

---

**PSO-PC DSP-BASED  
PROGRAMMABLE OUTPUT CONTROL  
CARD  
OPERATION & TECHNICAL MANUAL**

**P/N: EDO105 (V1.5)**

---



AEROTECH, Inc. • 101 Zeta Drive • Pittsburgh, PA. 15238-2897 • USA  
Phone (412) 963-7470 • Fax (412) 963-7459  
Product Service: (412) 967-6440; (412) 967-6870 (Fax)

**[www.aerotech.com](http://www.aerotech.com)**

If you should have any questions about the PSO-PC Control Card or comments regarding the documentation, please refer to Aerotech online at:

**<http://www.aerotech.com>.**

For your convenience, a product registration form is available at our web site.

Our web site is continually updated with new product information, updated manuals, free downloadable software and special pricing on selected products.

UNIDEX 500 and 600 are products of Aerotech, Inc.

Borland C is a product of Borland International, Inc.

IBM PC/AT bus is a registered trademark of International Business Machines, Inc.

Inductosyn is a registered trademark of Farrand Industries, Inc.

iSBX and Multimodule are registered trademarks of Intel Corporation.

MS-DOS, Microsoft Diagnostics and Windows are products of Microsoft Corporation

Windows NT and Windows 95 are products of Microsoft.

#### The PSO-PC DSP-based Programmable Output Control Card Operations and Technical Manual Revision History:

Rev 0.1	June 1, 1994
Rev 0.2	June 8, 1994
Rev 1.1	September 22, 1995
Rev 1.3	February 4, 1998
Rev 1.4	February 4, 2000
Rev 1.5	August 2, 2001

## TABLE OF CONTENTS

<b>CHAPTER 1: OVERVIEW</b> .....	1-1
1.1. Introduction .....	1-1
1.2. PSO-PC Overview.....	1-3
1.3. Programmed Output Modes.....	1-9
1.4. Safety Procedures and Warnings .....	1-10
<b>CHAPTER 2: GETTING STARTED</b> .....	2-1
2.1. Introduction .....	2-1
2.2. Unpacking and Inspecting the PSO-PC .....	2-1
2.3. PSO-PC Board Jumpers and Configurations .....	2-2
2.3.1. iSBX MINTR0/MINTR1 Request Jumper (JP1) .....	2-3
2.3.2. Host Base Address Jumpers (JP2-JP7).....	2-3
2.3.3. DSP-to-PC Host Interrupt Enable/Disable Jumper (JP8).....	2-5
2.3.4. SRDY_N PC Bus Timing Enable/Disable Jumper (JP9).....	2-5
2.3.5. DPRAM IRQ Jumpers (JP10-JP15).....	2-6
2.3.6. DPRAM Base Address Jumpers (JP16-JP23) .....	2-7
2.3.7. Laser Output Polarity Jumpers (JP24-JP26).....	2-8
2.3.8. Connect/Disconnect LASEN Input to DSP Interrupt (JP27).....	2-12
2.3.9. Enable/Disable Laser Enable Input LASEN (JP28).....	2-12
2.3.10. MINTR0/MINTR1 Selection Jumper (JP29).....	2-13
2.3.11. Encoder Data Format Selection Jumpers (JP30 through JP33).....	2-14
2.3.12. Global Encoder Direction Jumper (JP34) .....	2-15
2.3.13. Encoder Type Jumpers (JP35-JP42) .....	2-15
2.3.14. Watchdog Circuit Enable/Disable Jumper (JP43).....	2-17
2.3.15. Host Reset Capability Enable/Disable Jumper (JP44).....	2-17
2.3.16. Encoder Sampling Frequency Jumpers (JP45-JP48, JP52) .....	2-18
2.3.17. Internal/External Power Selection Jumpers (JP49- JP51) .....	2-20
2.4. Installing the PSO-PC Board.....	2-22
2.5. Connecting to the Optional Opto-22 I/O Port (P5).....	2-23
<b>CHAPTER 3: PROGRAMMING</b> .....	3-1
3.1. Introduction .....	3-1
3.2. Programming Commands .....	3-3
3.3. Conditional Tracking Based on Input States PSOC .....	3-4
3.3.1. MODE Arguments for PSOC.....	3-4
3.3.1.1. Enable Position Tracking Unconditionally PSOC,0 .....	3-4
3.3.1.2. Position Tracking Dependant on Input PSOC,1 .....	3-4
3.3.1.3. Position Tracking Dependent on Input (Position Counter set to 0) PSOC,2 .....	3-4

	3.3.1.4. Position Tracking Dependent on I/O Map	
	PSOC,3 .....	3-5
	3.3.1.5. Low, High Command PSOC,4 .....	3-5
3.3.2.	PSOC Arguments .....	3-5
	3.3.2.1. <i>i</i> Argument .....	3-5
	3.3.2.2. <i>n</i> Argument .....	3-5
	3.3.2.3. <i>in_map</i> Argument.....	3-5
	3.3.2.4. <i>out_map</i> Argument.....	3-6
3.4.	Position Synchronized Output Firing Distance Entry PSOD .....	3-7
3.4.1.	Mode Arguments for PSOD .....	3-7
	3.4.1.1. Pulse Output at a Constant Incremental	
	Distance PSOD,0 .....	3-7
	3.4.1.2. Pulse Output at Incremental Distances	
	Defined in Variables PSOD,1 .....	3-8
	3.4.1.3. Pulse Output at Absolute Distances Defined	
	in Variables PSOD,2 .....	3-8
3.4.2.	PSOD Arguments .....	3-10
	3.4.2.1. <i>distance</i> Argument .....	3-10
	3.4.2.2. <i>V#</i> Argument .....	3-10
	3.4.2.3. <i>±m</i> Argument .....	3-10
3.5.	Enable/Disable Position Synchronized Output Firing PSOF .....	3-12
3.5.1.	Mode Arguments for PSOF .....	3-12
	3.5.1.1. Disable Laser Output Pulse PSOF,0 .....	3-12
	3.5.1.2. Laser Output Fires Continuously PSOF,1 .....	3-12
	3.5.1.3. Laser Output Fires a Specified Number of	
	Times PSOF,2 .....	3-13
	3.5.1.4. Laser Output Synchronized with Position	
	PSOF,3 .....	3-13
	3.5.1.5. Laser Output According to Bitmap w/	
	Toggle Pulse PSOF,4 .....	3-14
	3.5.1.6. Laser Output According to Bitmap with	
	User-Defined Pulse PSOF,5 .....	3-15
3.5.2.	PSOF Arguments .....	3-17
	3.5.2.1. <i>num</i> Argument.....	3-17
	3.5.2.2. <i>axis#</i> Argument .....	3-17
	3.5.2.3. <i>±dist</i> Argument.....	3-17
3.6.	Position Synchronized Output Using Bit Mapping PSOM .....	3-19
3.6.1.	<i>var#</i> Argument .....	3-19
3.6.2.	<i>±size</i> Argument.....	3-19
3.7.	Position Synchronized Output Pulse Configuration PSOP .....	3-20
3.7.1.	Mode Arguments for PSOP .....	3-20
	3.7.1.1. Simple Single Pulse PSOP,0 .....	3-20
	3.7.1.2. Single Pulse with Lead, Width and Tail	
	Characteristics PSOP,1 .....	3-21
	3.7.1.3. Ramping Pulse Train PSOP,2 .....	3-22
	3.7.1.4. User-Defined Pulse Train PSOP,3 .....	3-22
	3.7.1.5. Simple One-Shot Pulse PSOP,4 .....	3-23
	3.7.1.6. Toggle Pulse PSOP,5 .....	3-23
3.7.2.	PSOP Arguments.....	3-24
	3.7.2.1. <i>l</i> Argument .....	3-24
	3.7.2.2. <i>w</i> Argument.....	3-24

3.7.2.3.	<i>t</i> Argument .....	3-24
3.7.2.4.	<i>r</i> Argument.....	3-24
3.7.2.5.	<i>g</i> Argument .....	3-24
3.7.2.6.	<i>var#</i> Argument .....	3-24
3.7.2.7.	<i>±n</i> Argument .....	3-25
3.8.	Position Synchronized Output with Real-time Control PSOR.....	3-26
3.8.1.	MODE Arguments for PSOR.....	3-26
3.8.1.1.	PSOR,0 .....	3-26
3.8.1.2.	PSOR,1 .....	3-26
3.8.1.3.	PSOR,3 .....	3-26
3.9.	Digital/Analog Output Command PSOT .....	3-27
3.9.1.	MODE Arguments for PSOT .....	3-27
3.9.1.1.	Set Individual Output States PSOT,0.....	3-27
3.9.1.2.	Set a Group of Output States PSOT,1.....	3-27
3.9.1.3.	Set Analog Outputs to Discrete Values PSOT,2 .....	3-28
3.9.1.4.	Velocity Ramping PSOT,4 .....	3-28
3.9.1.5.	Position Ramping PSOT,6.....	3-30
3.9.2.	PSOT Arguments .....	3-31
3.9.2.1.	<i>bit#</i> Argument .....	3-31
3.9.2.2.	<i>state</i> Argument.....	3-31
3.9.2.3.	<i>states</i> Argument .....	3-31
3.9.2.4.	<i>dac#</i> Argument.....	3-31
3.9.2.5.	<i>voltage</i> Argument.....	3-31
3.9.2.6.	<i>v0</i> Argument.....	3-31
3.9.2.7.	<i>vmax</i> Argument .....	3-32
3.9.2.8.	<i>velocity</i> Argument.....	3-32
3.9.2.9.	<i>position</i> Argument.....	3-32
3.10.	Programming Tools .....	3-33
3.10.1.	<i>aerq_psodlarray</i> .....	3-34
3.10.2.	<i>aer_send</i> .....	3-35
3.10.3.	<i>aer_initialize</i> .....	3-36
<b>CHAPTER 4:</b>	<b>TECHNICAL DETAILS.....</b>	<b>4-1</b>
4.1.	Introduction .....	4-1
4.2.	Specifications and Ratings.....	4-2
4.3.	Main Connector (P1) Pinouts .....	4-3
4.4.	Aerotech Bus (P3) Pinouts .....	4-4
4.5.	Intel iSBX Bus (P4) Pinouts.....	4-5
4.6.	Opto-22 I/O Connector (P5) Pinouts .....	4-6
4.7.	Single-ended Encoder Interface Connector (P6) Pinouts .....	4-7
4.8.	Differential Encoder Interface Connector (P7) Pinouts.....	4-9
4.9.	Test Points .....	4-10
4.10.	Analog Output Gain Adjustments.....	4-11
<b>CHAPTER 5:</b>	<b>TROUBLESHOOTING .....</b>	<b>5-1</b>
5.1.	Introduction .....	5-1
5.2.	Installation Problems .....	5-2
<b>APPENDIX A:</b>	<b>PSO COMMAND SUMMARY .....</b>	<b>A-1</b>
<b>APPENDIX B:</b>	<b>GLOSSARY OF TERMS .....</b>	<b>B-1</b>

**APPENDIX C: WARRANTY AND FIELD SERVICE .....C-1**  
**INDEX**  
**REVISION HISTORY**

▽ ▽ ▽

**LIST OF FIGURES**

Figure 1-1. Key Components of the PSO-PC Board .....1-1

Figure 1-2. Overview of Typical PSO Interfaces .....1-4

Figure 1-3. U500 System Interconnect.....1-5

Figure 1-4. U600 System Interconnect.....1-6

Figure 1-5. Overview of a Sample PSO-PC System Using the UNIDEX  
500.....1-7

Figure 1-6. How PSO Commands Are Routed to the PSO-PC .....1-9

Figure 1-7. PSO Output Signal Examples .....1-9

Figure 2-1. PSO-PC Board Jumpers .....2-2

Figure 2-2. Laser Interface (Using 4N33 Opto-isolators) Showing JP24-  
JP26 .....2-9

Figure 2-3. Laser Interface (Using 6N136 or HCPL-2601 Opto-Isolators)  
Showing JP24-JP26 .....2-10

Figure 2-4. Laser Enable Interface .....2-11

Figure 2-5. Typical Differential and Single-Ended Encoder Inputs .....2-15

Figure 2-6. The Encoder Sampling Process .....2-18

Figure 2-7. Optical Isolation of Analog Outputs .....2-21

Figure 2-8. Sample Connection to the Opto-22 I/O Port (P5).....2-23

Figure 3-1. Trigger Pulse Fired at Constant Increments.....3-7

Figure 3-2. Trigger Pulse Fired at Incremental Distances as Defined in  
Variables.....3-8

Figure 3-3. Trigger Pulse Fired at Absolute Distances as Defined in  
Variables.....3-9

Figure 3-4. Trigger Pulse Fired ON/OFF According to Bitmap .....3-15

Figure 3-5. Trigger Pulse Fired According to Bitmap with User-Defined  
Pulses.....3-16

Figure 3-6. Bit Mapping Scan Patterns for PSOM Commands .....3-19

Figure 3-7. Single Pulse Output .....3-20

Figure 3-8. Single Pulse Output with Lead, Width, and Tail .....3-21

Figure 3-9. Ramping Pulse Train .....3-22

Figure 3-10. Single One-Shot Pulse Output .....3-23

Figure 3-11. Toggle ON/OFF Pulse Defined by Distance .....3-23

Figure 3-12. User-Specified Analog Voltage .....3-28

Figure 3-13. Velocity Ramping .....3-29

Figure 3-14. Position Ramping .....3-30

Figure 4-1. User-Supplied Encoder Interface Cable .....4-7

Figure 4-2. PSO-PC Board.....4-11







**LIST OF TABLES**

Table 1-1.	Ports and Connector Types of the PSO-PC Board .....	1-8
Table 1-2.	I/O Summary of PSO-PC Connector P1 .....	1-8
Table 2-1.	iSBX MINTR0/MINTR1 Request Jumper Settings (JP1).....	2-3
Table 2-2.	Host Base Address Jumper Settings (JP2-JP7).....	2-4
Table 2-3.	DSP-to-host PC Interrupt Enable/Disable Jumper (JP8) .....	2-5
Table 2-4.	SRDY_N PC Bus Timing Enable/Disable Jumper (JP9).....	2-5
Table 2-5.	DPRAM IRQ Jumper Settings (JP10-JP15) .....	2-6
Table 2-6.	DPRAM Base Address Jumper Settings (JP16-JP23) .....	2-7
Table 2-7.	Laser Output Polarity Jumper Settings (JP24-JP26).....	2-8
Table 2-8.	Laser Output Opto-Isolator Specifications .....	2-11
Table 2-9.	Connect/Disconnect LASER Input DSP Interrupt (JP27) .....	2-12
Table 2-10.	Enable/Disable Laser Enable Input LASER (JP28) .....	2-12
Table 2-11.	MINTR0/MINTR1 Jumper Settings (JP29) .....	2-13
Table 2-12.	Encoder Data Format Jumper Settings (JP30-JP33).....	2-14
Table 2-13.	Global Encoder Direction Jumper Setting (JP34).....	2-15
Table 2-14.	Encoder Type Jumper Settings (JP35-JP42).....	2-16
Table 2-15.	Watchdog Circuit Enable/Disable Jumper Setting (JP43) .....	2-17
Table 2-16.	Host Reset Capability Enable/Disable Jumper Setting (JP44).....	2-17
Table 2-17.	Encoder Sampling Frequency Jumper Settings (JP45-JP48 and JP52).....	2-19
Table 2-18.	Internal/External Power Selection Jumper Settings (JP49- JP51).....	2-20
Table 2-19.	Port 1 External Power Connections for AOUT1 and AOUT2.....	2-20
Table 3-1.	Conventions for this Section.....	3-2
Table 3-2.	PSO-PC Programming Commands.....	3-3
Table 3-3.	Distance Calculations for Multiple Axes Using the PSOD Command .....	3-11
Table 4-1.	Main Connector (P1) Pinouts .....	4-3
Table 4-2.	Aerotech Bus (P3) Pinouts .....	4-4
Table 4-3.	Intel iSBX Bus (P4) Pinouts.....	4-5
Table 4-4.	Opto-22 I/O Connector (P5) Pinouts .....	4-6
Table 4-5.	Single-ended Encoder Interface Connector (P6) Pinouts .....	4-8
Table 4-6.	Single-ended Encoder Interface 25-pin D-type Connector .....	4-8
Table 4-7.	Differential Encoder Interface Connector (P7) Pinouts.....	4-9
Table 4-8.	Differential Encoder Interface 25-pin D-type Connector .....	4-9
Table 4-9.	Test Points of the PSO-PC .....	4-10
Table 5-1.	Troubleshooting for Common Installation Problems.....	5-2
Table A-1.	PSO Command Summary .....	A-1
Table R-1.	Revisions .....	R-1





## **PREFACE**

This section provides an overview of topics covered in each of the sections of this manual as well as conventions used in this manual. The PSO-PC Operation & Technical Manual contains information on the following topics:

### **CHAPTER 1: OVERVIEW**

This chapter contains an overview of the PSO-PC card as well as a system diagram. This chapter also contains precautionary notes about installing and using the PSO-PC card.

### **CHAPTER 2: GETTING STARTED**

This chapter contains information to familiarize you with how the PSO-PC may be used. This chapter also contains information about unpacking and inspecting the equipment, minimum hardware requirements, jumper configurations, hardware installation, software installation and software setup.

### **CHAPTER 3: PROGRAMMING**

This chapter supplies information required to understand the PSO-PC programming command set and how it is integrated into the motion control card (for example, the UNIDEX 500). This chapter also includes an in-depth discussion of the seven basic functional command groups and their associated arguments.

### **CHAPTER 4: TECHNICAL DETAILS**

This chapter supplies technical specifications for the PSO-PC such as test points, input and output ratings, pinouts, bus specifications, and others.

### **CHAPTER 5: TROUBLESHOOTING**

This chapter provides a reference tool if problems with the PSO-PC arise.

### **APPENDIX A: PROGRAMMING COMMAND SUMMARY**

Appendix A contains a quick reference of all PSO-PC commands.

### **APPENDIX B: GLOSSARY OF TERMS**

Appendix B contains a list of definitions of terms used in this manual.

### **APPENDIX C: WARRANTY AND FIELD SERVICE**

Appendix C contains the warranty and field service policy for Aerotech products.

### **REVISION HISTORY**

This section lists the changes made for the current revision.

Throughout this manual the following conventions are used:



- Capitalized letters within a command indicate the minimum entry for that command (e.g., DIisable)
- Danger and/or Warning symbols (see left) appear in the outer margins next to important precautions. Failure to observe these precautions could result in serious injury and/or damage to the equipment.
- Keys such as Shift, Ctrl, Alt and Enter are enclosed in brackets (e.g., <Shift>, <Ctrl>, <Alt> and <Enter>) to distinguish them from individual keystrokes.
- Hexadecimal numbers are listed using a preceding "0x" (for example, 0x300, 0x12F, 0x01EA, etc..) to distinguish them from decimal numbers.
- An "x" preceding a parameter number represents the axis number (1, 2, 3 or 4) for the corresponding axis (X, Y, Z or U, respectively). Therefore, parameter x38 (the Primary Feedback Channel), for example, actually corresponds to four distinct parameters:
  - 138 for the Primary Feedback Channel of axis X
  - 238 for the Primary Feedback Channel of axis Y
  - 338 for the Primary Feedback Channel of axis Z, and
  - 438 for the Primary Feedback Channel of axis U.
- The terms <Enter> and <Return> are used interchangeably throughout this document when referring to the keyboard.
- UNIDEX 500, U500, UNIDEX 600, and U600 are used interchangeably throughout this manual.
- This manual uses the symbol "▽ ▽ ▽" to indicate the end of a chapter.

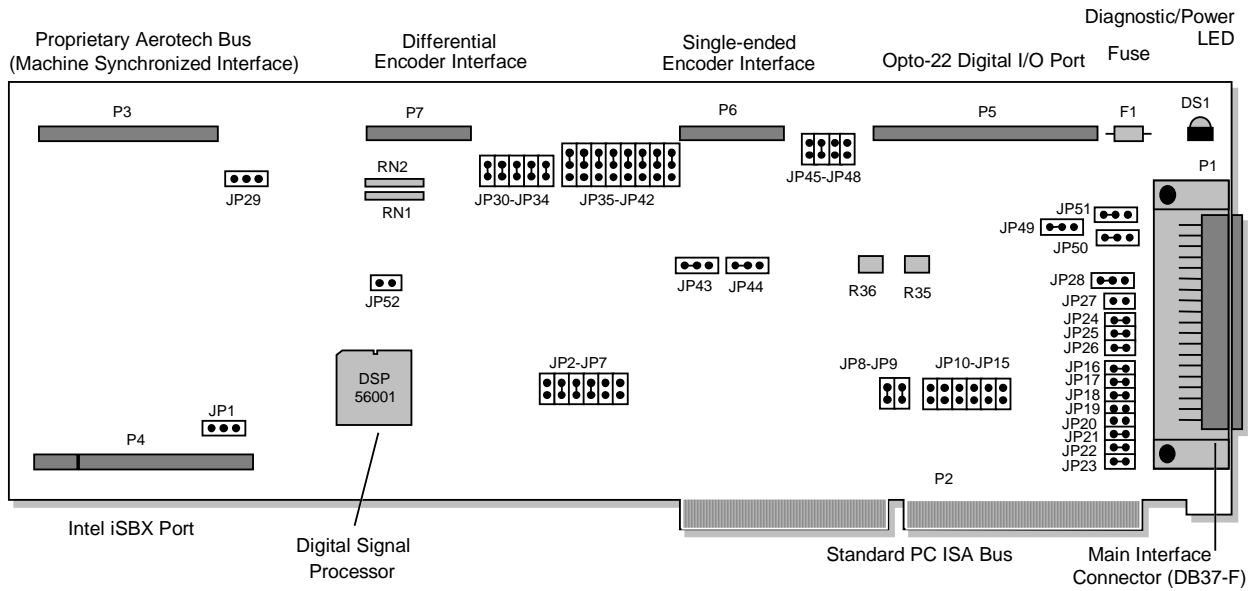
▽ ▽ ▽

## CHAPTER 1: OVERVIEW

<b>In This Section:</b>	
• Introduction .....	1-1
• PSO-PC Overview .....	1-3
• Programmed Output Modes.....	1-9
• Safety Procedures and Warnings .....	1-10

### 1.1. Introduction

The PSO-PC (Programmable Synchronized Output) board is a DSP-based PC card that provides versatile, on-the-fly synchronization of a laser's pulse and power outputs with the motions of any machine commanded by an appropriately equipped Aerotech controller (such as the UNIDEX 600 and in some cases the UNIDEX 500, for example). That is to say, the outputs of the PSO-PC are precisely linked to the axis motion/servo control card. Key components of the PSO-PC card are illustrated in Figure 1-1.



**Figure 1-1. Key Components of the PSO-PC Board**

Using advanced, high-speed, digital signal processing technology, the PSO-PC can acquire and analyze the positions of up to four axes at a maximum input frequency (quad) of 5 MHz (a maximum encoder channel data rate of 20 MHz). PSO-PC can also vector process two or three axes to provide laser control that is exactly synchronized with two- or three-dimensional contoured motions.

In addition to synchronized laser control, the PSO-PC can be used in non-destructive testing applications as well as force/flow control systems. Ultrasonic pulse/echo, through-transmission, eddy current and X-ray inspection systems all benefit from the PSO-PC's ability to provide fixed-pulse spacing along complex contours.

The PSO-PC board is programmable from the axis motion/servo controller (e.g., the UNIDEX 500's Startup program) using a suite of dedicated PSO-PC commands. These commands have the prefix "PSO" and are integrated directly into the motion control program. From there, the PSO commands perform a variety of functions ranging from conditional tracking based on input states to direct control of analog and digital outputs. U600 users refer to the U600 Programming Manual for the PSO programming commands. U500 users refer to Chapter 3: Programming for more information about the PSO commands.

Additional features of the PSO-PC include:

- Variable firing distances
- One, two or three axis synchronization
- Dynamic tracking of any combination of 4 axes
- Laser pulse train outputs programmable in 100 micro second increments
- One shot output with programmable pulse width (in 1 micro second increments)
- Two 16-bit programmable analog outputs
- Analog outputs synchronized to position or velocity
- OPTO-22 compatible I/O port with 16 outputs and 8 inputs
- Optically isolated laser interface with jumper selectable active polarity
- Optically isolated analog outputs, fail safe output and laser defeat input.

## 1.2. PSO-PC Overview

The PSO-PC should be used with a compatible motion control board like the U500 or U600, for example. The motion control board and the PSO-PC board must be inserted into an appropriate PC. In addition to sharing the same standard PC ISA bus, the two boards are connected by a 50-pin proprietary Aerotech bus (P3) using a flat ribbon cable (supplied with the PSO-PC). The P3 connection is used primarily for communicating PSO programming commands from the motion controller directly to the PSO-PC board for appropriate processing. Refer to Figure 1-2. Also, refer to Figure 1-3 for the U500 system interconnect and Figure 1-4 for the U600 system interconnect.

The PSO-PC and motion controller boards may also be connected on a single-ended encoder interface bus (P6) using a 26-pin ribbon cable that is also supplied with the PSO-PC. This interface is used to pass encoder (i.e., position) information from the motion controller card to the PSO-PC. Refer to Figure 1-2. The P6 encoder interface connection is intended to be used with the Aerotech internal encoder bus or single-ended encoders that are connected externally. If the application uses differential encoders, the user must use the differential encoder interface bus (P7). Any combination of single-ended and differential encoders may be used. The choices are programmable using jumpers located on the PSO-PC board. A user-supplied cable must be used for external encoder connections. Refer to Chapter 4: Technical Details for pinout details.

A digital I/O port (P5) provides 8 digital TTL inputs (IN0-IN7) and 16 digital TTL outputs (OUT0-OUT15). These inputs and outputs can be used in conjunction with the conditional tracking programming command (PSOC). For example, the user can command the PSO-PC to enable tracking based on some predefined input condition(s). If the input conditions are not met, the user can program the PSO-PC to set one or more outputs to a predefined state. The P5 port on the PSO-PC gives the user 8 digital inputs and 16 digital outputs to use for this purpose. Refer to Figure 1-2. This port can be connected to an optional Opto-22 I/O board (using a 50-pin ribbon cable) to provide screw terminal connections for these inputs and outputs.

The first four inputs (IN0-IN3) and first four outputs (OUT0-OUT3) of port P5 are also available on port P1. If conditional tracking is not needed, or if conditional tracking can be accomplished for the user's application using only four digital inputs and four digital outputs, then this option is not necessary (simply use the four inputs and four outputs from P1).



The PSO-PC board has a 44-pin Intel iSBX connector (P4). The iSBX port is an expansion port designed by Intel Corporation for use with Multimodule boards. It provides future expansion capabilities on the PSO-PC. Refer to Figure 1-2.

The main P1 interface of the PSO-PC consists of a DB37 (female) connector. This interface provides access to the three laser outputs (LOUT1, LOUT2 and READY), four TTL-type digital inputs (IN0-IN3), four open-collector TTL-type digital outputs (OUT0-OUT3), a laser shut-off input (LASEN\_N), an analog input for future use (AIN), two 16-bit programmable analog outputs (AOUT1 and AOUT2), and the associated power connections. A customized DB37 cable must be made for a particular application. Refer to Figure 1-2. The pinouts for this connector are shown in Chapter 4: Technical Details.

