USER'S MANUAL

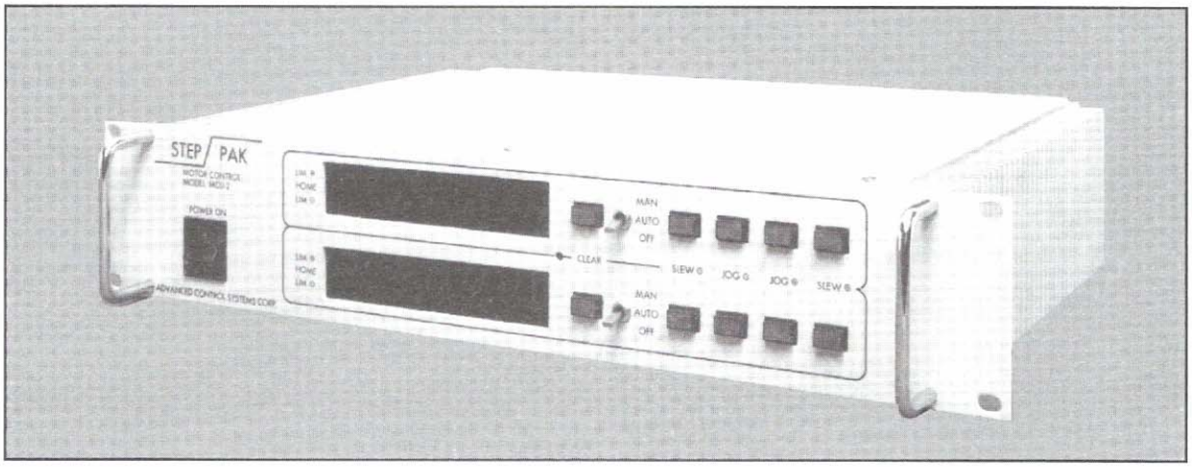
MCU-1 / MCU-2 STEPPING MOTOR CONTROLLERS

REVISION 1.1

ACS ADVANCED CONTROL SYSTEMS CORPORATION

35 Corporate Park Drive, Pembroke, Massachusetts 02359
PHONE: 781-829-9228 FAX: 781-829-9875
EMAIL: support.desk@acsmotion.com
www.ACSMotion.com

MCU-2
High Performance, Intelligent Stepping Motor Controller/Driver



FEATURES:

- □ Drives two four-phase motors rated from 2 to 6 Amps per phase; two motion stepping axes.
- □ Bi-level driver; switched holding current; no resitors in motor circuits.
- □ Speeds to 40,000 (half) steps per second use full dynamic range of motors.
- □ Simple, comprehensive manual controls with programmed resolution.
- □ Communication by serial link using either RS-232c or RS-485.
- □ Can be controlled from a single host port.
- □ Seven digit and sign position display by 0.56-inch LED in actual measuring (engineering) units, not steps.
- □ Accepts and displays limit and home switch inputs.
- □ Computes times for accel/decel rate changes in (trapezoidal) indexing functions using standard motion control variables.
- □ Decodes encoder signals for position indication.
- □ Non-volatile memory for motion control unit variables.

ACCESSORIES AVAILABLE:

- □ Power Supply
- □ Remote Controller

MCU-2

SPECIFICATIONS:

Instruction Set Overview: As a motion stepping package, the unit (1) needs information, (2) performs motion functions, (3) controls power and (4) stores information. Because it operates with a host computer it must receive instructions to (1) enter needed variable information, (2) call for execution of motion functions, (3) request changes in power flow and (4) report stored information. The unit accepts instructions expressed in standard stepping motor and motion control terminology.

In general, the instructions of each type, with the response produced by the controller/driver are: (1) Enter Data (variables), to which the unit responds "accepted". (2) Motion Execute, for which the unit (a) responds "accepted" and then (b) performs the motion function. (3) Output Control, for which the unit (a) responds "accepted" and then switches motor current. (4) Examine, to which the unit responds by sending the requested variable, either previously entered or computed and stored in the process of operation.

Communication: *Protocol* - A controller/driver operates as a "slave" under a "master", the host computer. The host computer issues commands, to which the controller/driver responds.

Modes - ASCII codes commands and responses are transmitted asynchronously. Standard baud rates are selected by convenience control: up to 19,200 using RS-232c, up to 300,000 using RS-485.

Position Displays: Using standard CNC practice, the position of each axis is expressed on the front panel as seven red decimal digits with sign and decimal point. For each axis the closure of switches, located on the equipment to mark plus and minus limits and home position, is indicated by the lighting of red LEDs.

If an encoder is used, its position is shown. The position of an axis without an encoder is derived from the motion execute instructions issued to that axis and the operation of any manual controls affecting that axis.

Manual Controls: For direct operator control, the front panel has: (1) an on/off switch for operation of the controller/driver, (2) a unit mode switch for each axis selecting auto(matic) operation under the host computer, man(ual) operation from the front panel, and off, turning off the motor current, and (3) pushbuttons for each axis, operating in manual mode, to set position display to zero and slew and jog at the entered rate in plus and minus directions.

Convenience Controls: To configure the unit for a particular equipment arrangement, the back panel has: (1) a screwdriver-actuated communication mode switch for the communication standard and baud rate, (2) a screwdriver-actuated address switch for each axis, and (3) two screwdriver-actuated potentiometers with locking nuts for each axis, setting motor torque.

User Connectors Mounted on the back panel and selected and located for safety are: (1) round-type, high current motor conectors for each axis, (2) a 25-pin D-type connector for the RS-232C communication standard, (3) a 10-pin flat cable connector for the RS-485 communication standard and multi-unit chaining, (4) 10-pin flat cable connectors for encoder, limit switch and home switch signals for each axis, (5) a motor power barrier terminal block, (6) a 115v. AC inlet connector, and (7) a switched 115v. AC plug.

Motor Requirements: Basic to the translator circuitry of the controller/driver, the unit operates four phase DC stepping motors with six or eight leads. The maximum rated current per phase is six amperes.

Encoder Requirements: ***(if used)*** Providing for bidirectionsl movement of an axis, the unit requires a dual channel pulse output incremental encoder with output signals phased in quadrature and a signal level of 5 volts (TTL).

Power Requirements: The architecture of the power system separates the requirements of the control/drive, as to displays, logic, communication and encoders, from the requirements of the motors. The unit requires 115v. AC power at 60 HZ, drawing up to 1.0 ampere. Motor power needed consists of unregulated DC in two voltage ranges, 10 to 12 and 36 to 42. Current draw can be up to 24 amperes, depending on the motors used. The motor supply should be able to supply and accept high current surges.

