Digital Scanning Controller (DSC)

User Manual

COMPONENTS PRODUCTS GROUP
GMAX™ SYSTEMS
MULTI-AXIS BEAM HANDLING
39 Manning Road
Billerica, Massachusetts 01821

GSI PN 12P-DSC
Rev. D

TABLE OF CONTENTS

Warranty 1

Customer Support 2

INTRODUCTION

Scope 3

Scanning System Configuration 4
XY Scan Head 5
Beam Generation/Modulation 5
Digital Scanning Controller 5
Host Computer/Controller 5

DSC Setup 6

THE DIGITAL SCANNING CONTROLLER

System Specifications 9
Mechanical 9
Electrical 9
Environmental 9

Euro Card Chassis 9
Euro Power Supply 10
Scanner Interface Card 11
Digital Backplane 11

Euro Servo Control Bus 11
Mechanical Specifications 12
Pin Assignments 12

Euro Diagnostic Bus 13
Mechanical Specifications 13
Pin Assignments 13
Functional Descriptions 14

Euro Digital Driver 15

Digital Interface Controller 16

HOST-TO-DSC INTERFACE

GSI Bus 19
DX versus DSC 19
Mechanical Specification 20
Electrical Specification 21
Pin Assignments 21
Functional Definitions 21
Scanner Bus Cycle Timing 22

EDD Command Interface Bus 23
Mechanical Specification 23
Electrical Specification 23
Pin Assignments 24
Functional Definitions 24
ECI Bus Cycle Timing 25
Write 25
Read 26
EDD Address Assignments 27

DSC OPERATION

XY Head Motion 29

Types of Scanner Drive Signals 30

Vector Scan Generation 31

Optimizing Vector Operation 34

Debugging Scanning Software 34

APPENDIXES

APPENDIX A: TROUBLESHOOTING MATRIX 36

APPENDIX B: SYSTEM CONNECTORS 38

APPENDIX C: INTERFACE HARDWARE. 40

APPENDIX D: IMPLEMENTING THE ECI BUS INTERFACE 41

ADDENDUMS

ADDENDUM A: CONNECTING THE SINGLE AXIS DSC POWER SUPPLY. . . .49

1.0 Introduction

The growth of laser technology and its applications over the past three decades has produced a commensurate growth in the types and sophistication of beam delivery methods. As system requirements demanded greater performance, both the equipment manufacturers and laser users extended their knowledge, expertise and expectations. General Scanning has been and continues to be committed to the laser industry.

Originally, GSI emerged at the forefront of beam delivery technology with our basic product, the closed-loop, "position controlled" galvanometer. As technology has advanced, so too has General Scanning in its effort to provide the laser working field with fast, easy and affordable methods of directed beam energy. General Scanning has maintained a leadership position in beam delivery technology by aggressively pursuing all aspects of optical path requirements.

Beginning with the galvanometer, GSI has evolved a family of complementary components: galvanometers (both linear & resonant), mirrors, XY head assemblies, dynamic focusing lens translators, and complete optical rails. To complement the optical and electro-mechanical components, a series of servo control products have been developed: servo control amplifiers (analog & digital), digital interface units (for analog amplifiers), digital correction electronics (for 2/3-axis, post-objective scanning), raster and vector control units, and integrated digital controllers. In addition, GSI creates and manufactures complete scanning systems to meet custom design requirements.

The Digital Scanning Controller is one member of the most advanced servo control system ever offered by General Scanning. As such it embodies years of experience and product knowledge acquired while providing laser scanning solutions to industry. We feel that it offers distinct advantages in performance, reliability and cost-effectiveness.

1.1 Scope

This manual is intended to provide users with the information necessary to correctly set up and operate the Digital Scanning Controller (DSC) and XY Head. The cabling and interconnection of the scan head and DSC is illustrated, along with a brief description of each element. A complete interface specification is presented for both the older GSI Scanner Bus and the newer EDD Command Interface (ECI) Bus. In order to facilitate the integration of the DSC into a user's system, guidelines and suggestions for implementing the ECI have been included.

The Digital Scanning Controller Manual has been expressly written to be a Users Manual and Reference Guide for the normal operation and use of the DSC. As the XY Head-DSC pair is provided as a beam delivery *module* or sub-system, circuit-level details and operational theory have been omitted from this document. Instead, broader descriptions of the DSC components have been included to identify each component's function.

1.2 Scanning System Configuration

When discussing a scanning system, certain features are common to all systems, others are common to most, and some are unique to specific systems. Although all practical systems must have a target (for image acquisition or deposition), the details of how the target interacts with the other system components is very application-dependent. To provide a useful manual to the greatest number of users, the focus of this section is restricted to only the common system elements. As Figure 1-1 illustrates, a typical laser system includes the following components:

- XY Scan Head
- Beam Generation/Modulation
- Digital Scanning Controller
- Host Computer/Controller

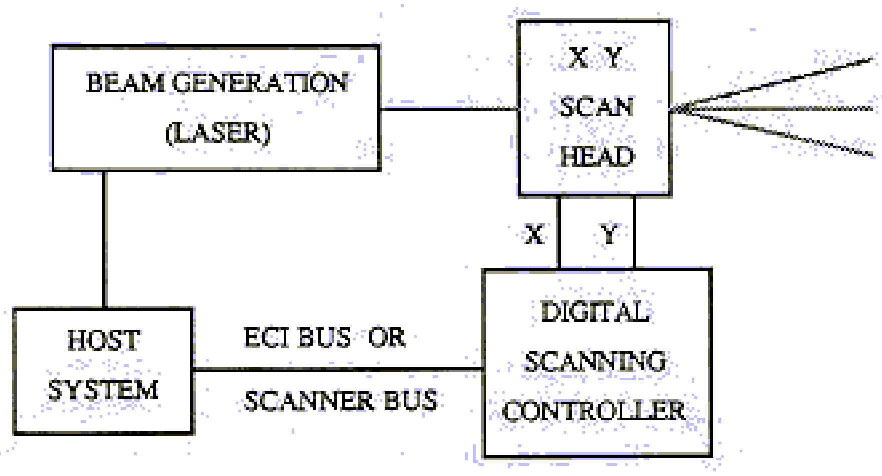


Figure 1-1: Laser System Block Diagram

1.2.1 XY Scan Head

The XY Scan Head contains two closed-loop position controlled galvanometers, two mirrors and the scanner mount. These units are pre-assembled and pre-tested with the DSC. The scan head provides two-axis beam deflection within the scan field. Each head includes mechanical limit stops at the full scan angle to protect the XY mirrors from unintended collisions. General Scanning has a complete line of XY Scan Heads to address every beam positioning need.

Scan Head	Pupil Size
XY0507	5 mm
XY1013	10 mm
XY2026	20 mm
XY3037	30 mm
XY5067	50 mm
XY2100	9.5 mm

1.2.2 Beam Generation/Modulation

This includes the user-provided laser source, attenuation, safety interlocks and lenses, if any.

1.2.3 Digital Scanning Controller

The DSC contains all of the circuitry necessary to control two closed-loop galvanometers. The DSC also contains a power supply and digital interface module. The DSC provides two-axis servo control for systems using digital scanning techniques. Host commands are translated from the active digital bus into appropriate scanner positions, with high speed and accuracy.

1.2.4 Host Computer/Controller

The Host Computer or Controller initiates all scanner motion by sending commands to the DSC. The Host can be connected to the DSC by either of two digital interfaces. Although different in many particulars, each bus is designed for parallel transmission of 16-bits of data (scanner position). The Host may also control the beam attenuation. Implementation of one GSI digital bus within the user-supplied "Host Computer/Controller" is required before the DSC can be used, as there are no analog signal inputs.

1.3 DSC Setup

NOTE: Please read the XY Scan Head User Manual and the remainder of this manual before powering up the DSC. The instructions rely on familiarity with the layout and parts of the XY Head and DSC, as well as specialized interface and operational knowledge.

[1] Unpack the XY Scan Head and the DSC & cables and save the boxes for return to the factory, if necessary.

[2] Check that the voltage selection jumper has been correctly installed (110 VAC) or removed (220 VAC) for your line voltage. (Ref. Figure 2-5.)

[3] Connect one end of an extension cable to the X scanner cable and the other end to SIC connector J4, behind the X EDD amplifier. (Ref. to Figure 2-3.) Secure the scanner end of the cable by closing the locking slide. Secure the EDD end of the cable by screwing the captive screws firmly, without over-tightening, into the locking standoffs. Connect the other extension cable between the Y scanner and Y EDD's SIC. Refer to the EDD nameplate to identify the X and Y channels.

Install both cables prior to energizing the DSC. **If the scanner cables are connected or disconnected while the DSC is powered up, catastrophic damage may occur.**

[4] Connect the Host Computer to either the GSI Scanner Bus connector or the ECI Bus connector. (Ref. to Figures 2-2 & 2-3.) Secure the cable by engaging and tightening the connector's screws in the locking standoffs. Ensure that the cable is connected to an interface card which has been properly designed for that bus. Install the cable prior to energizing the DSC. **If the host system is connected or disconnected while the DSC is powered up, catastrophic damage may occur.**

[5] Read the rest of this manual. Most importantly, read the sections covering the applicable digital interface and the DSC operation. **Improper operation of the DSC can result in damage to the XY Scan Head, DSC or both.**

[6] Ensure that the Host System is powered up and operating correctly.

