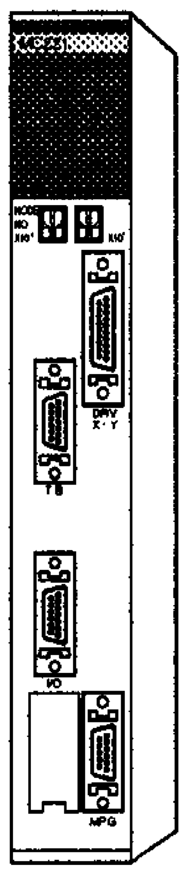
CV500-MC221/421 Motion Control Unit

Operation Manual: Introduction

Produced June 1995



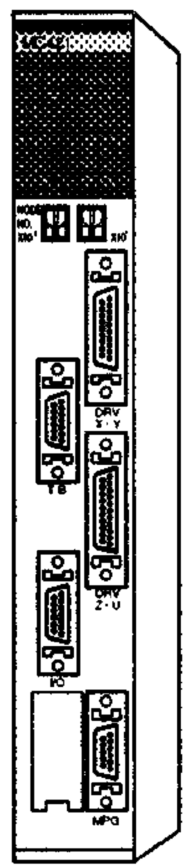


TABLE OF CONTENTS

TABLE OF CONTENTS

About this Manual:

This manual provides an introduction to the features and basic operation of the CV500-MC221 and CV500-MC421 Motion Control Units and includes the sections described below.

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

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

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

Section 1 describes fundamentals necessary for understanding this manual and for the successful operation of an MC Unit.

Section 2 describes the features and functions of the MC Unit.

Section 3 provides information on the servo system, basic mechanisms for positioning, and precautions to be heeded in using the system.

Section 4 provides information on the system configuration necessary for operating the MC Unit.

Section 5 provides information on position control as performed by the MC Unit.

Section 6 introduces the “G” language used for position control with the MC Unit.

Section 7 provides information on startup procedures, system configuration, and positioning operations using testing equipment, as well as explanations on the interface area necessary for creating ladder programs.

Section 8 explains the procedures involved in conducting tests using test equipment

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

SECTION 1
Introduction

This section describes fundamentals necessary for understanding this manual and for the successful operation of an MC Unit.

1-1 CV500-MC421 vs. CV500-MC-221

There are two CV-series MC Units available: the CV500-MC421 controls a maximum of four axes and the CV500-MC221 controls a maximum of two axes. In all other ways the basic features and functions of the CV500-MC221 are the same as those of the CV500-MC421. For further details, refer to *Section 2 Features*.

Both of the MC Units are supported by the following two peripheral devices:

- MC Support Software: CV500-ZN3AT1
- Teaching Box: CVM1-PRS71

The descriptions in this manual are based on the CV500-MC421, with supplementary remarks given if differences exist between the two models.

Note Although the Teaching Box and MC Support Software can be used for either model of MC Unit, pay attention to the following points when using the CV500-MC221.

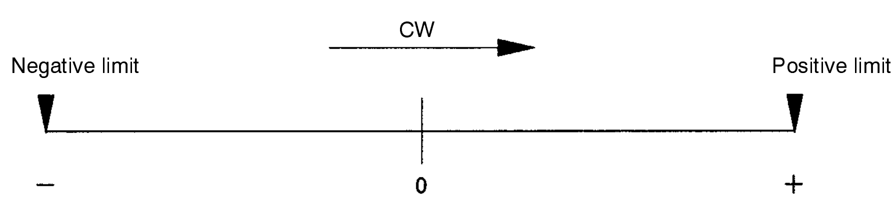
Item	Differences for CV500-MC221
The present position is displayed for four axes by the MC Support Software or Teaching Box.	"0" is displayed for the Z and U axes.
I/O signals monitored from the MC Support Software or Teaching Box are displayed for four axes.	"1" (ON) is displayed for the following signals for the Z and U axes. (The X and Y axes will operate normally.) CCW limit input CW limit input Immediate stop input Driver alarm input
If systems parameters or programs that control the Z and U axes are created and transferred by mistake, error will occur (see right column).	When the servo lock function is executed from the Teaching Box or ladder-diagram program, a Driver Alarm error will occur for the Z or U axes, prohibiting any operations on the Z or U axis.

1-2 CW and CCW

The abbreviations "CW" and "CCW" used in this manual to describe the operation of the MC Unit are defined as follows:

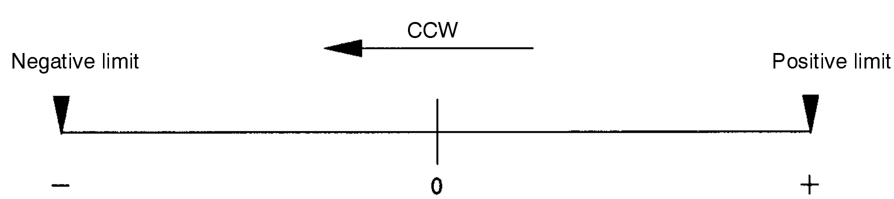
CW (Clockwise)

Clockwise is the direction in which the present position increases.



CCW (Counterclockwise)

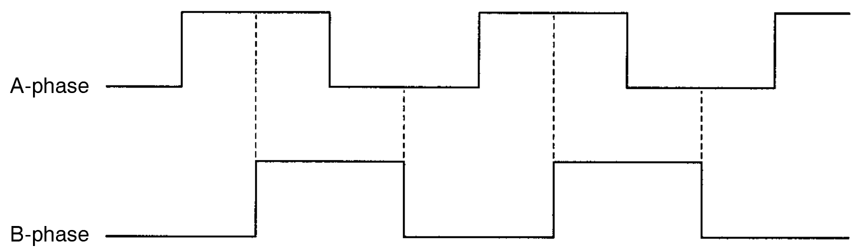
Counterclockwise is the direction in which the present position decreases.



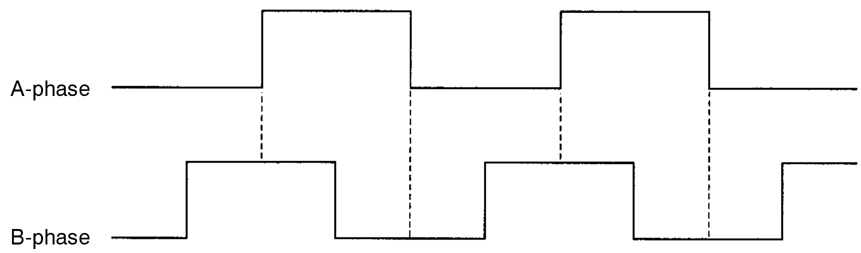
1-3 Feedback Pulse

Standard OMRON servomotors are designed for an advanced A-phase for forward rotation and an advanced B-phase for reverse rotation. The MC Unit is designed to comply with this phase advancement, allowing OMRON Connecting Cables to be connected without modification.

Forward Rotation (Positive Speed Command)



Reverse Rotation (Negative Speed Command)



When using servomotors by other makers, check carefully the encoder specifications. If the definition differs from the ones given above, take one of the following actions:

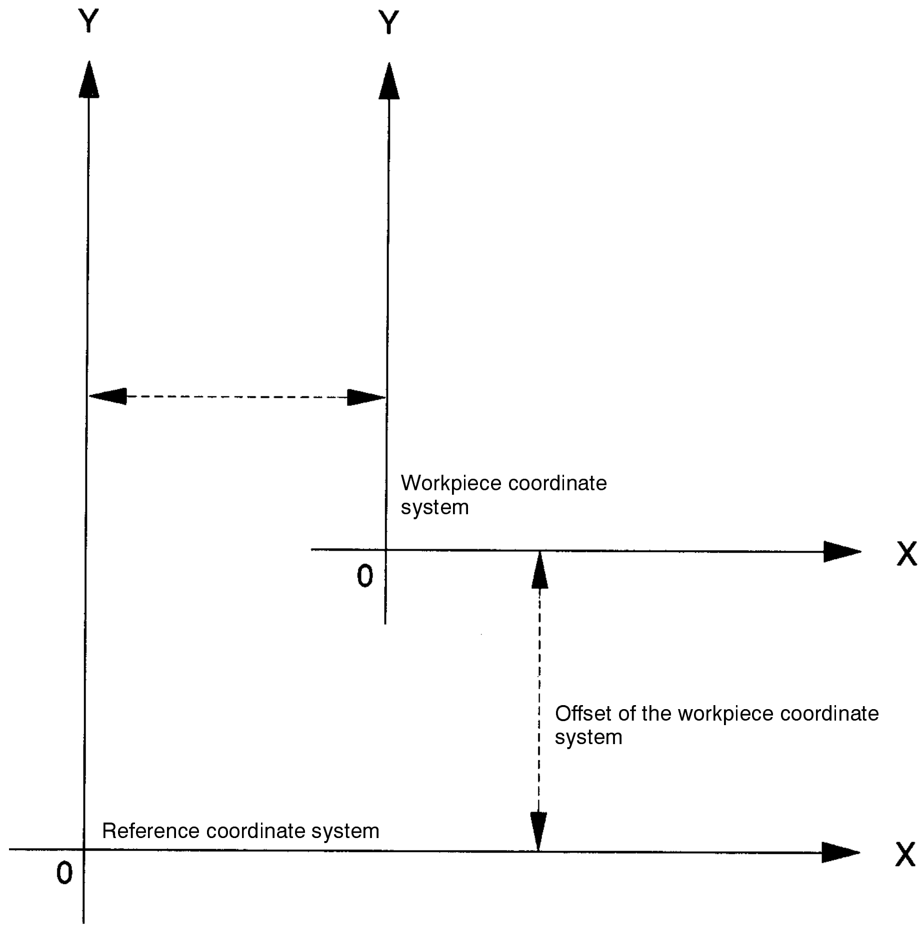
- Reverse the B-phase wiring between the MC Unit and the servodriver. (Reverse the +B terminal and the –B terminal.)
- Set the machine parameter “encoder polarity” in the system parameters to “reverse rotation for encoder increase.” It is initially set to the “forward rotation at the encoder increase.”

Note For more information on the machine parameter “encoder polarity,” refer to *MC Support Software Operation Manual*.

1-4 Coordinate Systems

Positioning operations using the MC Unit are performed based on two coordinate systems: a reference coordinate system and a workpiece coordinate system.

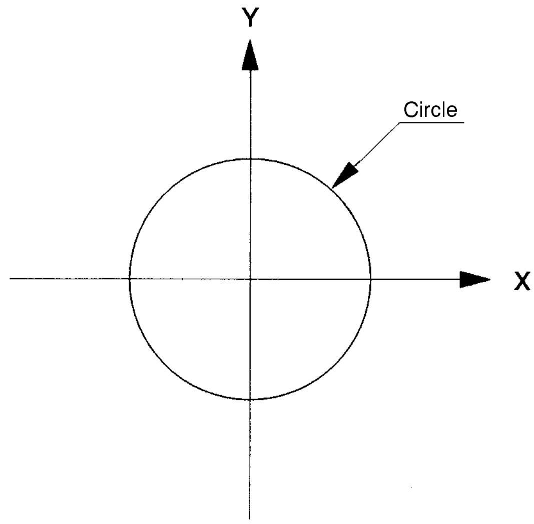
The reference coordinate system is the most fundamental one for positioning operations. The workpiece coordinate system is offset from the reference coordinate system by a specified amount, allowing the user to freely set a coordinate system.



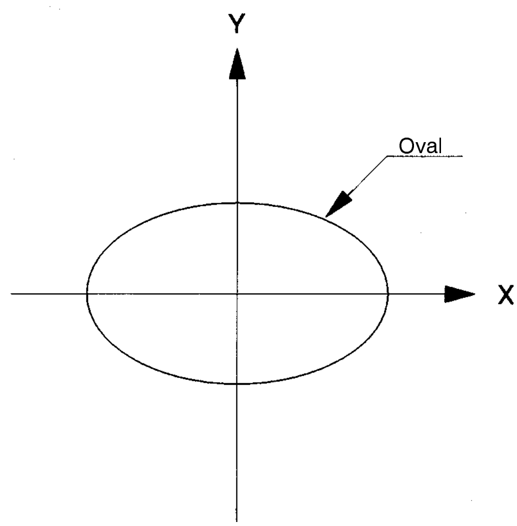
1-5 Circular Arc Interpolation and Pulse Rate

The MC Unit performs interpolation based on the assumption that the pulse rates for all axes (machine travel rate per feedback pulse) are all the same.

For circular interpolation, if the pulse rate of the X axis is equal to that of the Y axis, a circle is drawn.



If the pulse rate of the X axis differs from that of the Y axis, the axial locus turns out to be oval.



For linear interpolation, a straight line will be drawn regardless of whether or not the the pulse rate of one axis differs from that of the other.

1-6 Applicable Machines

The MC Unit has been developed for use in simple positioning applications using servomotors. Depending on the machine being controlled, the accuracy of the MC Unit should be about five to tens times higher that the machine being controlled. Applicable machines are as follows:

1, 2, 3...

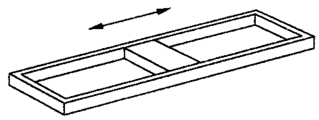
1. Conveyor machinery: X/Y tables, palletizers/depalletizers, loaders/unloaders, etc.
2. Assembling machinery: Simple robots (including orthogonal robots), simple automated assembling machines, etc.

Terminology

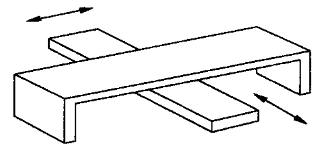
Palletizers/depalletizers: Devices used for loading goods onto pallets or for unloading them from pallets.

Loaders/unloaders: Devices that have shelves corresponding with the steps of a multistep press and used for inserting or removing all the materials at one time.

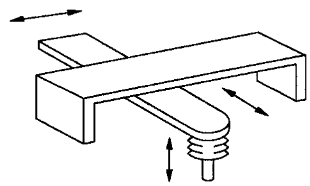
Orthogonal Robots



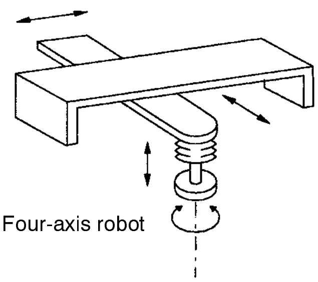
One-axis robot



Two-axis robot



Three-axis robot

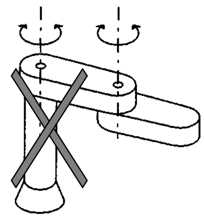


Four-axis robot

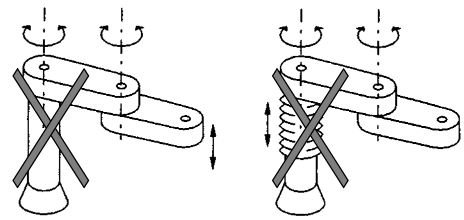
Horizontal Articulated Robots and Cylindrical Robots

The MC Units were not developed for controlling horizontal articulated robots and cylindrical robots.

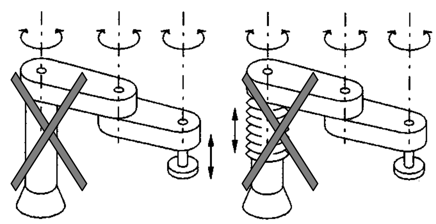
Horizontal Articulated Robots



Two-axis type

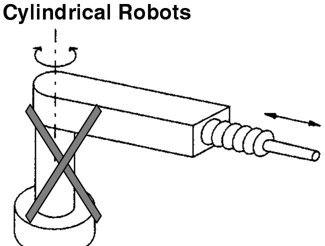


Three-axis type

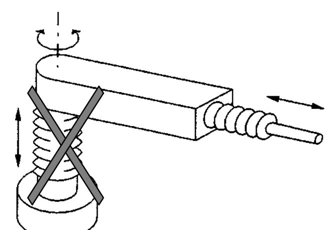


Four-axis type

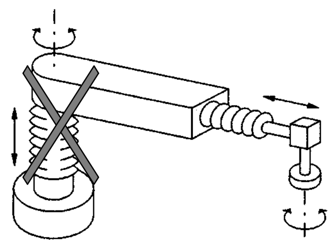
Cylindrical Robots



Two-axis type



Three-axis type



Four-axis type

1-7 Wiring Precautions

Heed the following precautions when wiring the MC Unit to the servodrivers and motors.

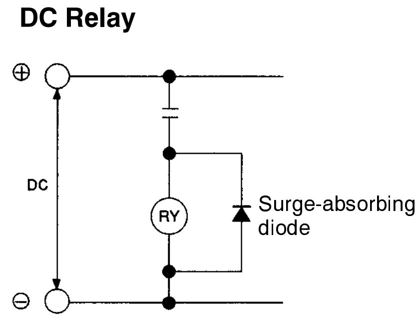
Electronically controlled equipment may malfunction because of noise generated by power supply lines or external loads. Such malfunctions are difficult to reproduce; hence, determining the cause often requires a great deal of time. The following tips should aid in avoiding noise malfunction and improving system reliability.

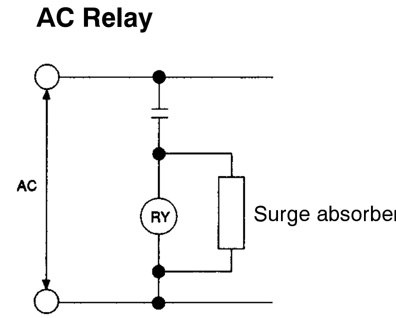
Use electrical wires or cables of designated sizes as specified in the instruction manual for the servodriver. Use larger size cables for the FG lines of the PC or the driver and ground them over the shortest possible distance.

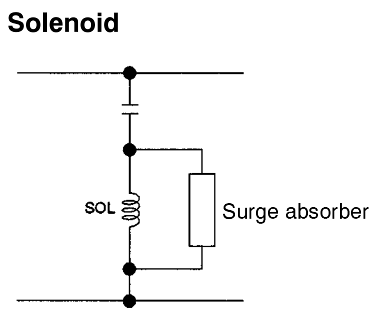
Separate power cables (AC power supply lines and motor power supply lines) from control cables (pulse output lines and external input signal lines). Do not group the two types of cable together or place them in the same conduit. Use shielded cables for control lines.

For inductive loads such as relays or solenoid valves, connect surge absorbers.

Note Connect a surge-absorbing diode or surge absorber close to the relay. Use a surge-absorbing diode with a voltage tolerance at least five times greater than the circuit voltage.





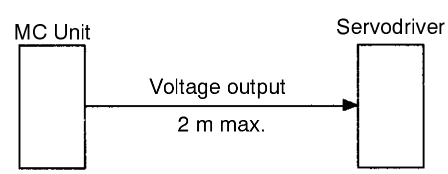


Noise may be generated on the power supply line if the same power supply line is used for an electric welder or electrical discharge unit. Provide an insulating transformer and a line filter in the power supply section to remove such noise.

It is recommended that twisted-pair cables be used for power supply lines. Use adequate grounds (i.e., to 100 Ω or less) with a wire cross section of 1.25 mm^2 or greater.

Use twisted-pair shielded cables for the control voltage output signals and the feedback input signals.

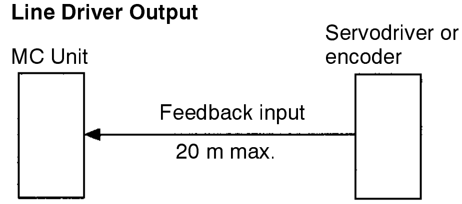
For the control voltage output signals, wire a maximum of two meters between the MC Unit and the servodriver.



Wire the distance between the MC Unit and the feedback pulse generator (i.e., encoder or the servodriver) as follows:

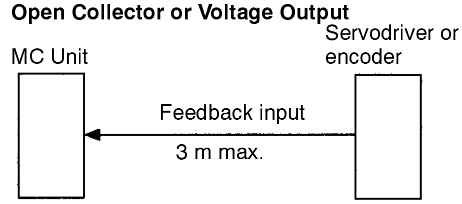
- When the feedback pulse is output from a line driver, wire a maximum length of 20 meters.

Line Driver Output



- When the feedback pulse output from an open collector or voltage source, wire a maximum of 3 meters.

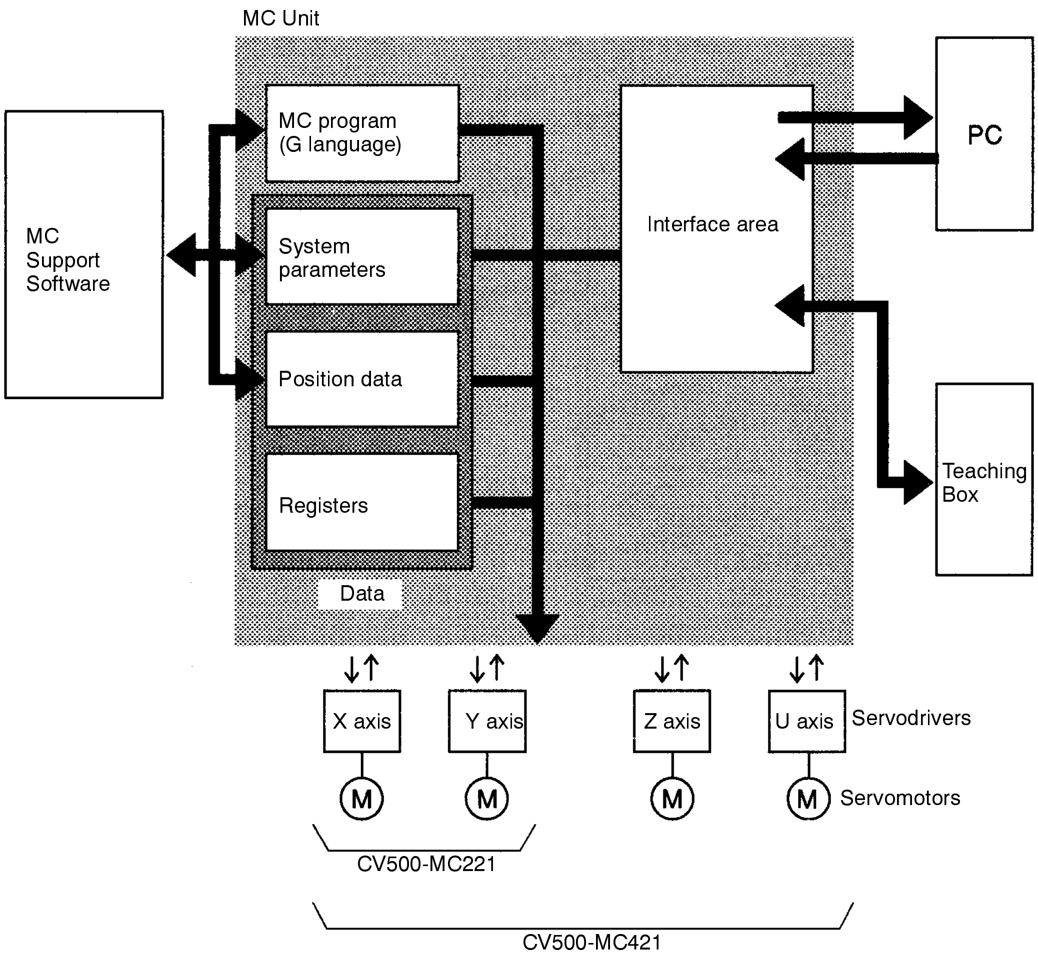
Open Collector or Voltage Output



Note The input terminals that operate the 24-V system are isolated with photocouplers to reduce external noise effects on the control system. Avoid connections between the analog control voltage ground (AG) and the 24-V system ground (DC GND).

1-8 System Configuration

The following diagrams shows an overview of the system configuration of the MC Unit and related devices.



The MC Unit receives commands from the PC through the interface area and executes the MC program to control the servomotors completely independent from the ladder-diagram program of the PC.

The MC programs, system parameters, and position data that are required for operating the MC Unit are set using the MC Support Software.

Note For further information on the MC program (G language), refer to *Section 6 G Language*.

For further information on the MC Support Software and the Teaching Box, refer to *Section 4 System Configuration*.

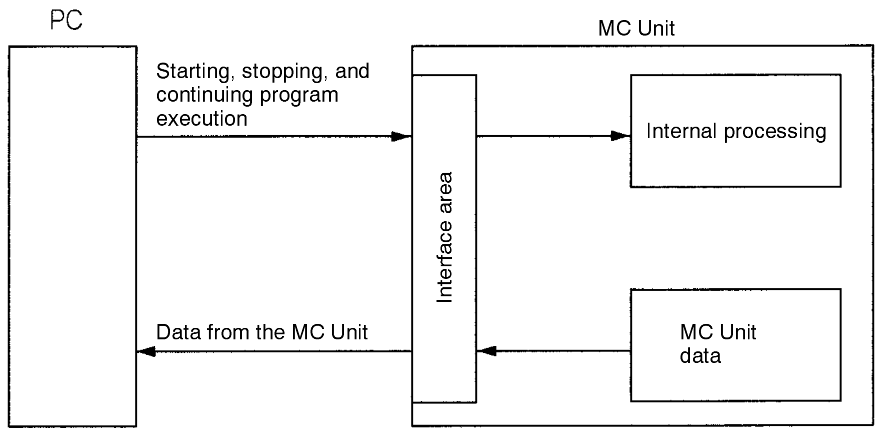
For further information on the interface area, refer to *1-9 Outline of the Interface Area* and *7-5 Interface Area*.

For further information on data, refer to *1-10 Data Configuration*.

1-9 Outline of the Interface Area

The interface area allows data such as commands from the PC, tasks of the MC Unit, and axis status, to be transferred between the PC and the MC Unit.

Interface Area



To control the interface area from the PC, it is necessary to allocate I/O word and a DM Area words to the MC Unit and to create a ladder program.

Refer to *7-5 Interface Area* for more information on the I/O word and DM Area word allocation and on the function of the allocated words.

Note For further information on the interface area, refer to the *MC Unit Operation Manual: Details*.

1-10 Data Configuration

The MC Unit handles the following three types of data:

System parameters
Position data
Registers

Programs created in the G language are not handled as data.

Types of Data

Name of data	Description
System parameters	System data processed by the MC Unit such as the number of axes used, number of tasks, feed rates (speeds) and operating ranges is stored as system parameters. The system parameters are classified into several groups.
Position data	Position data specifies positions for up to 2000 points. Position data are addressed using addresses A0000 to A19999 in G-language programs.
Registers	Registers are used to point to position data. There are 32 registers addressed using addresses E00 to E31.

System parameters and position data can be easily set using the MC Support Software. When the parameters and position data set by the MC Support Software are transferred to the MC Unit, they are stored in the system parameter area and position data area. Some of the system parameters and position data can be transferred to the MC Unit through the interface area.

Note Refer to the *MC Support Software Operation Manual* for information on setting system parameters and position data.

Refer to the *MC Unit Operation Manual: Details* for more information on data and data transfer.

1-11 Programs and Tasks

The CV500-MC221 MC Unit can perform up to two tasks; the CV500-MC421 MC Unit can perform up to four tasks. (A task is a unit of execution for a program.) The following descriptions is based on the CV500-MC421.

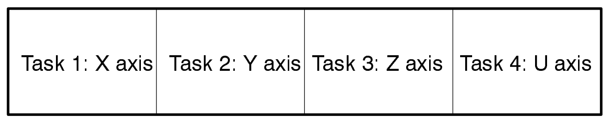
By executing four tasks at the same time, the MC Unit can perform the same functions as done by four NC controllers.

The number of tasks and axes to be used are set in advance using the MC Support Software by editing unit parameters.

Number of Tasks and Axes

The X axis, Y axis, Z axis, and U axis are available. Each axis can be used in only one task, i.e., any axis assigned to one task cannot be used for another task.

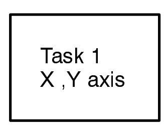
Example 1: 4 Tasks and 4 Axes



Example 2: 2 Tasks and 3 Axes

Example 3: 2 Tasks and 4 Axes

Example 4: 1 Task and 2 Axes



Tasks and Blocks

The MC Unit is capable of storing a total of 800 blocks of programming. The maximum number of blocks that can be executed in each task depends on the number of tasks as shown in the following table. These figures include subprograms.

Number of tasks	Maximum number of blocks
1	800 blocks
2	400 blocks/task
3 (CV500-MC421 only)	266 blocks/task
4 (CV500-MC421 only)	200 blocks/task

Tasks and Programs

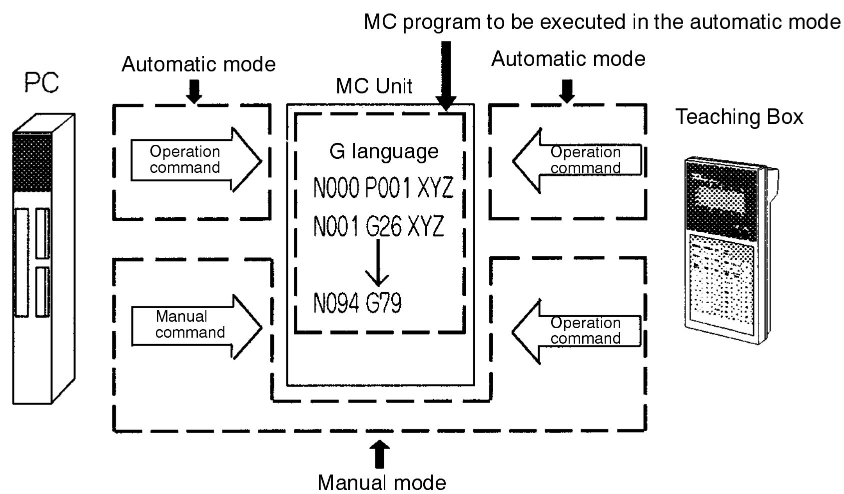
The maximum of 100 programs can be managed by the MC Unit. The number of programs that can be managed per task depends on the number of tasks as shown in the following table. These figures include subprograms.

Number of tasks	Maximum number of programs
1	100 programs
2	50 programs/task
3 (CV500-MC421 only)	33 programs/task
4 (CV500-MC421 only)	25 programs/task

Note
1. Each program number can be used only once; the same program number cannot be used in a different task.
2. Refer to *6-1 Introduction* for further information on the number of blocks and programs.

1-12 Manual and Automatic Operation

Each task of the MC Unit can be executed either in manual or automatic mode. In the automatic mode, MC programs created in the G language are executed. In the manual mode, manual commands from the PC or the Teaching Box are executed.



The interface area is used for executing the commands in the manual mode or the MC program in the automatic mode.

There are ten manual commands, including the origin search command, home shift command, jogging command, etc.

Note Refer to *7-5 Interface Area* for further information on the interface area.

Command	Description
Origin search	For determining origins of mechanical system when using an INC encoder.
Reference origin return	For returning to the reference origin.
Jogging	For moving the axis at a fixed speed.

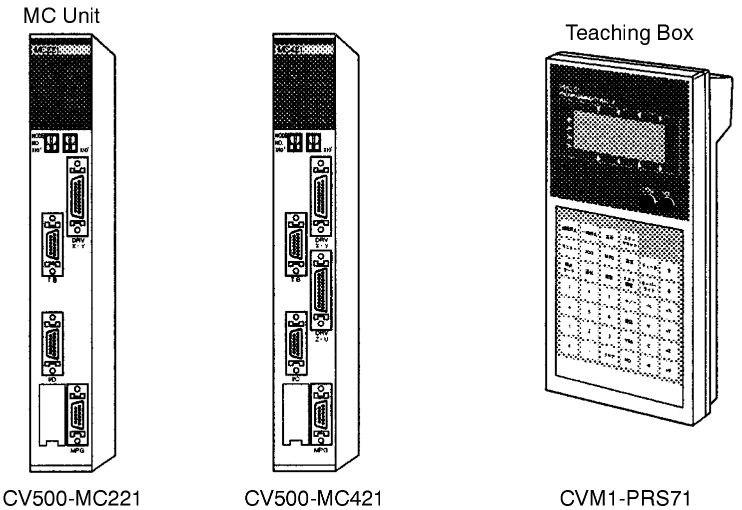
Note Refer to the *MC Unit Operation Manual: Details* for information on manual operation.

SECTION 2
Features

This section describes the features and functions of the MC Unit.

2-1 Features

The CV500-MC421 Motion Control Unit is designed to control positioning using servomotors.



The CV500-MC421 MC Unit provides the following features.

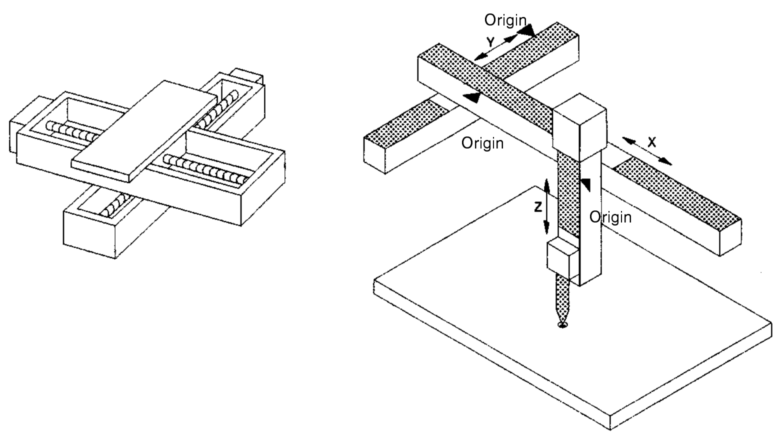
Four-axis Control

One Unit is capable of controlling up to four axes for linear, circular, or helical interpolation.