Physical Dimensions:

Width Over Heat Sinks	17.1 inches
Depth Behind Front Panel	13.0 inches
Height	3.5 inches
Depth Overall (Over Handles)	14.6 inches
Weight	12 pounds

OEM Versions: Several are available. An "A" model has no position displays, manual controls or rack mounting panel. The single-axis unit is also available (1) in an open frame without displays, controls, user connectors or convenience adjustment controls and (2) (modular) on a single-sided printed circuit board with first-stage heat sink only. Open frame and board both have barrier terminal blocks for motor power connections and flat cable connectors for other connections. OEM versions can be mounted in any orientation.

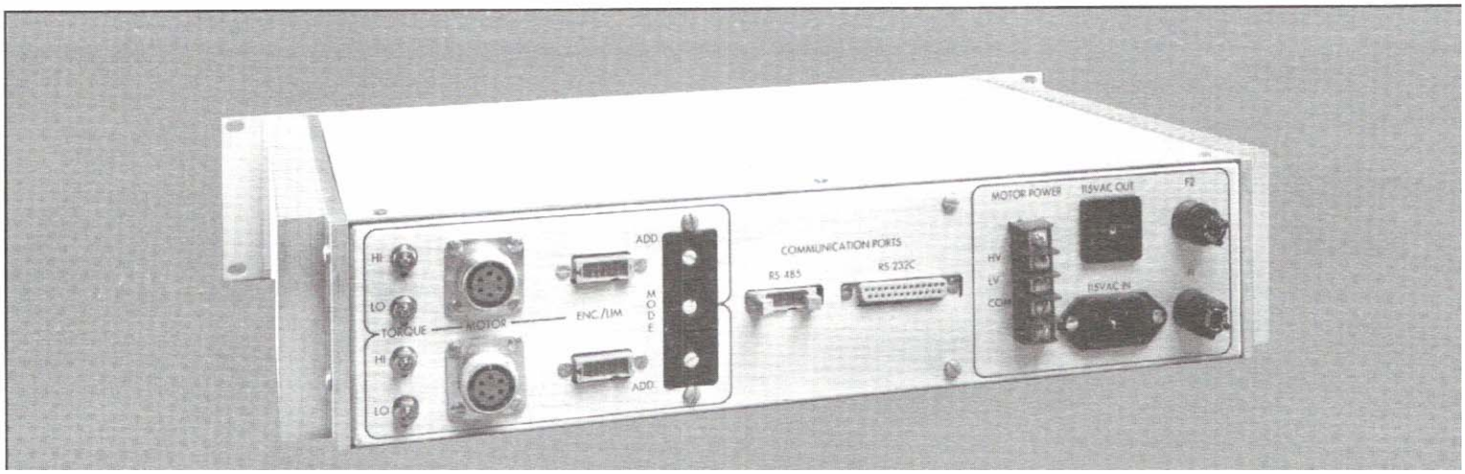


TABLE OF CONTENTS

GENERAL NOTICES

SECTION 1 GENERAL INFORMATION

1.1 Safety

“The use of electric motors and generators, like that of all other utilization of concentrated power, is potentially hazardous. The degree of hazard can be greatly reduced by proper design, selection, installation, and use, but hazards cannot be completely eliminated. The reduction of hazard is the joint responsibility of the user, the manufacturer of the driven or driving equipment, and the manufacturer of the motor or generator.”

> Standards Publication No. ANSI-C51-1/NEMA MG-2
> “Safety Standard for Construction and Guide for Selection,
> Installation, and Use of Electric Motors and Generators”.
> Available From:
> National Electrical Manufacturer’s Association
> 2101 L Street N. W.
> Washington, DC 20037

This manual should be completely read and understood prior to installing, operating, or troubleshooting the MCU-2. Observe all federal, state, and local safety standards that are pertinent to this piece of equipment.

1.2 MCU-2 Features

- The MCU-2 drives two four-phase motors, providing two motion stepping axes. The MCU-1 drives one motor.

- A proven design in the critical translator stage provides high power and high speed using conservative industrial voltages and other motor power supply characteristics.

- Available stepping rates are up to 40,000 (half) steps per second (eight-step input sequence, also called half-step mode) use the full mechanical speed range of the motors.

- Programmable reduction of stepping rates by factor of ten.

- No dropping resistors are used, meaning that there are no resistors in motor circuits. All necessary tuning of the unit/motor system is done with screwdriver-actuated adjustments of the unit.

- Communication is by serial link, which provides great flexibility in system architecture. Electrical standards are RS-232C and RS-485.

- If an incremental encoder is used on a motion stepping axis, the MCU-2 will decode the signals from the encoder to provide position information.

- MCU-2 performs indexing functions without depending on a host computer. This means that an axis can be driven a specified distance, starting at a specified base speed, accelerating to and decelerating from a specified top speed at a specified ramp, and stopping from the base speed. This substantially frees the host computer.

- In addition to stopping stepping of a motion axis upon closure of a limit switch, the MCU-2 responds to inquiries from the host computer as to the status, open or closed, of either of the limit switches or the home position switch.

- Each axis has an address with the number of available addresses allowing up to 100 motors to be controlled from a single host computer serial port.

- On instruction the unit will store data entered in a nonvolatile memory which retains the information even when the supply voltage to the control/drive is removed.

- Basic MCU-2 versions have a position display for each axis consisting of seven digits and sign, in conjunction with a conversion constant communicated from the host computer. Position is shown in large, bright numerals in engineering units.

- Basic MCU-2 versions have a three-position mode switch and just five spring-loaded push-button switches, for each axis. With speed rates and acceleration/deceleration ramp factors communicated from the host computer the manual controls can be simple and comfortable for the operator and appropriate to the physics of the equipment.

- MCU-1 is the basic versions of the single axis unit.

1.3 Controls and Instructions

The MCU-2 operates in one of three modes, automatic, manual, and motor current off, depending on the setting of the switch on the front panel.

In the manual slew mode push-buttons on the front panel set the position display to zero and slew and jog at the entered rates in positive or negative directions. (Thus there are five (5) push-button controls for each axis). If no value has been entered for any or all of the numeric values needed, jog or slew rate or ramp factor, including the case where the unit has never been connected to at host computer, the default values will be uses. Thus the MCU-2 can actually operate usefully without a host computer, although normally we think of the unit as a computer peripheral.

The mode causing motor current to be turned off removes power from the motor windings, thus removing torque except for a small value of residual motor torque.

In the automatic mode the MCU-2 follows the instructions of a host computer. The unit has such computing power, or intelligence, however, that depending on the physics of the equipment, characteristics of the motors, (and encoders, if used), instructions chosen by the programmer, and the supervisory program executed by the host computer, very few instructions from the host to controller/driver may be required. This affects the amount of time the host must devote to developing and transmitting instructions for a motion stepping axis, the other functions performed by the supervisory program of the host, and the number of MCU-2 control/drive units that can be supervised by a host.

