 85  
Reference Origin Return Bit, 150  
registers, 96  
  arithmetic operations, 96  
  indirect addressing, sample program, 203  
  specifying, 73  
  substitution, 97  
relative coordinates, 106

## S

SELECT REFERENCE COORDINATE SYSTEM, G50, 93  
SELECT WORKPIECE COORDINATE SYSTEM, G51, 93  
sensors, origin search, 178  
Servo Free Bit, 157  
Servo Lock Bit, 156  
Servo Lock ON Flag, 171  
servo parameters, 47  
servodrivers, connectors, 11  
Single Block Bit, 137  
soft reset, absolute encoder, 188  
special data, 36, 51  
SPEED CONTROL AXIS FEEDING, G30, 88  
STOP MODE, G11, 84  
stop mode, 84

stopping  
  position data, substituting, 209  
  program  
    general input, 208  
    substituting position data, 209  
SUBPROGRAM END, G73, 101  
SUBPROGRAM JUMP, G72, 101  
subprograms  
  end, 101  
  jump, 101  
SUBSTITUTION, G63, 97  
system  
  errors, 215  
  parameters, 30, 36, 37

## T

Task Error Flag, 164  
tasks  
  configuration, 6  
  errors, 216  
teaching, 194, 196  
  addresses, 194  
  errors, 197  
  setting, 195  
  errors, 197  
  ladder programs, 196  
  Teaching Box, 195  
Teaching Address Over Flag, 169  
Teaching Address Set Bit, 142  
Teaching Address Setting Completed, 168  
Teaching Bit, 142  
Teaching Box Enabled Flag, 162  
Teaching Box Protected Flag, 162  
Teaching Completed Flag, 168  
Teaching Error Flag, 169  
Teaching Type Bit, 143

timing  
  execution  
    IORD, 67  
    IOWR, 62  
  I/O transfer, 56  
timing charts, 225  
  control bits, 225  
  control flags, 225  
transfer time, 56  
transmission, 52  
  data, examples, 52  
Transmission Completed Flag, 52  
Transmission Completion Flag, 159  
Transmission Control Bit, 52, 144  
Transmission Error Flag, 52  
transmission/reception, 33  
  I/O transfer, 34  
  intelligent transfer, 35

## U

UNCONDITIONAL JUMP, G70, 99  
unit number, setting, 3  
unit parameters, 37

## W

wait time, 82  
workpiece coordinate system, changing origin, 94  
workpiece origin, 86  
  offset, sample program, 204  
WORKPIECE ORIGIN RETURN, G27, 86

## Z

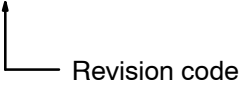
Z-phase signal, encoder, 179  
Zone Flags, 174  
zone parameters, 44





## Revision History

A manual revision code appears as a suffix to the catalog number on the front cover of the manual.

Cat. No. W315-E1-03  


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 1996	Original production
2	January 1999	Revised precautionary items and wording in <i>PRECAUTIONS</i> . <b>Pages 4, 146, 177, 205, etc.:</b> Added information on the Absolute Encoders. <b>Page 5:</b> Added <i>1-6 Changing the Task Configuration</i> . <b>Pages 12, 13, etc.:</b> Added information for dedicated driver cables and revised connection example diagrams. Minor changes and additions of supplementary information throughout the manual.
3	June 2010	Front matter: Added warranty limitation of liability information. <b>Pages 8 and 11:</b> Added connector model numbers and related information.





**OMRON Corporation** Industrial Automation Company  
Tokyo, JAPAN

Contact: [www.ia.omron.com](http://www.ia.omron.com)

**Regional Headquarters**

**OMRON EUROPE B.V.**

Wegalaan 67-69-2132 JD Hoofddorp  
The Netherlands

Tel: (31)2356-81-300/Fax: (31)2356-81-388

**OMRON ELECTRONICS LLC**

One Commerce Drive Schaumburg,  
IL 60173-5302 U.S.A.

Tel: (1) 847-843-7900/Fax: (1) 847-843-7787

**OMRON ASIA PACIFIC PTE. LTD.**

No. 438A Alexandra Road # 05-05/08 (Lobby 2),  
Alexandra Technopark,  
Singapore 119967

Tel: (65) 6835-3011/Fax: (65) 6835-2711

**OMRON (CHINA) CO., LTD.**

Room 2211, Bank of China Tower,  
200 Yin Cheng Zhong Road,  
PuDong New Area, Shanghai, 200120, China

Tel: (86) 21-5037-2222/Fax: (86) 21-5037-2200

**Authorized Distributor:**

© OMRON Corporation 1996 All Rights Reserved.  
In the interest of product improvement,  
specifications are subject to change without notice.

**Cat. No. W315-E1-03**

Printed in Japan  
0610