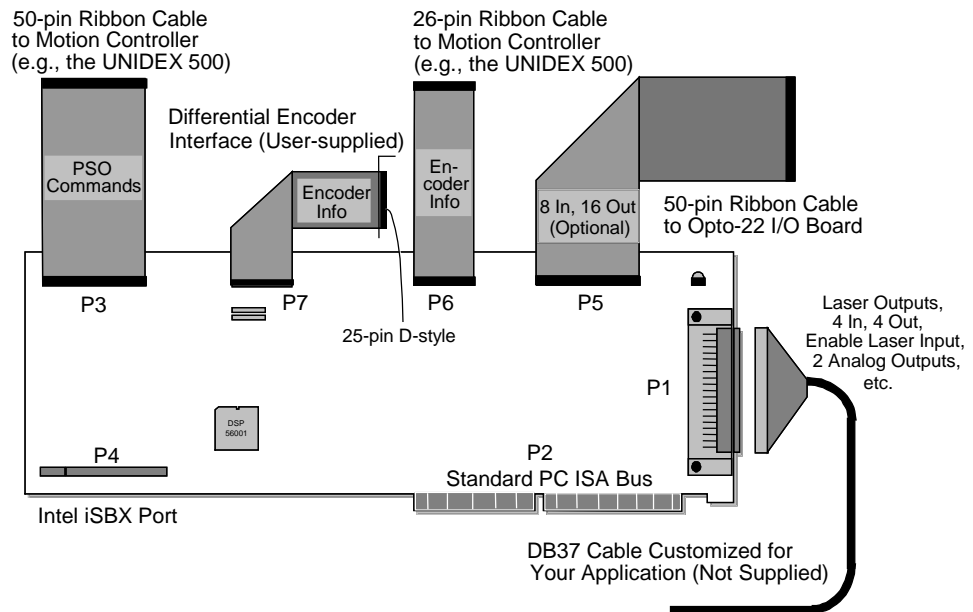


Figure 1-2. Overview of Typical PSO Interfaces



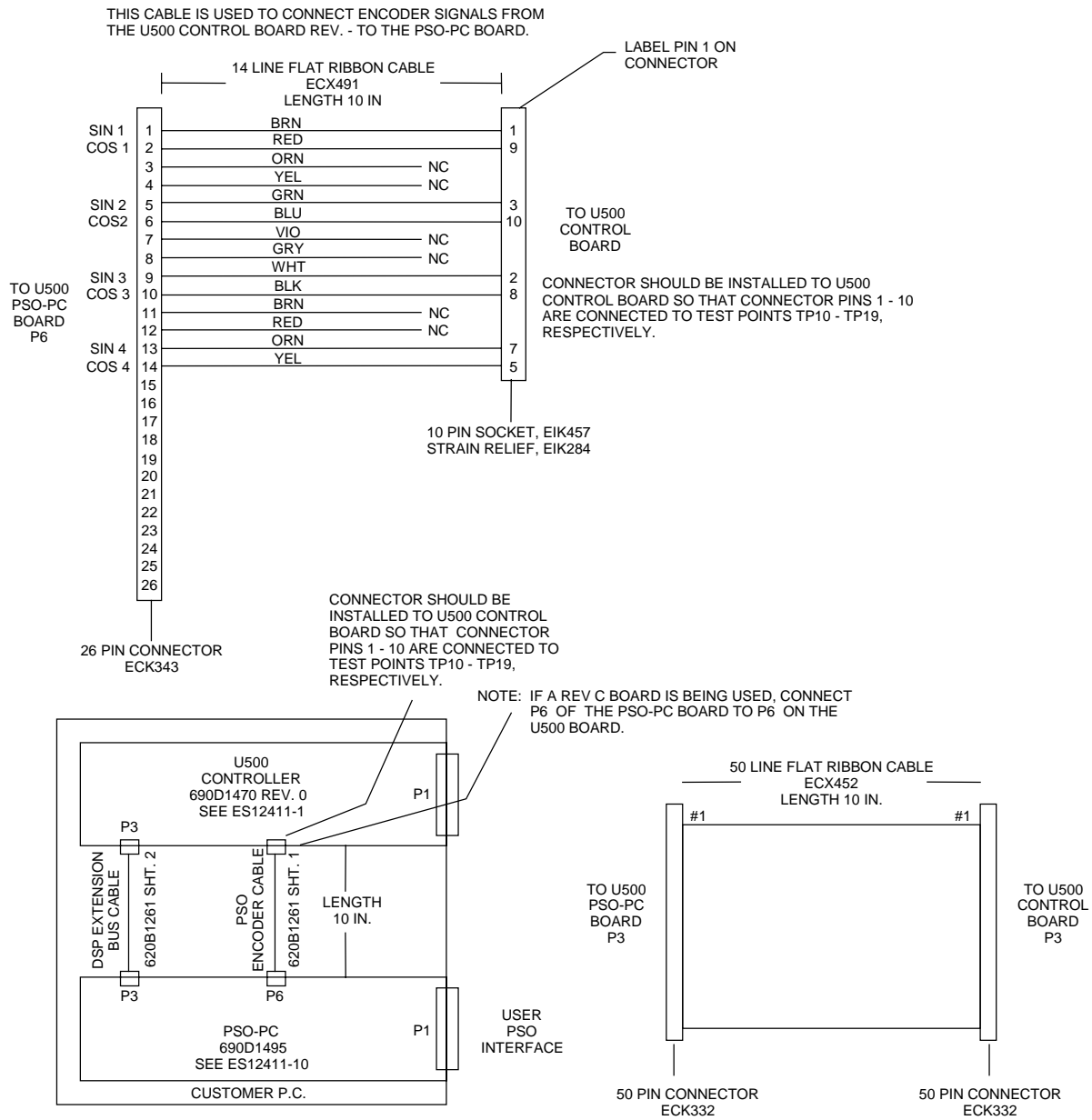


Figure 1-3. U500 System Interconnect

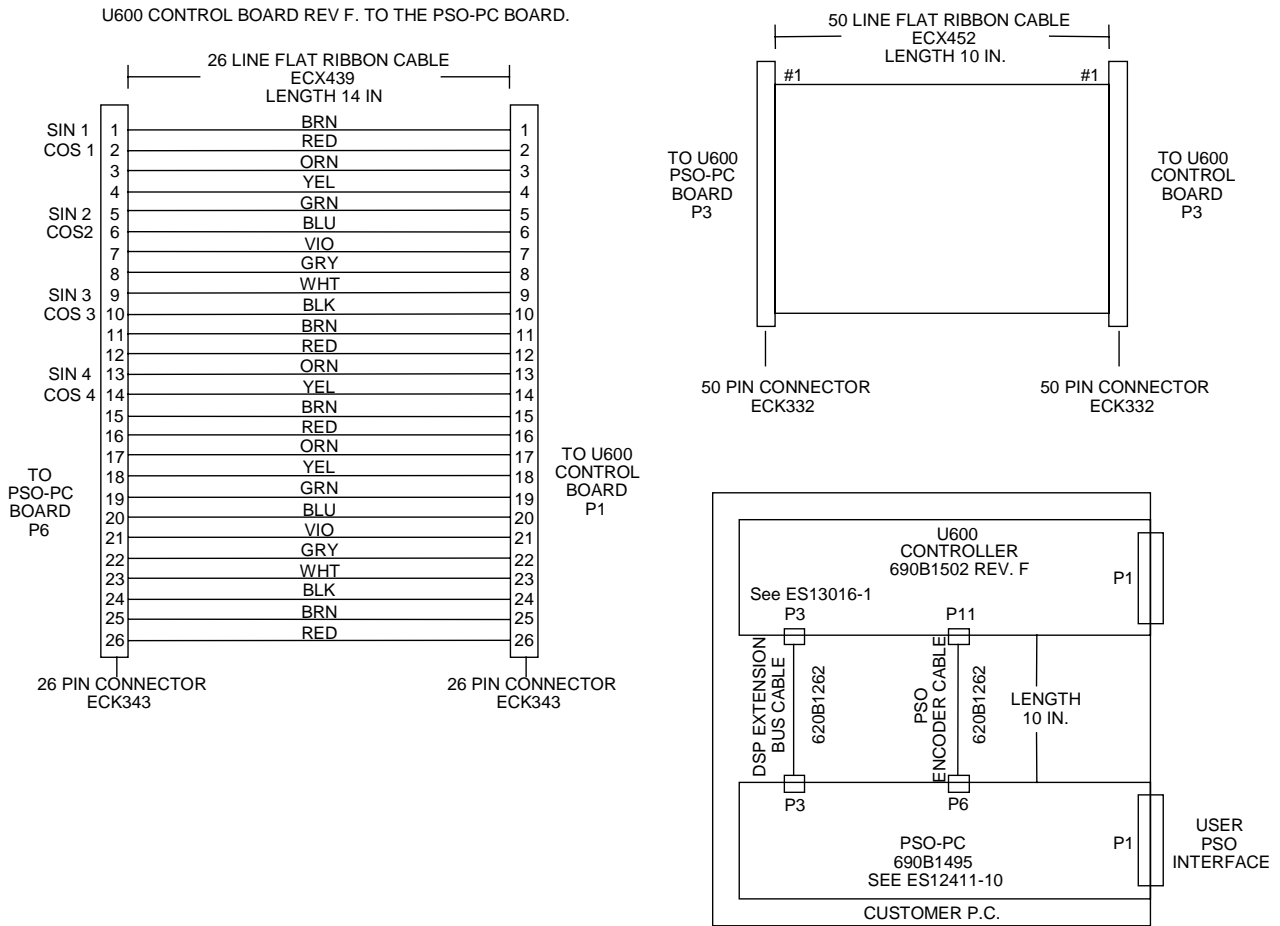
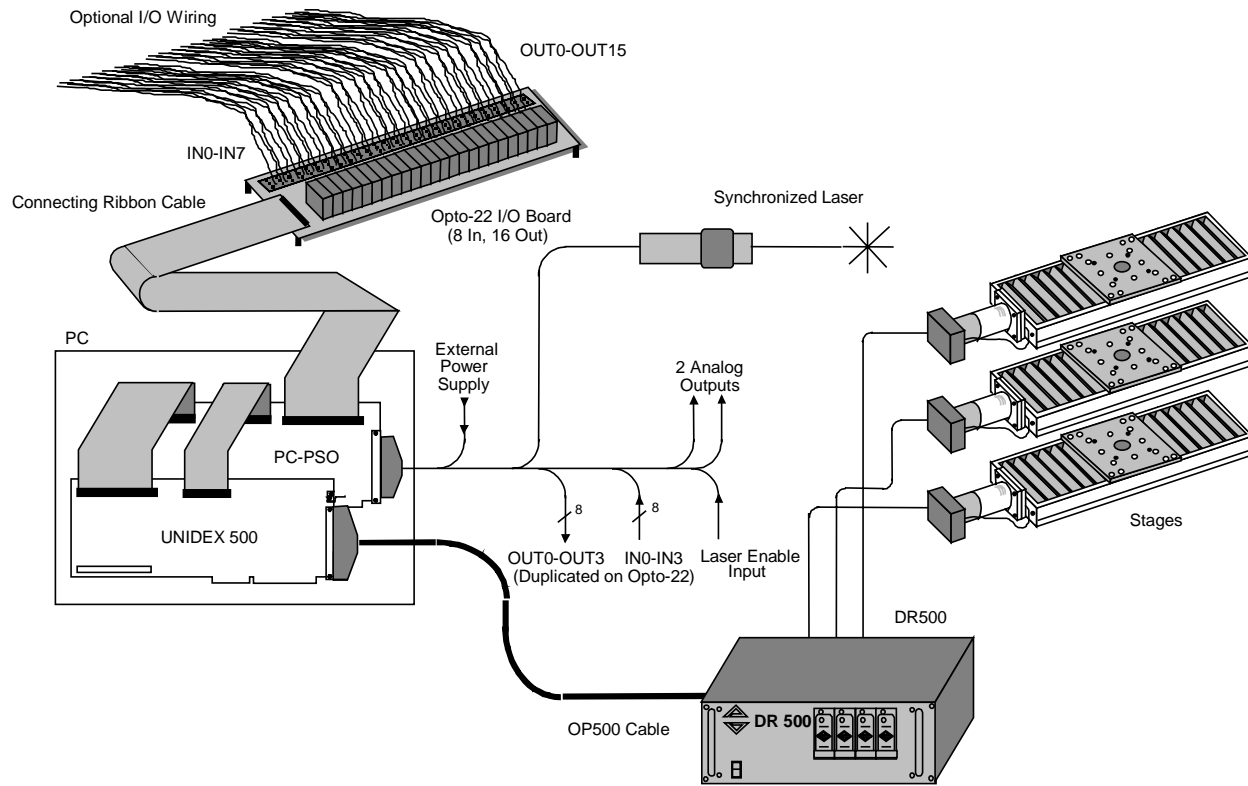


Figure 1-4. U600 System Interconnect



**Figure 1-5. Overview of a Sample PSO-PC System Using the UNIDEX 500**

Figure 1-5 illustrates a typical PSO-PC system. In this example, a UNIDEX 500 motion controller is used in conjunction with the PSO-PC board. The boards are seated in standard ISA bus slots in a PC and are connected to one another by way of port P3 (the proprietary Aerotech bus used primarily to exchange PSO programming commands) and port P6 (the single-ended encoder interface used for communicating position information to the PSO-PC board). The PSO-PC board is connected to an optional Opto-22 I/O board (by way of a ribbon cable connected to port P5) which provides 8 open-collector TTL inputs (IN0-IN7) and 16 TTL outputs (OUT0-OUT15) used for conditional tracking. The first four inputs (IN0-IN3) and the first four outputs (OUT0-OUT3) are provided at the P1 connection. If conditional tracking can be accomplished using these 8 I/Os, an Opto-22 board is not necessary. Port P1 of the PSO-PC board shows a customized DB37 cable that connects to various inputs and outputs as required for this application. The configuration may vary significantly from the system illustrated in Figure 1-5. For pinout and signal description information, refer to Chapter 4: Technical Details.

The interfaces of the PSO-PC board are listed in Table 1-1. This table shows the port number (P1 through P7), the associated connector type for each port, and a brief description of what connects to the port or how it is used. Pinouts and signal information for individual ports are listed in Chapter 4: Technical Details.

**Table 1-1. Ports and Connector Types of the PSO-PC Board**

Port	Connector Type	Description
P1	DB37-F	Main PSO-PC connector for input/output signals, power, etc.
P2	Edge Connector	Standard PC ISA bus
P3	50-pin (M)	Proprietary Aerotech bus connector (to motion controller, e.g., the UNIDEX 500) used primarily to communicate PSO programming commands to the PC-PSO card
P4	44-pin (F)	Intel iSBX expansion bus
P5	50-pin (M)	Opto-22 I/O port (provides 16 DO's, 8 DI's) used for conditional tracking
P6	26-pin (M)	External encoder interface for single-ended encoders (a combined total of 4 encoders are available on P6 and P7)
P7	26-pin (M)	External encoder interface for differential encoders (a combined total of 4 encoders are available on P6 and P7)

Table 1-2 lists an input/output summary of connector P1 of the PSO-PC board. I/O types are listed as AO (analog outputs), AI (analog inputs), DI (digital inputs) and DO (digital outputs). For additional information, refer to Chapter 4: Technical Details.

**Table 1-2. I/O Summary of PSO-PC Connector P1**

#	Type	Descriptions	Names	Use
2	AO	16-bit, -10 to +10 Volts	AOUT1, AOUT2	Optional Analog Outputs (see PSOT command)
1	AI	Extra analog input	AIN	Reserved for future use.
1	DO	Laser output signal 1	LOUT1	Main laser output
1	DO	Laser output signal 2	LOUT2	Reserved for future use.
1	DO	Laser ready	LREADY	Reset the output
1	DI	Laser enable	LASEN_N	Must be driven low to enable laser firing
4	DI	TTL inputs	IN0 - IN3	Conditional tracking (see PSOC command)
4	DO	Open Collector TTL outputs	OUT0 - OUT3	Conditional tracking (see PSOC command)

### 1.3. Programmed Output Modes

Using programming commands, the PSO-PC can be configured to operate in any one of several available output modes. Programming commands originate in the motion control program and are passed on to the PSO-PC card for processing. See Figure 1-6.

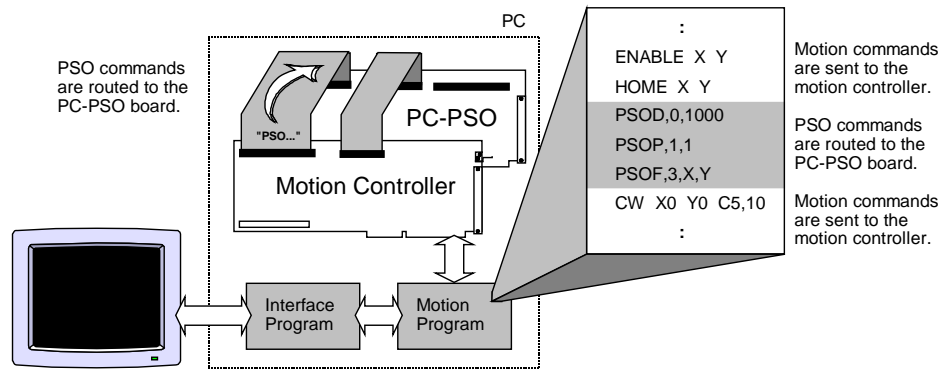


Figure 1-6. How PSO Commands Are Routed to the PSO-PC

The output modes of the PSO-PC define the operation of the PSO-PC board including the uses of inputs, the timing and functioning of outputs, and how the inputs and outputs are related. Several output modes including fixed pulse, ramped pulse, variable spacing, bitmapping, multiple pulse-at-position, and variable amplitude are available and easy to setup with the PSO-PC. Several PSO-PC output signal examples are illustrated in Figure 1-7. Refer to Chapter 3: Programming for more information about configuring the PSO-PC.

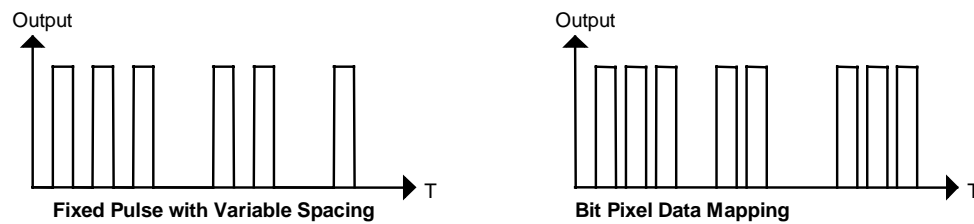


Figure 1-7. PSO Output Signal Examples

**1.4. Safety Procedures and Warnings**

The following statements apply wherever the Warning or Danger symbol appears within this manual. Failure to observe these precautions could result in serious injury to those performing the procedures and/or damage to the equipment.



To minimize the possibility of electrical shock and bodily injury, make certain that all of the electrical power switches are in the off position prior to making any electrical connections.



To minimize the possibility of electrical shock and bodily injury when any electrical circuit is in use, ensure that no person comes in contact with the circuitry.



When a motion controller is installed within a system, mechanical motion will occur. Care must be exercised that all personnel remain clear of any moving parts.



To minimize the possibility of bodily injury, make certain that all electrical power switches are in the off position prior to making any mechanical adjustments.



Never stare into the beam or the aperture of a laser head when it is turned on.

▽ ▽ ▽

## CHAPTER 2: GETTING STARTED

**In This Section:**

- Introduction .....2-1
- Unpacking and Inspecting the PSO-PC .....2-1
- PSO-PC Board Jumpers and Configurations .....2-2
- Installing the PSO-PC Board.....2-22
- Connecting to the Optional Opto-22 I/O Port (P5).....2-23

### 2.1. Introduction

This chapter steps through the logical sequence of unpacking the PSO-PC board, system (i.e., PC) requirements, board inspection, jumper settings and board installation. This chapter assumes that the user is familiar with the motion control system that is being used in conjunction with the PSO-PC.

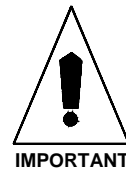
### 2.2. Unpacking and Inspecting the PSO-PC

Visually inspect the container of the PSO-PC for any evidence of shipping damage. If any such damage exists, notify the shipping carrier immediately.

The PSO-PC is wrapped in antistatic material. Make certain that the antistatic material is not damaged during unpacking. Before touching the PSO-PC board, be sure to observe the electrostatic discharge precautions that are listed below.

The PSO-PC board is sensitive to static electricity. To greatly reduce the possibility of board damage due to electrostatic discharge, adhere to the following precautions.

1. Do not remove the PSO-PC board from the antistatic bag until it is ready to be installed. When removing a card from a PC, immediately place the card in an antistatic bag.
2. Make certain that anyone who is handling the board (or any associated components) is wearing a properly grounded static strap.
3. When handling the PSO-PC board, hold the card by its edges and the mounting brackets. Avoid touching board components and the edge connectors that plug into the expansion slots
4. Do not slide the PSO-PC board over any surface.
5. Avoid plastic, Styrofoam or vinyl in the work area.
6. Static charge buildup may be removed from an object by touching the object to a properly grounded piece of metal.



The PSO-PC board was tested and inspected before being shipped from Aerotech, Inc. Vibration during shipment, however, may have loosened certain board components. Prior to installing the board, make certain all socketed ICs are firmly seated in their sockets. If a chip has become loose, carefully reinstall it into its socket. Be sure to observe the anti-static precautions mentioned above.

### 2.3. PSO-PC Board Jumpers and Configurations

This section outlines the jumper configurations of the PSO-PC control board. Descriptions are based on the following functional groups of jumpers:

- iSBX MINTR0/MINTR1 request jumper .....(JP1)
- Host base address jumpers .....(JP2-JP7)
- DSP host interrupt request to PC enable/disable jumper .....(JP8)
- SRDY\_N PC bus timing option enable/disable jumper .....(JP9)
- Dual-ported RAM PC IRQ setting jumper .....(JP10-JP15)
- Dual-ported RAM base address jumpers .....(JP16-JP23)
- Laser output polarity jumpers .....(JP24-JP26)
- Laser enable input control jumpers .....(JP27-JP28)
- Extension bus MINTR0/MINTR1 selection jumper .....(JP29)
- Encoder data format selection jumpers .....(JP30-JP33)
- Global encoder direction jumper .....(JP34)
- Encoder type (differential or single-ended) jumpers .....(JP35-JP42)
- Watchdog circuit enable/disable jumper .....(JP43)
- Host reset capability enabled/disabled jumper .....(JP44)
- Encoder sampling frequency selection jumpers .....(JP45-JP48, JP52)
- Internal/external power selection jumpers .....(JP49-JP51).

Refer to Figure 2-1 for jumper locations on the PSO-PC board.

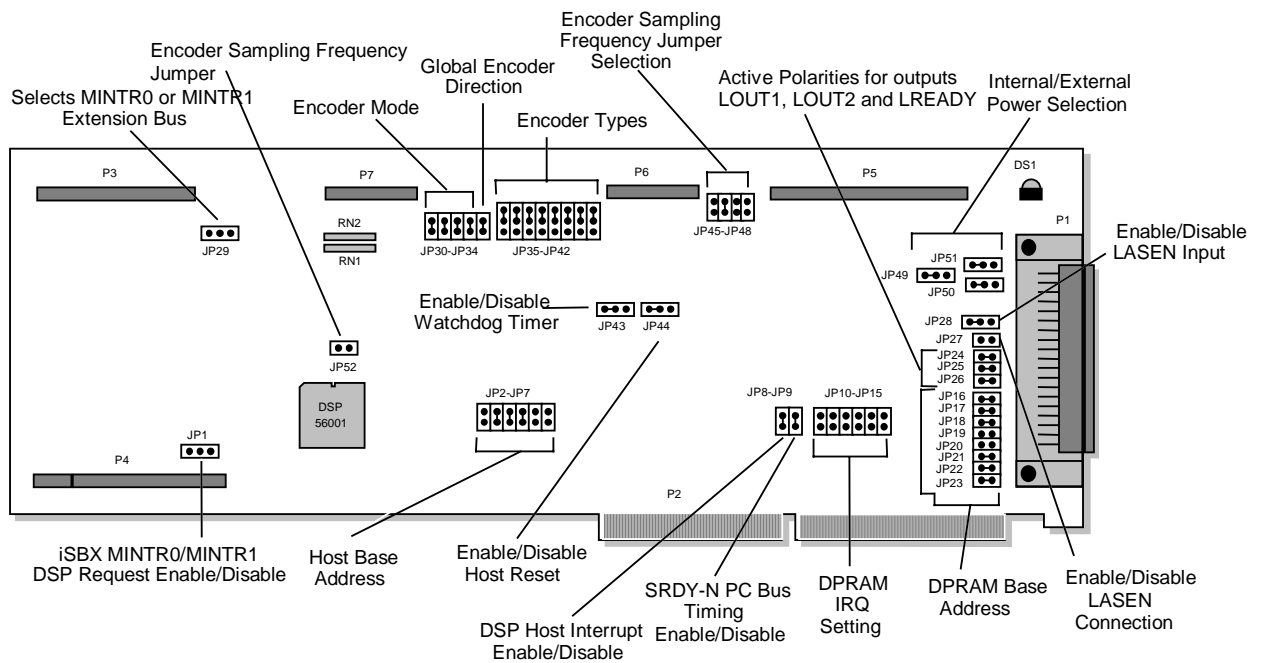


Figure 2-1. PSO-PC Board Jumpers






### 2.3.1. iSBX MINTR0/MINTR1 Request Jumper (JP1)

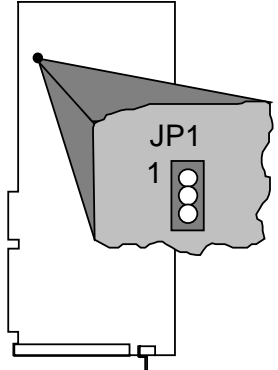
Jumper JP1 is a three-pin jumper block that is used to select which interrupt request line (MINTR0 or MINTR1) is used to interrupt the device attached to the iSBX port (P4) of the PSO-PC board. Placing a jumper on pins 1 and 2 enables MINTR0 for DSP host interrupts. Placing a jumper on pins 2 and 3 enables MINTR1 for DSP host interrupts. If a jumper is not connected at all, the PSO-PC assumes that the iSBX port is unused. This is the default setting for jumper JP1.

The settings for jumper JP1 are listed in Table 2-1.

**Table 2-1. iSBX MINTR0/MINTR1 Request Jumper Settings (JP1)**

iSBX MINTR0/MINTR1 Request	1&2	2&3	Settings
iSBX MINTR0 request to DSP	IN	OUT	JP1 
iSBX MINTR1 request to DSP	OUT	IN	JP1 
iSBX port unused (default - no jumper)	OUT	OUT	JP1 

Jumper JP1 should never be modified unless directed to do so by Aerotech.

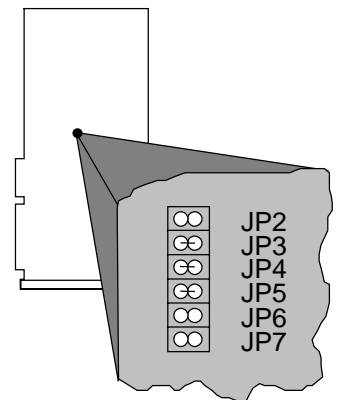
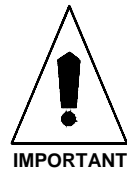


### 2.3.2. Host Base Address Jumpers (JP2-JP7)

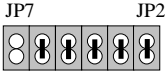





The PSO-PC has two communication ports to the PC: the host interface and the dual-ported RAM interface. The host interface is used to download system firmware and is the main communications channel to the PC. It is mapped in the PC's Input/Output (I/O) space. This mapping is accomplished using the host base address jumper settings. Jumpers JP2 through JP7 select the host base address on the PSO-PC.

Jumpers JP2 through JP7 are located near the center of the PSO-PC board to the right of the DSP chip. Each jumper has two pins. Each jumper (JP2 through JP7) is either connected or removed to create a unique host base address. The combinations of base address jumper settings are shown in Table 2-2.

The second communications port of the PSO-PC is a 2K byte dual-ported RAM. This port is used to transfer arrays of firing points and bit map information. Dual-ported RAM base addressing is accomplished using jumpers JP16-JP23, which are discussed later in this chapter.



**Table 2-2. Host Base Address Jumper Settings (JP2-JP7)**

Host Base Address Range	JP7	JP6	JP5	JP4	JP3	JP2	Settings
0x200 - 0x20F	OUT	IN	IN	IN	IN	IN	
0x210 - 0x21F	OUT	IN	IN	IN	IN	OUT	
0x300 - 0x30F	OUT	OUT	IN	IN	IN	IN	
0x310 - 0x31F (default)	OUT	OUT	IN	IN	IN	OUT	
0x350 - 0x35F	OUT	OUT	IN	OUT	IN	OUT	
0x360 - 0x36F	OUT	OUT	IN	OUT	OUT	IN	

A base address must be selected for each port by configuring the appropriate jumpers. The base address must then be used in all software communications with the PSO-PC card. If the user is using the UNIDEX 500 motion controller, parameter number 016 specifies (in hexadecimal) the host base address. UNIDEX 600 users must enter the selected host address into AerReg.exe.





If UNIDEX 500 parameter 016 (PSO Command DPRAM Base Address) is set to 0x0 (the software default which indicates that a PSO-PC board is not used), the PSO-PC will not initialize. This parameter must be set to a unique address for proper operation and must agree with the hardware address setting specified by jumpers JP2-JP7. The hardware jumpers default to the address 0x310.

### 2.3.3. DSP-to-PC Host Interrupt Enable/Disable Jumper (JP8)

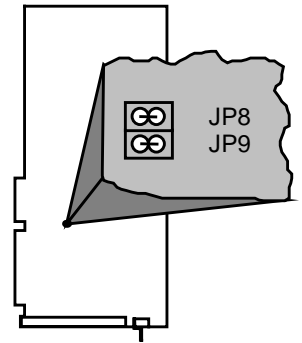
Jumper JP8 is a two-pin jumper block that is used to enable or disable interrupt requests that are initiated by the DSP to the host PC. Placing a jumper on JP8 enables DSP-to-host PC interrupts. Removing the jumper from P8 disables DSP-to-host PC interrupts. This jumper defaults to enabled (i.e., the jumper is in place).

