GFK-0840

GE Fanuc Manual Series 90-30

Power Mate APM Standard Mode

GE Fanuc Automation

PowerMotion ™ Products

Power Mate APM
for Series 90 ™ -30 PLC
Standard Mode

User's Manual

GFK-0840D

August 1996

Warnings, Cautions, and Notes as Used in this Publication

Warning

Warning notices are used in this publication to emphasize that hazardous voltages, currents, temperatures, or other conditions that could cause personal injury exist in this equipment or may be associated with its use.

In situations where inattention could cause either personal injury or damage to equipment, a Warning notice is used.

Caution

Caution notices are used where equipment might be damaged if care is not taken.

Note

Notes merely call attention to information that is especially significant to understanding and operating the equipment.

This document is based on information available at the time of its publication. While efforts have been made to be accurate, the information contained herein does not purport to cover all details or variations in hardware or software, nor to provide for every possible contingency in connection with installation, operation, or maintenance. Features may be described herein which are not present in all hardware and software systems. GE Fanuc Automation assumes no obligation of notice to holders of this document with respect to changes subsequently made.

GE Fanuc Automation makes no representation or warranty, expressed, implied, or statutory with respect to, and assumes no responsibility for the accuracy, completeness, sufficiency, or usefulness of the information contained herein. No warranties of merchantability or fitness for purpose shall apply.

The following are trademarks of GE Fanuc Automation North America, Inc.

Alarm Master	GEnet	PowerMotion	Series One
CIMPLICITY	Genius	ProLoop	Series Six
CIMPLICITY PowerTRAC	Genius PowerTRAC	PROMACRO	Series Three
CIMPLICITY 90–ADS	Helpmate	Series Five	VuMaster
CIMSTAR	Logicmaster	Series 90	Workmaster
Field Control	Modelmaster		

Preface

Content of This Manual

Chapter 1. **Introduction:** This chapter provides a brief overview of the software and hardware used to setup and operate a servo system.

Chapter 2. **Installing the Power Mate APM:** This chapter describes all the user interfaces of the Power mate APM and how to install the module on the Series 90-30 baseplate.

Chapter 3. **Configuring the Power Mate APM:** This chapter explains how to configure the Power Mate APM using the Logicmaster configuration package.

Chapter 4. **Automatic Data Transfers:** This chapter describes the %I, %AI, %Q, and %AQ data that is transferred between the PLC CPU and the Power Mate APM each sweep.

Chapter 5. **Servo System Startup Procedures:** This chapter explains the procedures for properly starting up a servo system.

Chapter 6. **Power Mate APM Motion Control:** This chapter provides practical examples illustrating how to program different types of moves, jumps, etc.

Appendix A. **Error Codes**

Appendix B. **Parameter Download Using COMM_REQ**

Appendix C. **Specifications**

Appendix D. **Wiring to SS-90 Drives**

Appendix E. **Ordering Information**

Appendix F. **LDT Interface**

Appendix G. **Serial Encoders**

Related Publications

GFK–0664 *Series 90-30 Axis Positioning Module (APM) Programmer's Manual*

GFK–0781 Power Mate APM for Series 90-30 PLC *Follower Mode User's Manual*

We Welcome Your Comments and Suggestions

At GE Fanuc automation, we strive to produce quality technical documentation. After you have used this manual, please take a few moments to complete and return the Reader's Comment Card located on the next page.

Contents

Contents

Chapter 1 Introduction

The Power Mate APM is a high-performance, easy-to-use, 1- or 2-axis motion control module which is highly integrated with the logic solving and communications functions of the Series 90-30 PLC.

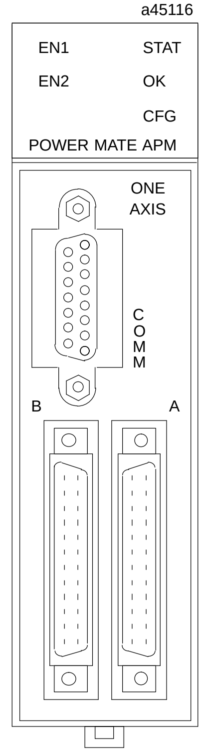
The Power Mate APM operates in two primary control loop configurations:

- Standard Mode
- Follower Mode

This manual describes the Power Mate APM in the *Standard* control loop mode. For information on the operation of the Power Mate APM in Follower control loop mode, refer to GFK-0781, *Power Mate APM for Series 90-30 PLC - Follower Mode User's Manual*.

Features of the Power Mate APM

High Performance



- Fast Servo Loop Update (1 ms for IC693APU301, 2 ms for IC693APU302).
- Block Processing time under 5 ms.
- Velocity Feed forward and Position Error Integrator to enhance tracking accuracy.
- High resolution of programming units.
  Position: -8,388,608...+8,388,607 User Units
  Velocity: 1 ... 8,388,607 User Units / sec
  Acceleration: 1 .. 134,217,727 User Units / sec / sec

Easy to Use

- Simple and powerful Motion Program instruction set.
- Simple 1- or 2-axis motion programs with synchronized block start.
- Program support for a short motion program which can be created in Logicmaster 90 configuration software.
- Non-volatile storage for 10 programs and 40 subroutines.
- Motion Programmer Status Screen for servo monitoring.

- User scaling of programming units (User Units).
- Generic programming using command parameters as operands for Acceleration, Velocity, Move, and Dwell Commands.
- Configured with Logicmaster 90 software.
- Automatic Data Transfer between PLC tables and Power Mate APM without user programming.
- Ease of I/O connection with factory cables and standard terminal blocks as well as a serial port for connecting programming devices.

Versatile I/O

- ±10 volt Velocity Command analog output.
- 12-bit plus sign analog input.
- Home and overtravel switch inputs.
- Position Capture Strobe Input.
- User-defined control inputs (8) and outputs (4).
- A Quad B Encoder Feedback (up to 175 kHz per channel).
- Control of Fanuc digital servos using Power Mate J Digital Servo Interface module.

Firmware Compatibility Tables

1 Axis Power Mate APM						
Released Firmware Version	**Catalog Number**	**Control Loop Capability**	**Power Mate APM Manual**	**PLC Programmer Version**	**Motion Programmer Version**	**Motion Programmer Manual**
1.00, 1.10	IC693APU301A-D	Standard	GFK-0707A	3.50 or later	1.01 or later	GFK-0664
2.02	IC693APU301E-H	Standard Follower	GFK-0840 GFK-0781	4.01 or later *	1.50 or later	GFK-0664A
2.11	IC693APU301J	Standard Follower	GFK-0840 GFK-0781	6.01 or later	1.50 or later	GFK-0664A
2.50	IC693APU301K	Standard Follower	GFK-0840 GFK-0781	6.01 or later	1.50 or later	GFK-0664A

* *The 1 axis Power Mate APM firmware release 2.02 requires PLC Programmer version 4.50 or later to allow configuration in Follower mode.*

2 Axis Power Mate APM						
Released Firmware Version	**Catalog Number**	**Control Loop Capability**	**Power Mate APM Manual**	**PLC Programmer Version**	**Motion Programmer Version**	**Motion Programmer Manual**
1.50	IC693APU302A-B	Follower	GFK-0781	3.50 or later	1.50 or later	GFK-0664A
2.01, 2.02	IC693APU302GG	Standard Follower	GFK-0840 GFK-0781	4.01 or later	1.50 or later	GFK-0664A
2.10, 2.11	IC693APU302HJ	Standard Follower	GFK-0840 GFK-0781	6.01 or later	1.50 or later	GFK-0664A
2.50	IC693APU302K	Standard Follower	GFK-0840 GFK-0781	6.01 or later	1.50 or later	GFK-0664A

Overview of Power Mate APM Operations

The Power Mate APM is an intelligent, fully programmable, motion control option module for the Series 90-30 Programmable Logic Controller (PLC). The Power Mate APM allows a PLC user to combine high performance motion control with PLC logic solving functions in one integrated system. The figure below illustrates the hardware and software needed to set up and operate a servo system. This section will discuss briefly each piece of the system to provide an overall picture of system operation.

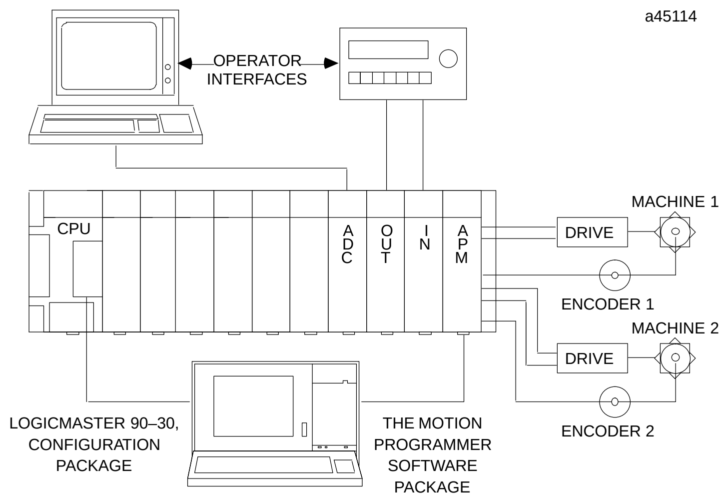


Figure 1-1. Hardware and Software Used to Configure, Program, and Operate an Power Mate APM Servo System

The system shown above can be divided into 4 categories:

1. The Power Mate APM and the Series 90-30 PLC
2. The Power Mate APM and the Servo System
3. Motion Programming Packages
4. Operator Interfaces

It is helpful to become familiar with the following Power Mate APM terms.

Commanded Velocity: the velocity generated by the Power Mate APM's internal path generator.

Commanded Position: the instantaneous position generated by the Power Mate APM's internal path generator.

Actual Velocity: the velocity of the axis indicated by the feedback.

Actual Position: the position of the axis indicated by the feedback.

Position Error: the difference between the Commanded Position and the Actual Position.

Position Loop Time Constant: a configurable value determining the response time of the closed Position Loop.

The Power Mate APM and the Series 90-30 PLC

The Power Mate APM and Series 90-30 PLC operate together as one integrated motion control package. The Power Mate APM communicates with the PLC through the backplane interface. Every PLC sweep, data such as *Commanded Velocity* and *Actual Position* within the Power Mate APM is transferred to the PLC in %I and %AI data. Also every sweep, %Q and %AQ data is transferred from the PLC to the Power Mate APM. The %Q and %AQ data is used to control the Power Mate APM. %Q bits perform functions such as initiating motion, aborting motion, and clearing strobe flags. %AQ commands perform functions such as initializing position and loading parameter registers.

In addition to %I, %AI, %Q, and %AQ reference data, COMM_REQ function blocks may be used to send parameters to the Power Mate APM. Details about parameter block downloading can be found in Chapter 6, Power Mate APM Motion Control.

The Power Mate APM is easily configured using the Logicmaster 90-30 Configuration software . The Power Mate APM is assigned to a particular slot and rack like any other PLC module. In addition, other types of configuration data must be entered such as:

- I/O addresses where the CPU to Power Mate APM transfers take place
- Serial Port Setup for connecting the Motion Programmer
- PM-Power Mate APM Setup Data
- An optional 20 line motion program (Program Zero)

All of the Power Mate APM configuration data is discussed in detail in Chapter 3, Configuring the Power Mate APM.

The Power Mate APM and the Servo System

The Power Mate APM communicates through the A and B connectors to the servo drive and position feedback device. The Power Mate APM controls an axis by sending an analog voltage to a servo. It interprets what the servo, or axis, does by reading a position feedback device.

Position Error is one of the controlling factors in closing the position loop. The Power Mate APM produces a velocity command output proportional to the position error.

The Power Mate APM also has Home Switch inputs, Overtravel Limit Switch inputs, and other control inputs and outputs.

The Power Mate APM and Digital Servos

Firmware release 2.50 allows the Power Mate APM to operate with a Digital Servo Interface module (DSI) to control GE Fanuc a or b series servo amplifiers and motors. The combination of the Power Mate APM controller module and Digital Servo Interface module is referred to as the Power Mate J system. Refer to the Power Mate J User's manual, GFK-1256 for additional information.

Motion Programming Packages

Motion programs are normally created using the Motion Programmer software package, but a single short program (Program Zero) can be created within the Logicmaster 90-30 configuration software.

Motion Programmer. The Power Mate APM has a configurable SNP port to communicate with the Motion Programmer. The Motion Programmer operates much like the Logicmaster 90 software package. It provides the capability of writing English-language motion programs, storing the programs on disks, and downloading the programs to the Power Mate APM as desired. A Status Screen supports servo system monitoring. An overview of Motion Programmer functions is provided at the end of this chapter. For detailed information on Motion Programmer, refer to GFK-0664A, *the Series 90 Axis Positioning Module Programmer's Manual*.

Program Zero. The Logicmaster Configuration package contains an editor for creating a short (20 commands maximum) program defined as Program Zero. The commands are entered in an English-language format similar to the Motion Programmer. No ladder programming is needed. Program Zero is discussed in detail in Chapter 3, Configuring the Power Mate APM.

Operator Interfaces

Operator interfaces provide a way for the operator to control and monitor the servo system through a control panel or CRT display. These interfaces communicate with the PLC through discrete I/O modules or an intelligent Series 90 PLC module such as CIMPLICITY 90 - ADS or a Programmable Coprocessor Module (PCM).

Operator data is automatically transferred between the PLC and the Power Mate APM through %I, %AI, %Q, and %AQ references which are specified when the module is configured. This automatic transfer of data provides a flexible and simple interface to a variety of operator interfaces in addition to the Motion Programmer.

Overview of Motion Programmer

This section summarizes programming using the Motion Programmer. For a comprehensive description of Power Mate APM programming, refer to GFK-0664A, *The Series 90 Axis Positioning Module Programmer's Manual*.

The Motion Programmer is a powerful, English-language programmer for the Power Mate APM. It can be executed as a stand-alone package or from the Logicmaster Startup Menu. Programs and Subroutines are developed by selecting commands using function keys and then entering values into Operand fields to complete the instruction.

Features of the Motion Programmer

- Create and Edit Programs and Subroutines
- Setup the Programmer Ports
- Load and Store Program files between the Motion Programmer and Power Mate APM
- Print Motion Programs, Subroutines, and Setup information

Features of the Program Editor

- Insert Program Lines
- Edit Program Lines
- Delete Program Line
- Check Program Syntax
- Renumber Block Numbers
- GOTO Function

The figure below illustrates the editor screen with sample instructions.

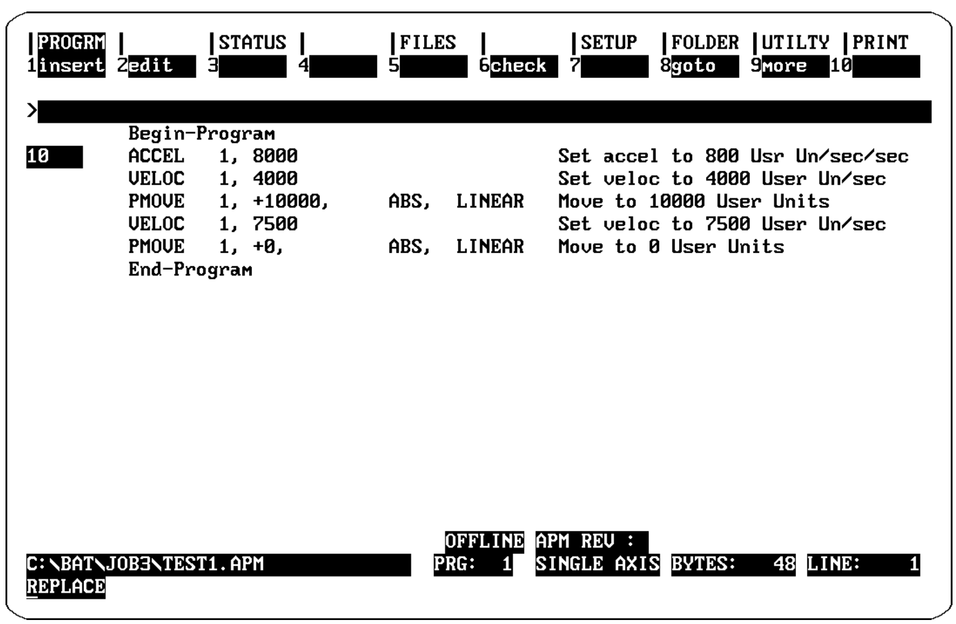


Figure 1-2. Example Motion Programmer Editor Screen

Motion Programmer Instruction Format

A program or subroutine instruction consists of a command and associated data describing the command. The command and data are entered into areas on the program editor screen called fields. These fields are described below. From left to right the fields are:

Block Number Field. This field is *optional*. The Block Number field permits the entry of a number to designate the beginning of a block of program instructions. It is used as a place for Jump commands to go to and for synchronizing 2-axis programs.

Synchronizing Block Indicator Field. This field appears only when editing multi-axis programs and is optional. An "X" will appear to indicate a block number is a sync block.

Command Name Field. This field is *required*. It is an English language designation for the command. The commands are selected from function keys at the top of the editor screen.

Operand Fields. Any operand field displayed is *required*. When a Command name is entered, the required Operand fields will be displayed with default values. The number of Operand fields displayed varies from one to four depending on the Command name. Operand field 1 is reserved for the axis number. The content of the other 3 fields vary depending on the command.

Comments Field. This field is *optional*. It allows approximately 30 characters of comment for each program instruction. A separate 80 character comment can be added on a separate line.

Table 1-1. The Motion Programmer Command List

Command Name Field	Operand 1	Operand 2	Operand 3	Operand 4
Acceleration (ACCEL)	Axis # (1, 2)	USER UNITS/sec/sec (1 - 134,217,727) or Parameter Number (0 - 255)	n/a	n/a
Call Subroutine (CALL)	n/a	Subroutine Number (1-40)	n/a	n/a
Continuous Move (CMOVE)	Axis # (1, 2)	USER UNITS (-8,388,608...+8,388,607)* or Parameter Number (0 - 255)	Positioning Mode (ABS, INCR)	Acceleration Mode (LINEAR, SCURVE)
Jump (JUMP)	n/a	CTL # (CTL01-CTL12, Unconditional)	Block Number (1-65,535)	n/a
Dwell (DWELL)	Axis # (1, 2)	Milliseconds (0-60,000) or Parameter Number (0 - 255)	n/a	n/a
Load Parameter (LOAD)	n/a	Parameter Number (0 - 255)**	Parameter Value (32-bit maximum)	n/a
Null (NULL)	n/a	n/a	n/a	n/a
Positioning Move (PMOVE)	Axis # (1, 2)	USER UNITS (–8,388,608...+8,388,607)* or Parameter Number (0 - 255)	Positioning Mode (ABS, INCR)	Acceleration Mode (LINEAR, SCURVE)
Velocity (VELOC)	Axis # (1, 2)	USER UNITS/sec (1 - 8,388,607) or Parameter Number (0 - 255)	n/a	n/a
Wait (WAIT)	Axis # (1, 2)	CTL # (CTL01 - CTL12)	n/a	n/a

* Absolute Moves (ABS) must specify a position within the End of Travel **AND** Count Limits or a stop error will occur when the move command is executed.

** Parameters 246 - 255 are reserved special purpose parameters. Load Parameter can load data into these parameters, but the Power Mate APM may overwrite its own data into these parameters.

Parameter Number	Special Purpose Function
246 - 251	Reserved for future use
252	Stores Axis 2 *Commanded Position* (user units) when Follower Enable Trigger occurs (2 Axis Power Mate APM only)
253	Stores Axis 1 *Commanded Position* (user units) when Follower Enable Trigger occurs
254	Stores Axis 2 *Strobe Position* value (user units), 2 Axis Power Mate APM only
255	Stores Axis 1 *Strobe Position* value (user units)

Chapter 2 — Installing the Power Mate APM

This chapter describes the 1- and 2-axis models of the Power Mate APM and how to install them on the Series 90-30 PLC baseplate. The chapter is divided into the following sections.

Section 1: Description of the Power Mate APM

This section describes the user interfaces of the 1- and 2-Axis Power Mate APM in standard mode.

LED Indicators

There are 5 LEDs which provide status indication for the Power Mate APM. These LEDs are described below.

Status. Normally ON. FLASHES to provide an indication of operational errors. Flashes *slow* (4 times/sec) for Status-Only errors. Flashes *fast* (8 times/sec) for errors which cause the servo to stop.

OK. The Power Mate APM OK LED indicates the current status of the Power Mate APM board.

ON: When the LED is steady ON, the Power Mate APM is functioning properly. Normally, this LED should always be ON.

OFF: When the LED is OFF, the Power Mate APM is not functioning. This is the result of a hardware or software malfunction.

CFG. It is ON when a valid module configuration has been received from the PLC. Flashes *slow* (4 times/sec) during the Motion Program Store function. Flashes *fast* (8 times/sec) during the Write User RAM to EEPROM operation.

EN1. When this LED is ON, the servo drive for Axis 1 is enabled.

EN2. When this LED is ON, the servo drive for Axis 2 is enabled. On a 1-axis Power Mate APM, this LED is ON only when a *Force D/A Immediate* command is used for the analog output on Connector B.

Serial COMM Connector

The Power Mate APM Front Panel contains a single 15-pin, female, D-connector for serial communications. It is used to connect the computer running the Motion Programmer software to the Power Mate APM. The Power Mate APM port uses the GE Fanuc proprietary SNP protocol and is RS-485 compatible. The baud rate is selectable from 300 to 19,200 baud.

The connection between the programming computer and the Power Mate APM is typically made from the RS-232 port of the computer through an RS-232 to RS-485 / RS-422 converter to the Serial Communications Connector. A Miniconverter Kit (IC690ACC901) which includes a converter and 6 ft cable is available for this purpose.

The port is configured using the Logicmaster Configuration Package or the 90-30 PLC Hand Held Programmer.

The pin definitions for the Power Mate APM serial port are listed below.

Table 2-1. Pin Definitions of the Serial COMM Connector

Pin	Signal	Description	Pin	Signal	Description
1	Shield	Cable Shield	9	RT	120 Ohm Termination for RXD (A)
2	DCD (A)	Carrier Detect	10	RD (A)	Receive Data
3	DCD (B)	Carrier Detect	11	RD (B)	Receive Data
4	ATCH	HHP Attach Input	12	SD (A)	Transmit Data
5	+5 V	+5 V HHP / 422-232 Converter Power	13	SD (B)	Transmit Data
6	RTS (A)	Ready to Send	14	RTS (B)	Ready to Send
7	0 V	0 V Signal Common	15	CTS (A)	Clear to Send
8	CTS (B)	Clear to Send			

Multidrop Connection

Power Mate APM modules can be connected in multidrop fashion. A sample setup is shown in the figure below. One cable is necessary for each Power Mate APM in the system.

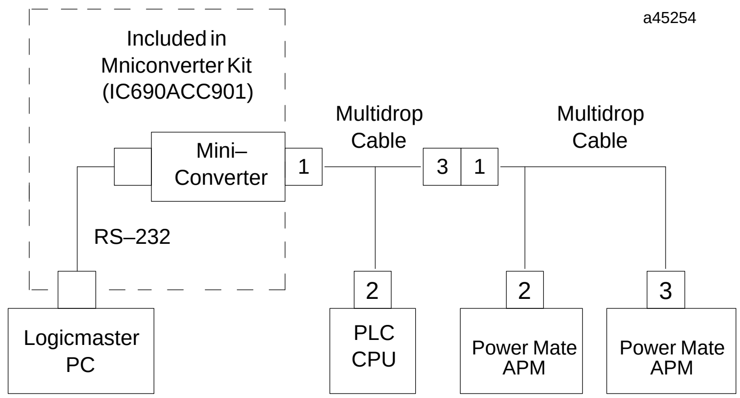


Figure 2-1. Connecting Power Mate APM Modules in a Multidrop Configuration

The multidrop cable should be made according to the following diagram.

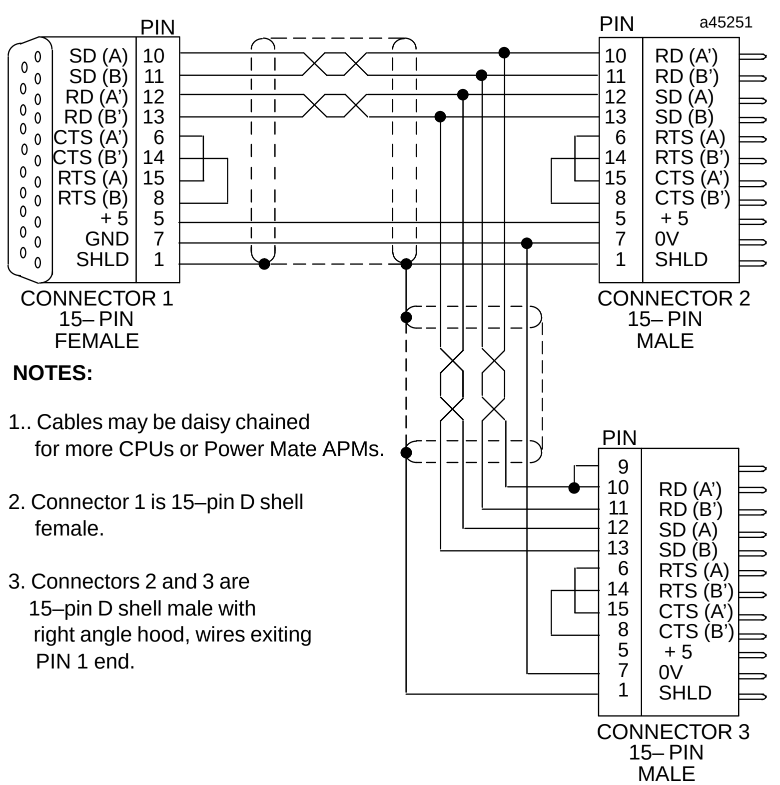


Figure 2-2. Multidrop Cable for the Power Mate APM

I/O Connectors

The Power Mate APM Front Panel contains two 24-pin, male high-density connectors for servo connections. Connector A contains connections for Axis 1. Connector B contains connections for Axis 2 and general purpose connections including the Analog Input and the Control Outputs. The tables briefly describe the function of the I/O connections.

I/O Cable and Terminal Block

High-density connectors are used on the Power Mate APM to permit a large number of I/O connections within the physical size limitations of the Power Mate APM. To facilitate wiring to the drive and machine, each high-density connector is typically connected by a short cable to a terminal block. Refer to Figure 2-3 below and Tables 2-2 through 2-5.

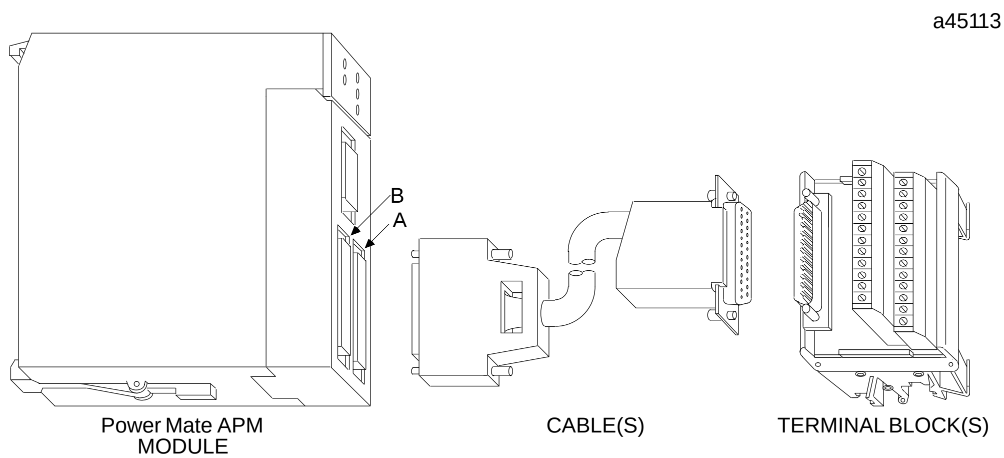


Figure 2-3. I/O Cable and Terminal Block

I/O Cable Connections for the 1-Axis Power Mate APM (IC693APU301)

Tables 2-2 and 2-3 define the cable connections for the 1-Axis Power Mate APM

Table 2-2. Cable Connections to Faceplate I/O Connector A (1-Axis Power Mate APM)

I/O Module Connector A Pin Number	Terminal Block Terminal Number	Description
A1	12	0 V
B1	24	Reserved
A2	11	Reserved
B2	23	CTL02 (+) Input
A3	10	CTL02 (–) Input
B3	22	Strobe 1 (+) / CTL01 (+) Input
A4	9	Strobe 1 (–) / CTL01 (–) Input
B4	15	Common for CTL03, 05, 06 Inputs
A5	2	Overtravel 1 (– direction) / CTL06 Input
B5	14	Overtravel 1 (+ direction) / CTL05 Input
A6	1	Home 1 / CTL03 Input
B6	16	Enable Relay 1 (–) Output
A7	3	Enable Relay 1 (+) Output
B7	17	Analog Output 1 (Velocity Command) Common*
A8	4	Analog Output 1 (Velocity Command)
B8	21	0 V
A9	8	+5 V Encoder Power
B9	20	Encoder Channel Z1+ / Linear Feedback (+) input
A10	7	Encoder Channel Z1– / Linear Feedback (–) input
B10	19	Encoder Channel B1+
A11	6	Encoder Channel B1–
B11	18	Encoder Channel A1+
A12	5	Encoder Channel A1–
B12	13	Cable Shield
	25	no connection

* Fused Output - an external source applied to this output could open the fuse.

Table 2-3. Cable Connections to Faceplate I/O Connector B (1-Axis Power Mate APM)

I/O Module Connector B Pin Number	Terminal Block Terminal Number	Description
A1	12	0 V
B1	24	Analog (+) Input
A2	11	Analog (–) Input
B2	23	CTL12 Output
A3	10	CTL11 Output
B3	22	CTL10 Output
A4	9	CTL09 Output
B4	15	Common for CTL04, 07, 08 Inputs
A5	2	CTL08 Input
B5	14	CTL07 Input
A6	1	CTL04 Input
B6	16	Enable Relay 2 (–) Output
A7	3	Enable Relay 2 (+) Output
B7	17	Analog Output 2 Common*
A8	4	Analog Output 2
B8	21	0 V
A9	8	+5 V Output
B9	20	Reserved
A10	7	Reserved
B10	19	Reserved
A11	6	Reserved
B11	18	Reserved
A12	5	Reserved
B12	13	Cable Shield
	25	no connection

* Fused Output - an external source applied to this output could open the fuse.

