INSTALLATION INSTRUCTIONS for SLO-SYN® SS2000I PROGRAMMABLE MOTION CONTROL

Codes added or modified by Revision B

New Codes Per Revision B

G19 Branch On Result
G21 Register Control
G55 Error Trapping
H44 Transfer Register Contents

Modified Codes Per Revision B

L01 Message #1
L02 Message #2
L03 Message #3
L04 Message #4
L05 Message #5
L20 System Configuration
G20 Branch On Input Condition
G22 Wait For Input Condition
G29 Set Designated L Code
G38 Read Feedrate
G47 Set Outputs
G51 Branch On Flag Condition
G52 Select External Data Source
G53 Select S2000 Interface I/O Port
G54 Register Execution
G60 Enable Continuous Execution Mode

Codes added or modified by Revision C

New Codes Per Revision C

G06 Hcode execution

Modified Codes Per Revision C

L20 System Configuration
G21 Register Control

CONTENTS

USING THIS MANUAL

INTRODUCTION

PRODUCT FEATURES

The SLO-SYN SS2000I is a high-performance programmable motion control (indexer) designed specifically for use with SLO-SYN SS2000D6 step motor drives. It will also work with most standard "pulse and direction" type drives, including digital servo drives. The unit can control one or two drives operating one at a time. This feature is useful in many applications, such as x-y tables for pin insertion or pick-and-place operations. Closed-loop operation is available as an easily-installed option via a front-panel plug-in card. This card also provides the user of open- or closed-loop systems with a way to store programs, via battery-backed memory.

Programming is accomplished via either of the two serial interfaces. Both RS-232 and RS-485 connectors are provided on the front panel. The programming language is a sub-set of the RS-274 CNC language standard (X, F, and G codes). Up to 99 SLO-SYN 2000I motion controls can be easily daisy-chained using their serial ports.

The SLO-SYN SS2000I inputs and outputs (I/O) feature:

- Single-point I/O with removable terminal strips
- Optical isolation, with built-in 24 Vdc 0.75 A power supply
- Sinking or sourcing (switch selectable)
- Software select for positive or negative logic
- Inputs: 13 programmable, 3 dedicated (high-level interrupt)
- Outputs: 8 programmable, with short-circuit protection
- Optional BCD switches (2 seven-bit and sign) for move data accessible via top-mounted 25-pin "D" connector
- Optional remote I/O expansion: 8 inputs and 8 outputs of "OPTO 22" type accessed via the BCD connector

Among the other high-performance capabilities of the SLO-SYN SS2000I programmable motion control are such features as:

- Mark registration ("index from run") repeatable to ± one pulse
- Breakpoint programming that allows changing velocity as a function of position, time, or the state of an input
- Outputs can be set and inputs can be read as a function of velocity, position, or time
- Subroutines and loops nestable to four levels
- Message display capability to user display via serial port
- Fast mechanical homing cycle
- Optional closed loop (factory supplied or field installed)

In addition the SLO-SYN SS2000I incorporates a Fault LED, on the front panel, which indicates when fault conditions exist. The conditions which are indicated by the Fault LED are as follows:

- Drive not ready
- IFR limit exceeded
- CW software limit active
- CCW software limit active
- CW hardware limit active
- CCW hardware limit active
- Program error encountered
- Closed loop error

CAUTIONS AND WARNINGS

- High voltages are present inside this unit. An electrical shock hazard exists that may cause serious injury or death if this unit is operated without its protective covers in place.

- Do not exceed the voltage or current ratings of the various inputs and outputs. Please read the electrical specifications.

- The 24-volt dc power supply is limited to a total current output of 0.7 amperes. Do not exceed this level or the Control may shut down or work erratically as the power supply's current limit operates to protect the unit from overload.

- Be sure to mount the unit so there is adequate space around it for proper cooling airflow.

- Please follow good wiring practices and keep low-level signal lines away from power and motor wiring; this will help to avoid electrical noise interference problems.

LOGIC AND VOLTAGE CONVENTIONS

- Motor rotation direction (CW and CCW) is properly oriented when viewing the motor from the end opposite the mounting flange.

- Please refer to the Glossary section of this manual for detailed descriptions of terms such as "sink and source I/O", and "positive and negative logic".

ASSUMPTIONS

What the user needs to know to properly apply this product.

This instruction manual is written in a simple and easy-to-follow format that should be suitable for both new and experienced motion control users. In order to get the most out of your Slo-Syn 2000 Programmable Motion Control, we assume the user will be knowledgeable in the following areas:

1. Basic electrical and electronic skills, including preparing and following an equipment wiring diagram.

2. The basics of motion control system application, such as torque, speed, move distances, and how to structure a motion task into move segments and input/output control.

3. Some familiarity with elementary computer programming, including defining the problem to be solved and coding it in a computer language.

QUICK STARTUP GUIDE

We strongly recommend that the user read and understand the details and specifications found in later sections of this manual before applying this product. However, listed below are the minimum steps necessary to get up and running, with references to the appropriate instruction manual sections where further details can be found.

1. Set the switch settings for the serial communications parameters and the input/output polarity (sinking - sourcing). Refer to page 2-1.

2. Mount the unit in the location where it will be used.

3. Make all the appropriate electrical connections - wire the signal lines first, and finish by connecting the appropriate ac power. Refer to pages 2-2 through 2-15.

4. Program the unit for the application, then debug, troubleshoot, and correct any problems found.

Complete specifications are found on pages 9-2 through 9-5, and a software reference guide is also provided (see Contents for specific page numbers).

Superior Electric also provides a programming aid, MS-2000, that runs on personal computers to assist in developing motion programs for this Slo-Syn product.

NOTE: The Slo-Syn 2000 Programmable Motion Control **is not** compatible with the Superior Electric Micro Series programming pendant SSP-525, nor with the MS-1 Application Generator Program for personal computers.

SETUP AND INSTALLATION

SWITCH SETTINGS

Before mounting and installing the Programmable Motion Control, it is best to set the top-mounted switches that govern various operating features.

SERIAL COMMUNICATIONS PARAMETER SWITCHES

1. The DIP switch needs to be set to match the Control's baud rate and parity with that of the host computer or terminal to which it is connected. The factory default is 9600 baud, and no parity.

2. The DIP switch needs to be set to match the type of host (RS-232 or RS-485) to which the Control is connected. The factory default is for RS-232.

3. The Control is capable of being "daisy-chained", with up to 99 units connected to a single host. This is described in further detail on pages 2-4 and 2-5. Each unit in such a chain needs a unique identification number (ID #). This value is entered via the pair of ID # select switches on top of the unit. The unit is shipped with a factory default ID # of 01.

If needed, set the ID # select switches to a different value. Set the pointers on the two switches to the desired value of ID #, from 01 to 99.

INSTALLATION GUIDELINES

GENERAL WIRING GUIDELINES

SLO-SYN SS2000I controls use modern solid-state electronics to provide the features needed for advanced motion control applications. In some cases, these applications produce electromagnetic interference (EMI, or electrical "noise").

In general, any equipment that causes arcs or sparks or that switches voltage or current at high frequencies can cause interference. In addition, ac utility lines are often "polluted" with electrical noise from sources outside a user's control (such as equipment in the factory next door). Some of the more common causes of electrical interference are:

- power from the utility ac line
- relays, contactors and solenoids
- light dimmers
- arc welders
- motors and motor starters
- induction heaters
- radio controls or transmitters
- switch-mode power supplies
- computer-based equipment
- high frequency lighting equipment
- dc servo and stepper motors and drives

The following wiring practices should be used to reduce possible noise interference.

- Solid grounding of the system is essential. Be sure that there is a solid connection to the ac system earth ground. Bond the drive case to the system enclosure. Use a single-point grounding system for all related components of a system (a "hub and spokes" arrangement). Keep the ground connection short and direct.

- Even though the control has a built-in line filter, be sure to use a "clean" ac input line. Particularly bad ac lines may need to be conditioned with a ferroresonant type isolation transformer to provide "clean" power to the control.

- Keep signal and power wiring well separated. If possible, use separate conduit or ducts for each. If the wires must cross, they should do so at right angles to minimize coupling.

  Note: Power wiring includes ac wiring, motor wiring, etc. and signal wiring includes inputs and outputs (I/O), serial communications (RS232 lines), etc.

- Use shielded, twisted-pair cables for Control I/O lines. BE SURE TO GROUND SHIELDS ONLY AT ONE END.

Suppress all relays not connected to the control to prevent noise generation (the control's outputs have built-in relay suppression). Typical suppressors are capacitors or MOV's. (See manufacturer's literature for complete information). Whenever possible, use solid-state relays instead of mechanical contact types to minimize noise generation.

If you are experiencing problems which might be related to EMI, refer to the Troubleshooting section for pointers.

INPUT/OUTPUT POLARITY SWITCH

Your Programmable Motion Control offers the optimum in flexibility of using the inputs and outputs (I/O). A top-mounted switch allows selection of either "sinking" or "sourcing" modes for the I/O. In addition, a software command allows the I/O states to be interpreted by the Control program using either "positive logic" or "negative logic" conventions. (Note: All these terms are described in detail in the Glossary at the back of this Manual.)

WIRING THE CONTROL FOR OPERATION

The following pages show how wiring connections are made to the various connectors on the control. All of the terminal strip connectors have their terminals clearly labeled. Please see the SPECIFICATION section for the pin-outs of the "D" connectors for the RS-232, BCD/OPTO-22, and optional Encoder interfaces.

Also included in this section are equivalent circuits for the inputs and outputs, operating in both sinking and sourcing modes.

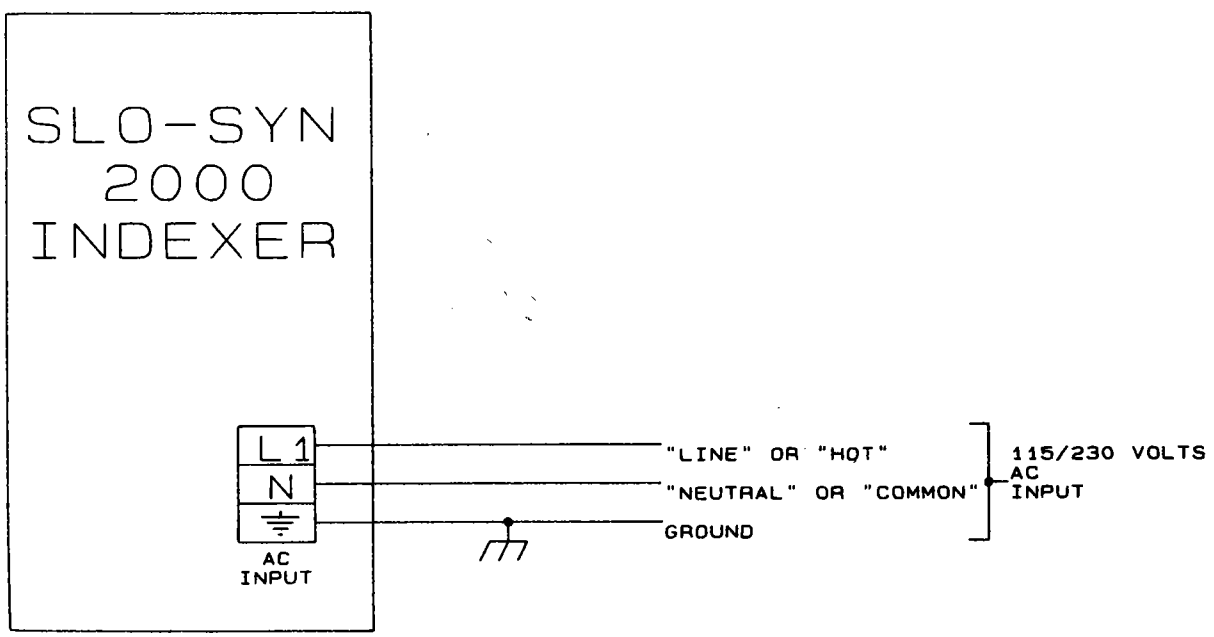
Further details on the use of the programmable inputs and outputs (IN1-IN13 and OUT1-OUT8) can be found in the Programming and Operating section and the Software Reference section of this manual.

Please observe the ratings of the ac input and the various I/O circuits as listed in the Specification section of this manual. This will ensure proper and reliable operation of your control.

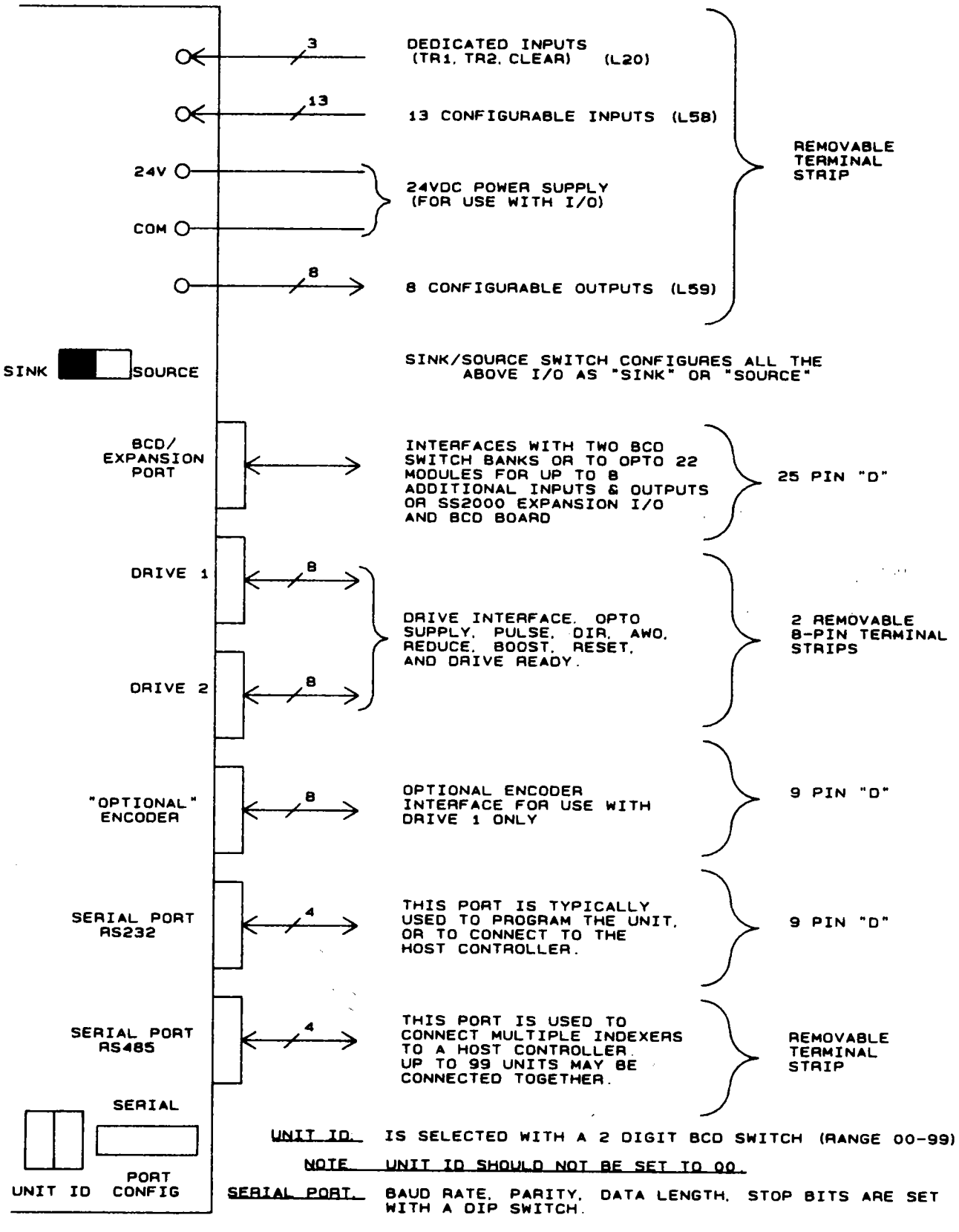
Power Input

The ac input is connected to a 3-screw terminal strip. The terminals are labeled as follows:

Terminal	Lead Color, North American Standard	Lead Color, European Standard (CEE)
"L1" for line or "Hot"	Black	Brown
"N" for Common or Neutral	White	Blue
"⊥" for Ground	Green	Green with Yellow Stripe

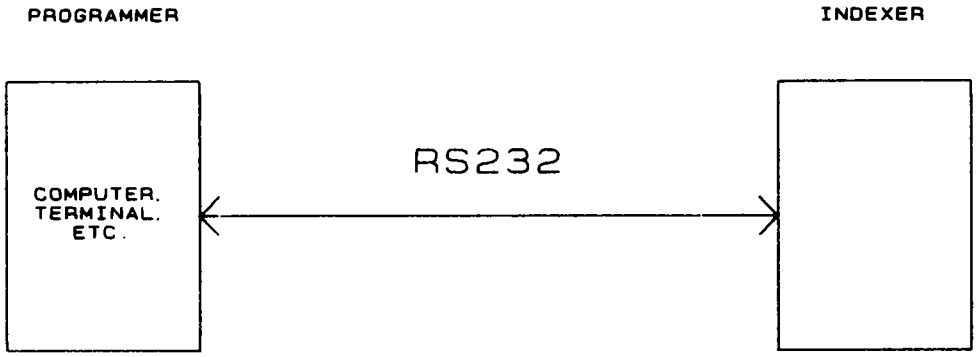


CONTROL I/O CONNECTIONS

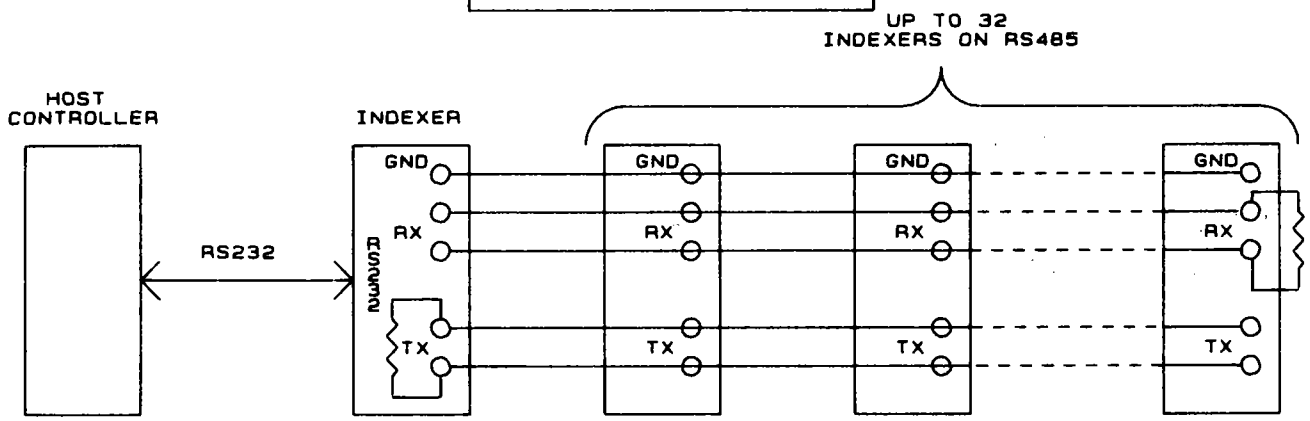


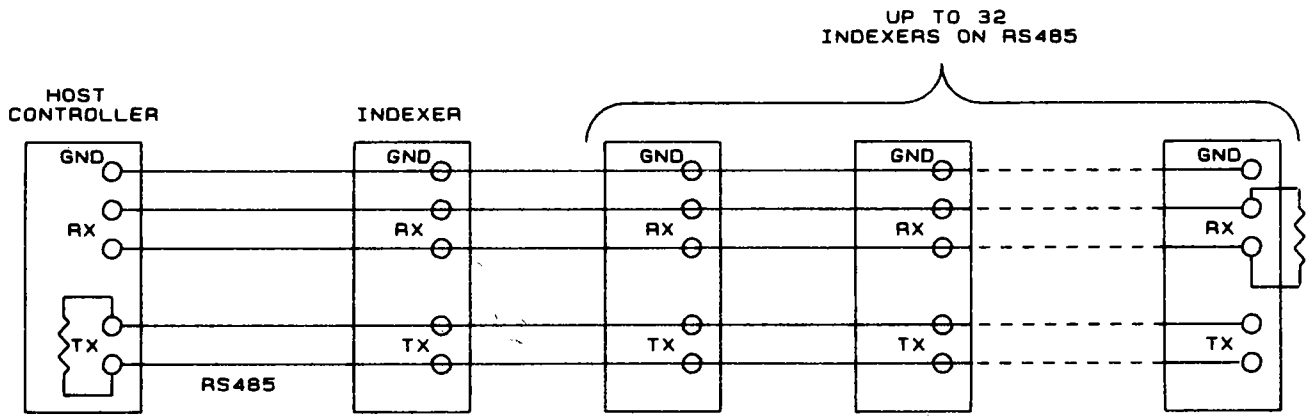
SERIAL PORT

PROGRAMMING



HOST CONTROL



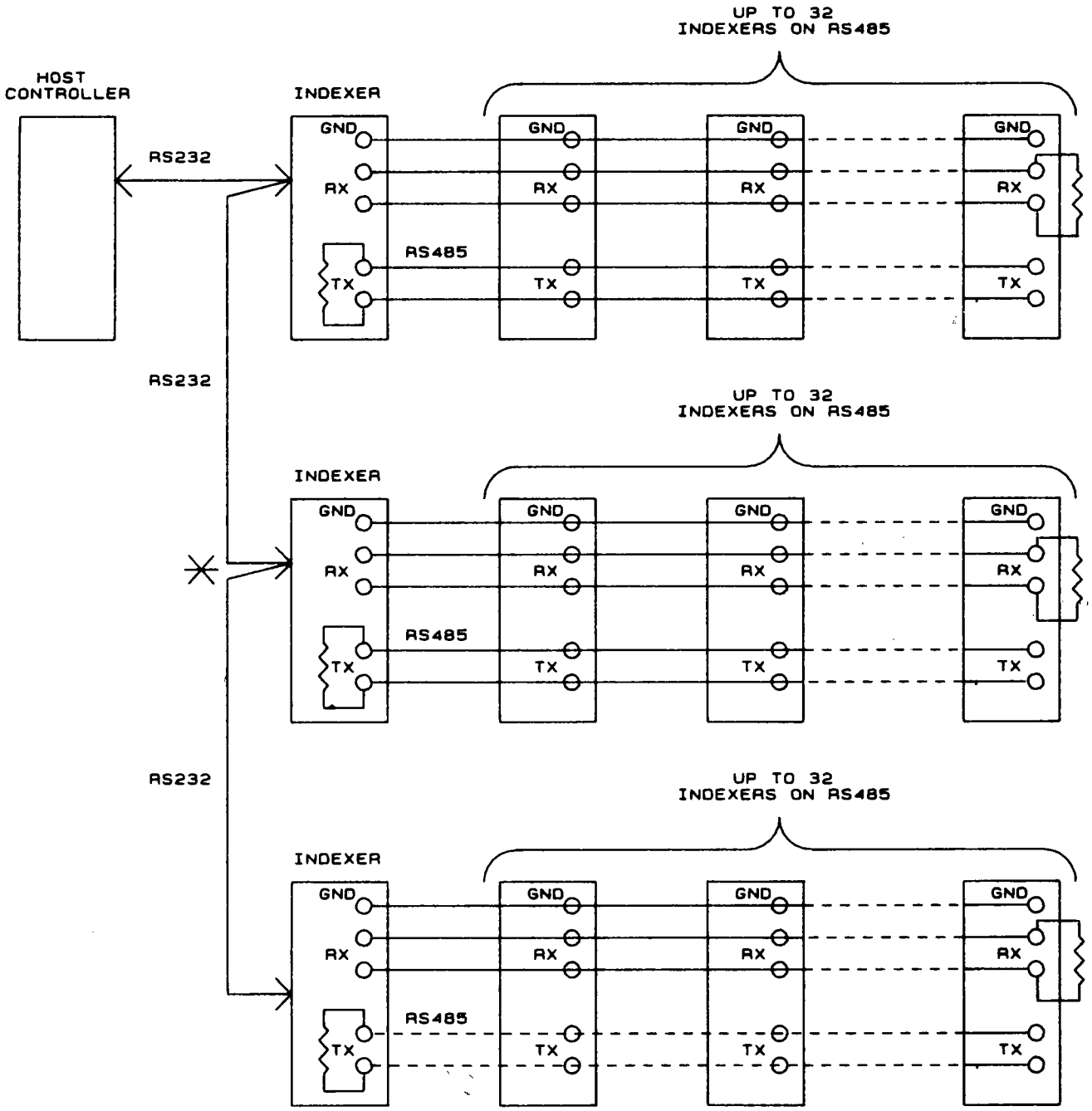


NOTES:

1. HOST COMMUNICATIONS TO FIRST INDEXER USE RS232.
2. REMAINING UNITS COMMUNICATE VIA RS485.
3. ALL UNITS MUST HAVE DISTINCT IDS.
4. AT THE LAST UNIT IN A CHAIN, USER MUST SUPPLY A 120 OHM 1/4 WATT TERMINATION RESISTOR.

HOST CONTROL

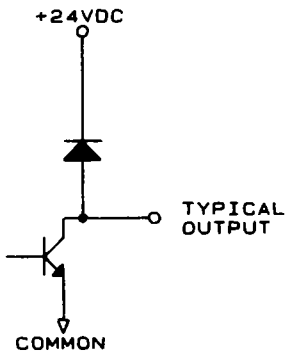
(UP TO 99 INDEXERS)



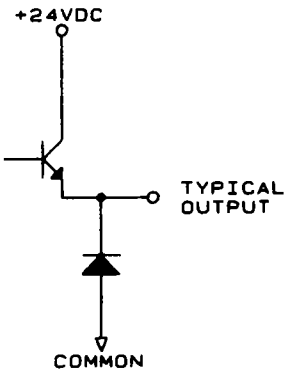
NOTES
1. HOST COMMUNICATIONS TO THREE LEFTMOST INDEXERS USE RS232. REMAINING UNITS COMMUNICATE VIA RS485.
2. ALL INDEXER IDS MUST BE DISTINCT.
3. AT THE LAST UNIT IN A CHAIN, USER MUST SUPPLY A 120 OHM 1/4 WATT TERMINATION RESISTOR, CONNECTED AS SHOWN ABOVE.

✳ USE RS232 ADAPTER "Y" CABLE, S.E., P/N 216298-002

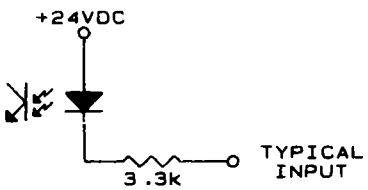
INPUT/OUTPUT EQUIVALENT CIRCUITS



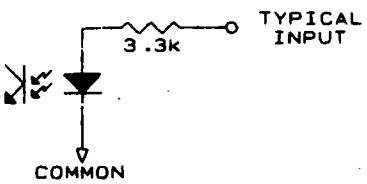
SINK OUTPUT



SOURCE OUTPUT



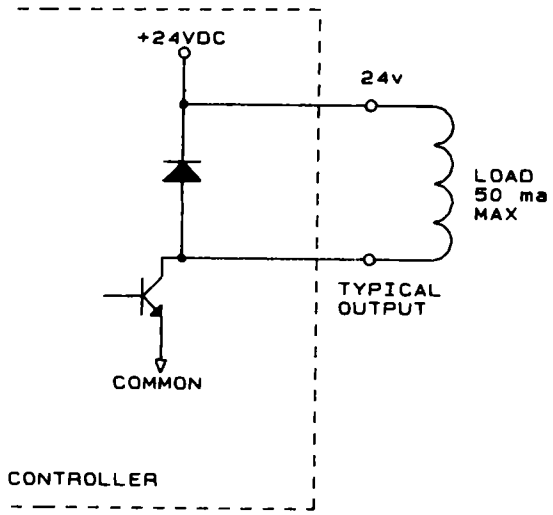
SINK INPUT



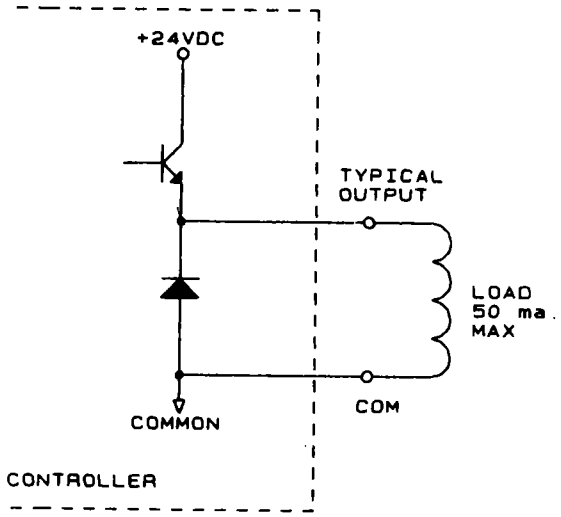
SOURCE INPUT

NOTE ALL INPUTS AND OUTPUTS ARE CONFIGURED TOGETHER FOR SINK OR SOURCE VIA THE SELECTOR SWITCH LABELED "I/O (SINK-SOURCE)" ON TOP OF THE CONTROLLER

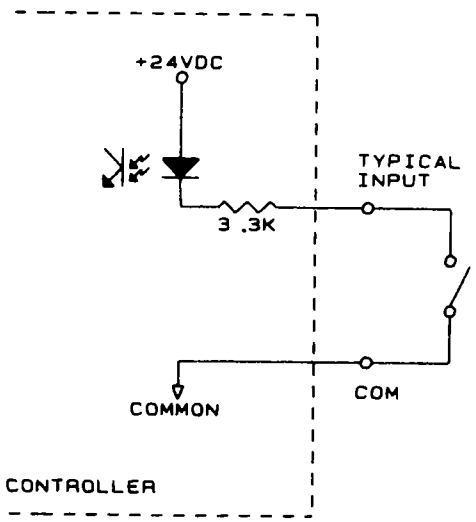
TYPICAL I/O CONNECTIONS



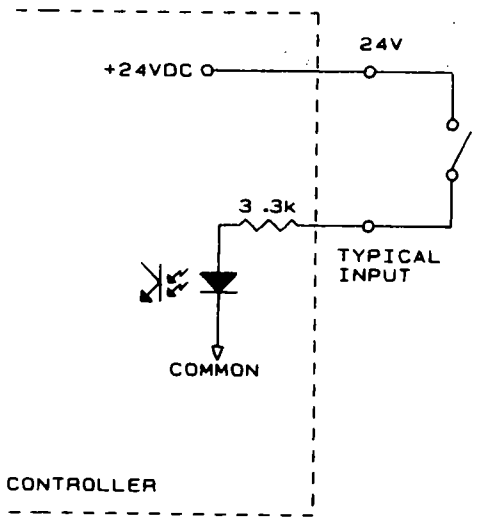
SINK OUTPUT



SOURCE OUTPUT



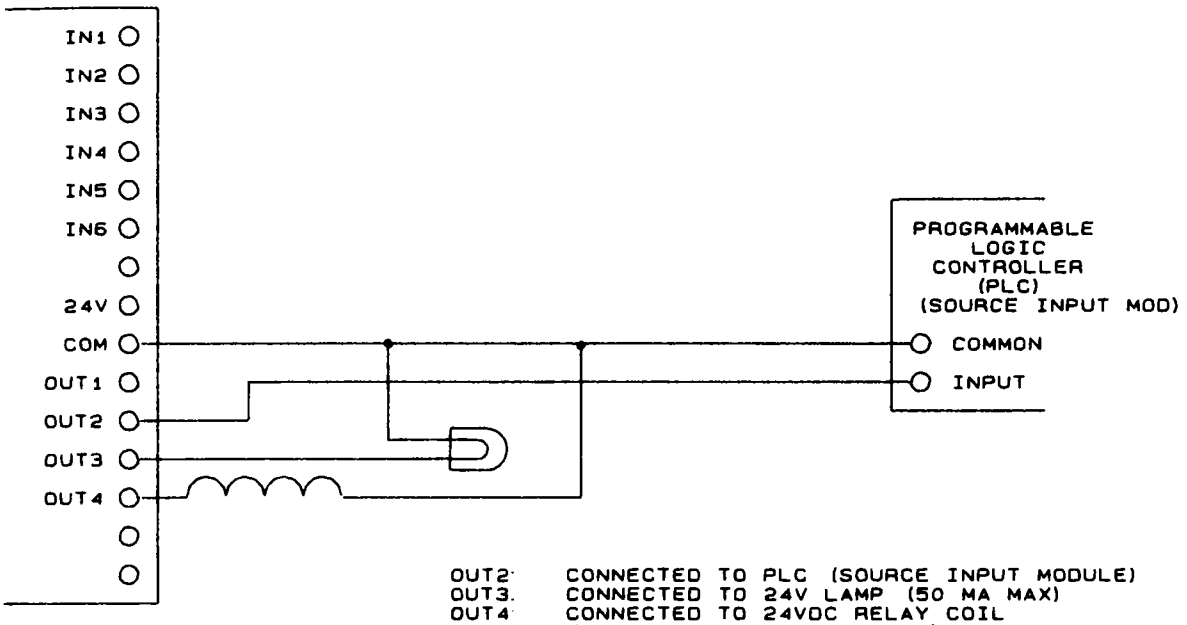
SINK INPUT



SOURCE INPUT

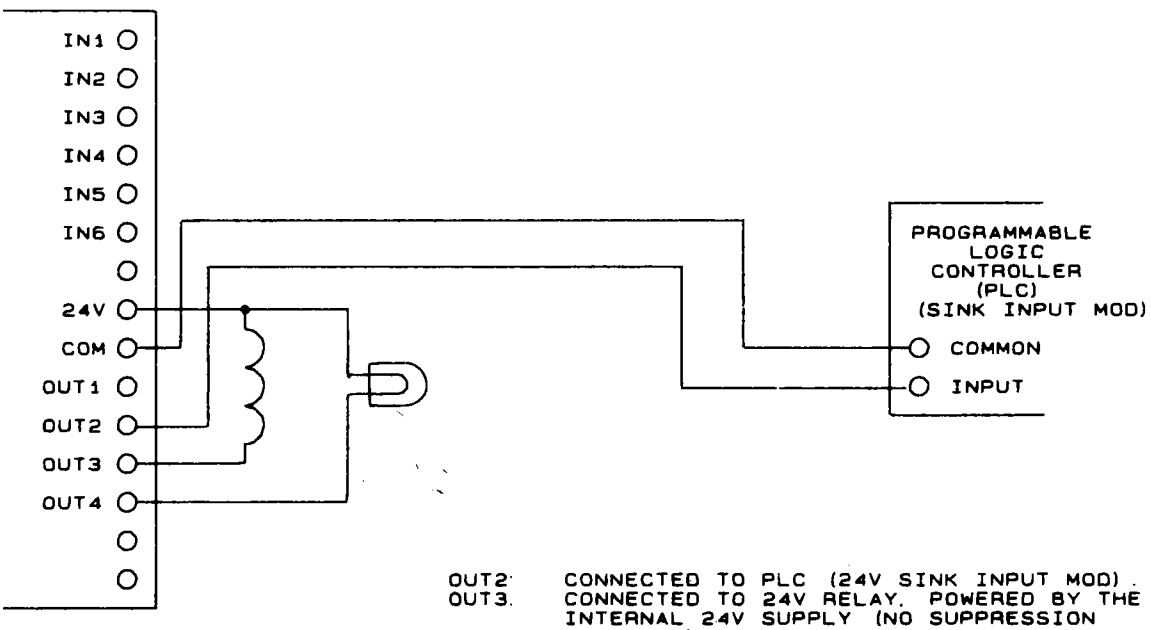
NOTES:
1. +24VDC AND COMMON ARE THE POWER SUPPLY OUTPUT AND COMMON TERMINALS, RESPECTIVELY.
2. TYPICAL OUTPUTS ARE OUT1-OUT8.
3. TYPICAL INPUTS ARE IN1-IN13, TR1, TR2, AND CLEAR.
4. ALL I/O'S ARE CONFIGURED TOGETHER FOR SINK OR SOURCE MODE VIA THE SWITCH ON TOP OF THE CONTROLLER LABELED "I/O (SINK-SOURCE)".

SOURCE OUTPUTS



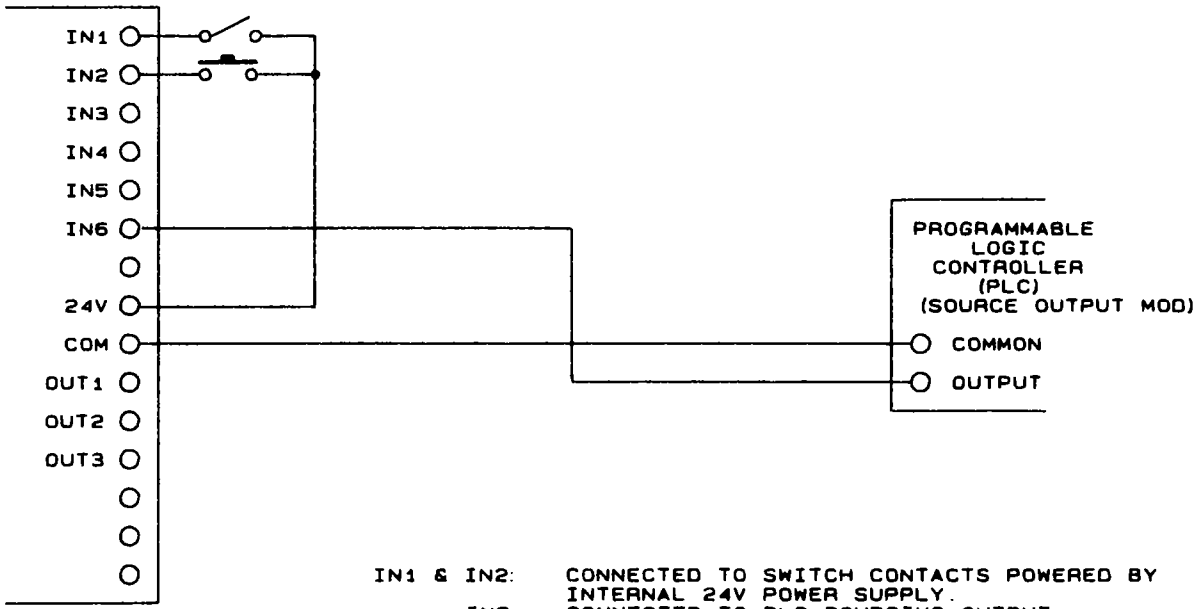
OUT2 CONNECTED TO PLC (SOURCE INPUT MODULE)
OUT3 CONNECTED TO 24V LAMP (50 MA MAX)
OUT4 CONNECTED TO 24VDC RELAY COIL (NO SUPPRESSION REQUIRED)

SINK OUTPUTS



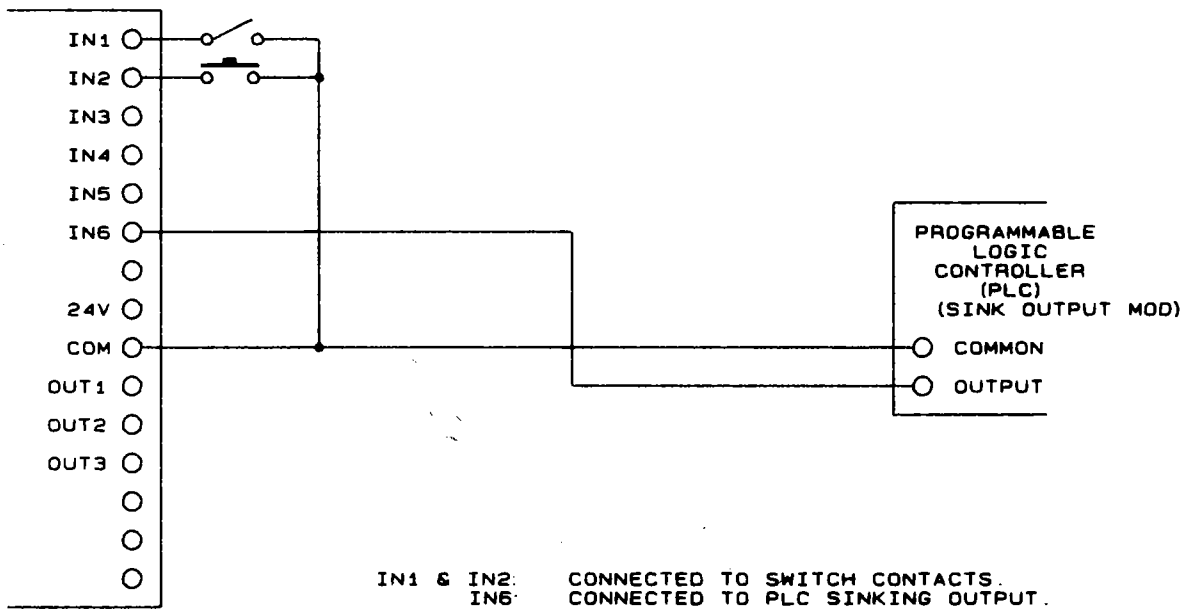
OUT2 CONNECTED TO PLC (24V SINK INPUT MOD).
OUT3 CONNECTED TO 24V RELAY, POWERED BY THE INTERNAL 24V SUPPLY (NO SUPPRESSION REQUIRED)
OUT4 CONNECTED TO 24V LAMP, POWERED BY THE INTERNAL 24V SUPPLY.