The settings for jumper JP8 are listed in Table 2-3.

**Table 2-3. DSP-to-host PC Interrupt Enable/Disable Jumper (JP8)**

DSP-to-host PC Interrupt	JP8	Settings
Enabled	IN	JP8 
Disabled	OUT	JP8 

Jumper JP8 should never be modified unless directed to do so by Aerotech.





### 2.3.4. SRDY\_N PC Bus Timing Enable/Disable Jumper (JP9)

Jumper JP9 is a two-pin jumper block that is used to enable or disable the optional SRDY\_N PC bus timing feature. Placing a jumper on JP9 enables SRDY\_N PC bus timing. Removing the jumper from JP9 disables SRDY\_N PC bus timing. This jumper defaults to enabled (i.e., the jumper is in place).

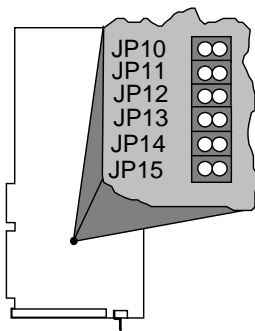
The settings for jumper JP9 are listed in Table 2-4.

**Table 2-4. SRDY\_N PC Bus Timing Enable/Disable Jumper (JP9)**

SRDY_N PC Bus Timing	JP9	Settings
Enabled	IN	JP9 
Disabled	OUT	JP9 

Jumper JP9 should never be modified unless directed to do so by Aerotech.

















### 2.3.5. DPRAM IRQ Jumpers (JP10-JP15)

The dual-ported RAM of the PSO-PC can be used to generate an interrupt request to the host PC. Currently this feature is not used by Aerotech. The interrupt level is jumper selectable and is outlined in Table 2-5. This table shows the available interrupt request (IRQ) lines that may be assigned using the PC bus interrupt jumpers.

The default jumper configuration has all jumpers removed. In this configuration, there is no interrupt connection.

**Table 2-5. DPRAM IRQ Jumper Settings (JP10-JP15)**

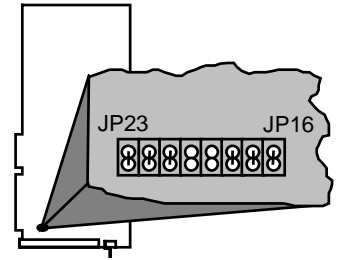
Jumper	State	Jumper Settings	Function
JP10	IN	JP10  JP15	Interrupt IRQ12 (AT unassigned)
	OUT	JP10  JP15	IRQ12 not selected (default)
JP11	IN	JP10  JP15	Interrupt IRQ11 (AT unassigned)
	OUT	JP10  JP15	IRQ11 not selected (default)
JP12	IN	JP10  JP15	Interrupt IRQ10 (AT unassigned)
	OUT	JP10  JP15	IRQ10 not selected (default)
JP13	IN	JP10  JP15	Interrupt IRQ7 (LPT)
	OUT	JP10  JP15	IRQ7 not selected (default)
JP14	IN	JP10  JP15	Interrupt IRQ4 (COM)
	OUT	JP10  JP15	IRQ4 not selected (default)
JP15	IN	JP10  JP15	Interrupt IRQ3 (COM)
	OUT	JP10  JP15	IRQ3 not selected (default)



A maximum of one DPRAM IRQ jumper may be installed at a time.

### 2.3.6. DPRAM Base Address Jumpers (JP16-JP23)

The PSO-PC has two communication ports to the PC: the host interface and the dual-ported RAM interface. The first (the host interface) was described earlier. The second is a 2K byte dual-ported RAM. It is used to transfer arrays of firing points and bit map information. Dual-ported RAM base addressing is accomplished using jumpers JP16-JP23. It is mapped in the first megabyte of system RAM. A base address for each port must be selected by configuring two banks of jumpers. This address must then be used in all software communications with the card. If the user is using the U500 motion controller, system parameters #15 and #16 specify (in hexadecimal) the base address of the dual-ported RAM and the base host address respectively. Some commonly used addresses for DPRAM are listed in Table 2-6. By default, the PSO-PC is configured to use DPRAM base address D800:0000.



**Table 2-6. DPRAM Base Address Jumper Settings (JP16-JP23)**

DPRAM Base Address	JP23	JP22	JP21	JP20	JP19	JP18	JP17	JP16	Settings
C000:0000	IN	IN	IN	IN	IN	IN	OUT	OUT	JP23 JP16 
C800:0000	IN	IN	IN	OUT	IN	IN	OUT	OUT	JP23 JP16 
D000:0000 (U600)	IN	IN	IN	IN	OUT	IN	OUT	OUT	JP23 JP16 
D800:0000 (Default)	IN	IN	IN	OUT	OUT	IN	OUT	OUT	JP23 JP16 

The dual PSO-PC hardware decodes 4K (4,096) byte blocks of system memory. The dual-ported RAM size is 2K (2,048) bytes. The dual-ported RAM therefore appears twice within the 4K byte memory block. The default jumper settings map the dual-ported RAM to memory locations D800:0000 and D800:0800. The first 2K byte block should be used for programming consistency.



A base address must be selected for each port by configuring the appropriate jumpers. The base address must then be used in all software communications with the PSO-PC card. If the user is using the U500 motion controller, parameter number 015 specifies (in hexadecimal) the DPRAM base address. The DPRAM address selected must not conflict with the AT Window defined for the U600 card within AerReg.exe.

It is possible that the address settings selected through software (parameter 015) and hardware (jumpers JP16-JP23) are not unique. If this is the case, the PSO-PC board will not function properly. This will be most evident during the use of programming commands “PSOD,1”, “PSOD,2”, “PSOM,0” and “PSOP,3”. If problems arise, the user may choose to use a program such as Microsoft Diagnostics (C:\WINDOWS\MSD.EXE) that shows memory map of programs that are currently in memory and can be used to determine memory conflicts under Windows 3.1. Likewise, under Windows NT the NT diagnostics program would be run, located under “Programs”, “Admin. Tools” on the

Start bar. In Win95, select the system icon within the control panel and view the resources by selecting the Device Manager tab. Several similar programs are also available on the market. The MSD.EXE program is part of Microsoft Windows and should be available to anyone using a PC with Windows installed on it.

It may be necessary for the setup of the PC to be reconfigured to allow communication with addresses within the first MB of memory. Typically, this is done from the setup utility of the PC by typing one or more specified keystrokes after rebooting the PC. Refer to the PC instruction manual for information on how to enter the setup program to change the setup settings on the PC.

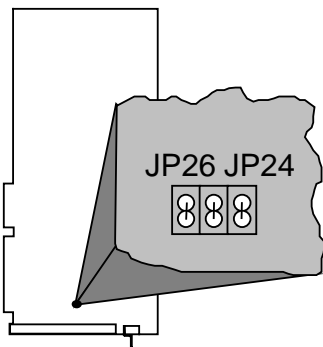


If U500 parameter 015 (PSO Board DPRAM Base Address) is set to 0x0 (the software default which indicates that a PSO-PC board is not used), the PSO-PC will not initialize. This parameter must be set to a unique address for proper operation and must agree with the hardware address setting specified by jumpers JP16-JP23. The hardware jumpers default to the DPRAM address D800:0000.

### 2.3.7. Laser Output Polarity Jumpers (JP24-JP26)

The PSO-PC has three laser outputs: LOUT1, LOUT2 and READY. The active polarity of these outputs is individually selectable from jumpers JP24, JP25 and JP26.

The READY output is directly tied to the reset line of the PSO-PC. The reset line becomes active during system initialization and download. This line also becomes active in case of a board failure. The laser output polarity jumpers are listed in Table 2-7.



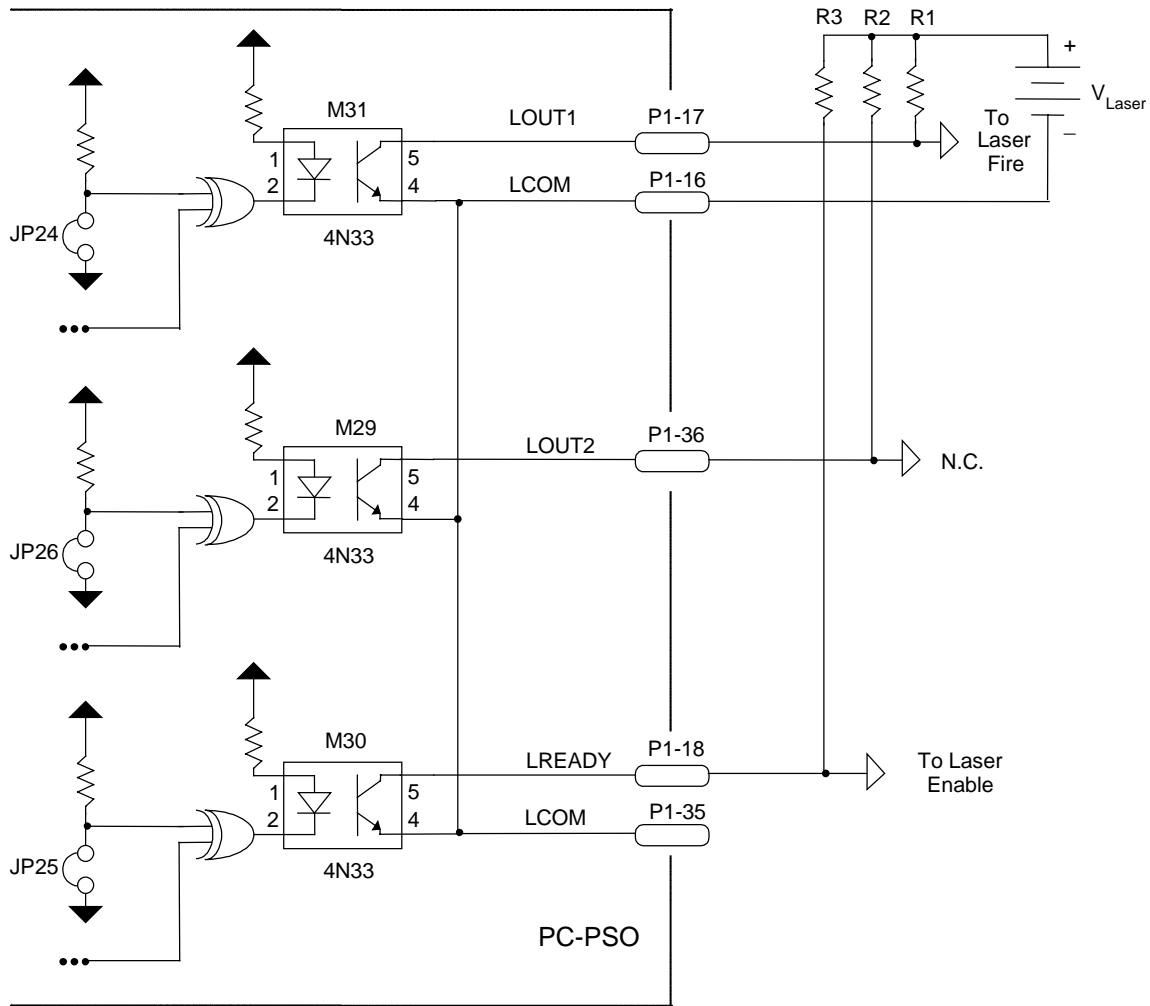
**Table 2-7. Laser Output Polarity Jumper Settings (JP24-JP26)**

Output	Jumper	Jumper "In" Settings (Active Low Polarity)	Jumper "Out" Settings (Active High Polarity)
LOUT1 (Main Output)	JP24	JP26 JP24 	JP26 JP24 
LOUT2 (Aux Output - Reserved)	JP25	JP26 JP24 	JP26 JP24 
READY (Reset Output)	JP26	JP26 JP24 	JP26 JP24 



Each of the laser outputs are optically isolated and socketed for either a 4N33 (default) or high-speed 6N136 device. The laser output circuits should never be connected to the PC power or ground. An external power supply referenced to the laser system should always be used.

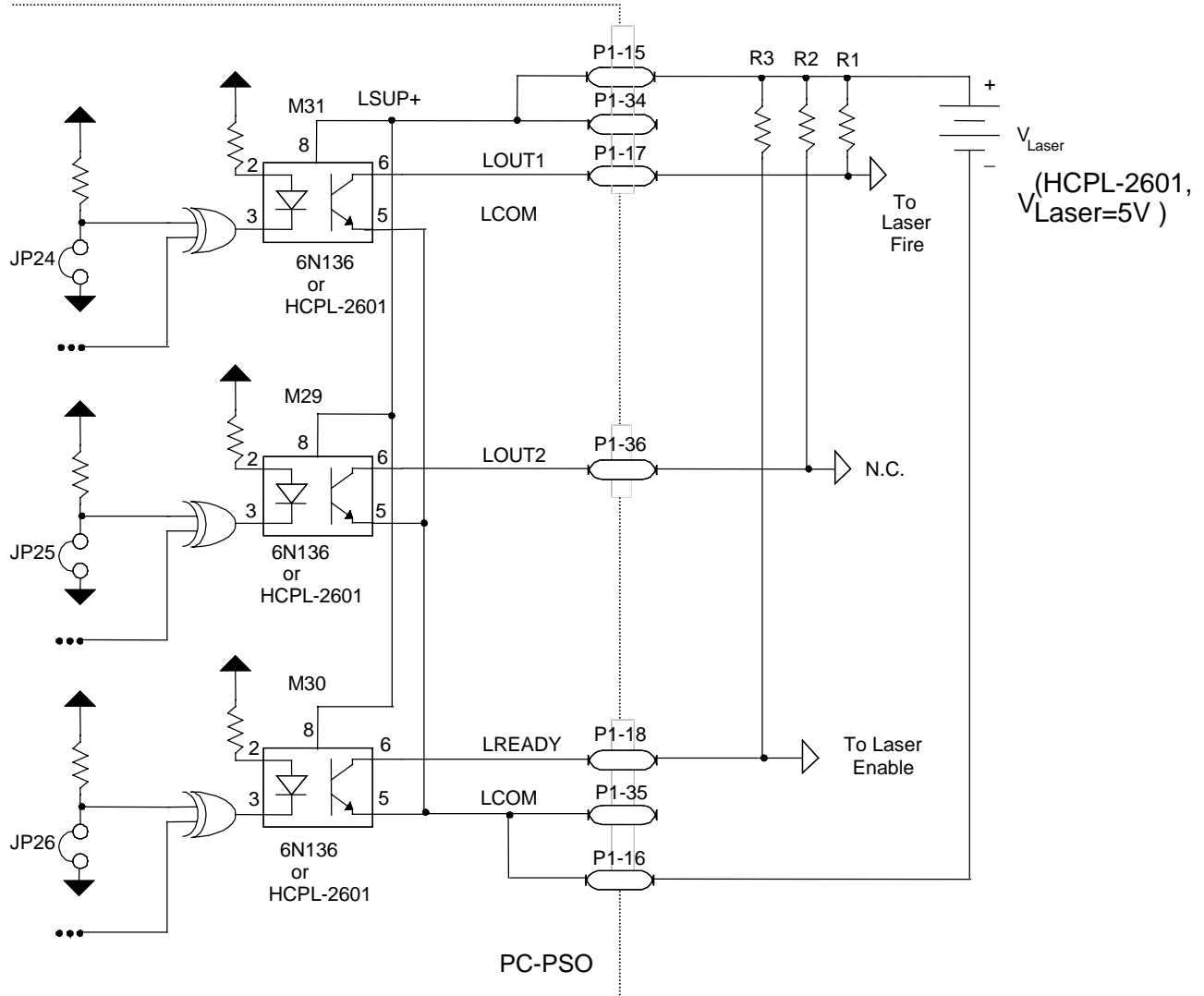
Figure 2-2 illustrates the laser interface (using 4N33 opto-isolators) and jumpers JP24 through JP26.



Note: Jumpers shown for active low outputs.

**Figure 2-2. Laser Interface (Using 4N33 Opto-isolators) Showing JP24-JP26**

Figure 2-3 illustrates the laser interface (using 6N136 or HCPL-2601 opto-isolators) and jumpers JP24 through JP26.

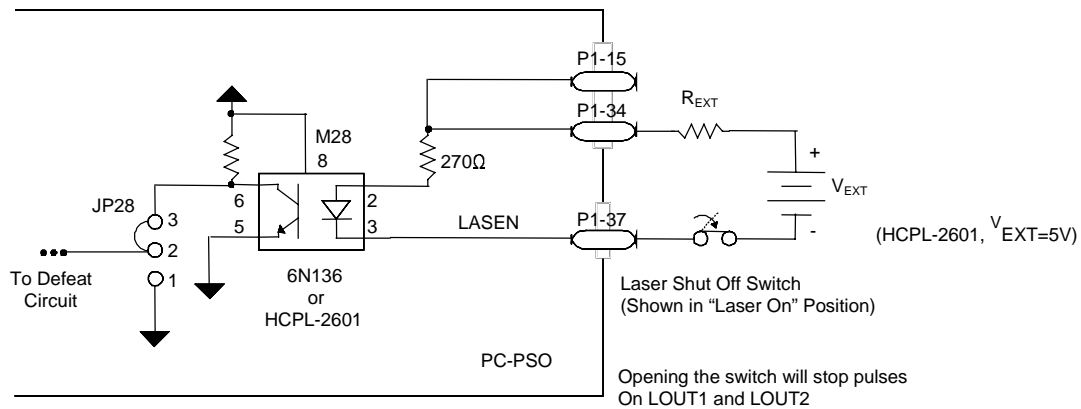


Note: Jumpers shown for active low outputs.

Figure 2-3. Laser Interface (Using 6N136 or HCPL-2601 Opto-Isolators) Showing JP24-JP26



Figure 2-4 illustrates the laser enable interface.



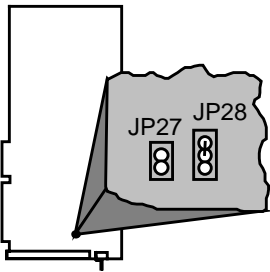
Note: Jumpers shown for active low outputs.

V <sub>Ext</sub>	R <sub>Ext</sub> (in Ohms)
5 VDC	0
12 VDC	470 Ω (1/4 w)
24 VDC	1500 Ω (1/2 w)

Figure 2-4. Laser Enable Interface

Table 2-8. Laser Output Opto-Isolator Specifications



Option (-n)	Nominal Frequency	Power Supply Voltage (V <sub>Laser</sub> )	Output Current Sink
HCPL-2601 (-1 option, std)	5 MHz	5 VDC	10 mA
6N136 (-2 option)	750 KHz	5-25 VDC	2 mA
4N33 (-3 option)	10 KHz	5-25 VDC	50 mA



### 2.3.8. Connect/Disconnect LASER Input to DSP Interrupt (JP27)

The laser enable input can be connected directly to the DSP interrupt on the PSO-PC. Enabling or disabling this connection is accomplished using jumper JP27. When the jumper is installed across JP27, then the laser opto input (signal LASEN) is connected to (and therefore driven by) the DSP. When the jumper is removed, the laser opto input (signal LASEN) is disconnected from the DSP. Refer to Table 2-9.

**Table 2-9. Connect/Disconnect LASER Input DSP Interrupt (JP27)**

Description	State	Settings
Laser opto input (signal LASEN) is connected to the DSP	IN	JP27 
Laser opto input is disconnected from the DSP	OUT	JP27 



Jumper JP27 should never be modified unless directed to do so by Aerotech.

### 2.3.9. Enable/Disable Laser Enable Input LASEN (JP28)

The PSO-PC has an optically isolated fail-safe input. When driven to the active state (low), this input enables the laser output. Otherwise, the output will remain in its inactive state.

The laser enable input of the PSO-PC can be configured in one of two ways. Configuration is accomplished using a three-pin jumper block (JP28) located near the top of the P1 connector. If enabled (jumpers 1 and 2 are connected), the laser input signal (LASEN) is ignored by the PSO-PC. If disabled (jumpers 2 and 3 are connected), the LASEN input signal must be driven low to enable laser firing. The settings for JP28 are shown in Table 2-10.

**Table 2-10. Enable/Disable Laser Enable Input LASEN (JP28)**

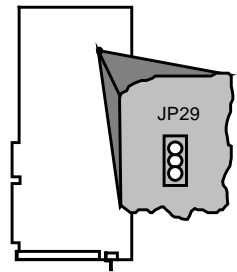
Description	1&2	2&3	Settings
LASEN input is ignored.	IN	OUT	JP28  <sup>1</sup>
LASEN input must be driven low to enable laser firing.	OUT	IN	JP28  <sup>1</sup>






Jumper JP28 should never be modified unless one is using the laser enable interface circuit in Figure 2-4.

### 2.3.10. MINTR0/MINTR1 Selection Jumper (JP29)

MINTR0 and MINTR1 are active high output lines that are used to make interrupt requests to the base board. The selection of MINTR0, MINTR1 or neither is made using JP29. JP29 is a three-pin jumper block located below and to the right of the Aerotech P3 bus connector. If jumpers 1 and 2 are connected, MINTR0 is used. If jumpers 2 and 3 are connected, MINTR1 is used. If a jumper is not used (default), then neither interrupt signal is used. The settings for JP29 are shown in Table 2-11.

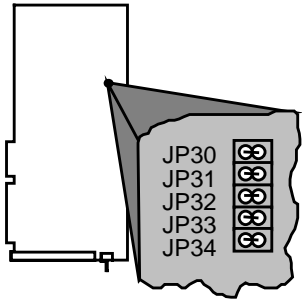


**Table 2-11. MINTR0/MINTR1 Jumper Settings (JP29)**

Description	1&2	2&3	Settings
No interrupt selected (default)	OUT	OUT	JP29  <sup>1</sup>
MINTR0 selected for extension bus P3	IN	OUT	JP29  <sup>1</sup>
MINTR1 selected for extension bus P3	OUT	IN	JP29  <sup>1</sup>

Jumper JP29 should never be modified unless directed to do so by Aerotech.

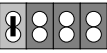
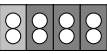
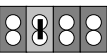
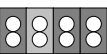
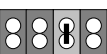
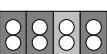
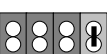
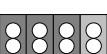


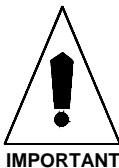


### 2.3.11. Encoder Data Format Selection Jumpers (JP30 through JP33)

The PSO-PC has four on-board encoder channels. Encoder connections are made to the 26-pin connectors (P6 and P7) located at the top of the PSO-PC board. Each of the four counter channels has two inputs (A and B). The data format for each pair of inputs can be selected by using jumpers JP30 through JP33. Each of these jumpers corresponds to one of the four encoder counters (JP30 is for encoder counter 1, JP31 is for encoder counter 2, etc.). The data format of the inputs for each encoder channel can be either SIN/COS mode (the jumper is installed) or clock/direction mode (the jumper is removed). Refer to Table 2-12.

**Table 2-12. Encoder Data Format Jumper Settings (JP30-JP33)**

Encoder Channel	Jumper Number	State	Encoder Data Format for Inputs A and B	Settings
1	JP30	IN	A = SIN (Quad) B = COS (Quad)	JP30  JP33
		OUT	A = Clock B = Direction	JP30  JP33
2	JP31	IN	A = SIN (Quad) B = COS (Quad)	JP30  JP33
		OUT	A = Clock B = Direction	JP30  JP33
3	JP32	IN	A = SIN (Quad) B = COS (Quad)	JP30  JP33
		OUT	A = Clock B = Direction	JP30  JP33
4	JP33	IN	A = SIN (Quad) B = COS (Quad)	JP30  JP33
		OUT	A = Clock B = Direction	JP30  JP33





Jumpers JP30 through JP33 must be configured prior to operating the PSO-PC.

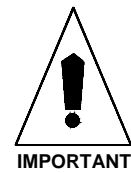
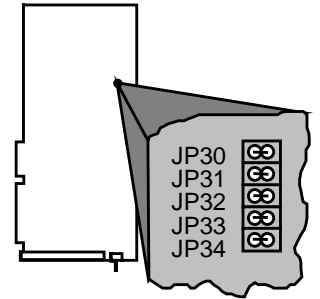
### 2.3.12. Global Encoder Direction Jumper (JP34)

The direction of all encoders used by the PSO-PC can be set to standard direction (default) or inverted direction using jumper JP34. Jumper JP34 is located at the top of the PSO-PC board between the encoder connectors P6 and P7. When the jumper is installed (default), the global encoder direction is the standard direction. If the jumper is removed, the global encoder direction is inverted. Refer to Table 2-13.

**Table 2-13. Global Encoder Direction Jumper Setting (JP34)**

State	Global Encoder Direction	Settings
IN	Global encoder direction is standard (default)	JP34 
OUT	Global encoder direction is inverted	JP34 

Jumper JP34 should never be modified unless directed to do so by Aerotech.

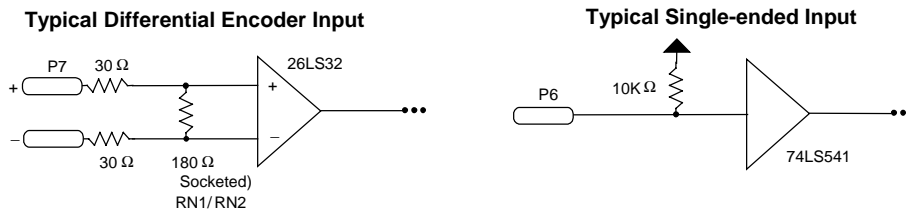
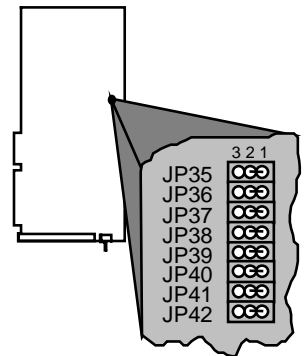


### 2.3.13. Encoder Type Jumpers (JP35-JP42)

The PSO-PC has four on-board encoder channels. Encoder connections are made to the 26-pin connectors (P6 and P7) located at the top of the PSO-PC board for external encoder interfacing. The PSO-PC usually connects to an internal encoder bus. Connector P6 is used for interfacing to single-ended encoders. P7 is used for interfacing to differential encoders. Refer to Figure 2-5. P7 has differential RS-422 inputs and terminating resistor networks RN1 and RN2. The default termination value is 180Ω.


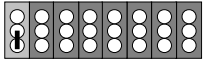
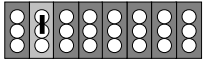
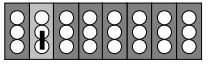
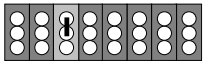
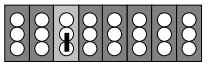
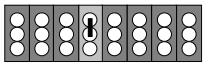
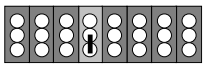
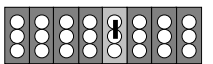
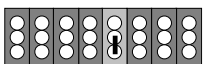




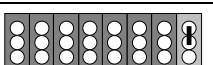
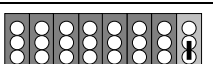
The P6 and P7 connectors designed to connect to a 25-pin ribbon cable (pin 26 is not used). A 25-pin D-style connector is then used in an adjacent PC slot for encoder connection. Refer to Chapter 4: Technical Details for pinouts.

Each of the four counter channels has two inputs (A and B). The data format for each pair of inputs can be selected as SIN/COS or Clock/Direction by using jumpers JP30 through JP33 (discussed earlier). Jumpers JP35 through JP42 define the encoder type (single-ended or differential) for each input (A and B) of all four encoder counter inputs. Refer to Table 2-14.



**Figure 2-5. Typical Differential and Single-Ended Encoder Inputs**



Table 2-14. Encoder Type Jumper Settings (JP35-JP42)

Encoder Channel	Jumper Number	Input	1&2 State	2&3 State	Encoder Type	Port	Settings
1	JP35	A	IN	OUT	Differential	P7	 1 2 3
		A	OUT	IN	Single-ended	P6	 1 2 3
1	JP36	B	IN	OUT	Differential	P7	 1 2 3
		B	OUT	IN	Single-ended	P6	 1 2 3
2	JP37	A	IN	OUT	Differential	P7	 1 2 3
		A	OUT	IN	Single-ended	P6	 1 2 3
2	JP38	B	IN	OUT	Differential	P7	 1 2 3
		B	OUT	IN	Single-ended	P6	 1 2 3
3	JP39	A	IN	OUT	Differential	P7	 1 2 3
		A	OUT	IN	Single-ended	P6	 1 2 3
3	JP40	B	IN	OUT	Differential	P7	 1 2 3
		B	OUT	IN	Single-ended	P6	 1 2 3
4	JP41	A	IN	OUT	Differential	P7	 1 2 3
		A	OUT	IN	Single-ended	P6	 1 2 3
4	JP42	B	IN	OUT	Differential	P7	 1 2 3
		B	OUT	IN	Single-ended	P6	 1 2 3

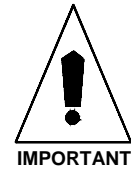
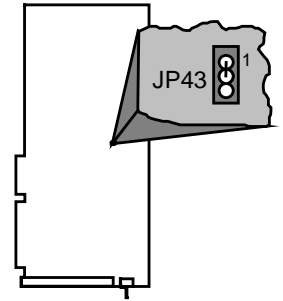
### 2.3.14. Watchdog Circuit Enable/Disable Jumper (JP43)

The PSO-PC has a built-in diagnostic watchdog feature. When this feature is enabled (default), the DSP strobes the watchdog timer. The watchdog feature can be disabled using jumper JP43. JP43 is a three-pin jumper located below the P6 connector. Refer to Table 2-15 for settings. If the watchdog fails, the LED on the PSO will flash.