I/O Cable Connections for the 2-Axis Power Mate APM (IC693APU302)

Tables 2-4 and 2-5 define the cable connections for the 2-Axis Power Mate APM

Table 2-4. Cable Connections to Faceplate I/O Connector A (2-Axis Power Mate APM)

I/O Module Connector A Pin Number	Terminal Block Terminal Number	Description
A1	12	0 V
B1	24	Reserved
A2	11	Reserved
B2	23	Strobe 2 (+) / CTL02 (+) Input
A3	10	Strobe 2 (–) / CTL02 (–) Input
B3	22	Strobe 1 (+) / CTL01 (+) Input
A4	9	Strobe 1 (–) / CTL01 (–) Input
B4	15	Common for CTL03, 05, 06 Inputs
A5	2	Overtravel 1 (– direction) / CTL06 Input
B5	14	Overtravel 1 (+ direction) / CTL05 Input
A6	1	Home 1 / CTL03 Input
B6	16	Enable Relay 1 (–) Output
A7	3	Enable Relay 1 (+) Output
B7	17	Analog Output 1 (Velocity Command) Common*
A8	4	Analog Output 1 (Velocity Command)
B8	21	0 V
A9	8	+5 V Encoder Power
B9	20	Encoder Channel Z1+ / Linear Feedback 1 (+) input
A10	7	Encoder Channel Z1– / Linear Feedback 1 (–) input
B10	19	Encoder Channel B1+
A11	6	Encoder Channel B1–
B11	18	Encoder Channel A1+
A12	5	Encoder Channel A1–
B12	13	Cable Shield
	25	no connection

* Fused Output - an external source applied to this output could open the fuse.

Table 2-5. Cable Connections to Faceplate I/O Connector B (2-Axis Power Mate APM)

I/O Module Connector B Pin Number	Terminal Block Terminal Number	Description
A1	12	0 V
B1	24	Analog (+) Input
A2	11	Analog (–) Input
B2	23	CTL12 Output
A3	10	CTL11 Output
B3	22	CTL10 Output
A4	9	CTL09 Output
B4	15	Common for CTL04, 07, 08 Inputs
A5	2	Overtravel 2 (– direction) / CTL08 Input
B5	14	Overtravel 2 (+ direction) / CTL07 Input
A6	1	Home 2 / CTL04 Input
B6	16	Enable Relay 2 (–) Output
A7	3	Enable Relay 2 (+) Output
B7	17	Analog Output 2 (Velocity Command) Common*
A8	4	Analog Output 2 (Velocity Command)
B8	21	0 V
A9	8	+5 V Output
B9	20	Encoder Channel Z2 + / Linear Feedback 2 (+) input
A10	7	Encoder Channel Z2 – / Linear Feedback 2 (–) input
B10	19	Encoder Channel B2 +
A11	6	Encoder Channel B2 –
B11	18	Encoder Channel A2 +
A12	5	Encoder Channel A2 –
B12	13	Cable Shield
	25	no connection

* Fused Output - an external source applied to this output could open the fuse.

Functional Connection Diagrams

The Figures below illustrates how the 1- and 2-Axis Power Mate APM I/O are connected to a drive and a machine in a typical application. Shielded cable should be used as indicated.

Functional Connection Diagrams for the 1-Axis Power Mate APM

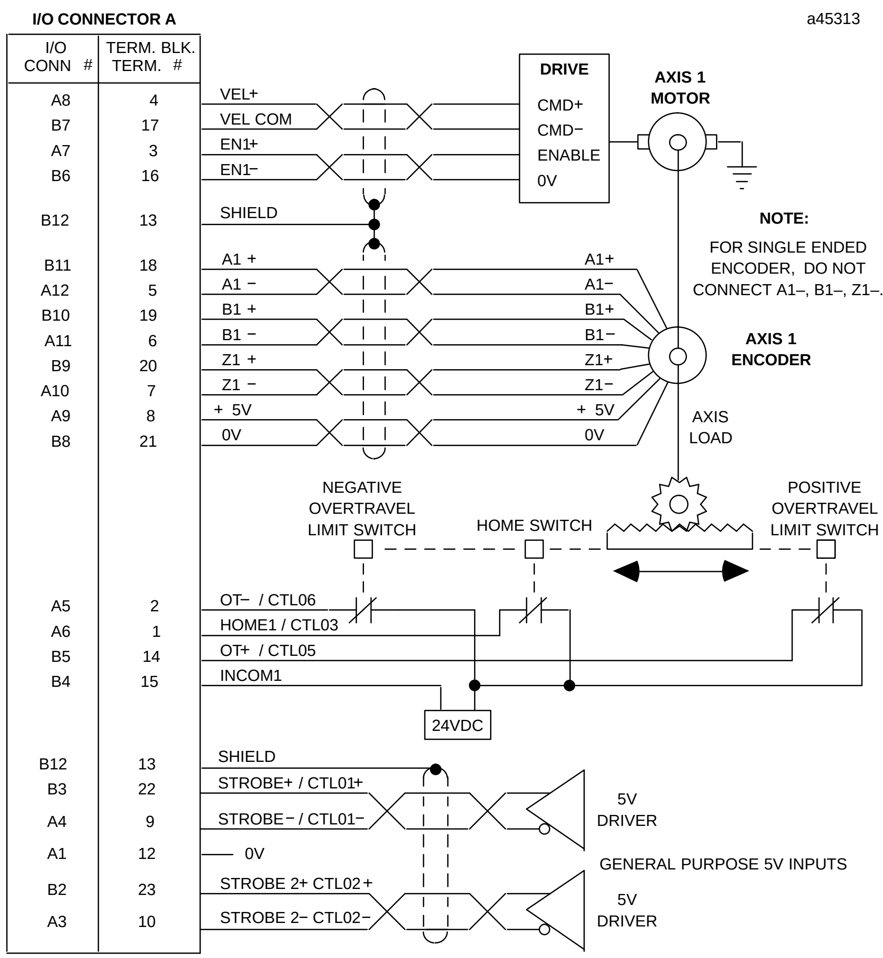


Figure 2-4. Functional Diagram for 1-Axis Power Mate APM I/O Connector A

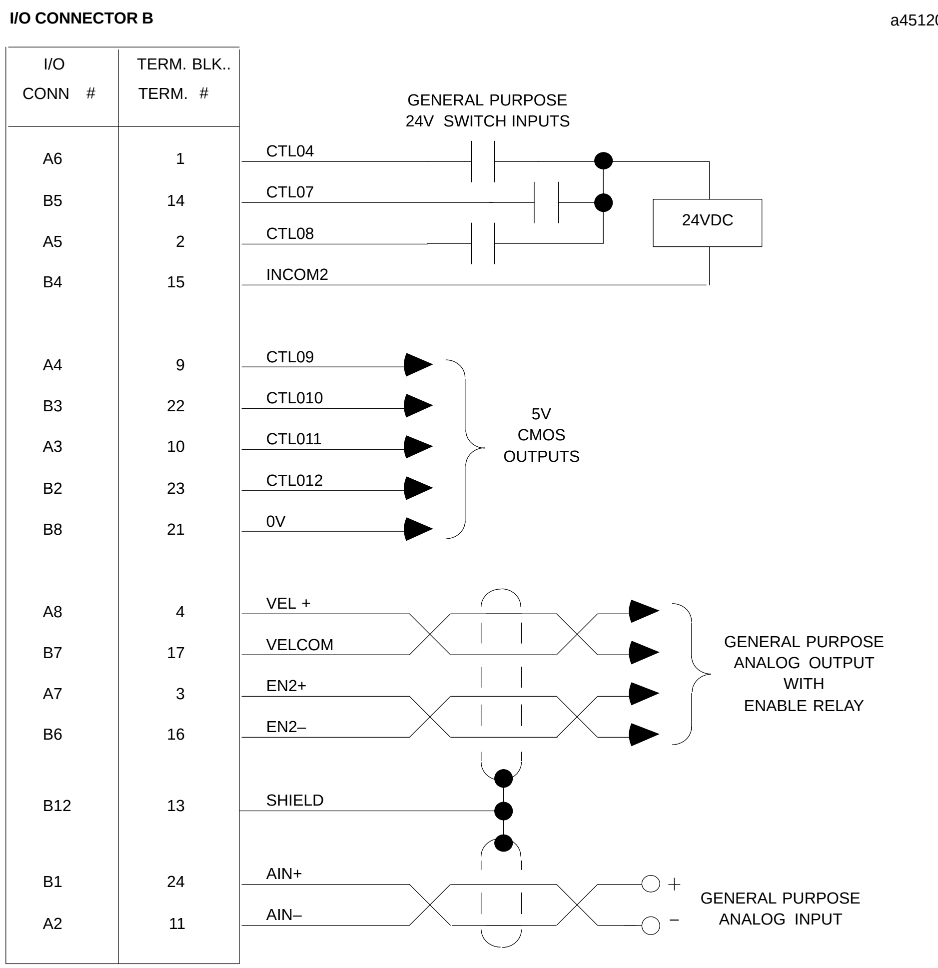


Figure 2-5. Functional Diagram for 1-Axis Power Mate APM I/O Connector B

Functional Connection Diagrams for the 2-Axis Power Mate APM

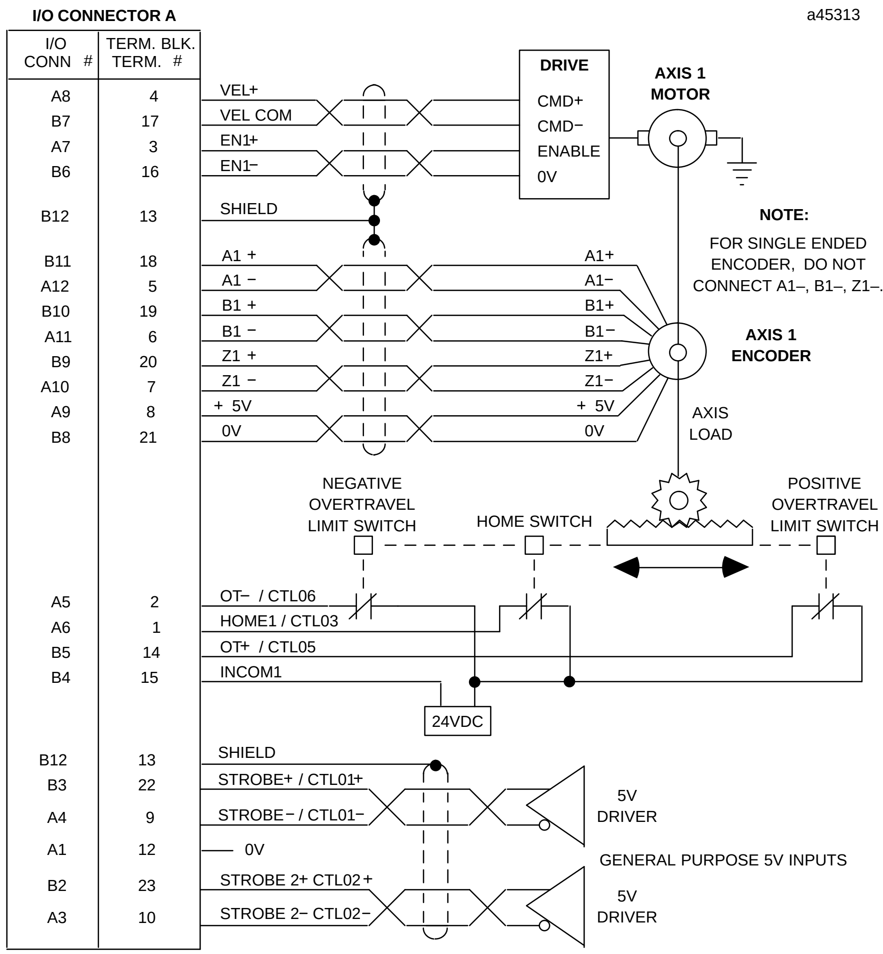


Figure 2-6. Functional Diagram for 2-Axis Power Mate APM I/O Connector A

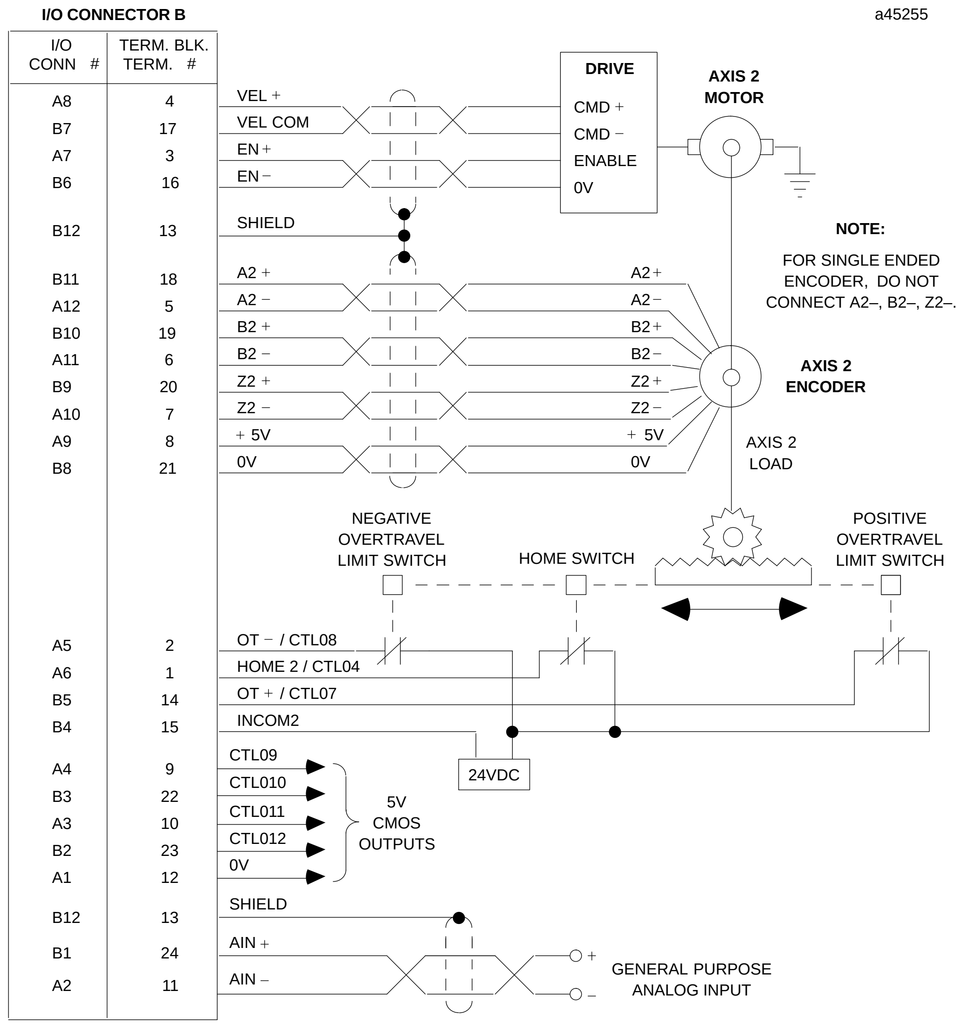


Figure 2-7. Functional Diagram for 2-Axis Power Mate APM I/O Connector B

Linear Feedback Transducer

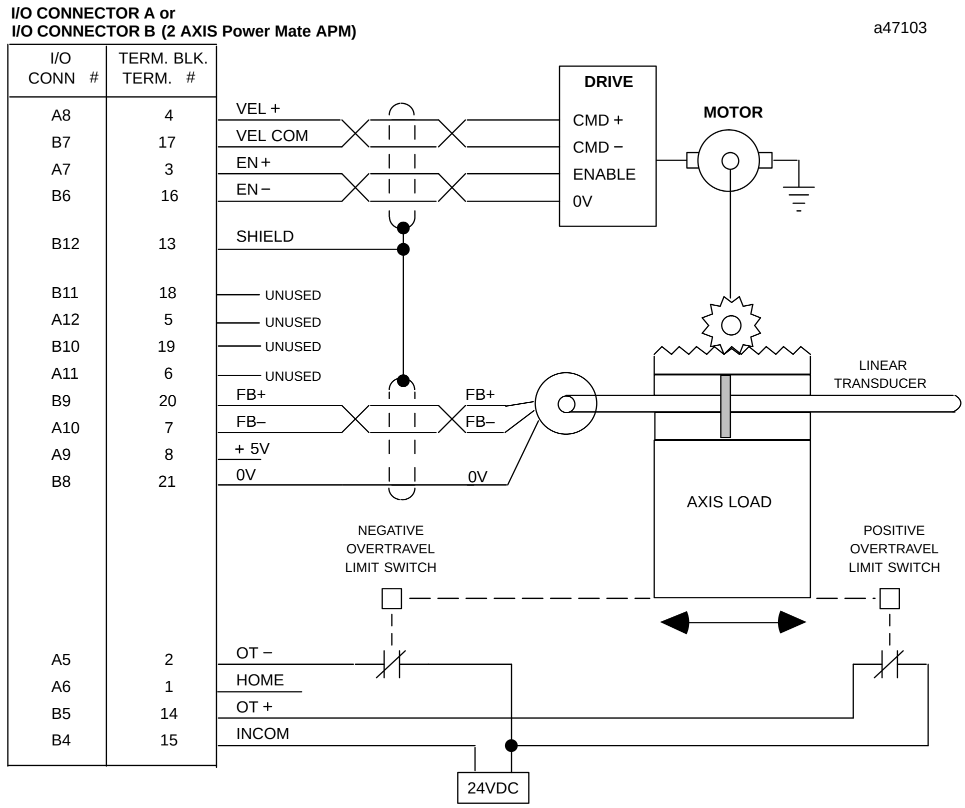


Figure 2-8. Connections for Linear Transducer

When a linear transducer is used (configured Feedback Type = Linear), the transducer differential 5V output must be connected as shown above. Linear feedback does not require a Home switch, so the Home switch input can be left unconnected or used as a general purpose 24V input. Strobe and analog I/O connections not shown above are unchanged from Figures 2-4 through 2-7.

Section 2: Installing the Power Mate APM

The Power Mate APM can operate in *any* Series 90-30 CPU or expansion baseplate (Series 90-30 release 3 or later). For limitations on the maximum number of Power Mate APM modules you can install per baseplate and system, refer to Appendix C, Module Specifications.

The configuration files created by Logicmaster 90 configuration software must match the physical configuration of the modules.

To install the Power Mate APM on the baseplate, follow these steps:

1. Use the Logicmaster 90 software or the Hand Held Programmer to stop the PLC. This will prevent the local application program, if any, from initiating any command that may affect the operation of the module.
2. Power down the Series 90-30 PLC system.
3. Align the module with the desired base slot and connector. Tilt the module upward so that the top rear hook of the module engages the slot on the baseplate.
4. Swing the module down until the connectors mate and the lock-lever on the bottom of the module snaps into place engaging the baseplate notch.
5. Refer to Figures 2-4 through 2-7 and Tables 2-2 through 2-5 for wiring requirements.
6. Power up the PLC rack. The Status LED of the Power Mate APM will turn ON when the Power Mate APM has passed its power-up diagnostics.
7. Repeat this procedure for each Power Mate APM.
8. Configure the Power Mate APM module(s) as explained in Chapter 3.

Chapter 3 Configuring the Power Mate APM

The Power Mate APM is configured using the Logicmaster 90-30 configuration software. Configuring the Power Mate APM is a two-part procedure consisting of:

- I/O Rack Configuration
- Module Configuration

I/O Rack Configuration

The Power Mate APM is configured using Logicmaster 90-30 software in the same way as other Series 90-30 modules. The software is used to define the type and location of all modules present in the PLC racks. This is done by completing setup screens which represent the modules in a baseplate and saving the information to a configuration file, which is then downloaded to the CPU.

Once a baseplate and slot location for the Power Mate APM is defined, you can continue to the second part of Power Mate APM configuration, Module Configuration.

Module Configuration

This section is divided into three parts:

- Setting the Configuration Parameters
- Essential Configuration Parameters
- Important Configuration Considerations

Setting the Configuration Parameters

As with I/O Rack Configuration, Module Configuration is done by completing screens in the Logicmaster 90-30 configuration software. The Hand Held Programmer can only configure the Module and Serial Port Configuration Data.

Power Mate APM module configuration data consists of 4 types:

- Module Configuration Data
- Programmer Port Configuration Data
- Axis Configuration Data
- Program Zero

Module Configuration Data

During each CPU sweep certain data is automatically transferred both ways between the Power Mate APM and the CPU. Power Mate APM to CPU Interface data references the starting locations for the automatic transfers. The configuration parameters in Module Configuration Data are described in Table 3-1.

Table 3-1. Module Configuration Data

Configuration Parameter	Description	Values	Defaults	Units
Ref Adr	Start address for %I ref type (32 bits)	CPU Dependent	%I00001 or next higher reference	n/a
Ref Adr	Start address for %Q ref type (32 bits)	CPU Dependent	%Q00001 or next higher reference	n/a
Ref Adr	Start address for %AI ref type (15 words for 1-axis, 28 words for 2-axis)	CPU Dependent	%AI00001 or next higher reference	n/a
Ref Adr	Start address for %AQ ref type (6 words)	CPU Dependent	%AQ00001 or next higher reference	n/a
%AI Pos Err	On Single Axis Power Mate APMs this parameter adds Position Error to %AI Data	DISABLED/ ENABLED	DISABLED	n/a
Fdback Type	Feedback Type[1]	ENCODER/ LINEAR RESOLVR [1] CUSTOM 1 [2] CUSTOM 2 [2] DIGITAL [1]	ENCODER	n/a
Ctl Loop	Control Loop Type	STANDARD FOLLOWER [3] CCL1 [2] CCL2 [2]	STANDARD	n/a
Servo Cmd	Servo Interface Type	ANALOG DIGITAL [1]	ANALOG	n/a
Motor1 Type Motor2 Type [4]	GE Fanuc Motor Type [1]	0 . . . 127	0 (no motor)	n/a
Motor1 Dir Motor2 Dir [4]	Motor direction for positive velocity command	POS NEG	POS	n/a

1 Reserved for future use; not implemented at this time.

2 For special-purpose applications only.

3 For information about the Follower Control Loop Type, refer to GFK-0781 *Series 90-30 Power Mate APM - Follower Mode User's Manual*.

4 Two axis Power Mate APM, IC693APU302H, or later.

NOTE: Refer to Appendix F for a discussion of LINEAR feedback.

Fdback Type. ENCODER selects A quad B (x4) incremental encoder input mode. LINEAR selects linear transducer (absolute feedback) input mode. DIGITAL selects GE Fanuc Digital AC servo encoder input mode. If DIGITAL is selected, the SERVO CMD configuration parameter must also be set to DIGITAL. CUSTOM1 and CUSTOM2 configure the Power Mate APM inputs for special applications. DIGITAL feedback is only supported in Power Mate APM firmware revisions 2.50 and higher. (Default = ENCODER).

Ctl Loop. STANDARD selects the normal Power Mate APM motion control loop. The STANDARD loop provides a velocity command proportional to position error, with optional Velocity Feedforward and Integrator gain terms. CCL1 and CCL2 (Customer Control Loops) are individually designed for special applications. FOLLOWER selects a control loop that allows ratio tracking of a master input with zero following error. (Default = STANDARD).

Servo Cmd. This parameter defines the type of command output provided to the servo system. ANALOG selects a Ç 10 volt velocity command for standard servo drives. DIGITAL selects a special digital output for GE Fanuc Digital AC servo drives. Digital AC servos are only supported in Power Mate APM firmware versions 2.50 and higher. (Default = ANALOG).

Motor Type. Selects the type of GE Fanuc Digital AC servo motor to be used with the Power Mate APM controller module and Digital Servo Interface module (Power Mate J system). Digital AC servos are only supported in Power Mate APM firmware versions 2.50 and higher. A motor type of 0 for a particular axis disables digital servo control by the Interface module for that axis. (Default = 0).

a Series Servo Motor

Motor Type Code	Motor Model	Motor Specification
13	a0.5 / 3000	0113
61	a1 / 3000	0371
46	a2 / 2000	0372
62	a2 / 3000	0373
15	a3 / 3000	0123
16	a6 / 2000	0127
17	a6 / 3000	0128
18	a12 / 2000	0142
19	a12 / 3000	0143
27	a22 / 1500	0146
20	a22 / 2000	0147
21	a22 / 3000	0148
28	a30 / 1200	0151
22	a 30 / 2000	0152
23	a30 / 3000	0153
30	a40 / 2000	0157
29	a40 / FAN	0158

a L Series Servo Motor

Motor Type Code	Motor Model	Motor Specification
56	aL3 / 3000	0561
57	aL6 / 3000	0562
58	aL9 / 3000	0564
59	aL25 / 3000	0571
60	aL50 / 2000	0572

a C Series Servo Motor

Motor Type Code	Motor Model	Motor Specification
7	aC3 / 2000	0121
8	aC6 / 2000	0126
9	aC12 / 2000	0141
10	aC22 / 1500	0145

a HV Series Servo Motor

Motor Type Code	Motor Model	Motor Specification
3	a12HV / 3000	0176
4	a22HV / 3000	0177
5	a30HV / 3000	0178

a M Series Servo Motor

Motor Type Code	Motor Model	Motor Specification
24	aM3 / 3000	0161
25	aM6 / 3000	0162
26	aM9 / 3000	0163

b Series Servo Motor

Motor Type Code	Motor Model	Motor Specification
35	b 1 / 3000	0031
36	b2 / 3000	0032
33	b3 / 3000	0033
34	b6 / 2000	0034

Motor Dir. A configured motor direction of POS (Positive) defines the positive axis direction as encoder channel A leading channel B (encoder feedback) or motion away from the feedback transducer head (linear feedback). A configured motor direction of NEG (Negative) defines the positive axis direction as encoder channel B leading channel A (encoder feedback) or motion towards the feedback transducer head (linear feedback). In practice, the motor direction configuration allows the user to reverse the motion caused by all commands without reversing motor or encoder wires. For digital servos, POS defines the positive axis direction as counter clockwise (CCW) motor shaft rotation when viewed looking into the front (output shaft end) of the motor.

Serial Communications Port Configuration Data

The Power Mate APM can be programmed using the Motion Programmer software. The computer running the Motion Programmer software connects to the Serial Communications Port (which supports the SNP protocol) on the faceplate of the Power Mate APM.

The Serial Communications Port must be configured properly to communicate with the Motion Programmer. Make sure the configuration parameters for the Motion Programmer and the Serial Communications Port match. The configuration parameters for the Serial Communications Port Configuration Data are described in Table 3-2.

Table 3-2. Serial Communications Port Configuration Data

Configuration Parameter	Description	Values	Defaults	Units
Baud Rate	Baud rate of SNP Port	300, 600, 1200, 2400, 4800, 9600, 19200	19200	n/a
Parity	Parity	ODD, EVEN, NONE	ODD	n/a
Stop Bits	Number of stop bits	1 or 2	1	n/a
Data Bits	Number of data bits	7 or 8	8	n/a
Modem TT	Modem turn-around time	0..2550, in multiples of 10 milliseconds	0	ms
Idle Time	Maximum link idle time	1..60	10	sec
SNP ID	SNP ID	6 characters consisting of A-F and 0-9. 1st char must be A-F	A00001	n/a

Axis Configuration Data

Power Mate APM Axis Configuration Data consists of base values for configuration parameters used by one or more motion programs. The values of these configuration parameters do not usually change, therefore they are not included in the motion program. The configuration parameters are defined and briefly described here.

Table 3-3. Axis Configuration Data

Configuration Parameter	Description	Values	Defaults	Units
User Units	User Units Value	1...65,535	1	n/a
Counts	Feedback Counts	1...65,535	1	n/a
OT Limit Sw	Overtravel Limit Sw En/Dis	ENABLED/DISABLED	ENABLED	n/a
Pos EOT	Positive End of Travel	–8,388,608...+8,388,607	+8,388,607	user units
Neg EOT	Negative End of Travel	–8,388,608...+8,388,607	– 8,388,608	user units
Pos Err Lim*	Position Error Limit	256...60,000	+ 4096	user units
In Pos Zone	In Position Zone	0..2000	10	user units
Pos Loop TC*	Position Loop Time Constant	0, 5...10,000	1000	ms
Vel at 10 V*	Velocity for 10 V Output	400...1,000,000	+4000	user units/sec
Vel FF %	Velocity Feed Forward	0..100	0	%
Intgr TC	Integrator Time Constant	0, 10..10,000	0	ms
Intgr Mode	Integrator Mode	OFF/CONT/IN ZONE	OFF	n/a
Rev Comp	Reversal Compensation	0...255	0	user units
DisDly	Drive Disable Delay	0...65,535	100	ms
Jog Vel	Jog Velocity	1...8,388,607	+1000	user units/sec
Jog Acc	Jog Acceleration	1...134,217,727	+10000	user units/sec/sec
Jog Acc Mod	Jog Acceleration Mode	LINEAR/SCURVE	LINEAR	n/a
Hi Limit	High Count Limit	–8,388,608...+8,388,607	+8,388,607	user units
Lo Limit	Low Count Limit	–8,388,608...+8,388,607	–8,388,608	user units
Home Positn	Home Position	–8,388,608...+8,388,607	0	user units
Home Offset	Home Offset Value	–32,768...+32,767	0	user units
Fnl Hm Vel	Final Home Velocity	1...8,388,607	+500	user units/sec
Find Hm Vel	Find Home Velocity	1...8,388,607	+2000	user units/sec
Home Mode	Find Home Mode	HOMESW/MOVE+/ MOVE–	HOMESW	n/a
Posn Offset	Position Offset	–8,388,608...+8,388,607	0	user units
Gradient	Transducer Gradient	8500...12000	9050	nanoseconds/inch
Recircs	Transducer Recirculations	1...128	4	n/a

* Scaling dependent configuration parameter, value range depends upon user scaling.

User Units, Counts. The ratio of User Units to Counts sets the number of programming units for each feedback count. This allows the user to program the Power Mate APM in units appropriate for the application. The RANGE for user units and counts is 1 to 65,535. The RATIO of user units to counts must be in the range 8:1 to 1:32. For example, if there is 1.000 inch of travel for 8000 feedback counts, a user units:counts ratio of 1000:8000 sets 1 user unit = 0.001 inch. Default is 1:1.

OT Limit Sw. Selects whether the Power Mate APM uses the hardware Overtravel Limit Switch inputs. If Limit Switches are DISABLED, the Limit Switch inputs may be used as general purpose inputs. If ENABLED, then 24 V dc must be applied to both of the inputs in order for the Power Mate APM to operate. If not, then whenever the drive is enabled a Limit Switch Input Error will occur. The Jog and Clear Error %Q bits may be used simultaneously to back away from the Limit Switch. Default: ENABLED.

Pos EOT. Positive Software End of Travel limit (User Units). If the Power Mate APM is programmed to go to a position greater than the Positive EOT then an error will result and the Power Mate APM will not allow axis motion. Default: +8,388,607

Neg EOT. Negative Software End of Travel limit (User Units). If the Power Mate APM is programmed to go to a position less than the Negative EOT then an error will result and the Power Mate APM will not allow axis motion. Default: –8,388,608

If Pos / Neg EOT limits are both set to zero, the Power Mate APM uses +8,388,607 / –8,388,608 instead.

Pos Err Lim. Position Error Limit (User Units). The maximum Position Error *(Commanded Position - Actual Position*) allowed when the Power Mate APM is controlling a servo. Position Error Limit should normally be set to a value 10% to 20% higher than the highest position error encountered under normal servo operation. Default: 4096. The range formula for Position Error Limit is:

$$256 \times (\text{user units / counts}) \le \text{Position Error Limit} \le 60{,}000 \times (\text{user units / counts})$$

If Velocity Feedforward is not used, Position Error Limit can be set to a value approximately 20% higher than the position error required to produce a 10 volt command. *The Position Error* (User Units) required to produce a 10V command with 0% Velocity Feedforward is:

$$\text{Position Error (user units)} = \frac{\text{Position Loop Time Constant (ms)} \times \text{Servo Velocity @ 10v (user units / sec)}}{1000}$$

If Velocity Feedforward is used to reduce the following error, a smaller error limit value can be used, but in general, the error limit value should be 10% - 20% higher than the largest expected following error.