SOURCE INPUTS



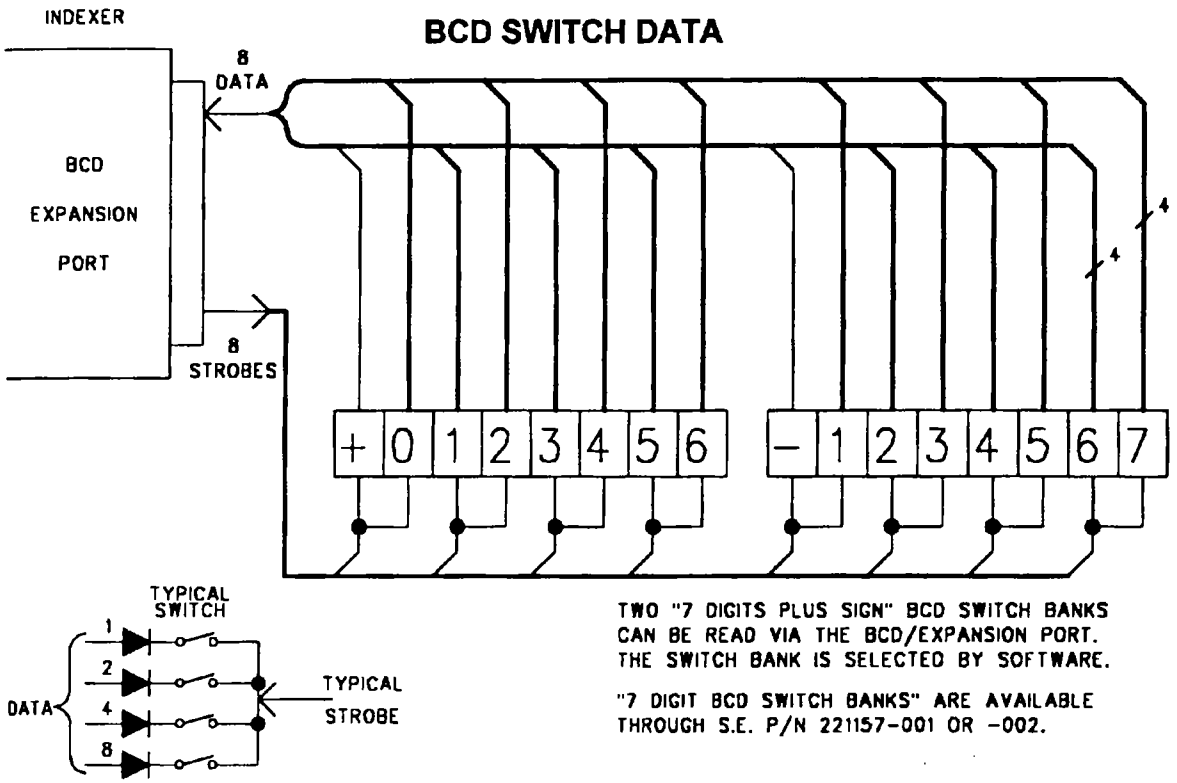
IN1 & IN2: CONNECTED TO SWITCH CONTACTS POWERED BY INTERNAL 24V POWER SUPPLY.
IN6: CONNECTED TO PLC SOURCING OUTPUT.

SINK INPUTS



IN1 & IN2: CONNECTED TO SWITCH CONTACTS.
IN6: CONNECTED TO PLC SINKING OUTPUT.

BCD SWITCH DATA



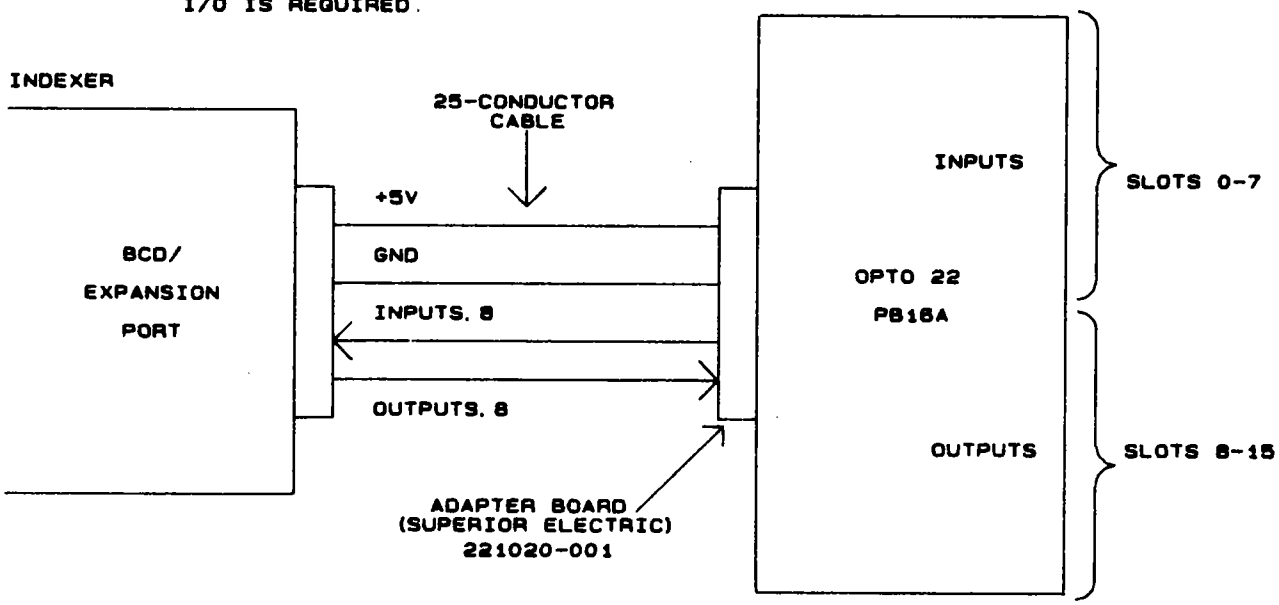
CONNECTOR PIN-OUTS

Indexer 25-pin D Connector Pin #	BCD Switch Function & Signal Name	
1		DATA 1
2		DATA 2
3		DATA 4
4		DATA 8
5		DATA 10
6		DATA 20
7		DATA 40
8		* DATA 80 or -SIGN
9-11		NC
12, 13		NOT USED
14	BCD BANK 1	STROBE X1
15		STROBE X100
16		STROBE X10,000
17		STROBE X1,000,000
18	BCD BANK 2	STROBE X1
19		STROBE X100
20		STROBE X10,000
21		STROBE X1,000,000
22, 23		NC
24, 25		NOT USED

* SIGN FOR STROBE X1,000,000; OTHERWISE DATA 80.

EXPANSION I/O

THE BCD/EXPANSION PORT CAN READILY INTERFACE TO STANDARD "OPTO 22" STYLE INPUT & OUTPUT MODULES WHEN HIGH POWER I/O IS REQUIRED.



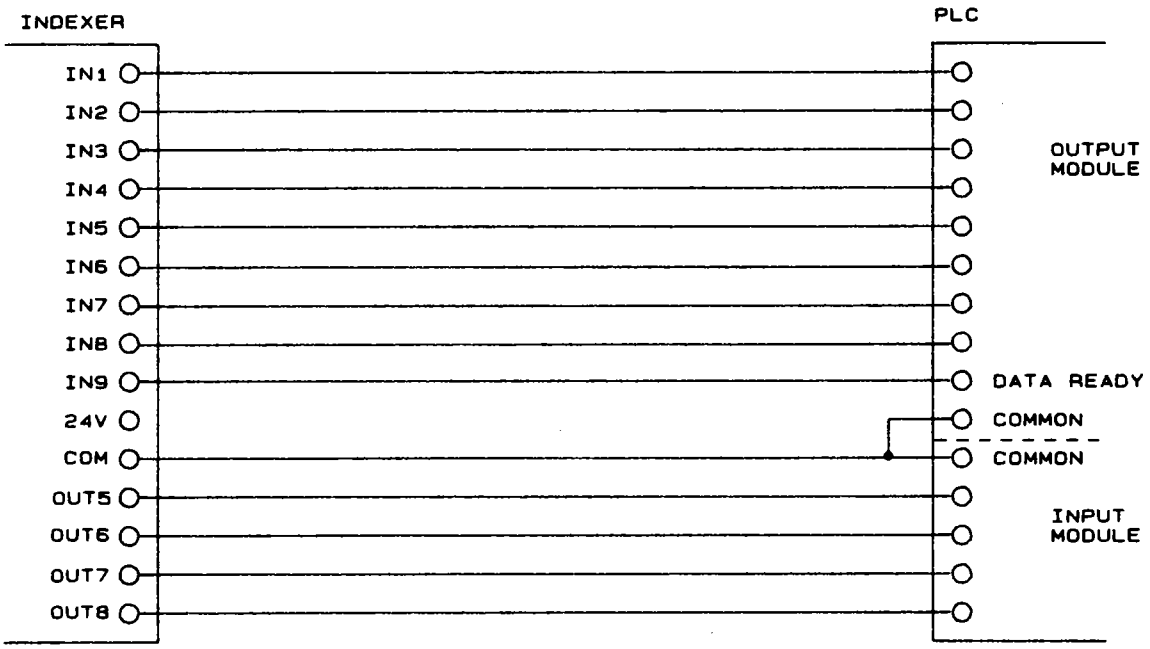
THE FOLLOWING LIMITATIONS APPLY:

MAXIMUM NUMBER OF INPUT MODULES IS 8.
MAXIMUM NUMBER OF OUTPUT MODULES IS 8.
SUPERIOR ELECTRIC SUPPLIED CABLING WILL RESTRICT INPUTS TO SLOTS 0-7 & OUTPUTS TO SLOTS 8-15.

CONNECTOR PIN-OUTS

INDEXER	SIGNAL NAME	OPTO 22
1	INPUT 0	47
2	INPUT 1	45
3	INPUT 2	43
4	INPUT 3	41
5	INPUT 4	39
6	INPUT 5	37
7	INPUT 6	35
8	INPUT 7	33
9-11	NC	NC
12, 13	+5V	49
14	OUTPUT 0	31
15	OUTPUT 1	29
16	OUTPUT 2	27
17	OUTPUT 3	25
18	OUTPUT 4	23
19	OUTPUT 5	21
20	OUTPUT 6	19
21	OUTPUT 7	17
22, 23	NC	NC
24, 25	GND	24

BCD DATA FROM PLC



READING BCD DATA FROM A PLC USES 4 OUTPUTS AND 9 INPUTS.

4 OUTPUTS [OUT1-OUT4] CONFIGURED AS STROBES.
8 INPUTS [IN1-IN8] CONFIGURED AS BCD DATA.
1 INPUT [IN9] CONFIGURED AS DATA READY.

I/O REMAINING: 4 INPUTS AND 4 OUTPUTS PLUS ANY EXPANSION I/O ON CONTROLLER, AND ANY SPARE PLC I/O

HOW I/O SINK VS. SOURCE POLARITY AND LOGIC CONVENTION (POSITIVE OR NEGATIVE LOGIC) AFFECT THE USE OF G22 AND G47 COMMANDS

The following pages show the way the control interprets the G22 (Wait for Input) and G47 (Set Output) commands, based on the L58 (Logic Convention, positive or negative) value and whether the inputs and outputs (I/O) are configured for sink or source mode.

Further details can be found in the Software Reference section of this manual, and the terms are explained in the Glossary.

INPUTS AND OUTPUTS AFFECTED BY LOGIC CONVENTION

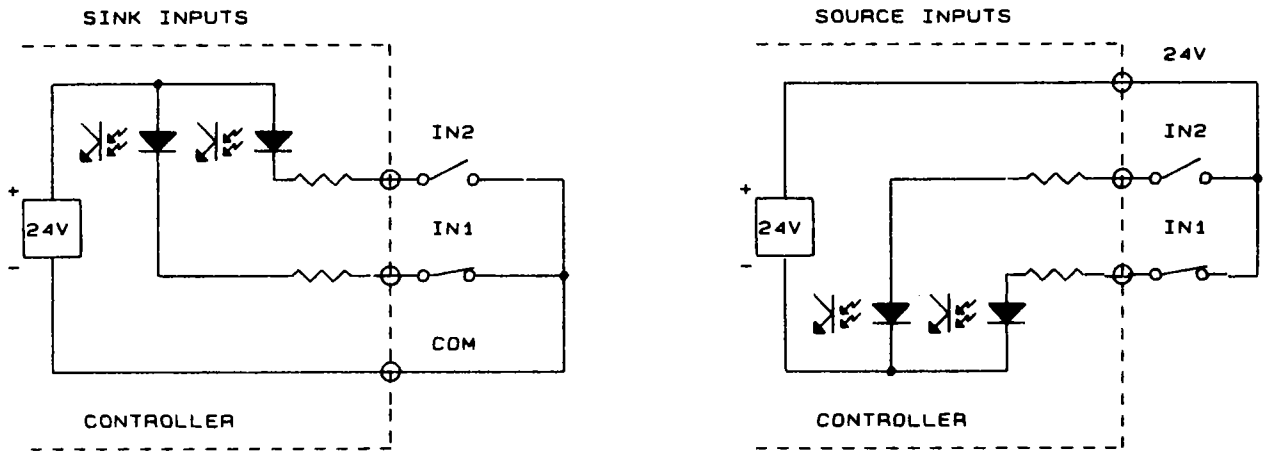
Input or Output	Positive Logic (L58 +)	Negative Logic (L58 -)
IN 1 - IN 13	Switch closed = 1	Switch closed = 0
OUT 1 - OUT 8	1 = output on	0 = output on
Expansion port configured for OPTO-22 I/O	Inputs: input active = 1	Inputs: input active = 0
	Outputs: 1 = output active	Outputs: 0 = output active
TR1, TR2, CLR inputs	Switch closed = 1	Switch closed = 0
Expansion port configured for BCD switches *	Switch closed = 1	Switch closed = 1
Fault LED *	LED on = fault	LED on = fault
Motion busy LED *	LED on = motion occurring	LED on = motion occurring

* Not affected by logic convention (L58 setting).

L58+

(POSITIVE LOGIC)

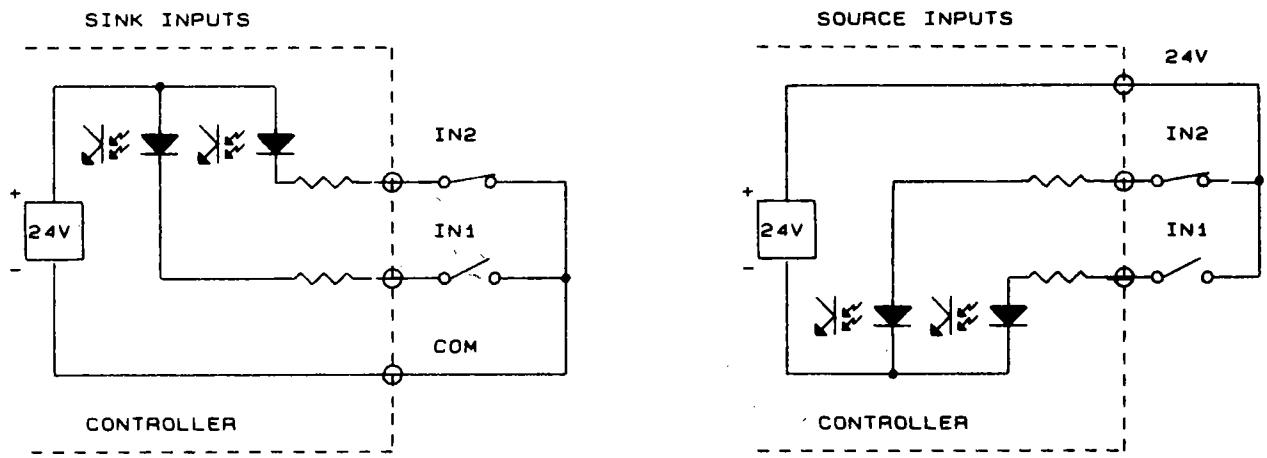
G22 X22222201 WAIT FOR
INPUT1=ON AND
INPUT2=OFF
(REMAINING INPUTS = "DON'T CARE")



L58-

(NEGATIVE LOGIC)

G22 X22222201 WAIT FOR
INPUT1=OFF AND
INPUT2=ON
(REMAINING INPUTS = "DON'T CARE")



EFFECT OF SINK/SOURCE & LOGIC CONVENTION ON G22 COMMAND (WAIT FOR INPUT CONDITION)

NOTE: Illustrations show switch positions needed to satisfy G22 examples given.

L58+

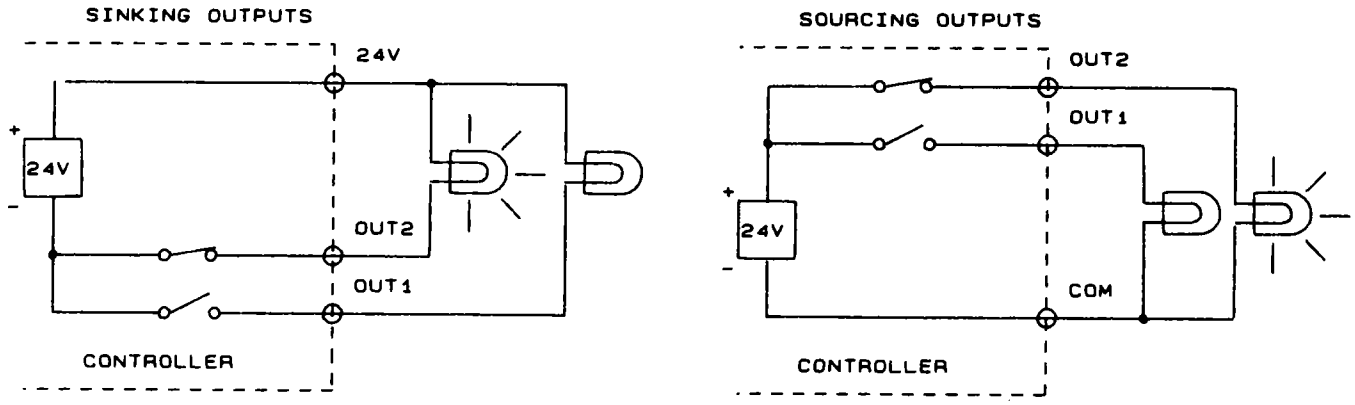
(POSITIVE LOGIC)

G47 X22222210

OUTPUT1=OFF

OUTPUT2=ON

(REMAINING OUTPUTS UNCHANGED)



L58-

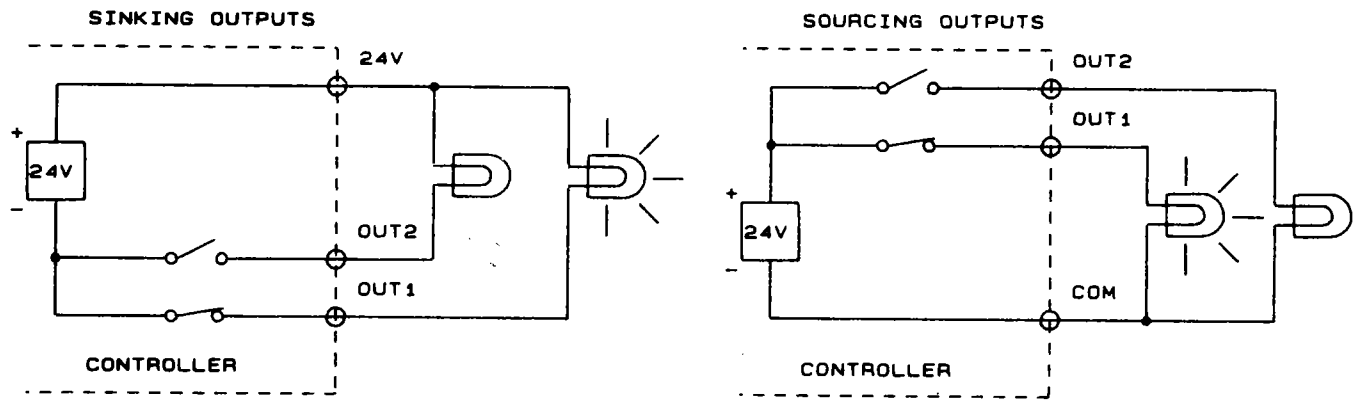
(NEGATIVE LOGIC)

G47 X22222210

OUTPUT1=ON

OUTPUT2=OFF

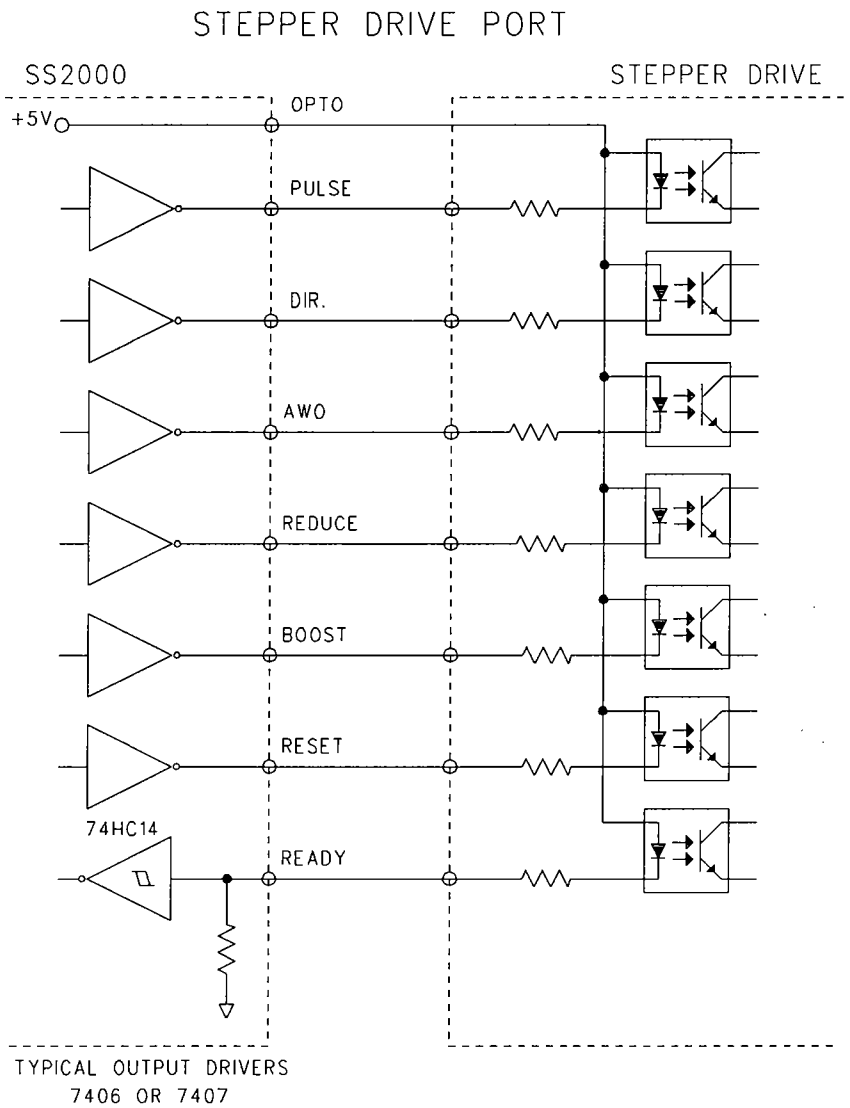
(REMAINING OUTPUTS UNCHANGED)



EFFECT OF SINK/SOURCE & LOGIC CONVENTION ON G47 COMMAND (SET OUTPUT CONDITION)

NOTE: Illustrations show output states resulting from G47 examples given.

DRIVE INTERFACE



NOTE: When connecting drives without a READY signal to the SS2000I connect the OPTO output to the READY line. The SS2000I will not operate unless the READY signal is activated (5 Vdc input).

This page intentionally left blank

PROGRAMMING AND OPERATING GUIDE

SERIAL COMMUNICATIONS ISSUES

GENERAL INFORMATION

EIA (Electronic Industries Association) Standard RS274-D, the programming guide for numerically controlled machines, has been used as the basis for the Slo-Syn 2000 Programmable Motion Control command structure. Although it was not technically feasible to conform to this standard in complete detail, it proved beneficial in the program structure to permit complex and varied operations to be performed using a simple format.

Communications between the Control and a host computer or terminal used for programming and operation is via either of two serial data interfaces; both RS-232 and RS-485 connectors are provided. The serial communications parameters and commands can be grouped into three general categories:

1. **"L Codes"** are used to set parameters for each Control. These commands do not cause motion, but simply establish the motion parameters.

2. **"H Codes"** are used to set Control modes, to control manual and program execution and to transmit parameters and Control status via the serial communications port.

3. **Line Data Codes** are the codes that define the motion that will be made (X Code or Y Code), the speed at which it will be done (F Code) and G Code which can control program execution and modify the way it operates.

BUFFERING OF COMMANDS

A Slo-Syn 2000 Programmable Motion Control has two buffers dedicated to RS232 communications. Each buffer holds 255 characters. One buffer holds commands which will be processed only when the Control is not "Busy" (when a cycle is in process or when previously transmitted commands are being processed). These commands are referred to as "Standard" commands and include all L codes, H codes and Feed Rate codes. They are processed in the order in which they are received, and each command must be completed before the next command is processed.

The second buffer holds commands which will be processed as soon as they are issued, even when the Control is "Busy". These commands are known as "Immediate" commands and include only a subset of the H codes and Line Data codes. To indicate that a command should be treated as an immediate command, simply precede the command with an exclamation mark (!). A list of immediate commands is given in Appendix C. Immediate commands are processed in the order in which they are received, as are standard commands.

Multiple commands can be transmitted to the Control at the same time. The Carriage Return (ASCII 13) and/or Line Feed (ASCII 10) will terminate a string of commands. If a string of immediate commands is to be sent, the "!" character need only be used in front of the first command. The Immediate Buffer selection is canceled when the Carriage Return or Line Feed is received.

The "Backspace And Delete" command, Control H (^H) can be used to delete a character in a buffer when data is being entered. In addition, the "Delete RS232 Buffer" command, Control X (^X), can be used to delete the last line entered in the active buffer.

The number of characters remaining in either the standard or the immediate character buffer may be requested at any time. To request the status of the standard buffer, send the "/" character (ASCII 47). Send the "\" character (ASCII 92) to request the status of the immediate buffer. Neither character is actually stored in the buffer. The response format from the Control is:

nnn CRLF

where nnn is the character count remaining.

Xon/Xoff PROTOCOL

Xon/Xoff is a serial communications protocol, executed in software, that allows communications between two devices without the need for additional hardware control. This method of controlling data flow requires only a three-wire connection between the devices. The protocol is used for controlling the flow of data between the Control and another device. **It should NOT be used to determine the status of the Control.**

The Xoff character (ASCII 19) is used to stop the transmission of RS232 characters. When one device sends an Xoff to the other device, it is telling the other to stop transmitting characters. The transmitting device should comply with the request.

The Xon character (ASCII 17) is used to resume transmission of the RS232 characters. When one device sends an Xon character to the other, it is signifying that it is ready to receive more data. If there is more data to be sent, the transmitting device may now resume transmission.

When the Control sends an Xoff command to the host, all activation commands (see below) and immediate H codes can still be processed.

Note: The Xon/Xoff protocol is only used when the L26 value is 0-3.

DEVICE IDENTIFICATION

In order to daisy chain multiple Controls to communicate with a single host, each Control must be given a unique identification. The Unit ID # select switch defines the identification number of the Control. This switch is interrogated on power turn on only. The factory setting is device 01. **Each Control must be given a unique identification before the system is wired.** The device addresses need not be consecutive, and the Controls can be placed in any positions in the chain regardless of their addresses.

SERIAL PORT CONFIGURATION

The Baudrate (9600,2400,1200 or 300), character length (7 or 8), Parity (Enable, Disable, Odd or Even) can be selected via a hardware dip switch. This switch is interrogated on power turn on only. The Factory setting is 9600 baud, 8 bit word length and parity disabled.

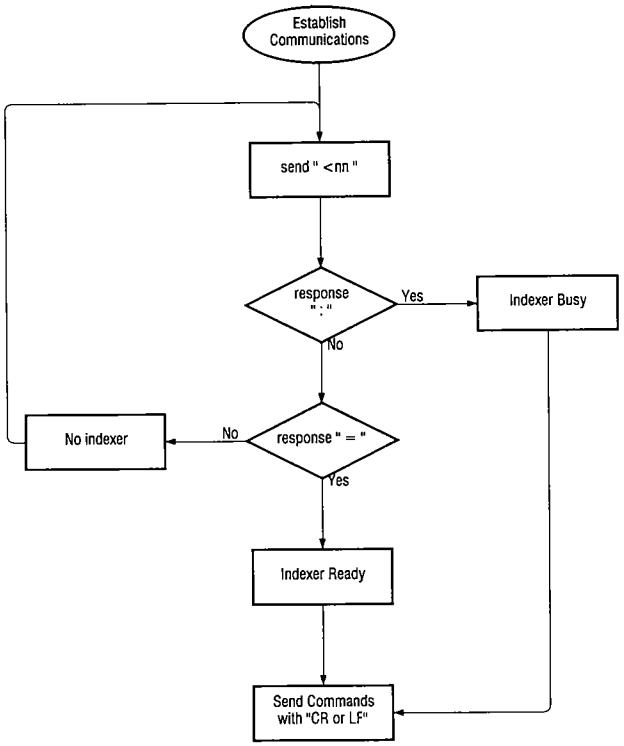
DEVICE ACTUATION

In order to accept commands from a host device, a Control must be set to the active mode. To do this, the host must send the device attention command (<) followed by the device identification number and, as always, a carriage return, line feed or a non-numeric character. For example, to activate device 01 send <01CR. The Control which has been activated will respond with an = if it is not busy, or with a : if it is busy. If you wish the Control to identify itself in the response, include a ? in the request (<01?). The response will be 01= if the Control is not busy, or 01: if it is busy. Device 01 will power up activated.

Note: If L26 = 0 thru 3, an Xon will be transmitted following the =. If L26 = 4 thru 7, no Xon will be transmitted.

All Controls can be placed in the listen mode only by issuing a <00CR from the host. No data can be transferred from the Control to the host in this mode. However, all other commands will honored by the individual controls.

If the host wishes to identify the active Control, the Device Acknowledgement command (?) can be issued. If the active device is not busy , it will respond with a nn=Xon, where nn is the device ID. If the active device is busy, it will respond with a nn:. If no device is active, no response will be received. (Xon will not be transmitted if the protocol is disabled.)



LISTEN MODE

In addition to the active mode, a Control can also be placed in the listen mode. To do this, the host must send the device attention command (<), followed by the device identification number, followed by an ampersand (&). For example, to place device 01 in the listen mode, send <01&CRLF. In response to the listen command, the Control will send a 01=Xon if it is not busy, or a 01: if it is busy. (Again, Xon will not be transmitted if the protocol is disabled.)

While in the listen mode, a Control will execute commands issued by the host. However, it will transfer data only in response to its device attention command for busy polling. All other transfer commands will be ignored while in the listen mode.

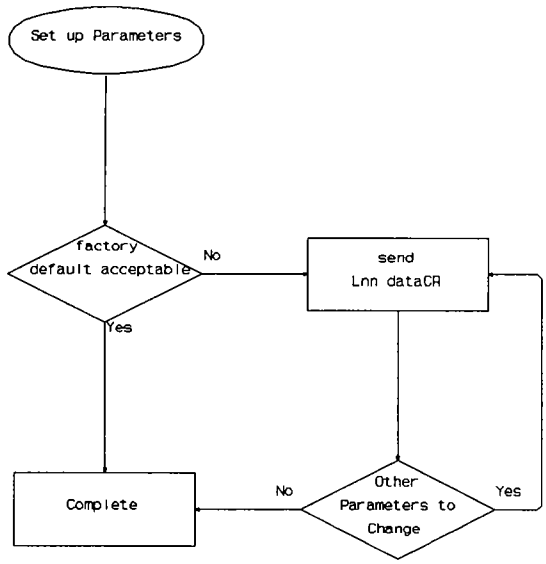
To cancel the listen mode, the host must transmit the device attention command (<), followed by the device attention character, followed by an @. For example, to cancel the listen mode for device 01, send <01@. This device will now become the active device.

All Controls can be placed in the listen mode by issuing a <00&CRLF from the host. To cancel this mode for all Controls, a <00@ can be sent from the host. When this is done, all Controls become active.

GENERAL PROGRAMMING COMMENTS

1. The first task that faces an operator is that of setting "L codes", the parameters of each Control.

2. It is important to note the **factory default values** that have been set for each parameter. These default values allow entry steps to be eliminated for those parameters where the factory default value is acceptable. (See the Appendix for values.)



3. If invalid data are entered for any parameter or field, the previous data will not be replaced.

4. If the number of characters entered for a parameter exceeds the number of characters required, the data are truncated to the correct length by eliminating the first characters sent. For example, if a parameter requires eight characters and ten are sent, only the last eight will be used.

5. In the command descriptions which follow, the field descriptions for each command use an "n" to designate a number and an "s" to designate a sign (+ or -). For example, the field designation "snnnnnnnn" designates that the field consists of a + or - sign followed by eight numbers

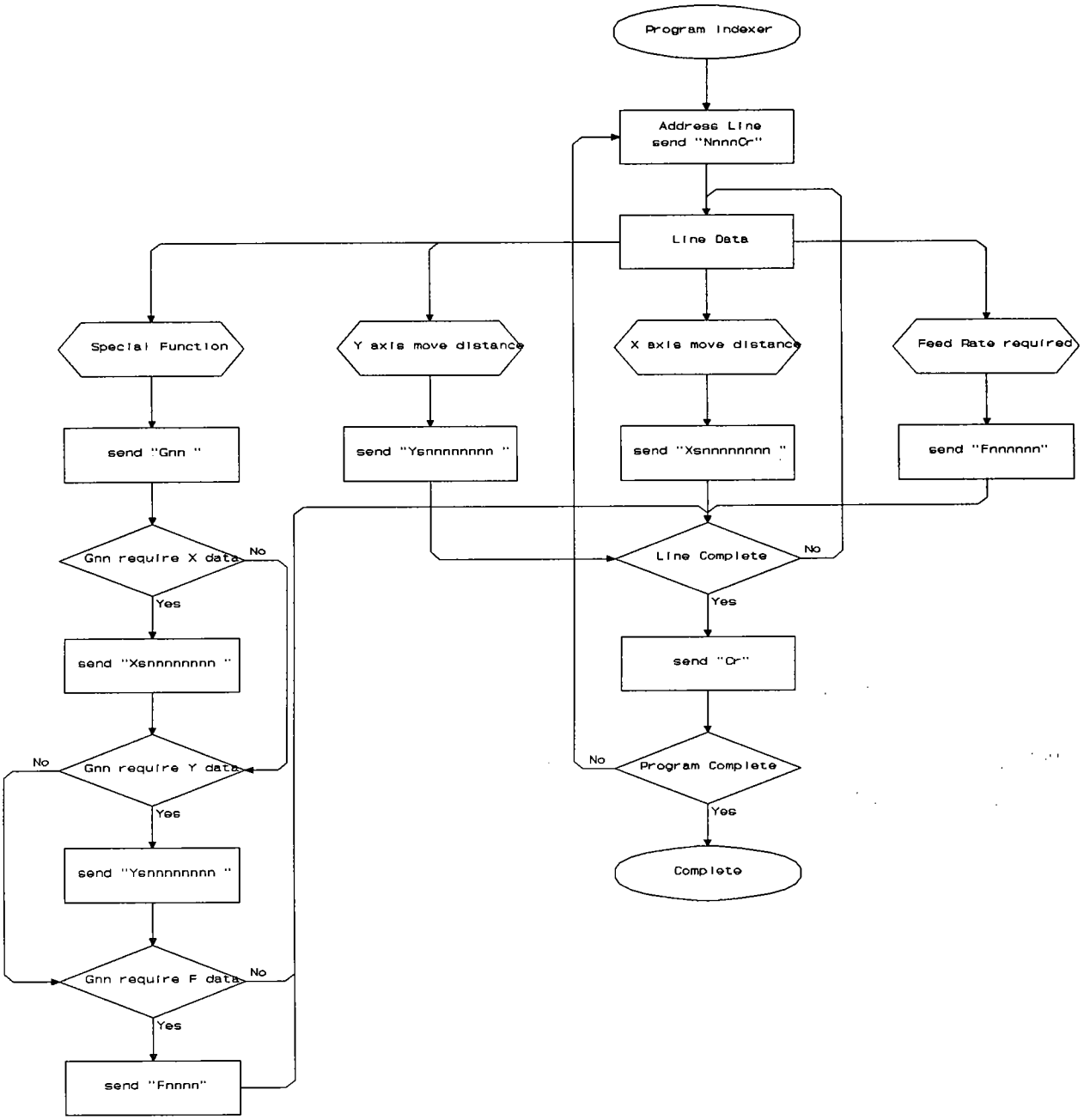
6. Some functions can be executed via switches connected to the programmable inputs. This allows the user to perform system set up through the serial port and then operate from a switch panel. **Note that if a parameter is changed over the serial port, it will override the switch setting on the panel.** For example, if the High Speed is selected on the panel and then the RS232 command for Low Speed mode is sent followed by a request for a CCW move from the panel, the move will be executed at Low Speed even though the panel switch is set at the high speed mode.

   Note also that switches are only honored if they are toggled (change state). Immediate H codes also override any switch settings.

7. When using any of the L, H and G codes, preceding zeros can be omitted. For example, H1CRLF will perform the same as H01CRLF.

8. If it is necessary to know when an index motion is complete, the Line Done Characters (L55 parameter) can be used. Note that if a Program Line Delay is programmed, the character(s) will not be transmitted until the delay has expired.

9. To enter a program line of code from your flow chart or pseudo code simply send the designated line number first, Nnnn ,. Now fill the contents of the line using the appropriate commands: Gnn for special functions, Xsnnnnnnnn or Ysnnnnnnn for move distance or G code field extension, Fnnnnnn for feed rate or G code field extension and Cr or Lf to load the line contents into non-volatile memory.



Notes: "Busy" means that the Control is executing a program or motion is occurring.

"Ready" means that motion has ceased and/or program execution is completed. The Control can now receive further information.

Immediate commands can be processed anytime.

NOTE: The Slo-Syn 2000 Programmable Motion Control **is not** compatible with the Superior Electric Micro Series programming pendant SSP-525, nor with the MS-1 Application Generator Program for personal computers.

This page intentionlly left blank.

PROGRAMMING

PROGRAMMING

This section provides an overview to the process of programming a Control. Once the "logic" behind the programming commands is understood, and the mix of numbers and letters explained, programming your Control will be seen as a straightforward process.

Programming of any sort requires planning and forethought. Programming your Control is no exception. This section will provide aids to facilitate your planning process.

Be patient! Allow time for adjustments ("debugging") and experimentation.

WHAT IS PROGRAMMING?

A program is a list of discrete numbered lines or Command Strings that, taken together in sequence, provide the information needed to get a machine to perform your predetermined sequence of instructions. These instructions can, in the case of a Programmable Motion Control, cause the motor to move at certain speeds and for given distances, plus read various inputs or set outputs to accomplish different machine-related tasks.

MEMORY TYPES AND USAGE

A program is stored in Memory. There are two kinds of memory. RAM (Random Access Memory) is called "Volatile Memory" because when power is removed from the Control, the information stored in that memory is lost. The Control, for example, stores manual commands (H Codes) in RAM.

The second kind of memory is "Nonvolatile Memory", such as BBRAM (Battery Backed RAM). The electrical codes stored in this type of memory are retained when external power is removed.

WHAT'S IN A PROGRAM?

A program for your Control will have several elements for each line or command string. These elements are: Line Numbers, G Codes, X (or Y) Fields for move distance and other values, and an F Field for Feedrate (motor speed) or other data.

A program in your Control can have up to 400 lines. Each line may contain up to 30 characters, including letters, numbers, marks and spaces, comprising one command string. You may store several programs as well, as long as the total number of lines of all the programs you have stored does not exceed 400. These different programs may be accessed by "pointing" to the line number on which they start This will be described in full detail later. In addition to program lines, the controller needs and stores (separate from the commands) a series of set-up parameters called L Codes.

Commands and Parameters

The types of commands and/or parameters your Control can accept are pre-set. Parameters are set in your Control with "L Codes," which are covered in more detail in the Software Reference Section of this Manual. As an example of how a parameter works, let's consider L12, which sets the motor's starting or "base" speed in pulses/sec. When this code is set at, say 450 pulses/sec, the motor's starting speed will be at this 450 pulses/sec rate every time a move is commanded by the program.

1. A "parameter" is not a part of your program. Rather, it is a set-up command that is entered prior to programming the Control.

2. A "command" in your program specifies the action you wish your motor to accomplish (speed, motion direction, delay time, etc).

   Commands (G Codes) are performed via numbered lines in your program. The program is a sequence of commands that control the motor and motion-related events you want to happen in a particular period of time.

A line number (N[nnn]):

G codes, X/Y codes and F codes are assigned line numbers. Parameters (L Codes) and Serial Commands (H Codes) do not have line numbers.

Line numbers must be entered by you. They are not generated by the Control. They take the form "N001" (an "N" followed by a number occupying up to three places - you may enter N1 for example, and when the program is listed the Control will display N001). Your Control can store up to 400 lines of program.

X and Y Fields

An X or Y Field is an area designated for specific information. The information in the field is a "Value" for a specific action.

"X/Y Fields" have three different purposes:

1. X/Y Fields are used alone or with G Codes specifying motion commands, to determine direction "+" (CW)/"-" (CCW), and number of motor pulses desired for the move.

2. X/Y Fields can also be used to control program operation by directing the flow of command. With the G11 command, for example, the X field specifies the line number of a subroutine for the program to go to. (An "F Field" value specifies how many times to repeat the subroutine.) When used with G20, the X field specifies the input condition for a conditional branch for the program to go to. (An F Field specifies the branch line number.) Other G Codes are used in similar fashion for program control.

3. X/Y Fields are used to input values for variables such as delay times in a program. This use of the X/Y Field is called for by the G04 command. Many other commands obtain their data from the X/Y field.

F Fields

There are two uses for F Fields.

1. F Fields are used in conjunction with X/Y Fields and G Codes to specify the "Feedrate" of these motions.

2. F Fields used in conjunction with commands such as G11 and G20 tell the program how to use the information in the subroutines and branches.

Examples of a program line:

Listed below is a typical program line with its line number, G Codes, and respective X and F Fields and Values. Here we can see how a Command String, or program line is structured.

N020 G91 X -2500 F2400

This line is interpreted as follows:

N020	-	Program line number 20
G91	-	Sets Incremental Mode
X-2500	-	The motor moves 2500 pulses in the "-" direction
F2400	-	Feedrate (speed) is 2400 pulses/sec.

Another example: N030 G11 X50 F2

N030	-	Program line number 30
G11	-	Calls subroutine
X50	-	Location of the subroutine (at line number 50)
F2	-	Specifies number of subroutine repetitions after the initial execution of the line

The subroutine will be repeated three times: once at the initial call, plus two more repetitions called for by the value 2 in the F field.