[7] Connect the AC power cable to the DSC input socket. (Ref. Figure 2-3.) Power up the DSC. The ON/OFF switch is located above the AC input socket.

2.0 The Digital Scanning Controller

The Digital Scanning Controller (DSC) is a versatile scanner servo control engine which is packaged in a closed electronics chassis. When configured for the DSC, the chassis is equipped with a power supply, two servo controller modules and a digital interface module. The DSC is a self-contained unit which accepts Host scanning commands and translates them into servo signals and, by extension, scanner positions.

The Digital Scanning Controller is a member of General Scanning's advanced line of servo control products. The DSC is designed to support applications requiring only two axes of scanning control, without built-in geometric correction or position interpolation. Other family members include the Linear Scanning Controller which accepts analog input signals and the DE series of two-axis and three-axis scanner control electronics. The DE 3000, for example, includes three EDD modules, the common power supply module, and an internal 68000 microprocessor to perform high-level command interpretation, geometric correction, and scanner position interpolation. Each of these products is designed to provide the best performance to the user.

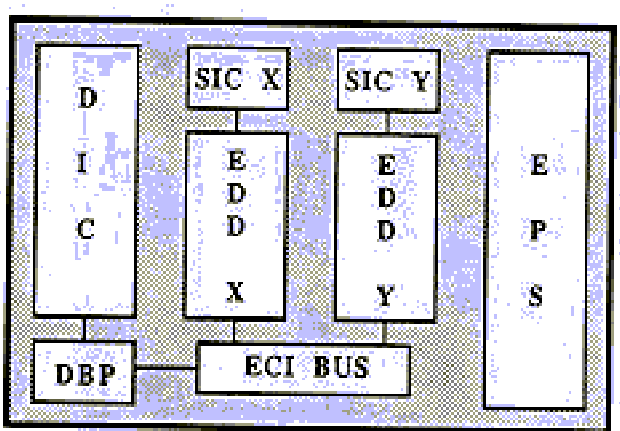


Figure 2-1: Block Diagram of DSC

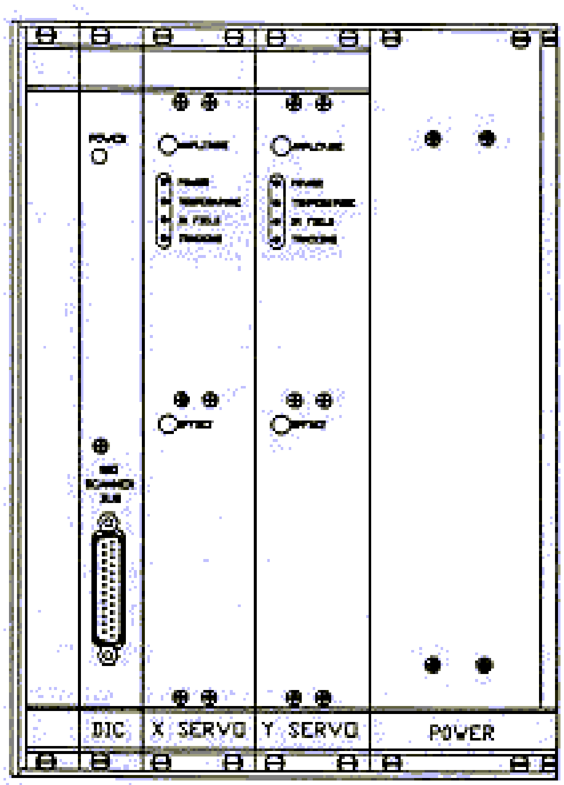


Figure 2-2: DSC Front View

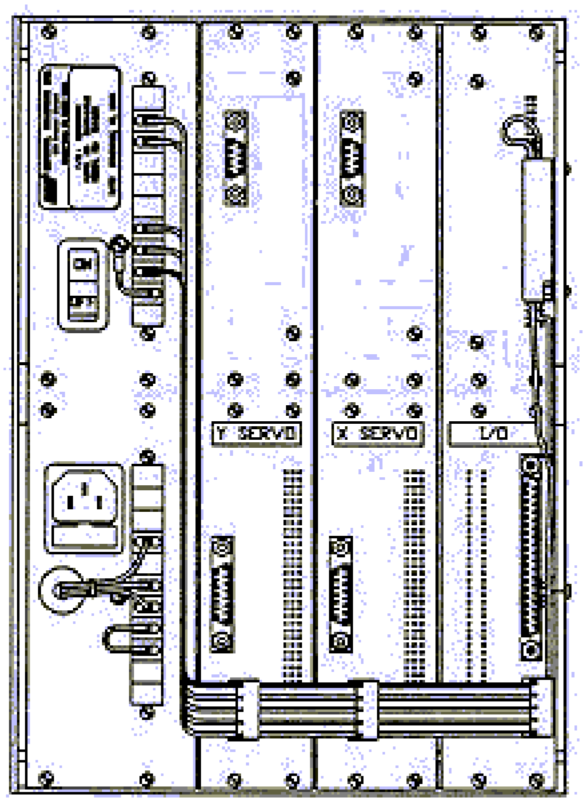


Figure 2-3: DSC Rear View

2.1 System Specifications

2.1.1 Mechanical

Dimensions:	10.5H x 7.75W x 8.0D (in) / 266H x 197W x 203D (mm)
Weight:	11 lbs, 1 oz / 4.57 kg

2.1.2 Electrical

Input Voltage:	110/220 VAC - User Selectable
Line Frequency:	50/60 Hz - No Modification Required
Input Current (MAX):	4 A @ 110 VAC / 2 A @ 220 VAC

2.1.3 Environmental

Operating Temperature:	15° C to 32° C
Storage Temperature:	-20° C to 80° C
Relative Humidity:	10% to 90% Non-Condensing

2.2 Euro Card Chassis

The Euro Card Chassis (ECC) consists of a VME-compatible card frame for structural support, a power supply module and a segmented backplane (see Figure 2-4). The leftmost slot (see Figure 2-2) accommodates the digital interface module, the center slots contain the X and Y servo control amplifiers, and the rightmost section houses the unit's power supply.

The majority of the ECC components are used in all of General Scanning's Euro-based scanning control products. This ensures compatibility between family members and provides user with an upward performance ladder, as their requirements change.

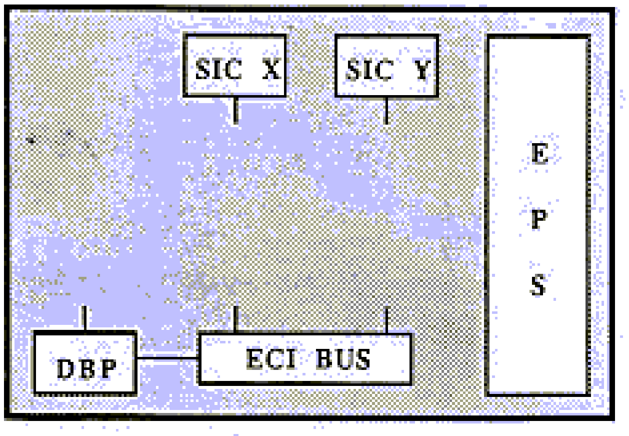


Figure 2-4: Block Diagram of ECC

2.2.1 Euro Power Supply

The Euro Power Supply (EPS) is a switching power supply which accepts AC line voltage and generates ±18 VDC and +5 VDC regulated outputs. It is the standard power supply of General Scanning's two and three axis scanning control systems. The EPS can be configured for operation with 110 or 220 VAC line voltage and 50 or 60 Hz line frequency.

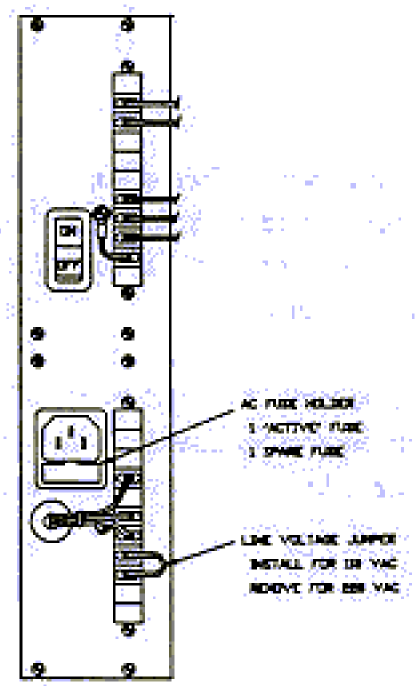


Figure 2-5: EPS Rear View

The standard shipping configuration is 110 VAC/60 Hz. To change the AC line selection, install or remove the voltage jumper shown in Figure 2-5. A replacement fuse is located within the AC input connector's fuse holder, on the rear panel (see also Figure 2-5). This fuse is rated for 5 amperes.

2.2.2 Scanner Interface Card

The EDD servo control amplifiers are linked to their galvanometers through the ECC's segmented backplane. Behind each EDD is a Scanner Interface Card (SIC) which connects that amplifier with DC power and the digital control bus. In addition, the SIC has two output connectors: one for scanner control and one for diagnostic information. The user connects the scanner cable (usually a scanner extension cable) to the scanner connector (J4) with the power OFF. The diagnostic connector (J3) is primarily used for EDD setup and tuning. It can also be used for software debugging by identifying significant problems, without having a scanner connected to the EDD.