**Table 2-15. Watchdog Circuit Enable/Disable Jumper Setting (JP43)**

Pins 1&2	Pins 2&3	Description	Settings
IN	OUT	DSP strobes the watchdog	JP43  1
OUT	IN	Watchdog is disabled (default)	JP43  1



Jumper JP43 should never be modified unless directed to do so by Aerotech.



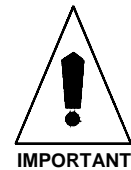
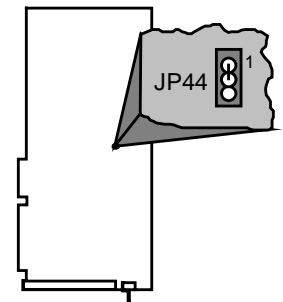
### 2.3.15. Host Reset Capability Enable/Disable Jumper (JP44)

Host reset capability of the PSO-PC can be enabled/disabled using jumper JP44. JP44 is a three-pin jumper located below the P6 connector. When the jumper is installed on pins 1 and 2 (default), the host drives the reset. If the jumper is installed on pins 2 and 3, the host is disconnected from the reset. Refer to Table 2-16 for settings.

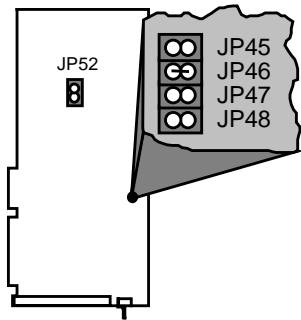
**Table 2-16. Host Reset Capability Enable/Disable Jumper Setting (JP44)**

Pins 1&2	Pins 2&3	Description	Settings
IN	OUT	Host drives reset (default)	JP44  1
OUT	IN	Host is disconnected from reset	JP44  1

Jumper JP44 should never be modified unless directed to do so by Aerotech.



### 2.3.16. Encoder Sampling Frequency Jumpers (JP45-JP48, JP52)



Encoder sampling frequency is the rate at which an encoder signal is fed to the control board (refer to Figure 2-6). This is also the frequency at which the encoder signals are passed from the motion control board to the PSO-PC. This frequency is given in counts/second (or Hertz [Hz]). The sample frequency of an encoder input signal for the PSO-PC is selectable using jumpers JP45 through JP48 and JP52. Five different maximum encoder sampling rates are available on the PSO-PC. These selections are 2.5 MHz, 5 MHz, 10 MHz (the default), 20 MHz, and the DSP clock out.

The quadrature encoder input signals are sampled and multiplied by four. For example, a sample frequency of 5 MHz allows a maximum input frequency of 625 kHz (and a 2.5 MHz data rate). This is the optimum setting for relatively low-resolution encoders. If a high-resolution encoder or a multiplier box is going to be used with the motion controller, the user may need to select the 10 MHz sample frequency. This allows a maximum input frequency of 1.25 MHz (5 MHz data rate).

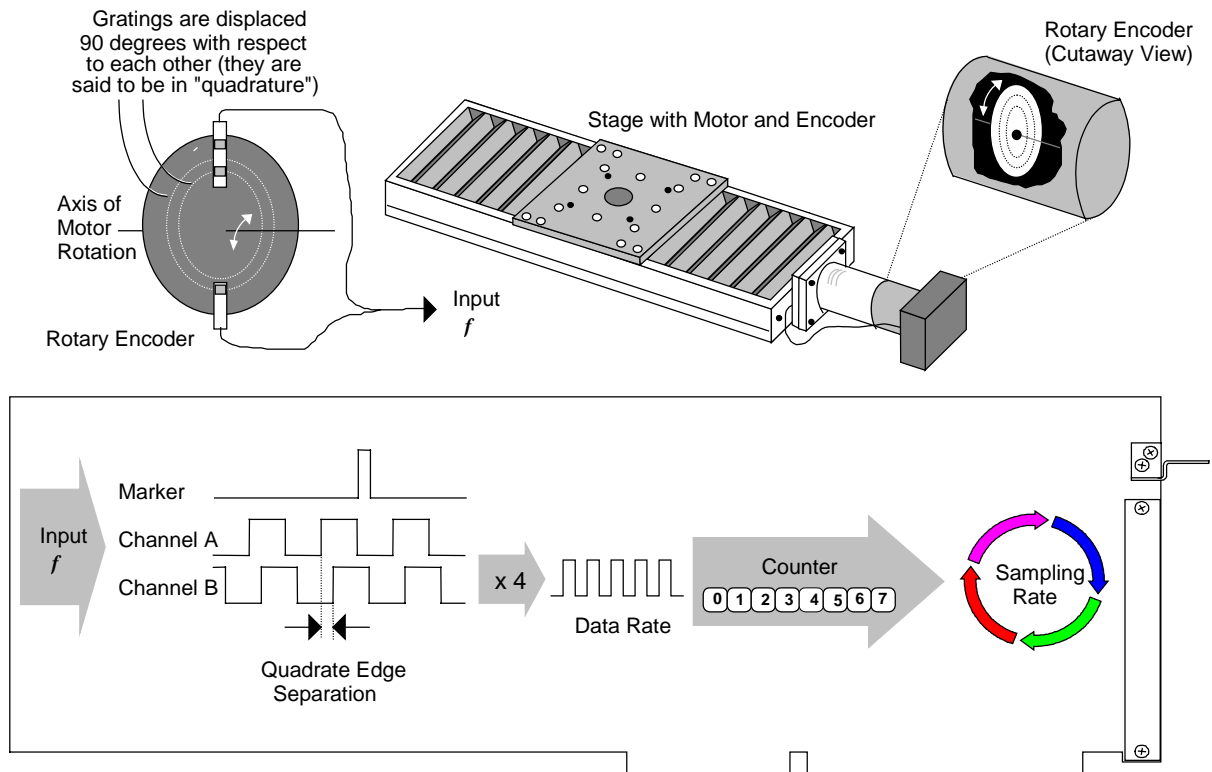




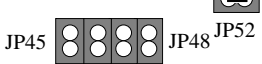


Figure 2-6. The Encoder Sampling Process



Jumper JP45 corresponds to a frequency of 20 MHz, JP46 corresponds to a frequency of 10 MHz, JP47 corresponds to a frequency of 5 MHz, JP48 corresponds to a frequency of 2.5 MHz, and JP52 is determined by the DSP (programmable). The available encoder sampling frequency options are shown in Table 2-17.

**Table 2-17. Encoder Sampling Frequency Jumper Settings (JP45-JP48 and JP52)**

Encoder Sampling Frequency	JP45	JP46	JP47	JP48	JP52	Settings
2.5 MHz	OUT	OUT	OUT	IN	OUT	
5 MHz	OUT	OUT	IN	OUT	OUT	
10 MHz	OUT	IN	OUT	OUT	OUT	
20 MHz	IN	OUT	OUT	OUT	OUT	
DSP Clock Out (Default)	OUT	OUT	OUT	OUT	IN	

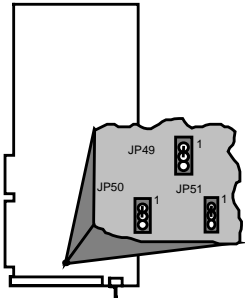
If the specific encoder requirements are unknown, do not change the default setting. Jumper JP52 should not be used unless directed to do so by Aerotech.



### 2.3.17. Internal/External Power Selection Jumpers (JP49-JP51)

The PSO-PC contains two 16-bit analog outputs (AOUT1 and AOUT2). These outputs may be completely isolated from the PC by selecting the proper jumper configurations for JP49 through JP51. An external bipolar supply must then be connected to power this circuit. The result is a floating analog output that may be referenced to the laser. This provides a high degree of PC protection if a fault occurs with the laser system.

The default jumper configuration disables the optical isolation and directly references the outputs to the PC power supply. Refer to Table 2-18 for settings.



**Table 2-18. Internal/External Power Selection Jumper Settings (JP49-JP51)**

Pins 1&2	Pins 2&3	Description	Settings
IN	OUT	Analog output power from PC's internal +12, -12 and GND (default).	
OUT	IN	Analog output power from P1 main connector. Must use external power supply.	



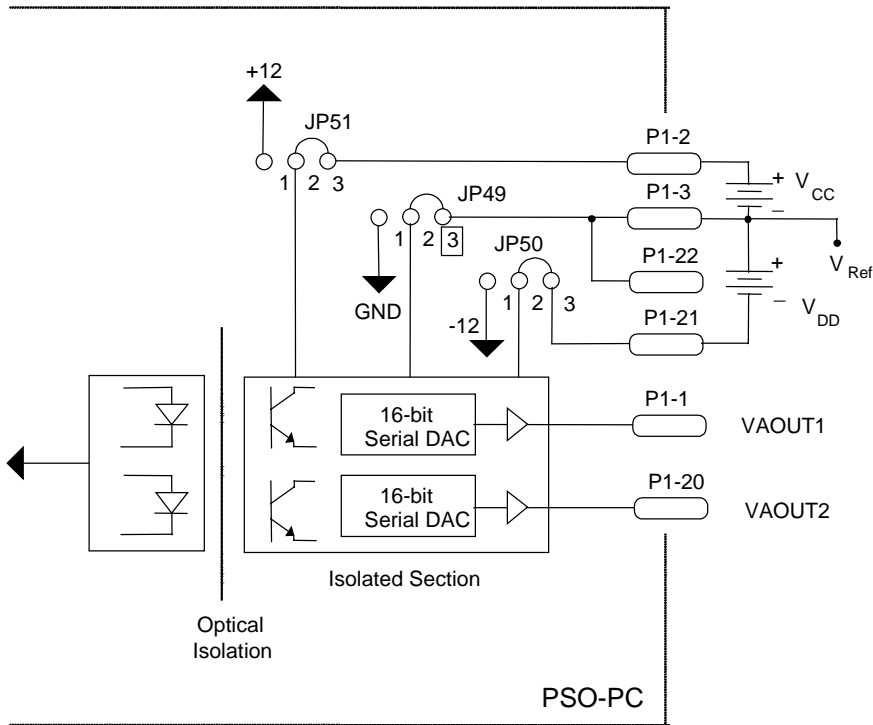
Jumpers JP49 through JP51 must always be set as a group.

If the external power option is selected, use the connections listed in Table 2-19.

**Table 2-19. Port 1 External Power Connections for AOUT1 and AOUT2**

P1 Pin #	Description
2	External power VCC +12 to +15 at 250 mA maximum.
3	External power common.
21	External power VDD -12 to -15 at 250 mA maximum.
22	External power common.

Figure 2-7 illustrates optical isolation of the analog outputs.



Note: Jumpers are set to enable optical isolation.

V<sub>CC</sub> = +12V to +15V

V<sub>DD</sub> = -12V to -15V

**Figure 2-7. Optical Isolation of Analog Outputs**

If jumpers JP49 to JP51 are in locations 1-2 (analog power supplied by PC), then VAOUT1 and VAOUT2 are referenced to PC common (pins 5, 8, 13, 24, 27, 31, and 32 on P1).



## 2.4. Installing the PSO-PC Board

The PSO-PC control board is a full-sized AT card that is installed into any of the PC's unused 16-bit expansion slots.



The PSO-PC control board may not fit in some smaller models of PC's.

The procedure for installation of the PSO-PC board is outlined in the steps that follow.

1. Turn OFF the power to the computer system unit and unplug the unit's power cord from the power source.



The possibility of electrical shock exists. Make certain that the computer system's power switch is in the OFF position and the power cord is disconnected before opening the computer's cabinet.

2. Open the computer cabinet. (Refer to the PC's User Manual for directions for opening the cabinet.)
3. Select an unused 16-bit (full-sized) expansion slot on the computer motherboard.
4. Locate the bracket of the selected expansion slot. Remove the screw and pull the bracket out of the expansion slot.
5. Observing anti-static safeguards, line up the PSO-PC board with the expansion slot and guide rails. Lower the board into the slot until each of its edge connectors rests on an expansion slot receptacle. Using evenly distributed pressure, push the board straight down until it is fully inserted into the expansion slot.
6. Secure the board to the chassis by reinstalling the bracket screw that was removed in step 3.
7. Attach the appropriate cables from the motion controller card to the PSO-PC.
8. Close and secure the PC's cover.
9. Reconnect the PC's power cord to the power source.
10. Move the PC's power switch to the ON position.

### 2.5. Connecting to the Optional Opto-22 I/O Port (P5)

The PSO-PC has an Opto-22 compatible I/O port (P5). This port has 16 open collector outputs. The outputs will go to the high impedance state during reset. This state is programmed as a “0” condition in the PSO-PC software. The “1” programmed state pulls the output bit to ground. Each bit will sink 20 mA of current in this state.

The digital I/O port also has 8 TTL inputs. Each input is tied to +5 V through a 10 KΩ resistor. An unconnected input pin will be read as a “1”. A pin that is pulled to ground will be read as a “0”.

A sample connection to port P5 is illustrated in Figure 2-8.

Optical isolation is recommended when interfacing the PSO-PC to any existing device.

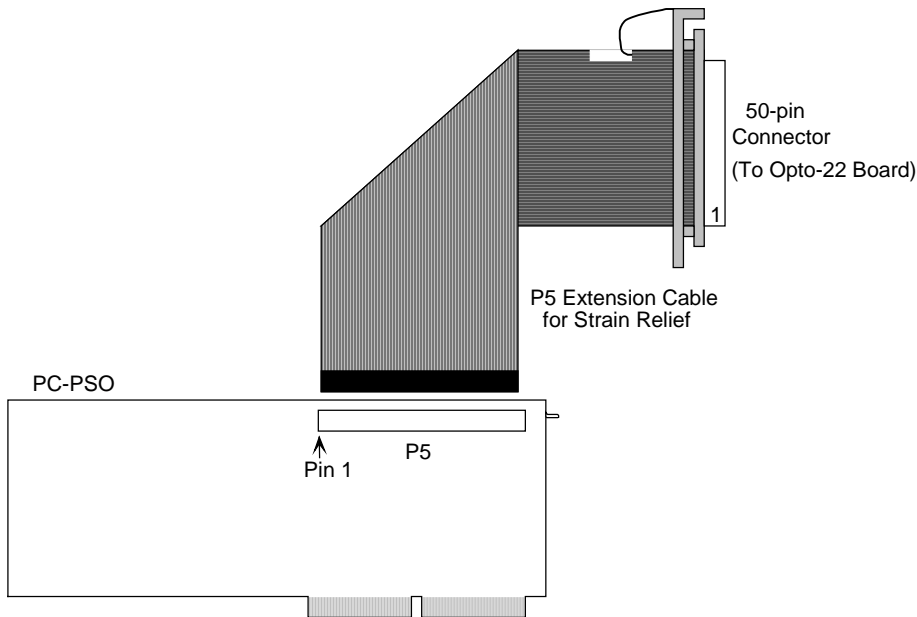


Figure 2-8. Sample Connection to the Opto-22 I/O Port (P5)





### CHAPTER 3: PROGRAMMING

**In This Section:**

- Introduction ..... 3-1
- Programming Commands ..... 3-3
- Conditional Tracking Based on Input States ..... PSOC ..... 3-4
- Position Synchronized Output Firing Distance Entry ..... PSOD ..... 3-7
- Enable/Disable Position Synchronized Output Firing ..... PSOF ..... 3-12
- Position Synchronized Output Using Bit Mapping ..... PSOM ..... 3-19
- Position Synchronized Output Pulse Configuration ..... PSOP ..... 3-20
- Position Synchronized Output with Real-time Control ..... PSOR ..... 3-26
- Digital/Analog Output Command ..... PSOT ..... 3-27
- Programming Tools ..... 3-33

#### 3.1. Introduction

**UNIDEX 600 users refer to the comprehensive *U600 Programming Manual, P/N EDU158* or the online help file within the U600MMI-NT/95.**



The PSO-PC board is programmed using commands that are integrated into the motion control program. As the motion control card (e.g., the UNIDEX 500) interprets these PSO-PC-specific commands, they are passed to the PSO-PC board (by way of the 50-pin ribbon cable on port P3). These programming commands are used to configure the PSO-PC for the desired type of operation. This chapter discusses the subset of machine control programming commands that are used to configure the PSO-PC board. For a complete list of motion control commands, refer to the operations and technical manual of the appropriate motion controller (e.g., the UNIDEX 500 or UNIDEX 600).

The PSO commands are not case sensitive, although throughout this chapter the PSO commands appear in uppercase letters for easy recognition.



This chapter uses the typographical conventions listed in Table 3-1.

**Table 3-1. Conventions for this Section**

Example	Description
<i>input_num</i>	Words in italic indicate information that the user <b>must</b> supply to validate the command.
[option]	Items between brackets are optional.
PSOC, <i>mode</i> ,[{ <i>i</i> , <i>n</i>   <i>in_map</i> , <i>out_map</i> }]	Braces and a vertical bar indicate a choice among two or more items. The user must choose one of the items unless brackets surround the braces.
PSOT, <i>case</i> , <i>condition</i> , <i>condition2</i> ,...	Three dots following an item indicate that more items having the same form may be included.
⋮	A column of three dots indicates that part of an example program has been omitted



### 3.2. Programming Commands

The PSO-PC supports seven functional groups of programming commands. These functional groups of commands are listed in Table 3-2 and explained in detail in the sections that follow.

**Table 3-2. PSO-PC Programming Commands**

Command	Page	Description
PSOC	3-4	Conditional tracking based on input states
PSOD	3-7	Firing distance entry
PSOF	3-12	Specify tracking axes and begin tracking
PSOM	3-19	Bit mapping data download
PSOP	3-20	Laser pulse output definition
PSOR	3-26	Real time control of tracking
PSOT	3-27	Digital and analog output control

Refer to the UNIDEX 600 Series online help file or Programming Manual (P/N EDU158) for the programming commands supported by the PSO-PC board when using the U600 motion controller.



### 3.3. Conditional Tracking Based on Input States

**PSOC**

The PSOC command is used to enable unconditional tracking or enable conditional tracking based on the state(s) of up to 8 PSO-PC inputs (IN0-IN7). The states of up to 16 outputs (OUT0-OUT15) can be controlled using this command when tracking is not enabled.



The PSO-PC board can accommodate 4 inputs (IN0-IN3) through the P1 connector. If the optional Opto-22 board is used (connected to P5), then 4 additional inputs (IN4-IN7) can be used to define conditional tracking using the PSOC command. Likewise, the PSO-PC can accommodate 4 outputs (OUT0-OUT3) through the P1 connector. If the optional Opto-22 board is used (connected to P5), then 12 additional outputs (OUT4-OUT15) can be used to define output conditions when conditional tracking is not enabled.

#### SYNTAX:

```
PSOC,mode [{ ,i,n | ,in_map,out_map}]
```

#### 3.3.1. MODE Arguments for PSOC

The mode argument defines one of four possible ways to use the PSOC command. The argument can range in values from 0 to 3 and the following sections describe their meaning.

##### 3.3.1.1. Enable Position Tracking Unconditionally

**PSOC,0**

Mode argument “0” enables position tracking unconditionally. Input signal conditions are ignored (Default). This mode has no additional arguments. Syntax for this mode is **PSOC,0**.

##### 3.3.1.2. Position Tracking Dependant on Input

**PSOC,1**

Mode argument “1” enables position tracking when a specified input number  $i$  ( $i=0-7$ ) is in a specified state  $n$  ( $n=0$  means a “low” state,  $n=1$  means a “high” state). Syntax for this mode is **PSOC,1,i,n** (where  $i$  specifies the input number and  $n$  specifies the required state of the input for tracking). In this mode, counter data is *retained* when the position counter is disabled.

##### 3.3.1.3. Position Tracking Dependent on Input (Position Counter set to 0)

**PSOC,2**

Mode argument “2” enables position tracking when a specified input number  $i$  ( $i=0-7$ ) is in a specified state  $n$  ( $n=0$  means a “low” state,  $n=1$  means a “high” state). Syntax for this mode is **PSOC,2,i,n** (where  $i$  specifies the input number and  $n$  specifies the required state of the input for tracking). In this mode, counter data is *reset to 0* when the position counter is disabled.

### 3.3.1.4. Position Tracking Dependent on I/O Map PSOC,3

Mode argument 3 enables position tracking when input bits 0-7 are configured as specified in the *in\_map* argument (0 = “low”, 1 = “high”, and x = input bit is not checked). If the inputs are not configured as specified in the *in\_map* argument (that is, the input condition is false), then the 16 outputs are set according to the *out\_map* argument. Syntax for this mode is PSOC,3,*in\_map*,*out\_map* (where *in\_map* is an 8-position bit map that corresponds to inputs IN0-IN7, and *out\_map* is a 16-position bitmap corresponding to the output states OUT0-OUT15).

### 3.3.1.5. Low, High Command PSOC,4

This command fires a pulse (PSOP, 4 mode only) when the specified axis is within the window defined by "low" and "high". These values are in machine steps and referenced to the point where firing was enabled (with PSOF, 3 command).

#### EXAMPLE:

```

PSOD, 0, 100           ; fire every 100 machine steps
PSOP, 4 , 1000        ; define pulse width (1000µsec = 1ms)
PSOC, 4, 10000,100000 ; define window in machine steps
PSOF, 3, X            ; start tracking

```

## 3.3.2. PSOC Arguments

The *arguments* used by the PSOC command vary based on the *mode* that is being used (refer to Section 3.3.1. Mode Arguments for PSOC). The following sections give a summary of all arguments used by the PSOC,*mode* command.

### 3.3.2.1. *i* Argument

The “*i*” argument specifies the single PSO-PC input that controls the conditional tracking. This argument can range from 0 to 7 and corresponds to inputs IN0 through IN7, respectively. If the state of the input specified by argument *i* equals the predefined state *n*, then the condition is true and position tracking is enabled. *This* argument is only used in modes 1 and 2 of the PSOC command.

### 3.3.2.2. *n* Argument

The “*n*” argument specifies the desired state for the selected input *i*. If the state of input *i* (0=low or 1=high) equals the desired state *n* (0=low or 1=high), then the condition is true and position tracking is enabled. *This* argument is only used in modes 1 and 2 of the PSOC command.

### 3.3.2.3. *in\_map* Argument

The *in\_map* argument is an 8-position bitmap that defines the desired states of one or more selected inputs. If the actual states of the 8 PSO-PC inputs (IN0-IN7) match the desired states specified in the *in\_map* argument, then the condition is true and position tracking is enabled. If this condition is false, then the second bitmap (*out\_maps*) defines the desired condition of the 16 PSO-PC outputs (OUT0-OUT15). This argument is only used in mode 3 of the PSOC command.

**3.3.2.4. out\_map Argument**

The *out\_map* argument is a 16-position bitmap that defines the desired states of one or more selected outputs when the input condition is not met (when position tracking is disabled). When position tracking is not enabled (i.e., the actual states of the 8 PSO-PC inputs do not match the desired states specified in the *in\_map* argument), then 16 PSO-PC outputs are set according to *out\_map*. This argument is only used in mode 3 of the PSOC command.



Each bit of the *in\_map* and *out\_map* bitmap arguments is configured using a “0”, “1” or “x”. A “0” indicates a low input/output and a “1” indicates a high input/output. For *in\_map*, “x” indicates that the associated input state should not be checked. For *out\_map*, “x” indicates that the associated output state remains unchanged. If an optional Opto-22 I/O extension board is not being used (thereby limiting the user to 4 inputs and 4 outputs through connector P1), then the user should use an “x” in positions IN4-IN7 of *in\_map* (e.g., xxxx1010).

**EXAMPLE:**

```

:                               ;Motion controller pre-processing
PSOC,3,xx1x0101,xxxx11000011xxxx ;Conditional PC PSO tracking enable
:                               ;Motion commands, post-processing, etc.
    
```

In this example, tracking is enabled only when inputs 0, 2 and 5 are high, and when inputs 1 and 3 are low. The states of the other inputs are ignored. If tracking is not enabled (i.e., when the inputs do not match the above criteria), then the PSO-PC outputs are set according to the final argument. In this case, outputs 4, 5, 10 and 11 are driven high, and outputs 6, 7, 8 and 9 are driven low. All other outputs are unaffected. This example assumes that an optional Opto-22 I/O extension module is being used. Otherwise the example might be PSOC,3,xxxx0101,xxxxxxxxxxxx1100.

### 3.4. Position Synchronized Output Firing Distance Entry PSOD

The PSOD command specifies the number of machine steps to be traveled before output synchronization occurs on laser output (LOUT1). Distances may be entered individually or sequentially through the use of the V0-V9999 variables of the UNIDEX 500. This command is only used in conjunction with the PSOF,3 and PSOP command.

**SYNTAX:**

**PSOD,mode {,distance |,V#, ±m}**

#### 3.4.1. Mode Arguments for PSOD

The mode argument defines one of three possible ways to use the PSOD command. The argument can range in values from 0 to 2 and the following sections describe their meaning.

##### 3.4.1.1. Pulse Output at a Constant Incremental Distance PSOD,0

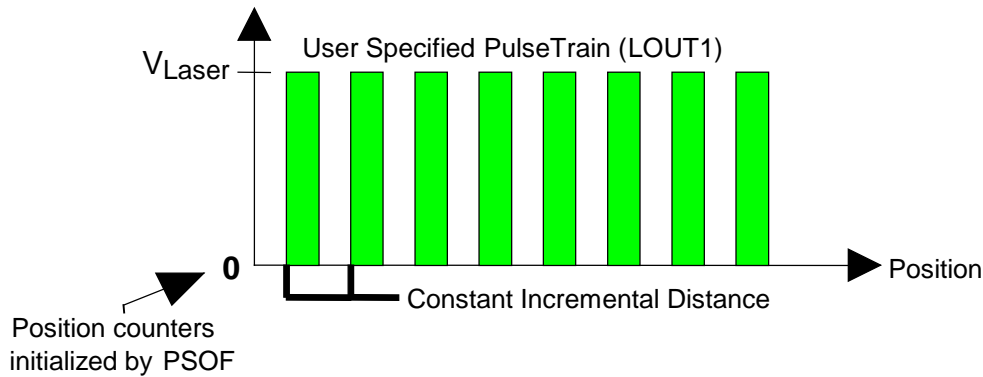
“PSOD,0” indicates that the pulse output will occur at a fixed incremental distance *distance*. An example illustrating this is shown in Figure 3-1.

**SYNTAX:**

**PSOD,0, distance.**

**EXAMPLE:**

PSOP, 0, 105 ; Single pulse 10.5 ms wide  
 PSOD, 0 , 5 ; That pulse every 5 machine steps



**Figure 3-1. Trigger Pulse Fired at Constant Increments**

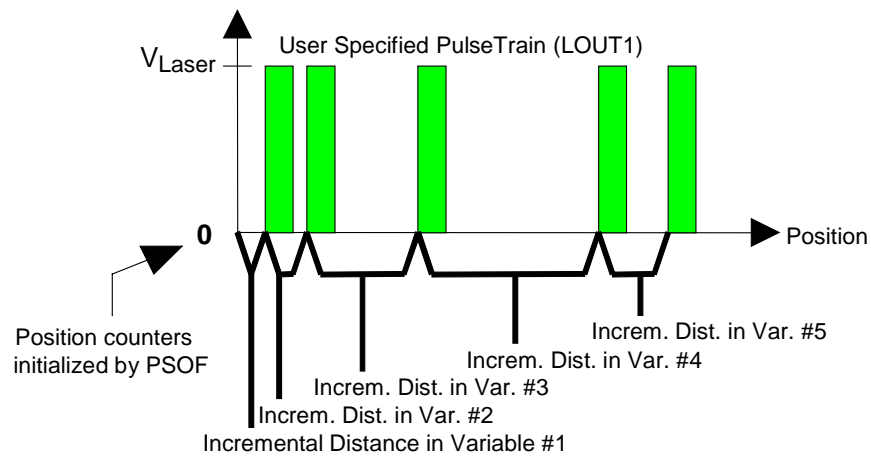
**3.4.1.2. Pulse Output at Incremental Distances Defined in Variables PSOD,1**

"PSOD,1" indicates that the pulse output will occur at incremental distances as defined in an array of variables (V0 to V9999). The variable indices may be incremental (i.e., V9 to V18) or decremental (i.e., V18 to V9). Refer to Figure 3-2. For example, the command PSOD,1,9,10 refers to: PSOD,1, starting at V9, firing incrementally over 10 firing distances, with the last distance being stored in V18. Once fired at the last incremental distance (V18), the incremental distances of V10, V11, ... will be used to start a new firing cycle. The first incremental firing distance is relative to the position at which the PSOF command is executed.