THE CODES: OVERVIEW

A complete listing of the codes together with their meanings and uses is provided in the Software Reference Section of this Manual. It will be helpful to refer to that section as needed while you are reading this section.

To further aid you in programming your Control, Superior Electric has prepared a Motion Control Program Development Software Package: MS-2000. This software simplifies the task of programming the Motion Control using a PC.

L CODES

L Codes are system parameters. Parameters are instructions that are active throughout all the operations of the Control program.

L Codes are not considered part of program, but are used as reference throughout the entire program operation.

L Codes are usually entered first, before you begin entering the first line of your program. They can be entered in any order. Also, they can be changed at any time, even during or after programming the Control.

This section describes the general types of L Codes used to set up your Control. Each L Code is explained in full detail in the Software Reference Section of this Manual.

Kinds of L Codes

There are four kinds of L Codes: Communication Parameters, Motion Parameters, Input/Output Control Parameters, and Execution Parameters.

1. Communication Parameters: These parameters control the flow of information between your Motion Control and the "host" device used to program and operate it. Included in this group are alphanumeric messages and characters that the Control can send to the host, as well as program and parameter transfer control.

2. Motion Parameters: These parameters are very important to consider in preparing your program. Jog speed (pulses/sec), acceleration and deceleration (pulses/sec/sec), low speed, home speed, offset distance from home, CW and CCW travel limits and Mechanical Home directions (+ or -) are all set by means of L Codes, not by G Codes or other program (line numbered) commands. Some of these parameters have factory-set default values. The default values are listed in a later section of this Manual.

   In addition, the closed loop option is set up and controlled through the use of certain L Codes.

3. Input/Output Control Parameters: These set up the assignments for the way the various input and output terminals are used.

4. Execution Parameters: These parameters determine how your program will be executed in relation to the motions you are commanding. Delays and program line counts are set here.

Some important facts about certain L Codes.

L26: Command Acknowledge

L26 is used to augment the XON/XOFF handshaking protocol to indicate Control readiness for situations where XON/OFF are inappropriate (such as a dumb terminal). Regardless of the L26 setting (0, 1, 2 or 3), the Control will always respond with an "=" to indicate the Control is ready or a ":" to indicate the Control is busy, in response to host device sending the attention character "<" and a valid device address "nn" This address must correspond with the value set on the ID# select switches on top of the Control.

G CODES AND RELATED X/Y and F FIELDS AND VALUES

G Codes are used to control program execution and modify the way a program operates. For example, subroutine calls, go to's, looping, and program branching are all accomplished using G Codes. Other uses include programming delays, sending messages, strobing data from BCD switches, and boosting or reducing the motor current.

Some G codes use the X/Y and F fields, and others do not.

Here are examples of G Codes that use neither X/Y Field nor F fields (X/Y and F are ignored):

G30: Program End

G31: Stop Program

G32: Return from Subroutine

Other G Codes require data in the X/Y field. Some examples are:

G04: Program Dwell Time

The X/Y field MUST be programmed with the delay time: range 0-9999 milliseconds.

G22: Wait for Input

The X/Y field MUST contain the desired input state:
0 = inactive, 1 = active, 2 or 9 for a Don't Care condition.

G47: Set/Reset Output Condition

The X/Y field MUST contain the desired output state:
0 = inactive, 1 = active, 2 or 9 for a Don't Care condition.

In the above examples, F field data is optional. If you wish to change motor speed on a line where one of those G Codes is used, simply program a new F value on the same program line as the G code.

Refer to Appendix E for a complete summary of the G codes and their related fields.

H CODES

H Codes are instructions that are issued to the Control from outside either the embedded program (using G Codes) or parameters (L Codes) you have set. H Codes are used for manual control and operation, or for transmitting to a computer (via the serial communications port) information and status conditions of the Control.

IMMEDIATE CODES

These are codes issued from your computer or other programming device that immediately act on the Motion Control's actions. They are not part of the embedded program and they are not stored in any memory. They are listed and fully described in the Software Reference Section and Appendices of this Manual.

These include:

*	CLEAR (Uncontrolled stop); The indexer immediately halts all motor motion and program execution
$	FEED HOLD (Controlled stop); The indexer immediately brings motor motion to a controlled stop with the programmed deceleration.
#	CYCLE STOP (Stop Program Execution Cycle); The indexer will stop program execution after the present program line completes its task
<nn	Device Attention Character (nn is the device ID#)
^H (Control H)	Backspace and Delete
^X (Control X)	Delete Line
?	Send active device ID number

A number of the H Codes are also available as Immediate codes by preceding them with an exclamation point (!).

IMPORTANT TERMS USED IN PROGRAMMING

ASCII:
"ASCII" stands for American Standard Code for Information Interchange. It is means of standardizing what numbers represent what characters on a typewriter or computer keyboard. For example a letter "A" is assigned the ASCII code 65, while "a" is assigned the code 97. While you use letters or symbols on a keyboard or keypad to enter information, the computer only recognizes the numerical value, or ASCII code.

A complete listing of ASCII characters is provided in an Appendix of this Manual.

Binary Coded Decimal:

Binary Coded Decimal numbers, or BCD's, are another means of representing characters (in this case the decimal digits 0 through 9) in binary code (0's and 1's). For instance, the decimal number 1 is equal to the binary (and BCD) number 0001; the decimal number 5 is equal to the binary or BCD 0101; decimal 9 is binary or BCD 1001. To form numbers larger than decimal 9, the individual digits are combined as in the following examples: decimal 10 is expressed as BCD 0001 0000; decimal 15 is expressed as BCD 0001 0101; and decimal 91 is expressed as BCD 1001 0001.

BCD's may be used to input data such as move distance or feedrate to your Control with thumbwheel switch panels.

NOTE: A more complete listing of BCD values is found in the Glossary.

WRITING YOUR OWN PROGRAMS

OVERVIEW

When writing your own program, there are several things to keep in mind:

1. You are providing instructions to a machine. **Keep it simple!**
2. Write down each program line as you have entered it, in sequence for later reference.
3. Be patient. Plan on going through several iterations.
4. Thoroughly familiarize yourself with this Instruction Manual, where the information is, what it is, and how it is presented, before you start.

WRITING AND LOADING THE PROGRAM:

1. Define your objectives. Write them down:

   "Move the belt 9 inches in 3 increments of 3 inches each, within a period of 6 seconds."

2. Break the objective down into discrete moves per axis. Write out the sequence for each axis. Make a Flow chart showing the sequence of the moves.

3. Find the G Codes that correspond to each of these moves in the table in the Software Reference Section, then read the detailed descriptions that are also listed in the Manual. Also consider the use of inputs and outputs, and code the appropriate program lines to control them.

4. Decide on your parameter settings. List the L Codes that are appropriate for this application (use the Programming Worksheet found at the back of this Manual). Write them down.

5. A sample "Programming Worksheet" has also been provided to help you prepare your program. It can be found at the back of this Manual. Please use it to help organize and record your program.

   The program can be written directly into the Control via the serial port, using either a terminal or a computer running a terminal emulation program (such as Superior Electric's MS-2000 disk). The program can also be written using a text editor, then downloaded to the Control via a communications program (such as "Crosstalk" or "Procomm" or MS-2000).

WRITING PROGRAMS FOR VARIOUS APPLICATIONS

This section explains various programming features of the control, the codes needed to accomplish typical applications, and examples of actual program lines which will make the control perform the required tasks. These examples are listed in order by complexity (from least complex to most) and by frequency of use (most frequent to least frequent). Many examples are included to aid you in learning how to program the control. It is important that you also consult the Software section for complete details on the use of each control parameter, code and command.

CONTROLLING SIMPLE MOVES

There are two basic types of moves: index and jog. Index moves specify a move distance, Jog moves do not. Jog moves the motor at a continuous controlled speed.

1. Setting the Basic Parameters

There are four parameters (L Codes) that specify the motor's velocity during a move. These are Base Speed, Acceleration, High Speed (or Feedrate) and Deceleration. During a simple move, the motor goes from standstill to base speed, accelerates to high speed, runs at high speed, decelerates to base speed and then stops.

Base Speed is the speed at which the motor starts and ends the move. It is typically set to a speed between 250 and 500 full steps/sec. L12 is the Base Speed parameter.

Acceleration and Deceleration specify the rate of change of motor speed in steps/sec/sec. They can differ by up to a factor of ten, i.e., the deceleration value can be between 1/10 and 10 times the acceleration value. Vertical lifting is an application where separate acceleration and deceleration can be used to advantage. L10 is the deceleration and L11 the acceleration parameter.

2. Setting the Move Distance

The move distance can be specified as Absolute or Incremental. An Incremental move specifies the distance to move from the present position. An Absolute move is specified in relation to electrical home, or absolute zero. The move distance is specified in the program or can be read in from thumbwheel switches or as BCD data from a PLC.

Example: N003 G91 X+200 F1000

In this example, G91 sets the X axis to incremental mode and calls for a move of 200 steps in the + direction with a high speed of 1000 steps/sec.

3. Choosing the Program Execution Mode

The program can be executed one line at a time (Single-Line), a fixed number of times (Automatic) or continuously (Continuous). This is controlled using parameter L06.

Program execution can be initiated by a cycle start command or automatically upon power up. There is a hardware input and a serial command for cycle start. For automatic startup upon power on, there is a 3 sec. delay built in to allow the motor drive to become ready. Program execution starts at the line number specified by L41.

Parameter L06, a 3 digit number, sets the execution mode. The Hundreds digit, 0 or 1, enables or disables motor motion. The Tens digit, 0 or 1, disables or enables autostart. The Units digit, 1,2 or 3, selects Single-Line, Automatic or Continuous program execution.

Applicable Codes:

- L06 Execution format
- L41 Program start line number
- L47 Repeat count

Examples:

L06 003	Motor motion enabled, program executes continuously upon receiving a cycle start
L06 102 L47 2	Motor motion disabled, program executes 3 times upon receiving a cycle start
L06 011 L41 22	Motor motion enabled, line 22 is executed upon power up.

USING THE BCD SWITCHES AND DATA SCALING FEATURE

The SS2000I Control has the capability of reading external data such as move distance, velocity and program line number from external BCD switches. This feature is quite useful for applications where the required data may vary and thus cannot be programmed with a fixed value. To take advantage of this ability the Control can be programmed using any of the G codes that instruct the it to read the data from the external BCD switches. When the Control executes one of these G codes the data that is present on the BCD switch is used as the data for that particular operation.

For example, under ideal conditions the Control would move a slide table a distance of 1 inch, however due to variances in the day to day process this distance must be adjusted to compensate for this variance. Rather than constantly reprogramming the Control with the adjusted distance value, an external BCD switch can be used to indicate the correct distance to move. The BCD value can then be adjusted as often as needed with out reprogramming the Control. Also, to facilitate the use of the external BCD data, a scale factor can be applied to compute the correct number steps to move a given distance.

In this example the slide table has a 10 pitch lead screw, meaning that 10 motor revolutions will cause 1 inch of table travel. Assuming there are 200 pulses per motor revolution, a total of 2000 pulses would be required to move the table 1 inch, with each pulse moving the table 0.0005 inches (1/2000). The requirement is to move in increments of 0.001 inches. This would require the operator to enter twice the desired value in order to obtain the proper move distance. By programming the scale factor with the correct value a direct 1 to 1 relationship can be obtained allowing the operator to enter the exact value to be moved. In this case a scale factor of 2 would be used achieve the desired move distance.

Applicable Codes:

L20	Power up configuration
L91	External BCD move distance scale factor
G52	Select BCD switch bank
G36	Read external move data and scale the value
G37	Read external line number
G38	Read external feedrate

Example:

N001 G36 F001000	Instructs the Control to read the BCD data, scale the data and move that distance at a velocity of 1000 pulses/sec
N001 G38	Instructs the Control to read the BCD data, and use that value as the new velocity

OPERATION FROM A PLC

The single point I/O facilitates connection to a PLC. The PLC can transfer data to the control using BCD format. The data is transferred 2 BCD digits at a time and the data length is 8 digits. The data range is 0 to +/- 79,999,999.

1. I/O setup

The transfer of data from the PLC requires 9 inputs and 4 outputs. Inputs 1-9 and Outputs 5-8 are configured using L58 and L59 respectively.

Applicable Codes:
L58 Input configuration
L59 Output configuration

Example: L58 "+0000BCCCCCCCC"
L59 "BBBB0000"

Inputs 1-8	configured as PLC Data ("C")
Input 9	configured as PLC Data Ready ("B")
Outputs 5-8	configured as PLC Strobes ("B")

The interface to the PLC uses handshaking. Refer to the detailed description of L58 for details. With L58 and L59 configured as above, the "PLC Mode" is enabled.

2. Data source selection

There are three sources of external data, PLC, BCD switch 1 and BCD switch 2. Code G52 selects the source of external data. Once selected, a given source will be used, to read all external data. When configured for PLC mode, the system defaults to select PLC data upon power up.

Applicable Codes: G52 Data source select

Examples: G52 X0 selects PLC data
G52 X1 selects BCD switch 1
G52 X2 selects BCD switch 2

3. Reading External Data

Applicable Codes: G10 Index from run
G29 Program L code
G36 Read move distance
G37 Read line number and branch
G38 Read feedrate

CHANGING SPEED DURING A MOVE

Changing speed during a move can be accomplished by 2 different methods. The first method involves issuing a speed control command via the serial port while the second method involves speed control commands that are part of the user's program.

Speed changes via the serial port requires the user to generate one of several different speed control commands to cause the Control to change the speed of the motor. These serial commands can be issued any time the motor is moving. All serial commands that control speed begin with an exclamation mark "!" and end with a carriage return and or linefeed. Speed control via the serial port is generally an synchronous approach to speed control since the timing of issuing the speed control commands can not always be repeated. For more precise control of speed changes, the programmed method of speed control should be used.

Speed changes under program control can be programmed to occur based on a variety of conditions. The most straightforward approach is to simply program the speed that you desire for each index. For example, "N001 X5000 F1000" would produce an indexed move of 5000 pulses at 1000 pulses per second. Speed changes while under program control may occur based on position, time or inputs.

Applicable Codes: G60 Enable continuous line execution mode
G22 Wait for input condition
G04 Delay
!H04 Set high speed mode
!H05 Set low speed mode
!H31 Increase speed
!H32 Decrease speed
!F Feedrate override

Speed change based on position

In this example the motor moves a total of 10,000 pulses. First it moves 2500 pulses at 1000 pulses/sec, then accelerates to 2000 pulses/sec. After 5000 pulses the speed changes to 750 pulses/sec which is held for the remainder of the move.

N001 G60 X+00010000 F001000
enable continuous execution mode, then move clockwise 10000 pulses at 1000 pulses/sec

N002 G62 X+00002500 F002000
wait for 2500 pulses to be moved then change speed to 2000 pulses/sec

N003 G62 X+00005000 F000750
wait for 5000 pulses to be moved then change speed to 750 pulses/sec

N004 G30
end program - program execution will wait here until the move is completed.

Speed change based on time

In this example the motor moves a total of 10,000 pulses. The motor first moves at 1000 pulses/sec for 2.5 seconds, then accelerates to 2000 pulses/sec. After an additional 3 seconds the motor then changes speed to 750 pulses/sec which is held for the remainder of the move.

N001 G60 X+00010000 F001000
enable continuous execution mode, then move clockwise 10000 pulses at 1000 pulses/sec

N002 G04 X+00002500 F002000
wait for 2.5 seconds to elapse, then change speed to 2000 pulses/sec

N003 G04 X+00003000 F000750
wait for 3 seconds to elapse, then change speed to 750 pulses/sec

N004 G30
end program - program execution will wait here until the move is completed.

Speed change based on Inputs

In this example, the motor moves a total of 10,000 pulses. The motor will run at 1000 pulses/sec until input 1 is activated. After activating input 1 the motor will then accelerate to 2000 pulses/sec and will run at this speed until input 2 is activated. After activating input 2 the speed changes to 750 pulses/sec which is held for the remainder of the move.

N001 G60 X+00010000 F001000
enable continuous execution mode, then move clockwise 10000 pulses at 1000 pulses/sec

N002 G22 X+22222221 F002000
wait for input 1 to become active, then change speed to 2000 pulses/sec

N003 G22 X+22222212 F000750
wait for input 2 to become active, then change speed to 750 pulses/sec

N004 G30
end program - program execution will wait here until the move is complete

USING BREAKPOINT PROGRAMMING

Breakpoint programming is used in the continuous line execution mode. This allows the Control to start a move on one line then continue to execute the next line. The lines that are executed while the motor is moving can be used to set up breakpoints. These breakpoints can be based on distance, time or velocity. For example, let's assume that the Control is in the continuous line execution mode and that a move of 25,000 pulses has been started at a velocity of 1000 pulses/sec. The application requires that once 3000 pulses have been moved the velocity needs to be changed to 5000 pulses/sec and when 16000 pulses have been moved the velocity needs to be changed to 1000 pulses/sec. This application would use position breakpoints to tailor the move profile.

Applicable Codes:
- G60 Enable continuous line execution mode
- G61 Disable continuous line execution mode
- G62 Wait for distance to be achieved
- G63 Wait for velocity to be achieved

Example:

N001 G60 X+0025000 F001000
continuous line execution mode: move 25000 pulses clockwise at a velocity of 1000 pulses/sec

N002 G62 X+0003000 F005000
wait for 3000 pulses to be moved, then change velocity to 5000 pulses/sec

N003 G62 X+0016000 F001000
wait for 16000 pulses to be moved, then change velocity to 1000 pulses/sec

N004 G30 end of program

PERFORMING I/O OPERATIONS DURING INDEXED MOTION

To perform input and output operations during indexed motion the continuous mode of execution must first be enabled. With this mode enabled, a line calling for indexed motion is executed, motion will start and then program execution will continue with the next programmed line. This line may call for a time delay or for waiting until a position or velocity is reached or an input is activated before continuing with program execution. By using these commands to control program execution, external devices may be signalled at the appropriate time by the setting of outputs.

Applicable Codes:
- L58 Input configuration
- L59 Output configuration
- G04 Delay
- G22 Wait for input condition
- G47 Set outputs
- G60 Enable continuous execution mode
- G61 Disable continuous execution mode
- G62 Wait for move distance
- G63 Wait for velocity condition

Example 1. In this example continuous execution is enabled. The motor then moves a total of 10,000 pulses at a velocity of 1,000 pulses/second. Output 1 is activated when the motor reaches a velocity of 1000 pulses/sec. The output remains active for 1 second and is then turned off.

N001 G60 X+00010000 F001000
enable continuous mode, then move 10,000 pulses at 1000 pulses/sec

N002 G63 F001000
wait for a velocity of 1000 pulses/sec to be reached

N003 G47 X+00000001
turn on output 1, outputs 2 thru 8 are turned off

N004 G04 X+00001000
delay for 1 second (1000 milliseconds)

N005 G47 X+00000000
turn off all outputs

N006 G30
wait for move to finish then end program

Example 2. In this example continuous execution is enabled. The motor then moves a total of 10,000 pulses at a velocity of 1,000 pulses/second. Output 1 is activated when the motor reaches a position of 2000 pulses. The output remains active until the motor reaches a position of 9000 pulses and is then turned off.

N001 G60 X+00010000 F001000
enable continuous mode, then move 10,000 pulses at 1000 pulses/sec

N002 G62 X+00001000
wait for a position of 1000 pulses to be reached

N003 G47 X+00000001
turn on output 1, outputs 2 thru 8 are turned off

N002 G62 X+00009000
wait for a position of 9000 pulses to be reached

N005 G47 X+00000000
turn off all outputs

N006 G30
wait for move to finish, then end program

USING ADDITIONAL INPUTS AND OUTPUTS

The Control has the capability to provide 8 program testable inputs and 8 program controlled outputs. Depending on the number of inputs and outputs assigned to functions other than program I/O, the expansion port which is normally used for accessing BCD data may instead be used for expanded program controlled I/O. An additional 8 Inputs and 8 outputs are available on this port. This allows for a total of 16 inputs and 16 outputs if the application requires it.

Expansion I/O is optional. The user may purchase industry-standard I/O boards such as "OPTO-22" type PB16A or "Potter & Brumfield" type 2IO-16. The expansion card is connected to the Control's top-mounted BCD / Expansion I/O connector via a 25-conductor cable (male 25-pin subminiature "D" connector at each end), by means of the accessory adapter board, Superior Electric part 221158-001 which attaches to the edge connector on the I/O card. On the I/O card, slots 0 to 7 must be inputs, and slots 8 to 15 must be outputs.

To access the additional I/O the L20 parameter would be modified to L20 nn1nn. This tells the Control that the expansion port is to be configured for additional I/O. The same G codes G20, G22 and G47 that access the standard I/O will now be able to access the expanded I/O. To have the Control use the expanded I/O simply use Y instead of X when programming the Control. For example "G20 Xnnnnnnnn" would instruct the Control to base a branch decision the state of the standard I/O while "G20 Ynnnnnnnn" would instruct the Control to base a branch decision on the state of the expanded I/O. Likewise "G47 Xnnnnnnnn" would instruct the Control to set or reset outputs using the standard I/O while G47 "Ynnnnnnnn" would instruct the Control to set or reset outputs using the expanded I/O.

Applicable Codes:

L20	Power up configuration
L59	Output configuration
G20	Branch on input condition
G22	Wait for input condition
G47	Set/reset outputs

Example:

N001 G22 X+00000111 Instructs the Control to wait for standard inputs 1, 2 and 3 to be active and standard inputs 4 through 8 to be inactive before continuing with program execution

N002 G22 Y+00000111 Instructs the Control to wait for expansion inputs 1, 2, and 3 to be active and expansion inputs 4 through 8 to be inactive before continuing with program execution

N003 G20 X+00000111 F000050
Instructs the Control to go to line 50 if standard inputs 1, 2 and 3 are active and standard inputs 4 through 8 are inactive.

N004 G20 Y+00000111 F000050
Instructs the Control to go to line 50 if expansion inputs 1, 2 and 3 are active and expansion inputs 4 through 8 are inactive.

N005 G47 X+00000001 Instructs the Control to turn on standard output 1 and turn off standard outputs 2 through 8.

N006 G47 Y+00000001 Instructs the Control to turn on expansion output 1 and turn off expansion outputs 2 through 8.

SENDING ASCII MESSAGES

In some applications it may be required to prompt an operator to select or start a specific process. In some cases this prompt may be something as simple as turning on a light and waiting for a button to be pressed before continuing with the program. In other cases more information must be presented to the operator before a decision can be made. If the application requires the ability to prompt the operator with detailed messages, the SLO-SYN 2000 Control's ability to send messages to a display may be used. The Control provides the user with the capability of programming 5 different messages with each message being 20 characters long. The messages may be transmitted one at a time or several messages may transmitted in sequence to be combined into longer messages. The messages are transmitted via the serial port, so the device connected to the serial port must be capable of receiving and displaying standard ASCII characters. After each message is transmitted, a carriage return and line feed will automatically be sent.

Applicable Codes: 1

L01	Message 1
L02	Message 2
L03	Message 3
L04	Message 4
L05	Message 5
G05	Transmit message 1 - 5

Example:

L01 "Load a part now"

N001 G05 X+00000001 send message #1

Example:

L01 "This is an example"
L02 "of combining text"
L03 "to send messages"
L04 "longer than 20"
L05 "characters."

N001 G05 X+00000001 send message #1
N002 G05 X+00000002 send message #2
N003 G05 X+00000003 send message #3
N004 G05 X+00000004 send message #4
N005 G05 X+00000005 send message #5
N006 G30 end program

USE OF THE SECOND AXIS

The SLO-SYN 2000 Control has the capability of driving a second axis of motion. The second axis has all the features of the primary axis except closed loop operation. The second axis is independent of the primary axis. A separate position counter and status register is maintained; also various L codes support two fields which allow for independent parameters. The only operation that is not supported, is for both the primary and secondary axes to move at the same time. Assuming that the default parameters are still loaded and that two drives are properly connected to the Control it is simple matter to program both axes for motion.

Example:

N001 G91 X+00001000 F001000
enable incremental positioning mode for the X axis, move the X axis clockwise 1000 pulses at 1000 pulses/sec

N002 G90 Y-00002300 F000750
enable the absolute positioning mode for the Y axis, move the Y axis to an absolute position of 2300 pulses at 750 pulses/sec

N003 X+00001000
move the X axis clockwise an additional 1000 pulses at 1000 pulses/sec

N004 Y+00002000
move the Y axis to an absolute position of 2000 pulses at 750 pulses/sec

N005 G30 end of program

MARK REGISTRATION (INDEX FROM RUN)

Mark registration (sometimes called Index From Run) is a means by which an indexed move can be performed when the total move distance is unknown. This incredible ability is actually quite simple, by connecting an external sensor to the Control's trigger input the Control can be made to run until this sensor detects a registration mark. At this point the Control will switch to an Index mode and will then move a preset distance and stop. This feature is useful when the distance from the registration mark is known but the distance between registration marks is unknown. The accuracy for this type of motion is typically +/- 1 pulse.

Applicable Codes: L16 Mark registration travel limit
L20 Power up configuration
G10 Mark registration mode

Example:

N001 G10 X+00001000 F001000
enable mark registration mode then start running clockwise at 1000 pulses/sec, wait for trigger to occur, and when trigger occurs move 1000 pulses at 1000 pulses/sec.

N002 G47 X+00000001
turn on output 1, all other outputs off

N003 G30 end of program

SOFTWARE REFERENCE

This page intentionally left blank

L CODES

CONTROL PARAMETER (L CODE) FUNCTIONAL LISTING

System Parameters

- L20 System Configuration
- L58 Input Configuration
- L59 Output Configuration

Travel Limit Parameters

- L16 Mark Registration Travel Limit
- L18 Clockwise Software Travel Limit
- L19 Counterclockwise Software Travel Limit

Serial Port Parameters

- L26 Transmission Protocol
- L48 Program Line Count List/Clear
- L49 First Parameter To List
- L50 Parameter List Count
- L98 Delay Between Automatic Transmissions

Serial Port Messages

- L01 Message #1
- L02 Message #2
- L03 Message #3
- L04 Message #4
- L05 Message #5
- L52 Buffer Full Warning Message
- L53 Following Error Warning Message
- L54 Unable To Reach Position Warning Message
- L55 Line Done Message
- L56 Program Done Message

Execution Parameters

- L06 Program Execution Format
- L41 Program Start Line Number
- L44 Delay After Motion
- L47 Program Execution Repeat Count
- L91 External Move Distance Scale Factor

Manual Motion Parameters

- L09 Jog Speed
- L13 Step Distance
- L73 Speed Increment

Motion Parameters

- L10 Deceleration
- L11 Acceleration
- L12 Low Speed
- L71 Maximum Speed
- L72 Velocity Profile

Home Parameters

- L08 Mechanical Home Direction
- L14 Home Speed
- L17 Offset From Home Switch

Backlash Parameters

- L43 Backlash Compensation Delay
- L66 Backlash Compensation

Closed Loop Parameters

- L87 Following Error
- L90 Closed Loop Configuration
- L93 Position Deadband
- L94 Encoder Counter Direction
- L95 Encoder Scale Factor
- L96 Position Correction Attempts
- L97 Delay Between Correction Attempts

```
L01  Message #1
L02  Message #2
L03  Message #3
L04  Message #4
L05  Message #5
```

Modified per Rev. B

Function

Defines the messages that are transmitted when a Transmit Message (G05) command is executed.

Command Format

L01 "aaaaaaaaaaaaaaaaaaaa"
L02 "aaaaaaaaaaaaaaaaaaaa"
L03 "aaaaaaaaaaaaaaaaaaaa"
L04 "aaaaaaaaaaaaaaaaaaaa"
L05 "aaaaaaaaaaaaaaaaaaaa"

Range

ASCII characters from 32 to 126 decimal with the following exceptions:

" decimal 34
$ decimal 36
* decimal 42
< decimal 60

Maximum message length is 20 characters.

Defaults

L01 = "Message #1 goes here"
L02 = "Message #2 goes here"
L03 = "Message #3 goes here"
L04 = "Message #4 goes here"
L05 = "Message #5 goes here"

Notes

Quotes must be used when programming L01 - L05. The designated message will be transmitted via the serial port when a G05 command is executed.

The message is sent only if the unit's ID is active.

An "*" is substituted for a "~" when the message is sent and "CRLF" is inhibited.

An "LF" is sustituted for a "^" when the message is sent and "CRLF" is inhibited

See Appendix H for further details.

L06 Program Execution Format

Function

Defines the way in which a program is executed.

Command Format

L06 nnn

n--	0 =	Program Debug Mode off, program executes as specified and motor motion will occur.
	1 =	Program Debug Mode on, program executes as specified but motor motion is not allowed.
-n-	0 =	AutoStart disabled, execution begins with a cycle start command.
	1 =	AutoStart enabled, execution begins upon power up.
--n	1 =	Single-Line Execution
	2 =	Complete Program Execution
	3 =	Continuous Program Execution

Default

L06 002 Program Debug Mode off
Autostart disabled
Complete Program Execution

Notes

In the Program Debug Mode (L06 1nn), the pulse output is disabled, thereby inhibiting all motion. This mode may be useful when debugging a program.

If AutoStart is enabled (L06 n1n), program execution commences upon power up, otherwise a cycle start command is required.

In the Single-Line Execution Mode (L06 nn1), each program line requires a Cycle Start for execution.

In the Complete Execution Mode (L06 nn2), the entire program is executed once plus the number of times indicated by the Program Execution Repeat Count parameter (L47).

In the Continuous Execution Mode (L06 nn3), the entire program is executed continuously.

Related parameters

L47 (Program Execution Repeat Count)
L41 (Program Start Line Number)

L08 Mechanical Home Direction

Function Sets motor rotation direction for a Mechanical Home Cycle.

Command Format
L08 s,s Set X and Y Home Directions.
L08 s Set X axis Home Direction.
L08 ,s Set Y axis Home Direction.

Range s + or - (CW or CCW rotation)

Default +,+ (CW rotation for X and Y)

Related parameters
L14 (Home Speed)
L17 (Offset From Home Switch)

Examples
L08 -,+ X Home Direction is - (CCW), Y is + (CW)

L08 ,- Y Home Direction is - (CCW)

L09 Jog Speed

Function Feedrate used to Jog or Step motor.

Command Format

L09 nnnnnn,nnnnnn	Set X and Y jog speeds.
L09 nnnnnn	Set X axis jog speed.
L09 ,nnnnnn	Set Y axis jog speed.

Range nnnnnn 1 - 999,999 pulses/sec

Default Value equivalent to 1000 Full Steps/sec, exact number depends on selected microstepping resolution. See Default table in Appendix A.

Examples

L09 2000,2000	Sets X and Y axis Jog Speed to 2000 pulses/sec
L09 2000	Sets X axis Jog Speed to 2000 pulses/sec

L10 Deceleration
L11 Acceleration

Function

L10 defines the rate at which the motor speed is decreased and L11 the rate at which it is increased.

Command Format

L10 nnnnnnnn,nnnnnnnn	Set Deceleration for both axes.
L10 nnnnnnnn	Set X axis Deceleration.
L10 ,nnnnnnnn	Set Y axis Deceleration.
L11 nnnnnnnn,nnnnnnnn	Set Acceleration for both axes.
L11 nnnnnnnn	Set X axis Acceleration.
L11 nnnnnnnn	Set Y axis Acceleration.

Range

nnnnnnnn 1 - 99,999,999 pulses/sec/sec
Deceleration (L10) 0.1 to 10 times Acceleration (L11)

Default

Value equivalent to 1000 Full Steps/sec/sec, exact number depends on selected microstepping resolution. See Default table in Appendix A.

Notes

The Deceleration value can be from 0.1 to 10 times the Acceleration value (L11).

If a Deceleration value is entered that is outside this range it is discarded, the old value retained and the Illegal L code program error status bit set.

If the Acceleration (L11) value is changed, the Deceleration value is checked against the range based on the new Acceleration. If the Deceleration value is outside this new range, it is set equal to the Acceleration value.

Related parameters

L72 (Velocity Profile)

Examples

L10 2000,2000	Set the X and Y Deceleration to 2000 pulses/sec/sec.
L11 ,2000	Set the Y axis Acceleration to 2000 pulses/sec/sec.

L12 Low Speed

Function

The starting/stopping speed in High Speed mode and the feedrate for all moves in Low Speed mode.

Command Format

L12 nnnnnn,nnnnnn	Sets X and Y axes Low Speed.
L12 nnnnnn	Sets X axis Low Speed.
L12 ,nnnnnn	Sets Y axis Low Speed.

Range

nnnnnn 0 - 999,999 pulses/sec

Default

300 Full Step/sec equivalent, exact number depends on selected microstepping resolution. See Default table in Appendix A.

Notes

During a typical move, the motor starts at Low Speed, accelerates to the target speed then decelerates to Low Speed and stops. When the target speed is less than the low speed, the motor will start and stop at the target speed.

When Low Speed mode is selected, all programmed feed rates are ignored and the Low Speed value is used for all motion.

Example

L12 320,320	Sets X and Y Low Speed to 320 pulses/sec.
L12 320	Sets X axis Low Speed to 320 pulses/sec.

L13 Step Distance

Function Distance moved in response to a step command.

Command Format

L13 nnnnnnnn,nnnnnnnn	Sets X and Y axes step distance.
L13 nnnnnnnn	Sets X axis step distance.
L13 ,nnnnnnnn	Sets Y axis step distance.

Range nnnnnnnn 1 - 99,999,999 pulses

Default 1 Full step equivalent, exact number depends on selected microstepping resolution. See Default table in Appendix A.

Notes A step command can be initiated by hardware inputs or by commands (H Codes) via the serial port.

Example

L13 200,200	Set X and Y step distance to 200 pulses
L13 200	Set X axis step distance to 200 pulses

L14 Home Speed

Function Speed used for Electrical (G76) or Mechanical Home (G78) cycles.

Command Format

L14 nnnnnn,nnnnnn	Set X and Y axis home speed.
L14 nnnnnn	Set X axis home speed.
L14 ,nnnnnn	Set Y axis home speed.

Default 1000 Full Steps/sec equivalent, exact number depends on selected microstepping resolution. See Default table in Appendix A.

Notes The feedrate during an Electrical or Mechanical Home cycle is either the Home Speed (L14) or the Low Speed (L12). The Low Speed is used when Low Speed Mode is active.

Related parameters
L08 (Mechanical Home Direction)
L17 (Offset From Home Switch)

Example:

L14 3000,3000	sets the X and Y home speed to 3000 pulses/sec
L14 ,3000	sets the Y axis home speed to 3000 pulses/sec

L16 Mark Registration Travel Limit

Function Defines the maximum move distance allowed in a Mark Registration Cycle.

Command Format L16 nnnnnnnn Set mark registration travel limit

Range nnnnnnnn 0 - 99,999,999 pulses

Default 0 (limit disabled)

Notes There is only one mark registration travel limit, it is used with both X and Y. The travel limit is an incremental distance, relative to the start of the move. A zero value denotes no limit (disabled).

The travel limit will terminate a Mark Registration cycle, should the registration input fail to occur. When the L16 distance is exceeded, the motion is decelerated to a controlled stop and bit 8 of System Error Status set.

Example L16 2000 Sets the Mark Registration Travel Limit to 2000 pulses.

L17 Offset From Home Switch

Function Defines the final position of a mechanical home cycle, as an offset (distance and direction) from the position, where the home input is activated.

Command Format

L17 snnnnnnnn,snnnnnnnn	Sets Offset From Home Switch for both X and Y.
L17 snnnnnnnn	Sets Offset From Home Switch for the X axis.
L17 ,snnnnnnnn	Sets Offset From Home Switch for the Y axis.

Range snnnnnnnn -99,999,999 to +99,999,999 pulses

Default +0,+0 (no offset)

Notes When the final position is achieved during a mechanical home cycle, the absolute position is set to 0.

The Offset From Home Switch parameter is not used during an Electrical Home cycle.

Related parameters L08 (Mechanical Home Direction)
L14 (Home Speed)

Example L17 +1000,+1000 Sets X and Y Offset From Home +1000 pulses for Switch values.

L17 +1000 Sets the X axis Offset From Home +1000 pulses for Switch value.

L18 Clockwise Software Travel Limit
L19 Counterclockwise Software Travel Limit

Function

CW motion is terminated or not allowed when absolute position is more positive or equal to L18. CCW motion is terminated or not allowed when the absolute position is more negative than L19.

Command Format

L18 snnnnnnnn,snnnnnnnn	Sets X and Y CW software travel limit.
L18 snnnnnnnn	Sets X axis CW software travel limit.
L18 ,snnnnnnnn	Sets Y axis CW software travel limit.
L19 snnnnnnnn,snnnnnnnn	Sets X and Y CCW software travel limit.
L19 snnnnnnnn	Sets X axis CCW software travel limit.
L19 ,snnnnnnnn	Sets Y axis CCW software travel limit.

Range

s	+ or -	
nnnnnnnn	0 - 99,999,999	pulses

Default

L18: -0, -0 CW (limit disabled)
L19: +0, +0 (CCW limit disabled)

Notes

A minus L18 value disables the CW software limit.

CW motion is not allowed unless the absolute position is less positive than the L18 value. When, during CW motion, the absolute position becomes more positive or equal to L18, the motion is decelerated to a controlled stop and bit 5 of Motion Error Status is set.

A positive L19 value disables the CCW software limit.

CCW motion is not allowed unless the absolute position is less negative than the limit. When, during CCW motion, the absolute position becomes more negative than L19, the motion is decelerated to a controlled stop and bit 4 of Motion Error Status is set.

Examples

L18 +100000,-100000	X CW software limit at +100000 pulses, Y disabled.
L19 -100000,+100000	X CCW software limit at -100000 pulses, Y disabled
X absolute position =	90000, X axis CW motion is allowed.
X absolute position =	100000, X axis CW motion is not allowed.
X absolute position =	-90000, X axis CCW motion is allowed.
X absolute position =	-100000, X axis CCW motion is not allowed.

L20 System Configuration

Modified per Rev. C

Function

Configures inputs TR1, TR2 and the expansion port. Specifies the electrical position, encoder position and feedrate defaults at power up.

Command Format

L20 nnnnn

TR1	n—	0 -	X axis Registration or Home input, positive edge triggered.
		1 -	X axis Registration or Home input, negative edge triggered.
		2 -	Y axis Registration or Home input, positive edge triggered.
		3 -	Y axis Registration or Home input, negative edge triggered.
		4 -	X axis Home input, encoder index and TR1 active.
TR2	-n—	0 -	X axis Registration or Home input, positive edge triggered.
		1 -	X axis Registration or Home input, negative edge triggered.
		2 -	Y axis Registration or Home input, positive edge triggered.
		3 -	Y axis Registration or Home input, negative edge triggered.
		4 -	X axis Home input, encoder index and TR2 active.
	—n—	0 -	Expansion Port interfaced to BCD switches.
		1 -	Expansion Port interfaced to expansion I/O.
		2 -	Expansion Port interfaced to SS2000 Expansion I/O board
	—n-	0 -	X and Y absolute position = 0. General purpose registers are set to zero on power up.
		1 -	The X and Y absolute positions saved at power down are restored. General purpose registers are set to zero on power up.
		2 -	(used with closed loop option) The encoder position saved at power down is restored. X absolute position = encoder position. Y absolute position = 0. General purpose registers are set to zero on power up.
		3 -	X and Y absolute position = 0. General purpose registers saved at power down are restored.
		4 -	The X and Y absolute positions saved at power down are restored. General purpose registers saved at power down are restored.
		5 -	(used with closed loop option) The encoder position saved at power down is restored. X absolute position = encoder position. Y absolute position = 0. General purpose registers saved at power down are restored.
	---n	0 -	Power up feedrate = L09 value.
		1 -	Power up feedrate = last index feedrate

Default

L20 02000

TR1 X axis positive edge
TR2 Y axis positive edge
Expansion port - BCD switches
X and Y abs. position = 0
Power up feedrate = L09
General purpose registers are set to zero on power up.

Notes

When a mechanical home or registration cycle is executed, the input signal source is determined by checking the assignment for TR1 first, then TR2. For example if TR1 is X positive and TR2 is X negative, TR1 X positive will be used.

Positive Edge Trigger definition:
Input transitions from inactive to active.
Negative Edge Trigger definition:
Input transitions from active to inactive.

The expansion port can be interfaced to BCD switches or expansion I/O, not both.

Related parameters

L58 (Input Configuration)
L59 (Output Configuration)

L26 Transmission Protocol

Function Selects the serial communication protocol

Command Format L26 n

Range n 0 - 7 (see below)

Default 0 (normal transmission mode, Xon/Xoff protocol)

Notes

Xon/Xoff Protocol Enabled

- n=0 normal transmission mode (no "EOT" or "=" characters)
- n=1 "EOT" follows each complete data transmission
- n=2 "=" is transmitted when ready for more commands
- n=3 "EOT" follows each complete data transmission and "=" is transmitted when ready for more commands

Xon/Xoff Protocol Disabled

- n=4 normal transmission mode (no "EOT" or "=" characters)
- n=5 "EOT" follows each complete data transmission
- n=6 "=" is transmitted when ready for more commands
- n=7 "EOT" follows each complete data transmission and "=" is transmitted when ready for more commands

EOT: ASCII 04 decimal.

see Xon/Xoff description for details

L41 Program Start Line Number

Function Line number from which program execution starts following a Power up, a Clear command, execution of an End of Program command or the occurance of a program error condition.

Command Format L41 nnn

Range nnn 0 - 400 line number

Default line 1

Example L41 100 Program Starts at line 100.

L43 Backlash Compensation Delay

Function Delay between an index move and its backlash cycle. Also the delay between the jog and index portions of a mechanical home cycle.

Command Format L43 nnnn

Range nnnn 0 - 9,999 milliseconds

Default 50 milliseconds

Notes The delay time should be sufficient to allow the motor to settle before the backlash correction move begins.

Related parameters L66 (Backlash Compensation)

Example L43 100 Sets Backlash Compensation Delay to 100 milliseconds.

L44 Delay After Motion

Function Delay that occurs following motion.

Command Format L44 nnnn

Range nnnn 0 - 9,999 milliseconds.

