

ARTISAN®
TECHNOLOGY GROUP

Your **definitive** source for quality pre-owned equipment.

Artisan Technology Group
(217) 352-9330 | sales@artisantg.com | artisantg.com

Full-service, independent repair center
with experienced engineers and technicians on staff.

We buy your excess, underutilized, and idle equipment
along with credit for buybacks and trade-ins.

Custom engineering
so your equipment works exactly as you specify.

- Critical and expedited services
- In stock / Ready-to-ship
- Leasing / Rentals / Demos
- ITAR-certified secure asset solutions

Expert team | Trust guarantee | 100% satisfaction

All trademarks, brand names, and brands appearing herein are the property of their respective owners.

Visit our website - *Click HERE*

Compumotor

System 7 User Guide

RS-232C Control Module and Servo Drive

Compumotor Division
Parker Hannifin Corporation
p/n 88-011270-01 D

Parker

System 7
RS-232C Control Module and Servo Drive User Guide
Compumotor Division of Parker Hannifin
88-011270-01 D
1/8/91

This User Guide applies only to System 7 equipped with RS-232C Control Modules and hybrid brushless or AC brushless servo motors.

For engineering assistance in the U.S.:
Compumotor Division of Parker Hannifin
5500 Business Park Drive
Rohnert Park, CA 94928
Telephone: (800) 358-9070
Fax: (707) 584-8015

For engineering assistance in Europe:
Parker Digiplan Ltd.
21 Balena Close
Poole, Dorset
England, BH17 7DX
Telephone: 0202-690911
Fax: 0202-6000820

User Guide Change Summary

The following is a summary of the changes to this user guide since the last version was released. This user guide, version 88-011270-01**D**, supersedes version 88-011270-01**C**.

When a user guide is updated, the new or changed text is highlighted with a change bar in the right margin (this paragraph is an example). If an entire chapter is changed, the change bar is located to the right of the chapter title.

The entire document is now organized as *Sections* instead of *Chapters*.

Section A. Introduction

- Revised the system functional block diagram

Section B. Getting Started

- Replaced transformer model numbers TO165/TO166 with TO185/TO186
- New bench test configuration diagram
- New RS-232C connection procedures for bench test
- Updated system testing LED information and software commands

Section C. Installation

- Removed warning (*AC Line Cabling*) stating that the transformer cable is inadequate for permanent operation at less than 240VAC — cable fix

Section D. Isolating Transformer

- Replaced transformer model numbers TO165/TO166 with TO185/TO186
- Transformers are factory configured by either 120VAC (single-phase) or 240VAC (three-phase)
- Transformer now comes with input cable that can be used with any input voltage.
- -F option available for TO170/TO171 transformers (in which the AC input fuses are factory-installed)
- Updated transformer connection diagrams

Section E. Power Supply Unit

No technical changes

Section F. Drive Module

- Included description and specs for drive # 7153.501
- Updated tables F1-2 and F1-3 for all possible drive/motor packages

Section G. Hybrid Servo Motor and Encoder

- Included specifications and dimensional drawings for all the possible motors used with this system
- Updated and clarified motor and encoder connection information/drawings
- Included information for using the Encoder Backup Supply

Section H. Control Module

- Added descriptions of new features (encoder following, encoder superposition, synchronous moves, linear interpolation, registration, servo self-tuning, dither facility, and speed change mode)
- Updated fault LED interpretations
- Updated signal descriptions and included step & direction input capability and Encoder Backup Supply
- Corrected and clarified RS-232C and ZDDC connection information and drawings
- Added information for servo self-tuning via the **TUNET** and **TUNEV** commands
- Added *Setting Up Servo Parameters* section

Section J. System Programming

- Added description of drive energize and de-energize functions
- Added illustration showing various types of homing
- Included description of sequence execution via hardware input from a switch or PLC
- Must enter the **SV** command (not the **SS** command) to save the independently saved commands
- Added new command descriptions and changed existing descriptions according to new software versions between version 1.4 and 2.4 (latest version):

  New Commands:

CIW	CONFIGURE INTEGRAL ACTION WINDOW
DTA	SET DITHER AMPLITUDE
DTF	SET DITHER FREQUENCY
FDA	ACCELERATION, LINEAR INTERP. MOVES
FDV	VELOCITY, LINEAR INTERP. MOVES
GL	GO, LINEAR MODE
GSY	GO, SYNCHRONOUS MODE
HSD	DAUGHTER BOARD OUTPUTS
MAS	MODE ASYNCHRONOUS
MSY	MODE SYNCHRONOUS
OSF	INITIALIZATION ON LIMIT
OSG	DITHER ENABLE
OSH	SELECT TRIANGULAR/SQUARE DITHER
OSI	INITIALIZATION MOVEMENT
OSJ	SELECT 24/16 BIT GEARBOX
PSY	PAUSE FOR SYNCHRONIZATION
RE	DRIVE STATUS REQUEST
RSE	REPORT SERVO ERRORS
SIM	SET INDEXER/FOLLOWING MODE
TRR	REGISTRATION MODE
TUNE	DISPLAY TUNING SETTINGS
TUNET	SERVO SELF-TUNING (TORQUE AMP.)
TUNEV	SERVO SELF-TUNING (VELOCITY AMP.)
VSY	VELOCITY CHANGE IN SYNCHRONOUS MODE

Section K. Maintenance & Troubleshooting

- Updated RS-232C diagnostics (software commands and LEDs
- Included sections on communications, drive control, and motor control problems
- Included a section on returning the system for repairs

PREVIOUS CHANGES: 01B → 01C

User Guide Change Summary

The following is a summary of the changes to this user guide since the last version was released. This user guide, version **88-011270-01C**, supersedes version **88-011270-01B**.

When a user guide is updated, the new or changed text is highlighted with a change bar in the right margin (this paragraph is an example). If an entire chapter is changed, the change bar is located to the right of the chapter title.

Chapter 1. Using This User Guide

No changes were made to this chapter.

Chapter 2. Introduction

Changes to Chapter 2 are summarized as follows:

- Added part numbers to the isolating transformers table (Page 6) as an additional reference convenience.
- Added a *What You Should Have* section that advises you compare the packing list with your original order to ensure that you have received a complete shipment.
- Added an *Additional Tools* section that lists the tools (**not provided with the product**) that you may need to complete the procedures in this user guide

Chapter 3. Getting Started

In the *Turning the System On* section (Page 10) , the command string was altered as follows:

LD3 STØ MN A10 V1Ø DØØØØ G

This change will ensure that the test move will be completed properly.

Chapters 4 - 11

No changes were made to these chapters.

Table of Contents

Table of Contents i
List of Figures iv
List of Tables v

Section A. Introduction A1-1
Organization of the User Guide A1-1
Warnings and Cautions A1-1
Conventions and Definitions A1-2
Motor Rotation A1-2
Encoder A1-2
Direction Signal A1-2
Step Signal A1-2
PWM Torque Signal A1-2
Rack Dimensions A1-2
System 7 General Description A1-3
Functional Block Diagram A1-3
Control Module A1-4
Drive Module A1-4
Power Supply A1-5
Isolating Transformer A1-5
What You Should Have A1-5
Additional Tools A1-5

Section B. Getting Started B1-1
Test Requirements B1-1
Cables B1-1
Bench Test Motor Mounting B1-1
Bench Test Configuration B1-1
Motor/Encoder B1-2
Estop Input B1-2
Serial Communication Input B1-2
Isolating Transformer B1-3
System Testing B1-4
System Parameters B1-4
Turning the System On B1-4
Functional Test B1-4
Summary B1-4

Section C. Installing the System C1-1
Installation Precautions C1-1
Mounting the Rack C1-1
Ventilation C1-2
Noxious Vapors C1-2
Accessibility C1-2
Installing the Modules C1-2
Installing the Motor C1-3
Mounting the Motor C1-3
Heatsinking C1-3
Coupling to the Motor C1-3
Installing the Transformer C1-4
Installing the Wiring C1-4
Cabling C1-4
Cable Routing C1-4
Grounding C1-4
Setting up the System C1-5
Summary C1-5

Section D. Isolating Transformer D-1
Transformer Description D-1
Single-Phase Supply D-1
Three-Phase Supply D-3

Section E. Power Supply Unit **E-1**
Description E-1
Specifications E-1
Fuse Ratings E-2
Installation E-3
Connections E-3
HV Supply Indicator E-3
Principles of Operation E-4
Summary E-5

Section F. Drive Module **F1-1**
Description F1-1
Principles of Operation F1-1
Specifications F1-3
Fuse Rating F1-4
Installation F1-4
Setting Up the Current Limit F1-5
Fault Indicators F1-6
Logic Supply On F1-6
Control Fault F1-6
Overload F1-6
HV Over-voltage F1-6
Over-temperature F1-6
Summary F1-6

Section G. Hybrid Servo Motor and Encoder **G1-1**
Description G1-1
Motor Specifications G1-1
Motor Dimensional Drawings G1-2
Connections G1-4
Cabling G1-4
Encoder Backup Supply G1-5
Couplings G1-6
Summary G1-6

Section H. RS-232C Control Module **H1-1**
Description H1-1
Features H1-1
Fault LED Indicator H1-2
Specifications H1-2
Motion Profiles H1-3
Operating Modes H1-4
Normal (Preset) Modes H1-4
Continuous Mode H1-5
Speed Change Mode H1-6
Registration mode H1-6
Asynchronous Mode H1-7
Synchronous Mode H1-7
Linear interpolation H1-7
Encoder following Mode H1-9
Dither H1-10
Motion on initialization H1-10
Program Storage H1-10
Motion Program Selection H1-11
Motherboard Terminal Functions H1-11
Signal Types H1-12
Signal Descriptions H1-13
External Connections H1-16
Step and Direction Inputs H1-16
Positive and Negative End-of-Travel Limit Inputs H1-16
The Home Limit Input H1-17
Trigger Inputs H1-17
± Jog Inputs H1-17
ESTOP Input H1-18

Sample Wiring Schemes H1-18
Coupling Outputs to Inputs H1-18
Fault Indicator H1-18
In-Position Indicator H1-19
Setting Up Terminal Communication H1-19
Communication Parameters H1-19
Basic RS-232C Connections H1-20
Serial Interface Status H1-21
Zero Delay Daisy Chain (ZDDC) H1-21
Control Module Address Jumpers H1-22
Servo Tuning H1-23
Tuning Parameters H1-23
Safety Precautions H1-24
Setting Up Servo Parameters H1-24
Manual Tuning Procedure H1-27
Notes on Tuning H1-29
Servo Self-Tuning H1-30

Section J. Software Reference J1-1
Communicating with the RS-232C Control Module J1-1
Buffered/Immediate Commands J1-1
Buffer Capacity J1-1
Individual Commands J1-1
Buffered Commands J1-1
Immediate Commands J1-2
Multiple Control Modules J1-2
Start Move Commands J1-3
Loop Commands J1-4
Stop Move Commands J1-4
Status Request Commands J1-6
Drive Energise/De-energise Function J1-6
Home Facilities J1-6
Operating Modes J1-7
Basic Programming Guide J1-8
Sequence Programming J1-10
Programming a Sequence J1-10
Running Sequences J1-11
Changing a Programmed Sequence J1-12
Checking the Sequence's Contents J1-12
Sequence Status Requests J1-12
Command Structure J1-13
Command List J1-15
Command Summary J1-86
Alphabetical Command Listing J1-89

Section K. Maintenance & Troubleshooting K1-1
Maintenance K1-1
Filters K1-1
Cables K1-1
Connectors K1-1
Fault Diagnosis K1-1
Problem Isolation K1-2
Power Supply and Drive Fault LEDs K1-2
Control Module Fault LED K1-3
Electrical Noise K1-4
Communications Problems K1-5
Drive Control Problems K1-5
Motor Control Problems K1-5
Returning The System K1-6
Summary K1-6

Appendices APP-1
Warranty APP-1
Index APP-3

List of Figures

Figure A1-1. System 7 Functional Block Diagram A1-3

Figure B1-1. Bench Test Configuration B1-2

Figure C1-1. Rack Dimensions C1-1
Figure C1-2. Module Positions in the Rack C1-2

Figure D-1. Single-Phase Transformer Mounting Details D-2
Figure D-2. 120V Single-Phase Transformer Connections D-3
Figure D-3. Three-Phase Transformer Mounting Details D-4
Figure D-4. 240V Three-Phase TO185 Connections D-5

Figure E-1. Fuse Location E-2
Figure E-2. Power Supply Circuit Diagram E-4
Figure E-3. Power Supply Motherboard Circuit Diagram E-5

Figure F1-1. Drive Block Diagram F1-2
Figure F1-2. Waveform Generator F1-3
Figure F1-3. Drive Current Limit Switch F1-5

Figure G1-1. 7521.350 Motor Dimensions (7252.500 & 7252.501 Motor/Drive Pkgs.) G1-2
Figure G1-2. 7521.250 Motor Dimensions (7252.520 Motor/Drive Package) G1-2
Figure G1-3. 7521.370 Motor Dimensions (7252.502 Motor/Drive Package) G1-3
Figure G1-4. 7501.350 Motor Dimensions (7252.300 Motor/Drive Package) G1-3
Figure G1-5. Motor and Encoder Connections G1-4
Figure G1-6. Encoder Backup Supply Connections G1-5

Figure H1-1. Triangular Profile H1-3
Figure H1-2. Trapezoidal Profile H1-3
Figure H1-3. Speed Change Mode Example Profile H1-6
Figure H1-4. Example Registration Move H1-6
Figure H1-5. Linear interpolation mode move profiles H1-8
Figure H1-6. Linear Interpolation X/Y table plot H1-8
Figure H1-7. Motherboard Terminal Functions H1-11
Figure H1-8. AN-OUT 1 Graph H1-14
Figure H1-9. AN-OUT 2 graph H1-15
Figure H1-10. Inputs and Outputs—General Arrangement H1-16
Figure H1-11. Step and Direction Input Connections H1-16
Figure H1-12. Limit and Home Switch Connections H1-17
Figure H1-13. Trigger and Jog Switch Connections H1-17
Figure H1-14. ESTOP Input Connection H1-18
Figure H1-15. Input-to-Output Coupling H1-18
Figure H1-16. Composite Fault Indicator H1-18
Figure H1-17. In-Position Indicator H1-19
Figure H1-18. RS-232C Permanent Connection H1-20
Figure H1-19. Temporary RS-232C Connection H1-20
Figure H1-20. Zero Delay Daisy Chain Connections H1-21
Figure H1-21. Control Module Address Jumper Location H1-22
Figure H1-22. Servo Control Loop Block Diagram H1-23
Figure H1-23. Servo Response Curve Example H1-26
Figure H1-24. Servo System Bandwidth H1-27
Figure H1-25. Overshoot During Positioning H1-29

Figure J1-1. Home Switch Operation J1-7
Figure J1-2. System Used in Example Programs J1-8
Figure J1-3. Example 2 Moves Diagram J1-9
Figure J1-4. Sequence Definition & Termination Example J1-10
Figure J1-5. Final Positioning J1-18
Figure J1-6. Integral Action J1-21
Figure J1-7. Servo Control Loop J1-53

List of Tables

Table A1-1. Isolation Transformers A1-5

Table B1-1. Fuse Ratings & Types B1-3

Table D-1. Fuse Ratings & Types for Single-Phase D-1
Table D-2. Single-Phase Transformer Specifications D-1
Table D-3. Single-Phase Transformer Dimensions D-2
Table D-4. Single-Phase Transformer Jumpers D-2
Table D-5. Fuse Ratings & Types for Three-Phase D-3
Table D-6. Three-Phase Transformer Specifications D-4
Table D-7. Three-Phase Transformer Dimensions D-4
Table D-8. TO185 & TO186 Transformer Primary Connections D-5
Table D-9. TO185 and TO186 Rack Connections & Jumpers D-5

Table E-1. Power Supply Specifications E-1
Table E-2. Location & Ratings E-2

Table F1-1. Specifications for Motor/Drive Packages F1-3
Table F1-2. DIP Switch Settings for F1-5

Table G1-1. Motor Specifications G1-1

Table H1-1. RS-232C Control Module Specifications H1-2
Table H1-2. Control Module Signal Types H1-12
Table H1-3. Command Input Connector Functions H1-21
Table H1-4. Control Module Address Settings H1-22
Table H1-5. HELP11 - Servo Setup Table H1-25
Table H1-6. DPE Readings During Final Positioning H1-29

Table K1-1. Power Supply and Drive Module Fault Conditions K1-3
Table K1-2. Control Module Fault Conditions K1-4

Section A. Introduction

Section Objective

Read this section to familiarize yourself with the organization of this user guide and with the product.

Organization of the User Guide

This user guide tells you how to configure and use the brushless servo drive systems. We have made it as easy as possible for you to check that the system is functioning properly. Once you have ensured that the system functions properly, you can go on to learn how to set up the drive and program the system in more detail.

Section A, *Introduction*, identifies the organization of this user guide. It tells you how the sections are arranged and describes the conventions we use. Section A also introduces System 7. It briefly describes the basic elements as an aid to users who have no previous knowledge of the system.

Section B is called *Getting Started*. It tells you the connections you will need to make before you turn on the power, and also how to check that the system is working by sending a simple command.

When you are sure that the system is working, you are ready to install it into your equipment. Section C lists some of the things you should consider before mounting the motor and system rack. It also gives recommendations on cable routing and grounding.

Sections D - H give details of the modules used in the system. You will need to refer to these, particularly Sections F and H, so that you can learn how to set up the drive and control card parameters and how to program the control card to suit your application.

Section J provides all of the information required to program the system, including a detailed list of commands and a quick reference list.

If the system develops a fault, or does not work when you configure it as described in Section B, refer to Section K. This chapter is called *Maintenance and Troubleshooting*. It will help you establish the source of the problem and assist you in deciding on the action you should take.

Warnings and Cautions

Please read the warnings and cautions carefully before you go on to a subsequent section. They are important for the safety of yourself and the equipment.

Conventions and Definitions

The following list gives some of the conventions and definitions that are used in this user guide.

Motor Rotation

The direction of rotation is defined with the motor viewed from the mounting flange end. Phase A leads Phase B for CW rotation.

Encoder

Channel A leads channel B for CW rotation of the motor.

Direction Signal

Open circuit inputs cause CW rotation of the motor.
DIRECTION+ greater than DIRECTION- will cause CW rotation.
DIRECTION- greater than DIRECTION+ will cause CCW rotation.

Step Signal

STEP+ greater than STEP- causes the motor to rotate by one step.

PWM Torque Signal

This is a pulse width modulated signal from the control card to the drive. The signal comprises two signals: *Mark* (the period of time when the pulse is on) and *Space* (the period when the pulse is off). When the mark-to-space ratio is equal to one, the torque demand is zero. When mark (PWM+) is greater than space (PWM-) this defines a CCW generated torque.

Rack Dimensions

The dimensions of this rack system are expressed in metric units—U and HP. To convert these units to inches, use the table below.

Metric Unit	Equivalent In Inches
1 HP	1.75 inches
1 U	0.20 inches

System 7 General Description

System 7 is a motion control system that uses standardized construction and communication techniques to control the full range of motor types. There are several versions of the System 7 that can incorporate various *drive modules* (open- and closed-loop steppers, microsteppers, and DC brush and AC brushless servos) and *control modules* (an RS-232C module, an IEEE-488 module, a Step & Direction module, and an Analog module).

This user guide relates to systems that include hybrid or AC brushless servo drives controlled by an RS-232C control module. Separate user guides are available for other System 7 configurations.

Functional Block Diagram

Figure A1-1 shows a diagram showing the functional interrelationships of the System 7 components.

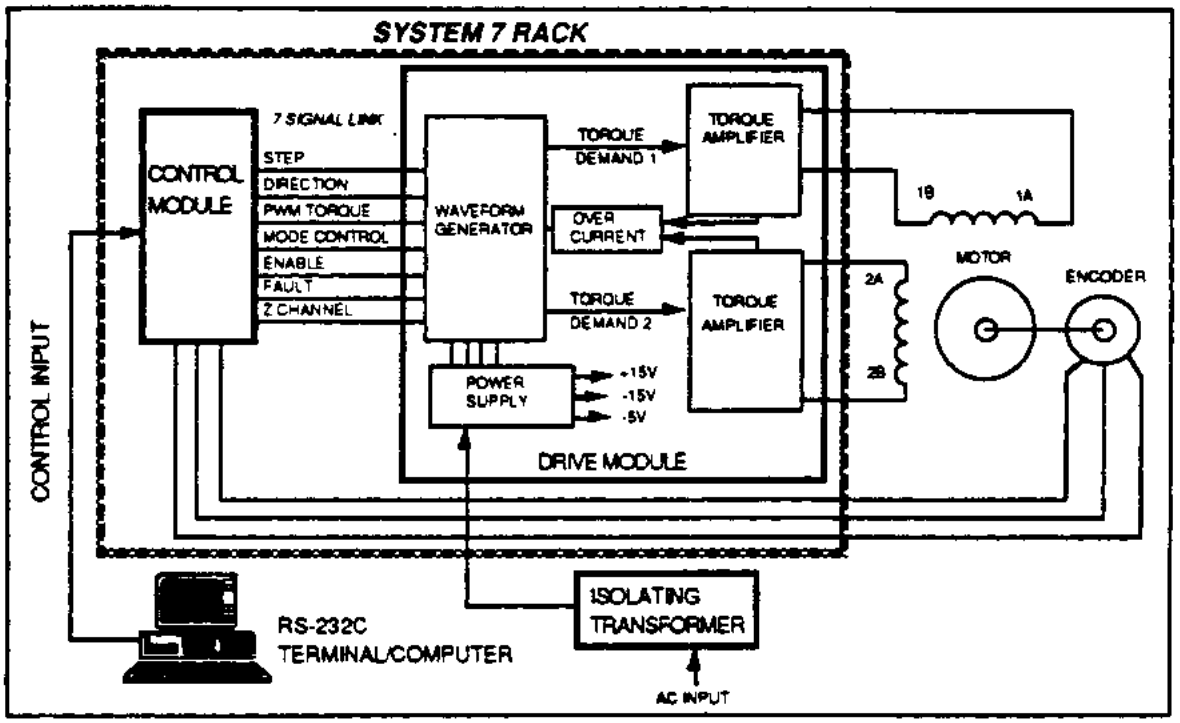


Figure A1-1. System 7 Functional Block Diagram

Control Module

This user guide is based on the System 7 using the RS-232C control module (see Figure A1-1).

To control the motor, the RS-232C control module accepts X Series commands (described in Section J) from a terminal or computer via an RS-232C link . It connects to the drive module via the standard 7-signal link to control any type of servo motor.

Drive Module

The drive consists of a dual torque amplifier to supply controlled current to the motor, a waveform generator (see Figure A1-1), and a switched mode power supply. Its integral power supply runs from the motor supply bus and provides power for the on-board circuitry and the isolated power supplies for a control card, encoder, and external switches.

7 Signal Link

The control module communicates with the drive module over the 7-signal link. The signals used depend upon the type of motor that you use. A hybrid or AC brushless system uses all 7 signals, an open-loop stepper uses STEP, DIRECTION, ENABLE, and FAULT, and a brush DC servo (*this capability is under development*) uses ENABLE, FAULT and PWM TORQUE.

These 7 signals, defined below, are fully opto-isolated at the drive end to eliminate the problems arising from electrical noise.

STEP AND DIRECTION

Step and direction, the usual inputs to a stepper drive, are also used with brushless servo drives. However, in this case, they are derived from the encoder and are used only for commutation.

ENABLE

This signal enables and disables the motor. With a servo drive, this signal can disable the motor if a controller fault occurs. It also resets a fault condition.

FAULT

This line is a signal from the drive to indicate a hardware, power supply, or temperature problem to the controller.

PWM TORQUE

The most significant of the seven signals is the PWM (pulse-width modulated) Torque signal, which consists of pulses at a constant frequency. The width of the pulses are related to the torque demand. As the torque demand changes, the mark-to-space ratio will change. A 5% mark-to-space ratio gives maximum negative torque and a 95% mark-to-space ratio gives maximum positive torque. Zero torque is produced with a 50% mark-to-space ratio. The motor current, accurately controlled by the PWM signal, is output by the torque amplifiers in the drive.

MODE CONTROL

This signal is specific to hybrid and AC brushless servo drives using encoder feedback. It is used in power-up initialization sequences.

Z CHANNEL

The position of the motor shaft is monitored by the encoder, which is housed within an enclosure at the end of the motor. The encoder's Z channel signal, an index pulse that occurs once per revolution, is processed in the control module to track rotor position.

Power Supply

Power for the control module and its associated input circuits is supplied by the power supply within the drive module, but the input to this module (150VDC) is provided by the rack power supply from a 107VAC source.

Isolating Transformer

Four types of isolating transformers can be supplied: two for single-phase operation and two for three-phase operation (see Table A1-1).

Transformer (Part #)	Type & Rating
TO170 (7451.330)	Single-phase 1.3 KVA
TO171 (7452.530)	Single-phase 2.5 KVA
TO185 (7452.570)	Three-phase 2.5 KVA
TO186 (7455.070)	Three-phase 5 KVA

Table A1-1. Isolation Transformers

Details of these transformers are given in Section D, *Isolating Transformer*.

What You Should Have

Before you proceed with Section B, *Getting Started*, review the packing list that was included with your System 7 shipment. If possible, compare the packing list with your original order to ensure that you have received all of the components necessary for your application.

Additional Tools

To complete some of the steps and procedures outlined in this user guide, you will need the following tools (not provided with your shipment):

- Phillips screwdriver
- Slot screwdriver
- Nut driver (or adjustable wrench)

Section B. Getting Started

Section Objective

This section tells you how to set up your system and check that it is working correctly. It contains all the information you will need to configure the system and to complete a series of functionality tests.

Test Requirements

Before you install the system into your equipment, particularly if you are not familiar with Compumotor equipment, we recommend that you complete a bench test to ensure that the system operates correctly. By following the simple procedures below, you should be able to have your system running within a short time.

WARNING

High voltages may be accessible at various points on the equipment. Ensure that you protect these areas so as to avoid inadvertent contact with the high-voltage bus.

Cables

Your system has all the cables required to connect the system modules. The motor and encoder cables will have been supplied to the length you specified. If you want to make your own cables, approved cable types and plug connection information is in Section G1 of this user guide.

Isolating transformer cables for connection between the transformer and the rack are supplied with the system. If you must extend these cables, use a gauge of wire that is the same as the supplied cable. Compumotor can supply complete cables.

Bench Test Motor Mounting

When you bench test your system, the motor may move on the bench unless it is restrained. The best way to secure the motor is to mount it on a bracket or in a cradle that is fixed to the bench.

Bench Test Configuration

Your system was pre-assembled and tested at Compumotor. If you reassemble the system as instructed, you should have little difficulty in getting it running. The rack is supplied ready-wired, with drive, control, and power supply modules already installed. When you have made the connections detailed below, you will be ready to power up the system. Figure B1-1 illustrates the required bench test configuration.

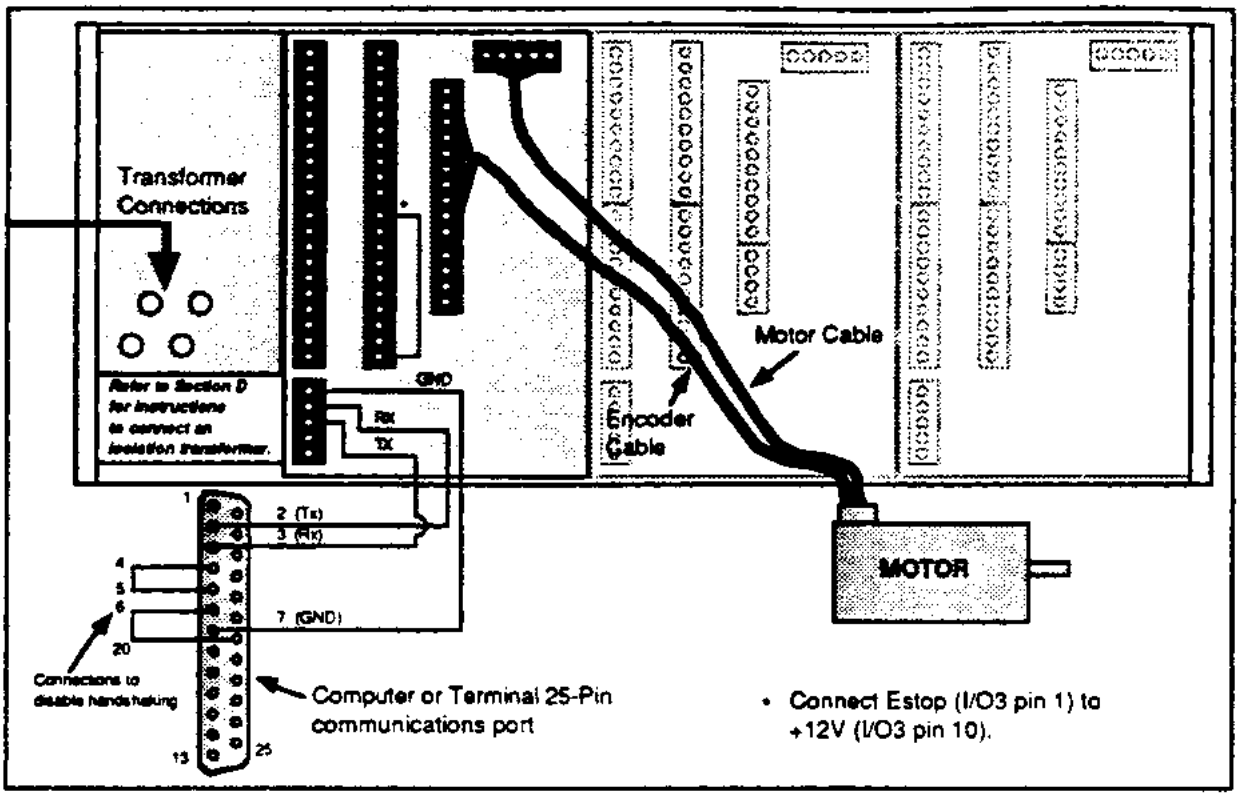


Figure B1-1. Bench Test Configuration

Motor/Encoder

The motor/encoder cable supplied with the system is pre-wired to the motor and encoder. For test purposes, plug the encoder cable plug into the **MOTOR FEEDBACK** connector on the appropriate drive/control motherboard and connect the motor cable to the **MOTOR** connector (see Figure B1-1).

Estop Input

You must physically connect the ESTOP input (I/O 3, pin 1) to +12V (I/O 3 pin 10) in order for this bench test configuration to operate (see Figure B1-1). ***The Estop input should be wired directly to +12V for the bench test only. In the final configuration, you must use a switch for this connection.***

Serial Communication Input

For the purpose of the temporary bench test configuration, connect your computer terminal to the **GND**, **RX**, and **TX** lines on the **SERIAL INPUT** connector. This connector is located on the back of the drive module (see Figure B1-1). You may obtain a standard 25-pin cable from Compumotor (part number 7967.100). If the front panel fault LED illuminates during the system bench test, the RS-232C link allows you to use the status commands (see Section J) to determine the cause of the fault.

After you permanently install the System 7 rack in an enclosure such as a rack cabinet, accessing the **SERIAL INPUT** connector (for testing and tuning purposes) is very difficult. Therefore, an alternative serial connector on the front of the control module board is accessible by removing the control module front panel. Refer to Section H for more information on alternative serial connections.

Communication Parameters

To communicate with the control module, the terminal or computer must be set to the communication parameters below. Refer to the terminal's operator handbook for information on how to do this.

- Baud Rate: 9,600 baud
- Data Bits: 8 (no parity)
- Start Bit: 1
- Stop Bit: 1

The control module does not accept handshaking in any form. Therefore, you should disable the handshaking via software or by jumpering RTS to CTS (usually pins 4 and 5) and DSR to DTR (usually pins 6 and 20) as shown in Figure B1-1.

Isolating Transformer

If you supply your own isolating transformer, it must be suitable for the AC line supply you intend to use. The secondary transformer must be wound to provide an output of 107V RMS **maximum** and should be rated to provide the necessary current (see specifications table in Section D).

If you are using a Compumotor transformer, its tappings will be jumpered for a 240V three-phase or a 120V single-phase AC supply, as you specified in your order. **Refer to Section D for details on Compumotor isolating transformers.**

Primary Fusing

When connecting the transformer to the AC line, you should protect it by installing an in-line fuse (refer to Table B1-1 for proper fuses for the various transformers).

Transformer (Part #)	Fuse Rating	Fuse Type	Voltage
TO170 (7451.330)	15A	TD	250V
TO171 (7452.530)	25A	TD	250V
TO185 (7452.570)	15A	TD	250V
TO186 (7455.070)	25A	TD	250V

Table B1-1. Fuse Ratings & Types

System Testing

Now that you have made all the necessary connections, the system should be functional. **Before you apply power you should make one final check to ensure that you have made all the connections correctly.**

System Parameters

A set of servo parameters have been programmed into the system to allow it to be used in a wide range of applications. These are the *default parameters*. Once your system is running you may need to tune it for optimum performance by changing these parameters. You will find information on how to do this in Section H, *RS-232C Control Module*.

Turning the System On

Apply AC power to the system (see Section D, *Isolating Transformer* for more information). If you connected the system properly, the green LED on the power supply front panel should illuminate, as should the green *Logic Supply* LED on the drive front panel. All other LEDs on these two modules should remain off. If any red LEDs (other than the control card LED) illuminate, refer to Section K, *Maintenance & Troubleshooting*, for instructions.

Functional Test

The System 7 is set prior to shipment to power up with the limits enabled (`LDØ`) and the drive disabled (`ST1`). This is done for safety considerations. Therefore, before proceeding, you must enable the drive by entering `STØ` `[cr]` and save this parameter by entering `SV` `[cr]`. The control card LED should now be off.

Enter the command string shown below. Put a space between each command and press the ENTER key after the Go (G) command.

```
LD3 MN A1Ø V1Ø D8ØØØ G
```

The motor should turn 2 revolutions CW (*4 revolutions for the 2340*). If the motor does not respond as described, switch off power and recheck your wiring. If the wiring is correct and the motor still does not respond, refer to Section K, *Maintenance and Troubleshooting*. The successful completion of this move indicates the following:

- The RS-232C connection is functioning properly
- The drive, motor/encoder, and control module are functioning properly
- The initial system parameters are properly set for the purposes of this *Getting Started* Exercise

Summary

When you are satisfied that the system is functioning properly, read the next section, which explains some items you need to consider.

Before dismantling your test set up, familiarize yourself with parameter adjustment and programming. You will find detailed information on these functions in Sections F and J.

Section C. Installing the System

Section Objective

This section discusses some of the things you must consider when you install the system. If you do not follow these guidelines, the system may operate unreliably and/or suffer damage.

Installation Precautions

The following precautions apply to the installation of most electronic equipment.

- Handle all units with care.
- Install System 7 in a clean, dry, well-ventilated area.
- Do not mount units on a hot surface or a heat source.
- Circulating air must be free from steam, chemical substances, combustible vapors, and oil.

Mounting the Rack

The rack is a standard 19" rack. You may mount it using any of the standard techniques.

- Open frame support
- Large cabinet containing other equipment
- Individual cabinet

Figure C1-1 shows the rack's dimensions. Before you mount the unit, examine the conditions under which the equipment will operate.

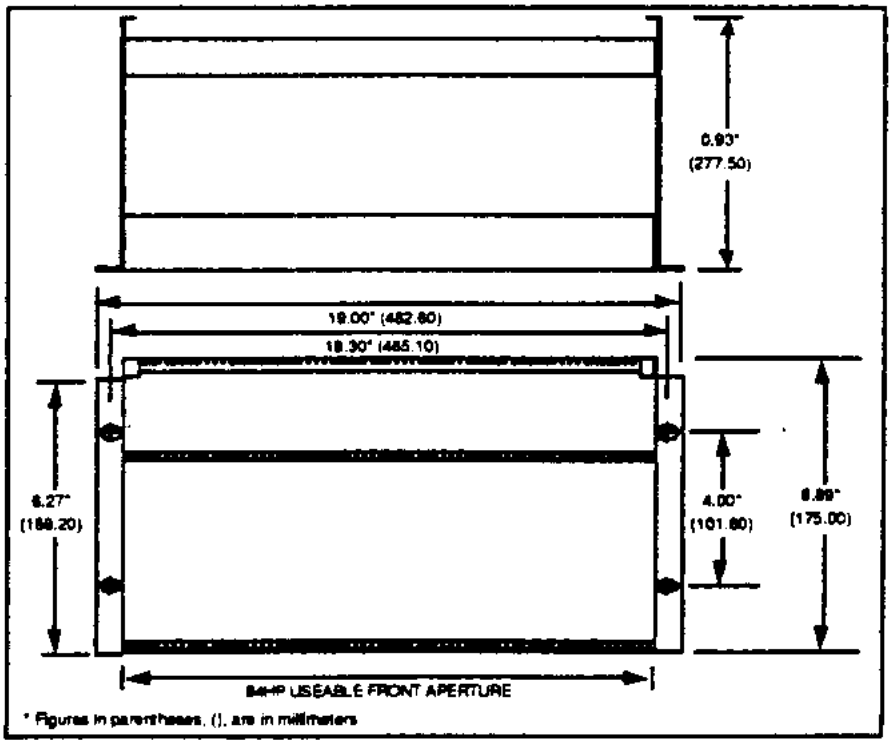


Figure C1-1. Rack Dimensions

Ventilation

The equipment is rated assuming a free flow of air at a maximum temperature of 104°F (40°C). When you mount the rack, you should ensure that the ventilation is adequate. Take particular care, if you are mounting the rack in a cabinet with other equipment, to ensure an adequate flow of cool air.

Noxious Vapors

Some industrial environments contain noxious chemical vapors and oils that can cause serious damage to electronic components. If your equipment will be installed in an industrial environment, you may have to enclose the rack in a sealed cabinet and ventilate the unit with special filters.

Accessibility

After you install the system, you may need to access the equipment from the front and back of the rack to complete modifications or repairs. You should consider this when you decide on the most suitable form of rack mounting.

Installing the Modules

The drive, power supply, and control modules are normally pre-mounted in the rack and plugged into their own motherboards, which are attached to the rack and pre-wired.

To remove a module, you should first unscrew the retaining screws located at the top and bottom of the front panel (drive modules have four screws and control modules have two screws). Once the screws are removed, slide the module out. Figure C1-2 is a front view of the rack showing the module positions. To install a module, simply reverse the removal procedure. Be sure to insert the module in the module guide.

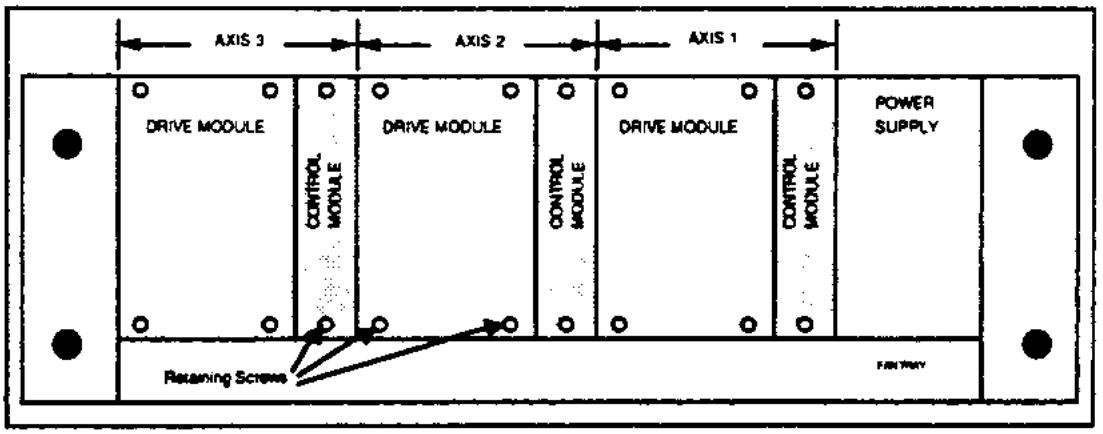


Figure C1-2. Module Positions in the Rack

Installing the Motor

The considerations that apply to handling and installing electronics also apply to motors. Some motor shafts are made from stainless steel and can easily be damaged if knocked or if the motor is dropped on its shaft.

Mounting the Motor

Servo motors and stepper motors are manufactured with a *pilot* machined on their mounting face. This pilot is closely toleranced in diameter and concentricity with the shaft. You should use this feature to locate the motor onto your equipment.

To attach the motor you should use all the bolt holes provided, installing the maximum possible diameter of screw. Mechanical details of the motor are in Section G, *Hybrid Servo Motor and Encoder.*

Heatsinking

The motor you are using will have a particular rating that depends on its operating case temperature. Normally, your equipment will be cooler than the motor and will provide some heat sinking. If your equipment is hot, however, it may heat the motor and reduce its rating.

Coupling to the Motor

Stepper motors should be mounted using flange bolts and centered by the pilot on the front face. Foot-mount configurations are a less desirable alternative because the torque of the motor is not evenly distributed around the motor case. Any radial load on the motor shaft is multiplied by a much longer lever arm when a foot mount is used rather than a face flange.

WARNING

Improper mounting can compromise system performance and jeopardize personal safety.

The motors used with the System 7 can produce very large torques. These motors can also produce high accelerations. This combination can shear shafts and mounting hardware if the mounting is not sufficient. High accelerations can produce shocks and vibrations that require much heavier hardware than would be expected for static loads of the same magnitude. The motor, under certain profiles, can produce low-frequency vibrations in the mounting structure. These vibrations can also cause metal fatigue in structural members if harmonic resonances are induced by the move profiles you are using. A mechanical engineer should check the machine design to ensure that the mounting structure is adequate.

CAUTION

Do not attach the load to the motor yet. Section G discusses methods for coupling the load to the motor.

Installing the Transformer

The transformer is a very heavy unit. It should be installed in the bottom of a large cabinet or rack. If you wish to separate the transformer from the rest of the system, you must ensure that high-voltage terminals are not exposed (covers are provided with Compumotor transformers). Section D, *Isolating Transformer* contains the mechanical details of the transformer.

Installing the Wiring

Your system is supplied with the cables cut to the lengths you specified in your order. Installation involves routing the cables in or from the control cabinet to the motor. Where cables are likely to be exposed in a *machine shop*, we suggest that you route them through a metal conduit.

Cabling

If you supply your own cables, we recommend certain types of cable for particular applications. For further information on these, refer to Section G.

Cable Routing

Where necessary, recommended cables are shielded to minimize problems caused by interference from external electrical sources and adjacent wiring.

To eliminate wiring problems, keep low-level signal cables, such as encoder cables, away from cables carrying high-level control signals or motor current. If possible, route high- and low-level signal cables in separate conduits. You should not encounter problems if you run a motor lead and its encoder cable in the same conduit, provided they are separately shielded.

Grounding

All cable shielding and other ground connections should be taken to one point in the control cabinet. You should leave the remote ends of shields non-terminated. This is to eliminate *ground loops* within the system wiring. If such loops exist, currents may circulate in them and these may cause unreliable or unpredictable performance.

Setting up the System

If you bench tested your system (as suggested in Section B) before mounting it into your equipment, you should be familiar with its operation. Having installed the system, you should be able to test your complete system. To test the system, you should perform a move. Be sure that the move can be completed with the mechanics of your system.