Similarly, decremental array indices can be specified by putting a negative sign in front of the variable number argument (i.e., PSOD,1,18,-10 will define 10 firing distances, first distance being stored in variable V18, last distance being stored in V9).

**SYNTAX:**

**PSOD, 1, V#, ±m**



**Figure 3-2. Trigger Pulse Fired at Incremental Distances as Defined in Variables**

**3.4.1.3. Pulse Output at Absolute Distances Defined in Variables PSOD,2**

"PSOD,2" indicates that the pulse output will occur at absolute distances as defined in an array of variables (V0 to V255). The variable indices may increment (from a lower index V9 to a higher index V18) or decrement (from V18 to V9). Refer to Figure 3-3. For example, the command PSOD,2,9,10 will define 10 firing distances with the first distance being stored in variable V9 and the last distance being stored in V18. Firing will start at the position stored in V9, keep firing at absolute positions specified by V10, V11, ..., V18, and, once fired at the last absolute distance, firing will cease. Firing will not occur as the axes move from the position contained in V18 to the portion contained in V9, you must re-execute the command with a negative increment (i.e., PSOD2,18,-10). The first incremental firing distance is relative to the position at which the PSOF command is executed.

Similarly decremental array indices can be specified by putting a negative sign in front of the variable number argument (i.e., PSOD,2,18,-10 will define 10 firing distances, the first distance stored in variable V18, and the last distance being stored in V9).

**SYNTAX:**

**PSOD, 2, V#, ±m**

The ±m argument assumes a positive direction if no sign is specified.



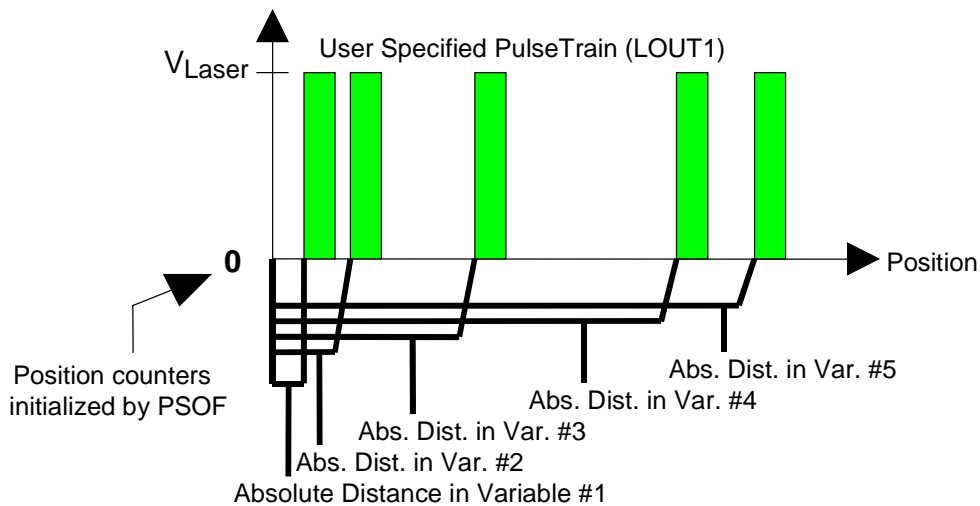
**EXAMPLE:**

```

PSOP,0,105      ;Single Pulse 10.5 ms wide
V10=100        ;Distance in Machine Steps
V11=200
V12=400
V13=800
V14=1600
PSOD,2,10,5    ;That Pulse at abs dist as defined in V10 to V14
PSOF,3,X,Y     ;Activate Pulse Firing
    
```

```

PSOP,0,105      ;Single Pulse 10.5 ms wide
V10=100        ;Distance in Machine Steps
V9=300
V8=900
V7=2700
V6=8100
PSOD,2,10,-5    ;That Pulse at abs dist as defined in V10 to V6
PSOF,3,X,Y     ;Activate Pulse Firing
    
```



**Figure 3-3. Trigger Pulse Fired at Absolute Distances as Defined in Variables**

### 3.4.2. PSOD Arguments

The arguments used by the PSOD command vary based on the mode that is being used (refer to Section 3.4.1., Mode Arguments for PSOD). The following sections give a summary of all arguments used by the PSOD,*mode* command.

#### 3.4.2.1. *distance* Argument

The “*distance*” argument specifies the fixed incremental firing distance in machine steps at which the pulsed output occurs. This distance must be less than  $2^{23}$  machine steps and is determined differently depending on how many axes are involved. Refer to Table 3-3.

#### 3.4.2.2. *V#* Argument

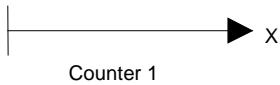
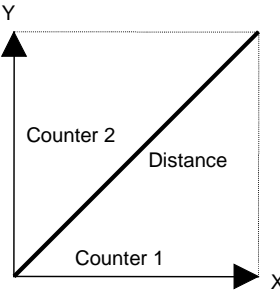
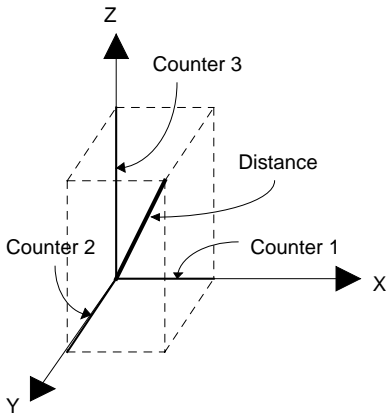
The *V#* argument specifies the starting variable number from the UNIDEX 500 that contains the first distance datum. This argument can range from 0 to 9,999 (corresponding to V0 to V9999) and is used in conjunction with the *m* argument.

#### 3.4.2.3. *±m* Argument

The *±m* argument is used to specify the number of times to increment or decrement (based on the sign) sequentially through the variables (from the starting variable number *V#*) to get additional firing distances. This argument represents the total number of variables to use (including *V#*) and can range from  $\pm 2$  to  $\pm 9,999$ . For example, the command PSOD,2,4,-5 gets the first distance datum from UNIDEX 500 variable V4. Subsequent distance values are read sequentially from variables V3, V2, V1 and V0.



**Table 3-3. Distance Calculations for Multiple Axes Using the PSOD Command**

# of Axes	Distance Calculation	Diagram
1	Distance = Counter1	
2	Distance = $\sqrt{\text{Counter1}^2 + \text{Counter2}^2}$	
3	Distance = $\sqrt{\text{Counter1}^2 + \text{Counter2}^2 + \text{Counter3}^2}$	

**EXAMPLES:**

PSOD,0,5 ;Pulsed output occurs every 5 machine steps

PSOF,3,X,Y ;Activate output firing pulse train

PSOD,1,199,-200;Pulsed output occurs at incremental distances found in V199,  
;V198, V197, V196, ... , V1, V0 (200 total firing distances)

PSOF,3,X,Y ;Activate output firing pulse train

PSOD,2,100,50 ;Pulsed output occurs at absolute distances found in V100,  
;V101, V102, ... , V148 and V149 (50 total firing distances)

PSOF,3,X,Y ;Activate output firing pulse train

### 3.5. Enable/Disable Position Synchronized Output Firing **PSOF**

The PSOF command activate or deactivates the pulse train output on LOUT1 and tracking features.

#### SYNTAX:

```
PSOF,mode [{,num | ,axis1 [,axis2 [,axis3]] | ,±dist ,axis1 [,axis2 [,axis3]] }
```

#### 3.5.1. Mode Arguments for PSOF

The *mode* argument defines one of six possible ways to use the PSOF command. The arguments for the mode command can range in values from 0 to 5 and the following sections describe their meanings.

##### 3.5.1.1. Disable Laser Output Pulse **PSOF,0**

“PSOF,0” disables the output firing pulse train and tracking features (default).

##### 3.5.1.2. Laser Output Fires Continuously **PSOF,1**

PSOF,1 activates the output firing pulse train (as established by the PSOP command). The pulse train will continue until it is disabled by the PSOF,0 command. No position tracking occurs in this mode.



PSOP,0 and PSOP,4 will not work with the PSOF,1 command.

#### EXAMPLE:

```
PSOP,1,50,105,75 ; Single Pulse:
                  ; 5ms lead, 10.5ms wide
                  ; 7.5ms tail
PSOF,1           ; Fires the Pulse continuously
```

### 3.5.1.3. Laser Output Fires a Specified Number of Times **PSOF,2**

“PSOF,2” activates the output firing pulse train (as established by the PSOP command) for *num* number of times. If *num*=0, then the output firing pulse train will not be activated until the previous output firing pulse train is complete. No position tracking occurs in this mode. The syntax for this command is **PSOF,2,num**.

PSOP,0 and PSOP,4 will not work with the PSOF,2 command.



#### EXAMPLE:

```
PSOP,1,50,105,75 ; Single Pulse:
                  ; 5ms lead, 10.5ms wide
                  ; 7.5ms tail
PSOF,2,525       ; Fires that Pulse 525 times
```

### 3.5.1.4. Laser Output Synchronized with Position **PSOF,3**

“PSOF,3” activates the output firing pulse train (as established by the PSOP command). The position counter locks on to the motions of the specified axes (X, Y, Z, U). Up to three axes may be “locked on” simultaneously. Output firing occurs at distances established by the PSOD command. The syntax for this command is **PSOF,3,axis1,axis2,axis3**.

#### EXAMPLE:

```
PSOP,1,50,105,75 ; Single Pulse:
                  ; 5ms lead, 10.5ms wide
                  ; 7.5ms tail
PSOD,0,5         ; Pulse every 5 Machine Steps
PSOF,3,X,Y       ; Enables Firing of that Pulse
```

### 3.5.1.5. Laser Output According to Bitmap w/ Toggle Pulse PSOF,4

“PSOF,4” activates the output firing pulse train and locks the position counters onto the specified axes (e.g., X, Y, Z). Up to three axes may be “locked on” simultaneously. The output firing pattern is determined by bit mapping as established by the PSOM command. The bit values serve the following functions:

0 - Causes the output to go/remain low

1 - Causes the output to go/remain high.

The pulse output occurs at a fixed incremental distance *dist*. If *dist* is positive (no sign or “+”), the bit pattern is run in a forward direction. If *dist* is negative (“-”), the bit pattern is run in reverse. Refer to Figure 3-4. The syntax for this command is **PSOF,4,*±dist,axis1,axis2,axis3***.



When the PSOF,4 command is issued, the first bit of the bitmap is evaluated. If it's a “1”, the laser output will immediately go high, otherwise it will go low.

#### EXAMPLE:

```

V10=0x4E738252      ;Hex value representing bitmap
V11=0x3842C5A2
V12=0xEEEE7624A
V13=0x1000EFF0
PSOM,0,10,4         ; Trigger Pulse based on bitmap in var's V10 to V13
PSOF,4,11,X         ; Activate ON/OFF Firing of bitmap, pixel width is
                    ; 11 Machine Steps. Forward Direction

V10=0x4E738252      ;Hex value representing bitmap
V11=0x3842C5A2
V12=0xEEEE7624A
V13=0x1000EFF0
PSOM,0,10,4         ; Trigger Pulse based on bitmap in var's V10 to V13
PSOF,4,-11,X        ; Activate ON/OFF Firing of bitmap, pixel width is
                    ; 11 Machine Steps. Backward Direction

```

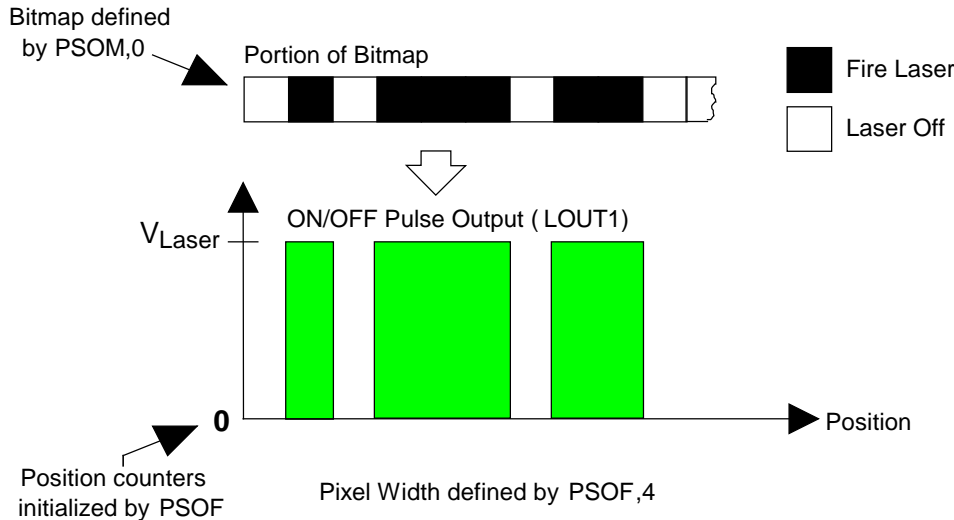


Figure 3-4. Trigger Pulse Fired ON/OFF According to Bitmap

**3.5.1.6. Laser Output According to Bitmap with User-Defined Pulse**

**PSOF,5**

“PSOF,5 activates the output firing pulse train and locks the position counters onto the specified axes (e.g., X, Y, Z). Up to three axes may be “locked on” simultaneously. The output firing pattern is determined by bit mapping as established by the PSOM command. The bit values serve the following function:

- 0 - Causes no output
- 1 - Causes the output to be a single pulse train as defined by PSOP.

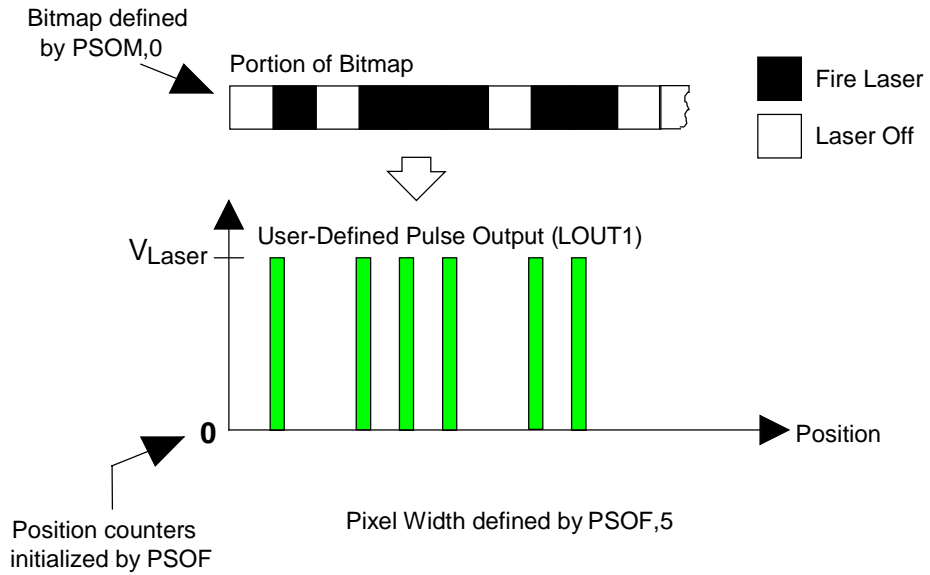
The pulse output occurs at a fixed incremental distance *dist*. If *dist* is positive (no sign or “+”), the bit pattern is run in a forward direction. If *dist* is negative (“-”), the bit pattern is run in reverse. Refer to Figure 3-5.

When the PSOF,5 is issued, the first bit of the bitmap is evaluated. If it’s a “1”, the laser output will immediately fire the pulse train as defined by PSOP.



The  $\pm dist$  argument assumes a positive distance if no sign is specified.





**Figure 3-5. Trigger Pulse Fired According to Bitmap with User-Defined Pulses**

**EXAMPLE:**

```

V10=0x4E738252 ; Hex value representing bitmap
V11=0x3842C5A2
V12=0xEE7624A
V13=0x1000EFF0
PSOM,0,10,4 ; Trigger Pulse based on bitmap in var's V10 to V13
PSOP,1,50,105,75 ; Single Pulse:
; 5ms lead, 10.5ms wide
; 7.5ms tail

PSOF,5,11,X,Y ; Activate Firing of bitmap w/ Single pulse, pixel width
; is 11 Machine Steps. Forward Direction

V10=0x4E738252 ; Hex value representing bitmap
V11=0x3842C5A2
V12=0xEE7624A
V13=0x1000EFF0
PSOM,0,10,4 ; Trigger Pulse based on bitmap in var's V10 to V13
PSOP,1,50,105,75 ; Single Pulse:
; 5ms lead, 10.5ms wide
; 7.5ms tail

PSOF,5,-11,X,Y ; Activate Firing of bitmap w/ Single pulse, pixel width
; is 11 Machine Steps. Backward Direction
    
```

### 3.5.2. PSOF Arguments

The *arguments* used by the PSOF command vary based on the *mode* that is being used (refer to Section 3.5.1. Mode Arguments for PSOF). Some forms of the PSOF,*mode* command have no additional arguments. The following sections give a summary of all arguments used by the PSOF,*mode* command.

#### 3.5.2.1. *num* Argument

The “*num*” argument specifies the number of times to activate the output firing pulse train (as established by the PSOP command). This argument is used only by the PSOF,2 command.

#### 3.5.2.2. *axis#* Argument

The “*axis#*” arguments specify which axes (up to 3) are to be locked on to the position counters. These arguments are specified using the axis names X, Y or Z. These arguments are used exclusively by commands PSOF,3, PSOF,4 and PSOF,5.

#### 3.5.2.3. *±dist* Argument

The *±dist* argument specifies the fixed incremental distance in machine counts at which the pulsed output occurs. If *dist* is positive (no sign or “+”), then the bit pattern will be run in a forward direction. If *dist* is negative (“-”), then the bit pattern will be run in the reverse direction. This distance argument is used only in commands PSOF,4 and PSOF,5. The value for this argument is determined differently depending on how many axes are involved. Refer back to Table 3-3.

**EXAMPLE:**

```
      ; Motion controller pre-processing
PSOF,0 ; Output firing pulse train and tracking disabled (default)

      ; Motion controller pre-processing
PSOP,1,5,10,0 ; Define pulse output train
PSOF,1 ; Activate the pulse train and continue until disabled

PSOP,1,5,10,0 ; Define pulse output train
PSOF,2,25 ; Activate the pulse train and continue 25 times
      ; Motion commands, post-processing, etc.

PSOD,0,50 ; Define pulsed output (firing occurs every 50 machine steps)
PSOP,1,5,10,0 ; Define pulse output train
PSOF,3,X,Y ; Activate the output firing pulse train and lock onto axes X and Y
      ; Motion commands, post-processing, etc.

      ; Motion controller pre-processing
PSOM,0,0,64 ; Define output bit mapping (in variables V0-V15)
PSOF,4,X,Y,Z ; Activate output firing pulse train and lock onto axes X, Y and Z
      ; Motion commands, post-processing, etc.

      ; Motion controller pre-processing
PSOM,0,0,64 ; Define output firing pattern (in variables V0-V15)
PSOF,5,-500,X ; Activate output firing pulse train, lock onto axis X, and run bit
               ; pattern in the reverse direction
      ; Motion commands, post-processing, etc.
```



### 3.6. Position Synchronized Output Using Bit Mapping PSOM

The PSOM command defines the characteristics of a bit pattern (stored in a programmable range of UNIDEX 500 variables V0-V9999). Bit patterns are used to specify when the laser is to be “On” (PSOF,4) or when to fire a pulse output (PSOF,5). If a bit is a 1, the laser output is on. If a bit is a 0, the laser output is off.

**SYNTAX:**

**PSOM,0,var#,±size**

#### 3.6.1. var# Argument

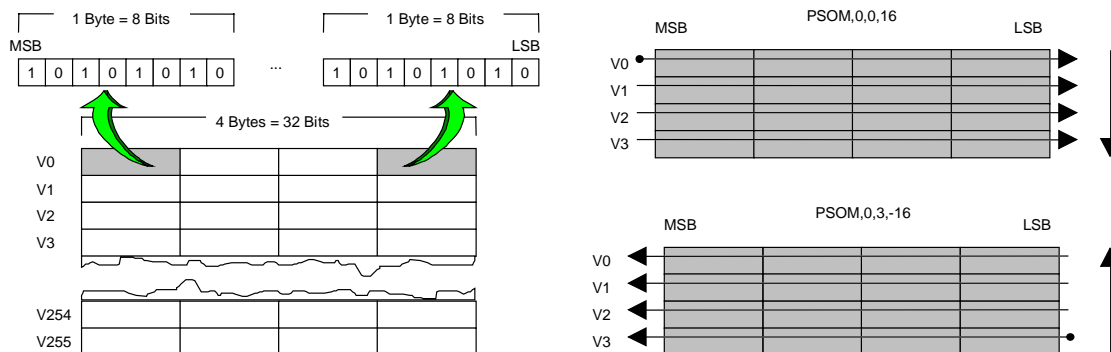
The *var#* argument specifies the number of the U500 direct variable (0-9,999 corresponds to V0-V9999) that contains the starting byte number of the first eight-bit firing pattern.

#### 3.6.2. ±size Argument

The *±size* argument specifies the total number of variables that make up the desired output firing bit map pattern. If *size* is positive (no sign or “+”), the bit pattern continues from the most significant bit of *var#* to the least significant bit. It then continues forward (to the next sequential variable *var#+1*, always starting at the most significant bit) for *size* variables. If *size* is negative (“-”), then the bit pattern continues from the least significant bit of *var#* to the most significant bit. It will then continue in reverse (to the previous variable number *var#-1*, always starting at the least significant bit) for *size* variables. This argument can range from ±2 to ±9,999. A variable V?? (V0-V9999) holds the equivalent of four bytes of information (or 32 bits of firing points). Refer to Figure 3-6.

**EXAMPLE:**

V10=0x4E738252 ; Hex value representing bitmap  
 V11=0x3842C5A2  
 V12=0xEEEE7624A  
 V13=0x1000EFF0  
 PSOM,0,10,4 ; Trigger Pulse based on bitmap in var’s V10 to V13  
 PSOF,4168,X ; Activate ON/OFF firing of bitmap, pixel width is 168 mach steps



**Figure 3-6. Bit Mapping Scan Patterns for PSOM Commands**

### 3.7. Position Synchronized Output Pulse Configuration PSOP

The PSOP command configures the pulse output train.

**SYNTAX:** `PSOP,mode { ,w | ,l,w,t | ,l,w,t,r,g | ,var#,±n }`

#### 3.7.1. Mode Arguments for PSOP

The *mode* argument defines one of six possible ways to use the PSOP command. The arguments for the mode command can range in values from 0 to 5 and the following sections describe their meaning.

##### 3.7.1.1. Simple Single Pulse PSOP,0

“PSOP,0” sets the width *w* of a single pulse output in tenths of milliseconds. Refer to Figure 3-7. The syntax for this mode is **PSOP,0,w**.

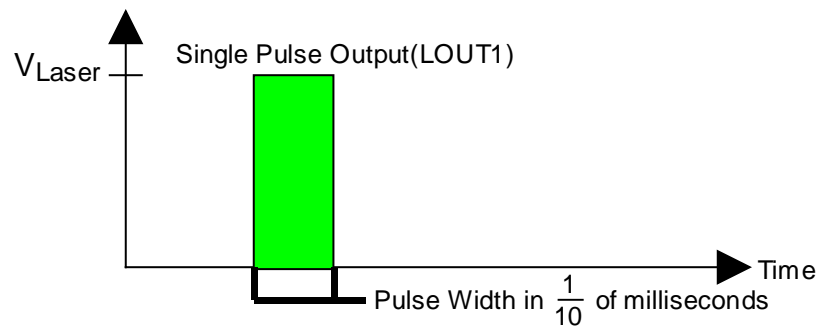


Figure 3-7. Single Pulse Output

**EXAMPLE:**

```
PSOP,0,105 ; Single Pulse 10.5 ms wide
PSOD,0,5 ; fire Pulse every 5 Machine Steps
PSOF,3,X,Y ; Activate Pulse Firing
```

**3.7.1.2. Single Pulse with Lead, Width and Tail**

**Characteristics**

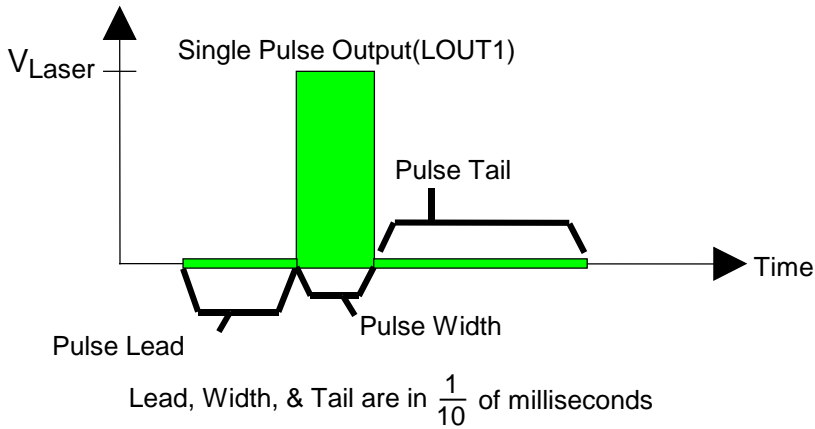
**PSOP,1**

“PSOP,1” sets a single pulse with definable pulse lead (*l*), pulse width (*w*) and pulse trail (*t*) characteristics. The pulse lead, width and trail arguments are specified in tenths of milliseconds. The syntax for this mode is **PSOP,1,*l,w,t***.

**EXAMPLE:**

```

PSOP,1,50,105,75           ; Single Pulse:
                           ; 5ms lead, 10.5ms wide
                           ; 7.5ms tail
PSOD,0,53                 ; fire Pulse every 53 Machine Steps
PSOF,3,X,Y                 ; Activate Pulse Firing
    
```



**Figure 3-8. Single Pulse Output with Lead, Width, and Tail**

**3.7.1.3. Ramping Pulse Train**

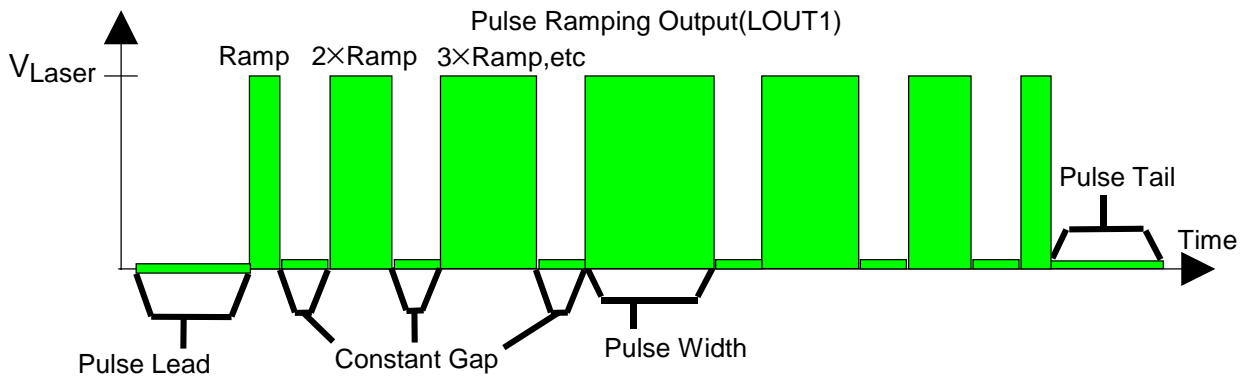
**PSOP,2**

“PSOP,2” sets a pulse train with definable pulse lead (*l*), pulse width (*w*), pulse trail (*t*), ramp up/down (*r*) and pulse gap interval (*g*) characteristics. The units used by the arguments are tenths of milliseconds. Refer to Figure 3-9. The pulse width starts at *r* and increments by *r* units until the pulse width reaches *w*. At this point, the pulse widths begin to decrease (in increments of *r* units) to a width of *r* units. The pulse gaps can be a fixed width (*g*>0) or can be made to ramp (i.e., match the size of the pulse width) (*g*=0). The syntax for this mode is **PSOP,2,*l,w,t,r,g***.