An instruction is a character or group of characters that will be interpreted by the unit as a command. An instruction is formatted as follows:

> **Start Character**
> **Axis Address/Instruction Mnemonic/Parameter/Terminator**
> **Example: #12G02**
> **Axis with address 2 will index the motor to next position,**
> **using parameter locations 02. "↵" is an instruction terminator.**

<u>Start Character</u>: All messages generated by the host computer must start with ASCII # character.

<u>Axis Address</u>: A decimal from "00" to "99" and "**". If not a valid address, the instruction will be ignored. "**" is an "ALL" or Global axis address. "*" can replace any address decimal digit.

Instruction Mnemonic: Consists of on to tow non-numeric characters from the list of commands. If not of this list the unit will ignore the instruction.

Parameter: Not required for all instructions. Must be numeric. Length depends on
instruction.

Terminator: Carriage Return character, a non-printing character, terminates all commands.

An instruction is communicated to the unit. When a terminator is received by the unit, the unit communicates its response.

An important aspect of operation is the fact that a conversion constant can be entered which allows use of measuring (engineering) units instead of motor steps to be displayed. The conversion constant is that number of motor steps which constitutes one measuring unit of position. The default value of this parameter is 1.

1.4 Displays and Connectors

Basic versions display the position of each axis on the front panel as a signed seven digit decimal number with decimal point floating. Light emitting diodes immediately to the left of the sign indicate closure of limit switches or the home position switch. Among the data to be entered via communication link with the host computer is that conversion constant which will result in position being displayed in the measuring units of interest rather than in motor steps. The height of each digit is 0.56 inches and the color of the display is red.

All units have a back panel on which are located connectors designed for multiple or repeated use. These connectors are of different types, depending on the kind of signal or power involved and safety considerations.

Motor Connectors - are round, mechanically shielded, 6 pins.
RS-232 Connector - is D-type, electrically shielded, 25 pins.
RS-485 Connector - is flat cable type, 10 pins.
Encoder and Switch Connectors - are flat cable type, 10 pins.
Motor Power (Input/Supply) Connector - is a barrier terminal block, 3 terminals.
115 VAC Inlet Connector - is three-pronged, mechanically shielded.

In addition basic units have a 115 VAC outlet connector which is three-pronged, mechanically shielded. Power to this connector is on when the front panel switch is on.

Again, excepting certain OEM versions, all units have convenient controls for configuring the unit to a particular equipment arrangement, a task that may have to be done more than once, but is not done repeatedly. These controls are all screwdriver-actuated.

Communication Mode Switch- detent -type, 16 position
Axis Address Switch - detent - type, 16 position
Motor Torque Potentiometers - with locking nuts

1.5 Communication Standards

The unit communicates with a host computer by ASCII codes transmitted in bit-serial fashion over a single transmission line. The rates at which data can be transmitted and received, in bits per second are:

COMMUNICATION MODE SWITCH SETTING	**BAUD RATE**
0	300 Parity Ignored
1	600 “ “
2	1200 “ “
3	2400 “ “
4	4800 “ “
5	9600 “ “
6	19,200 “ “
7	Spare
8	300 Parity Programmed
9	600 “ “
10	1200 “ “
11	2400 “ “
12	4800 “ “
13	9600 “ “
14	19,200 “ “
15	Spare

Each ASCII code is transmitted by sending a start bit, eight data bits, and a stop bit. This means each code in transmission is ten bits. Parity can be preprogrammed to be even, odd, or ignored.

1.6 Moving Elements: Motors and Encoders

The fundamental motor specification is that of a four phase DC stepping motor with six or eight leads and a rated phase current of six amperes or less. This is the type of digital motor found in commercial applications.

The encoder specification is for a dual channel pulse output incremental encoder with output signal phased in quadrature and a signed level of 5 volts (TTL, Transistor-Transistor Logic). This is the kind of encoder used for digitizing motion in commercial applications.

In the architecture of a motion system, an address must be assigned to each motion axis. The following table relates the back panel axis address switch to the logical address used for communication.

Address Switch Setting	**Address**
0	00 or Preprogrammed for any value up to 99
1	01
2	02
3	03
4	04
5	05
6	06
7	07
8	08
9	09
10	10
11	11
12	12
13	13
14	14
15	15

1.7 Power Plan

The requirements of the controller/driver are separated from the requirements of the motor to provide isolation of electrical noise, allow easy integration of heavy components into equipment and permit tailoring of electrical and mechanical arrangements.

The unit requires 115 VAC power at 60 Hz. drawing up to 1.0 ampere. This supplies the needs of displays, logic, communications and encoders.

Motor power consists of unregulated DC in two voltage ranges, 10 to 15 and 36 to 42. Current draw can be up to 24 amperes, depending on the motors used. The motor power source should be able to supply and accept high current surges. A 10,000 microfarad capacitor connected across each of the two voltages is adequate.

Back panel potentiometers are used to adjust motor torque, after which they are locked. Each motion axis has two torque potentiometers, one marked HI(gh), the other LO(w).

High (voltage) (pulse width) torque, or high speed torque, or torque at speed, is the adjustment for smooth operation of the motor at the (highest) speed required under equipment operating conditions. This adjustment is at its minimum when the potentiometer is fully counterclockwise, and at is maximum in the fully clockwise position. Motor temperature is directly affected by this adjustment.

Low (voltage) (current switching duty cycle) torque, or holding torque, is the adjustment for rated (idle) current in the motor winding when at standstill or zero speed. This adjustment is at its minimum when the potentiometer is fully counterclockwise, and at its maximum in the fully clockwise position. Motor temperature is directly affected by this adjustment.

1.8 Dimensions and Mounting

Dimensions and weight of basic versions are:

Width Over Heat Sink	17.1 inches
Depth Behind Front Panel	13.0 inches
Height	3.5 inches
Depth Overall (Over Handles)	14.6 inches
Weight	12 pounds

Basic versions have a front panel designed to allow the MCU-2 to be mounted on a standard 19 inch rack. In addition, the front panel has two handles mounted on it to make movement between locations easy.

OEM versions can be mounted in a wide variety of ways.

1.9 Motion Systems Overview

The motion requirements of a machine are considered to form a system, distinct from other systems within the machine. With the other systems the motion system is directed by a computer of some kind.

In relation to this computer the motion system is subordinate, the relationship being that of a “master” and “slave” or “host” can “guest”, hence the general use of the term host computer.

As a motion stepping package, the controller/driver is a key element in a motion system. It performs several functions:

1. Receiving information from various sources.
2. Performing motion function on command.
3. Controlling “un-phased” power to the motor, which exerts a holding torque at standstill or zero speed.
4. Transmitting stored information in various ways.

All of these functions have a discreet nature to them leading to the consideration of time throughout the controller/driver.