CAUTION

Use the Disable Limits (LD3) command carefully. The command disables limit switch operation. This may be convenient for system testing, but movement that exceeds the limit switch positions can damage your equipment.

If you need to modify the performance parameters of the system to meet the application's specifications, refer to Section F, *Drive Module*, and Section H, *RS-232C Control Module*. These sections provide information for you to customize the drive and control modules for your application.

Summary

Now that you have installed and set up your system, your equipment should function as required. Sections D through H describe the various modules/components in detail. If you have problems, refer to the Section K, *Maintenance and Troubleshooting*.

Section D. Isolating Transformer

Section Objective

This section gives you details of the isolating transformer. You will need to refer to it if you want to set up the voltage tappings on the isolating transformer.

Transformer Description

Depending on your system load requirements or on the available AC supply, you will have been supplied with either a single- or a three-phase transformer. This will have had its tappings jumpered for either 120V single-phase (see Figure D-2) or 240V three-phase (see Figure D-4) and for the appropriate output voltage (normally 107V).

It is our convention to quote transformer secondary voltages in the open circuit condition. To use the system on a different AC supply, or to obtain a different output voltage, you must adjust the transformer's jumpers. This section shows you how to do this. ***If you adjust the jumpers, test that 107VAC is obtained on the cables before you connect to them to the System 7.***

Single-Phase Supply

There are two types of single-phase transformers available: TO170 and TO171.

If you ordered the transformer with the -F option, the AC input fuse will be factory-installed on the transformer. If you intend to install the AC in-line fuse yourself, use the fuse supplied in the ship kit (see Table D-1 for specifications). Refer to Figure D-2 for fuse placement.

Transformer (Part #)	Fuse Rating	Fuse Type	Voltage
TO170 (7451.330)	15A	TD	250V
TO171 (7452.530)	25A	TD	250V

Table D-1. Fuse Ratings & Types for Single-Phase

The single-phase transformer specifications are presented in Table D-2.

Parameter	Specification
Primary 1, 2, 3, 4	0V, 20V, 110V, and 120V respectively (rms)
Secondary 1	TO170: 0 - 107V (rms) at 11A TO171: 0 - 107V (rms) at 22A
Secondary 2	0V - 115V (rms) at 1A
Regulation	Better than 3%
VA Rating	TO170: 1.3 KVA TO171: 2.5 KVA
Weight	TO170: 34.2 lbs [15.5 kgs] TO171: 54.0 lbs [24.5 kgs]

Table D-2. Single-Phase Transformer Specifications

Single-Phase Transformer Dimensions

Table D-3 contains the critical dimensions of the two single-phase transformers.

	TO170	TO171
Overall Height	8.0" (203)	9.5" (242)
Overall Width	5.3" (135)	6.8" (172)
Overall Length	7.7" (195)	8.5" (215)

Table D-3. Single-Phase Transformer Dimensions

Refer to Figure D-1 for mounting details

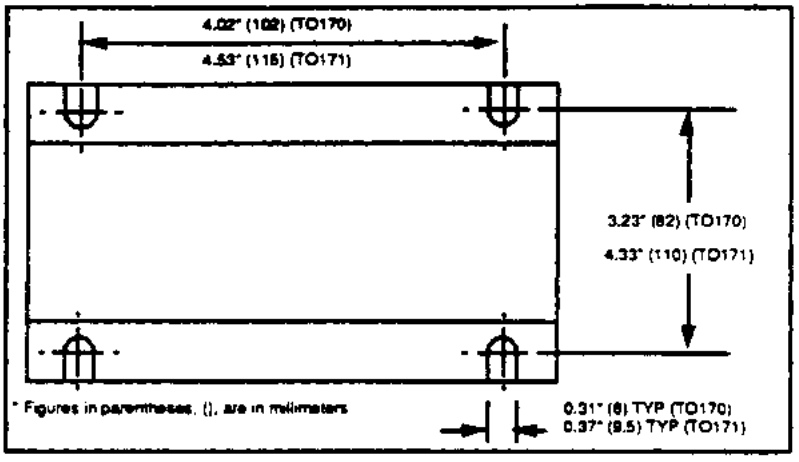


Figure D-1. Single-Phase Transformer Mounting Details

Voltage Adjustment

Figure D-2 shows the transformer connections for 120V (rms) input. If you must use other input voltages, Table D-4 gives details of the terminals you will need to jumper.

The input voltages in the range 360V to 480V are available for connecting the transformer across two phases of a three-phase supply.

Input Voltage	Primary Connections Line 1	Line 2	Terminals (Jumpering)
100	1	9	1-2-3-4, 9-10--11-12
110	5	13	5-6-7-8, 13-14-15-16
120*	1	13	1-2-3-4, 13-14-15-16
200	1	10	9-2, 11-4, 1-3,10-12
220	5	14	13-6, 15-8, 5-7, 14-16
230	1	14	13-6, 15-8, 14-16
240	1	14	13-2, 15-4, 14-16
360	5	12	9-6, 10-7, 11-8
380	5	16	9-6, 10-7, 11-8
400	1	12	9-2, 10-3, 11-4
420	1	16	9-2, 10-3, 11-4
440	5	16	13-6, 14-7, 15-8
460	5	16	13-6, 14-3, 15-4
480	1	16	13-2, 14-3, 15-4

*This is the default setting for TO171 & TO171 transformers

Table D-4. Single-Phase Transformer Jumpers

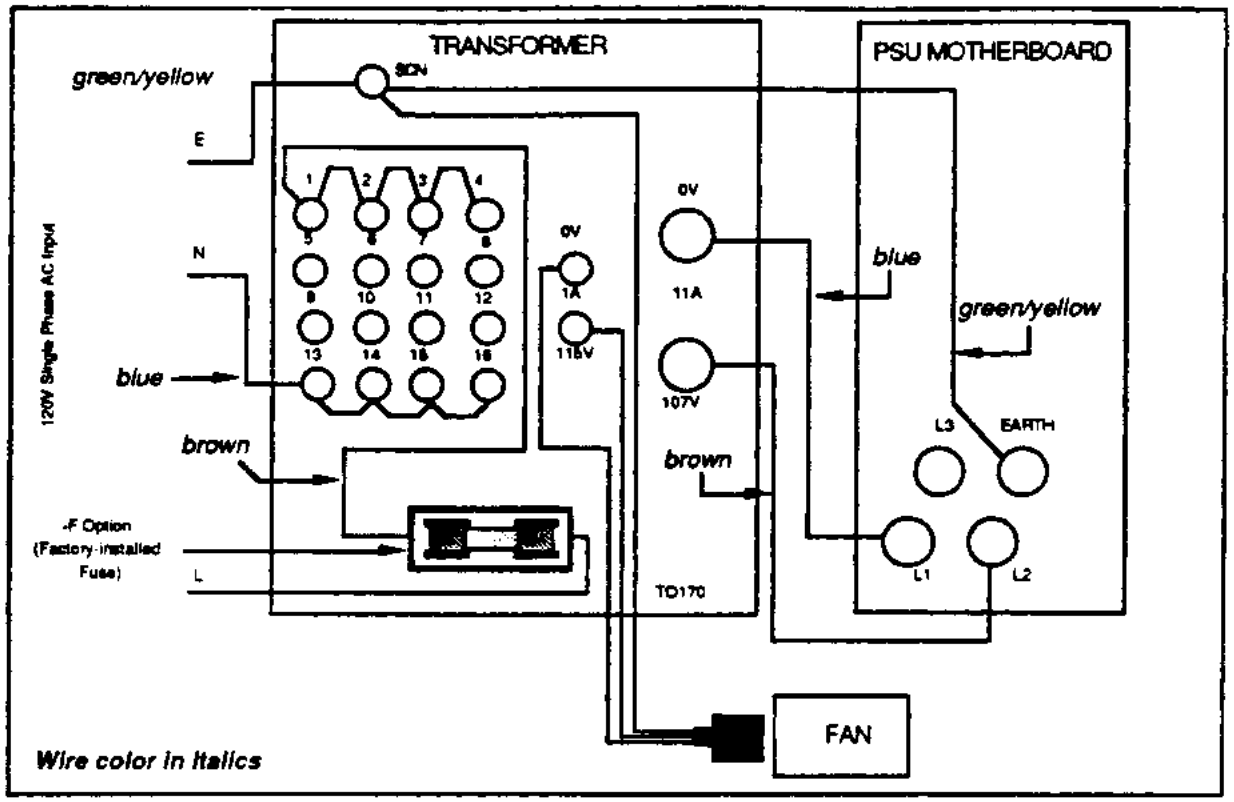


Figure D-2. 120V Single-Phase Transformer Connections

Three-Phase Supply

Two types of three-phase transformers are available: TO185 and TO186. The electrical and mechanical details of these are given in Table D-5.

Transformer (Part #)	Fuse Rating	Fuse Type	Voltage
TO185 (7452.570)	15A	TD	250V
TO186 (7455.070)	25A	TD	250V

Table D-5. Fuse Ratings & Types for Three-Phase

The three-phase transformer specifications are presented in Table D-6.

Parameter	Specification
Phase 1, Primary 1,2	20 - 0 - 180 - 220V (rms)
Phase 2, Primary 3,4	20 - 0 - 180 - 220V (rms)
Phase 3, Primary 5,6	20 - 0 - 180 - 220V (rms)
Phase 1, Secondary 1	TO185: 14 - 0 - 71 - 99 - 127V (rms) at 9A TO186: 14 - 0 - 71 - 99 - 127V (rms) at 18A
Phase 2, Secondary 2	TO185: 14 - 0 - 71 - 99 - 127V (rms) at 9A TO186: 14 - 0 - 71 - 99 - 127V (rms) at 18A
Phase 3, Secondary 3	TO185: 14 - 0 - 71 - 99 - 127V (rms) at 9A TO186: 14 - 0 - 71 - 99 - 127V (rms) at 18A
Phase 1, Secondary 4	TO185: 0-120V (rms) at 1A TO186: 2.5A
Phase 1, Secondary 5	TO185: 0-120V (rms) at 1A TO186: 0-120V (rms) at 2.5A
Phase 2, Secondary 6	TO185: 18 - 0 - 18 V (rms) at 3A TO186: 18 - 0 - 18 V (rms) at 6A
Regulation	Better than 2%
VA Rating	TO185: 2.5 KVA TO186: 5.0 KVA
Weight	TO185: 75.0 lbs [34 kgs] TO186: 110.2 lbs [50 kgs]

Table D-6. Three-Phase Transformer Specifications

Three-Phase Transformer Dimensions

The dimensions and mounting details of the transformers are shown in Table D-7.

	TO185	TO186
Overall Height	12.2" (310)	13.8" (350)
Overall Width	5.2" (132)	5.5" (140)
Overall Length	11.9" (302)	15.0" (380)

Table D-7. Three-Phase Transformer Dimensions

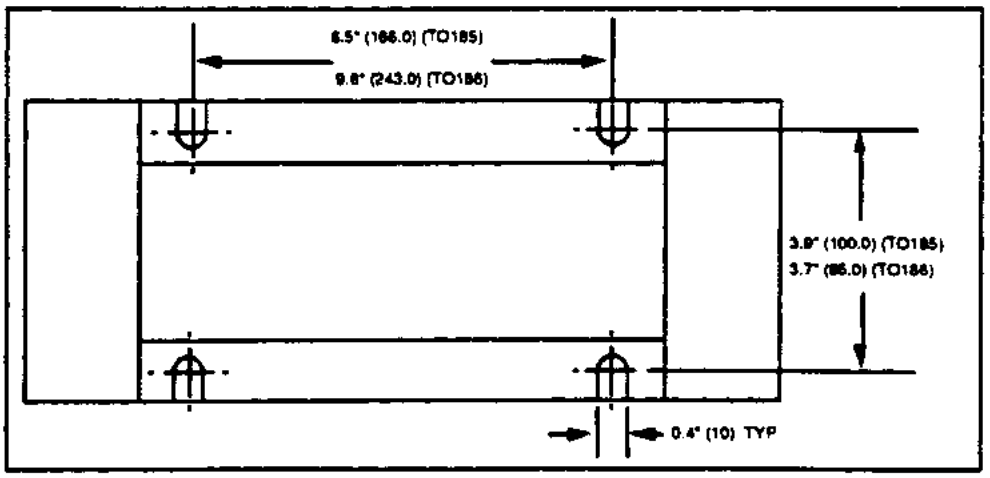


Figure D-3. Three-Phase Transformer Mounting Details

Voltage Adjustment

To adjust the transformer for the AC supply, follow the appropriate jumper connections illustrated in Tables D-8 and D-9.

Phase	Red	Yellow	Blue
Cable color	Brown	Black	Blue
Pin connection	1	9	17
Fan Connections: 45 & 46			

Table D-8. TO185 & TO186 Transformer Primary Connections

Primary Voltage	L1	L2	L3	Jumpers
240	26	32	38	1-5, 4-8, 8-9, 9-13, 12-16, 16-17, 17-21, 20-24, 1-24, 29-32, 35-38, 26-41
380	26	32	38	1-23, 3-6, 7-9, 11-14, 15-17, 19-22, 29-32, 35-38, 26-41
400	26	32	38	1-23, 3-5, 7-9, 11-13, 15-17, 19-21, 29-32, 35-38, 26-41
415/420	26	32	38	1-23, 4-6, 7-9, 12-14, 15-17, 20-22, 29-32, 35-38, 26-41
440	26	32	38	1-23, 4-5, 7-9, 12-13, 15-17, 20-22, 29-32, 35-38, 26-41
480	26	32	38	1-23, 4-5, 8-9, 12-13, 16-17, 20-21, 29-32, 35-38, 26-41

Table D-9. TO185 and TO186 Rack Connections & Jumpers

The transformer connections for 240V three-phase operation are shown in Figure D-4.

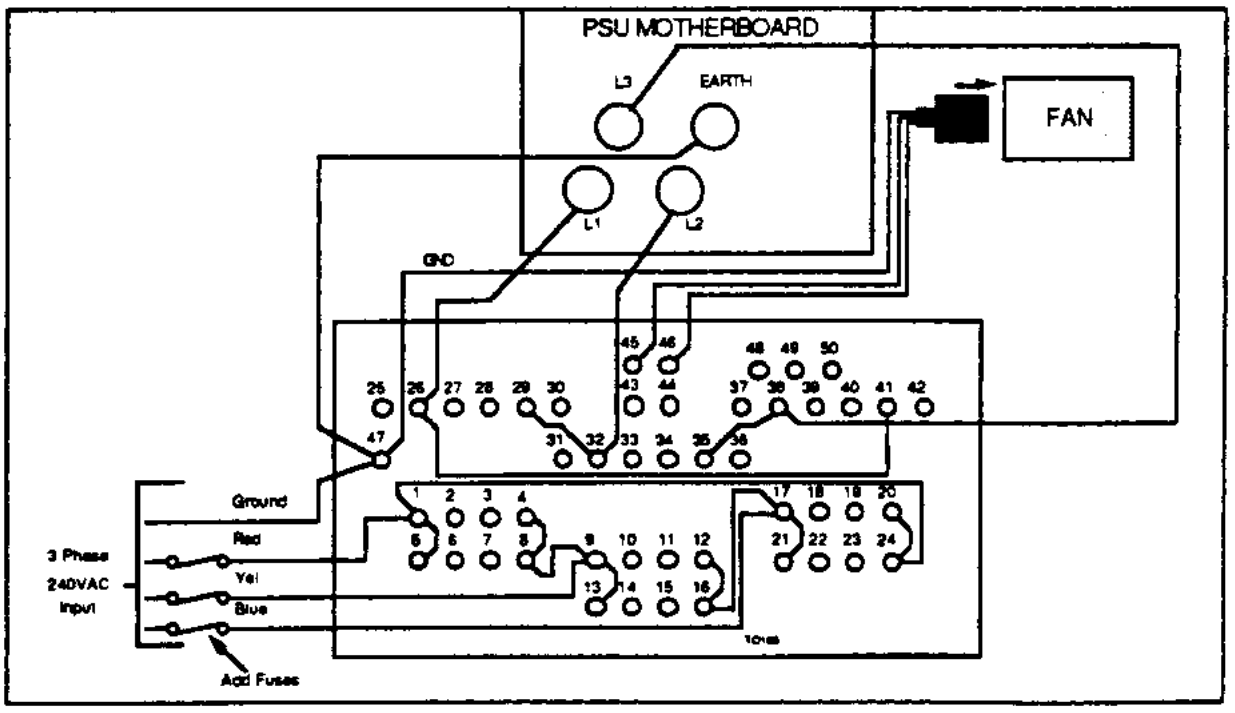


Figure D-4. 240V Three-Phase TO185 Connections

Section E. Power Supply Unit

Description

The 7715 power supply unit is a rack-mount module for use with Compumotor's range of brushless servo and stepping motor drives. The unit accepts a single- or three-phase AC input and provides a single DC output voltage. Input and output are sent via connections to a motherboard, which is mounted on the rack and connected to the power supply. The motherboard also carries the AC line fuses, two AC line suppression capacitors, and a DC supply bleed resistor.

Additional circuitry in the power supply *dumps* power generated by the motor during deceleration. The maximum power dump rating is 68 watts continuous. In high duty cycle applications, this rating may need to be increased. Provisions have been made for the addition of an external dump resistor (contact Compumotor for more information). The module can power up to three rack-mounted drives, depending on the total maximum current demand.

Adequate ventilation for the power supply unit should be provided. Rack-mounted systems usually require a cooling fan, depending on the maximum load. A built-in fan tray, with fans installed as necessary, is an available option with the Compumotor-supplied rack.

Specifications

Table E-1 provides the specifications for the power supply unit.

Parameter	Specification
AC Input	50 - 107V rms, single or three phase
DC Output	70 - 150VDC
VA Rating	Single-phase input: 2200VA at 107V Three-phase input: 3400VA at 107V
Max. Ambient Temperature	95°F (45°C)
Power Dump Current	11A (maximum)
Power Dump Rating	68 Watts continuous (provision for high-power external dump resistor if required)
Dimensions	3U/12HP
Weight	2 lbs (0.9 Kgs)

Table E-1. Power Supply Specifications

Since the power supply smoothing capacitor is distributed between the drive motherboards, maximum ratings will be achieved only when the power supply is used as part of a complete Compumotor rack system, or where suitable additional capacitors are included.

Fuse Ratings

Table E-2 lists the power supply module fuse ratings and types. Figure E-1 shows the location of the fuses. In the event of a failure, use only the specified fuse types for replacement.

Fuse	Location	Rating
FS1	Power Supply Module	2A TL HB 32 mm
FS1	PS Motherboard	25A QA HB 32 mm
FS2	PS Motherboard	25A QA HB 32 mm
FS3	PS Motherboard	25A QA HB 32 mm

Table E-2. Location & Ratings

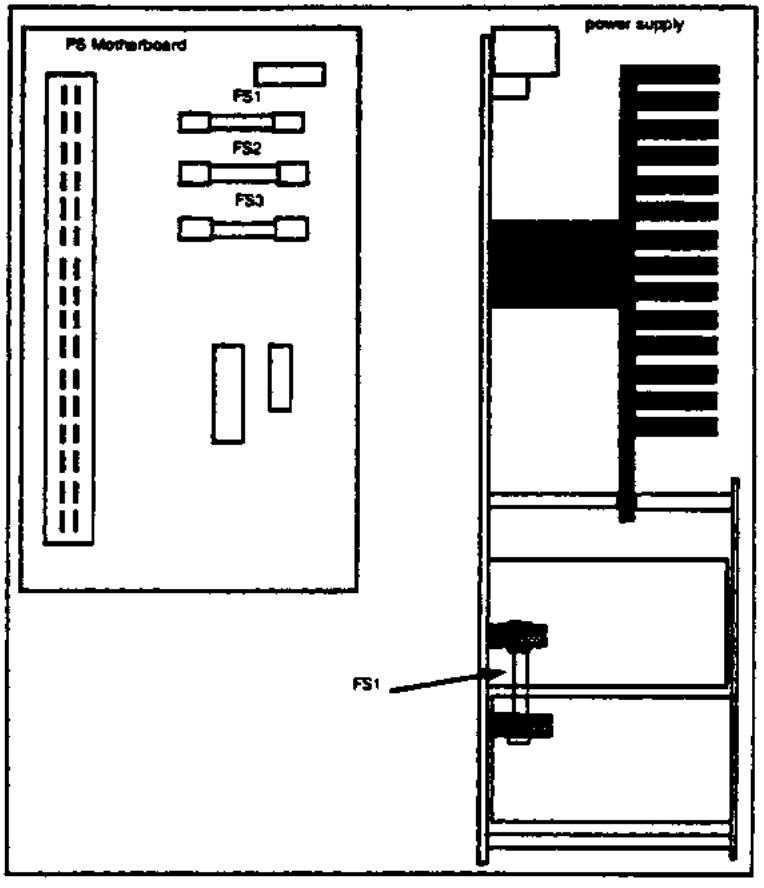


Figure E-1. Fuse Location

Installation

The power supply module is normally supplied as part of a complete rack system. If this is not the case, and you are mounting the power supply separately, you should observe the following precautions:

- Handle the unit with care.
- Install the unit in a clean, dry, well-ventilated area.
- Do not mount the unit on a hot surface.
- Do not mount the unit above a radiant heat source.
- The circulating air must be free from steam, chemical substances, combustible vapors, and oil.

Allow adequate ventilation above and below the rack. In some applications involving high duty cycles, additional dump resistors may be needed as described earlier in this chapter.

Connections

You will need to make the following connections to the power supply module via the power supply motherboard.

DC Output

The DC output is taken via a pair of standoffs mounted on the power supply motherboard. **Normally, the connections between the power supply and the rack-mounted modules are done for you.**

AC Input

The AC input to the power supply unit may be either single- or three-phase. Connection should be made between the power supply motherboard and the isolating transformer in accordance with the instructions in Section D, *Isolating Transformer*.

When making connections between the power supply and the isolating transformer or drive modules, be sure to use the appropriate gauge of wire.

HV Supply Indicator

A single green LED illuminates on the front panel to indicate the presence of the HV supply.

Principles of Operation

Figures E-2 and E-3 show the circuit diagrams of the power supply unit and its motherboard.

The AC input is rectified by the two bridges BR1 and BR2 and smoothed by capacitors C1 - C4, to provide the main HV supply. Additional smoothing is provided by capacitors mounted on the drive motherboards. When HV is present, the front panel LED will illuminate. The power dump circuit will remain operative if one of the input fuses fails when a three-phase supply is used.

The remainder of the circuitry absorbs power generated by the motor while it is deceleration. During deceleration, the motor generates power that causes the HV voltage to rise. If the HV voltage rises above the reference voltage (158V), the power dump circuit will be turned on and current will flow through the dump resistors (R8 - R11) until HV is restored to its nominal value. The dump fuse, FS1, is included to protect the dump resistors and the circuit from over-dissipation. If this fuse is blown and is not replaced, the drive may be damaged. When the AC power is switched off, the dump circuit becomes operative, reducing the HV supply to about 30V within 5 seconds.

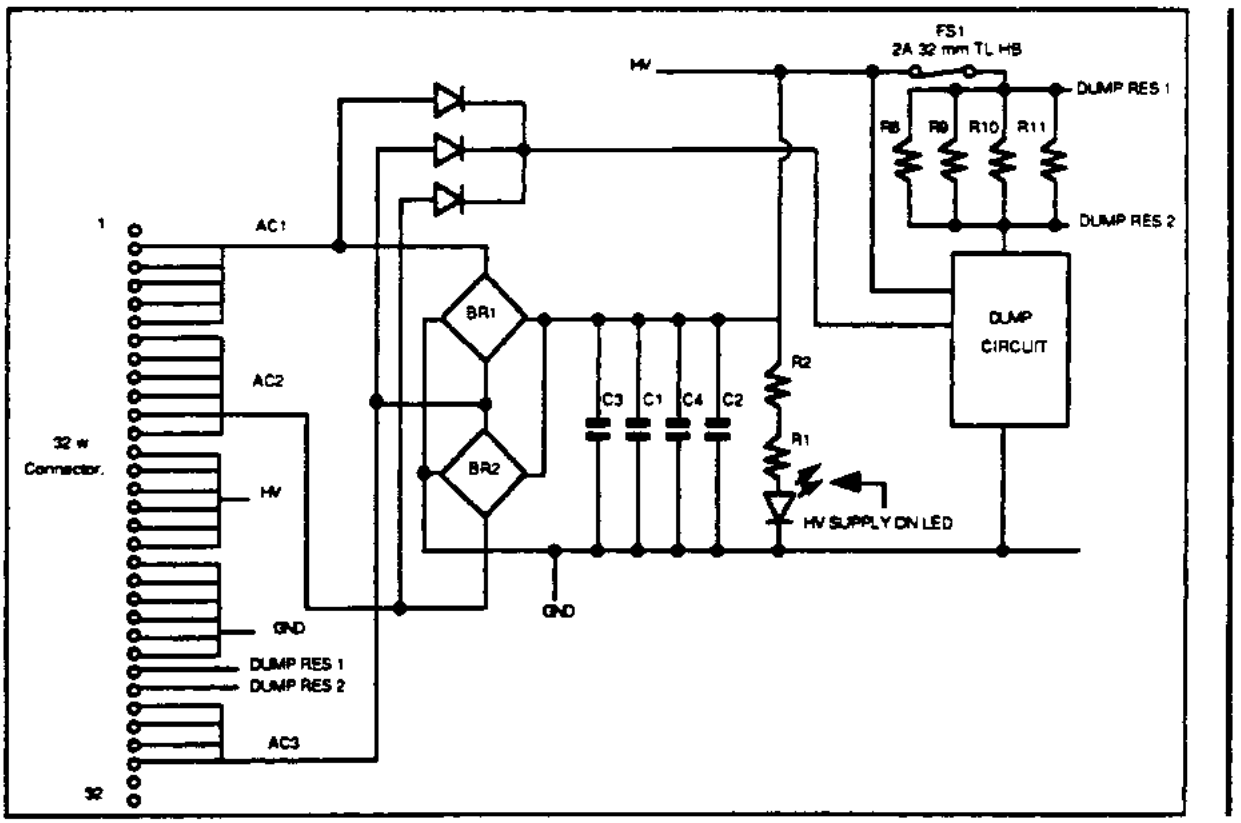


Figure E-2. Power Supply Circuit Diagram

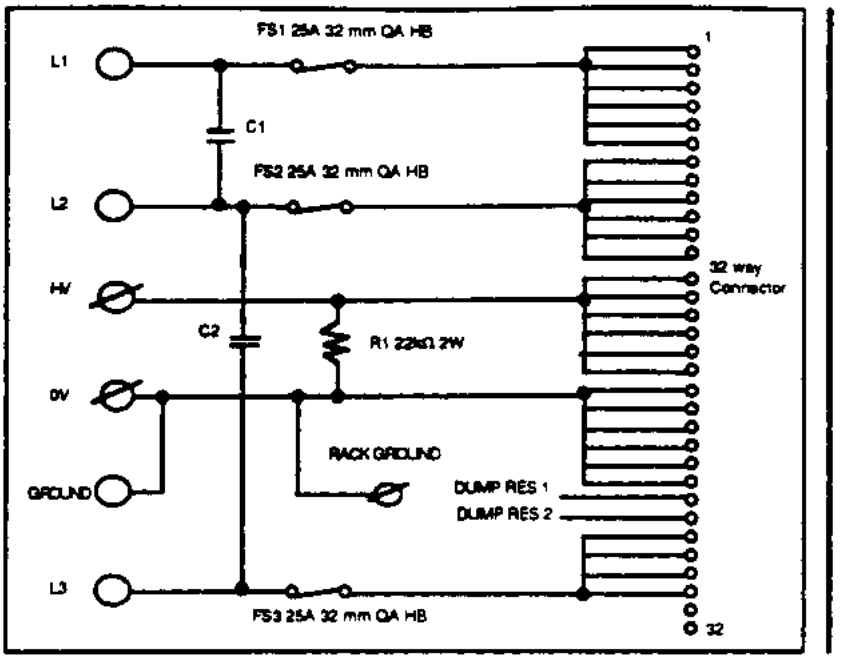


Figure E-3. Power Supply Motherboard Circuit Diagram

Summary

You should now have an understanding of the operation of the power supply circuit. You should be able to adjust the isolating transformer tappings to suit your supply voltage and to provide the required output voltage. If a fault occurs, refer to Section K, *Maintenance & Troubleshooting*.

Section F. Drive Module

Section Objective

This section gives details of the drive module. Refer to this section to set up the drive or to restart it if a fault develops.

Description

The 7153.500, 7153.501, and 7156.500 AC brushless drives and the 7153.300 hybrid servo drive are high-performance, low-loss, PWM drives designed for use with the Compumotor brushless servo motors. The drive consists of a dual torque amplifier, a waveform generator, and a switched mode power supply.

The drive module is designed to be mounted in an extended 3U rack, plugging into a rack-mounted motherboard. The drive is intended to be used with one of a number of different control cards mounted along side it in the rack and plugged into a common drive/control motherboard.

A single DC power bus is provided from the rack-mounted power supply which plugs into its own motherboard and supplies power to the rack via a pair of solid brass bus bars. Up to 3 drives, complete with their associated control cards, may be mounted in one rack.

Care must be taken to ensure adequate ventilation for the drives. Rack mounting will normally require the use of one or more cooling fans, depending on the number of drives installed. The Compumotor-supplied rack is 4U high, having a built in fan tray, with fans installed as necessary.

The drive is fully protected against overheating and against short circuits across the motor terminals, or from the motor terminals to ground. Additional protection circuitry monitors the voltage rails within the drive and disables the amplifiers if these fall outside the specifications. Fault indication is provided by a series of front panel LEDs.

Principles of Operation

Figure F1-1 is a block diagram of the drive and its connections to the system. It consists of two torque amplifiers, a switched mode power supply, and a two-phase waveform generator. These circuits are built on two printed circuit boards that also house the protection and fault indication circuits and the control input/output isolation.

Input to the waveform generator from the control card is via opto-isolators. These input signals are used within the waveform generator to generate the torque demand signals and send them to the torque amplifiers.

The torque amplifiers supply currents with the appropriate profile and phase to the motor windings under the control of the waveform generator.

The switched mode power supply operates at a frequency of 80 kHz and uses the HV supply to generate the internal supplies needed by the drive. In addition to these, two isolated supplies are generated. A 5V power supply is available to power the control card logic and the motor encoder. A 12V supply is available for use with external machine controls such as limit switches.

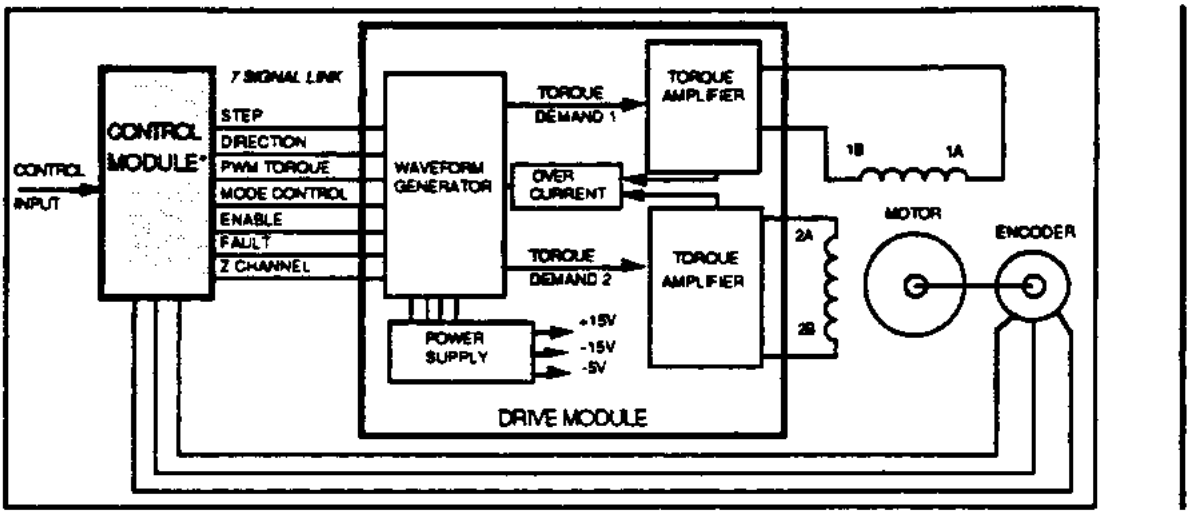


Figure F1-1. Drive Block Diagram

Figure F1-2 illustrates the waveform generator. The EPROM characterizes the drive (hybrid brushless) and contains the instructions necessary for the gate array to generate the appropriate current profiles. To generate the drive currents, an *electrical position latch* (EPL) monitors the position of the rotor using information input from the encoder, and uses this to generate the required current profile. The PWM torque signal determines the level of current required for the motor to follow the input motion profile. These two sets of data are fed into two digital-to-analog (DAC) convertors, which provide the analog torque demand signals for the two torque amplifiers. The output current from the amplifiers may be limited to a maximum level by the setting of the current limit switches (*refer to Figure F1-3 and Table F1-2 for details on current limit switch location and settings*).

When the system is switched on, the motor is powered up as an open-loop stepping motor and the rotor will rotate to locate its nearest step. This allows the drive to define a known *electrical* position. The *mode* input is then used to apply the necessary 90° torque angle and the current is adjusted to maintain position. If there is significant friction load on the motor, this procedure may result in operation at a non-optimum torque angle for up to one revolution of the motor shaft. The EPL will then be reset by the encoder index pulse, and any error in torque angle will be corrected.

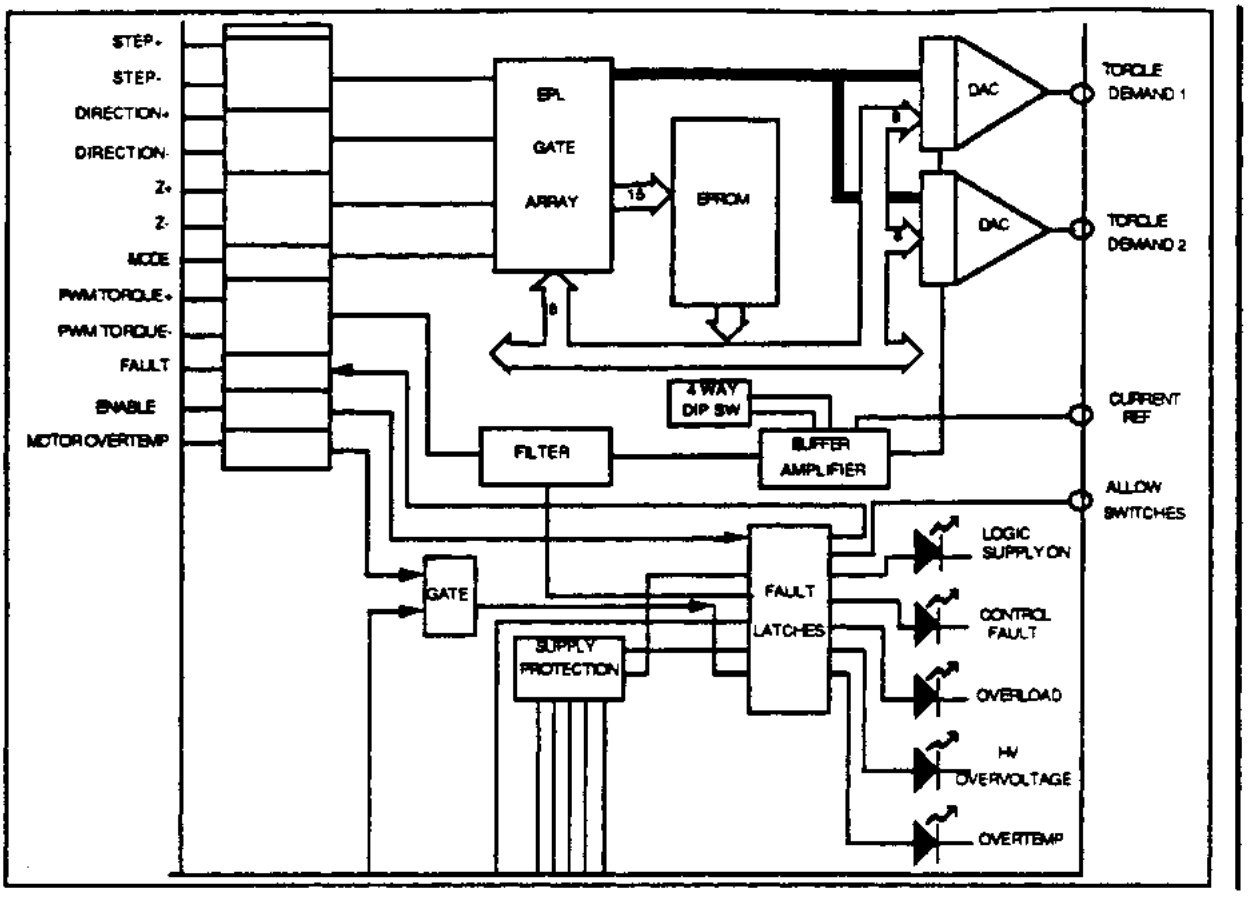


Figure F1-2. Waveform Generator

Specifications

Table F1-1 provides specifications for all five motor/drive packages.

Mtr/Drive Pkg #	7252.300	7252.501	7252.502	7252.500	7252.520
Weight. lbs (Kgs) Motor Drive	 7501.350: 10.14 (4.6) 7153.300: 3.86 (1.75)	 7521.350: 10.36 (4.7) 7156.500: 3.86 (1.75)	 7521.370: 13.23 (6.0) 7156.500: 3.86 (1.75)	 7521.350: 10.36 (4.7) 7153.500: 3.86 (1.75)	 7521.250: 4.19 (1.9) 7153.501: 3.86 (1.75)
DC Input Voltage	150VDC nominal (60VDC min., 165 max.)				
Current Rating	7156.500 - 15A rms				
Current Control	Recirculating PWM chopper with active current steering				
Switching Frequency	20kHz				
Maximum Ambient T	32-140°F (0-60°C)				
Maximum Heatsink T	185°F (85°C)				
Protection	Short circuit across phases or to 0V, drive over-temperature, power supplies out of specification. Fault indication by LEDs.				
Auxiliary Outputs (Isolated)	5V at 0.75A (for encoder and control card), 12V at 0.2A (for machine and manual inputs to control card).				
Dimensions	5.25 H x 3.6 W (3U x 18HP)				

Table F1-1. Specifications for Motor/Drive Packages

Fuse Rating

The type and rating of the HV fuse for the drive module (FS1, located on drive motherboard) is 16A QA HB 32 mm. If this fuse fails, use only the specified type for replacement.

Installation

The drive module is normally supplied as part of a complete rack system. If this is not the case and you are mounting the drive yourself, observe the installation precautions outlined in Section C.

The drive may be controlled by one of several control cards. This card is normally located in the rack next to the drive. If this is the case, you do not need to make any connections between the drive and the control card. Instructions for connecting the motor/encoder are provided in Section G, *Hybrid Servo Motor and Encoder*.

If you use a motor cable other than that supplied as part of a system, it should be of a rating appropriate for the motor current.

To reduce the effects of radiated electrical interference, shield the motor cables and, if possible, run them separately from cables carrying low-level signals (such as encoder outputs).

Setting Up the Current Limit

You can adjust the maximum rms current output from the drive with the DIP switches located in the drive module (see Figure F1-3). If the motor runs hot, you may want to reduce the current, *though this may affect the dynamics and tuning of the system.* Table F1-2 shows the switch settings for the range of current limit levels.

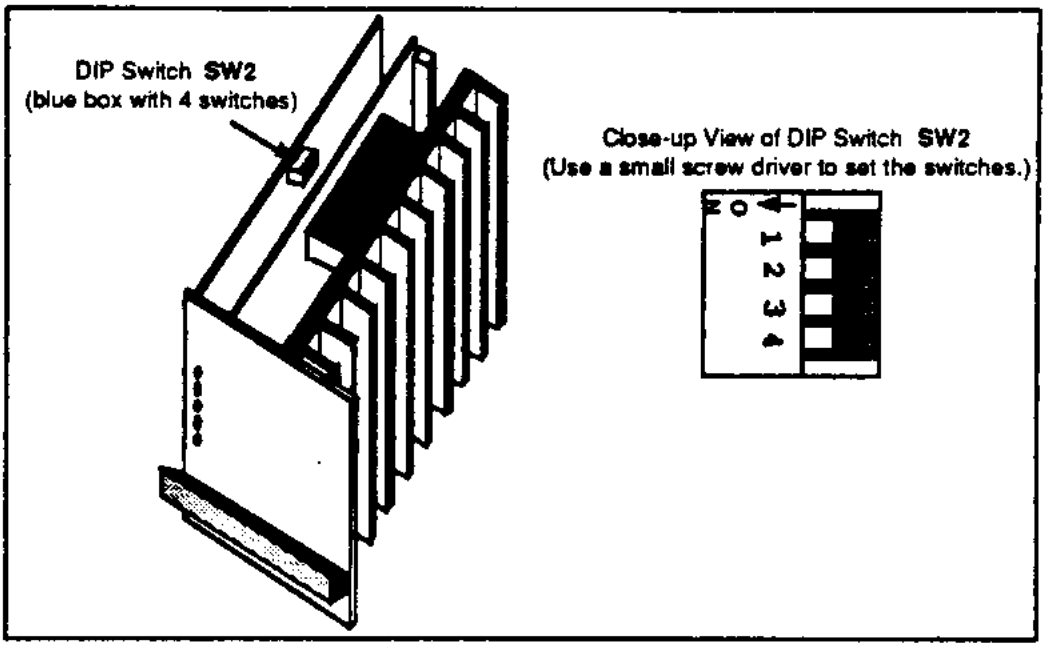


Figure F1-3. Drive Current Limit Switch

Max Current O/P (amps rms) *	Max Current O/P (amps rms) **	DIP Switch Settings (SW2) Bit 1	Bit 2	Bit 3	Bit 4
2.84	1.42	OFF	OFF	OFF	OFF
3.6	1.8	OFF	OFF	OFF	ON
4.52	2.26	OFF	OFF	ON	OFF
5.3	2.65	OFF	OFF	ON	ON
6.1	3.05	OFF	ON	OFF	OFF
6.88	3.44	OFF	ON	OFF	ON
7.76	3.88	OFF	ON	ON	OFF
8.52	4.26	OFF	ON	ON	ON
9.3	4.65	ON	OFF	OFF	OFF
10.08	5.04	ON	OFF	OFF	ON
11.8	5.90	ON	OFF	ON	ON
12.54	6.27	ON	ON	OFF	OFF
13.32	6.66	ON	ON	OFF	ON
14.24	7.12	ON	ON	ON	OFF
15***	7.5***	ON	ON	ON	ON

* Applies only to drive number 7156.500 (drive/motor pkgs. 7252.501 & 7252.502)
** Applies only to drive number 7153.500/7153.501 (drive/motor pkgs. 7252.500 & 7252.520)
*** Default setting

Table F1-2. DIP Switch Settings for Current Limit Levels

CAUTION

Two trim potentiometers (RV3 and RV4) are mounted to the PCBs of the drive module. They are factory set and sealed. Adjusting them may cause rough motor rotation or prevent rotation – and invalidate the warranty.

Fault Indicators

The drive is protected against overloads and short circuits. Operation of the protection circuits is indicated by LEDs on the drive front panel. Detection of a fault by the protection circuitry disable the drive being .