Default 50 milliseconds

Notes The delay time should be sufficient to allow the motor to settle before the next move begins.

Example L44 100 Sets Delay After Motion to 100 milliseconds.

L47 Program Execution Repeat Count

Function Number of times a program will be executed in the Complete Execution Mode. (L06 nn2)

Command Format L47 nnnn

Range nnnn 0 - 9999 times

Default 0 times

Notes This parameter only applies in the Complete Execution Mode (L06 nn2). In this mode, the program is executed once plus L47 times.

Related parameters L06 (Program Execution Format)

Example L47 50 Sets the Program Execution Repeat Count to 50.

L48 Program Line Count List/Clear

Function Number of program lines to be listed or cleared in response to serial port commands H14 and H12.

Command Format L48 nnn

Range: nnn 0 - 400 lines

Default: 20 lines

Notes: Starting at the present line number, transmit via the serial port the specified number of programmed lines in response to an H14 (List) command.

Starting at the present line number, clear the specified number of consecutive program lines in response to an H12 command. (clear)

If L48 = 0 then all lines will be listed or cleared.

Related parameters: L49 (First Parameter to List)
L50 (Parameter List Count)

Examples L48 0 All programmed lines listed or cleared.

L49 First Parameter to List
L50 Parameter List Count

Function

Control the listing of parameters in response to serial port command H16 (transfer parameters).

Command Format

L49 nn
L50 nn

Range

nn 0, valid L code number
nn 0 - 99

Default

L49 = 0	List all parameters, 3 per
L50 = 0	line format.

Notes

Attempts to program L49 with a non-existent parameter number, other than 0, are ignored.

The response to a transfer parameter (H16) command is:

L49 = 0 L50 = 0	List all parameters, three per line format. (3 columns)
L49 = 0 L50 non zero	List all parameters, one per line format. (1 column)
L49 non zero L50 = 0	List single parameter specified by L49.
L49 non zero L50 non zero	List L50 parameters, starting with the parameter specified by L49.

Related parameters

L48 (Program Line Count List/Clear).

Example

L50 = 4, L49 = 9 The response to an H16 command is:

L09 nnnnnn,nnnnnn
L10 nnnnnnnn,nnnnnnnn
L11 nnnnnnnn,nnnnnnnn
L12 nnnnnn,nnnnnn

L52 Buffer Full Warning Message

Function

Defines the message that is transmitted when either buffer is within 10 characters of overflowing.

Command Format

L52 "aaa"

Range

ASCII characters from 32 to 127 decimal except for the following:
" decimal 34
$ decimal 36
* decimal 42
< decimal 60

Maximum message length is three characters

Default

L52 "" (null string, no message sent)

Notes

Quotes must be used when programming L52.

The message is transmitted only if the unit's ID is active.

The message will be transmitted via the serial port when either buffer is within 10 characters of overflowing. A "CR LF" is transmitted following the message.

If the message is the null string (""), then nothing, not even "CR LF", is transmitted.

Example

L52 "Buf" Sets "BufCRLF" as the characters to be transmitted when either buffer is near to overflowing.

L53 Following Error Warning Message

Function

Defines the message that is transmitted when the position error exceeds the following error (L87).

Command Format

L53 "aaa"

Range

ASCII characters from 32 to 127 decimal except for the following:
" decimal 34
$ decimal 36
* decimal 42
< decimal 60

Maximum message length is three characters

Default

L53 "" (null string, no message sent)

Notes

This L code is only available with the closed loop option.

Quotes must be used when programming L53.

The message is transmitted only if the unit's ID is active.

The message will be transmitted via the serial port when the position error, during motion, exceeds the following error (L87) value. A "CR LF" is transmitted following the message.

If the message is the null string (""), then nothing, not even "CR LF", is transmitted.

Example

L53 "Fol" Sets "FolCRLF" as the characters to be to be sent transmitted when a following error occurs.

L54 Unable To Reach Position Warning Message

Function

Defines the message that is transmitted when the position error is excessive and could not be corrected.

Command Format

L54 "aaa"

Range

ASCII characters from 32 to 127 decimal except for the following:
" decimal 34
$ decimal 36
* decimal 42
< decimal 60

Maximum message length is three characters

Default

L54 "" (null string, no message sent)

Notes

This L code is only available with the closed loop option.

Quotes must be used when programming L54.

The message is transmitted only if the unit's ID is active.

The message will be transmitted via the serial port when excessive position error could not be corrected. A "CR LF" is transmitted following the message.

If the message is the null string (""), then nothing, not even "CR LF", is transmitted.

Example

L54 "Utc" Sets "UtcCRLF" as the characters to be transmitted when the excessive position error can not be corrected.

L55 Line Done Message

Function Defines the message that is transmitted when a program line is completed.

Command Format L55 "aaa"

Range ASCII characters from 32 to 127 decimal except for the following:
" decimal 34
$ decimal 36
* decimal 42
< decimal 60

Maximum message length is three characters

Default L55 "" (null string, no message sent)

Notes Quotes must be used when programming L55.

The message is transmitted only if the unit's ID is active.

The message will be transmitted via the serial port upon completion of a program line (line 1 thru 400). A "CR LF" is transmitted following the message.

If the message is the null string (""), then nothing, not even "CR LF", is transmitted.

Example L55 "Lin" Sets "LinCRLF" as the characters to be to be sent transmitted when a program line is completed.

L56 Program Done Message

Function Defines the message that is transmitted when a the program is completed.

Command Format L56 "aaa"

Range ASCII characters from 32 to 127 decimal except for the following:
" decimal 34
$ decimal 36
* decimal 42
< decimal 60

Maximum message length is three characters

Default L56 "" (null string, no message sent)

Notes Quotes must be used when programming L56.

The message is transmitted only if the unit's ID is active.

The message will be transmitted via the serial port upon completion of a the program. A "CR LF" is transmitted following the message.

If the message is the null string (""), then nothing, not even "CR LF", is transmitted.

Example L56 "Prg" Sets "PrgCRLF" as the characters to be to be transmitted when the program has been executed.

L58 Input Configuration

Function

Assigns specific functions to the 13 configureable inputs and selects the logic convention for both inputs and outputs.

Command Format

L58 "shhhhhhhhhhhhh"

Range

s + or - (logic convention)
h 0 - 9, A, B, C, D, E, F (function designator)

Default

+2187600000000 positive logic
In13 CCW limit X axis
In12 CW limit X axis
In11 Step/Jog
In10 CCW direction
In9 CW direction
In1-8 Program Testable Inputs

Notes

The sign determines the logic convention for the inputs and outputs (+ = positive, - = negative). With positive logic, an active input or output is denoted by a logic 1.

With negative logic, an active input or output is denoted by a logic 0.

Each digit in the L58 field represents a specific hardware input. The value programmed for a given digit determines the function assigned to that corresponding input.

```
L58   "shhhhhhhhhhhhh"
       s------------- Logic convention + pos, - neg.
       -h------------ Function for hardware Input 13
       --h----------- Function for hardware Input 12
       ---h---------- Function for hardware Input 11
       ----h--------- Function for hardware Input 10
       -----h-------- Function for hardware Input 9
       ------h------- Function for hardware Input 8
       -------h------ Function for hardware Input 7
       --------h----- Function for hardware Input 6
       ---------h---- Function for hardware Input 5
       ----------h--- Function for hardware Input 4
       -----------h-- Function for hardware Input 3
       ------------h- Function for hardware Input 2
       -------------h Function for hardware Input 1
```

Descriptions

Function 0 — Program Testable Input
Input state is checked when G20 (branch) or G22 (wait) command is executed.

Function 1 — Clockwise Travel Limit X Axis
When this input is activated, X axis CW motion is stopped immediately, further CW motion is not allowed while this input is active.

Function 2 — Counterclockwise Travel Limit X Axis
When this input is activated, X axis CCW motion is stopped immediately, further CCW motion is not allowed while this input is active.

Function 3 — Feed Hold
When this input is active motion comes to a controlled stop, if uncompleted, the remainder of the move will be finished when a cycle start is commanded.

Function 4 — Cycle Start
When this input transitions from inactive to active program execution starts or resumes from a Feed Hold state.

Function 5 — Cycle Stop
When this input becomes active, program execution stops at the completion of the current line.

Function 6 — Clockwise Direction (manual control)
Initiates Step or Jog motion in the CW direction. The commanded motion type is determined by the Step/Jog mode status of the active axis. Motion is initiated on the inactive to active transition of this input. The input must remain active to maintain Jog motion.

This input is ignored during program execution.

Function 7 — Counterclockwise Direction (manual control)
Initiates Step or Jog motion in the CCW direction. The commanded motion type is determined by the Step/Jog mode status of the active axis. Motion is initiated on the inactive to active transition of this input. The input must remain active to maintain Jog motion.

This input is ignored during program execution.

Function 8 — Step/Jog (manual control)
Sets the Step/Jog mode of the active axis. an inactive to active input transition selects Step Mode and an active to inactive transition selects Jog Mode.

This input is ignored during program execution, while motion is occurring or when in a Feed Hold state.

Function 9 — Low/High
Sets the High/Low speed mode of the active axis. When this input transitions from inactive to active, Low Speed Mode is selected, an active to inactive transition selects High Speed Mode.

Function A — Motor Windings Off/On
Sets the Windings Off/On mode of the active axis. When this input transitions from inactive to active, Windings Off Mode is selected, an active to inactive transition selects Windings On Mode.

This input is ignored during program execution, while motion is occurring or when in a Feed Hold state.

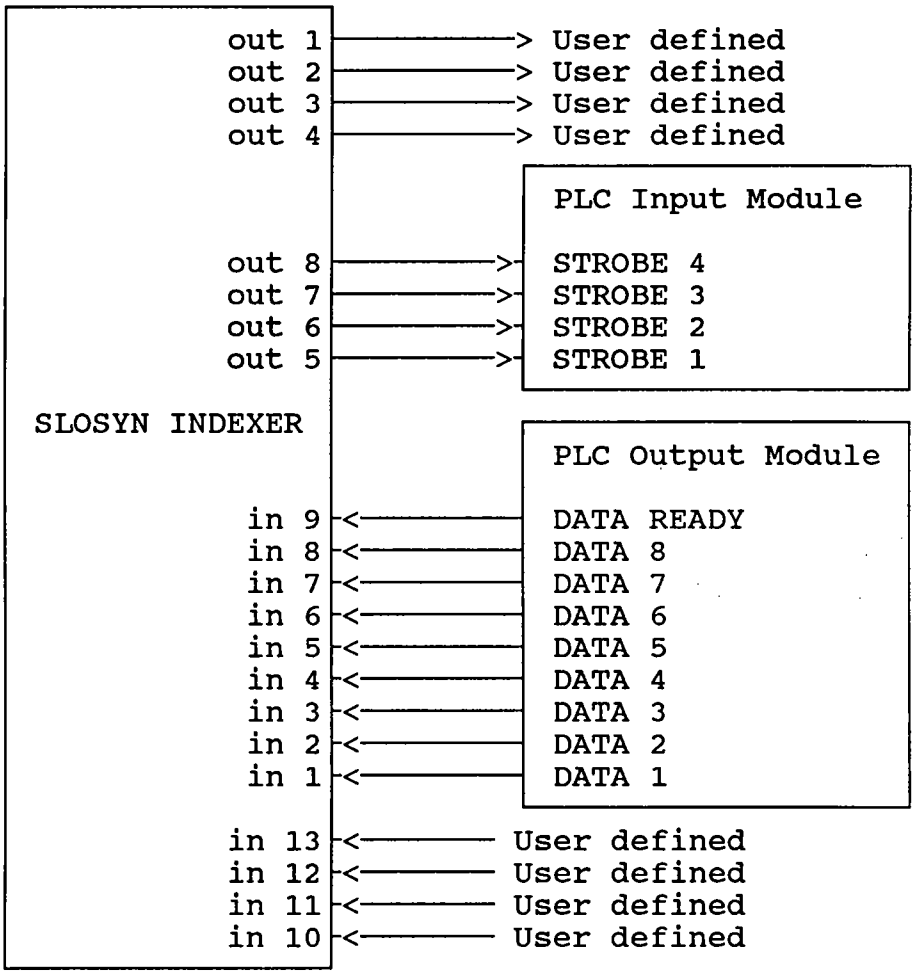
Function B — PLC DATA READY
Used for handshaking when reading data from a PLC. When active indicates that the PLC data is valid. This function can only be assigned to IN9.

Function C — PLC DATA
Reading PLC data requires 8 data inputs. The data inputs can only be assigned to IN1 - IN8. The data is coded as 2 BCD digits. The msb is IN8.

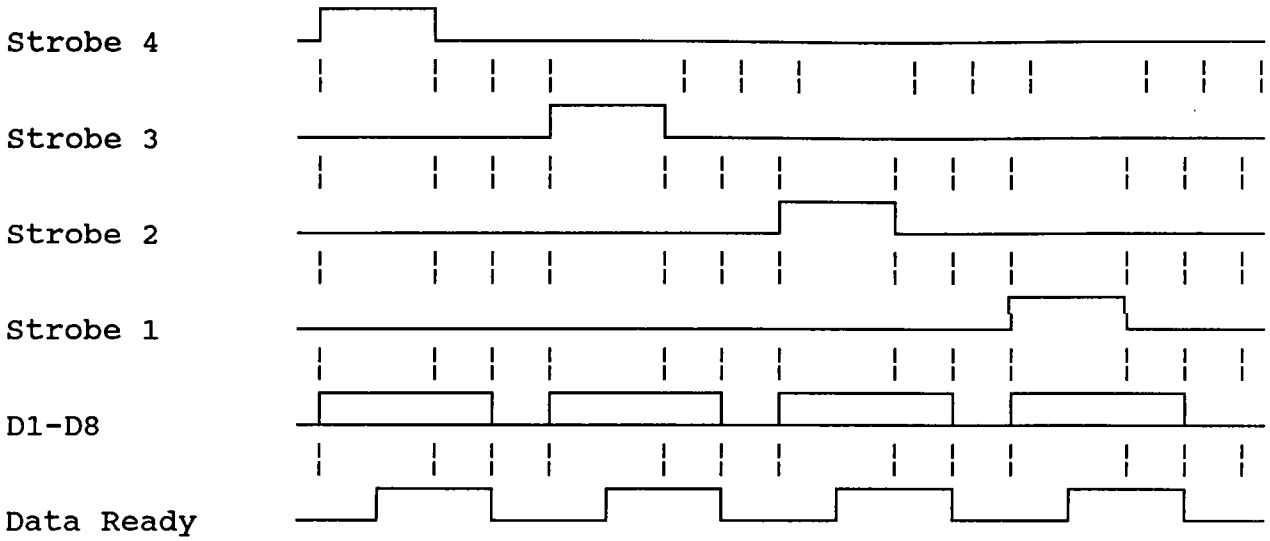
Function D — Y axis/ X axis select (manual control)
Selects the active axis. When this input transitions from inactive to active, Y becomes the active axis, an active to inactive transition selects X as the active axis. This input is ignored during program execution, while motion is occurring or when in a Feed Hold state.

Function E — Clockwise Travel Limit Y Axis
When this input is activated, Y axis CW motion is stopped immediately, further CW motion is not allowed while this input is active.

Function F — Counterclockwise Travel Limit Y Axis
When this input is activated, Y axis CCW motion is stopped immediately, further CCW motion is not allowed while this input is active.



PLC INTERFACE BLOCK DIAGRAM
FIGURE L-1



PLC TIMING DIAGRAM
FIGURE L-2

PLC Handshaking Sequence

1)	Control	Asserts Strobe 4 (OUT8)
2)	PLC	Outputs data (Sign, 10M's & 1M's) and asserts Data Ready
3)	Control	Reads data then removes Strobe 4
4)	PLC	Removes Data Ready and data
5)	Control	Asserts Strobe 3 (OUT7)
6)	PLC	Outputs data (100K's & 10K's) and asserts Data Ready
7)	Control	Reads data then removes Strobe 3
8)	PLC	Removes Data Ready and data
9)	Control	Asserts Strobe 2 (OUT6)
10)	PLC	Outputs data (1K's & 100's) and asserts Data Ready
11)	Control	Reads data then removes Strobe 2
12)	PLC	Removes Data Ready and data
13)	Control	Asserts Strobe 1 (OUT5)
14)	PLC	Outputs data (10's & 1's) and asserts Data Ready
15)	Control	Reads data then removes Strobe 1
16)	PLC	Removes Data Ready and data

PLC data Strobe Assignments

Strobe	In #8	In #7	In #6	In #5	In #4	In #3	In #2	In #1
Out 1	80	40	20	10	8	4	2	1
Out 2	8,000	4,000	2,000	1,000	800	400	200	100
Out 3	800,000	400,000	200,000	100,000	80,000	40,000	20,000	10.000
Out 4	- sign	40,000,000	20,000,000	10,000000	8,000,000	4,000,000	2,000,000	1,000,000

Related Parameters L20 (System Configuration)
L59 (Output Configuration)

Example:

L58 "-FE21BCCCCCCCC" Negative logic convention

Input #13,	Y axis CCW Travel Limit
Input #12,	Y axis CW Travel Limit
Input #11,	X axis CCW Travel Limit
Input #10,	X axis CW Travel Limit.
Input #9,	PLC Data Ready
Input #1-#8,	PLC Data

L58 "+3456789AD0000" Positive logic convention

Input #13,	Feedhold
Input #12,	Cycle Start
Input #11,	Cycle Stop
Input #10,	CW direction (manual control)
Input #9,	CCW direction (manual control)
Input #8,	Step/Jog
Input #7,	Low/High
Input #6,	Motor Windings On/Off
Input #5,	X axis/ Y axis select
Inputs #1-#4,	program testable inputs

L59 Output Configuration

Function Assigns specific functions to the 8 configurable outputs. See "Descriptions" below.

Command Format L59 "hhhhhhhh"

Range h 0 - 9, A, B, C, D, E, F (function designator)

Default 00000000 Out1 - 8 are programmable outputs

Notes Each digit in the L59 field represents a specific hardware output. The value programmed for a given digit determines the function assigned to that corresponding output.

L59 "hhhhhhhh"
h------ Function for hardware Output #8
-h----- Function for hardware Output #7
--h---- Function for hardware Output #6
---h--- Function for hardware Output #5
----h-- Function for hardware Output #4
-----h- Function for hardware Output #3
------h Function for hardware Output #2
-------h Function for hardware Output #1

Descriptions

Function 0 Programmable Output set by user program (G47 code).
This output is inactive on Power-On or following a "Clear" command.

Function 1 Programmable Output set by user program (G47 code).
This output is active on Power-On or following a "Clear" command.

Function 2 Motion busy
This output is active if either the X or Y axis is moving.

Function 3 System busy
This output is active when either axis is moving or program execution is occurring.

Function 4 At hard limit
This output is active if motion was terminated or inhibited by a travel limit and the source of that limit was a hardware input.

Function 5 At clockwise limit
This output is active if motion was terminated or inhibited by a CW travel limit. The travel limit can be from an input or the CW software travel limit (L18).

Function 6 At counter clockwise limit
This output is active if motion was terminated or inhibited by a CCW travel limit. The travel limit can be from an input or the CCW software travel limit (L19).

Function 7 At home position (electrical zero)
This output is active if the selected axis is at absolute position 0.

Function 8 Position error (closed loop option)
This output is active when the magnitude of the position error exceeds the position deadband (L93) value at standstill.

Function 9 X axis active
This output is active when the X axis is moving or selected.

Function A — Y axis active
This output is active when the Y axis is moving or selected.

Function B — PLC Strobe Output
Reading PLC data requires 4 strobes, one for each byte (2 BCD digits) of data. When used, this function must be assigned to Outputs #5-#8.

Function C — Fault
This output is active when the fault lamp is on.

Function D — Following error (closed loop option)
This output is active when the magnitude of the position error exceeds the following error (L87) value during motion.

Function E — Closed Loop Correction Cycle (closed loop option)
This output is active during the motion cycle that attempts to correct a following error.

Function F — Unable to Correct (closed loop option)
This output is active when all the allotted (L96) attempts to correct the position or following error have failed.

Related parameters L20 (System Configuration)
L58 (Input Configuration)

Example

L59 "BBBB0000"	Outputs #5-#8, PLC Strobes Outputs #1-#4, Programmable Outputs, inactive on Power-On or Clear.
L59 "11110000"	Outputs #5-#8, Programmable Outputs, active on Power-On or Clear. Outputs #1-#4, Programmable Outputs, inactive on Power-On or Clear.

L66 Backlash Compensation

Function Move distance to compensate for backlash.

Command Format

L66 snnnnnnnn,snnnnnnnn	Set X and Y axis backlash compensation distance.
L66 snnnnnnnn	Set X axis backlash compensation distance.
L66 ,snnnnnnnn	Set Y axis backlash compensation distance.

Range snnnnnnnn -99,999,999 to +99,999,999 pulses

Default +0,+0 (disabled)

Notes Backlash compensation is not performed with Mark Registration, step or jog motion.

Backlash Compensation is only performed when the move direction and the backlash compensation (L66) sign are the same. When this occurs, the motor is rotated past its target position by the backlash compensation distance. Following a delay time (L43) the motor is rotated back, in the opposite direction, the backlash compensation distance.

Related parameters L43 (Backlash Compensation Delay)

Examples

L66 +100,+100	Sets X and Y backlash compensation distance to 100 pulses in the CW direction.
L66 -100	Sets X axis backlash compensation distance to 100 pulses in the CCW direction.

L71 Maximum Speed

Function Maximum speed for all motion.

Command Format L71 nnnnnn

Range nnnnnn 1 - 999,999 pulse/sec.

Default Value equivalent to 10,000 Full Steps/sec, for microstepping resolution up to 100. Exact number depends on selected microstepping resolution. See Default table in Appendix A.

Notes

1) The maximum speed value applies to both X and Y. The speed of any motion is not allowed to exceed this value.

2) For best motor performance, set L71 as low as possible. When L71≤ 115,000, an internal pulse-smoothing algorithm is enabled. Therefore, when using drives with resolutions of full, half, 1/5, and 1/10 step it is best to keep L71 set no higher than 115,000.

Example L71 20000 Sets the maximum speed to 20,000 pulses/sec.

L72 Velocity Profile

Function Selects the velocity profile used during all motion.

Command Format L72 n

Range
0 = trapezoidal profile
1 = hyperbolic profile
2 = "S" shaped profile

Default: L72 0 (trapezoidal profile)

Notes The velocity profile determines the manner in which the accel/decel varies with speed, refer to illustrations of each profile.

Trapezoidal profile - The accel/decel is constant.

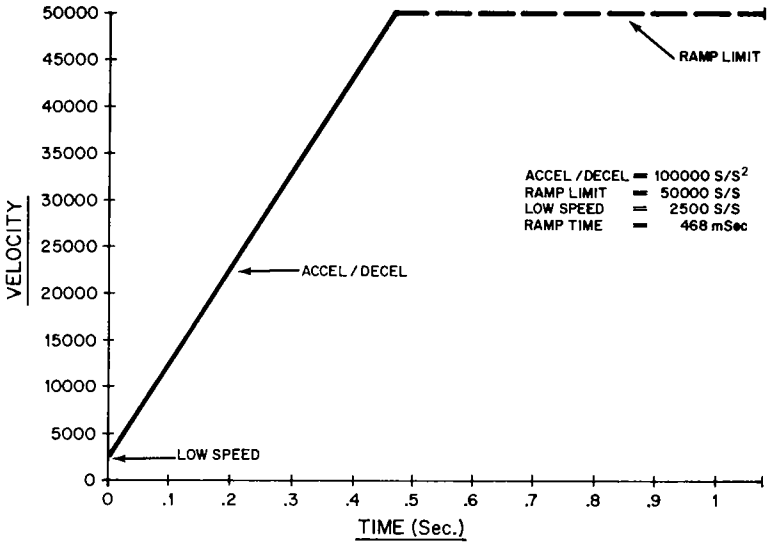
Hyperbolic profile - The accel/decel is at its maximum value at low speed and decreases to its minimum value at high speed.

"S" profile - The accel/decel is at its minimum value at low speed, increases to its maximum and then decreases back to its minimum value at high speed.

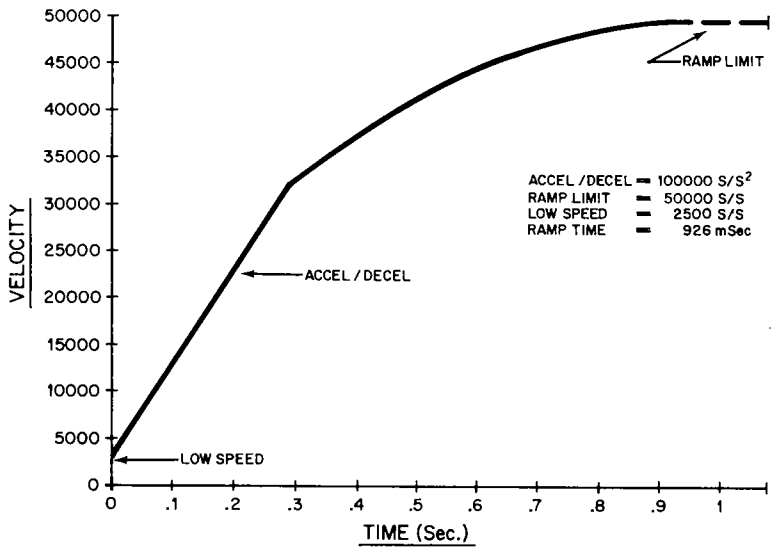
With some loads, a variable acceleration ("S" or Hyperbolic) will permit the load to achieve higher speeds than would be possible with constant acceleration (Trapezoidal).

For a given accel (L11) or decel (L10), the trapezoidal profile results in the shorter accel/ decel times.

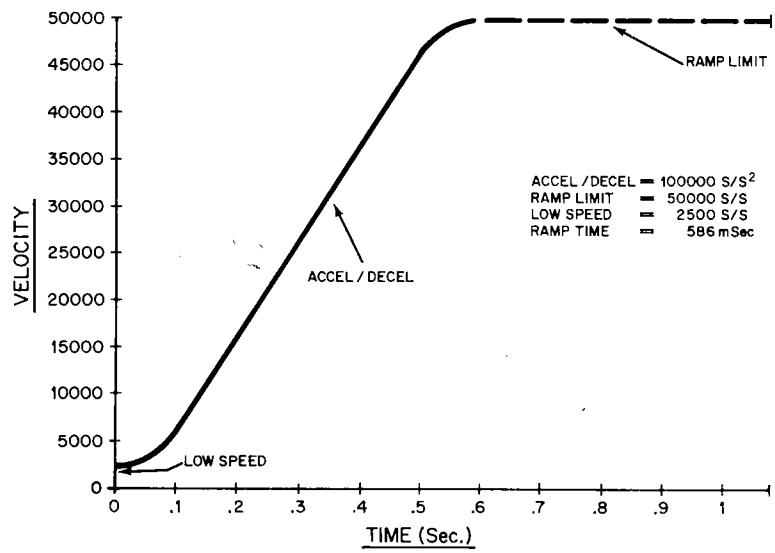
Related parameters
L10 (Deceleration)
L11 (Acceleration)



TRAPEZOIDAL PROFILE



HYPERBOLIC PROFILE



"S" PROFILE

L73 Speed Increment

Function Size of the speed change that occurs in response to serial port commands Target Velocity Increase (H31) and Target Velocity Decrease (H32).

Command Format L73 nnnnnn

Range nnnnnn 1 - 999,999 pulses/sec.

Default Value equivalent to 100 Full Steps/sec, exact number depends on selected microstepping resolution. See Default table in Appendix A.

Notes The speed can not be increased above the maximum speed (L71) or decreased below 0.

Example L73 100 Sets the speed increment to 100 pulses/sec.

L91 External Move Distance Scale Factor

Function Scale factor used with external BCD move data.

Command Format

L91 nnnnnnnn	(scale factor for X axis)
L91 ,nnnnnnnn	(scale factor for Y axis)
L91 nnnnnnnn,nnnnnnnn	(scale factors for X and Y)

Range nnnnnnnn 0 - 99999999

Default 00010000,00010000 (X and Y scale factor = 1.0000)

Notes The scale factor has 4 fractional and 4 whole digits. (10000 = 1.0000).

The scale factor is only used with the following G codes:

- G10 (Mark Registration Cycle)
- G36 (Read Move Distance)
- G79 (Set Absolute Position)

When one of the above codes is executed and the distance is not specified, the external data is read and multiplied by the scale factor to form the distance.

Examples L91 105000,15000 X axis scale factor = 10.5000
Y axis scale factor = 1.5000

L98 Delay Between Automatic Transmissions

Function Sets the time between repeated transmissions of data called for by continuous (%) H code commands.

Command Format L98 nnnn

Range nnnn 0 - 9,999 milliseconds

Default 100 milliseconds

Notes The continuous H code commands are:

%H13 Transfer Current Program Line
%H15 Transfer Current Line Number
%H17 Transfer Active Axis Absolute Position
%H19 Transfer Mode Status
%H20 Transfer Output Status
%H21 Transfer Input Status
%H22 Transfer Encoder Position
%H26 Transfer Error Count
%H27 Transfer Position Verification Status
%H60 Transfer Present Velocity
%H88 Transfer Execution Status

The remaining L codes pertain to the closed loop option and its operation.

L87 Following Error

Function Magnitude of position error, while motion is occurring, that signals a following error.

Command Format L87 nnnn

Range nnnn 0 - 9,999 pulses

Default Value equivalent to 10 Full Steps, exact number depends on selected microstepping resolution. See Default table in Appendix A.

Notes Closed loop operation is only available on the X axis.

The Following Error should be greater than four full steps (Drive Resolution/50).

200/50 = 4 (four full steps)
25000/50 = 500 (four full steps)

Related parameters L93 (Position Deadband)

Example L87 200 Sets the Following Error at 200 pulses.

L90 Closed Loop Configuration

Function Sets the closed loop operating mode.

Command Format L90 n

Range n 0 - 3

Default 0 (Open Loop Mode)

Notes The Encoder position and the position error are calculated in all operating modes. The encoder position can be examined with a H22 command.

A Following Error occurs when the magnitude of the position error exceeds the Following Error (L87) during motion.

L90=0 Encoder feedback has no effect on operation. (Open Loop Mode)

L90=1 Position maintenance is enabled. When a Following Error occurs, motion is terminated and the fault light turned on.

L90=2 Position maintenance is enabled. When a Following Error occurs, motion is terminated, then restarted in an attempt to complete the original move. If after L96 attempts, the move is not completed, the cycle is terminated and the fault light turned on. Position Verification Status will indicate an "unable to correct" condition.

L90=3 Position maintenance is enabled. When a Following Error occurs, indexed motion is terminated, the motor is returned to the position from which it started, then the original move retried. Up to L96 attempts are allowed for returning to the starting position and also for completing the original move. If either the return to starting position or subsequent completion of the original move is unsuccessful, the active cycle is terminated and the fault light turned on. Position Verification Status will indicate an "unable to correct" condition.

A Following Error occurring during jog or any motion without a target position, will be handled in the same manner as L90=2 mode.

Position maintenance

Maintenance is only active while motion is not being commanded (standstill). If the magnitude of the position error exceeds the Position Deadband (L93), the motor is moved so as to produce zero position error.

Attempts to correct the position error will be limited to the Position Corrections Attempts (L96) number, at which point the fault light is turned on. Position Verification Status will indicate an "unable to correct" condition.

L93 Position Deadband

Function Magnitude of position error, at standstill, that signals a position error.

Command Format L93 nnnn

Range nnnn 0 - 9,999 pulses

Default see default table Appendix A.

Note If L95 (Encoder Quadrature Scale Factor) is set to 5, the L93 value should be a multiple of 5.

Related parameters L87 (Following Error)

Example L93 1 Sets the position deadband to 1 pulse.

L94 Encoder Counter Direction

Function Sets the encoder position counter direction (up or down) vs. encoder output Ch.A & Ch.B phasing.

Command Format L94 n

Range
n = 0 count up if Ch.A leads Ch.B
n = 1 count down if Ch.A leads Ch.B

Default 1 (count down, Ch.A leads Ch.B)

Notes For the closed loop to work, the encoder position and X absolute position counters must both count in the same direction.

Related parameters L95 (Encoder Scale Factor)

Example L94 1 count down if, Ch.A leads Ch.B

L95 Encoder Scale Factor

Function Scales the encoder counts/rev. to match the motor pulses/rev.

Command Format L95 n

Range
0 = no scaling
2 = divide encoder counts/rev by 2
4 = divide encoder counts/rev by 4
5 = multiply encoder counts/rev by 5
8 = divide encoder counts/rev by 8

Default see default table Appendix A.

Related parameters L94 (Encoder Counter Direction)

Examples A 400 line Encoder (1600 pulses/rev) is used with a half step drive (400 pulses/rev). The scale factor required is divide by 4. (L95 = 4)

A 1250 line Encoder (5000 pulses/rev) is used with a 1/125 microstepping drive (25000 pulses/rev). The scale factor required is multiply by 5. (L95 = 5)

L96 Position Correction Attempts

Function Number of consecutive correction attempts allowed for Position Maintenance or for the correction of a Following Error.

Command Format L96 nnnn

Range nnnn 0 - 9,999 attempts

Default 100 attempts

Notes When the number of attempts to correct a position error equals the L96 value, an unable to correct flag is set in Position Verification status. No further attempts will be allowed until a new motion is started.

Related parameters L97 (Delay Between Correction Attempts)

Example L96 200 Sets number of correction attempts to 200.

L97 Delay Between Correction Attempts

Function Delay between correction attempts for Position Maintenance or Following Error correction.

Command Format L97 nnnn

Range nnnn 0 - 9,999 milliseconds

Default 100 milliseconds

Notes When a following error or a motion clear occurs, the encoder position is transferred to the X absolute position, L97 milliseconds later.

Related parameters L96 (Position Correction Attempts)

Example L97 1000 Sets the delay between correction attempts to 1000 milliseconds.

This page intentionally left blank

G CODES

PROGRAMMING COMMAND (G-CODE) FUNCTIONAL LISTING

PROGRAM COMMANDS
G04 Delay
G11 Call Subroutine
G12 Go to line Number
G25 Loop start
G26 Loop end
G30 End Of Program
G31 Program stop
G32 Return from subroutine
G90 Absolute position mode
G91 Incremental position mode

HOMING
G76 Return to electrical home
G77 Set electrical home
G78 Return to mechanical home
G79 Set absolute position

INPUT/OUTPUT
G20 Branch on input condition
G22 Wait for input condition
G47 Set output(s)

MOTOR CURRENT CONTROL
G64 Enable reduce current
G65 Disable reduce current
G66 Enable boost current
G67 Disable boost current
G68 Motor windings off
G69 Motor windings on

SPEED CONTROL
G23 Increase feed rate
G24 Decrease feed rate
G27 High speed mode
G28 Low speed mode

ADVANCED PROGRAM COMMANDS
G05 Transmit message
G06 Simulate the reception of an Hcode
G19 Branch on register result
G21 Register control
G29 Set designated L code
G50 Set flag(s)
G51 Branch on flag condition
G54 Register execution
G55 Error trapping

EXTERNAL DATA
G29 Set designated L code
G36 Read move distance
G37 Read line number and branch
G38 Read feed rate
G52 Select external data source
G53 Select SS2000 interface I/O port

DYNAMIC MOTION
G10 Mark registration cycle
G48 Start jog cycle
G49 Stop jog cycle

BREAK POINT MOTION
G60 Enable continuous execution mode
G61 Disable continuous execution mode
G62 Wait for move distance
G63 Wait for velocity condition

G04 Delay

Function Delay program execution.

Command Format
G04 Xnnnn {Fnnnnnn}
G04 Ynnnn {Fnnnnnn}

Range

X or Y field	0 - 9,999	delay time (milliseconds)
F field	0 - 999,999	feedrate (pulses/sec)

Notes

This command can not be used on program line 0 (MDI mode)

If no value has been entered for the X or Y field, or the value is out of range, program execution will terminate.

The F field, if programmed, will become the new feedrate after the delay time has elapsed.

Examples

N1 G04 X1000	delays 1 second.
N1 G04 Y2000	delays 2 seconds.
N1 G04 X1000 F2000	delays 1 second then changes the feedrate to 2000 pulses/second.

G05 Transmit Message

Function Transmit the message number specified by the X or Y field. The messages are stored in parameters L01 thru L05.

Command Format
G05 Xn {Fnnnnnn}
G05 Yn {Fnnnnnn}

Range

X or Y field	1 - 5	message number
F field	0 - 999,999	feedrate (pulses/sec)

Notes

The F field, if programmed, will become the new feedrate.

If no value has been entered for the X or Y field, or the value is out of range, program execution will terminate.

Related commands
L01 "Message 1"
L02 "Message 2"
L03 "Message 3"
L04 "Message 4"
L05 "Message 5"

Examples

N1 G05 X1	send message #1 (L01).
N1 G05 Y2	send message #2 (L02).
N1 G05 X1 F2000	send message #1 (L01) and change the feedrate to 2000 pulses/second.

G06 Hcode execution

New code per Rev. C

Function Simulate the reception of an Hcode

Command Format
G06 Xnn
G06 Ynn

Range X or Y field any legal immediate Hcode number

Notes The F field, if programmed, is ignored.

If no value has been entered for the X or Y field, or the value is out of range, program execution will terminate.

Legal values for X or Y field

04	High Speed Mode
05	Low Speed Mode
13	Transfer Current Program Line
15	Transfer Current Line Number
17	Transfer Absolute Electrical Position
19	Transfer Mode Status
20	Transfer Output Status
21	Transfer Conditional Input Status
22	Transfer Encoder Position
24	Program Trace Mode On
25	Program Trace Mode Off
26	Transfer Pulse Error Count
27	Transfer Closed Loop Status
31	Target Velocity Increase
32	Target Velocity Decrease
41	Transfer Remaining Repeat Value
42	Transfer Last External BCD Value
43	Transfer OPTO 22 Input/Output Status
44	Transfer Register Contents
60	Transfer Present Velocity
85	Transfer Motion Error Status
86	Transfer Data Error Status
87	Transfer Program Error Line Number
88	Transfer Execution Status

Examples

N1 G06 X17	transmit present absolute position
N1 G06 X22	transmit present encoder position.

G10 Mark Registration Cycle

Function

Execute a Mark Registration cycle.

Command Format

G10 X {Fnnnnnn}	X axis, using external data
G10 Xsnnnnnnnn {Fnnnnnn}	X axis, using X field data
G10 Y {Fnnnnnn}	Y axis, using external data
G10 Ysnnnnnnnn {Fnnnnnn}	Y axis, using Y field data
G10 {Fnnnnnn}	Active axis, using external data

Range

X or Y field -99,999,999 to +99,999,999 index distance
F field 0 - 999,999 feedrate (pulses/sec)

Notes

This command can not be used during a G48 jog cycle.

The index distance of a Mark Registration cycle is always incremental.

The cycle begins as a Jog cycle. When the registration input occurs, the index portion of the cycle begins.

Parameter L16 can be used to limit the move distance during a Registration cycle. This feature allows the cycle to be terminated in the event of a loss of the registration input.

The motor speed is limited so that the distance needed to decelerate to a stop is less than the index distance. In some cases this results in a speed less than the feedrate.

The F field, if programmed, will become the new feedrate.

The repeatability of the Registration cycle is ±1 pulse.

Related commands

L16 (Registration Travel Limit)
L20 (TR1 & TR2 Configuration)
G52 (Select External Data Source)

Examples

N1 G10 X	Mark Registration using X axis. Index distance is external BCD data multiplied by the L91 X axis scale factor.
N2 G10 Y1000	Mark Registration using Y axis. Index distance is 1000.

G11 Call Subroutine

Function

Call the subroutine at the line indicated by the X or Y field and repeat the number of times indicated by the F field.

Command Format

G11 Xnnn Fnnnn
G11 Ynnn Fnnnn

Range

X or Y field	1 - 400	line number
F field	0 - 9,999	repeat count

Notes

This command can not be used on program line 0 (MDI mode)

The X or Y field is the subroutine line number. The F field is the subroutine repeat count. The total number of times the subroutine is executed is the F field value plus one.

Subroutine nesting (calling a subroutine from within a subroutine) is allowed up to four levels. Nesting beyond four levels will cause program execution to terminate.

When a Return From Subroutine (G32) command is encountered during, the Control returns to the line calling the subroutine and the cycle is repeated F + 1 times. The Control will then go on to the next program line.

Related commands

G32 (Return From Subroutine)

Example

```
N1 G11 X100 F3          call subroutine line 100 & repeat 3 times
N2 G30          end of program

N100 Y+500 F1000        moves Y+500 at 1000 pulses/sec
N101 G32                return from subroutine
```

G12 Go To Line Number

Function Branch to the line indicated by the X or Y field

Command Format
G12 Xnnn {Fnnnnnn}
G12 Ynnn {Fnnnnnn}

Range

X or Y field	1 - 400	line number
F field	0 - 999,999	feedrate (pulses/sec)

Notes This command can not be used on program line 0 (MDI mode)

The F field, if programmed, will become the new feedrate.

Example N10 G12 X2 Program execution branches to line 2

G19 Branch On Result

Function

This command allows the program to branch to a specified program line based on the result of a register-to-register comparison.

Command Format

G19 X0000nnnn Fnnn Branch if comparison result matches.

Range:

F field 1 to 400 line number
X field as shown below register comparison result

X0000nnnn

0000n--- Comparison result greater than or equal to (>=)
0000-n-- Comparison result greater than (>)
0000--n- Comparison result equal to (=)
0000---n Comparison result less than (<)

n = 0 false
n = 1 true
n = 2 don't care

Notes

If the field data are out of range, program execution will be terminated.