Note

An *Out of Sync* error will occur and cause a fast stop if this Error Limit Value is exceeded by more than 1000 counts. The Power Mate APM attempts to prevent an *Out of Sync* error by temporarily halting the internal command generator whenever position error exceeds the Position Error Limit. Halting the command generator allows the position feedback to catch up and reduce position error below the error limit value.

If the feedback does not catch up and the position error continues to grow, the *Out of Sync* condition will occur. Possible causes are:

1. erroneous feedback wiring
2. feedback device coupling slippage
3. servo drive failure.

In Pos Zone. In Position Zone (User Units). When the magnitude of servo position error is less than or equal to this value and neither Jog nor Move at Velocity is commanded, the *In Zone* status bit will be set. In Position Zone also determines the position error at which *Pmove*s are considered to be complete. Default: 10

Pos Loop TC. Position Loop Time Constant (milliseconds). The desired servo position loop time constant. The lower the value, the faster the system response. Values which are too low will cause system instability and oscillation. For accurate tracking of the commanded velocity profile, "Pos Loop TC" should be 1/4 to 1/2 of the MINIMUM system deceleration time. Setting Position Loop Time Constant to 0 will place the Power Mate APM in "open loop" mode where only Velocity Feedforward is used to produce the analog velocity command output. **The Position Loop Time Constant will not be accurate unless the "Vel at 10v" value is set correctly.** Default: 1000

Vel at 10v. Actual Servo Velocity (User Units/second) for an Power Mate APM velocity command output of 10v. **This value must be configured correctly in order for the "Pos Loop TC" and "Vel FF %" factors to be accurate.** The Power Mate APMs *Force D/A Output* %AQ immediate command and the *Actual Velocity* %AI status word can be used to determine the proper configuration value. The allowed range for Velocity at 10 Volts, which is affected by scaling, is:

400 * (user units/counts) ≤ Velocity at 10 Volts ≤ 1,000,000 * (user units/counts)

For digital servos, the *Velocity at 10 v:* configuration field should be set to a conversion constant value of 139820 multiplied by the decimal value of the *User Units* to *Counts* ratio. For example: with a *User Unit* value of 1 and a *Counts* value of 2 the decimal value of the ratio would be 0.5. The conversion constant multiplied by 0.5 yields the value 69910 for the *Velocity at 10 v (1/2 * 139820 = 69910)*

Default: 4000

Vel FF %. Velocity Feedforward gain (percent). The percentage of Commanded Velocity that is added to the Power Mate APM velocity command output. Increasing Velocity Feedforward causes the servo to operate with faster response and reduced position error. Optimum feedforward values are 80-90 %. **The "Vel at 10v" value must be set correctly for proper operation of Velocity Feedforward. Default: 0**

Intgr TC. Integrator Time Constant (ms). This is the time constant for the position error integrator in milliseconds. This value indicates the length of time in which 63% of the position error will be removed. For example, if the Integrator Time Constant is 1000, or 1 second, the *Position Error* would be reduced to 37% of its initial value after 1 second. A value of zero turns off the integrator. The Integrator Time Constant should be 5 to10 times greater than the Position Loop Time Constant to prevent instability and oscillation.

In Digital mode the position error integrator is not used and Intgr TC is used to configure the digital servo Velocity Loop Gain.

Default: 0

Intgr Mode. Integrator Mode. Operating mode for position error integrator. OFF means the integrator is not used. CONTINUOUS means the integrator runs continuously, even during servo motion. IN ZONE means the integrator only runs when the In Zone status bit is set. Default: OFF

When Digital Servo Mode is selected, the position error integrator is not used. The Integrator Mode field is used to configure Incremental or Absolute feedback type for the digital serial encoder. OFF or IN ZONE means that feedback counts from the serial encoder will be interpreted as *INCREMENTAL* counts, encoder battery alarms are not reported. CONTINUOUS means the serial encoder feedback will be considered as *ABSOLUTE,* battery alarms will be reported. See Appendix G for more information about using absolute mode encoders.

Rev Comp. Reversal Compensation (User Units). A compensation factor which allows the servo to reverse direction and still provide accurate positioning in systems exhibiting backlash. Default: 0.

DisDly (ms). Servo Drive Disable Delay (milliseconds). The time delay from zero velocity command to the drive enable output switching off. Disable Delay is effective when the *Enable Drive* %Q bit is turned off or certain error conditions occur. Disable Delay should be longer than the deceleration time of the servo from maximum speed. Default: 100

Jog Vel. Jog Velocity (User Units / second). The velocity at which the servo moves during a *Jog* operation. Default: 1000

Jog Acc. Jog Acceleration Rate (User Units / second / second). The acceleration rate used during *Jog, Find Home, Move at Velocity,* and *Abort* operations. The *Jog Acceleration* is used when an acceleration has not been programmed. Default: 10000

Jog Acc Mod. Jog Acceleration Mode (LINEAR or SCURVE). The acceleration mode for *Jog, Find Home, Move at Velocity,* and *Abort* operations. LINEAR causes commanded velocity to change linearly with time. SCURVE causes commanded velocity to change more slowly than the linear mode at the beginning and end of acceleration intervals. Default: LINEAR

Hi Limit. High Count Limit (User Units). When moving +, the Actual Position will roll over to the low limit when this value is reached. The Count Limits are used for rotary applications. Default: 8,388,607

Lo Limit. Low Count Limit (User Units). When moving –, the Actual Position will rollover to the high limit when this value is reached. The Count Limits are used for rotary applications. Default: –8,388,608

If Hi / Low Count limits are both set to zero, the Power Mate APM uses default limits +8,388,607 / –8,388,608 instead.

Home Positn. Home Position (User Units). The value assigned to *Actual Position* at the end of a *Find Home* cycle. Default: 0

Home Offset. Home position Offset (User Units). The offset of the servo final stopping point at the completion of a *Find Home* cycle. Home Offset adjusts the final servo stopping point relative to the Encoder marker. Default: 0

Fnl Home Vel. Final Home Velocity (User Units / second). The velocity at which the servo seeks the final Home Switch transition and Encoder Marker pulse at the end of a Find Home cycle. Final Home Velocity must be slow enough to allow a 10 ms (filter time) delay between the final Home Switch transition and the Encoder Marker pulse. Default: 500

Find Home Vel. Find Home Velocity (User Units / second). The velocity at which the servo seeks the initial Home Switch transitions during the *Find Home* cycle. If desired, Find Home Velocity can be set to a high value to allow the servo to quickly locate the Home Switch. Default: 2000

Posn Offset. Absolute Position Offset (User Units). This configuration item is only used with absolute feedback devices (LINEAR). Posn Offset is used to adjust the absolute position read from the feedback device and reported as *actual position* in the Power Mate APM %AI data. Default: 0

Gradient. Transducer Gradient (nanoseconds / inch), LINEAR feedback type only. Gradient should be set to the calibration value for the transducer actually used. A typical transducer gradient is 9.050 microseconds / inch = 9050 nanoseconds / inch. Default: 9050

Recircs. (LINEAR feedback type only). The value chosen for Recircs represents a multiplier for transducer resolution and update time. A high value for recircs allows high resolution but may slow the transducer update time enough to degrade system stability. Default: 4

Program Zero

Program Zero is a short default motion program (20 commands maximum) which is defined in the configuration software and downloaded whenever the Power Mate APM is initialized by the PLC. Program Zero is programmed by entering motion commands in an English-language format similar to that of the Motion Programmer.

In a 2-axis module, the Power Mate APM determines whether Program Zero is an axis 1, axis 2 or multi-axis program according to which axis or axes are used in the program. For example, if all Program Zero commands contain Axis number = 1, Program Zero will be classified as a Single Axis program for Axis 1. Therefore Program Zero will be allowed to execute **concurrently** with another Single Axis Program for Axis 2.

The Program Zero commands are entered using the Function keys (F1-F9). Paired with each command is a data field for entering either a signed integer or the number of an Power Mate APM parameter, as appropriate for the configured command.

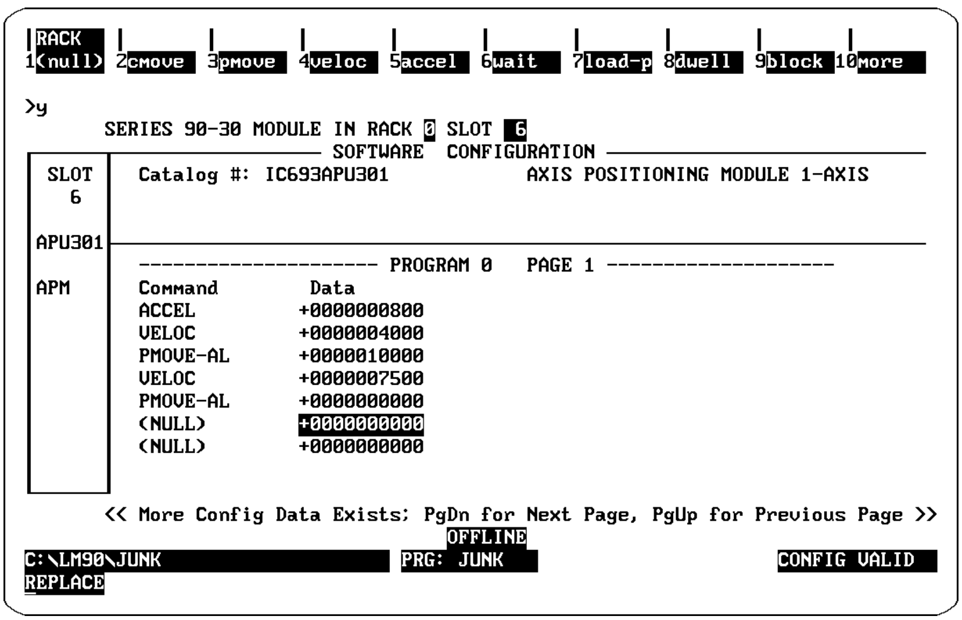


Figure 3-1. Example Program Zero Editor Screen

Program Zero Programmer Instruction Format

A Program Zero instruction consists of a command and associated data describing the command. The command and data are entered into areas on the program editor screen called fields. These fields are described below. From left to right the fields are:

Command Name Field. The English-language designation for the command. To enter a command, position the cursor on a command name field then press the desired function key (F1-F9). Most of the commands fall into groups such as CMOVE, PMOVE, VELOC, ACCEL, and so forth. The variations of these commands can be displayed (using the Tab key) after pressing the function key for the particular group.

For example, to program a CMOVE-IS-P (Continuous Move, incremental, s-curve, using a parameter), first select a command field then press the *CMOVE* function key. This causes CMOVE-AL to be displayed in the command name field of the screen. The variations of the command are cycled through by pressing the Tab key.

To program a JUMP command press F10 (MORE) and then press F1 (JUMP).

Operand Field. Paired with each command is a data field. In the data field enter either a signed integer or the number of a parameter (0- 255), as appropriate for the configured command. Parameters 1-20 can be loaded with data using the LOAD-P command.

Table 3-4. Program Zero Motion Program Commands

Command Name	Definition	Range	Default
(NULL)	Do nothing	0	0
BLOCK	Define Block Number	1..65,535	1
BLOCK-SYNC	Define Synchronous Block Number	1..65,535	1
CMOVE-AL	Continuous move, Absolute, Linear	–8,388,608.. 8,388,607	1
CMOVE-AL-P	Continuous move, Absolute, Linear, Use data in Parameter	0..255	1
CMOVE-AS	Continuous move, Absolute, S-curve	–8,388,608.. 8,388,607	1
CMOVE-AS-P	Continuous move, Absolute, S-curve, Use data in Parameter	0..255	1
CMOVE-IL	Continuous move, Incremental, Linear	–8,388,608.. 8,388,607	1
CMOVE-IL-P	Continuous move, Incremental, Linear, Use data in Parameter	0..255	1
CMOVE-IS	Continuous move, Incremental, S-curve	–8,388,608.. 8,388,607	1
CMOVE-IS-P	Continuous move, Incremental, S-curve, Use data in Parameter	0..255	1
PMOVE-AL	Positioning move, Absolute, Linear	–8,388,608.. 8,388,607	1
PMOVE-AL-P	Positioning move, Absolute, Linear, Use data in Parameter	0..255	1
PMOVE-AS	Positioning move, Absolute, S-curve	–8,388,608.. 8,388,607	1
PMOVE-AS-P	Positioning move, Absolute, S-curve, Use data in Parameter	0..255	1
PMOVE-IL	Positioning move, Incremental, Linear	–8,388,608.. 8,388,607	1
PMOVE-IL-P	Positioning move, Incremental, Linear, Use data in Parameter	0..255	1
PMOVE-IS	Positioning move, Incremental, S-curve	–8,388,608.. 8,388,607	1
PMOVE-IS-P	Positioning move, Incremental, S-curve, Use data in Parameter	0..255	1
VELOC	Set Velocity	1..8,388,607	2000
VELOC-P	Set Velocity to data in Parameter	0..255	1
ACCEL	Set Acceleration	1..134,217,727	5000
ACCEL-P	Set Acceleration to data in Parameter	0..255	1
WAIT	Wait for CTL XX bit to go high before moving	1..12	1
LOAD-P01 ↓ **LOAD-P20**	Load Power Mate APM Parameter register number	–8,388,608.. 8,388,607	0
DWELL	Wait for specified time in milliseconds	0..60,000	0
DWELL-P	Wait X milliseconds, X is the value in the Parameter	0..255	1
JUMP-UNCOND	Continue program execution at given block #	1..65,535	1
JUMP-CTL01 ↓ **JUMP-CTL12**	When the given CTL bit goes high during the current block execution, transfer program execution to the given block #	1..65,535	1

Program Zero Motion Command Descriptions

Each command is briefly described below. For a comprehensive explanation of Motion Programming on the Power Mate APM, refer to GFK-0664A, *The Series 90 Power Mate APM Programmer's Manual*.

Acceleration (ACCEL). This command is used to specify the axis acceleration and deceleration rate for subsequent moves. Once encountered, the specified rate will remain in effect until overridden by a later *Acceleration* command.

Block (BLOCK). Block numbers are used to monitor and synchronize program execution, terminate jump testing, and as jump destinations. Block number is an actual command in the Program Zero Editor.

Cmove (CMOVE). Continuous Move - this command is used when it is not necessary for the axis to be within the configured *In Position Zone* before proceeding to the next command. If no previous acceleration or velocity has been specified in a motion program, the configured *Jog Acceleration* and / or *Jog Velocity* will be used.

Dwell (DWELL). This command causes motion to cease for a specified time period (in milliseconds) before proceeding to the next command.

Jump (JUMP). This command is used to jump to another location in the program under certain specified states of the Faceplate control inputs (CTL 1-8) and %Q control outputs (CTL 9-12). The jump will occur when the condition tests "true" (logic 1). An unconditional *Jump* can also be selected. The jump may be forward or backward in the program. The jump condition will be tested as soon as the move prior to the *Jump* command has completed. A maximum of 220 jumps can exist for all programs and subroutines.

Once the condition testing is allowed to start, the test will occur once each millisecond (every 2 milliseconds for 2 Axis Power Mate APM) until a Block Number or another Jump command is encountered. This will allow continued testing while a move or series of moves takes place, if the *Jump* command is located ahead of the moves in the block. If a jump occurs during a move, the remainder of the move is aborted and the command at the destination location is immediately effective. Jump destinations must be limited to the bounds of the program containing the *Jump* command.

Load Parameter (LOAD). This command initializes or changes a Power Mate APM parameter value. The new value becomes effective immediately when encountered in the program. Program Zero may load parameters 1 to 20.

Pmove (PMOVE). Positioning Move - this command is used when it is necessary for the axis to be within the configured *In Position Zone* before proceeding to the next command. If no previous acceleration or velocity has been specified in a motion program, the configured *Jog Acceleration* and / or *Jog Velocity* will be used.

Velocity (VELOC). This command specifies the velocity of axis motion. Once encountered, this command will remain in effect until overridden by a later *Velocity* command.

Wait (WAIT). This command synchronizes the start of axis motion with an external input or event reported in CTL 1-12. The start of motion is suspended until the bit being monitored is true.

Essential Configuration Parameters

To correctly configure the Power Mate APM, several configuration parameters must be properly set. This section explains how these parameters affect Power Mate APM operation and how the parameters relate to each other. While all parameters are important, some parameters are absolutely essential to correct operation.

Velocity at 10 volts

All Power Mate APM and servo functions depend on this value being correct for proper operation. This should be the velocity of the axis when the Power Mate APM outputs 10 volts to the servo. See Chapter 5, Servo System Startup Procedures to determine the correct value. The range for Velocity at 10 Volts will depend upon the user scaling. The allowed range is:

400 * (user units / counts) ≤ Velocity at 10 Volts ≤ 1,000,000 * (user units / counts)

For digital servos, he *Velocity at 10 v:* configuration field should be set to a conversion constant value of 139820 multiplied by the decimal value of the *User Units* to *Counts* ratio. For example: with a *User Unit* value of 1 and a *Counts* value of 2 the decimal value of the ratio would be 0.5. The conversion constant multiplied by 0.5 yields the value 69910 for the *Velocity at 10 v (1/2 * 139820 = 69910)*

Position Loop Time Constant

The lower the Position Loop Time Constant, the faster the axis will respond. However, if the Time Constant is too low, the system may become unstable or even oscillate. When decelerating, especially at high speeds, the Power Mate APM could command a servo to stop at a certain point faster than the servo could respond. This would result in overshoot.

For accurate tracking of the commanded velocity profile, “Pos Loop TC” should be 1/4 to 1/2 of the MINIMUM system deceleration time.

For users familiar with servo bandwidth expressed in rad/sec:

Bandwidth (rad/sec) = 1000 / Position Loop Time Constant (ms)

For users familiar with servo gain expressed in ipm/mil:

Gain (ipm/mil) = 60 / Position Loop Time Constant (ms)

Gain (ipm/mil)	Bandwidth (rad/sec)	Position Loop Time Constant (ms)
0.5	8.5	120
0.75	12.5	80
1.0	16.6	60
1.5	25.1	40
2.0	33.4	30
2.5	41.8	24
3.0	50	20

Open Loop Mode

For applications which do not require feedback or employ very crude positioning systems, an Open Loop Mode exists. Setting a zero Position Loop Time Constant, which indicates that the positioning loop is disabled, selects this mode. Note that in Open Loop Mode, the only way to generate motion is to program a non-zero Velocity Feedforward. The *Position Error* is no longer used to generate motion because *Position Error* is based on feedback and Open Loop Mode ignores all feedback.

User Units and Counts

The Power Mate APM has a very powerful scaling feature. A User Units to Counts ratio can be configured to allow programming in other than default counts. In a simplified example, suppose an encoder feedback application has an encoder which produces 1,000 counts per revolution (250 lines) and is geared to a machine which produces one inch per revolution. The default units would be one thousandth of an inch per count. However, you may want to write programs and use the Power Mate APM with metric units. A ratio of 2540 User Units to 1000 Counts can be configured to allow this. With this ratio, one user unit would represent .01 millimeters. 2540 user units would represent 25.4 millimeters (one inch) of travel.

The range for both User Units and Counts is 1 to 65,535. However, the ratio of User Units to Counts must be between 8:1 and 1:32. A more detailed example is described below.

Example

A machine has a motor with a 2000 line encoder connected through a 20:1 gear reduction to a 14.336 inch diameter spur gear. The programmer wishes to program in 0.01 inch resolution.

The following is the given data:

> 14.336 inch diameter spur gear
>
> 2000 line encoder
>
> 20:1 reduction gear
>
> 0.01 inch is the desired programming unit

First, determine the number of Encoder Counts per revolution of the spur gear.

> 2000 lines per rev * 4 * 20 = 160,000 Counts per rev
>
> > The 4 results from A Quad B generating 4 counts per encoder line
> >
> > The 20 results from the reduction gear

Next, determine the number of User Units per revolution using 0.01 inch resolution.

14.336 inches * π = 45.0378 inches per revolution

45.0378 inches per rev / 0.01 inches per User Unit = 4503.78 User Units per revolution

This User Units to Counts ratio would be 4503 / 160,000 or 0.02815 which is about 1:36. This ratio is greater than 1:32 so the resolution must be adjusted. Assume the new programming unit is 0.001 inch.

Determine the revised User Units per revolution using 0.001 inch resolution.

14.336 inches * π = 45.0378 inches per revolution

45.0378 inches per rev / 0.001 inches per User Unit = 45,037.8 User Units per revolution

This User Units to Counts ratio would be 45,037 / 160,000 or 0.2815 which is about 1:3.6. Thus values of 2815 for User Units and 10,000 for Counts could be used to obtain 0.001 inch resolution.

Acceleration Mode

The Power Mate APM supports two types of acceleration, Linear and Scurve. Linear motion consists of constant linear acceleration and deceleration to specified velocities. A profile of velocity versus time would show straight lines. Scurve motion uses a variable acceleration. An Scurve acceleration would begin slowly and increase, up to the specified acceleration, then decrease back to zero when the specified velocity was reached. ***Scurve motions require twice the time and distance to change velocity compared to linear motions with the same acceleration.***

CMOVE and PMOVE motion commands specify the acceleration mode for all programmed motion, except the jump stop condition which is described in Chapter 6. *Jog, Find Home, Abort, and Move at Velocity* use the configured *Jog Acceleration Mode.*

Important Configuration Considerations

The Integrator will attempt to remove all Position Error and can be used in two modes: IN ZONE and CONTINUOUS. The Integrator Time Constant is the amount of time in which 63% of the *Position Error* will be removed. When using the IN ZONE mode, at the end of a move when the *Actual Position* nears the *Commanded Position* and the servo is about to stop, the axis may stop a little early if the servo cannot take out all of the error. To increase precision, the Integrator can then work to integrate out all error. When using the CONTINUOUS mode, the Integrator will always try to remove all *Position Error,* even when moving. Rapidly changing velocities can make the Integrator over or under compensate. CONTINUOUS mode is not recommended for positioning applications. The Integrator Time Constant must be five to ten times greater than the *Position Loop Time Constant* or oscillation might occur.

Software End of Travel Limits, positive and negative, are used to bound commanded motion. The Power Mate APM will not execute any programmed motion which takes the *Commanded Position* to or past an EOT Limit. If the *Position Valid* %I bit is ON, a *Jog* will immediately be stopped at the EOT Limit. *Jogs* with *Position Valid* OFF and *Move at Velocity* immediate commands ignore the EOT Limits. This enables movement outside the EOTs while setting up a system.

High and Low Count Limits can be used for rotary type motion in which a servo can move forever in either direction. When a Count Limit is reached, the reported position "wraps around" to the opposite limit where it can continue changing. Thus if the Count Limits are set equal to or within the *End of Travel* Limits, the *End of Travel Limits* will never be reached. ***In 1-axis Power Mate APM release 1.xx firmware, Count Limits are not used and MUST be set to zero in order to configure the Power Mate APM.***

In Rotary mode, Jogs or Incremental Moves can be used to cause continuous motion in either direction and the EOT limits will never be exceeded.

In EOT Mode (Non Rotary Mode) Jogs and Programmed Motions beyond either EOT limit are not allowed.

Rotary Mode selection is based on the relative values of Hi / Lo Count Limits and the Pos / Neg EOT limits:

> The Power Mate APM will be in Rotary Mode if the Hi Count Limit is <= the Pos EOT limit **AND** the Lo Count limit is >= the Neg EOT limit.
>
> The Power Mate APM will be in EOT Mode (Non Rotary Mode) if the Hi Count Limit is > the Pos EOT limit **OR** if the Lo Count limit is < the Neg EOT limit.

Rotary Mode is the DEFAULT operating mode for the Standard Control Loop because the default Count limits and EOT limits are +8,388,607 / –8,388,608.

The Overtravel Limit Switches are external hardware switches. The switch status is returned in the CTL bits CTL05 - CTL08. The Power Mate APM can be configured to use these CTL bits as Overtravel Limit Switches. If the switches are enabled, whenever the drive is enabled and an Overtravel bit is OFF the Power Mate APM commands all motion on that axis to immediately cease. If this happens, the *Jog* and *Clear Error* %Q bits can be turned on simultaneously to move the servo away from the Overtravel switch.

A Reversal Compensation feature on the Power Mate APM allows accurate positioning on systems with backlash. Backlash is exhibited by a servo that must move a small amount before the system begins moving when it switches directions. For example, consider a deadbolt door lock. Imagine the servo controls the key in the lock and the feedback reports movement of the bolt. When the servo turns the key counter-clockwise, the bolt moves left. However, as the servo turns the key clockwise, the bolt does not move until the key turns to a certain point. The Reversal Compensation feature adds in the necessary motion to move the servo to where motion will begin on the feedback device. The Power Mate APM removes the compensation when a move in the negative direction is commanded, and adds the compensation before a move in the positive direction.

The Servo Drive Disable Delay specifies how long the Power Mate APM will wait after a zero velocity is commanded before it turns OFF the Drive Enable output. Because turning OFF the Drive Enable Relay stops the Power Mate APM from commanding the servo, there are times when the relay should stay ON. For example, if the servo runs into an *End of Travel Limit* and the Drive Enable Relay was immediately turned OFF because of the error, the servo would continue moving until it coasted to a stop. Thus to allow the Power Mate APM to command and control a fast stop, the Drive Disable Delay should be longer than the deceleration time of the servo from maximum speed.

Chapter 4 Automatic Data Transfers

This chapter defines the data that is transferred automatically each sweep, without user programming, between the CPU and the Power Mate APM. This data is categorized as follows.

- **Input Status Data (Transferred from Power Mate APM to CPU)**
  - Status Bits: 32 bits of (%I) data
  - Status Words: 15 words of (%AI) data for a 1 axis Power Mate APM
    28 words of (%AI) data for a 2 axis Power Mate APM
- **Output Command Data (Transferred from CPU to Power Mate APM)**
  - Discrete Commands: 32 bits of (%Q) data
  - Immediate Commands: 6 words of (%AQ) data

%I Status Bits

The following %I Status Bits are transferred automatically from the Power Mate APM to the CPU each sweep. The actual addresses of the Status Bits depend on the starting address configured for the %I references. See Table 3-1, Module Setup Data. The bit numbers listed in the following table are offsets to this starting address.

Table 4-1. %I Status Bits for the 1-Axis Power Mate APM (IC693APU301)

Bit *	Description	Bit *	Description
00	Axis Enabled Axis 1	16	Front Panel Input CTL01 Status (Strobe 1 State)
01	Position Valid Axis 1	17	Front Panel Input CTL02 Status
02	Drive Enabled Axis 1	18	Front Panel Input CTL03 Status (Home Switch 1)
03	Program Active Axis 1	19	Front Panel Input CTL04 Status
04	Moving Axis 1	20	Front Panel Input CTL05 Status (+ Overtravel)
05	In Zone Axis 1	21	Front Panel Input CTL06 Status (– Overtravel)
06	Position Strobe Axis 1	22	Front Panel Input CTL07 Status
07	In Error Limit Axis 1	23	Front Panel Input CTL08 Status
08	reserved	24	Configuration Complete
09	reserved	25	reserved
10	reserved	26	reserved
11	reserved	27	reserved
12	reserved	28	reserved
13	reserved	29	reserved
14	reserved	30	PLC Control Active
15	reserved	31	Error

* The bit numbers represent an offset to the starting address for %I references.

Table 4-2. %I Status Bits for the 2-Axis Power Mate APM (IC693APU302)

Bit *	Description		Bit *	Description
00	Axis Enabled	Axis 1	16	Front Panel Input CTL01 Status (Strobe 1 State)
01	Position Valid	Axis 1	17	Front Panel Input CTL02 Status (Strobe 2 State)
02	Drive Enabled	Axis 1	18	Front Panel Input CTL03 Status (Home Switch 1)
03	Program Active	Axis 1	19	Front Panel Input CTL04 Status (Home Switch 2)
04	Moving	Axis 1	20	Front Panel Input CTL05 Status (+ Overtravel 1)
05	In Zone	Axis 1	21	Front Panel Input CTL06 Status (– Overtravel 1)
06	Position Strobe	Axis 1	22	Front Panel Input CTL07 Status (+ Overtravel 2)
07	In Error Limit	Axis 1	23	Front Panel Input CTL08 Status (– Overtravel 2)
08	Axis Enabled	Axis 2	24	Configuration Complete
09	Position Valid	Axis 2	25	reserved
10	Drive Enabled	Axis 2	26	reserved
11	Program Active	Axis 2	27	reserved
12	Moving	Axis 2	28	reserved
13	In Zone	Axis 2	29	reserved
14	Position Strobe	Axis 2	30	PLC Control Active
15	In Error Limit	Axis 2	31	Error

* The bit numbers represent an offset to the starting address for %I references.

Axis Enabled. The *Axis Enabled* status bit is ON when the Power Mate APM is ready to receive commands and control a servo. An error condition which stops the servo will turn *Axis Enabled* OFF.

Position Valid. The *Position Valid* status bit indicates that the value in the %AI *Actual Position* status word has been initialized by a *Set Position* command or successful completion of the Find Home cycle. *Position Valid* must be ON in order to execute a motion program.

If the Power Mate APM is configured to use a Linear absolute feedback transducer, *Position Valid* is automatically set whenever a correct transducer signal is received.

When A Quad B encoder feedback is used, the *Position Valid* status bit can be cleared by PLC commands. To do this, the PLC should send the %Q commands for Find Home and Abort on the same PLC sweep.

Drive Enabled. The *Drive Enabled* status bit indicates the state of the *Enable Drive* discrete command and the relay contact supplied by the Power Mate APM. The ON state of the *Drive Enabled* status bit corresponds to the CLOSED state of the relay contact. *Drive Enabled* is cleared following power-up or an error condition which stops the servo.

Program Active. The *Program Active* status bit for each axis indicates that a Motion Program (0– 10), or a %AQ Move command (27h) is executing on that axis. On a two axis Power Mate APM, executing a multi-axis program will set both *Program Active* bits.

Moving. The *Moving* status bit is set when commanded velocity is non-zero, otherwise it is cleared. All *Positioning Move*, *Continuous Move*, *Jog*, and *Move at Velocity* commands will cause the *Moving* bit to be set. The *Force D/A Output* command will not set the *Moving* bit.

In Zone. The *In Zone* status bit indicates that the position error is equal to or less than the configured *In Position Zone* value. This condition occurs at the end of each *Positioning Move* command or any time the axis commands are halted and the actual position has caught up to the commanded position (e.g. for *Dwell*, *Feedhold*, or % *Feedrate* = 0).

Position Strobe. The *Position Strobe* status bit indicates that the Strobe Input at the I/O connector has captured an axis position that is currently indicated by the *Strobe Position* %AI status word. The data will remain in the *Strobe Position* status word until the *Position Strobe* bit is cleared by the *Reset Strobe Flag* %Q bit. Once the *Position Strobe* bit is cleared, new data may be captured by another Strobe Input. Position data will be captured within 1 millisecond from Strobe Input (2 milliseconds for 2 axis Power Mate APM).