The faults are latched and may be reset once the fault has been corrected, either by temporarily disconnecting the AC supply or by enabling the drive with the **ST** or **ON** commands.

Logic Supply On

Illumination of this green LED indicates that the low voltage supplies to the drive are functioning. The LED should normally be illuminated. However, if the logic supply drops for a short period and then returns to normal, the LED could extinguish and not re-illuminate even though the drive may still be functioning normally. This condition can be reset by removing and re-applying the *Enable* signal (pin 1 on connector I/O 3).

Control Fault

Illumination of this red LED indicates that the PWM torque demand signal has gone permanently high or low.

Overload

Illumination of this red LED indicates an excessive current demand from the drive as a result of a short circuit across the two motor terminals or a short to ground.

HV Over-voltage

Illumination of this red LED indicates that the supply bus is not within the specification level (55 - 170V). This condition is most likely to occur if excessive regeneration is taking place. If this is the case, it may be necessary to install an external power dump resistor.

The built-in power dump resistor is located on the power supply unit. If you need to increase its rating, reference Section E, *Power Supply Unit*.

Over-temperature

If the drive heatsink temperature rises above 185°F (85°C), this red LED will illuminate.

Summary

You should now have a basic understanding of the operation of the drive module. If a fault develops, you should be able to decide whether or not it is a drive module fault by referring to the above section on *Fault Indicators* and to Section K, *Maintenance & Troubleshooting*.

Section G. Hybrid Servo Motor and Encoder

Section Objective

This section contains the electrical and mechanical specifications for the motor and encoder. It shows details of the cable connections and lists the recommended cable types.

Description

The motors are hybrid brushless servo motors, with a built-in incremental encoder that provides shaft position and commutation information.

The pulse count from the encoder may be multiplied, on the control card, by a factor of 4 (see the CCS command description in Section J).

The motor/encoder package is supplied complete with cables, which are terminated at the motor-end by means of screw terminal connectors.

Motor Specifications

Table G1-1 below gives the specifications of the AC brushless and hybrid servo motors.

Motor/Drive Package #	7252.501	7252.520	7252.502	7252.500	7252.300
Motor Part #	7521.350	7521.250	7521.370	7521.350	7501.350
Rotor Inertia oz-in² (kg-cm²)	8.75 (1.6)	1.53 (0.28)	13.12 (2.4)	8.75 (1.6)	9.8 (1.8)
Insulation	NEMA Class F	NEMA Class F	NEMA Class F	NEMA Class F	NEMA Class B
Built-in Encoder	1000-line	500-line	1000-line	1000-line	1000-line
Maximum Resolution	4000 counts/rev	2000 counts/rev	4000 counts/rev	4000 counts/rev	4000 counts/rev
NEMA Frame Size	34	23, 2-pole pair	34, 3-pole pair	34, 3-pole pair	34, 50-pole pair
Weight (Inc. Cable) lbs (kgs)	10.3 (4.7)	4.2 (1.9)	13.2 (6.0)	10.3 (4.7)	10.1 (4.6)
Max. Case Temperature	212°F (100°C) same for all motors				
Environmental	IP50 same for all motors				
Standard Cable	10 ft (3 m) same for all motors				

Table G1-1. Motor Specifications

Motor Dimensional Drawings

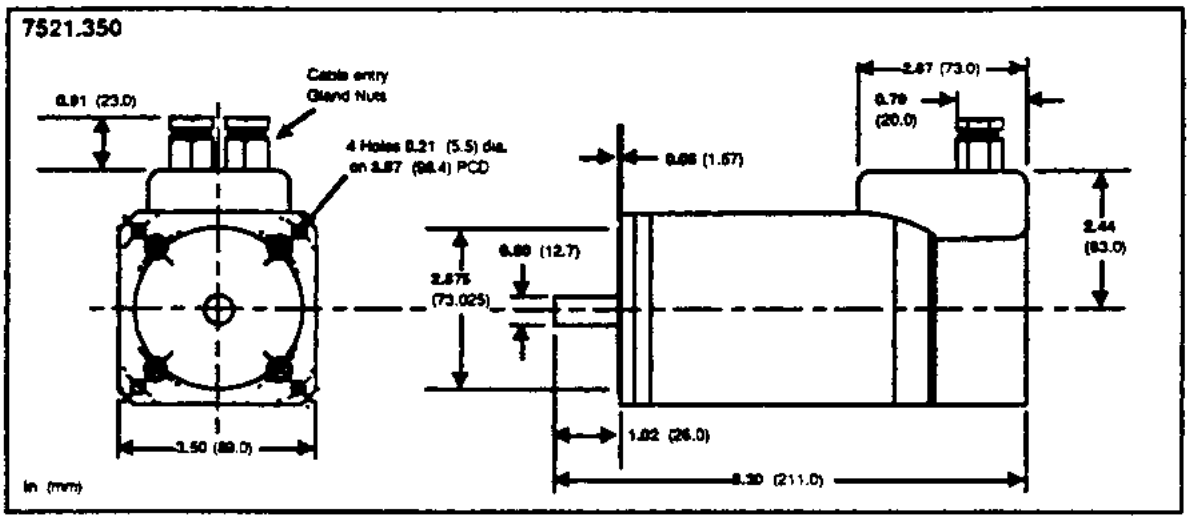


Figure G1-1. 7521.350 Motor Dimensions (7252.500 & 7252.501 Motor/Drive Pkgs.)

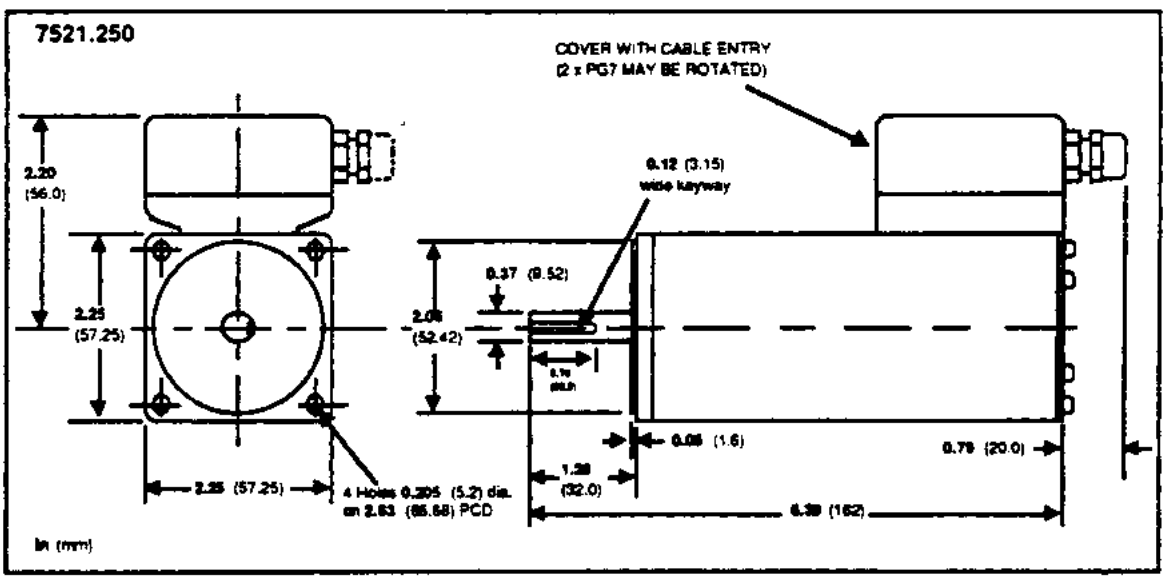


Figure G1-2. 7521.250 Motor Dimensions (7252.520 Motor/Drive Package)

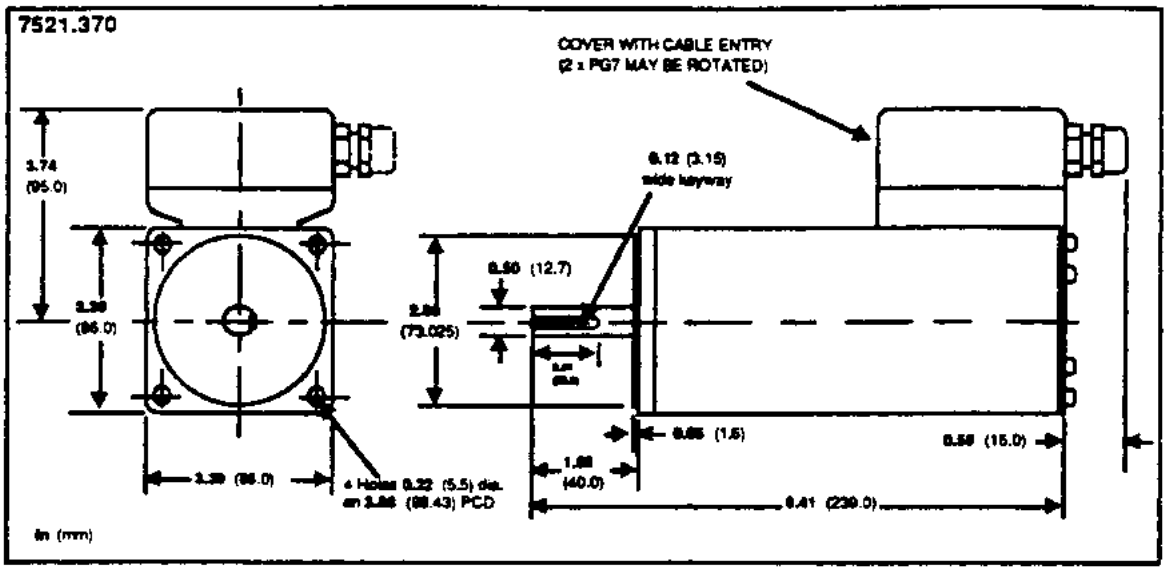


Figure G1-3. 7521.370 Motor Dimensions (7252.502 Motor/Drive Package)

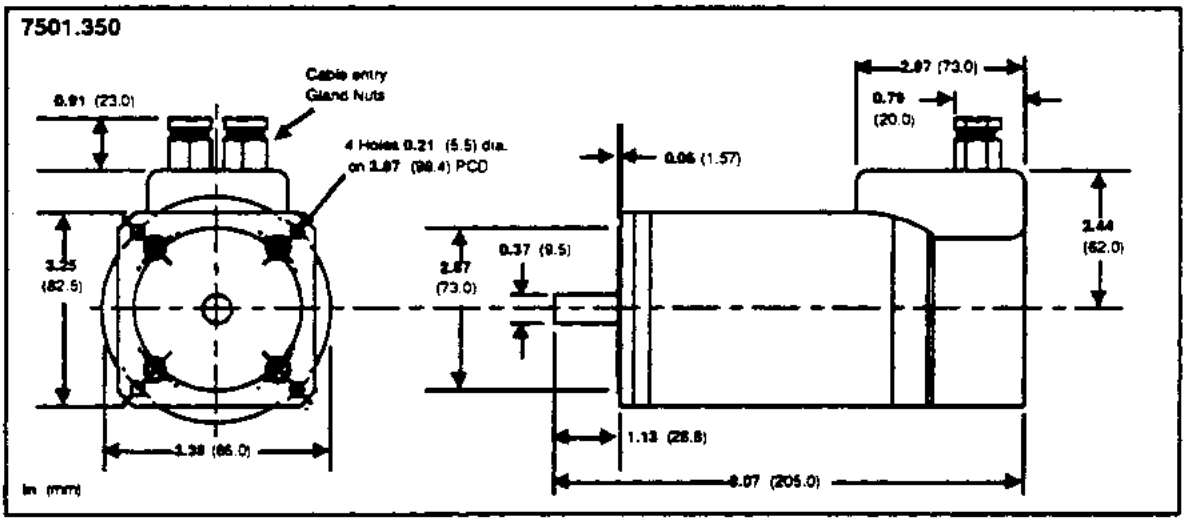


Figure G1-4. 7501.350 Motor Dimensions (7252.300 Motor/Drive Package)

Connections

Compumotor motors are shipped with pre-wired cables, ready to connect to the System 7 motherboard **MOTOR** and **MOTOR FEEDBACK** (encoder) connectors (see Figure G1-5 for cable connections). The cables are supplied at the length you specified in your order.

If you need to attach longer cables, the *Cable* section of this section shows the recommended types.

Figure G1-5 illustrates cable color and number codes. The number codes are printed directly on the motor leads. Each pair of encoder wires (e.g., CHA+ and CHA-) is sealed together in the same clear plastic tubing. This is done to ensure that you use the proper black ground wire for the appropriate encoder input.

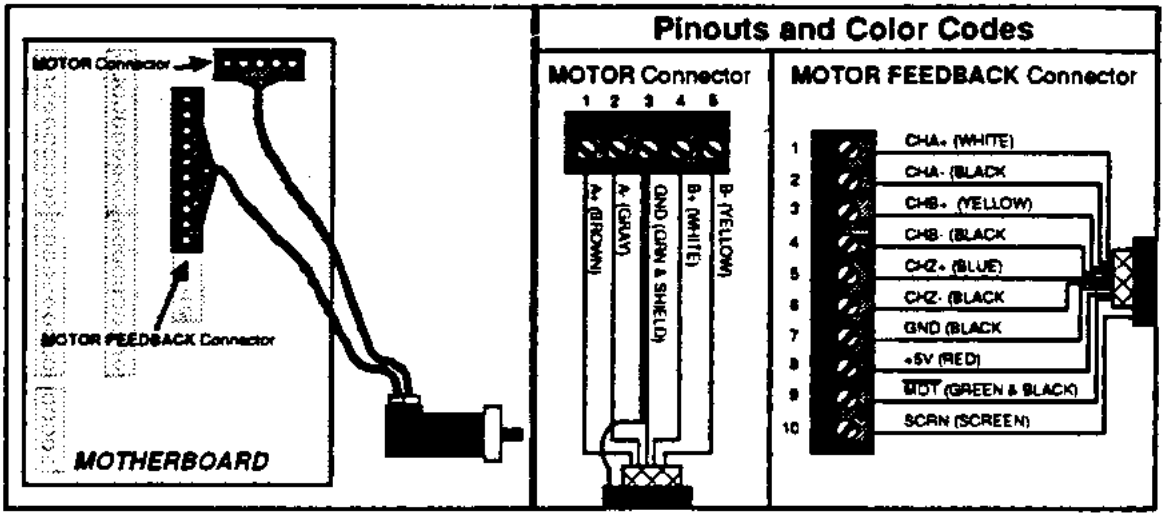


Figure G1-5. Motor and Encoder Connections

Cabling

If you use cables other than those supplied as part of the system, use cables that meet the following specifications.

Motor

You can get additional cables from Parker Compumotor (part # 4011.350). The specifications for the cable are listed below.

- BS6360, VDE0295 class 5 & 6
- Color code: DIN47100
- 5 core 1 mm square screened
- Type: Olflex CY/1.0/5

Encoder

- Five twisted pairs, 7/0.203 mm (24 AWG)
- PVC insulated and wrapped in aluminized tape
- Tinned copper drain wire
- Characteristic impedance 77Ω
- UL number 2464
- Manufacturers' type numbers: Alpha—5475; Belden—9505

Encoder Backup Supply

By connecting an external +5V power supply to the **ENCODER BACKUP SUPPLY** connector (see Figure G1-6), the control module can retain the motor's position if the power is interrupted. To determine the motor's position after a temporary power outage, simply use the **DR** or **PR** status commands.

To use the encoder backup supply, you must first access the motherboard and remove the wire (*link*) labeled **ENCODER INT. 5V LINK**. Use the following procedure to access the 5V link:

1. **Make sure the power is off.**
2. Using a wire cutter, remove the 5V *link* (located next to the ENCODER BACKUP SUPPLY connector) on the motherboard.

Using pins 3 and 4 on the same connector, you can daisy-chain multiple units from one power supply (see Figure G1-6). This reduces the wiring effort.

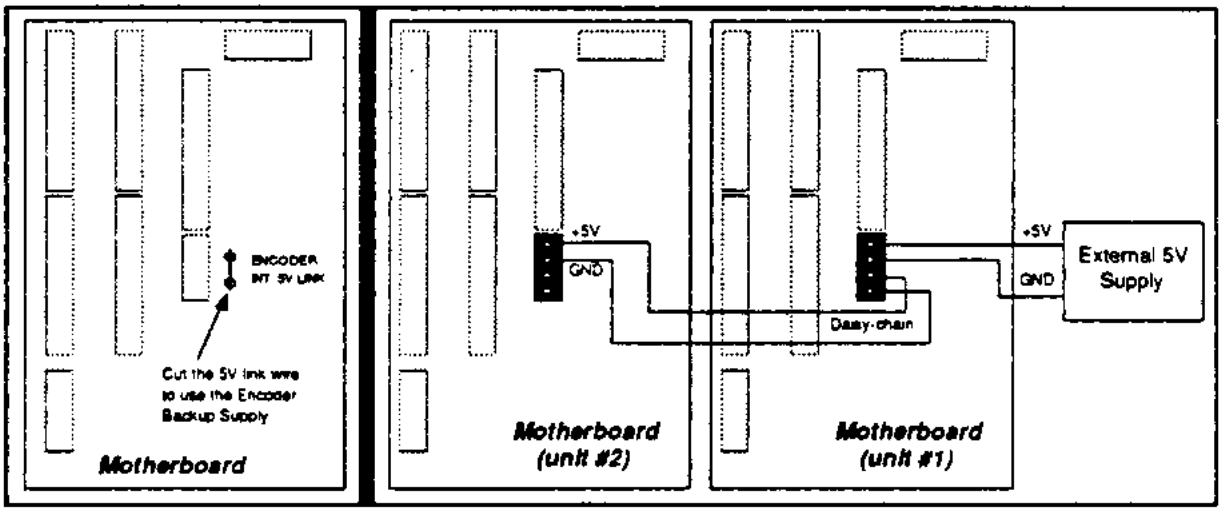


Figure G1-6. Encoder Backup Supply Connections

Couplings

Special couplings that accommodate different types of misalignments are available. There are three types of misalignments which can exist in any combination:

- Parallel Misalignment. The offset of two mating shaft center lines, although the center lines remain parallel to each other
- Angular Misalignment. When two shaft center lines intersect at an angle other than zero degrees
- End Float. A change in the relative distance between the ends of two shafts

Special couplings accommodate the above misalignments and to transmit the desired torque. You should consult the coupling manufacturer to ensure that your application is within the coupling's specified torque capacity and misalignment ranges.

There are three types of shaft couplings: single-flex, double-flex, and rigid. Like a hinge, a single-flex coupling accepts angular misalignment only. A double-flex coupling accepts both angular and parallel misalignments. Both single-flex and double-flex, depending on their design, may or may not accept end-play. A rigid coupling does not compensate for any misalignment.

Single-Flex Coupling

When a single-flex coupling is used, one and only one of the shafts must be free to move in the radial direction without constraint. *Do not use a double-flex coupling in this situation because it will allow too much freedom and the shaft will rotate eccentrically, causing large vibrations and immediate failure.*

Double-Flex Coupling

Use a double-flexed coupling whenever two shafts are joined that are fixed in the radial and angular direction (angular misalignment). *Do not use a single-flex coupling with a parallel misalignment; this will bend the shafts, causing excessive bearing loads and premature failure.*

Rigid Coupling

Rigid couplings are generally not recommended. They should be used only if the motor is on some form of floating mounts that allow for alignment compensation.

For unusual motor installations contact your Compumotor applications engineer for assistance at (800) 358-9070.

Summary

This section has provided mechanical information, including shaft loading (for the motor) and the connection information needed to lengthen or re-terminate the motor or encoder cables. In the next section, we will explain the details of the control card and how it is programmed.

Section H. RS-232C Control Module

Section Objective

This section discusses the set up and operation of the RS-232C control module. This information should be thoroughly understood before carrying out the final set-up of the system.

Description

The RS-232C control module is designed to work with the System 7 rack, performing position control and indexing functions using an industry standard RS-232C interface. It is easily controlled from computers, terminals, and most programmable controllers. The control module can function with the following types of motors:

- Open-loop stepper motors
- Hybrid brushless servo motors
- AC brushless servo motors

It can function as a Step & Direction control module with step and direction position command from an external source or as a brush servo controller.

The programming language is based on Compumotor's X Language and the RS-232C control module is capable of storing and executing complex motion programs from its non-volatile memory (battery backed-up RAM).

Features

- Differential pulse width modulated outputs
- High-speed operation
- RS-232C command interface
- 1 to 8 devices can be daisy-chained on a single RS-232C port using zero delay daisy chaining (ZDDC)
- 16K of nonvolatile program memory for storing up to 16 sequences and parameters.
- Automatic load and execution of motion programs (sequences) at power up
- Sequence execution can be initiated by external switches, computer, or programmable controller
- Sequence upload, download, and memory verification
- Optically isolated inputs for home position, end-of-travel limits, and emergency stop and jog functions
- Seven optically-isolated, user-definable inputs
- Programmable resolution
- Programmable analog output for signal monitoring
- Watchdog output (Optically isolated)
- In-position output (Optically isolated)
- Three programmable outputs (Optically isolated)
- Motherboard has 12V supply for opto-isolated I/O
- Encoder following - axis A follows the encoder of axis B
- Encoder superposition - motion is the sum of the encoder following input and an internally generated index

- Linear interpolation on two axes
- Registration mode
- Synchronous moves
- Servo self-tuning as a torque or velocity amplifier
- Dither facility - for pneumatic and hydraulic valves

Fault LED Indicator

A single control module LED, visible through the front panel, functions as a combined fault indicator. If the control module disables the drive when no terminal is connected to the RS-232C control module, you can count the number of LED flashes to determine the reason for the disabling.

1 flash = RST command issued (disabled)
3 flashes = New memory map
4 flashes = Excessive following error
5 flashes = Checksum failure
7 flashes = Prolonged max. torque demand
8 flashes = ESTOP input seen

The sequence of flashes continues until the fault is cleared. A one- to two-second delay separates the LED flash sequences.

Specifications

Parameter	Value
RS-232C Communication	
Type	3-wire (Tx, Rx, GND), minimum voltage swing on Rx line = ±3V
Parameters	9,600 baud, 8 data bits, 1 stop bit, no parity
Connector	Removable screw terminals
Configuration	Up to 8 interfaces can be controlled from a single RS-232C port using ZDDC. All commands can be preceded by a device address. The address is set by jumpers JU7, JU8, & JU9 on the board.
Motor/Drive Compatibility	Open-loop micro-stepper, AC brushless, or hybrid servo drives (selected by software commands)
Digital Servo Loop	
Update Time (frequency)	550 µs
Servo Tuning	RS-232C values stored in battery-backed RAM
Tuning Parameters	PIVF or PID options with digital filter
Move Parameter Ranges	
Position	±1 to 2,097,152 steps
Velocity	0.0001 to 100 rps (0.488 to 4 x 10^6 steps/sec)
Acceleration	0.06 to 999,999.00 rps^2 (244 to 4 x 10^9 steps/sec^2)
Motion Program Storage	
Memory type, Capacity	Battery-backed RAM; capacity = 6400 characters total
Number of Programs, Lengths	16 programs with variable lengths up to memory limit
Program Selection	Via RS-232C, Auto-execution on power-up, or binary address on sequence-select inputs
Maximum Encoder Frequency	100 kHz (lines/sec before multiplication)
Resolution Ranges	
Feedback Encoder	1 to 32,767 counts/rev
User-programmed	1 to 32,767 counts/rev
Coordinate System	Incremental or absolute
Operating Modes	Preset, Preset with speed change, & continuous
Indexer Loop Update	Every 2 ms
Optically-Isolated Inputs	Home, Limits, Jogs, ESTOP, & Inputs 1 through 7: 10-30V ON maximum; 5mA at 30V OFF maximum; maximum reverse voltage: -5V
Optically-Isolated Outputs	Watchdog, in-position, & fault outputs: NPN open-collector common to isolated ground. 300mA ON maximum 30V OFF maximum, 2.5V at 300mA maximum voltage in the ON state
Analog Monitor Outputs	Velocity and position error 0 to ±2.5V relative to interface 0V
Power Requirements	+5V at 300mA, derived from drive module

Table H1-1. RS-232C Control Module Specifications

Motion Profiles

In any motion control application, the most important requirement is precise shaft rotation, whether it be with respect to position, time, or velocity. The type of motion profile needed depends on the motion control requirement. The following sections describe the basic motion profiles.

Velocity, acceleration, and distance must be defined before any preset move can be executed. The value of these parameters determines the type of motion profile as either triangular or trapezoidal.

Triangular Profile

A triangular profile results when the velocity and acceleration are set such that the designed velocity is not attained before half of the specified distance has been travelled. This results from either a very low acceleration, a very high velocity, or both over a relatively short distance. For example, if the acceleration is set to 1 rps^2, velocity is set to 2 rps, and distance is set to 8,000 steps (2 revs), a triangular motion profile will result. This is because by the time the motor shaft has reached the desired velocity of 2 rps, it will also have traveled half of the defined distance due to the acceleration setting of 1 rps^2. The motion profile for this move is shown in Figure H1-1.

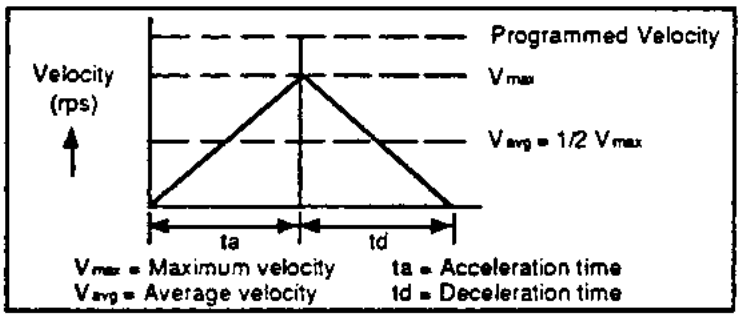


Figure H1-1. Triangular Profile

Trapezoidal Profile

A trapezoidal move profile results when the defined velocity is attained before the motor shaft has moved half of the specified distance. This is due to a defined velocity that is low, a defined acceleration that is high, a move distance that is long, or a combination of all three. For example, if the acceleration is set to 10 rps^2, velocity is set to 1 rps, and distance is specified as 20,000 steps (5 revs), the resulting motion profile is shown in Figure H1-2.

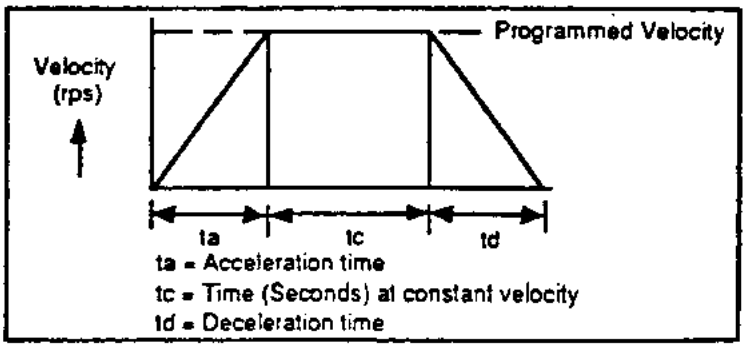


Figure H1-2. Trapezoidal Profile

User Profiles

You may define a move profile with the MC command to establish the continuous mode. Velocity can be programmed on-the-fly with the V command. Sending commands to the RS-232C control module in rapid succession allows smooth profiling that will allow circles and arcs to be traced using a two-axis system, or allow smooth (low jerk) motion by virtue of S-curve acceleration rather than straight ramping as with triangular and trapezoidal moves. Once you have generated a sequence of commands to trace the correct profile, the profile will be very repeatable. This may require some trial and error. You can combine V commands with time delays in a sequence buffer to create very complex move profiles.

The RS-232C control module also has a Speed Change mode (MQ), which is like the Preset mode in that the move distance is pre-defined, but it is possible to change speed in the middle of the move as required based on a distance, input, or time delay trigger.

Operating Modes

This section provides an overview of the various operating modes and special features.

Normal (Preset) Modes

A *preset move* refers to a move of a specified distance (in motor steps). Preset moves allow you to position the motor in relation to the motor's previous stopped position (incremental moves) or in relation to a defined zero reference position (absolute moves). You can select preset moves by putting the RS-232C control module into *mode normal* with the MN command.

Incremental Preset Moves

If the RS-232C control module is in the incremental mode (MN or MPI command), a preset move will move the motor shaft the specified distance (in motor steps) from its starting position. For example, to move the motor shaft 1.5 revolutions, a preset move of +6,000 steps would be specified, assuming a 4,000 step/rev encoder resolution setup. Every time this move is executed, the motor shaft will move 1.5 revolutions positive from its current position. You can specify the direction of the move with the distance command by using the optional ± sign (D+6000 or D-6000). You can also define it separately with the H command (H+ or H-).

Example Program

Command	Description
1MPI	Sets incremental mode
1DPA	Response=1500 (current position)
1D4000	Sets move to be 4000 steps CW
1G	Executes move to position 5500
1D-3000	Sets move to 3000 steps CCW
1G	Executes move to position 2500

Absolute Preset Moves

A preset move in absolute mode (MPA command) will move the shaft of the motor the specified distance (in motor steps) from the absolute zero position. The absolute position can be set to zero with the PZ command. For instance, at the end of a go home move (GH command). The absolute zero position is initially the power-up position, and will remain that way until it is changed with a PZ command. Any preset move performed while in the absolute mode will position the motor shaft the defined distance (in motor steps) from the absolute zero position. For example, with the RS-232C control module at the absolute zero position a move with a distance of +4,000 will cause the motor shaft to turn 1 revolution in the positive direction. If a move with the same defined distance is executed immediately after this move, the motor shaft will not turn, since it is already +4,000 steps from the absolute zero position. The direction of an absolute preset move will depend upon the shaft position at the beginning of the move and the position that it is being commanded to move to. For example, if the motor shaft is at absolute position +12,800, and position commanded is +5000, the motor shaft will move in the negative direction a distance of 7,800 steps to absolute position +5000.

The RS-232C control module powers up in incremental mode. Issuing the MPA command will set the mode to absolute. Issuing the MPI command will switch the mode from absolute back to incremental. The RS-232C control module retains the absolute position referenced, even while in incremental mode. However, the counter does have an upper limit just slightly more than 2,000,000 counts.

Example Program

Command	Description
1MPA	Sets absolute mode
1PZ	Sets current position to be absolute zero
1D4000	Sets move to absolute position 4000
1G	Execute a CW move to position 4000
1D2500	Sets move to absolute position 2500
1G	Executes move of 1500 steps CCW to position 2500

Continuous Mode

Continuous moves (MC command) cause the motor to accelerate and attain the specified velocity and then continue to move. To change velocity while the motor is moving use the Velocity (V) command. Issuing a stop (the S command) will decelerate the motor to a stop at the last defined acceleration rate. The distance parameter is not used, although it is saved in case the mode is returned to preset.

This mode is useful for applications that require constant spinning of the load, when the motor must stop after a period of time has elapsed rather than after a fixed distance, or when the motor must be synchronized to external events such as trigger input signals.

Example Program

Command	Description
MC	Sets continuous mode
D4000	Ignored in continuous mode
A50	Sets acceleration to 50 rps^2
V20	Sets velocity to 20 rps
G	Motor runs at 20 rps continuously
V10	Motor speed changes to 10 rps
S	Motor stops

Speed Change Mode

The MQ command allows speed changes to be made during a preset move. The speed changes can be triggered at a set distance using the TRD command or by an input using the TRE command.

Example Program

Command	Description
3MQ	Set MQ mode for axis 3
3A50	Set axis 3 acceleration to 50 rps^2
3V15	Set velocity to 15 rps
3D8000	Set move distance to 8000 steps
3G	Start move
3TRD3000	Go to next V command at D = 3000 steps
3V5	Change velocity to 5 rps
3TRE_X1	Go to next command when Input 2 goes high
3V2	Change velocity to 2 rps

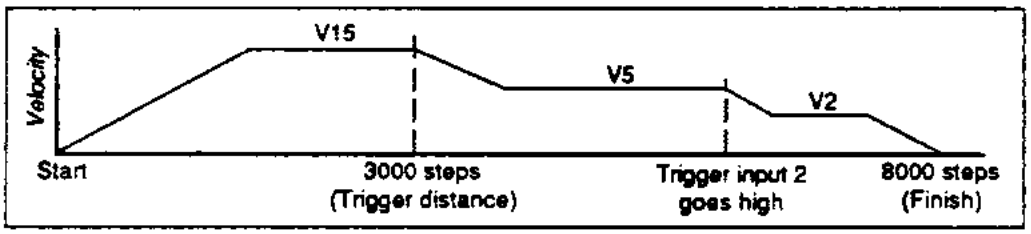


Figure H1-3. Speed Change Mode Example Profile

Registration Mode

A move may be programmed to end a specified distance after a registration pulse appears at Input 6. The mode is selected using the TRR command with the required registration distance in the MQ or MC modes. Figure H1-4 illustrates an example registration move.

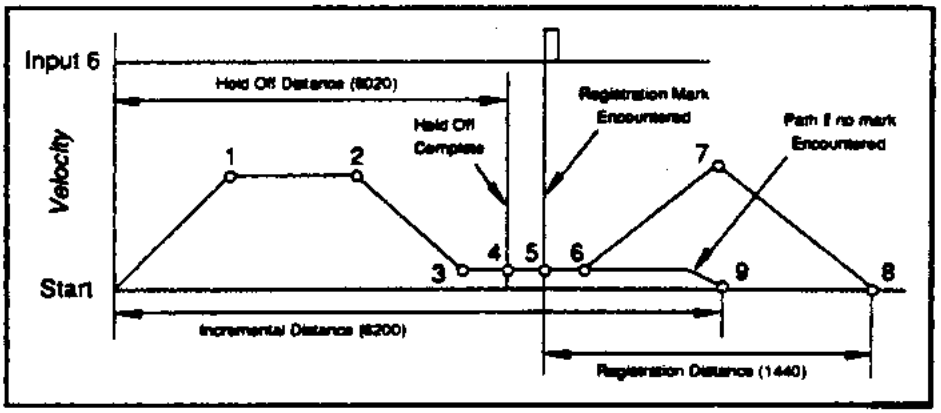


Figure H1-4. Example Registration Move

Example Program

Command	Description
MQ	Select MQ mode
D6200	Set move distance to 6200 steps
A100	Set all accelerations to 100 rps^2
V10	Set velocity to 10 rps
G	Go
TRD4000	At distance = 4000 steps
V1	Change velocity to 1 rps
TRD6020	At distance 6020 look for registration pulse
TRR1440	Travel 1440 steps from registration pulse
V50	At a velocity of 50 rps
TRI2	Set Output 2 when in position

Program Operation

When executing this program, the axis first accelerates at 100 rps^2 to 10 rps (1). When it reaches a distance of 4000 steps (2), its speed reduces to 1 rps (3) while it waits for the registration pulse. Shortly after the pulse is received (4), the axis accelerates toward 50 rps, but will not reach the speed before starting to decelerate (5) to stop at 1440 steps from where the pulse occurred (6).

If the pulse was not received after 6020 steps (the *hold off distance*) and before 6200 steps (the incremental distance), the axis will stop at 6200 steps. This sets a window of 180 steps during which the pulse is expected.

Asynchronous Mode

The MAS command selects the asynchronous mode in which axes may start and stop moves independently of other axes in the ZDDC system. In this mode, moves are executed in the order that they are programmed.

Synchronous Mode

The normal operating mode is the asynchronous mode, but in this mode it cannot be guaranteed that moves will be started simultaneously. In synchronous mode, however, a simultaneous start for moves on all axes can be achieved. When the go command (GSY) is given at the start of a synchronous move, the system waits for the axes to be ready and then starts them together. If they are not all ready within two seconds of the go command being given, the move is aborted, unless a PSY command has previously been given, in which case the wait for readiness is indefinite.

After the synchronized start, the moves will be at the programmed speed, acceleration, and distance for each axis. The synchronous mode, entered with the MSY command, is not exclusive of other modes. Exit from the mode is by use of the MAS command to return to asynchronous mode.

Linear Interpolation

The RS-232C Control Module has the capability of linear interpolation on two axes. This allows the axes to start moving at the same time and then move with the same speed relationship to each other until their moves end simultaneously.

The function requires that both of these axes be in mode synchronous (MSY). The individual speeds and accelerations for the axes are calculated by the processor so that moves are started and ended at the same time and the axes remain at the same speed and position relationship to each other throughout the moves.

The commands FDA and FDV are used to set the required acceleration and velocity of the resulting vector movement . Distances for both axes to be used for the linear interpolation move may be either positive or negative and they are set as part of the GL command, which is the command used to initiate the move. Figure H1-5 shows the move profiles followed by the two axes during a linear interpolation move resulting from the following commands:

```
MSY
FDV5
FDA100
GL60000,30000
```

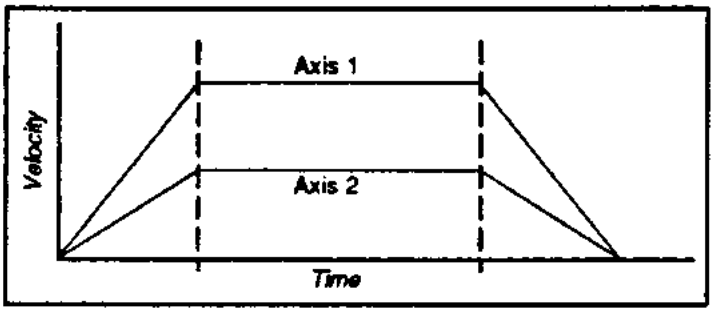


Figure H1-5. Linear Interpolation Mode Move Profiles

The effect of these profiles on an X/Y table would be as shown in Figure H1-6. Axis 1 has twice the distance of axis 2 to move in the same time, so the processor sets its acceleration, velocity and deceleration to twice the axis 2 values. Both sets of values are calculated from the distances and the FDA and FDV values set.

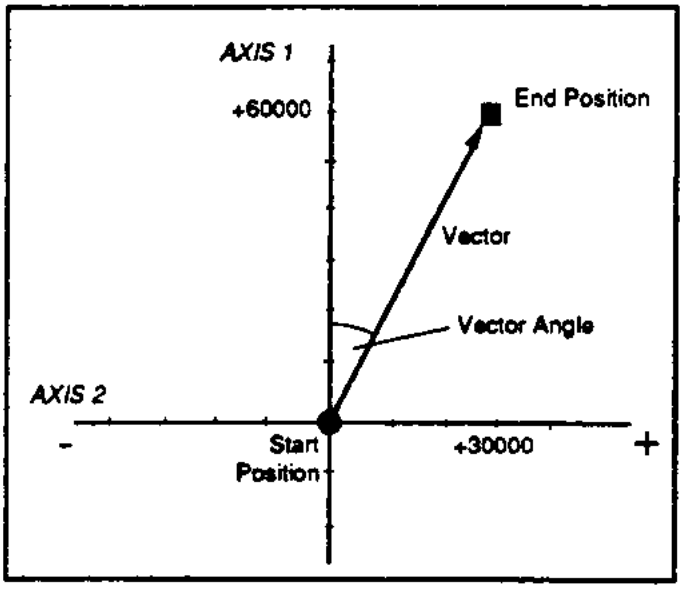


Figure H1-6. Linear Interpolation X/Y Table Plot

Since the relationship between the speeds of the two axes is maintained, the plot of the axis 1 distance moved against the axis 2 move is a straight line. Along this line the resultant combined move accelerates at the rate set by **FDA** to the speed set by **FDV**, decelerating to a stop as the axes reach their programmed distances. This is called vector programming and **FDA** and **FDV** are the vector acceleration and velocity respectively. The distances set the vector angle and vector distance.

Encoder Following Mode

Encoder following is the technique of moving the motor of one axis according to the input from an encoder on a different axis. To use the control module in the encoder following mode, you must connect the *following encoder's* leads to the clock (step) and direction terminals as shown below (see Figure H1-6 for terminal locations). Refer to your encoder's user documentation to determine the lead functions.

Following Encoder's Leads		Motherboard Terminals
CHA+	to	CLK+
CHA-	to	CLK-
CHB+	to	DIR+
CHB-	to	DIR-

Use the **SIM** and **CCS** commands (see Section J) to configure the encoder following operation.

You can use the **CCS** command to configure the CLK+, CLK-, DIR+ and DIR- motherboard terminals as encoder inputs (x1, x2 or x4) or as clock and direction inputs.

The **SIM** command may be used to select the following:

- Normal control module operation (**SIMØ**)
- Encoder following with control module motion commands inoperative (**SIM1**).
- Encoder following with motion control commands operative (**SIM2**).
- Software scaled encoder following (**SIM3**).
- Software scaled encoder following with reversal (**SIM4**).

For **SIM3** and **SIM4** operation, the **CMR** and **CUR** commands define the scaling ratio and the rate can be calculated from the following formula:

Motor (output) rate = input rate x **CMR**/**CUR**

The step distribution can be calculated from the formula for a limited number of steps:

Number of motor steps to move = **INT** (**CMR** x number of pulses received + previous remainder/**CUR**) where **INT** means "take the integer part of". The value is rounded towards zero whether positive or negative. This formula is applied iteratively with the remainder carried forward so that no pulses are lost.

The software scaling requires that when the ratio CMR/CUR is reduced to its lowest terms both the numerator and the denominator must be in the range 1 to 255. This means that at the maximum division, 255 pulses in results in 1 motor step moved and at the maximum multiplication, 1 pulse in results in 255 motor steps moved.

Comparison of Following Features

Hardware Scaling SIM1, SIM2	Software Scaling SIM3, SIM4
The motor always moves by the same or fewer pulses than received. The pulse stream is effectively divided.	The motor can move by more or less pulses than received. The input pulse stream can be multiplied or divided
The ratio is programmed by the RAT command with 16 or 24 bit resolution as a binary fraction	The RAT command has no effect.
CMR and CUR have no effect.	The ratio is programmed by CMR and CUR.
Only available in servo (ESC1) mode.	Available for all drive types.
Superposition onto an additional internally generated indexer motion is available.	Superposition is not available.

Dither

This facility allows a triangular or square wave signal to be added to the torque demand to combat problems on systems that are non-linear in the crossover region such as pneumatic and hydraulic valves. The commands associated with dither operation are :

OSG — Enables/disables dither
OSH — Selects triangular or square wave signal
DTA — Sets dither signal amplitude
DTF — Sets dither signal frequency

They are defined in detail in Section J.

Motion On Initialization

On energizing the drive, a brushless non-hybrid servo motor shaft may move through an angle of up to sixty degrees. The OSI command permits either normal initialization which allows this movement or reduced movement initialization which is designed to protect systems which could be damaged by the normal movement. The reduced movement option is not suitable for some types of load and with this option initialization can take considerably longer than normal initialization.

Program Storage

The user memory is battery-backed up RAM with memory retention of 1,000 hours. The capacity of the RAM is 16K. A write-protect facility is provided. Programmed sequences are stored in variable-length buffers and the total length, including servo and other set-up parameters, cannot exceed 16K.

Motion Program Selection

The system may be set up so that no sequence is executed at power-up. In this case, sequence execution would be initiated from the controller over the RS-232C link.

Alternatively, a single sequence to be automatically executed at power-up can be programmed. After that sequence is executed, control can pass to another sequence or to the RS-232C interface. Sequence selection may also be initiated via inputs. Up to 16 user-defined sequences can be selected for execution in this way.