2.2.3 Digital BackPlane

Similar to the EDDs, the Digital Interface Controller (DIC) is linked to the remainder of the DSC through the Digital BackPlane (DBP). With the DBP, the DIC provides a signal path to the EDD modules from the Host system. Inside the DSC, the EDD and DIC cards bus the ECI digital signals. The DBP to connects this "private" pathway to the external world. Although it is a vital component of the DSC, there is no active circuitry on the DBP PC card.

On the visible (outer) side, the DBP has the ECI Bus's 37 pin D connector. This connector allows interconnection of external equipment (Host) to the full EDD Command Interface Bus. Since the DBP is part of the segmented backplane in the ECC, this connector is only accessible from the rear of the chassis. On the inner, "hidden" side, the DBP has a mating connector for the DIC PC board, analogous to the SIC's mating with the EDD card. Thus, the DBP links the physical ECI bus, the 37 pin D connector, and the DIC-converted-to-ECI Scanner Bus.

2.3 Euro Servo Control Bus

The EDD servo control amplifiers are connected to the galvanometers through the segmented backplane of the ECC. Behind each EDD is a Scanner Interface Card (SIC) which connects each amplifier with DC power and the digital control bus. In addition, the SIC has two output connectors: one for diagnostic information and one for scanner control.

The scanner is operated through a 15 pin D connector located near the bottom of the SIC. (Please refer to the Figure 2-3.) The scanner connector is labelled J4 and contains the following signal groups:

- High & low drive signals to the scanner coils
- Thermal drive and control signals required by the heater blanket and thermistor control circuit to regulate the scanner temperature.
- Control signals used by the position detection circuitry and Automatic Gain Control (AGC) circuit.

2.3.1 Mechanical Specifications

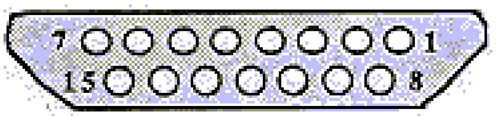


Figure 2-6: Scanner D Connector (J4)

Manufacturer's Specification:

PLUG	On scanner/extension cable:	AMP p/n 205206-1 or eq.
RECEPTACLE	On chassis (SIC) rear panel:	AMP p/n 205205-1 or eq.

2.3.2 Pin Assignments

15-pin D Connector	Signal Name	15-pin D Connector	Signal Name
1 & 9	Drive Current High	7	Heater Current High
2 & 10	Drive Current Low	8	AGC Output
3	Position In (+)	11	Position In (-)
4	Signal GND	12	AGC Sense
5	Heater Current Low	13	AGC GND
6 & 14	(Reserved)	15	Thermistor Input

2.4 Euro Diagnostic Bus

As stated in section 2.3, each EDD mates with a Scanner Interface Card (SIC) which connects the amplifier with its inputs (DC power & the ECI bus) and outputs (diagnostic & scanner control signals).

The diagnostic signals are provided through a 9 pin D connector located near the top of the SIC. (Please refer to the Figure 2-3.) The scanner connector is labelled J3 and contains the following signal groups:

- Drive signals to the scanner coils and position sensor.
- Scanner position-related signals: velocity, position. position error & DAC output.
- Optional analog input (requires a hardware modification.)

2.4.1 Mechanical Specifications

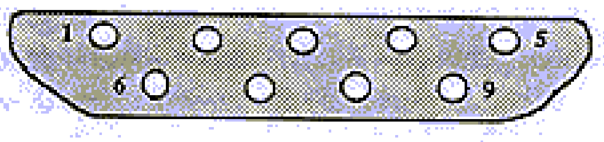


Figure 2-7: Diagnostic D Connector (J3)

Manufacturer's Specification:

PLUG	On mating cable:	3M p/n 3730-1000 or eq.
RECEPTACLE	On chassis (SIC) rear panel:	AMLAN p/n CDF9P or eq.

2.4.2 Pin Assignments

9-pin D Connector	Signal Name	9-pin D Connector	Signal Name
1	Position Error (10x)	5	Scanner Current
2	Position Out	6	Velocity Out
3	DAC Out	7	External In
4 & 9	Signal GND	8	AGC Voltage

2.4.3 Functional Definitions

Position Error: The voltage measured is equal to ten times (10x) the position error signal. The position error signal is defined to be the difference between the position input signal (desired position) and the position output signal (actual position), from the transducer. The position error signal is used to tune the scanner/amplifier pair.

NOTE: The signal is often several millivolts offset from 0V, due to normal op amp offset voltages.

Position Output: The voltage measured is proportional to the angle of the mirror. It is generated by the position transducer within the galvanometer and the position amplification circuit on the EDD.

DAC Output: The voltage measured is proportional to the digital position values generated by the Host system. When used, the DAC output is also the position input signal to the servo amplifier circuits.

Scanner Current: The voltage measured is one-half the galvanometer drive signal current (i.e. 2 amps produces 1 volt). The voltage is generated by sampling the drive current across a 0.5 Ω resistor. The drive signal frequency is rolled off with a 3 KΩ series resistor and 0.01 μF capacitor (~5.3 kHz).

Velocity Output: The voltage measured is proportional to the speed of mirror motion (or, slew rate of mirror deflection.) The velocity signal is generated by differentiating the position output signal.

External Input: This pin can be used to directly supply the EDD card with an analog input signal. (Please contact factory.)

NOTE: To use this input pin, the DAC must be switched off.

AGC Voltage: The voltage measured drives the position sensor circuitry in the scanner. The voltage is generated by the Automatic Gain Correction circuit on the EDD card, and connected via a 3 KΩ series resistor. The AGC supplies a stable voltage to the position sensor, for greater position accuracy.

Signal Ground: Ground is provided for circuit reference and signal return for all diagnostic connector signals.

2.5 Euro Digital Driver

The Euro Digital Driver (EDD) contains an analog closed-loop servomotor control amplifier which generates the current required to drive a scanner. The EDD also contains digital interface circuitry and a Digital-to-Analog Converter (DAC) which changes the digital position data into analog position voltage for the servo circuits. The desired position voltage is compared with the actual position signal being returned by the galvanometer. The servo amplifier produces drive signals to minimize the difference between the actual and desired scanner positions.

The four status indicators, located on the front of each EDD module, give visual confirmation that the scanner is operating correctly. (See Figure 2-8.)

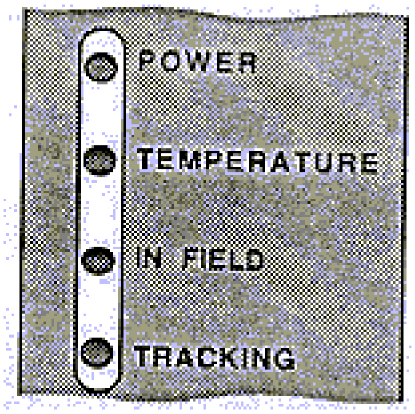


Figure 2-8: EDD Front Panel Indicators

POWER:	When lit, POWER indicates that EDD is receiving +18 VDC and +5 VDC. The LED must be ON for the EDD to function.
TEMPERATURE:	When lit, TEMPERATURE indicates that the scanner temperature has stabilized. When the LED is OFF, the scanner is either warming up or is overheating. It is highly recommended that the user wait until the scanner temperature has stabilized before scanning.
IN FIELD:	When lit, IN FIELD indicates that the scanner is within its maximum rated excursion (angle). When the LED is OFF, the over-position circuitry has detected that the scanner is beyond its maximum rated excursion.

TRACKING: When lit, TRACKING indicates that the position error (the difference between the desired and the actual scanner positions) is within the preset limit. When the LED is OFF, the position error signal is greater than the preset limit. This is not necessarily a fault condition. The TRACKING LED may blink for many applications. The amplitude of the position error signal is proportional to the scanning speed of the galvanometer, the slew rate of the mirrors, and the amplitude of the position signal.

Name	State	Description
Power	LED ON	EDD module is receiving power
	LED OFF	EDD module is not receiving power
Temperature	LED ON	Scanner temperature is stabilized.
	LED OFF	Scanner is warming up or overheating.
In Field	LED ON	Scanner is within maximum excursion.
	LED OFF	Scanner is beyond maximum excursion.
Tracking	LED ON	Scanner is following the position input signal (DAC output) within preset error threshold.
	LED OFF	Scanner has exceeded the preset position error threshold.

NOTE: When the scanners are at rest and warmed up, all status indicators should be ON. The G100 and G300 series scanners warm up in approximately 5 and 30 minutes, respectively. Ambient temperature influences the warm up time.

Table 1: Status Indicator Summary

2.6 Digital Interface Controller

The Digital Interface Controller (DIC) is an adaptor card which permits the host system to connect to the EDD modules using either the GSI Scanner Bus or the EDD Command Interface (ECI) Bus. This permits users to benefit from the features of the newer interface, while providing complete compatibility with existing applications and equipment.

The DIC provides a complete implementation of both digital interfaces and it contains all of the circuitry required to convert signals from the Scanner Bus to the ECI Bus. The DIC plugs into a special member of the ECC's segmented backplane, the DBP (see also section 2.2.3). The Scanner Bus connector is a 25 pin D, located on the front of the DIC module. The ECI connector is a 37 pin D, located at the back of the DSC chassis and mounted on the DBP.