**EXAMPLE:**

```

PSOP,2,305,200,75,50,50 ; Ramping Pulse Train:
                          ; 30.5ms lead, 20ms wide
                          ; 7.5ms tail, 5ms ramp,
                          ; 5ms gap
PSOD,0,450                ; fire Pulse every 450 Machine Steps
PSOF,3,X,Y                ; Activate Pulse Firing
    
```



**Conditions**                      Lead, Ramp, Gap, Width, & Tail are in  $\frac{1}{10}$  of milliseconds  
 Ramp < Width  
 $n \times \text{Ramp} = \text{Width}$ , where *n* is an integer (*n* > 0)

**Figure 3-9. Ramping Pulse Train**

**3.7.1.4. User-Defined Pulse Train**

**PSOP,3**

“PSOP,3” sets a pulse train by defining a series pulse widths and gap widths (in tenths of milliseconds) in a series of direct variables (e.g., V0-V9999) of the UNIDEX 500. The argument *var#* specifies the direct variable number (0-9999 corresponding to V0-V9999) that contains the first gap width (in tenths of milliseconds). The next sequential variable contains the first pulse width (in tenths of milliseconds).

**SYNTAX:**

```

PSOP,3,var#,±m.
    
```

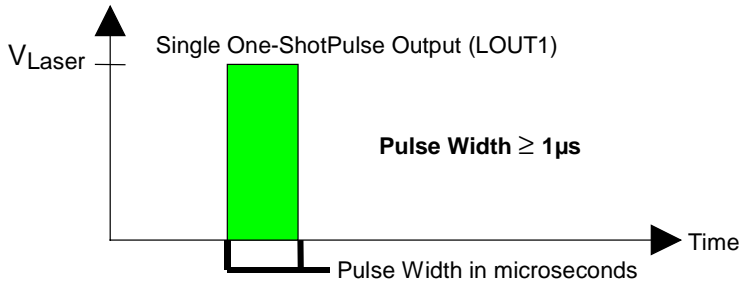
**3.7.1.5. Simple One-Shot Pulse**

**PSOP,4**

“PSOP,4” sets the width *w* of a single pulse output in *microseconds* (with a minimum value of 1 microsecond). Refer to Figure 3-10. The syntax for this mode is **PSOP,4,w**.

**EXAMPLE:**

```
PSOP,4,25 ; Single One-Shot Pulse 25 μs wide
PSOD,0,7 ; fire Pulse every 7 Machine Steps
PSOF,3,X,Y ; Activate Pulse Firing
```



**Figure 3-10. Single One-Shot Pulse Output**

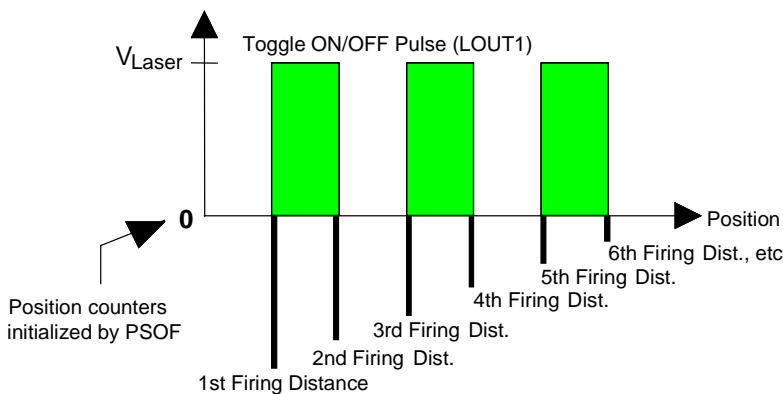
**3.7.1.6. Toggle Pulse**

**PSOP,5**

“PSOP,5” sets output toggle mode. For example, if the pulse output is configured to occur at a fixed incremental distance of 500 machine steps (PSOD,0,500), then each 500-step motion is considered to be an event. Refer to Figure 3-11. After the first 500 steps, the pulse output is enabled. After the second 500 steps, the pulse output is disabled (i.e., odd-numbered events enable the pulse output, and even-numbered events disabled the pulse output).

**EXAMPLE:**

```
PSOP,5 ;Toggle ON/OFF Pulse
PSOD,0,500 ;Pulse toggles every 500 Machine Steps
PSOF,3,X,Y ;Activate Pulse Firing
```



**Figure 3-11. Toggle ON/OFF Pulse Defined by Distance**

### 3.7.2. PSOP Arguments

The *arguments* used by the PSOP command vary based on the *mode* that is being used (refer to Section 3.7.1. Mode Arguments for PSOP). Some forms of the PSOP,*mode* command have no additional arguments. The following sections give a summary of all arguments used by the PSOP,*mode* command.

#### 3.7.2.1. *l* Argument

The “*l*” argument specifies the pulse lead in tenths of milliseconds. This argument is used with commands PSOP,1 and PSOP,2 only.

#### 3.7.2.2. *w* Argument

The “*w*” argument specifies a pulse width in tenths of milliseconds. This argument is only used with commands PSOP,0 through PSOP,2 and PSOP,4.

#### 3.7.2.3. *t* Argument

The “*t*” argument specifies the pulse trail in tenths of milliseconds. This argument is only used with commands PSOP,1 and PSOP,2.

#### 3.7.2.4. *r* Argument

The “*r*” argument specifies a ramp up/down time in milliseconds. This argument is only used with command PSOP,2. This argument allows the pulse width (and therefore the power output of the laser) to vary (i.e., increment then decrement) from *r* milliseconds to *w* (the full pulse width) in increments of  $\pm r$  milliseconds, and then back again. This feature is useful in applications involving laser drilling. In such applications, a pulse train that begins with small-width pulses (low power) and gradually increases to continuous constant-width output pulses (full power, for example), may be preferred over a steady pulse train of full-power pulses.

#### 3.7.2.5. *g* Argument

The “*g*” argument specifies the pulse gap interval (between ramps) in tenths of milliseconds. These gaps can be variable-width based on the pulse train width *w* ( $g=0$ ) or fixed width ( $g>0$ ). This argument is only used with command PSOP,2.

#### 3.7.2.6. *var#* Argument

The “*var#*” argument specifies the UNIDEX 500 direct variable number (0-9,999 corresponds to V0-V9999) that contains the “off/on” pulse train timing characteristic data in tenths of milliseconds. The specified variable contains the number of “off” tenths of milliseconds, the second sequential direct variable contains the number of “on” tenths of milliseconds, etc. for a total of *m* values.

### 3.7.2.7. $\pm m$ Argument

The “ $\pm m$ ” argument specifies the number of changes (i.e., on-to-off or off-to-on) that make up the desired output train. If  $m$  is positive (no sign or “+”), then the PSO-PC gets sequential pulse/gap width data from increasing sequential direct variables starting with  $var\#$ . If  $m$  is negative (“-”), then the PSO-PC gets sequential pulse/gap width data from decreasing sequential direct variables starting with  $var\#$ .

The  $\pm m$  argument assumes a positive distance if no sign is specified.



#### EXAMPLE:

```

:           ; Motion controller pre-processing
PSOP,0,100  ; Define a single pulse output (For PSOP,0,100, pulse width =
; 10.0 ms)
:           ; Motion controller, post processing, etc.

PSOP,1,50,10,10 ; Define pulse output train
PSOF,2,25      ; Activate the pulse train and continue 25 times
:           ; Motion commands, post-processing, etc.

PSOD,0,5000   ; Define pulsed output (firing occurs every 5,000 machine
; steps)
PSOP,1,5,10,0 ; Define pulse output train
PSOF,3,X,Y    ; Activate the output firing pulse train and lock onto axes X
; and Y
:           ; Motion commands, post-processing etc.

```

### 3.8. Position Synchronized Output with Real-time Control PSOR

The PSOP command provides various configurations of the PSO-PC's position counter.

**SYNTAX:** PSOR,*mode*

#### 3.8.1. MODE Arguments for PSOR

The *mode* argument defines one of three possible configurations (0, 1 or 3) for the position counters. There are no additional arguments for this command. The following sections describe these modes

##### 3.8.1.1. PSOR,0

Clears all previous real-time control data from the counter.

##### 3.8.1.2. PSOR,1

Stops the position counter from recording new data and retains current data under operator command.

##### 3.8.1.3. PSOR,3

Stops the position counter from recording new data and returns the counter to zero.



### 3.9. Digital/Analog Output Command

### PSOT

The PSOT command is used to set/clear digital output lines (OUT0-OUT15) and/or set the desired voltage (-10.0 V to 10.0 V) of the analog outputs (AOUT1 and AOUT2).

#### SYNTAX:

```
PSOT,mode{,bit#,state,... | ,states | ,dac#,voltage,... | ,dac#,v0,vmax,velocity |
,dac#,v0,vmax,position }
```

#### 3.9.1. MODE Arguments for PSOT

The *mode* argument defines one of five possible configurations (0, 1, 2, 4, or 6) for setting digital and analog outputs. The following sections describe those modes.

##### 3.9.1.1. Set Individual Output States

### PSOT,0

“PSOT,0” is used to set individual digital output lines (argument *bit#*) to either a high signal (*state*=1) or a low signal (*state*=0). The syntax for this mode is **PSOT,0,bit#,state**.

#### EXAMPLE:

```
PSOT,0,4,1           ;Turns output #4 ON
PSOT,0,4,0           ;Turns output #4 OFF
PSOT,0,3,0,13,1,4,1,10,0 ;Turns output #3 OFF, turns output #13 ON,
; turns output #4 ON, turns output #10 OFF,
; at the same time
```

##### 3.9.1.2. Set a Group of Output States

### PSOT,1

“PSOT,1” is used to set a group of digital output lines high or low using a hexadecimal or decimal number specified by the *states* argument. The syntax for this mode is **PSOT,1,states**.

#### EXAMPLE:

```
PSOT,1,0x93AD       ; Turns output #0 ON, turns output #1 OFF,
; turns output #2 ON, turns output #3 ON,
; turns output #4 OFF, turns output #5 ON,
; turns output #6 OFF, turns output #7 ON,
; turns output #8 ON, turns output #9 ON,
; turns output #10 OFF, turns output #11 OFF,
; turns output #12 ON, turns output #12 OFF,
; turns output #14 OFF, turns output #15 ON,
; at the same time
```

### 3.9.1.3. Set Analog Outputs to Discrete Values

PSOT,2

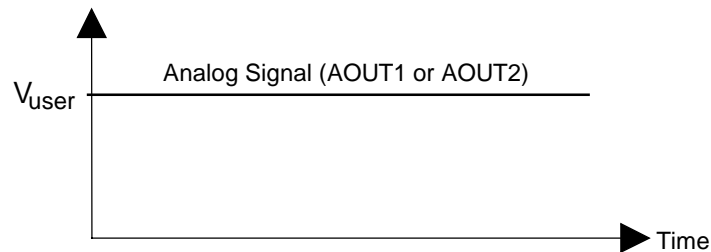
“PSOT,2” is used to set the output voltage of DAC0 (*dac#=0*) which is AOUT1 or DAC1 which is AOUT2 (*dac#=1*) to a constant value. Refer to Figure 3-12. DAC output voltages can range from -10.0 V to 10.0 V and have a minimum step size of 0.3 mV. The syntax for this mode is **PSOT,2,*dac#,voltage***.



The outputs (AOUT1 and AOUT2) are 16 bit D/A's.

#### EXAMPLE:

```
PSOT,2,0,5.25 ;AOUT1 is 5.25 VDC
PSOT,2,1,-3.70 ;AOUT2 is -3.70 VDC
```



(Voltage between  $\pm 10$ VDC can be specified to the D/A)

Figure 3-12. User-Specified Analog Voltage

### 3.9.1.4. Velocity Ramping

PSOT,4

“PSOT,4” is used to set the output voltage of DAC0 (*dac#=0*) which is AOUT1 or DAC1 which is AOUT2 (*dac#=1*) to a value that is proportional to the velocity (velocity ramping). Refer to Figure 3-13. DAC output voltages can range from a programmable minimum value (at zero velocity) specified by argument *v0* to a maximum voltage (at target velocity *velocity*) specified by argument *vmax*. The syntax for this mode is **PSOT,4,*dac#,v0,vmax,velocity***.



The target velocity is specified in machine steps per millisecond.

Keep in mind that the PC-PSO works in units of Machine Step/ms. A Machine Step is an integer and can not contain a fraction. Likewise, the velocity is the same way. Therefore, a 16 Machine Step/ms maximum velocity with a minimum voltage of 0 VDC and a maximum of 5 VDC will cause the voltage to step at 0.3125 VDC increments  
 $([5 - 0] / 16 = 0.3125)$ , even though a 16 bit D/A can step at 0.0003 VDC increments. So, going at 1 Machine Step/ms is +0.3125 VDC; going at 2 Machine Steps/ms is +0.6250 VDC, etc. For that reason, expect some variation at slow velocities and at velocities that are not integers when converted to Machine Steps/ms. This also includes Velocity Errors when traveling at a constant speed. For this example, the amount of variation will be the incremental voltage, which is 0.3125 VDC.

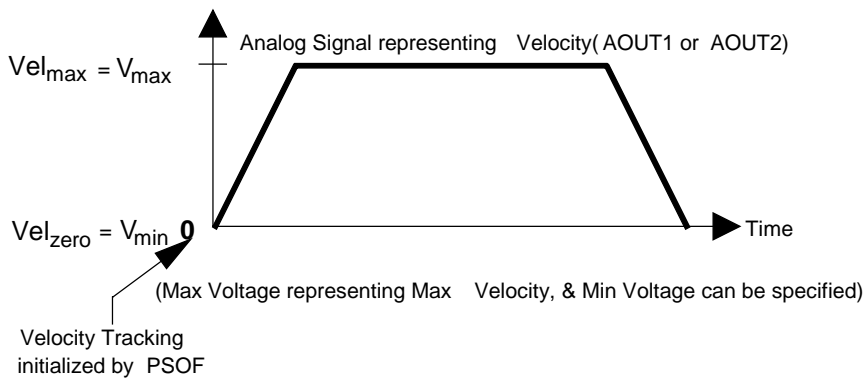


Velocity Tracking is engaged by the "PSOF,3", "PSOF,4", or "PSOF,5" command. Keep in mind for the PSOF to work it must have a pulse defined with the PSOP, PSOD, and/or PSOM commands. One must define a pulse although they are not using it.



**EXAMPLE:**

- PSOP,0,105 ; Single Pulse 10.5 ms wide
- PSOD,0,5 ; That Pulse every 5 Machine Steps
- PSOT,4,0,0,5,16 ; Velocity Tracking using D/A #0 = AOUT1  
 ; min voltage = 0 VDC, max voltage = +5 VDC  
 ; max velocity = 16 Mach Steps/ms
- PSOF,3,X,Y ; Activate Pulse Firing & Vel Track.



**Figure 3-13. Velocity Ramping**

**3.9.1.5. Position Ramping**

**PSOT,6**

“PSOT,6” is used to set the output voltage of DAC0 (*dac#=0*) which is AOUT1 or DAC1 which is AOUT2 (*dac#=1*) to a value that is proportional to the position (position ramping). Refer to Figure 3-14. DAC output voltages can range from a programmable minimum value (at initial position) specified by argument *v0* to a maximum voltage (at target position) specified by argument *vmax*. The syntax for this mode is **PSOT,6,dac#,v0,vmax,position**.



Keep in mind that the PC-PSO works in units of Machine Steps for position. A Machine Step is an integer and can not contain a fraction. Therefore, a 250 Machine Step target position with a minimum voltage of -1 VDC and a maximum of 8 VDC will cause the voltage to step at 0.0360 VDC increments ( $[8 - (-1)] / 250 = 0.0360$ ), even though a 16 bit D/A can step at 0.0003 VDC increments. So, when at 1 Machine Step beyond the current position the voltage is -0.9640 VDC; when at 2 Machine Steps the voltage is -0.9280 VDC, etc. For that reason, expect some variation in your analog signal as you step at 0.0360 VDC increments. For our example, the amount of variation will be the incremental voltage, which is 0.0360 VDC.

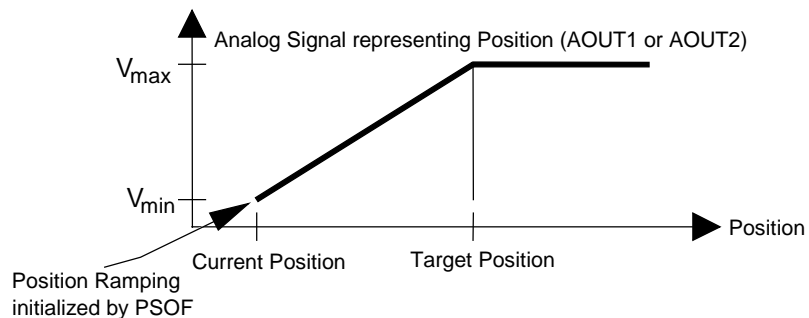


Position Ramping is engaged by the “PSOF,3”, “PSOF,4”, or “PSOF,5” command. Keep in mind for the PSOF to work it must have a pulse defined with the PSOP, PSOD, and/or PSOM commands. You must define a pulse although you are not using it.

**EXAMPLE:**

```

PSOP,0,105           ; Single Pulse 10.5 ms wide
PSOD,0,5             ; That Pulse every 5 Machine Steps
PSOT,6,0,-1,8,250   ; Position Ramping using D/A #0 = AOUT1
                    ; min voltage = -1 VDC, max voltage = +8 VDC
                    ; Target Position = 250 Machine Steps
PSOF,3,X,Y           ; Activate Pulse Firing & Pos Ramping
    
```



(Max Voltage representing Max Position, & Min Voltage can be specified)

**Figure 3-14. Position Ramping**

### 3.9.2. PSOT Arguments

The arguments used by the PSOT command vary based on the *mode* that is being used (refer to Section 3.9.1. Mode Arguments for PSOT). The following sections give a summary of all arguments used by the PSOT,*mode* command.

#### 3.9.2.1. *bit#* Argument

The “*bit#*” argument specifies an individual bit number that corresponds to one of the 16 digital outputs (0=OUT0, 1=OUT1, ..., 15=OUT15). This argument is only used by the command PSOT,0.

#### 3.9.2.2. *state* Argument

The “*state*” argument specifies the digital state (0=low, 1=high) of the previous output line (specified by the *bit#* argument). This argument is used by the PSOT,0 and PSOT,1 commands. Multiple groups of bit numbers and states may be specified.

#### 3.9.2.3. *states* Argument

The “*states*” argument specifies a single hexadecimal or binary number that defines the desired output states of the 16 digital outputs. OUT0 corresponds to the least significant bit of *states* and OUT15 corresponds to the most significant bit of *states*. For example, the programming command PSOT,1,0xA3C1 would set the output bits (OUT15-OUT0) as follows: 1010 0011 1100 0001 (1010=A, 0011=3, 1100=C and 0001=1). The *states* argument may also be in decimal format (41921).

#### 3.9.2.4. *dac#* Argument

The *dac#* argument specifies which of the two digital-to-analog converters (0 or 1) the user wants to set. The value 0 corresponds to AOUT1 and the value 1 corresponds to AOUT 2. This argument is only used by commands PSOT,2, PSOT,4 and PSOT,6.

#### 3.9.2.5. *voltage* Argument

The “*voltage*” argument defines the desired output voltage (-10.0 to 10.0 volts) of the specified DAC channel (0 or 1). Both DAC channels and their respective voltages may be listed in the same PSOT command (e.g., PSOT,2,0,5,1,-5). This argument is only used by command PSOT,2.

#### 3.9.2.6. *v0* Argument

The “*v0*” argument specifies the zero-state analog output voltage for proportional output modes PSOT,4 (*v0* is the analog output voltage at zero velocity) and PSOT,6 (*v0* is the analog output voltage at the initial position). This argument can range from -10.0 volts to 10.0 volts and is used only by modes PSOT,4 and PSOT,6.

### 3.9.2.7. *vmax* Argument

The “*vmax*” argument specifies the maximum analog output voltage at target velocity *velocity* in mode PSOT,4 or at the target position *position* in mode PSOT,6. This argument can range from -10.0 volts to 10.0 volts and is used only by modes PSOT,4 and PSOT,6.

### 3.9.2.8. *velocity* Argument

The “*velocity*” argument defines the target velocity (in machine steps per millisecond) at which the analog output defined by *dac#* will be at its maximum (as defined by *vmax*). Velocity can range from  $-2^{23}$  to  $2^{23}-1$  machine steps per millisecond. This argument is used only in mode PSOT,4.

### 3.9.2.9. *position* Argument

The “*position*” argument defines the target position (in machine steps) at which the analog output defined in *dac#* will be at its maximum (as defined by *vmax*). *Position* can range from  $-2^{23}$  to  $2^{23}-1$  machine steps. This argument is used only in mode PSOT,6.



Programming an output bit to a logical “1” state results in the corresponding pin being pulled to ground. A programmed “0” state results in a high impedance state on the corresponding output pin. During a reset, all outputs go to the high impedance state. All input lines are pulled to +5V by a 10K $\Omega$  resistor.

#### EXAMPLE:

```

PSOT,0,0,1,1,1,2,1,3,1 ; Sets digital output lines OUT0-OUT3 high.
PSOT,1,0x4AB9          ; Sets output lines to 0100 1010 1011 1001
                        ; (1=high,0=low)
PSOT,2,0,2.5          ; Sets DAC0 (AOUT1) to 2.5 V.
PSOT,4,0,3,7,125     ; Sets DAC0 (AOUT1) for velocity ramping
                        ; AOUT1 ranges proportionally from 3 V to 7 V
                        ; as velocity ranges from 0 to 125 machine steps/ms.
PSOP,5                ; Toggle mode laser firing
V0=300                ; Laser off for 300 machine steps
V1=700                ; Laser on for 700 machine steps
PSOD,1,0,2            ; Incremental firing distance with 2 elements (V0 & V1)
PSOF,3,X,Y            ; Enable firing, track vector velocity in the X-Y plane
:                      ; Motion commands, post-processing, etc.

```

### 3.10. Programming Tools

All UNIDEX 600 PSO-PC programming support is provided as part of the standard Util600-NT/95 library software package (see EDU156 – the U600 library reference manual).



Toolkit is a Windows software interface that is included with the UNIDEX 500 system. This program provides an interface between the user and the UNIDEX 500 system by providing such features as easy-to-use menu bars, axis position displays, machine control buttons, a built-in text editor for writing motion control programs, a display window, etc. It may be desirable to create a customized operator interface, instead of using the Toolkit Windows interface.

When the user buys a UNIDEX 500, installation disk 2 contains the library files. These library (.LIB) files are provided for C and BASIC programs. Different versions of .LIB files are provided for different compiler manufacturers and different compiler versions. Depending upon the application, be sure to use the appropriate library file. For additional information, refer to the help files under the appropriate \LIB (library) directory after installing Toolkit.

If the user plans to write a customized software interface program rather than use the Toolkit program included with the UNIDEX 500, it will be necessary to perform the following functions: downloading arrays of information, sending PSO commands to the PSO-PC board, and initializing the PSO-PC board. These functions can be accomplished using C library functions `aer_qpsodlarray`, `aer_send` and `aer_initialize`. These functions are explained in the sections that follow. Also, there are Windows functions for `aer_send` and `aer_initialize`. Refer to the *UNIDEX 500 Operation and Technical Manual P/N EDU150* for additional information.

The functions `aer_send` and `aer_initialize` have additional arguments and return codes that are not listed in this manual. These arguments and return codes do not directly relate to the PSO-PC board and have therefore been omitted. Refer to the UNIDEX 500 Operation and Technical Manual (EDU150) for additional information.



If the user is using the Windows functions and PSO commands that require the DPRAM such as “PSOD,1”, “PSOD,2”, “PSOM,0”, and “PSOP,3”, then the Windows function `WAPIAerSetPSOPtr()` must be called following the `WAPIAerInitialize()` function.



### 3.10.1. aery\_psodlarray

This function is used to download a large array of firing distances (up to 40,960 23-bit firing distance points) to the PSO-PC board through the dual-ported RAM. This is done in blocks of 639 points. The size of each incremental point must be less than  $2^{23}$  (8,388,608). The data is verified automatically by a checksum that is calculated after the last point is downloaded.

*SHORT aery\_psodlarray( SHORT board, long \_far \*data, long timeout, long nPoints*

where board = the board number in the C library interface (0-6)  
 \*data = pointer to the integer firing distances  
 timeout = communications check cycle counter (100,000 typical)  
 nPoints = number of points to be downloaded



The PSO-PC must have the large RAM option (part number PSO-PC-2) installed to use this option.

This function returns the following values:

- 0 = the data transfer is complete
- 1 = time-out occurred while waiting for the “ready” flag to clear in DPRAM. The “ready” flag is bit #0 at the top of DPRAM.
- 2 = too many data points (>40,960) or negative number (<0)
- 3 = checksum error occurred after data transmission
- 4 = address not set in parameter file
- 5 = *reserved.*



This function is available only through the C libraries. An attempt to use this function without the optional RAM will cause RAM corruption on the PSO-PC board. If such RAM corruption occurs, the system must be re-initialized.



This function transfers data points at a rate of 100 KHz with a 80486 CPU running at 33 MHz.



### 3.10.2. aer\_send

This function sends a UNIDEX 500 command to the queue buffer. All PSO commands can be sent to the PSO-PC board using this command. The syntax is as follows:

**SYNTAX:**

```
int aer_send("ps0 command here") ; send new PSO command
int aer_retry(void) ; if aer_send command got a 102 bus busy
; error message, use this command to retry
```

This function returns the following values:

0	=	command taken/accepted
1	=	command name error
2	=	incomplete/incorrect argument
3	=	requires a separate block
9	=	wait for previous command to finish
10	=	can't open file
14	=	this board # not defined
22	=	PSO mailbox doesn't have enough empty space
101	=	bus time out
102	=	bus busy
103	=	system is not initialized
126	=	message input command

### 3.10.3. aer\_initialize

This function initializes the PSO-PC board if (1) the dual-ported RAM base address (parameter 015) does not equal 0, (2) the host address (parameter 016) does not equal 0, and (3) the file PCPSO.FRM exists in the same directory as the UNIDEX 500 firmware file U500.JWP.

**SYNTAX:**

```
int aer_initialize( )
```

This function returns the following error codes relative to the PSO-PC initialization process:

- 11 = PSO-PC time-out occurred
- 12 = PSO-PC firmware file (PCPSO.FRM) is corrupted
- 13 = PSO-PC checksum error has occurred
- 14 = PSO-PC communications time-out

▽ ▽ ▽

**CHAPTER 4: TECHNICAL DETAILS**

<b>In This Section:</b>	
• Introduction .....	4-1
• Specifications and Ratings.....	4-2
• Main Connector (P1) Pinouts .....	4-3
• Aerotech Bus (P3) Pinouts .....	4-4
• Intel iSBX Bus (P4) Pinouts.....	4-5
• Opto-22 I/O Connector (P5) Pinouts.....	4-6
• Single-ended Encoder Interface Connector (P6) Pinouts .....	4-7
• Differential Encoder Interface Connector (P7) Pinouts.....	4-9
• Test Points.....	4-10
• Analog Output Gain Adjustments.....	4-11

**4.1. Introduction**

This chapter supplies technical specifications for the PSO-PC board. Connector pinouts, test points, input/output ratings, etc. are found in this chapter. For details on setting jumpers on the PSO-PC board, refer to Chapter 2: Getting Started.

## 4.2. Specifications and Ratings

Below are specifications and ratings that pertain to the PSO-PC board.

### Specifications

Number of single encoder channels.....	4
Two analog outputs (16 bit) .....	-10 to +10 Volts
Analog input range (12 bit) (Optional).....	-10 to +10 Volts
Digital I/O - Opto-22 compatible .....	16 out, 8 in
Digital Signal Processor (DSP) .....	20 MHz
Single axis scan time .....	500 KHz
Interface to internal encoder bus .....	P6 and P7
iSBX Expansion port.....	P4

### Absolute Maximum Ratings

Encoder channel maximum input frequency (quad) .....	5 MHz
Encoder channel maximum data rate.....	20 MHz
Analog Output - Current drive .....	± 100 mA
Analog Output - Power supply input voltage .....	± 18 V
Analog Output - Power supply input current.....	250 mA
Digital output sink current.....	20 mA
Maximum Operating Temperature .....	100° F

### 4.3. Main Connector (P1) Pinouts

Table 4-1 lists the signals and descriptions of the pins on the main connector (P1) of the PSO-PC board. This connector extends through the back of the PC and consists of a DB37 female connector.