Control setups can be varied depending on applications. For example two axes each can be used in pairs, or the X, Y and Z axes can be used in a group and the U axis used separately.

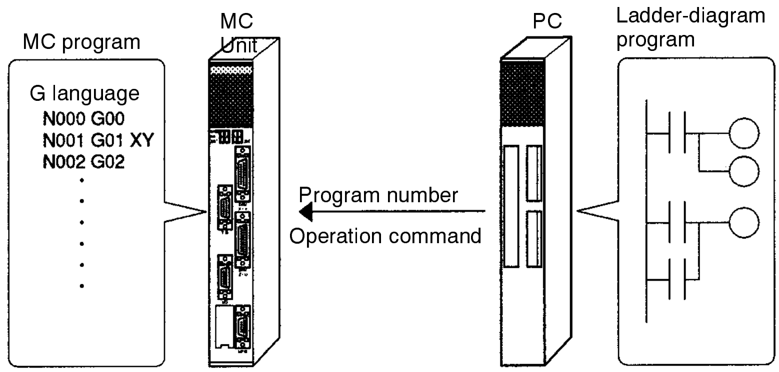
Two systems like the following one using two axes each.

Or the following system using three axes plus one more axis used separately.



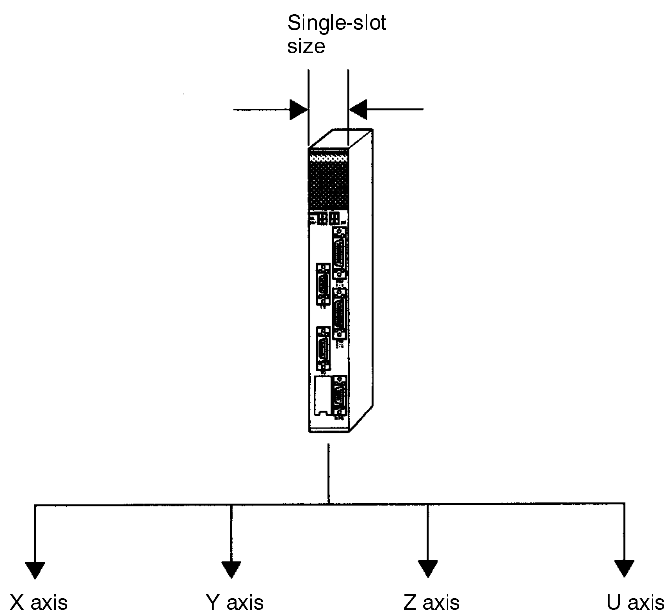
Easy Control through Multi-task G Language

The G language is used to program the MC Unit. Each program is controlled as a task completely independent from other tasks. Programs can be easily executed by simply designating a program number and giving a run command from the ladder-diagram program.



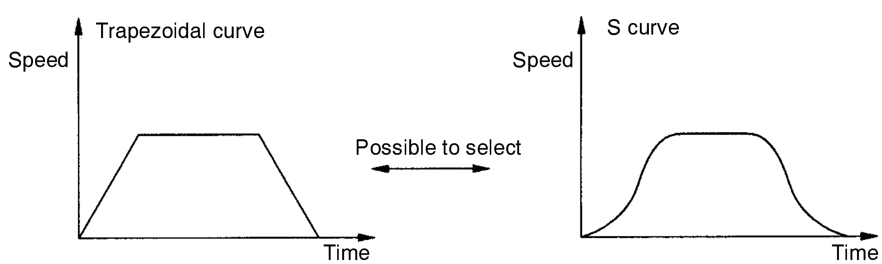
Space-saving 4-axis control

The MC Unit is as compact as CV-series I/O Unit and yet can control up to four axes.



Reduced Machine Wear

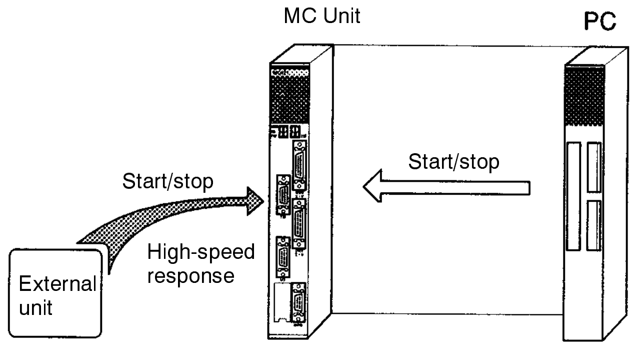
In addition to the traditional trapezoidal curve, an S curve has been adopted to reduce vibration and extend the service life of the controlled machine.



Note Refer to *5-6 Acceleration and Deceleration Curves* for further information on the trapezoidal and S curves.

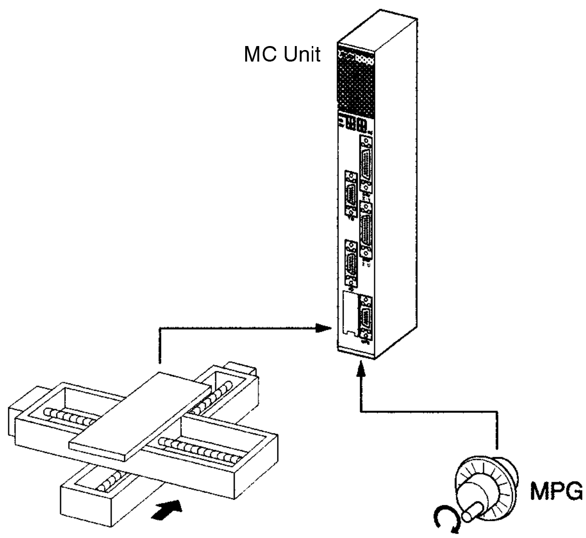
General-purpose I/O Signals

Starting and stopping can be controlled without the use of the PC. The time required until a control command voltage is output is shorter than the time necessary to start the Unit by a command from the ladder-diagram program.



Manual Pulse Generator Supported

An MPG (Manual Pulse Generator) can be connected if operations such as teaching require fine position data.



2-2 Specifications

General Specifications

The general specifications are the same for both the CV500-MC221 and the CV500-MC421.

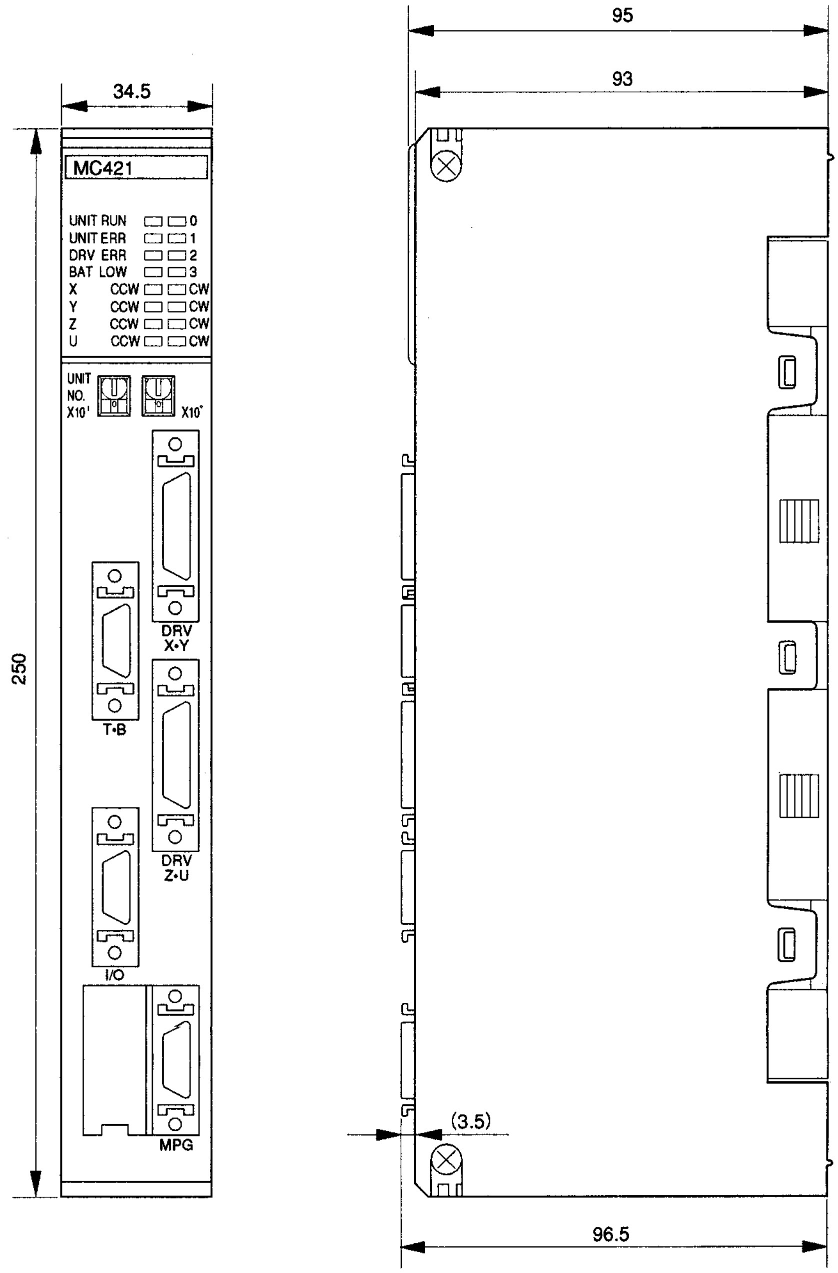
Item	Specifications
Power supply voltage	5 VDC (from Backplane.)
	24 VDC (from external power supply.)
Voltage fluctuation tolerance	4.75-5.25 VDC (from Backplane.)
	21.6-26.4 VDC (from external power supply.)
Internal current consumption	1 A or less for 5 VDC
	50 mA or less for 24 VDC
Battery service life	5 years maximum at 25°C (See note.)
Weight	600 grams or less
External dimensions	250.0 x 34.5 x 93.0 mm (H x W x D) (Occupies one slot.)

Note The battery service life is five years at 25°C. The period of time that it can be used for memory backup when the power is off is shortened at high temperatures. Replace the battery immediately when the BAT LOW indicator light turns on.

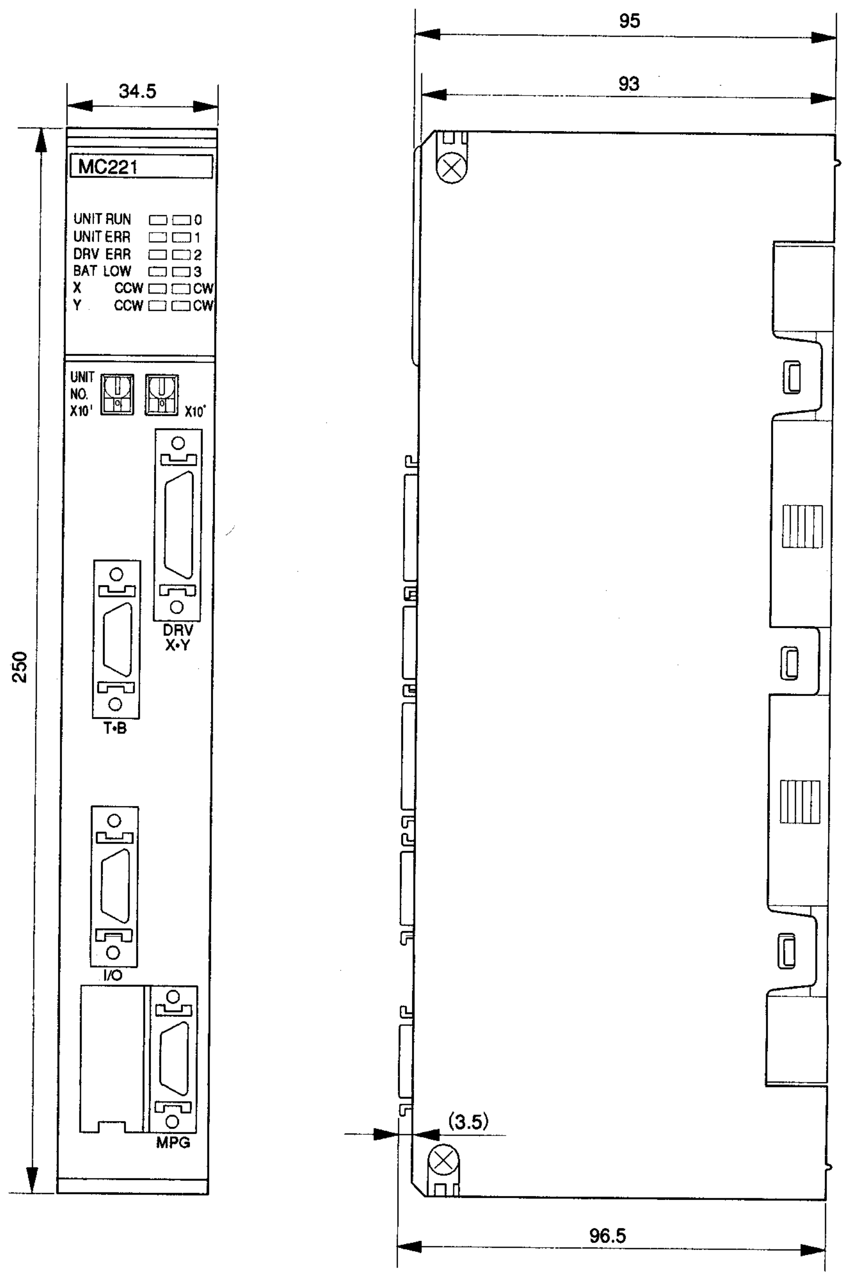
Specifications other than those shown above conform to those for the SYSMAC CV Series.

2-3 Outside Dimensions

CV500-MC421



CV500-MC221


34.5
250
MC221
UNIT RUN 0
UNIT ERR 1
DRV ERR 2
BAT LOW 3
X CCW CW
Y CCW CW
UNIT NO. X10¹ X10⁰
DRV X•Y
T•B
I/O
MPG
95
93
(3.5)
96.5

2-4 Functions

The following tables summarize the capacity and functions of the MC Unit.

Item		Contents
Number of I/O words		IOM: 25 words; DM: 100 words; total: 125 words
External connected devices		Teaching Box, MPG (manual pulse generator)
Controlled driver		Analog Input Servodriver (OMRON H Series, M Series, etc.)
Control	Control method	Semi-closed loop using incremental/absolute encoder (speed command voltage output)
	Number of controlled axes	MC221; 2 axes max. MC421: 4 axes max.
	Number of simultaneously controlled axes	MC221; 2 axes max. MC421: 4 axes max.
	PTP (independent) control	Execution by independent programs, operation modes for each axis
Interpolation control	Linear interpolation	MC221; Linear interpolation for maximum of two axes MC421: Linear interpolation for maximum of four axes
	Circular interpolation	Circular interpolation for two axes on a plane
	Helical interpolation	MC221; Not possible MC421: Helical interpolation on a plane + one axis
Control unit	Minimum unit settings	1, 0.1, 0.001, 0.0001
	Units	mm, inch, degree, pulse
Maximum command value		–39,999,999 to +39,999,999

Note 1. When displaying units in MC Support Software as anything other than pulses, first change the display unit and then set the amount of workpiece movement by setting the pulse rate.

2. The maximum position command values are as shown in the following table for each minimum setting unit.

Minimum setting unit				
1	**0.1**	**0.01**	**0.001**	**0.0001**
–39999999 to +39999999	–3999999.9 to +3999999.9	–399999.99 to +399999.99	–39999.999 to +39999.999	–3999.9999 to +3999.9999

The maximum command value ranges may become smaller than those shown above depending on the pulse rate. The maximum command value must be set to satisfy the following two conditions:

- Maximum command value (C) ≤ 1073741823 x P
- Maximum command value (C) ≤ 39999999 (C)

P: Pulse rate (Pulse/pulse, mm/pulse, degrees/pulse and inch/pulse)
C: Minimum setting unit (1, 0.1, 0.01, 0.001, 0.0001)

Example: If the minimum setting unit is 0.01 and pulse rate is 0.0001, the maximum command value can be calculated as follows:

$$1073741823 \times 0.0001 = 10734.1823 < 399999.99$$

Therefore, the maximum command value range is between –10734.18 and 10734.18 (because the minimum set range is 0.01).

Item		Contents	
External I/O	Teaching Box	RS-422, one channel (baud rate: 9,600 bps)	
	Encoder	Line receiver input MC221: For two axes MC421: For four axes (170 kpps before multiplication) Multiplication factor fixed at 4.	
	Servodriver relationships	The following signals are each provided for two axes for the MC221 ad four axes for the MC421: Input: Driver alarm signal Output: Driver alarm reset signal High-speed command voltage output (±10 V) Operation command output	
	MPG (See note.)	One channel, line driver-type MPG (50 kpps max.)	
	Individual axis control (See note.)	The following signals are each provided for two axes for the MC221 ad four axes for the MC421: Input: CCW limit input CW limit input Origin proximity input Immediate stop input	
	Others (See note.)	General inputs and outputs, four each	
Acceleration/deceleration curve		Trapezoid or S curve	
Acceleration/deceleration time		0 to 9,998 ms, which can be set separately for acceleration and deceleration (in 2-ms units).	
Feed operations	Rapid feed speed	Example: 36.86 m/min	Conditions Encoder resolution: 2,048 p/r Motor speed: 4,500 r/m Control unit: 0.001 mm/pulse
	Interpolation feed speed	Example: 36.86 m/min	
	Rapid feed override	0% to 100%	Setting unit: 0.1%
	Interpolation feed override	0% to 199%	
	Jog feed override	0% to 100%	

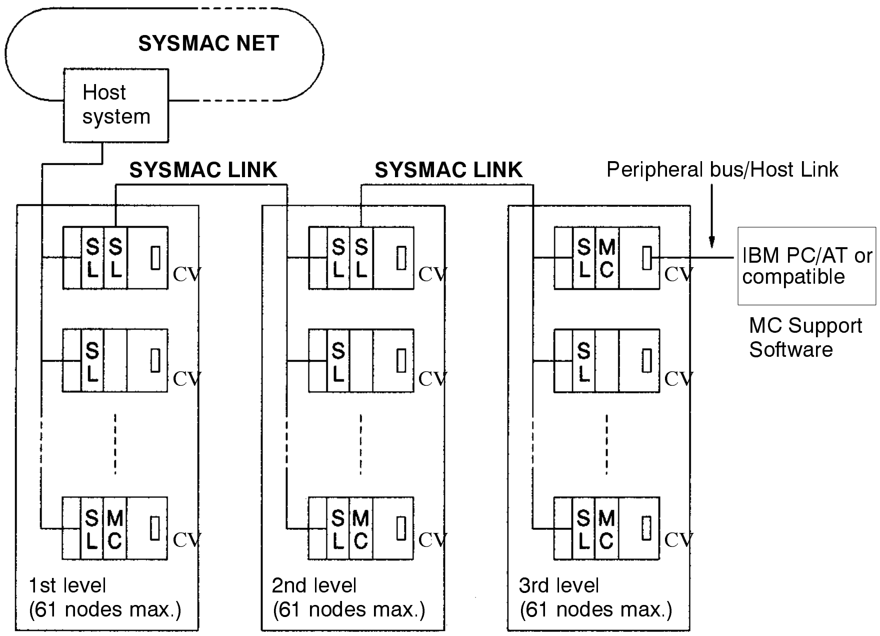
Note Power supply must be provided by the user.

Item			Contents
Operation modes	Automatic	Automatic operation	Automatic operation by program.
	Manual	Jogging	Moves axes continuously by manual operation.
		Handle feed	Moves axes by MPG.
		Axis selection	Specifies axes for handle feed operation.
		Multiplication rate selection	Specifies the rate per pulse during handle feed operation.
		Manual origin search	Searches for mechanical origin.
		Manual origin return	Moves to origin in standard coordinate system.
Origin return	Manual origin return		Manually returns to origin in standard coordinate system.
	Automatic origin return		Automatically returns to origin in standard coordinate system.
	Workpiece origin return		Automatically returns to origin in workpiece coordinate system.
Zone setting			8/axis
Optional inputs			20 points (inputs for block control referenced by special G codes) Four of the 20 points can be designated as general inputs for the MC Unit.

Item		Contents
Program and data retention	MC Unit	Battery backup by lithium battery
	External peripheral devices	Personal computer diskette or hard disk
Self-diagnostic function		Detection of memory corruption Detection of disconnected lines
Error detection function		Error counter alarm Error counter overrun Battery alarm CPU error Communication error (Teaching Box) Software limit overrun Z-phase error Error history Overtravel Immediate stop FINS communication error Unit number error CV error detection Driver alarm detection EEPROM error
Task program management	Number of tasks	4 max. (program execution units)
	Number of programs	The maximum number of programs differs according to the number of tasks. When 1 task is used: 100 When 2 tasks are used: 50/task When 3 tasks are used: 33/task (MC421 only) When 4 tasks are used: 25/task (MC421 only)
	Program capacity	The maximum program capacity differs according to the number of tasks. When 1 task is used: 800 blocks When 2 tasks are used: 400/task When 3 tasks are used: 266/task (MC421 only) When 4 tasks are used: 200/task (MC421 only)
	Position data capacity	2,000 max. (when only one axis is used)
	Number of registers	32 (Mainly used for specifying position data numbers.)
	Sub-program nesting	5 levels max.
Task control	Program number designation	Specifies the number of the program to be executed.
	Cycle start	Executes the program
	Single block	Executes the program one block at a time.
	Pause	Temporarily halts program execution.
	Control block end	Forcibly ends execution of a block.
	Error reset	Resets a task in which an error has occurred.
	Optional inputs	Specifies input information to be referenced by special G code (20 points).
	Teaching	Creates position datas for each task.
Auxiliary function	M code	0 to 999
Axis control	Current position preset	Changes the current position to any coordinates.
	Servo-lock	Creates a position loop and turns ON the operation command output to the servodriver.
	Servo-free	Cancels the position loop and turns OFF the operation command output to the servodriver.
	In-position	Pulses can be set within a range of 0 to 999.
	Driver alarm reset	Resets servo-driver alarm.
Data transfer		Transfers position data and acceleration/deceleration time parameters between the PC and MC Unit.

2-5 Network Compatible

Networks can be configured so that data can be exchanged on three levels using MC Support Software and a host system.



SL: SYSMAC Link Unit
MC: MC Unit
CV: CV-series PC

MC Support Software Operations

The following operations are available using the MC Support Software.

- Uploading and downloading MC Unit files (programs, positioning data, and system parameters)
- Monitoring the MC Unit present position and other data.
- Reading MC Unit error histories.
- Reading other status.

SECTION 3
Servo System Principles and Precautions

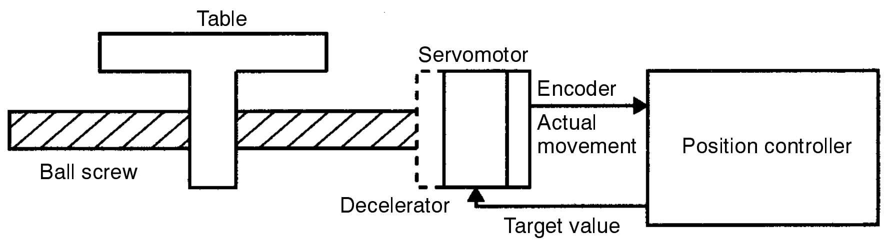
This section provides information on the servo system, basic mechanisms for positioning, and precautions to be heeded in using the system.

3-1 Servo System

The servo system used by and the internal operations of the MC Unit are briefly described below.

Semi-closed Loop System

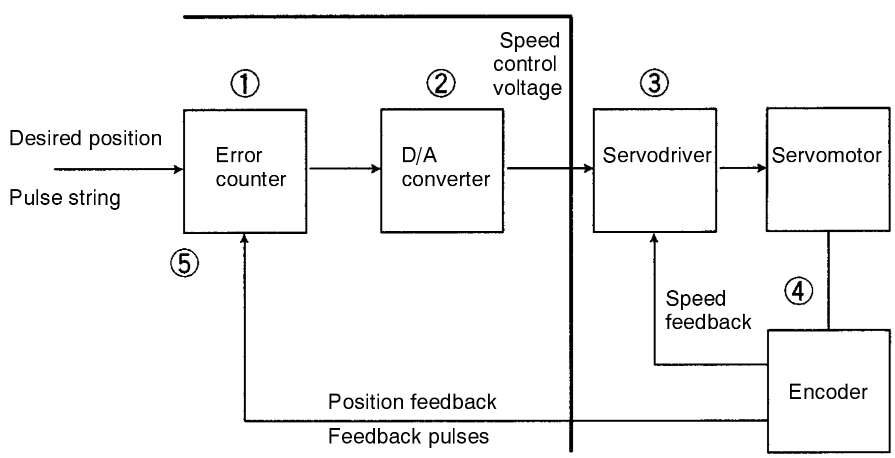
The servo system of the MC Unit uses a semi-closed loop system. This system is designed to detect actual machine movements by rotation of the motor in relation to a target value. It computes the error between the target value and actual movement, and zeroes the error through feedback.



Semi-closed loop systems occupy the mainstream in modern servo systems applied to positioning devices for industrial applications.

Internal Operations of the MC Unit

Commands to the MC Unit, speed control voltage to the servodriver, and the feedback signals from the encoder are described in the next few pages.

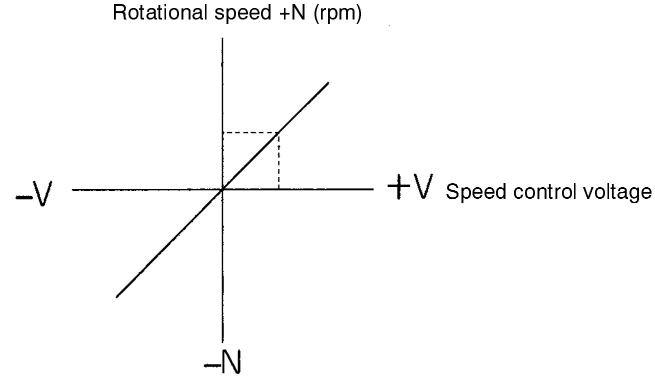


1, 2, 3...

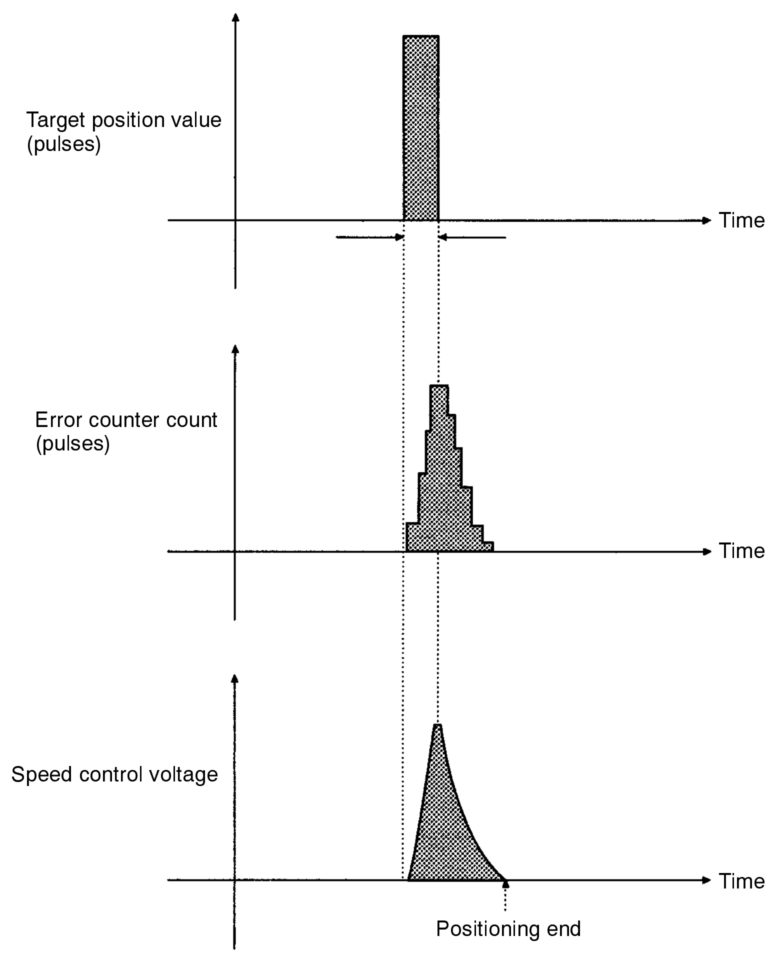
1. The error counter receives a target position in units of encoder pulses. This is called a pulse string.
2. The error counter is directly connected to the D/A converter where the pulses received by the error counter are converted to analog voltages. These analog voltages are sent to the servodriver as the speed control voltages.

3. When the speed control voltage is received by the servodriver, it rotates the motor at a speed corresponding to the speed control voltage. The rotational speed is in proportion to the speed control voltage.

Servodriver Speed Characteristics



4. The rotary encoder directly connected to the motor axis rotates in sync with the motor and generates feedback pulses.
5. The feedback pulses are subtracted until the error counter goes to zero. When the error counter goes to zero, the speed control voltage to the servodriver becomes zero and the motor stops rotating.

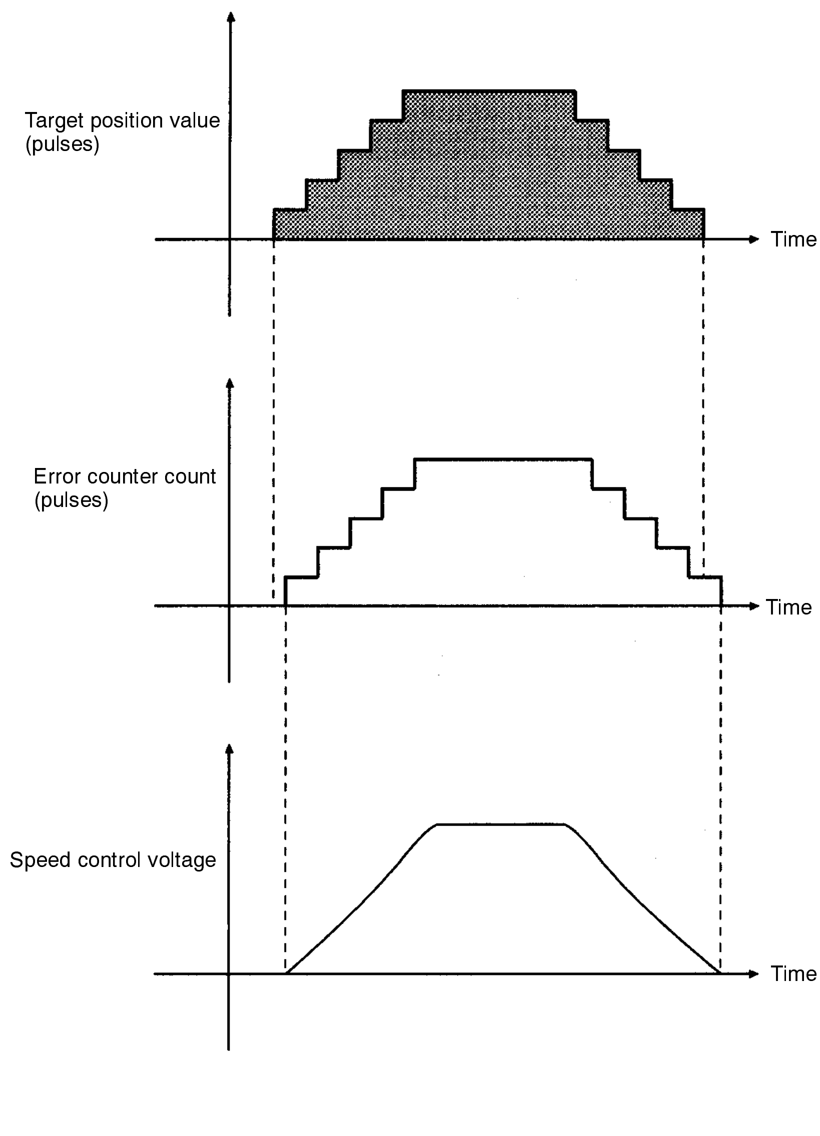


6. Unless the target position is given, the error counter constantly maintains the stopped position.

7. If the motor axis moves slightly due to a drift in the driver or voltage output, the error counter receives a feedback pulse from the rotary encoder and a speed control voltage is output in the reverse direction, causing the motor to rotate toward its original position. This corrective operation for maintaining the present position is called servolock or servoclamp.

8. Using this principle, positioning with acceleration and deceleration is executed by continuously setting target positions in the error counter.

9. The target position set in the error counter becomes the error counter count as shown below. The count is converted to a speed control voltage for the servodriver to control the motor.