In Error Limit. The *In Error Limit* status bit is set when the absolute value of the position error exceeds the configured *Position Error Limit* value. When the *In Error Limit* status bit is set, commanded velocity and commanded position are frozen to allow the axis to ”catch up” to the commanded position.

Faceplate Input Status CTL01-08 These inputs always indicate the state of the external input devices connected to the Power Mate APM faceplate terminals CTL01-08. These inputs (as well as CTL09-CTL12 from the PLC %Q table) may be tested by the Power Mate APM during the execution of *Wait* and *Conditional Jump* commands.

Several inputs may serve alternate purposes:

CTL01	Axis 1 Strobe input	CTL02	Axis 2 Strobe input
CTL03	Axis 1 Find Home switch input	CTL04	Axis 2 Find Home switch input
CTL05	Axis 1 Overtravel (+) switch input	CTL07	Axis 2 Overtravel (+) switch input
CTL06	Axis 1 Overtravel (–) switch input	CTL08	Axis 2 Overtravel (–) switch input

Note

In Digital mode, CTL04, CTL07, and CTL08 are not available because faceplate connector B is used for the Digital Servo Interface module communications cable. In this mode, CTL05 is used for the axis 2 Home Switch if axis 1 Overtravel switches are disabled in the configuration software.

Configuration Complete. The *Configuration Complete* status bit is set by %AQ Immediate command 49h. This status bit is cleared whenever the PLC sends a reset command or new configuration to the Power Mate APM. Configuration Complete can be set by a PLC program after other %AQ Immediate commands such as *In Position Zone* or *Position Loop Time Constant* have been sent to the Power Mate APM. The status bit can then be monitored by the PLC. If the bit is ever cleared, then the Power Mate APM has been reset or reconfigured and the PLC should resend all necessary %AQ configuration commands before setting the bit again.

PLC Control Active. Normally the *PLC Control Active* status bit is set, indicating that the %Q discrete commands or %AQ immediate commands from the PLC can control the Power Mate APM. PLC Control Active is cleared only when the Status screen in the Motion Programmer is used instead of the PLC to control the Power Mate APM, a capability not yet implemented.

Error. This status bit is set when the Power Mate APM detects any error. When set, the %AI *Status Code* word identifies the error condition. *Clear Error* is the only command that will clear the *Error* status bit and the associated *Status Code* word. If the condition causing the error is still present, the *Error* status bit will not be cleared.

%AI Status Words

The following %AI Status Words are transferred automatically from the Power Mate APM to the CPU each sweep.

The actual addresses of the Status Words depend on the starting address configured for the %AI references. See Table 3-1, Module Setup Data. The word numbers listed in the following table are offsets to this starting address.

Table 4-3. %AI Status Words for the 1-Axis Power Mate APM (IC693APU301)

Word*	Description	
000	Status Code	
001	Command Block Number	Axis 1
002–003	Commanded Position**	Axis 1
004–005	Actual Position	Axis 1
006–007	Strobe Position	Axis 1
008–009	Commanded Velocity	Axis 1
010–011	Actual Velocity	Axis 1
012–013	Position Error***	Axis 1
014	Analog Input Value	

* The word numbers represent an offset to the starting address for %AI references.

** The %AQ Immediate Command *Select Return Data* (40h) can be used to select Commanded Position or other data such as module firmware revision.

*** If Module configuration parameter *%AI Pos Err* is DISABLED, Analog Input Value will be reported instead of Position Error in %AI word 012. This allows compatibility with APM release 1.00 and 1.10 firmware.

Table 4-4. %AI Status Words for the 2-Axis Power Mate APM (IC693APU302)

Word*	Description	
000	Status Code	
001	Command Block Number	Axis 1
002–003	Commanded Position**	Axis 1
004–005	Actual Position	Axis 1
006–007	Strobe Position	Axis 1
008–009	Commanded Velocity	Axis 1
010–011	Actual Velocity	Axis 1
012–013	Position Error	Axis 1
014	Analog Input Value	
015	Command Block Number	Axis 2
016–017	Commanded Position**	Axis 2
018–019	Actual Position	Axis 2
020–021	Strobe Position	Axis 2
022–023	Commanded Velocity	Axis 2
024–025	Actual Velocity	Axis 2
026–027	Position Error	Axis 2

* The word numbers represent an offset to the starting address for %AI references.

** The %AQ Immediate Command *Select Return Data* (40h) can be used to select Commanded Position or other data such as module firmware revision.

Status Code. *Status Code* indicates the current operating status of the module. When the *Error* flag is set, it contains an error code number which describes the condition causing the error.

For a list of Power Mate APM error codes refer to Appendix A, Error Codes.

Command Block Number. *Command Block Number* indicates the block number of the command that is presently being executed in the active Program or Subroutine. It changes at the start of each new block as the program commands are executed, and thus identifies the present operating location within the program.

Commanded Position. *Commanded Position* (user units) is where the axis is commanded to be at any instant in time. The difference between *Commanded Position* and *Actual Position* is the *Position Error* value which produces the Velocity Command to drive the axis. The rate at which the *Commanded Position* is changed determines the velocity of axis motion.

If *Commanded Position* moves past either of the count limits, it will "roll over" to the other limit and continue in the direction of the axis motion.

Note

The *Select Return Data* %AQ Immediate Command can be used to select the reporting of alternate data in the *Commanded Position* location.

Actual Position. *Actual Position* (user units) is a value maintained by the Power Mate APM to represent the physical position of the axis. It is set to an initial value by the *Set Position* command or to *Home Position* by the *Find Home* cycle. It is updated by the motion of the feedback device.

If *Actual Position* moves past either of the count limits, it will "roll over" to the other limit and continue in the direction of the axis motion.

Strobe Position. *Strobe Position* (user units) contains the axis position when a Strobe Input occurs. When a Strobe Input occurs, the *Position Strobe* %I bit is set to indicate to the PLC that new Strobe data is available in the *Strobe Position* status word. The PLC must set the *Reset Strobe Flag* %Q bit to clear the Position Strobe %I bit.

Strobe Position will be maintained and will not be overwritten by additional Strobe Inputs until the Position Strobe %I bit has been cleared. If the Reset Strobe Flag output is left in the On state (thus holding Position Strobe input flag in the cleared state), then each Strobe Input that occurs will cause the axis position to be captured in *Strobe Position*.

The axis 1 *Strobe Position* is also placed in parameter 255 for use within motion programs. In a 2-axis Power Mate APM, the axis 2 *Strobe Position* is also placed in parameter 254.

This feature allows the strobe input to trigger a Conditional JUMP to a program block using the strobe position as the destination of a CMOVE or PMOVE command.

Commanded Velocity. *Commanded Velocity* (user units/sec) is a value generated by the Power Mate APM that indicates the instantaneous velocity command that is producing axis motion. At the beginning of a move it will increase at the acceleration rate, and once the programmed velocity has been reached, it will stabilize at the programmed velocity value.

Actual Velocity. *Actual Velocity* (user units/sec) is a value maintained by the Power Mate APM that is derived from the feedback device. Therefore, it represents the velocity of the axis movement.

Position Error. *Position Error* (user units) is the difference between *Commanded Position* and *Actual Position* at any instant in time.

Analog Input. *Analog Input* returns the digital value representing the voltage applied to the analog input terminal. +10 V is indicated by +32,000 and –10 V by –32,000.

The *Analog Input* is not available in Digital mode because faceplate connector B is used for the Digital Servo Interface module communications cable.

%Q Discrete Commands

The following %Q Outputs representing Discrete Commands are sent automatically to the Power Mate APM from the CPU each PLC sweep. A command is executed simply by turning on the Output Bit of the desired command.

The actual addresses of the Discrete Command bits depend on the starting address configured for the %Q references. See Table 3-1, Module Setup Data. The bit numbers listed in the following table are offsets to this starting address.

Table 4-5. %Q Discrete Commands for the 1-Axis Power Mate APM (IC693APU301)

Bit *	Description		Bit *	Description
00	Abort All Moves	Axis 1	16	CTL09 Output Control
01	Feedhold	Axis 1	17	CTL10 Output Control
02	Enable Drive	Axis 1	18	CTL11 Output Control
03	Find Home	Axis 1	19	CTL12 Output Control
04	Jog Plus	Axis 1	20	Execute Motion Program 0
05	Jog Minus	Axis 1	21	Execute Motion Program 1
06	Reset Strobe Flag	Axis 1	22	Execute Motion Program 2
07	reserved		23	Execute Motion Program 3
08	reserved		24	Execute Motion Program 4
09	reserved		25	Execute Motion Program 5
10	reserved		26	Execute Motion Program 6
11	reserved		27	Execute Motion Program 7
12	reserved		28	Execute Motion Program 8
13	reserved		29	Execute Motion Program 9
14	reserved		30	Execute Motion Program 10
15	reserved		31	Clear Error

* The bit numbers represent an offset to the starting address for %Q references.

Table 4-6. %Q Discrete Commands for the 2-Axis Power Mate APM (IC693APU302)

Bit *	Description		Bit *	Description
00	Abort All Moves	Axis 1	16	CTL09 Output Control
01	Feedhold	Axis 1	17	CTL10 Output Control
02	Enable Drive	Axis 1	18	CTL11 Output Control
03	Find Home	Axis 1	19	CTL12 Output Control
04	Jog Plus	Axis 1	20	Execute Motion Program 0
05	Jog Minus	Axis 1	21	Execute Motion Program 1
06	Reset Strobe Flag	Axis 1	22	Execute Motion Program 2
07	reserved		23	Execute Motion Program 3
08	Abort All Moves	Axis 2	24	Execute Motion Program 4
09	Feedhold	Axis 2	25	Execute Motion Program 5
10	Enable Drive	Axis 2	26	Execute Motion Program 6
11	Find Home	Axis 2	27	Execute Motion Program 7
12	Jog Plus	Axis 2	28	Execute Motion Program 8
13	Jog Minus	Axis 2	29	Execute Motion Program 9
14	Reset Strobe Flag	Axis 2	30	Execute Motion Program 10
15	reserved		31	Clear Error

* The bit numbers represent an offset to the starting address for %Q references.

Abort All Moves. This command command causes *any* motion in progress to halt at the **current Jog Acceleration** rate. Any pending programmed or immediate command is canceled and therefore not allowed to become effective. The abort condition is in effect as long as this command is on. If motion was in progress when the command was received, the *Moving* status bit will remain set and the *In Zone* status bit will remain cleared until the commanded velocity reaches zero and the *In Zone* condition is achieved.

Feedhold (On Transition). This command causes any motion in progress to halt at the active acceleration rate. Once the motion is stopped, the *Moving* status bit is cleared and the *In Zone* status bit is set when the In Zone condition is attained. *Jog* commands are allowed when in the Feedhold condition. After an ON transition, program motion will stop, even if the command bit transitions back OFF before motion stops.

Feedhold (Off Transition). This command causes any programmed motion interrupted by *Feedhold* to resume at the programmed acceleration and velocity rate. Additional program moves will then be processed and normal program execution will continue.

If jogging occurred while *Feedhold* was ON, the interrupted *Move* command will resume from where the axis was left after the *jog*. The *Move* finishes at the correct programmed velocity and continues to the original programmed position as if no *jog* displacement occurred.

Enable Drive. If the *Error* and *Drive Enabled* status bits are cleared, this command will cause the Drive Enable relay contact to close (enabling the drive) and the *Drive Enabled* bit to be set; otherwise, it has no effect. When the *Drive Enabled* bit is set, the path generation and position control functions are enabled and servo motion can be commanded. *Enable Drive* must be maintained ON to allow normal servo motion (except when using *Jog* commands).

Enable Drive must be turned off whenever power is removed from the servo drive. If this is not done, any servo drift could cause a rapid jump of the servo when power is reapplied to the drive.

Find Home. This command causes the Power Mate APM to establish the *Home Position* for systems with an incremental feedback device that also provides a marker pulse. A Home Limit Switch Input from the I/O connector roughly indicates the reference position for Home and the next marker encountered indicates the exact position. A configuration option allows the Home Limit Switch to be ignored. The configured *Home Offset* defines the location of *Home Position* as the offset distance from the Home Marker. Position Valid indication is set at the conclusion of the Home cycle.

Refer to Home Cycle in Chapter 6, Power Mate APM Motion Control for a description of the Find Home Cycle.

Jog Plus. When this command bit is ON, the axis moves in the positive direction at the configured *Jog Acceleration* and *Jog Velocity* as long as the *Jog Plus* command is maintained and the configured *Positive End Of Travel* limit is not encountered. *Jog Plus* may be used to jog off of the Negative End of Travel Limit Switch if the %Q Clear Error bit is also maintained on.

Jog Minus. When this command bit is ON, the axis moves in the negative direction at the configured *Jog Acceleration* and *Jog Velocity* as long as the *Jog Minus* command is maintained and the configured *Negative End Of Travel* limit is not encountered. *Jog Minus* may be used to jog off of the Positive End of Travel Limit Switch if the %Q Clear Error bit is also maintained on.

Reset Strobe Flag. The *Position Strobe* %I status bit flag informs the PLC that a Strobe Input has captured an axis position that is now stored in the *Strobe Position* status word. When the PLC acknowledges this data, it may use the *Reset Strobe Flag* %Q command bit to clear the *Position Strobe* %I status bit flag. Once the *Position Strobe %I* bit is set, additional Strobe Inputs will not cause new data to be captured. The flag must be cleared before another Strobe Position will be captured. As long as the *Reset Strobe Flag* %Q command bit is set, the *Position Strobe* bit flag will be held in the cleared state. In this condition, the latest Strobe Input position is reflected in the *Strobe Position* status word, although the flag cannot be used by the PLC to indicate when new data is present.

CTL09-CTL12 Output Controls. These command bits control the CMOS digital outputs on Faceplate I/O Connector B. The bits may also be tested by the Power Mate APM during execution of *Wait* or *Conditional Jump* commands.

Execute Motion Program 0-10. These commands are used to select stored programs for immediate execution. Each command uses a one shot action; thus a command bit must transition from OFF to ON each time a program is to be executed. Programs may be temporarily interrupted by a Feedhold command.

*When a program begins execution, Rate Override is always set to 100%***.** A Rate Override %AQ immediate command can be sent on the same sweep as the Execute Program %Q bit and will be effective as the program starts.

Only one Motion Program can be executed at a time per axis; the Program Active %I status bit must be OFF or Motion Program execution will not be allowed to start. A multi-axis Motion Program uses both axis 1 and axis 2, so both Program Active bits must be OFF to start a multi–axis Motion Program.

Clear Error. When an error condition is reported, this command is used to clear the *Error* status bit and its associated *Status Code* word. Error conditions that are still present (such as an *End of Travel* limit switch error) will not be cleared and must be cleared by some other corrective action .

%AQ Immediate Commands

Each PLC sweep, 6 words of data are automatically transferred from the CPU %AQ data to the Power Mate APM. These six words are used to send Immediate Commands from the PLC to the Power Mate APM. The first three words, offsets 0 through 2, are dedicated to axis 1 of the Power Mate APM. The second three words, offsets 3 through 5, are dedicated to axis 2 of the Power Mate APM. Thus one command may be sent to each axis of the Power Mate APM every sweep.

The only exception is the Load Parameter Immediate command which is axis independent. This command may be sent using either or both sets of three words. Thus two Load Parameter Immediate commands can be sent on the same sweep (one in the first three %AQ words and the other in the second three %AQ words).

In a 1-axis Power Mate APM, the only effective commands for the second set of %AQ words are Force D / A Output (Analog Output 2) and Load Parameter Immediate.

Even though the commands are sent each sweep, the Power Mate APM will act on a command ONLY if it changed since the last sweep. When any of the 6-byte data changes, the Power Mate APM will accept the data as a new command and respond accordingly.

The 6-byte format for the Immediate Commands is defined in Table 4-7. The actual addresses of the Immediate Command Words depend on the starting address configured for the %AQ references. See Table 3-1, Module Setup Data. **The word numbers listed in the following table are offsets to this starting address.**

The word offsets are shown in reverse order and in hexadecimal to simplify the data entry. The following example sends the Set Position command to axis 1. The first word, word 0, contains the actual command number. For the Set Position command, the command number is 0023h. The second and third words contain the data for the Set Position command which is a position. The second word, word 1, is the least significant word of the position and the third word, word 2, is the most significant word. To set a position of 3,400,250, first convert the value to hexadecimal. 3,400,250 decimal equals 0033E23A hexadecimal. For this value, 0033 is the most significant word and E23A is the least significant word. The data to be sent to the Power Mate APM would be:

Word 2	Word 1	Word 0	Command
0033	E23A	0023	Set Position 3,400,250

Setting up word 0 as a hex word and words 1 and 2 as a double integer in the PLC would simplify immediate command entry.

Table 4-7. Immediate Commands Using the 6-Byte Format (for both 1 and 2-Axis Power Mate APM)

Word 2		Word 1		Word 0 *		Immediate Command Definition
Word 5		Word 4		Word 3 *		
Byte 5	Byte 4	Byte 3	Byte 2	Byte 1	Byte 0	
xx	xx	xx	xx	00	00h	Null
xx	xx	xx	RO%	00	20h	Rate Override RO% = 0 ...120%
xx	xx	**	Incr	00	21h	Position Increment Without Position Update Incr = –128 ... +127 User Units
Velocity				00	22h	Move At Velocity Vel. = –8,388,608 ... + 8,388,607 User Units / sec
Position				00	23h	Set Position Pos. = –8,388,608 ... + 8,388,607 User Units
xx	xx	D / A Output		00	24h	Force D / A Output D / A Output = –32000 ... + 32000
xx	xx	**	Incr	00	25h	Position Increment With Position Update Incr = –128 ... +127 User Units
xx	xx	In Position Zone		00	26h	In Position Zone Range = 0 ... 2000
Data				Move Type	27h	Move Command
Velocity				00	28h	Jog Velocity Vel. = +1 ... + 8,388,607 User Units / sec
Acceleration				00	29h	Jog Acceleration Acc. = +1 ... + 8,388,607 User Units / sec / sec
xx	xx	Time Constant		00	2Ah	Position Loop Time Constant Time Constant = 0,5 ... 10000
xx	xx	xx	VFF%	00	2Bh	Velocity Feedforward VFF% = 0 ... 100%
xx	xx	Integrator Time Constant		00	2Ch	Integrator Time Constant ITC = 0, 10 ... 65,535 ms
xx	xx	xx	VLGN	00	2Eh	Velocity Loop Gain VLGN = 0 ... 255
xx	xx	xx	Torque Limit %	00	2Fh	Torque Limit Range=1-100%

Table 4-8. Immediate Commands Using the 6-Byte Format (for both 1 and 2-Axis Power Mate APM) (continued)

xx	xx	Offset		Mode	40h	Select Return Data
xx	MCON CTRL	xx	xx	00	45h	Configuration Modes
Feedback Offset (counts)				xx	46h	Absolute Feedback Offset
xx	xx	xx	xx	00	49h	Set Configuration Complete
xx	xx	xx	xx	00	4Ah	Update Flash Memory
Parameter Data				Par #h	50h	Load Parameter Immediate Par # = 0 ... 255 Parameter Data = Range depends on parameter usage.

* = The word numbers represent an offset to the starting address for the %AQ references.
** = Only 00 or FFh are acceptable.
xx = don't care

Null. This is the default %AQ Immediate command. Since the %AQ words are automatically transferred each PLC sweep, the *Null* command should always be used to avoid inadvertent execution of another Immediate command.

Rate Override. This command immediately changes the % feedrate override value (frequently referred to as MFO value). This new value will become effective immediately when received by the Power Mate APM. It is stored and will remain effective until overwritten by a different value. A rate override has no effect on non–programmed motion. Feedrate is set to 100% whenever a program is initiated.

Position Increment Without Position Update. (user units) This command offsets the axis motion from –128 to +127 user units without updating the Actual or Commanded Position. The Power Mate APM will immediately move the axis by the increment commanded if the servo is enabled.

Move At Velocity. (user units/sec) This command is executed from the PLC to move the axis at a constant velocity. The configured *Jog Acceleration* rate is used for *Move at Velocity* commands. Axis position data will roll over at the configured Hi or Lo Limit when reached during these moves.

Set Position. (user units) This command changes the axis position register values without moving the axis. The *Commanded Position* and *Actual Position* values will both be changed so that no motion command will be generated. The *Actual Position* will be set to the value designated and the *Commanded Position* will be set to the value + *Position Error*. *Set Position* cannot be performed when the %I *Moving* bit or the %I *Program Active* bit is ON. The position value must be within the End of Travel Limits and Count Limits or a status error will be reported. *Position Valid* indication is set after a successful *Set Position* command.

Force D/A Output. This command forces the Velocity Command D/A outputs at the I/O connector to a constant output level. The Force D/A Output command sets the output to any value within the +/– 10 volt range. A command of +32,000 will produce an output of + 10 V and -32,000 will produce an output of -10 V.

In Digital mode, a *D/A Output* value of +4095 will produce a motor output of + 4095 RPM and -4095 will produce a motor output of -4095 RPM.

The *Enable Drive %Q* bit must be active with no other motion commanded for the Force D/A Output command to operate. Force D/A output is the only continuous %AQ immediate command. It must remain continuously in the %AQ data for proper operation. Thus any other %AQ immediate command will remove the Force D/A command.

For the single axis Power Mate J only, a *Force D/A Output* immediate command and a *D/A Output* value in the %AQ Words 3-5, will operate the analog output on I/O connector A. A *D/A Output* value of +32000 will produce +10.00 Vdc and a *D/A Output* value of -32000 will produce -10.00 Vdc.

Position Increment With Position Update. (user units) This command is similar to the Position Increment without position update command (#21h) except that Actual and Commanded Position (returned in %AI data) **are both updated** by the increment value. If the servo is enabled, the Power Mate APM will immediately move the axis by the increment value.

In Position zone. This command can be used to set the active *In Position* zone to a value different than the configured value.

The *In Position* zone is used by the Power Mate APM to determine when a PMOVE is complete and also when the axis motion (feedback position) is close enough to the commanded position to allow position critical operations (such as *Set Position*) to take place. The *In Zone* %I bit is set to indicate this.

If the Power Mate APM is power cycled or the PLC CPU is reset for any reason, the value set by this command will be lost and the *In Position* zone value set by configuration software will be reinstated.

Move Command. This command will produce a single move to the commanded position each time it is sent. The current Jog acceleration and velocity (which can also be changed by %AQ commands) will be used for the move.

The data field for this command may contain the **move position or distance** in bytes 2–5 with the command type (in hexadecimal format) as defined below:

Move Type (byte 1):

00h = Abs, Pmove, Linear
01h = Abs, Cmove, Linear
10h = Abs, Pmove, S
11h = Abs, Cmove, S
40h = Inc, Pmove, Linear
41h = Inc, Cmove, Linear
50h = Inc, Pmove, S
51h = Inc, Cmove, S

The data field for this command may contain a parameter number in byte 2 (bytes 3–5 unused) with the command type (in hexadecimal format) as defined below:

Move Type (byte 1):

80h = Abs, Pmove, Linear
81h = Abs, Cmove, Linear
90h = Abs, Pmove, S
91h = Abs, Cmove, S
C0h = Inc, Pmove, Linear
C1h = Inc, Cmove, Linear
D0h = Inc, Pmove, S
D1h = Inc, Cmove, S

The **Move** command is executed as a single move motion program, therefore, all the restrictions that apply to motion program execution will also apply to it. For example, if a program is already active for axis 1, then an attempt to send this command for axis 1 will result in an error condition reported.

Jog Velocity. (user units/sec) This command sets the velocity used when a *Jog* %Q bit is used to jog in the positive or negative direction.

Jog Acceleration. (user units/sec/sec) This command sets the acceleration value used by *Jog*, *Move at Velocity*, a *Home Cycle*, and *Abort All Moves*.

Position Loop Time Constant. (milliseconds) This command allows the servo position loop time constant to be changed from the configured value. The lower the value, the faster the system response. Values which are too low will cause system instability and oscillation. For accurate tracking of the commanded velocity profile, the *Position Loop Time Constant* should be 1/4 to 1/2 of the MINIMUM system deceleration time.

Velocity Feedforward. This command sets the *Velocity Feedforward* gain (percent). It is the percentage of Commanded Velocity that is added to the Power Mate APM velocity command output. Increasing *Velocity Feedforward* causes the servo to operate with faster response and reduced position error. Optimum *Velocity Feedforward* values are 80–90 %. **The "Vel at 10 V" value must be set correctly for proper operation of the *Velocity Feedforward* gain factor.**

Integrator Time Constant. (milliseconds) This command sets the *Integrator Time Constant* for the position error integrator. This value specifies the length of time in which 63% of the position error will be removed. The *Integrator Time Constant* should be 5 to 10 times greater than the *Position Loop Time Constant* to prevent instability and oscillation.

Velocity Loop Gain. (VLGN) The velocity control loop gain for an axis may be set with the *Velocity Loop Gain* command. The VLGN value is used to match the load inertia (J_L) to the motor inertia (J_M). VLGN is defined with a value of 16 representing an inertia ratio of 1 to 1. The VLGN value is calculated assuming that the load is rigidly applied to the motor. Therefore, in actual machine adjustment the set value may significantly differ from the calculated value due to rigidity, friction, backlash, and other factors. A PLC reset or power cycle returns this value to the configured data. The user must set the velocity loop gain (VLGN) such that it satisfies the following equation:

$$\mathrm{VLGN} = \frac{J_L}{J_M} * 16$$

Where:

JL = Load Inertia

JM = Motor Inertia

VLGN = Velocity Control Loop Gain (0 - 255)

For example: The rotor inertia (J_M) of a particular servo is 0.10 lb-in-s^2. The load inertial (J_L) in this application is 0.05 lb-in-s^2. VLGN = (0.05 / 0.10) * 16 = 8

The default *Velocity Loop Gain* is set using the *Integrator Time Constant* setting in the Configuration Software. Since the Integrator is unused by the controller module in Digital mode, the Integrator Time Constant data is used to set the initial *Velocity Loop Gain*.

Caution

Changing the VLGN value may cause an axis to be unstable and care should be used when making any change to the VLGN value.

Select Return Data. This command allows alternate data to be reported in the %AI Commanded Position location for each axis. The alternate data includes information such as Parameter Register contents, controller module's Firmware Revision and interface module's Firmware Revision.

The *Select Return Data* command uses a mode selection and an offset selection. The mode selection (byte offset +1 of the six byte command) determines the Return Data type. The offset selection (byte offsets +2, +3 of six byte command) selects an individual data item for some modes. Setting the mode to 00h causes the default Commanded Position to be reported. The following Return Data selections are allowed:

Selected Return Data	Data Mode	Data Offset
Commanded Position	00h	not used
Absolute Feedback Offset(cts)	07h	not used
Parameter Register	08h	Parameter Number (0–255)
APM Firmware Revision	10h	not used
DSI Firmware Revision	11h	not used

Absolute Feedback Offset is the position offset that is used to initialize Actual Position when Absolute Encoder is used. Actual Position = Absolute Encoder Data + Absolute Feedback Offset.

Controller and interface module Firmware Revisions should be interpreted as two separate words for major-minor revision codes. At least three PLC sweeps or 20 milliseconds (whichever represents more time) must elapse before the new Selected Return Data is available in the PLC.

Torque Limit (TRLMT). The *Torque Limit Command* provides a method of limiting the torque produced by a digital servo motor on each axis. The controller module will set the torque limit (TRLMT) at the default 100% whenever a power cycle or reset occurs. The PLC application logic must set any other value for desired TRLMT. The valid range for TRLMT is 1 to 100 % in units of 1%. TRLMT can be changed during axis motion. Refer to the appropriate servo motor manual for the actual motor torque curve to determine the actual value of torque output at a given velocity.

Configuration Modes. This configuration command permits various configuration changes to the controller module by modifying the byte in the command string associated with the desired feature. The values established by the *Configuration Mode* command are not saved through a power cycle or reset and must be re-established on system startup. Use of the *Set Configuration Complete* command is recommended, with appropriate application logic to enable the PLC verification of desired modes.

MCON CTRL has the following possible values:

- MCON CTRL = 00h = Digital Servo Magnetic Contactor (MCON) NOT controlled by CTL Output (Default).
- MCON CTRL = 01h = Digital Servo Magnetic Contactor (MCON) is controlled by CTL Output.

With the default mode of operation the %Q *Enable Drive* command On will energize the digital servo power relay (MCON) and enable servo power to the amplifier. The *Enable Drive %Q* Off will cause the MCON and digital servo power to be removed after the configured *Servo Drive Disable Delay (DisDly).*

With MCON CTRL (byte 4) set to 01h, the *%Q Enable* Drive ON / OFF command Enables and Disables servo power to the motor. The motor shaft will have no torque when the servo power is Off. The MCON relay will remain energized. This function is useful when the torque must be removed and reapplied repeatedly from the motor and will help avoid wear on the MCON relay.

With MCON CTL set to 01h, a %Q CTL output will control the ON / OFF state of the magnetic contactor (MCON).

CTL11 is the Axis 1 MCON control output. CTL12 is the Axis 2 MCON control output for the two axis Power Mate J.

Absolute Feedback Offset. This command updates the Absolute Feedback Offset (counts) used to initialize Actual Position from an AbsoluteDigital Encoder. To permanently save the encoder offset entered with this command, the *Update Flash Memory* %AQ command must also be used. Normally the Absolute Feedback Offset is calculated automatically by the *Set Position* or *Find Home* commands. Refer to Appendix D, *Serial Encoders,* for additional information.

Set Configuration Complete. This command sets the *Configuration Complete* %I bit. Once set, the *Configuration Complete* bit is only cleared when the PLC resets or reconfigures the Power Mate APM. The PLC can monitor the bit to determine if it must resend other %AQ commands, such as *In Position Zone* or *Jog Acceleration*. This would only be necessary if the %AQ commands were used to override Power Mate APM configuration data programmed with the PLC configuration software.

Update Flash Memory. This command copies important RAM data including Motion Programs and the latest Absolute Feedback Offset into Flash memory. Normally the *Update Flash Memory* command is sent after a Set Position or Find Home operation when an Absolute Digital Encoder is used. Updating the Flash memory permanently saves the Absolute Feedback Offset so that Actual Position can be restored correctly after a system loss of power. Refer to Appendix D, *Serial Encoders,* for additional information.

Load Parameter Immediate. This command is executed from the PLC to immediately change an Power Mate APM parameter value. Parameters are only used by motion programs. A command for each parameter change is required.

Chapter 5 Servo System Startup Procedures

Follow the procedures below to startup your servo system.

1. Connect the motor to the servo amplifier according to the manufacturer's recommendations.
2. Connect the Power Mate APM **Drive Enable** Relay and **Velocity Command** outputs to the servo amplifier. Connect the position feedback device (A Quad B Encoder or Linear Transducer) to the Power Mate APM inputs.

Note

> If these connections are incorrect or there is slippage in the coupling to the Feedback Device, an ***Out of Sync*** error condition can occur when motion is commanded.