Motherboard Terminal Functions

The Motherboard front panel is shown in Figure H1-7.

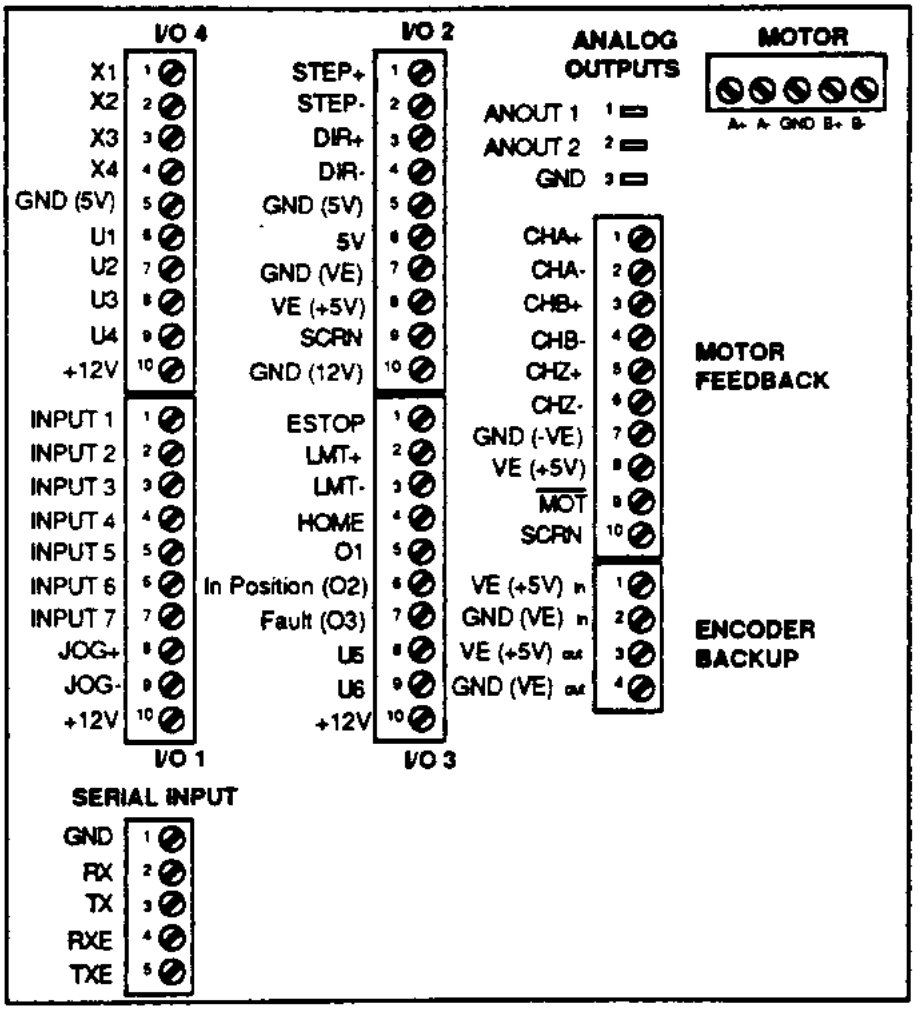


Figure H1-7. Motherboard Terminal Functions

Signal Types

Connector/Pin #	Signal Name	Function	Signal Type
I/O 1			
Pins 1 - 7	INPUT 1 - INPUT 7	User Inputs	A
Pin 8	JOG +	Jog Input	A
Pin 9	JOG -	Jog Input	A
Pin 10	+12V	Power Supply	C
I/O 2			
Pin 1	STEP+ (CLK+)	Step Input	A
Pin 2	STEP- (CLK-)	Step Input	A
Pin 3	DIR+	Direction Input	A
Pin 4	DIR-	Direction Input	A
Pin 5	GND (5V)	Power Supply	----
Pin 6	GND (5V)	Power Supply	----
Pin 7	GND (VE)	Power Supply	----
Pin 8	VE (+5V)	Power Supply	----
Pin 9	SCRN	Shield	----
Pin 10	GND (12V)	Power Supply	----
I/O 3			
Pin 1	ESTOP	Emergency Stop Input	A
Pin 2	LMT+	Positive (CW) Limit	A
Pin 3	LMT-	Negative (CCW) Limit	A
Pin 4	HOME	Home Limit	A
Pin 5	01	User Output	B
Pin 6	02 (In Position)	User Output	B
Pin 7	03 (Fault)	User Output	B
Pin 8	U5	*reserved*	K
Pin 9	U6	*reserved*	K
Pin 10	+12V	Power Supply	C
I/O 4			
Pins 1 - 4	X1 - X4	*reserved*	K
Pin 5	GND (5V)	Power Supply	C
Pins 6 - 9	U1 - U4	*reserved*	K
Pin 10	+12V	Power Supply	C
ANALOG OUTPUTS	Not Used	*reserved*	K
SERIAL INPUT			
Pin 1	GND	RS-232C Com. Connections	F
Pin 2	RX	RS-232C Com. Connections	E
Pin 3	TX	RS-232C Com. Connections	E
Pin 4	RXE	Aux. RS-232C Connections	E
Pin 5	TXE	Aux. RS-232C Connections	E
MOTOR FEEDBACK			
Pin 1	CHA+	Encoder Connections	G
Pin 2	CHA-	Encoder Connections	G
Pin 3	CHB+	Encoder Connections	G
Pin 4	CHB-	Encoder Connections	G
Pin 5	CHZ+	Encoder Connections	G
Pin 6	CHZ-	Encoder Connections	G
Pin 7	GND (VE)	Encoder Connections	H
Pin 8	VE (+5V)	Encoder Connections	----
Pin 9	MOT	Motor Over-temperature	----
Pin 10	SCRN	Shield	----
ENCODER BACKUP			
Pin 1	VE (+5V)	Backup Encoder Connections	----
Pin 2	GND (VE)	Backup Encoder Connections	H
Pin 3	VE (+5V)	Backup Encoder Connections	----
Pin 4	GND (VE)	Backup Encoder Connections	H

Table H1-2. Control Module Signal Types

KEY

Signal Type	Description
A	Optically-isolated inputs referenced to isolated GND. 7V-12V applied at input represents logic 1
B	Open-collector optically isolated outputs referenced to isolated GND
C	Isolated 12V supply from drive
E	RS-232C data signals
F	Control Module 0V (RS-232C)
G	Differential encoder inputs TTL levels relative to pin 8
H	Encoder supply voltage
I	Control module 0V
J	Motor over-temperature inputs for switch to GND
K	Terminal not used
L	Analog monitor output ±2.5V relative to control module 0V

Signal Descriptions

The following are the descriptions of the signals identified in Figure H1-7 and Table H1-2.

Step

When step and direction control is used, the step signal controls the acceleration, deceleration, and velocity of the motor. It is a balanced input.

Direction

The two direction inputs, + and -, are balanced inputs that control the direction of rotation. Positive (CW) rotation results when the + input is positive with respect to the - input. Negative (CCW) rotation results when the + input is negative with respect to the - input.

Inputs 1 to 7

The RS-232C control module reads these optically isolated connections on connector I/O 1 so that the controller can use them. They can also trigger sequences (see commands **IS** and **TRE**). You can use the **SSE** command to set input I5 as a controlled stop line. This input triggers an internal **S** command that interrupts all other indexing activity. You can assign alternative functions to inputs 4 - 7 as follows:

- Input 4 is a sequence select line if **SSF** is set to 1.
- Input 5 is a stop input if **SSE** is set to 1.
- Input 6 clears a pause if **SSB** is set to 1.
- Input 7 is a sequence select line if **OSD** is set to 1.

Jog+ & Jog-

These inputs on connector I/O 1 provide a switch activated method of moving in the positive or negative direction respectively at a constant velocity previously defined using the Jog Velocity (**JV**) command.

ESTOP

The ESTOP connection on the I/O 3 connector is a fail-safe input that needs current to be sourced into it to keep the drive enabled. It provides a hardware-only path to enable the drive. If current is momentarily stopped, the resident RS-232C control module software will suppress re-enabling the drive when the supply returns. The ESTOP input must be physically connected to I/O 3 pin 10 (+12V), preferably through a normally closed emergency stop switch.

+ (CW) & - (CCW) Limits

When you activate the + and - limit inputs on connector I/O 3, you will trigger a controlled stop and prevent further movement in the same direction as the active limit (the convention is positive direction = CW rotation). These inputs need current sourced into them to allow motion in the specified direction. You can disable all limits with the LD3 command. The system will not operate without limit switch connections or use of the LD3 command.

CAUTION

Damage may occur to the system due to excessive travel if you use the Disable Limits (LD3) command.

Home

The home position is a reference position that you can define. It is usually activated by a switch. You can program the RS-232C control module to search for the home position. See the GA, GHF, GH and PZ command descriptions in Section J.

Output O1

If the SSD command value is set to 0, Output O1 becomes a user-defined output. If the SSD command value is set to 1, the output becomes a composite fault indicator that sinks current during normal operation. It also functions as a watchdog so that if the software enters an illegal state, the output will become high-impedance and the drive disables. Refer to the IO, O, and SSD command descriptions.

Output O2

If the SSC command value is set to 0, Output O2 functions as a user-definable output. If the SSC command is set to 1, the output is configured to indicate *in-position*. The user defines the criteria for being in-position with the CIT and CEW commands (see commands IO, O, SSC, CIT, and CEW).

Output O3

This is a user-definable output with no alternative fixed function.

AN-OUT 1

This is an analogue output for position monitoring. The output voltage is from +2.5V (representing a position error of 128 steps lagging the commanded position) to -2.5V (representing a position error of 128 steps in advance of the commanded position). The error corresponding to intermediate voltages is proportional to the full scale.

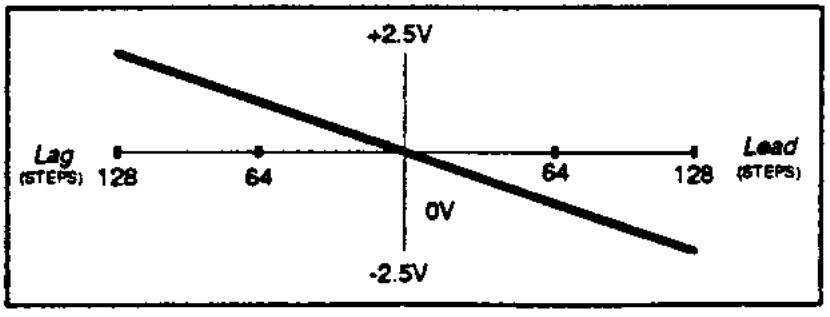


Figure H1-8. AN-OUT 1 Graph

AN-OUT2

This is an analogue output for velocity monitoring. The output is +2.5V (representing a clockwise speed of 256,000 steps/sec) to -2.5V (representing an counter-clockwise speed of 256,000 steps/sec). The speed corresponding to intermediate voltages is proportional to the full scale.

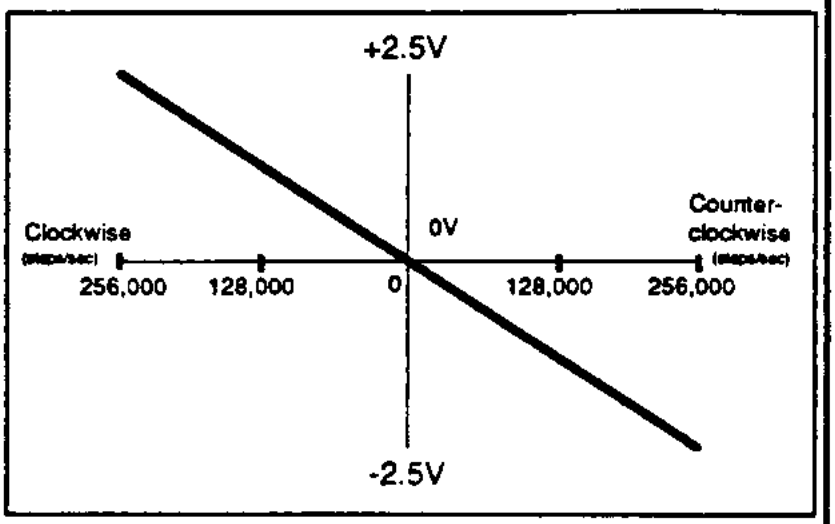


Figure H1-9. AN-OUT 2 Graph

Indexer Inputs

There are two limit inputs: end-of-travel and home. End-of-travel limits must be normally *active* to allow motor motion. If a limit goes *inactive* (current stops flowing to the opto-isolator) while the motor is moving in the direction of that limit, the control module will stop motor motion at a user-defined deceleration rate determined by the **LA** command. The **RA** command tests limit status after a move.

The home input establishes a home reference position. The **GH** command causes the motor to search for a transition on its home input. After finding the home position, you can set the absolute position with the **PZ** or **SP** commands.

Encoder Backup Supply

The purpose of the Encoder Backup Supply facility is to provide an external 5V backup power supply for the control module to retain the motor's position if the power is interrupted. For more information on this facility, refer to Section G.

External Connections

Figure H1-10 shows the general circuit arrangement of the inputs and outputs of the interface. Both circuits are optically isolated to give immunity from noise and transients.

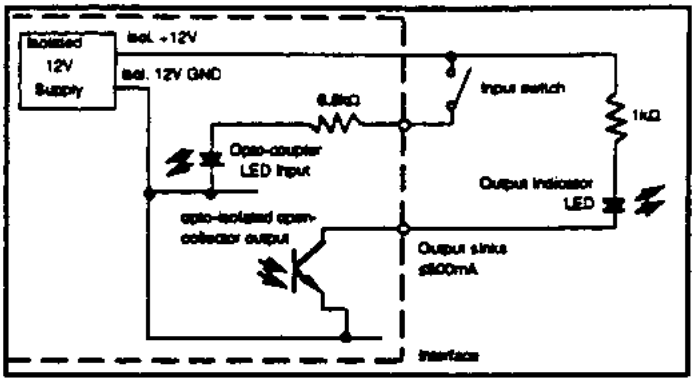


Figure H1-10. Inputs and Outputs—General Arrangement

Step and Direction Inputs

These balanced inputs may be driven from a single-ended output via a differential line driver, such as the National Semiconductor DS8830, that accepts a TTL-level input and provides a balanced output. Each of the two circuits in the DS8830 should have their four inputs connected together and to the input signal if this device is used (see Figure H1-11). Connecting the step/direction source equipment ground to the control module ground should be avoided since this would violate the isolation conditions.

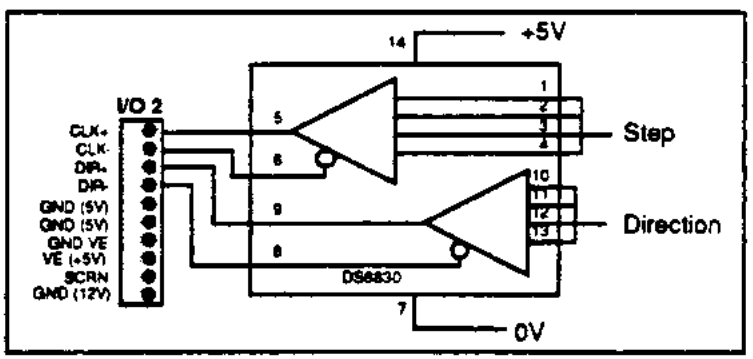


Figure H1-11. Step and Direction Input Connections

Positive and Negative End-of-Travel Limit Inputs

The RS-232C control module is equipped with positive (CW) and negative (CCW) end-of-travel limit inputs. These inputs require a normally-closed, load-activated switch connected to a current source (see Figure H1-12). If either input goes high, motor motion in the corresponding direction will not be possible until the input goes low again. The inputs prevent the load from crashing into a mechanical stop and damaging equipment or injuring personnel. You can disable the limit switches for testing purposes or for rotary systems (which have no end-of-travel requirement) with the LD3 command. When disabled, these inputs are available as user inputs similar to I1 through I7. You can read them with the Input Status (IS) command.

The Home Limit Input

You can initiate the RS-232C control module's Go Home function with the Go Home (GH) command. The direction and velocity in which the motor should search for home must be included in the command syntax. Refer to the GA, SS, SR, GHP and GHF commands.

The home limit input on the RS-232C control module is optically isolated. A normally-open, load-activated switch to a current source is the most common way of detecting the home position (see Figure H1-12). When the control module receives the GH command, it initiates a move in the direction and at the velocity specified, looking for the home limit input state to change.

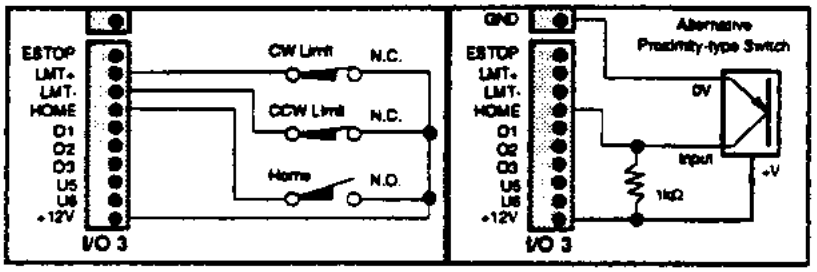


Figure H1-12. Limit and Home Switch Connections

Trigger Inputs

The RS-232C control module has 13 optically isolated inputs, any of which may be used as triggers with the Trigger On Input Equal (TRE) command. The semi-dedicated inputs such as JOG+ and JOG- must have their dedicated functions disabled when used as trigger inputs. They may be used either high true or low true, but you must supply an external 7V to 12V at 12mA per input to turn on the opto-isolators (12V at 0.2A is available on the motherboard). Refer to Figure H1-13 for trigger input connection guidelines.

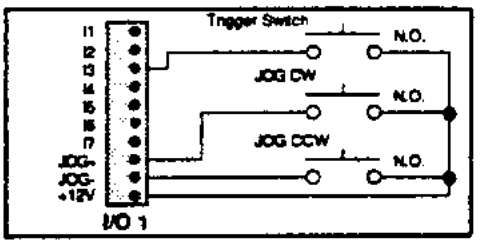


Figure H1-13. Trigger and Jog Switch Connections

± Jog Inputs

The jog switches provide manual control of the motor position. Motor acceleration and speed when the jog switches are operated are set by the JA and JV commands respectively. Normally-open pushbuttons (as shown in Figure H1-13 above) are frequently used for this function.

ESTOP Input

The ESTOP input must have current sourced to it to keep the drive enabled. It is usually connected to the +12V supply via a normally-closed pushbutton (see Figure H1-14). Pressing the pushbutton immediately disables the drive.

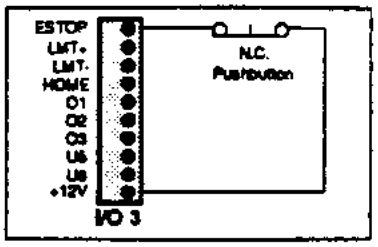


Figure H1-14. ESTOP Input Connection

Sample Wiring Schemes

The following are sample I/O wiring schemes.

Coupling Outputs to Inputs

An output can be coupled to an input on another control module as shown in Figure H1-15. This connection may be required to run sequences on two axes with a need to synchronize the two sequences. Using this arrangement, setting the output to 1 turns on the open collector output transistor, diverting current from the input. With no input current, the input is at 0 level, so a data inversion occurs.

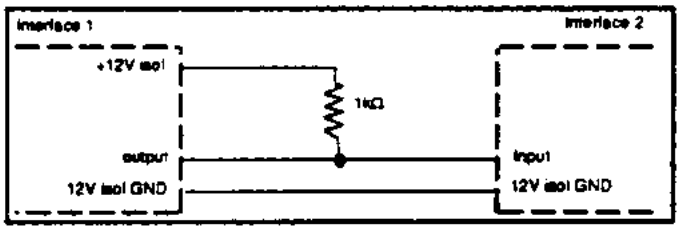


Figure H1-15. Input-to-Output Coupling

Fault Indicator

Figure H1-16 shows a composite fault indicator that requires Output 1 to be configured as a composite fault output (refer to the SSD command). O1 sinks current while no fault exists and switches off if a fault occurs.

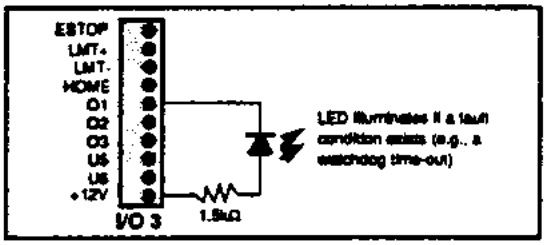


Figure H1-16 . Composite Fault Indicator

In-Position Indicator

Figure H1-17 shows an *in-position* indicator, which requires Output O2 to be configured as an in-position output (with the CCS command). Output O2 sinks current while it is inside of the deadband and switches off when outside it.

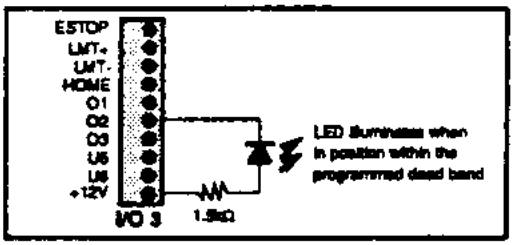


Figure H1-17 . In-Position Indicator

Setting Up Terminal Communication

The RS-232C terminal is used for setting up system operating parameters, requesting the system's status, and tuning. Once entered, a parameter will be held in *volatile* memory until it is actively changed or the control module is reset. It is not necessary to keep re-stating the same parameter value.

When you power up the control module, all parameters are assigned default values, which the **1DR** command will show you. You can change any of these and make them the new default values by entering new set-up commands and typing the Save (**SV**) command. The **RFS** and **RIFS** commands return the set-up parameters to the original factory settings.

RS-232C communication parameters and connection instructions are provided below.

Communication Parameters

The control module communication is fixed as follows:

- Baud Rate: 9,600
- Data Bits: 8
- Parity: No
- Start Bit: 1
- Stop Bit: 1
- Device address: 1-8 (set with control card DIP switches)
- Echo function: all characters received are immediately re-transmitted. You can enable communications with the **E** command and disable it with the **F** command.

Handshaking is not supported in any form. The computer or terminal sending characters to the interface should have its handshaking disabled by either hardware or software (refer to Figure H1-18 for the hardware method). XON, XOFF software handshaking is supported.

Basic RS-232C Connections

The control module communicates with the terminal via a three-wire (**Tx, Rx, GND**) implementation of RS-232C. The **Rx** line requires a minimum voltage swing of ±3V.

You can make *permanent* or *temporary* RS-232C connections as described below:

- For permanent RS-232C connection, use the **SERIAL INPUT** connector on the motherboard (refer to Figure H1-18). A standard cable can be obtained from Compumotor (part number 7967.100).

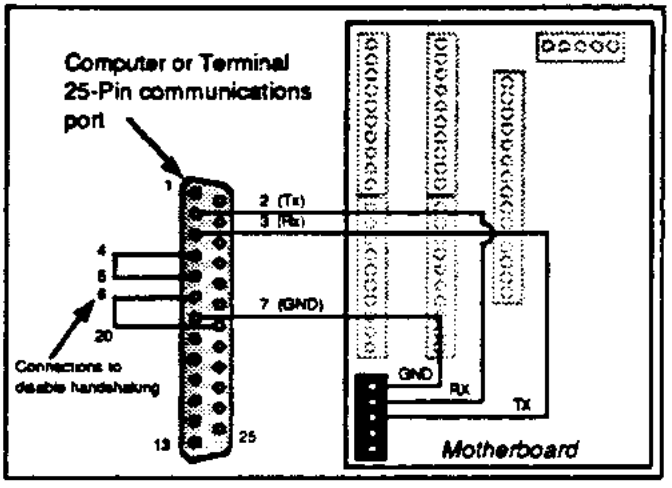


Figure H1-18. RS-232C Permanent Connection

- For temporary connection (for testing or sequence downloading), an alternative serial connector is available on the front of the board (see Figure H1-19 and Table H1-3). To access this alternative connector, you must remove the control module front panel.

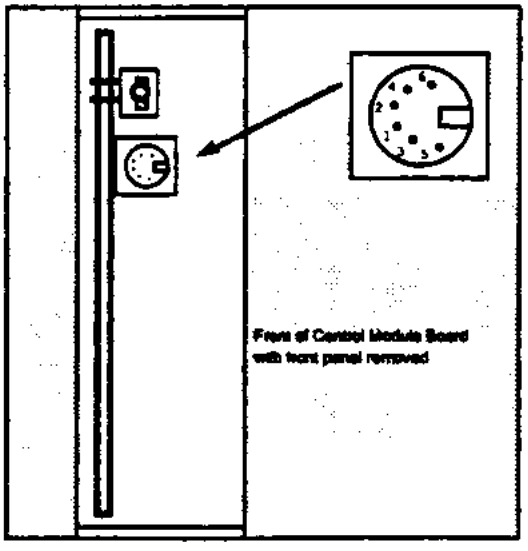


Figure H1-19. Temporary RS-232C Connection

Pin	Function
1	TX
2	RX
3 & 6	OV (GND)
4 & 5	+5V

Table H1-3. Command Input Connector Functions

Serial Interface Status

After power up, the sign-on message **DON'T PANIC TYPE HELP** and the > prompt will appear on the screen, unless you set the RS-232C control module into the terse computer communications mode with the Set Communication Style (**EX0**) command. In any case, the **DR** command will display the status of the RS-232C control module.

If you do not receive an echo from the control card, type **^Q<space>E<space>1DR<CR>**. This should re-establish communication. *^Q represents holding down the control key while pressing the Q key.*

Zero Delay Daisy Chain (ZDDC)

You can use the Zero Delay Daisy Chain (ZDDC) configuration to connect up to eight control modules to a single RS-232C port and keep propagation delays to a minimum. Characters sent by the terminal are echoed back to verify that the character has been correctly received. In the ZDDC, for practical purposes, all devices are connected in parallel. To avoid confusion of data on the return line, only the addressed interface performs the echo back.

Universal commands will be echoed back by the last interface to be addressed. This function defaults to device no.1 at power-up. Each interface is equipped with an additional RS-232C receiver and transmitter to enable successive interfaces to be linked together.

The first interface in the chain can be connected to the terminal via the **SERIAL INPUT** connector as described above in *Basic RS-232C Connections*. The other modules are linked by ribbon cable, in via PL9 and out via PL11 on the inside of the motherboard near the cooling fans (see Figure H1-20).

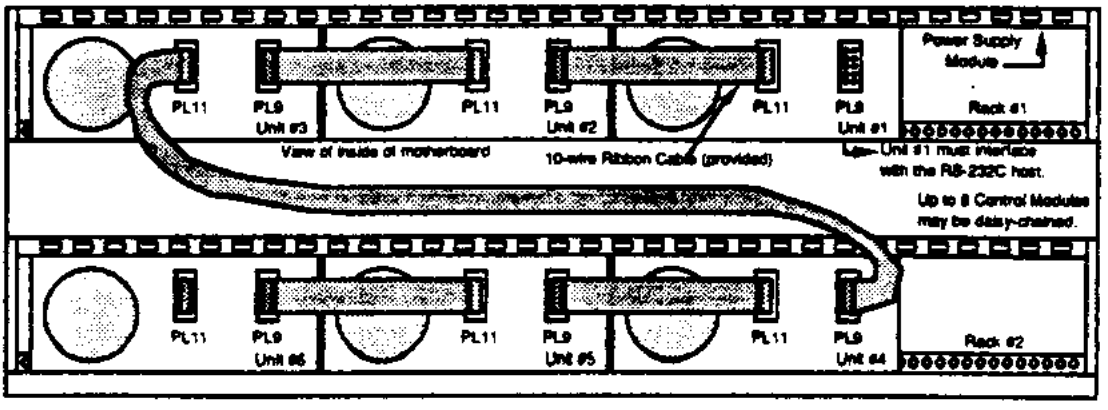


Figure H1-20. Zero Delay Daisy Chain Connections

Control Module Address Jumpers

Each control module in the chain must have a unique address. Normally, Control Module #1 is positioned next to the power supply in the rack and is the first control module in the chain. The control module address is assigned by configuring jumpers LK7, LK8, and LK9 on each control module card in a binary pattern as shown in Table H1-4. The jumpers are located on the control card (see Figure H1-21). To access the jumper, you must remove the control card front panel and pull the control card from the rack. **Be sure to use proper electro-static discharge precautions**.

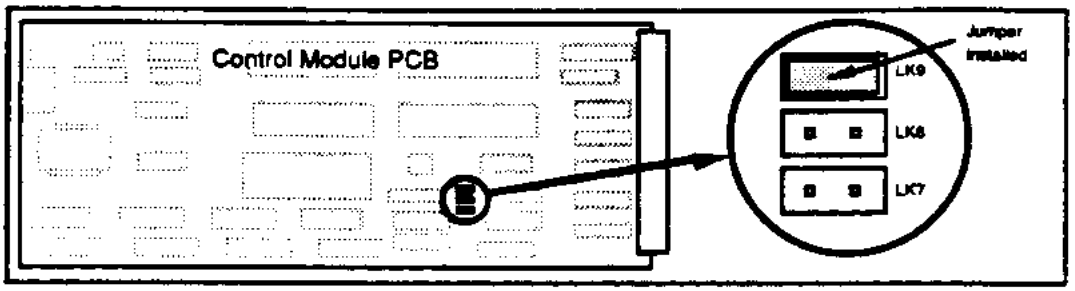


Figure H1-21. Control Module Address Jumper Location

Address	Jumper 7	Jumper 8	Jumper 9
Control Module 1	OFF	OFF	ON
Control Module 2	OFF	ON	OFF
Control Module 3	OFF	ON	ON
Control Module 4	ON	OFF	OFF
Control Module 5	ON	OFF	ON
Control Module 6	ON	ON	OFF
Control Module 7	ON	ON	ON
Control Module 8	OFF	OFF	OFF
ON = jumper installed, OFF = no jumper installed			

Table H1-4. Control Module Address Settings

Servo Tuning

To obtain smooth motor rotation at optimum speed and accurate positioning without overshoot or oscillation, you must tune the gain and damping characteristics of the servo loop. The **TUNET** and **TUNEV** commands provide a self-tuning feature for torque amplifier or velocity amplifier operation, but for some applications the best results will be obtained by manual tuning using servo tuning parameter commands from the *Command List* in Section J. The servo tuning parameters (shown in the servo loop block diagram — Figure H1-22) are described below.

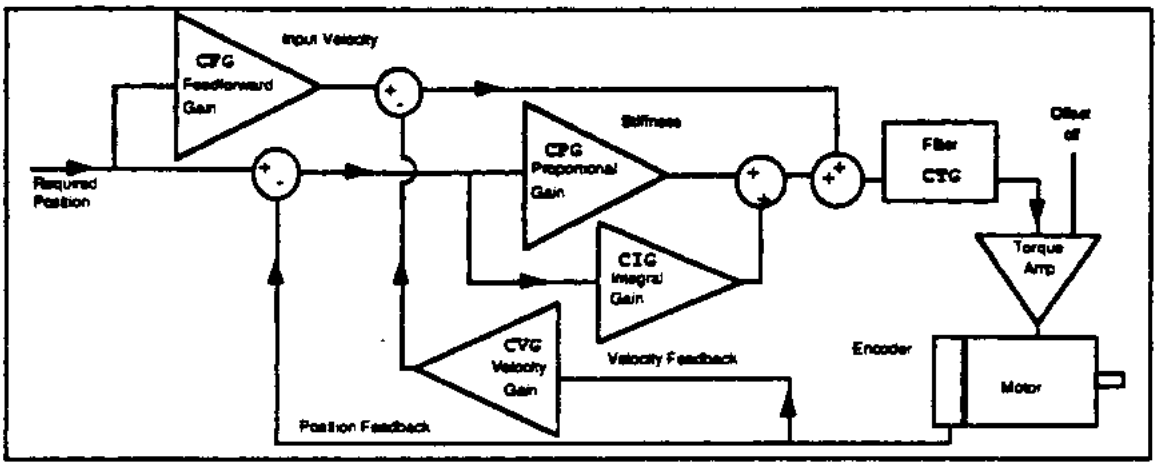


Figure H1-22. Servo Control Loop Block Diagram

Tuning Parameters

The following are descriptions of the parameters that affect tuning. You can examine these parameters by using the **DR** command.

Proportional Gain (CPG Command)

Proportional gain determines the amount of torque produced in response to a given position error. It sets the stiffness of the system and affects the following error (see below). A high proportional gain creates a stiff, responsive system, but results in overshoot and oscillation, which require damping.

Velocity Feedback Gain (CVG Command)

Velocity feedback is a signal that increases with shaft speed. It acts in a negative sense opposing the the proportional action and helping to stabilize the motion. The damping action of velocity feedback allows a higher proportional gain to be used.

Integral Gain (CIG Command)

Proportional action may be insufficient to overcome static position errors due to friction. Integral action accumulates a steady state error until sufficient torque is produced to move the load. It improves overall positioning accuracy but may produce low-frequency oscillation around the commanded position.

Velocity Feedforward Gain (CFG Command)

The opposing action of proportional and velocity gains result in a position error which depends on speed. This is called *following error*. Feedforward gain can be used to offset the following error and improve tracking accuracy. This is important in contouring applications.

Filter Time Constant (CTG Command)

Fast positioning systems need high proportional and velocity gains. By limiting the bandwidth, the digital filter prevents a high gain system from becoming too lively. The filter also serves to average the effects of the digital control loop, reducing jitter at standstill.

Safety Precautions

Wire the ESTOP input to an emergency stop switch that is located in an easily accessible position so that the system can be stopped quickly if instability causes oscillation. You can also cut power with the OFF command, but this takes longer.

Unless you saved data while the motor was enabled, the RS-232C control module will not move or enable the motor when power is applied. If any of the motor or encoder connections have been changed, the first step is to check the integrity and polarity of the motor connections. Proceed as follows:

1. Disable the axis. Type R if you are not sure the axis is disabled. Use the Display Signal (DS3) command to display the feedback position continuously (DS3 status command response should be approximately zero).
2. Turn the motor shaft CW; the displayed value should increase. Turn the shaft CCW and it should decrease. If this does not happen, do not close the loop with the ON command at any time, since the feedback will be positive.
3. Press the carriage return <cr> key to restore the > prompt.

Setting Up Servo Parameters

The default tuning parameters set up in your system when you receive it will allow the system to run but for optimum performance you should set up servo parameters in accordance with your type of application as follows.

Tracking Applications

For tracking applications (PID tuning), where minimum following error is required, velocity feedforward should be used to improve the tracking accuracy. This can be set using the CFG command or the CDG command can be used to set feedforward gain and velocity feedback gain to the same value.

Point-to-point Positioning Applications

For fast point to point positioning applications (PIVF tuning), a high proportional gain (CPG) damped by velocity gain (CVG) should be used.

HELP11

A guide to servo loop parameter settings for a fully damped non-oscillating response is provided in **HELP11**. This uses the maximum low speed motor torque entered by the CTQ command and the total moment of inertia entered by the CJL command to calculate suggested settings for the loop parameters.

First enter the values of CTQ and CJL (Motor + load inertia) for your system, then use **HELP11** to display recommended servo loop parameter settings calculated from the data you have entered. For example, if 3.6 is entered for the CTQ command to indicate a maximum torque of 3.6 Nm available and 1.6 is entered for the CJL command indicating a total inertia of 1.6 kg/cm^2 , the following table will be displayed:

Gain (CPG)	Velocity Gain (CVG/CFG/CDG)	Stiffness mNm/step	Error @ maximum torque (fb steps)	Doubled T Constant (msecs)	1/2 moved time (msecs)	Closed l'p bandwidth (3db) (Hz)
92	57	10	360	5	8	20
46	40	5	720	7	12	14
23	29	2	1400	10	17	10
11	20	1	2900	14	24	7
5	14	0	5700	20	34	5

Table H1-5. **HELP11** - Servo Setup Table

Each successive line of the table corresponds to a less highly tuned servo system with reduced stiffness and speed of response. These reduced values may be used in matching the servo to lower specification mechanics.

The following describes in more detail the terms used in the table.

VELOCITY GAIN

The settings of CVG, CDG and CFG depend upon the application as described above. For the figures in the rest of the line of the table to apply, the CDG or CVG setting should be as in column 2.

For PID applications, setting CDG to the figure quoted in the table will automatically set CVG and CFG to the same value.

STIFFNESS

Stiffness is the restoring force per unit of displacement. The force is dependent on the number of steps displaced from the demanded position. For example, a displacement of 100 steps with a stiffness of 10 mNm/step would produce a restoring force of 1 Nm.

ERROR @ MAXIMUM TORQUE

As the axis is increasingly displaced from the demanded position, the restoring torque increases at the quoted stiffness until the maximum available torque is applied. The Error @ Max Torque is the number of steps of displacement required to produce a restoring force of the maximum available torque.

DOUBLED T CONSTANT

When the damping is critical, the response of the servo loop is that of two simple filters in series, each having a time constant T. The value of T is quoted in the table.

1/2 MOVED TIME

The following is an example move using servo values taken from the top line of the **HELP11** table to show the 1/2 moved time. The move is at maximum acceleration and velocity and it is initiated by the command string:

```
D1ØØ AØ VØ G
```

Figure H1-23 represents instantaneous position plotted against time for the resulting servo movement .

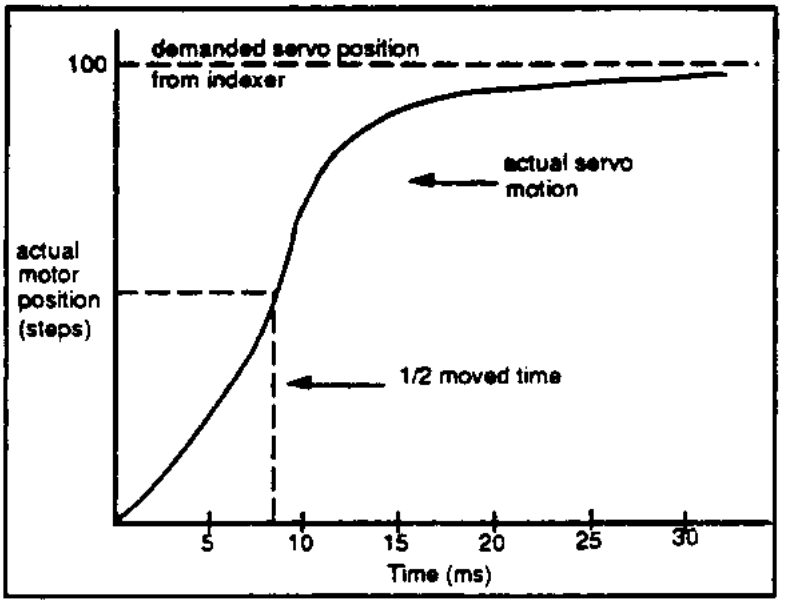


Figure H1-23. Servo Response Curve Example

The 1/2 moved time is the time taken to complete half of the move, 50 steps in this example.

The move will be half complete in the time 1.7T (column 6 of the table) or 1.7 x 5 = 8 milliseconds (approximately).

The move will be 96% complete at time 5T. In the example, the motor will have moved 96 steps in 25ms.

BANDWIDTH

The bandwidth of the servo system is the range of frequencies at which a sine wave move pattern input will be followed within 3db by the actual motion of the motor. Figure H1-24 shows the sine wave position demand input and the graphical representation of the motor movement.

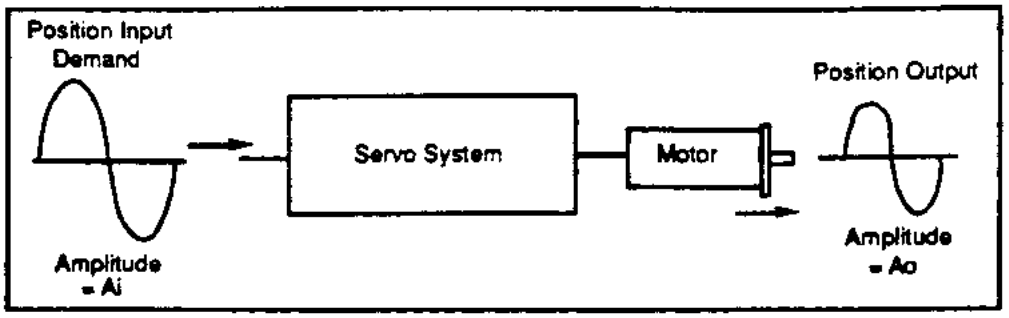


Figure H1-24. Servo System Bandwidth

As the frequency of the sine wave of demanded positions at the input (Ai) increases, the ability of the servo system to follow is reduced and the amplitude of the sine wave of the resulting output positions (Ao) decreases. The frequency at which it is reduced by 3db is the upper limit of the bandwidth.

Manual Tuning Procedure

Type the following commands to disable the output (refer to Figure H1-22): **CPG0 CPG0 CVG0 COFF0 CIG0 CTG0.** Check the result of these commands by issuing the DR command.

When the axis is enabled, it will be open-loop with a slight torque demand due to amplifier offsets. This may cause slow motion (*drift*) of the motor. You may have to restrain the axis and be prepared to switch the unit off, or issue the **OFF** command. In any case, the **CPE** command sets an automatic software limit on the distance the motor will drift.

Position Error Window Setting

The drive is disabled to protect the system if the position error goes beyond the position error window. The window can be set to a low number of steps if required to avoid damage due to overshoot or oscillation.

The value of **CPE** command defines the size of the window in steps. During tuning, **CPE** is normally set to 4,000 steps (i.e., 1 rev either way at the usual 4,000 steps/rev). If in doubt, set this parameter to a lower value. The effect of going beyond the set error limit will be to automatically disable the motor.

Type the Report Servo Errors (**RSE**) command. This may show quite a list of errors. **Drive disabled by software command, error limit exceeded,** and **failed checksum of backup ram** are all acceptable errors at this point, but the **drive fault** error should not come up. If it does, check the LED on the drive front panel and try turning power off and on. **DO NOT PROCEED IF A DRIVE FAULT PERSISTS.**

Enabling the Axis

Enable the axis with the **ON** command. You may need to restrain the axis to prevent it from drifting out of the error limit and disabling. If it disables, you can restart it by typing **ON**.

Cancelling Drift

Cancel the drift with the COFF command. Negative values stop CW drift; positive values stop CCW drift. Values up to a few thousand are acceptable (the maximum is 32,000 corresponding to full torque demand).

Check that increasing the positive value of COFF tends to turn the motor CW, and increasing the positive value of COFF tends to turn it CCW. If this does not happen, do not proceed, as the loop will have a net positive feedback and will oscillate (although probably slowly) when the loop is closed.

Closing the Loop

Set a small position error limit (2,000 steps for example) so that if oscillation occurs during tuning, the resulting overshoot producing large position errors will exceed the limit set by CPE and shut down the drive.

If necessary, first move the axis to a good starting point, then type ON to enable the motor and CPG1 to close the loop. If a slight vibration appears at this point, it indicates positive feedback. This should only occur if you have changed the motor or encoder connections. Taking care, displace the axis slightly and then let go. You should experience a slight spring effect and the axis should return to its original position approximately. The closed loop is now working.

Coarse Tuning the Loop

Attach a nylon cable tie to the shaft of the motor. The loose end can be used as a coarse indicator for overshoot and oscillation. Use the following command string to produce a short, sharp rotation of the shaft:

```
A0 V0 D50 G
```

Observe the movement. Watch the cable tie for overshoot and oscillation. Repeat the movement, increase the value of the CPG command using the settings suggested in HELP 11 until overshoot occurs.