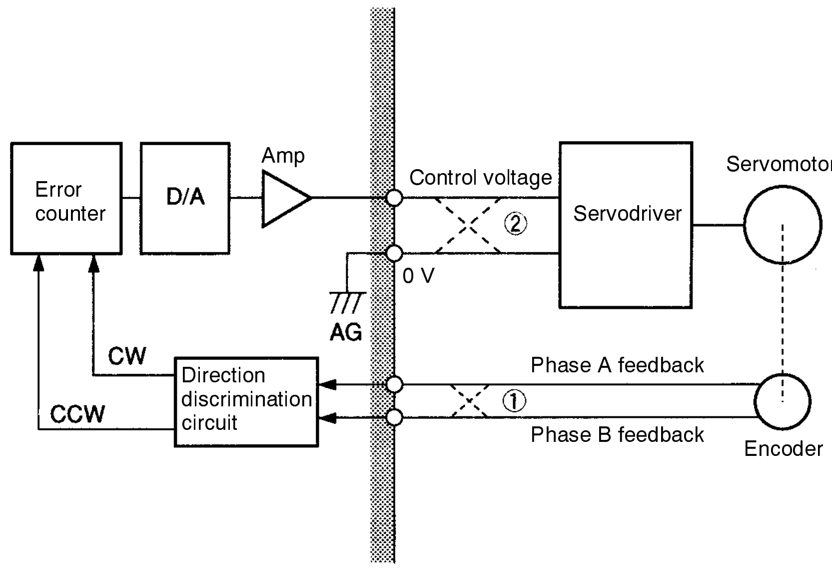


Thus, the position equals the total count of target positions (shaded area in the figure), and the speed will depend on the target position value per unit time.

3-2 Motor Runaway

In a servo system employing a servomotor, faulty or disconnected wiring may cause the servomotor to run out of control. Therefore, careful attention must be paid to preventing faulty or disconnected wiring.

When the wiring is correct, the servomotor maintains the stopped position through corrective operations as long as a position loop is formed and servolock is in effect.



If the motor rotates in the CW direction due to a factor such as temperature drift, it is detected by the encoder and the internal error counter of the Motion Control Unit is notified of the direction and amount of rotation by means of feedback signals given by the encoder.

The count of the error counter is ordinarily zero unless otherwise designated. When the motor moves in the CW direction, the feedback signal transfers the direction and amount of movement as a count to the error counter. Then the Motion Control Unit outputs a control voltage to rotate the motor in the CCW direction so as to zero this count figure.

The control voltage is output to the servodriver, and the motor rotates in the CCW direction. Again, when the motor rotates in this CCW direction, the encoder detects the direction and amount of movement and notifies the error counter in the Motion Control Unit with feedback signals to subtract and zero the count figure.

This position loop subtracts the count in the error counter to zero it.

Runaway Due to Faulty Wiring

If the phase-A and phase-B feedback input lines are wired in reverse (crossed dotted lines at 1 in the figure), the servolock will not be effective and the motor will run out of control.

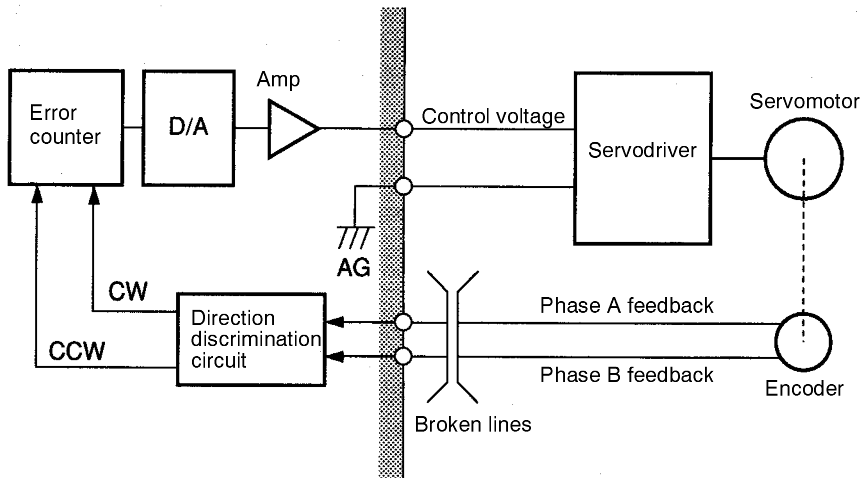
1, 2, 3...

1. If the motor rotates in the CW direction due to drift or some other cause, the encoder detects the direction and amount of movement and transmits feedback signals to the error counter in the Motion Control Unit.
2. If the phase A and phase B feedback input lines are wired in reverse, the error counter receives the information as a rotation in the CCW direction.
3. As a result, the error counter having a count in the CCW direction attempts to zero the count by outputting a control voltage to the motor driver in the CW direction.
4. The servomotor rotates in the CW direction, repeating the above steps 1. to 3. Eventually, the motor runs out of control.

Runaway can occur not only from reversed wiring of phases A and B of the feedback inputs, but also from reversed wiring of the speed control voltage and the ground lines (crossed dotted lines at 2 in the figure above).

Runaway Due to Disconnected Wiring

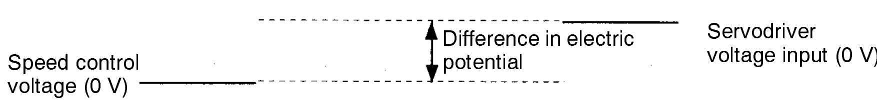
The servomotor runs out of control not only when the position loop is not correctly formed but also when the position loop is not interrupted due to disconnected wiring.



1, 2, 3...

1. Wire breakage while the servomotor is rotating:
   While the servomotor is rotating, the speed control voltage is not 0 V because of the signal from the error counter. If the feedback line is broken, no feedback signals will be given to the error counter and the speed control voltage remains unchanged from the value existed before the line breakage, causing motor runaway.
2. Wire breakage while the motor is stopped:
   If the feedback line is broken while the servomotor is stopped and correct feedback signals cannot be returned, the speed control voltage remains at zero without changing. Therefore, the servomotor also remains stopped. In fact, however, the motor may move in one direction without stopping.

This is caused by a discrepancy between the 0 V of the MC Unit's control voltage and the 0 V of the servodriver's voltage input.



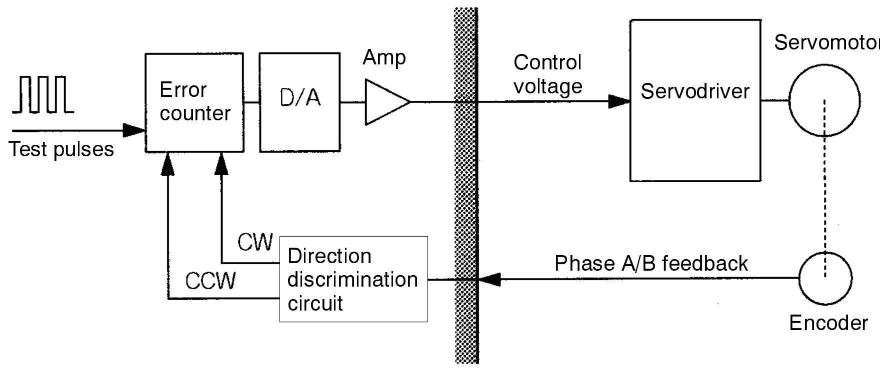
When the two 0 voltages do not match, an electric potential difference is generated, resulting in a false control voltage. This in turn causes the servomotor to move in one direction without stopping.

In order to prevent this, repair the wiring or adjust the 0 V of either the MC Unit or the servodriver so that the 0 V levels match.

3-3 Wiring Check Function

The MC Unit is provided with a wiring check function. This function is designed to check for reverse wiring and disconnected wiring when the power supply switch is turned on in order to prevent motor runaway. Whether or not to a execute wiring check can be determined by setting a system parameter.

This function is designed to output a specified number of pulses in the specified direction and to check whether correct feedback pulses are read.



Set a specified number of test pulses in the error counter. After a set period of time (the time required for the system parameter to perform wiring check), check the content of the error counter to determine whether the number of pulses that were initially set and also the direction are correctly returned.

When the test pulses are set in the CW direction, the feedback pulses corresponding to the set number of pulses will be returned in the CW direction. If the direction is reversed, it is treated as a reverse wiring error and the servolock is cleared, dropping the voltage output to 0 V.

To correct this faulty wiring, either turn off the power and repair the reversed wiring or change the machine parameter "encoder polarity" from the MC Support Software from the preset setting to the opposite setting, i.e., change from "forward rotation for encoder increase" to "reverse rotation for encoder increase" or from "reverse rotation for encoder increase" to "forward rotation for encoder increase."

Note Refer to the *MC Support Software Operation Manual* for further information on whether to execute wiring check or not, for further information on the test pulse setting, and for further information on the time setting before error counter check.

In addition to the descriptions on the previous page, if the feedback pulses returned is less than the number of test pulses, a disconnected wiring error will be generated. Just as with a faulty wiring error, the servolock will be cleared and the voltage output will drop to 0 V.

To correct a disconnected wiring, turn off the power and repair the wiring.

Faulty wiring checks and disconnected wiring checks can be carried out simultaneously. Both checks can be skipped by setting the machine parameter "wiring check" from the MC Support Software to NO. It is initially set to YES.

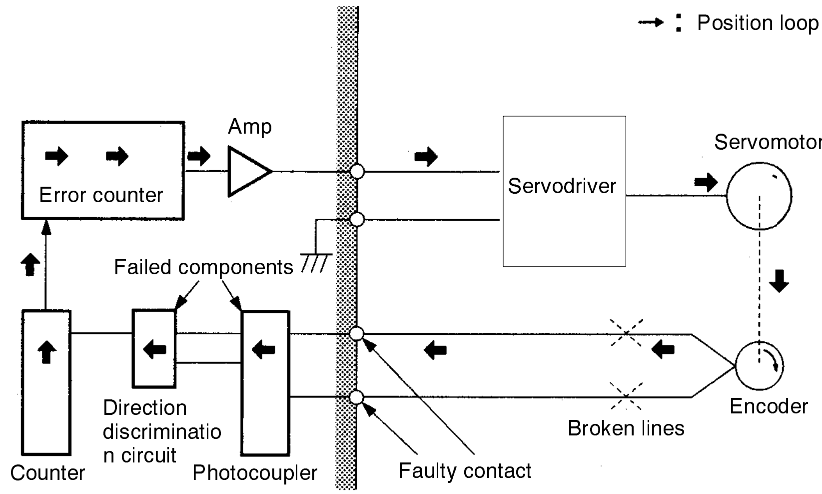
3-4 Failsafe Circuits

To protect against unforeseen problems that may occur during operation, provide failsafe circuits, as those shown below, in the positioning system in which the MC Unit is used.

Errors during Positioning

As illustrated below, motor runaway may occur during operation without a position loop being formed for these reasons:

- Failure of internal components in the MC Unit
- Disconnected external wiring or faulty connections



Error Counter Overflow Check

The MC Unit checks for errors during positioning according to the count in the error counter and executes the following processes if motor runaway occurs as described above:

a) Outputs an “Error counter overflow” error

b) Sets the Error Counter Alarm Bit in the interface bits to ON according to the parameter settings.

Error Counter Capacity Setting

The error counter capacity can be set according to the operation conditions by means of a parameter. This parameter can be used to make the error counter overflow detection more sensitive. Follow the procedure outlined below to set the error counter capacity.

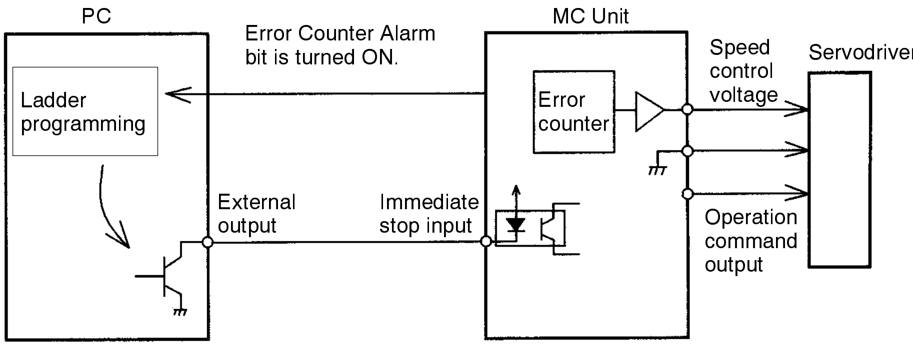
1, 2, 3...

1. Conduct a trial operation of the machine, and use the Teaching Box to check how the count changes in the error counter.
2. Check the maximum count value, and set the parameter data so that the error counter capacity is 10% to 20% greater than that value.

Example 1, below, shows an error occurring during positioning. The MC Unit checks the count of the error counter and the Error Counter Alarm Bit of the interface bits is turned ON. The PC processes input data and gives an immediate stop input to the MC Unit by means of an external output.

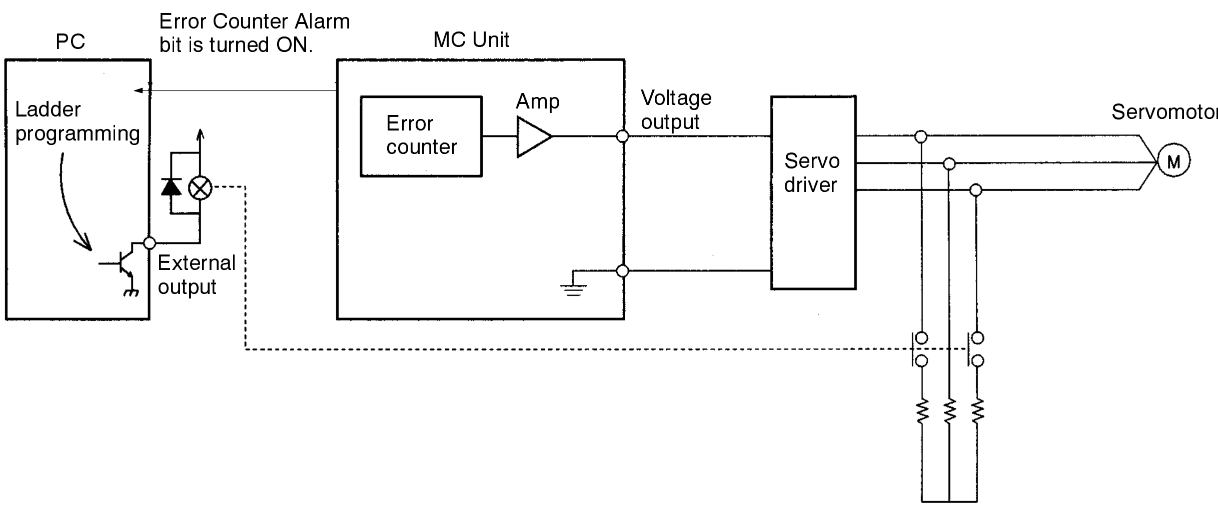
When an error counter overflow occurs, the immediate stop input is turned ON using an external output, the voltage output to the servodriver is dropped to 0 V and at the same time the operation command output is turned OFF.

Example 1



In the Example 2, when the Error Counter Alarm Bit is turned ON, the dynamic brake of the motor is actuated by an external output from the PC to stop the motor.

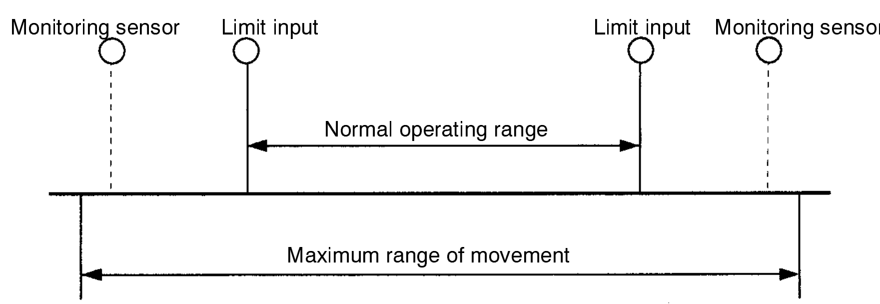
Example 2



In either of the above examples, the motor makes a sudden stop when the error counter overflows. Make sure that this sudden stop will not result in damage to the system.

External Emergency Stop Circuit

In addition to the failsafe circuits shown above, a failsafe circuit is normally set up using monitoring sensors installed at the edges of the workpiece's range of movement to detect abnormal workpiece movement and stop operation if a runaway occurs.



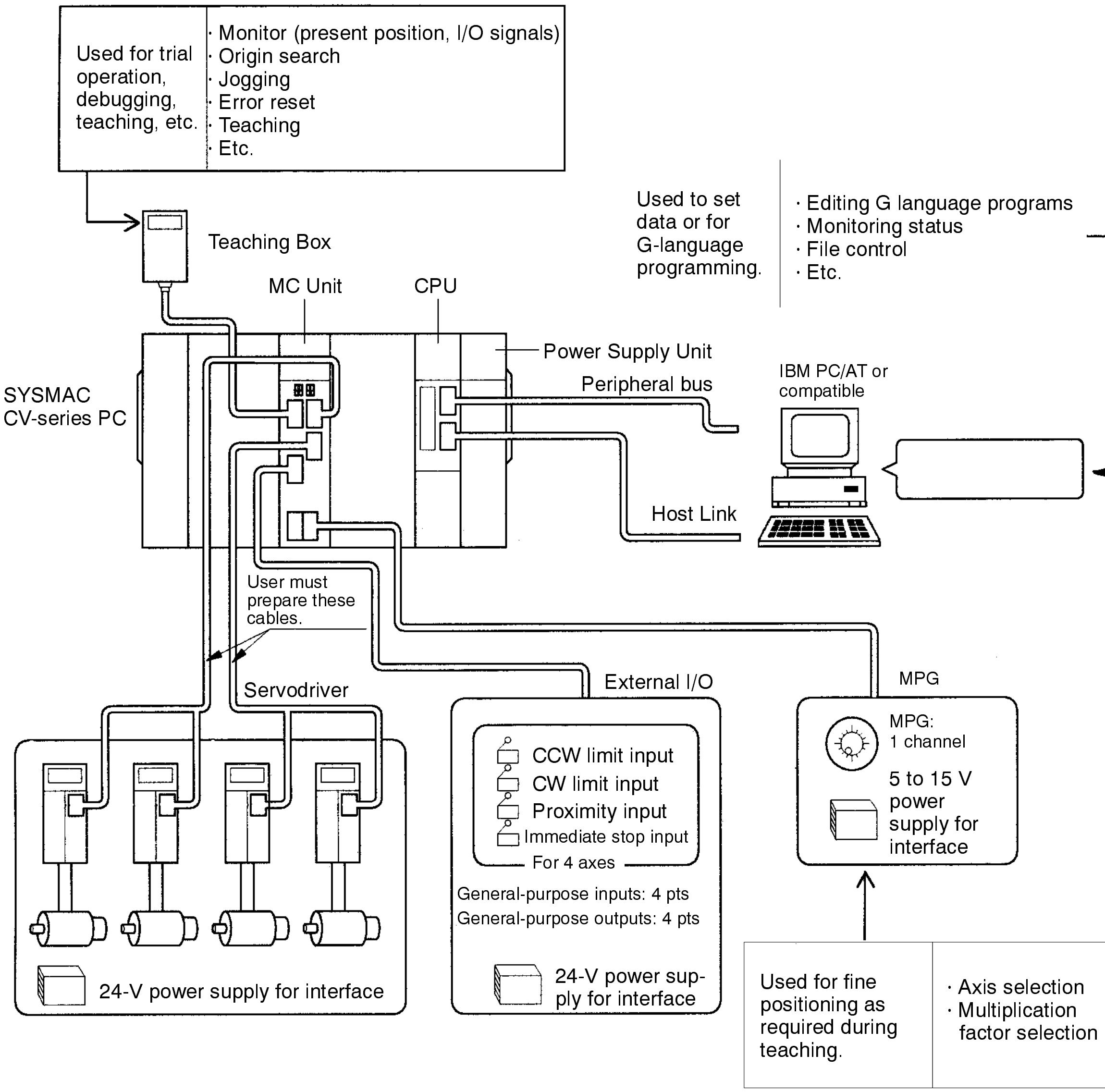
Monitoring sensors are installed outside of the limit inputs. If the workpiece reaches one of the sensors, the power is turned off to the servodriver and then the dynamic brake is applied to stop the motor.

SECTION 4
System Configuration

This section provides information on the system configuration necessary for operating the MC Unit.

4-1 Basic System Configuration

The following is an example of a system configuration using a CV500-MC421 MC Unit.



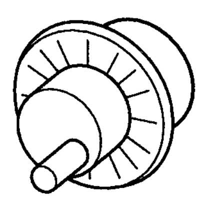
Devices used in the system configuration example are as shown in the following table.

Devices	Model
Motion Control Unit	CV500-MC421
CPU	One of the following: CV500-CPU01-V1 CV1000-CPU01-V1 CV2000-CPU01-V1 CVM1-CPU01-V2 CVM1-CPU11-V2 CVM1-CPU21-V2
Power Supply Unit	One of the following: CV500-PS221/PS211 CVM1-PA208
CPU Backplane	CV500-BC101/BC051/BC031
Teaching Box	CVM1-PRS71
Personal computer for CV Support Software	IBM PC/AT or compatible
CV Support Software	CV500-ZS3AT1-EV2 (3.5-inch floppy disk) CV500-ZS5AT1-EV2 (5-inch floppy disk)
Personal computer for MC Support Software	IBM PC/AT or compatible
MC Support Software	CV500-ZN3AT1-E (3.5-inch floppy disk)
Servodriver	Analog voltage control type
MPG	Line driver type

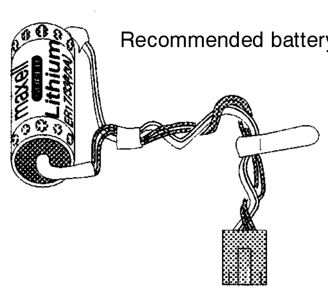
4-2 Items Supplied by the User

If using a manual pulse generator (MPG) in your system, prepare the following items.

MPG (Line driver output type: Compatible with RS-422)
Power source for the MPG: 5 V

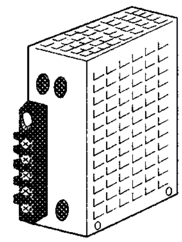


If using an absolute encoder, prepare a battery for data backup.



Recommended battery; 3G2A9-BAT08

In addition to the above, prepare power sources for the servodriver interface and for external I/O.



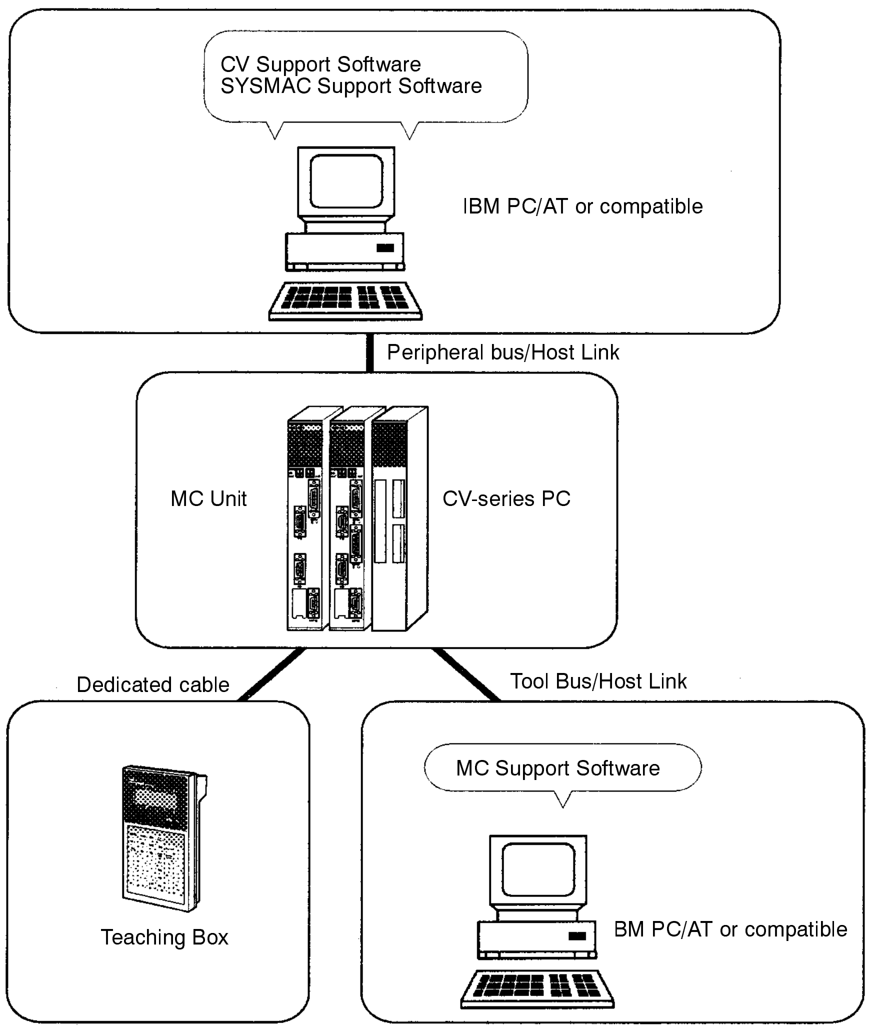
Power source for the servodriver interface: 24 V
Power source for the external I/O: 24 V

Driver Connecting Cables

The user must prepare the driver connecting cables. Refer to *8-5 Connecting Servodrivers* and the *MC Unit Operation Manual: Details* for information on driver connecting cables.

4-3 Peripheral Devices

Shown below is a brief explanation of peripheral devices used to operate the MC Unit.



Note The SYSMAC Support Software supports only CVM1 PCs. When using other CV-series PCs, use the CV Support Software.

CV Support Software Used for creating an I/O table and a ladder-diagram program, and for monitoring various data.

MC Support Software Used for creating MC Unit control programs in G language or for setting parameters.

Teaching Box Used for jogging operations, origin searches, executing control programs previously transferred to the MC Unit, displaying present positions, and teaching.

Teaching Box Functions The following table lists the functions of the Teaching Box.

Function		Description
Deceleration stop		Decelerates all axes to a stop.
Error reset	MC Unit error reset	Resets errors that have occurred in the MC Unit.
	Servodriver error reset	Resets alarms for the servodriver.
Monitor	Present position	Monitors the following present position: Present position in the reference coordinate system (using user-set unit such as “mm”). Present position in the reference coordinate system (in pulses). Error counter value.
	Position data	Reads position data stored in the MC Unit.
	Errors	Reads errors that have occurred in the MC Unit.
	I/O signals	Monitors and changes I/O signals connected to the MC Unit.
	Z-phase tolerance	Monitors the number of pulses to the Z phase from the origin input.
Origin search		Searches for the origin.
Program execution	Task/program No. designation	Designates the desired task and program to be executed.
	Cycle run	Executes tasks.
	Single block run	Executes the program block by block.
Jogging		Jogs individual axes. More than one axis can be jogged at the same time.
MPG feeding	Axis	Designates the axis to be fed by MPG (manual pulse generator).
	Multiplication factor	Designates the multiplication factor for 1 pulse for the MPG.
Override		Increases or decreased the operating speed during memory operation.
Teaching		Registers the present position as position data.
Extension	Mode	Changes the mode used to control the MC Unit.
	Servo-lock/servo-free	Locks or frees the servomotor.
	Memory protect	Protects or clears protection for memory (position data area) in the MC Unit.
Error detection		CPU errors Communications errors

MC Support Software Functions

Function	Specifications
Applicable computer	IBM PC/AT or compatible (CPU 80286/80386/80486)
Operating environment	Memory: Open area of 490K bytes or larger Hard disk: Open area of 1 megabyte or larger Operating system: PC-DOS/MS-DOS
Editing programs	MC program can be created, modified, or deleted.
Editing position data	Position data can be created, modified, or deleted.
Editing parameters	System parameters can be created or modified.
Transferring and comparison	The contents of MC programs, system parameters, and position data are transferred and compared between the MC Unit and the personal computer.
Printing	The contents of MC programs, system parameters, and position data can be printed.
Monitoring	Monitoring while the MC program is running. Monitoring the present position Present positions in the reference coordinate system (either in user-set display unit such as mm or in pulses) Present position in the workpiece coordinate system Work origin shift Error counter count FAL status of the MC Unit I/O status of the MC Unit Error log
File control	Displaying a list of files, loading, saving or deleting a file Formatting floppy disks
System settings	Setting the network address for communications with another node. Setting the communications system (peripheral bus/Host Link)

SECTION 5
Positioning

This section provides information on position control as performed by the MC Unit.

5-1 Positioning Controls

The MC Unit offers the following two positioning controls:

PTP control

CP control (linear interpolation, circular interpolation and helical circular interpolation)

Control programs are created in the G language.

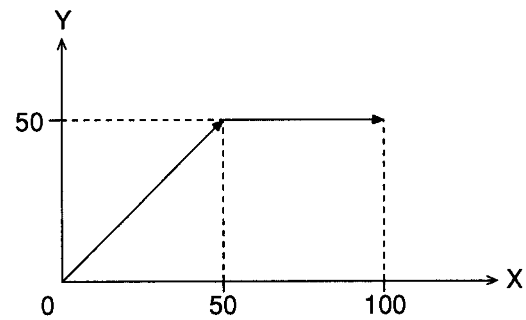
Positioning is described based on the CV500-MC421. If you are using the CV500-CV221, remember that only up to two axes can be controlled.

Note Refer to *Section 6 G Language* for information on the G language.

PTP Control

The PTP control is used to position each axis (X, Y, Z or U axis) independently from other axes. Positioning time depends on the travel distance and speed of each axis.

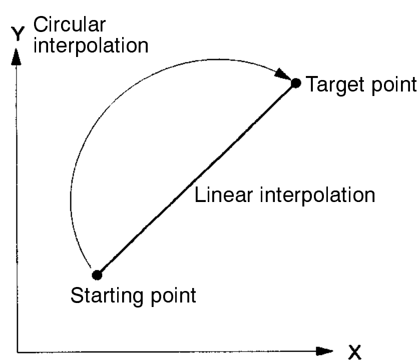
Example: Moving from the origin to the X-axis coordinate of 100 and Y-axis coordinate of 50.



The illustration shows movements when the high-speed feed rate for the X axis is set to the same feed rate as the Y axis.

CP Control

CP control is used to position by designating not only the starting point and the target point but also the path between these two points.

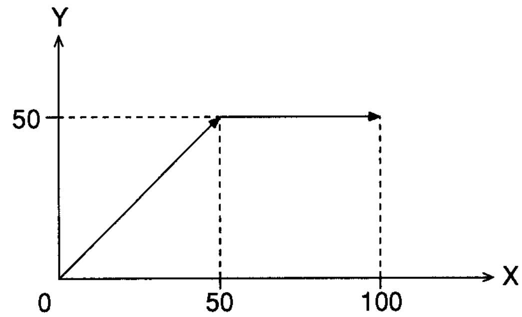


CP control offers three control methods: linear interpolation to connect two points by a straight line, circular interpolation that designates a circular path between the two points using the radius or the circle center coordinates, and helical interpolation to combine linear and circular interpolation.

5-2 PTP Control

Positioning each axis independently from the rest of the axes is called PTP control. Each axis moves at the preset seeds: at the acceleration speed for the preset acceleration time, the maximum high-speed feed rate, and the deceleration speed for the preset deceleration time.

For example, suppose a control program is executed to move from the origin to the X-axis coordinate of 100 and Y-axis coordinate of 50. If the speed for the acceleration time is the same as that for the deceleration time, X-axis and Y-axis movements will be as illustrated below.



The illustration shows movements when the high-speed feed rate for the X axis is set to the same feed rate as the Y axis.

Both the X axis and Y axis move to a coordinate of 50 over the same duration of time. At this point, the Y axis stops and the X axis moves to a coordinate of 100.

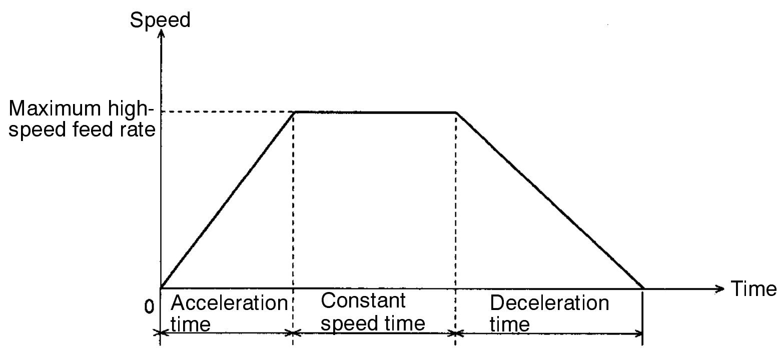
Jogging in the manual mode is also controlled by the PTP control. When jogging is controlled by the PTP control, the speed will be the maximum jogging speed.

Acceleration Time and Deceleration Time

Acceleration and deceleration times under the PTP control are as follows:

Acceleration time: Time required until the single axis speed reaches the maximum high-speed feed rate

Deceleration time: Time required until the speed control voltage drops to zero from the maximum high-speed feed rate of the single axis.

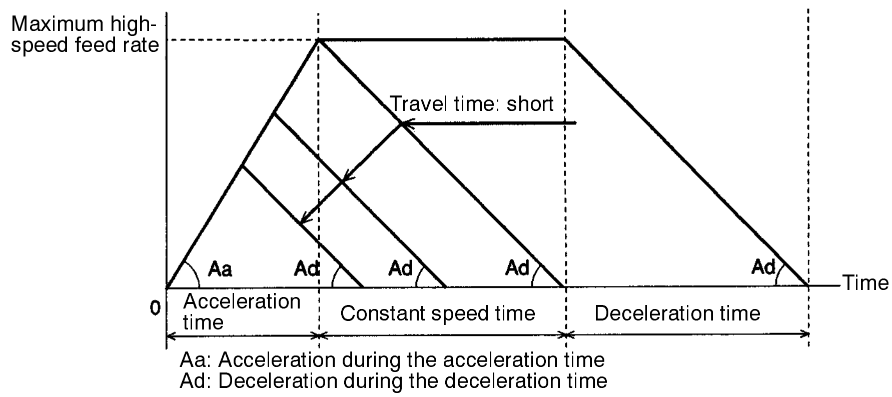


The acceleration time and deceleration time varies depending on the override values. (Acceleration speed is constant.)