This is a consideration throughout an entire machine. Each element of each system must be reviewed, at least, briefly, to make sure that the speed with which it operates is consistent with the speed of neighboring elements and the work performed by the machine.

1.10 Controller/Driver Design

A useful element with which to begin might be that of position. As a motion stepping package, the unit must keep track of the position of moving element(s). It does this with a counter for each axis which has a range of ±8,388,606. This counter is driven up and down by signals from either of two sources. If the motion system is equipped with an encoder, the signals from the device are counted. If no encoder is used, signals are taken from the step control section of the unit itself.

If an encoder is used, its signals are decoded, then sent to the counter.

Connected to the counter, on basic versions, is the position display.

Control of the movement of the electrical motor lies in a step control section which emits a signal indicating clockwise or counterclockwise rotation and a pulse of proper duration, voltage and rise and fall time to actuate the driver. The number of pulses and interval between each of them determine the distance of movement of the load and its speed at all times while moving. The step control also controls the state of holding of idle current when the motor is at standstill or zero speed. Pulses from the step control are counted for position display if no encoder is used.

Power to the motor is modulated by an electronic drive which translates the information received from the step control into electric power for an electro-magnetic-mechanical device, the motor. The MCU-2 uses a proven bi-level design which operates on conservative voltages, appropriate to commercial machinery and provides excellent overall performance.

Storage of variables without depending on power is done by a nonvolatile memory under the direct control of the central processor.

SECTION 2 INSTALLATION AND INTERCONNECTIONS

2.1 MCU-2 Motor Operation

The MCU-2 Driver/Controller is designed to operate 2 stepping motors. Running only 1 motor can cause damage to the channel not connected to a motor if the channel is powered-up. The board level fuses to the channel not connected to a motor must be manually removed. If you plan to run a single motor with the MCU-2, follow the instructions below.

WARNING! DO NOT SUPPLY POWER TO THE MCU-2 DRIVER/CONTROLLER UNTIL THIS PROCESS IS COMPLETE!

1. CONNECT THE MOTOR TO BE DRIVEN TO THE MCU-2 VIA ONE OF THE 2 (TWO) BACK PANEL CONNECTORS.

2. OPEN THE MCU-2 TOP COVER BY REMOVING THE 6 RETAINING SCREWS

3. IDENTIFY THE CCB-5 DRIVER BOARD WIRED TO THE CORRESPONDING MOTOR CONNECTOR

4. REMOVE BOTH TWO (2) FUSES ON THE CCB-5 DRIVER BOARD ***NOT CONNECTED*** TO THE STEPPING MOTOR IN STEP 3 ABOVE.

5. REPLACE THE MCU-2 TOP COVER AND INSTALL THE 6 RETAINING SCREWS

6. FOLLOW THE REMAINING INSTALLATION INSTRUCTIONS IN THIS USER’S MANUAL

2.2 Internal Settings

The MCU-2 is factory preset for the following operating parameters:

Operation without encoder
Home switch input normally open
Limit switches inputs normally closed

Refer to table 2.1 for alternate jumper setting. Remove top plate and move jumpers accordingly. See also board pictorial.

JUMPER	**FUNCTION**
A1 - A2, B1 - B2	Encoder Inputs
A2 - A3, B2 - B3	No Encoder Inputs

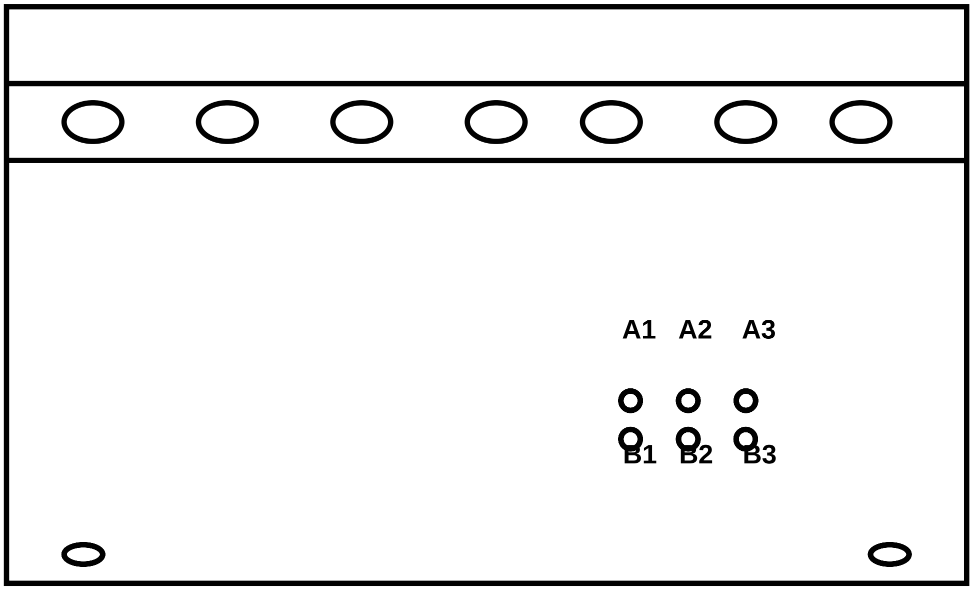


Figure 2.1 Board Pictorial

2.3 Motor Connection

Motor is to be connected via round motor connector on the back panel. Refer to Fig. 2.2 for pin assignment and proper stepping motor connection. Refer to Motor Manufacturer data sheets for connection lead color codes or use an Ohm meter to determine the required motor lead configuration.

S-6S (639)

Mating Plug:	Amphenol MS3106A 14S-6P (639)
Cable Clamp:	Amphenol 97-3057-1007 (639)
Cable Boot:	Amphenol 9779-513-100

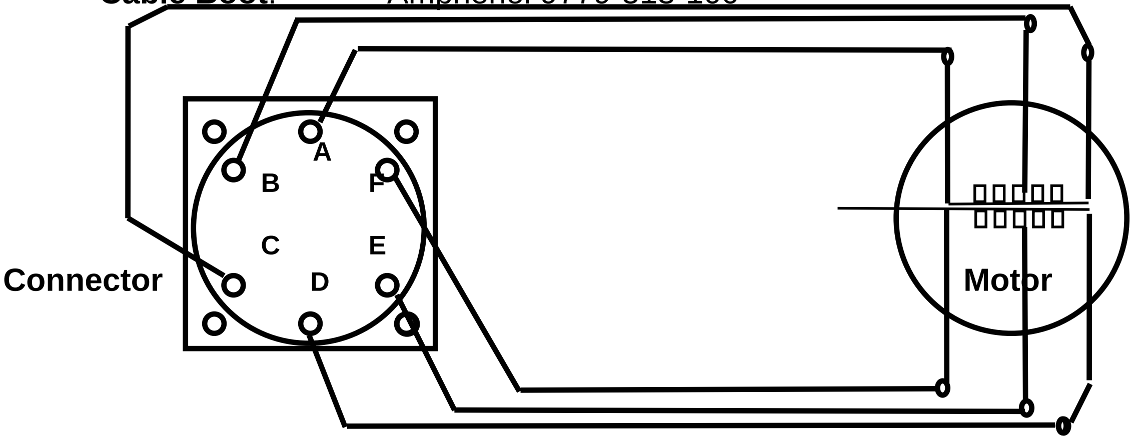


Figure 2.2 Motor Connection To MCU-2 Unit

2.4 Power Connections

Connect external power supply to motor power terminal block as indicated on the back panel of MCU-2 unit; common ground to “COM”, +12VDC to “LV”, +36VDC to “HV”. Connect power supply line cable to “115 VAC OUT” connector on the back of MCU-2 unit. Use supplied AC power cable to connect MCU-2 to AC line power.