General Scanning, recommends that all new product designs implement the ECI Bus. The Scanner Bus has been included in the DSC for applications which require it, rather than as an equivalent. A clear example of this is using the DSC with the DG series of digital scanner control electronics. The Scanner Bus is not supported on other new GSI products and may not be included in future versions of the DSC.

A user may not physically connect to both the Scanner Bus and the ECI Bus, simultaneously. The design of the DIC permits signals from only one digital interface to reach the amplifiers, at any time. The active interface is selected by a set of five shorting jumpers. The jumpers, labelled W1 through W5, are located near the top edge of the DIC card (see Figure 2-9).

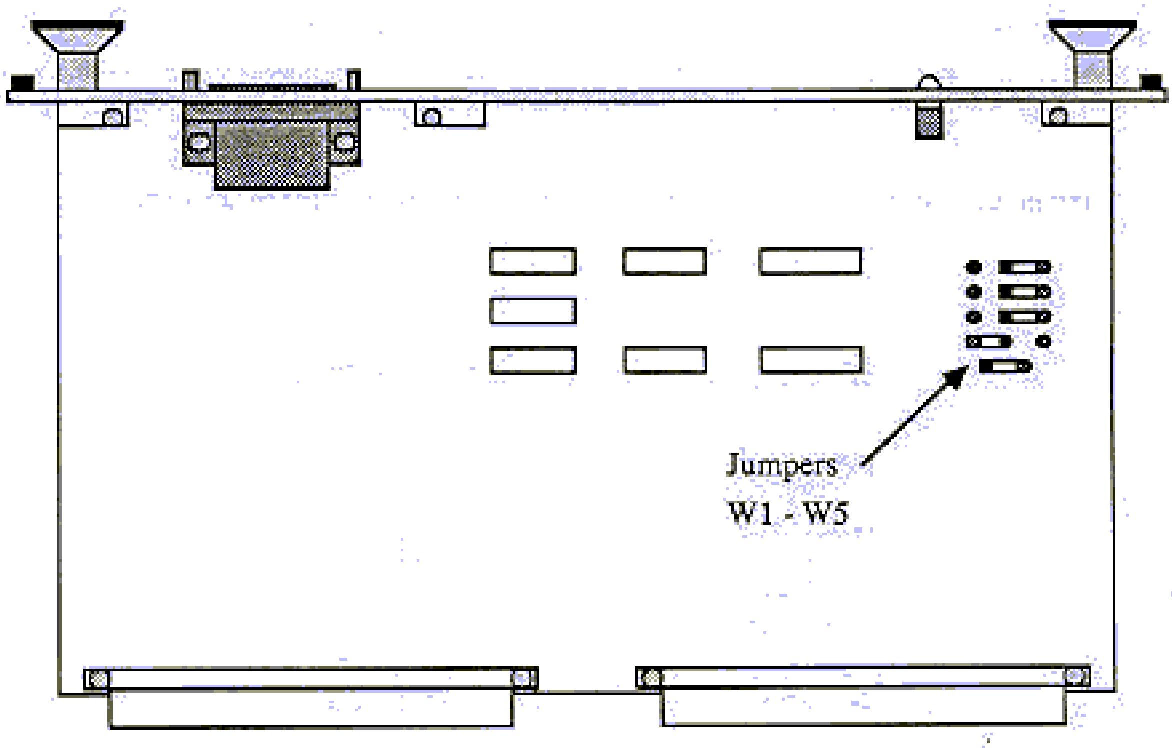


Figure 2-9: DIC Layout Reference Drawing

The DSC comes pre-configured for operation with either the Scanner Bus or the ECI Bus. The user may re-configure the DSC to use the other digital interface. This requires removal of the DIC card to change the jumpers. The modification can be made quickly, following a few simple steps.

> NOTE: **The DSC and its modules are static-sensitive.** Please perform all modifications at a static-controlled station, observing ESD-preventive precautions.

[1] Switch OFF the DSC. The ON/OFF switch is located above the AC input socket.

[2] Loosen the captive locking screws (above and below the DIC finger pull grips) until they do not engage the card frame.

[3] Using both hands, carefully pull the Digital Interface Controller out of the DSC. If the DIC moves forward a small distance and then stops, at least one of the screws is not completely freed. Re-loosen the screws and remove the DIC.

[4] Following Figure 2-10, move the shorting jumpers W1 through W4 onto the correct pairs of pins. Also, remove or install a shorting jumper across W5, as illustrated.

[5] Carefully slide the DIC card into the chassis. Ensure that the edges of the PC board are seated in the card guides. When the connectors engage, continue to push the DIC module (using the grips) firmly, until the front panel is flush with the EDD's panel.

[6] Tighten the locking screws until completely seated. Be careful to not over-tighten.

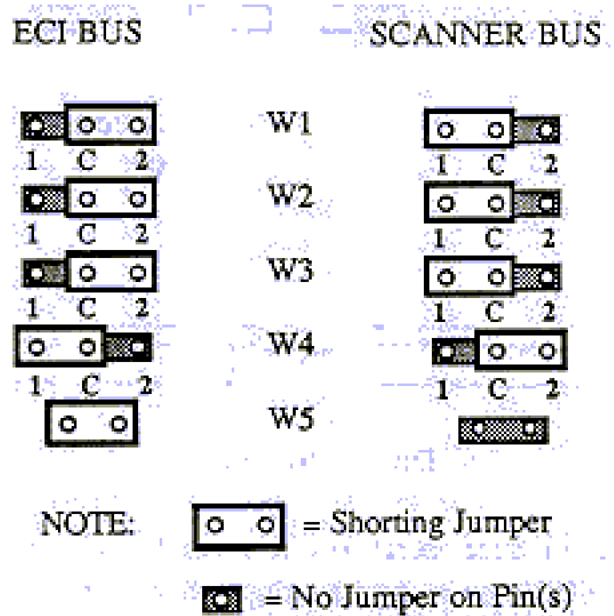


Figure 2-10: Jumper Settings for Digital Bus Selection

3.0 HOST-TO-DSC INTERFACE

The host computer initiates scanner motion by sending commands (positions) to the DSC. The DSC supports two physical interfaces: (1) the GSI (General Scanning Interface) Bus; (2) the ECI (EDD Card Interface) Bus.

Although different in their control mechanisms, each bus is designed for 16-bit, parallel transmission of data (scanner position). Users are free to choose either interface depending on their needs. However, all new customer designs should incorporate the ECI Bus, as the GSI bus will not be supported in future hardware releases.

3.1 GSI BUS

The GSI Bus consists of sixteen data lines (D15-0), three strobes (STB X*, STB Y*, STB Z*) and signal grounds. Data is transferred from the host to the DSC by first applying the data to the data bus and then pulsing (Active Low) the appropriate strobe line (for X, Y, or Z).

For exact timing specifications, refer to Figure 3-2 on page 22. Note that data is latched into the DSC on the rising edge of strobe. As long as data has been stable for 100 nS or more prior to the rising edge of strobe, the cycle will be successful.

3.1.1 DX VERSUS DSC

If you opt to use the DSC to replace a DX series servo control amplifier, you may have to make minor alterations to the host hardware. If the host system strictly adheres to the Scanner Bus timing requirements, as presented in section 3.1.6, then no modifications should be necessary.

Most problems will involve how the control signals are configured and, unless these lines are hard-wired, the changes should be easy to identify and implement.

In an application where DX series controllers were used previously, each scanner axis was connected to a single servo amplifier PC board (DA160). These boards were designed to directly support the GSI Scanner Bus, as they include a decoding section where the amplifier is configured to respond to only one of the three strobe signals X, Y or Z.

This completely isolates that PC board from any activity on the other two strobe lines. If more than one strobe signal were activated, each axis would independently respond to its own strobe signal.

A DSC contains the newer EDD servo amplifier PC boards. These boards are designed to directly support the ECI Bus, as they include a decoding section for ECI Bus signals, but not for Scanner Bus signals.

In the DSC, the Scanner Bus is decoded on the DIC card. The DIC synthesizes the correct ECI address, direction and strobe signals for the EDD servo amplifiers from the Scanner Bus input signals.

Because all Scanner Bus signals are channelled through only one set of decoding circuits, the DIC cannot process two strobes simultaneously. If a second strobe is activated while the first strobe is still active, the ECI signals will be invalid.

In a DX 2000 system, the Z strobe signal does not affect the X or Y scanners. If the host system design held the Z strobe in the active state (a logic 0 or LOW), this might go unnoticed.

It is possible for the designers to overlook this error, since it would not produce any errors. Because of the independence of each axis, the DX is very forgiving of this type of timing violation. But if a DSC is used to replace the DX, the decoding circuitry on the DSC never generates the correct sequence of ECI Bus signals and neither the X or Y scanner rotates. To correct this problem, the Z strobe must be set to the inactive state (a logic 1 or HIGH). This is likely to be the most common problem when a DX is replaced with a DSC.

3.1.2 MECHANICAL SPECIFICATION

The digital connector is a 25-pin D which fastens to the cable in the same manner as the scanner cables do (See Figure 3-1 below). The system has been designed to work reliably with unshielded ribbon cable (cable length recommended to be less than 6 feet) to minimize costs.