Note Refer to *5-8 Override Function* for information on the override.

Triangular Control

If the travel time is shorter than the sum of acceleration time and deceleration time, movements are controlled by the triangular control shown below.

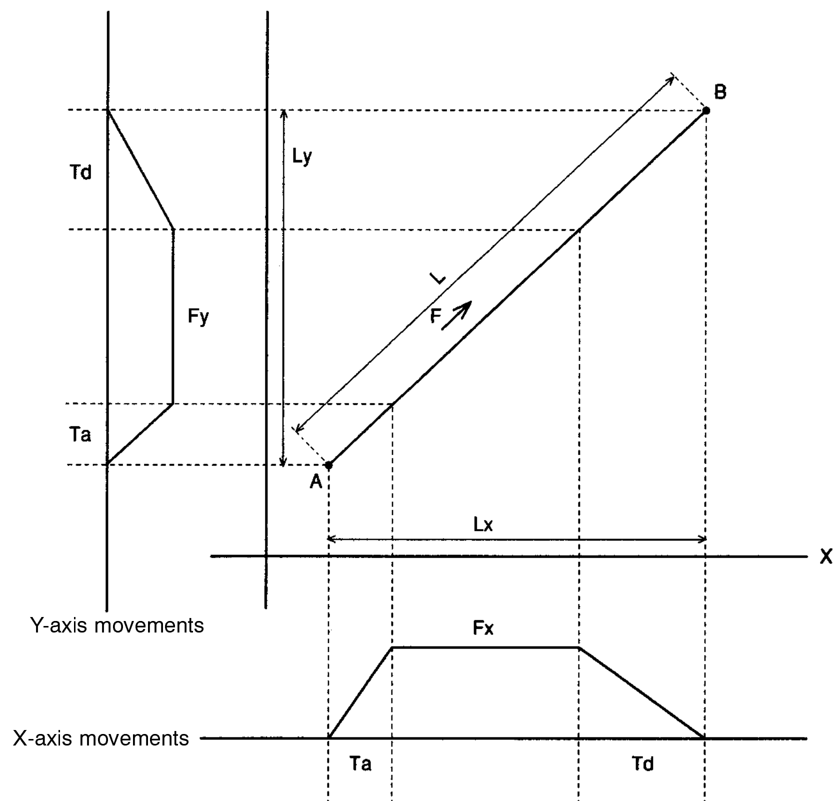


The ratio of acceleration time and deceleration time in the triangular control is the same as the ratio of acceleration time and deceleration time set as a system parameter.

5-3 Linear Interpolation

Positioning for linear interpolation produces a straight line that connects a preset starting point to a preset end point using all axes.

Linear interpolation from the point A to the point B will be as shown below when using the X and Y axes.



F: Designated interpolation feed rate
Fx: Interpolation feed rate of the X axis based on F
Fy: Interpolation feed rate of the Y axis based on F
Ta: Interpolation acceleration time
Td: Interpolation deceleration time

Fx and Fy can be expressed as follows:
$Fx = Lx/L \times F$
$Fy = Ly/L \times F$
Where, L is the travel distance in the specified locus, Lx is the travel distance along the X axis, and Ly is the travel distance along the Y axis.

Interpolation Acceleration and Deceleration Times

Interpolation acceleration and deceleration times for linear interpolation are defined as follows:

Interpolation acceleration time: Time required to reach the specified interpolation feed rate on the composite axial locus.

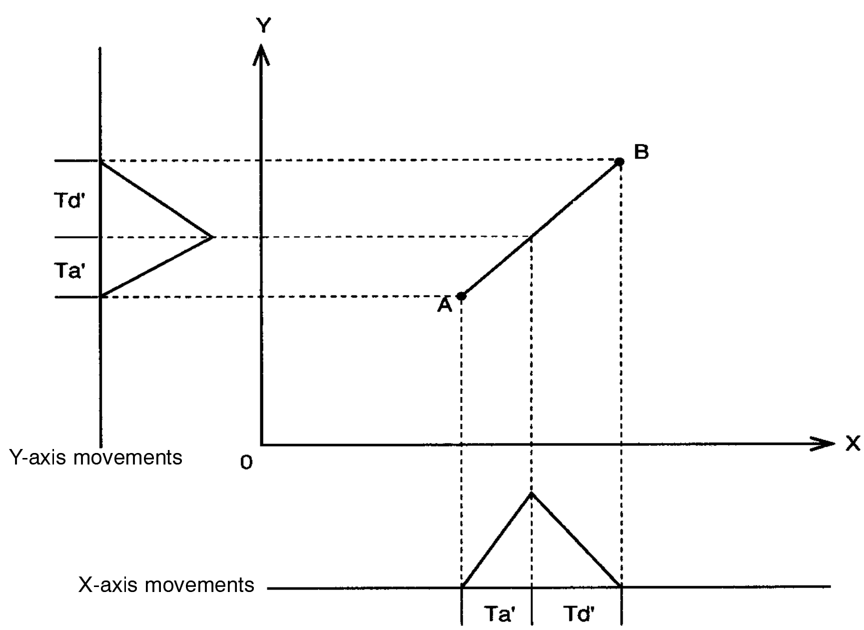
Interpolation deceleration time: Time required until the speed control voltage drops to zero from the specified interpolation feed rate on the composite axial locus.

Unlike those in the PTP control, linear interpolation acceleration and deceleration times are not affected by the speed. Acceleration changes according to the movement to satisfy the preset interpolation acceleration and deceleration times.

Triangular Control

When the mode is set to the stop mode and if the travel time is shorter than the sum of the interpolation acceleration time and the deceleration time, movements are subject to the triangular control as in the PTP control.

When the mode is set to the pass mode, the travel time won't become shorter than the preset interpolation acceleration and deceleration times, but the speed will become slower than the designated speed.



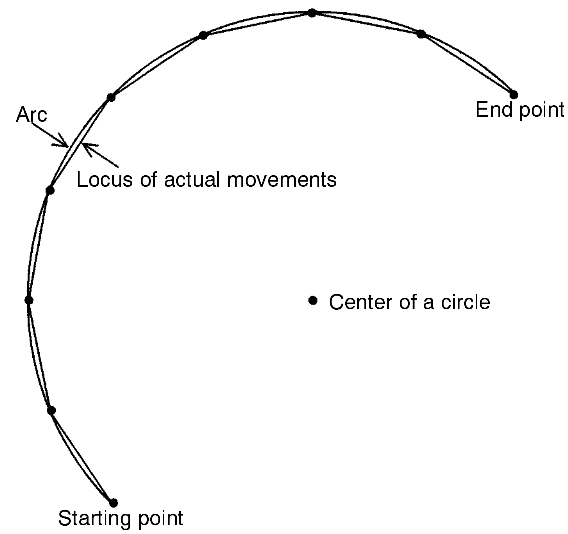
The ratio between the interpolation acceleration time and the deceleration time in the triangular control is equal to the ratio between the preset interpolation acceleration time (Ta) and the deceleration time (Td). Td'/Ta/=Td/Ta

5-4 Circular Interpolation

Positioning for circular interpolation is performed using two axis from a starting point to an ending point and traveling through a circular arc.

Basic Idea

Circular interpolation is achieved by repeating successive linear interpolation along straight lines drawn by dividing a circular arc (in the shape of a polygon). Actual locus of circular interpolation is as shown below.



The interpolation acceleration time, the interpolation deceleration time, and triangular control are the same as for the linear interpolation.

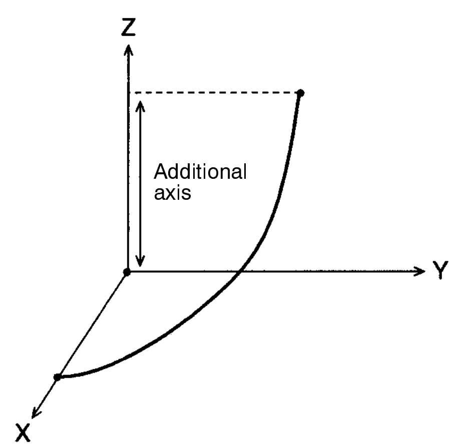
Note Refer to *5-3 Linear Interpolation* for information on the interpolation acceleration time, interpolation deceleration time, and triangular control.

5-5 Helical Interpolation

One axis that makes parallel movements is added to those for the circular interpolation to realize the helical interpolation.

Helical interpolation is not possible for the CV500-MC221, which controls only two axes.

A helical arc is divided into a three-dimensional polygon as for helical interpolation. Shown below is a locus of helical interpolation where the X and Y axes are used for circular interpolation and the Z axis is the third axis.



The feed rate of the Z axis is as follows:
Feed rate = Designated interpolation feed rate x Z axis length/Arc length

The interpolation acceleration time, interpolation deceleration time, and triangular control are the same as for the linear interpolation.

In helical interpolation, the travel time is computed with priority placed on the feed rate along the circular arc. As a result, the feed rate of the third axis may exceed the maximum interpolation feed rate set in the system parameter, which is detected as an error to result in stopping operations. If this happens, lower the feed rate.

Note Refer to *5-3 Linear Interpolation* for information on the interpolation acceleration time, interpolation deceleration time, and triangular control.

5-6 Acceleration and Deceleration Curves

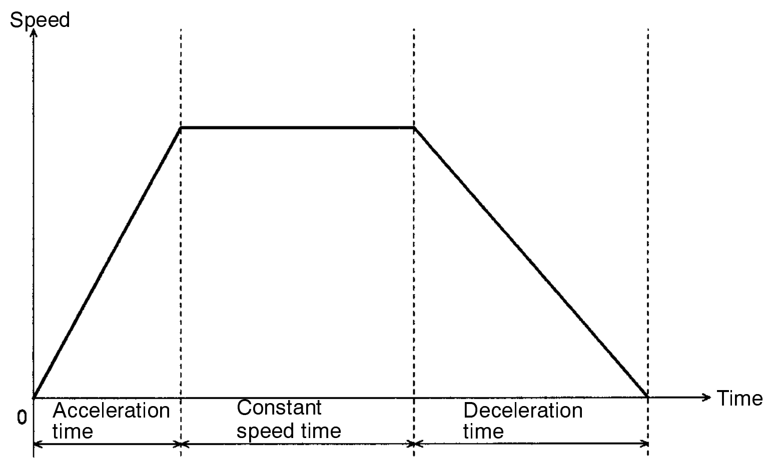
In positioning actions, operating speed is accelerated gradually at the beginning and decelerated gradually toward the end to achieve smooth movement. For the MC Unit, either a trapezoidal curve or an S curve can be used as the acceleration/deceleration curve for the starting/stopping operations for each axis.

The acceleration time and deceleration time can be set separately.

Note Refer to the *MC Support Software Operation Manual* to set the acceleration/deceleration curve and acceleration/deceleration times.

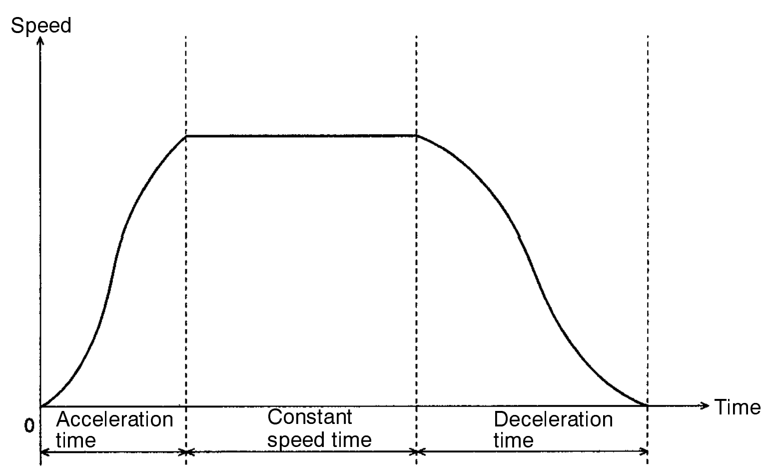
Trapezoidal Curve

With the trapezoidal curve, acceleration is constant during the acceleration time and deceleration time.



S Curve

With the S curve, acceleration during the acceleration time and deceleration time changes with time.



Note The maximum acceleration in the S curve is 1.5 times that in the trapezoidal curve. Therefore, when using a program set for a trapezoidal curve for an S curve, the acceleration and deceleration times set for the trapezoidal curve must be multiplied by 1.5. The maximum acceleration in the S curve will then fall within the acceleration set for the trapezoidal curve, allowing the motor to drive smoothly.

5-7 Operation Modes

Two modes, a stop mode and a pass mode, are available for determining movements from one operation to another when executing continuous positioning. This mode is designated in the control program created in the G language.

Note Refer to the *MC Unit Operation Manual: Details* to designate the operation mode in the G language.

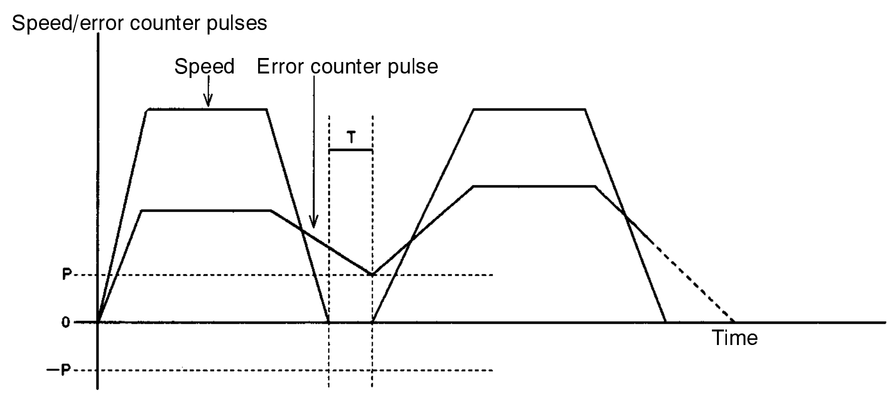
Stop Mode

If the stop mode is specified for continuous positioning operations, one operation is followed by the subsequent operation after completely finishing the first positioning operation.

The following illustration shows changes in the time, speed, and pulses when the axis is controlled as shown below.

Example:

`N010 G00 X100`	Moves X axis to 100-mm point by PTP control
`N011 G00 X200`	Moves X axis to 200-mm point by PTP control



P: In-position width
T: Time required for determining whether positioning is completed or not

Positioning is completed when the error counter pulse falls within the in-position zone.

Pass Mode

If the pass mode is specified for continuous positioning operations, one operation is followed by the subsequent operation without any pause in motion, i.e., without stopping to determine whether positioning is completed or not. When the power is turned on, this mode is automatically selected.

The following illustrations show changes in the speed in the stop mode and the pass mode when the continuous operation commands are given as shown below.

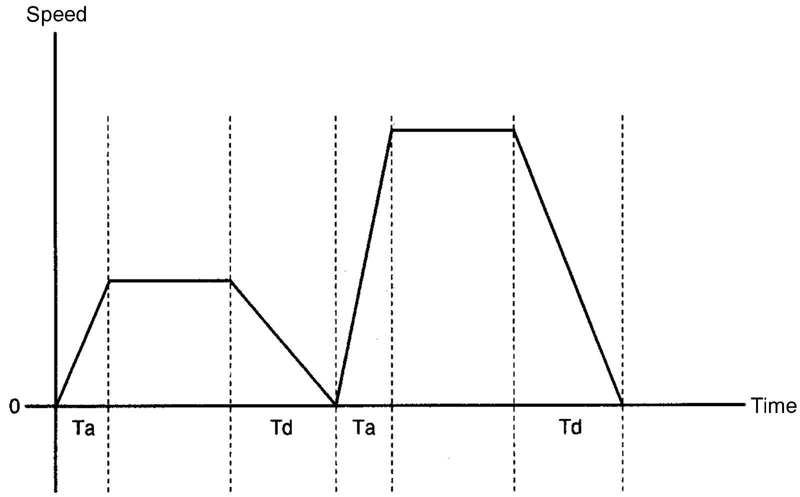
Example:

```
N010 G01 X100 F10
```
Moves X axis to 100-mm point at 10 mm/s via linear interpolation

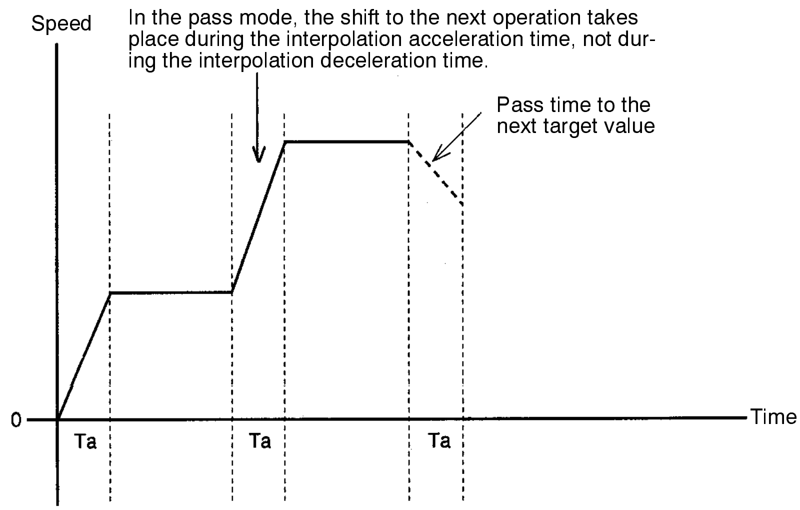
```
N011 G01 X300 F20
```
Moves X axis to 300-mm point at 20 mm/s via linear interpolation

⋮

Stop Mode



Pass Mode



Ta: Interpolation acceleration time
Td: Interpolation deceleration time

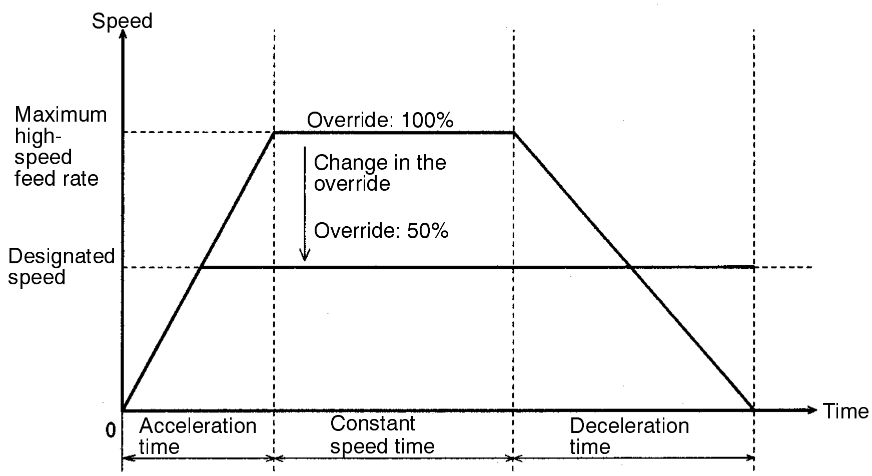
When the pass mode is selected, the time required for movements is reduced because no interpolation deceleration time is required.

5-8 Override Function

The override function is designed to change the operation speed through multiplying the speed that is set in the system parameters or G commands by a desired factor. Setting methods are different depending on whether the PTP control is used or interpolation is used.

Override in PTP Control

The override in the PTP control can be set to a value between 0.1% and 100.0%. "100%" means the maximum high-speed feed rate designated by the feed rate parameter. The override function is used when the maximum high-speed feed rate is too fast. For example, if the override is set to 50%, the travel speed drops to half the maximum high-speed feed rate.



The acceleration remains constant even if the speed is changed by using the override function. As a result, both the acceleration time and the deceleration time are proportionally reduced.

Override in CP Control

In interpolation control, the designated interpolation feed rate is used as the maximum feed rate. The override for the interpolation feed rate can be set to a value between 0.1% and 199.9%. If the result set by using the override function to over 100% exceeds the maximum interpolation speed, maximum interpolation feed rate is used instead. Unlike the PTP control, the acceleration and deceleration times remain constant regardless of the override set value and, as a result, the acceleration and deceleration rates vary.

SECTION 6
G Language

This section introduces the "G" language used for position control with the MC Unit.

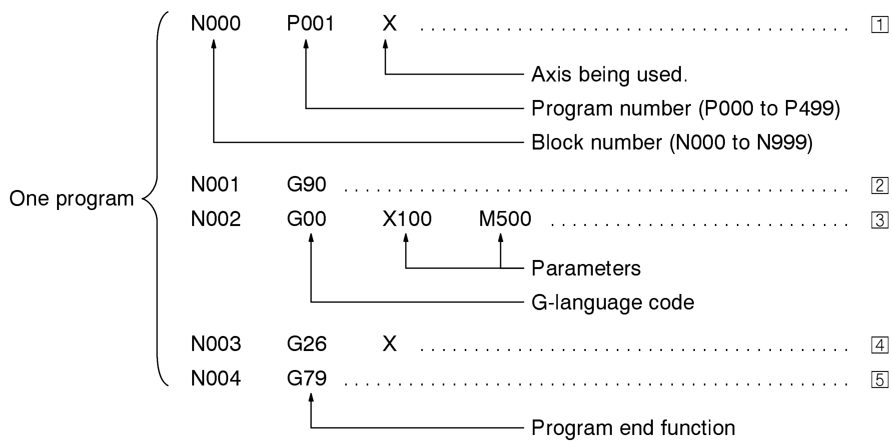
6-1 Introduction

The G language is used widely in position control and its main feature is that it is very easy to write for programming. Program functions can be entered simply by entering a “G” and a 2-digit numerical code, then adding any needed parameters. G-language codes G00 through G91 are used in the MC Units. For example, the function “PTP control positioning” is assigned to G00.

Note Refer to *6-1-2 G-language Codes* for a table showing the functions assigned to the G-language codes.

6-1-1 Example Program

The following diagram shows the format of a basic G-language program.



Block Numbers

Programs are composed of blocks, which are distinguished by block numbers N000 through N999. Block numbers are equivalent to the program’s line numbers.

Always declare the program number and axis in block number N000 ([1]).

Program Numbers

Program numbers range from P000 through P499. The program shown above begins with block number N000 ([1]) and ends with the block (N004) that contains the program end function, G79.

Sub-programs have program numbers ranging from P500 through P999, and end with a sub-program end function, G73.

The block numbers and program numbers are the total of the blocks and programs being used. A maximum of 800 blocks and 100 programs can be used in an MC Unit.

G-language Codes

Code G00 in line [3] is a G-language code representing the PTP control positioning function.

Parameters

Positions are input as parameters. In this example “X100” has been input to indicate 100 on the X-axis.

The next parameter, “M500,” outputs M-code 500 when the positioning to X100 has been completed. Refer to MC *Unit Operation Manual: Details* for details on M codes.

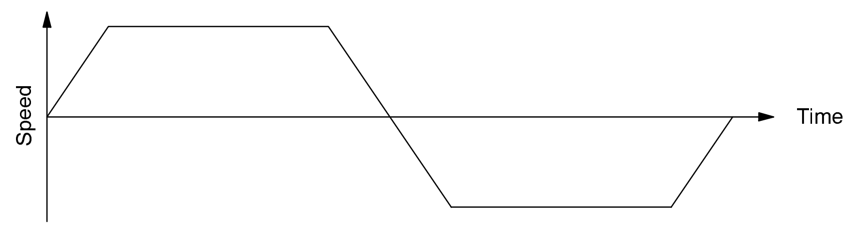
Some functions, such as G90 in line [2], don’t require parameters. The absolute specification function (G90) indicates that coordinates are to be treated as absolute coordinates.

Format

The following table explains the contents of the MC program.

Line	MC program blocks	Function
1	N000 P001 X	Declares program number (001) and the axis being used (X).
2	N001 G90	Specifies positioning by absolute coordinates.
3	N002 G00 X100 M500	Moves to X-axis coordinate 100. Outputs M-code 500 when positioning is completed.
4	N003 G26 X	Returns to the reference origin.
5	N004 G79	Ends the MC program.

The following diagram shows the operation of the example program.



6-1-2 G-language Codes

The following table provides summary and brief description of the G-language functions. Refer to the *MC Unit Operation Manual: Details* for details on these functions. (Codes with asterisks are not supported by the CV500-MC221.)

Code	Name	Function
G00	POSITIONING	Positions up to 4 axes simultaneously with PTP control at the maximum speed.
G01	LINEAR INTERPOLATION	Performs linear interpolation on up to 4 axes simultaneously at the specified interpolation feed speed.
G02	CIRCULAR INTERPOLATION (CLOCKWISE)	Performs 2-axis circular interpolation in the clockwise direction at the specified interpolation feed speed.
G03	CIRCULAR INTERPOLATION (COUNTERCLOCKWISE)	Performs 2-axis circular interpolation in the counterclockwise direction at the specified interpolation feed speed.
G04	DWELL TIMER	Waits for the specified length of time.
G10	PASS MODE	Performs operations one-by-one in sequence without deceleration to stop.
G11	STOP MODE	Performs the next operation after completing positioning.
G17	CIRCULAR PLANE SPECIFICATION (X-Y)	Sets the X-Y plane as the plane for circular interpolation.
G18*	CIRCULAR PLANE SPECIFICATION (X-Z)	Sets the X-Z plane as the plane for circular interpolation.
G19*	CIRCULAR PLANE SPECIFICATION (Y-Z)	Sets the Y-Z plane as the plane for circular interpolation.
G20*	CIRCULAR PLANE SPECIFICATION (X-U)	Sets the X-U plane as the plane for circular interpolation.
G21*	CIRCULAR PLANE SPECIFICATION (Y-U)	Sets the Y-U plane as the plane for circular interpolation.
G22*	CIRCULAR PLANE SPECIFICATION (Z-U)	Sets the Z-U plane as the plane for circular interpolation.
G26	REFERENCE ORIGIN RETURN	Moves to the reference origin.
G27	WORKPIECE ORIGIN RETURN	Moves to the workpiece origin.
G28	ORIGIN SEARCH	Performs an origin search on the specified axis.
G50	SELECT REFERENCE COORDINATE SYSTEM	Specifies the reference coordinate system.
G51	SELECT WORKPIECE COORDINATE SYSTEM	Specifies the workpiece coordinate system.
G53	CHANGE WORKPIECE ORIGIN OFFSET	Changes the origin of the workpiece coordinate system.
G54	CHANGE REFERENCE COORDINATE SYSTEM PV	Changes the present value in the reference coordinate system.
G60	ARITHMETIC OPERATIONS	Performs arithmetic operations on numerical values, position data, and registers.
G63	SUBSTITUTION	Substitutes numerical values, position data, or registers into other position data or registers.
G69	CHANGE PARAMETER	Changes the specified parameter.
G70	UNCONDITIONAL JUMP	Unconditionally jumps to the specified block.
G71	CONDITIONAL JUMP	Jumps to the specified block when the condition is met.
G72	SUBPROGRAM JUMP	Calls the specified subprogram.
G73	SUBPROGRAM END	Ends the subprogram.
G74	OPTIONAL END	Ends the block currently being executed when the specified optional input is ON.
G75	OPTIONAL SKIP	Skips the block after this function when the specified optional input is ON.
G76	OPTIONAL PROGRAM STOP	Pauses the program when the specified optional input is ON.
G79	PROGRAM END	Ends the main program.
G90	ABSOLUTE SPECIFICATION	Positions with absolute coordinates when performing axis operations.
G91	INCREMENTAL SPECIFICATION	Positions with relative coordinates when performing axis operations.

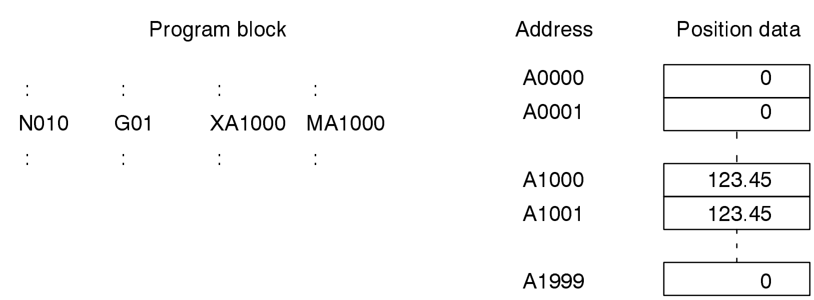
6-2 G-language Symbols

The following table lists the symbols used to define parameters in G-language programming. (Symbols with asterisks are not supported by the CV500-MC221.)

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

6-2-1 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 a parameter in an axis operation command or M code. For example, when the following program is executed, the contents of A1000 (123.45) will be used for the X-axis data and the contents of A1001 (123) will be used for the M code.

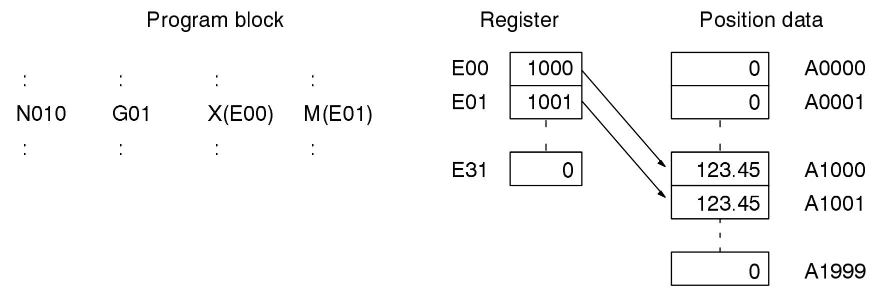


6-2-2 Specifying Registers (E00 to E31)

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

Register		Position data	
Address	**Contents**	**Address**	**Contents**
E00	1000	A1000	123.45
E01	1001	A1001	123.45

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



Registers and position data aren't allocated to tasks; they can be used as desired in 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.

SECTION 7
Preparations for Operation

This section provides information on startup procedures, system configuration, and positioning operations using testing equipment, as well as explanations on the interface area necessary for creating ladder-diagram programs.

7-1 System Startup Procedures

This section describes positioning control procedures using the testing equipment. Refer to the sections shown in parentheses for each step. Descriptions on mechanical design are omitted here.

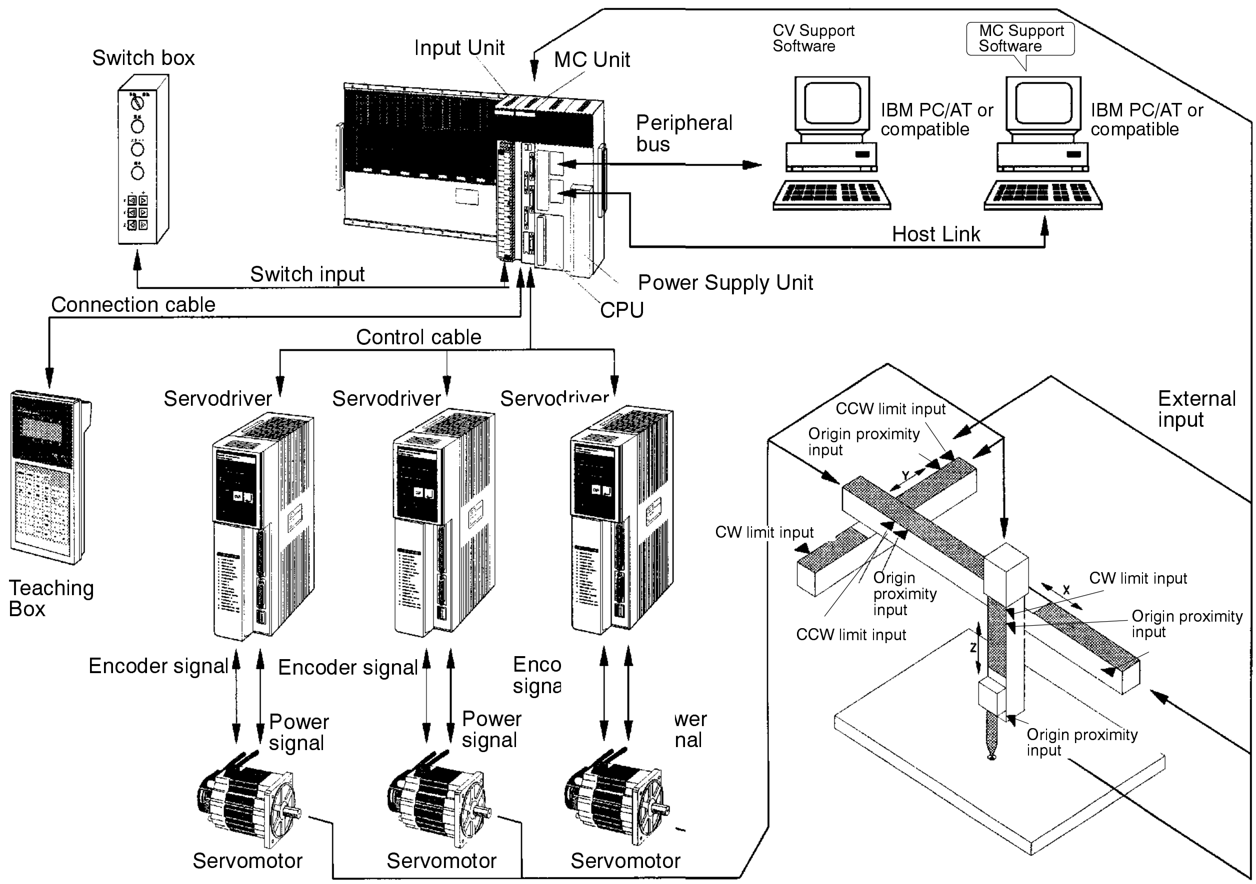
Startup Procedures

1, 2, 3...