2.5 Encoder/Limits Connections

When incremental encoder and limit inputs are utilized, connect these devices per Fig. 2.4. Busy Line: TTL Compatible, LO when stepping.

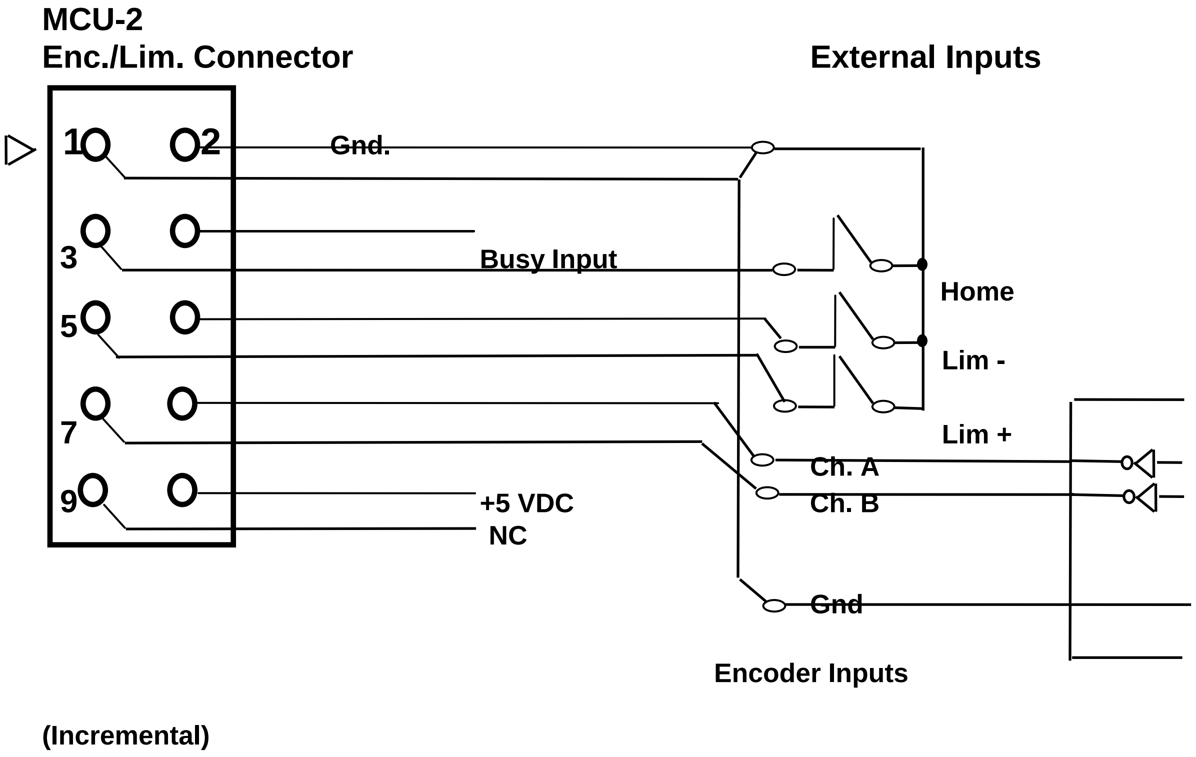


Figure 2.5 Encoders/Limits/Home/Interconnect

2.6 Communication Ports Connections

Connect RS-232 Port of MCU-2 (DB25 type connector) to terminal or host computer serial port per connection diagram on Fig. 2.5A or Fig. 2.5B. Pin 5 on MCU-2 is an output. LO on Pin 5 indicates that MCU-2 is ready for new command.

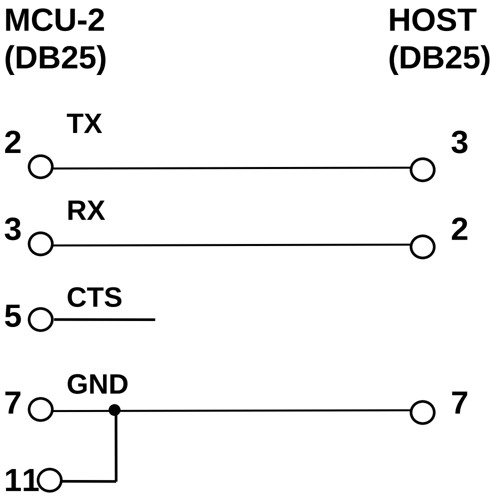


Fig. 2.5A DB25 to DB25 Communication Cable

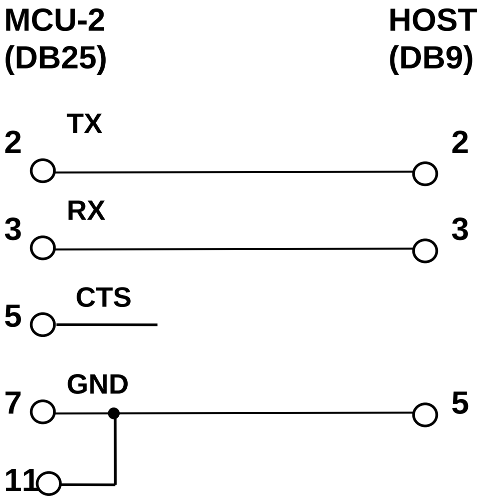


Fig. 2.5B DB25 to DB9 Communication Cable

When there is more than one MCU-2 in the system, they must be daisy chained via RS-485 communication ports and only one MCU-2 must be connected via RS-232 communication cable to the host.

RS-485 Standard defines drives and receivers for balanced digital multipoint communication system. The configuration of a system normally consists of several drivers, several receivers and terminating resistor. There can be up to 32 Driver/Receiver pairs on a single balanced line. By utilizing line repeaters the system can be expanded even more. RS-485 is compatible with RS422-A.