Related Commands

G21 Register Control
G52 Select External Data Source
G54 Register Execution

Example

Program select 1-4 via BCD input. Program 1 starts at line 100, Program 2 starts at line 150, Program 3 starts at line 200, and Program 4 starts at line 250.

```
N1  G52  X1          select BCD bank 1
N2  G21  F30         BCD data > register 3
N3  G21  X1  F40     1 > register 4
N4  G21  X4  F34     BCD data = 1?
N5  G19  X2212  F100 if yes, go to line 100
N6  G21  X2  F40     2 > register 4
N7  G21  X4  F34     BCD data = 2?
N8  G19  X2212  F150 if yes, go to line 150
N9  G21  X3  F40     3 > register 4
N10 G21  X4  F34     BCD data = 3?
N11 G19  X2212  F200 if yes, go to line 200
N12 G21  X4  F40     4 > register 4
N13 G21  X4  F34     BCD data = 4?
N14 G19  X2212 F250  if yes, go to line 250
N15 G12  X2          go to line 2
```

G20 Branch on Input Condition

Function

If the condition specified by the X or Y field occurs, the program branches to the line specified by the F field, otherwise the next line is executed.

Command Format

G20 Xnnnnnnnn Fnnn (check Programmable Input conditions)
G20 Ynnnnnnnn Fnnn (L20 nn1nn check Expansion input conditions)
G20 Ynnnn Fnnn (L20 nn2nn check SS2000 Expansion input conditions)

Range

X or Y field 0 - 99,999,999 input condition
F field 1 - 400 line number

Notes

This command can not be used on program line 0 (MDI mode)

Each digit in the X or Y field represents an input as shown below.

Xnnnnnnnn	L20 nn1nn Ynnnnnnnn	L20 nn2nn Ynnnn
n------ Input 8	n------ Expansion Input 8	n--- S2000 Expansion Input 4
-n----- Input 7	-n----- Expansion Input 7	-n-- S2000 Expansion Input 3
-n----- Input 7	-n----- Expansion Input 7	--n- S2000 Expansion Input 2
--n---- Input 6	--n---- Expansion Input 6	---n S2000 Expansion Input 1
---n--- Input 5	---n--- Expansion Input 5	
----n-- Input 4	----n-- Expansion Input 4	
-----n- Input 3	-----n- Expansion Input 3	
------n Input 2	------n Expansion Input 2	
------n Input 1	------n Expansion Input 1	

The input condition "n" for each input is a value 0 - 9. A 0 specifies an inactive input, a 1 an active input and 2 - 9 that the input state doesn't matter. Programming the X field will cause the inputs that have been configured as programmable inputs, via the L58 parameter, to be tested.

Programming the Y field will cause the expanded inputs to be tested. If the Y field is programmed and the expansion port is not configured for I/O, the programmable inputs will be tested.

The L20 parameter determines which set of expansion I/O to test. L20 nn1nn selects the Expansion inputs, while L20 nn2nn selects the S2000 Expansion board inputs (4 inputs per port).

Related commands

L20 (System Configuration)
L58 (Input Configuration)

Example

The following example has inputs 1 - 8 configured as testable inputs and the expansion port as I/O.

```
N1 G20 X22220011 F10   If Inputs 1,2,3,4 = 1,1,0,0 respectively then go to line 10
N2 G20 Y11022220 F10   If Expansion Inputs 1,6,7,8 = 0,0,1,1 respectively then go to line 10.
N3 G12 X1                                                    Go to line 1.

N10 X10000 F10000      Index X axis 10,000 pulses at 10,000 pulses/sec.
N11 G30                Program end
```

G21 Register Control

Modified per Rev. C

Function

This command controls nine registers; two dedicated and seven general purpose. These registers can be loaded, adjusted or compared.

Command Format

G21 Xsnnnnnnnn Fn0	load register Fn (1-9) with X value
G21 Fn0	load register Fn (1-9) with external BCD data
G21 Fn1	scale register Fn (1-9) using L91 X/Y scale factor
G21 Xn Fn2	move contents of register Xn (1-9) to register Fn (1-9)
G21 X+n Fn3	add register Xn (1-9) to register Fn (1-9), result in register Fn
G21 Y+nnnnnnnn Fn3	add nnnnnnnn to Register Fn (1-9) result in Register Fn
G21 X-n Fn3	subtract register Xn (1-9) from register Fn (1-9), result in register Fn
G21 Y-nnnnnnnn Fn3	subtract nnnnnnnn from Register Fn (1-9), result in Register Fn
G21 Xn Fn4	compare register Xn (1-9) with register Fn (1-9)
G21 Ysnnnnnnnn Fn4	compare Ysnnnnnnnn to Register Fn (1-9)
G21 X Fn5	load register Fn (1-9) with X absolute position
G21 Y Fn5	Load register Fn (1-9) with Y absolute position
G21 X Fn6	load X absolute position with register Fn (1-9)
G21 Y Fn6	load Y absolute position with register Fn (1-9)
G21 X Fn7	load register Fn (1-9) with X encoder position
G21 X Fn8	load Feedrate with register Fn (1-9)
G21 Xnn Fn9	load L nn X axis parameter with register Fn (1-9)
G21 Ynn Fn9	load L nn Y axis parameter with register Fn (1-9)

Range

Function 0	X field +99,999,999 to -99,999,999 F field 10-90
Function 1	X field none F field 11-91
Function 2	X field 1-9 F field 12-92
Function 3	X field 1-9 Y field -99999999 to +99999999 F field 13-93
Function 4	X field 1-9 Y field -99999999 to +99999999 F field 14-94
Function 5	X field none F field 15-95
Function 6	X field none F field 16-96
Function 7	X field none F field 17-97
Function 8	X field none F field 18-98
Function 9	X field any legal L code Y field any legal L code F field 19-99

Notes

The register assignments are as follows:

register 1	event counter 1 (counts inactive to active transitions on input 1)
register 2	event counter 2 (counts inactive to active transitions on input 2)
register 3	general purpose register
register 4	general purpose register
register 5	general purpose register
register 6	general purpose register
register 7	general purpose register
register 8	general purpose register
register 9	general purpose register

Register 1 counts inactive to active transitions on the input terminal assigned to Program testable Input 1. Refer to the L58 code in this manual for assignment

Register 2 counts inactive to active transitions on the input terminal assigned to Program testable Input 2. Refer to the L58 code in this manual for assignment

The LSB digit of the F field selects the function to be performed. The functions are:

0	load register Fn (1-9) with X value or BCD data
1	scale register Fn (1-9) using L91 scale factor
2	move register Xn (1-9) to register Fn (1-9)
3	add (X+n) or subtract (X-n) register Xn to or from register Fn (1-9)
4	compare register Fn (1-9) with the X or Y absolute position
5	load register Fn (1-9) with the X or Y absolute position
6	load the X or Y absolute position with register Fn (1-9)
7	load register Fn (1-9) with the X encoder position
8	load Feedrate with register Fn (1-9)
9	load L code parameter with register Fn (1-9)

If the F field data are out of range, program execution will be terminated.

Register scaling (G21 Fn1) uses the active axis scale factor (L91 value)

All registers are cleared when a Cycle Start is executed, but left unchanged if resuming from a Feedhold or Program Stop. If L20 nnnXn is 3, 4, 5 the registers are not cleared.

Related Commands

G52	Select External Data Source
G54	Register Execution
G58	Input Configuration
L91	Scale Factor

Examples

Incrementally index Y axis 10 times using BCD data and return to starting position

N1 G91 Y F1000	select Y axis and feedrate of 1000 pps
N2 G90	absolute position mode
N3 G52 X1	select BCD data bank 1
N4 G21 F20	BCD data > register 2
N5 G21 F21	scale register 2 BCD data using L91 Y value
N6 G21 Y F95	Y absolute position > register 9
N7 G21 X9 F82	register 9 > register 8
N8 G25 X9	loop 10 times
N9 G21 X2 F83	add register 2 to register 8
N10 G54 Y+8 F0	index Y axis to register 8 position
N11 G26	loop end
N12 G54 Y+9 F0	index Y axis to starting position

Test encoder position. If different than X absolute position, load encoder position into X absolute position.

N1 G21 X F37	encoder position > register 3
N2 G21 X F45	X absolute position > register 4
N3 G21 X3 F44	compare register 3 with register 4
N4 G19 X2212 F6	branch if registers alike to line 6
N5 G21 X F36	encoder position > X absolute position
N6 G30	

G22 Wait for Input Condition

Function

Wait until the input states match the X or Y field value.

Command Format

G22 Xnnnnnnnn {Fnnnnnn}	(wait for Prog. Input conditions)
G22 Ynnnnnnnn {Fnnnnnn}	(L20 nn1nn wait for Expansion input conditions)
G22 Ynnnn {Fnnnnnn}	(L20 nn2nn wait for S2000 Expansion input condition)

Range

X or Y field	0 - 99,999,999	input condition
F field	0 - 999,999	feedrate (pulses/sec)

Notes

This command can not be used on program line 0 (MDI mode).

The F field, if programmed, will become the new feedrate after the input condition is satisfied.

Each digit in the X or Y field represents an input as shown below.

Xnnnnnnnn	L20 nn1nn Ynnnnnnnn	L20 nn2nn Ynnnn
n------ Input 8	n------ Expansion Input 8	n--- S2000 Expansion Input 4
-n----- Input 7	-n----- Expansion Input 7	-n-- S2000 Expansion Input 3
--n---- Input 6	--n---- Expansion Input 6	--n- S2000 Expansion Input 2
---n--- Input 5	---n--- Expansion Input 5	---n S2000 Expansion Input 1
----n-- Input 4	----n-- Expansion Input 4	
-----n- Input 3	-----n- Expansion Input 3	
------n Input 2	------n Expansion Input 2	
------n Input 1	------n Expansion Input 1	

The input condition "n" for each input is a value 0 - 9. A 0 specifies an inactive input, a 1 an active input and 2 - 9 that the input state doesn't matter.

Programming the X field will cause the inputs that have been configured as programmable inputs, via the L58 parameter, to be tested.

Programming the Y field will cause the expanded inputs to be tested. If the Y field is programmed and the expansion port is not configured for I/O, the programmable inputs will be tested.

The L20 parameter determines which set of expansion I/O to test. L20 nn1nn selects the Expansion Inputs, while L20 nn2nn selects the S2000 Expansion board Inputs (four inputs per port).

Related commands

L20 (System Configuration)
L58 (Input Configuration)
G53 (Select S2000 Interface Input Port)

Example

The following example has inputs 1 - 8 configured as testable inputs and the expansion port as I/O.

N1 G22 X22220011 F10	Wait until Inputs 1,2,3,4 = 1,1,0,0 respectively.
N2 G22 Y11022220 F10	Wait until Expansion Inputs 1,6,7,8 = 0,0,1,1 respectively.
N10 X10000 F10000	Index X axis 10,000 pulses at 10,000 pulses/sec.
N11 G30	Program end

G23 Increase Feedrate

Function Increase feedrate by the Feedrate increment (L73).

Command Format G23

Notes This command can not be used on program line 0 (MDI mode) or during normal execution mode.

Each time G23 is executed, the feedrate is increased by the feedrate increment, but not beyond the feedrate limit (L71). Changes to the feedrate using G23, during a G48 jog cycle, are not saved.

If a value is entered for the X or Y field, program execution will be terminated.

Related Commands L73 (Feedrate increment)
L71 (Feedrate limit)

Example Feedrate increment L73 = 100

```
N1 G48 X+ F500      start Jog cycle, feedrate = 500 pulses/sec
N2 G23              increment feedrate by 100 pulses/sec
N3 G04 X1000        delay 1 second
N4 G49              stop Jog cycle
N5 G30              end of program
```

G24 Decrease Feedrate

Function Decrease feedrate by the Feedrate Increment (L73).

Command Format G24

Notes This command can not be used on program line 0 (MDI mode) or during normal execution mode.

Each time G24 is executed, the feedrate is decreased by the feedrate increment, but not below zero. Changes to the feedrate using G24, during a G48 jog cycle, are not saved.

If a value is entered for the X or Y field, program execution will be terminated.

Related Commands L73 (Feedrate Increment)

Example Feedrate increment L73 = 100

```
N1 G48 X+ F500      start Jog cycle, feedrate = 500 pulses/sec
N2 G24              decrease feedrate by 100 pulses/sec
N3 G04 X1000        delay 1 second
N4 G49              stop Jog cycle
N5 G30              end of program
```

G25 Loop Start
G26 Loop End

Function

Defines the start (G25) and the end (G26) of a Loop. The loop will be executed the number of times specified by the X or Y field plus one.

Command Format

G25 Xnnnn {Fnnnnnn}	nnnn = repeat count
G25 Ynnnn {Fnnnnnn}	nnnn = repeat count
G26 Xnnnn {Fnnnnnn}	repeat count is read from external data
G26 Ynnnn {Fnnnnnn}	

Range

X or Y field	0 - 9,999	repeat count
F field	0 - 999,999	feedrate (pulses/sec)

Notes

These commands cannot be used on program line 0 (MDI mode)

The loop executes lines from the loop start (G25) to the loop end (G26). When the loop is completed, execution continues with the line following the loop end (G26).

The X or Y field is the loop repeat count. The number of loop executions is the X or Y value plus 1. If the X or Y value is out of range, program execution will terminate. If no value has been entered for the X or Y field, the repeat count will be read from the selected external data source.

Loop nesting, calling another loop from within a loop, beyond four levels is not allowed. If loop nesting beyond four levels occurs, program execution will terminate.

The F field, if programmed, will become the new feedrate.

Example

Move Y+500, 4 times.

N1 G25 X3	Loop start; repeat 3 times
N2 Y+500	move Y+500 pulses
N3 G26	end of loop
N4 G30	end of program

G27 High Speed Mode

Function Sets the High Speed Mode of Operation.

Command Format

G27 X {Fnnnnnn}	Set X axis to High Speed Mode
G27 Xsnnnnnnnn {Fnnnnnn}	Set X axis to High Speed Mode followed by a move
G27 Y {Fnnnnnn}	Set Y axis to High Speed Mode
G27 Ysnnnnnnnn {Fnnnnnn}	Set Y axis to High Speed Mode followed by a move
G27 {Fnnnnnn}	Set active axis to High Speed Mode.

Range

X or Y field	-99,999,999 to +99,999,999 index distance
F field	0 - 999,999 feedrate (pulses/sec)

Default Mode High Speed Mode

Notes The X or Y field value, if programmed, will become the index distance and direction.

The F field, if programmed, will become the new feedrate.

Example N1 G27 X Sets the X axis to High Speed Mode

G28 Low Speed Mode

Function Sets the Low Speed Mode of Operation.

Command Format

G28 X {Fnnnnnn}	Set X axis to Low Speed Mode
G28 Xsnnnnnnnn {Fnnnnnn}	Set X axis to Low Speed Mode followed by a move
G28 Y {Fnnnnnn}	Set Y axis to Low Speed Mode
G28 Ysnnnnnnnn {Fnnnnnn}	Set Y axis to Low Speed Mode followed by a move
G28 {Fnnnnnn}	Set active axis to Low Speed Mode.

Range

X or Y field	-99,999,999 to +99,999,999 index distance
F field	0 - 999,999 feedrate (pulses/sec)

Default Mode High Speed Mode

Notes The X or Y field value, if programmed, will become the index distance and direction.

The F field, if programmed, will become the new feedrate.

Example N1 G28 X Sets the X axis to Low Speed Mode

G29 Set Designated L Code

Function

The L code, designated by the (F field), is loaded with the X or Y field data.

Command Format

G29 Xsnnnnnnnn Fnn	X axis L code "nn" = "snnnnnnnn"
G29 X Fnn	X axis L code "nn" = external data
G29 Ysnnnnnnnn Fnn	Y axis L code "nn" = "snnnnnnnn"
G29 Y Fnn	Y axis L code "nn" = external data
G29 Fnn	active axis L code "nn" = external data

Range

X or Y field	range depends on L code	L code data
F field	valid entries listed below	L code

The following L codes can be set using this command:

L08 Mechanical Home Direction
L09 Jog Speed
L10 Deceleration
L11 Acceleration
L12 Low Speed
L13 Step Size
L14 Home Speed
L16 Index From Run Travel Limit
L17 Mechanical Home Offset
L18 Software CW Limit
L19 Software CCW Limit
L20 System Configuration
L41 Auto Start Line Number
L43 Delay between Index & Backlash
L44 Delay after Motion
L47 Program Execution Repeat Count
L66 Backlash Compensation
L71 Ramp Frequency Limit
L73 Deviation Frequency
L87 Following Error
L90 Position Verification Enable/Disable
L91 External Move Distance Scale Factor
L93 In Position Bandwidth

Notes

This command cannot be used during a G48 jog cycle or continuous execution mode.

If the X or Y field data is omitted the data will be read from the externally selected data source.

Examples

N1 G29 X+1000 F09	the X axis jog speed (L09) = 1000
N2 G29 Y F11	the Y axis accel (L11) = external data

G30 End of Program

Function Denotes the end of the program and resets the line pointer to the line designated by the L41 parameter.

Command Format G30

Notes This command can not be used on program line 0 (MDI mode)

Following the execution of line 400, the line pointer is set to the line designated by the L41 parameter.

Related Commands L41 (Auto Start Line Number)
L47 (Repeat Count)

G31 Program Stop

Function The program stops execution and increments the line pointer. Index motion is allowed to finish and Jog motion is stopped.

Command Format G31

Notes This command can not be used on program line 0 (MDI mode)

Program execution can be resumed, with a Cycle Start command.

G32 Return from Subroutine

Function Indicates the end of a subroutine.

Command Format G32

Notes This command can not be used on program line 0 (MDI mode)

When encountered within a subroutine, the line pointer is restored to the line number from which the subroutine call (G11) was made.

If this command is encountered outside of a subroutine, program execution will terminate.

Related commands G11 (Call Subroutine)

G36 Read Move Distance

Function	Read data from the selected external data source, multiply by the L91 scale factor and use as the index distance.	
Command Format	G36 {Fnnnnnn}	Move the active axis
	G36 X {Fnnnnnn}	Move the X axis
	G36 Y {Fnnnnnn}	Move the Y axis
Range	External data -79,999,999 to +79,999,999 index distance	
Notes	This command can not be used during a G48 jog cycle.	
	The move distance is treated as either incremental or absolute, depending on the selected mode.	
	The F field, if programmed, will become the new feedrate.	
Related commands	G52 (Select External Data Source)	
Example	N1 G36 F1000 read the external data, multiply by the L91 scale factor and move the active axis with a feedrate of 1000 pulses/second.	

G37 Read Line Number and Branch

Function	Read data from the selected external data source and branch to the indicated line number.	
Command Format	G37 {Fnnnnnn}	
Range	External Data 1 - 400	line number
Notes	This command can not be used on program line 0 (MDI mode)	
	If the external data is outside the valid range, program execution will terminate.	
	The F field, if programmed, will become the new feedrate.	
Related commands	G52 (Select External Data Source)	

G38 Read Feedrate

Function Read the selected external data source and use as the feedrate.

Command Format
G38 {Xsnnnnnnnn}
G38 {Ysnnnnnnnn}

Range X or Y field -99,999,999 to +99,999,999 index distance External Data 0 - 999,999 feedrate

Notes
If the external data are outside the valid range, program execution will terminate.

The X or Y field value is the index distance and direction.

The BCD data will be scaled using the L91 scale factor.

Related commands G52 (Select External Data Source)

Example

N10 G38 X-1000	Move -1000 on the X axis using the feedrate read from the external data source.

G47 Set Outputs

Function

Set Programmable or Expansion outputs to the states specified by the X or Y field data.

Command Format

G47 Xnnnnnnnn {Fnnnnnn}	Set Programmable outputs G47
Ynnnnnnnn {Fnnnnnn}	L20 nn1nn (set Expansion Outputs)
Ynnnn {Fnnnnnn}	L20 nn2nn (set S2000 Expansion Outputs)

Range

X or Y field 0 to 99,999,999

Notes

The state of each output is specified by a digit in the X or Y field data. A 0 value sets the output inactive, a 1 sets the output active and 2-9 cause no change to the output.

The X field specifies the states of the programmable outputs. (Outputs assigned as "0" or "1" in L59)

The Y field specifies the states of the Expansion Port outputs. If the Expansion Port is not configured for I/O (L20 = nn0nn), then the programmable outputs will be set instead.

If the X or Y field is missing, program execution will terminate.

Each digit in the X or Y field represents an output as shown below:

	L20 nn1n		L20 nn2nn
Xnnnnnnnn		Ynnnnnnnn	Ynnnn
n------	Output 8	n------ Expansion Output 8	n--- S2000 Expansion Output 4
-n-----	Output 7	-n----- Expansion Output 7	-n-- S2000 Expansion Output 3
--n----	Output 6	--n---- Expansion Output 6	--n- S2000 Expansion Output 2
---n---	Output 5	---n--- Expansion Output 5	---n S2000 Expansion Output 1
----n--	Output 4	----n-- Expansion Output 4	
-----n-	Output 3	-----n- Expansion Output 3	
------n	Output 2	------n Expansion Output 2	
------n	Output 1	------n Expansion Output 1	

The F field, if programmed, will become the new feedrate.

The L20 parameter determines which set of expansion I/O to test. L20 nn1nn selects the Expansion outputs, while L20 nn2nn selects the S2000 Expansion board outputs.

Related commands

L20 (Expansion Port Configuration)
L59 (Output Configuration)
G53 (Select S2000 Interface Input Port)

Examples

Expansion Port configured as I/O (L20 = nn1nn) and Outputs 1 - 4 configured as programmable outputs

N1 G47 X00001111 (Outputs 1 - 4 all active)

N2 G47 Y10101010 (Expansion Outputs 1,3,5,7 inactive, Expansion Outputs 2,4,6,8 active)

Expansion Port configured as BCD data (L20 = nn0nn) and Outputs 5 - 8 configured as programmable outputs

N3 G47 Y10101010 (Outputs 5 & 7 inactive, Outputs 6 & 8 active)

Expansion port configured for S2000 Expansion I/O Board (L20 nn2nn)

N1 G53 X1	select Expansion Port 1
N2 G47 Y1000	Port 1 outputs 4 and 3 active, 2 and 1 inactive

G48 Start Jog Cycle

Function

Starts a Jog cycle during program execution in the direction specified by the X or Y field. The feedrate will be the Jog Speed (L09) or the value specified in the F field.

Command Format

G48 Xs {Fnnnnnn}	start a program Jog using the X axis
G48 Ys {Fnnnnnn}	start a program Jog using the Y axis

Range

X or Y field +,- direction
F field 0 - 999,999 feedrate (pulses/sec)

Notes

This command can only be used in Normal execution mode.

The following commands are allowed during a G48 jog cycle:

G04	Delay
G05	Transmit Message
G11	Call Subroutine
G12	Go To Line Number
G19	Branch On Result
G20	Branch on Input Condition
G21	Register Control
G22	Wait For Input Condition
G23	Increase Feedrate
G24	Decrease Feedrate
G25	Loop Start
G26	Loop End
G27	High Speed Mode
G28	Low Speed Mode
G30	End of Program
G31	Program Stop
G32	Return From Subroutine
G37	Read Line Number and Branch
G38	Read Feedrate
G47	Set Outputs
G49	Stop Jog Cycle
G50	Set Flags
G51	Branch on Flag Condition
G52	Select External Data Source
G53	Select S2000 Interface I/O Port
G54	Register Execution
G55	Error Trapping
G62	Wait for Move Distance
G63	Wait for Velocity Condition
G64	Enable Reduce Current
G65	Disable Reduce Current
G66	Enable Boost Current
G67	Disable Boost Current
G68	Motor Windings Off
G69	Motor Windings On
G90	Absolute Position Mode
G91	Incremental Position Mode
Fnnnnnn	Feedrate

The F field, if programmed, will become the new feedrate.

The Single Line execution mode (L06 = nn1), is suspended when a G48 command is executed. The program will execute continuously and Single Line execution mode is restored following a Stop Jog Cycle (G49), End of Program (G30), Program Stop (G31) or executing line 400.

Related Command G49 (Stop Jog Cycle)

Example

N1 G48 Y- F2000	Start CCW Y Axis JOG at 2000 pulses/sec
N2 G04 X1000 F3000	Delay 1 second, change feedrate to 3000 pulses/sec
N4 G04 X1000	Delay 1 second
N5 G30	Program End

G49 Stop Jog Cycle

Function Causes a Jog cycle to come to a controlled stop.

Command Format G49

Notes This command can only be used during a G48 jog cycle.

When executed this command brings a programmed (G48) Jog cycle to a controlled stop and pauses program execution until motion has stopped.

The X or Y field, if programmed, will terminate program execution.

Related commands G48 (Start Jog Cycle)

G50 Set Flags

Function

Set/Reset 8 internal flags based on X or Y field data.

Command Format

G59 Xnnnnnnnn {Fnnnnnn}
G59 Ynnnnnnnn {Fnnnnnn}

Range

X or Y field	0 - 99,999,999
F field	0 - 999,999 feedrate (pulses/sec)

Notes

This command can not be used on program line 0 (MDI mode)

Each digit in the X or Y field corresponds to one of 8 flags. A value of 0 clears the flag, a 1 sets it and 2 - 9 leave it unchanged.

If the X or Y field is missing, program execution will terminate.

The X or Y field output assignments are as follows:

Xnnnnnnnn
n------ flag 8
-n----- flag 7
--n---- flag 6
---n--- flag 5
----n-- flag 4
-----n- flag 3
------n flag 2
-------n flag 1

The F field, if programmed, will become the new feedrate.

Example

The following subroutine changes the state of output #1 each time it is called. The Flag is used to keep track of the output state. The flag is tested (G51) and the output is changed accordingly.

```
N100 G51 X22222220 F104     If flag 1 = 0 go to line 104
N101 G47 X22222220          Output 1 = 0
N102 G50 X22222220          flag 1 = 0
N103 G32                    else
N104 G47 X22222221          Output 1 = 1
N105 G50 X22222221          flag 1 = 1
N106 G32
```

G51 Branch on Flag Condition

Function

If the Flags match the condition specified by the X or Y field, the program branches to the line specified by the F field, otherwise the next line is executed.

Command Format

G51 Xnnnnnnnn Fnnn
G51 Ynnnnnnnn Fnnn

Units

Flag condition	X or Y field)
Line number	(F field)

Range

X or Y field	0 - 99,999,999	flag condition
F field	1 - 400	line number

Notes

This command can not be used on program line 0 (MDI mode)

If no value is entered for the X or Y field, program execution will terminate.

The flag condition "n" for each flag is a value 0 - 9. A 0 requires that the flag be cleared, a 1 requires that the flag be set and a value 2 - 9 indicates the state of the flag doesn't matter.

The X or Y field input assignments are as follows:

Xnnnnnnnn	Ynnnnnnnn
n------ flag 8	n------ 1 = CW 0 = CCW
-n----- flag 7	-n----- Software Limit
--n---- flag 6	--n---- Hardware Limit
---n--- flag 5	---n--- Mark Registration Limit
----n-- flag 4	----n-- Trigger 2 flag
-----n- flag 3	-----n- Trigger 1 flag
------n flag 2	------n velocity at speed flag
-------n flag 1	-------n motion busy flag

The MSB in the Y field selects the direction for the Software Limit or the Hardware Limit.

Examples

L20 02000	X axis registration cycle TR1 positive edge trigger, Y axis registration cycle TR2 positive edge trigger. Registration cycle starts on 2nd TR2 trigger.
N1 G60	Set Continuous Execution mode
N2 G21 X F30	Load BCD data into register 3
N3 G54 Y+3 F3	Start Y axis registration cycle using register 3, but do not arm trigger
N4 G51 Y22220222 F4	Wait for trigger 2 to be active
N5 G51 Y22221222 F5	Wait for trigger 2 to be inactive
N6 G54 F4	Arm registration cycle trigger
N7 G51 Y22222221 F7	Wait for cycle complete
L58 "+2187600000000"	Input 13 X axis CCW limit, Input 12 X axis CW limit. Jogs between CW and CCW limits five times and stops.
N1 G55 X20	Set error trapping routine starting at line 20
N2 G21 X5 F30	5 > register 3
N3 G21 X1 F40	1 > register 4
N4 G21 X5 F34	Set compare result > 0
N5 G48 X+ F2000	Start X axis Jog in CW direction
N6 G12 X6	
N20 G19 X2212 F23	Branch to line 23 if register 3 = 0
N21 G51 Y12122222 F30	If CW hard limit, go to line 30
N22 G51 Y02122222 F24	If CCW hard limit, go to line 24
N23 G30	Program end
N24 G21 X-4 F33	Register 3 minus register 4 > register 3

G51, Continued

```
N25 G21 X5 F34          Test register 3 for 0
N26 G12 X5              Start X axis Jog cycle in CW direction
N30 G48 X- F2000        Start X axis Jog cycle CCW direction @ 1000 pps
N31 G12 X6

N1 G48 X+ F1000         Start X axis Jog cycle CW @ 1000 pps
N2 G51 Y22222202 F2     Wait until at speed
N3 G04 X1000            Delay, one second
N4 G49
```

G52 Select External Data Source

Function

Selects the external data source from which, data is read.

Command Format

G52 Xn {Fnnnnnn} (L20 xx0xx, L20 xx1xx or L20 xx0xx)
G52 Yn {Fnnnnnn} (L20 xx1xx)
G52 Yn {Fnnnnnn} (L20 xx2xx)

Range

With L20 xx0xx, the range is 0-2
1 selects BCD Switch Bank 1 (Strobes 1-4)
2 selects BCD Switch Bank 2 (Strobes 5-8)

With L20 xx@xx, S2000 Expansion I/O, the range is 0-4
1 selects BCD Switch Bank 1
2 selects BCD Switch Bank 2
3 selects BCD Switch Bank 3
4 selects BCD Switch Bank 4

Default

PLC interface, if I/O is configured for it, otherwise BCD Switch Bank 1

Notes

If the PLC interface is selected and the PLC Mode is not enabled, program execution will be terminated. Refer to the L58 and L59 parameters for PLC interface selection,

If BCD switch bank 1, 2, 3 or 4 is selected and the expansion port is configured for Expansion I/O (L20 nn0nn), program execution will be terminated.

The F field, if programmed, will become the new feedrate.

Related commands

G10 (Mark Registration Cycle)
G21 (Register Control)
G29 (Set Designated L Code)
G36 (Read Move Distance)
G37 (Read Line Number and Branch)
G38 (Read Feedrate)
G79 (Set Absolute Position)

Example

This program requires a distance and feedrate selectable from external BCD switches. Bank 1 will be the move distance and Bank 2 will be the velocity for that move.

```
N1 G52 X2     select Bank 2 BCD switches
N2 G38 X      load feed rate from Bank 2 BCD switches
N3 G52 X1     select Bank 1 BCD switches
N4 G36 X      load distance from Bank 1 BCD switches and move
```

G53 Select S2000 Interface I/O Port

Function Selects the I/O port to be accessed via the S2000 Expansion I/O when an I/O function is performed during program execution.

Command Format
G53 Xn {Fnnnnnn}
G53 Yn {Fnnnnnn}

Range 1 through 6 (port 1 through port 6)

Default Port 1

Notes
If a number that is out of range is encountered, program execution will be terminated.

This command is active when the S2000 Expansion I/O is enabled (L20 nn2nn)

The F field value, if programmed, will become the High Speed Rate value.

Related Commands
G20 (Branch On Input Condition)
G22 (Wait For Input Condition)
G47 (Set Outputs)

G54 Register Execution

Function This command can be used to start an Index, Mark Registration, Wait For Move Distance or Return To Mechanical Home, or can be used to abort the current motion cycle.

Command Format

	G54 X+n F0	X axis index using Register Xn distance and direction
	G54 X-n F0	X axis index using Register Xn distance and opposite direction
	G54 Y+n F0	Y axis index using Register Yn distance and direction
	G54 Y-n F0	Y axis index using Register Yn distance and opposite direction
	G54 Xsn F1	Wait Register Xn move distance
	G54 X+n F2	X axis Mark Registration cycle using Register Xn distance and direction
	G54 X-n F2	X axis Mark Registration cycle using Register Xn distance and opposite direction
	G54 Y+n F2	Y axis Mark Registration cycle using Register Yn distance and direction
	G54 Y-n F2	Y axis Mark Registration cycle using Register Yn distance and opposite direction
(Trigger is not armed)	G54 X+n F3	X axis Mark Registration cycle using Register Xn distance and direction
(Trigger is not armed)	G54 X-n F3	X axis Mark Registration cycle using Register Xn distance and opposite direction
(Trigger is not armed)	G54 Y+n F3	Y axis Mark Registration cycle using Register Yn distance and direction
(Trigger is not armed)	G54 Y-n F3	Y axis Mark Registration cycle using Register Yn distance and opposite direction
(Arm Trigger)	G54 F4	Arm Trigger for Mark Registration cycle started by function 3 above
	G54 F5	Abort current motion cycle, if any (this will be a controller stop)
	G54 X F6	Modified Mechanical Home cycle for X axis
	G54 Y F6	Modified Mechanical Home cycle for Y axis

Range

X or Y field	1-9
F field	0-6

Notes

If the F field is out of range, program execution will be terminated.

The F field selects the function to be executed. The functions are:

0	Index cycle for X or Y axis
1	Wait for Move Distance cycle (Reference G62)
2	Mark Registration cycle for X or Y axis (Reference G10)
3	Mark Registration cycle for X or Y axis (Reference G10), do not arm Trigger
4	Mark Registration cycle for X or Y axis (Reference G10), arm Trigger only
5	Abort current motion cycle (controlled stop)
6	Modified Mechanical Home cycle

For functions 0-3, the X field selects X as the motion axis and the Y selects Y as the motion axis. If neither is programmed or the value is out of range, program execution will be terminated.

The Continuous Execution mode must be enabled when functions 3-5 are programmed. If not, enable program execution will be terminated.

If a Mark Registration cycle with no trigger armed (G54 Xsn F3) is taking place and another motion command id encountered, program execution will be terminated. The Control is waiting for an Arm The Trigger command.

The Abort Current Motion Cycle command is only effective if the Continuous Execution mode is selected (G60). This command will not stop a Home cycle.

The X or The Y field selects the register to be used for the selected function.

X+n or Y+n	Register contents
X-n or Y-n	Negate register contents

When the Mechanical Home cycle is started (G54 X F6), the motor moves in the Home direction (L08) at the Home Speed (L14). When the Home switch is activated, the absolute position is captured and the motor brought to a stop. The Absolute Position is now adjusted to reflect the current position in relationship to the Home switch.

Related Commands

G10 (Mark Registration Cycle)
G21 (Register Control)
G62 (Wait For Move Distance)

Examples

```
N1 G60 F1000          Set Continuous Execution mode. Feed Rate is 1000 pps
N2 G21 X10000 F50     10000 > register 5
N3 G21 X400 F30       400 > register 3
N4 G21 X3 F42         Register 3 > Register 4
N5 G54 X+5 F0         Index X axis 10000 pulses at 1000 pps Register 5 direction and
                      value
N6 G25 X23            Loop 24 times
N7 G54 X4 F1          Wait for compare distance using Register 4 value
N8 G47 X22222221      Output 1 active
N9 G21 X+3 F43        Register 3 + Register 4 > Register 4, add 400
N10 G04 X50           Delay 50 milliseconds, time output is active
N11 G47 X22222220     Output 1 inactive
N12 G26               Loop end
N13 G61               Disable Continuous Execution mode

N1 G52 X1             BCD bank 1 selected
N2 G21 F30            BCD data > Register 3
N3 G54 X-3 F2         X axis Mark Registration cycle, Register 3 value, opposite
                      direction
```

N1 G60 F1000	Set Continuous Execution mode. Feed Rate is 1000 pps
N2 G21 X-1000 F30	-1000 > Register 3
N3 G54 Y+3 F3	Y axis Mark Registration cycle, Trigger not armed, Register 3
N4 G51 Y22220222 F4	Wait for TR2 input active
N5 G51 Y22221222 F5	Wait for TR2 input inactive
N6 G54 F4	Arm Mark Registration trigger
N7 G20 X22222221 F10	Branch to line 10 if input 1 active, error occurred
N8 G51 Y22222221 F7	If cycle is still taking place, branch to line 7
N9 G12 X11	Go to line 11
N10 G54 F5	Abort cycle
N11 G61	Disable Continuous Execution mode

G55 Error Trapping

Function

This command detects the line to be executed if a Software Limit, Hardware Limit or Mark Registration Limit is encountered during program execution.

Command Format

G55 Xnnn {Fnnnnnn}
G55 Ynnn {Fnnnnnn}

Range

X or Y field	0-400	line number
F field	0-999,999	feed rate in pps

Notes

If X0 is programmed on the line, the Error Trapping becomes disabled and normal program termination occurs when a limit is encountered.

If the X or Y field is out of range, program execution will be terminated.

If the X field is 1 to 400 and a Software Limit, Hardware Limit or Mark Registration Limit is encountered during program execution, the program execution continues at the line specified by the X field of this command. The loop and subroutine commands are cancelled at this time.

Examples

N1 G55 X10	Error Trapping enabled at line 10
N2 G21 X6 F30	6 > Register 3
N3 G21 X1 F40	1 > Register 4
N4 G48 X+ F1000	Start X axis Jog in CW direction
N5 G12 X5	Wait for soft limit
N6 G21 X-4 F33	Decrement Register 3
N7 G21 X5 F34	Compare Register 3 for zero
N8 G19 X2202 F5	If not zero, branch to line 5
N9 G30	Program end
N10 G51 Y11222222 F13	Branch to line 13 if CCW Software Limit set
N11 G51 Y01222222 F3	Branch to line 3 if CCW Software Limit set
N12 G30	Program end, Hard Limit active
N13 G48 X- F1000	Start X axis Jog CCW direction
N14 G12 X5	Decrement the loop count

G60 Enable Continuous Execution Mode

Function

Allows program lines that don't require motion or a change of active axis to be executed during a programmed index or Jog motion.

Command Format

G60 Xsnnnnnnnn {Fnnnnnn} Continuous Execution mode enabled; move X axis
G60 Ysnnnnnnnn {Fnnnnnn} Continuous Execution mode enabled; move Y axis
G60
Continuous Execution mode enabled

Range

X or Y field -99,999,999 to +99,999,999 index distance
F field 0 - 999,999 feedrate (pulses/sec)

Notes

This command can not be used on program line 0 (MDI mode) or during a G48 Jog Cycle.

The following commands are allowed in Continuous Execution Mode:

G04	Delay
G05	Transmit Message
G10	Mark Registration Cycle
G11	Call Subroutine
G12	Go To Line Number
G19	Branch On Register Result
G20	Branch on Input Condition
G21	Register Control
G22	Wait For Input Condition
G23	Increase Feedrate
G24	Decrease Feedrate
G25	Loop Start
G26	Loop End
G27	High Speed Mode
G28	Low Speed Mode
G30	End of Program
G31	Program Stop
G32	Return From Subroutine
G36	Read Move Distance
G37	Read Line Number and Branch
G38	Read Feedrate
G47	Set Outputs
G50	Set Flags
G51	Branch on Flag Condition
G52	Select External Data Source
G53	Select S2000 Interface I/O Port
G54	Register Execution
G55	Error Trapping
G61	Disable Continuous Execution Mode
G62	Wait for Move Distance
G63	Wait for Velocity Condition
G64	Enable Reduce Current
G65	Disable Reduce Current
G66	Enable Boost Current
G67	Disable Boost Current
G68	Motor Windings Off
G69	Motor Windings On
G76	Return to Electrical Home
G77	Set Electrical Home
G78	Return to Mechanical Home
G79	Set Absolute Position
G90	Absolute Position Mode

G91	Incremental Position Mode
Xsnnnnnnnn	Index X axis
Ynnnnnnnnn	Index Y axis
Fnnnnnn	Feedrate

The F field, if programmed, will become the new feedrate.

The X or Y field value, if programmed, will set the X or Y axis active and will become the index distance and direction.

The Single Line execution mode (L06 = nn1), is suspended when a G60 command is executed. The program will execute continuously and Single Line execution mode restored following a Disable Continuous Execution Mode (G61), End of Program (G30), Program Stop (G31) or executing line 400.

Related Commands G61 (Disable Continuous Execution Mode)

G61 Disable Continuous Execution Mode

Function End Continuous Execution Mode and resume normal program execution.

Command Format

G61 {Xsnnnnnnnn} {Fnnnnnn}	disable Continuous Execution mode; index X axis
G61 {Ynnnnnnnnn} {Fnnnnnn}	disable Continuous Execution mode; index Y axis
G61	disable Continuous Execution mode

Range

X or Y field	-99,999,999 to +99,999,999 index distance
F field	0 - 999,999 feedrate (pulses/sec)

Notes This command can only be used during continuous execution mode.

If an X or Y field is programmed on the line, the index motion will not start until any previous motion is completed.

The F field, if programmed, will become the new feedrate.

Related Commands G60 (Enable Continuous Execution Mode)

Example

N20 G60 X2000 F1000	Start Continuous execution, Move the X axis 2000 pulses at a feedrate of 1000.
N21 G04 X1000	Delay 1000 milliseconds.
N22 G47 X22222221	Turn on Output 1.
N23 G61 X-2000	End Continuous execution, when the motion is complete, move the X axis -2000 pulses at a feedrate of 1000.

G62 Wait for Move Distance

Function Program execution is paused until the distance specified by the X or Y field has been reached.

Command Format
G62 Xnnnnnnnn {Fnnnnnn} wait for distance to be achieved
G62 Ynnnnnnnn {Fnnnnnn} wait for distance to be achieved

Range
X or Y field 0 - 99,999,999 move distance
F field 0 - 999,999 feedrate (pulses/sec)

Notes
This command can not be used on program line 0 (MDI mode) or during normal execution mode.

If the F field is programmed, it will become the new feedrate when the distance has been achieved.

If the absolute mode is active, then the compare distance is an absolute position, otherwise the compare distance is an incremental distance.

Related Commands G60 Enable Continuous Execution Mode

Example:

N1 G60 X10000 F2000	Continuous Execution Mode; Move X 10000 pulses at 2000 pulses/sec.
N2 G62 X1000	Wait for the occurrence of 1000 pulses from start.
N3 G47 X22222221	Output 1 On
N4 G62 X9000	Wait for the occurrence of 9000 pulses from start.
N5 G47 X22222220	Output 1 Off
N6 G61	Cancel Continuous Execution Mode
N7 G30	Program End

G63 Wait for Velocity Condition

Function

Program execution is paused until the specified F field velocity is achieved.

Command Format

G63 X+ Fnnnnnn	wait for velocity to be >= Fnnnnnn
G63 X- Fnnnnnn	wait for velocity to be < Fnnnnnn
G63 Y+ Fnnnnnn	wait for velocity to be >= Fnnnnnn
G63 Y- Fnnnnnn	wait for velocity to be < Fnnnnnn

Range

F field 1 - 999,999	compare velocity

Notes

This command can not be used on program line 0 (MDI mode) or during normal execution mode.