1. **Preparation of Devices**
   Prepare devices to be used for operation. (7-2 and 7-3)
2. **Installation of Units**
   Install the CPU, Power Supply Unit, MC Unit, and Input Units. (8-1)
3. **Connection and Wiring**
   Connect the peripheral devices to the Teaching Box, and conduct wiring for the external input, servomotor, servodriver, and Input Unit. (8-2 though 8-6)
4. **Startup of the PC**
   Set the program type and PC model and create an I/O table. (8-7)
5. **Setting, Storing, and Transferring Parameters**
   Set required parameters using the MC Support Software. After storing them in a data disk, transfer them to the MC Unit. (8-8 through 8-10)
6. **Creating, Storing, and Transferring an MC Program**
   Create a program in the G language for test operations using the MC Support Software. After storing the program in a data disk, transfer it to the MC Unit. (8-11 though 8-13)
7. **Creating, Storing, and Transferring a Ladder-diagram Program**
   Create a ladder-diagram program for executing the MC program. After storing the program in a data disk, transfer it to the PC. (8-14)
8. **Operation Verification**
   Execute the MC program to verify the test operation. (8-15 and 8-16)

7-2 Testing Equipment Configuration

The equipment used for testing operations consists of the following devices.



7-3 Items to Be Prepared

Prepare the following items. If more than one models is listed, select one of them. The shaded model is the one used in this manual. All the models without manufactures' names are OMRON products.

Controllers

Name	Model
SYSMAC CV-series PC CPU	CV500-CPU01-EV1 CV1000-CPU01-EV1 CV2000-CPU01-EV1 CVM1-CPU01-EV2 CVM1-CPU11-EV2 CVM1-CPU21-EV2
Motion Control Unit	CV500-MC421
Input Unit	3G2A5-ID213 (See note)
Power Supply Unit	CV500-PS221 (AC type) CV500-PS211 (DC type) CVM1-PA208 (AC type)
Power supply	DC power source (+24 V)
CPU Backplane	CV500-BC101 (10 slots) CV500-BC051 (5 slots) CV500-BC031 (3 slots)
Teaching Box	CVM1-PRS71

Note For other applicable Input Units, refer to the *CVM1/CV500/CV1000/CV2000 PC Installation Guide*.

CV Support Software

Name	Model
Personal computer for CV Support Software	IBM PC/AT or compatible
For a personal computer	CV500-Z53AT1-EV2 (3.5-inch FD) CV500-ZS5AT1-EV2 (5-inch FD)

MC Support Software

Name	Model
Personal computer for MC Support Software	IBM PC/AT or compatible
MC Support Software	CV500-Z3AT1-E (3.5-inch FD)

Servomotor Controller

Name	Model
Servodriver Servomotor Power cable Encoder cable Control cable	Prepare the ones that fit the system.

Other Items

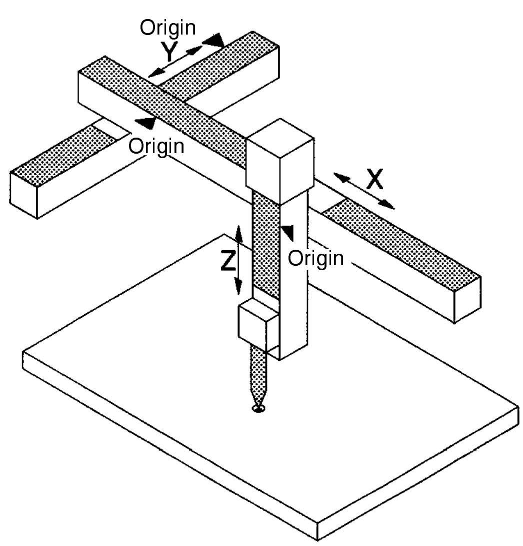
Nonfuse breaker
Magnet relay
Noise filter
Surge killer
Surge absorber
Switch box
Switch
Cables and wires

Note Refer to the operation manual for the servodriver to be used and prepare required items.

7-4 Positioning Operations

Positioning operations are explained below using a 3-axis mechanical system.

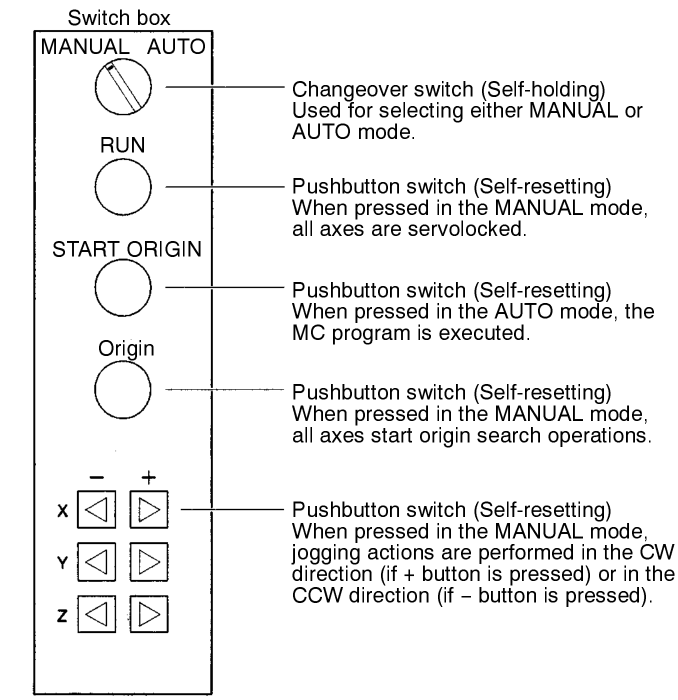
Mechanical System



Note An incremental encoder is used for each axis. Descriptions of drill revolution control is omitted here. The drill is moved to a desired position by controlling the X and Y axes and moved up and down along the Z axis.

Switch Box Specifications

The switch box used in the testing equipment is illustrated below. A switch box and switches should be prepared by the users.

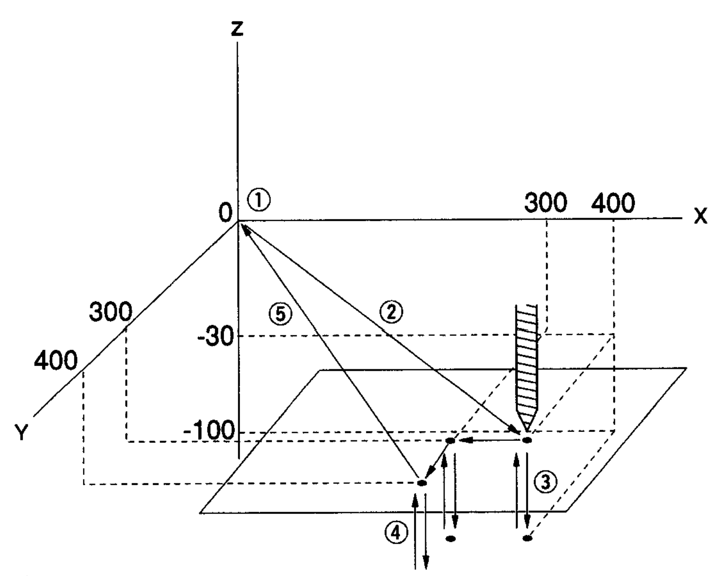


Automatic Operation

Select the AUTO mode using the changeover switch on the switch box and press the START button to execute the MC program. The 3-axis (X/Y/Z) robot will drill a hole at a predetermined position. Drilling operations are performed according to the following steps:

1, 2, 3...

1. Each axis waits at its reference origin.
2. X and Y axes are started to move the drill to a position where the first hole is to be drilled.
3. Z axis is started to lower the drill to drill a hole and then to return to the origin.
4. The robot moves to the next position and repeats the same actions.
5. After drilling three holes, the robot returns to the reference origin.



Manual Operation

Select the MANUAL mode using the changeover switch on the switch box and then press the RUN switch, ORIGIN SEARCH switch, and or JOG switch. The following operations are performed.

RUN Switch

If any other manual command is not being executed and if axes are not servo-locked, X, Y and Z axes are servolocked.

ORIGIN SEARCH Switch

If any other manual command is not being executed and if the origin has not been defined yet, X, Y and Z axes start origin searches.

JOG Switch

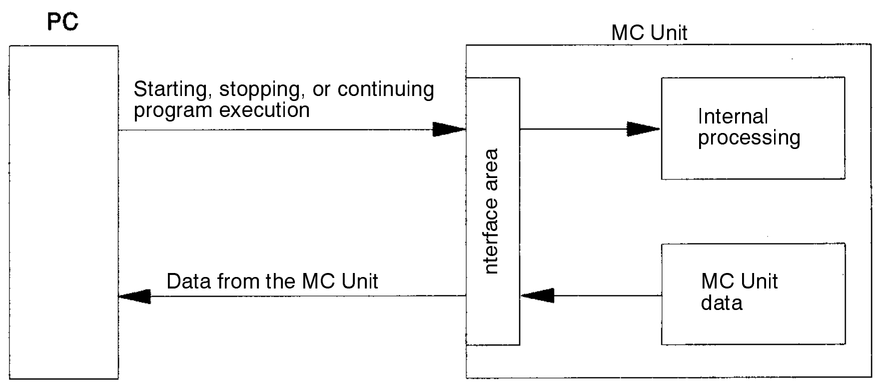
If any other manual command is not being executed and if the axes are servo-locked, X, Y and Z axes start jogging. The motors remains ON while the switch is pressed.

7-5 Interface Area

This section provides a brief explanation of the interface area necessary for creating a ladder-diagram program that executes the MC program.

Definition of Interface Area

The interface area is an area where commands from the PC, status of the MC Unit, and task data are transferred between the PC and the MC Unit.



I/O words and DM Area words are allocated as the interface area by setting the unit number of the MC Unit.

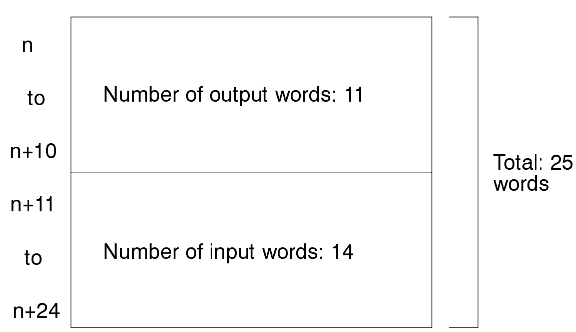
Interface Area Allocation

The interface area used by the cyclic processing consists of the following:
I/O words: 25 words
DM Area: 100 words

The I/O words consist of bits allocated specific functions. These bits are called interface bits. The DM Area words are handled in units of words.

I/O Word Allocation

The 25 I/O words are divided into 11 output (PC to MC Unit) words and 14 input (MC Unit to PC) words.



Note Refer to the *MC Unit Operation Manual: Details* for unit number settings.

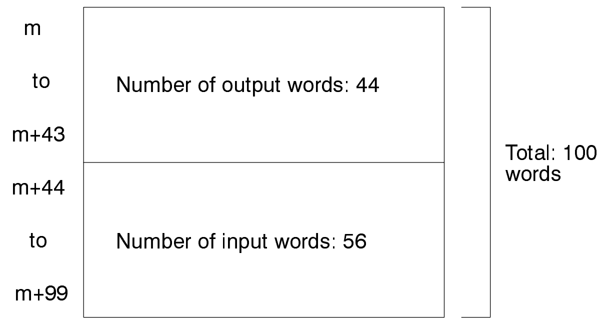
I/O words are allocated based on the unit number setting of the MC Unit:

First I/O word: n = 1500 + Unit number x 25

For example, if the unit number is set to 00, I/O words are allocated from CIO 1500 to CIO 1524.

DM Area Word Allocation

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



DM Area words are allocated based on the unit number setting of the MC Unit.

First DM Area word: m = 2000+ Unit number x 100

For example, if the unit number is set to 00, DM Area words are allocated from D2000 to D2099.

Outline of the I/O Words

Each word is assigned a control function or a status from the MC Unit. Furthermore, each bit of each word is assigned a function or a status.

Output area (PC to MC Unit): 11 words

Word	Contents
n	All tasks common 1 System error reset, etc.
n+1	All tasks common 2 Optional inputs
n+2	Task 1 control bits Cycle start, etc.
n+3	Task 2 control bits
n+4	Task 3 control bits
n+5	Task 4 control bits
n+6	Not used
n+7	X-axis control bits Origin search, etc.
n+8	Y-axis control bits
n+9	Z-axis control bits
n+10	U-axis control bits

Input area (MC Unit to PC): 14 words

Word	Contents
n+11	All tasks common 3 System error, etc.
n+12	All tasks common 4 General-purpose I/O
n+13	Task 1 status Task error, etc.
n+14	Task 2 status
n+15	Task 3 status
n+16	Task 4 status
n+17	X-axis status 1 Reference origin, zone bit, etc.
n+18	X-axis status 2 Servolocked, CCW limit input, etc.
n+19	Y-axis status 1
n+20	Y-axis status 2
n+21	Z-axis status 1
n+22	Z-axis status 2
n+23	U-axis status 1
n+24	U-axis status 2

n+7 X-axis control bit

Bit	Bit name
00	Deceleration stop
01	Origin search
02	Reference origin return
03	Jogging
04	Not used
05	Present position preset
06	Reserved
07	Reserved
08	MPG enabled
09	Servo lock
10	Servo free
11	Driver alarm reset
12	Override set
13	Jogging direction
14	MPG multiplication factor 1
15	MPG multiplication factor 2

For example, if the bit “01” (origin search) of the word n+7 is turned ON, the X axis executes origin search operations.

Note Refer to the *MC Unit Operation Manual: Details* for further information.

Outline of the DM Area Words DM Area is assigned to command data from the PC or status from the MC Unit.

Words	Contents	Area
m to m+3		Output area (PC to MC Unit): 44 words
m+4 to m+7	Same as the above (except number is 2)	
m+8 to m+11	Same as the above (except number is 3)	
m+12 to m+15	Same as the above (except number is 4)	
m+16 to m+18	Task 1 command information Program number Teaching start address Present position designation	
m+19 to m+21	Task 2 command information	
m+22 to m+24	Task 3 command information	
m+25 to m+27	Task 4 command information	
m+28 to m+30	Not used	
m+31 to m+33	X-axis command information Present position preset value Override	
m+34 to m+36	Y-axis command information	
m+37 to m+39	Z-axis command information	
m+40 to m+42	U-axis command information	
m+43	Not used	
m+44 to m+65	All tasks common 2 System version System error code Task error code Axis error code Receive data (1 to 4) supplementary information Receive data (1 to 4)	Input area (MC Unit to PC): 56 words
m+66 to m+70	Task 1 status Execution program number Execution block number M code Teaching execution address (m+70 is not used.)	
m+71 to m+75	Task 2 status (m+75 is not used)	
m+76 to m+80	Task 3 status (m+80 is not used)	
m+81 to m+85	Task 4 status (m+85 is not used)	
m+86	Not used	
m+87 to m+89	X-axis status Additional information of the present position Present position Present position	
m+90 to m+92	Y-axis status	
m+93 to m+95	Z-axis status	
m+96 to m+98	U-axis status	
m+99	Not used	

Note Refer to the *MC Unit Operation Manual: Details* for further information.

SECTION 8
Test Operation

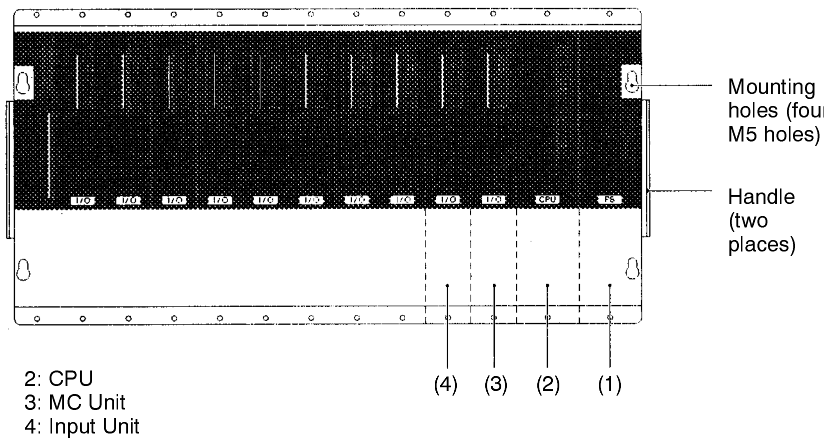
This section explains the procedures involved in conducting tests using test equipment.

8-1 Mounting the Units

The Power Supply Unit, CPU, MC Unit, and Input Unit are all mounted to the CPU Backplane.

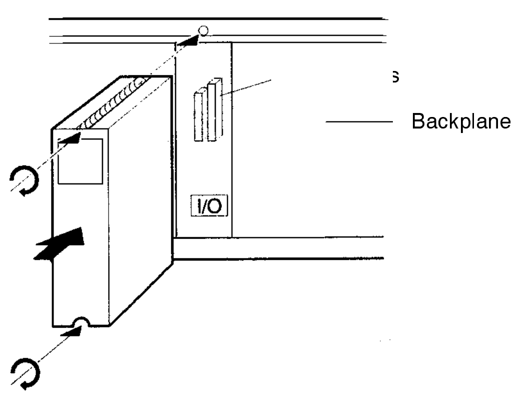
8-1-1 Mounting Positions

The mounting positions of the Power Supply Unit and the CPU are fixed. The positions of the MC Unit and the Input Unit are not fixed, but for the purposes of this explanation, assume that they are mounted in the positions shown in the following illustration.



8-1-2 Mounting the Units

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



After the MC Unit has been mounted, set the unit number to 00.

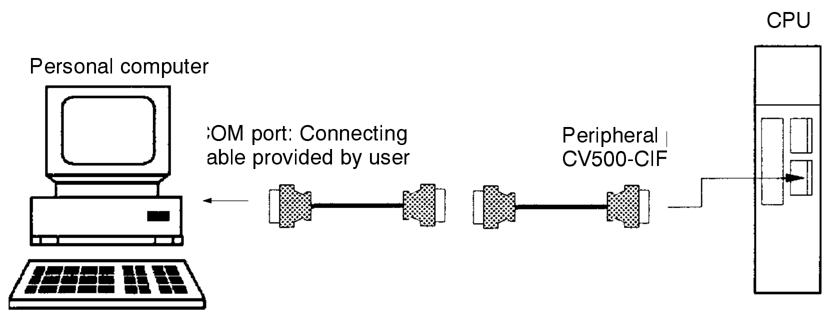
Note For details concerning Unit numbers, refer to the *MC Unit Operation Manuals: Details*.

8-2 Connecting the Computer

Two methods are explained below for connecting the computer running the CV Support Software and MC Support Software. The first method connects the computer and the CPU via the peripheral bus. The second method connects them via a Host Link.

8-2-1 Connecting via the Peripheral Bus

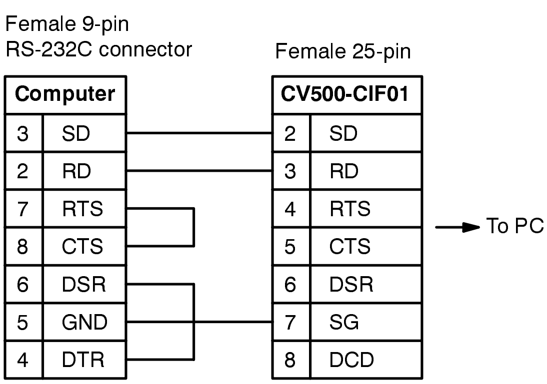
Use a CV500-CIF01 Connecting Cable (6 meters) to connect the computer and CPU via the peripheral bus. The user will also need to provide a separate connecting cable that fits the connectors at the computer.



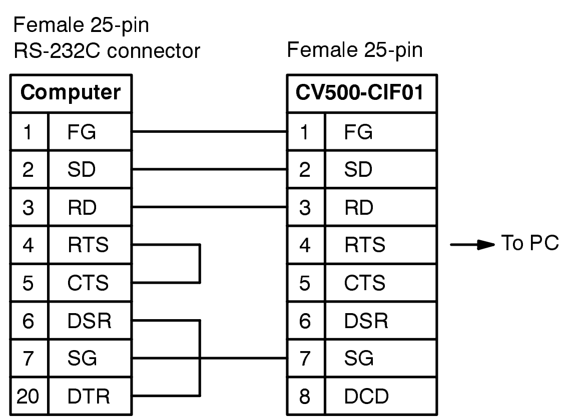
⚠ Caution Always connect the CV500-CIF01 to the computer before connecting it to the CPU. Elements inside the CPU may be damaged if the cable is connected to the CPU first.

Wiring Diagrams An adapter is required to connect the CV500-CIF01 cable to a computer. Wire the adapter as shown below.

IBM PC/AT Computer



IBM PS/2 Computer



Communications Settings After the cables have been connected, check to make sure that the DIP switch at the CPU is set to a baud rate of 50.0 kbps.

Note For details concerning CPU DIP switch settings, refer to the *CV500/CV1000/CV2000/CVM1 Operation Manual*.

Pin number		Baud rate (bps)
1	2	
OFF	OFF	50,000
ON	OFF	19,200
OFF	ON	9,600
ON	ON	4,800

Turn on the power to the computer and start up the CV Support Software. In the CVSS system setup, set the communications format to binary. Then set the baud rate to 9,600 bps and the response monitoring time to 10 seconds.

Note For details concerning the CVSS system setup, refer to the *CV Support Software Operation Manual*.

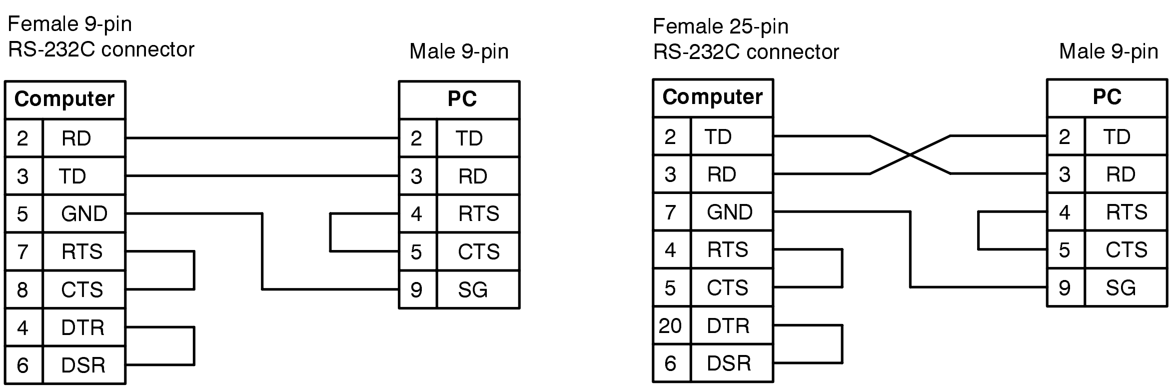
When the settings have been made, set the baud rate at the CPU's DIP switch so that it matches the setting made in the CVSS system setup.

8-2-2 Connecting via the Host Link

The user must prepare the cable for connecting the computer and CPU by Host Link.

Connection Diagram

Wiring the connecting cable as shown in the following illustration.



Turn off the computer and PC when the connecting the cables. The maximum cable length is 15 meters.

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

Applicable Connectors

Connected to	Connector
Personal computer (25 pins)	XM2A-2501 (connector)
	XM2S-2511 (connector hood)
CPU (9 pins)	XM2A-0901 (connector)

Connection Signals at the Computer

Pin number	Signal	Name
1	FG	Frame Ground
2	SD	Send Data
3	RD	Receive Data
4	RS	Request Send
5	CS	Can Send
7	SG	Signal Ground
6	DR	Data Ready
8	CD	Carrier Detect
20	ER	Equipment Ready

In the CVSS system setup, set the communications format to Host Link. Then set the communications specifications. The specifications should normally be set as shown below.

Baud rate:	9,600 bps
Unit number:	00
Parity:	Even
Data length:	7 bits
Stop bits:	2
Response monitoring time:	10 seconds

Note For details concerning the CVSS system setup, refer to the *CV Support Software Operation Manual*.

Set the Host Link RS-422/232 switch at the PC to RS-232C. Then set the communications format and specifications at the PC to match the communications format that have been set for the MCSS. The procedure is as follows:

1, 2, 3...

1. Turn on the power to the computer.
2. Switch the CVSS to online mode.
3. Select “A: PC setup” from the main online menu.
4. Select “A: PC setup” from the PC Setup menu.
5. Select “H: Host Link” from the PC Setup menu.
6. Make the settings just as with MCSS, or check to make sure that the settings are the same.

Note The PC setup can be specified in either the online or offline mode. If the PC setup is changed in the offline mode, execute “T: Transfer PC settings” in the online mode PC setup to transfer the settings to the PC. If the settings are not transferred, the information at the PC will not be changed. For details concerning the CVSS system setup and PC setup, refer to the *CV Support Software Operation Manual*.

8-3 Connecting the Teaching Box

This section explains how to connect the Teaching Box and MC Unit.

Connecting Cable

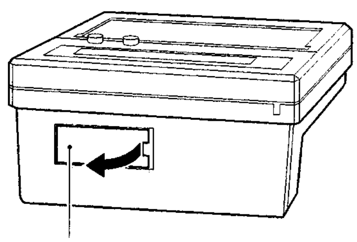
Use the connecting cables shown in the following table to connect the Teaching Box to the MC Unit.

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

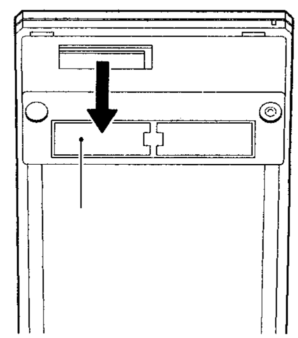
Connecting to the MC Unit

1, 2, 3...

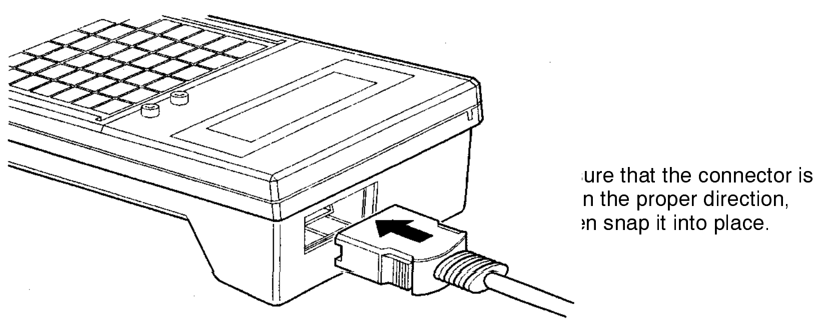
1. Remove the connector cover.



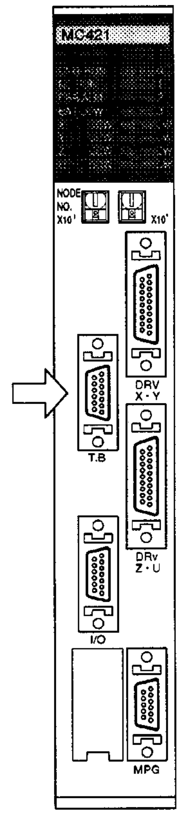
The cover that has been removed can be attached to the back of the Unit for convenience, to keep it from getting lost.



2. Insert the cable connector into the Teaching Box.

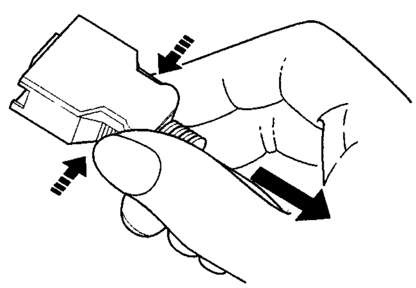


3. Insert the other cable connector into the MC Unit connector marked “T.B.”



Removing the Connecting Cable

Squeeze the lock release buttons located on both sides of the connector, and pull the connector out.



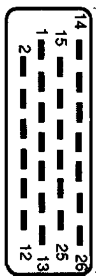
8-4 Wiring I/O Connectors

This section shows how to wire the CW and CCW limit inputs and the origin proximity inputs for each axis, and provides the names and wiring of the I/O connector terminals.

Connector

Use a half-pitch, bellows plug.

Pin Assignments



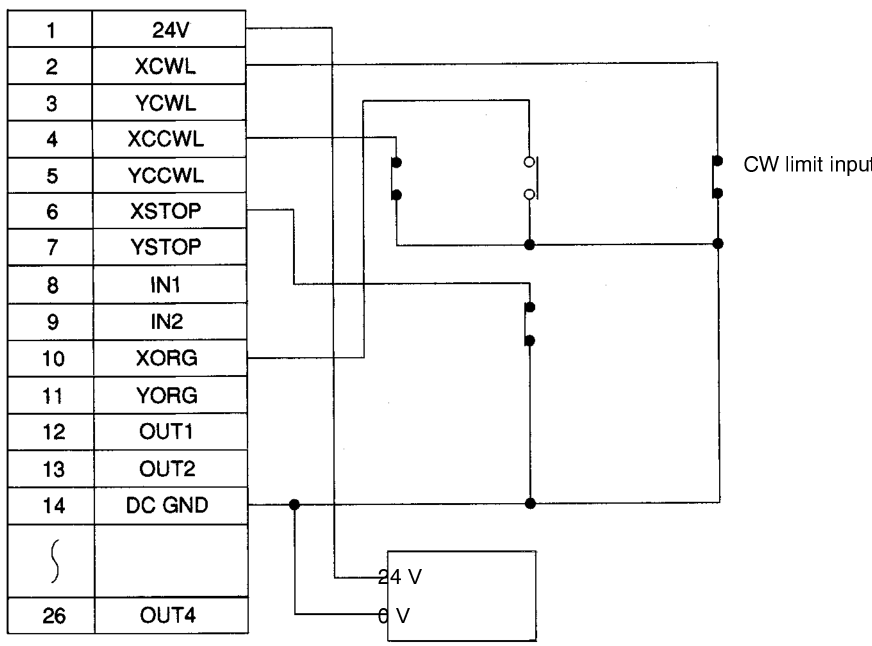
Connector pin arrangement

2	XCWL	1	+24	15	ZCWL	14	DC GND
4	XCCWL	3	YCWL	17	ZCCWL	16	UCWL
6	XSTOP	5	YCCWL	19	ZSTOP	18	UCCWL
8	IN1	7	YSTOP	21	IN3	20	USTOP
10	XORG	9	IN2	23	ZORG	22	IN4
12	OUT1	11	YORG	25	OUT3	24	UORG
		13	OUT2			26	OUT4

Connection Diagram

The following diagram shows the wiring for the X axis only. Wire the Y and Z axes in the same way.