Balance the increasing gain with damping by setting the value of the CVG command to 10, 20, 40. . . Increase it until the oscillation and overshoot indicated by the cable tie ceases.

Fine Tuning

Continue the balancing monitoring the overshoot at the terminal by typing DPE G. The carriage return <cr> should be entered quickly after typing the space after G. This will stop the position error display with the final positioning readings on screen to allow you to analyze them. Overshoot is indicated by the position error reading passing through zero at the end of the movement, then returning through zero in the opposite direction, oscillating with decreasing error readings about the zero position before finally settling within the position error window (refer to Figure H1-25). The damping should be increased to the point where overshoot disappears. Too much damping causes sluggish movement.

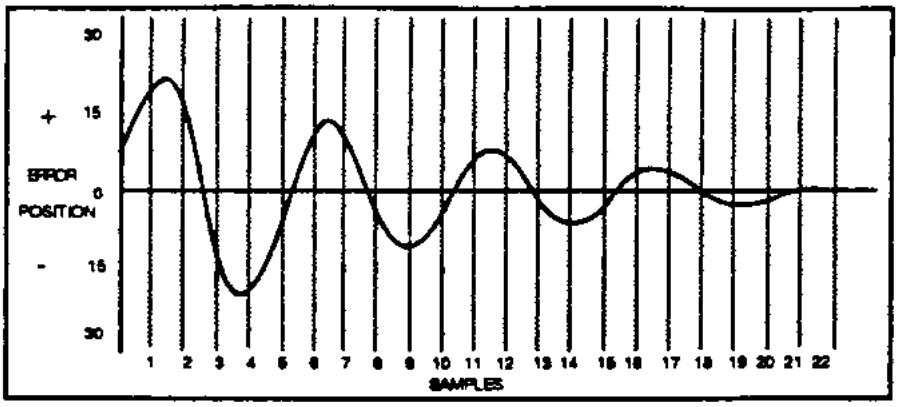


Figure H1-25. Overshoot During Positioning

The following table shows the Display Position Errors (DPE) command's readings for the final positioning graph of Figure H1-25. This is only an example to illustrate the general principles involved in tuning. In practice, the resulting figures may differ considerably from those given, but they will follow the general pattern shown. The overshoot shown here is more severe than might be generally expected.

Samples	Readings
1 - 9	23 22 -17 -20 -5 14 12 -7 -10
10 - 18	0 6 4 -3 -4 -2 4 3 -2
19 - 22	-3 -1 0

Table H1-6. DPE Readings During Final Positioning

Notes on Tuning

The CPG command should never be set more than 20% greater than the CVG command.

An increase in gain is always best matched by an increase in damping. When this is no longer true, you have reached the optimum values. A useful guideline is that an increase of times X in gain is best matched by an increase of times square root X in velocity feedback (think of this as an X% increase in gain matches an $^{X}/_{2}$ % increase in damping (e.g., 30% more gain matches 15% more damping). This will keep the shape of the vibration the same and just increase its frequency (you are holding a quantity called the *damping ratio constant*).

The rule for speed or frequency or bandwidth is the same. An increase of times X in gain produces an increase of times square root X in the speed or frequency (e.g., 20% more gain means 10% faster).

It is also useful to know that the power required increases as the cube of the speed and that the torque required increases as the square of the speed. In summary, to position in 90% of the time, 10% faster means:

- You must set 10% more velocity feedback (CVG).
- You must set 20% more proportional gain (CPG).

More torque and power will be required and these must be available.

Servo Self-Tuning

A facility for self-tuning the servo is included in the control module software. The commands **TUNET** and **TUNEV** (see detailed description in Section J) are used to tune the positioning servo as a torque amplifier or as a velocity amplifier respectively. The servo parameters are set at the time that they are executed and there is no continuous on-going adjustment (known as *adaptive control*).

Tuning is for the following characteristics:

- Stability — does not oscillate as it moves or settles
- Maximized stiffness and accuracy — deflects as little as possible in response to external forces
- Maximized bandwidth and speed of response — settles to the commanded position as quickly as possible at the end of a move
- Does not overshoot the end position in a move
- Minimal sound during motion or when holding position
- Tracks a profile of constantly changing position or velocity with as little error as possible
- Resistant to change as friction and other load characteristics vary from time to time, machine to machine, application to application and with production variations
- Operates for smaller loads than that used for tuning down to no load.

There are a number of other move related requirements:

- Make full use of the torque and power available so that the longest, fastest move possible can be made
- Perform equally well for fast or slow motions and for short or long moves
- Minimize heating effects in the motor and drive

It is not correct to adjust the servo loop parameters to meet these requirements as the general servo performance would be adversely affected. They are best met by adjusting the move profile (**D**, **V** and **A** values) and possibly the feedforward **CFG** value whilst watching for saturation effects.

If the application is purely velocity control with no requirement for positioning, then the tuning is best carried out manually. The self-tuning settings will work, but they will not produce the best achievable results.

Section J. Software Reference

Section Objectives

This section provides full information on programming the system to perform the required user functions. The command language is described in detail and in quick-reference form at the end of the chapter.

Communicating with the RS-232C Control Module

The control module is designed to work in one of two modes:

- Human Interactive mode
- Computer Host mode

The **EX1** command enables the Human Interactive mode and the **EX0** command enables the Computer Host mode.

In the Human Interactive mode, the control module responds to a dumb terminal in an intelligent, user-friendly manner. This mode provides help facilities and error messages as the errors occur.

In the Host Computer mode, all unsolicited output to the RS-232C is suppressed so that communication may be handled by software that resides in the host computer.

Buffered/Immediate Commands

All commands annotated with * can be prefixed by the character B, in which case they become buffered rather than immediate commands.

Buffer Capacity

Any combination of individual commands and command groups can be entered in the buffer until the total number of characters currently stored (including the delimiters) equals 2,000. The control module uses a first-in/first-out, serial buffer. As the commands are read from the buffer, additional commands can be entered to replace them. Therefore, the possibility exists that a command set could actually consist of more than 2,000 characters. Examples of command sets that might be used for different applications are presented later.

The sequence buffer is different from the command buffer in that it can be copied to battery-backed RAM using the **SV** command; consequently, its contents are saved when power is removed from the system (turned off). Since the command buffer is not battery-backed, its commands are lost when power is removed.

Individual Commands

An individual indexer command controls a single parameter, function, or action such as acceleration, velocity, position, time delay, pause, loop, go, etc. There are two classes of individual commands, immediate and buffered.

Buffered Commands

Buffered commands are received by the control module and stored (in the command buffer) if the RS-232C control module is not free to execute them. Stored commands are executed as soon as the RS-232C control module gets to them, in the order that they were received.

Immediate Commands

Immediate commands are executed as soon as the control module receives them. They have priority over whatever operation is in progress. These commands include stop commands that clear the command buffer and status request commands, which do not affect the command buffer.

Individual commands are variable in length. They can consist of one or more letters with a delimiter (carriage return or space character) and one or more letters and numbers with a delimiter. Each command is entered as a character/delimiter combination. Some commands include a sign (±) to denote direction of motion. The number of characters used depends on the type of command.

Some typical commands are shown below:

Command	Description
1E	Enable communications on address #1
S	Stop
A10	Set acceleration to 10 rps^2
V3	Set velocity to 3 rps
D46000	Set distance to 46,000 steps in the cw direction

When two or more individual commands are entered on the same line, they are separated by spaces. Multiple command entries are shown on a single line of your terminal screen. The example below shows a set of individual command entries with space delimiters.

```
MN  A10  V2  D25000 L10  G  M
```

If spaces are used as the delimiter and a large number of multiple command entries are made, you could exceed the ability of your terminal to display characters on a single line (80 characters per line is a typical value). When the carriage return is used as a delimiter, the cursor returns to the beginning of the next line each time the carriage return is pressed.

In interactive mode (EX1), the RS-232C control module echoes carriage returns with a carriage return line feed . If you get double line spacing, you must turn off your terminal's auto line feed function. In computer mode (EX0), only the characters received are echoed.

The following example shows the effect of using a carriage return as the delimiter when the terminal is not in auto line feed.

```
MN A10 V2 D25000 L10 G
```

Multiple Control Modules

When multiple control modules are on the communications line, commands like the previous example are executed by all control modules on the line. To send commands to a single control module, the device address of that control module should be put in front of the instruction. The device address is set using jumpers on the individual control module.

Example

Three control modules are on an RS-232C ZDDC (zero delay daisy chain). They are sent the following commands:

Command	Description
MN	Set indexer to normal mode
A10	Set acceleration to 10 rps^2
V10	Set velocity to 10 rps
1D25000	Set distance to 25,000 steps (unit 1)
2D50000	Set distance to 50,000 steps (unit 2)
3T1	Wait for 1 second (unit 3)
3D100000	Set distance to 100,000 steps (unit 3)
G	Execute moves

Unit #1 moves 25,000 steps, unit #2 moves 50,000 steps, and unit #3 waits 1 second and moves 100,000 steps. All three units use the same acceleration and velocity parameters.

Because multiple control modules receive characters in parallel, universal commands are always echoed by the last addressed axis (not by all control modules). An addressed axis continues to echo characters up to and including the next address (i.e., the 2 of 2D50000 of the above example is echoed by axis #1 whereas D50000 is echoed by axis #2). After power-up, axis #1 is the default axis.

During sequence download to one axis in a ZDDC chain, all of the other axes must have their communication switched off with the F command. The alternative is to make each command in the sequence device-specific by adding the device address.

Start Move Commands

The RS-232C control module reads and stores individual commands in the order they are entered. Each command is read and executed in order. The Go (G) command is the principal buffered command that initiates shaft motion. When the buffer reads a Go command, the motor moves in accordance with the move parameters that reside in the RS-232C control module at the time the Go command is read.

When the Go command is read, the indexer incorporates any motion parameter command that was encountered prior to the Go command. For example, the A command would be combined with the Go command to change the acceleration value as follows:

Command	Description
A123	Sets acceleration to 123 rps^2
G	Initiates the move (Go)

This command instructs the indexer to change the acceleration value to 123 rps^2 and then execute the move using the last mode, velocity, and distance values that were previously entered.

When the RS-232C interface is initially activated, a series of commands of the form MN An Vn Dn G is required. These commands set up the initial operating conditions for the RS-232C control module. After the indexer has been initialized, single move parameter commands can be used.

If only G is entered, the indexer will repeat the previous motion pattern. For example, a string of Go commands would instruct the indexer to move in the same direction four times.

```
G G G G
```

There is no need to re-enter all the shaft motion commands to change one of the variables. If one or more of the motion parameters is changed by a command entry and the Go command is then entered, the prior pattern will be repeated using the new motion parameter(s) that were entered prior to the latest Go command.

Command	Description
A14	Set acceleration to 14 rps^2
G	Execute move (Go)
V2.6	Set velocity to 2.6 rps
G	Execute move (Go)
D-27634	Set distance to 27,634 ccw steps
G	Execute move (Go)

The previous example would change the acceleration to 14 rps^2 and move, then change velocity to 2.6 rps and move with 14 rps^2 as the acceleration value, then change position and move again using 14 rps^2 and 2.6 rps as the acceleration and velocity values.

The Continue C command is used following execution of a Pause command (PS or U). The Continue command allows execution of the next command waiting in the command buffer (if any command is stored).

Loop Commands

The Loop command allows a cycle to be repeated continuously (L) or a given number of times (Ln) up to 65,535. The End-of-Loop command (N) indicates where the loop ends. The End-of-Loop command can be used to indicate that the RS-232C control module should proceed with further commands after the designated number of loops has been executed, or in combination with the Y command to indicate where execution is to stop. The U command may be used to temporarily halt Loop execution, the C command will then cause the loop to resume execution.

Stop Move Commands

The Stop move commands are presented in order of severity of response. Software reset (Z) command is the most abrupt and severe stop command you can use. The Stop at end of current loop command (Y) is the least abrupt and severe stop command.

Controlled Stop Command—S

In the Preset or Continuous mode, this command decelerates the motor to a stop at the last used acceleration rate.

An S command will always cause a deceleration to velocity zero at the last used acceleration.

The S command clears any remaining commands in the command buffer unless prevented from doing so via the SSH command.

Normally, the motor is decelerated to a stop at the same rate as it was accelerated. A different deceleration rate may be programmed in CONTINUOUS mode only by calling for a velocity of zero rev/sec with a new acceleration and executing a V0 rather than an S command. The following is an example:

Command	Description
MC	Set indexer to continuous mode
A1	Set acceleration to 1 rps^2
V10	Set velocity to 10 rps
C	Execute move
A100	Change acceleration to 100 rps^2
V0	Change velocity to 0 rps
C	Execute A and V changes (motor stops)

Controlled Stop Command—LS

The LS (Limit Stop) command will decelerate the motor to a stop at the deceleration rate defined by the LA (Limit Acceleration) command. This will guarantee a quick stop for any move.

Stop at End of Current Loop—Y

The Stop at end of current loop command (Y) (also the *stop sequence* command) will not halt processing until command processing reaches the character N at the end of the command loop. At that time, the indexer will execute the next command in the buffer after N, if any. The command loop cannot be restarted without re-entering the commands.

Kill Command—K

This command stops RS-232C control module commands to the motor. In addition, it terminates a loop, ends a time delay, and aborts a command sequence download in progress (XD command). The command buffer is also cleared.

Software Power-On Reset—Z

The Z command is equivalent to cycling the AC power to the indexer; that is, it disables the communications interface and returns all internal settings to their power-on values. The command buffer is cleared. Like the K command, Z causes an immediate cessation of output pulses to the motor. Z also disables and resets the drive ready for re-initialization at power up.

When the Z command is used, the indexer is busy for up to 2 seconds and will ignore any commands.

Pause Commands

There are four types of pause commands available as follows:

- PS: Pause and wait for continue command C to continue. A common use of the Buffered Pause PS is to hold execution of individual commands in a *command loop* until the entire loop has been loaded.
- T: Pause in processing commands for a given number of seconds and then continue.
- U: Pause now and wait for continue. This is an immediate command.
- TRE: Pause in processing commands until the designated TRIGGER input pattern is present.

  Triggers are used to synchronize RS-232C control module operations with external events. They can be used to implement a *handshaking* function with other devices. See command details for exact syntax.

Status Request Commands

All status request commands result in transmission by the RS-232C control module. To prevent multiple indexers from all responding at once, the status request commands are given the classification *Device Specific*, meaning that the device address of the responding RS-232C control module must be placed in front of the command. No RS-232C control module will execute a Device Specific command without its device number.

The use of status request commands must be conducted in an orderly fashion. Commands should only be issued when the host is ready to read the response. New commands should not be sent until the response is received. In particular, after a buffered status command, an immediate status command should not be sent until the response has been received by the host. If this procedure is not followed, the command responses will be intertwined, rendering the information useless.

There are two status commands that can be used to request position: **DPA** and **PR**. The **DPA** command will report how many steps the motor has moved relative to the absolute zero position. This position is calculated by summing the total number of moves commanded since the control module was at the absolute zero position. If position reports are needed that are relative to the beginning of each individual move, the **PR** command should be used.

Drive Energise/De-energise Function

Energizing and de-energizing the drive is subject to the commands given and error status as follows:

The drive will energize if:

- The **ST0** or **ON** command is given when the ESTOP, WATCHDOG, FOLLOWING ERROR and EXCESS TORQUE DEMAND signals are inactive.

The drive will de-energize if :

- The **ST1** or **OFF** command is given.
- The ESTOP or WATCHDOG signals become operative.
- A FOLLOWING ERROR or an EXCESS TORQUE demand occurs.

The de-energizing signal must be restored, then the **ST0** or **ON** command must be given to re-energize the drive.

Home Facilities

The Go Home (**GH**) command instructs the motor to rotate in the direction and at the speed specified until the home position is reached. In practice, it continues at little beyond this point, but you can locate the home position accurately by using the following series of commands:

```
MPA  D0  G
```

The **GH** command always makes the home position zero. You can use the **SP** command to change the accurate home position from zero if required.

If the home position is not found in the direction given in the GH command, the limit switch position will be reached and the switch will be operated. The motor will reverse and seek the home position in the opposite direction. For recognized home detection, the home switch must always be activated from the same direction of motor rotation (shown by the arrow in Figure J1-1). Two limit switch reversals may occur if the direction given in the GH command is away from the home position (GH+ in the illustration).

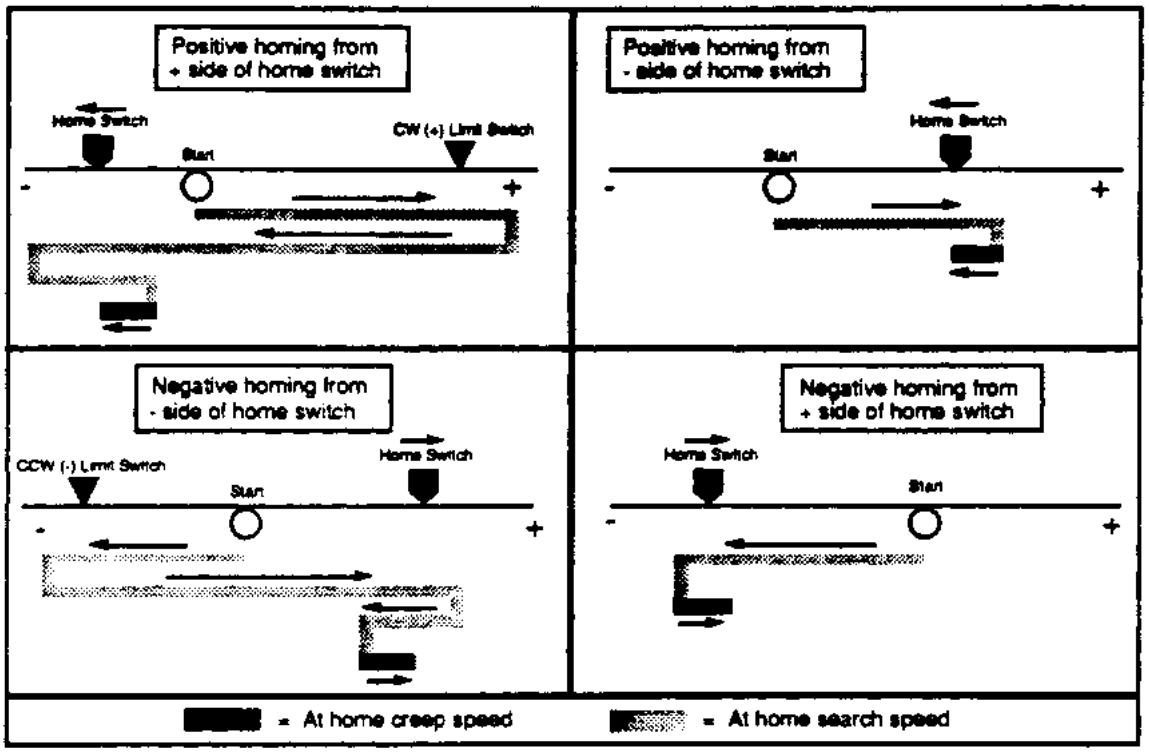


Figure J1-1. Home Switch Operation

Operating Modes

The RS-232C Control Module can be operated in several different operating modes. These modes, identified below, are described in detail in Section H, *RS-232C Control Module*.

- Normal (Preset)
  - Incremental
  - Absolute
- Continuous
- Speed Change
- Registration
- Asynchronuous
- Synchronuous
- Linear Interpolation
- Encoder Following
- Dither
- Motion on Initialization

Basic Programming Guide

This section uses examples of simple programs as an introduction to System 7 programming. This section will give you an insight to the usage of the command language. The examples become progressively more complex, so start with Example 1 and work through the entire set.

In each example, the System 7 configuration is for two axes. The first axis is an AC brushless servo motor with a 1,000-line/rev encoder. The second axis is a hybrid motor that has a 500 line/rev encoder. The sample configuration is shown in Figure J1-2. 4X decoding is used on both axes. The user resolution for motor #1 is set to 200 steps/rev and motor #2 is set to 500 steps/rev. Tuning the servo loop is discussed in the *Servo Tuning* section. We will assume that tuning has been completed for both axes.

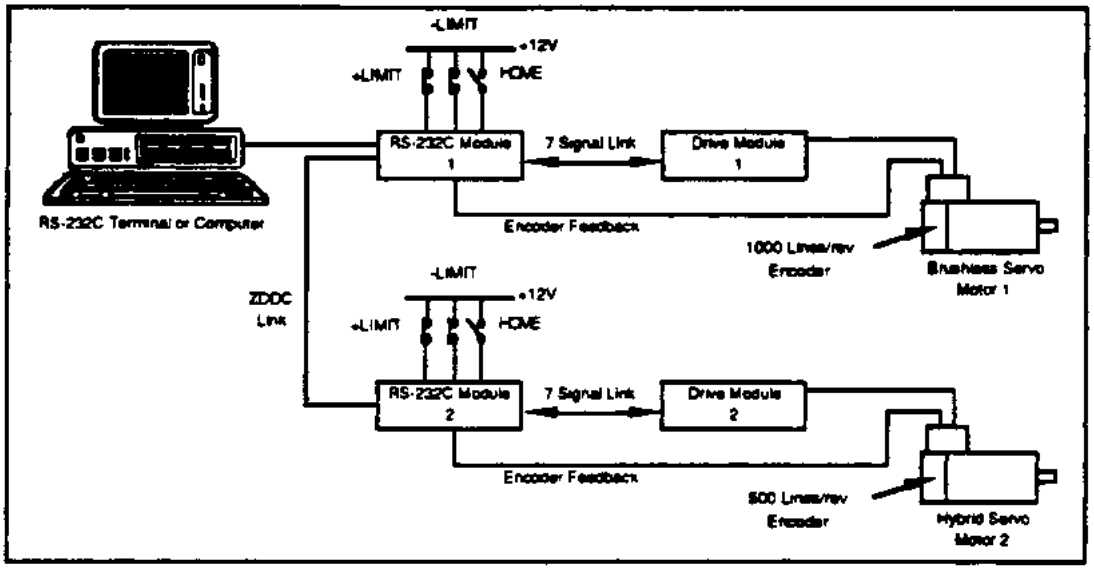


Figure J1-2. System Used in Example Programs

Example 1

In this example, motor #1 must perform two revolutions in a CW direction, then motor #2 must perform 3.5 revolutions in the CCW direction. The following commands instruct the axes to perform this movement:

Command	Description
MN	Set move mode to normal
1A10	Set acceleration motor #1 = 10 rps^2
1V5	Set velocity motor #1 = 5 rps
1D400	Set move distance motor #1 = cw 2 revs
1G	Start move - motor #1 moves 2 revs cw
2A20	Set acceleration motor #2 = 20 rps^2
2V10	Set velocity motor #2 = 10 rps
2D-1750	Set move distance motor #2 = 3.5 revs ccw
2G	Start move - motor #2 moves 3.5 revs ccw

Description

Commands 2 - 5 cause motor #1 to accelerate at 10 rps^2 to a velocity of 5 rps travelling CW for 400 steps. Commands 6 - 9 set motor #2 to accelerate at 20 rps^2 to a velocity of 10 rps. travelling CCW for 1,750 steps.

Example 2

This example is concerned with X and Y positioning using axis #1 to drive the Y positioning and axis #2 to drive the X positioning. The home switches on each axis are 1.000 steps CCW from the zero position. The motion requirements are as follows:

1. Move to position zero (Y=0/X=0) and wait 1 second
2. Move to position 1 (Y= 20.000/X=13.000) and wait 5 seconds
3. Move to position 2 (Y=5.400, X=15.300)and wait 10 seconds
4. Return to position zero (Y=0/X=0)

Figure J1-3 is a diagram of the required moves. A program to complete this operation is shown below.

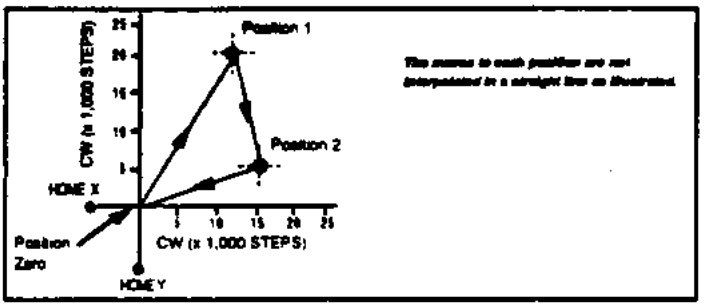


Figure J1-3. Example 2 Moves Diagram

To perform this function we will couple the in-position output (O2) of each axis to Input 1 on the other axis. This allows us to confirm that both axes are in position before the next command is executed. A series of commands to complete this operation is as follows:

Command	Purpose
`ST0`	Enable both drives
`SSC1`	Set O2 of axes as in-position o/p's
`MPI`	Select incremental positioning
`A10`	Set acceleration to 10 rps^2
`V10`	Set velocity to 10 rps
`GB`	Send both axes to home position
`1D10000`	Set axis 1 to go to zero position
`2D10000`	Set axis 2 to got to zero position
`G`	Both axes move to zero position
`TRE_1`	Pause until axes are in position
`T1`	Hold zero position for 1 second
`1D20000`	Set axis 1 to move to position 1
`2D13000`	Set axis 2 to move to position 1
`G`	Both axes move to position 1
`TRE_1`	Pause until axes are in position
`T5`	Wait 5 seconds at position 1
`1D-14600`	Set axis 1 to move to position 2
`2D2300`	Set axis 2 to move to position 2
`G`	Both axes move to position 2
`TRE_1`	Pause until axes are in position
`T10`	Wait for 10 seconds at position 2
`1D-5400`	Set axis 1 to return to position zero
`2D-15300`	Set axis 2 to return to position zero
`G`	Both axes return to position zero

Sequence Programming

A sequence is a series of commands that is executed in order whenever the sequence is run. Immediate commands cannot be stored in a sequence just as they cannot be stored in the command buffer. The only *sequence legal* commands are buffered commands.

The control module has a section of non-volatile RAM totalling 6,400 characters allocated for storing up to 16 sequences. The sequence buffers are of variable length, so they can store long sequences of up to 6,400 characters or several shorter ones, provided that the total length of all sequences does not exceed the allocated 6,400 character space.

During normal operation, a command can specify the execution of any sequence stored in memory. Sequence identifiers: number from 1 - 16.

When a sequence is downloaded the RS-232C control module automatically calculates a checksum for the sequence and stores the length of the sequence and its checksum in an internal directory.

You can use the **XSS** command to verify the existence of a sequence.

If you use a power-on sequence, the sequence and its sequence number are stored in nonvolatile RAM. A power-on sequence execution failure is indicated by flashing the LED.

Programming a Sequence

To download a sequence, first send a special start download command (XD), which clears the command buffer. All subsequent commands are stored in the command buffer (but not executed) until an end sequence command is received. At this point, the sequence is stored in the back-up RAM. Any condition that prevents proper recording of the sequence is saved and can be accessed with the download status command (XSD). The downloaded sequence can be tested with the run sequence command (XR), which executes any specified sequence.

You can use the **XU** command to read back the entered sequence.

To begin the definition of a sequence the **XD** command immediately followed by sequence identifier number (1 - 16) and a delimiter must be entered. An **XT** command ends the sequence definition. You should use the Save command to store sequences in the control module's nonvolatile memory.

The commands entered after the **XD** command and before the **XT** command will be executed in the order in which they were entered when the sequence is run.

Example

The > prompt is replaced by the $ prompt when in sequence edit mode.

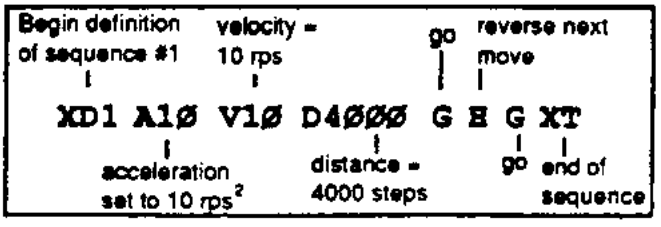


Figure J1-4. Sequence Definition & Termination Example

The definition of sequence #1 is terminated by XT, but the sequence is NOT stored in nonvolatile memory until the save command is given.

Remove jumper 1 to write-protect sequences.

Whenever the above sequence is run, the motor will turn 1 revolution CW and then 1 revolution CCW.

Running Sequences

You can run sequences in by command input from a host computer or by hardware input from a switch or PLC. It is also possible to have a sequence executed on power-up.

Sequence Execution via Command Input

A sequence can be run by entering the **XR** command immediately followed by a sequence identifier number (1 - 16) and a delimiter.

Sequence Execution via Hardware Input

A sequence selected as a binary pattern set by input switches can be run in response to a trigger on Input 4 if **SSF1** is set.

Selection is by Inputs 1-3 if **OSDØ** is set.
Selection is by Inputs 1-3 and Input 7 if **OSD1** is set .

EXAMPLE

Set **SSF1** and set Inputs 1-3 and Input 7 as follows:

Input 1 = 1
Input 2 = Ø
Input 3 = Ø
Input 7 = 1

On an Input 4 transition from low to high level, sequence number 9 will be executed if **OSD1** and **SSF1** have been set. The Input 4 trigger will not be accepted if the axis is busy. When a PLC is used, it is recommended that an output (O3 for example) is set at the beginning of a sequence to indicate to the PLC that the axis is busy. A command could then be added at the end of the sequence to reset the output and indicate to the PLC that the axis is ready to receive a new trigger.

Sequence Execution on Power-up

The RS-232C control module may be programmed to execute a sequence of commands once or continuously on power up.

One of the stored sequences can be designated to be executed when the device is powered up. The sequence to be executed can be selected by prior RS-232C command (**XP1**-**XP16**). The single sequence specified is executed once; control then passes to the normal command processor loop. Alternatively, power-on execution can be disabled with either the **XPØ** or **XZ** command.

It is possible to have the control module execute a sequence on power up by defining the commands within a sequence using the **XD**, **XT**, and **XP** commands. To define sequence #1, enter the following commands:

```
XD1 MN A1Ø V5 D4ØØØØ G XT
```

In this example, you would not apply power to the unit until you were ready for the sequence to begin.

The XP1 command enables a single run of sequence #1 after power on. The XD1 command signifies the start of the definition of sequence #1 and the XT command signifies the end of the sequence #1 definition. When the operator applies power to the control module, sequence #1 will be executed (the motor will turn 10 revolutions in the positive direction). The S command stops the sequence execution. Use the K command to stop sequence execution if a an emergency stop must be made.

Changing a Programmed Sequence

Once a sequence is defined, it cannot be redefined until it has been deleted. A new sequence cannot be downloaded over an existing one.

To change a sequence, it must be be deleted from memory and then redefined as a new sequence using the same number.

First, the sequence must be deleted from memory using the XE command. It will not be deleted from back-up RAM unless the SV command is used after the XE command.

The changed sequence should then be entered. It will not be saved to nonvolatile memory until the SV command is issued.

Checking the Sequence's Contents

You can use the 1XU1 command to check the contents of the sequence. With the 1XU1 command, the control module sends the contents of sequence #1 (control module #1) to the terminal's screen. The 1 preceding the XU command is the device address that must be present since the XU command is a *device-specific* command. After issuing this command, the following message will be displayed on the terminal's screen:

```
MN A1Ø V5 D4ØØØØ G
```

These commands are the contents of sequence #1.

If the operator does not wish to have the sequence executed on power up, it can be cancelled by issuing either the XPØ or XZ command.

Sequence #1 can also be run manually using the XR command.

Sequence Status Requests

Sequence status commands include XC, XSD, XSP, XSR, XSS. The XC command can be used to verify that the memory in the back-up RAM has not been corrupted in any way since last power down. The XC command will cause the control module to send a checksum in the form of three decimal digits (000 - 255) to the terminal. The checksum may be recorded when sequences have been defined, so that it may be used for comparison later, such as each time the RS-232C control module is powered up.

The RS-232C control module is equipped with a battery-backed RAM which enables it to store, recall, and run sequences of operation on command or on power up.

Command Structure

This section defines the basic format for RS-232C control module commands.

①A ②Motion	③Acceleration			④VALID Software Version 1.1
⑤SYNTAX <a>An	⑥UNITS n = rps²	⑦RANGE 0.0306 to 999999.0 (motor dependent)	⑧DEFAULT 10	⑨ATTRIBUTES Buffered Savable in Sequence
⑩EXECUTION TIME		⑪SEE ALSO D, G, V		
RESPONSE TO aA IS A = nnnnnn.nnn revs/sec/sec				(*nnnnnn.nnn) ⑫

1. ***Mnemonic Code***

   This box contains the command's mnemonic code and the command type. The command types are described below. All commands annotated with an asterisk (*) can be prefixed by the character B, in which case they become buffered rather than immediate commands.

2. ***Type***

   This portion of the box contains the command's type. The four command types are listed below.

   Motion — Motion commands affect motor motion (i.e., acceleration, velocity, distance, go home, stop, direction, mode, etc.).

   Status — Status commands respond (report back) with information.

   Set-Up — Setup commands define setup conditions for the application. Setup commands include the following types of commands:

   - Homing (Go home acceleration and velocity, etc.)
   - Input/Output (Limits, scan time, in-position time, etc.)
   - Tuning (Servo or position tracking)
   - General (Set switches, return to factory settings, etc.)

   Programming — Programming commands affect programming and program flow. Examples are trigger, output, all sequence commands, quote, time delays, pause and continue, enable and front-panel, loop and end loop, line feed, carriage return, and backspace.

3. ***Full Name***

   This field contains the full command name.

4. ***Valid Revision Level***

   This field contains the revision history of the command. It includes the revision of software when the command was added or modified. If the revision level of the software you are using is equal to or greater than the revision level listed in this field, you are using the proper version of the software.

5. *Syntax*

The proper syntax for the command is shown here. The specific parameters associated with the command are also shown. If any parameter appears within brackets, such as <a>, it is optional. Definitions of the parameters are described below.

- **a** This indicates that a device address must accompany the command. Only the device specified by this parameter receives and executes the command. If the **a** does not appear in the syntax box, no address is allowed.
- **n** This represents an integer. You may use an integer to specify a variety of values (acceleration, velocity, etc.).
- **s** This indicates that a sign character, either positive or negative (+ or -) is required.

6. *Units*

This field describes what unit of measurement the parameter in the command syntax represents.

7. *Range*

This is the range of valid values that you can specify for n (or any other parameter specified).

8. *Default*

The default setting for the command is shown in this box. A command will perform its function with the default setting if you do not provide a value.

9. *Attributes*

This box indicates if the command is ***immediate*** or ***buffered.*** The system executes immediate commands as it receives them. Buffered commands are executed in the order that they are received with other buffered commands. You can store buffered commands in a sequence and save them in permanent memory.

The lower portion of the box explains how to save the command.

- Automatically Saved
- Independently Saved
- Savable in Sequence
- Never Saved

Commands that are ***automatically saved*** are stored in the system's memory when you enter them. You do not have to save them with a software command (SV). ***Savable in sequence*** commands can be saved only within a sequence of commands. Once a sequence is defined and saved, you can recall it to perform its function. You must enter the SV command to maintain parameters that are ***independently saved.*** A command that is ***never saved*** is never placed in the system's permanent memory.

10. *Execution Time*

The execution time is the span of time that passes from the moment you issue a command to the moment the system begins to execute it.

11. *See Also*

Commands that are related or similar to the command described are listed here.

12. *Response*

A sample status command is given (next to **RESPONSE TO**) and the system response is shown in the adjoining box. Computer mode responses are in parentheses. Interactive mode responses are not in parentheses. The response parameters are represented as follows:

- **n** Represents a numeric character
- **b** Represents a binary number (0 or 1)
- **z** Represents any non-numeric character

This box will only be provided if the system provides a response to the command. If no response is provided, this box will not be included with the command description.

Command List

A Motion	Acceleration Rate			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>An	n = rps^2	0.0306 to 999999.000	10	Buffered Savable in Sequence
EXECUTION TIME		**SEE ALSO** D, G, V		
RESPONSE TO aA IS A = nnnnnn.nnn revs/sec/sec (*nnnnnn.nnn)				

Description

This command sets the acceleration rate. The acceleration remains set until it is changed or the drive is reset. The default is part of the factory settings and is not user-changeable.

If n = 0 there is no acceleration period (start/stop move).

Example

Command	Description
1A10	Sets acceleration of device 1 to 10 rps^2
A5	sets acceleration of all devices to 5 rps^2

B Status	Buffer Status Request			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
aB	None	None	None	Immediate Never Saved
EXECUTION TIME		**SEE ALSO** BS		
RESPONSE TO aB IS Buffer is more (or less) than 90% full (*x)				

Description:

This command requests the status of the command buffer. The response, *R or *B, indicates if more or less than 10% of the command buffer is free.

*R = more than 10% of the buffer is free
*B = less than 10% of the buffer is free

This command is normally sent after a long series of commands to ensure that room exists for more commands. The communications buffer is 2,000 bytes long.

Example

Command	Response
1B	*B (Less than 10% of command buffer is free)

BS Status	Buffer Size Request			VALID Software Version 1.4
SYNTAX aBS	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO BS		
RESPONSE TO aBS IS nnnn bytes free				(nnnn)

Description

This command requests a count of the remaining bytes available in the command buffer. The response, 4 decimal digits (range of 0 - 2000) followed by a [CR], indicates the number of bytes remaining in the command buffer. When entering long strings of commands, check the buffer status to ensure that there is enough room in the buffer, otherwise commands will be lost. Each character (including delimiters) uses one byte.

Example

Command	Response
1BS	*122 (122 bytes remaining in the buffer)

C Programming	Continue			VALID Software Version 1.4
SYNTAX <a>C	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO PS, U		

Description

The continue command ends a pause state. It enables the RS-232C control module to continue executing buffered commands after a pause has been initiated with the PS command or the U command. This command is useful when you want to transmit a string of commands before you actually execute them.

Example

Command	Description
MC	Set indexer to continuous mode
A1Ø	Set acceleration to 10 rps^2
V1Ø	Set velocity to 10 rps
PS	Pause execution until C command is received
C	Execute move (Go)
T1Ø	Wait 10 seconds
VØ	Set velocity to 0 rps
C	Cancel Pause and execute move (C will not be recognized until PS is cleared with C)

CCS Set-Up	Configure Command Source			VALID Software Version 1.4
SYNTAX aCCSn	**UNITS** n = Decode Setting	**RANGE** 0 - 3	**DEFAULT** 0	**ATTRIBUTES** Buffered Independently Saved
EXECUTION TIME		**SEE ALSO** DR, SIM		
RESPONSE TO aCCS **IS** *n				

Description

This command configures the control module for various command input types according to the value of n, as follows:

n = 0	Quadrature encoder (x 4)	4,000 steps/rev with 1,000-line encoder
n = 1	Quadrature encoder (x 2)	2,000 steps/rev with 1,000-line encoder
n = 2	Quadrature encoder (x 1)	1,000 steps/rev with 1,000-line encoder
n = 3	Step and Direction input	Module accepts only step and direction inputs

*CDG Set-Up	Configure Derivative Gain			VALID Software Version 1.4
SYNTAX <a>CDGn	**UNITS** n = gain	**RANGE** 0 to 32,767	**DEFAULT** 57	**ATTRIBUTES** Immediate Independently Saved
EXECUTION TIME		**SEE ALSO** DR		
RESPONSE TO aCDG **IS** *D as in PID = nn				(*nn)

Description

This command configures the derivative gain when using PID tuning. This term represents the derivative of position error, in other words the rate at which position error is changing. It produces a damping action in a similar way to velocity feedback and sets the velocity feedback and feedforward gains to equal values.

*CEW Set-Up	Configure In-Position Window			VALID Software Version 1.4
SYNTAX <a>CEWn	**UNITS** n = steps	**RANGE** 0 to 32,767	**DEFAULT** 50	**ATTRIBUTES** Immediate Independently Saved
EXECUTION TIME		**SEE ALSO** DR		
RESPONSE TO aCEW IS *In posn window (deadband) = nn steps (*nn)				

Description

This command, together with the CIT command, can be used to configure an in position window, which in its turn can be used to indicate that the preceding index has terminated. Output 02 can be configured with the SSC command to reflect the state of the in position window, thus allowing the user to trigger external hardware from the in position condition. The in position condition is met when:

1. The indexer algorithm has finished (no input position command).
2. The CEW condition is met (i.e., the following error is less than that specified by CEW).
3. The above condition has been true for the length of time specified by the CIT command.

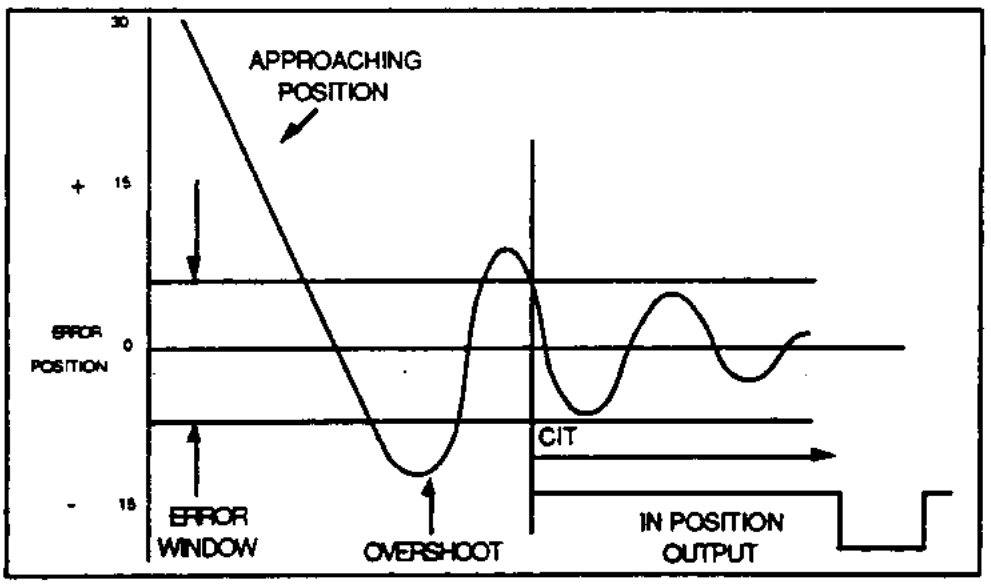


Figure J1-5. Final Positioning

*CFG Set-Up	Configure Feedforward Gain			VALID Software Version 1.4
SYNTAX <a>CFGn	**UNITS** n = gain	**RANGE** 0 to 32,767	**DEFAULT** 57	**ATTRIBUTES** Immediate Independently Saved
EXECUTION TIME		**SEE ALSO** DR		
RESPONSE TO aCFG IS *Velocity feed forward = nn (*nn)				

Description

Use this command to set the velocity feedforward gain. The opposing action of proportional and velocity feedback results in a position error which depends on speed. This is called *following error*. Velocity feedforward can be used to offset the following error and improve tracking accuracy. This is important in contouring applications. For true PID, this can be set to the same value as the velocity feedback, or the CDG command may be used instead.