3. If **Overtravel Limit** switches are used (24 Vdc), wire them to the correct Power Mate APM inputs. Otherwise the overtravel limits must be disabled using Configuration Software. If a **Home** switch is used (24 Vdc), wire it to the correct Power Mate APM inputs. The **Home** switch must be wired so that it is ALWAYS ON (closed) when the axis is on the negative side of home and ALWAYS OFF (open) when the axis is on the positive side of home.
4. Use the Configuration Software to set the desired user scaling factors and other configurable parameters. Store the configuration to the PLC.
5. Clear the program from the PLC, turn off all Power Mate APM %Q bits and place the PLC in RUN mode. Monitor the %I bits for **CTL03, 4 (Home)**, **CTL05, 7 (+ Overtravel)** and **CTL06, 8 (– Overtravel)** and confirm that each bit responds to the correct switch.
6. Turn on the %Q *Enable Drive* bit and place the command code for *Force D/A Output* 0 in the %AQ table. Confirm that the servo amplifier is enabled. If the motor moves, adjust the amplifier zero offset until the motor stops moving. Note: The %Q *Enable Drive* bit must be maintained on in order for the *Force D/A Output* command to function.
7. Send the command code for *Force D/A Output* +3200 (+1.0v). Confirm that the motor moves in the desired POSITIVE direction and the *Actual Velocity* reported in

the Power Mate APM %AI table is POSITIVE. If the motor moves in the wrong direction, consult the manufacturers instructions for corrective action. The Motor Dir parameter in the Configuration Software can also be used to swap the positive and negative axis directions (Power Mate APM firmware 2.10 or later). If the motor moves in the POSITIVE direction but the Power Mate APM reports that *Actual Velocity* is NEGATIVE, then the encoder channel A and channel B inputs must be swapped.

8. Record the actual motor velocity reported by the Power Mate APM with a 1.0 volt velocity command. Multiply this velocity by 10 and update the *Vel @ 10V* entry in the Power Mate APM configuration. Initially set the *Pos Loop TC* configuration entry to a high value (typically 100 to 1000 ms).

9. Turn on the %Q *Jog Plus* bit. Confirm that the servo moves in the proper direction and that the *Actual Velocity* reported by the Power Mate APM in the %AI table matches the configured jog velocity.

10. With the *Drive Enabled* %Q bit on and no servo motion commanded, adjust the servo drive zero offset for zero position error (Power Mate APM *Commanded Position* = Power Mate APM *Actual Position*). The Integrator should be disabled during this process.

11. Check for proper operation of the *Find Home* cycle by momentarily turning on the %Q *Find Home* bit (the *Drive Enabled* %Q bit must also be maintained on). The axis should move towards the Home Switch at the configured *Find Home* velocity, then seek the Encoder Marker at the configured *Fnl Home* velocity. If necessary, adjust the configured velocities and the location of the **Home Switch** for consistent operation. The final **Home Switch** transition MUST occur at least 10 ms before the Encoder Marker Pulse is encountered. The physical location of *Home Position* can be adjusted by changing the *Home Offset* value with the Configuration Software.

12. Monitor servo performance and use the %Q *Jog Plus* and *Jog Minus* bits to move the servo motor in each direction. The *Pos Loop Time Constant* can be temporarily modified by placing the correct command code in the %AQ table. For most systems the *Position Loop Time Constant* can be reduced until some servo instability is noted, then increased to a value approximately 50% higher. Once the correct time constant is determined, the Power Mate APM configuration should be updated using the Configuration Software. *Velocity Feedforward* can also be set to a non zero value (typically 80 – 90 %) for optimum servo response.

Note

For proper servo operation, the Configuration entry for *Vel at 10v* MUST be set to the actual servo velocity (in User Units / sec) caused by a 10 V command.

System Troubleshooting Hints

1. The Series 90-30 Power Mate APM requires PLC firmware release 3.52 or greater and LM90 software release 4.0 or greater.

2. The default Power Mate APM configuration for the Overtravel inputs is ENABLED. Therefore 24 vdc must be applied to the Overtravel inputs or the Power Mate APM

will not operate. If Overtravel inputs are not used, the Power Mate APM configuration should be set to Overtravel inputs DISABLED.

3. The ENABLE DRIVE %Q control bit must be set continuously to **1** or no motion other than Jogs will be allowed. If no STOP errors have occurred, the DRIVE ENABLED %I status bit will mirror the state of the ENABLE DRIVE %Q bit. A STOP error will turn off DRIVE ENABLED even though ENABLE DRIVE is still a **1**. The error condition must be corrected and the CLEAR ERROR %Q control bit turned on for one PLC sweep to re-enable the drive.

4. If the ERROR %I status bit is **1** and the AXIS ENABLED and DRIVE ENABLED %I status bits are **0**, then a STOP error has occurred (Status LED flashing fast). In this state, the Power Mate APM will not respond to any commands other than the CLEAR ERROR %Q control bit.

   The CLEAR ERROR %Q control bit uses one-shot action. Each time an error is generated, the bit must be set to **0** then set to **1** for at least one PLC sweep to clear the error.

5. The *CFG OK* LED must be ON or the Power Mate APM will not respond to PLC commands. If the LED is OFF then a valid Power Mate APM configuration has not been received from the PLC.

Chapter 6 Power Mate APM Motion Control

This chapter provides practical information on Power Mate APM Motion Control with a number of examples. The main topics discussed are:

- Position Feedback Types
- Non-Programmed Motion
- Programmed Motion

Position Feedback Types

Power Mate APM firmware revision 2.50 or higher supports three types of feedback devices:

- A Quad B Incremental Encoder
- Linear Absolute Transducer
- Digital Serial Encoder (requires Digital Servo Interface module)

A Quad B Incremental Encoder

A Quad B Encoders provide three output signals to the Power Mate APM: Quad A, Quad B, and Marker. The Quad A and Quad B signals transition as the encoder turns, allowing the Power Mate APM to count the number of signal transitions and calculate the latest encoder position change and direction of rotation.

A Quad B Encoders are incremental feedback devices; they do not provide a continuous indication of absolute shaft angle as the input shaft rotates. For this reason, the Power Mate APM Actual Position must be initialized with a known physical position before positioning control is allowed. This position alignment can be accomplished using the *Set Position* %AQ Immediate command (described in Chapter 4) or the *Home Cycle* (described later in this chapter under *Non-Programmed Motion*). The *Home Cycle* makes use of the encoder Marker channel, which is a once per revolution pulse produced at a known encoder shaft angle. Successful completion of the *Home Cycle* or a *Set Position* command causes the Power Mate APM to set the axis *Position Valid* %I bit. *Position Valid* must be set before motion programs will be allowed to execute. *Position Valid* is only cleared by an encoder Quadrature Error (Quad A and Quad B switching at the same time) or by turning on the *Find Home* and *Abort* %Q bits simultaneously.

Linear Absolute Transducer

Linear Absolute transducers for use with the Power Mate APM provide an output signal that has constant frequency and variable on time. The on time of the signal represents the absolute position reported by the transducer. The Power Mate APM continuously measures the on time to update the %AI Actual Position data. Because the transducer provides an absolute position, the *Position Valid* %I bit is automatically set by the Power Mate APM as soon as a valid signal is received from the transducer. If the transducer signal is lost for more than 20 milliseconds, the Power Mate APM will report a *Loss of Feedback* error and stop all motion until the error is cleared and the transducer signal is restored.

When the Power Mate APM is configured for Linear Feedback, the *Home Cycle* is disabled and the %Q *Find Home* command bit has no effect. After receiving a valid transducer signal, the Power Mate APM attempts to set *Actual Position* to the (transducer reading + configured Position Offset). If this value is outside the configured Hi / Lo Count Limit range, an error is reported, *Position Valid* stays cleared and *Actual Position* is set to zero. If this value is within the Hi / Lo Count Limit range, *Actual Position* is initialized and *Position Valid* is set. *Actual Position* can be shifted by changing the Position Offset parameter with the Configuration software. Once *Actual Position* is initialized, the Power Mate APM continuously reads the linear transducer and updates Actual Position with any position changes.

The *Set Position* command can also be used to set *Actual Position* to a user specified value. For additional details on specification and configuration of linear transducers, see Appendix F: *LDT Interface*

Digital Serial Encoder

The initial release of the Power Mate J supports only digital serial encoders used with the a Series or b Series digital servo. Encoder resolution is fixed at 8192 encoder counts per motor shaft revolution. Incremental or Absolute (battery backed) encoders may be configured. Refer to Appendix G for additional information about absolute encoder usage.

Non-Programmed Motion

The Power Mate APM can generate motion in an axis in one of five ways without the use of any motion programs.

- *Find Home* and *Jog* use the %Q bits to command motion.
- *Move at Velocity*, *Force D/A Output*, and *Position Increment* use %AQ immediate commands.

During *Jog*, *Find Home*, *Move at Velocity*, and *Force D/A Outpu*t, any other commanded motion, programmed or non-programmed, will generate an error. The only exception is *Position Increment* which may be commanded any time. See the description of *Position Increment* motion for more details.

Non-programmed motions (Abort, Jog, Move at Velocity) use the JOG acceleration and acceleration mode. FEEDHOLD uses the programmed acceleration and acceleration mode.

Power Mate APM Home Cycle

A Home Cycle establishes the Home Position for systems with an incremental feedback device that also provides a marker pulse. The configured Home Offset defines the location of Home Position as the offset distance from the Home Marker.

The *Enable Drive %Q bit and Drive Enabled %I* bit must be ON during an entire Home Cycle. However, the *Find Home %Q* bit does not need to be held ON during the cycle; it may be one-shot. Note that turning ON the *Find Home* bit immediately turns OFF the *Position Valid %I* bit until the end of the Home Cycle. The *Abort All Moves %Q* bit halts a Home Cycle, but the *Position Valid* bit does not turn back ON. No motion programs may be executed unless the *Position Valid* bit is ON.

Home Switch Mode

If the Find Home Mode is configured as Home Switch, the **Home Switch** input from the I/O connector is used to roughly indicate the reference position for Home. The next marker encountered when traveling in the negative direction indicates the exact location. An open **Home Switch** indicates the servo is on the positive side of the **Home Switch**. An OFF to ON transition of the *Find Home* command yields the following Home Cycle. Unless otherwise specified, acceleration is at the current *Jog Acceleration* and configured *Jog Acceleration Mode.* (If initiated from a position on the positive side of the **Home Switch**, the cycle starts with step 1, otherwise the cycle starts with step 3)

1. The axis is moved in the negative direction at the configured *Find Home Velocity* until the **Home Switch** closes.
2. The axis decelerates and is stopped.
3. The axis is accelerated in the positive direction and moved at the configured *Find Home Velocity* until the **Home Switch** opens.
4. The axis decelerates and is stopped.
5. The axis is accelerated in the negative direction and moved at the configured *Find Home Velocity* until the **Home Switch** closes.
6. The Power Mate APM continues negative motion at the configured *Final Home Velocity* until a marker pulse is sensed. The marker establishes the Home reference position.
7. The axis decelerates and is stopped.
8. The axis is moved, at the current *Jog Velocity*, the number of user units specified by the *Home Offset* value from the Home reference position.
9. The axis decelerates and is stopped.
10. An internal *Set Position* sets the *Commanded* and *Actual* Positions to the configured *Home Position* value. Finally, the Power Mate APM sets the *Position Valid* %I bit to indicate the Home Cycle is complete.

Move+ and Move– Modes

If Find Home Mode is configured as Move+ or Move–, the first marker pulse encountered when moving in the appropriate direction (positive for Move+, negative for Move–) is used to establish the exact location. An OFF to ON transition of the Find Home %Q bit will perform the following operation.

1. The axis is accelerated at the *Jog Acceleration rate* and moved at the configured *Final Home Velocity* (positive direction for Move+, negative direction for Move–) until a marker pulse is sensed. This marker pulse establishes the Home reference position.
2. The axis is stopped at the configured *Jog Acceleration* rate and with the configured *Jog Acceleration Mode.*
3. The axis is moved, at the configured *Jog Velocity* and with the configured *Jog Acceleration* rate and *Jog Acceleration Mode*, the number of user units specified by the *Home Offset* value from the Home reference position.
4. The axis is stopped at the configured *Jog Acceleration* rate and with the configured *Jog Acceleration Mode.*

5. An internal *Set Position* sets the *Commanded* and *Actual* Positions to the configured *Home Position* value; the Power Mate APM sets the *Position Valid* %I bit to indicate the Home Cycle is complete.

Jogging with the Power Mate APM

The *Jog Velocity*, *Jog Acceleration*, and *Jog Acceleration Mode* are configurable in the Power Mate APM. These values are used whenever a *Jog Plus* or *Jog Minus* %Q bit is turned ON. Note that both bits ON generates no motion. The *Jog Acceleration* and *Jog Acceleration Mode* are also used during a *Find Home* Cycle and when a *Move at Velocity* immediate command is performed. Programmed motions use the *Jog Velocity* and *Jog Acceleration* as defaults.

A *Jog* can be performed when no other motion is commanded, or while programmed motion is temporarily halted due to a Feedhold command. The *Enable Drive* %Q bit does not need to be ON to Jog, but it may be. Turning on a *Jog* %Q bit will automatically close the Enable Relay, and turn on the *Drive Enabled* %I bit. When an **Overtravel Limit Switch** is OFF, *Jog* and *Clear Error* may be turned on simultaneously to move away from the open Limit Switch. Thus a *Jog Plus* will not work while the **Positive End of Travel Switch** is open and *Jog Minus* will not work while the **Negative End of Travel Switch** is open.

Move at Velocity Command

A *Move at Velocity* command is generated by placing the value 22h in the first word of %AQ data assigned to an axis. The second and third words together represent a signed 32 bit velocity. Note that the third word is the most significant word of the velocity. Once the command is given, the %AQ data may be cleared by sending a NULL command, or changed as desired. *Move at Velocity* will not function unless the servo drive is enabled (%Q *Enable Drive* and %I *Drive Enabled* are set).

The listing of %AQ immediate commands shows the words in reverse order to make understanding easier. For example, to command a velocity of 512 user units per second in a Power Mate APM configured with %AQ data starting at %AQ1, the following values should be used: 0022h (34 decimal) in %AQ1, 0200h (512 decimal) in %AQ2, and 0 in %AQ3. When the Power Mate APM receives these values, if *Drive Enabled %I* is ON, *Abort All Moves %Q* is OFF, and no other motion is commanded it will begin moving the axis at 512 user units per second in the positive direction using the current Jog Acceleration and Acceleration Mode.

The *Drive Enabled* %I bit must be ON before the Power Mate APM receives the immediate command or an error will occur. Also, if a *Move at Velocity* command is already in the %AQ data, the velocity value must change while the *Drive Enabled* bit is ON for the Power Mate APM to accept it. The Power Mate APM detects a *Move at Velocity* command when the %AQ values change.

When the Power Mate APM is performing a *Move at Velocity*, it ignores the *Software End of Travel Limits*. Hardware Overtravel Limits must be ON if they are enabled.

A *Move at Velocity* can be stopped without causing an error in two ways: a *Move at Velocity* command with a velocity of zero, or turning the *Abort All Moves* %Q bit ON for at least one PLC sweep.

Force D/A Command

To obtain a specific output voltage from the Power Mate APM to the servo, use the *Force D/A* immediate command. Command values between –32,000 and +32,000 are scaled to the range –10V to +10V. Thus +3,200 is +1V. The *Drive Enabled* %I bit must be on for the *Force D/A* command to work. An exception is the second axis of a one axis Power Mate APM. Since there is no axis 2 *Enable Drive* bit on a one axis Power Mate APM, placing the Force D / A command code in the second axis %AQ data automatically enables the servo drive.

The *Force D/A* command is the only command that must be maintained in the %AQ data for proper operation. If any other immediate command is sent to the Power Mate APM, the *Force D/A* operation will end. A one-shot Force D / A command will operate only during the sweep in which it appears.

Position Increment Commands

To generate small corrections between the axis position and the Power Mate APM tracking, the *Position Increment* commands can be used to offset Actual Position by a specific number of user units. If the *Drive Enabled %I* bit ON, the axis will immediately move the increment amount. If the position increment without position update is used (%AQ command 21h), the *Actual Position* reported by the Power Mate APM will remain unchanged. If the position increment with position update is used (%AQ command 25h), the *Actual Position* and *Commanded Position* reported by the Power Mate APM will be changed by the increment. *Position Increment* may be used at any time, though simultaneous use with the *Force D/A* command is impossible because the *Force D/A* command must continuously appear in the %AQ data.

Other Considerations

Other considerations when using non-programmed motion are as follows:

- An ON *Abort All Moves* bit will prevent any non-programmed motion from starting.
- Turning ON the *Abort All Moves* bit will immediately stop any current non-programmed motion at the current *Jog Acceleration*.
- A *Set Position* command during non-programmed motion will cause a status error.
- Turning OFF the *Enable Drive* bit while performing a Home Cycle or *Move at Velocity* will cause a stop error.
- The *Feedhold* bit has no effect on non-programmed motion.
- The *Rate Override* command has no effect on non-programmed motion.
- Changing the *Jog Velocity* or *Jog Acceleration* will not affect non-programmed moves in progress.

Programmed Motion

The Power Mate APM executes program motion commands sequentially in a block-by-block fashion once a program is selected. The program commands can be categorized as follows:

Type 1 Commands

- Call Subroutine
- Jump

Type 2 Commands

- Block #
- Null
- Acceleration
- Velocity

Type 3 Commands

- Positioning Move
- Continuous Move
- Dwell
- Wait
- End of Program

Type 1 commands can redirect the program path execution, but do not directly affect positioning. Call executes a subroutine before returning execution to the next command. Jump either continues execution at another location, or it tests CTL bits and, based on the bit condition, may or may not alter the program path.

Type 2 commands also do not affect position. The Block # command provides an identification or label for the following Type 3 command. If no Block # is found in the current program block, the previous Block # is used. The Velocity and Acceleration commands specify velocity and acceleration rates for motion.

Type 3 commands start or stop motion and thus affect positioning control. Positioning and Continuous moves command motion; Dwell, Wait, and End of Program stop motion.

A program block consists of one (and only one) Type 3 command with any number and combination of preceding Type 1 and 2 commands. Type 2 commands are optional; a program block could contain a single Type 3 command. Type 2 commands, and Conditional Jumps, do not take effect until the following Type 3 command is executed.

While the Power Mate APM is executing one program block, the following program block is processed into a buffer command area to minimize the transition time from one block to another. Thus parameters used in a move must be loaded before the move two blocks previous completes execution.

When a 2-axis Power Mate APM is executing a 2-axis program, the program commands are scanned independently by each axis and only the data designated for that axis is executed. Note that some commands do not specify an axis (Block #, Jump, Call, and End) and therefore will apply to both axes.

A 2-axis program can contain Sync Block # commands to synchronize the 2 motions at designated points. When the first axis reaches the Sync Block, it will not execute the block until the other axis has also reached the Sync Block. Refer to Example 18 for an illustration of this type of operation.

Several aspects of programmed motion are discussed below.

Prerequisites for Programmed Motion

The following conditions must be satisfied before a motion program can be initiated (for a multi-axis program, the conditions must be met for BOTH axis 1 and axis 2):

- The Enable Drive %Q bit must be ON
- The *Drive Enabled* %I bit must be ON
- The *Position Valid* %I bit must be ON
- The *Moving* %I bit must be OFF
- The *Program Active* %I bit must be OFF
- The *Abort All Moves* %Q bit must be OFF
- The axis position must be within the configured End of Travel Limits
- The Overtravel Limit Switches must be asserted if enabled
- A Force D / A command must not be active
- The program to be executed must be a valid program stored in the Power Mate APM

Conditions Which Stop a Motion Program

A motion program will immediately cease when one of the following conditions occurs:

- The *Abort All Moves* %Q bit turns ON
- The *Enable Drive %Q* bit turns OFF
- An **Overtravel Limit Switch** turns OFF when OT Limits are configured to be enabled.
- The next programmed move, either PMOVE or CMOVE, will pass a *Software End of Travel Limit*
- A Stop Error occurs. See Appendix A (Error Codes)

Parameters for Programmed Moves

Programmed moves have three parameters:

1. The *distance* to move or *position* to move to,
2. The *type of positioning reference* to use for the move, and
3. The *type of acceleration* to use while performing the move.

Types of Positioning Reference

For the type of reference to use for the move, the choices are ABSOLUTE and INCREMENTAL. This reference determines how the first parameter, the distance to move or position to move to, is interpreted.

Absolute Positioning

In an absolute positioning move, the first parameter is the position to move to. The following is an example of an absolute positioning move.

```
PMOVE 5000, ABS, LINEAR
```

This move will move the axis from its current position, whatever it may be, to the position 5000. Thus the actual distance moved depends upon where the axis is when the move is encountered. If the initial position was 0, the axis would move 5000 user units in the positive direction. If the initial position was 10000, the axis would move 5000 user units in the negative direction. And, if the initial position was 5000, no motion would be generated.

Incremental Positioning

In an incremental move, the first parameter is interpreted as the distance to move from the position where the move begins. The Power Mate APM translates incremental move distances into absolute move positions so no error accumulates. The following is an example of an incremental positioning move.

```
PMOVE 5000, INC, LINEAR
```

This incremental move will move the axis from its current position to a position 5000 user units greater. With an incremental move, the first parameter specifies the actual number of user units the axis moves.

Types of Acceleration

Choices for the last parameter, which specifies the type of acceleration to use while performing the move, are LINEAR and SCURVE.

Linear Acceleration

A sample profile of a linear move plotting velocity versus time is shown in Figure 6–1. The straight lines on the graph show that a linear move uses constant acceleration. The area under the graph is the distance moved.

```
ACCEL     1000
VELOC     2000
PMOVE     6000, INC, LINEAR
```

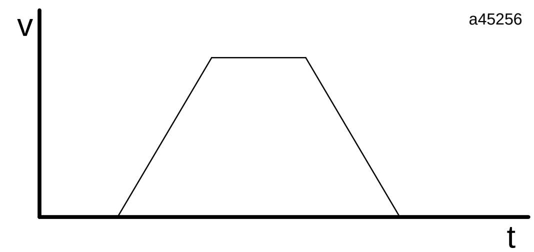


Figure 6-1. Sample Linear Motion

S-Curve Acceleration

An s-curve motion sample, again plotting velocity versus time, is shown below. The curved lines on the graph indicate that the acceleration was not constant. When the move begins, the acceleration starts slowly and builds until it reaches the programmed acceleration. This should be the midpoint of the acceleration. Then, the acceleration begins decreasing until it is zero, at which time the programmed velocity has been reached. An s-curve move requires twice the time and distance to accelerate and decelerate that a linear move needs if the acceleration is the same. The area under the velocity vs time graph is also the distance moved.

```
ACCEL      2000
VELOC      2000
PMOVE      8000, INC, SCURVE
```

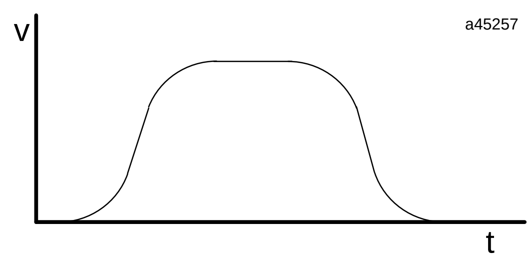


Figure 6-2. Sample S-curve Motion

Types of Programmed Move Commands

Positioning Move (PMOVE)

A PMOVE uses the most recently programmed velocity and acceleration. If a VELOC command has not been encountered in the motion program, the *Jog Velocity* is used as a default. If an ACCEL command has not been encountered in the motion program, the *Jog Acceleration* is used as a default.

A PMOVE will always stop when it is completed to allow the *IN ZONE* %I bit to turn ON.

Continuous Move (CMOVE)

A CMOVE uses the most recently programmed velocity and acceleration. If a VELOC command has not been encountered in the motion program, the *Jog Velocity* is used as a default. If an ACCEL command has not been encountered in the motion program, the *Jog Acceleration* is used as a default.

A CMOVE does not stop when completed unless it is followed by a DWELL or a WAIT, the next programmed velocity is zero, or it is the last program command. It does not wait for the position to be IN ZONE before going to the next move. A CMOVE reaches its programmed position at the same time it reaches the velocity of the following Move command.

A special form of the CMOVE command can be used to force the Power Mate APM to reach the programmed CMOVE position *before* starting the velocity change associated with the next move command (that is, execute the entire CMOVE command at a constant velocity). Programming an incremental CMOVE command with an operand of 0 (CMOVE INC 0) will force a delay in the servo velocity change for the next move command in sequence. The following sequence of commands illustrates this effect:

Command	**Data**	**Comments**
VELOC	10000	Set velocity of first move = 10000
CMOVE	15000, ABS, LINEAR	Reach velocity of second move at position = 15000
VELOC	20000	Set velocity of second move = 20000
CMOVE	0, INC, LINEAR	Force next velocity change to wait for next move command
CMOVE	30000, ABS, LINEAR	Stay at velocity = 20000 until position = 30000, then change to velocity = 5000
VELOC	5000	Set velocity of third move = 5000
PMOVE	45000, ABS, LINEAR	Final stop position = 45000

Programmed Moves

By combining CMOVEs and PMOVES, absolute and incremental moves, and linear and scurve motion, virtually any motion profile can be generated. The following examples show some simple motion profiles, as well as some cases of incorrect motion programming.

Example 1: Combining PMOVEs and CMOVEs

This example shows how simple PMOVEs and CMOVEs combine to form motion profiles.

```
ACCEL     1000
VELOC     2000
PMOVE     5000, ABS, LINEAR
VELOC     1200
PMOVE     10000, ABS, SCURVE
ACCEL     1500
VELOC     2800
CMOVE     6000, INC, LINEAR
VELOC     1200
CMOVE     23000, ABS, SCURVE
ACCEL     1000
VELOC     2800
PMOVE     5000, INC, LINEAR
```

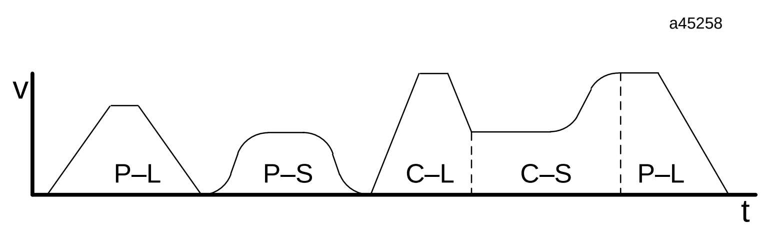


Figure 6-3. Combining PMOVEs and CMOVEs

The move types are indicated under the corresponding move; for example, P-L indicates linear PMOVE.

The first PMOVE accelerates to programmed velocity, moves for a distance, and decelerates to a stop. This is because motion stops after all PMOVEs. When the first move stops, it is at the programmed distance.

The second move is an scurve PMOVE. It, like the first, accelerates to the programmed velocity, moves for a time, and decelerates to zero velocity because it is a PMOVE.

The next move is a linear CMOVE. It accelerates to programmed velocity, moves for a time, and then decelerates to a lower velocity using linear acceleration. When a CMOVE ends, it will be at the programmed position of the move just completed, and at the velocity of the next move. Thus when the fourth move begins, it is already at its programmed velocity.

The fourth move is a CMOVE, so as it approaches its final position, it accelerates to be at the velocity of the fifth move when it completes. The graph shows the acceleration of the fourth move is scurve.

Finally, the fifth move begins and moves at its programmed velocity for a time until it decelerates to zero. Any subsequent moves after the fifth would begin at zero velocity because the fifth move is a PMOVE.

Example 2: Changing the Acceleration Mode During a Profile

The following example shows how a different acceleration, and even acceleration mode, can be used during a profile using CMOVEs. The first CMOVE accelerates linearly to the programmed velocity. Because the second CMOVE's velocity is identical to the first, the first CMOVE finishes its move without changing velocity. The acceleration of the second move is scurve as it decelerates to zero velocity.

```
ACCEL     2000
VELOC     6000
CMOVE     13000, ABS, LINEAR
ACCEL     4000
CMOVE     15000, INC, SCURVE
```

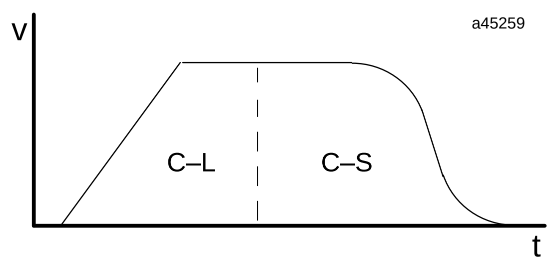


Figure 6-4. Changing the Acceleration Mode During a Profile

Example 3: Not Enough Distance to Reach Programmed Velocity

CMOVES and PMOVES can be programmed which do not have enough distance to reach the programmed velocity. The following graph shows a CMOVE which could not reach the programmed velocity. The Power Mate APM accelerates to the point where it must start decelerating to reach the programmed position of C1 at the velocity of the second CMOVE.

```
ACCEL       2000
VELOC       8000
CMOVE       7000, INC, LINEAR
ACCEL       10000
VELOC       2000
CMOVE       4400, INC, LINEAR
```

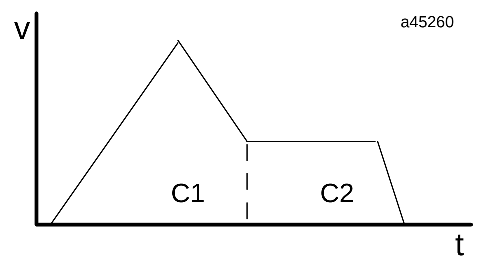


Figure 6-5. Not Enough Distance to Reach Programmed Velocity

Example 4: "Hanging" the Power Mate APM When the Distance Runs Out

A serious programming error involves "hanging" the Power Mate APM at a high velocity when the distance runs out. In the following example, the first CMOVE accelerates to a high velocity. The second CMOVE has an identical velocity. However, the distance specified for the second CMOVE is very short. Thus the axis is running at a very high velocity and must stop in a short distance. If the programmed acceleration is not large enough, the following profile could occur. In order to not pass the final position, the Power Mate APM instantly commands a zero velocity. This rapid velocity change is undesirable and could cause damage to the controlled machine.

```
ACCEL     500
VELOC     3000
CMOVE     9000, ABS, LINEAR
ACCEL     600
CMOVE     4800, INC, LINEAR
```

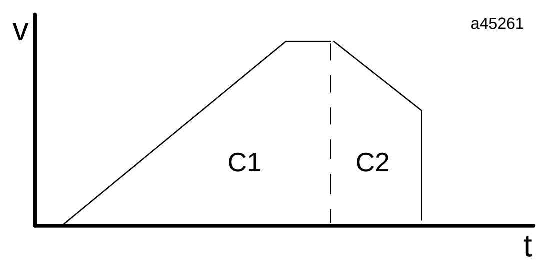


Figure 6-6. "Hanging" the Power Mate APM When the Distance Runs Out

Dwell Command

A DWELL command is used to generate no motion for a specified number of milliseconds. A DWELL after a CMOVE will make the CMOVE perform similar to a PMOVE, even if the specified dwell duration is zero.

A DWELL command using a parameter to set the dwell time will be treated as a NULL command if the parameter value is 65000. This feature allows a DWELL-P command between a CMOVE and another Move to be skipped if the DWELL-P value is 65000 (The CMOVE continues to the Move following the DWELL without stopping).