If an X or Y value is programmed, program execution will be terminated.

Related Commands

G62 (Wait for Move Distance)

Example

N1 G48 X+ F1000	Start Program Jog; Target Velocity 1000 pulses/sec.
N2 G63 X+ F1000	Wait for Velocity = 1000 pulses/sec.
N3 G47 X22222221 F0	Turn on Output 1, Feedrate = 0
N4 G63 X- F500	Wait for Velocity < 500 pulses/sec.
N5 G47 X22222220	Turn off Output 1
N6 G49	Stop Jog

G64 Enable Reduce Current

Function

Motor current is reduced by 50% at standstill. Reducing the motor current lowers heating, but reduces holding torque.

Command Format

G64 {Fnnnnnn}	active axis reduce current mode On
G64 X {Fnnnnnn}	X axis reduce current mode On
G64 Xsnnnnnnnn {Fnnnnnn}	X axis reduce current mode On, index X axis
G64 Y {Fnnnnnn}	Y axis reduce current mode On
G64 Ysnnnnnnnn {Fnnnnnn}	Y axis reduce current mode On, index Y axis

Range

X or Y field -99,999,999 to +99,999,999 index distance
F field 0 - 999,999 feedrate (pulses/sec)

Default Mode

Reduce Current Disabled

Notes

When the Reduce Current mode is active, an additional time delay is introduced during motion cycles. This delay allows winding current to build in the motor prior to motion and to decay after motion has stopped. The delay is approximately 10 milliseconds prior to motion and 10 milliseconds after motion has stopped. Therefore, the shortest possible time between moves is approximately 20 milliseconds.

The Reduce Current feature should only be used when maximum holding torque at standstill is not needed.

The X field, if programmed, will define the X axis index distance and direction.

The Y field, if programmed, will define the Y axis index distance and direction.

The F field, if programmed, will become the new feedrate.

The Reduce Current command will be ignored when the Windings Off (G68) mode is active.

Related commands

G65 (Disable Reduce Current)

G65 Disable Reduce Current

Function

Cancels the Reduce Current mode.

Command Format

G65 {Fnnnnnn}	active axis reduce current mode Off
G65 X {Fnnnnnn}	X axis reduce current mode Off
G65 Xsnnnnnnnn {Fnnnnnn}	X axis reduce current mode Off, index X axis.
G65 Y {Fnnnnnn}	Y axis reduce current mode Off
G65 Ysnnnnnnnn {Fnnnnnn}	Y axis reduce current mode Off, index Y axis.

Range

X or Y field -99,999,999 to +99,999,999 index distance
F field 0 - 999,999 feedrate (pulses/sec)

Default Mode

Reduce Current Disabled

Notes

Canceling a Reduce Current condition with the G65 command provides nominal holding torque at standstill and will also permit shorter cycle times.

The X field, if programmed, will define the X axis index distance and direction.

The Y field, if programmed, will define the Y axis index distance and direction.

The F field, if programmed, will become the new feedrate.

Related commands

G64 (Enable Reduce Current)

G66 Enable Boost Current

Function

Motor current is increased during accel and decel by 50%. This increases torque, but motor heating will increase.

Command Format

G66 {Fnnnnnn}	active axis boost current mode On
G66 X {Fnnnnnn}	X axis boost current mode On
G66 Xsnnnnnnnn {Fnnnnnn}	X axis boost current mode On, index X axis
G66 Y {Fnnnnnn}	Y axis boost current mode On
G66 Ysnnnnnnnn {Fnnnnnn}	Y axis boost current mode On, index Y axis

Range

X or Y field -99,999,999 to +99,999,999 index distance
F field 0 - 999,999 feedrate (pulses/sec)

Default Mode

Boost Current Disabled

Notes

When accelerating Boost Current will be applied until the motor has reached the programmed speed.

When decelerating Boost Current will be applied until the motor has stopped.

When the Boost Current feature is active, an additional time delay is introduced during motion cycles to allow time for motor winding current to build prior to motion and to decay after motion has stopped. This delay is approximately 10 milliseconds prior to motion and 10 milliseconds after motion has stopped. Therefore, the shortest possible time between moves is approximately 20 milliseconds.

The X field, if programmed, will define the X axis index distance and direction.

The Y field, if programmed, will define the Y axis index distance and direction.

The F field, if programmed, will become the new feedrate.

The Boost Current feature should be used when additional torque is required for starting or stopping.

Related commands

G67 (Disable Boost Current)

G67 Disable Boost Current

Function Cancels the Boost Current mode.

Command Format

G67 {Fnnnnnn}	active axis boost current mode Off
G67 X {Fnnnnnn}	X axis boost current mode Off
G67 Xsnnnnnnnn {Fnnnnnn}	X axis boost current mode Off, index X axis
G67 Y {Fnnnnnn}	Y axis boost current mode Off
G67 Ysnnnnnnnn {Fnnnnnn}	Y axis boost current mode Off, index Y axis

Range

X or Y field -99,999,999 to +99,999,999 index distance
F field 0 - 999,999 feedrate (pulses/sec)

Default Mode Boost Current Disabled

Notes

The X field, if programmed, will define the X axis index distance and direction.

The Y field, if programmed, will define the Y axis index distance and direction.

The F field, if programmed, will become the new feedrate.

Related commands G66 (Enable Boost Current)

G68 Motor Windings Off

Function

Motor current is reduced to zero at standstill.

Command Format

G68 {Fnnnnnn}	active axis windings off mode
G68 X {Fnnnnnn}	X axis windings off mode
G68 Xsnnnnnnnn {Fnnnnnn}	X axis windings off mode, index X axis
G68 Y {Fnnnnnn}	Y axis windings off mode
G68 Ysnnnnnnnn {Fnnnnnn}	Y axis windings off mode, index Y axis

Range

X or Y field -99,999,999 to +99,999,999 index distance
F field 0 - 999,999 feedrate (pulses/sec)

Default Mode

Motor Windings On

Notes

The G68 command should only be used when no holding torque is required at standstill.

When the Motor Windings Off function is active, an additional time delay is introduced during motion cycles to allow winding current to build in the motor prior to motion and to decay after motion has stopped. This delay is approximately 50 milliseconds prior to motion and 50 milliseconds after motion has stopped. Therefore, the shortest possible time between moves is approximately 100 milliseconds.

The X field, if programmed, will define the X axis index distance and direction.

The Y field, if programmed, will define the Y axis index distance and direction.

The F field, if programmed, will become the new feedrate.

Related commands

G69 (Motor Windings On)

G69 Motor Windings On

Function

Cancels the Motor Windings Off condition, causing current to be applied to the motor windings.

Command Format

G69 {Fnnnnnn}	active axis windings on mode
G69 X {Fnnnnnn}	X axis windings on mode
G69 Xsnnnnnnnn {Fnnnnnn}	X axis windings on mode, index X axis
G69 Y {Fnnnnnn}	Y axis windings on mode
G69 Ysnnnnnnnn {Fnnnnnn}	Y axis winding on mode, index Y axis

Range

X or Y field -99,999,999 to +99,999,999 index distance
F field 0 - 999,999 feedrate (pulses/sec)

Default Mode

Motor Windings On

Notes

When Windings On mode is active, motor winding current is applied at all times, providing holding torque at standstill and allowing less time between moves, by eliminating the 50 millisecond turn on and turn off delay.

The X field, if programmed, will define the X axis index distance and direction.

The Y field, if programmed, will define the Y axis index distance and direction.

The F field, if programmed, will become the new feedrate.

Related commands

G68 (Motor Windings Off)

G76 Return to Electrical Home

Function

Move to absolute position zero.(Electrical Home)

Command Format

G76 {Fnnnnnn}	Move active axis to Electrical Home
G76 X {Fnnnnnn}	Move X axis to Electrical Home
G76 Xsnnnnnnnn {Fnnnnnn}	Move X axis to Electrical Home, then move X axis as specified
G76 Y {Fnnnnnn}	Move Y axis to Electrical Home
G76 Ysnnnnnnnn {Fnnnnnn}	Move Y axis to Electrical Home, then move Y axis as specified

Range

X or Y field -99,999,999 to +99,999,999 index distance
F field 0 - 999,999 feedrate (pulses/sec)

Notes

This command can not be used during a G48 jog cycle.

The move to Electrical Home uses the home speed (L14).

A program line may also specify a move in addition to the home cycle.

If programmed, the X field defines the X axis move to be performed following the return to electrical home.

If programmed, the Y field defines the Y axis move to be performed following the return to electrical home.

The F field, if programmed, will become the new feedrate.

Related commands

L14 (Home Speed)

Examples

N10 G76 X	Moves X axis to Electrical Home
N11 G76 Y-1000 F1000	Moves Y axis to Electrical Home, then moves -1000 steps at 1000 pulses/sec

G77 Set Electrical Home

Function

Sets the absolute and encoder positions to zero.

Command Format

G77 {Fnnnnnn}	active axis position = 0
G77 X {Fnnnnnn}	X axis position = 0
G77 Xsnnnnnnnn {Fnnnnnn}	X axis position = 0, then move X axis as specified
G77 Y {Fnnnnnn}	Y axis position = 0
G77 Ysnnnnnnnn {Fnnnnnn}	Y axis position = 0, then move Y axis as specified

Range

X or Y field -99,999,999 to +99,999,999 index distance
F field 0 - 999,999 feedrate (pulses/sec)

Notes

This command can not be used during a G48 jog cycle.

Encoder feedback is only available on the X axis, When the command references the X axis both X absolute and encoder position are cleared, only Y absolute position is cleared for references to the Y axis.

If programmed, the X field defines the X axis move to be performed after the present position is cleared.

If programmed, the Y field defines the Y axis move to be performed after the present position is cleared.

The F field, if programmed, will become the new feedrate.

Examples

N10 G77 X	Set X absolute and encoder position to zero
N11 G76 Y-1000 F1000	Set Y absolute position to zero, then move -1000 steps at 1000 pulses/sec

G78 Return to Mechanical Home

Function

Move to mechanical home position.

Command Format

G78 {Fnnnnnn}	move active axis to mechanical home
G78 X {Fnnnnnn}	move X axis to mechanical home
G78 Xsnnnnnnnn {Fnnnnnn}	move X axis to mechanical home, then do specified move
G78 Y {Fnnnnnn}	move Y axis to mechanical home
G78 Ysnnnnnnnn {Fnnnnnn}	move Y axis to mechanical home, then do specified move

Range

X or Y field -99,999,999 to +99,999,999 index distance
F field 0 - 999,999 feedrate (pulses/sec)

Notes

This command can not be used during a G48 jog cycle.

Parameter L20 selects the home switch input (TR1 or TR2) and it's activation state (rising edge, falling edge or input active and encoder index). The latter activation option requires an encoder and is limited to use with the X axis.

When a mechanical home cycle is started, the motor moves in the home direction (L08) at the home speed (L14).

When the home switch is activated, the absolute position is captured and the motor brought to a stop.

The home offset (L17) is added to the captured position and the move to that position is started after the L43 delay has expired.

When this move is complete, the absolute position is set to zero (Homing the X axis additionally sets the encoder position to zero) and the cycle is done.

A program line may also specify a move in addition to the home cycle.

If programmed, the X field defines the X axis move to be performed following the return to mechanical home.

If programmed, the Y field defines the Y axis move to be performed following the return to mechanical home.

The F field, if programmed, will become the new feedrate.

Related commands

L08 (Mechanical Home Direction)
L14 (Home Speed)
L17 (Offset from Mechanical Home)
L20 (TR1 & TR2 Configuration)

Examples

N10 G78 X	Moves X axis to Mechanical Home
N11 G78 Y-1000 F1000	Moves Y axis to Mechanical Home, then moves -1000 steps at 1000 pulses/sec

G79 Set Absolute Position

Function

Sets the Absolute Position and Encoder Position to the value specified in the X or Y field.

Command Format

G79 {Fnnnnnn}	set active axis position using external BCD data
G79 X {Fnnnnnn}	set X axis position using external BCD data
G79 Xsnnnnnnnn {Fnnnnnn}	set X axis position using X field data
G79 Y {Fnnnnnn}	set Y axis position using external BCD data
G79 Ysnnnnnnnn {Fnnnnnn}	set Y axis position using Y field data

Range

X or Y field -99,999,999 to +99,999,999 index distance
F field 0 - 999,999 feedrate (pulses/sec)

Notes

This command can not be used during a G48 jog cycle.

Encoder feedback is only available on the X axis. When the command references the X axis both X absolute and encoder position are set, only Y absolute position is set for references to the Y axis.

If X or Y field data is not specified, external BCD data is read, multiplied by the External move distance scale factor (L91) and used to set the position.

The F field, if programmed, will become the new feedrate.

Related commands

G52 (Select external data source)

Example

N1 G79 X0 X absolute position = 0
(Alternative to using G77)

G90 Absolute Position Mode

Function

This command selects the Absolute Position mode.

Command Format

G90 {Fnnnnnn}	set active axis position mode to Absolute
G90 X {Fnnnnnn}	set X axis to absolute position mode
G90 Xsnnnnnnnn {Fnnnnnn}	set X axis to absolute position mode and move to the specified absolute position
G90 Y {Fnnnnnn}	set Y axis to absolute position mode
G90 Ysnnnnnnnn {Fnnnnnn}	set Y axis to absolute position mode and move to the specified absolute position

Range

X or Y field -99,999,999 to +99,999,999 move distance
F field 0 - 999,999 feedrate (pulses/sec)

Default Mode

Incremental Mode

Notes

Once set, the absolute position mode will be active until changed by a G91 command or until the unit is powered off. All move distances are absolute distances.

The X field, if programmed, will define the X axis index distance and direction.

The Y field, if programmed, will define the Y axis index distance and direction.

The F field, if programmed, will become the new feedrate.

Related Commands

G91 (Incremental Position Mode)

Examples

N1 G90 X0		Move X axis to absolute position 0 (Alternative to using G78)
N2	X1000 F2000	Move X axis to absolute position 1000 with a feedrate of 2000 pulses/sec
N3	X1000 F2000	No motion occurs, already at absolute position 1000

G91 Incremental Position Mode

Function

This command selects the Incremental Position mode.

Command Format

G91 {Fnnnnnn}	set active axis position mode to Incremental
G91 X {Fnnnnnn}	set X axis to Incremental position mode
G91 Xsnnnnnnnn {Fnnnnnn}	set X axis to Incremental position mode and move the specified incremental distance
G91 Y {Fnnnnnn}	set Y axis to Incremental position mode
G91 Ysnnnnnnnn {Fnnnnnn}	set Y axis to Incremental position mode and move the specified incremental distance

Range

X or Y field -99,999,999 to +99,999,999 index distance
F field 0 - 999,999 feedrate (pulses/sec)

Default Mode

Incremental Mode

Notes

The incremental position mode, active at power up, remains active until changed by a G90 command. All moves specify incremental distances.

The X field, if programmed, will define the X axis index distance and direction.

The Y field, if programmed, will define the Y axis index distance and direction.

The F field, if programmed, will become the new feedrate.

Examples

N1 G91 X	Set X axis to incremental position mode
N2 X1000 F2000	Move X axis +1000 pulses with a feedrate of 2000 pulses/sec
N3 X1000 F2000	Move X axis +1000 pulses with a feedrate of 2000 pulses/sec

H CODES

RS232 COMMAND (H-CODE) FUNCTIONAL LISTING

MOTION COMMANDS
H01 Cycle start
H06 CW direction
H07 CCW direction
H31 Increase speed
H32 Decrease speed

HOMING
H08 Return to electrical home
H09 Set electrical home
H10 Return to mechanical home

PROGRAM
H11 Clear current program line
H12 Clear program
H13 TX current program line
H14 TX program
H15 TX current line number
H16 TX parameters

STATUS QUERIES
H17 TX absolute position
H19 TX mode status
H20 TX output status
H21 TX input status
H23 TX software revision level
H29 TX program execution time
H41 TX program repeat count
H42 TX last external BCD value
H43 TX opto22 I/O status
H44 TX Registers Contents
H60 TX present speed
H88 TX execution status
H99 TX unit type

CLOSED LOOP
H22 TX encoder position
H26 TX position error
H27 TX verification status

MODE CONTROL COMMANDS
H02 Step mode
H03 Jog mode
H04 High speed mode
H05 Low speed mode
H33 Set incremental position mode
H34 Set absolute position mode

MOTOR CURRENT CONTROL
H35 Motor windings on
H36 Motor windings off
H37 Enable boost current
H38 Disable boost current
H39 Enable reduce current
H40 Disable reduce current

PROGRAM DEBUG COMMANDS
H24 Enable trace mode
H25 Disable trace mode
H85 TX system error status
H86 TX data error status
H87 TX program error line number

NOTE: A number of the H codes are also available as immediate codes by preceding them with an exclamation point (!).

H01 Cycle Start

Function This command starts program execution.

Default Mode none

Command Format H01(crlf)

Notes The Cycle Start command causes program execution to begin at the currently selected line number. Program execution is performed in a sequential fashion.

Related commands
L06 (Line Execution Format)
L41 (Auto Start Line Number)
N (Line Number)

H02 Step Mode

Function This command selects the Step mode for the active axis.

Default Mode Jog mode (on power turn-on)

Command Format H02(crlf)

Notes The H02 command does not cause motion to occur.

When the Control is in the Step mode and an H06 (CW) or an H07 (CCW) command is given, the Control will step the motor in the desired direction. The step distance for this move is the value entered for the L13 (Step Increment) parameter.

H02 will not be processed during a Feed Hold condition.

Related commands
L09 (Jog Speed)
L13 (Step Increment)
H04 (High Speed Mode)
H05 (Low Speed Mode)
H06 (CW Direction)
H07 (CCW Direction)
H19 (Transfer Mode Status)
X, (Select X axis)
Y, (Select Y axis)

H03 Jog Mode

Function The H03 command selects the Jog mode for the active axis manual motion operation.

Default Mode Jog Mode (on power turn-on)

Command Format H03(crlf)

Notes The H03 command does not cause motion to occur.

When the Control is in the Jog mode and an H06 (CW) or an H07 (CCW) command is given, the motor will Jog in the desired direction. The motor will jog until a Feed Hold or a Clear command is issued or until a limit is exceeded.

H03 will not be processed during a Feed Hold condition.

Related commands
L09 (Jog Speed)
H04 (High Speed Mode)
H05 (Low Speed Mode)
H06 (CW Direction)
H07 (CCW Direction)
H19 (Transfer Mode Status)
X, (Select X axis)
Y, (Select Y axis)

H04 High Speed Mode

Function The H04 command selects the High Speed mode for the active axis.

Default Mode High Speed mode (on power turn-on)

Command Format
H04(crlf)
!H04(crlf)

Notes The H04 command selects the L09 value as the High Speed rate for the Jog cycle or Step execution and the F value as the High Speed rate for Program Execution.

If !H04 is issued during Low Speed motion, the appropriate High Speed value will become the target velocity for the motion in progress.

Related commands
L09 (Jog Speed)
H05 (Low Speed Mode)
H19 (Transfer Mode Status)
X, (Select X axis)
Y, (Select Y axis)

H05 Low Speed Mode

Function The H05 command selects the Low Speed mode for the active axis.

Default Mode High Speed mode (on power turn-on)

Command Format
H05(crlf)
!H05(crlf)

Notes This command selects the L12 (Low Speed) parameter value as the velocity for a Jog cycle, Program Execution or Step Execution.

If an !H05 command is issued during High Speed motion, the L12 (Low Speed) value becomes the target velocity for the motion in progress. The motor will be ramped down to the L12 (Low Speed) parameter value.

Related commands
H04 (High Speed Mode)
H19 (Transfer Mode Status)
L12 (Low Speed)
X, (Select X axis)
Y, (Select Y axis)

H06 CW Direction

Function This command is used to start a Jog or Step motion in the CW direction for the active axis.

Default Mode none

Command Format H06(crlf)

Notes:If any motion other than CW manual motion is active or if a Program Cycle is active, The CW command will not be honored.

The following chart lists the action which will occur with the possible combinations of inputs that affect CW Direction.

INPUTS THAT AFFECT CW DIRECTION			
CW MOTION	**STEP/JOG**	**SPEED**	**FUNCTION**
active	Jog (H03)	High (H04)	Ramp to Jog Speed rate and Jog
active	Jog (H03)	Low (H05)	Jog at Low Speed rate
active	Step (H02)	High (H04)	Ramp to Jog Speed rate and Step
active	Step (H02)	Low (H05)	Step at Low Speed rate
active	Jog or Step	Low to High	Ramp to Jog Speed rate
active	Jog or Step	High to Low	Ramp to Low Speed rate

Related commands
L09 (Jog Speed)
L12 (Low Speed)
L13 (Step Increment)
H02 (Step Mode)
H03 (Jog Mode)
H04 (High Speed Mode)
H05 (Low Speed Mode)
X, (Select X axis)
Y, (Select Y axis)

H07 CCW Direction

Function This command is used to start a Jog or Step motion in the CCW direction the active axis.

Default Mode none

Command Format H07(crlf)

Notes **If any motion other than CCW manual motion is active or if a Program Cycle is active, the CCW command will not be honored.**

The following chart lists the action which will occur with the possible combinations of inputs that affect CCW Direction.

INPUTS THAT AFFECT CCW DIRECTION			
CCW MOTION	**STEP/JOG**	**SPEED**	**FUNCTION**
active	**Jog (H03)**	**High (H04)**	**Ramp to Jog Speed rate and Jog**
active	**Jog (H03)**	**Low (H05)**	**Jog at Low Speed rate**
active	**Step (H02)**	**High (H04)**	**Ramp to Jog Speed rate and Step**
active	**Step (H02)**	**Low (H05)**	**Step at Low Speed rate**
active	**Jog or Step**	**Low to High**	**Ramp to Jog Speed rate**
active	**Jog or Step**	**High to Low**	**Ramp to Low Speed rate**

Related commands
L09 (Jog Speed)
L12 (Low Speed)
L13 (Step Increment)
H02 (Step Mode)
H03 (Jog Mode)
H04 (High Speed Mode)
H05 (Low Speed Mode)
X, (Select X axis)
Y, (Select Y axis)

H08 Return To Electrical Home

Function The H08 command will cause the Control to perform an index move to return to the location which has been designated Electrical Home. This is an Absolute position of zero.

Default Mode none

Command Format H08(crlf)

Notes The direction for a Return To Electrical Home index move will be the reverse of the direction of the present Absolute position and the distance will be that of the present Absolute electrical position.

Related commands
H04 (High Speed Mode)
H05 (Low Speed Mode)
H09 (Set Electrical Home)
L14 (Home Speed)
X, (Select X axis)
Y, (Select Y axis)

H09 Set Electrical Home

Function The H09 commands the Control to establish the present mechanical position as Electrical Home and Encoder Home Position.

Default Mode none

Command Format H09(crlf)

Notes All Absolute positions are referenced to the Electrical Home position.

This home position will continue to be the Electrical Home position until a new Set Electrical Home command is given, a Return to Mechanical Home cycle is executed or the Control is deenergized.

H09 will not be processed while a Feed Hold condition is active.

Related Commands
H08 (Return to Electrical Home)
H10 (Return to Mechanical Home)
X, (Select X axis)
Y, (Select Y axis)

H10 Return To Mechanical Home

Function This code commands the Control to perform a Return To Mechanical Home cycle.

Default Mode none

Command Format H10(crlf)

Notes The Control begins a Return To Mechanical Home cycle by jogging at the rate programmed for the Home Speed parameter (L14). The direction of the motion is determined by the Mechanical Home Direction parameter (L08).

The Control will Jog until the Home Limit switch is activated, and will then ramp to a halt. The point at which the Home Limit switch was activated will be registered as the Home position.

When motion ceases, the Control will index to the Home position and will establish this position as the Absolute Electrical Home Position as well as the Encoder Home Position.

Related commands
L08 (Mechanical Home Direction)
L12 (Low Speed)
L14 (Home Speed)
H04 (High Speed Mode)
H05 (Low Speed Mode)
X, (Select X axis)
Y, (Select Y axis)

H11 Clear Current Program Line

Function The H11 commands the Control to clear the current program line. The line will be irretrievably erased and the line pointer will not change.

Default Mode none

Command Format H11(crlf)

Notes This command when used with the Nnnn command can be used to clear specific program lines.

Examples H11(crlf) will cause the current line to be cleared.
N004 H11(crlf) will cause line 4 to be cleared.

H12 Clear Program

Function This command, together with L48 = 0, will clear the entire program. The lines will be irretrievably erased and the line pointer will then be set to the line selected by L41.

Default Mode none

Command Format H12(crlf)

Notes If L48 = nnn and the H12 command is issued, nnn lines will be cleared starting from the current line. The line pointer will then be set to one line past the last cleared line.

Examples L48 0 H12(crlf) will clear lines 1 through 400. The line pointer will be set to the L41 value.

N1 L48 10 H12(crlf) will clear lines 1 through 10. The line pointer will be set to 11.

H13 Transfer Current Program Line Number and Contents

Function Requests the transfer of the current program line number and contents.

Default Mode none

Command Format H13(crlf)
!H13(crlf)

Notes This command when used with the Nnnn command can be used to transfer specific program lines. If the line is not programmed only the line number will be transferred.

Depending on what fields are programmed. Spaces will be transferred to represent unprogrammed fields.

Examples H13(crlf) will cause the current line to be transferred.
N004 H13(crlf) will cause line 4 to be transferred.

H14 Transfer Program

Function This command, together with L48 = 0, will transfer all programmed lines. The line pointer will then be set to the line selected by L41.

Default Mode none

Command Format H14(crlf)

Notes If L48 = nnn and this command is given, nnn programmed lines will be transferred, starting from the current line. If the line is not programmed the line will not be transferred.

If no lines are programmed a "crlf" will be transferred only and the line pointer will be set to the L41 value.

Examples L48 0 H14(crlf) will transfer all programmed lines starting with the first line programmed.

N1 L48 10 H14(crlf) will transfer the first 10 programmed lines. The line pointer will be set to the line after the last line transferred.

N40 L48 20 H14(crlf) will transfer 20 programmed lines starting at line 40. The line pointer will be set to the line after the last line transferred.

H15 Transfer Current Line Number

Function This command requests the transfer of the current line number.

Default Mode none

Command Format H15(crlf)
!H15(crlf)

Notes The data will be transferred in the following format:
Nnnn(crlf)

H16 Transfer Parameters

Function

This command, together with L49 and L50 will cause the transfer of parameters (L codes). The format of the transfer and the number parameters transferred are controlled with the L49 and L50 values.

Default Mode

none

Command Format

H16(crlf)

Notes

L49 = 0
L50 > 0
The L codes are transferred one per line with all L codes being transferred.

L49 = 0
L50 = 0
The L codes are transferred three per line with all L codes being transferred.

L49 = nn
L50 = 0
Only the Lnn code is transferred.

L49 = nn
L50 = xx (where xx is a number 1 thru 99)
Only xx number of L codes will be transferred starting with the Lnn code. The L codes will be transferred one per line.

Examples

command: L50 0 L49 09 H16(crlf)
response: L09 nnnnnn,nnnnnn(crlf)

command: L50 3 L49 11 H16(crlf)
response: L11 nnnnnnnn,nnnnnnnn(crlf)
L12 nnnnnn,nnnnnn(crlf)
L13 nnnnnnnn,nnnnnn(crlf)

H17 Transfer Absolute Position

Function

This code requests the transfer of the active axes Absolute Electrical Position.

Default Mode

none

Units

pulses

Command Format

H17(crlf)
!H17(crlf)

Notes

The data will be transferred in the following format:
Xsnnnnnnnnn(crlf)
Ysnnnnnnnnn(crlf)

The absolute position limits are +/- 2,147,483,647 pulses.

Related Commands

X, (Select X axis)
Y, (Select Y axis)

H19 Transfer Mode Status

Function The H19 code requests the transfer of the status of the Control modes.

Default Mode none

Command Format
H19(crlf)
!H19(crlf)

Notes The data will be transferred in the following format:
nnnnnnnn nnnnnnnn(crlf)

n------- --------	0 = X Incremental Mode	1 = X Absolute Mode
-n------ --------	0 = X Motor Windings Off	1 = X Motor Windings On
--n----- --------	0 = X Boost Current Off	1 = X Boost Current On
---n---- --------	0 = X Reduce Current Off	1 = X Reduce Current On
----n--- --------	0 = X Low Speed Mode	1 = X High Speed Mode
-----n-- --------	0 = X Step Mode	1 = X Jog Mode
------n- --------	0 = X Selected	1 = Y Selected
-------n --------	0 = CCW direction selected	1 = CW direction selected
-------- n-------	0 = Y Incremental Mode	1 = Y Absolute Mode
-------- -n------	0 = Y Motor Windings Off	1 = Y Motor Windings On
-------- --n-----	0 = Y Boost Current Off	1 = Y Boost Current On
-------- ---n----	0 = Y Reduce Current Off	1 = Y Reduce Current On
-------- ----n---	0 = Y Low Speed Mode	1 = Y High Speed Mode
-------- -----n--	0 = Y Step Mode	1 = Y Jog Mode
-------- ------n-	0 = X Selected	1 = Y Selected
-------- -------n	0 = CCW direction selected	1 = CW direction selected

Example !H19(crlf) sent. The following data transfer may result:

00000010 00000010(crlf)

This data transfer indicates that the Y axis is selected.

H20 Transfer Output Status

Function Transfers the status of the Control programmable outputs (L59 assignments).

Default Mode none

Command Format
H20(crlf)
!H20(crlf)

Notes The data will be transferred as follows:
nnnnnnnn(crlf)

n------- Output 8	0= inactive	1= active
-n------ Output 7	0= inactive	1= active
--n----- Output 6	0= inactive	1= active
---n---- Output 5	0= inactive	1= active
----n--- Output 4	0= inactive	1= active
-----n-- Output 3	0= inactive	1= active
------n- Output 2	0= inactive	1= active
-------n Output 1	0= inactive	1= active

The L58 parameter sign selects the polarity of the output state.
L58 +nnnnnnnnnnnnn
A logic state of 1 indicates that the output is in the active state.
L58 -nnnnnnnnnnnnn
A logic state of 1 indicates that the output is in the inactive state.

Example H20(crlf) sent. The following data transfer may result:

00000010(crlf)

This data transfer indicates that Output 2 is active. All others are inactive.

H21 Transfer Input Status

Function Transfers the status of all the Control inputs.

Default Mode none

Command Format
H21(crlf)
!H21(crlf)

Notes The data will be transferred as follows:

```
nnnnnnnn nnnnnnnn(crlf)
n------- -------- Clear Input        0 = Inactive  1 = active
-n------ -------- Trigger 2          0 = Inactive  1 = active
--n----- -------- Trigger 1          0 = Inactive  1 = active
---n---- -------- Input 13           0 = Inactive  1 = active
----n--- -------- Input 12           0 = Inactive  1 = active
-----n-- -------- Input 11           0 = Inactive  1 = active
------n- -------- Input 10           0 = Inactive  1 = active
-------n -------- Input 9            0 = Inactive  1 = active

-------- n------- Input 8            0 = Inactive  1 = active
-------- -n------ Input 7            0 = Inactive  1 = active
-------- --n----- Input 6            0 = Inactive  1 = active
-------- ---n---- Input 5            0 = Inactive  1 = active
-------- ----n--- Input 4            0 = Inactive  1 = active
-------- -----n-- Input 3            0 = Inactive  1 = active
-------- ------n- Input 2            0 = Inactive  1 = active
-------- -------n Input 1            0 = Inactive  1 = active
```

The L58 parameter sign selects the polarity of the input states for Input 1 thru Input 13, Clear, TR1 and TR2.

L58 +nnnnnnnnnnnnn
A logic state of 1 indicates that the input is in the active state.

L58 -nnnnnnnnnnnnn
A logic state of 1 indicates that the input is in the inactive state.

Example H21(crlf) sent. The following data transfer may result:

00000000 10000010(crlf)

This data transfer indicates that Input 8 and Input 2 are active. All others are inactive.

H22 Transfer Encoder Position

Closed Loop Control Only

Function This command requests the transfer of the Encoder Position.

Default Mode none

Units pulses

Command Format
H22(crlf)
!H22(crlf)

Notes The transfer format is:

Xsnnnnnnnnnn(crlf)

H23 Transfer Software Revision Level

Function This command request the transfer of the Software Revision Level.

Default Mode none

Command Format H23(crlf)

Notes The data will be transferred in the following format:
SloSyn mm/yy/x(crlf)

where:
mm = Month
yy = Year
x = Revision Level

H24 Enable Trace Mode

Function This command causes the Control to transmit the contents of each line as it is executed during program execution.

Default Mode Off

Command Format H24(crlf)
!H24(crlf)

Notes If the line is not programmed the line will not be transferred.

If a field is not programmed, the field will not be transferred.

Depending on what fields are programmed. Spaces will be transferred to represent unprogrammed fields.

H25 Disable Trace Mode

Function This command will cancel the Program Trace Mode.

Default Mode Off

Command Format H25(crlf)
!H25(crlf)

H26 Transfer Position Error

Closed Loop Control Only

Function The H26 code requests the transfer of the Position Verification Error Pulse Count. This is the difference between Absolute Electrical Position and Encoder Position.

Default Mode none

Units pulses

Command Format H26(crlf)
!H26(crlf)

Notes The transfer format is:
Xsnnnnnnnnnn(crlf)

H27 Transfer Verification Status

Closed Loop Control Only

Function The H27 code requests the transfer of the Position Verification status.

Default Mode none

Command Format H27(crlf)
!H27(crlf)

Notes The transfer format is:
Xnnnnn000(crlf)

```
n------- Position Verification          0=inactive   1=active
-n------ Following Error Detect         0=inactive   1=active
--n----- Correction Cycle In Progress   0=inactive   1=active
---n---- Unable To Correct Position     0=inactive   1=active
----n--- Position Maintenance Cycle     0=inactive   1=active
-----0-- Always zero
------0- Always zero
-------0 Always zero
```

H29 Transfer Program Execution Time

Function This command transfers the program execution time, in milliseconds.

Units milliseconds

Default Mode none

Command Format H29(crlf)

Notes A Cycle Start command restarts the timing from zero (except when restarting from a Feed Hold condition). Timing is stopped by a Feed Hold, Clear, Program Line Error, Program Stop or Program Execution Complete.

Individual line execution times can be monitored in the Single Execution mode.

Times for executing entire programs can be monitored in the Automatic Execution mode.

Accuracy of the timing is +/-2 milliseconds.

The data will be transferred as follows:

nnnnnnnn(crlf)

H31 Increase Speed

Function This command requests that the target velocity be increased by the Deviation Frequency (L73) value.

Default Mode none

Command Format !H31(crlf)

Notes If the !H31 command is given while the motor is moving, the motor speed will increase by the Deviation Frequency (L73) value.

Related commands L73 (Deviation Frequency)

H32 Decrease Speed

Function The H32 command requests that the target velocity be decreased by the Deviation Frequency (L73) value.

Default Mode none

Command Format !H32(crlf)

Notes If the !H32 command is given while the motor is moving, the motor speed will decrease by the Deviation Frequency (L73) value.

Related Commands L73 (Deviation Frequency)

H33 Set Incremental Position Mode

Function This command selects the Incremental Position mode. All moves will be made in the plus or minus direction from the **present position.**

Default Mode Incremental mode (H33)

Command Format H33(crlf)

Notes **H33 will be ignored if a Feed Hold ($) is active.**

Related Commands X, (Select X axis)
Y, (Select Y axis)

H34 Set Absolute Position Mode

Function This command selects the Absolute Position mode. All moves in this mode are referenced from the **electrical home position.**

Default Mode Incremental mode (H33)

Command Format H34(crlf)

Notes **H34 will be ignored if a Feed Hold ($) is active.**

Related Commands X, (Select X axis)
Y, (Select Y axis)

H35 Motor Windings On

Function The H35 code causes current to be applied to the motor windings at all times.

Default Mode Windings On (H35)

Command Format H35(crlf)

Notes **Use the H35 command when holding torque is required at standstill.**

Related Commands X, (Select X axis)
Y, (Select Y axis)

H36 Motor Windings Off

Function This command causes current to be removed from the motor windings when motion ceases.

Default Mode Windings On (H35)

Command Format H36(crlf)

Notes Without winding current, the motor will have no holding torque. Therefore, **the H36 command should only be used when holding torque at standstill is not needed.**

Whenever the Motor Windings Off function is active, an additional time delay is introduced during programmed motion cycles to allow winding current to build in the motor prior to the start of motion and to decay after motion has occurred. This delay is approximately 50 milliseconds prior to motion and 50 milliseconds after motion has stopped. Therefore, the shortest possible time between moves is approximately 100 milliseconds.

When the Control is in the Windings Off mode, the Reduce Current feature will be ignored.

Related Commands X, (Select X axis)
Y, (Select Y axis)

H37 Enable Boost Current

Function When the H37 command is issued, motor winding current will be increased during acceleration and deceleration by 50% of the value set by the dip switches on the drive.

Default Mode Disable Boost Current (H38)

Command Format H37(crlf)

Notes An additional time delay is introduced when the Boost Current feature is active to allow time for the motor winding current to build prior to motion and to decay after motion. This delay is approximately 10 milliseconds before the motion and 10 milliseconds after motion has stopped.

The Boost feature should be used when additional torque is needed for starting or stopping.

Related Commands X, (Select X axis)
Y, (Select Y axis)

H38 Disable Boost Current

Function This command cancels Boost current during acceleration and deceleration.

Default Mode Disable Boost Current (H38)

Command Format H38(crlf)

Related Commands X, (Select X axis)
Y, (Select Y axis)

H39 Enable Reduce Current

Function This command will cause the motor current to be reduced at standstill by 50% of the value which has been selected on the drive. This condition lowers heating of the motor and also reduces holding torque at standstill.

Default Mode Disable Reduce Current (H40)

Command Format H39(crlf)

Notes An additional time delay is introduced during programmed motion cycles whenever the Reduce feature is active. This delay allows winding current to build in the motor prior to motion and to decay after motion has stopped. The delay is approximately 10 milliseconds prior to motion and 10 milliseconds after motion has stopped.

Activate the Reduce feature only when maximum holding torque at standstill is not needed.

The Reduce command will be ignored when the Windings Off feature is active.

Related Commands X, (Select X axis)
Y, (Select Y axis)

H40 Disable Reduce Current

Function This command cancels the Reduce Current mode.

Default Mode Disable Reduce Current (H40)

Command Format H40(crlf)

Related Commands X, (Select X axis)
Y, (Select Y axis)

H41 Transfer Program Repeat Count

Function This command sends the remaining Repeat Count value.

Default Mode none

Command Format H41(crlf)
!H41(crlf)

Notes If the present Repeat Count is zero and a Cycle Start command is given, the L47 (Repeat Count) value is transferred to the Repeat Count. The count remaining can be interrogated using this command.

The Repeat Count is decremented each time a Program End or line 400 is encountered during program execution.

The Repeat Count is zero when the Control is in the Continuous Execution or the Single Execution mode.

The data will be transferred in the following format:
nnnn(crlf)

H42 Transfer Last External BCD Value

Function This command sends the Last External BCD read value, execution of a G36, G37, or G38 command.

Default Mode none

Command Format H42(crlf)
!H42(crlf)

Notes The data will be transferred in the following format:
snnnnnnnn(crlf)

H43 Transfer Expansion I/O Status

Function This command transfers the present status of Expansion Inputs and Outputs.

Default Mode none

Command Format H43(crlf)
!H43(crlf)

Notes The data will be transferred in the following format:

```
nnnnnnnn nnnnnnnn(crlf)
n------- -------- Expansion Input 8
-n------ -------- Expansion Input 7
--n----- -------- Expansion Input 6
---n---- -------- Expansion Input 5
----n--- -------- Expansion Input 4
-----n-- -------- Expansion Input 3
------n- -------- Expansion Input 2
-------n -------- Expansion Input 1

-------- n------- Expansion Output 8
-------- -n------ Expansion Output 7
-------- --n----- Expansion Output 6
-------- ---n---- Expansion Output 5
-------- ----n--- Expansion Output 4
-------- -----n-- Expansion Output 3
-------- ------n- Expansion Output 2
-------- -------n Expansion Output 1
```

This command is only valid if the OPTO 22 option is selected, L20 nn1nn.

H44 Transfer Register Contents

New code per Rev. B

Function This command transfers the contents of the nine registers.

Command Format
H44 crlf
!H44 crlf

Notes The transfer format is:

snnnnnnnnnn (crlf)	Register 1
snnnnnnnnnn (crlf)	Register 2
snnnnnnnnnn (crlf)	Register 3
snnnnnnnnnn (crlf)	Register 4
snnnnnnnnnn (crlf)	Register 5
snnnnnnnnnn (crlf)	Register 6
snnnnnnnnnn (crlf)	Register 7
snnnnnnnnnn (crlf)	Register 8
snnnnnnnnnn (crlf)	Register 9

H60 Transfer Present Speed

Function The H60 command requests that the present velocity be transferred.

Default Mode none

Units pulses per second

Command Format
H60(crlf)
!H60(crlf)

Notes The data will be transferred in the following format:
Xnnnnnn(crlf) or Ynnnnnn(crlf)

H85 Transfer System Error Status

Function The H85 code requests the transfer of the Motion Error status of the Control.

Default Mode none

Command Format H85(crlf)

Notes The data will be transferred in the following format:

```
nnnnnnnnn(crlf)

n-------- Drive not ready
-n------- Index From Run Limit exceeded
--n------ CW Software Limit Exceeded
---n----- CCW Software Limit Exceeded
----n---- CW Hardware Limit Exceeded
-----n--- CCW Hardware Limit Exceeded
------n-- Program Error
-------n- Closed Loop Error
--------n Hardware or Software Clear
```

Example !H85(crlf) The data transfer results might be as follows:

000001000(crlf)

This transfer indicates the active axis CCW hardware Limit was exceeded.

H86 Transfer Data Error Status

Function The H86 code is used to request the transfer of the Control's Data Error Status.