*CIG Set-Up	Configure Integral Gain			VALID Software Version 1.4
SYNTAX <a>CIGn	**UNITS** n = gain	**RANGE** 0 to 32,767	**DEFAULT** 0	**ATTRIBUTES** Immediate Independently Saved
EXECUTION TIME		**SEE ALSO** DR		
RESPONSE TO aCIG IS *Integral action gain = nn (*nn)				

Description

Use this command to set the integral gain. If proportional feedback is insufficient to overcome static position errors due to friction, integral action accumulates a steady-state position error until sufficient torque is applied to move the load to reduce the error. It improves overall positioning accuracy, but low frequency oscillation may occur around the commanded position.

•CIT Set-Up	Configure In-Position Time			VALID Software Version 1.4
SYNTAX <a>CITn	UNITS n = ms	RANGE 0 to 32,767	DEFAULT 20	ATTRIBUTES Immediate Independently Saved
EXECUTION TIME		SEE ALSO SS, CEW		
RESPONSE TO aCIT IS *In position wait time = nn milliseconds (*nn)				

Description

This command is used to specify the time period that the servo is to be within the deadband region before the *in position* signal is generated. The range of n is 0 to 32,767 and it represents the number of milliseconds to be used as the testing time frame. The shortest time for which the motor must be stopped and within the deadband before it is considered *in-position* is 2 milliseconds. If at any point during that 2 milliseconds the motor is out of deadband, the in-position output will remain inactive.

Example

Command	Description
1SSC1	Enable output 2 as an in-position output
1CIT3Ø	Set in-position time to 30 ms
CIW2Ø	Set in-position window to 20 steps

This sequence will make the *in-position line* (output #2) turn on when the motor has completed a move. This happens when the shaft is within the 20 step deadband and has remained within the deadband for 30 milliseconds.

CIW Set-Up	Configure Integral Action Window			VALID Software Version 1.5
SYNTAX <a>CIWn	UNITS n = steps	RANGE 0 to 32,767	DEFAULT 40	ATTRIBUTES Immediate Independently Saved
EXECUTION TIME		SEE ALSO OSB		
RESPONSE TO aCIW IS *Switched int act window = nn input steps (*nn)				

Description

This command sets the width of the position window within which integral action is active when the indexer demanded motion is complete.

Example

Command	Description
1CIW5	Figure J1-6 illustrates the integral action window set to 5 user steps. It initiates final positioning under integral gain.

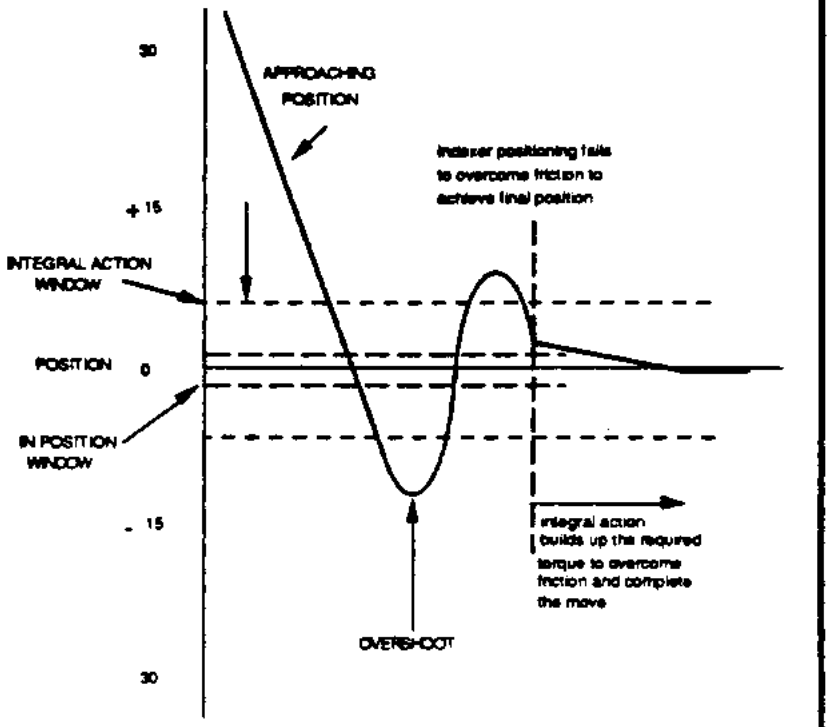


Figure J1-6. Integral Action

*CJL Set-Up	Enter Combined Motor and Load Inertia			VALID Software Version 1.4
SYNTAX aCJLn	**UNITS** n = kg-cm^2	**RANGE** None	**DEFAULT** 1.6	**ATTRIBUTES** Immediate Independently Saved
EXECUTION TIME		**SEE ALSO** HELP11, CTQ		
RESPONSE TO aCJL IS Motor + load inertia = n.nen kg cms cms (*n.nen)				

Description

This command has no effect on the behavior of the RS-232C control module. It allows entry of the combined motor and load inertia in calculations when using HELP11 as an aid to setting up the servo system parameters.

*CMR Set-Up	Configure Motor Resolution			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>CMRn	n = encoder counts/rev	1 to 32,767	4000	Immediate Independently Saved
EXECUTION TIME		**SEE ALSO** CUR		
RESPONSE TO aCMR IS *Feedback (motor) resolution = nnnn steps/rev (*nnnn)				

Description

This command programs the RS-232C control module to match the motor resolution for the servo motor or the open loop stepper motor. The number n is determined by the encoder fitted to the motor or the number of steps per revolution of an open loop stepper. It will normally be 4,000. This parameter does not need to be changed unless the motor is changed to one having a different encoder resolution or number of steps per revolution.

Example

Command: 1CMR2000

Response: would set the resolution of device 1 for a 500 line encoder. Note that the control module uses a x4 decoding circuit giving an effective resolution of four times the number of encoder lines.

*COFF Set-Up	Configure Amplifier Offset			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>COFFn	n = offset	-1023 to +1023	0	Immediate Independently Saved
EXECUTION TIME		**SEE ALSO** DR		
RESPONSE TO aCOFF IS *Amplifier offset = n (*n)				

Description

This command can be used to cancel the effect of an offset in the torque amplifier. An excessive offset error can sometimes be the cause of movement at power up but it is unlikely to affect the closed loop operation unless it is very large. The value should be chosen such that at standstill, position demand equals actual position. It can be set by opening the loop (make CPG, CVG, CIG all zero) and adjusting COFF for zero drift, or reading back the position using the DPE command and setting COFF for the smallest error.

*CPE Programming	Configure Position Error			VALID Software Version 1.4
SYNTAX <a>CPEn	**UNITS** n = steps	**RANGE** 0 to 32,767	**DEFAULT** 4000	**ATTRIBUTES** Immediate Independently Saved
EXECUTION TIME		**SEE ALSO** DR		
RESPONSE TO aCPE IS *Dynamic error limit = nnnn (*nnnn)				

Description

This command defines maximum allowed following error. If the absolute position error is greater than this number, the RS-232C control module will disable the drive. If a valid number in steps is entered, it will become the new maximum following error, otherwise the current setting is reported. Exceeding the maximum following error is a fault condition that will cause the amplifier to be shutdown. If the maximum following error is defined as 32,768 the shutdown function is disabled and no amount of following error will generate an error condition or shutdown the motor.

In the event that the gain used during setting up is too great, a small position error setting will prevent oscillation and potential mechanical damage to the connected system.

Example

Command	Response
1CPE1000	Set the following error limit of device #1 to 1,000 steps.

*CPG Set-Up	Configure Proportional Gain			VALID Software Version 1.4
SYNTAX <a>CPGn	**UNITS** n = gain	**RANGE** 0 to 32,767	**DEFAULT** 92	**ATTRIBUTES** Immediate Independently Saved
EXECUTION TIME		**SEE ALSO** DR		
RESPONSE TO aCPG IS *Proportional gain = nn (*nn)				

Description

This command is used to set the proportional gain. The proportional gain determines the amount of torque produced in response to a given position error. It sets the stiffness of the system and also affects the following error. A high proportional gain gives a stiff, responsive system but results in overshoot and oscillation which require damping. *If no value is supplied with the command, the previous setting is reported.*

*CTG Set-Up	Configure Filter Time Constant			VALID Software Version 1.4
SYNTAX aCTGn	**UNITS** n = ms	**RANGE** 0 to 32,767	**DEFAULT** 2.94	**ATTRIBUTES** Immediate Independently Saved
EXECUTION TIME		**SEE ALSO** DR		
RESPONSE TO aCTG **IS** *Filter time constant = n.nn milliseconds (*n.nn)				

Description

This command sets the time constant of the filter used to reduce the effect of transients at the input of the drive amplifier and to smooth the response of the incremental velocity feedback loop.

*CTQ Set-Up	Enter Motor Torque			VALID Software Version 1.4
SYNTAX aCTQn	**UNITS** n = Nm	**RANGE** None	**DEFAULT** 3.6	**ATTRIBUTES** Immediate Independently Saved
EXECUTION TIME		**SEE ALSO** DR, HELP11, CJL		
RESPONSE TO aCTQ **IS** *Available motor torque = n.nen Newton meters (*n.nen)				

Description

This command has no effect on the behavior of the RS-232C control module. It allows entry of the available motor torque in calculations when using **HELP11** as an aid to setting up the servo system parameters.

*CUR Set-Up	Configure User Resolution			VALID Software Version 1.4
SYNTAX aCURn	**UNITS** n = user steps	**RANGE** 1 to 32,767	**DEFAULT** 4000	**ATTRIBUTES** Immediate Independently Saved
EXECUTION TIME		**SEE ALSO** CMR, DR, HELP11		
RESPONSE TO aCUR **IS** *User resolution = nnnn steps/rev (*nnnn)				

Description

Use this command to configure the user resolution. This command is used to define the number of steps per revolution required by the user .

Example

Command	Response
1CUR2000	Defines the user resolution as 2,000 steps/rev.
D4000	Defines distance as 4,000 user steps (2 revolutions)
G	Executes the move (Go)

*CVG	Configure Velocity Gain			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>CVGn	n = gain	0 to 32,767	57	Immediate Independently Saved
EXECUTION TIME		**SEE ALSO** DR		
RESPONSE TO aCVG **IS** *Velocity feedback = nn				(*nn)

Description

Use this command to set the gain of the the velocity feedback loop. The velocity feedback is a signal which increases with shaft speed. It acts in opposition to proportional feedback to stabilize the motion. This setting is generally used to damp vibrations in the servo response, allowing a higher proportional gain to be used. See *Setting Up Servo Parameters* in Section H for further details.

D Motion	Distance			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>D<s>n	n = steps	±2,097,152 steps	4000	Buffered Savable in Sequence
EXECUTION TIME		**SEE ALSO** A, MN, MPA, MPI, MQ, V, G		
RESPONSE TO aD **IS** *D = nnnn steps				(*nnnn)

Description

This command is used in MN and MQ to move the motor a number of steps (n) in the direction specified by <s>. The direction is assumed to be positive if no sign is given.

Example

Command	Response
MN	Set indexer to normal mode
A1Ø	Set acceleration to 10 rps^2
V1Ø	Set velocity to 10 rps
D-1ØØØØØ	Set distance to 100,000 ccw steps
G	Execute the move (a 4,000 steps/rev motor will turn 25 revolutions in the negative direction).

DFX Status	Display Flags Indexer			VALID Software Version 1.4
SYNTAX aDFX	**UNITS** None	**RANGE** None	**DEFAULT** None	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME		**SEE ALSO** None		
RESPONSE TO aDFX IS (see below)		(*bbbb_bbbb_bbbb_bbbb_bbbb_bbbb_bbbb_bbbb)		

Description

This command reports all indexer status flags as 32 bits where the response is as follows:

```
>1DFX
  N/A          N/A          N/A          N/A          _
  N/A          N/A          N/A          N/A          _
  PUSEQ        EXSEQ        TRIPPAUSE    TRDPAUSE     _
  TRIGPAUSE    UPAUSE       PAUSE        WAIT         _
  HOMESLOW     HOMEFAST     HOMEINDEX    HOMELIMIT    _
  HITLIMIT-    HITLIMIT+    JOGGING      SKIPPING     _
  DIR          NEG          MQ           MC           _
  ABSOLUTE     QPERFORM     PRPERFORM    CPERFORM     _
  *bbbb_bbbb_bbbb_bbbb_bbbb_bbbb_bbbb_bbbb
```

The order of the bits in the last line of the response (*bbbb_bbbb_bbbb_bbbb_bbbb_bbbb_bbbb_bbbb) is, from left to right, *31, 30, 29, 28....3, 2, 1, 0*.
Response Bits: 31-24 are all reserved for future use

23	1= Run sequence on power up
22	1= Executing a sequence
21	1= Paused, waiting on in-position
20	1= Paused, waiting on distance trigger bits
19	1= Paused, waiting on trigger bits
18	1= Paused by U command flag, waiting on a C (continue)
17	1= Paused by PS command, waiting on C (continue)
16	1= Performing a wait
15	1= Homing 2nd leg, low speed move back to home limit
14	1= Homing 1st leg of home move, high speed to home limit
13	1= Go home move to encoder position
12	1= Home limit switch has been hit
11	1= +limit switch has been hit
10	1= +limit switch has been hit
9	1= Jogging is enabled, we are jogging
8	1= Skip next buffered command
7	1= Set if continuous move direction is negative
6	1= Set if current move direction is negative
5	1= Set if we want to change velocity
4	1= Set if we are in continuous mode,clear if we are in preset mode.
3	1= Set if we are in absolute mode, clear if in incremental mode.
2	1= Performing a variable speed move
1	1= Performing a preset move
0	1= Performing a continuous move

DIC Status	Display Indexer Counter			VALID Software Version 1.4
SYNTAX aDIC	UNITS Steps	RANGE None	DEFAULT 4000	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO D		
RESPONSE TO aDIC IS Indexer target position = nnn (*nnn)				

Description

This command requests a single display of the contents of the indexer counter as a single value in steps at the user resolution. This is the programmed D value.

DPA Status	Display Position Actual			VALID Software Version 1.4
SYNTAX aDPAn	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO PZ, SP		
RESPONSE TO aDPA IS ±nnnn (continuously scrolled)				

Description

This command continuously displays the actual position. The response is the position in user steps resulting from the move commands sent to the drive from the RS-232C control module since the drive was enabled or a PZ or SP command was issued.

DPE Status	Display Position Error			VALID Software Version 1.4
SYNTAX aDPE	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO PZ		
RESPONSE TO aDPE IS nnnn (continuously scrolled)				

Description

This command continuously displays the position error. The response is the difference between the setpoint and the actual position in user steps. It is used by the position control algorithm to control motor current. The difference between command setpoint and actual position is also used to determine if the motor is within the deadband specified by the CEW command. The response is a single instantaneous value reported at 150 ms intervals until the return key is pressed.

DPS Status	Display Position Setpoint			VALID Software Version 1.4
SYNTAX aDPS	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO None		
RESPONSE TO aDPS IS ±nnnnn (continuously scrolled)				

Description

This command continuously displays the position setpoint. The response is repeatedly updated until the return key is pressed. It is the absolute number of pulses sent to the drive from the control module since the drive was enabled (or reset).

DR Status	Display Report			VALID Software Version 1.4
SYNTAX aDR	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO PS, U		
RESPONSE TO aDR IS (see below)				

Description

This command displays a user-friendly screen of the main parameters and the values of their settings (see below).

```
>1DR
Drive is a brushless servo.  Indexer mode is:  MN
Load Position (user steps).=0              Position Error.............=-1
Programmed Distance.......D=4000           Programmed Velocity.......V=1
Programmed Acceleration...A=10.00976       Proportional Gain.........PG=92
Velocity Feedback.........VG=57            Filter Time Constant......TG=2.94
Velocity Feed Forward.....FG=57            Integral Action Gain......IG=0
Dynamic Error Limit.......PE=4000          In position wait time.....IT=20
Amplifier offset..........OFF=0            In pos window (deadband).EW=50
User resolution...........UR=4000          Motor (fb) resolution.....MR=4000
Go Home Final vel'y......GHF=0.09997558    Go Home Acceleration......GA=10.00976
Jog Velocity..............JV=1            Jog Acceleration..........JA=99
Limit Acceleration........LA=900.0244      Limit Disable.............LD=0
Int. action window........IW=40
Drive status is:
*No Errors
*0000_0000_0000_0000
Software Switches are:
-RS232_ECHO I6(PS) O2(IN_POS) O1(WDOG) _ CTRL_STOP I4(SEQ_STRB) LIM_SV STOP_SV
*0000_0000
Other switches are:
N/A SWITCH_IACT SINGLE_DS I7(SEQ_SEL) _ JOG_ENABLE N/A N/A N/A
*0000_0000
```

DS Status	**Display Signal**			**VALID** Software Version 1.5
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
aDSn	n = signal number	1 - 14	None	Immediate Never Saved
EXECUTION TIME		**SEE ALSO** PIC		
RESPONSE TO aDSn **IS** *n (signal value) displayed continuously				
RESPONSE TO aDS **IS** (list of signal displays - see below)				

Description

This command requests the continuous display of variable parameters. The PIC screen (see the PIC command description) represents the control algorithm. Any parameter (such as DS3) in the PIC diagram can be displayed. If n is omitted, a list of the signal displays is produced.

The individual parameters are as follows:

DS1	Motor velocity	DS8	Position error
DS2	Input velocity	DS9	Error action + integral action
DS3	Motor position	DS10	Velocity action difference
DS4	Velocity feedback action	DS11	Integral of position error
DS5	Scaled input position	DS12	Integral action
DS6	Filtered torque demand	DS13	Error times gain (=torque)
DS7	Input velocity action	DS14	Unfiltered output demand

The individual units are as follows:

DS1	mtr steps/500µsec	DS8	mtr steps
DS2	mtr steps/500µsec	DS9	32767 gives max o/p
DS3	mtr steps	DS10	32767 gives max o/p
DS4	32767 gives max o/p	DS11	mtr steps
DS5	mtr steps	DS12	32767 gives max o/p
DS6	1024 gives max o/p	DS13	32767 gives max o/p
DS7	32767 gives max o/p	DS14	32767 gives max o/p

Example

Command	Response
1DS8	Shows the position error for RS-232C control module 1. The significance of the numbers displayed will vary with the signal point chosen. Input and motor velocities are expressed in encoder edges per 550 µS period, with a maximum value of ±255. All other parameters have values between +32767 and -32768, except the torque demand which covers the range +1023 to -1024 for full torque. A continuous display of the maximum value indicates saturation.

*DTA Set-up	Set Dither Amplitude			VALID Software Version 2.3
SYNTAX aDTAn	UNITS n = amplitude	RANGE 0 - 1023	DEFAULT 300	ATTRIBUTES Immediate Independently Saved
EXECUTION TIME		SEE ALSO OSG		
RESPONSE TO aDVA IS nnnn (continuously scrolled)				

Description

When the dither function is enabled (OSG command), this command sets the amplitude of the superimposed dither signal.

*DTF Set-up	Set Dither Frequency			VALID Software Version 2.3
SYNTAX aDTFn	UNITS n = Hz	RANGE 38 - 500	DEFAULT 50	ATTRIBUTES Immediate Independently Saved
EXECUTION TIME		SEE ALSO OSG		
RESPONSE TO aDTF IS *nnn				

Description

When the dither function is enabled (OSG1 command), this command sets the frequency of the superimposed dither signal. There is a rounding up or down of the entered value. The exact value can be displayed by giving the DTF command with no parameter.

DVA Status	Display Velocity Actual			VALID Software Version 1.4
SYNTAX aDVA	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO DS		
RESPONSE TO aDVA IS nnnn (continuously scrolled)				

Description

This command causes the continuous display of the actual velocity. The number is reported in motor steps per 500µs and is repeatedly updated until the return key is pressed. This value is the shaft velocity being read from the encoder measured over a 500 µs period. The DVA command has the same effect as the DS1 command.

DVS Status	Display Velocity Setpoint			VALID Software Version 1.4
SYNTAX aDVS	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO DS		
RESPONSE TO aDVS IS nnnn (continuously scrolled)				

Description

This command requests a continuous display of the velocity setpoint. The displayed value is the velocity being sent to the velocity part of the servo loop by the servo algorithm. It is repeatedly updated until the return key is pressed. The DVS command has the same effect as the DS2 command.

E Set-Up	Enable Communications			VALID Software Version
SYNTAX <a>E	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO F		

Description

This command allows the RS-232C control module to accept commands over the serial communications interface. You can re-enable the the communications interface with this command if you had previously disabled the RS-232C interface with the F command. If several units are using the same communications interface, use of the E and F commands can help to streamline programming.

Example

Command	Response
F	Disable communications to all units
1E	Enable communication to unit 1
4E	Enable communication to unit 4
A10	Set acceleration to 10 rps^2
V5	Set velocity to 5 rps
D5000	Set distance to 5,000 steps
G	Execute move

Disables all axes on the communication interface, then enables the control module for axes 1 and 4. Sets acceleration to 10 rps^2, velocity to 5 rps and distance to 5,000 steps. On execution, only axes #1 and #4 will move.

EX Set-Up	Set Communication Style			VALID Software Version 1.4
SYNTAX aEXn	UNITS n = terminal or computer mode	RANGE 1 or 0	DEFAULT 1	ATTRIBUTES Immediate Independently Saved
EXECUTION TIME		SEE ALSO None		
RESPONSE TO aEX IS *b				

Description

This command sets the style of communication between the RS-232C control module and the terminal/computer.

n=1: sets terminal mode (i.e., User-friendly messages sent)
n=0: sets computer mode (i.e., No user-friendly messages)

F Set-Up	Disable Communications			VALID Software Version 1.4
SYNTAX <a>F	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO E		

Description

The F command disables communication with devices connected to the RS-232C control module. It is useful when you are programming multiple devices on a single interface. Devices that are not intended to respond to universal commands should be disabled using device-specific F commands. This enables you to program other devices without specifying a device identifier on every command.

Example

Command	Response
1F	Disable communications to unit #1
3F	Disable communications to unit #3
G	Execute move using last defined parameters

The communications interface on units with device addresses 1 and 3 are disabled. All RS-232C control modules (except 1 and 3) will execute a move.

FDA Set-Up	Acceleration, Linear Interpolation Moves			VALID Software Version 2.3
SYNTAX FDAn	UNITS n = rps^2	RANGE 0.0306 - 999999	DEFAULT 10	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO FDV, GL, MSY		

Description

This command defines the common feed acceleration along the line of motion for a linear interpolation move. It may not be device specific.

Example

Command	Response
MSY	Set synchronous mode
FDA100	Set feed acceleration to 100 rps^2
FDV10	Set feed velocity to 10 rps
GL60000,35000	Start linear interpolation move

FDV Set-Up	Velocity, Linear Interpolation Moves			VALID Software Version 2.3
SYNTAX	UNITS	RANGE	DEFAULT	ATTRIBUTES
FDVn	rps	0.0001 - 999999	100	Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO FDV, GL, MSY		

Description

This command defines the common feed velocity along the line of motion for a linear interpolation move. It may not be device specific.

Example

Command	Response
MSY	Set synchronous mode
FDA10Ø	Set feed acceleration to 100 rps^2
FDV1Ø	Set feed velocity to 10 rps
GL5ØØØØ,3ØØØØ	Start linear interpolation move

G Motion	Go			VALID Software Version 1.4
SYNTAX	UNITS	RANGE	DEFAULT	ATTRIBUTES
<a>G	None	None	None	Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO S, MN, MC, MQ, A, V, D		

Description

The Go command allows you to make a move using the previously entered parameters. It is not necessary to re-enter A, V, and D.

Example

Command	Response
MN	Set indexer to normal mode
A1Ø	Set acceleration to 10 rps^2
V1Ø	Set velocity to 10 rps
D1ØØØØ	Set distance to 100,000 steps
G	Execute move

A 4,000 steps/rev motor will turn 25 revolutions after the G is entered.

GA Motion	Go Home Acceleration			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>GAn	n = rps^2	0.001 to 999,999	10	Buffered Savable in Sequence
EXECUTION TIME		**SEE ALSO** GH, SS, GHF, GHP		
RESPONSE TO aGA IS GA = nn.nnn revs/sec/sec (*nn.nn)				

Description

This command sets the acceleration rate that the GH command will use. The value can be saved in non-volatile RAM. The default is changed by the user if the GA command is issued and then a save is performed (SV command).

GH Motion	Go Home			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>GH<s>n	n = rps	0.0001 to 100	None	Buffered Savable in Sequence
EXECUTION TIME		**SEE ALSO** GA, GHF, RG		

Description

This command causes the RS-232C control module to rotate the motor in the direction and at the speed specified until its home limit input is activated.

The GH+2 command causes the controller to seek the home position at 2 rps. The sign is optional (a + or positive is assumed if omitted). The controller reverses direction if a limit is activated and it will cease the attempt to go home if the second limit is also activated.

The GH command resets the absolute position counter to zero. If you want to set the counter to another value, use the SP command after the GH is complete.

Example

GH-2 (Motor will turn negative at 2 rps and look for Home limit input to go active. It will then stop, reverse, and finally stop completely when the home switch is reached for the second time.

GHF Motion	Go Home Final			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>GHFn	n = rps	0.01 to 100.0	0.1	Buffered Savable in Sequence
EXECUTION TIME		**SEE ALSO** GA, GH, RG, SS		
RESPONSE TO aGHF IS GHF = n.nnnn revs/sec (*n.nnn)				

Description

This command sets the velocity for the final move in the go home sequence. The value can be saved in battery-backed up RAM.

GL Motion	Go, Linear Mode			VALID Software Version 2.3
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
GHFn,m	n & m = user steps	-268435455 to +268435455	0	Buffered Savable in Sequence
EXECUTION TIME		**SEE ALSO** FDA, FDV, MSY		

Description

Go - make a linear interpolation move using the incremental distances entered for axis 1 and axis 2, where n is the axis 1 distance and m is the axis 2 distance. The feed acceleration and feed velocity, **FDA** and **FDV**, should be set prior to this command.

Example

Command	Response
GL8000,4000	Go - start linear interpolation move of axis 1 for 8000 steps and axis 2 for 4000 steps.

GSY Motion	Go, Synchronous Mode			VALID Software Version 2.3
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
GHFn,m	None	None	None	Buffered Savable in Sequence
EXECUTION TIME		**SEE ALSO** MSY, PSY		

Description

Go - make a synchronous move using the parameters entered for all axes in synchronous mode (**MSY**). The acceleration, velocity and distance should be set for each axis prior to this command.

Example

Command	Response
MSY	Set synchronous mode
A10	Set acceleration to 10 revs/sec/sec
V10	Set velocity to 10 revs/sec
1D100000	Set distance axis 1to 100,000 steps
2D5000	Set distance axis 2 to 5000 steps
GSY	Go - start move on both axes simultaneously

H Motion	Change Direction			VALID Software Version 1.4
SYNTAX <a>H	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO H+, H-, D		

Description This command reverses direction of the next move.

Example

Command	Response
MH	Set indexer to normal mode
A10	Set acceleration to 10 rps^2
V10	Set velocity to 10 rps
D8000	Set distance to 8,000 steps
G	Execute move
H	Change direction of move
G	Execute move

The motor will turn two revolutions positive and then 2 revolutions negative.

^H Motion	Backspace			VALID Software Version 1.4
SYNTAX ^H	UNITS None	RANGE None	DEFAULT one character (Control H, ASCII 8)	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO None		

Description This command produces a backspace during command input, deleting last character. It will not prevent the execution of an immediate command.

The ^H command (^H indicates that the **CONTROL** or **CTRL** key is held down when the H key is pressed) backspaces one character provided a delimiter has not been sent. A new character may be entered at that position to replace the existing character. The effect of this command character is to cause the RS-232C control module to back up one character in the command buffer regardless of what appears on the terminal. On some terminals pressing the **BACKSPACE** key will produce the same character.

Because the RS-232C control module processes each command upon receipt of the delimiter, it is not possible to backspace once the delimiter is encountered. If you type a device address wrongly, you must re-type the whole command.

H± Motion	Change Direction			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>Hs	s = pos/neg direction	+ or -	+	Buffered Savable in Sequence
EXECUTION TIME		**SEE ALSO** H, D		

Description This command sets the default direction of all moves according to s = + (positive, usually CW) or - (negative, usually CCW).

Example H- (All moves default to negative unless otherwise specified in the command string. A D command resets the motor's direction for subsequent moves).

HELP Programming	Produce Help Screens			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>HELP<n>	n = help screens	1 - 12	None	Immediate Never Saved
EXECUTION TIME		**SEE ALSO** None		
RESPONSE TO aHELP<n> **IS** (see below)				

Description This command displays the help screens. It provides a list of commands and a brief description of each command (see below). The command on its own will produce the base menu followed by all other help screens. Individual screens can be accessed by typing HELP followed by the appropriate screen number. HELP11 provides a worked example on servo set up. This is available in terminal mode only (see EX command).

The return key will take you out of the help menu.

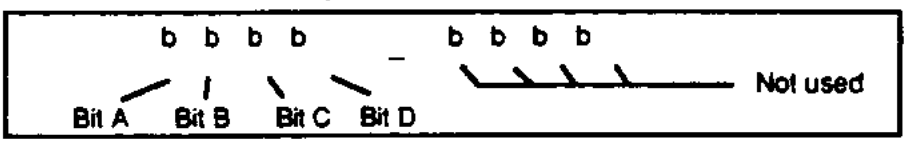
With the exception of HELP11, all control modules return the same help information, so no device address is needed. The gain values returned in HELP11 relate to the torque and inertia figures programmed by the CJL and CTQ commands.

```
DON'T PANIC Type 'HELP'
>HELP
HELP1 Help Commands
------------------
HELP....Produce all help displays          HELP1...List of HELP's
HELP2...Indexer Commands                   HELP3...Start/Stop Commands
HELP4...Sequence Commands                  HELP5...Homing, Jog, Limit Commands
HELP6...Option Set Commands                HELP7...Information Commands
HELP8...Inputs and Outputs                 HELP9...Control and Memory Commands
HELP10..Using the indexer                  HELP11..Recommended servo settings
HELP12..Servo Setup Procedure
PRESS SPACE BAR TO CONTINUE OR RETURN TO EXIT
>
>HELP3
HELP3 Start/Stop Commands
------------------------
K.......Kill indexer instantly             LS......Stop now at LA deceleration
S.......Stop motion, decelerate now        SB......Stop motion, buffered
ST1.....Buffered stop = de-energize        STB.....Buffered start = re-energize
OFF.....Immediate stop = de-energize       ON......Immediate stop = re-energize
Y.......Terminate loop or sequence         Z.......Power up reset
>
```

HS Status	**Display Hardware Configured Switches**			**VALID** Software Version 1.5
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
aHS	None	None	None	Immediate Never Saved
EXECUTION TIME		**SEE ALSO** HS commands		
RESPONSE TO aHS IS *bbbb_bbbb (b = 0 or 1)				

Description

The command displays the status of hardware configured switches (i.e., how the control module is configured for its environment). The command displays a binary representation of this status. The response , *bbbb_bbbb. is interpreted as follows:



Bit A : 1 = Resolver ø = Other
Bit B : 1 = Brush ø = Brushless initialization
Bit C : 1 = Servo ø = Open loop
Bit D : 1 = Hydraulic ø = Not hydraulic

If the HS commands are not configured, the response to the status command is ?.

Example

Command	Response
1HS	*ø91ø_øøøø This indicates that the control module is configured for brushless servo operation.

HSB Set-Up	Configure Motor Type			VALID Software Version 1.4
SYNTAX aHSBn	UNITS n = motor type	RANGE 0 or 1	DEFAULT 0	ATTRIBUTES Immediate Independently Saved
EXECUTION TIME		SEE ALSO HSC		
RESPONSE TO aHSB IS *n				

Description

This command sets the configuration for a brush or brushless motor.

n=0: closed loop brushless
n=1: other

HSB and **HSC** Combined Commands:

	HSB	HSC
STEPPER	1	0
HYBRID SERVO	0	1
PM AC BRUSHLESS SERVO	0	1
DC BRUSH SERVO	1	1

HSC Set-Up	Configure Motor Type			VALID Software Version 2.0
SYNTAX aHSCn	UNITS n = motor type	RANGE 0 or 1	DEFAULT 1	ATTRIBUTES Immediate Independently Saved
EXECUTION TIME		SEE ALSO HSB		
RESPONSE TO aHSC IS *n				

Description

This command sets the configuration for a stepper motor.

n=0: open-loop stepper
n=1: other

HSB and **HSC** Combined Commands:

	HSB	HSC
STEPPER	1	0
HYBRID SERVO	0	1
PM AC BRUSHLESS SERVO	0	1
DC BRUSH SERVO	1	1

HSD Set-Up	Daughter Board Outputs			VALID Software Version 2.1
SYNTAX aHSDn	UNITS n = On or Off	RANGE 0 or 1	DEFAULT 0	ATTRIBUTES Immediate Independently Saved
EXECUTION TIME		SEE ALSO O		

Description

Used with output daughter boards to signify that the board is fitted. The O command has an extended argument when #SD1 is set (e.g., OXXX1X1) to drive the 3 extra outputs. The extra outputs are connected to terminals X1-X3 of connector I/O 4 on the motherboard.

n = 0: No daughter board fitted
n = 1: Daughter board with 3 additional outputs fitted

IO Programming	Immediate Output			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>IOnnn	n = output bit settings	0, 1, or X	None	Immediate Never Saved
EXECUTION TIME		**SEE ALSO** None		
RESPONSE TO aIO IS O1/watchdog O/in-position O3 *nnn (*nnn)				

Description

This command sets the output bits as specified in the pattern on an immediate basis. The variable n may be set to 0, 1 or X (X leaves the output unchanged). The variable corresponds, from left to right, to O1, O2 (watchdog), and O3 (in-position).

Example

Command	Description
2IO1ØX	Sets the outputs of device 2 : O1 to 1 and O2 to Ø. It leaves O3 unchanged.

IS Status	Input Status			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
aIS	None	None	None	Immediate Never Saved
EXECUTION TIME		**SEE ALSO** None		
RESPONSE TO aIS IS *bbbbbbb_bbbbbbb (b = 0 or 1)				

Description

This is an immediate command that will report the status of all the inputs no matter what the configuration. The pattern of the response is, from left to right:

HOME, LMT-, LMT+, +JOG, -JOG, EIN, INDEX TRACK - INPUT 1, INPUT 2, INPUT 3, INPUT 4, INPUT 5, INPUT 6, INPUT 7.

Ø = Input active
1 = Input inactive
EIN = Drive energized
Open circuit inputs are indicated by Øs.

Example

Command	Response
1IS	*ØØØØØØØ_1ØØØØØØ

This indicates that only Input #1 is active.

JA Motion	Jog Acceleration			VALID Software Version 1.4
SYNTAX <a>JAn	**UNITS** n = rps^2	**RANGE** 0.01 to 999999.00	**DEFAULT** 99	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME		**SEE ALSO** JV, OSE		
RESPONSE TO aJA IS JA = nn revs/sec/sec				

Description

The JA command sets the jog acceleration. The acceleration rate used in jog operations is set by this command.

Example

Command	Description
JV1.5	Sets Jog velocity to a top speed of 1.5 rps when the jog function is activated and the appropriate input is active
JA20	Sets acceleration at 20 rps^2

JV Motion	Jog Velocity			VALID Software Version 1.4
SYNTAX <a>JVn	**UNITS** n = rps	**RANGE** 0.0001 to 100.0000	**DEFAULT** 1	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME		**SEE ALSO** JA, OSE		
RESPONSE TO aJV IS JC = nn revs/sec				

Description

The JV command sets the jog velocity. The velocity used in jog operations is set using this command. The factory default can be changed by saving a new JV value to non-volatile RAM with the SV command.

K Motion	Kill			VALID Software Version 1.5
SYNTAX <a>K	**UNITS** None	**RANGE** None	**DEFAULT** None	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME		**SEE ALSO** JA		

Description

This command stops RS-232C control module commands to the drive. In addition it terminates a loop, ends a time delay, and aborts a command sequence download in progress (XD command). The command buffer is also cleared.

LA Motion	Limit Deceleration			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>LAn	n = rps^2	0.001 to 999,999	900	Immediate Savable in Sequence
EXECUTION TIME		**SEE ALSO** SSE		
RESPONSE TO <a>LA **IS** LA = nnn.nnn revs/sec/sec				(*nnn.nnn)

Description

This command defines or reports the deceleration rate after limit switch operation. This allows a rapid deceleration in response to a limit switch regardless of the rate programmed by the A command.

LD Programming	Limit Disable			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>LDn	n = enable/ disable limits	0 - 3	0	Immediate Independently Saved
EXECUTION TIME		**SEE ALSO** None		
RESPONSE TO <a>LD **IS** *n				

Description

This command is used to disable the limit switch functions.

n = 0: Enable all limits (default)
n = 1: Disable + limit
n = 2: Disable - limit
n = 3: Disable + and - limit

LS Motion	Limit Stop			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>LS	None	None	None	Immediate Never Saved
EXECUTION TIME		**SEE ALSO** LA		

Description

This command stops the motor at the deceleration rate previously set with the LA command. The deceleration rate set by LA is usually set higher than the normal acceleration (A) rate. LA is used as an alternative to the S command, to facilitate a faster deceleration.

MAS Set-Up	Mode Asynchronous			VALID Software Version 2.3
SYNTAX <a>MAS	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO MSY		

Description

MAS sets the axis mode to asynchronous. It causes subsequent moves to start asynchronously. Axes in MAS may start, stop and change speed independently of other axes in the ZDDC system. The command is used to return an axis to normal (asynchronous) operation after it has been in synchronous mode.

Example

Command	Description
MSY	Set synchronuous mode
FDA1Ø	Set acceleration to 10 rps^2
FDV1Ø	Set velocity to 10 rps
GL4ØØØ,1ØØØ	Go - start linear interpolation move
MAS	Return axes to asynchronous mode

MC Set-Up	Mode Continuous			VALID Software Version 1.4
SYNTAX <a>MC	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO MN, A, V		

Description

The MC command sets the move mode to continuous. It causes subsequent moves to ignore any distance parameter and move continuously at the programmed velocity until stopped by a limit switch or the K, LS, or S commands.

Example

Command	Description
MC	Sets indexer to continuous mode
D-	Sets direction to negative (CCW)
A1Ø	Sets acceleration to 10 rps^2
V1Ø	Sets velocity to 10 rps
G	Executes move with last distance parameter

(Motor turns CCW at 10 rps until stopped). D+ or D- sets the direction in MC.

MN Set-Up	Mode Normal			VALID Software Version 1.4
SYNTAX <a>MN	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO MQ, MC, MPA, MPI, A, V, D		

Description

The MN command sets the move mode to normal preset distance. It causes last issued distance parameter to be used as the distance for the current move. The MN command will change the mode of operation from continuous back to preset.

Example

Command	Description
MN	Sets indexer to normal mode
A10	Sets acceleration to 10 rps^2
V10	Sets velocity to 10 rps
D8000	Sets distance to 8,000 steps
G	Executes move

The motor turns 2 revolutions in the positive direction.

MPA Set-Up	Mode Position Absolute			VALID Software Version 1.4
SYNTAX <a>MPA	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO MN, MC, MQ, MPI, D, PZ		

Description

The MPA command sets the position mode to absolute. In this mode all move distances are referenced to absolute zero. Units are scaled by the CUR command. You must be in the preset mode (MN or MQ command) before the MPA command will take effect.

Example

Command	Description
MN	Sets indexer to normal mode
MPA	Sets indexer to absolute position mode
A10	Sets acceleration to 10 rps^2
V10	Sets velocity to 10 rps
D10000	Sets move to position +10,000
G	Executes move
D2000	Sets move to position +2,000
G	Executes move

Motor goes to position +10,000 and then turns 8000 steps in a negative direction to position absolute +2000. In the absolute mode, issuing two G (go) commands in succession will cause the motor to move only once as the motor will have achieved the desired absolute position at the end of the first move.

MPI Set-Up	Mode Position Incremental			VALID Software Version 1.4
SYNTAX <a>MPI	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO MPA, D		

Description

The MPI command sets the position mode to incremental. In this mode, all move distances are referenced to the starting position of each move. You must be in the Preset mode (MN or MQ command) before the MPI command will take effect. MPI is the power-up default on the MN command.

Example

Command	Description
MN	Sets indexer to normal mode
MPI	Sets indexer to incremental position mode
A10	Sets acceleration to 10 rps^2
V10	Sets velocity to 10 rps
D8000	Sets distance to 8,000 cw steps
G	Executes move
D4000	Sets distance to 4,000 cw steps
G	Executes move

The motor turns two revolutions positive and then one more revolution positive.

MQ Set-Up	Speed Change Mode			VALID Software Version 1.4
SYNTAX <a>MQ	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO T, TRD, TRE, TRN		

Description

This command allows velocity to be changed using buffered commands during a preset move.

Example

Command	Description
MQ	Sets indexer to speed change mode
D80000	Sets distance to 80,000 cw steps
A200	Sets acceleration to 200 rps^2
V35	Sets velocity to 35 rps
G	Executes move
TRD60000	Sets distance at which velocity changes to 60,000 cw steps
V2	Sets velocity to 2 rps

The motor accelerates to a speed of 35 rps and when distance 60,000 is reached, it slows to 2 rps. It stops at position 80,000 exactly. TRD may be replaced by the TRE, TRN, or T commands.

MSY Set-Up	Mode Synchronous			VALID Software Version 2.3
SYNTAX <a>MSY	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO GL, GSY, PSY, VSY		

Description This command sets the mode to synchronous. In this mode moves may be started simultaneously on all axes.

Example

Command	Description
MSY	Set synchronous mode
A10	Set acceleration to 10 rps^2
V10	Set velocity to 10 rps
1D8000	Set axis 1 distance to 8000 steps
2D-4000	Set axis 2 distance to 4000 steps negative
GSY	Go - move axis 1 to position +8000 steps from the starting position and axis 2 to position -4000 from the starting position, starting the moves simultaneously.

N Programming	End Loop			VALID Software Version 1.4
SYNTAX <a>N	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO L, Y		

Description This command marks the end of a loop. When used in conjunction with the L command, it causes the buffered commands between the L and the N to be executed as many times as the number following L.

Example

Command	Description
L5	Loop 5 times
A1Ø	Sets acceleration to 10 rps^2
V1Ø	Sets velocity to 10 rps
DØØØØ	Sets distance to 8,000 (2 revs) CW steps
G	Executes move
N	End loop

The motor will turn 2 revolutions positive 5 times.

O Programming	Programmable Output			VALID Software Version 1.4
SYNTAX <a>Onnn	UNITS n = on/off	RANGE 1 = on 0 = off X = unchanged	DEFAULT 000	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO SS, OS		
RESPONSE TO <a>O IS (see below)				(*bbb)

Description The O command turns outputs on and off. The three outputs are set by the three digits nnn where the first n sets output #1, the second n output #2 and the third n output #3.

n = 1 sets the output to sink current (i.e., output on)
n = Ø turns the output current off
n = X leaves output unchanged

The response to the status request is as follows:

```
>1O
  Outputs are:
  O1/watchdog O2/in-position O3
  *bbb
```

Examples

The command 10111 will cause all outputs on device 1 to sink current from an externally supplied source.
The command O@@@ will cause current to stop flowing in all outputs. The command OXIX would leave outputs one and three unchanged while turning output 2 on.

Using the configuration switches it is possible to dedicate output 1 as the *Composite fault* line and/or output 2 as an *in-position* line. The O command will not control these outputs if they have been configured in this way.