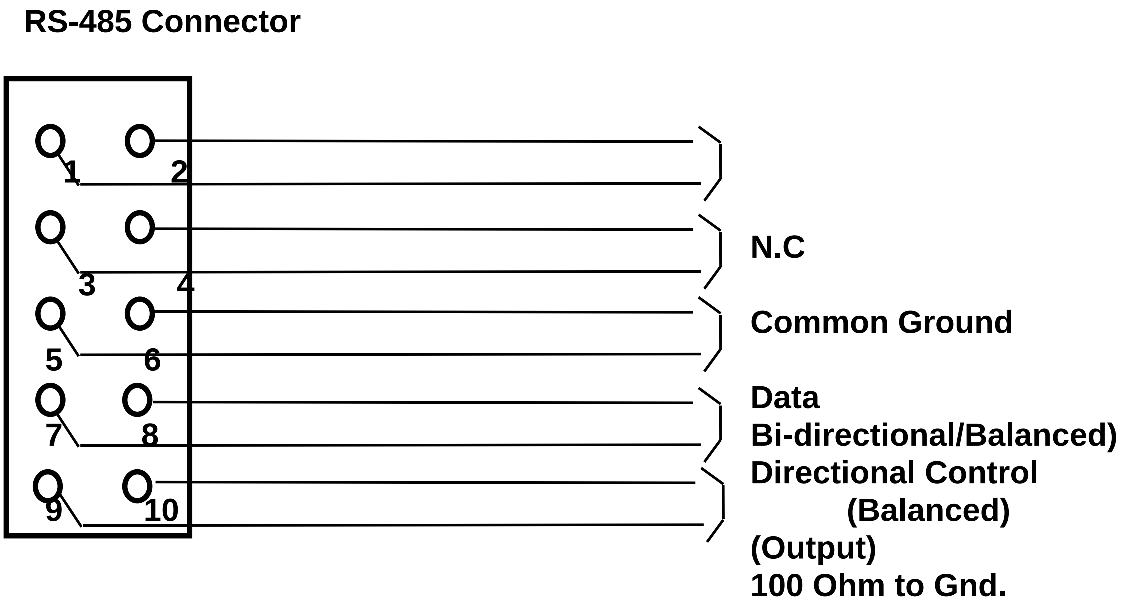


Figure 2.6; RS-485 Pin Assignments

SECTION 3 MCU-2 INSTRUCTION SET

3.1 Instruction Set Abbreviations and Notes

- Start Character
aa - Controller Address: Range 00 to 99. ** is All Address
b - Status Indicator: b = 1 Winding Current On
b = 0 Winding Current Off
c - Conversion Constant; Range 1 to 65535
dd - Data Block Address; Range 00 to 80
k - Ramp Data; Range 1 to 100 (1,000 to 100,000 step/sec^2)
r - Rate Data 10 to 40,000 steps/sec
n - Position or Step Count Data; Range -8,388,606 to +8,388,606
↵ - Indicates carriage return character

bbbbb

O.........Motor is at zero speed
R.........Motor is ramping
S..........Motor is at speed

O..........Controller is in OFF mode
A..........Controller is in AUTO mode
M.........Controller is in MAN mode

O...........Positive Direction Limit is not activated
+............Positive Direction Limit is activated

O..........Home Input is not activated
H..........Home Input is activated

O...........Negative Direction Limit is not activated
- ....Negative Direction Limit is activated

For example: OOAS: Limits and home inputs are not activated,
controller is in auto mode, and motor is at speed.

MCU-2 does not respond to "All" Address.
MCU-2 does not respond to unrecognized addresses.

MCU-2 responds with aa? when address is recognized but structure of the instruction is wrong.

3.2 Data Enter Instructions

MNEMONIC	DESCRIPTION	RESPONSE
#aaA↵	Accept Data to NVRAM	aa↵
#aaB=r↵	Base Rate Enter	aa↵
#aaC=r↵	Conversion Constant Error	aa↵
#aaDdd=±n↵	Distance Data Enter	aa↵
#aaJ=r↵D	Jog Rate (Manual) Enter	aa↵
#aaK=1↵;aaK=10↵	Rate Division Constant Enter	aa↵
#aaP=±n↵	Position Enter	aa↵
#aaRdd=k↵	Ramp Data Enter	aa↵
#aaVdd=r↵	Velocity (Rate)Data Enter	aa↵
#aaW=b↵	Winding (Motor) Current On (b=1) on Off (b=0)	aa↵
#aaZ↵	Zero Current Position	aa↵
#aaV00=r↵	Velocity Data for Manual and immediate Slew Enter	aa↵
#aaR00=k↵	Ramp Data for Manual and immediate Slew Enter	aa↵

Unit Configuration Instructions

#aaUPE↵	Set Even Parity	aa↵
#aaUPO↵	Set Odd Parity	aa↵
#aaUPN↵	Set No Parity	aa↵
#aaUaa↵	Set Unit Address to aa	aa↵
#aaPRES↵	Preset NVRAM to Default Conditions	aa↵

3.3 Data Examine Instructions

MNEMONIC	DESCRIPTION	RESPONSE
#aaB↵	Base Rate Examine	aaB=r↵
#aaC↵	Conversion Constant Examine	aaC=c↵
#aaDdd↵	Distance Data Examine	aaDdd=±n↵
#aaE↵	Examine Limits/Home Inputs	aaE=bbb↵
#aaJ↵	Jog Rate (Manual) Examine	aaJ=r↵
#aaK↵	Rate Division Constant Examine	aaK=1↵ or aaK=10↵
#aaM↵	Motion Status	aaM=bb↵
#aaP↵	Present Position Examine	aaP=±n↵
#aaRdd↵	Ramp Data Examine	aaRdd=k↵
#aaVdd↵	Velocity (Rate) Data Examine	aaVdd=r↵
#aaW↵	Winding (Motor) Current Examine	aaW=b↵

Data Block 00: Velocity data for Slew function
Ramp Index for Slew function

Data Block 01: Default Ramping Index
Velocity data for Go function and Index function
Ramp Index for Go function and Index function

Note: There are 80 data blocks. Each block contains Velocity (top rate), Distance, and Ramp index. All this information is needed for motion execute commands.