Example 5: Dwell

A simple motion profile, which moved to a specific point, waited, and returned to the original point could use the following program and would have the following velocity profile.

```
ACCEL     30000
VELOC     15000
PMOVE     120000, ABS, LINEAR
DWELL     4000
PMOVE     0, ABS, LINEAR
```

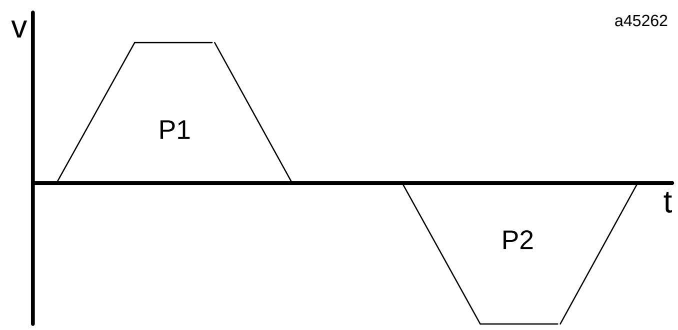


Figure 6-7. Dwell

Wait Command

The WAIT command is similar to the DWELL command; instead of generating no motion for a specified period of time, a WAIT stops program motion and monitors a CTL bit until it is ON. Thus motion will stop any time a WAIT is encountered, even if the CTL bit is on before the WAIT is reached in the program. The trigger to continue the program may be any of the twelve CTL bits.

If, in the previous example WAIT was substituted for DWELL, the motion profile would be the same except the second PMOVE would not start until the CTL bit turned ON. If the CTL bit was ON when the program reached the WAIT, the second PMOVE would begin immediately when the first PMOVE finished.

Also, if WAIT was used instead of DWELL in the previous example, CMOVEs and PMOVEs would generate similar velocity profiles. The WAIT will stop motion whether the previous move is a CMOVE or PMOVE.

Subroutines

The Power Mate APM can store up to ten separate programs and forty subroutines. Subroutines can be defined as two types: *single axis* and *multi axis*. Commands within single axis subroutines do not contain an axis number; this allows single axis subroutines to be called from any single axis program written for either axis 1 or axis 2. Commands within multi axis subroutines contain axis numbers just like commands within multi axis programs. Multi axis subroutines can only be called from multi axis programs or subroutines. Single axis subroutines can only be called from single axis programs or subroutines. On a two axis Power Mate APM, a single axis program for axis 1 and a single axis program for axis 2 may call the same single axis subroutine simultaneously.

The CALL command is used to execute a subroutine, with the subroutine number specified as an argument. Program execution continues at the start of the subroutine and resumes at the command after the CALL command when the subroutine finishes. Subroutines can be called from another subroutine, but once a subroutine has been called, it must be completed before it can be called again for the same axis. Thus recursion is not allowed.

Block Numbers and Jumps

Block numbers are used as reference points within a motion program and to control jump testing. A %AI data word displays the current block number which can be monitored to insure correct program execution or to determine when events should occur. A block number may also be the destination of a JUMP command.

Unconditional Jumps

Jumps are divided into unconditional and conditional. An unconditional jump command simply tells the Power Mate APM to continue program execution at the destination block number. An example of an unconditional jump follows.

Example 6: Unconditional Jump

The program executes a PMOVE, dwells for 2 seconds, and then unconditionally jumps back to the beginning of the program at block 1. Thus the PMOVE will be repeated until an *End of Travel Limit* or **Overtravel Limit Switch** is reached.

```
ACCEL    10000
VELOC    30000
BLOCK    1
PMOVE    200000, INC, LINEAR
DWELL    2000
JUMP     UNCOND, 1
```

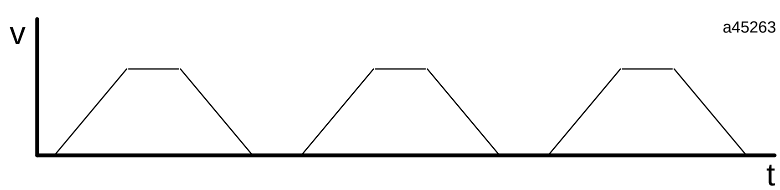


Figure 6-8. Unconditional Jump

Conditional Jumps

A conditional jump is a JUMP command with a CTL bit specified in the command. Conditional jumps are Type 1 commands in that they affect program path execution, but they are also similar to Type 2 commands because they do not take effect until a Type 3 command following the JUMP command is executed. When a conditional JUMP command is executed, the Power Mate APM examines the specified CTL bit. If the bit is ON, program execution jumps to the destination BLOCK #; if the bit is OFF, the program continues executing the command after the JUMP. Note that the Type 3 command after the conditional jump and at the jump destination will affect jump behavior.

Conditional JUMP commands should not be used with multi–axis programs containing sync blocks unless the Jump is triggered while both axes are testing the same JUMP command.

Conditional Jump testing **starts** when the next PMOVE, CMOVE, DWELL, WAIT or END Program command following a Conditional JUMP becomes active.

When Conditional Jump testing is active, the designated CTL bit is tested once every millisecond for the 1 axis Power Mate APM and once every 2 milliseconds for the 2 axis Power Mate APM.

Conditional Jump testing **ends** when the designated CTL bit turns ON (Jump Trigger occurs) or when a new Block Number becomes active.

If more than one Conditional JUMP is programmed without an intervening PMOVE, CMOVE, DWELL, WAIT or END Program command, only the last Conditional JUMP will be recognized.

Motion Program Example:

```
Begin-Program
1  JUMP     CTL01, 2                This JUMP command will be ignored
   JUMP     CTL02, 3                This JUMP command will be recognized
   CMOVE 1, +40000,    INC, LINEAR
2  CMOVE 1, +20000,    INC, LINEAR
3  PMOVE 1, +100000,   ABS, LINEAR
4  DWELL 1, 100
End-Program
```

When a new Block Number becomes active AFTER a Conditional JUMP command, Jump testing occurs one more time.

Motion Program Example:

```
Begin-Program
1  CMOVE 1, +20000,    ABS, LINEAR
   JUMP     CTL01, 3
2  PMOVE 1, +40000,    ABS, LINEAR              CTL01 tested only once
3  DWELL 1, 100
End-Program
```

In the example above, The CTL01 bit will only be tested once because the PMOVE following the JUMP contains a new Block Number (2).

Changing the location of Block Number 2 will cause the CTL bit to be tested throughout the PMOVE following the JUMP:

```
Begin-Program
1  CMOVE 1, +20000,   ABS, LINEAR
2  JUMP     CTL01, 3
   PMOVE 1, +40000,   ABS, LINEAR                CTL01 tested throughout PMOVE
3  DWELL 1, 100
End-Program
```

The Power Mate APM can perform a Conditional JUMP from an active CMOVE to a program block containing a CMOVE or PMOVE without stopping. **For the axis to jump without stopping, the distance represented by the CMOVE or PMOVE in the Jump block must be greater than the servo stopping distance.** The servo stopping distance is computed using the present commanded velocity and the acceleration parameters that would be in effect when the jump block became active.

The axis will STOP before jumping if a Conditional Jump trigger occurs under any of the following conditions:

- When a PMOVE is active.
- When a CMOVE is active and the Jump destination block contains a CMOVE or PMOVE representing motion in the opposite direction.
- When a CMOVE is active and the Jump destination block contains a CMOVE or PMOVE representing motion in the same direction with insufficient distance for the axis to stop.
- When a CMOVE is active and the Jump destination block contains a DWELL, WAIT or END (program) command.

If the axis does STOP before a Conditional Jump, the JOG acceleration and acceleration mode will be used.

Unconditional Jumps do not force the axis to stop before jumping to a new program block. For example, a CMOVE followed by a JUMP Unconditional to another CMOVE will behave just as if the two CMOVEs occurred without an intervening Unconditional JUMP.

If Conditional Jump testing is active when the Power Mate APM command processor encounters a CALL SUBROUTINE command, the axis will **stop** and terminate jump testing before the CALL is executed.

If Conditional Jump testing is active when the Power Mate APM command processor encounters an END SUBROUTINE command, the axis will **stop** and terminate jump testing before the END SUBROUTINE is executed.

Jump Testing

Conditional jumps perform jump testing. If the CTL bit is ON, the jump is immediately performed. If the CTL bit is OFF, the Power Mate APM watches the CTL bit and keeps track of the JUMP destination. This monitoring of the CTL bit is called jump testing. If during jump testing the CTL bit turns ON before a BLOCK command, another JUMP command, or a CALL command is encountered, the jump is performed. These commands will end jump testing.

Example 7: Jump Testing

Consider the following two program sections. In the program on the left, the move to position 2000 is completed before jump testing begins. The BLOCK command immediately after the JUMP command ends jump testing. Thus the duration for which the CTL bit is monitored is very short. In the program on the right, however, the JUMP command is encountered before the move command. This starts jump testing before motion begins and jump testing continues as long as the move lasts. If the CTL bit turns ON while the move is being performed, the jump will be performed. After the move completes, the BLOCK command ends jump testing and program execution continues normally. Jump testing would continue during subsequent moves encountered before the BLOCK command.

ACCEL	5000	ACCEL	5000
VELOC	1000	VELOC	1000
BLOCK	1	BLOCK	1
CMOVE	2000, ABS, LINEAR	JUMP	CTL01, 3
JUMP	CTL01, 3	CMOVE	2000, ABS, LINEAR
BLOCK	2	BLOCK	2

Normal Stop before JUMP

A conditional jump command is similar to Type 2 commands in that jump testing does not start until the Type 3 command immediately after the JUMP is executed. If this Type 3 command would normally stop motion, then motion will stop before jump testing begins. Type 3 commands that will stop motion are: DWELL, WAIT, End of Program, and moves in the opposite direction.

Thus even though the CTL bit may be ON before the block with the conditional JUMP and Type 3 command is executed, axis motion will stop before program execution continues at the jump destination. This stopping is NOT a Jump Stop, which is described in Example 10.

Example 8: Normal Stop before JUMP

The following example contains a jump followed by a DWELL command. The Power Mate APM, because it processes ahead, knows it must stop after the CMOVE command. Thus it comes to a stop before the DWELL is executed. Since jump testing does not begin until the DWELL is executed, testing begins after motion stops. Jump testing ends when the following CMOVE begins because of the BLOCK command associated with it.

The dotted lines in the velocity profile indicate when jump testing is taking place. The CTL03 bit does not turn ON during the program.

```
BLOCK    1
ACCEL    5000
VELOC    10000
CMOVE    60000, INC, LINEAR
BLOCK    2
JUMP     CTL03, 4
DWELL    4000
BLOCK    3
ACCEL    10000
VELOC    5000
CMOVE    15000, INC, LINEAR
BLOCK    4
NULL
```

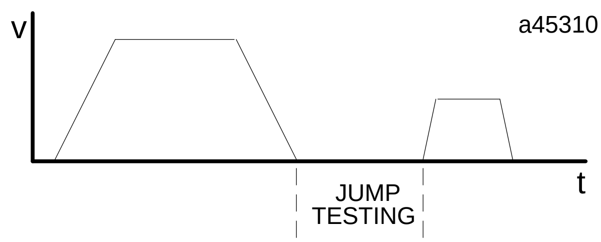


Figure 6-9. Normal Stop before JUMP

Jumping Without Stopping

If the Type 3 command following a conditional jump is a CMOVE and the Type 3 command at the destination is a move command with sufficient distance to fully decelerate to zero when completed, the jump will be executed without stopping. This is the only way to sustain motion when a jump is performed.

Example 9: JUMP Without Stopping

This is a simple example of a conditional jump from one CMOVE to another. While jump testing the CTL03 bit, the first CMOVE accelerates to the programmed velocity. Before the dotted line, the CTL03 bit is OFF, but at the dotted line the CTL03 bit turns ON. Program execution is immediately transferred to block 3 and the CMOVE there begins. Because the velocity at the jump destination is different, the velocity changes at the acceleration programmed of the jump destination block. Finally, as the second CMOVE completes, velocity is reduced to zero and the program ends.

```
BLOCK    1
ACCEL    2000
VELOC    10000
JUMP     CTL03, 3
```

```
CMOVE    120000, INC, LINEAR
BLOCK    3
ACCEL    20000
VELOC    5000
CMOVE    15000, INC, LINEAR
```

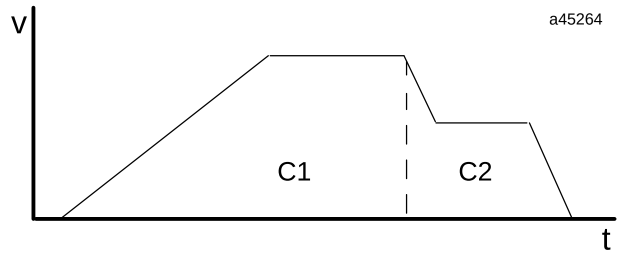


Figure 6-10. JUMP Without Stopping

Jump Stop

A jump stop is a stop that is caused by a jump. When a jump stop occurs, the *Jog Acceleration* and *Jog Acceleration Mode* are used instead of any programmed acceleration. Note that scurve motion will achieve constant velocity before using the *Jog Acceleration* and beginning to decelerate. See the scurve jump examples for more details. The *Jog Acceleration* is used because a jump stop may indicate something is wrong. The current *Jog Acceleration*, which can be changed with an immediate command, provides more versatility than the programmed acceleration. There are two ways of generating a jump stop, each described below.

A conditional JUMP triggered during a PMOVE will always generate a jump stop. Because a PMOVE always stops before continuing to a subsequent motion, a jump stop always occurs when a jump takes place during a PMOVE.

When a conditional jump trigger occurs during a CMOVE, however, a jump stop will not occur if the motion programmed at the jump destination is a PMOVE or CMOVE representing sufficient distance in the same direction.

In an scurve move, a jump stop will do one of two things. If the jump takes place after the midpoint of the acceleration or deceleration, the acceleration or deceleration is completed before the jump stop is initiated. If the jump occurs before the midpoint of the acceleration or deceleration, the profile will immediately begin leveling off. Once acceleration or deceleration is zero, the jump stop begins. See the scurve jump examples.

Example 10: Jump Stop

The following is an example conditional jump with a jump stop. An enhancement on Example 5, DWELL, would be to watch an external CTL bit which would indicate a problem with the positive motion. If the CTL bit never turns on, the profile for the following program will be identical to the profile shown in the DWELL example. If the CTL bit turned on during the first PMOVE or the DWELL, the reverse movement would immediately commence.

The following profile would appear if the CTL bit turned on during the first PMOVE, at the dotted line, and the *Jog Acceleration* was 75000. Because the first move completed early due to the CTL bit and a faster acceleration (*Jog Acceleration* versus programmed acceleration) the second move would not have to move as far to get back to 0 position as it did in the DWELL example. Note that because the motion programmed at the jump destination is in the opposite direction as the initial motion, the profile would be identical if the moves were CMOVEs instead of PMOVEs.

```
ACCEL    30000
VELOC    15000
BLOCK    1
JUMP     CTL09, 2
PMOVE    120000, ABS, LINEAR
DWELL    4000
BLOCK    2
PMOVE    0, ABS, LINEAR
```

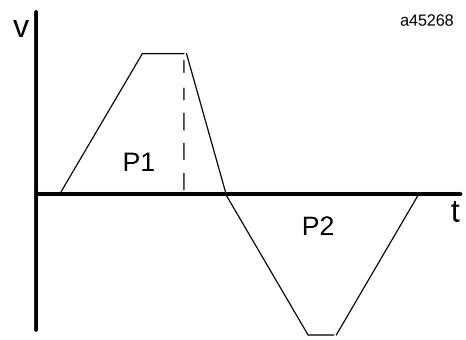


Figure 6-11. Jump Stop

Example 11: Jump Followed by PMOVE

In this JUMP example, the command after the JUMP is a PMOVE in the same direction. The velocity profile below shows the acceleration and movement for the first CMOVE and the deceleration to the PMOVE's velocity. The CTL01 bit, OFF when the PMOVE begins, turns ON at the dotted line. Motion stops after a PMOVE, even if a conditional jump goes to another block. Thus the CTL01 bit triggers a deceleration to zero before the final CMOVE begins.

```
BLOCK    1
ACCEL    2000
VELOC    8000
CMOVE    76000, INC, LINEAR
BLOCK    2
ACCEL    1000
VELOC    4000
JUMP     CTL01, 3
PMOVE    50000, INC, LINEAR
BLOCK    3
ACCEL    6000
VELOC    6000
CMOVE    36000, INC, LINEAR
```

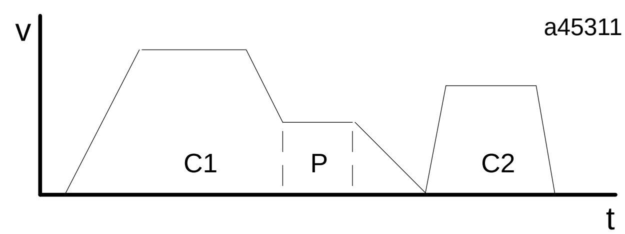


Figure 6-12. Jump Followed by PMOVE

SCURVE Jumps

Jumps during linear motion and jumps during scurve motion at constant velocities immediately begin accelerating or decelerating to a new velocity. Jumps during an scurve acceleration or deceleration, however, require different rules in order to maintain an s-curve profile. What happens when a jump occurs during an scurve move while changing velocity depends on whether the jump occurs before or after the midpoint (the point where the acceleration magnitude is greatest) and whether the velocity at the jump destination is higher or lower than the current velocity.

If the jump occurs after the midpoint of the change in velocity, the change will continue normally until constant velocity is reached; then the velocity will be changed to the new velocity using the acceleration mode of the move at the jump destination.

Example 12: SCURVE – Jumping After the Midpoint of Acceleration or Deceleration

In the following example, a jump occurs during the final phase of deceleration, at the dotted line. The deceleration continues until constant velocity is reached and then the acceleration to the higher velocity begins.

```
ACCEL       50000
VELOC       100000
BLOCK       1
JUMP        CTL01, 3
CMOVE       500000, ABS, SCURVE
BLOCK       2
VELOC       60000
CMOVE       –500000, INC, SCURVE
BLOCK       3
VELOC       85000
ACCEL       100000
CMOVE       250000, INC, SCURVE
```

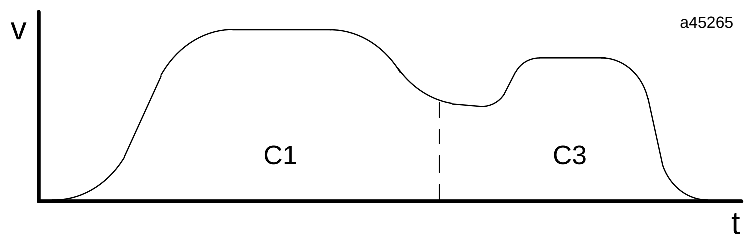


Figure 6-13. Jumping After the Midpoint of Acceleration or Deceleration

If a jump takes place before the midpoint of acceleration or deceleration, the result depends on whether the velocity at the jump destination is higher or lower than the velocity before the jump took place. In the first case, when accelerating but the new velocity is lower, or decelerating and the new velocity is greater, the Power Mate APM will immediately begins reducing the acceleration or deceleration to zero; once zero, the Power Mate APM will use the jump destination acceleration and velocity and change to the new velocity.

Example 13: SCURVE – Jumping Before the Midpoint of Acceleration or Deceleration

In the following example, during the acceleration of the first CMOVE, a jump takes place at the first dotted line. Because the velocity at the jump destination is lower than the velocity of the first CMOVE the Power Mate APM slows the acceleration to zero. Constant velocity, zero acceleration, occurs at the second dotted line. There, the Power Mate APM begins decelerating to the new velocity using the acceleration at the jump destination. Finally, the second CMOVE finishes.

```
ACCEL      1000
VELOC      50000
BLOCK      1
JUMP       CTL01, 3
CMOVE      50000, INC, SCURVE
BLOCK      3
VELOC      5000
ACCEL      10000
CMOVE      15000, INC, SCURVE
```

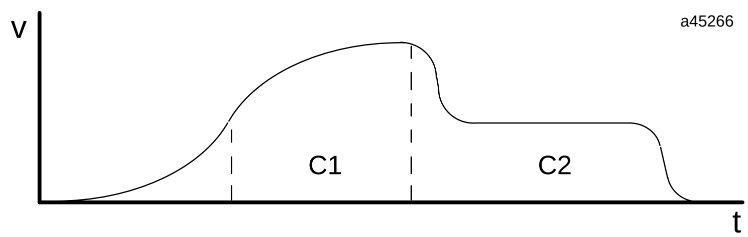


Figure 6-14. Jumping before the Midpoint of Acceleration or Deceleration

The second case involves jumping to a higher velocity while accelerating or a lower velocity while decelerating. When this occurs, the Power Mate APM continues to the first move's acceleration or deceleration. This acceleration or deceleration is maintained, similar to be a linear acceleration, until the axis approaches the new velocity. Then the normal scurve is used to reduce acceleration or deceleration to zero.

Example 14: SCURVE – Jumping to a Higher Velocity While Accelerating or Jumping to a Lower Velocity While Decelerating

In this example, a JUMP command is triggered during the initial phase of acceleration (before the first dotted line) and the velocity at the jump destination is higher than that of the current move. The first dotted line indicates the maximum acceleration of the first CMOVE. This value is held as the axis continues to accelerate until it scurves back to constant velocity. Constant velocity, the second dotted line, indicates the beginning of the second CMOVE. This move continues until it decelerates to zero at the end of the program.

```
ACCEL       50000
VELOC       30000
BLOCK       1
JUMP        CTL02, 2
CMOVE       150000, INC, SCURVE
BLOCK       2
VELOC       90000
ACCEL       25000
CMOVE       500000, INC, SCURVE
```

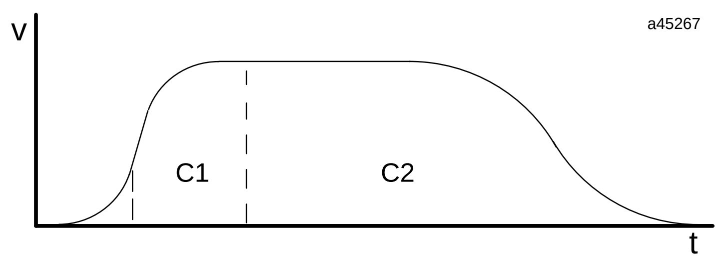


Figure 6-15. Jumping to a Higher Velocity While Accelerating or Jumping to a Lower Velocity While Decelerating

Other Programmed Motion Considerations

Maximum Acceleration Time

The maximum time for a programmed acceleration or deceleration is 64 seconds for the 1 axis Power Mate APM and 128 seconds for the 2 axis Power Mate APM. If the time to accelerate or decelerate is computed to be longer than this time, the Power Mate APM will compute an acceleration to be used based on 64 or 128 seconds. To obtain longer acceleration times, multiple CMOVEs with increasing or decreasing velocities must be used.

Example 15: Maximum Acceleration Time

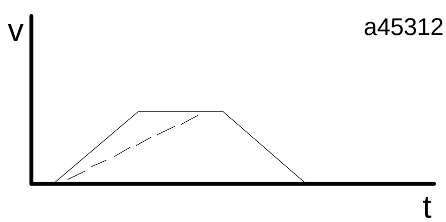
The following 1 axis program shows a problem with a very long acceleration time and a solution. In the first program and profile, 120 seconds, two minutes, is required to reach the programmed acceleration. Since this is greater than 64 seconds, the Power Mate APM calculates that an acceleration of 188 would allow a velocity of 12000 to be reached in 64 seconds. The lefthand velocity profile below shows the slightly higher 188 acceleration used. Also shown is a dotted line indicating the programmed acceleration to constant velocity.

One solution for obtaining a low acceleration for a long period of time breaks the move up into separate moves with individual acceleration times less than 64 seconds. This method requires some calculation. Each acceleration and deceleration must be broken into moves with acceleration times less than 64 seconds. Thus to allow an acceleration of 100 during acceleration and deceleration, three moves will be required.

The second program and profile, those on the right, show how the first program can be broken into three parts. The distance at the midpoint of each acceleration, when velocity is 6000, is calculated to be 180,000, one fourth the distance required to accelerate to 12000. An initial CMOVE uses this distance. The next CMOVE will then accelerate to its velocity at the same acceleration rate. The final PMOVE is the midpoint distance, 180,000 user units, from the final position. The second CMOVE will automatically decelerate to the PMOVE's velocity as it approaches its final position. The dotted lines indicate when the second CMOVE begins and ends.

```
ACCEL    100
VELOC    12000
PMOVE    1500000, INC, LINEAR
```

```
ACCEL    100
VELOC    6000
CMOVE    180000, INC, LINEAR
VELOC    12000
CMOVE    1,140000, INC, LINEAR
VELOC    6000
PMOVE    180000, INC, LINEAR
```



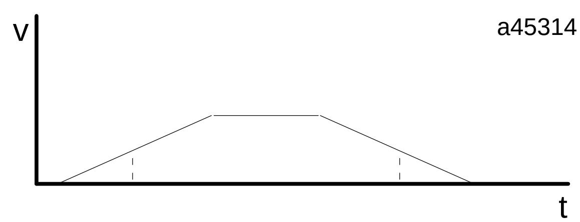


Figure 6-16. Maximum Acceleration Time

Feedhold with the Power Mate APM

Feedhold is used to pause program execution without ending the program, often to examine some aspect of a system. It causes all axis motion to end at the programmed acceleration. When feedhold is ended, program execution resumes. Interrupted motion will resume at the programmed acceleration and velocity.

Feedhold is asserted by turning ON the *Feedhold* %Q bit and lasts until the %Q bit is turned OFF. The *Abort All Moves* %Q bit turning ON or an error which would normally cause a stop error will end feedhold as well as terminate the program. During feedhold, jogging positive and negative is allowed, but no other motion. When feedhold is terminated and program execution resumes, the Power Mate APM remembers and will move to its previous destination.

Example 16: Feedhold

The following example illustrates a motion profile when feedhold is applied. The linear move accelerates to the programmed velocity at the programmed rate. Feedhold is applied at the dotted line, so velocity decreases at the programmed acceleration to zero. Then, a Jog is performed using the *Jog Minus %Q* bit. This is evident because the *Jog Velocity* is negative. Note the acceleration used during the Jog is the current *Jog Acceleration* which is different than the programmed acceleration. Note also, the feedhold must be applied during the entire duration of the Jog. After the jog motion has ceased, the feedhold is ended and the program continues to completion.

```
ACCEL      1000
VELOC      2000
PMOVE      12000, INC, LINEAR
```

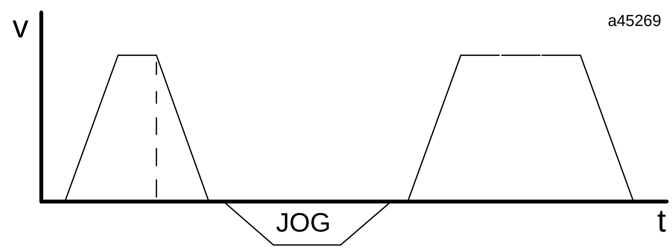


Figure 6-17. Feedhold

Feedrate Override

Some applications require small modifications to a programmed velocity to handle outside changes. A *Feedrate Override* immediate command allows changes to a programmed velocity during program execution. Whenever a program begins, the override rate is initially set to 100%. Thus changes to feedrate before the execute program bit is turned ON will be ignored, but a feedrate commanded on the same sweep as an execute program bit will be effective.

A percentage may be assigned to the feedrate from 0% to 120%. When a *Feedrate Override* is commanded, the Power Mate APM internally multiplies the feedrate percentage by programmed velocity to obtain a new velocity. If the axis is moving, the current move's *Jog Acceleration Mode* is used to change velocity to the new velocity. All future move velocities will be affected by the feedrate change. Note that when a feedrate of 0% is applied, no motion will be generated until a new feedrate is commanded. Also note the *Moving %I* bit stays ON when the feedrate is 0%.

Feedrate Override has no effect on non-programmed motion such as *Jog, Find Home,* or *Move at Velocity.*

Example 17: Feedrate Override

During execution of this program, feedrate changes of + or -10% are commanded. Dotted lines indicate -10%, dashed lines indicate +10%.

ACCEL	1000
VELOC	6000
PMOVE	110000, INC, LINEAR

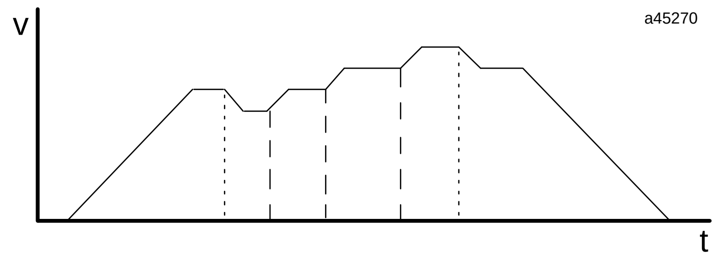


Figure 6-18. Feedrate Override

Multi-Axis Programming

Sync Blocks may be used in a multi-axis program to synchronize the axis motion commands at positions where timing is critical.

Example 18: Multi-Axis Programming

This example assumes that axis 1 controls vertical motion and axis 2 controls horizontal motion. The objective is to move a piece of material from point A to point C as quickly as possible while avoiding the obstacle which prevents moving directly from A to C.

A simple way would be to move from point A to point B, and then from point B to point C. This sequence, however, wastes time. A better way would begin the horizontal movement before reaching point B. It has been determined that after axis 1 has moved to a position of 30,000 user units, axis 2 could then start and still clear the obstacle. The program segment could be programmed as follows:

```
BLOCK 10            CMOVE, 30000, INC, AXIS 1
BLOCK 20 [SYNC]     PMOVE, 50000, INC, AXIS 1
                    PMOVE, 150000, INC, AXIS 2
```

When this program is executed, axis 1 immediately begins its 30,000 unit move. Axis 2 would ignore the first command, because it applies only to axis 1, and see the Sync Block. Axis 2 waits for axis 1 to reach the Sync Block before it continues executing the program. When axis 1 reaches the 30,000 unit mark, it begins the 50,000 unit PMOVE at the Sync Block without stopping (the first move was a CMOVE). Now that axis 1 has reached the Sync Block, axis 2 begins its 150,000 unit move. Looking at the position profile below, axis 1 completes its move first and stops at the end of the PMOVE. When axis 2 reaches point C, it also stops.

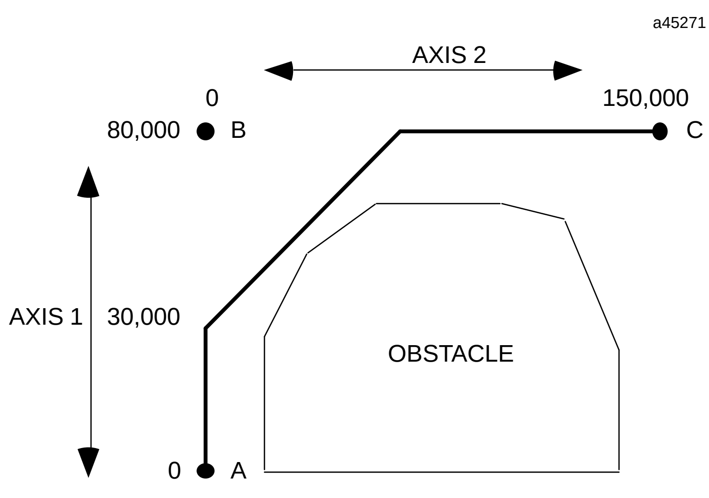


Figure 6-19. Multi-Axis Programming

If this program segment is not at the beginning of a program and for some reason axis 2 has not yet reached Block 20 when axis 1 has moved 30,000 counts, an error would occur. Axis 1 would continue to 80,000 counts, and the Power Mate APM would report a Block Sync Error during a CMOVE in the Status Code.