I/O Connector



I/O Connector Terminals

Pin no.	Symbol	Name	Explanation
1	24 V	24-V Input	Connects the positive terminal of the 24-V external power supply.
2	XCWL (NC)	X-axis CW Limit Input	Limits X-axis movement in the CW direction.
3	YCWL (NC)	Y-axis CW Limit Input	Limits Y-axis movement in the CW direction.
4	XCCWL (NC)	X-axis CCW Limit Input	Limits X-axis movement in the CCW direction.
5	YCCWL (NC)	Y-axis CCW Limit Input	Limits Y-axis movement in the CCW direction.
6	XSTOP (NC)	X-axis Emergency Stop Input	Disables the X-axis command output and stops the X axis.
7	YSTOP (NC)	Y-axis Emergency Stop Input	Disables the Y-axis command output and stops the Y axis.
8	IN1 (NO)	General Input 1	General-purpose input no. 1
9	IN2 (NO)	General Input 2	General-purpose input no. 2
10	XORG (NC, NO)[1]	X-axis Origin Proximity Input	Used for X-axis origin search.
11	YORG (NC, NO)[1]	Y-axis Origin Proximity Input	Used for Y-axis origin search.
12	OUT1	General Output 1	General-purpose output no. 1
13	OUT2	General Output 2	General-purpose output no. 2
14	DC GND	24-V Input Ground	Connects the – terminal (0 V) of the 24-V external power supply.
15	ZCWL (NC)	Z-axis CW Limit Input	Limits Z-axis movement in the CW direction.
16	UCWL (NC)	U-axis CW Limit Input	Limits U-axis movement in the CW direction.
17	ZCCWL (NC)	Z-axis CCW Limit Input	Limits Z-axis movement in the CCW direction.
18	UCCWL (NC)	U-axis CCW Limit Input	Limits U-axis movement in the CCW direction.
19	ZSTOP (NC)	Z-axis Emergency Stop Input	Disables the Z-axis command output and stops the Z axis.
20	USTOP (NC)	U-axis Emergency Stop Input	Disables the U-axis command output and stops the U axis.
21	IN3 (NO)	General Input 3	General-purpose input no. 3
22	IN4 (NO)	General Input 4	General-purpose input no. 4
23	ZORG (NC, NO)[1]	Z-axis Origin Proximity Input	Used for Z-axis origin search.
24	UORG (NC, NO)[1]	U-axis Origin Proximity Input	Used for U-axis origin search.
25	OUT3	General Output 3	General-purpose output no. 3
26	OUT4	General Output 4	General-purpose output no. 4

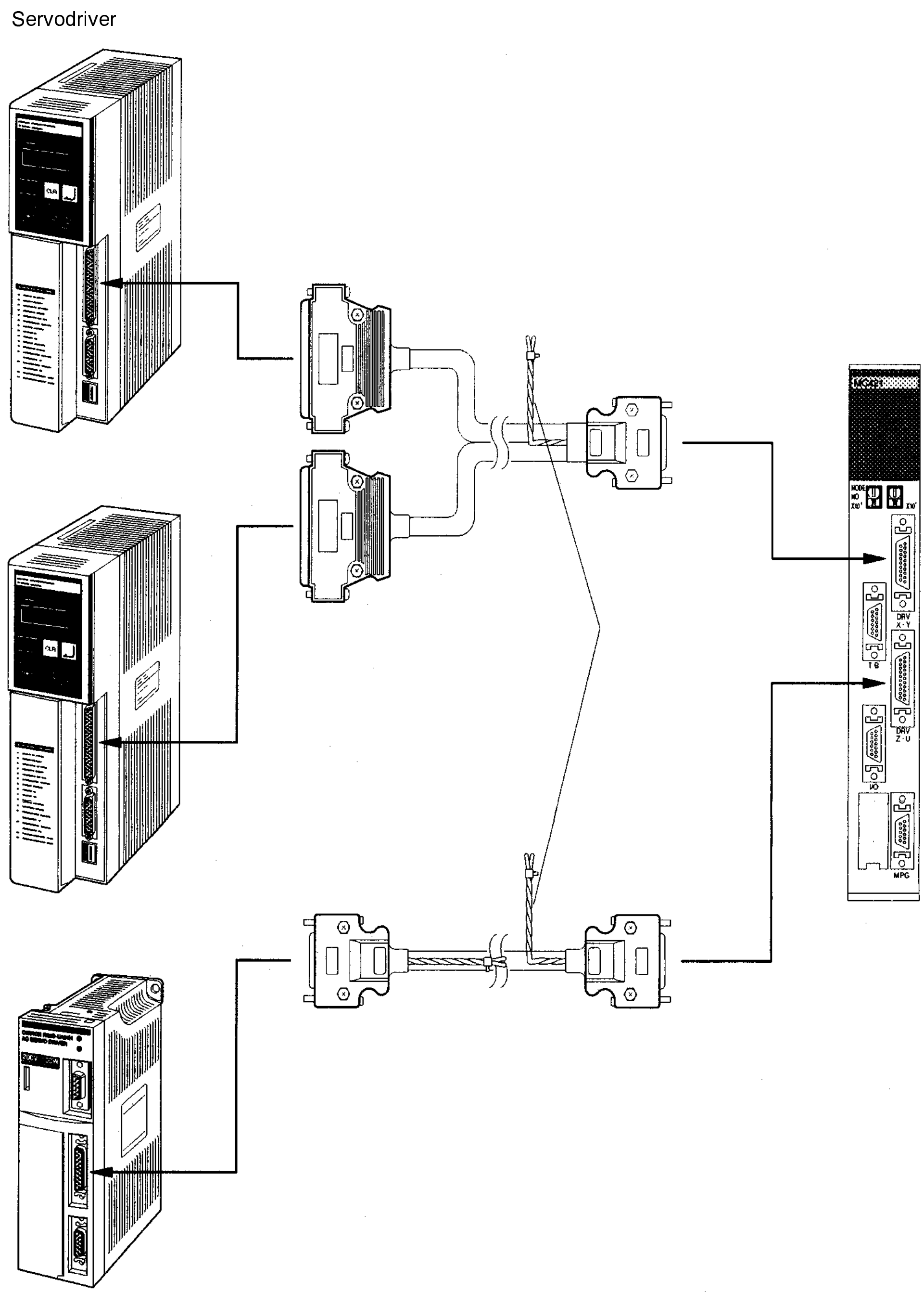
Note

1. The origin proximity input can use any of the settings for the origin proximity input logic parameter set from the MCSS.
2. NC stands for normally closed, and NO stands for normally open.
3. Normally-closed input terminals must be shorted-circuited at the connector when not used.

8-5 Connecting Servodrivers

Connect the MC Unit and servodrivers using control cables. The following is an example of the servodriver connections.

Connecting Control Cables

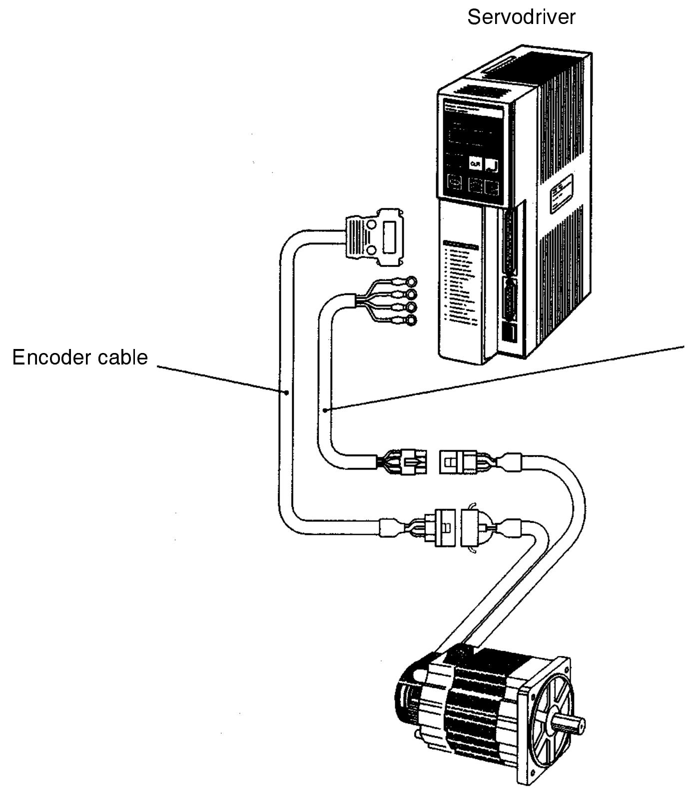


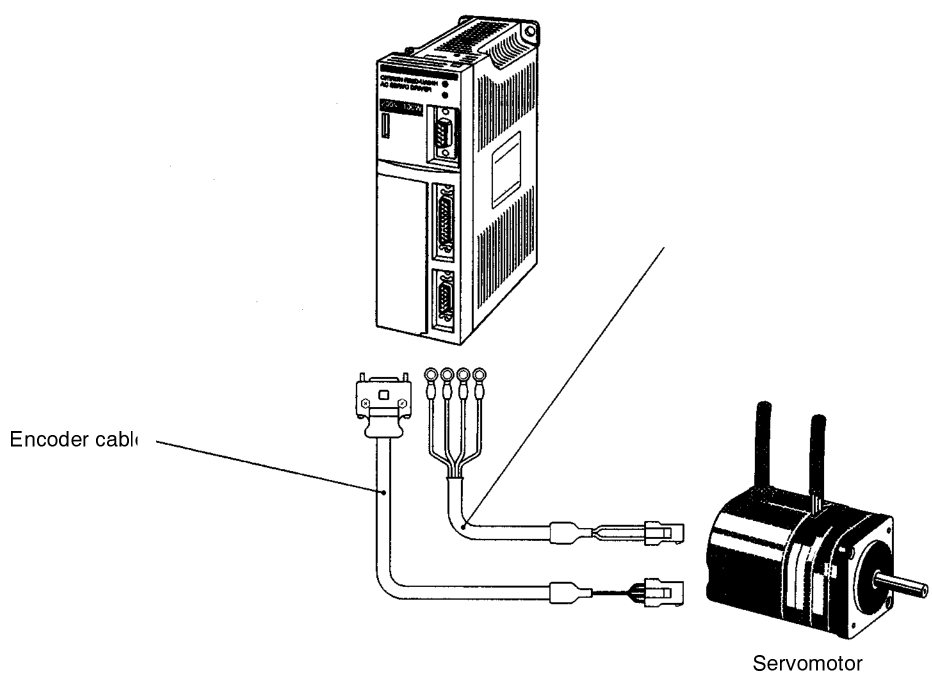
Control Cables

Refer to the *MC Unit Operation Manual: Details* for information on control cables.

Connecting Power Cables and Encoder Cables

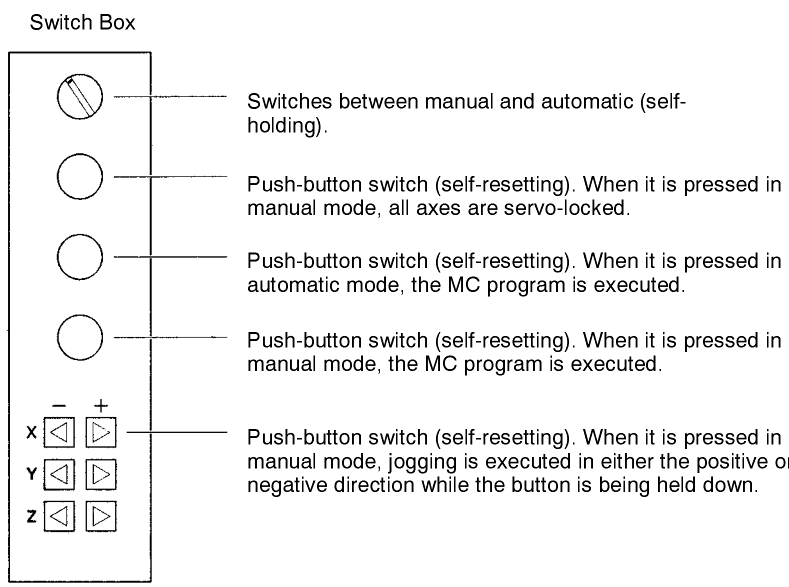
The following are examples using servodrivers. Refer to the applicable servodriver manual for more details.



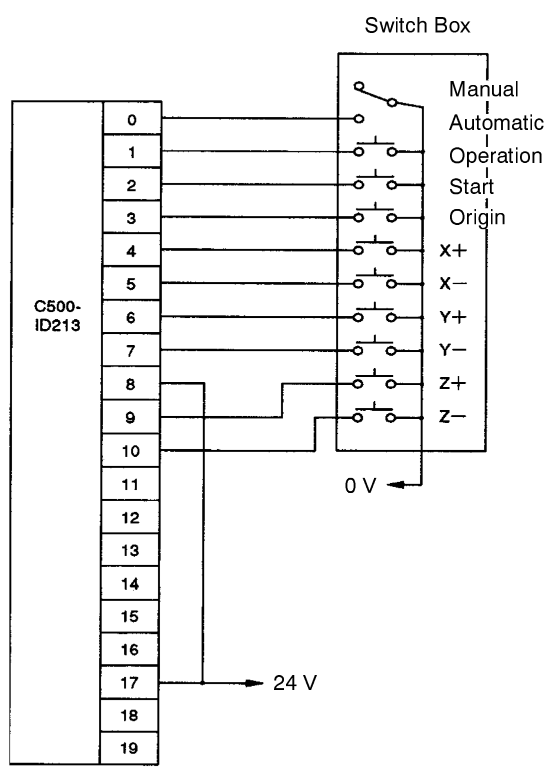


8-6 Wiring Input Units

This section shows how to wire a switch box to an Input Unit. First, prepare a switch box as shown in the following illustration. (This must be done by the user.)



After the switch box has been prepared, wire it to the Input Unit as shown below.



8-7 Starting Up the PC

This section explains how to set the program type and the PC model, and how to create the I/O table. These settings must be made when first mounting the CPU and MC Unit to the CPU Backplane.

It is assumed here that the CVSS has already been installed. Refer to the relevant sections of the *CV Support Software Operation Manual* for explanations of installation operations.

1, 2, 3...

1. Start up the CVSS, and then set the program to “ladder” and the PC model to “CV1000-CPU01.”
2. Switch the CVSS to online mode.
3. Create the I/O table according to the status of the Units that are mounted to the Backplane.
4. Return the CVSS to offline mode so that the ladder-diagram program can be created.

8-8 Setting Parameters

This section explains how to set the parameters that are necessary for controlling the MC program. (*8-11 Creating the MC Program* explains how to create the program.)

Set the parameters by changing some of the default values for the MCSS. After the parameters have been set, transfer them to the MC Unit.

The parameters that must be changed or reset are shown in the following table.

Parameters	Item	Set value
MC parameters	Number of axes	3 axes
	Task 1 axes	XYZ
Machine parameters	Minimum setting unit	0.01
	Display unit	mm
	Pulse rate	1/100
	Positive and negative software limits	X axis, negative: –50.00
		X axis, positive: 600.00
		Y axis, negative: –50.00
		Y axis, positive: 500.00
		Z axis, negative: –500.00
		Z axis, positive: 100.00
Coordinate parameters	Reference origin offset value	0
	Workpiece origin offset value	0
Feed rate parameters	Maximum high-speed feed rate	2,048.00
	Maximum interpolation feed rate	2,048.00
	Origin search high speed	204.80
	Origin search low speed	20.48
	Maximum jog feed rate	204.8
Servo parameters	In-position	5

For items with no axis settings, set the same values for axes X, Y, and Z. The methods for setting individual parameters are explained below and on subsequent pages in this section. The procedures are shown using the X-axis setting screens as an example. To make the settings for either the Y or the Z axis, press the End Key to display the Machine Parameter Edit, Feed Rate Parameter Edit, and Servo Parameter Edit menus, and select the axes that are to be edited.

It is assumed here that the MCSS has already been installed.

8-8-1 Number of Axes

1, 2, 3...

1. Turn on the power supply to the computer, and start up the MCSS.
2. Select “W:Edit unit parameters” from the MC Parameter Edit Menu.

```
                                                 Unit Parameters
<CV500-MC421 >
                                            ESC: previous
  (          )
   [ Unit Parameter Edit ]
   A:Number of axes                                   4 axis
   B:Number of tasks (1 to 4)                         1 item
   C:Task 1 axis                                      XYZU
   D:Task 2 axis                                        *
   E:Task 3 axis                                        *
   F:Task 4 axis                                        *
```

3. Select “A:Number of Axes” from the Unit Parameter Edit Menu.

```
<CV500-MC421 >                    Number of axis     Unit Parameters
                                            ESC: previous
  (          )
   [ Unit Parameter Edit ]
                                                      4 axis
    [ # of Axis ]   (1 to 4)                          1 item
    1:1 axis                                          XYZU
    2:2 axis                                            *
    3:3 axis                                            *
    4:4 axis                                            *
```

4. Select “3:3 Axis” from the Axis Configuration Menu.

8-8-2 Task 1 Axes

1, 2, 3...

1. Select “W:Edit Unit Parameters” from the MC Parameter Edit Menu.
2. Select “C:Task 1 axis” from the Unit Parameter Edit Menu.

```
<CV500-MC421 >                    Task 1 Axis      Unit Parameters
                                            ESC: previous
  (          )
  [ Unit Parameter Edit ]
                                                 3 axis
   [ Task 1 Axis ]                               1 item
   A:XYZU                                         XYZU
   B:XYZ                                            *
   C:XYU                                            *
   D:XZU                                            *
   E:YZU
   F:XY
   G:XZ
   H:XU
   I:YZ
   J:YU
   K:ZU
   L:X
   M:Y
   N:Z
   O:U
   P:*
```

3. Select “B:XYZ” from the Task 1 Axis Menu.

```
<CV500-MC421 >                                     Unit Parameters
                                            ESC: previous
  (          )
  [ Unit Parameter Edit ]
  A:Number of axes                               3 axis
  B:Number of tasks (1 to 4)                     1 item
  C:Task 1 axis                                   XYZU
  D:Task 2 axis                                     *
  E:Task 3 axis                                     *
  F:Task 4 axis                                     *
```

8-8-3 Minimum Unit

1, 2, 3...

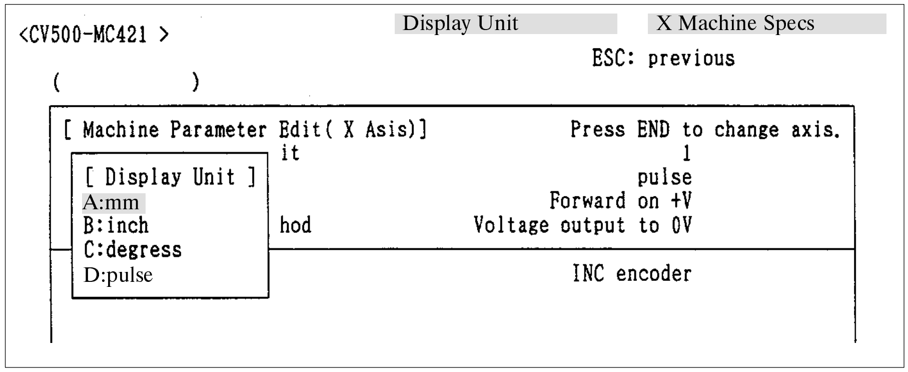
1. Select “A:Minimum Setting Unit” from the Machine Parameter Edit screen.

```
<CV500-MC421 >             Minimum Setting Unit   X Machine Specs
                                            ESC: previous
  (          )
  [ Machine Parameter Edit: X Asis ]        Press END to change axis.
                                                 1
   [ Minimum Setting Unit ]                    pulse
   A:1                                    Forward on +V
   B:0.1                              Voltage output to 0V
   C:0.01
   D:0.001                                 INC encoder
   E:0.0001                                   2048 ppr
```

2. Select “C.”

8-8-4 Display Unit

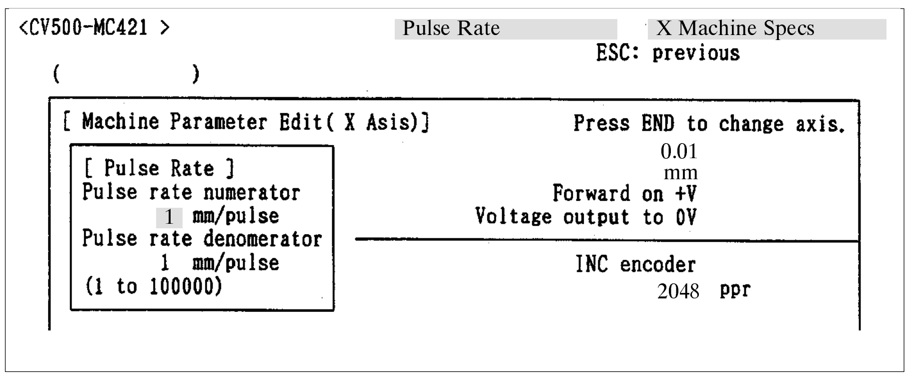
1, 2, 3... 1. Select “B:Display Unit” from the Machine Parameter Edit screen.



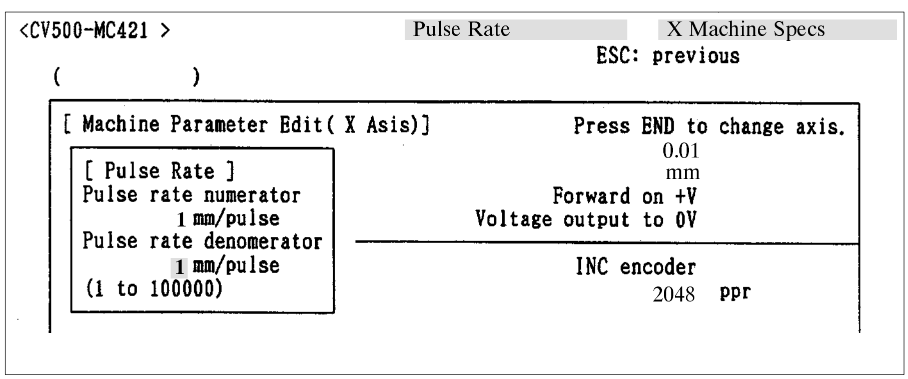
2. Select “A.”

8-8-5 Pulse Rate

1, 2, 3... 1. Select “H:Pulse Rate” from the Machine Parameter Edit screen.



2. The pulse rate numerator is not changed here, so press either the Down (↓) Key or the Return Key. Move the cursor to the pulse rate denominator value.



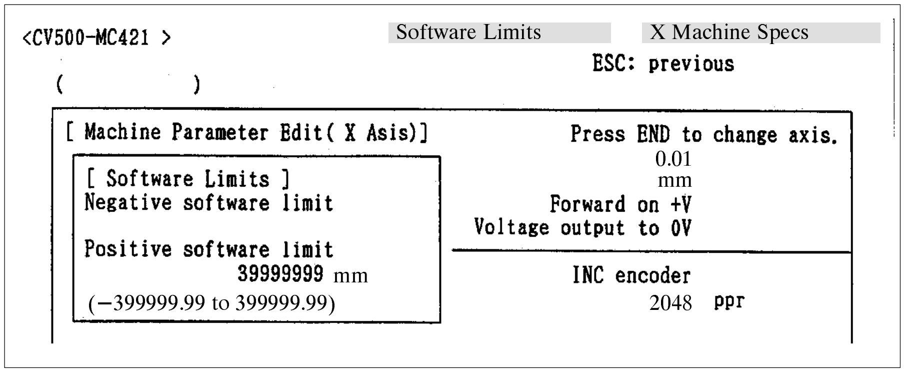
3. Enter the pulse rate denominator value (100 in this example). After the correct setting has been made, return to the Machine Parameter Edit screen.

Note If the pulse rate is set to a value greater than 1, it will not be possible to return to the Machine Parameter Edit screen by pressing the Return Key. If this situation occurs, reset the pulse rate and then try again.

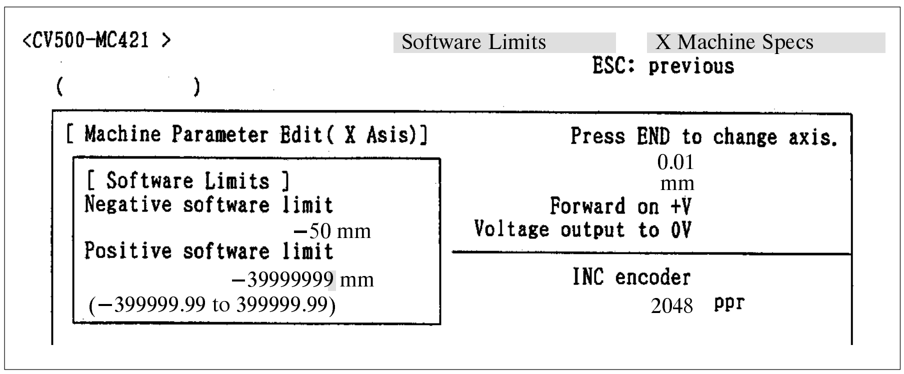
8-8-6 Positive and Negative Software Limits

1, 2, 3...

1. Select "J:Negative Software Limit" from the Machine Parameter Edit screen.



2. Enter the negative software limit (for this example, enter "–50") and then press the Return Key.

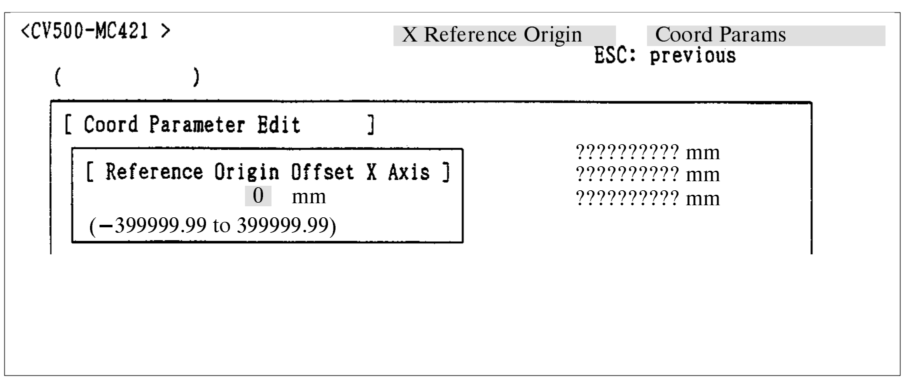


3. Enter the positive software limit (for this example, enter "600") and then press the Return Key.

8-8-7 Reference Origin Offset

1, 2, 3...

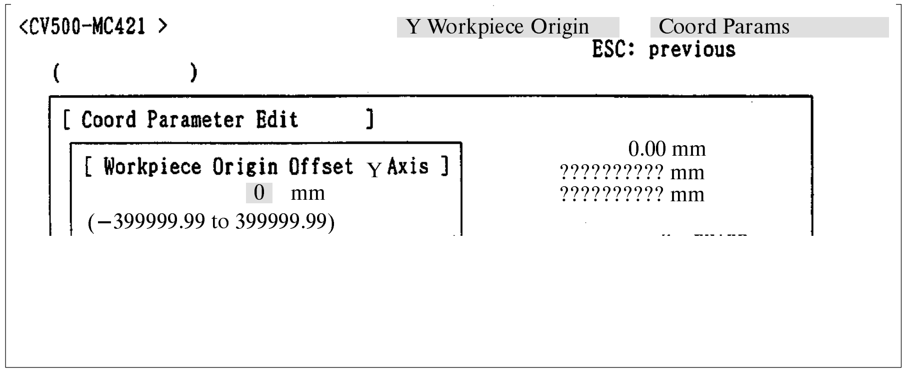
1. Select "A:Reference Origin Offset X Axis" from the Coordinate Parameter Edit screen.



2. Enter "0" for the offset value and press the Return Key.

8-8-8 Workpiece Origin Offset Value

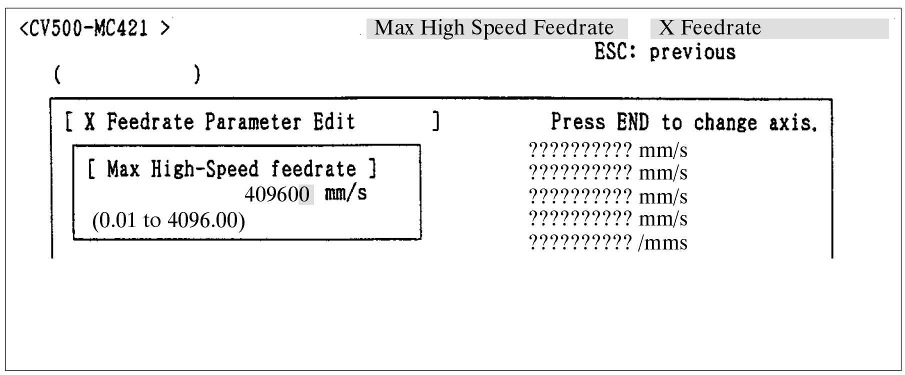
1, 2, 3... 1. Select "D:Workpiece Origin Offset Y Axis" from the Coordinate Parameter Edit screen.



2. Enter "0" for the offset value and press the Return Key.

8-8-9 Maximum High-speed Feed Rate

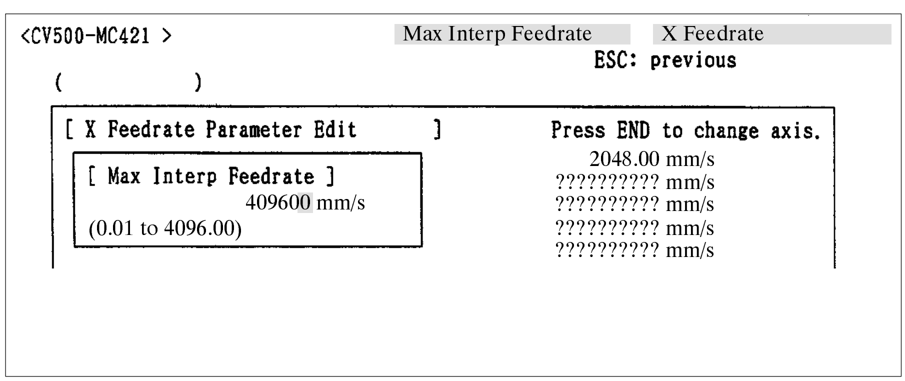
1, 2, 3... 1. Select "A:Max High-speed Feedrate" from the Feedrate Parameter Edit screen.



2. Enter the maximum feed rate ("2,048" in this example).

8-8-10 Maximum Interpolation Feed Rate

1, 2, 3... 1. Select "B:Maximum Interpolation Feedrate" from the Feedrate Parameter Edit screen.



2. Enter the maximum feed rate ("2,048" in this example).

8-8-11 Origin Search High Speed

1, 2, 3... 1. Select "C:Origin Search High Speed" from the Feedrate Parameter Edit screen.

```
<CV500-MC421 >                      Orgn Srch High Speed    X Feedrate
                                                ESC: previous
  (          )
 [ X Feedrate Parameter Edit      ]        Press END to change axis.
  [ Orgn srch High Speed ]                      2048.00 mm/s
            40960 mm/s                          2048.00 mm/s
  (0.01 to 4096.00)                          ?????????? mm/s
                                             ?????????? mm/s
                                             ?????????? mm/s
```

2. Enter "204.8" and press the Return Key. An error warning will appear, but just disregard it and press any key.

8-8-12 Origin Search Low Speed

1, 2, 3... 1. Select "C:Origin Search Low Speed" from the Feedrate Parameter Edit screen.

```
<CV500-MC421 >                      Orgn Srch Low Speed     X Feedrate
                                                ESC: previous
  (          )
 [ X Feedrate Parameter Edit      ]        Press END to change axis.
  [ Orgn Srch Low Speed ]                     2048.00 mm/s
             4096  mm/s                       2048.00 mm/s
  (0.01 to 4096.00)                            204.80 mm/s
                                           ?????????? mm/s
                                           ?????????? mm/s
```

2. Enter "20.48" and press the Return Key.

8-8-13 Maximum Jog Feed Rate

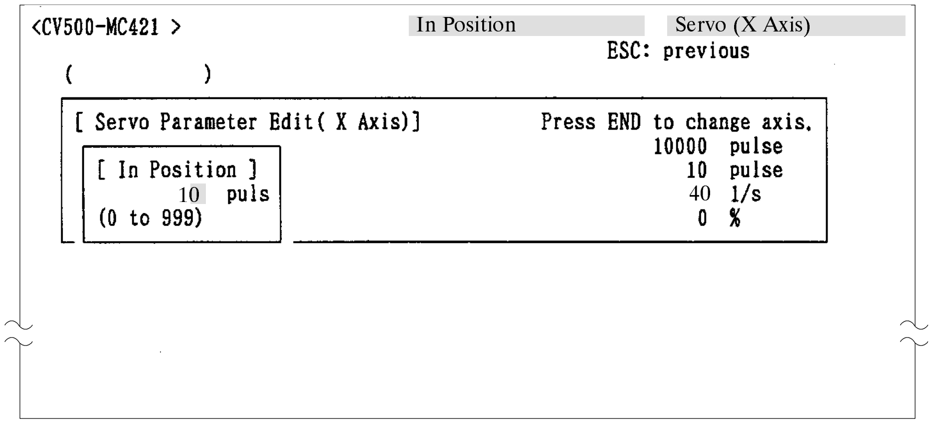
1, 2, 3... 1. Select "E:Maximum Jog Feedrate" from the Feedrate Parameter Edit screen.

```
<CV500-MC421 >                      Max Jog Feedrate        X Feedrate
                                                ESC: previous
  (          )
 [ X Feedrate Parameter Edit      ]        Press END to change axis.
  [ Max Jog Feedrate ]                        2048.00 mm/s
                 40960  mm/s                  2048.00 mm/s
  (0.01 to 4096.00)                            204.80 mm/s
                                                20.48 mm/s
                                           ?????????? mm/s
```

2. Enter "204.8" and press the Return Key.

8-8-14 In-position Value

1, 2, 3... 1. Select “B:In-position” from the Servo Parameter Edit screen.



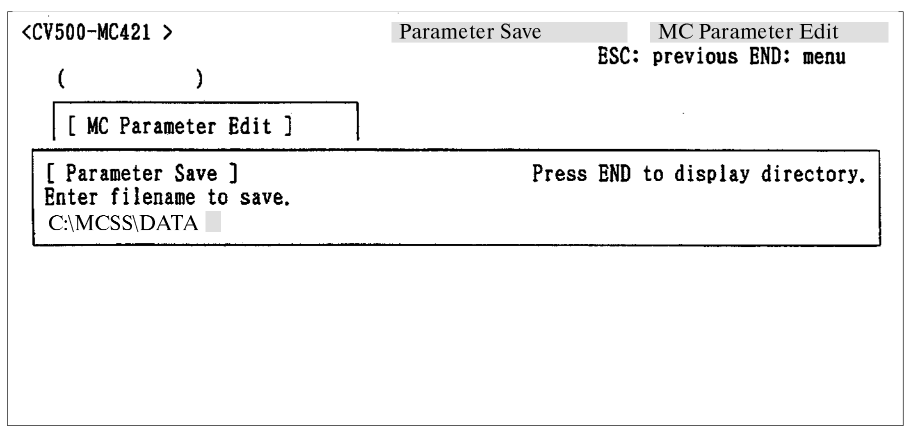
2. Enter “5” and press the Return Key.