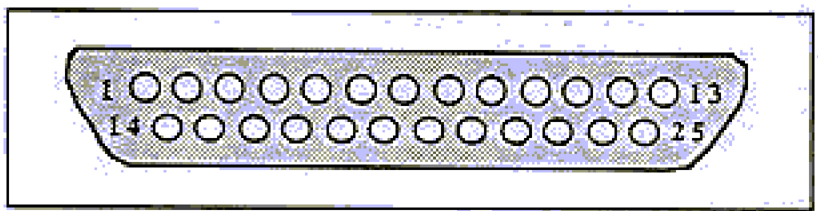


Figure 3-1: Scanner Bus Connector

Manufacturer's Specification:

PLUG On Chassis (DIC) front panel: AMP p/n 206771-1 or eq.
RECEPTACLE On ribbon interconnect cable: AMP p/n 206770-1 or eq.

3.1.3 ELECTRICAL SPECIFICATION

All signals are TTL and Low-Power Schottky (LS) TTL compatible.

Inputs:

Logic 0: Guaranteed if level is less than 0.8 V.
Logic 1: Guaranteed if level is greater than 2.0 V.
Fanout: Each signal line should be capable of driving 3 LS TTL loads.

3.1.4 PIN ASSIGNMENTS

25-pin D Connector	Signal Name
1	GND
2	STB X-
3	D14
4	D12
5	D10
6	D8
7	GND
8	STB Y-
9	D6
10	D4
11	D2
12	D0 (LSB)
13	STB Z- (unused)
14	GND
15	N.C.
16	D15 (MSB)
17	D13
18	D11
19	D9
20	GND
21	N.C.
22	D7
23	D5
24	D3
25	D1

3.1.5 FUNCTIONAL DEFINITIONS

D0-D16 16 data lines (uni-directional)

STB X- Edge-sensitive control line, which gates bus activity to X EDD

STB Y- Edge-sensitive control line, which gates bus activity to Y EDD

GND System ground/Signal Return

3.1.6 Scanner Bus Cycle Timing

Description	Symbol	Min.	Typ.	Max.	Units
Data Setup	TDVSH	0	-	-	nS
Strobe-	TSLSH	100	-	-	nS
Data Hold	TSHDX	500	-	-	nS
DAC cycle	TSLCC	-	8	-	µS

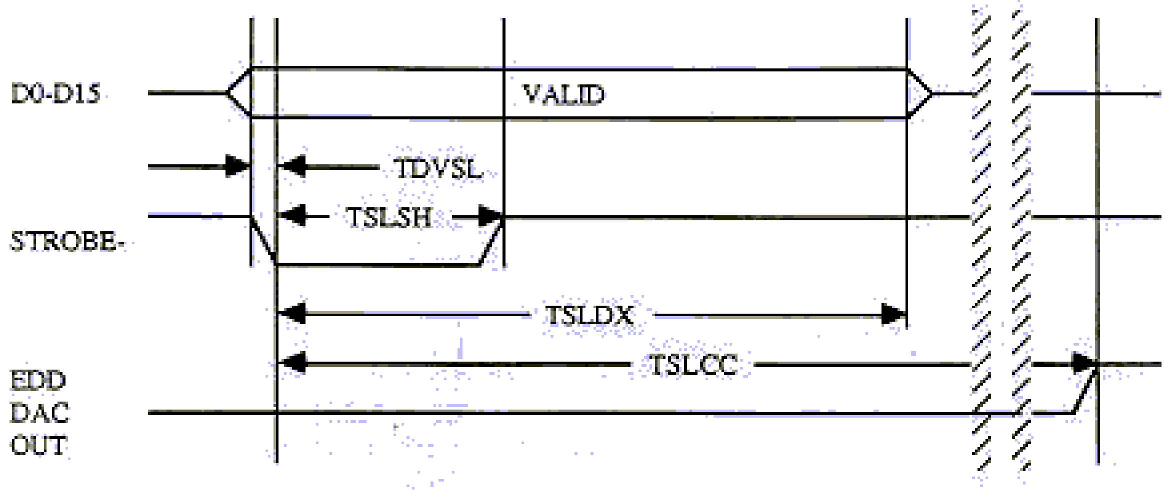


Figure 3-2: Individual Write Cycle

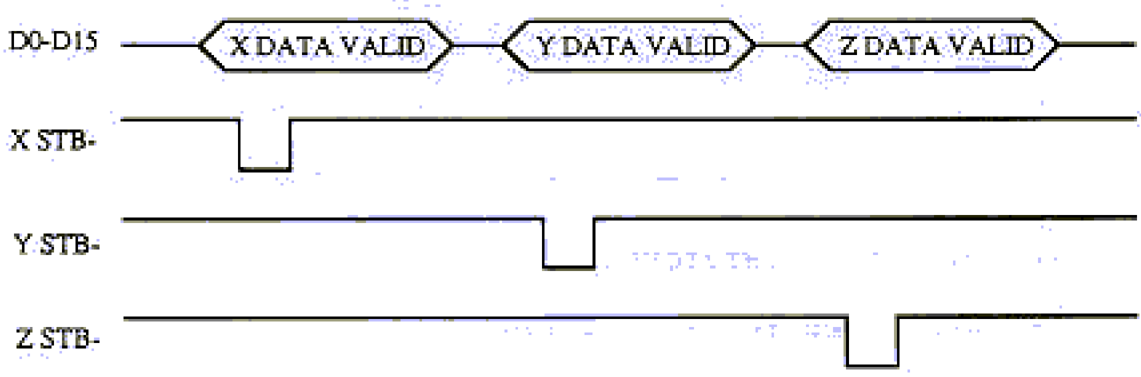


Figure 3-3: Typical 3-Axis Input Sequence

3.2 EDD COMMAND INTERFACE BUS

The EDD Command Interface (ECI) Bus was developed for General Scanning's third generation of integrated, digitally-controlled scanning systems. The ECI is incorporated in the DE Series of scanner control electronics, the Linear Scanning System (LSS) and the Digital Scanning Controller (DSC). The DSC provides external access to the ECI, whereas the other products incorporate it only internally.

The ECI offers greater flexibility and performance than the Scanner Bus, with little increase in integration or implementation complexity. The ECI is bi-directional, so status information on the scanners is available from the servo amplifiers by using an appropriate read command.

The ECI can support many devices on one cable, permitting future product line development without forcing interface change. It can be driven with programmed Input/Output from a variety of readily available parallel I/O PC-Interface cards. The minimum number of active I/O lines required to implement the ECI is 23.

3.2.1 MECHANICAL SPECIFICATION

The ECI connector is a 37-pin D which fastens to the cable in the same manner as the scanner cables do (See Figure 3-4). The bus is designed to work with unshielded ribbon.

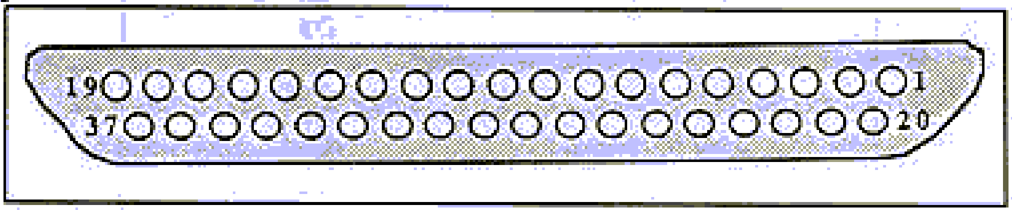


Figure 3-4: ECI Bus D Connector

Manufacturer's Specification

PLUG On Chassis (DIC) rear panel: AMP p/n 206650-1 or eq.
RECEPTACLE On ribbon interconnect cable: AMP p/n 206655-1 or eq.

3.2.2 ELECTRICAL SPECIFICATION

All signals are TTL and Low-Power Schottky (LS) TTL compatible.

Inputs:

Logic 0: Guaranteed if level is less than 0.8 V.
Logic 1: Guaranteed if level is greater than 2.0 V.
Fanout: Each signal line should be capable of driving 3 LS TTL loads.

Outputs:

Levels: Logic 0 and 1 will meet voltages specified under "Inputs" above.
Fanout: Each output can drive at least 7 LS TTL loads.

3.2.3 PIN ASSIGNMENTS

37-pin D Connector	Signal Name
1	D0 (LSB)
2	GND
3	D3
4	D4
5	GND
6	D7
7	D8
8	GND
9	D11
10	D12
11	GND
12	D15 (MSB)
13	A4
14	A2
15	RD/WR-
16	GND
17	GND
18	FS1 (UNUSED)
19	FS3 (UNUSED)
20	D1
21	D2
22	GND
23	D5
24	D6
25	GND
26	D9
27	D10
28	GND
29	D13
30	D14
-	-
32	A3
33	A1
34	RESET-
35	STB-
36	FS0 (UNUSED)
37	FS2 (UNUSED)

3.2.4 FUNCTIONAL DEFINITIONS

RD / WR- Level-sensitive control line to differentiate a READ bus transaction from a WRITE bus transaction.

D0-D15 16 data lines (bi-directional), D15=MSB & D0 = LSB.

A1-A4 4 address lines (unidirectional), A4=MSB & A1 = LSB.

STROBE- Edge-sensitive control line which gates all bus activity, READ or WRITE.

GND System ground/Signal Return.