If it is imperative that the axes synchronize at Block 20, then changing Block 10 to a PMOVE would guarantee synchronization, but then axis 1 would always stop momentarily at 30,000 counts

Parameters in the Power Mate APM

The Power Mate APM maintains 256 double word parameters (0 through 255) in memory. These values may be used as a parameter in ACCEL, VELOC, DWELL, PMOVE, and CMOVE motion commands. Note that range limits still apply and errors may occur if a parameter contains a value out of range. The last ten parameters are special purpose parameters. The Power Mate APM can load data into these parameters which might overwrite user data. The following table describes the function of the special purpose parameters.

Parameter Number	Special Purpose Function
246 - 251	Reserved for future use
252	Stores Axis 2 *Commanded Position* (user units) when Follower Enable Trigger occurs (2 Axis Power Mate APM only)
253	Stores Axis 1 *Commanded Position* (user units) when Follower Enable Trigger occurs
254	Stores Axis 2 *Strobe Position* value (user units), 2 Axis Power Mate APM only
255	Stores Axis 1 *Strobe Position* value (user units)

Parameters are all reset to zero after a power cycle or after a Power Mate APM configuration is stored by the PLC. Parameters can be assigned in three ways: the motion program command LOAD, the immediate command Load Parameter Immediate, and the COMM_REQ function block in the PLC. The COMM_REQ function block is described in Appendix B. Assigning a value to a parameter overwrites any previous value. Parameter values may be changed during program execution, but the change must occur before the Power Mate APM begins executing the block previous to the block that uses the parameter.

Appendix A — Error Codes

The *Status Code* word of the %AI Status Words contains a code which describes the error indicated when the *Error* status bit is set. There are three categories of errors reported by the *Status Code.*

- Programming errors which generate a Status Only (warning) message.
- Programming errors that halt the servo
- Hardware errors (encoder out of sync, PLC Run switches off, loss of programmer communications, and so forth)

Note

> The Status LED on the faceplate of the module flashes slow (4 times / second) for Status Only errors and fast (8 times / sec) for errors which cause the servo to stop.

Error Codes are placed in the Status Code %AI word. The format for the Status Code word is:

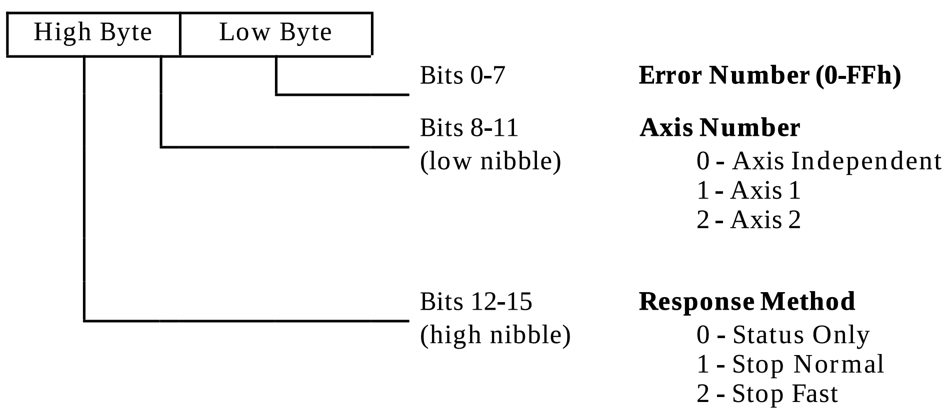


Figure A-1. Status Code Organization

Response Methods

1. **Status Only Errors:** Set the error flag and status code, but do not affect motion.
2. **Stop Normal Errors:** Perform an internal abort of any current motion. The *Drive Enabled* and *Axis Enabled %I* bits are turned OFF after the configured *Drive Disable Delay.*
3. **Stop Fast Errors:** Instantly abort all motion by setting the analog output voltage to zero. The *Drive Enabled* and *Axis Enabled %I* bits are turned off after the configured *Drive Disable Delay.*

Table A-1. Status Word Error Codes

Error Number (Hexadecimal)	Response	Description
0	None	No Error
		Configuration Errors
2	Status Only	Scaled data too big, maximum value in range used
3	Status Only	Home Position > Positive EOT, Positive EOT used
4	Status Only	Home Position < Negative EOT, Negative EOT used
		Configuration Parameter Errors
10	Status Only	Position Loop Time Constant too large, Immediate command ignored
11	Status Only	Position Loop Time Constant too small, Immediate command ignored
12	Status Only	Position Loop Time Constant computation overflow, reduced to non-overflow value
1E	Status Only	Immediate command Jog Velocity out of range, command ignored
1F	Status Only	Immediate command Jog Acceleration out of range, command ignored
		Program Errors
20	Status Only	Program Acceleration overrange, defaults to 16.7 million cts / sec / sec
21	Status Only	Program Acceleration too small, defaulted to 32 cts / sec / sec
22	Status Only	Scaled Velocity greater than 1 million cts / sec, 1 million cts / sec is used
23	Status Only	Program Velocity is zero, defaulted to 1 count / sec used
24	Stop Normal	Program Position too large
25	Stop Normal	Unconditional Jump Destination not found
26	Stop Normal	Jump Mask error
27	Stop Normal	Wait Mask error
28	Stop Normal	Parameter Position too large
29	Status Only	Dwell time greater than 60 seconds, 5 seconds used
		Position Increment Errors
2C	Status Only	Position Increment Overrange error, increment ignored
		Find Home Errors
30	Status Only	Find Home while Drive Not Enabled error *
31	Status Only	Find Home while Program Selected error *
32	Status Only	Find Home while D / A forced error *
33	Status Only	Find Home while Jog error *
34	Status Only	Find Home while Move at Velocity error *
36	Status Only	Find Home while Abort bit set error *
		Move at Velocity Errors
39	Status Only	Move at Velocity while Drive Not Enabled error *
3A	Status Only	Move at Velocity while Program Selected error *
3B	Status Only	Move at Velocity while Home Cycle active error *
3C	Status Only	Move at Velocity while Jog error *
3D	Status Only	Move at Velocity while Abort All Moves bit is set error *
3E	Status Only	Move at Velocity Data greater than 8,388,607 user units / sec *
3F	Status Only	Move at Velocity Data greater than 1 million cts / sec error *
		Jog Errors
40	Status Only	Jog while Find Home error *
41	Status Only	Jog while Move at Velocity error *
42	Status Only	Jog while Force D / A error *
43	Status Only	Jog while Program Selected and not Feedholding error *

* Status error is reported, command is not executed.

Table A-1. Status Word Error Codes (Continued)

Error Number (Hexadecimal)	Response	Description
		Force D/A Errors
47	Status Only	Force D / A while Jog error *
48	Status Only	Force D / A while Move at Velocity error *
49	Status Only	Force D / A while Program Selected error *
		Set Position Errors
50	Status Only	Set Position while Program Selected error *
51	Status Only	Set Position Data overrange error *
52	Status Only	Set Position while not In Zone error *
53	Status Only	Attempt to initialize position before digital encoder passes reference point.
54	Status Only	Digital encoder position invalid, must use Find Home or Set Position.
		End of Travel and Count Limit Errors
56	Status Only	Commanded Position > Positive End of Travel or High Count Limit *
57	Status Only	Commanded Position < Negative End of Travel or Low Count Limit *
58	Status Only	(Absolute Position + Position offset) > Positive End of Travel or High Count Limit
59	Status Only	(Absolute Position + Position offset) < Negative End of travel or Low Count Limit
		Drive Disable Errors
5B	Stop Normal	Drive Disabled while Moving
5C	Stop Normal	Drive Disabled while Program Active
		Software Errors
5F	Status Only	Software Error (Call G.E. Fanuc Field Service)
		Program and Subroutine Errors
61	Stop Normal	Subroutine not in list
62	Stop Normal	Call Error (subroutine already active)
63	Stop Normal	Subroutine End command found in Program
64	Stop Normal	Program End command found in Subroutine
		Program Execution Errors
70	Status Only	Request Program 0 with other programs active *
71	Status Only	Too many programs requested in same PLC sweep *
72	Status Only	Request Program 1-10 with multi-axis program active *
73	Status Only	Request two programs on same sweep with program active *
74	Status Only	Request two programs for same axis, lower number program executed
75	Status Only	Empty or Invalid Program requested
		Program Execution Conditions Errors
80	Status Only	Execute Program while Home Cycle active *
81	Status Only	Execute Program while Jog *
82	Status Only	Execute Program while Move at Velocity *
83	Status Only	Execute Program while D / A Forced *
84	Status Only	Execute Program while Program Selected *
85	Status Only	Execute Program while Abort All Moves bit set *
86	Status Only	Execute Program while Position Valid not set *
87	Status Only	Execute Program while Drive Enabled not set *

* Status error is reported, command is not executed.

Table A-1. Status Word Error Codes (Continued)

Error Number (Hexadecimal)	Response	Description
		Program Synchronous Block Errors
8C	Status Only	Sync Block Error during CMOVE
8D	Status Only	Sync Block Error during Jump
		EEPROM Errors
90	Status Only	Flash EEPROM memory programming failure
		Hardware Limit Switch Errors
A0	Stop Fast	Limit Switch (+) error
A1	Stop Fast	Limit Switch (–) error
		Hardware Errors
A8	Stop Fast	Out of Sync error
A9	Stop Fast	Encoder Loss of Quadrature or Linear Feedback Loss of Signal error
AA	Stop Normal	Analog Input Failure
		Digital Servo Configuration Errors
B1	Status Only	CRC , SCRDY or SAL1–SAL0 error during configuration
B2	Status Only	Wait > 10 seconds for initial PM–J communications
B3	Status Only	PCGACK not = 1 error
B4	Status Only	PCGACK not = 0 error
B5	Status Only	DSI not ready for control mode
		Digital Servo Control Errors
B8	Stop Fast	DSI interrupt missing
B9	Stop Fast	CRC error Communication to DSI
BE	Stop Fast	Communicationsalarm from DSI
BF	Stop Fast	Hardwarealarm from DSI
C0	Stop Fast	Servo not ready when MCON command is on - may be caused by E-STOP input to amplifier.
C1	Status Only	Serial Encoder Battery Low
C2	Stop Normal	Serial Encoder Battery Failed
C3	Stop Normal	Servo Motor Over Temperature
C4	Stop Fast	Servo Motor Over Current
C5	Stop Fast	Serial Encoder Communications Alarm
C6	Stop Fast	Serial Encoder Alarm
C7	Stop Fast	Servo Unit Alarm
		Special Purpose Errors
E0	Status Only	Custom Loop Type Mismatch
E1	Status Only	Digital mode not supported by hardware
EF	Status Only	Firmware / Hardware Axis Number Mismatch(1-axix firmware in 2-axis module, or 2-axis firmware in 1-axis module)

Appendix B Parameter Download Using COMM_REQ

This appendix describes an alternate method of loading Power Mate APM Parameter Memory from the PLC using a the COMM_REQ function block with command code E501h. This function block can send up to 16 PM-APM Parameter values at once. The total data length of the COMM_REQ must be set to 68 bytes (34 words) organized as shown below.

Word Offset	Byte Offset	Data
0	0 - 1	Starting parameter number (0 - 255)
1	2 - 3	Number of parameters to load
2 - 3	4 - 7	1st parameter data (4 bytes)
4 - 5	8 - 11	2nd parameter data (4 bytes)
...	...	...
32 - 33	64 - 67	16th parameter data (4 bytes)

Only the number of parameters specified in word offset 1 will be loaded into parameter memory. However, the 68 byte data block must always be initialized in the PLC. If the last parameter to be loaded is greater than 255, the COMM_REQ will be rejected. A parameter block download is illustrated in the following PLC program segment.

```
REFERENCE NICKNAME     REFERENCE DESCRIPTION
---------    ---------    ---------------------------------

%R0195 CMREQST   COMM_REQ STATUS WORD (Updated by COM_REQ)
%R0196 HDR_WDS   COMM_REQ HEADER LENGTH IN WORDS (ALWAYS 4)
%R0197 NO_WAIT   NO WAIT (ALWAYS 0)
%R0198 STMEMTP   STATUS MEMORY TYPE (8=REG)
%R0199 STLOCM1   STATUS WORD LOCATION MINUS 1 (194 = %R0195)
%R0200 NO_USE1   NOT USED (ZEROED BY BLKMV)
%R0201 NO_USE2   NOT USED (ZEROED BY BLKMV)

%R0202 CMDTYP    COMMAND TYPE (E501 FOR PM-APM)
%R0203 BYTECNT   BYTE COUNT OF DATA
%R0204 MEMTYP    MEMORY TYPE OF DATA (8=REG)
%R0205 DATAST    START OF DATA BLOCK –1 (205 = %R0206)
%R0206 PAR_NO    STARTING PARAM NUMBER
%R0207 NO_VALS   NUMBER OF PARAMETERS TO SEND

%R0208–%R0239 PAR_DAT   Data for 16 Parameters (32 words)
```

```
|   (*************************************************************************)
|   (*  CLEAR THE REGISTER BLOCK, THEN LOAD THE COMM_REQ HEADER DATA       *)
|   (*************************************************************************)
|
|   SEND
| %T001   +-----+                 +-----+
+--] [----+ BLK_+-----------------+BLKMV+-
|         | CLR_|                 | INT |
|         | WORD|                 |     |
| CMREQST |     |         CONST -+IN1 Q+-%R0195
| %R0195  +IN   +         +00000 |     |
|         | LEN |                 |     |
|         |00045|         CONST -+IN2  |
|         |     |         +00004 |     |
|         +-----+                 |     |
|                         CONST -+IN3  |
|                         +00000 |     |
|                                 |     |
|                         CONST -+IN4  |
|                         +00008 |     |
|                                 |     |
|                         CONST -+IN5  |
|                         +00194 |     |
|                                 |     |
|                         CONST -+IN6  |
|                         +00000 |     |
|                                 |     |
|                         CONST -+IN7  |
|                         +00000 +-----+
|
|
|   (*************************************************************************)
|   (* PUT THE COMMAND TYPE (E501) IN  THE FIRST DATA WORD (R202)           *)
|   (* PUT THE BYTE COUNT OF THE DATA IN THE NEXT WORD (R203).  PUT THE     *)
|   (* MEMORY TYPE OF THE DATA (8=REG) IN THE NEXT WORD (R204). PUT THE     *)
|   (* STARTING LOCATION OF THE DATA BLOCK -1 IN THE NEXT WORD (R205)       *)
|   (* PUT THE STARTING PARAMETER NUMBER (1) IN THE LOWER BYTE OF (R206)   *)
|   (* AND THEN THE NUMBER OF PARAMETERS TO BE SENT (16) IN THE LOWER      *)
|   (* BYTE OF R207                                                          *)
|   (*************************************************************************)
|
|   SEND
| %T0001  +-----+                +-----+                 +-----+
+--] [----+MOVE_+----------------+MOVE_+-----------------+MOVE_+-
|         | WORD|                | INT |                 | INT |
|         |     |                |     |                 |     |
|         |     | CMDTYP         |     | BYTECNT         |     | MEMTYP
|   CONST -+IN  Q+-%R0202  CONST -+IN  Q+-%R0203   CONST -+IN  Q+-%R0204
|    E501  | LEN |         +00068 | LEN |          +00008 | LEN |
|         |00001|                |00001|                 |00001|
|         |     |                |     |                 |     |
|         +-----+                +-----+                 +-----+
|
|   SEND
| %T0001  +-----+                +-----+                 +-----+
+--] [----+MOVE_+----------------+MOVE_+-----------------+MOVE_+-
|         | INT |                | INT |                 | INT |
|         |     |                |     |                 |     |
|         |     | DATAST         |     | PAR_NO          |     | NO_VALS
|   CONST -+IN  Q+-%R0205  CONST -+IN  Q+-%R0206   CONST -+IN  Q+-%R0207
|   +00205 | LEN |         +00001 | LEN |          +00016 | LEN |
|         |00001|                |00001|                 |00001|
|         |     |                |     |                 |     |
|         +-----+                +-----+                 +-----+
|
```

```
|
|  (***********************************************************************)
|  (* ADD LOGIC HERE TO MOVE THE PROPER CONSTANTS INTO THE REGISTERS      *)
|  (* (%R208 - %R239) SO THEY CAN BE SENT TO THE PM-APM PARAMETERS.       *)
|  (***********************************************************************)
|
|  (***********************************************************************)
|  (* NOW ACTIVATE THE COMM_REQ TO SEND THE PARAMETER DATA TO THE PM-APM   *)
|  (***********************************************************************)
|
|  SEND
| %T0001   +-----+
+--] [-----+COMM_+-
|          | REQ |     NOTE:  SYSID HIGH BYTE = COMM_REQ RACK DESTINATION
|          |     |            SYSID  LOW BYTE = COMM_REQ SLOT DESTINATION
|  HDR_WDS |     |
|  %R0196 -+IN FT+-           TASK ALWAYS = 0 FOR PM-APM COMM_REQ
|          |     |
|   CONST -+SYSID|
|    0007  |     |
|          |     |
|   CONST -+TASK |
| 00000000 +-----+
```

Appendix C Specifications

This appendix provides specifications for the Power Mate APM module, and wiring information for the I / O connections.

Module Specifications

Power Supply Voltage:	5 Vdc from backplane
Power Supply Current:	800 mA + (1.4 x Encoder Current Drain) + (95 mA SNP Mini-converter Current, if used)
Available Encoder +5V Current/Module:	500 mA @40C (104F) derated to 300 mA @ 55C (131F)
Maximum Number of modules/system: **Model 311, 313, 321, 323 PLCs :**	3 IC693APU301 modules in CPU baseplate (limited by power supply) 2 IC693APU302 modules in CPU baseplate (limited by %AI data)
Model 331 PLC :	8 IC693APU301 or 4 IC693APU302 modules in CPU, expansion and remote baseplates (limited by %AI data). Maximum of 3 IC693APU modules per baseplate (limited by power supply)
Model 341 PLC :	14 IC693APU301 or IC693APU302 modules in CPU, expansion and remote baseplates. Maximum of 2 IC693APU modules in CPU baseplate and 3 in each expansion / remote baseplate (limited by power supply)
Model 351, 352 PLC :	23 IC693APU301 or IC693APU302 modules in CPU, expansion and remote baseplates. Maximum of 2 IC693APU modules in CPU baseplate and 3 in each expansion / remote baseplate (limited by power supply) *Note that a Model 351/352 system can have up to 8 baseplates (CPU baseplate and 7 expansion/remote baseplates)*

NOTE: Refer to GFK-0867B (GE Fanuc Product Agency Approvals, Standards, General Specifications), or later version, for product standards and general specifications.

I/O Specifications

The specifications and circuitry for the I/O connections are provided below.

Velocity Command

Output of D/A converter with the following characteristics.

- Resolution: 13 bits including sign
- Linearity: .02% of full scale output
- Offset Voltage: ± 500 µV maximum
- Maximum Output: ± 10.0 V , ± 0.3 V
- Minimum Load Resistance: 2000 Ω
- Maximum allowable Voltage Between Analog Common and Ground: ± 1.0 V

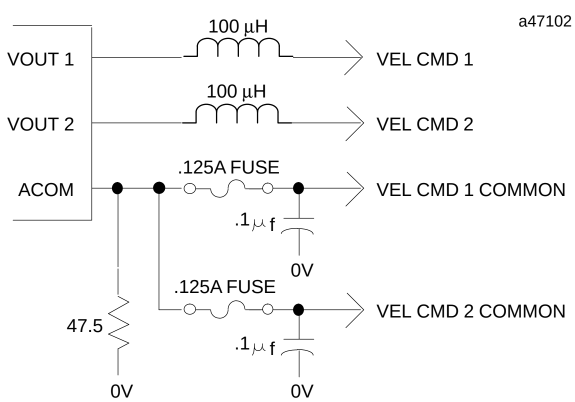


Figure C-1. Velocity Command Circuitry

Enable Relay Output

Normally-open dc solid state relay contact; Contacts rated at 30 V, 100 mA dc. Resistive load only. The off-state leakage current is 10 A maximum.

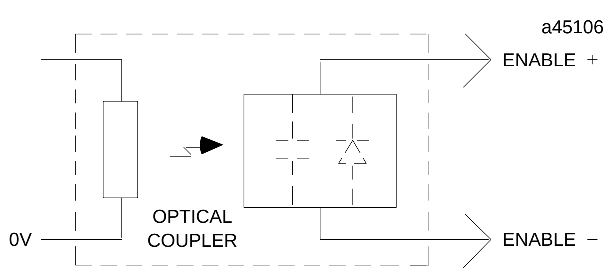
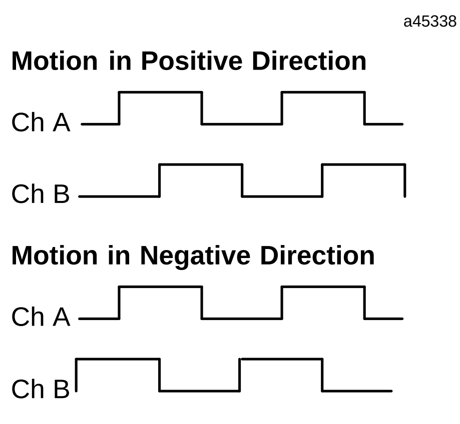


Figure C-2. Enable Relay Output Circuitry

Encoder Inputs

- Input Type: 5V differential or single ended
- Input Impedance: 4000 Ω (common mode)
- Input Threshold: Single ended: +1.5 V nominal (±0.4V) , Differential: +0.5 V nominal 〈±0.4V)
- Input Common Mode Range: ±15 V
- Single Ended Input Voltage: +15 V maximum
- Maximum Input Frequency: 175 kHz / Channel (X4 count rate = 700 kHz)
- Input Filtering: Noise pulses shorter than 1 Microsecond will be rejected.
- Encoder Power: Non-isolated current limited +5 V supply is available at the PM-APM front panel I / O connectors for use by one or more encoders. Maximum load must be limited to 500 mA @ 40°C, 300 mA @ 55°C.
- Quadrature Tolerance: 90 degrees ± 30 degrees.
- Quadrature Error Detection: Simultaneous transitions on the Channel A and Channel B inputs will be detected as loss of quadrature.
- Z Channel (Marker Operation: A positive going edge on the marker channel will be used to latch the present encoder position. The minimum pulse width is 4 microseconds.)
- Encoder direction of travel: Channel A leading Channel B indicates motion in positive direction:



Strobe Inputs

- Input Type: 5V differential or single ended
- Input Impedance: 4000 Ohms (common mode)
- Input Threshold: Single ended: +1.5 V nominal ⟨± 0.4V), Differential: +0.5 V nominal (± 0.4V)
- Input Common Mode Range: ± 15 V
- Single Ended Input Voltage: + 15 V maximum
- Minimum 3 microsecond pulse width is required
- A pulse on the strobe input causes the latest value of Actual Position to be reported in the ”Strobe Position” portion of %AI data.
- Position Capture Delay: 0 to 1 ms (IC693APU301)
  0 to 2 ms (APU302).

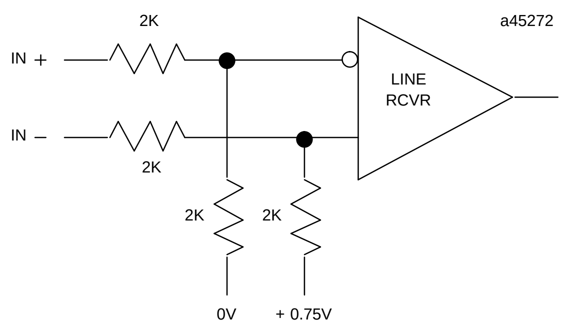


Figure C-3. Encoder and Strobe Input Circuitry

Encoder Power

- Encoder Power: Non-isolated current limited +5 V supply is available at the PM-APM front panel I/O connectors for use by one or more encoders. Maximum load must be limited to 500 mA @ 40C, 300 mA @ 55C.

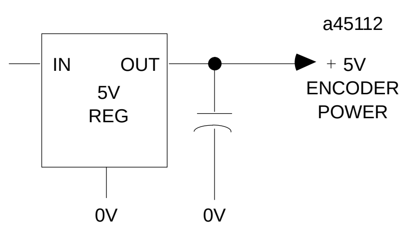


Figure C-4. Encoder Power Circuitry

General Purpose 24 VDC Digital Inputs

I/OConnector A:

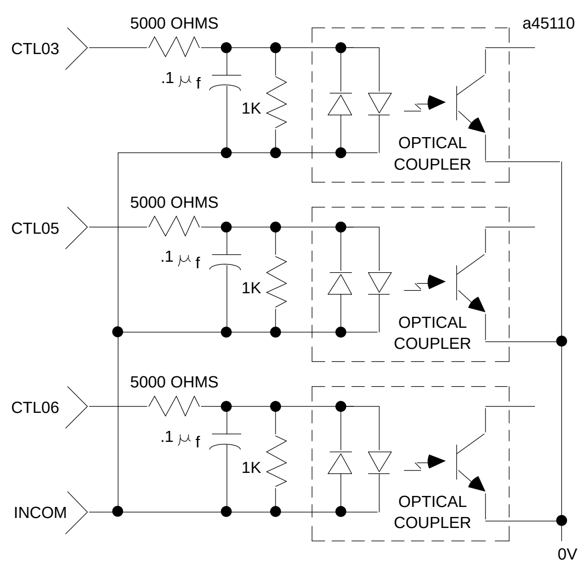
- Axis 1 Home Switch, CTL03
- Axis 1 Positive Overtravel Switch, CTL05
- Axis 1 Negative Overtravel Switch, CTL06

I/O Connector B:

- Axis 2 Home Switch, CTL04
- Axis 2 Positive Overtravel Switch, CTL07
- Axis 2 Negative Overtravel Switch, CTL08

Optically Isolated with the following specifications:

- Optically isolated DC source/sink (bidirectional input optocoupler)
- Input ON threshold: 18.0 V to 30.0 V
- Input OFF threshold: 0 V to 4.0 V
- Input resistance: 5000 Ω ± 10%
- Input filtering: 5 ms nominal
- Isolation voltage: 1500 V peak transient



NOTE: I/O Connector A is shown.
INCOM on Connector A is isolated from INCOM on Connector B.

Figure C-5. General Purpose Input Circuitry

General Purpose Outputs

- CTL09
- CTL10
- CTL11
- CTL12

5V signal level, 5V power generated by PM-APM

- Driver is a CMOS buffer with 20 Ohm Output Impedance
- Suitable for driving indicators or logic inputs
- Must not be used to drive inductive loads, including relay coils

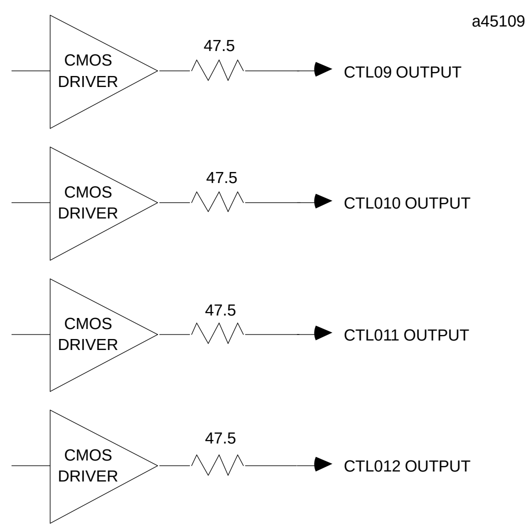


Figure C-6. General Purpose Output Circuitry

Analog Input

12-bit plus sign A / D converter with the following specifications:

- Input range: ± 10.0 V
- Input impedance: 50 k Ω
- Common mode range: ± 20 V
- Resolution: 12 bits magnitude plus sign bit
- Linearity: < 1 LSB
- Accuracy: 2% of reading ± 4 LSBs
- Scale Factor: +10.0 V = 32,000, -10V = -32,000
- Update Rate: 16 ms (does not include PLC scan time)

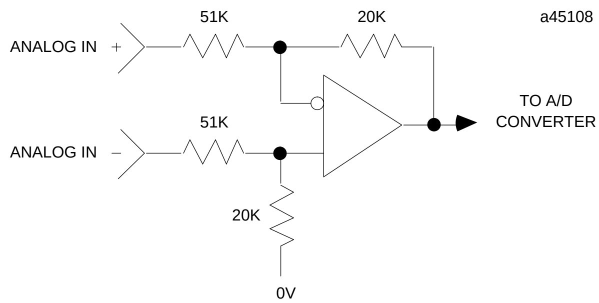


Figure C-7. Analog Input Circuitry

I/O Connector Cable Specifications

The cable that connects from the I / O connector to an external terminal block can be shortened to meet the requirements of your installation. Refer to Table C-1 to correctly match cable wires with connector pins. Also, refer to Figures 2-4 through 2-7 and Tables 2-2 through 2-5 for specific wiring requirements.

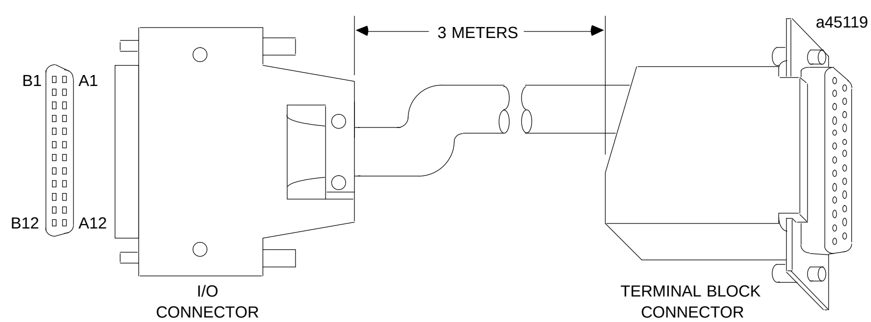


Figure C-8. I/O Connector Cable Specifications

I/O Cables

An I/O Cable assembly for the PM-APM modules consisting of a 24-pin connector, a cable, and a 25-pin D-type terminal block connector is available in two lengths. Catalog number IC693CBL311 is 10 feet (3 meters) in length and IC693CBL319 is 3 feet (1 meter) in length. The cable wire code list for these cables is provided in Table C-1.

An I/O cable assembly similar to IC693CBL311, but with the drain shield wire disconnected from pin B12 and brought outside of the cable housing through an 8" pigtail, is also available in two lengths. Catalog number IC693CBL317 is 10 feet (3 meters) in length and IC693CBL320 is 3 feet (1 meter) in length. *This cable improves the PM-APM's noise immunity*. The cable wire code list for these cables is provided in Table C-2.