OFF Programming	**Immediate Stop - Disable Drive**			**VALID** Software Version 1.5
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>OFF	N/A	N/A	N/A	Buffered Never Saved
EXECUTION TIME		**SEE ALSO** ST, ON		

Description

This command immediately disables the drive. This command may be used to quickly shut down the drive in an emergency. Issuing an SV command after the OFF command causes the drive to power up in the disabled state.

ON Programming	**Immediate Start - Enable Drive**			**VALID** Software Version 1.5
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>ON	N/A	N/A	N/A	Buffered Never Saved
EXECUTION TIME		**SEE ALSO** ST, OFF		

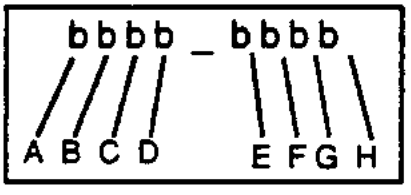
Description

This command immediately enables the drive. This command may be used to quickly restart the drive after a shutdown. Issuing an SV command after the ON command causes the drive to power up in the enabled state.

OS Status	**Other Switches**			**VALID** Software Version 1.4
SYNTAX aOS	**UNITS** None	**RANGE** None	**DEFAULT** None	**ATTRIBUTES** Buffered Never Saved
EXECUTION TIME		**SEE ALSO** OS commands		
RESPONSE TO aOS IS *bbbb_bbbb (b = 0 or 1)				

Description

This command reports the set up of the RS-232C control module functions as follows:



The command OS reports the state of the OS switches with a *bbbb_bbbb where the response pattern is **ABCDEFGH** (i.e., OSB, OSC, etc.).

OSB Set-Up	**Integral Action Selection**			**VALID** Software Version 1.4
SYNTAX aOSBn	**UNITS** n = function on/off	**RANGE** 0 or 1	**DEFAULT** 0	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME		**SEE ALSO** CIW		
RESPONSE TO aOSB IS *b				

Description

This command selects whether integral action will occur all the time or only when in position.

n = 1: Integral action will only occur when *in-position*
n = 0: Integral action occurs all the time

OSC Set-Up	Monitor Command Reporting			VALID Software Version 1.4
SYNTAX aOSCn	**UNITS** n = function on/off	**RANGE** 0 or 1	**DEFAULT** 0	**ATTRIBUTES** Buffered Independently Saved
EXECUTION TIME		**SEE ALSO** None		
RESPONSE TO aOSC IS *b				

Description: The OSC command selects continuous reporting of monitor commands (DS, DPA etc.) or reporting of one value only.

n=1: Signal monitor commands report only one value
n=0: Monitor commands report continuously

OSD Set-Up	Input 7 Sequence Select			VALID Software Version 1.4
SYNTAX aOSDn	**UNITS** n = function (seq. select/ no seq. select)	**RANGE** 0 = seq. select 1 = no seq. select	**DEFAULT** 0	**ATTRIBUTES** Buffered Independently Saved
EXECUTION TIME		**SEE ALSO** None		
RESPONSE TO aOSD IS *b				

Description This command configures Input 7 as a sequence select line enabling up to 16 sequence selections.

n = 1: Input 7 defined as sequence select
n = 0: Input 7 not sequence select

OSE Set-Up	Jog Enable			VALID Software Version 1.4
SYNTAX aOSEn	**UNITS** n = function enabled/ disabled	**RANGE** 1 = jog enabled 0 = jog disabled	**DEFAULT** 0	**ATTRIBUTES** Buffered Independently Saved
EXECUTION TIME		**SEE ALSO** JA, JV		
RESPONSE TO aOSE IS *b				

Description The OSE command enables or disables the jog function.

n = 1: jog enabled
n = 0: jog disabled

OSF Set-Up	Initialization on Limit			VALID Software Version 2.2
SYNTAX aOSFn	UNITS n = function on/off	RANGE 0 or 1	DEFAULT 0	ATTRIBUTES Buffered Independently Saved
EXECUTION TIME		SEE ALSO LD, OSI		
RESPONSE TO aOSF IS *b				

Description:

WARNING: This command is used to override a safety feature. Initial energization of the brushless drive involves a motion of the motor. With **OSFØ** (the default) set, the drive will not energize with a limit active and an error message is displayed in response to the energise command (**ON** or **STØ**). With **OSF1** set, the drive will energize with a limit active and the motion on initialization is allowed. *Care should be taken to ensure that the equipment will not be damaged by this motion if OSF1 operation is chosen.*

OSG Set-Up	Dither Enable			VALID Software Version 2.3
SYNTAX aOSGn	UNITS n = function on/off	RANGE 0 or 1	DEFAULT 0	ATTRIBUTES Buffered Independently Saved
EXECUTION TIME		SEE ALSO DTA, DTF, OSH		
RESPONSE TO aOSG IS *b				

Description

The OSG command is used to enable or disable the dither function, which is described in Section H.

n = 1: Dither enabled
n = 0: Dither disabled

OSH Set-Up	Select Triangular/Square Dither			VALID Software Version 2.3
SYNTAX aOSHn	UNITS n = function on/off	RANGE 0 or 1	DEFAULT 0	ATTRIBUTES Buffered Independently Saved
EXECUTION TIME		SEE ALSO DTA, DTG, OSG		
RESPONSE TO aOSH IS *b				

Description

The OSH command selects square or triangular wave dither signal.

n = 1: Square dither signal
n = 0: Triangular dither signal

OSI Set-Up	Initialization Movement			VALID Software Version 2.4
SYNTAX	UNITS	RANGE	DEFAULT	ATTRIBUTES
aOSIn	n = function std/reduced	0 or 1	0	Buffered Independently Saved
EXECUTION TIME		SEE ALSO None		
RESPONSE TO aOSI IS *b				

Description

The OSI command selects standard (n=0) or reduced (n=1) movement of the axis on initialization. It is for use with brushless non-hybrid servo motors to reduce the effect of axis movement as the motor comes to position at power up.

OSJ Set-Up	Select 24/16 Bit Gearbox			VALID Software Version 2.3
SYNTAX	UNITS	RANGE	DEFAULT	ATTRIBUTES
aOSJn	n = function 16/24	0 or 1	0	Buffered Independently Saved
EXECUTION TIME		SEE ALSO RAT		
RESPONSE TO aOSJ IS *b				

Description

The OSJ command selects 24 or 16 bit gearbox used with the RAT command for external pulse stream following.

n = 1: 24 bit gearbox operation —
RAT command range = +/-16777216

n = 0: 16 bit gearbox operation —
RAT command range = +/-65535

The 24 bit range has been included in this software version for higher resolution, but the 16 bit range has been retained for compatibility with earlier software verions.

P Status	Position			VALID Software Version 1.4
SYNTAX aP	**UNITS** None	**RANGE** None	**DEFAULT** None	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME		**SEE ALSO** G		
RESPONSE TO aP IS *±n (n = user steps)				

Description

This command displays the position relative to the start of the last index. It is cleared by the next G command. This is a single response in user steps. It can be used during a move.

PIC Status	Picture			VALID Software Version 1.4
SYNTAX aPIC	**UNITS** None	**RANGE** None	**DEFAULT** None	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME		**SEE ALSO** DR		
RESPONSE TO aPIC IS (see below)				

Description

This command displays a picture of the servo loop with the signal monitor numbers (see Figure J1-7 below). A device address is not necessary since any control module will return the same display.

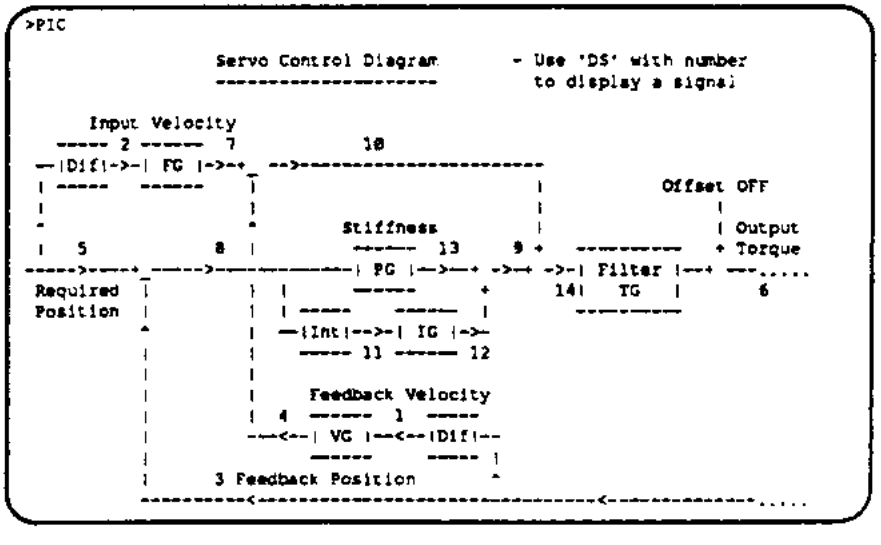


Figure J1-7. Servo Control Loop

PR Status	Position Report			VALID Software Version 1.5
SYNTAX aPR	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO MPA, MPI, D, PZ, SP		
RESPONSE TO aPR IS Absolute indexer position = ±nnn (*±nnn)				

Description

This command requests the current absolute position. The report is a number preceded by a sign and followed by a carriage return * (s) a<cr>. The range of the response is ±2,097,152. The number represents the cumulative position in user steps (n) with respect to the position at power up or the last point at which an SP or PZ command was issued to reset the absolute counter to zero. The sign indicates which side of the absolute zero position the motor is on.

Example

Command	Response
1PR	*-25600 (Motor is at absolute position -25600).

PS Programming	Pause			VALID Software Version 1.4
SYNTAX <a>PS	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO C		

Description

This command pauses the execution of the current command string or sequence. Execution will then be resumed after a C command (continue) is received. The command is used to allow the entering of a complete command string before the commands are executed.

Example

Command	Response
PS	Pause execution until a c command is received
A10	Sets acceleration to 10 rps^2
V10	Sets velocity to 10 rps
D25600	Sets distance to 25,600 cw steps
G	Executes move
C	Cancels Pause and executes move

The motor will not turn until the c command is entered.

PSY Programming	Pause for Synchronization			VALID Software Version 2.3
SYNTAX PSY	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO GSY, MSY, VSY		

Description

In synchronous mode (MSY), this command pauses the execution of buffered commands indefinitely while waiting for axes to be ready for a synchronized operation. Execution of subsequent commands will then be commenced almost simultaneously on all of the axes. The wait for readiness will be timed out after 2 seconds and the synchronized move aborted if this command has not been given.

Example

Command	Response
1MSY	Select Synchronous mode axis 1
2MSY	Select Synchronous mode axis 2
PSY	Wait for axes to be ready
O1	Set output 1 almost simultaneously on Axes 1 and 2

PZ Status	Position Zero			VALID Software Version 1.4
SYNTAX <a>PZ	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO MN, MPI, MPA, PR, D, SP		

Description

This command sets the current position to absolute zero.

Example

Command	Response
MPA	Sets indexer to absolute position mode
A10	Sets acceleration to 10 rps^2
V10	Sets velocity to 10 rps
D2500	Sets distance to 25,000 cw steps
G	Executes move
1DPA	Display absolute position
1PZ	Sets absolute position counter to zero
1DPA	Display absolute position

This will cause the motor to move to the absolute position +2500 counts from zero and report that position. Then the absolute feedback counter will be reset to zero (in effect moving the zero point) and the unit would report an absolute position of zero when the second 1DPA command is issued. Where a servo is used, due to offsets in the loop the demanded position and the feedback position may not be exactly equal. This command sets the physical or feedback position zero, so the demand position indicated by DPS may show a non-zero value.

QS Programming	Transmit an Identifier			VALID Software Version 1.4
SYNTAX <a>QS	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Immediate Savable in Sequence
EXECUTION TIME		SEE ALSO None		
RESPONSE TO <a>QS IS *nn!				

Description

The QS command responds with the identity of the echoing device. The command can be used to confirm which device is echoing after a series of universal commands. The response consists of an asterisk, a two-digit device address, an exclamation mark and a carriage return.

Example

Command	Response
MN	Sets indexer to normal mode
D4ØØØ	Sets distance to 4,000 cw steps
A1ØØ	Sets acceleration to 100 rps^2
V5Ø	Sets velocity to 50 rps
G	Executes move
1QS	Identify which device is echoing

If this is executed as a sequence by axis #1, the response *01! will be received over the RS-232C line when the sequence is complete.

R Status	Report RS-232C Control Module Status			VALID Software Version 1.4
SYNTAX aR	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO RA, RB, RC, RG, RS, OS, SS, XC, XSP, XSD, XSR, XSS, DFS, DFX		
RESPONSE TO aR IS *x				

Description

This command requests the status of the RS-232C control module. The possible responses are as follows:

*R = ready
*B = busy (when a command is being executed)
*S or *C is returned if any error conditions exist

Example

Command	Response
1R	*R (RS-232C control module ready to accept a command) If it's a buffered command, it will be executed immediately.

RA Status	Report A—Limit Status Request			VALID Software Version 1.4
SYNTAX aRA	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO R, RB		
RESPONSE TO aRA IS *x				

Description

The RA command requests the status of the limits. The response, in accordance with the following table, is the current and last move limit status @ <cr> to O <cr>.

Example

Command	Response
1RA	*I The current move is limited by the negative limit and the last move was terminated by the positive limit.

Response Character	Last Move Terminated by Positive	Last Move Terminated by Negative	Current Move Limited by Positive	Current Move Limited by Negative
*@	NO	NO	NO	NO
*A	YES	NO	NO	NO
*B	NO	YES	NO	NO
*C	YES	YES	NO	NO
*D	NO	NO	YES	NO
*E	YES	NO	YES	NO
*F	NO	YES	YES	NO
*G	YES	YES	YES	NO
*H	NO	NO	NO	YES
*I	YES	NO	NO	YES
*J	NO	YES	NO	YES
*K	YES	YES	NO	YES
*L	NO	NO	YES	YES
*M	YES	NO	YES	YES
*N	NO	YES	YES	YES
*O	YES	YES	YES	YES

The RA command is useful when the motor will not move in either or both directions. The report back will indicate whether or not the last move was terminated by a limit switch activation and if the current move is disabled by an active limit.

RAT Set-up	Set Rate Multiplier Value			VALID Software Version 2.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
aRATn	n = rate mult. value	-32,768 to +32,767	0	Buffered Independently Saved
EXECUTION TIME		**SEE ALSO** CCS, OSJ, SIM		

Description

This command is used in servo mode (ASC1) to configure a rate multiplier on the second encoder input when using the following mode of operation. The range of the command is determined by the setting of the OSJ command as follows:

OSJ = 0: RAT range -65536 to +65536
OSJ = 1: RAT range -16777216 to +16777216

The effect is that of a gearbox on the second encoder input that obeys the following formula:

Output rate to the motor (OSJ = 0) = $\frac{n}{65536}$ x input rate from the second encoder.

Output rate to the motor (OSJ = 1) = $\frac{n}{16777216}$ x input rate from the second encoder.

For a limited number of pulses:

Pulses to motor (OSJ = 0) = INT $\frac{n \times \text{pulses in}}{65536}$

Pulses to motor (OSJ = 1) = INT $\frac{n \times \text{pulses in}}{16777216}$

where INT means 'the integer part of'. The value is rounded down towards zero whether the value is positive or negative.

Negative values of RAT *allow the direction of following to be reversed.*

The number of second encoder input pulses is also affected by the CCS *command.*

Example

A 4000 step/rev motor is required to follow a 1000 line/rev encoder at half the encoder speed in the opposite direction. 1 revolution of the encoder produces 1000 x 4 (CCSØ set) = 4000 pulses; -2000 pulses are required to produce a half revolution of the motor shaft in the opposite direction.

Therefore, n = $\frac{65536 \times (-2000)}{4000}$ = -32768

Command	Description
1CCSØ	Select x 4 decode
1OSJØ	Set OSJ for 65536 RAT range
1SIM1	Select encoder following mode
1RAT-32768	If the tracked encoder turns 1 revolution clockwise, the number of pulses sent to the motor is:

$\frac{-32768 \times 4000}{65536}$ = -2000

The motor of axis 1 therefore turns a half revolution counter-clockwise.

RB Status	Report B—Miscellaneous Status Request			VALID Software Version 1.4
SYNTAX aRB	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO TR, PS, L, ST		
RESPONSE TO aRB IS *x				

Description

This command requests the status of a loop, a pause, a shutdown or an input. The response is @ <cr> to O<cr>according to the following table.

Response Character	Loop Active	Pause Active	Shutdown Active	Input Active
*@	NO	NO	NO	NO
*A	YES	NO	NO	NO
*B	NO	YES	NO	NO
*C	YES	YES	NO	NO
*D	NO	NO	YES	NO
*E	YES	NO	YES	NO
*F	NO	YES	YES	NO
*G	YES	YES	YES	NO
*H	NO	NO	NO	YES
*I	YES	NO	NO	YES
*J	NO	YES	NO	YES
*K	YES	YES	NO	YES
*L	NO	NO	YES	YES
*M	YES	NO	YES	YES
*N	NO	YES	YES	YES
*O	YES	YES	YES	YES

Loop active means that a loop is in progress.

Pause active means that buffered commands are not being executed and the RS-232C control module is waiting for a C command.

Shutdown active means that the motor is shutdown. Input active means that at least one input is active.

Example

Command	Response
1RB	*J There is no loop active, pause is active, the motor is not shutdown and there is an input active.

RE Status	Drive Status Request			VALID Software Version 1.4
SYNTAX aRE	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO ST, OFF, ON, ST		
RESPONSE TO aRE IS *x				

Description

This command requests the enabled/disabled status of the drive. The response is *0 if the drive is enabled or *1 if the drive is disabled.

RFS Set-Up	Return Servo to Factory (EPROM) Settings			VALID Software Version 1.4
SYNTAX <a>RFS	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Immediate Independently Saved
EXECUTION TIME		SEE ALSO Z, RIFS		

Description

This command returns the servo loop variables to the factory default settings. It brings up a set of example default sequences. ***These settings are not specific to the system - they will be the same for all motor/drive packages.***

The Z command restores the parameters to their values prior to entry of the RFS command if the settings have not been saved using the SV command.

After issuing the RFS command, the drive must be redefined using the #SD and #SC commands.

RG Status	Report Go Home Status			VALID Software Version 1.4
SYNTAX aRG	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO GH, GA, GHP, GHF		
RESPONSE TO aRG IS *x				

Description

This command requests the status of the last Go Home attempt. The response is 0 <cr> or A<cr>, indicating the success or failure of the last go home attempt as follows:

Response	Go Home Successful
*0	NO
*A	YES

RIFS Set-Up	**Return Indexer to Factory (EPROM) Settings**			**VALID** Software Version 1.4
SYNTAX <a>RIFS	**UNITS** None	**RANGE** None	**DEFAULT** None	**ATTRIBUTES** Immediate Independently Saved
EXECUTION TIME		**SEE ALSO** Z, RFS		

Description

This command will set up all the indexer parameters and values (distances, acceleration, velocities and sequences) to the factory setting. It brings up a set of example default sequences. ***These settings are not specific to the system.***

The **Z** command will restore the parameters to their values prior to entry of the **RIFS** command if the settings have not been saved using the **SV** command.

RPO Status	**Report Power-On Time**			**VALID** Software Version 1.4
SYNTAX aRPO	**UNITS** None	**RANGE** None	**DEFAULT** None	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME		**SEE ALSO**		
RESPONSE TO aRPO IS Power on time = nn.n hours				*nn.n

Description

This command requests the RS-232C control module power on time (hours) in decimal, for example 33.7 hours. A continuous record of usage is maintained by the control module.

RS Set-Up	Report Sequence Status			VALID Software Version 1.4
SYNTAX aRS	**UNITS** None	**RANGE** None	**DEFAULT** None	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME		**SEE ALSO** XR, XP		
RESPONSE TO aRS IS *x				

Description

This command requests the status of the last sequence execution. The response will be * <cr> to D<cr> according to the following table:

Response Character	Sequence Started	Sequence Ended
*@	NO	NO
*A	YES	NO
*B	NO	YES
*C	YES	YES
*D	NO	YES

Whenever a sequence is started, the sequence start bit is set and the sequence end bit is cleared (this only occurs if the sequence is valid and is actually run). Whenever a sequence is ended, the start bit is cleared and the end bit is set. Any abrupt move termination (e.g., limit activation), or a K or S command clears both bits.

*D is reported when there is an unbalanced number of loops and loop terminators inside a sequence. Starting a loop in one sequence and terminating it in another sequence is not allowed. Nested loops require complete closure before execution will begin.

Example

Command	Response
1RS	*A Sequence in progress.

RSE Set-Up	Report Servo Errors			VALID Software Version 1.6
SYNTAX aRSE	**UNITS** None	**RANGE** None	**DEFAULT** None	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME		**SEE ALSO** DR, DS		
RESPONSE TO aRSE IS (see below)			(*bbbb_bbbb_bbbb_bbbb)	

Description

This command returns all servo error flags as a set of error messages. Possible errors include:

```
#20 Shutdown by RS-232C command
#22 Shutdown by changed memory map (new software)
#23 Shutdown by an excessive position error
#24 Shutdown by a failed checksum of backup ram
#25 Shutdown by undefined drive
#26 Shutdown by prolonged max. torque at low speed
#27 Output to drive is Ø torque and drive de-energized
#28 Probable ESTOP input
#29 Shutdown by limit hit during initialization
```

The Bit pattern in the RSE command response *bbbb_bbbb_bbbb_bbbb is as follows (from right to left *bbbb_bbb8_7654_321Ø):

Bit Ø Shutdown by RS-232C command
Bit 1 Shutdown by limit hit during initialization
Bit 2 Shutdown by changed memory map (new software)
Bit 3 Shutdown by an excessive position error
Bit 4 Shutdown by a failed checksum of backup ram
Bit 5 Shutdown by undefined drive
Bit 6 Shutdown by prolonged maximum torque demand at low speed
Bit 7 Output to drive is Ø torque and drive de-energized
Bit 8 Probable ESTOP input

RV Status	Revision			VALID Software Version 1.7
SYNTAX aRV	**UNITS** None	**RANGE** None	**DEFAULT** None	**ATTRIBUTES** Buffered Never Saved
EXECUTION TIME		**SEE ALSO**		
RESPONSE TO aRV IS * ISSUE: n.n DATE: nn-xxx-nn nn:nn				

Description

With this command, you can request the system's software version number. An example response is as follows:

```
* ISSUE: 1.7        DATE: 30-April-90 15:32
```

This is the software revision and the date and time of the revision.

S Motion	Stop			VALID Software Version 1.4
SYNTAX <a>S	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Immediate Never Saved
EXECUTION TIME		SEE ALSO K		

Description

If SSØ (Don't save command buffer on stop) is set, the command buffer is cleared (at the end of a move if one is in progress).

A command sequence download is aborted (XD command) and a time delay is terminated.

The motor is decelerated to stop using the current acceleration value (A) but the drive stays energized.

If SS1 (Save command buffer on stop)is set, the command buffer is not cleared and only the move that is in progress is terminated.

Example

Command	Response
MC	Sets indexer to continuous mode
A1Ø	Sets acceleration to 10 rps^2
V1Ø	Sets velocity to 10 rps
G	Executes continuous move
S	Stops move

The motor will decelerate to a stop at a rate of 10 rps^2 as soon as S is entered.

SB Motion	Stop – Buffered			VALID Software Version 1.4
SYNTAX <a>SB	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO MC, MQ, S		

Description

The motor is decelerated to a stop using the current acceleration value (A). The command can be used in the MC or MQ modes.

SIM Set-up	**Set Indexer/Following Mode**			**VALID** Software Version 1.5
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
aSIMn	n = mode number	0 - 2	0	Buffered Independently Saved
EXECUTION TIME		**SEE ALSO** CCS		
RESPONSE TO aSIM IS *n				

Description

This command is used to configure the serial control module for following a second encoder. The mode of operation is selected by n as follows:

n = 0 Normal control module operation. In this mode any pulses arriving at the second encoder input are ignored.

n = 1 Encoder following operation. In this mode the position of the motor follows the pulse stream on the second encoder input and the control module motion commands are inactive.

n = 2 Super position operation. In this mode both the incoming pulses from the second encoder and control module motion commands are active. The position of the motor is the sum of both motions.

The stop commands and the resolution commands, CMR and CDR, do not affect the second encoder input. Enable and disable commands are effective.

Example

Command	Description
1SIM0	Set normal control mode
GH2	Go home
1SIM1	Set control by second encoder

SKE Programming	**Skip on Equals**			**VALID** Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>SKEn	n = input status	0, 1, or X	None	Buffered Savable in Sequence
EXECUTION TIME		**SEE ALSO** TRE		

Description

This command instructs the system to skip the following command on an input pattern equal to n. This command is used in sequences to control execution flow. The pattern, from left to right, is as follows:

HOME, LMT-, LMT+, +JOG, -JOG, EIN, INDEX TRACK - INPUT 1, INPUT 2, INPUT 3, INPUT 4, INPUT 5, INPUT 6, INPUT 7.

Each input can be Ø (off), 1 (on) or X (don't care). You can omit trailing X specifiers within each group of 7 specifiers. For example the following two specifiers are equivalent:

ØØ1XXXX_XXX11XX and **ØØ1_XXX11**

The next two are also equivalent:

XXXXXXX_X1XXXXX and **_X1**

Example

Command	Response
1SKIXX1	The next command will be skipped if the positive limit input is active

SKN Programming	**Skip on Not Equal**			**VALID** Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>SKNn	n = input status	0, 1, or X	None	Buffered Savable in Sequence
EXECUTION TIME		**SEE ALSO** TRN		

Description

This command instructs the indexer to skip the next command when the input pattern is not equal to n. The command is used in sequences to control execution flow. The format for n is nnnnnnn_nnnnnnn. This represents the state of all inputs to the control module. The pattern from left to right is shown below:

HOME, LMT-, LMT+, +JOG, -JOG, EIN, INDEX TRACK - INPUT 1, INPUT 2, INPUT 3, INPUT 4, INPUT 5, INPUT 6, INPUT 7.

Each input can be Ø (off), 1 (on) or X (don't care). You can omit trailing X specifiers within each group of 7 specifiers. For example the following two specifiers are equivalent:

ØØ1XXXX_XXX11XX and **ØØ1_XXX11**

The next two are also equivalent:

XXXXXXX_X1XXXXX and **_X1**

Example

Command	Description
1SKN1_	The next command will be skipped if the HOME input is not active

SP Programming	**Set Position**			**VALID** Software Version 2.2
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
aSP<s>n	n = motor steps	-268435455 to +268435455	0	Buffered Savable in Sequence
EXECUTION TIME		**SEE ALSO** MPI, MPA, PR, D, PZ		

Description

This command adds the value of n to the absolute position counter. The counter is set to n if a PZ command is given prior to the SP command. This command is useful for setting the absolute zero point to some location other than that of the physical hardware home. If you have a cut-off saw for example, you may not be able to mount the home switch at the cut point. However, by mounting the home a known distance away and resetting the reference point with the SP command, the system may be made to function as if the home switch were at the cut point.

The units of the SP command are scaled by the CMR command. You can get a position report in the same units by using the DPA command.

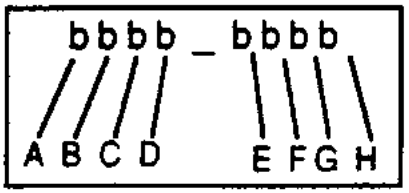
Example

Command	Description
GH2	Go Home at 2 rps
P2	Set position to zero
SP4000	The absolute position will be set to 4000 so that the absolute zero will be one motor revolution away from the switch location.

SS Status	**Software Switch Status**			**VALID** Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
aSS	None	None	None	Buffered Independently Saved
EXECUTION TIME		**SEE ALSO** SS commands		
RESPONSE TO aSS IS *bbbb_bbbb (b = 0 or 1)				

Description

This command displays the RS-232C control module option switches. The response is:



The response *bbbb_bbbb represents ABCDEFGH (i.e., SSA, SSB etc.).

SSA Set-Up	RS-232C Echo Control			VALID Software Version 1.4
SYNTAX	UNITS	RANGE	DEFAULT	ATTRIBUTES
aSSAn	n = echo on/off	0 or 1	0	Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO None		
RESPONSE TO aSSA IS *b				

Description

This command turns the RS-232C echo on and off.

- **SSA0** = Echo on
- **SSA1** = Echo off

In the Echo On mode, characters that are received by the RS-232C control module are echoed automatically. In the Echo Off mode, characters are not echoed. This command is useful if your computer cannot handle echoes.

Example

Command	Response
SSA1	Turns the echo off (Characters sent to the RS-232C control module are not echoed back to the host).

SSB Set-Up	Set Input 6 as a Clear Pause Input			VALID Software Version 1.4
SYNTAX	UNITS	RANGE	DEFAULT	ATTRIBUTES
aSSBn	n = function on/off	0 or 1	0	Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO IS		
RESPONSE TO aSSB IS *b				

Description

This command allows you to set input 6 to clear a pause.

n =0: normal
n =1: Input 6 clears a pause

SSC Set-Up	Set Output 2 as an In-Position Output			VALID Software Version 1.4
SYNTAX	UNITS	RANGE	DEFAULT	ATTRIBUTES
aSSCn	n = function on/off	0 or 1	0	Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO IO		
RESPONSE TO aSSC IS *b				

Description

This command allows you to set Output 2 as an In-position indicator output.

n =0: Normal
n =1: Output 2 is configured as an *in-position* output

SSD Set-Up	Set Output 1 as Composite Fault Signal			VALID Software Version 1.6
SYNTAX aSSDn	UNITS n = function on/off	RANGE 0 or 1	DEFAULT 0	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO IO		
RESPONSE TO aSSD IS *b				

Description

This command allows you to set Output 1 as a composit fault indicator output.

n = 0: Normal
n = 1: Output 1 is configured as a composite fault indicator

When functioning as a composite fault indicator, the output is normally on and any fault that disables the drive generated by software, watchdog time-out, or an incoming emergency stop signal will cause the output to turn off. This does not include the effect of ST1, which deliberately disables the drive.

SSE Set-Up	Set Input 5 as a Controlled Stop Input			VALID Software Version 1.4
SYNTAX aSSEn	UNITS n = function on/off	RANGE 0 or 1	DEFAULT 0	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO IS, LA, S		
RESPONSE TO aSSE IS *b				

Description

n = 0 : Normal
n = 1: Input 5 is configured as a controlled stop input

If you issue the SSE1 command and if Input 5 goes high, the axis will stop, using the LA setting.

SSF Set-Up	Set Input 4 as Sequence Strobe			VALID Software Version 1.5
SYNTAX	UNITS	RANGE	DEFAULT	ATTRIBUTES
aSSFn	n = function on/off	0 or 1	0	Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO IS, SN		
RESPONSE TO aSSF IS *b				

Description

n = 0: Normal
n = 1: Input #4 defined as a sequence strobe

When active, Input 4 causes execution of the sequence number found at the value on Inputs 1-3 or inputs 1-3 and input 7 if OSD1 is selected.

SSG Set-Up	Save Command Buffer on Limit			VALID Software Version 1.4
SYNTAX	UNITS	RANGE	DEFAULT	ATTRIBUTES
aSSGn	n = function on/off	0 or 1	0	Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO LD, SSH		
RESPONSE TO aSSG IS *b				

Description

n = 0: Normal
n = 1: Command buffer saved on limit

Normally, when a limit is reached, the current command buffer or sequence is scrapped thus preventing further execution. Setting this bit to 1 prevents this activity.

SSH Set-Up	Save Command Buffer on Stop			VALID Software Version 1.4
SYNTAX	UNITS	RANGE	DEFAULT	ATTRIBUTES
aSSHn	n = function on/off	0 or 1	0	Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO S, SSG		
RESPONSE TO aSSH IS *b				

Description

n =0: Normal
n =1: Save command buffer on Stop (S) command

ST Programming	Shutdown			VALID Software Version 1.5
SYNTAX <a>STn	**UNITS** n = function on/off	**RANGE** 0 or 1	**DEFAULT** 1	**ATTRIBUTES** Buffered Never Saved
EXECUTION TIME		**SEE ALSO** OFF, ON		

Description

Ø The drive is enabled and any brushless or stepper motor initialized if required.

1 The drive is disabled.

Any move commands given during motor shutdown will not be executed. **STØ** ramps the currents up to normal level (at the position read upon execution of the **ST1** command) and re-enables all move commands. This function is normally used to allow manual positioning of the load.

Example

Command	Description
ST1	Disables the drive

SV Programming	Save			VALID Software Version 1.4
SYNTAX <a>SV	**UNITS** None	**RANGE** None	**DEFAULT** None	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME		**SEE ALSO** RIFS, RFS		

Description

This command causes the current servo, set up, and indexer parameters to be saved in non-volatile RAM.

To successfully save data, jumper 1 must be in place. Remove the jumper to write protect the RAM.

T Programming	Time Delay			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>Tn	n = seconds	0.01 to 999.99	2	Buffered Savable in Sequence
EXECUTION TIME		**SEE ALSO** NONE		

Description

This command causes the RS-232C control module to wait the number of milliseconds specified before it executes the next command in the buffer.

Example

Command	Description
MN	Sets indexer to normal mode
A5	Sets acceleration to 5 rps^2
V5	Sets velocity to 5 rps
D25000	Sets distance to 25,000 cw steps
T.5	Sets time delay to 0.5 seconds
G	Executes move
T1	Sets time delay to 1 second
G	Executes move

Sets the mode to normal, acceleration to 5 rps^2, velocity to 5 rps and distance to 25,000 steps. Pauses the motor for 0.5 seconds, then executes the move. After a further delay of 1 second, it again executes the move. The resolution is 2 ms.

TRD Set-Up	Trigger on Input Distance			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>TRDn	n = user steps	-2097152 to +2097151	None	Buffered Savable in Sequence
EXECUTION TIME		**SEE ALSO** MQ		

Description

This command defines the moment at which the velocity changes when the RS-232C control module has reached a specified position within this move. The distance specified is always incremental.

Example

Command	Description
TRD3000	Set speed change distance to 3000 steps
V10	Change speed to 10 revs/sec

TRE Programming	Trigger on Input Equal			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>TREn	n = input status	0, 1, or X	None	Buffered Savable in Sequence
EXECUTION TIME		**SEE ALSO** MQ, SKE		

Description

Use this command to pause a move until the input status = nnnnnnn_nnnnnnn. The format nnnnnnn_nnnnnnn is from left to right:

HOME, LMT-, LMT+, +JOG, -JOG, EIN, INDEX TRACK - INPUT 1, INPUT 2, INPUT 3, INPUT 4, INPUT 5, INPUT 6, INPUT 7.

Each input can be Ø (off), 1 (on) or X (don't care). You can omit trailing X specifiers within each group of 7 specifiers. For example, the specifiers:

ØØ1XXXX_XXX11XX and ØØ1_XXX11 are equivalent.

The next two specifiers are also equivalent:

XXXXXXX_X1XXXXX and _X1

TRIP Programming	Trigger on In-Position			VALID Software Version 1.5
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>TRIP	None	None	None	Buffered Savable in Sequence
EXECUTION TIME		**SEE ALSO** MC, MQ, SSC		

Description

After use of this command, further buffered commands are not executed until the motor has stopped within the deadband region for the specified time. The command can be used in the MC or MQ modes.

Example

Command	Description
MQ	Sets indexer to speed change mode
1SSC1	Enables output 2 as an in-position output
V35	Sets velocity to 35 rps
A2ØØ	Sets acceleration to 200 rps^2
DØØØØØ	Sets distance to 80,000 steps
L	Loop
G	Executes move
TRIP	Delays next move until motor is stopped within deadband
N	End loop

The motor will accelerate to 35 rps and travel 80,000 steps. It will then wait in position before repeating the move.

TRN Programming	Trigger on Input Not Equal			VALID Software Version 1.4
SYNTAX <a>TRNn	UNITS n = Input Status	RANGE 0, 1, or X	DEFAULT None	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO SSC, SKN		

Description

This command pauses execution of a move until the input status is not equal to nnnnnnn_nnnnnnn. The format nnnnnnn_nnnnnnn is from left to right:

HOME, LMT-, LMT+, +JOG, -JOG, EIN, INDEX TRACK - INPUT 1, INPUT 2, INPUT 3, INPUT 4, INPUT 5, INPUT 6, INPUT 7.

Each input can be 0 (off), 1 (on) or X (don't care).
You can omit trailing X specifiers within each group of 7 specifiers. For example, the following specifiers are equivalent.

ØØ1XXXX_XXX11XX and ØØ1_XXX11

The next two specifiers are also equivalent:

XXXXXXX_X1XXXXX and _X1

TRR Set-up	Registration Mode			VALID Software Version 2.3
SYNTAX <a>TRRn	UNITS n - User steps	RANGE ±2097151	DEFAULT 0	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO SSC, SKN		

Description

This command selects a *registration move*, a move ending a specified distance in user steps after a mark signal appears at Input 6. The move is used in modes MC or MQ. Registration moves are described in detail in Section H.

TUNE Status	**Display Tuning Settings**			**VALID** Software Version 2.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
aTUNE<n>	N/A	1 - 11	None	Buffered Savable in Sequence
EXECUTION TIME		**SEE ALSO** DR, TUNET, TUNEV		
RESPONSE TO aTUNEn IS *(report of specified tuning setting - see below)				
RESPONSE TO aTUNE IS *(report of all tuning settings - see below)				

Description

The TUNE command with no parameter causes the control module to report tuning information calculated or measured during the previous move. The response format is as follows:

1. Measured largest error (steps)
2. Measured largest torque demand
3. Measured settling time within window (ms)
4. Measured total indexing + settling time (ms)
5. Calculated stiffness (mNm/step)
6. Calculated error at maximum torque (steps)
7. Estimated load inertia (Kg/cm^2)
8. Estimated maximum available acceleration ($revs/sec^2$)
9. Estimated double time constant (ms)
10. Estimated 1/2 settling time (ms)
11. Estimated closed loop bandwidth (Hz)

If a numeric parameter <n> is entered, any one of the above values as defined by the number will be reported.

Example

Command	Description
1TUNET	Self-tune servo
CIN2	Set narrow 'in position' window
D400	Set distance to 400 steps
A100	Set acceleration to 100 rps^2
V50	Set velocity to 50 rps
G	Go - start the move
1TUNE3	Report data item 3 (settling time) only

TUNET Status	Servo Self-Tuning (Torque amplifier settings)			VALID Software Version 2.4
SYNTAX	UNITS	RANGE	DEFAULT	ATTRIBUTES
<a>TUNET	None	None	None	Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO TUNE, TUNEV		

Description

This command self-tunes the servo as a torque amplifier. The following parameters are set:

CPG Maximized to the point where the standing digital vibration is reasonable.

CVG Set for stability — little overshoot beyond the end point of a move and no oscillations.

COFF Set so that the average error is zero. This cancels the effect of amplifier and gravitational offsets on positioning accuracy.

CTG Set to cut off the velocity feedback action at high frequency.

CIG Set to zero. Integral action is always de-stabilizing and a specialized feature associated with a requirement for extreme accuracy.

CFG Set to zero, assuming a point-to-point application. You should set this to the value of **CVG** (PID equivalent) if you have a following/contouring type application.

Example

Command	Description
1TUNET	Self-tune servo for torque amp.
1SV	Save parameters for future use

Safety

The tuning process moves the load a distance of 178 steps backwards and forwards using step changes of position demand that result in about 50% of the maximum available motor torque being suddenly applied to the load. This process continues for up to 30 seconds depending on the load conditions. During this time buffered commands typed at the keyboard or pending in a sequence will not be executed. However, the unit will respond to immediate commands such as **K**, **S** or **OFF**, to enabled end-of-travel inputs, to an enabled Input 5 stop input, and to the emergency stop input. If the tuning is interrupted in this way, the servo parameters will then be in an intermediate state and it may be necessary to cycle the power or use the **Z** command to restore them to a viable state.

TUNEV Status	Servo Self-Tuning (Velocity amplifier settings)			VALID Software Version 2.4
SYNTAX <a>TUNEV	UNITS None	RANGE None	DEFAULT None	ATTRIBUTES Buffered Savable in Sequence
EXECUTION TIME		SEE ALSO TUNE, TUNET		

Description

This command self-tunes the servo as a velocity amplifier. The process may also be used for pneumatic and hydraulic actuators. It is a limited process, maximising CPG starting from the value you enter so that there is little overshoot and no oscillations. The final value of CPG is dependent on the velocity loop gain which you have set. The following parameters are set:

- **CPG** Maximized, starting from the entered value and taken to the point where there is little overshoot beyond the end of move position and no oscillations occur.
- **CVG** Not altered — normally zero since damping is provided by the velocity amplifier.
- **COFF** Set so that the average error is zero. This cancels the effect of amplifier and gravitational offsets on positioning accuracy.
- **CTG** Not altered.
- **CIG** Set to zero. Integral action is always de-stabilizing and a specialized feature associated with a requirement for extreme accuracy.
- **CFG** Set to zero.

Example

Command	Description
1CPG50	CPG set for a low gain
1TUNEV	Self-tune servo for velocity amp.
1SV	Save parameters for future use

Safety

The tuning process moves the load a distance of 178 steps backwards and forwards using step changes of position demand that result in about 50% of the maximum available motor torque being suddenly applied to the load. This process continues for up to 30 seconds depending on the load conditions. During this time buffered commands typed at the keyboard or pending in a sequence will not be executed. However, the unit will respond to immediate commands such as K, S or OFF, to enabled end-of-travel inputs, to an enabled Input 5 stop input, and to the emergency stop input. If the tuning is interrupted in this way, the servo parameters will then be in an intermediate state and it may be necessary to cycle the power or use the Z command to restore them to a viable state.

U Programming	Pause			VALID Software Version 1.4
SYNTAX <a>U	**UNITS** None	**RANGE** None	**DEFAULT** None	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME		**SEE ALSO** PS, C		

Description

Use this command to pause execution of a command string at the point where a U command is entered. The C command will cause execution to be resumed at the point where it was paused.

Example

Command	Description
MN	Sets indexer to normal mode
A10	Sets acceleration to 10 rps^2
V10	Sets velocity to 10 rps
L	Begin Loop
D8000	Sets distance to 8,000 cw steps
G	Executes move
M-	Changes direction of move
G	Executes move
N	End loop

In this example, the motor turns 2 revolutions in the positive direction, then two revolutions in the negative direction in a continuous loop. If a U command is entered during the execution of the loop, the motor will stop at the end of its current move. Upon receipt of a C command, execution of the loop will be resumed from the point where it was paused.

V Motion	Velocity			VALID Software Version 1.5
SYNTAX <a>Vn	**UNITS** n = rps	**RANGE** 0.0001 to 100.00	**DEFAULT** 1	**ATTRIBUTES** Buffered Savable in Sequence
EXECUTION TIME		**SEE ALSO** A, CPE, D		
RESPONSE TO aV IS *V = nn revs/sec (*nn)				

Description

This command sets the velocity. If no rate is given, the last defined velocity is used for the move.

If the velocity is set too high, the following error will exceed the maximum position error (set by CPE) and the drive will be shut down. *The actual top speed of the motor is limited by the motor type and drive.*

Example

Command	Description
A10	Sets acceleration to 10 rps^2
V10	Sets velocity to 10 rps
D8000	Sets distance to 8,000 cw steps
G	Executes move

VSY Motion	Velocity Change in Synchronous Mode			VALID Software Version 2.2
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>VSYn	n = rps	0.0001 to 100.00	1	Buffered Savable in Sequence
EXECUTION TIME		**SEE ALSO** GSY, MSY, V		

Description

This command is used to produce a simultaneous speed change across a number of axes, all of which are running in modes MC or MQ and executing VSY commands. This is illustrated in the example shown, where axis 1 waits for VSY commands for the other axes in synchronous mode and then their speed changes are executed at the same time.