FS0-FS3 4 Function Selector Lines (reserved for future implementation).

3.2.5 ECI BUS CYCLE TIMING

3.2.5.1 WRITE CYCLE

An ECI Bus write cycle is performed by providing the necessary amplifier address signals A1-A4 (see Table 2, page 27), setting the RD/WR* line low, applying valid data to data pins D0-D15, and finally bringing STB* low and then high. Figure 3-5 below shows the timing required. The most common method to accomplish this operation is to provide a generic parallel interface card and program the necessary control operations.

When using software to control this interface, the proper sequence is to (1) apply data, address and RD/WR* signals to the interface; (2) wait the necessary setup time (20nS); (3) pulse the STB* line low. The low level of STB* must be at least 100nS wide. Data will actually be latched on the rising edge of STB*. Data must remain stable for a minimum of 100nS wide after the rising edge of STB*.

Description	Symbol	Min.	Typ.	Max.	Units
Control Setup	TCVSL	10	-	-	nS
Data Setup	TDVSH	20	-	-	nS
Strobe-	TSLSH	100	-	-	nS
Control Hold	TSHCX	20	-	-	nS
Data Hold	TSHDX	100	-	-	nS
DAC cycle	TSHCC	-	8	-	µS

NOTE: Address Setup & Hold times are identical to those of Control lines.

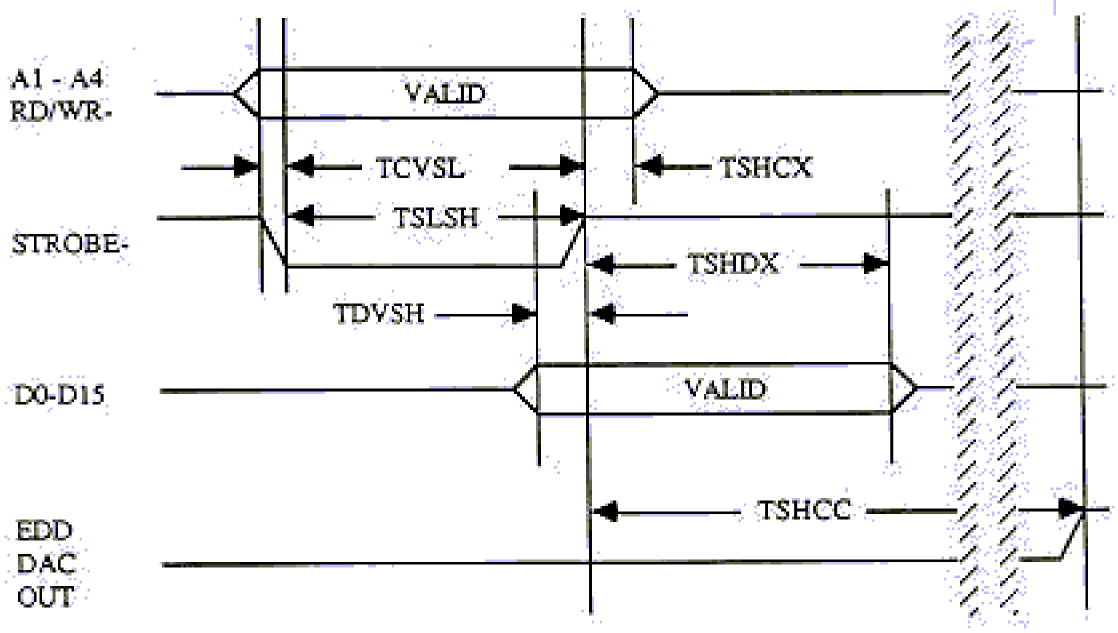


Figure 3-5: ECI Bus Write Cycle

3.2.5.2 READ CYCLE

An ECI Bus read cycle is performed by providing the necessary amplifier address on signals A1-A4 (See Table 2 next page), setting the RD/WR* line High, then after a short delay (10nS), lowering STB*. The timing required is shown in Figure 3-6. The status information will be available in 100nS (max). The status bit encoding is shown below.

```
                      Status Word Error Bit Definitions

 X Power
  |  X Temperature
  |   |  X In Field
  |   |   |  X Tracking
  |   |   |   |  Y Power
  |   |   |   |   |  Y Temperature
  |   |   |   |   |   |  Y In Field
  |   |   |   |   |   |   |  Y Tracking
  |   |   |   |   |   |   |   |  Z Power
  |   |   |   |   |   |   |   |   |  Z Temperature
  |   |   |   |   |   |   |   |   |   |  Z In Field
  |   |   |   |   |   |   |   |   |   |   |  Z Tracking
  |   |   |   |   |   |   |   |   |   |   |   |
  v   v   v   v   v   v   v   v   v   v   v   v
-----------------------------------------------------------------
|   |   |   |   |   |   |   |   |   |   |   |   | X | X | X | X |
-----------------------------------------------------------------
MSB                                                           LSB
(D15)                                                        (D0)

     If bit is set - Error is present
     X = Don't Care (Not Used)

     For Description of Errors refer to Section 2.5
```

Description	Symbol	Min.	Typ.	Max.	Units
Control Setup	TCVSL	10	-	-	nS
Data Latency	TSLDV	-	-	100	nS
Strobe-	TSLSH	100 + TRD	-	-	nS
Control Hold	TSHCX	0	-	-	nS
Data Hold	TSHDX	50	-	-	nS

NOTE: Address Setup & Hold times are identical to those of Control lines.

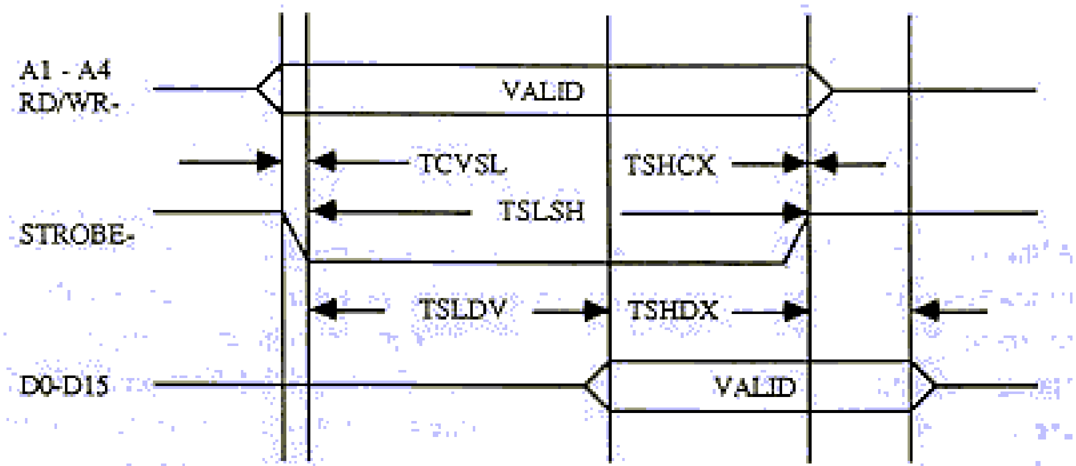


Figure 3-6: ECI Bus Read Cycle

3.2.6 EDD AMPLIFIER ADDRESSING

The ECI Bus address assignments for the X and Y servo drivers are shown in Table 2. The DSC amplifiers addresses are preset by the factory and should not require modification by the user. These settings are consistent with the setup of X and Y EDD modules in other controllers.

TABLE 2. ECI Bus I/O Address Assignments

Control Signals					Selected Function
RD/WR*	A4	A3	A2	A1	
0	1	X	0	0	Z Servo Write (if applicable)
0	1	X	0	1	Y Servo Write
0	1	X	1	0	X Servo Write
1	1	X	0	1	Status Read

NOTE: X = Don't Care; 1 = High Level; 0 = Low Level

4.0 DSC Operation

--- WARNING ---

Each DSC is paired with a specific XY Scan Head. The factory has pre-adjusted the X and Y EDD modules for their respective scanners. The same axis servo amplifier and scanner (i.e. EDD X & scanner X) must be connected together for proper operation. Mis-connection of the XY Head to the DSC (ex. EDD X with scanner Y) may result in **serious damage** to the **amplifier, scanner or both.**

--- EXPLANATION ---

Although some XY Heads (i.e. those with smaller pupils) have similar inertia for the X and Y mirrors, others do not. Even when these two values are close, they are not identical. The EDD servo loop control parameters (servo gain, integrator gain and damping) are tuned to the specific inertial load of the scanner and mirror. The inertial difference, if the X & Y mirrors are swapped, most likely would result in somewhat sub-optimal system performance. This reduction might be small enough to be unnoticed except in the most demanding applications (or when large pupil XY Heads are used.)

In addition, each scanner has a unique position transducer sensitivity and offset. The position amplification circuit of each EDD has been calibrated to the position transducer of the scanner it controls. The probable variation of position signal strength from one scanner to the next is quite large. We would expect it to appreciably affect the system performance whenever incorrectly calibrated. Thus, the scanner and same axis amplifier comprise a matched set and must be used as such.

NOTE: Alterations to the scanner's inertial load (i.e. swapping mirrors), changing scanners or changing amplifiers requires complete re-tuning and calibration of the amplifier and the XY Scan Head. Whenever one channel is re-tuned, because the DSC is optimized for vector operation (see 4.2), the two channels must be re-matched. Therefore, if a DSC or XY Head requires repair, the XY Head and both EDD modules (alternately the entire DSC) must be sent to the designated repair facility.