**Table 4-1. Main Connector (P1) Pinouts**

Pin	Signal	Signal Description	Pin	Signal	Signal Description
1	AOUT1	Analog output 1 ( $\pm 10V$ )	20	AOUT2	Analog output 2 ( $\pm 10V$ )
2	VASUP+	Analog supply + (12 to 15V)	21	VASUP-	Analog supply (-12 to -15V)
3	VACOM	Analog common	22	VACOM	Analog common
4	SHIELD	Frame connection	23	NC	No connection
5	COM	Signal common	24	COM	Signal common
6	OUT0	TTL output 0	25	OUT1	TTL output 1
7	OUT2	TTL output 2	26	OUT3	TTL output 3
8	COM	Signal common	27	COM	Signal common
9	IN0	TTL input 0	28	IN1	TTL input 1
10	IN2	TTL input 2	29	IN3	TTL input 3
11	SHIELD	Frame connection	30	NC	No connection
12	AIN	Analog input	31	COM	Signal common
13	COM	Signal common	32	COM	Signal common
14	SHIELD	Frame connection	33	NC	No connection
15	LSUP+	Laser input supply (+5V)	34	LSUP+	Laser input supply (+5V)
16	LCOM	Laser output common	35	LCOM	Laser output common
17	LOUT1	Laser output signal 1	36	LOUT2	Laser output signal 2
18	READY	Laser output	37	LASEN_N	Enable input
19	SHIELD	Frame connection			

#### 4.4. Aerotech Bus (P3) Pinouts

Table 4-2 lists the signals and descriptions of the pins on the Aerotech bus (P3) of the PSO-PC board. This connector is located on the upper left corner of the PSO-PC board and connects to the motion controller card (e.g., the U500 or U600). PSO commands are sent from the motion controller to the PSO-PC board via this connector.

**Table 4-2. Aerotech Bus (P3) Pinouts**

Pin	Signal	Signal Description	Pin	Signal	Signal Description
1	MA3	I/O address line 3	26	D4	Data line 4
2	GND	Ground	27	OPT1	Option line 1 ( <i>unused</i> )
3	MA4	I/O address line 4	28	D3	Data line 3
4	GND	Ground	29	OPT0	Option line 0 ( <i>unused</i> )
5	MCLK	Timing Signal	30	D2	Data line 2
6	MRESET	Initialize PSO-PC board	31	MA11	I/O address line 11
7	MA5	I/O address line 5	32	D1	Data line 1
8	MA2	I/O address line 2	33	D16	Data line 16
9	MA6	I/O address line 6	34	D0	Data line 0
10	MA1	I/O address line 1	35	D17	Data line 17
11	MINTR1	Interrupt request 1	36	D18	Data line 18
12	MA0	I/O address line 0	37	D19	Data line 19
13	MINTR0	Interrupt request 0	38	D20	Data line 20
14	IOWR	I/O write command line	39	D21	Data line 21
15	MA7	I/O address line 7	40	D22	Data line 22
16	IORD	I/O read command line	41	D23	Data line 23
17	MA8	I/O address line 8	42	D14	Data line 14
18	GND	Ground	43	MWAIT	Base board CPU wait for PSO-PC to finish requested operation
19	MCS1	I/O chip select signal 1	44	D12	Data line 12
20	D7	Data line 7	45	D15	Data line 15
21	MCS0	I/O chip select signal 0	46	D10	Data line 10
22	D6	Data line 6	47	D13	Data line 13
23	MA9	I/O address line 9	48	D8	Data line 8
24	D5	Data line 5	49	D11	Data line 11
25	MA10	I/O address line 10	50	D9	Data line 19

#### 4.5. Intel iSBX Bus (P4) Pinouts

Table 4-3 lists the signals and descriptions of the pins on the Intel iSBX bus connector (P4) of the PSO-PC board. This connector is located at the bottom left of the PSO-PC board.

**Table 4-3. Intel iSBX Bus (P4) Pinouts**

Pin	Signal	Signal Description	Pin	Signal	Signal Description
43	MD8	Bidirectional data line 8	44	MD9	Bidirectional data line 9
41	MDA	Bidirectional data line 10	42	MDB	Bidirectional data line 11
39	MDC	Bidirectional data line 12	40	MDD	Bidirectional data line 13
37	MDE	Bidirectional data line 14	38	MDF	Bidirectional data line 15 (MSB)
35	GND	Ground	36	+5V	Power supply voltage ( $\pm 5\%$ )
33	MD0	Bidirectional data line 0 (LSB)	34	MDRQT	Direct memory access (DMA) cycle request from PSO-PC to base board
31	MD1	Bidirectional data line 1	32	MDACK/	DMA cycle acknowledgment
29	MD2	Bidirectional data line 2	30	OPT0	Option line 0 ( <i>unused</i> )
27	MD3	Bidirectional data line 3	28	OPT1	Option line 1 ( <i>unused</i> )
25	MD4	Bidirectional data line 4	26	TDMA	Terminate DMA activity
23	MD5	Bidirectional data line 5	24	<i>reserved</i>	
21	MD6	Bidirectional data line 6	22	MCS0/	I/O chip select signal 0
19	MD7	Bidirectional data line 7	20	MCS1/	I/O chip select signal 1
17	GND	Ground	18	+5V	Power supply voltage ( $\pm 5\%$ )
15	IORD/	I/O read command line	16	MWAIT/	Base board CPU wait for PSO-PC to finish requested operation
13	IOWRT/	I/O write command line	14	MINTR0	Interrupt request 0 to base board
11	MA0	I/O address line 0	12	MINTR1	Interrupt request 1 to base board
9	MA1	I/O address line 1	10	<i>reserved</i>	
7	MA2	I/O address line 2	8	MPST/	Signal that PSO-PC board is installed
5	RESET	Initialize PSO-PC board	6	MCLK	Timing signal
3	GND	Ground	4	+5V	Power supply voltage ( $\pm 5\%$ )
1	+12V	Power supply voltage ( $\pm 5\%$ )	2	-12V	Power supply voltage ( $\pm 5\%$ )

**4.6. Opto-22 I/O Connector (P5) Pinouts**

Table 4-4 lists the signals and descriptions of the odd-numbered pins on the Opto-22 Input/Output connector (P5). This connector is located at the upper right of the PSO-PC board. The even-numbered pins (2-50) are all connected to ground.

**Table 4-4. Opto-22 I/O Connector (P5) Pinouts**

Pin	Signal	Description	Pin	Signal	Description
1	IN7	Digital input 7	27	OUT10	Digital output 10
3	IN6	Digital input 6	29	OUT9	Digital output 9
5	IN5	Digital input 5	31	OUT8	Digital output 8
7	IN4	Digital input 4	33	OUT7	Digital output 7
9	IN3	Digital input 3	35	OUT6	Digital output 6
11	IN2	Digital input 2	37	OUT5	Digital output 5
13	IN1	Digital input 1	39	OUT4	Digital output 4
15	IN0	Digital input 0	41	OUT3	Digital output 3
17	OUT15	Digital output 15	43	OUT2	Digital output 2
19	OUT14	Digital output 14	45	OUT1	Digital output 1
21	OUT13	Digital output 13	47	OUT0	Digital output 0
23	OUT12	Digital output 12	49	+5V	Power supply voltage
25	OUT11	Digital output 11			



#### 4.7. Single-ended Encoder Interface Connector (P6) Pinouts

Table 4-5 lists the signals and descriptions of the pins on the single-ended encoder interface connector (P6) of the PSO-PC board. This connector is located in the top center portion of the PSO-PC board and consists of a 26-pin header. A user-supplied cable (see Figure 4-1) may be used to interface to external encoders (P6 or P7).

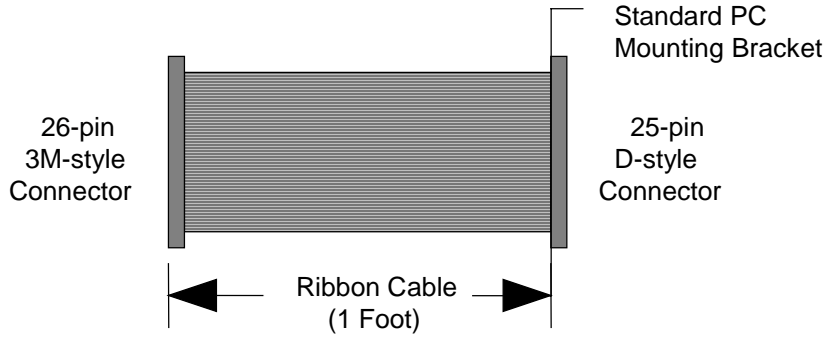


Figure 4-1. User-Supplied Encoder Interface Cable

**Table 4-5. Single-ended Encoder Interface Connector (P6) Pinouts**

Pin	Signal	Signal Description	Pin	Signal	Signal Description
1	CH1A	SIN or clk1	2	CH1B	COS or direction 1
3	NC	No connection	4	NC	No connection
5	CH2A	SIN or clk2	6	CH2B	COS or direction 2
7	NC	No connection	8	NC	No connection
9	CH3A	SIN or clk3	10	CH3B	COS or direction 3
11	NC	No connection	12	NC	No connection
13	CH4A	SIN or clk4	14	CH4B	COS or direction 4
15	NC	No connection	16	NC	No connection
17	NC	No connection	18	GND	Common
19	NC	No connection	20	GND	Common
21	NC	No connection	22	GND	Common
23	NC	No connection	24	GND	Common
25	SHIELD	Frame connection	26	+5V	Power

Table 4-6 lists the 25-pin D-type connector pinouts.

**Table 4-6. Single-ended Encoder Interface 25-pin D-type Connector**

Pin	Signal	Signal Description	Pin	Signal	Signal Description
1	CH1A	SIN or clk1	14	CH1B	COS or direction 1
2	NC	No connection	15	NC	No connection
3	CH2A	SIN or clk2	16	CH2B	COS or direction 2
4	NC	No connection	17	NC	No connection
5	CH3A	SIN or clk3	18	CH3B	COS or direction 3
6	NC	No connection	19	NC	No connection
7	CH4A	SIN or clk4	20	CH4B	COS or direction 4
8	NC	No connection	21	NC	No connection
9	NC	No connection	22	GND	Common
10	NC	No connection	23	GND	Common
11	NC	No connection	24	GND	Common
12	NC	No connection	25	GND	Common
13	SHIELD	Frame connection			

**4.8. Differential Encoder Interface Connector (P7) Pinouts**

Table 4-7 lists the signals and descriptions of the pins on the differential encoder interface connector (P7) of the PSO-PC board. This connector is located in the top center portion of the PSO-PC board (to the left of P6) and consists of a 26-pin header.

**Table 4-7. Differential Encoder Interface Connector (P7) Pinouts**

Pin	Signal	Signal Description	Pin	Signal	Signal Description
1	CH1A+	SIN or clk1+	2	CH1B+	COS or direction 1+
3	CH1A-	SIN or clk1-	4	CH1B-	COS or direction 1-
5	CH2A+	SIN or clk2+	6	CH2B+	COS or direction 2+
7	CH2A-	SIN or clk2-	8	CH2B-	COS or direction 2-
9	CH3A+	SIN or clk3+	10	CH3B+	COS or direction 3+
11	CH3A-	SIN or clk3-	12	CH3B-	COS or direction 3-
13	CH4A+	SIN or clk4+	14	CH4B+	COS or direction 4+
15	CH4A-	SIN or clk4-	16	CH4B-	COS or direction 4-
17,19	NC	No connection	18,20	GND	Common
21,23	NC	No connection	22,24	GND	Common
25	SHIELD	Frame connection	26	+5V	Power

Table 4-8 lists the 25-pin D-type connector pinouts.

**Table 4-8. Differential Encoder Interface 25-pin D-type Connector**

Pin	Signal	Signal Description	Pin	Signal	Signal Description
1	CH1A+	SIN or clk1+	14	CH1B+	COS or direction 1+
2	CH1A-	SIN or clk1-	15	CH1B-	COS or direction 1-
3	CH2A+	SIN or clk2+	16	CH2B+	COS or direction 2+
4	CH2A-	SIN or clk2-	17	CH2B-	COS or direction 2-
5	CH3A+	SIN or clk3+	18	CH3B+	COS or direction 3+
6	CH3A-	SIN or clk3-	19	CH3B-	COS or direction 3-
7	CH4A+	SIN or clk4+	20	CH4B+	COS or direction 4+
8	CH4A-	SIN or clk4-	21	CH4B-	COS or direction 4-
9-12	NC	No connection	22-25	GND	Common
13	SHIELD	Frame connection			

### 4.9. Test Points

Test points are located across the top and down the left side of the PSO-PC board. They are used as an aid in troubleshooting the board and to gain easy access to the PSO-PC signals.

Test points for the PSO-PC are listed and explained in Table 4-9.

**Table 4-9. Test Points of the PSO-PC**

TP#	Signal	TP#	Signal
TP4	GND	TP29	BS_N
TP5	MDPRAM_N	TP30	WT_N
TP6	MBSYR_N	TP31	BEN_N
TP7	MIRQR_N	TP32	HRW_N
TP8	MBSYL_N	TP33	PS_N
TP9	MIRQL_N	TP34	DS_N
TP10	MCEL_N	TP35	RD_N
TP11	BSMEMWR_N	TP36	HEN_N
TP12	BSMEMRD_N	TP37	HREQ_N
TP13	BSMEMWR_N	TP38	Pulse 1
TP14	MAE_N	TP39	Pulse 2
TP15	SMEMSL_N	TP40	Encoder Channel 1 phase A
TP16	EIRQ_N	TP41	Encoder Channel 1 phase B
TP17	EBSYR_N	TP42	Encoder Channel 2 phase A
TP18	EDPRAM_N	TP43	Encoder Channel 2 phase B
TP19	EIRQL_N	TP44	Encoder Channel 3 phase A
TP20	EBSYL_N	TP45	Encoder Channel 3 phase B
TP21	ECEL_N	TP46	Encoder Channel 4 phase A
TP22	BEIOWR_N	TP47	Encoder Channel 4 phase B
TP23	WR_N	TP48	Host Address Select
TP24	IRQA_N	TP49	Buffered Analog Input Voltage
TP25	RAMSL_N	TP50	ADSTB_N
TP26	IRQB_N	TP51	Reset
TP27	1 Shot	TP52	VAOUT1
TP28	BIORD_N	TP53	VAOUT2

### 4.10. Analog Output Gain Adjustments

Each of the analog outputs has gain adjustment pot. These pots have been factory adjusted for an output of 5 volts for a 5 volt command. Gain adjustment pot (R36) controls analog output #1 gain and (R35) control analog output #2 gain, refer to Figure 4-2. Turning the gain pots in a clockwise (CW) direction results in an increase in the gain.

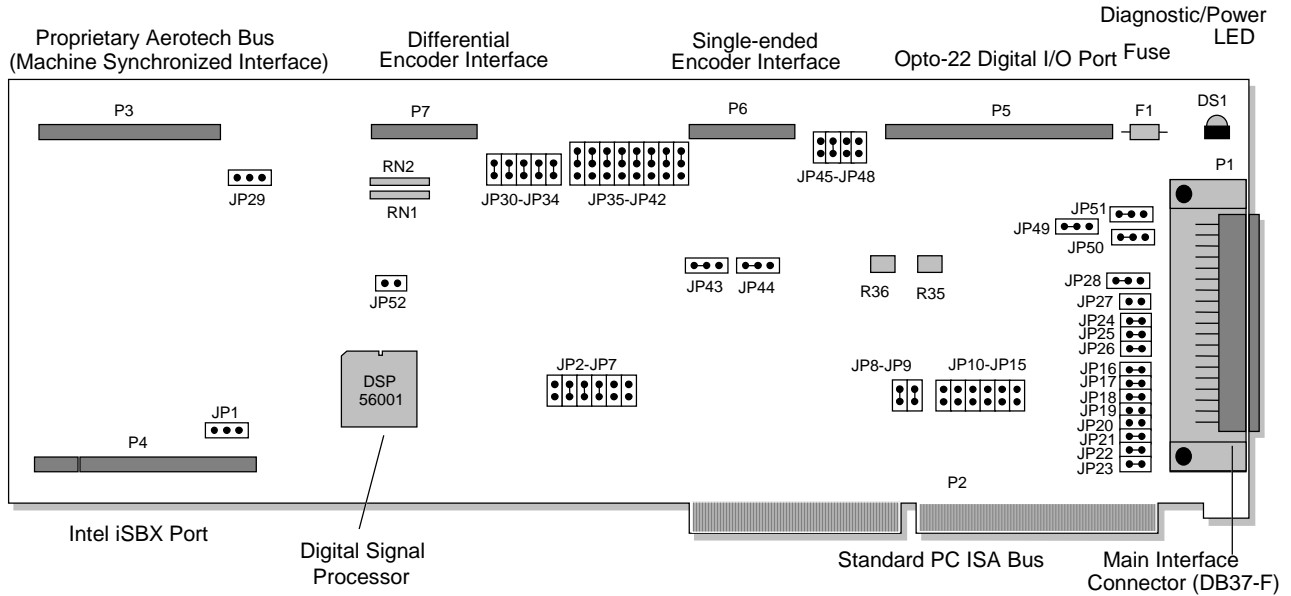


Figure 4-2. PSO-PC Board





## CHAPTER 5: TROUBLESHOOTING

**In This Section:**

- Introduction ..... 5-1
- Installation Problems ..... 5-2

### 5.1. Introduction

If there are any technical support questions, please have the following information available before calling:

1. Customer order number.
2. We may also need to know the type of PC being used (brand name, CPU, available memory), the current version of DOS, and the contents of the AUTOEXEC.BAT and CONFIG.SYS files.
3. If a customized application is being developed, we will need to know what compiler and version number is being used (e.g., Borland C v3.1, Microsoft Visual C, etc.).
4. If at all possible, try to be in front of the system where the problems are occurring.

### 5.2. Installation Problems

Some common problems that relate to installation, startup and communications are listed and diagnosed in Table 5-1.

**Table 5-1. Troubleshooting for Common Installation Problems**

Problem	Possible Causes / Solutions
Initialization failure, communication failure or "Bus Timeout !" error occurs...	The UNIDEX Controller board is not installed. The UNIDEX Controller board is not seated properly. The software address does not match the hardware address. Another device in the PC is set to the same base address as the Controller board. Select a unique base address.
Checksum failure ...	The UNIDEX Controller board is faulty. Another device in the PC is set to the same base address as the Controller board. Select a unique base address.
The board initializes properly, but later fails ...	The UNIDEX Controller board is not seated properly. Another device in the PC is set to the same base address as the Controller board. Select a unique base address.
The PSO LED never lights up after power up ...	The UNIDEX Controller board is not seated properly. Another device in the PC is set to the same base address as the Controller board. Select a unique base address. The power supply has been overloaded and has shut down. Install a larger power supply to correct the problem.
The PSO LED lights after power up and remains on even after attempts to initialize ...	The software base address does not match the base address jumper settings on the UNIDEX Controller board. Another device in the PC is set to the same base address as the Controller board. Select a unique base address.
The PC power supply goes dead when the UNIDEX Controller is installed ...	The power supply has been overloaded and has shut down. Install a larger power supply to correct the problem. External wiring problems exist. Remove the main interface cable and recheck. The UNIDEX Controller board is faulty.
A PC bus timeout error occurs and the Controller LED is blinking...	Try resetting the PC and re-initializing the Controller. Using Rev- firmware (Controller.JWP) with RevC Controller board. Using RevC firmware (Controller.JWP) with Rev- Controller board.
The Opto-22 board does not appear to be getting any power...	Fuse F1 may be blown. Call the Aerotech Technical Support Department for information.
The PSO LED is flashing	The PSO watchdog has failed. Disable the watchdog by setting JP43 to off.





## APPENDIX A: PSO COMMAND SUMMARY

**In This Section:**

- PSO-PC Programming Commands

The following section provides a quick reference for the U500 PSO-PC programming commands. The commands listed in Table A-1 include the command statement syntax and a brief description for reference.

**UNIDEX 600 users refer to the comprehensive *U600 Programming Manual, P/N EDU158* or the online help file within the U600MMI-NT/95 for command help.**



**Table A-1. PSO Command Summary**

Command	Description
PSOC,0	Enables unconditional position tracking
PSOC,1, <i>i</i> , <i>n</i>	Enables conditional position tracking when the current state (0 or 1) of input <i>i</i> (0-7) equals <i>n</i> (0 or 1), while retaining position data when the position counter is disabled.
PSOC,2, <i>i</i> , <i>n</i>	Enables conditional position tracking when the current state (0 or 1) of input <i>i</i> (0-7) equals <i>n</i> (0 or 1), while resetting position data to 0 when the position counter is disabled.
PSOC,3, <i>in_map</i> , <i>out_map</i>	Enables conditional position tracking when the current states (0 or 1) of inputs IN0-IN7 match the states defined in the bit map <i>in_map</i> (0=low, 1=high, x=ignore the input). If no match occurs, then the 16 outputs (OUT0-OUT15) are set according to the bit map <i>out_map</i> (0=low, 1=high, x=output state remains unchanged).
PSOD,0, <i>distance</i>	Specifies a fixed incremental distance ( <i>distance</i> ) given in machine steps at which pulse outputs will occur.
PSOD,1, <i>V#</i> , <i>±m</i>	Specifies incremental distances given in machine steps at which pulse outputs will occur. Firing distances are specified sequentially forward (+) or backwards (-) for a total of <i>m</i> variables starting at the variable number specified by <i>V#</i> (0-9999 corresponds to V0-V9999).
PSOD,2, <i>V#</i> , <i>±m</i>	Specifies absolute distances given in machine steps at which pulse outputs will occur. Firing distances are specified sequentially forward (+) or backwards (-) for a total of <i>m</i> variables starting at the variable number specified by <i>V#</i> (0-9999 corresponds to V0-V9999).
PSOF,0	Disables the output firing pulse train and position tracking features.
PSOF,1	Activate the output firing pulse train as defined by the PSOP command, and continues until disabled by the PSOF,0 command. No position tracking occurs in this mode.
PSOF,2, <i>num</i>	Activate the output firing pulse train as defined by the PSOP command, and continues <i>num</i> number of times. No position tracking occurs in this mode.

Table A-1. PSO Command Summary Continued

Command	Description
PSOF,3,axis1,axis2,axis3	Activates the output firing pulse train as defined by the PSOP command, and locks the position counters onto the motions of up to three specified axes ( <i>axis1</i> , <i>axis2</i> and /or <i>axis3</i> ).
PSOF,4,±dist,axis1,axis2,axis3	Activates the output firing pulse train as defined by the PSOP command, and locks the position counters onto the motions of up to three specified axes ( <i>axis1</i> , <i>axis2</i> and /or <i>axis3</i> ). The output firing pattern is defined by bit mapping using the PSOM command (0=output goes/remains low, 1=output goes/remains high). The pulse output occurs at a fixed incremental distance <i>dist</i> . The ± sign define whether the bit pattern is run forward (“+” or no sign) or backwards (“-” sign).
PSOF,5,±dist,axis1,axis2,axis3	Same as PSOF,4, except that the output firing pattern is defined by bit mapping using the PSOM command, where a bit value of 0 causes no output, and a bit value of 1 causes the output to be a single pulse train.
PSOM,0,V#,±size	<p>Defines a bit pattern that is used in conjunction with commands PSOF,4 and PSOF,5. When used with PSOF,4, the bit pattern specifies when the laser output is to go/remain low (bit=0) or when the laser output is to go/remain high (bit=1). When used with PSOF,5, the bit pattern specifies whether there should be no output (bit=0) or a single pulse train (bit=1). Bit patterns begin with variable number <i>V#</i> (0-9999 corresponds to V0-V9999) and increment (if “+”) or decrement (if “-”) sequentially for a total of <i>size</i> bytes (groups of 8 bits). Note that each variable can contain 4 bytes (32 bits).</p> <p>In the “+” direction, firing data starts with the most significant bit of the 32-bit pattern specified by variable number <i>V#</i> and proceeds sequentially through the least significant bit. The process is repeated for the next variable number <i>V#+1</i>.</p> <p>In the “-” direction, firing data starts with the least significant bit of the 32-bit pattern specified by variable number <i>V#</i> and proceeds sequentially through the most significant bit. The process is repeated for the previous variable number <i>V#-1</i>.</p>
PSOP,0,w	Sets the width <i>w</i> of a single pulse output in tenths of milliseconds (e.g., <i>w</i> =100 means 10.0 ms).
PSOP,1,l,w,t	Sets a single pulse with definable pulse lead <i>l</i> , pulse width <i>w</i> and pulse trail <i>t</i> characteristics. Arguments are specified in tenths of milliseconds (e.g., <i>w</i> =100 means 10.0 ms).
PSOP,2,l,w,t,r,g	Same as PSOP,1, except that ramp up/down ® and pulse gap interval ( <i>g</i> ) characteristics are definable. Arguments <i>r</i> and <i>g</i> are given in tenths of milliseconds. The pulse width starts at <i>r</i> and increments by <i>r</i> units until the pulse width reaches <i>w</i> . At that point, the pulse widths begin to decrease (in decrements of <i>r</i> units) to a width of <i>r</i> units. The pulse gaps can be a fixed width ( <i>g</i> >0) or can be made to ramp (i.e., to match the size of the pulse width) ( <i>g</i> =0).

Table A-1. PSO Command Summary Continued

Command	Description
PSOP,3,V#,±m	Sets a pulse train by defining a series pulse widths and gap widths (in tenths of milliseconds) in a series of direct variables (e.g., V0-V9999) of the UNIDEX 500. The argument <i>var#</i> specifies the direct variable number (0-9999 corresponding to V0-V9999) that contains the first gap width (in tenths of milliseconds). The next sequential variable contains the first pulse width (in tenths of milliseconds).
PSOP,4,w	Sets the width <i>w</i> of a single pulse output in <i>microseconds</i> (with a minimum value of 1 microsecond).
PSOP,5	Sets output toggle mode. For example, if the pulse output is configured to occur at a fixed incremental distance of 500 machine steps (PSOD,0,500), then each 500-step motion is considered to be an event. After the first 500 steps, the pulse output is enabled. After the second 500 steps, the pulse output is disabled (i.e., odd-numbered events enable the pulse output, and even-numbered events disabled the pulse output).
PSOR,0	Clears all previous real-time control data from the counter.
PSOR,1	Stops the position counter from recording new data and retains current data under operator command.
PSOR,3	Stops the position counter from recording new data and returns the counter to zero.
PSOT,0,bit#,state,...	This mode is used to set individual digital output lines (argument <i>bit#</i> ) to either a high signal ( <i>state</i> =1) or a low signal ( <i>state</i> =0).
PSOT,1,states	This mode is used to set a group of digital output lines high or low using a hexadecimal or decimal number specified by the <i>states</i> argument.
PSOT,2,dac#,voltage	This mode is used to set the output voltage of DAC0 ( <i>dac#</i> =0) or DAC1 ( <i>dac#</i> =1) to a constant value. DAC output voltages can range from -10.0 V to 10.0 V and have a minimum step size of 0.3 mV.
PSOT,4,dac#,v0,vmax,velocity	This mode is used to set the output voltage of DAC0 ( <i>dac#</i> =0) or DAC1 ( <i>dac#</i> =1) to a value that is proportional to the velocity (velocity ramping). DAC output voltages can range from a programmable minimum value (at zero velocity) specified by argument <i>v0</i> to a maximum voltage (at target velocity <i>velocity</i> ) specified by argument <i>vmax</i> .
PSOT,6,dac#,v0,vmax,position	This mode is used to set the output voltage of DAC0 ( <i>dac#</i> =0) or DAC1 ( <i>dac#</i> =1) to a value that is proportional to the position (position ramping). DAC output voltages can range from a programmable minimum value (at initial position) specified by argument <i>v0</i> to a maximum voltage (at target position <i>position</i> ) specified by argument <i>vmax</i> .

▽ ▽ ▽



## APPENDIX B: GLOSSARY OF TERMS

**In This Section:**

- Terms Used In This Manual
- Definitions

This appendix contains definitions of terms that are used throughout this manual.

**absolute positioning** - Absolute positioning is positioning that is done with respect to an initial starting position (typically referred to as the home position) and typically uses a standard coordinate system (using [X,Y] coordinates is an example of absolute positioning). In contrast, incremental (or relative) positioning is done using a series of relative moves. These moves are relative to the previous location rather than a single reference point (for example, relative changes in position [ $\Delta X, \Delta Y$ ] are examples of incremental positioning).

**accuracy** - Accuracy is the difference between an expected value and an actual value expressed as a percentage.

**axis** - An axis is a direction along which movement occurs.

**base address** - A base address is a number that represents the memory location in the computer where input/output (I/O) information can be stored. All devices (e.g., the PSO-PC card, the UNIDEX 500 card, network cards, tape backup cards, etc.) within a computer must have unique I/O base addresses.

**batch file** - A batch file is a file that contains a series of commands (e.g., the AUTOEXEC.BAT file is a batch file).