3.4 Motion Execute Instructions

MNEMONIC	RESPONSE	DESCRIPTION	MOTOR RESPONSE
#aaF↵	aa↵	Finish Function	Motor decelerates to Base Rate and stops.
#aaG+n↵aa↵	Go Function	Motor moves to ±n absolute	
#aaG-n↵	aa↵		position using rate and ramp data stored in Location 01.
#aaGdd↵	aa↵	Go Function	Motor moves to next position rate and ramp data stored at Location dd.
#aaH+↵	aa↵	Home Function	Motor moves at Base Rate to
#aaH-↵	aa↵		Home position in positive and negative direction.
#aaI+n↵	aa↵	Index Function	Motor moves n steps in
#aaI-n↵	aa↵		positive or negative using rate and ramp data store at Location 01.
#aaIdd↵	aa↵	Index Function	Motor moves preset number of steps using data stored at Location dd.
#aaL+↵	aa↵	Limit Function	Motor moves at base rate to
#aaL-↵	aa↵		positive or negative limit switch.
#aaQ↵	aa↵	Quit Function	Motor stops stepping immediately.
#aaS+↵	aa↵	Slew Start Function	Motor ramps in selected
#aaS-↵	aa↵		direction from Base rate using rate and ramp data stored in Location dd.
#aaSdd↵	aa↵	Slew Function	Motor ramps to next rate using rate and ramp data stored in Location dd.
#aaS+r↵	aa↵	Slew Function	Motor ramps in selected
#aaS-r↵	aa↵		direction using ramp data stored in Location 00; and for speed using immediate data (provided for instructions).

3.5 Instruction Set: Specifics and Examples

A: Accept Data to NVRAM

Instruction: #03A↵ Response: 03↵

Function: Controller with address set to 03 will copy all data from volatile memory into nonvolatile storage. It responds with 03.

Instruction: #*3A↵ No Response

Function: All Controllers with second address digit equal 3 will memorize data.

Instruction: #0*A↵ No Response

Function: All Controllers with first address digit equal 0 will memorize data.

NOTE: * is a global address character and can be used on any instruction. No response is generated.

B: Base Rate Enter/Examine

Instruction: #03B=400↵ Response: 03↵

Function: Controller loads base rate registers with base rate of 400 steps per second, and responds with its own address and carriage return character.

NOTE: Base Rate is starting and stopping rate when motor accelerates to or decelerates from higher speeds. Base rate should be set below start/stop rate of motor load combination from accurate positioning.

Instruction: #03B↵ Response: 03↵

Function: Examination of base rate; base rate is 400 steps/sec.

3.5 Instruction Set: Specifics and Examples

C: Conversion Constant Enter/Examine

Instruction: #03C=200↵ Response: 03↵

Function: Controller loads conversion constant register with value 200.

NOTE: Conversion Constant is used to display calculations. Number of steps is divided by conversion constant and displayed. For example, to move load linearly for one inch, motor has to execute 200 steps. Conversion Constant should be 200 in order to display l”. Conversion Constant is 1 to 65535 in unite increments.

Instruction: #03C↵ Response: 03C=200↵

Function: Examination of Conversion Constant; it is 200

D: Distance Data Enter/Examine

Instruction: #03D12=+3500↵ Response: 03↵

Function: Controller with address 03 loads positive direction and count of 3500 into memory block #12.

Instruction: #03D12↵ Response: 03D12+3500↵

Function: Controller #03, Data Block 12 contains +3500 distance/position information.

NOTE: Count of 3500 can be number of steps when used by “I” instruction, or
absolute position when used by “G” instruction.

3.5 Instruction Set: Specifics and Examples

E: Examine Limits/Home Inputs

Instruction: #03E↵ Response: 03E=000AS↵

Function: Controller 03 responded with 000AS code which means that both limit input and home input are not activated, controller is in automatic mode, and motor is at speed.

F: Finish Function

Instruction: #03F↵ Response: 03↵

Function: Motor 03 starts decelerating and will stop when reaching base rate.

NOTE: Deceleration rate is equal to the last acceleration rate.

3.5 Instruction Set: Specifics and Examples

G: Go Function

Instruction: #03G-30000↵ Response: 03↵

Function: Motor 03 executes index to absolute position -30000 steps using preprogrammed base rate, top rate and ramping index.

NOTE: Base rate is stored in base rate register. Data Block 01 contains the top
rate and ramping index.

Instruction: #03G15↵ Response: 03↵

Function: Motor 03 executes index to absolute position stored in data block 15.

NOTE: Data Block 15 contains also top rate and ramping index for this particular move.
F and Q functions are enabled.

H: Home Function

Instruction: #03H+↵ Response: 03↵

Function: Motor 03 finds home position moving in positive direction.

NOTE: Motor moves at base rate in positive direction until LO is recognized
on home input. If it find positive limit first, it reverses direction.

3.5 Instruction Set: Specifics and Examples

I: Index Function

Instruction: #03I+10000↵ Response: 03↵

Function: Motor 03 executes index of 10000 steps in positive direction using preprogrammed base rate, top rate, and ramping index.

NOTE: Base rate is stored in base rate register. Data Block 01 contains top
rate and ramping index.

Instruction: #03I07↵ Response: 03↵

Function: Motor 03 executes index of preprogrammed number of steps in preprogrammed direction.

NOTE: Data Block 07 contains distance, direction, top rate and ramping index.

J: Jog Rate Enter/Examine

Instruction: #03J=500↵ Response: 03↵

Function: Rate 500 steps/sec is entered into jog rate register for later use in manual jog mode.

NOTE: Jog rate is usually set below start/stop speed. There is no ramping in manual jog mode.

Instruction: #03J↵ Response: 03J=500↵

Function: Jog rate is examined; it is 500.

3.5 Instruction Set: Specifics and Examples

K: Rate Division Constant Enter/Examine

Instruction: #03K=10↵ Response: 03↵

Function: Controller is programmed with rate division constant 10. All preprogrammed stepping and ramping rates are divided by 10.

Instruction: #03K↵ Response: 03K=10↵

Function: Rate division is examined; it is 10.

L: Limit Function

Instruction: #03L+↵ Response: 03↵

Function: Motor 03 moves in positive direction at base speed until HI is recognized on limit + input.

NOTE: F and Q instructions are active in limit function.

M: Motion Status Examine

Instruction: #03M↵ Response: 03M=00↵

Function: Controller 03 responded with 00 Code which means that the motor is at standstill.

Response: 03M=10↵ Motor Ramping
Response: 03M=11↵ Motor at Speed

3.5 Instruction Set: Specifics and Examples

P: Position Enter/Examine

Instruction: #03P=+5600↵ Response: 03↵

Function: Controller 03 position counter is preset to +5600 count (or steps).

NOTE: Sign has to always be in the message structure. Position range is from
+8,388,606 to -8,388,606.

Instruction: #03P↵ Response: 03P=+5600↵

Function: Position is examined; it is +5600.

Q: Quit Function

Instruction: 03Q↵ Response: 03↵

Function: Motor 03 stops stepping immediately.

NOTE: This instruction is operable for all automatic motion functions.

3.5 Instruction Set: Specifics and Examples

R: Ramp Index Enter/Examine

Instruction: #03R08=5↵ Response: 03↵

Function: Controller 03 ramp index 5 into data block 08.

NOTE: Ramp index 5 represents 5,000 step/sec² ramping constant. Range of
ramp index is 1 to 100.