Default Mode none

Command Format H86(crlf)

Notes The format of the transferred data will be as follows:

```
nnnnnnnn nnnn0000(crlf)
n------- -------- Illegal G Code                                  0=inactive   1=active
-n------ -------- Illegal L Code                                  0=inactive   1=active
--n----- -------- Subroutine or Loop nesting                      0=inactive   1=active
---n---- -------- F data out of range                             0=inactive   1=active
----n--- -------- X or Y data out of range                         0=inactive   1=active
-----n-- -------- N data out of range                              0=inactive   1=active
------n- -------- Line requires F data                            0=inactive   1=active
-------n -------- Line requires X or Y data
                                                                   0=inactive   1=active

-------- n------- G code illegal in current cycle                 0=inactive   1=active
-------- -n------ Switching Axis while Jogging                    0=inactive   1=active
-------- --n----- Attempted Index during a Jog cycle              0=inactive   1=active
-------- ---n---- G Code cannot be executed from Line 0           0=inactive   1=active
-------- ----n--- No External Data Source                        0=inactive   1=active
-------- -----n-- Illegal comparison opposite direction          0=inactive   1=active
-------- ------n- X/Y sign required                              0=inactive   1=active
-------- -------n No Trigger for IFR/RMH cycle                   0=inactive   1=active
```

Illegal G Code:
Set whenever an illegal G code is encountered during a continuous motion cycle.

Illegal L Code:
Set when the data field for an L code is outside the legal range.
Set when an illegal L code is entered.
Reset when L Code data is in range or when program execution begins.

Subroutine or Loop Nesting:
Set during program execution if Subroutine or Loop nesting beyond four levels is detected.
Reset when program execution begins.

F data out of range:
Set during program execution if the F value is outside the legal range.
Set if the Strobe F (G38) value is out of range.
Set if the feedrate override value is out of range. (!F)
Reset when F data is in range or program execution begins.

X or Y data out of range:
Set during program execution if the X or Y value is outside the legal range.
Reset when program execution begins.

N data out of range:
Set when an addressed line number is out of range (0 to 400).
Set if the Strobe N (G37) value is out of range (1 to 400).
Reset when an addressed line is in range or when program execution begins.

Line requires F data:
Set during program execution if a required F field is omitted.
Reset when program execution begins.

Line requires X or Y data:
Set during program execution if a required X or Y field is omitted.
Reset when program execution begins.

G Code illegal in current cycle:
Set during program execution when an illegal G code is encountered during a program jog (G48) or Continuous Execution Mode is enabled (G60).
Reset when program execution begins.

Switching Axis while Jogging:
Set during a program Jog (G48) if a command requires Axis switching.
Reset when program execution begins.

Attempted Index during a Jog Cycle:
Set during a program Jog (G48) if a command requiring an Index is encountered.
Reset when program execution begins.

G Code cannot be executed from Line 0:
Set if an illegal G code is executed on Line 0.
Reset when program execution begins.

No External Data Source:
Set when a G code requires External BCD Data but the external source is not enabled.
Reset when program execution begins.

Illegal comparison opposite direction:
Set when the Control is in the Absolute Position Mode and Wait for distance command (G62) cannot be achieved, motion requires opposite direction.
Reset when program execution begins.

X/Y sign required:
Set when a G code requires an Axis designation and a direction.
Reset when program execution begins.

No Trigger for IFR/RMH cycle:
Set when an IFR (G10) or RMH (G78) command is executed and a Trigger source for the command is not defined, L20 code defines trigger source.
Reset when program execution begins.

If a data error occurs during program execution the line it occurred on can be interrogated using the H87 command.

H87 Transfer Program Error Line Number

Function This command requests the transfer of the program line number with an error detected during program execution.

Default Mode none

Command Format H87(crlf)

Notes The data will be transferred in the following format:

Nnnn(crlf)	(if program was terminated on a programming error)
(crlf)	(if no program error was detected)

H88 Transfer Execution Status

Function This command requests the transfer of the present execution cycle.

Default Mode none

Command Format H88(crlf)
!H88(crlf)

Notes The data will be transferred in the following format:

```
nnnnnnnn nnnnnnn0(crlf)
n------- -------- Program Execution          0=inactive  1=active
-n------ -------- Switch Execution           0=inactive  1=active
--n----- -------- Program Stop              0=inactive  1=active
---n---- -------- Slide Hold                0=inactive  1=active
----n--- -------- Delay occurring           0=inactive  1=active
-----n-- -------- Waiting for input          0=inactive  1=active
------n- -------- Waiting for velocity       0=inactive  1=active
-------n -------- Read external data         0=inactive  1=active

                                                  0=X axis     1=Y axis
-------- n------- Active Axis
-------- -n------ REH execution               0=inactive  1=active
-------- --n----- RMH execution               0=inactive  1=active
-------- ---n---- IFR execution               0=inactive  1=active
-------- ----n--- Jogging                     0=inactive  1=active
-------- -----n-- Stepping                    0=inactive  1=active
-------- ------n- Indexing                    0=inactive  1=active
-------- -------n Waiting for position        0=inactive  1=active
```

When a Slide Hold has been executed, the current cycle will still be active.

H99 Transfer Unit Type

Function This command transfers the Factory established Unit designation.

Default Mode none

Command Format H99(crlf)

Notes Up to 32 characters and a (crlf) will be sent in response to this command.

% Activate Continuous H Code Transfer Mode

Function

This command will set the active Control into the Continuous Transfer mode for designated H codes. When active the data requested by the designated H Code will be transferred continuously until the mode is canceled. This mode will be canceled by receipt of a "%CRLF" or a hardware clear or RS232 Clear "*"

Command Format

%Hnn(crlf)
!%Hnn(crlf)

Notes

Designated H Codes:

H13	Transfer Current Program Line
H15	Transfer Current Line Number
H17	Transfer Active Axis Absolute Electrical Position
H19	Transfer Mode Status
H20	Transfer Output Status
H21	Transfer Input Status
H22	Transfer Encoder Position (Closed Loop Control Only)
H26	Transfer Error Count (Closed Loop Control Only)
H27	Transfer Position Verification Status (Closed Loop Control Only)
H41	Transfer remaining repeat value
H43	Transfer Opto 22 Input/Output Status
H60	Transfer Present Velocity
H88	Transfer Execution Status

% is ignored if the Listen mode is active (&).

The L98 (Delay Between Continuous H Codes) sets the delay between continuous H code transfers.

Transfer Format

(data)(crlf) see the individual H codes for the specific format

This page intentionally left blank

CLOSED LOOP OPERATION

CLOSED LOOP OPERATION (Optional)

DESCRIPTION

The Closed Loop Option provides a means of interfacing the Control to an incremental encoder either on the motor, or on the driven mechanism. This may be needed in some applications to maintain a desired position within a particular deadband or to assure that an index command attains the destination position. This option can be easily-installed via a front-panel plug-in module and is only applicable to the X axis. This module contains the operating system program, battery-backed memory and encoder interface circuitry for detecting mechanical position.

INSTALLING THE CLOSED-LOOP OPTION

CAUTION: The module for open- or closed-loop operation contains electronic components that may be damaged by static electricity (ESD). To prevent ESD damage, you must be certain to work in a grounded, static-protected area, and only handle the module by its metal front, or by the edges of the circuit card. Keep the module in its static-protective plastic bag except when installing or removing it from the control.

CAUTION: Be sure to disconnect ac power before installing or removing the open- or closed-loop electronic module.

To install the optional plug-in module, first be sure to disconnect ac power to the control. Then, remove the two screws that fasten the module in place in the lower left corner of the control enclosure (when viewed from the front). Next, carefully unplug the original (open-loop) module from its DIN connector socket by pulling it firmly out from the front of the control chassis. Then install the closed-loop module by firmly seating it in the connector socket, noting the proper orientation of the module (observe the screw hole locations on the metal shroud). Finally, replace the two screws, connect the proper encoder to the "D" connector (see Encoder Selection, below), and restore ac power.

DESCRIPTION OF VARIOUS CLOSED LOOP OPERATING MODES

The closed loop option has four operating modes which are selected by programming the Position Verification Enable/Disable parameter, L90, with a numeric value 0 through 3. These four operating modes are:

L90 0 Following Error Detection, Position Maintenance at Standstill, and Automatic Index Command Correction are all disabled in this mode. However, absolute encoder position is maintained. This mode is used primarily for setting up the closed loop parameters.

L90 1 Following Error Detection and Position Maintenance at Standstill are enabled, but Automatic Index Command Correction is disabled in this mode. This mode is used to maintain position at standstill and to abort a cycle if a following error is detected (stalled motor detected during motion).

L90 2 Following Error Detection, Position Maintenance at Standstill, and Automatic Index Command Correction are all enabled in this mode. If a following error is detected during motion, then that cycle will automatically be stopped and restarted after the designated delay (L97 parameter value). This action will continue until the cycle is completed or the number of attempts allowed (L96 parameter value) has elapsed. A Jog cycle will simply be restarted in the designated direction. An Index will be restarted in the designated direction until the target position is attained.

L90 3 This mode is identical to the L90 2 mode except for the restarting of an Index after a following error is detected. If a following error is detected when the motor is moving toward the target position, the control will command the motor to return to the original starting position after a designated delay, the L97 parameter value. When this position is attained, the motor will move toward the target position after the designated delay (L97 parameter value). This cyclic action will continue until the target is attained or the number of attempts allowed, the L96 parameter value, has elapsed.

RELATED COMMANDS

L53 Following Error Exceeded Characters
L54 Unable to Correct Characters
L59 Output Configuration
L87 Following Error
L90 Position Verification Enable/Disable
L93 In Position Bandwidth
L94 Invert Encoder Direction
L95 Encoder Quadrature Scale Factor
L96 Number of Correction Attempts allowed
L97 Delay between Correction Attempts
H22 Transfer Absolute Encoder Position
H26 Transfer Pulse Error Count
H27 Transfer Closed Loop Status

ENCODER SELECTION

The encoder used with the Closed Loop Option should be a 5-volt incremental encoder with differential outputs. A shielded cable with a maximum length of 25 feet (7.6 meters) can be used. If a cable length greater than 25 feet is required an external 5 volt supply located at the encoder must be provided, and pin 9 of the encoder cable must not be used (this is what normally carries +5 Vdc from the control to the encoder). The 9-pin "D" connector at the Control end must have a metal case with the shield tied to the metal case to provide immunity from electrical interference. The encoder input and power supply specification are located in the SPECIFICATION section of this manual. An Encoder Compatibility Chart is shown below.

Note: If a single ended encoder is used a shielded cable with a maximum length of 5 feet (1.5 meters) can be used.

Encoder Compatibility Chart

Encoder Line Count	Drive Resolution	L95 Value	Encoder Line Count	Drive Resolution	L95 Value
50	200	0	100	200	2
200	200	4	400*	200	8
100	400	0	200	400	2
400*	400	4	800	400	8
500*	1000	2	1000	1000	4
500*	2000	0	1000	2000	2
800	3200	0	360	7200	5**
500*	10000	5**	1000	20000	5**
1250	25000	5**	2500	50000	5**

* Encoders with 400, 500, and 1250 lines and with differential outputs are available from Superior Electric.

** When L95=5, the quadrature pulse count is multiplied by 5 to obtain the encoder count. All other values of L95 result in a division into the quadrature pulse count. For example, when using a 400 line encoder with a drive setting of 200 pulses/revolution, L95=8 results in dividing the 1600 quadrature pulse count by 8 to obtain 200 counts per revolution. When using a 500 line encoder with a drive setting of 10,000 counts per revolution, L95=5 results in multiplying the 2000 quadrature pulse count by 5 to obtain 10,000 counts per revolution.

Encoder Input Connector Assignments (9-pin "D" connector)

C12 Lead Color*	C2, C4, C5 Lead Color*	All Types Terminal #	Function	Cable Wire Color	9-Pin "D" Connector Pin Number
Green	Green	1	B+ "Count"(single-ended)	Black	1
Blue	White/Green	2	B- "Count"	White	2
White	Orange	3	A+ "Gate"(single ended)	Red	3
Yellow	White/Orange	4	A- "Gate"	Green	4
Black	Black	5	Encoder common Vo	Brown	5
			not used	Blue	6
Orange	White	7	I+ "Index"	Orange	7
Brown	White/Black	8	I- "Index" (single ended)	Yellow	8
Red	Red	6	Encoder +5 Vdc	Purple	9

* Cable supplied with control must be modified for connection to encoder leads.

CAUTION DO NOT CONNECT THE ENCODER TO THE SERIAL OUTPUT CONNECTOR OR DAMAGE MAY OCCUR.

Note **If a single-ended encoder is used (the complement is not available), the B- and A- signal lines must be left unconnected.**

ENCODER SETUP

NOTE **In the following discussion, the terms "Cr" and "Lf" denote carriage return and line feed, respectively.**

1) Establish Control communications first. Transmit the following command:

<01Cr

Note **If the Control ID is not 01, substitute the correct Control ID for 01.**

2) The position verification Enable/Disable parameter should be disabled before attempting to do the encoder setup. Transmit the following command line to disable the position verification option:

L90 0CrLf

3) Select the proper encoder from the Encoder Compatibility chart and enter the appropriate value in the Encoder Quadrature Scale Factor Parameter, L95.

Example: Drive Resolution is 10000 pulses/revolution. A 500 line encoder, 2000 pulses per revolution, was selected. From the Encoder Compatibility Chart the L95=5 should be selected. Transmit the following command line to set the value.

L95 5CrLf

4) A parameter for setting the allowable Following error, which is the difference between the mechanical and electrical position, should be decided on first. The minimum value for this parameter should be four full steps. This value is entered as the L87 value. The factory default for this parameter is 10 full steps. Transmit the following command line to set the Following Error value "nnn":

L87 nnnCrLf

Example: L87 50CrLf
Enters 50 pulses as the allowable Following Error.

5) The allowable In Position Bandwidth, position deviation about the ideal position, which is the L93 parameter value, should be decided on next. This value should be attainable without excessive hunting. If the Encoder is not connected directly to the motor, this value must include the lash in the mechanics. The L93 value should be smaller than the L87 value. The factory default varies with the step resolution selected.

Example: L93 5CrLf
Enters 5 pulses as the In Position Deadband.

Refer to Appendix A for the factory default values. The use of L93 values less than the factory default may result in excessive hunting.

6) The number of correction attempts should now be decided upon. This value should be entered as the L96 value. The factory default is 100.

Example: L96 10CrLf
Enters 10 as the Number of Correction Attempts Allowed.

7) The delay between Correction Attempts (L97 parameter). This delay time should be sufficient to allow the motor to settle after a motion. In general, the larger the inertia load the longer the delay should be. The factory default is 100 milliseconds.

Example: L97 500CrLf
Enters 500 milliseconds as the delay between Correction Attempts.

8) Load the following program on the MDI line, Line 0, in the control:

N0 X1000 f1000CrLf
Move the X axis 1000 pulses in Clockwise direction at a feedrate of 1000 pulses/second.

9) Set the X axis to the Home Position by transmitting the following command line:

X,H9H17h22CrLf
Set Home position (H9), transmit Electrical Position (H17) and transmit Encoder Position (h22).
response:

X+0000000000 (Electrical Position)
X+0000000000 (Encoder Position)

10) Execute move and verify positions.

H1H17h22CrLf
Response after motion:

X+0000001000 (Electrical Position)
X+0000001000 or
X-0000001000 (Encoder Position)

if response is X-0000001000 then change the Invert Encoder Direction parameter, L94, by transmitting one of the following commands:

L94 0CrLf or L94 1CrLf
Then repeat steps 8 and 9.

11) The desired operating mode must now be selected, L90 0 or L90 1 or L90 2 or L90 3.

Refer to **Description of various Closed Loop Operating Modes** section.

The desired operating mode "n" can be selected by transmitting the following command line:

L90 nCrLf
Selects Mode "n" (0-3) as the desired operating mode.

OPERATION

When Position Maintenance is enabled, (when L90=1 or 2 or 3), the motor position will be maintained within the L93 deadband value at standstill. The Control moves the motor to the ideal position, center of the deadband, when the In Position Deadband (the L93 parameter value) is exceeded. The L97 parameter sets the minimum time between correction attempts. This timeout begins when motor motion is completed.

A positional error is detected when the Following Error, (the L87 parameter value) is exceeded during motion. When a following error condition is detected, motion stops with no deceleration. The Delay between correction Attempts, L97 parameter, begins. After the timeout, one of the following will occur:

1) Cycle termination will occur, L90=1 or Number of Position Correction Attempts, L96 parameter value, has elapsed. At this time the Encoder Position is transferred to the X Axis Absolute Position.

Note **When a Closed Loop Error occurs the Fault Led is illuminated. This error is reset when program execution begins or a motion cycle begins.**

2) Restart of Current cycle and decrement of Position Correction Attempt count, L96 parameter value.

Note **The Position Correction Attempt count value is set to the Position Correction Attempt parameter, L96, value when a non-correction cycle, Jog or Step or Index, takes place in the X axis.**

This page intentionally left blank

SECURITY CODE

SECURITY CODE

DESCRIPTION

A security code provides a means of denying access to parameters and program information. A security code of 00000000 disables the function. This is the factory setting for the security code.

When the security code feature is enabled, access to the parameters and program information upon power turn-on until the security code is entered. The code can only be changed under the following conditions:

The security code is 00000000
A matching security code has been entered
Initialization of Factory Defaults (see Appendix G), Z99 nnn, has been performed. When this is done, the parameters will be changed to the factory default values and the program contents will be cleared. It is recommended that a listing be made of the parameters and program contents before using the Z99 command.

RELATED COMMANDS

The following commands are disabled when the security code is active:

Lnn nnnnnnnn	Parameter changing via RS232
H11	Clear Current Program Line
H12	Clear Program
H13	Transfer Current Program Line And Contents
H14	Transfer Program
H16	Transfer Parameters
H24	Enable Trace Mode
Gnn, Xsnnnnnnnn, Ysnnnnnnnn and Fnnnnnn	Line Data (0-400)
X,	Select X Axis
Y,	Select Y Axis

COMMAND SYNTAX

s nnnnnnnn nnnnnnnn range is 00000000 - 99999999

NOTES

When the security code is active, the Y axis cannot be selected via RS232. However, a manual axis select can be utilized if one of the inputs is configured for X/Y select (L58, function D).

Program line data cannot be altered when the security code is active. Therefore, manual index motion cannot be executed unless a specific line was programmed for an index motion while the security code was inactive.

Example: **N000 X1000 F2000** was programmed while the security code was inactive. Index motion can be commanded when the security code is active using **N0H1crlf**.

Manual Step (H2H6 or H2H7) and Jog motions (H3H6 or H3H7) can be commanded when the security code is active

When the security code is active, L codes (parameters) cannot be changed via RS232 codes, but can be altered during program execution using the G29 command.

TROUBLESHOOTING

TROUBLESHOOTING

The status information provided by the SLO-SYN 2000 Control can be invaluable as a trouble shooting aid. The most obvious status information is provided by the front panel LED's.

Using the "FAULT" led as an error indicator, it is a relatively simple procedure to determine just what caused the error condition. If you are using a "dumb terminal" or computer that can emulate a "dumb terminal", first establish communications with the Control by issuing the command "<01 crlf". If the Control ID number is set to something other than 1 substitute that number instead of "01". The Control will respond by transmitting a "=" character. This character indicates the Control is ready to communicate. After receiving the "=" response, issue the command "H85 crlf", The Control will respond by transmitting the system error status.

Refer to the description of the H85 command in the Software Reference section of the manual for a complete listing of the possible system error responses.

In the case of a programming error, more specific information can be obtained by issuing the command "H86 crlf". The Control will respond by transmitting the program error status. Refer to the description of the H86 command in the Software Reference section of the manual for a complete list of the possible program errors.

When a fault occurs during program execution, it is possible to determine just where in the program the fault occurred by issuing the command "H87 crlf", the Control will respond by transmitting the program line number that was being executed when the error occurred. For example if the Control had transmitted "N007" in response to the "H87" command, it can be determined that program line number 7 was executing when an error occurred.

Applicable Codes: H85 Transfer system error status
H86 Transfer program error status
H87 Transfer program error line #

TRACE MODE

Trace mode allows the user to view the program line by line as the program executes. When trace mode is enabled the Control will transmit via the serial port the contents of the program line that is being executed. The device connected to the serial port must be capable of receiving and displaying standard ASCII characters. A "dumb terminal" or computer emulating a "dumb terminal" is a perfect device for viewing the program lines as they are transmitted from the Control. To enable the trace mode issue the following command "!H24 crlf".

To disable the trace mode issue an "!H25 crlf" command. The time it takes to execute the program will increase when trace mode is active.

Applicable Codes: H24 Program trace on
H25 Program trace off

USING THE "DRY RUN" FEATURE

The SLO-SYN 2000 Control has a "dry run" capability that allows the program to be executed in real time without motor motion. This feature is most useful in the development and debugging stages of program development. To enable the "dry run" feature the L06 parameter must be set to "L06 1nn" where "nn" represents the desired execution format. Once the "dry run" feature has been enabled motor motion will not be allowed. The program, however, will execute normally. To return to "normal" operating conditions where motor motion is allowed, set the L06 parameter to "L06 0nn" where "nn" again represents the desired execution format.

Applicable Codes: L06 Execution format

INDICATOR LED's

The "POWER" led will be illuminated whenever adequate AC power is provided to the Control. This led must be on before the Control will be able to communicate or execute its stored program. The "MOTION BUSY" led will be on when the motor is commanded to move. In general if the light is on, the Control has received or generated a motion command.

The "FAULT" led will light when a motion or program error has occurred or when a drive error has occurred. This led being on is the first indication that the Control has encountered a condition which has caused an error. The conditions which are indicated by the Fault LED are as follows:

- Drive not ready
- IFR limit exceeded
- CW software limit active
- CCW software limit active
- CW hardware limit active
- CCW hardware limit active
- Program error encountered
- Closed loop error

HALTING A FAULTY PROGRAM THAT KEEPS REPEATING

A situation can occur where an incorrect program may contain commands that cause it to keep repeating. If this happens, the program can be halted by downloading another program to the SS2000I Programmable Motion Controller. Downloading of a new program will cause the control to cease operation of any program that is in process.

TROUBLESHOOTING COMMANDS

Two Z commands are provided to aid in troubleshooting a problem. These commands, Z81 and Z99, are described in Appendix G of the manual.

MOST FREQUENTLY ASKED QUESTIONS

DOES THE ABSOLUTE POSITION COUNTER UPDATE DURING THE JOG AND STEP COMMANDS?

The absolute position counter maintains the electrical position during any motion regardless of the motion command or the source (serial or parallel) of the command. If a motion clear, clockwise limit or counterclockwise limit is actuated during motion, the absolute position counter contains erroneous data. The home position should be reestablished subsequent to those actions.

I PROGRAMMED A DELAY ON A PROGRAM LINE, BUT WHY IS THE DELAY LONGER THAN I PROGRAMMED?

The total delay time on a line is the value programmed in the X field plus the L44 line delay value. The L44 value must be accounted for to attain the desired dwell time between moves. The L44 value is used to allow sufficient "settling time" for the motor to come to rest before another move is attempted. If this time is not long enough, position errors may result because the motor has not stopped its "ringout" (oscillation) following a move. L44 is only added following program lines calling for motion to occur.

As an example, the following program is executed with:

L44 = 100 milliseconds:

N001 G91 X+1000 F500	(index 1000 pulses)
N002 G04 X+1000	(delay)
N003 X-1000 F500	(index -1000 pulses)
N004 G30	(program end)

The total delay time from the end of line 1 to the beginning of line 3 is the L44 delay when line 1 is done, plus the G04 delay of line 2 (the 1000 msec value in the X Field). Thus the total delay equals 100 + 1000 = 1100 milliseconds. The G04 X field should be set to the desired dwell time minus the L44 value.

G04 delay=(desired dwell)-(L44 value)

In this case, if the desired dwell is 1000 milliseconds, then

G04 delay=(1000)-(100)

= 900 milliseconds

WHAT IS THE DIFFERENCE BETWEEN "RETURN TO MECHANICAL HOME" AND "RETURN TO ELECTRICAL HOME"?

The Electrical Home (+000000 in the absolute position counter) is set either at the motor position occupied upon energizing the Control or with a "Set Electrical Home" command (H09 or G77) at any time.

The Return to Electrical Home command (H08 or G76) causes the motor to index from its current position (as indicated in the absolute position counter) in the direction opposite the absolute counter sign. This will continue until the motor reaches the absolute position counter setting of +0000000000 (Electrical Home).

Return to Mechanical Home establishes the Electrical Home with a mechanical switch. An H10 or G78 causes the motor to run in the direction programmed in the L08 parameter until the Home Limit Switch input is active. Then the motor will offset the direction and distance set with the L17 parameter. When that motion is complete, the motor position becomes Electrical Home (the absolute position counter is reset to +0000000000).

Therefore, use the Return to Mechanical Home (H10 or G78) to accurately establish Electrical Home upon power up or when the absolute position counter contains erroneous data (i.e., subsequent to a Motion Clear *). Then use the Return to Electrical Home (H08 or G76)to position the motor at a known location.

WHAT IS THE DIFFERENCE BETWEEN ABSOLUTE AND INCREMENTAL MODES OF MOTION?

Incremental Motion (set by G91) moves the number of pulses contained in the X field with the direction determined by the X field sign (+ indicating clockwise direction and counterclockwise direction set by -). The moves are made relative to the present motor position.

Absolute Motion (set by G90) moves to the position contained in the X field by moving the direction and the number of pulses to achieve that position as related to the present absolute position counter. The absolute position counter is set to zero or the home position upon power up and with the H09 and G77 commands.

HOW DO MY COMPUTER AND CONTROL PROGRAMS ACTUALLY CONNECT?

This process is called a "Handshake." Some computers use a hardware connection to accomplish this. The SS2000I Control does not. The handshaking technique used by the Control is a software method utilizing the "XON/XOFF protocol." These are discussed in detail in the Glossary under "XON/XOFF".

I AM UNABLE TO ESTABLISH SERIAL COMMUNICATIONS BETWEEN MY CONTROL AND MY HOST COMPUTER. WHAT CAN THE PROBLEM BE?

First, make sure that all the hardware connections are made, cable lengths do not exceed specified limits and that no power cables are along side of communications cables.

If no communications occur, it could be a problem with your communications parameters. Check to see that the serial communications settings on the top-mounted switches are identical for the host device and the Control. The factory default settings for the Control are 9600 baud, 8 data bits, 2 stop bits, and no parity.

SPECIFICATIONS

SPECIFICATIONS FOR PROGRAMMABLE MOTION CONTROL SS2000I

I. MECHANICAL

Dimensions:	9.5"H x 2.5"W x 5.6"D (24 x 6.4 x 14.2 mm)
Weight:	3 lb. 9 oz. (1.62 kg)

II. ENVIRONMENTAL

TEMPERATURE	
Operating	0° C to 50° C
Storage	-40° C to 75° C
HUMIDITY	95% noncondensing

III. ELECTRICAL

INPUT POWER	
Voltage:	90-265 VAC 50/60 HZ Current: less than 0.4 amperes at 115 Vac Fuse: 2 A (normal blow)
ISOLATED CURRENT-LIMITED POWER SUPPLY FOR USE WITH SINGLE POINT I/O)	24Vdc (+/- 10%) at 0.75A
INPUTS & OUTPUTS:	
SINGLE POINT I/O ELECTRICAL SPECS.	Outputs shall be able to drive a shorted load indefinitely without damage. No suppression required for inductive loads.
OUTPUTS (SINK MODE)	
Load Power Supply	Can use built-in 24 Vdc supply or external 12 to 24 Vdc supply
Current Rating	50 mA
Voltage Rating	24 Vdc
ON STATE VOLTAGE	
@ 50 mA	2.0 V max.
OFF STATE LEAKAGE	
@ 24VDC	0.6 mA max.
OUTPUTS (SOURCE MODE)	
Current Rating	50 mA
ON STATE VOLTAGE	
@ 50 mA	20 V min.
Off State Leakage	0.6 mA max.
INPUTS (SINK MODE)	
On State Voltage Range	0 - 12 volts
Input Current @ 12 V	2.3 mA
Input Current @ 0 V	6.5 mA
INPUTS (SOURCE MODE)	
On State Voltage Range	10 - 24 Vdc
Input Current @ 10 V	2.3 mA
Input Current @ 24 V	6.5 mA

IV. INTERFACE FUNCTIONS AND CONNECTOR PIN-OUTS

BCD SWITCH INTERFACE / I/O EXPANSION PORT
(25 PIN "D" FEMALE CONNECTOR)

This port consists of eight inputs and eight outputs. It is capable of reading two banks of BCD switch data (each with seven digits plus sign). The control can also interface to "OPTO 22" type PB16A I/O modules (or equivalent, such as Potter & Brumfield type 210-16), providing an additional eight inputs and eight outputs.

BCD PORT CONNECTOR PIN NUMBER	BCD SWITCH FUNCTION	EXPANSION I/O FUNCTION
1	DATA 1	INPUT 0
2	DATA 2	INPUT 1
3	DATA 4	INPUT 2
4	DATA 8	INPUT 3
5	DATA 10	INPUT 4
6	DATA 20	INPUT 5
7	DATA 40	INPUT 6
8	*DATA 80 or (-) SIGN	INPUT 7
9-11	NC	NC
12, 13	NOT USED	+5V
14	BANK 1 - X1	OUTPUT 0
15	BANK 1 - X100	OUTPUT 1
16	BANK 1 - X10, 000	OUTPUT 2
17	BANK 1 - X1, 000, 000	OUTPUT 3
18	BANK 2 - X1	OUTPUT 4
19	BANK 2 - X100	OUTPUT 5
20	BANK 2 - X10, 000	OUTPUT 6
21	BANK 2 - X1,000,000	OUTPUT 7
22, 23	NC	NC
24, 25	NOT USED	GND

* Sign for x1,000,000; otherwise, Data 80

DRIVE INTERFACE

There are two drive interfaces. Each has its own eight-position removable terminal strip connector. The pin assignments for each are as follows:

PULSE 0-1 MHZ SQ. WAVE
DIRECTION
WINDINGS OFF
REDUCE
BOOST
RESET
DRIVE READY INPUT
OPTO SUPPLY (+5V)

When "Drive Ready" is on, the drive is powered and ready.
When "Drive Ready" is off, the drive is not powered or has failed.

SERIAL PORT

The serial port is used to program the unit or to interface to a host controller. There are two serial interfaces: RS232 and RS485. Units can be daisy chained via either interface. Communication to the host via RS232 or RS485 is switch selectable.

RS232 (9 Pin "D" Female Connector)

Pin Assignments For RS232 Connector

1	V0
2	CHAIN OUT (TX)
3	Rx
4	V0
5	V0
6	ECHO
7	CHAIN IN
8	+5V
9	+5V

RS485 (5 Position Removable Terminal Strip Connector)
Pin Assignments are As Follows:

Tx -
Tx +
Rx -
Rx +
GND

SERIAL PORT PARAMETERS:

BAUD RATE	(300,1200,2400,9600) hardware switch selectable
DATA FORMAT	Hardware switch selectable
DATA LENGTH	7 or 8 bits
PARITY	Odd, even or none
UNIT ID (01-99)	Hardware switch selectable

OPTIONAL ENCODER INTERFACE:

9 pin "D" female connector (on plug-in Memory Module). Operates with differential or single-ended encoders. Count rate to 400,000 pulses per second.

ENCODER CONNECTOR	DIFF ENCODER	SINGLE ENDED ENCODER
1	B	B
2	B-	NOT USED
3	A	A
4	A-	NOT USED
5	ENCODER GND	ENCODER GND
6	NC	NC
7	Z	NOT USED
8	Z-	Z
9	ENCODER +5V	ENCODER +5V

This page intentionally left blank

GLOSSARY

GLOSSARY

ABSOLUTE MODE

Motion mode in which all motor movements are specified in reference to an electrical home position.

ABSOLUTE POSITION COUNTER

A data register in the Control which counts pulses delivered to the motor (via the drive). When the count is zero, this position is designated "Electrical Home".

ACCELERATION

The rate at which the motor speed is increased from its present speed to a higher speed (specified in pulses/second/second).

ACCURACY

The noncumulative incremental error which represents step to step error in one full motor revolution.

ALL WINDINGS OFF

Applying an average zero motor current at standstill to alleviate motor heating or eliminate holding torque.

AMBIENT TEMPERATURE

The temperature of the air surrounding the motor or drive.

ASCII

(American Standard Code for Information Interchange). A format to represent alphanumeric and control characters as seven-or eight-bit codes for data communications.

ATTENTION CHARACTER

<nn, where "nn" is a unique integer from 1-99 (set by use of the unit ID# select switches) that is assigned to a Motion Control arrayed in a multi-Control system. The Attention Character directs the program command to the specified Motion Control.

BASE SPEED

Starting speed for the motor (also known as low speed).

BAUD RATE

The rate of serial data communications expressed in binary bits per second.

BCD

(Binary Coded Decimal), a format to represent the digits 0 through 9 as four digital signals. Systems using thumbwheel switches may program commands using BCD digits.

A BCD digit uses a standard format to represent the digits 0 through 9 as four digital signals. The following table lists the BCD and complementary BCD representation for those digits. The Motion Control uses the complementary BCD codes because the signals are active low.

BCD code table (0 = low state, 1=high state)

digit	BCD code	complementary BCD code
0	0000	1111
1	0001	1110
2	0010	1101
3	0011	1100
4	0100	1011
5	0101	1010
6	0110	1001
7	0111	1000
8	1000	0111
9	1001	0110

To represent numbers greater than 9, cascade the BCD states for each digit. For example, the number 79 is 0111:1001.

BOOST CURRENT

Increase of motor current during acceleration and deceleration to provide higher torques and faster ramp rates.

CLEAR, *	Immediate Code which when received from your computer will command the indexer to immediately halt all motor motion and program execution.
COLLECTORS (OPEN)	A transistor output that takes the signal to a low voltage level with no pull-up device; resistive pull-ups are added to provide the high voltage level.
CYCLE START	Command H01 to initiate program execution.
CYCLE STOP, #	Immediate Code which when received from your computer will command the indexer to stop program execution after the present program line completes its task.
DAISY-CHAIN	A method to interface multiple Motion Controls via RS232 to a single host using only one serial port.
DAMPING	A method of applying additional friction or load to the motor in order to alleviate resonance and ringout.
DECELERATION	The rate in which the motor speed is decreased from its present speed to a lower speed (specified in pulses/second/second).
DEVICE ADDRESS	A unique number used to assign which Motion Control in a multi-drive stepper system is to respond to commands sent by a host computer or terminal. Device addresses from 1 - 99 are set by means of the ID # select switch. "00" is reserved to address all Motion Controls in a system. Factory default is 01.
DWELL	A programmed delay in program execution (specified in milliseconds. The duration set is in the X/Y field.
ELECTRICAL HOME	The motor position at which the Absolute Position Counter is zero.
EOT	Character (ASCII code 4) appended to an end of a data transmission to signify the conclusion of the data transfer.
F CODE	A six-digit field on a program line that either designates the Feedrate or contains additional data as required by the X/Y field in a G code.
FEEDHOLD, $	Immediate Code which when received from your computer will command the indexer to immediately bring motor motion to a controlled stop with the programmed deceleration.
FEEDRATE	Specified in an F field, the motor speed (in pulses per second) in which a move will occur; stored and executed in association with several G and H codes.
FRICTION	Force that is opposite to the direction of motion as one body moves over another.
FULL-STEP	Position resolution in which 200 pulses corresponds to one motor revolution in a 200 step per revolution motor.
G CODES	Preparatory commands that are stored and executed within a program that designate the function of the program line.
H CODES	Commands that set Motion Control modes and control manual and program execution.
HALF-STEP	Position resolution in which 400 pulses corresponds to one motor revolution for a 200 step per revolution motor.
HANDSHAKE	A computer communications technique in which one computer's program links up with another's. The Motion Control uses a software "Xon, Xoff" handshake method. See "XON" below.

HIGH SPEED MODE

Command H4 that allows the motor to accelerate to a high speed as all motion occurs at the programmed feedrates.

INCREMENTAL MODE

Motion mode in which all motor movements are specified in reference to the present motor position.

INDEXER

A Microprocessor-based programmable motion control that controls move distance and speeds; possesses intelligent interfacing and input/output capabilities.

INERTIA

Measurement of a property of matter that a body resists a change in speed (must be overcome during acceleration).

INERTIAL LOAD

A "flywheel" type load affixed to the shaft of a step motor. Sometimes used as a damper to eliminate resonance.

INSTABILITY

Also frequently called, "mid-range instability" or "mid-range resonance," this term refers to a resonance that occurs in the 500 - 1,500 steps/sec range. Mid-range instability is important because it refers to a loss of torque or a stalled motor condition at higher stepping rates.

Since step motors do not start instantaneously above the mid-range resonance frequency, an acceleration scheme will have to be used to pass through the troublesome region.

JOG MODE

Command H3 that configures H6 (clockwise motion) and H7 (counterclockwise motion) to move the motor continuously until a Feedhold $ command is issued.

L CODES

Contain the values for the global parameters.

LOAD

This term is used several ways in this and other manuals.

LOAD (ELECTRICAL): The current in Amperes passing through the motor windings.

LOAD (MECHANICAL): The mass to which motor torque is being applied (the load being moved by the system).

LOAD (PROGRAMMING): Transmits a program from one computer to another. "UPLOAD" refers to transmitting a program from a host computer (where a program has been written) to the Motion Control where it will be used. "DOWNLOAD" refers to transmitting a program from a Motion Control back to the host computer.

LOW SPEED MODE

Command H5 that inhibits the motor from accelerating to a high speed as all motion occurs at the programmed low speed value (also known as "Base Speed Only")

MECHANICAL HOME

The position where a switch input is used as a reference to establish electrical home.

MDI

"Manual Data Interface". A program line (line 0) designated for manual operation to facilitate rapid program loading and execution. Line 0 is stored in non-volatile memory (BBRAM). When the MDI line is executed, the execution format (L06) is ignored as line 0 executes once per each Cycle Start (H01) command, and the line pointer remains at 0.

MICROSTEPPING

A sophisticated form of motor control that allows finer resolution than full step (200 pulses per revolution) or half step (400 pulses per revolution) by adjusting the amount of current being applied to the motor windings.

For 200 step per revolution motors, typical microstepping levels are 1/10-step and 1/125 step (2000 pulses per revolution and 25,000 pulses per revolution, respectively).

N CODE

Sets the program line number.

NEGATIVE LOGIC

An "inverted" way of interpreting the state of inputs and outputs. When current flows through that input or output, the control's program treats it as inactive or "off". Set L58 - (minus sign) for negative logic.

Physical Condition	How Program Interprets I/O State
current flow = switch closed	"off" = inactive = logic 0
no current flow = switch open	"on" = active = logic 1

NESTING

The ability of an active subroutine to call another subroutine. The Motion Control can nest up to four levels.

NONVOLATILE MEMORY

Data storage device that retains its contents even if power is removed. Examples are EEPROM and battery-backed RAM.

OPTO-ISOLATION

The electrical separation of the logic section from the input/output section to achieve signal separation and to limit electrical noise. The two systems are coupled together via a transmission of light energy from a sender (LED) to a receiver (photo transistor).

PARITY

An error checking scheme used in serial communications (via the RS-232 port) to ensure that the data is received by a Motion Control is the same as the data sent by a host computer or programmer such as the SSP525.

POSITIVE LOGIC

The conventional way of interpreting the state of inputs and outputs. When current flows through that input or output, the control's program treats it as active or "on". Set L58 + (plus sign) for positive logic.

Physical Condition	How Program Interprets I/O State
current flow = switch closed	"on" = active = logic 1
no current flow = switch open	"off" = inactive = logic 0

REDUCE CURRENT

Reduction of motor current during standstill to alleviate motor heating.

RESOLUTION

The minimum position command that can be executed. Specified in steps per revolution or some equivalent.

RETURN TO ELECTRICAL HOME

Function which allows the Motion Control to position the motor to a known reference point. See also Absolute Position Counter and Mechanical Home.

RETURN TO MECHANICAL HOME

Function which allows the Motion Control to position the motor to seek a switch to establish electrical home.

RINGOUT

The transient oscillatory response (prior to settling down) of a step motor about its final position.

RS232-C

EIA (Electronic Industries Association) communication standard to interface devices employing serial data interchanges.

RS274-D

EIA (Electronic Industries Association) programming standard for numerically-controlled machines.

SET HOME

A command which, after executing, references the present motor position as "home" and the absolute position counter is set to 0. See also Absolute Position Counter.

SINKING

An input or output that is brought to a low level (signal common or low side of the input/output power supply) when active.

SOURCING

An input or output that is brought to a high level (the voltage used for the input/output power supply) when active.

STEP MODE

Command H2 that configures H6 (clockwise motion) and H7 (counterclockwise motion) to move the motor one pulse.

SUBROUTINE

A sequence of lines that may be accessed from anywhere in a program to preclude having to program those lines repetitively. This allows shorter and more efficient programs.

TORQUE

Product of the magnitude of a force and its force arm (radius) to produce rotational movement.

TRANSLATOR

A motion control device (also called "translator drive") that converts pulses to motor phase currents to produce motion.

X/Y Field

An eight digit field with a sign on a program line that either designates the motor move or contains additional data as required by the G code.

XON/XOFF

A computer software "handshaking" scheme used by a Motion Control.

The Motion Control sends an XOFF character (ASCII Code 19) when it receives a command string with a Carriage Return and has less than 50 characters remaining in its serial port buffer. The Control sends and Xon when available buffer space reaches 100 characters or in response to an ID attention with adequate buffer space remaining. Since it is impossible for the host device to immediately cease transmissions, the next three characters (subject to the total serial buffer capacity of forty characters) received subsequent to the Motion Control sending the XOFF character will be stored in the Motion Control's serial buffer (a memory dedicated to store characters that are in the process of transmission).

Similarly, the Motion Control will not transmit data if the host device has sent an XOFF character to the Control; Motion Control transmissions will resume when the Control receives an XON character.