Table C-1. I/O Cable Wire Coding for IC693CBL311 and IC693CBL319

I/O Connector Pin Number	Cable Wire Color Codes	25-Pin Connector Terminal Number*
no connection	Wire 1 Pair1(Brown / Black)	25
A1	Wire 2 Pair 1 (Brown)	12
B1	Wire 1 Pair2(Red / Black)	24
A2	Wire 2 Pair 2 (Red)	11
B2	Wire 1 Pair3(Orange / Black)	23
A3	Wire 2 Pair 3 (Orange)	10
B3	Wire 1 Pair 4 (Yellow / Black)	22
A4	Wire 2 Pair 4 (Yellow)	9
B4	Wire 1 Pair5(Green / Black)	15
A5	Wire 2 Pair 5 (Green)	2
B5	Wire 1 Pair6(Blue / Black)	14
A6	Wire 2 Pair 6 (Blue)	1
B6	Wire 1 Pair 7 (Violet / Black)	16
A7	Wire 2 Pair 7 (Violet)	3
B7	Wire 1 Pair8(White / Black)	17
A8	Wire 2 Pair 8 (White)	4
B8	Wire 1 Pair9(Gray / Black)	21
A9	Wire 2 Pair 9 (Gray)	8
B9	Wire 1 Pair10(Pink / Black)	20
A10	Wire 2 Pair 10 (Pink)	7
B10	Wire 1 Pair11(LightBlue / Black)	19
A11	Wire 2 Pair 11 (Light Blue)	6
B11	Wire 1 Pair12(LightGreen / Black)	18
A12	Wire 2 Pair 12 (Light Green)	5
B12	Drain Wire (Shield)	13

* Same as Terminal Block Terminal Number.

Table C-2. I/O Cable Wire Coding for IC693CBL317 and IC693CBL320

I/O Connector Pin Number	Cable Wire Color Codes	25-Pin Connector Terminal Number[1]
no connection	Wire 1 Pair1(Brown / Black)	25
A1	Wire 2 Pair 1 (Brown)	12
B1	Wire 1 Pair2(Red / Black)	24
A2	Wire 2 Pair 2 (Red)	11
B2	Wire 1 Pair3(Orange / Black)	23
A3	Wire 2 Pair 3 (Orange)	10
B3	Wire 1 Pair 4 (Yellow / Black)	22
A4	Wire 2 Pair 4 (Yellow)	9
B4	Wire 1 Pair5(Green / Black)	15
A5	Wire 2 Pair 5 (Green)	2
B5	Wire 1 Pair6(Blue / Black)	14
A6	Wire 2 Pair 6 (Blue)	1
B6	Wire 1 Pair 7 (Violet / Black)	16
A7	Wire 2 Pair 7 (Violet)	3
B7	Wire 1 Pair8(White / Black)	17
A8	Wire 2 Pair 8 (White)	4
B8	Wire 1 Pair9(Gray / Black)	21
A9	Wire 2 Pair 9 (Gray)	8
B9	Wire 1 Pair10(Pink / Black)	20
A10	Wire 2 Pair 10 (Pink)	7
B10	Wire 1 Pair11(LightBlue / Black)	19
A11	Wire 2 Pair 11 (Light Blue)	6
B11	Wire 1 Pair12(LightGreen / Black)	18
A12	Wire 2 Pair 12 (Light Green)	5
External Ring Terminal	Drain Wire (Shield)[2]	13

1 Same as Terminal Block Terminal Number.
2 16 gauge wire, green w / yellow tracer. 8” length (from back of connector), terminates with a #10 ring terminal.

The 24-pin I / O connector itself (which mates with the I / O Connector on the faceplate of the PM-APM) is available as three types and can be ordered as an accessory kit as listed below. Three types of connectors are available; solder pin, crimp pin, and ribbon cable. *Each accessory kit contains enough components (D-connectors, backshells, contact pins, etc.) to assemble ten single-ended cables of the type specified for each kit.*

IC693ACC316	FCN–361J024–A U	Solder eyelet receptacle
	FCN–360C024–B	Backshell (for above)
IC693ACC317	FCN–363J024	Crimp wire receptacle
	FCN–363J–A U	Crimp pin (for above; 24 needed)
	FCN–360C024-B	Backshell (for above)
IC693ACC318	FCN–367J024- AUF	IDC (ribbon) receptacle - closed cover
	FCN–367J024- AUH	IDC (ribbon) receptacle - open cover

Note that additional tools from Fujitsu are required to properly assemble the crimped contact and ribbon cable type connectors. The solder eyelet connectors (as provided in IC693ACC316) do not require any special tooling.

Crimped Contact Connectors (as provided in IC693ACC317) require :

Hand Crimping Tool	FCN-363T-T005 / H
Contact Extraction Tool	FCN-360T-T001 / H

Ribbon Cable Connectors (as provided in IC693ACC318) require :

Cable Cutter	FCN-707T-T001 / H
Hand Press	FCN-707T-T101 / H
Locator Plate	FCN-367T-T012 / H

These tools can be ordered from an authorized Fujitsu distributor. Three of the largest US distributors for Fujitsu connectors are Marshall at (800)522-0084, Milgray at (800)MILGRAY, and Vantage at (800)843-0707. If none of these distributors service your area, then contact Fujitsu Microelectronics in San Jose, California, USA via telephone at (408) 922-9000 or via fax at (408) 954-0616 for further information.

It is recommended that you order any necessary connector tooling with sufficient lead time to meet your assembly requirements for these connectors. These tools are generally not stock items and can have significant lead times from distribution. If you have any further questions about this issue, please feel free to contact the GE Fanuc PLC Hotline at 1-800-828-5747 or 804-978-5747.

Appendix D
Wiring to SS-90 Drives

When the Power Mate APM is used with the SS-90 Servo (Digital 100/Digital 200) use the connection diagram below. The pin numbers shown correspond to the 25 pin female connector of the IC693CBL311 and also the terminals on the recommended terminal block. Machine inputs such as Overtravel and Home Switches are not shown in this diagram. Refer to Chapter 2, *Installing the Power Mate APM* for connection diagrams.

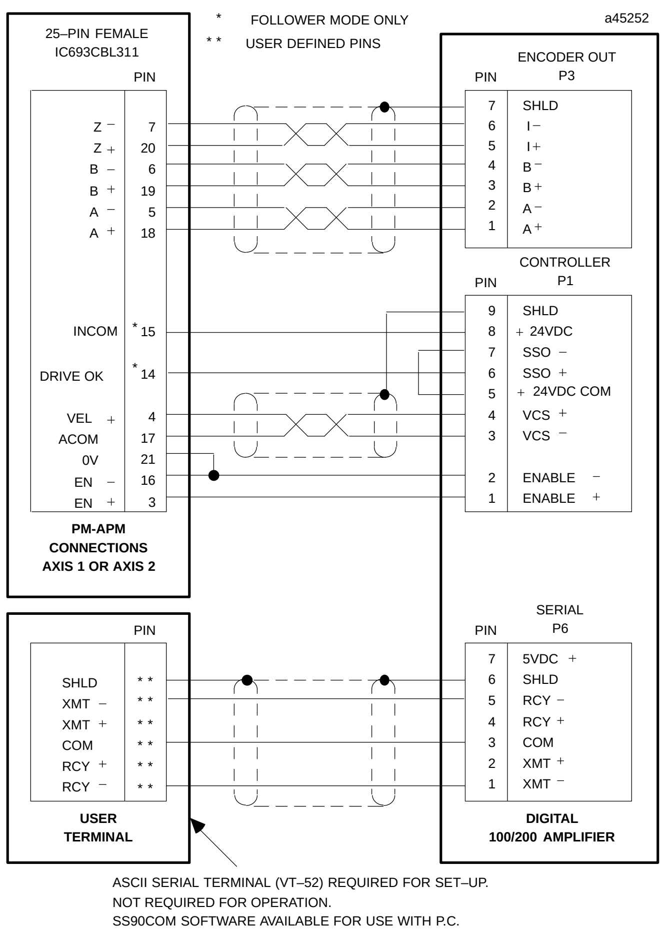


Figure D-1. SS-90 (Digital 100/Digital 200) Wiring

Note that the FAC+ and FAC– pins on the SS–90, as well as the RAC+ and RAC– pins, MUST be closed for the SS–90 to work.

Refer to the following documents for more information:

GFT–106 – SS-90 Servo System Product Selection Guide

GFK–0776 – Digital 100/Digital 200 Brushless Servo Amplifier Instruction Manual

GFK–0777 – Analog 100 Brushless Servo Amplifier Instruction Manual

Appendix E

Ordering Information

Series 90-30 Power Mate APM	IC693APU301(1Axis) IC693APU302(2Axis)
Motion Programmer Software	IC641SWP065
I/O Cable, 10 feet (3 meters) I/O Cable, 3 feet (1 meter) I/O Cable, 10 feet (3 meters) w/external shield wire I/O Cable, 3 feet (1 meter) w/external shield wire	IC693CBL311 IC693CBL319 (Consists of a 24-pin I/O connector, a cable, and a 25-pin, D-type terminal block connector). IC693CBL317 IC693CBL320 (Consists of a 24-pin I/O connector, a cable, and a 25-pin, D-type terminal block connector– also has an 8” external shield pigtail)
Terminal Block	Weidmuller RD25 910648 or equivalent (must be consistent with I/O cable IC693CBL311)
24-pin I/O Connector Accessory Kits (three types available, each kit has enough components to assemble ten single-ended cables)	IC693ACC316 (solder eyelet receptacle) IC693ACC317 (crimp wire receptacle) IC693ACC318 (IDC (Ribbon) receptacle)
Serial Communications Miniconverter Kit	IC690ACC901

Appendix F LDT Interface

TEMPOSONICS® II or LDT Interface to Power Mate APM

Release 2.10 of the Series 90-30 Power Mate APM, catalog numbers IC693APU301 / 302 firmware includes an LDT interface (or LINEAR feedback) option. The LINEAR feedback option is for the STANDARD control loop and will not work in FOLLOWER mode. This appendix explains what an LDT interface is and how to specify one for use with the Series 90-30 Power Mate APM.

TEMPOSONICS II is a brand name for a device more commonly known as a linear displacement magnetostrictive transducer (LDT). An LDT is a sealed stainless steel tube constructed to contain a waveguide. A conductor is threaded coaxially through the waveguide. Electronics at one end of the waveguide tube transmit accurately timed interrogation pulses down the conductor and measure the amount of time required for the interrogation pulse response to return through the waveguide. The effective length of the tube is either the end of the tube or the time required for the transmitted pulse to be reflected back to the electronics package.

A simple method of electronically making the tube length shorter is to surround the waveguide tube with a magnetic field of sufficient intensity (magnetostrictive) to reflect the transmitted pulse to the electronic detection circuitry. Permanent magnets, mounted to a moveable collar or slide are commonly used for this purpose. As the magnet is moved along the length of the tube, the transmitted pulse is reflected back to the electronics with a time delay proportional to the magnet position. The LDT electronics can now accurately generate an output that is proportional to the absolute position of the magnet along the rod or waveguide tube.

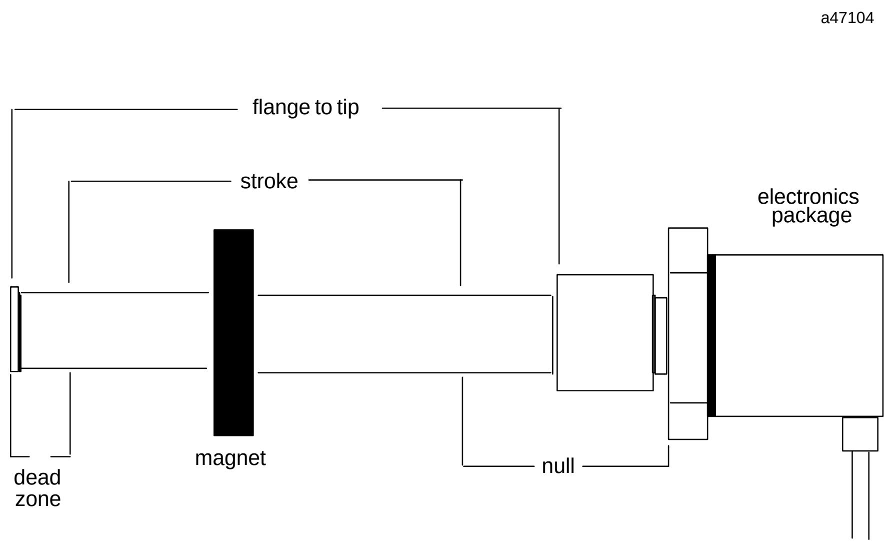


Figure F-1. Example of a Linear Displacement Magnetostrictive Transducer (LDT)

LDTs have been used successfully in motion applications for more than fifteen years. The LDT has many desirable characteristics; absolute position sensing, rugged sealed construction for use in harsh environments and non-contact measurement. These devices are typically used in linear motions of relatively short length. Measuring hydraulic cylinder stroke length is one application where LDTs are often specified. Custom LDTs may be ordered for certain radial motion applications. There is a large selection of mounting options, output types and resolutions available when ordering LDTs. The magnet of a vertically positioned LDT may even be mounted in a floatation device to measure liquid level.

LDTs used to provide feedback to the Series 90-30 Power Mate APM must be specified as TEMPOSONICS II, DPM (Digital Personality Module), with Internal Interrogation or compatible. Additionally an external power supply may be required to power the LDT electronics.

Accuracy, Resolution and Response Considerations.

To specify and use a particular LDT for an application requires an understanding of some terminology particular to this type of transducer. The DPM option specified above will convert the LDT pulse duration to a differential TTL (RS422 line driver) signal. The logical *ON* duration of the differential TTL signal is proportional to the position of the magnet along the *rod* of the transducer. The Power Mate APM measures the *ON* time of the LDT output and converts the duration of the signal to a position or distance.

Physical Length

The LDT you specify as a certain stroke length will actually be several inches longer than you need. The waveguide portion of the rod has a *Dead Zone* (2.5 inches nominal for TEMPOSONICS II) at the tip and a *Null* zone (nominal 2 inches for TEMPOSONICS II) from the mounting flange end. The additional length of the rod may be a mechanical or mounting consideration.

Resolution: The smallest increment of a stroke that can be detected and indicated on the output of the LDT. The resolution is also limited to the smallest signal interval that can accurately be detected by the Series 90-30 Power Mate APM. *The Series 90-30 Power Mate APM (APU301 or APU302) will sample the LDT interface at a 16.896 MHz rate, yielding a 59.186 nanosecond counter resolution.*

Gradient: The rate at which pulses propagate through the waveguide of the LDT, this is approximately 9.05 microseconds per inch (0.356 microseconds per millimeter) for TEMPOSONICS II, but will vary slightly from transducer to transducer. The gradient for any particular transducer will be indicated on the label affixed to the transducer. The nominal gradient for a Balluff LDT is 8.90 microseconds per inch.

Recirculations (circulations): A process that improves the resolution of a LDT system. The on time of the LDT pulse duration signal (differential TTL signal from the DPM) is multiplied by a specified factor of 1-128. This provides more counting time for the Power Mate APM, improving resolution. The LDT response slows in direct proportion to the number of recirculations The recirculation value for the LDT is specified when ordering a particular LDT or may be adjustable within the LDT. The following formula will allow you to calculate the number of recirculations to specify when ordering a particular LDT.

$$\text{Recirculations} = \frac{1}{(G)(F)(R)}$$

Where:

G = Gradient (approximately 9.05ms per inch for TEMPOSONICS II)

F = Crystal frequency of the counter (16.896 MHz for Series 90-30 Power Mate APM, APU301/302).

R = Resolution desired in inches.

$$\text{Example: } \frac{1}{(9.05\text{msec/in})(16.896\text{MHz})(0.01\text{in})} = 0.654 \text{ (round up to 1)}$$

Update Time: Update time is defined as the time interval between maximum update (refresh) of the LDT output. With the Series 90-30 Power Mate APM crystal frequency and resolution fixed, the update time is proportional to the number of recirculations and the total length of the LDT. The update time for the LDT is calculated using the formula below.

$$\text{Update Time (milliseconds)} = \frac{(DZ + S + N)(C)(G)}{1000}$$

Where:

DZ = *D*ead Zone of the waveguide (2.5 inches for TEMPOSONICS II)

S = Stroke Length

N = *Null* of the waveguide (2.0 inches for TEMPOSONICS II)

C = Number of Recirculations

G = Gradient (approximately 9.05ms per inch for TEMPOSONICS II)

$$\text{Example: } \frac{2.5\text{in} + 24\text{in} + 2\text{in})(9.05\text{ms / in})}{1000} = 0.258 \text{ ms}$$

Counts per inch: The actual counts per inch resolution of a given LDT can be calculated with the formula;

Rt = (C)(G)(F)

Where:

Rt = Resolution of the LDT in counts per inch

C = Number of circulations specified

G = Gradient of the sensor in msec/inch

F = Frequency of the counter in MHz (16.896 MHz for APU301/302)

Example: Rt = (3 recirc.)(9.05msec/in)(16.896 MHz) = 458 ct/in.

Temperature coefficient: Expressed as PPM / C or F, is the degree to which the output signal (and therefore the indicated position) is affected by ambient temperature changes. For example, a given LDT with less than or equal 40 PPM / C temperature coefficient at 20C ambient, outputs a value for 6 inches. Should the ambient temperature increase 10C the same LDT at the same position would indicate a position change of less than or equal 0.0024 inches (6*0.00004). The temperature induced offset remains repeatable to within indicated resolution at the new constant temperature.

Non-Linearity, Hysteresis and Repeatability: These factors may also be a consideration to overall system performance. Non-linearity is basically the deviation in full scale travel from one end of the rod to the other. Hysteresis is the difference in indicated position when the same point on the rod is reached from two different directions. Repeatability refers to the difference in position indicated when the same point on the rod is reached, from the same direction, with different moves.

Table F-1. Sources for LDTs

Name of Company	Phone	Type of LDT to Specify
MTS Systems Corporation Research Triangle Park, NC	919 677-0100	TEMPOSONICS II (DPM, Internal Interrogate)
GEMCO Clawson, MI	810 435-0700	Quick - Stik II Series 951VP (Variable Pulse, Internal)
Balluff Florence, KY	606 727-2200	BTL-2-R1-xxxx-x-xxx-Ix (pulse width recirc, Internal)

How to select an LDT for interface to the Power Mate APM (APU301/302)

Begin with the known requirements for the application:

D. required stroke length

E. resolution needed. (divide this value by two to increase output stability)

F. update time required. (less than 3.6 milliseconds preferred for Power Mate APM)

Example: An application has a required stroke length of 24 inches. The desired resolution is 0.01inches (divided by two to increase stability = 0.005inches). It is necessary to calculate the LDT update time from the first two variables.

1. Divide the stroke length by the desired resolution to calculate the minimum number of counts required.

   Example: 24in Ï 0.005in / count =4,800 counts

Note: This result must not exceed the 16 bit count (65,535) ability of the Series 90-30 Power Mate APM linear counter input. Use a lower resolution if necessary.

2. Calculate number of recirculations from the formula: Recirculations = $\frac{1}{(G)(F)(R)}$

   Example: 1 Ï (9.05msec/in * 16.896 MHz * 0.005 in) = 1.3 (round up to 2).

Note: Round the result of the recirculation calculation up to the next integer.

3. Calculate LDT update time from the formula:

$$\text{Update (milliseconds)} = \frac{(DZ + S + N)(C)(G)}{1000}$$

Example: ((2.5in + 24in + 2in) * 2 * 9.05msec / in)Ï 1000 = 0.516 milliseconds

Note:. The update time of the LDT should not be greater than 3.6 milliseconds. Consider a lower resolution or lower value for recirculations if necessary.

4. Determine LDT specifications.

Example:

The specified LDT stroke is 24 inches

The specified number of recirculations = 2

The Power Mate APM counter frequency is 16.896 MHz

The Power Mate APM clock divider = 1

Note: When ordering an LDT specify stroke and recirculations. The manufacturer of the LDT will adjust the sensor to specification. The actual resolution obtained is based on the gradient of the specific sensor.

5. The Logicmaster 90-30 configuration of the Power Mate APM, LINEAR feedback, will make an on-line calculator available to you. Zoom into the APU module and select **Fdback Type = LINEAR**. Page down into the axis dependent configuration. Observe the following fields;

Clk Divider: Always 1 in current releases.

Recircs: Number of recirculations.

Gradient: The gradient specified for the LDT

MaxLen (in): The maximum stroke available.

Counts / Inch: The LDT resolution.

The LogicMaster configuration will calculate the resolution and maximum stroke length automatically. Enter a value for **Recircs**: and if necessary change **Gradient:** to calculate **MaxLen (in):** and **Counts/Inch:**.

6. Another method of calculating the counts per inch is to use the formula
Rt = (C)(G)(F)

Example: Rt = 2 recirculations * 9.05msec / in * 16.896 MHz = 305.8

Note: Truncate the value for Rt to the integer. In the above example truncate 305.8 down to 305 counts/inch. Logicmaster will calculate this value automatically. This example is included to illustrate how the counts/inch value is achieved.

TEMPOSONICS II is a registered trademark of MTS Systems Corporation

Quick-Stik II is a trademark of GEMCO, Inc.

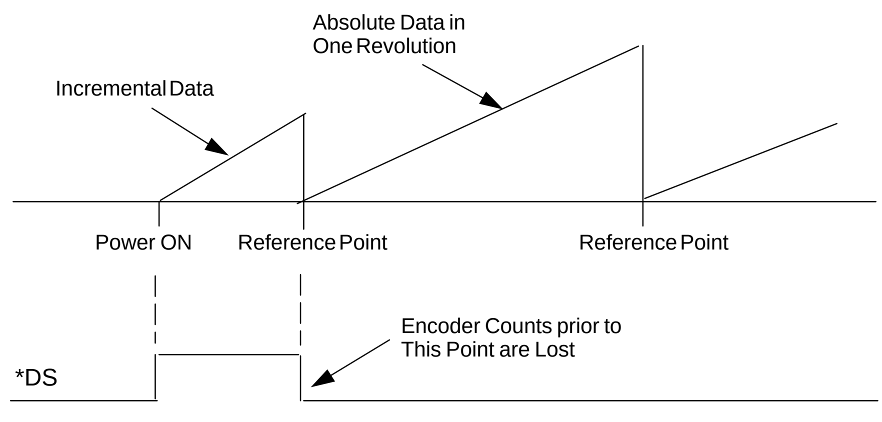
Appendix G Serial Encoders

Serial Encoder Models

There are three pulse encoder models that will function with the Power Mate APM, 8K, 32K or 64K counts per revolution. All encoders return position information in the form of a 32 bit value. To maintain equal number of counts (scaling) per motor revolution, the 8K encoder position increments its absolute position word data by eight counts, whereas the 32K encoder increments its data by two counts, and the 64K encoder increments its data by one count. However, the Power Mate APM overrides these additional count inputs; thus, 8K, 32K and 64K encoders are identical for a Power Mate APM. **The Power Mate APM will always sense 8192 counts per motor shaft revolution from a serial encoder.** The one million count (1M) encoder is not supported.

Serial Encoder - First Time Use or Use After Loss of Encoder Battery Power

The encoder temporarily provides incremental data during the first use or after restoring encoder battery power. The incremental data is lost when motor shaft rotation causes the encoder to pass a reference point (similar to a marker signal) within one revolution of the motor shaft. The diagram below describes this situation, as well as the state of the *DS internal signal. The signal *DS is not directly available to the user but serves to illustrate the sequence of events. The *DS signal indicates that absolute position data is available. If no battery pack is used, *DS will be reset on each power cycle.



Note

The a Series digital encoder must be rotated up to one full revolution after the absolute mode battery has been reattached. Within one revolution the encoder will reference itself and report a referenced status (*DS = 0) to the Power Mate APM. The encoder will set it's internal counts accumulator to zero at the point where *DS switches to a logic zero. Counts accumulated prior to the *DS transition are lost.

Serial Encoder Modes

The a Series pulse encoders can be operated in either Incremental mode or Absolute mode. The Incremental mode is selected by setting the Power Mate APM's Logicmaster 90-30 configuration field *Intgr Mode* (Integrator Mode) to *OFF* or to *IN ZONE*. Absolute mode is selected by setting the Power Mate APM's Logicmaster 90-30 configuration field *Intgr Mode* (Integrator Mode) to *CONTINU*. Additionally, proper operation of the Absolute mode requires an external battery pack that must be connected to the appropriate digital servo amplifier connector. Refer to the appropriate amplifier manual for selection and installation of the battery pack.

Note

The Position Error Integrator is not used when the Power Mate APM is used to control digital servos. Therefore the *Intgr Mode* (Integrator Mode) configuration is used for the alternate purpose of selecting Incremental or Absolute operating mode of the a Series pulse encoders.

Limitations on Total Travel for EOT Mode

The maximum value that can be used for the *+EOT* is +8,388,607 user units. User units are configured as a ratio to the encoder count value and are bound to the range of 8:1 to 1:32. Additionally, the feedback from a digital servo is fixed at 8192 counts per revolution. Using a simple formula will determine the maximum number of motor shaft revolutions possible for axis travel when EOT (non rotary) mode is used.

$$\textit{Maximum Revolutions} = 1024 * \frac{\text{Counts}}{\text{User Units}}$$

Using the above calculation, the maximum possible move is realized with the User Unit to Count ratio at the maximum 1:32 resulting in a maximum travel of +/– 32,768 motor shaft revolutions. (The maximum travel originates at the *Actual Position* zero and can move the number of maximum travel revolutions plus or minus of the zero position.)

There is no restriction on maximum travel for Rotary mode. refer to *Important Configuration Considerations* in Chapter 2 for additional information on EOT Mode and Rotary Mode.

Incremental Encoder Mode Considerations

The digital serial encoder can be used as an incremental encoder returning 8192 counts per shaft revolution, with no revolution counts retained through a power cycle. The equivalent of a *marker* pulse will occur once each motor shaft revolution. All Home modes (Home Switch, Move+, Move– and Set Position) reference the axis, and set the *Position Valid* bit upon successful completion. High count limits and Low Count limits are valid and the A*ctual Position* as reported by the Power Mate APM module will wrap from high to low count or from low to high count values. This is an excellent mode for rotary applications that will always operate via incremental moves, in the same direction. *Home Offset* and *Home Position* configuration items allow simple referencing to the desired location. Incremental mode is selected by setting the Power Mate APM"s Logicmaster 90-30 configuration field *Intgr Mode* (Integrator Mode) to *OFF* or to *IN ZONE*.

Absolute Encoder Mode Considerations

The a Series serial pulse encoder can be used as an absolute type encoder by adding a battery pack to retain servo position while system power is off. A Find Home cycle or Set position must be performed initially or whenever encoder battery power is lost with the servo amplifier also in a powered down state. Absolute Encoder mode must be configured in the Power Mate APM module to function properly. Absolute mode is selected by setting the Power Mate APM's Logicmaster 90-30 configuration field *Intgr Mode* (Integrator Mode) to *CONTINU*.

Absolute Encoder Mode - Position Initialization

When a system is first powered up in Absolute Encoder mode, a position offset for the encoder must be established. This can be done by using the Find Home cycle or the Set Position command.

Find Home Cycle - Absolute Encoder Mode

The Find Home cycle mode can be configured for Move +, Move – or Home Switch operation. Refer to Chapter 4 for additional details of Home Cycle operation. The Home Offset and Home Position configuration items function the same as in Incremental Encoder mode. At the completion of the Home Cycle, Actual Position is set to the configured Home Position value. The Power Mate APM internally calculates the encoder Absolute Feedback Offset needed to produce the configured Home Position at the completion of the Home Cycle. This Absolute Feedback Offset must be permanently saved in the Power Mate APM by sending %AQ command 4Ah *Update Flash Memory* after completion of the Home Cycle.

Once the Absolute Feedback Offset is saved in Flash memory, the Power Mate APM will automatically initialize Actual Position after a power cycle and set the Position Valid %I bit.

Set Position Command - Absolute Encoder Mode

The %AQ Set Position command functions the same as in Incremental Encoder mode. At the completion of the Set Position operation, Actual Position is set to the Set Position value. The Power Mate APM internally calculates the encoder Absolute Feedback Offset needed to produce the commanded Set Position value. This Absolute Feedback Offset must be permanently saved in the Power Mate APM by sending %AQ command 4Ah *Update Flash Memory* after the Set Position command.

Once the Absolute Feedback Offset is saved in Flash memory, the Power Mate APM will automatically initialize Actual Position after a power cycle and set the Position Valid %I bit.

Absolute Encoder Mode - Power Mate APM Power-Up

The battery pack attached to the servo subsystem will maintain power to the encoder counter logic. Once the encoder has referenced through first time start up, the actual position is automatically maintained by the encoder, even if the axis is moved during servo power loss. The encoder will monitor the status of the battery pack, and report loss of battery power or low battery power to the Power Mate APM.

The Power Mate APM will complete a power-on diagnostic, and when configured for absolute encoder mode, interrogate the referenced status of the serial encoder. A valid referenced status from the encoder will signal the Power Mate APM to read the encoder absolute position. The Power Mate APM will report the *Actual Position* as the sum of the encoder position and the A*bsolute Feedback* Offset established by the initial Find Home cycle or Set Position command.

Absolute Encoder Mode with Rotary Mode

Some restrictions are necessary when Absolute Encoder mode is selected along with Rotary mode.

Absolute Encoder mode causes the Power Mate APM to automatically initialize Actual Position from the battery backed absolute encoder after a power cycle. Absolute Encoder mode is selected by setting the Intgr Mode to CONTINU in the Configuration software.

Rotary mode allows the Power Mate APM to create continuous motion in one direction by the use of multiple CMOVE or PMOVE incremental commands. As the axis moves, Actual Position will reach a Hi or Low Count Limit then roll over to the other limit. The distance between Actual Position rollovers is the Rotary Count Modulus. Rotary Mode is selected by configuring the +EOT Limit >= Hi Count Limit and –EOT Limit <= Low Count Limit.

The battery backed absolute encoder has a total absolute counting range of (8192 cts / rev) x (32767 revs) = Ç 268,427,264 counts. This is the maximum number of counts the encoder can move after a position reference operation (Find Home or Set Position) and still retain absolute position. If the encoder rotates more than this number of counts after a position reference operation, the absolute position is lost and the encoder starts a new counting cycle.

Restrictions when Absolute Encoder Mode is used with Rotary Mode

1. If the Rotary Count Modulus (in counts) is a power of 2, then no restrictions exist on rotary travel. This means the distance defined as ((Hi Limit – Low Limit) + 1) in counts must be a number which is a power of 2 such as 128, 256, 512 ... 8192, 16384, and so forth. Under this condition the Power Mate APM will always initialize Actual Position to the correct rotary position after a power cycle.

2. If the power of 2 condition for Rotary Count Modulus is not met, the rotary motion must be limited to a range of + / – 268,427,264 counts after a Set Position or Find Home position reference operation. This restriction can be handled in some systems by periodically performing a Set Position when the axis is stopped and holding a known rotary position.

Note

Performing a Set Position or Find Home cycle in Absolute Encoder mode causes the Power Mate APM to recalculate the encoder Absolute Feedback Offset. An Update Flash Memory command must always be sent to permanently save the new Absolute Feedback Offset. The flash memory in the Power Mate APM has a limited number of update cycles (20,000 – 100,000 cycles). This limitation should be considered if a system design required the Absolute Feedback Offset to be saved more than 10 times per day.

Symbols

A

M

N

O

P

R

S

T