**C** - C is a high-level programming language (developed at Bell laboratories) that is able to manipulate a computer at a low level like assembly language. A C program can be used to write a customized software interface for the UNIDEX 500 and PSO-PC, rather than using the Startup program (and machine programs containing PSO commands) that is supplied with the system. If a customized interface program is created using C, the appropriate C library must be included when the program is compiled.

**DOS** - DOS is an acronym for Disk Operating System--a master control program that runs a computer and acts as a scheduler, providing job, task, data and device management. DOS is a generic term that refers to an operating system. MS-DOS is a single-user operating system for PCs that was designed by Microsoft Corporation. Since the DOS and MS-DOS programs are very, the terms DOS and MS-DOS are frequently interchanged.

**DR500 Chassis/Drive Rack** - The DR500 Chassis (or Drive Rack) is a housing for the axis amplifiers (for microstepping, DC brush and brushless drivers) and the driver power supply. The DR500 is available in rack mount, panel mount and desktop packaging.

**DSP** - DSP is an acronym for Digital Signal Processor - an advanced, high-speed processor technology that analyzes input signals by first converting them into digital data and then uses various algorithms (such as fast Fourier transforms) for analysis. Once a signal has been converted to digital data, the individual components of the signal can be isolated and interpreted more readily than in analog form.

**encoder** - An encoder is a rotary device that transmits a pulsed signal based on the number of revolutions of the device.

**hexadecimal number format** - Hexadecimal number format is a method of representing large numbers using base 16 rather than the standard base 10. In base 16 or hexadecimal number format (often abbreviated "hex"), the number positions represent powers of 16 (rather than powers of 10 in decimal). The decimal number positions (1's, 10's, 100's, 1,000's, 10,000's, etc.) are replaced with hexadecimal number positions (1's, 16's, 256's, 4096's, etc.). Also, while the individual numerals for the decimal system are 0-9, the numerals for the hexadecimal number system (which requires 16 unique "numerals") are 0-9 then A-F (where  $A_{16}=10_{10}$ ,  $B_{16}=11_{10}$ ,  $C_{16}=12_{10}$ ,  $D_{16}=13_{10}$ ,  $E_{16}=14_{10}$ , and  $F_{16}=15_{10}$ ). For simplicity in this manual, hexadecimal numbers are written with a preceding "0x" rather than using the subscript 16. For example, the hexadecimal number 12A5 is written 0x12A5. Numbers without the preceding "0x" are assumed to be decimal unless otherwise indicated.

**incremental positioning** - Incremental (or relative) positioning is done using a series of relative moves. These moves are relative to the previous location rather than a single reference point (for example, relative changes in position  $[\Delta X, \Delta Y]$  are examples of incremental positioning). In contrast, absolute positioning is positioning that is done with respect to an initial starting position (typically referred to as the home position) and typically uses a standard coordinate system (using  $[X, Y]$  coordinates is an example of absolute positioning).

**IRQ** - IRQ (interrupt request) is a term associated with generating an interrupt request to the PC. A PC has many IRQs (e.g., IRQ3 and IRQ4 are typically configured as COM ports on the PC, IRQ7 is typically configured as the LPT port, et. al.). Some PC boards must have an IRQ configured through software and/or hardware.

**iSBX expansion port** - The iSBX expansion port (P4 on the PSO-PC card) is a standard Intel interface that uses either an 8 or 16 bit data bus and is used primarily for communications-oriented additions to the system. The iSBX expansion port has a communications data rate of approximately 1 MB/sec (1,048,576 bytes per second).

**jumpers** - Jumpers are hardware *ties* that you manually position into different sockets to configure the hardware platform. Jumpers on the UNIDEX 500 board are used to configure the base address, the encoder sampling frequency, termination resistors, and other features.

**LED** - LED is an acronym for light-emitting diode. An LED is a semiconductor diode that converts electrical energy into visible electromagnetic radiation. The PSO-PC board has an LED (visible from the back of the PC after installation) that is used for diagnostic purposes.

**position synchronized output card (PSO-PC)** - The position synchronized output card is an optional PC-bus based card that can be used in conjunction with the U500 (via connection P3 on the U500 card) to provide programmable laser-firing control.

**program** - A program is a set of instructions that are carried out in some predefined logical order. A UNIDEX 500 program is a sequential list of UNIDEX 500 programming commands which tell the U500 control board how to perform specific motions for a particular application as well as PSO-PC control. UNIDEX 500 programs may be created/edited on-line (from within the Program menu of the Startup software) or off-line (using any standard ASCII text editor). U500 program files (which are optional components of project or .PRJ files) use .PRG as their extension.

**quadrature** - Quadrature is the state of two signals that are displaced 90 degrees with respect to each other. In most rotary incremental optical encoders, light (from an LED, for example) is measured after it is passed through slits in a grating disk (which is attached to the axis being measured). Typically, two tracks on the disk have their gratings displaced 90 degrees with respect to each other (that is, the tracks are said to be in quadrature).

**software** - The term software refers to a computer program. Contrast software with hardware, the physical machinery, components and support peripherals through which the software runs.

**time-based motion** - A time-based motion is a motion that arrives at a specified location in a desired amount of time. After the target position of a move is programmed, the controller chooses any speed to achieve that position on time.

**variables** - Variables are programming terms that are used as temporary storage locations for calculations. Direct variables (V0 through V9999) are general purpose, double precision storage locations. Indirect variables (VV0 through VV9999) are used to indirectly address other variables. For example, if V35=999 and V1=35, then you can indirectly address the contents of V35 using the statement V0=VV1. In this case, the contents of V1 (35) is used as an index to V35. The value of V35 (999) is placed in V0.

▽ ▽ ▽





**APPENDIX C: WARRANTY AND FIELD SERVICE****In This Section:**

- Laser Product Warranty
- Return Products Procedure
- Returned Product Warranty Determination
- Returned Product Non-warranty Determination
- Rush Service
- On-site Warranty Repair
- On-site Non-warranty Repair

Aerotech, Inc. warrants its products to be free from defects caused by faulty materials or poor workmanship for a minimum period of one year from date of shipment from Aerotech. Aerotech's liability is limited to replacing, repairing or issuing credit, at its option, for any products which are returned by the original purchaser during the warranty period. Aerotech makes no warranty that its products are fit for the use or purpose to which they may be put by the buyer, whether or not such use or purpose has been disclosed to Aerotech in specifications or drawings previously or subsequently provided, or whether or not Aerotech's products are specifically designed and/or manufactured for buyer's use or purpose. Aerotech's liability or any claim for loss or damage arising out of the sale, resale or use of any of its products shall in no event exceed the selling price of the unit.

Aerotech, Inc. warrants its laser products to the original purchaser for a minimum period of one year from date of shipment. This warranty covers defects in workmanship and material and is voided for all laser power supplies, plasma tubes and laser systems subject to electrical or physical abuse, tampering (such as opening the housing or removal of the serial tag) or improper operation as determined by Aerotech. This warranty is also voided for failure to comply with Aerotech's return procedures.

***Laser Products***

Claims for shipment damage (evident or concealed) must be filed with the carrier by the buyer. Aerotech must be notified within (30) days of shipment of incorrect materials. No product may be returned, whether in warranty or out of warranty, without first obtaining approval from Aerotech. No credit will be given nor repairs made for products returned without such approval. Any returned product(s) must be accompanied by a return authorization number. The return authorization number may be obtained by calling an Aerotech service center. Products must be returned, prepaid, to an Aerotech service center (no C.O.D. or Collect Freight accepted). The status of any product returned later than (30) days after the issuance of a return authorization number will be subject to review.

***Return Procedure***

After Aerotech's examination, warranty or out-of-warranty status will be determined. If upon Aerotech's examination a warranted defect exists, then the product(s) will be repaired at no charge and shipped, prepaid, back to the buyer. If the buyer desires an air freight return, the product(s) will be shipped collect. Warranty repairs do not extend the original warranty period.

***Returned Product  
Warranty Determination***

***Returned Product Non-warranty Determination***

After Aerotech's examination, the buyer shall be notified of the repair cost. At such time the buyer must issue a valid purchase order to cover the cost of the repair and freight, or authorize the product(s) to be shipped back as is, at the buyer's expense. Failure to obtain a purchase order number or approval within (30) days of notification will result in the product(s) being returned as is, at the buyer's expense. Repair work is warranted for (90) days from date of shipment. Replacement components are warranted for one year from date of shipment.

***Rush Service***

At times, the buyer may desire to expedite a repair. Regardless of warranty or out-of-warranty status, the buyer must issue a valid purchase order to cover the added rush service cost. Rush service is subject to Aerotech's approval.

***On-site Warranty Repair***

If an Aerotech product cannot be made functional by telephone assistance or by sending and having the customer install replacement parts, and cannot be returned to the Aerotech service center for repair, and if Aerotech determines the problem could be warranty-related, then the following policy applies:

Aerotech will provide an on-site field service representative in a reasonable amount of time, provided that the customer issues a valid purchase order to Aerotech covering all transportation and subsistence costs. For warranty field repairs, the customer will not be charged for the cost of labor and material. If service is rendered at times other than normal work periods, then special service rates apply.

If during the on-site repair it is determined the problem is not warranty related, then the terms and conditions stated in the following "On-Site Non-Warranty Repair" section apply.

***On-site Non-warranty Repair***

If any Aerotech product cannot be made functional by telephone assistance or purchased replacement parts, and cannot be returned to the Aerotech service center for repair, then the following field service policy applies:

Aerotech will provide an on-site field service representative in a reasonable amount of time, provided that the customer issues a valid purchase order to Aerotech covering all transportation and subsistence costs and the prevailing labor cost, including travel time, necessary to complete the repair.

***Company Address***

Aerotech, Inc.  
101 Zeta Drive  
Pittsburgh, PA 15238-2897  
USA

Phone: (412) 963-7470  
Fax: (412) 963-7459

▽ ▽ ▽

**A**

Accessories  
 OP500, 4-9  
 Addresses, Setting, 2-7  
 aer\_initialize, 3-33  
 aer\_initialize Function, 3-36  
 aer\_send Function, 3-33  
 aer\_send Function, 3-35  
 Aerotech Bus (P3), 1-3, 1-7  
 Aerotech Bus (P3) Pinouts, 4-4  
 aerq\_psodlarray Function, 3-33, 3-34  
 AIN, 1-8  
 AIN Input, 1-4  
 Analog Output  
   current drive, 4-2  
   power supply input current, 4-2  
   power supply input voltage, 4-2  
 Analog Output (Floating), 2-20  
 Analog Output Control Using PSOT Command, 3-3  
 Analog Outputs  
   optical isolation, 2-21  
 Analog Outputs (16-bit), 4-2  
 Analog/Digital Output Command (PSOT), 3-27  
 Antistatic Bag, 2-1  
 Antistatic Precautions, 2-22  
 Antistatic Wrapping, 2-1  
 AOUT1, 1-8  
 AOUT1 and AOUT2, 2-20  
   port 1 external power connections, 2-20  
 AOUT1 and AOUT2 Outputs, 1-4  
 AOUT2, 1-8  
 Arrays of Firing Points  
   transferring, 2-7  
 Axis Positions  
   acquiring and analyzing, 1-1  
   maximum input frequency for analyzing, 1-1  
 Axis Scan Time, 4-2

**B**

Base Address Jumpers, 2-3, 2-8  
 Bipolar Power Supply (External), 2-20  
 Bit Mapping, 3-3, 3-19

**C**

C  
 \WINDOWS\MSD.EXE Diagnostic Program, 2-7  
 C Library Functions, 3-33, 3-35  
 Cable for Encoder Interfacing, 4-7  
 Cables, 1-6, 1-8  
   25-pin D style, 2-15  
   50-pin cable for port P3, 3-1  
   customized DB37, 1-4

  Opto-22, 50-pin ribbon, 1-3  
 Cables, User-supplied, 1-3  
 Checksum Error, 3-36  
 Checksum Errors, 3-34  
 Checksum Failure, 5-2  
 Clock/Direction Encoder Data Format, 2-14  
 CMOS (PC's), Reconfiguring, 2-8  
 Commands, 3-3  
 Communications Error, 3-36  
 Communications Failure, 5-2  
 Communications Ports, 2-3, 2-7  
 Communications Problems, 5-1, 5-2  
 Components, 1-1  
   reseating, 2-1  
 Conditional Tracking, 1-3, 3-3, 3-4  
 Connect/Disconnect LASEN Input to DSP Interrupt (JP27), 2-12  
 Connector Types, 1-8  
 Connectors, 2-2  
 Counter Data  
   resetting, 3-4  
   retaining, 3-4  
 Customer Order Number, 5-1

**D**

DACs  
   setting output voltages for, 3-28, A-3  
 Data Rates for Encoders, 2-19  
 DB37 Connector for P1, 1-4  
 Differential Encoder Interface (P7), 1-3  
 Differential Encoder Interface Bus (P7), 1-3  
 Differential Encoder Interface Connector (P7)  
   Pinouts, 4-9  
 Differential Encoders, 1-3, 2-15  
   RS-422 inputs, 2-15  
   termination resistances, 2-15  
 Digital I/O Port (P5), 1-3  
 Digital Output Control Using PSOT Command, 3-3  
 Digital Outputs  
   sink current, 4-2  
 Digital Signal Processing. *See* DSP  
 Digital/Analog Output Control Command (PSOT), 3-27  
 Downloading Data, 3-3  
 Downloading Large Arrays to the PC-PSO, 3-34  
 DPRAM Base Address Jumper Settings (JP16-JP23), 2-7  
 DPRAM Base Address Jumpers (JP16-JP23), 2-7  
 DPRAM IRQ Jumper Settings (JP10-JP15), 2-6  
 DPRAM IRQ Jumpers (JP10-JP15), 2-6  
 DR500, 1-7  
 DSP, 4-2  
 DSP Interrupts, 2-5  
 DSP-to-host PC Interrupt Enable/Disable Jumper (JP8), 2-5  
 Dual Ported RAM, 2-3

**E**

Eddy Current Inspection Applications, 1-2  
 Electrostatic Discharge Precautions, 2-1  
 Enable/Disable Laser Enable Input LASEN (JP28), 2-12  
 Enable/Disable Position Synchronized Output Firing (PSOF Command), 3-12  
 Encoder Bus Interface, 4-2  
 Encoder Connections, 2-14  
 Encoder Data Format  
   clock/direction, 2-14  
   SIN/COS, 2-14  
 Encoder Data Format Selection Jumpers (JP30-JP33), 2-14  
 Encoder Direction Jumper (Global), 2-15  
 Encoder Input Frequency, 4-2  
 Encoder Input Signals, 2-18  
 Encoder Interface Cable, 4-7  
 Encoder Interfaces (P6 and P7), 1-3  
 Encoder Maximum Data Rate, 4-2  
 Encoder Sampling Frequency  
   default, 2-18  
   for high resolution encoders, 2-18  
   for low resolution encoders, 2-18  
   use with signal multipliers, 2-18  
 Encoder Sampling Frequency Jumper, 2-19  
 Encoder Sampling Frequency Jumpers (JP45-JP48), 2-18  
 Encoder Sampling Process, 2-18  
 Encoder Sampling Rates, 2-18  
 Encoder Type Jumper Settings (JP35-JP42), 2-16  
 Encoder Type Jumpers, 1-3  
 Encoder Type Jumpers (JP35-JP42), 2-15  
 Encoders, 2-15  
   differential, 1-3  
   input diagrams, 2-15  
   quadrature signal, 2-18  
   sampling frequency, 2-18  
     default, 2-18  
     for high resolution encoders, 2-18  
     for low resolution encoders, 2-18  
     using a multiplier box, 2-18  
   sampling rates, 2-18  
   Single-ended, external, 1-3  
   termination resistances for differential types, 2-15  
   using a combination of single-ended and differential, 1-3  
 Expansion Slot, 2-22  
 External Single-ended Encoders, 1-3

**F**

Features, 1-2  
 Field Service Policy, C-1  
 Firing Distance  
   calculations using multiple axes, 3-11  
   maximum, 3-10  
 Firing Distance Command, 3-7  
 Firing Distance Entry, 3-3

Firing Distances  
   downloading using C library function, 3-34  
 Firing Points  
   transferring, 2-7  
 Force/Flow Control System Applications, 1-2

**G**

Getting Started, 2-1  
 Global Encoder Direction Jumper (JP34), 2-15

**H**

Host Base Address Jumper Settings, 2-4  
 Host Base Address Jumpers, 2-3  
 Host Interface, 2-3  
 Host Interrupts, 2-3  
 Host Reset Capability Enable/Disable Jumper (JP44), 2-17

**I**

I/O Summary, 1-8  
 IN0-IN3, 1-8  
 IN0-IN3 Inputs, 1-4  
 IN0-IN7 Inputs, 1-3, 1-7  
 Initialization, 3-36  
 Initialization Failure, 5-2  
 Input Frequency for Encoders, 2-19  
 Inputs, 4-2  
   "i" argument, 3-5  
   Analog Input AIN, 1-4  
   conditional tracking based on the states of, 3-4  
   IN0-IN3, 1-4  
   IN0-IN7, 1-3, 1-7  
   Laser Shutoff, 1-4  
 Inputs and Outputs  
   available on P1, 1-3  
   duplication between P1 and P4, 1-3  
   duplication between P1 and P5, 1-3  
   optional, 2-23  
   optional using Opto-22 Board on P5, 1-3  
   Opto-22 board, 1-7  
 Installation, 2-22  
 Installation Problems, 5-1, 5-2  
 Intel Bus (P4) Pinouts, 4-5  
 Intel iSBX Port, 2-2  
 Interfaces, Typical, 1-6  
 Internal Encoder Bus, 1-3  
 Internal/External Power Selection Jumpers (JP49-JP51), 2-20  
 Interrupt Requests, 2-13  
 Interrupt the PC, 2-6  
 IRQ Settings, 2-6  
 IRQ10, 2-6  
 IRQ11, 2-6  
 IRQ12, 2-6  
 IRQ3, 2-6

IRQ4, 2-6  
 IRQ7, 2-6  
 ISA Bus, 1-3  
 iSBX Bus (P4) Pinouts, 4-5  
 iSBX Expansion Port, 4-2  
 iSBX MINTR0/MINTR1 Request Jumper (JP1), 2-3  
 iSBX Port, 2-2  
 iSBX Port (P4), 1-4  
 Isolated Outputs, 2-20  
 Isolation of Analog Outputs, 2-21

**J**

JP1, 2-3  
 JP10-JP15, 2-6  
 JP16-JP23, 2-7  
 JP24-JP26, 2-8  
 JP27, 2-12  
 JP28, 2-12  
 JP29, 2-13  
 JP2-JP7, 2-3, 2-4  
 JP30-JP33, 2-14  
 JP34, 2-15  
 JP35-JP42, 2-15, 2-16  
 JP43, 2-17  
 JP44, 2-17  
 JP45-JP48, 2-18  
 JP49-JP51, 2-20  
 JP8, 2-5  
 JP9, 2-5  
 Jumpers, 2-2  
   base address, 2-8  
   encoder sampling frequency, 2-19

**L**

LASEN\_N, 1-8  
 LASEN\_N Input, 1-4  
 Laser Output Polarity Jumpers (JP24-JP26), 2-8  
 Laser Outputs, 1-4  
 Laser Pulse Output Definition, 3-3  
 Laser Shutoff Input, 1-4  
 LED, 2-2  
 LED Problems, 5-2  
 LIB Files, 3-33  
 LOUT1 and LOUT2, 1-8, 2-8  
 LREADY, 1-8

**M**

Main Connector (P1), 4-3  
 Main Interface P1, 1-4  
 Memory Conflicts, 2-7  
 Microsoft Diagnostic Program (MSD.EXE), 2-7  
 MINTR0/MINTR1 Selection Jumper (JP29), 2-13  
 MSD.EXE Diagnostic Program, 2-7  
 Multimodule Boards, 1-4

Multiplier Board, 2-18

**N**

Non-destructive Testing Applications, 1-2

**O**

OP500 Cable, 1-7  
 Opto 22 I/O Bus, 2-2  
 Opto-22 Board, 1-7, 3-4, 4-2  
 Opto-22 Connector (P5) Pinouts, 4-6  
 Opto-22 I/O Port (P5), 1-3, 2-23  
 OUT0-OUT15 Outputs, 1-3, 1-7  
 OUT0-OUT3, 1-8  
 OUT0-OUT3 Outputs, 1-4  
 Output Definition of Laser Pulse, 3-3  
 Output Firing, Enabling and Disabling, 3-12  
 Output Signal Examples, 1-9  
 Output Toggle Mode, 3-23, A-3  
 Outputs, 1-4  
   "out\_map" argument, 3-6  
   AOUT1 and AOUT2, 1-4  
   configuration of DACs to provide position ramping, 3-30, A-3  
   control using PSOT command, 3-3  
   controlling states based on inputs, 3-4  
   isolated, 2-20  
   LOUT1, 1-4  
   LOUT2, 1-4  
   OUT0-OUT15, 1-3, 1-7  
   OUT0-OUT3, 1-4  
   PSOT Command, 3-27  
   READY, 1-4  
   setting based on inputs, 1-3  
   setting output voltages of DACs using PSOT, 3-28  
   setting states of, 3-5  
   setting using PSOT command, 3-27  
 Outputs OUT0-OUT15  
   setting states of, 3-5

**P**

P1 Connector, 2-2  
 P1 Main Connector Pinouts, 4-3  
 P1 Main Interface, 1-4  
 P2 Connector, 2-2  
 P3, 3-1  
 P3 Aerotech Bus, 1-3, 1-7  
 P3 Aerotech Bus Pinouts, 4-4  
 P3 Connector, 2-2  
 P4 Connector, 2-2  
 P4 Intel iSBX Bus Pinouts, 4-5  
 P5 Connector, 2-2  
 P5 Opto-22 Connector Pinouts, 4-6  
 P5 Opto-22 Port, 1-3  
 P6 and P7 Connections, 2-15

- P6 Connector, 2-2
- P6 Single-ended Encoder Interface Connector
  - Pinouts, 4-7, 4-8
- P7 Connector, 2-2
- P7 Differential Encoder Interface Connector Pinouts, 4-9
- Parameter 015, 3-36
- Parameter 016, 3-36
- Parameter Summary, B-1
- Parameters 015 and 016 of the UNIDEX 500, 2-7
- PC Board, 2-2
- PC Expansion Slot, 2-22
- PC ISA Bus, 1-3, 1-7
- PC Power Problems, 5-2
- PC-PSO Board, 2-2
- PC-PSO installation, 2-22
- PCPSO.FRM File, 3-36
- Pinouts, 4-3, 4-4, 4-6, 4-7, 4-8, 4-9
- Ports and Connector Types, 1-8
- Ports of the PSO-PC, 1-6
- Position Counter
  - clearing, 3-26
  - stopping, 3-26, A-3
- Position Counters
  - setting and locking, 3-14
- Position Ramping, 3-30, A-3
- Position Synchronized Output Firing Distance Command, 3-7
- Position Synchronized Output Pulse Configuration (PSOP), 3-20
- Position Synchronized Output Using Bit Mapping (PSOM), 3-19
- Position Synchronized Output with Real-time Control, 3-26
- Position Tracking
  - based on inputs, 3-4
  - unconditional, 3-4
- Precautions, 1-10
- Programming
  - Command Summary, A-1, A-2, A-3
  - commands, 3-3
  - PSO Commands, 1-2
  - role of the motion controller card, 1-2
- Programming Conventions, 3-2
- Programming Tools, 3-33
- PSO Command Bus (P3), 1-3
- PSO Command Prefix, 1-2
- PSO Commands, 4-4
  - case sensitivity, 3-1
  - routing, 1-9
- PSOC
  - using optional inputs and outputs, 1-3
- PSOC Command, 3-4
- PSOD Command, 3-7
- PSOF Command, 3-12
- PSOM Command, 3-19
- PSOP Command, 3-20

- PSO-PC Card, 1-7
- PSOR Command, 3-26
- PSOT Command, 3-27
- Pulse Configuration Command (PSOP), 3-20
- Pulse Gap Interval, 3-22
- Pulse Lead Time, 3-22
- Pulse Output
  - incremental distances, 3-8
- Pulse Ramp Up/Down Time, 3-22
- Pulse Trail Time, 3-22
- Pulse Train
  - components, 3-22
  - defining, 3-22
- Pulse Width Time, 3-22

## Q

- Quadrature Edge Separation for Encoders, 2-19
- Quadrature Signals, 2-18

## R

- RAM (Dual Ported), 2-3, 2-6
- Ratings, 4-2
- READY, 2-8
- Real Time Tracking Control, 3-3
- Real-time Control (PSOR), 3-26
- Reseating Loose Components, 2-1
- Resistor Networks RN1 and RN2, 2-15
- Revision History, R-1
- RN1 and RN2, 2-15
- Routing PSO Commands, 1-9

## S

- Safety Procedures, 1-10
- Sampling Process for Encoders, 2-18
- Sampling Rates for Encoders, 2-19
- Sending PSO Commands Using C Library Function, 3-35
- Signal Examples, 1-9
- SIN/COS Encoder Data Format, 2-14
- Single Pulse, Defining, 3-21
- Single-ended Encoder Channels, 4-2
- Single-ended Encoder Interface (P6), 1-3
- Single-ended Encoder Interface Connector (P6)
  - Pinouts, 4-7, 4-8
- Single-ended Encoder Port (P6), 1-7
- Single-ended Encoders, 2-15
- Specifications, 4-2
- SRDY\_N PC Bus Timing Enable/Disable Jumper (JP9), 2-5
- Stages, 1-7
- Startup Problems, 5-1, 5-2
- Static Charge Buildup, Removing, 2-1
- Static Discharge Precautions, 2-1
- Synchronization, 3-7

**T**

Technical Details, 4-1, 5-1  
Technical Support Questions, 5-1  
Test Points, 4-10  
Through Transmission Applications, 1-2  
Time-out Error, 3-36  
Time-out Error During Array Download, 3-34  
Toggle Mode for Outputs, 3-23, A-3  
Tracking  
    based on input conditions, 1-3  
Tracking Axes, Specifying, 3-3  
Tracking Control, Real Time, 3-3  
Troubleshooting, 4-10  
    address problems, 2-7  
    memory problems, 2-7

**U**

U500.JWP File, 3-36  
Ultrasonic Pulse/Echo Applications, 1-2  
UNIDEX 500, 1-7  
    parameter 016, 2-4

    parameters 015 and 016, 2-7  
    programming, 3-1  
Unpacking the PC-PSO, 2-1  
User-supplied Cables, 1-3  
    customized DB37, 1-4

**V**

Variable V0-V255, 3-19  
Variables V0-V255, 3-7  
    size in bytes, 3-19  
Vector Processing, 1-1  
Velocity Ramping, 3-28, A-3

**W**

Warnings, 1-10  
Warranty Policy, C-1  
Watchdog Circuit Enable/Disable Jumper (JP43), 2-17

**X**

X-ray Inspection Applications, 1-2

▽ ▽ ▽





## REVISION HISTORY

**In This Section:**

- Revisions ..... R-1

### Revisions

The following section provides the user with general information regarding the latest changes to this manual. Extensive changes, if made, may not be itemized – instead, the section or chapter will be listed with “extensive changes” in the corresponding General Information cell.

**Table R-1. Revisions**

Revision	Section(s) Affected	General Information
1.5	Chapter 3	V255 changed to V9999 (U500 variables)
	3.1.	U600 Programming Note
	3.4.1.2. & 3.4.1.3.	PSOD,1 and PSOD,2 text updated
	3.7.1.6	“8” changed to “500” in example
	4.2	Max temp specification added
	Appendix A	V255 changed to V9999 (U500 variables)
		U600 Programming Note

▽ ▽ ▽





## READER'S COMMENTS

### PSO-PC DSP-based Programmable Output Control Card Operation & Technical Manual P/N EDO 105, August, 2001

Please answer the questions below and add any suggestions for improving this document. Is the information:

	<u>Yes</u> _____	<u>No</u> _____
Adequate to the subject?	_____	_____
Well organized?	_____	_____
Clearly presented?	_____	_____
Well illustrated?	_____	_____
Would you like to see more illustrations?	_____	_____
Would you like to see more text?	_____	_____

How do you use this document in your job? Does it meet your needs?

What improvements, if any, would you like to see? Please be specific or cite examples.

---

---

---

---

---

---

---

Your name \_\_\_\_\_  
Your title \_\_\_\_\_  
Company name \_\_\_\_\_  
Address \_\_\_\_\_  
\_\_\_\_\_

Remove this page from the document and fax or mail your comments to the technical writing department of Aerotech.

AEROTECH, INC.  
Technical Writing Department  
101 Zeta Drive  
Pittsburgh, PA. 15238-2897 U.S.A.  
Fax number (412) 967-6870