Example

For 2-axis synchronization:

Command	Description
MC	Set continuous mode on all axes
MSY	Set synchronous mode on all axes
1V2Ø	Set speed for axis 1 to 20 rps
2V1Ø	Set speed for axis 2 to 10 rps
G	Start all axes
1VSY1Ø	Axis 1 does not change speed
2VSY5	until axis 2 changes speed

XC Status	Checksum			VALID Software Version 1.5
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
aXC	None	None	None	Buffered Savable in Sequence
EXECUTION TIME		**SEE ALSO** None		
RESPONSE TO aXC IS (see below)				

Description

This command causes the checksums for the volatile and non-volatile RAM to be computed and reported .

```
>1XC
  Current ram checksum = nnnn
  Backup ram checksum = nnnn
  Computed backup ram checksum = nnnn
  Current map version = nn
  Backup map version = nn
```

XD Programming	Sequence Definition			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>XDn	n = sequence number	1 - 16	None	Buffered Never Saved
EXECUTION TIME		**SEE ALSO** XT, XE, XR, XRP		

Description

This is the *Start sequence* indicator for defining a sequence. It clears the command buffer and all subsequent commands will be stored in the buffer until receipt of an XT command. A sequence will not be stored if a sequence with the same number identifier already exists, or if there is an error in writing the sequence. *The sequence is not stored in nonvolatile memory until the SV command is given.*

Example

Command	Description
XD1	Starts definition of sequence 1
A1Ø	Sets acceleration to 10 rps^2
V1Ø	Sets velocity to 10 rps
D8ØØØ	Sets distance to 8,000 cw steps
G	Executes move
D4ØØØ	Sets distance to 4,000 cw steps
I	Changes direction
G	Executes move
XT	Ends definition of sequence 1

The sequence is defined as sequence #1. Each time XR1 is entered the motor will turn 2 revolutions positive and then 1 revolution negative.

XE Programming	Sequence Delete			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>XEn	n = sequence number	1 to 16	None	Buffered Independently Saved
EXECUTION TIME		**SEE ALSO** XT, XD, XR, XRP		

Description

Use this command to delete sequence n from RAM.

Example

Command	Description
XE1	Erases sequence 1
SV	Save

Sequence #1 is deleted from RAM.

XP Programming	Power-On Sequence Number			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>XPn	n = sequence number	1 to 16	None	Buffered Independently Saved
EXECUTION TIME		**SEE ALSO** XQ, XSR, XZ, XSP, IM, SS		
RESPONSE TO <a>XP IS Power on sequence number = *nn (*nn)				

Description

This command sets the power-on sequence mode and sequence number to be executed on power up. The command must be saved before power off to enable the sequence to be executed on power up. If the command XPØ is sent, no power-on sequence will be executed. This facility is normally used for stand alone operation.

Example

Command	Description
XP1	Specifies that sequence #1 will be executed on power-up

XR Programming	Run Sequence			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>XRn	n = sequence number	1 to 16	None	Buffered Savable in Sequence
EXECUTION TIME		**SEE ALSO** XT, XD, XE, XRP		

Description

An XR command can be used within one sequence to start execution of another sequence; all commands in the first sequence following the XR will be executed after the sequence called by the XR command is complete (in this respect an XR acts like a subroutine call).

You must be certain that the number of loops and loop terminators are balanced within a single sequence. You cannot start a loop in one sequence and end it in another called by the XR command.

Example

XR1 (Sequence #1 will be executed.)

XRP Programming	Run/Pause Sequence			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>XRPn	n = sequence number	1 to 16	None	Buffered Savable in Sequence
EXECUTION TIME		**SEE ALSO** XT, XD, XE, XR		

Description

The **XRP** command operates in the same way as **XR** except that a pause condition, which must be cleared before the command buffer is executed, is automatically generated. The pause condition is executed only if the sequence is valid and actually run.

Example

XRP1 (Sequence #1 is loaded into the command buffer and is paused until a continue [C command] is entered).

XRT Programming	Return From Sequence			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>XRT	None	None	None	Buffered Savable in Sequence
EXECUTION TIME		**SEE ALSO** XT		

Description

This command is used in a sequence to terminate execution of that sequence. It is typically used in branched sequences where the RS-232C control module must exit the sequence before the **XT** command.

Used in conjunction with **SKIP** commands.

XSD Status	Sequence Download Status Report			VALID Software Version 1.4
SYNTAX aXSD	**UNITS** None	**RANGE** None	**DEFAULT** None	**ATTRIBUTES** Buffered Never Saved
EXECUTION TIME		**SEE ALSO** XD		
RESPONSE TO aXSD IS *n				

Description

This command requests a status report of the previous sequence download from the controller to the indexer. The response is a single ASCII character followed by [CR]:

Ø: Download ok<cr>

1: Cannot overwrite existing sequence <cr> (sequence with this number already exists - erase previous one first)

2: Sequence buffer full<cr>

In order to retain the sequence it is necessary to save it using the SV command.

Example

Command	Response
XSD	*Ø (Sequence download ok)

XSR Status	Sequence Run Status Report			VALID Software Version 1.4
SYNTAX aXSR	**UNITS** None	**RANGE** None	**DEFAULT** None	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME		**SEE ALSO** XR, XRP		
RESPONSE TO aXSR IS *n				

Description

This command requests the sequence run status. The response is a single ASCII character followed by <cr>.

Ø<cr> = running (or ran successfully)
2<cr> = invalid sequence - requested sequence doesn't exist

Example

Command	Response
XSR	*Ø (sequence ok)

XSS	Sequence Status			VALID Software Version 1.4
SYNTAX aXSSn	**UNITS** n = sequence number	**RANGE** 1 to 16	**DEFAULT** None	**ATTRIBUTES** Buffered Never Saved
EXECUTION TIME		**SEE ALSO** XD, XT, XE		
RESPONSE TO aXSSn IS *n				

Description

This command requests the status of sequence n. The response is a single ASCII character followed by <cr>.

ø<cr> Empty - sequence doesn't exist
2<cr> OK - sequence does exist

The XSS command will inform the user if the sequence indicated by the number immediately following (no space) the XSS command does in fact exist in memory.

Example

Command	Response
XSS1	*ø (Sequence not defined)

XT Programming	Sequence Terminator			VALID Software Version 1.4
SYNTAX <a>XT	**UNITS** None	**RANGE** None	**DEFAULT** None	**ATTRIBUTES** Buffered Independently Saved
EXECUTION TIME		**SEE ALSO** XD		

Description

The XT command is used to terminate a sequence download, thus returning the system to command mode. The executable end of sequence is also marked at this point.

XU Programming	Sequence Upload			VALID Software Version 1.4
SYNTAX aXU<n>	**UNITS** n = sequence number	**RANGE** 1 to 16	**DEFAULT** None	**ATTRIBUTES** Immediate Never Saved
EXECUTION TIME		**SEE ALSO** XD, RIFS		
RESPONSE TO aXU<n> IS (see description)				

Description

The sequence upload command displays the sequences stored in RAM. The value n is the sequence number and if it is omitted a display of all stored sequences will occur. *The RS-232C control module is supplied with default sequences to get you started (see RIFS command).*

XZ Programming	Reset Power-Up Sequence Mode			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>XZ	None	None	None	Immediate Independently Saved
EXECUTION TIME		**SEE ALSO** Z, XP		

Description

This command sets the power-up sequence mode to zero (thereby disabling sequence activation on power-up). This is the same function as XP0 but it also calls the SV command to save this mode.

*Y	Terminate Loop			VALID Software Version 2.3
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
<a>Y	None	None	None	Immediate Never Saved
EXECUTION TIME		**SEE ALSO** L, N		

Description

This command terminates a loop at the end of the current pass.

Example

Command	Description
L	Begins loop
A10	Sets acceleration to 10 rps^2
V10	Sets velocity to 10 rps
D8000	Sets distance to 8,000 CW steps
T1000	Sets time delay to 1,000 ms
G	Executes move
W	Ends definition of loop
Y	Terminates loop

This loop will continuously cause the motor (4,000 steps/rev) to move 2 revolutions positive and then wait for 2 seconds. Loop will be terminated at the end of the pass during which the Y command is entered).

Z Programming	Reset			VALID Software Version 1.4
SYNTAX	**UNITS**	**RANGE**	**DEFAULT**	**ATTRIBUTES**
Z	None	None	None	Immediate Never Saved
EXECUTION TIME		**SEE ALSO** SV		

Description

This command applies a reset as if the power had been switched off and then on again. The command is universal and it returns all axes to their last saved settings - it cannot be made device-specific.

Command Summary

Help Command (HELP1)

HELP	Produce all help displays
HELP1	HELP list
HELP2	Indexer Commands
HELP3	Start/Stop Commands
HELP4	Sequence Commands
HELP5	Homing, Jog, Limit
HELP6	Option Set Commands
HELP7	Information Commands
HELP8	Inputs and Outputs
HELP9	Control and Memory Commands
HELP10	Using the Indexer
HELP11	Recommended Servo Settings
HELP12	Servo Setup Procedure

Motion Commands (HELP2)

A(n)	Set acceleration
C	Continue
D(n)	Set move distance
FDA	Set feed acceleration
FDV	Set feed velocity
G	Go, make a move
GL	Go linear
GSY	Go synchronous
H	Reverse motor direction
H+	Set positive direction
H-	Set negative direction
L(n)	Loop instructions
MAS	Mode asynchronous
MC	Move continuous
MN	Normal preset moves
MPA	Absolute position mode
MPI	Incremental position mode
MQ	Velocity change preset moves
MSY	Mode synchronous
N	End of loop indicator
PS	Wait for continue
PSY	Pause synchronous
SIM	Set indexer/following mode
T(n)	Delay for n secs
TRD	Pause until distance reached
TRE	Pause until inputs equal
TRIP	Pause until in position
TRN	Pause until inputs not equal
TRR	Pause for I6 registration
U	Pause, wait for continue
V(n)	Set velocity

Start/Stop Commands (HELP3)

K	Kill motion
LS	Stop now at LA deceleration
OFF	Immediate stop - de-energise
ON	Immediate start - re-energise
S	Stop motion, decelerate now
SB	Stop motion, buffered
ST0	Energise drive
ST1	De-energise drive
Y	Terminate loop or sequence
Z	Power up reset

Sequence Commands (HELP4)

XC	Report checksum
XD(n)	Begin download sequence
XE(n)	Delete sequence
XP(n)	Select power up sequence
XR(n)	Execute sequence
XRP(n)	Execute sequence with pause
XRT	Return from sequence
XSD	Report status of download
XSR	Report run sequence status
XSS(n)	Report sequence status
XT	Sequence terminator
XU(n)	Sequence upload
XZ	Turn off power up sequence mode

Homing, Jog, Limit Commands (HELP5)

GA(n)	Set go home acceleration
GH+/-(n)	Go home with direction + velocity
GHF(n)	Final home move velocity
JA(n)	Set jog acceleration
JV(n)	Set jog velocity
LA(n)	Set limit accn
LD(n)	Set limit function
PZ	Set current position as absolute zero
SP	Set current position to value

Option Set Commands (HELP6)

E	Enable communications
EX	1=terminal 0=computer mode
F	Disable communications
HS	Display H/W switches
HSA	1=resolver feedback 0=other
HSB	0=closed loop brushless
HSC	0=open loop stepper 1=other
HSD	0=no daughter board 1=board fitted
OS	Display 'OS' switches
OSB	1=Switched integral action on
OSC	1=Single value returned by OS
OSD	1=Input 7 is sequence select line
OSE	1=enable jog
OSF	1=allow initialise on limit
OSG	1=allow dither
OSH	1=square dither 0=triangular
OSI	1=reduced movement initialise
OSJ	1=24 bit gearbox 0=16 bit
SS	Display SS switches
SSA	0=echo on 1=echo off
SSB	1=Input 6 clears and store pause
SSC	1=Output 2 is in position
SSD	1=Output 1 is composite fault
SSE	1=Input 5 is stop line
SSF	1=Input 4 is sequence select
SSG	1=save comm buffer on limit
SSH	1=Save command buffer on stop

Information Commands (HELP7)

B	Command buffer status request
BS	Command buffer size request
DFX	Returns index status (32 bits)
DIC	Report Control Module position
DPA	Display position-actual
DPE	Continuous error display
DPS	Display position setpoint
DR	Display report
DVA	Display velocity-actual
DVS	Display velocity setpoint
P	Request position this move
PIC	Display servo loop diagram
PR	Request absolute position
QS	Send a response at move end
R	Status request
RA	Limit switch status request
RB	Pause status request
RE	Report drive status
RG	Go home status request
RPO	Report power on time
RS	Sequence status request
RSE	Report servo error conditions
RV	Report software revision

Inputs and Outputs (HELP8)

IO	Sets outputs immediately
IS	Reports input status
O	Turns outputs on and off
OSD	1=input 7 is sequence select line
SKE	Skip on inputs equal
SKN	Skip on inputs not equal
SSB	1=Input 6 clears pause
SSC	1=Output 2 is in position
SSD	1=Output 1 is composite fault
SSE	1=Input 5 is controlled stop line
SSF	1=Input 4 is sequence select strobe
SSG	1=save command buffer on limit
SSH	1=save command buffer on stop
TRD	Pause until distance reached
TRE	Pause until inputs equal
TRIP	Pause until in position
TRN	Pause until inputs not equal

Control and Memory Commands (HELP9)

CCS	Configure command source
CDG	Set D gain in PID
CEW	Set in position window
CFG	Set feed forward gain
CIG	Set integral action gain
CIT	In position wait time (2ms ints)
CIW	Set integral action window
CJL	Set motor + load inertia (Kg-cms^2)
CMR	Set motor resolution
COFF	Set amplifier offset
CPE	Set position error limit
CPG	Set forward path gain
CTG	Set filter time constant
CTQ	Set available motor torque (Nm)
CUR	Set user resolution
CVG	Set velocity feedback gain
DS	Signal display by number
OFF	Immediate de-energise
ON	Immediate energise
RFS	Servo to factory settings
RIFS	Indexer to factory settings
RSE	Reports servo error conditions
SIM	Select indexer operation
ST0	Energise drive
ST1	De-energise drive
SV	Save current settings in backup RAM
TUNE	Report tuning information
TUNET	Automatic torque amp tuning
TUNEV	Automatic velocity amp tuning

Alphabetical Command Listing

A	ACCELERATION RATE
B	BUFFER STATUS REQUEST
BS	BUFFER SIZE REQUEST
C	CONTINUE
CCS	CONFIGURE INPUT DECODE
CDG	CONFIGURE DERIVATIVE GAIN
CEW	CONFIGURE IN-POSITION WINDOW
CFG	CONFIGURE FEEDFORWARD GAIN
CIG	CONFIGURE INTEGRAL GAIN
CIT	CONFIGURE IN-POSITION TIME
CIW	CONFIGURE INTEGRAL ACTION WINDOW
CJL	ENTER MOTOR + LOAD INERTIA
CMR	CONFIGURE MOTOR RESOLUTION
COFF	CONFIGURE AMPLIFIER OFFSET
CPE	CONFIGURE POSITION ERROR
CPG	CONFIGURE FILTER TIME CONSTANT
CTQ	ENTER MOTOR TORQUE
CUR	CONFIGURE USER RESOLUTION
CVG	CONFIGURE VELOCITY GAIN
D	DISTANCE
DFX	DISPLAY FLAGS INDEXER
DIC	DISPLAY INDEXER COUNTER
DPA	DISPLAY POSITION ACTUAL
DPE	DISPLAY POSITION ERROR
DPS	DISPLAY POSITION SETPOINT
DR	DISPLAY REPORT
DS	DISPLAY SIGNAL
DTA	SET DITHER AMPLITUDE
DTF	SET DITHER FREQUENCY
DVA	DISPLAY VELOCITY ACTUAL
DVS	DISPLAY VELOCITY SETPOINT
E	ENABLE COMMUNICATIONS
EX	SET COMMUNICATION STYLE
F	DISABLE COMMUNICATIONS
FDA	ACCELERATION, LINEAR INTERP. MOVES
FDV	VELOCITY, LINEAR INTERP. MOVES
G	GO
GA	GO HOME ACCELERATION
GH	GO HOME
GHF	GO HOME FINAL
GL	GO, LINEAR MODE
GSY	GO, SYNCHRONOUS MODE
H	CHANGE DIRECTION
^H	BACKSPACE
H+	SET DIRECTION
H-	SET DIRECTION
HELP	PRODUCE HELP SCREENS
HS	DISPLAY H/W CONFIGURED SWITCHES
HSB	CONFIGURE MOTOR TYPE
HSC	CONFIGURE MOTOR TYPE
HSD	DAUGHTER BOARD OUTPUTS
IO	IMMEDIATE OUTPUT
IS	INPUT STATUS
JA	JOG ACCELERATION
JV	JOG VELOCITY
K	KILL
L	LOOP
LA	LIMIT DECELERATION
LD	LIMIT DISABLE
LS	LIMIT SWITCH FAST STOP
MAS	MODE ASYNCHRONOUS
MC	MODE CONTINUOUS
MN	MODE NORMAL
MPA	MODE POSITION ABSOLUTE
MPI	MODE POSITION INCREMENTAL
MQ	SPEED CHANGE MODE
MSY	MODE SYNCHRONOUS
N	END LOOP
O	PROGRAMMABLE OUTPUT
OFF	IMMEDIATE STOP
ON	IMMEDIATE START
OS	OTHER SWITCHES
OSB	INTEGRAL ACTION SELECTION
OSC	MONITOR COMMAND REPORTING
OSD	INPUT 7 SEQUENCE SELECT
OSE	JOG ENABLE
OSF	INITIALIZATION ON LIMIT
OSG	DITHER ENABLE
OSH	SELECT TRIANGULAR/SQUARE DITHER
OSI	INITIALIZATION MOVEMENT
OSJ	SELECT 24/16 BIT GEARBOX
P	POSITION
PIC	PICTURE
PR	POSITION REPORT
PS	PAUSE
PSY	PAUSE FOR SYNCHRONIZATION
PZ	POSITION ZERO
QS	TRANSMIT AN IDENTIFIER
R	REPORT CONTROL MODULE STATUS
RA	REPORT A - limit status request
RAT	SET RATE MULTIPLIER VALUE
RB	REPORT B - miscellaneous status request
RE	DRIVE STATE REQUEST
RFS	RETURN SERVO TO FACTORY SETTINGS
RG	REPORT GO HOME STATUS
RIFS	RETURN INDEXER TO FACTORY SETTINGS
RPO	REPORT POWER-ON TIME
RSE	REPORT SERVO ERRORS
RS	REPORT SEQUENCE STATUS
RV	REVISION
S	STOP
SB	STOP BUFFERED
SKE	SKIP ON 'EQUALS'
SKN	SKIP ON 'NOT EQUAL'
SP	SET POSITION
SS	SOFTWARE SWITCHES STATUS
SSA	RS-232C ECHO CONTROL
SSB	SET INPUT 6 = CLEAR PAUSE INPUT
SSC	SET OUTPUT 2 = IN-POSITION OUTPUT
SSD	SET OUTPUT 1 AS COMPOSITE FAULT SIGNAL
SSE	SET INPUT 5 = CONTROLLED STOP INPUT
SSF	SET INPUT 4 AS SEQUENCE STROBE
SSG	SAVE COMMAND BUFFER ON LIMIT
SSH	SAVE COMMAND BUFFER ON STOP
ST	SHUTDOWN
SV	SAVE
SAVE	SAVE
T	TIME DELAY
TRD	TRIGGER ON INPUT DISTANCE
TRE	TRIGGER ON INPUT EQUAL
TRP	TRIGGER ON IN POSTION
TRN	TRIGGER ON INPUT NOT EQUAL
TRR	REGISTRATION MODE
TUNE	DISPLAY TUNING SETTINGS
TUNET	SERVO SELF-TUNING (TORQUE AMP.)
TUNEV	SERVO SELF-TUNING (VELOCITY AMP.)
U	PAUSE
V	VELOCITY
VSY	VELOCITY CHANGE IN SYNCHRONOUS MODE
XC	CHECKSUM
XD	SEQUENCE DOWNLOAD
XE	SEQUENCE DELETE
XP	POWER-ON SEQUENCE NUMBER
XR	RUN SEQUENCE
XRP	RUN/PAUSE SEQUENCE
XRT	RETURN FROM SEQUENCE
XSD	SEQUENCE DOWNLOAD STATUS REPORT
XSR	SEQUENCE RUN STATUS REPORT
XSS	X SEQUENCE STATUS
XT	SEQUENCE TERMINATOR
XU	SEQUENCE UPLOAD
XZ	RESET POWER-UP SEQUENCE MODE
Y	TERMINATE LOOP
Z	RESET

Section K. Maintenance & Troubleshooting

Section Objectives

This section will help you identify the part of your system (unit) that developed a fault. This should enable you to get the equipment working again with minimal difficulty. The LEDs mounted on some modules indicate the nature of a fault. Using this information, you can find a fault and prevent a recurrence of the problem.

Maintenance

CAUTION

You should never dismantle a suspect motor. If you do, you will demagnetize the permanent magnets. After re-assembly, the motor may still run, but it will not be capable of giving its full performance.

Your system will not normally need any routine maintenance, other than for the few items covered below.

Filters

If you mounted the electronics rack in a cabinet through which filtered cooling air is passed, you will need to inspect and clean the filters at regular intervals. Failure to do this may result in reduced cooling, leading to overheating and possible failure of the electronics.

Cables

If any of your cabling is exposed, you should inspect the cable regularly inspection to prevent a machine breakdown.

Connectors

Over a period of time, and particularly if subjected to vibration, connectors may come loose. Inspect the connectors regularly to prevent this problem.

Fault Diagnosis

To minimize field failures, all electronic units are fully tested on automatic test equipment and then soak-tested at elevated temperatures before they are shipped. In spite of these precautions, however, occasional failure may still occur.

WARNING

Before you attempt any work on the system, ensure that AC power is disconnected, and that the HV supply is given sufficient time (at least 5 seconds) to reach a safe level.

Problem Isolation

If your system malfunctions, you must identify and isolate the problem. When you accomplish this, you can begin to eradicate and resolve the problem.

The first step is to isolate each system component and ensure that each component functions properly when it is run independently. You may have to dismantle your system and put it back together piece by piece to detect the problem. If you have additional units available, you may want to use them to replace existing components in your system to help identify the source of the problem.

Try to determine if the problem is mechanical, electrical, or software-related. *Can you repeat or re-create the problem?* Do not attempt to make quick rationalizations about problems. Random events may appear to be related, but they are not necessarily contributing factors to your problem. You must carefully investigate and decipher the events that occurred before the subsequent system problem.

You may be experiencing more than one problem. You must solve one problem at a time. Log (document) all testing and problem isolation procedures. You may need to review and consult these notes later. This will also prevent you from duplicating your testing efforts.

Once you isolate the problem, take the necessary steps to resolve it. Refer to the problem solutions contained in this chapter. If your system's problem persist, contact Compumotor's Applications Department.

Power Supply and Drive Fault LEDs

The power supply and drive modules of your system are fitted with several LEDs that indicate certain fault conditions. If a fault develops in the system, you should first check these LEDs.

Table K1-1 will help you to identify the causes of common faults related to the power supply and the drive.

PSU LED not illuminated	AC supply not connected. AC fuse failure (fuses located on PSU motherboard)
Drive logic supply LED is not illuminated	If this LED is not illuminated, but the drive still functions, the logic supply may have dropped for a short period and then returned to normal. If you cycle the drive *enable* signal, the LED should illuminate. If the drive does not function, check that HV is present (the green LED on the PSU should be illuminated). If it is, the problem may be due to excessive load on the external 5V supply. Check that the maximum current rating (0.75A) is not being exceeded.
Drive control fault LED illuminated	This indicates loss of the torque demand signal from the control card.This could be due to a fault either in the control card, the drive, or less likely in the interconnection via the motherboard.
Drive overload LED illuminated	Indicates a short circuit across the motor terminals, or between one terminal and earth. Check the connections to the drive and examine the motor cable. If necessary, check the motor windings with an ohm-meter, after first disconnecting the motor cable from the drive.
Drive HV over-voltage LED illuminated	This will happen if excessive regeneration occurs. To confirm this, check that the fault occurs during deceleration; if so, first check the dump fuse (FS1) on the PSU. If this is intact, it may be necessary to increase the rating of the power dump resistors. Before you do this, contact Compumotor for further advice and fitting instructions.
Drive over-temperature LED illuminated	Over-temperature faults can be caused by a failure within the drive or motor, an elevated ambient temperature, or poor ventilation. Check or replace the motor and check ventilation fans and filters (where fitted) and ambient temperature. If there is no problem in these areas, the drive itself is probably faulty. *The LEDs on the Drive Module are latched in the fault condition and the drive is disabled.* To reset the latch, you should remove and re- apply the Enable signal.
Poor system high speed performance	Failure of one AC fuse when a 3-phase supply is being used. If one phase of the supply is lost, the system will continue to function, but with a reduced supply voltage under heavy load giving rise to reduced high-speed performance.
Power dump fuse fails on switch on	Failure of power dump circuit. Power dump transistor (Q5 on PSU) short circuit.
Power dump fuse fails during system operation.	Excessive dump current. This is most likely to be caused by excessive regeneration and may be accompanied by an HV over-voltage indication on the drive and overheating of the power dump resistors. If this is the case, a higher wattage power dump resistor may be needed. Before attempting to fit this, you should contact us for further advice and instructions.
Wiring Problems	If all appears to be well, but the system does not operate, you should carefully check all motor and signal connections and wiring (for breakage and loose or incorrect connection). An incorrectly wired limit switch, for example, may prevent motor rotation in one direction while allowing rotation in the other.

Table K1-1. Power Supply and Drive Module Fault Conditions

Control Module Fault LED

A single LED, visible through the front panel, functions as a combined fault indicator. If the control module disables the drive when no terminal is connected to the step and direction control module, you can count the number of LED flashes to determine the first reason for the disabling (see Table K2-2). The sequence of flashes continues until the fault is cleared. A one- to two-second delay separates the LED flash sequences.

RS-232C Diagnostics and Problem-Solving

Via an RS-232C terminal or computer, you can use the System 7 status commands to locate faults. Once you locate the faults you can often resolve the problem by tuning the system with the tuning set-up commands. The most commonly used status command for diagnostics is the Report Servo Error Conditions (RSE) command. This command reports the cause of the fault condition that disabled the drive. Other useful diagnostic status commands are as follows:

DFX	Display interface status flags
R	Status request
RA	Limit switch status request
RB	Miscellaneous status request
RG	Go Home status request
RS	Sequence status request
XC	Report checksum
XSD	Sequence download status request
XSR	Sequence run status request
XSS	Sequence status

Refer to Section J2, *Software Reference*, for status and set-up command descriptions. Refer to Section G1, *Control Module*, for RS-232C connection instructions.

Table K1-2 provides solutions to common parameter set-up problems.

LED off	No fault
LED flashes once	The ST1 or OFF command has been sent. Send the ST0 or ON command to re-enable the drive.
LED flashes three times	The EPROM has been fitted with a different memory map. Check that the EPROM has the correct part number.
LED flashes four times	There has been a following error exceeding the value set by the CPE command. Reset the value of CPE if required and re-enable with ST0.
LED flashes five times	The checksum failed at power up.
LED flashes seven times	There has been a prolonged maximum torque demand causing the drive to disable. Investigate the excessive torque requirement.
LED flashes eight times	The ESTOP input has been seen by the indexer. To re-enable the drive, make sure the ESTOP input is connected to +12V.

Table K1-2. Control Module Fault Conditions

Electrical Noise

If the system is affected by electrical noise, it will normally manifest itself as erratic motor movement (the commanded position is not reached), or as movement of a motor which you expect to see at rest. Sources of electrical noise can often be difficult to find. All electrical systems are potential generators of electrical noise.

If you are having problems, refer to Section C1, *Installing the System*, and ensure that you have followed the recommendations. You should also consult the *Technical Data Section* of the Compumotor catalog.

Communications Problems

This section helps you diagnose problems related to RS-232C communications.

- Check the terminal communication parameters set up as follows:

  9600 baud
  8 data bits
  no parity
  1 start bit
  1 stop bit

- Check that the terminal's Tx line is connected to the Control Module's Rx line and vice versa.
- Check that upper case characters are used.
- Send [CTRL] Q if XON/XOFF communications have been used.
- Send `E` to ensure that the `F` command has not been used to disable communication.
- Type <CR>1 to set echo back.
- Send `nEX1` to set terminal mode if there is no prompt.
- Send `SSRØ` and then `S` to clear the command buffer.
- Check the status with the `R` command. The response should be `*R` if the controller is ready to accept a command or `*B` if it is busy.

Drive Control Problems

This section helps you diagnose problems related to drive control. If the communication is working correctly but the drive cannot be enabled, check the following:

- Drive may have to be defined by using the `DR` command. If the response is that the drive is undefined, use the `HSB` and `HSC` commands to configure the drive/motor type.
- The ESTOP input may be active. Current must be sourced to the ESTOP input to allow the drive to be enabled. If a switch is fitted, it should be normally-closed and connected to the +12V supply.
- Send `S` to clear the command buffer.
- Check the status of the servo error flags using the `RSE` command and note the number of flashes of the Fault LED on the control module.

Motor Control Problems

This section helps you diagnose problems related to motor control.

Motor Velocity Incorrect

If the motor does not run at the expected speed, check the following:

- Velocity signal
- HV supply voltage
- Main fuse on the three-phase supply
- Definition of the motor/drive (`HSB` command)
- Motor resolution (`CMR` command)
- User resolution (`CUR` command)
- Servo loop tuning
- Available motor torque
- Condition of the load
- Motor/Encoder connections
- Motor winding resistances (Do not strip the motor or its magnets may be affected.)

Motor Acceleration/ Deceleration Incorrect

If the motor does not accelerate or decelerate as expected, check the following:

- Input acceleration rate demand
- Condition of the load
- Available motor torque
- Servo loop tuning

Returning The System

If you must return your system to effect repairs or upgrades, use the following steps:

1. Get the serial number and the model number of the defective unit, and a purchase order number to cover repair costs in the event the unit is determined by Parker Compumotor to be out of warranty.

2. Before you ship the system to Parker Compumotor, have someone from your organization with a technical understanding of the system and its application include answers to the following questions:

   - What is the extent of the failure/reason for return?
   - How long did it operate?
   - How many units are still working?
   - How many units failed?
   - What was happening when the unit failed (i.e., installing the unit, cycling power, starting other equipment, etc)?
   - How was the product configured (in detail)?
   - What, if any, cables were modified and how?
   - With what equipment is the unit interfaced?
   - What was the application?
   - What was the system sizing (speed, acceleration, duty cycle, inertia, torque, friction, etc.)?
   - What was the system environment (temperature, enclosure, spacing, unit orientation, contaminants, etc.)?
   - What upgrades, if any, are required (hardware, software, user guide)?

3. Call Parker Compumotor for a Return Material Authorization (RMA) number. Returned products cannot be accepted without an RMA number. The phone number for Parker Compumotor Applications Department is (800) 358-9070.

4. Ship the unit to: Parker Compumotor Corporation
   5500 Business Park Drive
   Rohnert Park, CA 94928
   Attn: RMA # xxxxxxx

Summary

Using the fault LEDs and the information in this section, you should be able to identify and resolve most of the faults that arise within the system.

Appendices

Warranty

The items described in this document are hereby offered for sale at prices to be established by Parker Hannifin Corporation, its subsidiaries, and its authorized distributors. This offer and its acceptance by any customer (*Buyer*) shall be governed by all of the following Terms and Conditions. Buyer's order for any item described in its document, when communicated to Parker Hannifin Corporation, its subsidiary, or an authorized distributor (*Seller*) verbally or in writing, shall constitute acceptance of this offer.

1. Terms and Conditions of Sale
All descriptions, quotations, proposals, offers, acknowledgements, acceptances, and sales of Seller's products are subject to and shall be governed exclusively by the terms and conditions stated herein. Buyer's acceptance of any offer to sell is limited to these terms and conditions. Any terms or conditions in addition to, or inconsistent with those stated herein, proposed by Buyer in any acceptance of any offer by Seller, are hereby objected to. No such additional, different or inconsistent terms and conditions shall become part of the contract between Buyer and Seller unless expressly accepted in writing by Seller. Seller's acceptance of any offer to purchase by Buyer is expressly conditional upon Buyer's assent to all the terms and conditions stated herein, including any terms in addition to, or inconsistent with those contained in Buyer's offer. Acceptance of Seller's products shall in all events constitute such assent.

2. Payment
Payment shall be made by Buyer net 30 days from the date of delivery of the items purchased hereunder. Amounts not timely paid shall bear interest at the rate of 1.5% for each month or a portion thereof that Buyer is late in making payment. Any claims by Buyer for omissions or shortages in a shipment shall be waived unless Seller receives notice thereof within 30 days after Buyer's receipt of the shipment.

3. Delivery
Unless otherwise provided on the face hereof, delivery shall be made F.O.B. Seller's plant. Regardless of the method of delivery, however, risk of loss shall pass to Buyer upon Seller's delivery to a carrier. Any delivery dates shown are approximate only and Seller shall have no liability for any delays in delivery.

4. Warranty
Seller warrants that the items sold hereunder shall be free from defects in material or workmanship for a period of 2 years from the date of shipment to Buyer. THIS WARRANTY COMPRISES THE SOLE AND ENTIRE WARRANTY PERTAINING TO ITEMS PROVIDED HEREUNDER. SELLER MAKES NO OTHER WARRANTY, GUARANTEE, OR REPRESENTATION OF ANY KIND WHATSOEVER. ALL OTHER WARRANTIES, INCLUDING BUT NOT LIMITED TO, MERCHANTABILITY AND FITNESS FOR PURPOSE, WHETHER EXPRESS, IMPLIED, OR ARISING BY OPERATION OF LAW, TRADE USAGE, OR COURSE OF DEALING ARE HEREBY DISCLAIMED. NOTWITHSTANDING THE FOREGOING, THERE ARE NO WARRANTIES WHATSOEVER ON ITEMS BUILT OR ACQUIRED WHOLLY OR PARTIALLY, TO BUYER'S DESIGNS OR SPECIFICATIONS.

5. Limitation of Remedy
SELLER'S LIABILITY ARISING FROM OR IN ANY WAY CONNECTED WITH THE ITEMS SOLD OR THIS CONTRACT SHALL BE LIMITED EXCLUSIVELY TO REPAIR OR REPLACEMENT OF THE ITEMS SOLD OR REFUND OF THE PURCHASE PRICE PAID BY BUYER, AT SELLER'S SOLE OPTION. IN NO EVENT SHALL SELLER BE LIABLE FOR ANY INCIDENTAL, CONSEQUENTIAL, OR SPECIAL DAMAGES OF ANY KIND OR NATURE WHATSOEVER, INCLUDING BUT NOT LIMITED TO LOST PROFITS ARISING FROM OR IN ANY WAY CONNECTED WITH THIS AGREEMENT OR ITEMS SOLD HEREUNDER, WHETHER ALLEGED TO ARISE FROM BREACH OF CONTRACT, EXPRESS OR IMPLIED WARRANTY, OR IN TORT, INCLUDING WITHOUT LIMITATION, NEGLIGENCE, FAILURE TO WARN, OR STRICT LIABILITY.

6. Changes, Reschedules and Cancellations
Buyer may request to modify the designs or specifications from the items sold hereunder as well as the quantities and delivery dates thereof, or may request to cancel all or part of this order. However, no such requested modification or cancellation shall become part of the contract between Buyer and Seller unless accepted by Seller in a written amendment to this Agreement. Acceptance of any such requested modification or cancellation shall be at Seller's discretion, and shall be upon such items and conditions as Seller may require.

7. Special Tooling
A tooling charge may be imposed for any special tooling, including without limitation, dies, fixtures, molds and patterns, acquired to manufacture items sold pursuant to this contract. Such special tooling shall be and remain Seller's property notwithstanding payment of any charges therefore by Buyer. In no event will Buyer acquire any interest in apparatus belonging to Seller which is utilized in the manufacture of the items sold hereunder, even if such apparatus has been specially converted or adapted for such manufacture and notwithstanding any charges paid by Buyer therefor. Unless otherwise agreed, Seller shall have the right to alter, discard or otherwise dispose of any special tooling or other property in its sole discretion at any time.

8. Buyer's Property
Any designs, tools, patterns, materials, drawings, confidential information, or equipment furnished by Buyer or any other

items that become Buyer's property, may be considered obsolete and may be destroyed by Seller after two (2) consecutive years have elapsed without Buyer placing an order for the items that are manufactured using such property. Seller shall not be responsible for any loss or damage to such property while it is in Seller's possession or control.

9. Taxes

Unless otherwise indicated on the face hereof, all prices and changes are exclusive of excise, sales, use, property, occupational or like taxes which may be imposed by any taxing authority upon the manufacture, sale, or delivery of the terms sold hereunder. If any such taxes must be paid by Seller or if Seller is liable for the collection of such tax, the amount thereof shall be in addition to the amount for the items sold. Buyer agrees to pay all such taxes or to reimburse Seller therefore upon receipt of its invoice. If Buyer claims exemption from any sales, use, or other tax imposed by any taxing authority, Buyer shall save Seller harmless from and against any such tax, together with any interest or penalties thereon which may be assessed if the items are held to be taxable.

10. Indemnity for Infringement of Intellectual Property Rights

Seller shall have no liability for infringement of any patents, trademarks, copyrights, trade dress, trade secrets, or similar rights except as provided in this Part 10. Seller will defend and indemnify Buyer against allegations of infringement of U.S. patents, U.S. trademarks, copyrights, trade dress, and trade secrets (hereinafter 'Intellectual Property Rights'). Seller will defend at its expense and will pay the cost of any settlement or damages awarded in any action brought against Buyer based on an allegation that an item sold pursuant to this contract infringes the Intellectual Property Rights of a third party. Seller's obligation to defend and indemnify Buyer is contingent on Buyer notifying Seller within ten (10) days after Buyer becomes aware of such allegations of infringement, and Seller having sole control over the defense of any allegations or actions including all negotiations for settlement or compromise. If an item sold hereunder is subject to a claim that it infringes the Intellectual Property Rights of a third party, Seller may, at his sole expense and option, procure for Buyer the right to continue using said item, replace or modify said item so as to make it non-infringing, or offer to accept return of said item and return the purchase price less a reasonable allowance for depreciation. Notwithstanding the foregoing, Seller shall have no liability for claims of infringement based on information provided by Buyer, or directed to items delivered hereunder for which the designs are specified in whole or part by Buyer, or infringements resulting from the modification, combination, or use in a system of any item sold hereunder. The foregoing provisions of this Part 10 shall constitute Seller's sole and exclusive liability and Buyer's sole and exclusive remedy for infringement of Intellectual Property Rights.

If a claim is based on information provided by Buyer or if the design for an item delivered hereunder is specified in whole or in part by Buyer, Buyer shall defend and indemnify Seller for all costs, expenses, or judgments resulting from any claim that such item infringes any patent, trademark, copyright, trade dress, trade secret, or any similar right.

11. Force Majeure

Seller does not assume the risk of and shall not be liable for delay or failure to perform any of Seller's obligations by reason of circumstances beyond the reasonable control of Seller (hereinafter 'Events of Force Majeure'). Events of Force Majeure shall include without limitation, accidents, acts of God, strikes or labor disputes, acts, laws, rules or regulations of any government or government agency, fires, floods, delays or failures in delivery of carriers or suppliers, shortages of materials, and any other cause beyond Seller's control.

12. Entire Agreement/ Governing Law

The terms and conditions set forth herein, together with any amendments, modifications, and any different terms or conditions expressly accepted by Seller in writing, shall constitute the entire Agreement concerning the items sold, and there are no oral or other representations or agreements which pertain thereto. This Agreement shall be governed in all respects by the law of the State of Ohio. No actions arising out of the sale of the items sold hereunder of this Agreement may be brought by either party more than two (2) years after the cause of action accrues.

Index

7-signal link A1-4

Absolute mode H1-5
Absolute zero J1-67
Accessibility C1-2
Address settings H1-22
Asynchronous mode H1-7

Bench test configuration B1-1
Buffered commands J1-1
Buffers H1-10, J1-1

Command buffer J1-1
Communications problems K1-5
Computer Host mode J1-1
Continuous moves H1-5
Control module (RS-232C) A1-4
  address settings H1-22
  fault conditions K1-3
  LED H1-2
  specifications H1-2

Daisy-chaining H1-21
Dimensions A1-1
Dither H1-10
Drive module F1-1
  control problems K1-5
  current limit levels F1-5
  fault conditions K1-3
  functional block diagram F1-2
  HV fuse F1-4
  LEDs F1-6
  specifications F1-3

Encoder
  connections B1-2, G1-4
  couplings G1-5
Encoder backup supply G1-5, H1-15
Encoder following H1-9
ESTOP input H1-13

Fault LED diagnostics K1-2
Following H1-9
Functional block diagram A1-3

Handshaking H1-19
HELP11 ..... H1-25
Hold off distance H1-7

Immediate commands J1-2
Incremental mode H1-4
Initialization H1-10
  normal H1-10
  reduced movement H1-10
Installation
  cables C1-4
  drive F1-4
  heat sinking C1-3
  motors C1-3
  precautions C1-1
  shielding C1-4
  transformer C1-4
Interactive mode J1-1

Limit inputs H1-16
  end-of-travel H1-16
  home H1-17

Linear interpolation H1-7

Module removal C1-2
Motor G1-1
  cabling G1-4
  connections B1-2, G1-4
  control problems K1-5
  dimensions G1-2
  specifications G1-1
  motor/encoder couplings G1-6

Parameter settings H1-25
Position error H1-27
Position monitoring H1-14
Power supply module A1-5, E-1
  AC input E-3
  DC output E-3
  fault conditions K1-3
  fuses E-2
  specifications E-1
Preset moves H1-4
Pulse width modulated signal A1-2

Rack Dimensions C1-1
  HP vs. U A1-2
Reduced movement initialization H1-10
Registration H1-6
Return Material Authorization (RMA) number K1-6
RS232C
  communication parameters B1-3, H1-19
  diagnostics K1-4
  front-panel access H1-20
  motherboard connections H1-20
  permanent or temporary connection ... H1-20

Safety precautions H1-24
Self-tuning H1-23, H1-30
Sequence buffer J1-1
Sequence execution via command input J1-11
Sequence execution via hardware input J1-11
Servo control loop H1-23
Simultaneous start H1-7
Software scaling H1-10
Speed relationship H1-7
Synchronous mode H1-7

Test (functional) B1-4
Transformers A1-5, D-1
  fuse D-3
  single-phase D-1
  three-phase D-3
Tuning
  parameters H1-23, H1-24
  procedure H1-27

User memory H1-10

Velocity monitoring H1-15
Ventilation C1-2

Warranty APP-1

Z channel C1-5
ZDDC H1-21

Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

(217) 352-9330 | sales@artisantg.com | artisantg.com

ARTISAN® TECHNOLOGY GROUP