Instruction: #03R08↵ Response: 03R08+5↵

Function: Ramp index is examined; it is 5.

S: Slew Function

Instruction: #03S-↵ Response: 03↵

Function: Controller 03 starts slew function in negative direction. It ramps from base speed to preprogrammed speed in data block 00.

NOTE: Ramp index is also stored in data block 00.

Instruction: #03S-10000↵ Response: 03↵

Function: Controller 03 starts slew function in negative direction and ramps to 10,000 steps/second using ramp data in data Block 00.

Instruction: #03S25↵ Response: 03↵

Function: Controller 03 ramps to preprogrammed speed in data block 25 using direction and ramp index in data block 25.

NOTE: Slew function enables the host controller to generate various velocity
profiles. The very first slew instruction defines the direction of rotation.
Speed can be changed by use of the three slew formats; direction information is being ignored.

3.5 Instruction Set: Specifics and Examples

U: Unit Configuration

Instruction: #03UPE↵ Response: 03↵

Function: Controller 03 is programmed for even parity.

NOTE: Even parity mode will become active on the next power reset.

Instruction: #03UPO↵ Response: 03↵

Function: Controller 03 is programmed for odd parity.

Instruction: #03UPN↵ Response: 03↵

Function: Controller 03 is programmed to ignore parity bit on receive, and to transmit “0” for parity bit on transmit.

Instruction: #03UA17↵ Response: 03↵
#03A↵ Response: 03↵

Function: Controller 03 new address is programmed to be 17.

NOTE: This new address becomes active on the next power reset. Address select
switch must be set to 0.

NOTE: #03A instruction must be executed for permanent store of new configuration before power OFF.

3.5 Instruction Set: Specifics and Examples

V: Velocity Data Enter/Examine

Instruction: #03V08=5000↵ Response: 03↵

Function: Controller 03, data block 08 is programmed for velocity (rate) 5000 step/sec.

NOTE: Velocity range is 10 to 40,000 steps/sec.

Instruction: #03V00=2000↵ Response: 03↵

Function: Controller 03, data block 00 is programmed for 200 steps per second.

NOTE: Data Block 00 is used for slew function.

3.5 Instruction Set: Specifics and Examples

W: Winding (Motor) Current Examine

Instruction: #03W=0↵ Response: 03↵

Function: Controller 03 shuts off motor current.

Instruction: #03W=1↵ Response: 03↵

Function: Controller 03 turns on motor current.

NOTE: This is also motor phases reset; motor phases are turned on is a preset (known) state.

Instruction: #03W↵ Response: 03W=1↵ or 03W=0↵

Function: Motor current is examined.

NOTE: “1” response indicates the current is on.

Z: Zero Position Counter

Instruction: #03Z↵ Response: 03↵

Function: Controller 03 position counter is set to zero.

NOTE: This instruction is the same as #03P=+0↵.

SECTION 4

4.1 Motor Current Adjustments

The MCU-2 is to be adjusted to achieve the best performance of the selected stepping motor. The "LO" potentiometer adjustment on the back panel of the MCU-2 controls idle motor winding current, or holding torque. It can be adjusted in the range of 2 to 6 Amp. The "HI" potentiometer adjustment controls the running torque as well as compensates for motor resonances.

Procedure:

4.11 Power up the unit.

Turn mode switch to "Manual" for one motor. Adjust corresponding "LO" pot for required holding torque. This is done by rotating the motor shaft back and forth by hand while slowly turning the "LO" speed pot from extreme CCW to CW. At approximately one third of the pots rotation there will be a noticeable increase in motor shaft resistance. This is the point at which you stop turning the potentiometer, i.e. you have reached the setpoint for the "LO" speed and holding torque. It is important that you locate the pot setpoint at exactly the shaft resistance needed to provide a minimum holding torque required. Actuate the jog pushbutton and ensure that the jog function operates the motor smoothly. It may be necessary to provide a small increase in the "LO" speed pot to remove any missteping that may occur due to insufficient "LO" speed torque. WARNING! The motor temperature should not become too hot to touch (about 65 deg. c). High motor temperature indicates the "LO" pot is set too high. This will cause damage to both the motor and the MCU-2 output transistors. Allow unit to sit powered up for 20 minutes to check motor temperature. It is important to note that low voltage motors (typ 1.2 to 5.0 volts) require less "LO" and "HI" pot adjustment. They also require more careful adjustment as their low winding impedance can result in higher MCU-2 output and motor currents.

4.12 Repeat step 4.11 for 2nd axis.

4.13 Set automatic mode.

Enter your top rate into data block 00, i.e. execute command #aaV00=r. Set manual mode. Using one of the slew pushbuttons, ramp motor to the top speed and adjust "HI" pot from maximum CCW to a position that allows the motor to run smoothly. This should occur at one-half to three-quarters potentiometer setting. Run the motor for 20 minutes at the slew rate to ensure that the motor does not overheat. Readjust the "HI" and "LO" controls a small amount for smooth ramping. Do not turn up the "HI" pot to compensate for too high a ramp setting. Motor and load inertia may mandate a slower ramp rate.

4.14 Repeat step 4.13 for 2nd axis.

4.15 If the motor falls out of sync, i.e. stalls, it may be necessary to change ramp data. To do this, you must choose a lesser value for #aaR00. You may also want to ensure that an extra margin of torque exists for the high speed setting. Slew at maximum speed with motor load in place. Using a gloved hand, carefully apply an additional load to the motor shaft. If the motor immediately stalls, you can advance the "HI" torque setting a small
amount until additional motor shaft load is required to stall motor.

4.2 Manual Mode of Operation

In manual mode of operation the front panel pushbuttons are operative. The unit does not accept serial port motion execute commands.

Pushing “CLEAR” pushbutton sets position counter and display to zero.

Short push of “JOG” pushbutton executes single step. Holding “JOG” pushbutton depressed motor executes jog motion - constant low speed of operation, no ramping.

Holding “SLEW” pushbutton depressed motor goes into slew motion - starts at base speed, ramps up to slew speed.

Jog, base and slew speed are to be preprogrammed using appropriate data enter instructions.

4.3 Automatic Mode of Operation

Host computer can be simulated by RS-232 Terminal; either hand-held CRT. It must be set to correspond to communications parameters mode, with internally generated “line feed” after receiving carriage return character. This enables the operator to see all the control messages and responses.

Communication on RS-485 interconnect link is half-duplex, therefore, communication on RS-232 line is simplex (only one direction at a time). The control program must operate on master/slave principle, when any of the MCU-2 units are responding, Pin 5 of RS-232 connector goes LO.

In Normal Auto Mode all MCU-2s are in receive mode. When host sends a message it is received and analyzed by all units. At that time the units are not able to receive next message. The host program should insert approximately 50mSec delay between consecutive commands, or wait for a response.