8-9 Saving Parameters

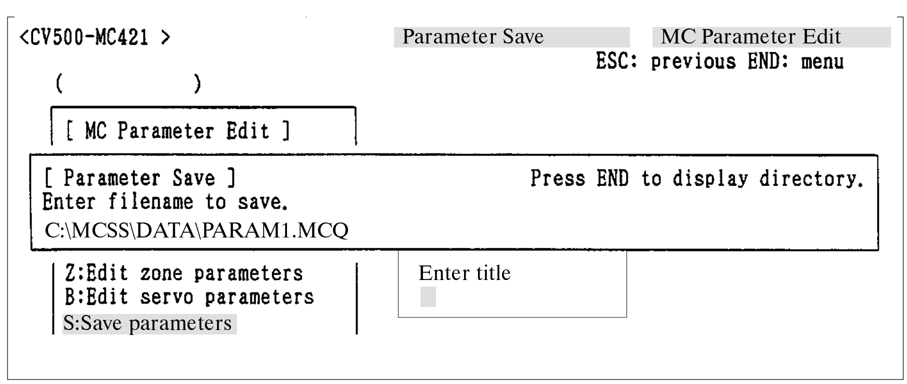
This section explains how to save the parameters to a data disk after they have been set.

Filenames must be eight characters or less. (The entire path name can be up to 74 characters.) Comments must be 30 characters or less.

1, 2, 3... 1. Select “S:Save Parameter” from the MC Parameter Edit Menu.

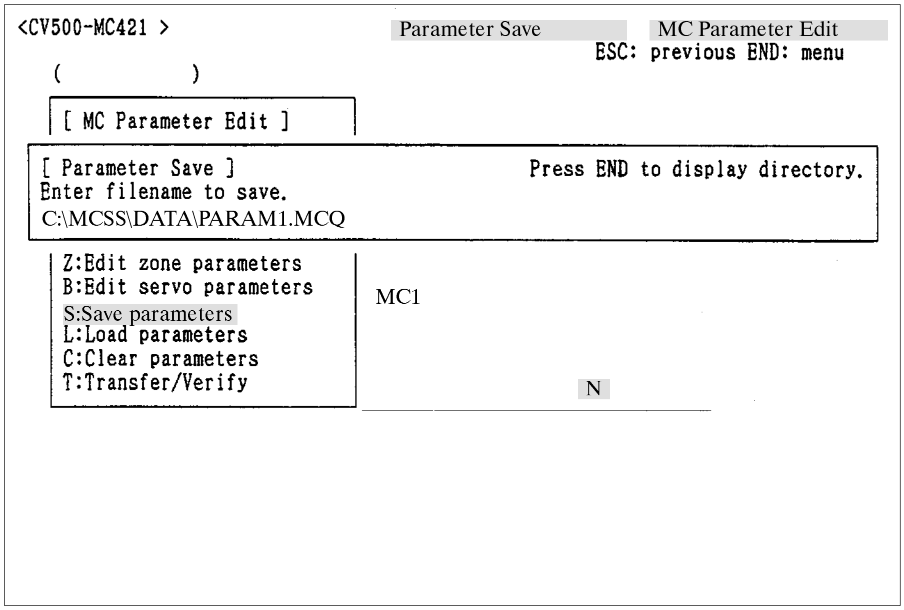


2. Enter the filename (“PARAM1,” for this example) and then press the Return Key.



3. Enter a comment ("MC1," for this example) and then press the Return Key.

   If the same filename already exists, a message will be displayed for confirmation. To overwrite the existing file, input "Y" and press the Return Key. To cancel the save operation, just press the Return Key.

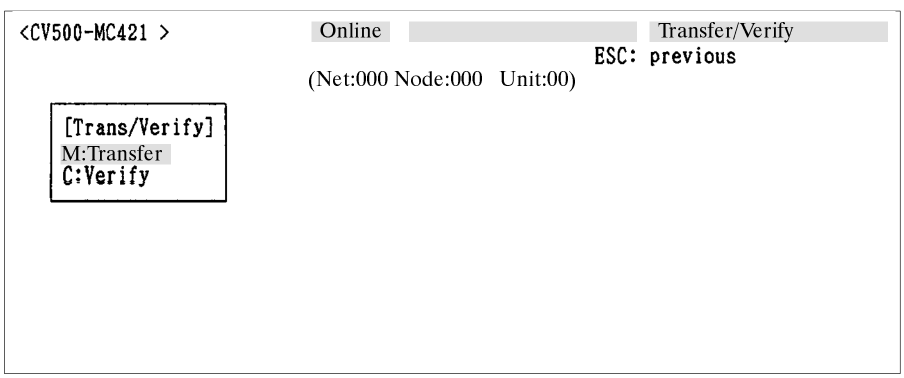


8-10 Transferring Parameters

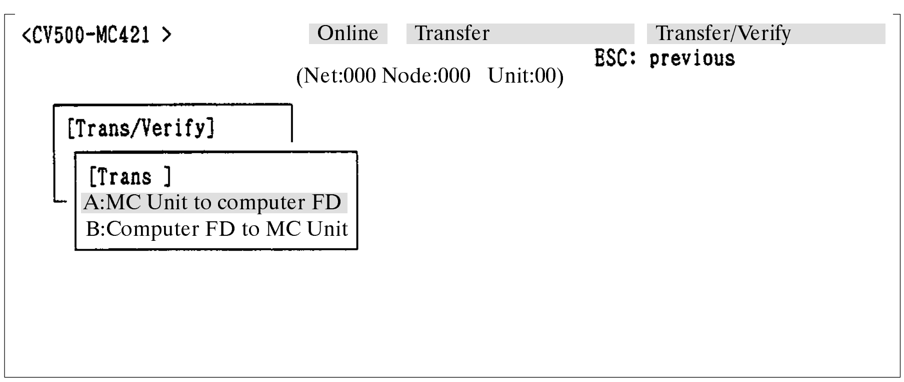
This section explains how to transfer the parameters to the MC Unit once they have been set.

1, 2, 3...

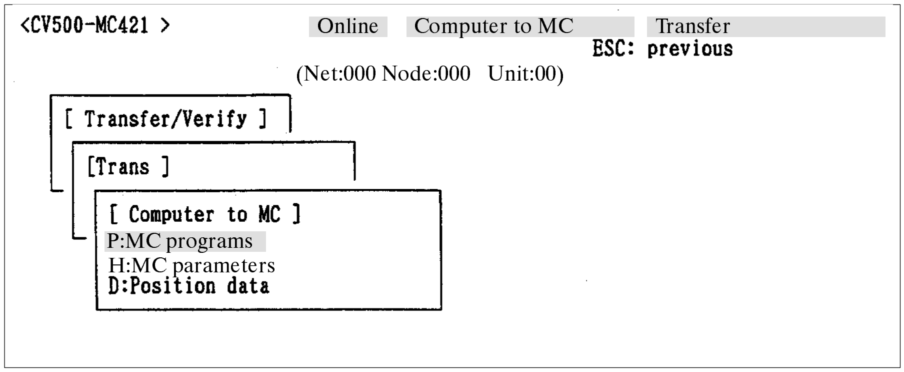
1. Select "T:Transfer/Verify" from the Main Menu.



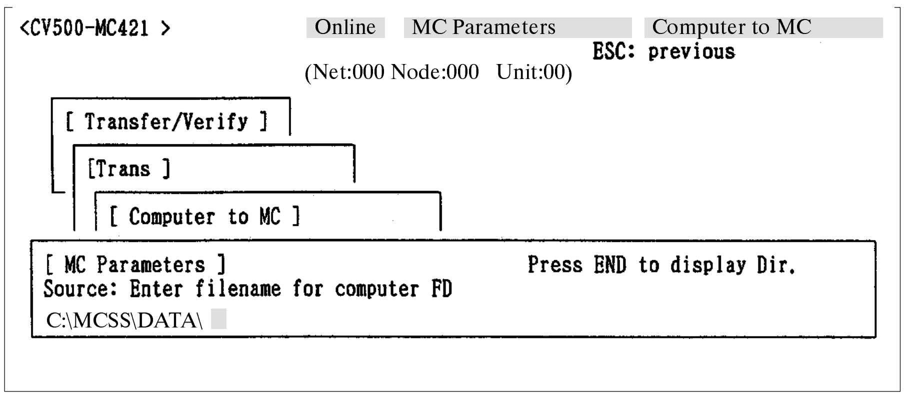
2. Select "M:Transfer" from the Transfer/Verify Menu.



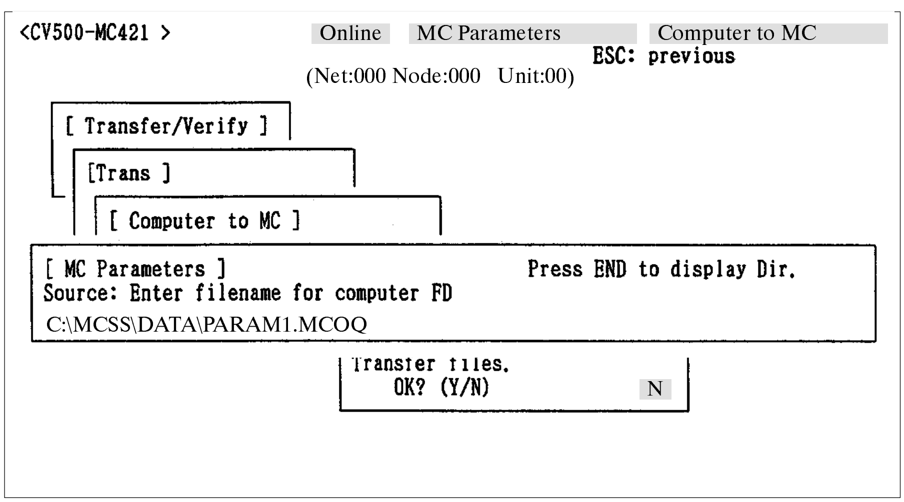
3. Select “B:Computer FD to MC Unit” from the Transfer Menu.



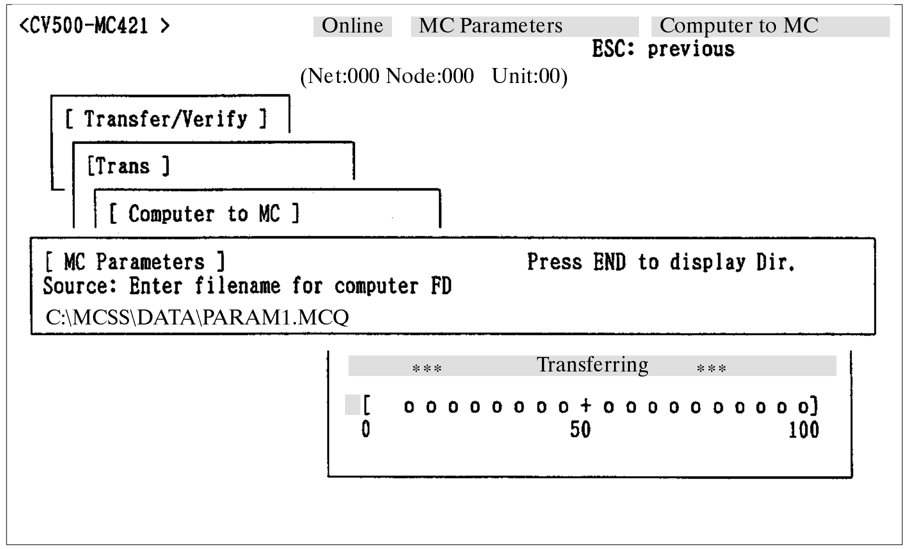
4. Select “H:MC Parameters” from the Computer FD to MC Menu.



5. Enter the name of the file that is to be transferred (“PARAM1,” for this example). If the file already exists, the filename can be selected from a list by first pressing the End Key to display a list of existing files.



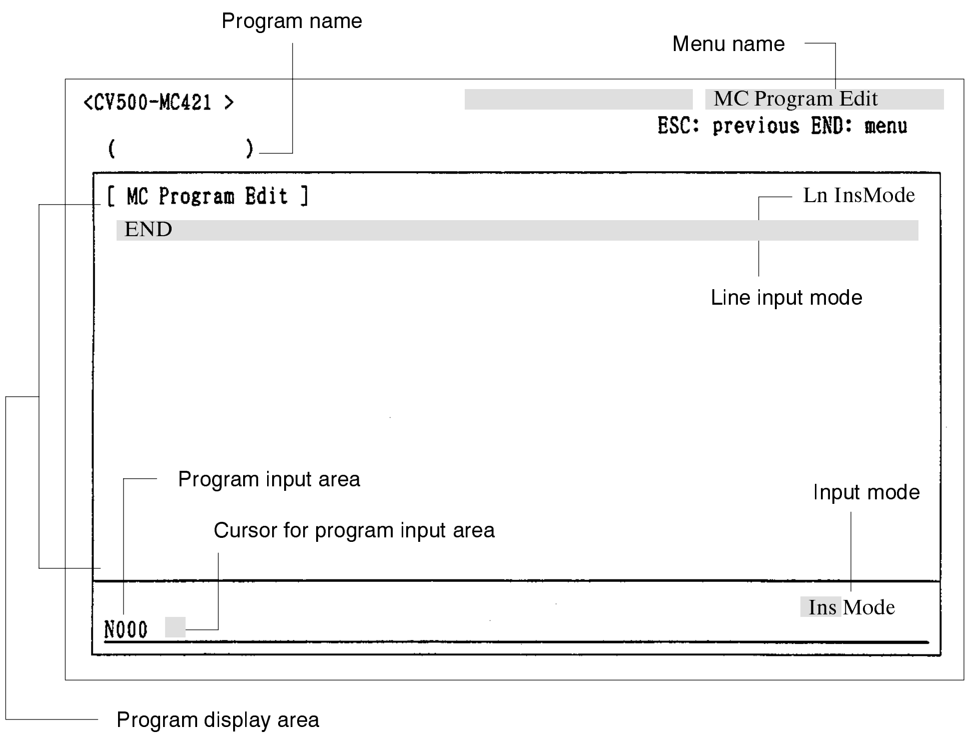
6. A message will be displayed at the bottom of the screen for confirmation. To transfer the file, enter "Y" and press the Return Key. To cancel the transfer operation, just press the Return Key. The transfer status will be displayed while the file is being transferred.



8-11 Creating MC Programs

This section provides a simple explanation of how to use the MCSS to create an MC program. For a more detailed explanation, refer to the *MC Support Software Operation Manual*.

1, 2, 3... 1. Select "P:Edit MC Programs" from the Main Menu. The MC Program Edit screen will be displayed.



2. Check to make sure that the line input mode is "line insert," and that the input mode is "insert."

   If the line input mode is "line overwrite," press the F4 (Insert) Key. If the input mode is "overwrite," press the Insert Key.

3. Input the program, pressing the Return Key at the end of each line.

MC Program

The program to be created is shown below.

```
N000    P001 XYZ
*001    SAMPLE PROGRAM
N002    G04 5
N003    G26 XYZ
N004    G11
N005    G01 X400 Y300 Z–30 F30
N006    G01 Z–100 F5
N007    G01 Z–30
N008    G01 X300 F10
N009    G01 Z–100 F5
N010    G01 Z–30
N011    G01 Y400 F10
N012    G01 Z–100 F5
N013    G01 Z–30
N014    G26 XYZ
N015    G79
```

Program Contents

N000 States that the program number is "P001," and that the X, Y, and Z axes are to be used.

*001 This is a comment.

N002 Waits five seconds.

N003 Returns the X, Y, and Z axes to their reference origins.

N004 Selects the stop mode for positioning.

N005 Moves to X400 Y300 Z–30 at speed 30.

N006 Lowers the Z axis to Z–100 at speed 5.

N007 Raises the Z axis to Z–30 at speed 5.

N008 Moves the X axis to X300 at speed 10.

N009 Lowers the Z axis to Z–100 at speed 5.

N010 Raises the Z axis to Z–30 at speed 5.

N011 Moves the Y axis to Y400 at speed 10.

N012 Lowers the Z axis to Z–100 at speed 5.

N013 Raises the Z axis to Z–30 at speed 5.

N014 Returns the X, Y, and Z axes to their reference origins.

N015 Ends the program.

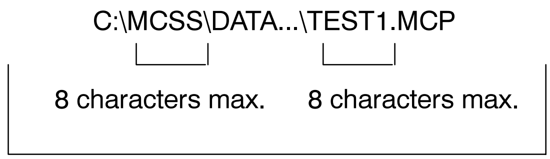
8-12 Saving MC Programs

This section explains how to save an MC program to the data disk once it has been created.

Filenames, Directory Names, and Comments

A filename must be entered when a program is saved. Filenames and directory names can contain up to eight characters each. Letters of the alphabet are converted to upper case even if they are entered in lower case. The maximum path length is 78 characters.

Example



The maximum number of characters that can be used for a comment is 30.

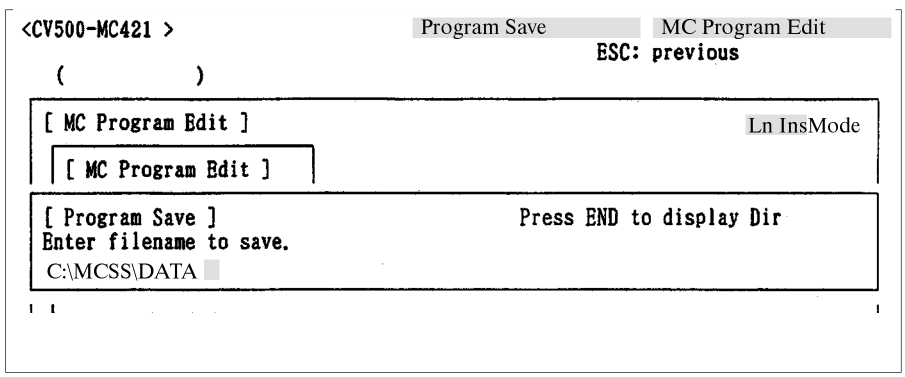
Floppy Disks

Floppy disks used as data disks must be initialized and write-enabled. There must also be sufficient capacity available for the amount of data that is to be saved.

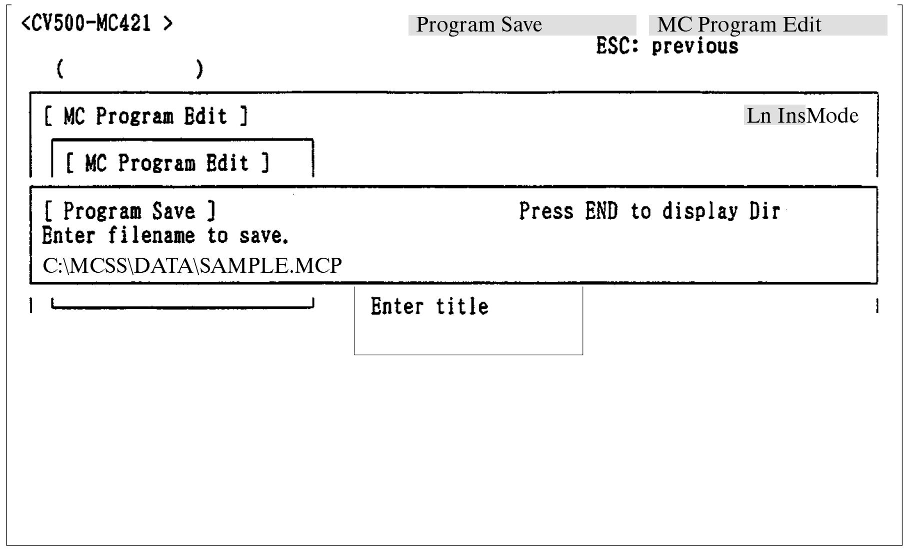
Saving a Program

1, 2, 3...

1. Press the End Key to display the MC Program Edit Menu. Then select “S:Save Programs.”

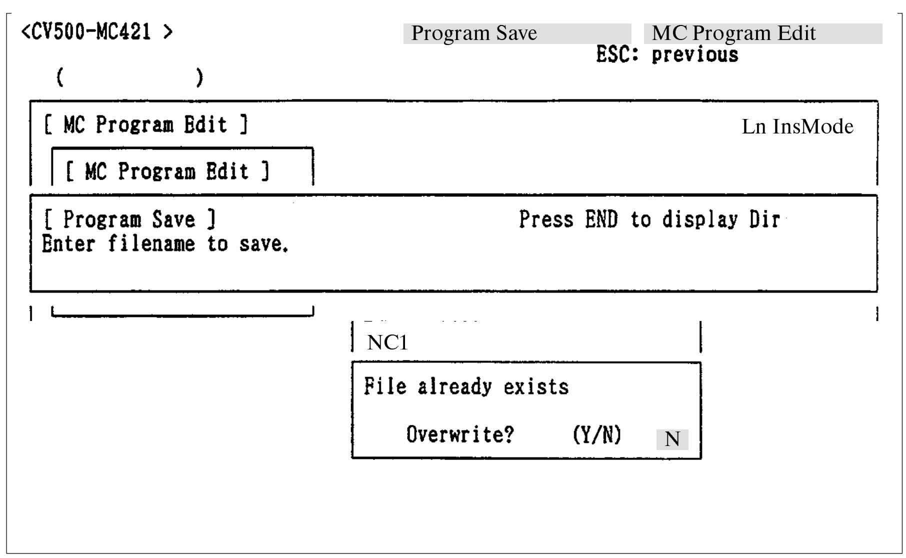


2. Enter the name of the file (SAMPLE, for this example) that is to be saved, and press the Return Key. If the file already exists, the filename can be selected from a list by first pressing the End Key to display a list of existing files.



3. Enter a comment ("NC1," for this example). To omit the comment, just press the Return Key.

   If the same filename already exists, a message will be displayed for confirmation. To save the new file with that name, input "Y" and press the Return Key. To cancel the save operation, just press the Return Key.



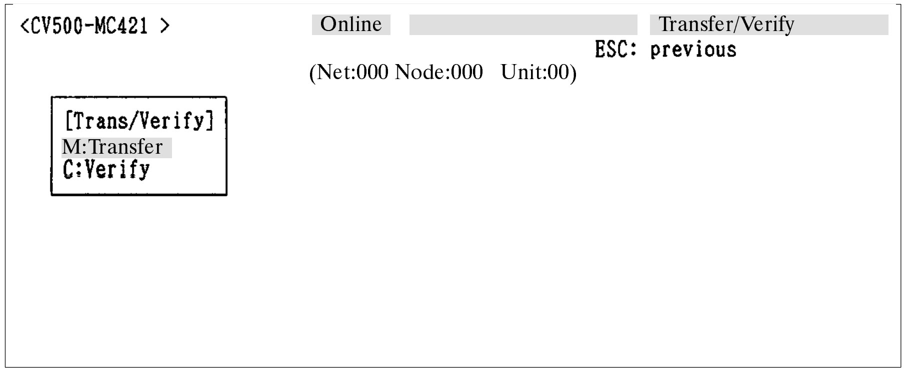
The time required to save the file will depend on the capacity of the program.

8-13 Transferring MC Programs

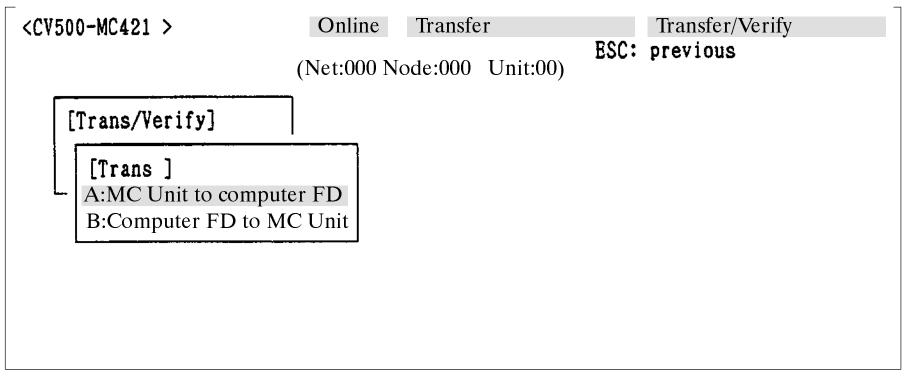
This section explains how to transfer MC programs to the MC Unit.

1, 2, 3...

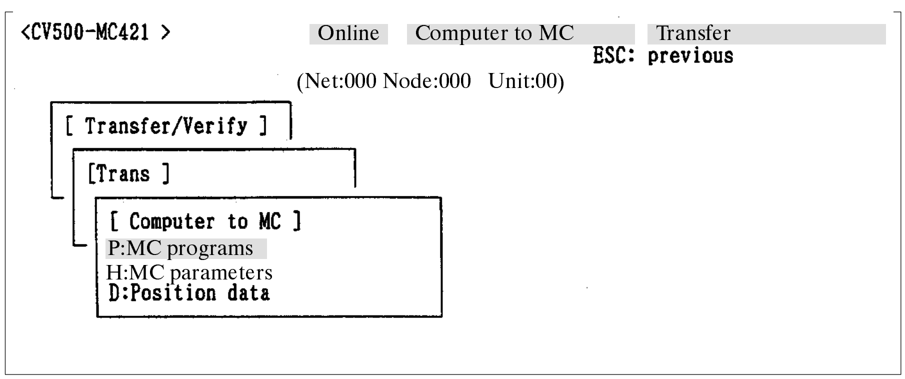
1. Select “T:Transfer/Verify” from the Main Menu.



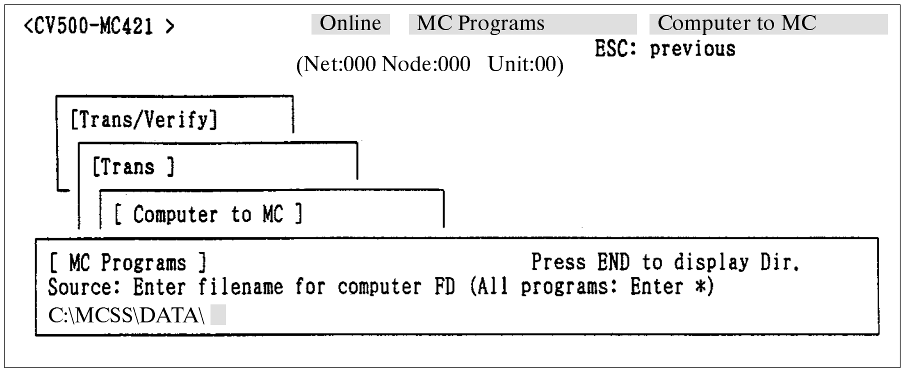
2. Select “M:Transfer” from the Transfer/Verify Menu.



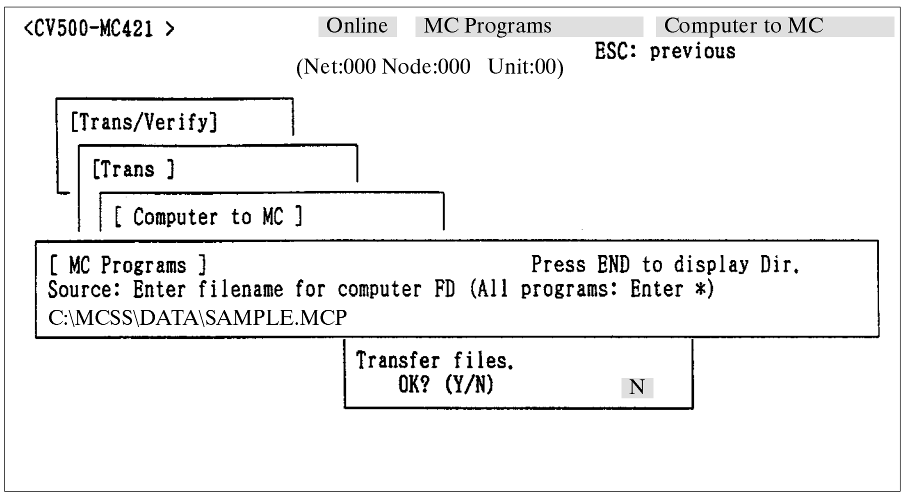
3. Select “B:Computer FD to MC Unit” from the Transfer Menu.



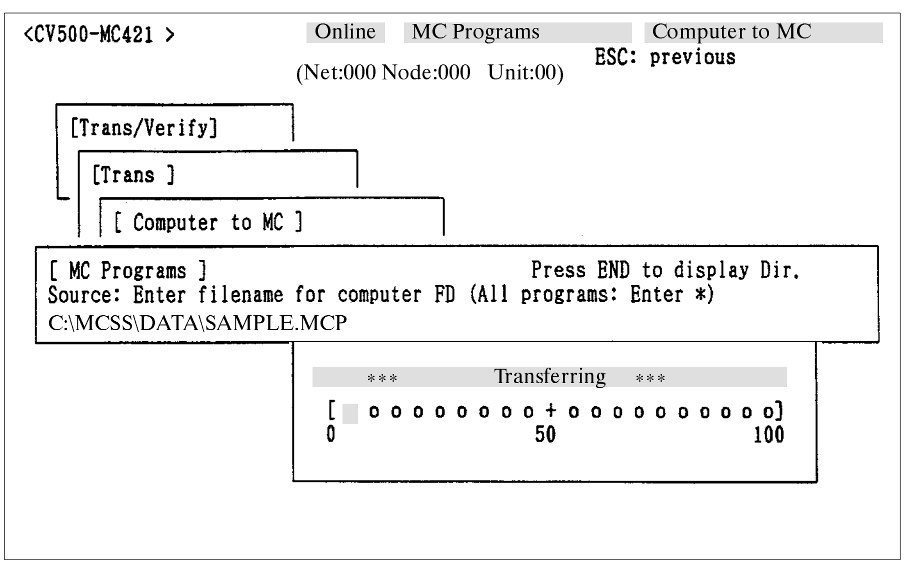
4. Select “P:MC Programs” from the Computer FD to MC Unit Menu.



5. Enter the name of the file that is to be transferred (“SAMPLE,” for this example). If the file already exists, the filename can be selected from a list by first pressing the End Key to display a list of existing files.



6. A message will be displayed at the bottom of the screen for confirmation. To transfer the file, enter “Y” and press the Return Key. To cancel the transfer operation, just press the Return Key. The transfer status will be displayed while the file is being transferred.



8-14 Creating Ladder-diagram Programs

This section explains how to create ladder-diagram programs using the CVSS. These programs can be used in manual mode to execute servo-locks, origin searches, and jogging, and in automatic mode to execute MC programs.

After a ladder-diagram program has been created, it must be transferred to the PC.

Note For more details regarding ladder-diagram programs, refer to the *CV Support Software Operation Manual*.

Ladder-diagram Program Notation

Bit names with pointed brackets < > indicate interface output bits. Bit names with parentheses ( ) indicate interface input bits.

Note For more details regarding interface bits, refer to the *MC Unit Operation Manual: Details*.

Word Allocation

The ladder-diagram programs explained here are for MC Unit #00. Therefore the interface area is allocated 25 words beginning with CIO 1500, and 100 words beginning with DM 2000. CIO 006400 to CIO 006407 are used as work bits.

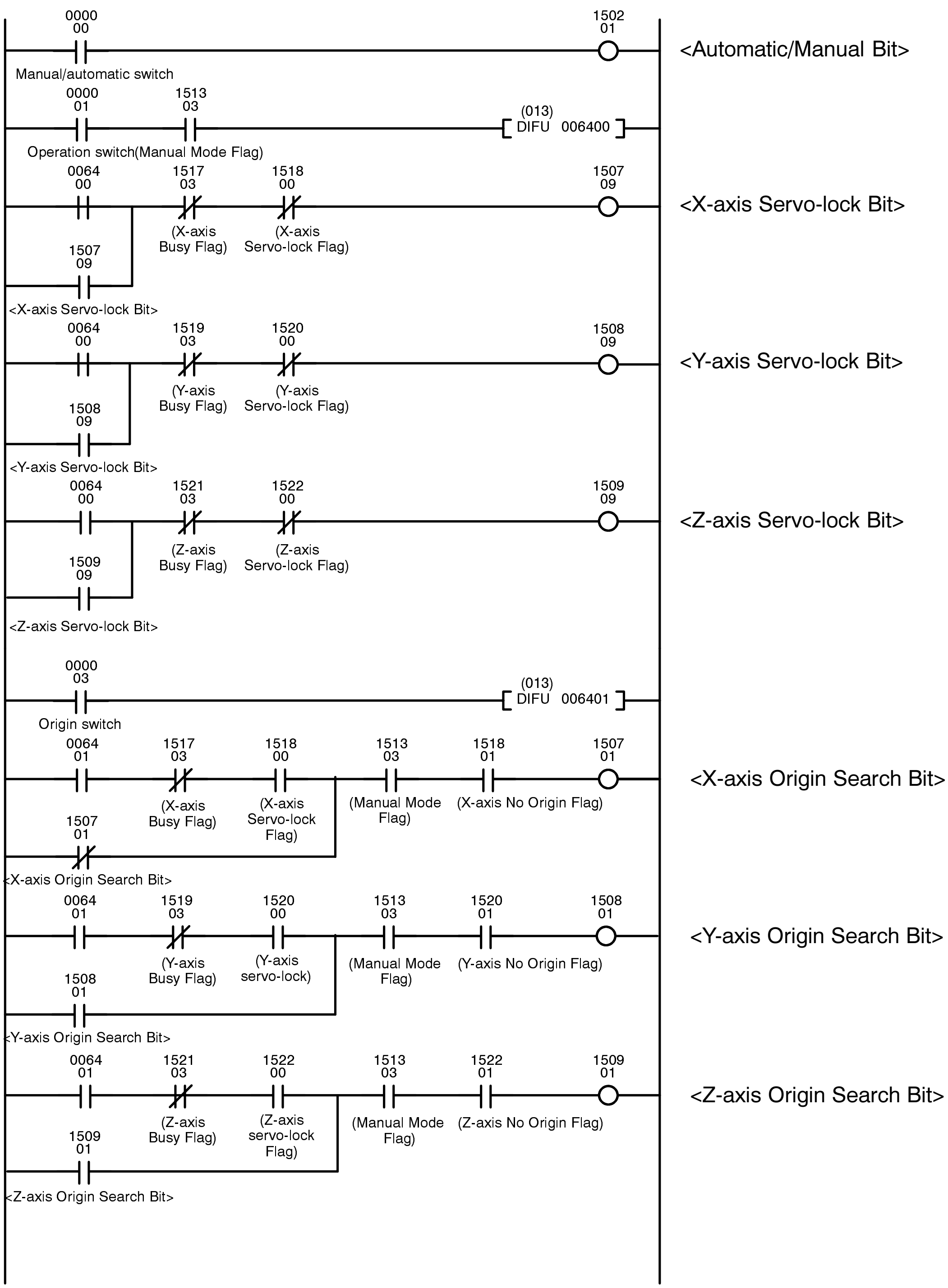
Procedure

For detailed instructions regarding any of these procedures, refer to the relevant sections in the *CV Support Software Operation Manual*.