4.1 XY Head Motion

The EDD accepts binary 16-bit digital inputs, which will be referred to as "digital position" or simply "position". These digital positions are transmitted to the EDD via the Scanner or ECI Bus (as detailed in sections 3.1 & 3.2). There is a linear relationship between the input position and the output beam. A correctly calibrated EDD will produce a beam deflection (rotation), proportionate to the input, as shown in Table Y:

Digital Position	Scanner Rotation	Optical Deflection
0	1 x CCW	-20°
16384	1/2 x CCW	-10°
32768	Centered	0°
48152	1/2 x CW	10°
65535	1 x CW	20°

NOTE: The input values as shown are decimal (base 10). The EDD requires binary input values (base 2): conversion to binary must be performed by the Host System.

Table 3: Digital Position versus Beam Deflection

The actual beam deflection is the end product of a series of signal transformations. In reverse order, the chain of transformations is:

- The new beam deflection is established by the angular displacement of the mirrors.
- Mirror angle is a direct function of scanner rotation.
- The amount of scanner rotation is set by the steady-state current in the scanner drive coils and the closed-loop position signal.
- The position reference signal (desired position) is generated by a 16-bit digital-to-analog converter. This signal and the scanner's (actual) position signal determine how much current is needed in the drive coils to reach and maintain the desired position.
- The Host System supplies the 16-bit digital position signal which initiates the sequence.

4.2 Types of Scanner Drive Signals

--- WARNING ---

> The standard EDD servo driver amplifiers are tuned for vector input ONLY. Briefly, this method of tuning (and operation) provides the highest possible small signal bandwidth, with the loss of stable operation for large input signals. **All input signals (vector and step) must be restricted to within certain safe limits for proper operation.** Exceeding these limits may result in uncontrolled scanner motion which can produce mirror crash or internal damage.

Exceptions:

At customer request, General Scanning will tune the DSC for non-vector operation. If your EDD amplifiers have been tuned for step operation, they will have been optimized for a particular range (size). Inputs outside this range will produce under- and over-damped response for smaller and larger steps, respectively. It is less likely that uncontrolled scanner motion will occur with step-tuned servo drivers.

There are three primary types of scanner motion: step, raster and vector. For comparison purposes, a short description of the advantages and disadvantages of each scanning method is included below:

STEP: Step operation directs the scanner to jump directly to an X,Y coordinate in one motion and with one X,Y input pair. Tuning an amplifier for step performance compromises the speed and performance of the scanner-amplifier system. A relatively narrow range of steps can be tuned for. Outside of this range, the scanner will exhibit degraded performance. This may require large steps to be split into two or more smaller steps.

However, step operation permits the use of simple destination coordinates (without slewing through intermediate points.) This reduces the programming effort and real-time calculations needed to generate X,Y positions.

RASTER: Raster operation orient the scanning along either the X or Y axis. The fast axis (usually X) is scanned at a constant velocity with a digital ramp (created by slewing through intermediate data points.) The slow axis (usually Y) steps, during the retrace interval of the fast axis, to produce the raster lines. Data must be synchronized on the scan line to allow margins at each end for scanner acceleration and deceleration.

Raster tuning compromises vector performance by allowing some overshoot at the fast axis margins (to provide higher speed retrace motion) and by stepping the slow axis. Raster tuning compromises step performance by under-damped scanner motion which dramatically increases settling time.

VECTOR: Vector operation synchronizes the scanning of both the X and Y axes. Both are scanned at a constant velocity by a digital ramp. The Host System generates a sequence of X,Y pairs which are actually very small, equidistant steps. These steps are transmitted to the DSC/EDD at short, regular intervals. The result is a very constant velocity over the entire vector.

Vector tuning produces the best all-around scanner performance. A high slew-rate vector will often reach "stepped" coordinates more rapidly than step tuning can. Raster scans can be executed (when generated as vector streams) at approximately 75% of raster-tuned performance. Vector operation does require a high rate of X,Y coordinates to create the digital input ramp. Step inputs cannot be used with vector-tuned scanner-amplifier pairs.

4.3 Vector Scan Generation

As stated in section 4.2, a vector is a type of ramp waveform, which consists of many small steps. In fact, vectors are generated and sent to the EDD as digital ramps. The velocity of such a ramp is defined as the change in position over time: $\Delta p/t$. As shown in Figure 4-1, this is controlled by both the magnitude of the incremental step (step size: *SS*) and the time interval between successive step outputs (step period: *SP*). Although the Host may be able to generate data over a wide range of step periods (e.g. from very slow to very fast), the capabilities of the EDD and scanner may limit the minimum and maximum transmission rates.

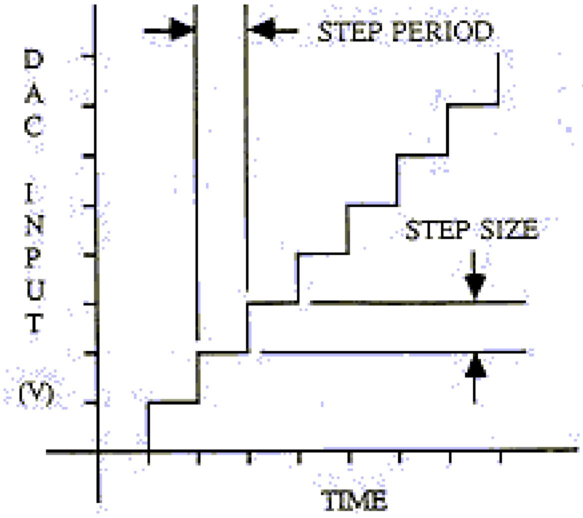


Figure 4-1: Vector Velocity

In general, the effectiveness of vector operation (or vector slewing) is highly dependent on the uniformity of the data steps and their frequency. Each vector has a start and finish endpoint (digital position). The length of the vector must be subdivided into a sequence of incremental (intermediate) points, all of which must be equidistant. (See Figure 4-2.)

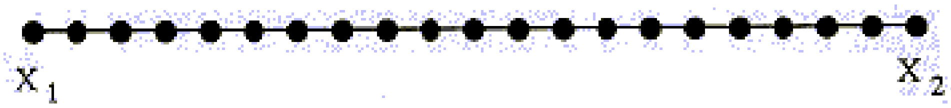


Figure 4-2: 1-Axis Vector Point Generation

For application to two axes, we would say, "Each vector must be subdivided into a sequence of incremental points (pairs), each point lying along the line of the vector, and all points being equidistant." In fact, the 2-dimensional vector is decomposed into the sum of its component X and Y vectors. These vectors are then subdivided into an equal number of uniformly spaced intermediate points, as in Figure 4-3. Note that the point spacing along the X vector is usually not equal to that of the Y vector.

4.4 Optimizing Vector Operation

Different scan heads require different scanner speed to achieve good or optimal performance. It is also very true that the best scanning results are highly dependent on the application. In general, faster scanning will often produce smoother vectors. Yet, for some applications, faster scan rates (vector speeds) may decrease accuracy. If laser power is limited, slower scan rates may be necessary to insure sufficient energy deposition or uniformity.

Be prepared to experiment to determine the most effective parameters for your particular system.

It may also be useful to alter the vector speed for different length vectors. This may improve the system performance in many demanding, high-resolution applications. Laser beam modulation will also be dependent on the scan head performance, as well as many other factors, including the type of laser and the modulation/output control method. Again, the user should anticipate the need for experimentation to determine the best control parameters.

4.5 Debugging Scanning Software

When writing new software for XY scanning, it may be helpful and easier to verify the gross performance of the programs without connecting the scanners. Although many of General Scanning's XY heads have integral position stops, those without such stops can be damaged by uncontrolled scanner motion. It is possible to operate the DSC without the scanners connected. This could alleviate any anxiety over the operation of un-tested software.

There are two approaches to observing the "potential" scanner motion: one requires the user to make up cables for the diagnostic connectors (see section 2.4), the other uses an accessory available from General Scanning, the TOM. If the user chooses to construct their own cables, they should limit the cable length to less than six (6) feet. The signal of primary interest is the DAC output.

The Tuning Operation Module (TOM) [GSI p/n E00-701550] was designed to assist the tuning and trouble-shooting of GSI's scanner control systems. The TOM connects to the diagnostic connector (J3) on the SIC and provides BNC terminations for three of the diagnostic signals: Position Error (10x), Position Output, and DAC Output. As above, the DAC output signal is the important one.

When the Host System sends digital position values to the EDDs, the DACs generate matching output voltages (ref. Figure 4-1). The voltage range is ±5 Volts. These signals can be displayed on an oscilloscope. For repetitive scanning motion, such as a raster waveform, one axis at a time could be displayed and seen quite easily. For vector slewing or jumping to test points, one axis might be more difficult to verify or visualize.

If the oscilloscope is configured for X-Y display and connected to both the X and Y DAC outputs, the 'scope trace (dot) will show where the beam would be deflected. This can be very helpful in finding major errors as well as verifying overall scan operation. However, full-size images on film or other substrates produced by actual beam energy are the only method of proving fine detail correctness.