APPENDIX

L CODE LISTING

Control Parameters

L13 nnnnnnnn,nnnnnnnn	Step Increment for X,Y axes
L16 nnnnnnnn	Index From Run Travel Limit
L18 snnnnnnnn,snnnnnnnn	Clockwise Travel Limit for X,Y axes
L19 snnnnnnnn,snnnnnnnn	Counterclockwise Travel Limit for X,Y axes
L20 nnnnn	Power Up Configuration
L41 nnn	Auto Start Line Number

RS232 Parameters

L01 "aaaaaaaaaaaaaaaaaaaa"	Message # 1
L02 "aaaaaaaaaaaaaaaaaaaa"	Message # 2
L03 "aaaaaaaaaaaaaaaaaaaa"	Message # 3
L04 "aaaaaaaaaaaaaaaaaaaa"	Message # 4
L05 "aaaaaaaaaaaaaaaaaaaa"	Message # 5
L26 n	Command Acknowledge
L48 nnn	Program Line Count Designator
L49 nn	Parameter Transfer Designator
L50 nn	Parameter Transfer Count
L52 "aaa"	Buffer Warning Characters
L53 "aaa"	Following Error Exceeded Characters
L54 "aaa"	Unable To Correct Characters
L55 "aaa"	Line Done Characters
L56 "aaa"	Program Done Characters
L98 nnnn	Delay Between %Hnn Code Transmissions

Program Parameters

L06 nnn	Execution Format
L44 nnnn	After Motion Delay
L47 nnnn	Repeat Count
L58 "shhhhhhhhhhhhh"	Input Configuration
L59 "hhhhhhhh"	Output Configuration
L91 nnnnnnnn,nnnnnnnn	External Distance BCD Scaling factor

Feed Rate Parameters

L09 nnnnnn,nnnnnn	Jog Speed for X,Y axes
L12 nnnnnn,nnnnnn	Low Speed for X,Y axes
L14 nnnnnn,nnnnnn	Home Speed for X,Y axes
L73 nnnnnn	Deviation Frequency

Ramp Parameters

L10 nnnnnnnn,nnnnnnnn	Deceleration Rate for X,Y axes
L11 nnnnnnnn,nnnnnnnn	Acceleration Rate for X,Y axes
L71 nnnnnn	Ramp Frequency Limit
L72 n	Ramp Profile Select

Mechanical Home Parameters

L08 s,s	Mechanical Home Direction for X,Y axes
L17 snnnnnnnn,snnnnnnnn	Offset From Mechanical Home for X,Y axes

Backlash Compensation

L43 nnnn	Delay Between Index and Backlash
L66 snnnnnnnn,snnnnnnnn	Backlash Compensation for X,Y axes

Closed Loop Parameters

L87 nnnn	Following Error
L90 n	Position Verification Enable/Disable
L93 nnnn	In Position Bandwidth
L94 n	Invert Encoder Direction
L95 n	Encoder Quadrature Scale Factor
L96 nnnn	Number of Attempts for Position Correction
L97 nnnn	Delay Between Correction Attempts

FACTORY DEFAULT SUMMARY

Parameter	Default for all resolutions (Z99 nnn)	Parameter	Default for all resolutions (Z99 nnn)
L01	"Message #1 goes here"	L48	020
L02	"Message #2 goes here"	L49	00
L03	"Message #3 goes here"	L50	00
L04	"Message #4 goes here"	L52	"
L05	"Message #5 goes here"	L53	"
L06	002	L54	"
L08(X,Y)	"+"	L55	"
L16	00000000	L56	"
L17(X,Y)	+00000000	L58	"+2187600000000"
L18(X,Y)	-00000000	L59	"00000000"
L19(X,Y)	+00000000	L66(X,Y)	+00000000
L20	02000	L72	0
L26	0	L90	0
L41	001	L91	10000
L43	0050	L94	1
L44	0050	L96	0100
L47	0000	L97	0100
		L98	0100

	Step Size (Z99 nnn)									
Parameter	1	2	5	10	16	36	50	100	125	250
L09(X,Y)	1000	2000	5000	10000	16000	36000	50000	100000	125000	250000
L10(X,Y)	1000	2000	5000	10000	16000	36000	50000	100000	125000	250000
L11(X,Y)	1000	2000	5000	10000	16000	36000	50000	100000	125000	250000
L12(X,Y)	300	600	1500	3000	4800	10800	15000	30000	37500	75000
L13(X,Y)	1	2	5	10	16	36	50	100	125	250
L14(X,Y)	1000	2000	5000	10000	16000	36000	50000	100000	125000	250000
L71	10000	20000	50000	100000	160000	360000	500000	999999	999999	999999
L73	100	200	500	1000	1600	3600	5000	10000	12500	25000
L87	10	20	50	100	160	360	500	1000	1250	2500
L93	0	0	0	1	2	5	5	10	15	25
L95	8	4	2	0	0	5	5	5	5	5

Recommended Encoder (Lines/Rev) for Step Size (Z99 nnn)									
1	2	5	10	16	36	50	100	125	250
400	400	500	500	800	360	500	1000	1250	2500

Note: The factory default value is "Z99 50".

APPENDIX B: H CODES

H01	Cycle Start
H02	Step Mode
H03	Jog Mode
H04	High Speed Mode
H05	Low Speed Mode
H06	CW Direction
H07	CCW Direction
H08	Return To Electrical Home
H09	Set Electrical Home
H10	Return To Mechanical Home
H11	Clear Current Program Line
H12	Clear Program Lines Using L48
H13	Transfer Current Program Line
H14	Transfer Program Lines Using L48
H15	Transfer Current Line Number
H16	Transfer Parameters
H17	Transfer Absolute Electrical Position
H19	Transfer Mode Status
H20	Transfer Output Status
H21	Transfer Input Status
H22	Transfer Absolute Encoder Position
H23	Transfer Software Revision Level
H24	Program Trace Mode On
H25	Program Trace Mode Off
H26	Transfer Pulse Error Count
H27	Transfer Closed Loop Status
H29	Transfer Program Execution Time
H31	Target Velocity Increase
H32	Target Velocity Decrease
H33	Incremental Position Mode
H34	Absolute Position Mode
H35	Motor Windings On
H36	Motor Windings Off
H37	Enable Boost Current
H38	Disable Boost Current
H39	Enable Reduce Current
H40	Disable Reduce Current
H41	Transfer Remaining Repeat Value
H42	Transfer Last External BCD value
H43	Transfer Opto 22 Input/Output Status
H44	Transfer Register Contents
H60	Transfer Present Velocity
H85	Transfer Motion Error Status
H86	Transfer Data Error Status
H87	Transfer Program Error Line Number
H88	Transfer Execution Status
H99	Transfer Model Type

APPENDIX C: IMMEDIATE COMMANDS

!Fnnnnnnn	Feed Rate Override
!H04	High Speed Mode
!H05	Low Speed Mode
!H13	Transfer Current Program Line
!H15	Transfer Current Line Number
!H17	Transfer Absolute Electrical Position
!H19	Transfer Mode Status
!H20	Transfer Output Status
!H21	Transfer Conditional Input Status
!H22	Transfer Encoder Position
!H24	Program Trace Mode On
!H25	Program Trace Mode Off
!H26	Transfer Pulse Error Count
!H27	Transfer Closed Loop Status
!H31	Target Velocity Increase
!H32	Target Velocity Decrease
!H41	Transfer Remaining Repeat Value
!H42	Transfer Last External BCD value
!H43	Transfer OPTO 22 Input/Output Status
!H44	Transfer Register Contents
!H60	Transfer Present Velocity
!H85	Transfer Motion Error Status
!H86	Transfer Data Error Status
!H87	Transfer Program Error Line Number
!H88	Transfer Execution Status

APPENDIX D: MISCELLANEOUS COMMANDS

<nnCR	Device Attention
! (ASCII 33)	Immediate Command
%Hnn	Activate Continuous H Code Transfer Mode
%CR	Deactivate Continuous H Code Transfer Mode
Nnnn	Line Number
Gnn	Preparatory Command
Xsnnnnnnnn	Program Index Distance or G Code data field
Ysnnnnnnnn	Program Index Distance or G Code data field
Fnnnnnnn	Program Feed Rate or G Code data field
Xoff (ASCII 19)	Stop Transmission
Xon (ASCII 17)	Resume Transmission
DC2 (ASCII 18)	Enable Transmission of L code characters (L52-L56) and G05 characters
DC4 (ASCII 20)	Disable Transmission of L code characters (L52-L56) and G05 characters
^H (ASCII 8)	Backspace And Delete
^X (ASCII 24)	Delete RS232 Buffer
<00CR	All Devices Listen Only
<00&	All Devices Listen Mode
<00@	Cancel Listen Mode For All Devices
<nn?	Device Active And Acknowledge ID
nn&	Device Listen Mode
nn@	Device Cancel Listen Mode And Become Active
? (ASCII 63)	Device Acknowledge ID
* (ASCII 42)	Clear command, \stops program execution, motion, and clears the RS232 buffer
$ (ASCII 36)	Feed Hold (Controlled stop)
# (ASCII 35)	Program Stop
/ (ASCII 47)	Transfer Normal Buffer Character Count Remaining
\ (ASCII 92)	Transfer Immediate Buffer Character Count Remaining

Notes: "s" is an ASCII sign (+ or -)
"n" is an ASCII number (0 to 9)
The remaining symbols are ASCII characters (decimal value shown)
"*", "$", "#", "<nn", "^H", "^X", and "?" are immediate commands which when received from your computer will command the indexer to perform the indicated action.

APPENDIX E: G CODE LISTING

code	X/Y Required	X/Y Optional	F Required	F Optional	Description
G04 G04	[Xnnnn] [Ynnnn]			{Fnnnnnn} {Fnnnnnn}	Delay (X) milliseconds Delay (Y) milliseconds
G05 G05	[Xn] [Yn]			{Fnnnnnn} {Fnnnnnn}	Transmit message (X) Transmit message (Y)
G06 G06	[Xn] [Yn]			{Fnnnnnn} {Fnnnnnn}	Hcode Execution Hcode Execution
G10 G10		{Xsnnnnnnnn} {Ysnnnnnnnn}		{Fnnnnnn} {Fnnnnnn}	Start Index From Run Primary Axis Start Index From Run Secondary Axis
G11 G11	[Xnnn] [Ynnn]		[Fnnnn] [Fnnnn]		Call Subroutine Line (X) repeat (F) times Call Subroutine Line (Y) repeat (F) times
G12 G12	[Xnnn] [Ynnn]			{Fnnnnnn} {Fnnnnnn}	Go to Line (X) Go to Line (Y)
G19	[X0000nnnn]		{Fnnn}		Test Register Compare Results
G20 G20	[Xnnnnnnnn] [Ynnnnnnnn]		[Fnnn] [Fnnn]		Branch on input Condition (X) primary inputs. Branch on input Condition (Y) secondary inputs.
G21 G21		{Xsnnnnnnnn} {Ysnnnnnnnn}	{Fnn} {Fnn}		Register Control Register Control
G22 G22	[Xnnnnnnnn] [Ynnnnnnnn]			{Fnnnnnn} {Fnnnnnn}	Wait for input Condition (X) primary inputs. Wait for input Condition (Y) secondary inputs.
G23					Increment Velocity.
G24					Decrement Velocity.
G25 G25	[Xnnnn] [Ynnnn]			{Fnnnnnn} {Fnnnnnn}	Loop start Repeat loop (X) times. Loop start Repeat loop (Y) times.
G26				{Fnnnnnn}	Loop End.
G27 G27		{Xsnnnnnnnn} {Ysnnnnnnnn}		{Fnnnnnn} {Fnnnnnn}	Set High Speed Mode X axis. Set High Speed Mode Y Axis.
G28 G28		{Xsnnnnnnnn} {Ysnnnnnnnn}		{Fnnnnnn} {Fnnnnnn}	Set Low Speed Mode X axis. Set Low Speed Mode Y Axis.
G29 G29		{Xsnnnnnnnn} {Ysnnnnnnnn}	[Fnn] [Fnn]		Program L code (F) X axis value. Program L code (F) Y axis value.
G30					End of Program.
G31					Stop Program.
G32					Return From Subroutine.
G36 G36				{Fnnnnnn} {Fnnnnnn}	Strobe X data. Strobe Y data.
G37				{Fnnnnnn}	Strobe N data and go to that line.
G38 G38		{Xsnnnnnnnn} {Ysnnnnnnnn}			Strobe F data. Strobe F data.
G47 G47	[Xnnnnnnnn] [Ynnnnnnnn]			{Fnnnnnn} {Fnnnnnn}	Set/Reset Outputs primary Outputs. Set/Reset Outputs secondary Outputs.
G48 G48	[Xs] [Ys]			{Fnnnnnn} {Fnnnnnn}	Start Continuous Jog cycle primary Axis. Start Continuous Jog cycle secondary Axis.
G49 G49					Stop Continuous Jog Cycle. Stop Continuous Jog Cycle.
G50 G50	[Xnnnnnnnn] [Ynnnnnnnn]			{Fnnnnnn} {Fnnnnnn}	Set/Reset internal flag condition. Set/Reset internal flag condition.
G51 G51	[Xnnnnnnnn] [Ynnnnnnnn]		[Fnnn] [Fnnn]		Branch on internal flag condition (X) to line (F). Branch on internal flag condition (Y) to line (F).
G52 G52	[Xn] [Yn]			{Fnnnnnn} {Fnnnnnn}	Select BCD switch bank 0 or 1 Select BCD switch bank 0 or 1
G53 G53	[Xn] [Yn]			{Fnnnnnn} {Fnnnnnn}	Select SS2000 Interface I/O Port Select SS2000 Interface I/O Port
G54 G54		{Xsn} {Ysn}	[Fn] [Fn]		Register Execution Register Execution
G55 G55	[Xnnn] [Ynnn]			{Fnnnnnn} {Fnnnnnn}	Error Trapping Error Trapping

APPENDIX E: G CODE LISTING, Continued

code	X/Y Required	X/Y Optional	F Required	F Optional	Description
G60		{Xsnnnnnnnn}		{Fnnnnnn}	Enable Continuous Execution Mode.
G60		{Ysnnnnnnnn}		{Fnnnnnn}	Enable Continuous Execution Mode.
G61		{Xsnnnnnnnn}		{Fnnnnnn}	Disable Continuous Execution Mode.
G61		{Ysnnnnnnnn}		{Fnnnnnn}	Disable Continuous Execution Mode.
G62	[Xnnnnnnnn]			{Fnnnnnn}	Wait for Distance (X) to be achieved.
G62	[Ynnnnnnnn]			{Fnnnnnn}	Wait for Distance (Y) to be achieved.
G63	[Xs]		{Fnnnnnn}		Wait for velocity (F) to be achieved in (X) axis.
G63	[Xs]		{Fnnnnnn}		Wait for velocity (F) to be achieved in (Y) axis.
G64		{Xsnnnnnnnn}		{Fnnnnnn}	Reduce Current On Mode for (X) axis.
G64		{Ysnnnnnnnn}		{Fnnnnnn}	Reduce Current On Mode for (Y) axis.
G65		{Xsnnnnnnnn}		{Fnnnnnn}	Reduce Current Off Mode for (X) axis.
G65		{Ysnnnnnnnn}		{Fnnnnnn}	Reduce Current Off Mode for (Y) axis.
G66		{Xsnnnnnnnn}		{Fnnnnnn}	Boost Current On Mode for (X) axis.
G66		{Ysnnnnnnnn}		{Fnnnnnn}	Boost Current On Mode for (Y) axis.
G67		{Xsnnnnnnnn}		{Fnnnnnn}	Boost Current Off Mode for (X) axis.
G67		{Ysnnnnnnnn}		{Fnnnnnn}	Boost Current Off Mode for (Y) axis.
G68		{Xsnnnnnnnn}		{Fnnnnnn}	Windings Off Mode for (X) axis.
G68		{Ysnnnnnnnn}		{Fnnnnnn}	Windings Off Mode for (Y) axis.
G69		{Xsnnnnnnnn}		{Fnnnnnn}	Windings On Mode for (X) axis.
G69		{Ysnnnnnnnn}		{Fnnnnnn}	Windings On Mode for (Y) axis.
G76		{Xsnnnnnnnn}		{Fnnnnnn}	Return to Electrical Home Primary Axis.
G76		{Ysnnnnnnnn}		{Fnnnnnn}	Return to Electrical Home Secondary Axis.
G77		{Xsnnnnnnnn}		{Fnnnnnn}	Set Electrical Home Primary Axis.
G77		{Ysnnnnnnnn}		{Fnnnnnn}	Set Electrical Home Secondary Axis.
G78		{Xsnnnnnnnn}		{Fnnnnnn}	Return to Mechanical Home Primary Axis.
G78		{Ysnnnnnnnn}		{Fnnnnnn}	Return to Mechanical Home Secondary Axis.
G79		{Xsnnnnnnnn}		{Fnnnnnn}	Set Absolute Position Primary Axis.
G79		{Ysnnnnnnnn}		{Fnnnnnn}	Set Absolute Position Secondary Axis.
G90		{Xsnnnnnnnn}		{Fnnnnnn}	Set Absolute Position Mode Primary Axis.
G90		{Ysnnnnnnnn}		{Fnnnnnn}	Set Absolute Position Mode Secondary Axis.
G91		{Xsnnnnnnnn}		{Fnnnnnn}	Set Incremental Position Mode Primary Axis.
G91		{Ysnnnnnnnn}		{Fnnnnnn}	Set Incremental Position Mode Secondary Axis.

APPENDIX F: PROGRAM WORKSHEETS

Control Parameters	
L Code Data	Function
L01 _	Message # 1
L02 _	Message # 2
L03 _	Message # 3
L04 _	Message # 4
L05 _	Message # 5
L06 _ _ _	Execution Format
L08 _ , _	Mechanical Home Direction
L09 _ _ _ _ _ , _ _ _ _ _	Jog Speed
L10 _ _ _ _ _ _ _ , _ _ _ _ _ _ _	Deceleration
L11 _ _ _ _ _ _ _ , _ _ _ _ _ _ _	Acceleration
L12 _ _ _ _ _ , _ _ _ _ _	Low Speed
L13 _ _ _ _ _ _ _ , _ _ _ _ _ _ _	Step Size
L14 _ _ _ _ _ , _ _ _ _ _	Home Speed
L16 _ _ _ _ _ _ _	IFR Travel Limit
L17 _ _ _ _ _ _ _ _ , _ _ _ _ _ _ _ _	Mechanical Home Offset
L18 _ _ _ _ _ _ _ _ , _ _ _ _ _ _ _ _	CW Software Travel Limit
L19 _ _ _ _ _ _ _ _ , _ _ _ _ _ _ _ _	CCW Software Travel Limit
L20 _ _ _ _ _	Power Up Configuration
L26 _	Command Acknowledge
L41 _ _ _	Auto Start Line Number
L43 _ _ _ _	Backlash Delay
L44 _ _ _ _	After Motion Delay
L47 _ _ _ _	Repeat Count
L48 _ _ _	Program Line Count
L49 _ _	Parameter Designator
L50 _ _	Parameter Count
L52 _ _ _	Buffer Warning Characters
L53 _ _ _	Following Error Characters
L54 _ _ _	Unable To Correct Characters
L55 _ _ _	Line Done Characters
L56 _ _ _	Program Done Characters
L58 _ _ _ _ _ _ _ _ _ _ _ _ _	Input Configuration
L59 _ _ _ _ _ _ _ _	Output Configuration
L66 _ _ _ _ _ _ _ , _ _ _ _ _ _ _	Backlash Compensation
L71 _ _ _ _ _	Ramp Frequency Limit
L72 _	Ramp Profile
L73 _ _ _ _ _	Deviation Frequency
L87 _ _ _ _	Following Error
L90 _	Position Verification Enable/Disable
L91 _ _ _ _._ _ _ _ , _ _ _ _._ _ _ _	External BCD Move Distance Scale Factor
L93 _ _ _ _	In Position Deadband
L94 _	Invert Encoder Direction
L95 _	Encoder Quadrature Scale Factor
L96 _ _ _ _	Correction Attempt Count
L97 _ _ _ _	Delay Between Attempts
L98 _ _ _ _	Delay between %Hnn transfers

APPENDIX F: PROGRAM WORKSHEETS, Continued

Hardware Settings	
RS232 Baud Rate (9600, 2400, 1200, 300)	_ _ _ _
RS232 Word Length (7,8)	_
RS232 Parity (Enable,Disable)	_ _ _ _ _ _ _
RS232 Parity (Odd,Even)	_ _ _ _
RS232,RS485	_ _ _ _ _
Control Identification (01-99)	_ _

APPENDIX F: PROGRAM WORKSHEETS, Continued

Program							
Line Number	G Code	Move Distance or Data	Feed Rate or Data		Line Number	G Code	Move Distance or Data	Feed Rate or Data
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____
N___	G__	X_____ Y_____	F_____		N___	G__	X_____ Y_____	F_____

APPENDIX G: TROUBLESHOOTING COMMANDS

Z81 Transfer Checksum

Function Checksums Program Lines or Parameters.

Units none

Range 0 - 1

Command Format

Z81 0 Checksum Program Lines 1 - 400
Z81 1 Checksum Parameters

Notes

Transfer response:
hhhhCrLf (hhhh = hexadecimal checksum)

Z99 Factory Defaults

Function Sets factory defaults for parameters, clears Program and verifies contents.

Units none

Range 1, 2, 5, 10, 16, 36, 50, 100, 125, 250

Command Format

Z99	sets 1/50 step (10000 pulses/rev)
Z99 nnn	sets 1/50 step (1000 pulses/rev)
Z99 1	sets full step (200 pulses/rev)
Z99 2	sets half step (400 pulses/rev)
Z99 5	sets 1/5 step (1000 pulses/rev)
Z99 10	sets 1/10 step (2000 pulses/rev)
Z99 16	sets 1/16 step (3200 pulses/rev)
Z99 36	sets 1/36 step (7200 pulses/rev)
Z99 50	sets 1/50 step (10000 pulses/rev)
Z99 100	sets 1/100 step (20000 pulses/rev)
Z99 125	sets 1/125 step (25000 pulses/rev)
Z99 250	sets 1/250 step (50000 pulses/rev)

Default Z99 50

Notes

Match the drive setting in pulses per revolution

Defaults for each setting are listed in Appendix A.

Responses

00CrLf	contents verified
15CrLf	error encountered

APPENDIX H
RED LION MESSAGE CENTER

A Red Lion Message Center can be used with the SS2000I Programmable Motion Control to display user messages. The contents of the messages are controlled by the parameters L01 through L05. The selection and transmission of messages are controlled by the execution of a program line containing a G05 code.

Substituting a "*" for a programmed "~" signifies the End Of Transmission to the Message Center Slave unit. Substituting an "|" for a programmed "^" identifies the top line of the message text on the 2 x 20 line display.

Example The following messages are displayed on the 2 x 20 line dispaly:

Select Program 1-3 Then Press Select	Program Executing	Select Input active

L01 "Select Program 1-3"	(Select Program 1-3|)
L02 "Then Press Select~"	(Then Press Select*)
L03 " ~"	(*)
L04 "Program Executing^"	(Program Executing|)
L05 "Select Input Active^"	(Select Input Active|)
N1 G20 X22222220 F5	If Select input inactive, go to line 5
N2 G05 X5	display "Select input active
N3 G05 X3	Blank lower display
N4 G22 X22222220	wait for Select input inactive
N5 G52 X1	select BCD bank 1
N6 G05 X1	display "Select Program 1-3~"
N7 G05 X2	display "Then Press Select"
N8 G22 X22222221	wait for Select input active
N9 G21 X3 F30	3 > Register 3 (max. prog #)
N10 G21 F40	load BCD data into Register 4
N11 G21 X5 F44	BCD = 0?
N12 G19 X2212 F1	branch if yes to line 1
N13 G21 X4 F34	BCD > 3?
N14 G19 X2122 F1	branch if yes to line 1
N15 G05 X4	display "Program Executing"
N16 G05 X3	blank lower display
N17 G19 X2212 F300	branch to line 300 if program 3
N18 G21 X2 F30	2 > Register 3
N19 G21 X4 F34	BCD = 2?
N20 G19 X2212 F200	branch if yes to line 200
N21	start of Program 1
N200	start of Program 2
N300	start of Program 3

SS2000I Control Setup

Unit ID BCD selection
01

DIP switch settings
232 Host position
Parity Off position
8-bit data selection
2400 baud position only

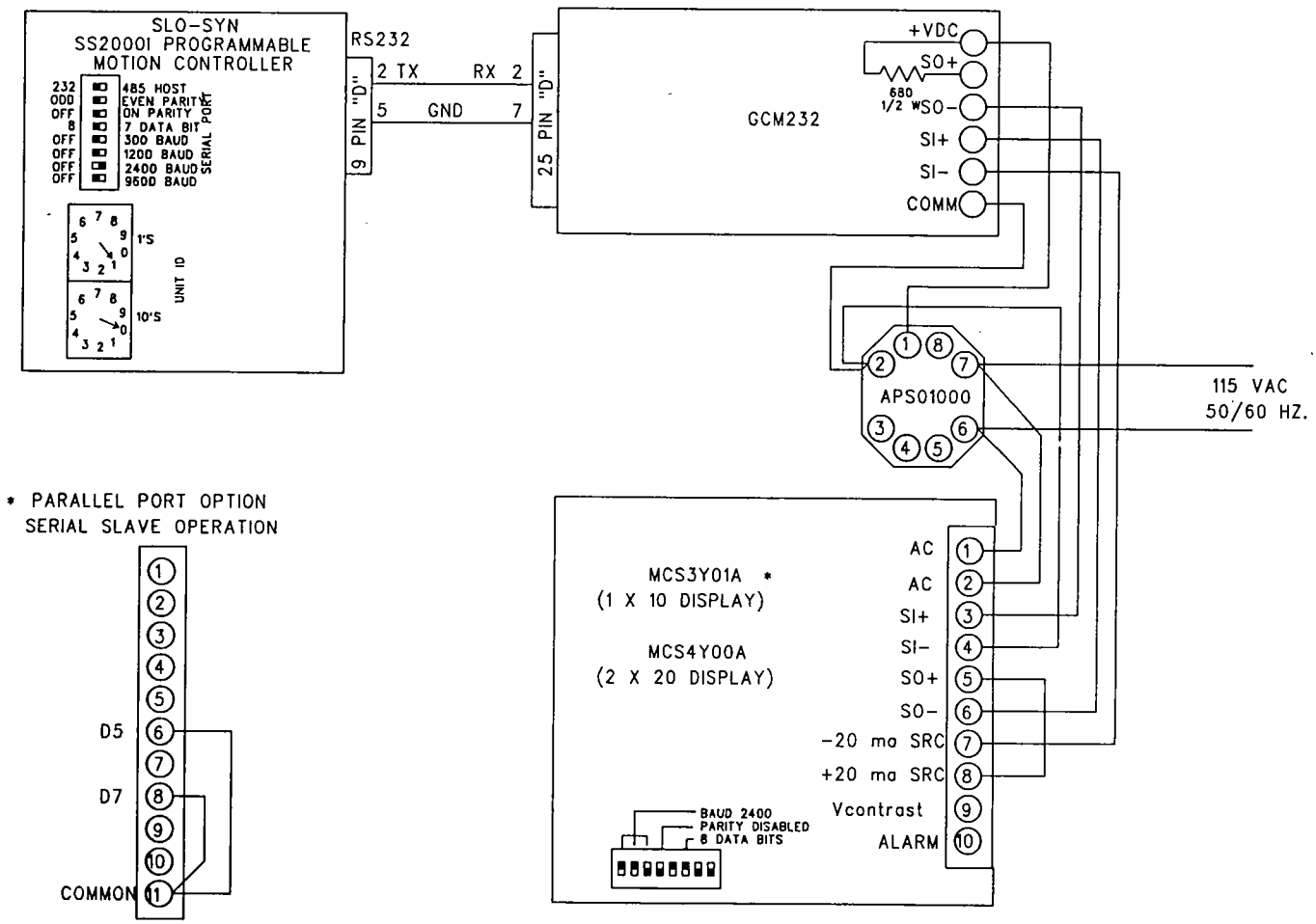
Message Control Setup

DIP switch settings

BR0 on position
BR1 on position
BR2 off position
PARITY Off position (parity off)
EVEN/ODD on position
7/8 BIT on position

2400 Baud

Wiring Use Wiring Diagram shown below

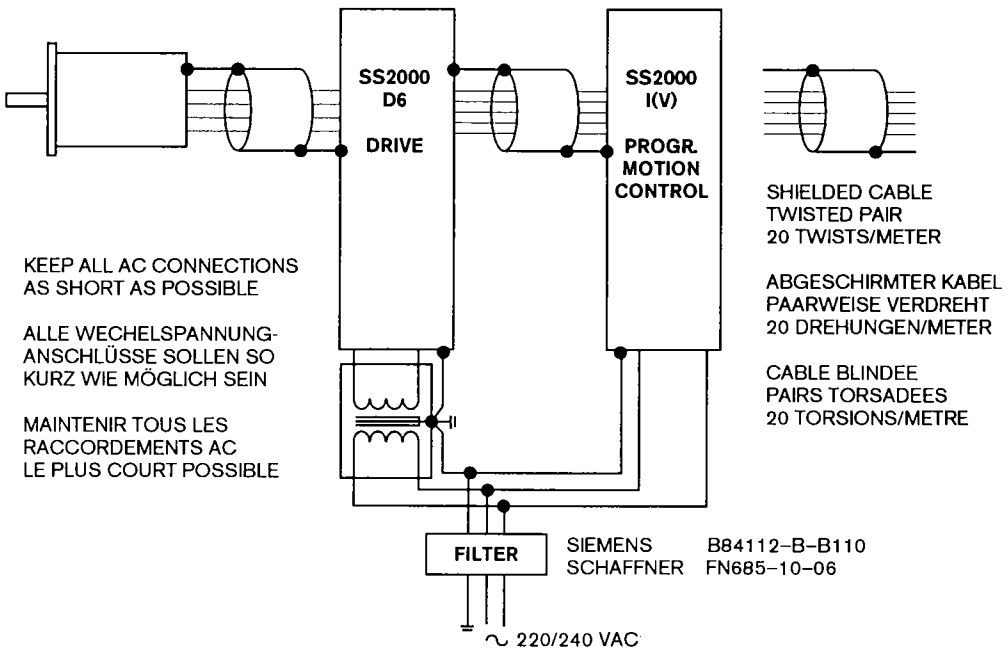


Wiring Diagram for RED LION CONTROLS SLAVE MESSAGE CENTER

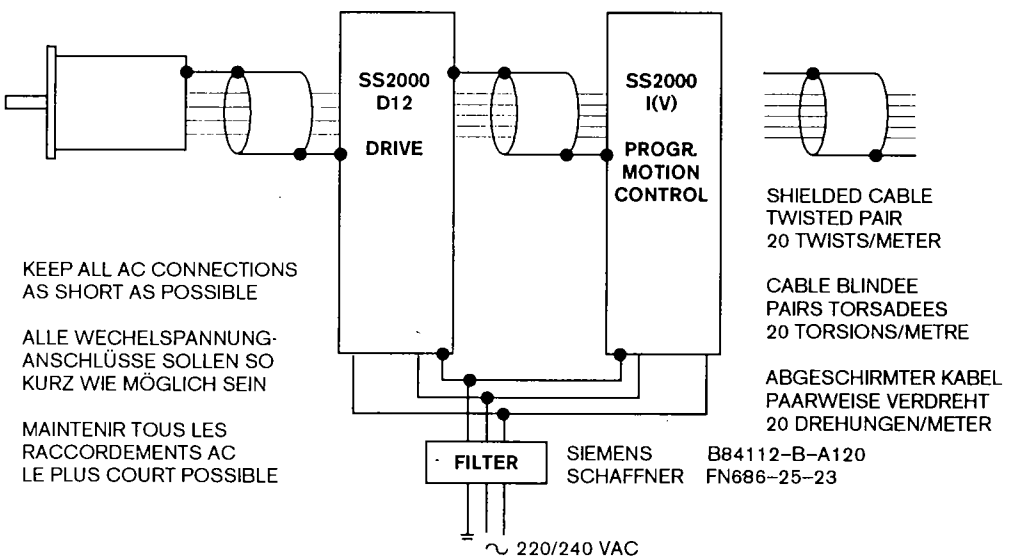
APPENDIX I
CONNECTIONS FOR CE EMC REQUIREMENTS

Mandatory connections to meet CE EMC requirements
Vorgeschriebener Anschluss zur Übereinstimmung mit CE EMC Normen
Branchement obligatoîre afin de respecter la norme CE - EMC

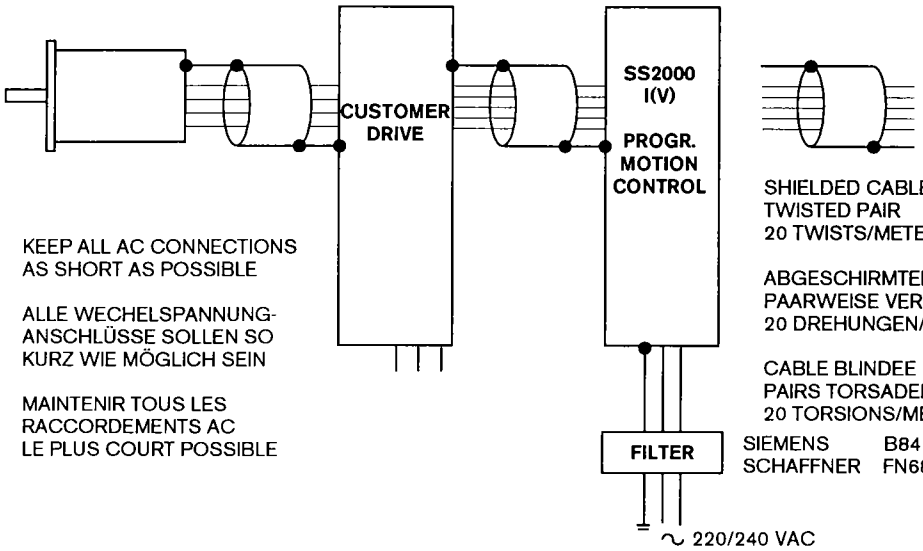
1. SLO-SYN SS2000I(V) + SS2000D6 + MOTOR



2. SLO-SYN SS2000I(V) + SS2000D12 + MOTOR



3. SLO-SYN SS2000I(V) + CUSTOMER DRIVE + MOTOR



KEEP ALL AC CONNECTIONS
AS SHORT AS POSSIBLE

ALLE WECHELSPANNUNG-
ANSCHLÜSSE SOLLEN SO
KURZ WIE MÖGLICH SEIN

MAINTENIR TOUS LES
RACCORDEMENTS AC
LE PLUS COURT POSSIBLE

SHIELDED CABLE
TWISTED PAIR
20 TWISTS/METER

ABGESCHIRMTER KABEL
PAARWEISE VERDREHT
20 DREHUNGEN/METER

CABLE BLINDEE
PAIRS TORSADEES
20 TORSIONS/METRE

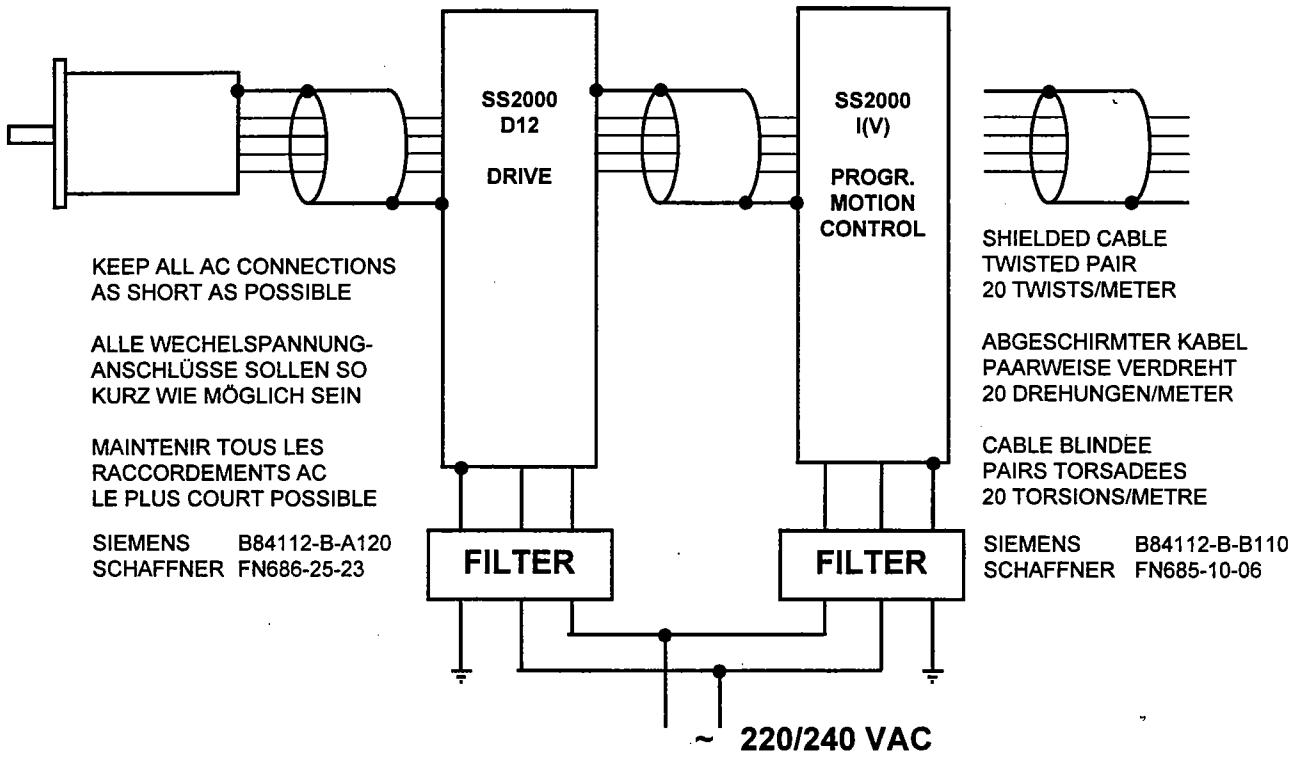
SIEMENS B84112-B-B110
SCHAFFNER FN685-10-06

This page intentionally left blank

ADDENDUM TO
INSTRUCTIONS FOR SLO-SYN® SS2000I PROGRAMMABLE MOTION CONTROL
400030-025 REV. E

The following diagram illustrates the proper connections between the SS2000D12 drive and an SS2000I or SS2000I-V programmable motion control when the system must meet CE EMC requirements. This diagram replaces diagram 2. at the bottom of page 13-15 of Instruction 400030-025 Rev. E and will be added to the instruction at the next printing. Diagram 1. on page 13-15 for connection of the SS2000D6 and an SS2000I(V) to meet CE EMC requirements is correct as is.

2. SLO-SYN SS2000I(V) + SS2000D12 + MOTOR



Printed in U.S.A.

DISTRIBUTION COAST-TO-COAST AND INTERNATIONAL

Superior Electric products are available nationwide through an extensive authorized distributor network. These distributors offer literature, technical assistance and a wide range of models off the shelf for fastest possible delivery.

In addition, Superior Electric sales engineers and manufacturers' representatives are conveniently located to provide prompt attention to customers' needs. Call the nearest office listed for ordering and application information or for the address of the closest authorized distributor.

WARRANTY AND LIMITATION OF LIABILITY

Superior Electric (the "Company"), Bristol, Connecticut, warrants to the first end user purchaser (the "purchaser") of equipment manufactured by the Company that such equipment, if new, unused and in original unopened cartons at the time of purchase, will be free from defects in material and workmanship under normal use and service for a period of one year from date of shipment from the Company's factory or a warehouse of the Company in the event that the equipment is purchased from the Company or for a period of one year from the date of shipment from the business establishment of an authorized distributor of the Company in the event that the equipment is purchased from an authorized distributor.

THE COMPANY'S OBLIGATION UNDER THIS WARRANTY SHALL BE STRICTLY AND EXCLUSIVELY LIMITED TO REPAIRING OR REPLACING, AT THE FACTORY OR A SERVICE CENTER OF THE COMPANY, ANY SUCH EQUIPMENT OR PARTS THEREOF WHICH AN AUTHORIZED REPRESENTATIVE OF THE COMPANY FINDS TO BE DEFECTIVE IN MATERIAL OR WORKMANSHIP UNDER NORMAL USE AND SERVICE WITHIN SUCH PERIOD OF ONE YEAR. THE COMPANY RESERVES THE RIGHT TO SATISFY SUCH OBLIGATION IN FULL BY REFUNDING THE FULL PURCHASE PRICE OF ANY SUCH DEFECTIVE EQUIPMENT. This warranty does not apply to any equipment which has been tampered with or altered in any way, which has been improperly installed or which has been subject to misuse, neglect or accident.

THE FOREGOING WARRANTY IS IN LIEU OF ANY OTHER WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE, and of any other obligations or liabilities on the part of the Company; and no person is authorized to assume for the Company any other liability with respect to equipment manufactured by the Company. The Company shall have no liability with respect to equipment not of its manufacture. **THE COMPANY SHALL HAVE NO LIABILITY WHATSOEVER IN ANY EVENT FOR PAYMENT OF ANY INCIDENTAL OR CONSEQUENTIAL DAMAGES, INCLUDING, WITHOUT LIMITATION, DAMAGES FOR INJURY TO ANY PERSON OR PROPERTY.**

Written authorization to return any equipment or parts thereof must be obtained from the Company. The Company shall not be responsible for any transportation charges.

IF FOR ANY REASON ANY OF THE FOREGOING PROVISIONS SHALL BE INEFFECTIVE, THE COMPANY'S LIABILITY FOR DAMAGES ARISING OUT OF ITS MANUFACTURE OR SALE OF EQUIPMENT, OR USE THEREOF, WHETHER SUCH LIABILITY IS BASED ON WARRANTY, CONTRACT, NEGLIGENCE, STRICT LIABILITY IN TORT OR OTHERWISE, SHALL NOT IN ANY EVENT EXCEED THE FULL PURCHASE PRICE OF SUCH EQUIPMENT.

Any action against the Company based upon any liability or obligation arising hereunder or under any law applicable to the sale of equipment, or the use thereof, must be commenced within one year after the cause of such action arises.

These products are sold subject to the standard Limitation of Liability and/or Warranty of Superior Electric.

The right to make engineering refinements on all products is reserved. Dimensions and other details are subject to change.

DANA