1, 2, 3...

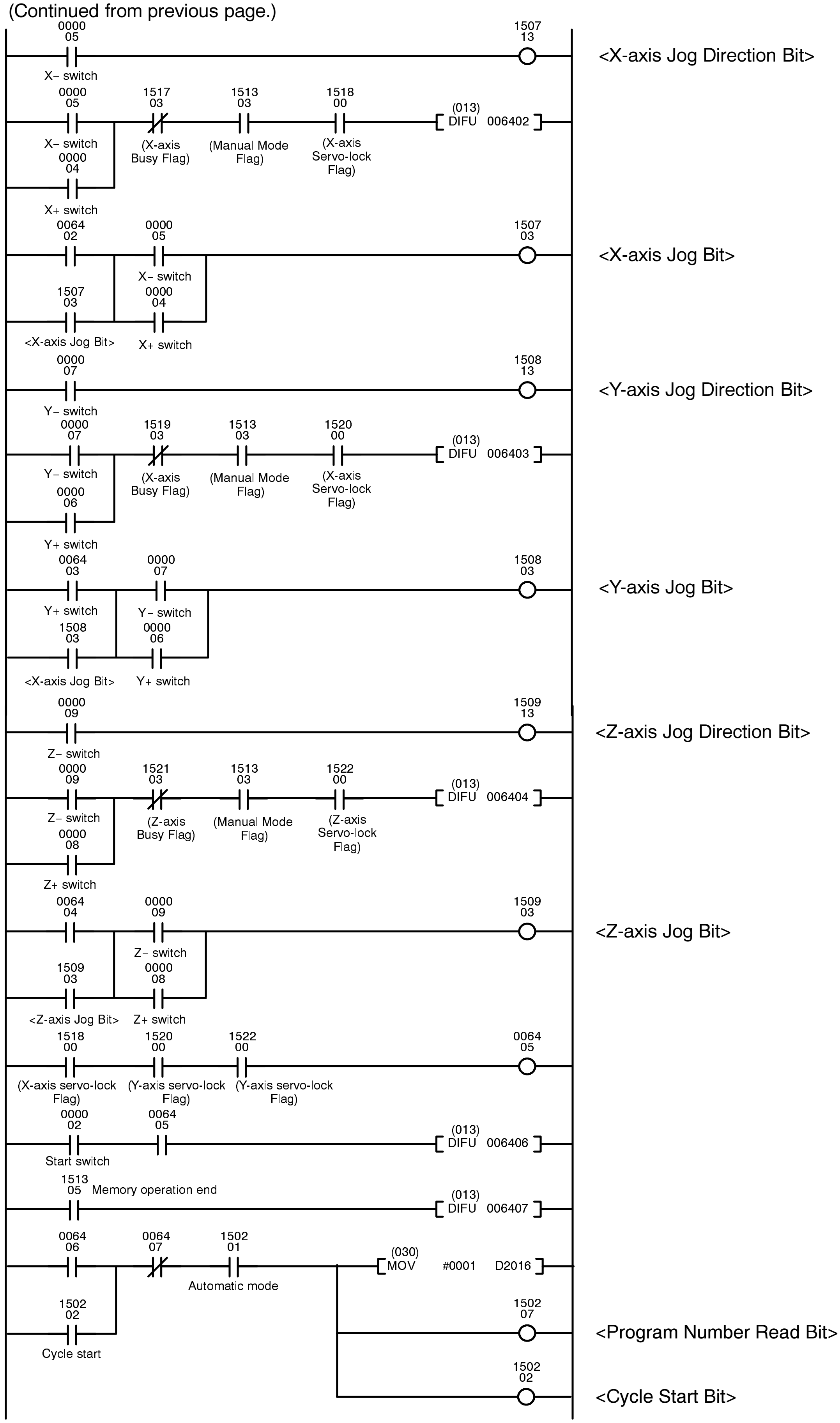
1. Create the ladder-diagram program for executing the MC program.
2. Save the ladder-diagram program to a data disk.
3. Switch the CVSS to the online mode.
4. Transfer the ladder-diagram program to the PC.
5. Monitor the MC program by switching the PC to MONITOR mode and then executing the MC program.

Ladder-diagram Program Input the ladder-diagram program shown below.

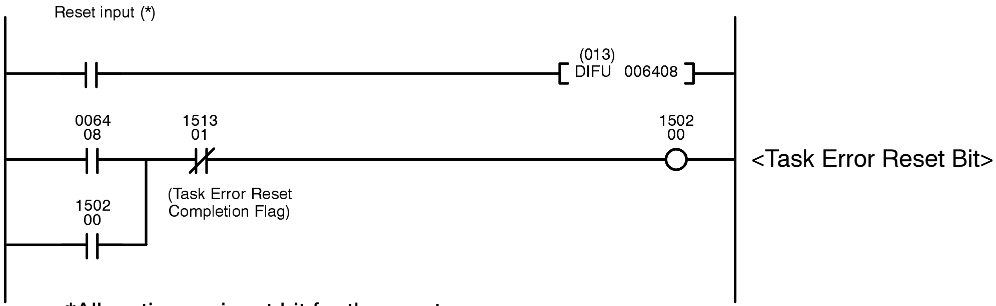


(Continued on next page.)

(Continued from previous page.)



Note A task error reset is required for when an error occurs during trial operation. Add an error reset like the one shown below to the program.



*Allocation an input bit for the reset.

Interface bit CIO 151300 is used in an NC condition to break the self-holding bit (CIO 150200) when the error is reset.

8-15 Checking Operation from the MCSS

After the wiring has been completed, the parameters have been set, and the MC program and the ladder-diagram program have been input and transferred, check the operation by means of the following procedure. Each of the steps in the procedure is explained in more detail on subsequent pages.

1, 2, 3...

1. Turn on the power to the Power Supply Unit, the servodriver, and the computer.
2. Start MC monitoring.
3. Set the automatic/manual switch to manual, and then press the Program Execution Key.
4. Press the Origin Search Key.
5. Press the Jog Key.
6. Set the automatic/manual switch to automatic, and press the Start Key.

Note When checking operation using a ladder-diagram program, be sure that the Teaching Box mode is set to T. BOX LIMITED. Commands from the PC cannot be received in T. BOX RESERVED or T. BOX ENABLED mode. For details, refer to *8-16 Checking Operation from the Teaching Box*.

Turning On the Power Supply

When power is turned on to the MC Unit, an error will not be generated if the servodriver alarm input is ON from the beginning. An error will be generated if it the servodriver alarm turns ON, or if a command is executed accompanying axis operation with the alarm input turned ON. This allows the power-up sequence of the MC Unit and the servodriver to be easily coordinated.

The power must be turned on to the Programmable Controller when carrying out online operations from the computer. If it is not on, an error will be generated.

Starting MC Monitoring

MC monitoring is used to check the present position for each axis. To use MC monitoring, start up the MCSS and select “M:MC Monitoring” from the Main Menu. (For more details regarding MC monitoring, refer to the *MCSS Operation Manual*.

```
<CV500-MC421 >            Online                       MC Monitoring
                                              ESC: previous END: menu
                        (Net:000 Node:000   Unit:00)

 [ PV Display ]
     <Reference Coordinate>             <Workpiece Coordinate>
 X Axis                 0.01 mm     X Axis                 0.01 mm
 Y Axis                 0.00 mm     Y Axis                 0.00 mm
 Z Axis                −0.02 mm     Z Axis                −0.02 mm
 U Axis                 0.00 mm     U Axis                 0.00 mm
     <Reference Coordinate>                <Deviation Count>
 X Axis                  1 puls     X Axis                 0 puls
 Y Axis                  0 puls     Y Axis                -1 puls
 Z Axis                 −2 puls     Z Axis                 0 puls
 U Axis                  0 puls     U Axis                 1 puls
      <Workpiece Origin Shift>
 X Axis                 0.00 mm
 Y Axis                 0.00 mm
 Z Axis                 0.00 mm
 U Axis                 0.00 mm
```

Checking Servo-lock

The servos for axes X, Y, and Z will be locked when the Program Execution Key is pressed while in manual mode. Power will be supplied for each axis, and the CW and CCW indicators on the MC Unit will light for the X, Y, and Z axes.

The present position and the number of error counter pulses are changed in the vicinity of “0” on the MC Monitoring Screen.

```
<CV500-MC421 >            Online                       MC Monitoring
                                              ESC: previous END: menu
                        (Net:000 Node:000   Unit:00)

 [ PV Display ]
     <Reference Coordinate>             <Workpiece Coordinate>
 X Axis                −0.01 mm     X Axis                −0.01 mm
 Y Axis                 0.02 mm     Y Axis                 0.02 mm
 Z Axis                 0.01 mm     Z Axis                 0.01 mm
 U Axis                 0.00 mm     U Axis                 0.00 mm
     <Reference Coordinate>                <Deviation Count>
 X Axis                 −1 puls     X Axis                −3 puls
 Y Axis                  2 puls     Y Axis                 1 puls
 Z Axis                  1 puls     Z Axis                −2 puls
 U Axis                  0 puls     U Axis                 0 puls
```

Checking Origin Search

When the Origin Search Key is pressed while in manual mode, an origin search is executed along the X, Y, and Z axes. Positioning moves to the origin at each axis. When the origin search is completed, the reference coordinate system’s present position will be near “0” on the MC Monitoring Screen.

Checking the Jog Operation

Positioning will move in the positive direction along the X axis, at jog speed, while the +X Key is being pressed, and the X-axis present position will be incremented on the MC Monitoring Screen. Likewise, positioning will move in the negative direction along the X axis, at jog speed, while the –X Key is being pressed, and the X-axis present position will be decremented on the MC Monitoring Screen. The jog operation can be checked in the same way for the Y and Z axes.

```
<CV500-MC421 >            Online                          MC Monitoring
                                                  ESC: previous END: menu
                         (Net:000 Node:000   Unit:00)
 [ PV Display ]
     <Reference Coordinate>                  <Workpiece Coordinate>
 X Axis              136.36 mm       X Axis               136.36 mm
 Y Axis                0.00 mm       Y Axis                 0.00 mm
 Z Axis                0.01 mm       Z Axis                 0.01 mm
 U Axis                0.00 mm       U Axis                 0.00 mm
     <Reference Coordinate>                   <Deviation Count>
 X Axis               13636 puls     X Axis                   –4 puls
 Y Axis                   0 puls     Y Axis                   –1 puls
 Z Axis                   1 puls     Z Axis                   –0 puls
 U Axis                   0 puls     U Axis                    0 puls
```

Note Be sure that the present position does not exceed the software limit that has been set. If that limit is exceeded, either a CW or CCW direction software limit error will be generated.

Checking MC Program Operation

The MC program can be checked by setting the automatic/manual switch to automatic and then pressing the Start Key to execute the MC program. Follow the procedure shown below to display the MC program on the MC Monitoring screen and check it.

1, 2, 3...

1. Set the automatic/manual switch to automatic.
2. Press the Start Key.
3. Press the End Key from the MC Monitoring screen to display the Monitoring Menu.

```
<CV500-MC421 >            Online                          MC Monitoring
                                                  ESC:  previous
                         (Net:000 Node:000   Unit:00)
 [ PV Display ]
                                          <Workpiece Coordinate>
  [ MC Monitoring ]                  X Axis                 0.00 mm
  P:Display MC programs              Y Axis                 0.00 mm
  F:Display FAL status               Z Axis                 0.00 mm
  I:Display error log                U Axis                 0.00 mm
  S:Display MC I/O status                 <Deviation Count>
  N:Dest network addr                X Axis                    0 puls
```

Note If steps 3 through 5 are executed before the Start Key is pressed, a message will be displayed at the bottom left of the screen indicating that the program has not been executed. Monitoring can be carried out if program execution is begun by pressing the Start key.

4. Select “P:Display MC Programs” from the MC Monitoring Menu.

```
<CV500-MC421 >              Online   MC Program Display   MC Monitoring
                       (Net:000 Node:000   Unit:00)   ESC: previous

 [ PV Display ]
                                       <Workpiece Coordinate>
   [ MC Monitoring ]              X Axis                    0.00 mm
                                  Y Axis                    0.00 mm
     [ MC Program Display ]       Z Axis                    0.00 mm
      1:Display Task 1            U Axis                    0.00 mm
      2:Display Task 2                    <Deviation Count>
      3:Display Task 3            X Axis                       0 puls
      4:Display Task 4            Y Axis                      -1 puls
                                  Z Axis                       0 puls
```

5. Select “1:Display Task 1” from the MC Program Display Menu. The program will be displayed on the screen, and the block being executed will be shown in reverse video.

```
<CV500-MC421 >              Online   MC Program Display   MC Monitoring
                                           ESC: previous END: menu
 (Tsk1,P00 )             (Net:000 Node:000   Unit:00)
 [ MC Program Display ]
  N000 P001 XYZ
  *001 SAMPLE PROGRAM
  N002 G04  5 :
  N003 G26 x Y Z
  N004 G11
  N005 G01 X400 Y300 Z-30 F30
  N006 G01 Z-100 F5
  N007 G01 Z-30
  N008 G01 X300 F10
  N009 G01 Z-100 F5
  N010 G01 Z-30
  N011 G01 Y400 F10
  N011 G01 Z-100 F5
 [ PV  Display ]
 X Axis   * * * * * * * * * * *     Y Axis   * * * * * * * * * * *
 Z Axis   * * * * * * * * * * *     U Axis   * * * * * * * * * * *
 Task status   [ 1 *  2 *  3 -  4 -   ]          Running  *  Stopped
```

6. Press the F1 Key to display the present position at the bottom of the screen. Check to see that positioning is proceeding to the coordinates set by the MC program that is being executed.

```
 [ PV  Display ]
 X Axis              −0.07 mm      Y Axis                    0.02 mm
 Z Axis               0.01 mm      U Axis                    0.00 mm
 Task status   [ 1 *  2 *  3 -  4 -   ]          Running  *  Stopped
```

8-16 Checking Operation from the Teaching Box

This section explains how to use the Teaching Box to execute the MC program and the jog operation. It explains cycle run, in which the specified program is executed from the beginning, and single-block run, in which the specified program is executed one block at a time.

When the Teaching Box is connected to the MC Unit, it first goes into T. BOX LIMITED mode, in which only monitoring is possible. In order to execute the cycle run, single-block run, or jogging operation, the mode must be switched to either T. BOX RESERVED or T. BOX ENABLED. The operation mode can only be changed from the Teaching Box itself.

If servo-lock and origin search have not been executed using the switch box, execute the origin search from the Teaching Box after setting the servo-lock status before executing the MC program and jog operation.

Changing the Mode to T. BOX RESERVED

1, 2, 3...

1. Press the EXT Key.

```
EXT|                        1/2
 1.CHANGE MODE
 2.SERVO LOCK
 3.SERVO FREE
```

2. Press “1.”

```
EXT|–CHG MODE–
    | T.BOX
    | LIMITED
    |SWITCH:↑,↓
```

3. Press the Down Key. (Pressing the Up and Down Keys switches among the three modes.)

```
EXT|–CHG MODE–
    | T.BOX
    | RESERVED
    |SWITCH:↑,↓
```

Executing Servo-lock

1, 2, 3...

1. Select “2. SERVO LOCK” from the Extension Mode Menu.

```
EXT|–SERVO LOCK–
  | 0.ALL AXES
  | 1.X   3.Z
  | 2.Y   4.U
```

2. Select the axis or axes.

```
EXT|–SERVO LOCK–
    | OK?
ALL|
    | YES / NO
```

The specified axis/axes are shown here.

3. Press the YES Key. Servo-lock will be executed, and then the following screen will be displayed.

```
EXT|–SERVO LOCK–
   |
ALL| COMPLETE!
   |
```

Executing Origin Search for All Axes

1, 2, 3...

1. Press the ORIG SRCH Key.

```
ORG|0.ALL AXES
   |1.X  3.Z
   |2.Y  4.U
   |
```

2. Select the axis or axes.

```
ORG|
   | OK?
ALL|
   | YES / NO
```

The specified axis/axes are shown here.

3. Press the YES Key. An origin search will be started and the present position monitoring screen will be displayed.

```
ORG|X*           12.00
   |Y*           12.32
ALL|Z*           12.34
   |U*            0.00
```

The asterisk (*) indicates that the origin is not defined.

The present position is refreshed for moving axes.

The origin search operation is completed when the axes have been moved to the reference coordinate system origin and stopped. The asterisks disappear at that point, and the present position is shown as “0.”

When the origin search operation has been completed for the specified axes, the following screen will be displayed.

```
ORG|
   |
ALL| COMPLETE!
   |
```

4. Press the CLR Key. The axis selection screen in step one of this procedure will return.

Note The CLR Key is not effective during the origin search operation. To halt the origin search operation in progress, press the PAUSE Key.

Clearing Errors

When an error occurs at the MC Unit or the servodriver, clear it by means of the following procedures.

Resetting MC Unit Errors

1, 2, 3...

1. Press the ERR CLR Key.

```
[RT]

 1.MC UNIT
 2.DRIVER
```

2. Press "1."

```
[RT]                MC UNIT

 RESET ERROR ?
        YES / NO
```

3. Press the YES Key. The error will be reset for the task in which the error occurred.

```
[RT]                MC UNIT

 ERROR RESET!
```

Resetting Servodriver Errors

1, 2, 3...

1. Select "2.DRIVER" from the MC Unit's Error Reset Procedure 1 Menu.

```
[RT]     DRIVER
    0.ALL AXES
    1.X   3.Z
    2.Y   4.U
```

2. Press a number from 0 to 4 to select the number of axes to be reset.

```
[RT]                 DRIVER
 <ALL AXES>
 RESET ERROR ?
        YES / NO
```

The specified axis or axes are displayed here.

0: All axes 1: X axis
2: Y axis 3: Z axis
4: U axis

Executing the Jog Operation

1, 2, 3...

1. Press the JOG Key. The Unit will go into jog mode and the following screen will be displayed.

```
JOG|X                  0.00
   |Y                  0.00
   |Z                  0.00
050|U*                 0.00
```

The override values are displayed here. The default value is 50%. Press the Up and Down Keys to set the value from 10% to 100% in increments of 10%.

2. To execute jogging along the X, Y, or Z axes, press any of the following six jog feed keys: +X, +Y, +Z, –X, –Y, –Z. The specified axis will move in the specified direction, and the present position will be refreshed on the screen.

Executing Cycle Run

1, 2, 3...

1. Press the PROG EXEC Key.

```
RUN|1.CYCLE RUN
   |2.SINGLE RUN
   |
   |
```

2. Select “1. CYCLE RUN” from the menu.

```
RUN|–CYCLE RUN–
CYC|PROGRM P000
TK1|BLOCK  N000
   | RUN: START
```

Task 1 is displayed.

3. Specify the program number. In this case, the program number is 001, so just input “1.”

```
RUN|–CYCLE RUN–
CYC|PROGRM P001
TK1|BLOCK  N000
   | RUN: START
```

4. Press the START Key.

```
RUN|STATUS RUN
CYC|PROGRM P050
TK1|BLOCK  N010
   |
```

When the PAUSE Key is pressed, program execution is paused and the previous screen is restored. Program execution is resumed by pressing the START Key. The program cannot be executed while an error is in effect.

5. Press the CLR Key after the program has been completed. The menu for selecting the operation mode will be restored. (The CLR Key is not effective during program execution.)

Executing Single-block Run

1, 2, 3...

1. Select “2.SINGLE RUN” from the menu.

```
RUN|–SINGLE RUN–
SIN|PROGRM P000
TK1|BLOCK  N000
   | RUN: START
```

Task 1 is displayed.

2. Specify the program number. In this case, the program number is 001, so just input “1.” If the program number is already set to 001 after the execution of cycle run, there is no need to enter anything here.

```
RUN|STATUS RUN
SIN|PROGRM P001
TK1|BLOCK  N002
   |
```

3. Press the START Key. The first block of program no. 001 will be executed. In this case, since the first block is a comment, block N002 will be executed.

```
RUN|STATUS RUN
SIN|PROGRM P001
TK1|BLOCK  N002
   |
```

To halt the operation in progress, press the PAUSE Key.

4. When execution of the first block has been completed, “pause” status will go into effect and program execution will be stopped. The number of the next block to be executed will be displayed.

```
RUN|STATUS PAUSE
SIN|PROGRM P001
TK1|BLOCK  N003
   | RUN: START
```

Press the PROG EXEC Key to execute the next block. Press the CLR Key to return to the program number input screen. Then press the CLR Key again to return to the operation mode menu.

Glossary

absolute position A position given in respect to the origin rather than in respect to the present position.

acceleration/deceleration curve Curves which determine the rate of acceleration to the maximum feed rate and the rate of deceleration from the maximum feed rate.

auxiliary bit A bit in the Auxiliary Area.

Backplane A base to which Units are mounted to form a Rack. Backplanes provide a series of connectors for these Units along with buses to connect them to the CPU and other Units and wiring to connect them to the Power Supply Unit. Backplanes also provide connectors used to connect them to other Backplanes.

basic instruction A fundamental instruction used in a ladder diagram. See *advanced instruction*.

baud rate The data transmission speed between two devices in a system measured in bits per second.

BCD Short for binary-coded decimal.

binary A number system where all numbers are expressed in base 2, i.e., numbers are written using only 0's and 1's. Each group of four binary bits is equivalent to one hexadecimal digit. Binary data in memory is thus often expressed in hexadecimal for convenience.

bit The smallest piece of information that can be represented on a computer. A bit has the value of either zero or one, corresponding to the electrical signals ON and OFF. A bit represents one binary digit. Some bits at particular addresses are allocated to special purposes, such as holding the status of input from external devices, while other bits are available for general use in programming.

block number Numbers used to distinguish blocks in MC programs. Block numbers are roughly equivalent to program line numbers.

bus A communications path used to pass data between any of the Units connected to it.

channel See *word*.

CIO Area A memory area used to control I/O and to store and manipulate data. CIO Area addresses do not require prefixes.

comment block A program block that contains comments input by the programmer. Comment blocks and program blocks share the same block numbers, but comment blocks begin with an asterisk rather than an "N."

control bit A bit in a memory area that is set either through the program or via a Programming Device to achieve a specific purpose, e.g., a Restart Bit is turned ON and OFF to restart a Unit.

counter A dedicated group of digits or words in memory used to count the number of times a specific process has occurred, or a location in memory accessed through a TC bit and used to count the number of times the status of a bit or an execution condition has changed from OFF to ON.

CPU The name of the Unit in a PC that contains the main CPU and other main PC components. See also *central processing unit*.

CPU Backplane A Backplane used to create a CPU Rack.

CPU Bus Unit A special Unit used with CV-series PCs that mounts to the CPU bus. This connection to the CPU bus enables special data links, data transfers, and processing.

CPU Bus Unit Area A part of the CIO Area allocated to CPU Bus Units. The use of the words and bits in this area is determined by the Unit to which they are allocated.

CPU Rack The main Rack in a building-block PC, the CPU Rack contains the CPU, a Power Supply, and other Units. The CPU Rack, along with the Expansion CPU Rack, provides both an I/O bus and a CPU bus.

CV-mode A form of communications useable only with CV-series PCs. See *C-mode*.

CV-series PC Any of the following PCs: CV500, CV1000, CV2000, or CVM1

CVSS See *CV Support Software*.

CW and CCW Abbreviations for clockwise (CW) and counterclockwise (CCW). CW and CCW are defined for a motor shaft in reference to a viewer facing the shaft on the end of the motor from which the shaft extends from the motor for connection.

CW/CCW limits Limits on the CW and CCW sides of the origin which can be internally set to restrict rotation of the shaft.

data area An area in the PC's memory that is designed to hold a specific type of data.

decimal A number system where numbers are expressed to the base 10. In a PC all data is ultimately stored in binary form, four binary bits are often used to represent one decimal digit, via a system called binary-coded decimal.

dwell time A setting that specifies the period of time during which positioning will stop before execution of the next positioning action.

error counter A counter used to ensure positioning accuracy when positioning via pulse trains. The error counter receives a target position as a specific number of pulses in a pulse train from the Motion Control Unit and outputs analog speed voltages to drive a servomotor accordingly. The specified number of pulses in the error counter is counted down by feedback from an encoder measuring actual motor shaft movement, causing voltage output to stop when the number of pulses equals zero, i.e., when the target position has been reached.

feedback The return of a portion of the output of a circuit or device to its input. It is used in servocontrol systems to help bring actual values closer to target values.

flag A dedicated bit in memory that is set by the system to indicate some type of operating status. Some flags, such as the carry flag, can also be set by the operator or via the program.

gain The increase in signal power produced by an amplifier.

G language A programming language used widely in position control. Program functions are entered simply by entering a "G," a 2-digit numerical code, and adding any needed parameters.

hunting The tendency, in servosystems, to overcompensate when the system's momentum carries it past the target position.

IBM PC/AT or compatible A computer that has similar architecture to, that is logically compatible with, and that can run software designed for an IBM PC/AT computer.

inching Manual feeding wherein positioning is executed one pulse at a time.

incremental position A position given in respect to the present position, rather than in respect to the origin.

initial position The present position when a start command is executed.

in position The range within which the system is determined to be at the target position.

input The signal coming from an external device into the PC. The term input is often used abstractly or collectively to refer to incoming signals.

interpolation The mathematical calculation of missing values based pm known values. The Motion Control Unit uses interpolation when positioning along two or more axes simultaneously. There are three types of interpolation possible: linear, circular, and helical (a combination of linear and circular).

interface An interface is the conceptual boundary between systems or devices and usually involves changes in the way the communicated data is represented. Interface devices such as NSBs perform operations like changing the coding, format, or speed of the data.

least-significant (bit/word) See *rightmost (bit/word)*.

leftmost (bit/word) The highest numbered bits of a group of bits, generally of an entire word, or the highest numbered words of a group of words. These bits/words are often called most-significant bits/words.

linear interpolation Dual-axis, linear positioning from the present position to a point designated as the interpolation end point based on specified points.

load The processes of copying data either from an external device or from a storage area to an active portion of the system such as a display buffer. Also, an output device connected to the PC is called a load.

local In network communications, the node or device from which communications are being viewed. See *remote*.

LSS Abbreviation for Ladder Support Software.

M code An abbreviation for machine code. The user can set various M codes for various positions so that each M code will be output when the workpiece passes its respective position.

MC program A G-language program that controls the MC Unit's operation.

megabyte A unit of storage equal to one million bytes.

most-significant (bit/word) See *leftmost (bit/word).*

MS-DOS An operating system in common use on smaller computers.

NC contacts	Normally-closed contacts. A pair of contacts on a relay that open when the relay is energized.
negative software limit	The lower limit on the number of pulses set as a software parameter.
nesting	Programming one loop within another loop, programming a call to a subroutine within another subroutine, or programming an IF–ELSE programming section within another IF–ELSE section.
NO contacts	Normally-open contacts. A pair of contacts on a relay that close when the relay is energized.
OFF	The status of an input or output when a signal is said not to be present. The OFF state is generally represented by a low voltage or by non-conductivity, but can be defined as the opposite of either.
offline	The state in which a Programming Device is not functionally connected to the CPU, although it may be connected physically.
offset	A positive or negative value added to a base value such as an address to specify a desired value.
ON	The status of an input or output when a signal is said to be present. The ON state is generally represented by a high voltage or by conductivity, but can be defined as the opposite of either.
online	The state in which a Programming Device is functionally connected to the CPU so that CPU data and programs can be monitored or accessed.
online edit	An edit to a program made from a peripheral device connected to and currently online with a PC in PROGRAM or MONITOR mode. In MONITOR mode, this means that the program is changed while it is actually being executed.
origin proximity input	A signal input to indicate that the axis is near the origin.
origin search	An operation used to automatically move the axes to the origin or to define the origin.
output	The signal sent from the PC to an external device. The term output is often used abstractly or collectively to refer to outgoing signals.
parameters	Data which determines limits and other conditions under which an operation will be carried out.
PC	An acronym for Programmable Controller.
PC Setup	A group of operating parameters set in the PC from a Programming Device to control PC operation.
positive software limit	The upper limit on the number of pulses set as a software parameter.
present value	The current value registered in a device at any instant during its operation. Present value is abbreviated as PV. The use of this term is generally restricted to timers and counters.
program block	A unit of programming in MC programs roughly equivalent to program lines.
Programmable Controller	A computerized device that can accept inputs from external devices and generate outputs to external devices according to a program held in memory. Pro-

grammable Controllers are used to automate control of external devices. Although single-unit Programmable Controllers are available, building-block Programmable Controllers are constructed from separate components. Such Programmable Controllers are formed only when enough of these separate components are assembled to form a functional assembly, i.e., there is no one individual Unit called a PC.

Programming Device A Peripheral Device used to input a program into a PC or to alter or monitor a program already held in the PC. There are dedicated programming devices, such as Programming Consoles, and there are non-dedicated devices, such as a host computer.

pulses Discrete signals sent at a certain rate. The Motion Control Unit outputs pulses, each of which designates a certain amount of movement. Such pulses are converted to an equivalent control voltage in actual positioning.

pulse rate The distance moved the motor shaft divided by the number of pulses required for that movement.

pulse train A series of pulses output together.

remote In network communications, the node or device with which communications are taking place. See *local*.

retrieve The processes of copying data either from an external device or from a storage area to an active portion of the system such as a display buffer. Also, an output device connected to the PC is called a load.

rightmost (bit/word) The lowest numbered bits of a group of bits, generally of an entire word, or the lowest numbered words of a group of words. These bits/words are often called least-significant bits/words.

RUN mode The operating mode used by the PC for normal control operations.

servicing The process whereby the PC provides data to or receives data from external devices or remote I/O Units, or otherwise handles data transactions for Link Systems.

servolock An operation whereby a rotary encoder is used to maintain the position of a motor while it is stopped. Whenever the motor axis moves, the rotary encoder sends a feedback pulse to an error counter, causing a rotation voltage to be generated in the reverse direction so that the motor rotates back to its original position.

software error An error that originates in a software program.

sub-program A group of instructions that are executed independently of the main program.

target position A parameter for a positioning action that designates what position is to be reached at the completion of the action.

teaching Automatically writing the present position into memory, via the Teaching Box, as the target position for the designated positioning action.

transfer The process of moving data from one location to another within the PC, or between the PC and external devices. When data is transferred, generally a copy of the data is sent to the destination, i.e., the content of the source of the transfer is not changed.

unit address A number used to control network communications in FINS protocol. Unit addresses are computed for Units in various ways, e.g., 10 hex is added to the unit number to determine the unit address for a CPU Bus Unit.

unit number A number assigned to some Link Units, Special I/O Units, and CPU Bus Units to facilitate identification when assigning words or other operating parameters.

uploading The process of transferring a program or data from a lower-level or slave computer to a higher-level or host computer. If a Programming Devices is involved, the Programming Device is considered the host computer.

watchdog timer A timer within the system that ensures that the scan time stays within specified limits. When limits are reached, either warnings are given or PC operation is stopped depending on the particular limit that is reached.

WDT See *watchdog timer*.

wiring check A check performed automatically at startup to detect wiring problems such as reversed polarity or disconnections.

word A unit of data storage in memory that consists of 16 bits. All data areas consists of words. Some data areas can be accessed only by words; others, by either words or bits.

work bit A bit that can be used for data calculation or other manipulation in programming, i.e., a ‘work space’ in memory. Also see *work word*.

write-protect A state in which the contents of a storage device can be read but cannot be altered.

zone A range of positions or values which can be defined so that flags are turned ON whenever the present position is within the range.

Index

V

W

Revision History

A manual revision code appears as a suffix to the catalog number on the front cover of the manual.

Cat. No. W254-E1-1

Revision code

The following table outlines the changes made to the manual during each revision. Page numbers refer to the previous version.

Revision code	Date	Revised content
1	June 1995	Original production