APPENDIX B: System Connectors

EDD Command Interface Bus:

PLUG	On chassis (DIC) rear panel:	AMP p/n 206650-1 or eq.
RECEPTACLE	On ribbon interconnect cable:	AMP p/n 206655-1 or eq.

37-pin D Connector	Signal Name	37-pin D Connector	Signal Name
1	D0 (LSB)	20	D1
2	GND	21	D2
3	D3	22	GND
4	D4	23	D5
5	GND	24	D6
6	D7	25	GND
7	D8	26	D9
8	GND	27	D10
9	D11	28	GND
10	D12	29	D13
11	GND	30	D14
12	D15 (MSB)	31	ERR THRESH- (opt)
13	A4	32	A3
14	A2	33	A1
15	RD/WR-	34	RESET-
16	GND	35	STB-
17	GND	36	FS0 (unused)
18	FS1 (unused)	37	FS2 (unused)
19	FS3 (unused)		

Euro Diagnostic Bus:

PLUG	On mating cable:	3M p/n 3730-1000 or eq.
RECEPTACLE	On chassis (SIC) rear panel:	AMLAN p/n CDF9P or eq.

9-pin D Connector	Signal Name	9-pin D Connector	Signal Name
1	Position Error (10x)	5	Scanner Current
2	Position Out	6	Velocity Out
3	DAC Out	7	External In
4 & 9	Signal GND	8	AGC Voltage

Euro Servo Control Bus:

PLUG	On scanner/extension cable:	AMP p/n 205206-1 or eq.
RECEPTACLE	On chassis (SIC) rear panel:	AMP p/n 205205-1 or eq.

15-pin D Connector	Signal Name	15-pin D Connector	Signal Name
1 & 9	Drive Current High	7	Heater Current High
2 & 10	Drive Current Low	8	AGC Output
3	Position In (+)	11	Position In (-)
4	Signal GND	12	AGC Sense
5	Heater Current Low	13	AGC GND
6 & 14	(Reserved)	15	Thermistor Input

GSI Scanner Control Bus:

PLUG	On chassis (DIC) front panel:	AMP p/n 206771-1 or eq.
RECEPTACLE	On ribbon interconnect cable:	AMP p/n 206770-1 or eq.

25-pin D Connector	Signal Name	25-pin D Connector	Signal Name
1	GND	14	GND
2	STB X -	15	N.C.
3	D14	16	D15 (MSB)
4	D12	17	D13
5	D10	18	D11
6	D8	19	D9
7	GND	20	GND
8	STB Y-	21	N.C.
9	D6	22	D7
10	D4	23	D5
11	D2	24	D3
12	D0 (LSB)	25	D1
13	STB Z- (unused)		

APPENDIX C: Interface Hardware

Disclaimer:

The inclusion of specific interface hardware for the IBM PC/compatible computer is done solely for the convenience of General Scanning's customers. Those products listed are simply commercially available products, of which we are cognizant. The listing of such hardware should not be misconstrued as a characterization of these products as better (or worse) than competing products. There is no implied warranty of fitness for scanner control using these products and GSI is not responsible for consequential damages or misapplication.

Note: A customized interface cable will be necessary to route the active signals to the correct pins on the Scanner Bus or ECI Bus. This can be done by splitting one end of a ribbon cable and soldering each wire independently to a D connector. Alternately, an in-line re-routing board could be designed (using wire-wrapped connectors) so that normally terminated ribbon cables can be used.

Company: Personal Computing Tools, Inc.
17419 Farley Road
Los Gatos, CA 95030
(408) 395-6600

Product: D-MAX 54: Universal Digital I/O Interface Card

Company: Real Time Devices, Inc.
531 East Marylyn Avenue
P.O. Box 906
State College, PA 16804
(814) 234-8087

Product: DG24: Digital I/O Interface Card

APPENDIX D: Implementing The ECI Bus Interface

Introduction:

This application note details one implementation of General Scanning's ECI (EDD Command Interface) bus. As such, a specific combination of hardware, software, and cabling is used to show how a user could meet the bus requirements. The primary goal of this application note is to convey the idea that using and implementing the ECI bus is really quite simple.

All of the code has been compiled and run and several of the suggested improvements have also been successfully used in scanner testing. The suggestions are not shown in the code for two reasons. First, the code examples would get much larger for very little additional benefit. Second, the changes are quite straightforward and it benefits the interested reader much more to make these changes after reflection, rather than by copying them from this note.

The selection of hardware (I/O card) for this example was arbitrary and is not intended to be a statement of support for one supplier's products over another. The software methods illustrated in this application note are general and can be applied, with sufficient adaptation for hardware differences, to any general purpose I/O card using the 8255 IC. Furthermore, the choice of a programmable IC for this example should not preclude or prevent a user from designing a custom circuit to implement the ECI Bus. Often the hardware solution will be superior when weighing all aspects of a real-world design: speed, overhead, noise immunity, commonality of resources, etc.

ECI Bus Overview:

As covered in section 3.2, the EDD Command Interface (ECI) Bus has been developed for use with highly integrated, digitally-controlled scanning systems. The experience gained from years of creating, using and providing multi-axis scanning systems was central to the creation of the bus specification. As a result, it offers good performance and flexibility without being overly complex. ECI is incorporated into many of General Scanning's current scanning systems, including the DE 2xxx and DE 3xxx, LSS, and DSC. On the DSC, it is accessible to the user as the primary Host scanner control interface.

ECI allows many devices to share one physical cable by using bus addresses to differentiate the devices. This expands the usefulness of the bus beyond simple X, Y, Z 3-axis systems, by allowing other receivers to be added. In addition, the ECI bus provides status readback from the EDD servo control amplifiers. This permits the directing controller (an internal CPU or an external host) to identify scanner or amplifier problems and take appropriate action.

Hardware Overview:

The sample software is written for IBM PC-compatible computers. A low-end clone with 4.77 MHz 8088 CPU, monochrome display and 5.25" floppy disk drives will be quite adequate to execute the software. A faster computer (higher speed 8086, 80286 or 80386) with a hard drive will make program development (compilation and debugging) much easier.

The user will need to purchase a D-MAX 54 Digital I/O Interface Card (see Appendix C) and install it in an empty PC expansion slot. If an alternate I/O card is to be used, changes to the driver routines and cable will probably be required.

An interconnection cable diagram and part numbers are also included in this application note. The cable is correct for the D-MAX 54's 8255 connector pinout. The cable uses unshielded ribbon cable, which may be susceptible to EMI generated by the computer. It is recommended that the cable length be kept as short as possible.

Software Driver Overview:

The software in this application note was written in Turbo C, version 2.0. One goal was to provide software examples that would be easy to understand, self-documenting if possible, and highly portable, making it easy for a user to try them out. This narrowed the choice of programming language to either Pascal or C. C was chosen, not because it had any specific advantages over Pascal, but because it is a "hotter" language presently and more likely to be available to the average user. However, it is possible to code these routines in almost any language with minor difficulty. Once the reader understands the concepts shown, translation into another language will be straightforward.

In keeping with the portability objective, no assembly language has been used in these examples. Not only would this produce much greater machine dependency, it would also complicate the code unnecessarily, obfuscating the techniques being illustrated. It is true that using assembly language would allow faster execution of some routines, but through appropriate coding methods C alone has been adequate even when executing on the slowest (4.77 MHz, 8088) PC-clones.

Turbo C was selected as the development environment, due to its popularity for PC-based applications and its interactive capability. Although most, if not all, PC-based C compilers could easily generate the sample code, Turbo does offer advantages when considering full program development requirements.

Interface Card Architecture:

The D-MAX 54 Digital I/O Interface Card contains two 8255 Programmable Peripheral Interface ICs, two 8253 Counter/Timer ICs,an oscillator clock for the 8253s, an address selection switch, and an interrupt selection block. Physically, the I/O card is an 8-bit "short" PC expansion card. Thus, it can be used in PC, XT or AT style computers.

Each IC (8255 or 8253) occupies four consecutive addresses which are added to the I/O card's base address (see below). The base address is decoded on the D-MAX 54 through the address selection switch. Both the software and hardware must agree on the choice of address. In addition, this address cannot conflict with any other device in the computer. The default address set by the factory (280H) is the base address used in the sample code.

Of the four ICs, only the first 8255 is used in the code samples. Neither the 8253s or their interrupt controller block are used. Although, as a product, the D-MAX 54 provides greater functionality by including these devices, they are not needed to implement the ECI Bus.

Device	Address	Register	Connector
8255 #1	BASE + 00H	Port A	J0
	BASE + 01H	Port B	J0
	BASE + 02H	Port C	J0
	BASE + 03H	Control	—
8255 #2	BASE + 04H	Port A	J1
	BASE + 05H	Port B	J1
	BASE + 06H	Port C	J1
	BASE + 07H	Control	—
8253 #1	BASE + 08H	Counter 0	J2
	BASE + 09H	Counter 1	J2
	BASE + 0AH	Counter 2	J2
	BASE + 0BH	Control	—
8253 #2	BASE + 0CH	Counter 0	J3
	BASE + 0DH	Counter 1	J3
	BASE + 0EH	Counter 2	J3
	BASE + 0FH	Control	—

Table 1: I/O Device Assignments

As can be seen from the table, each of the ICs has a dedicated connector header. The headers are dual-row, 0.1 in center-spaced pins to which a ribbon cable can be mated. The 8255s have 26 pin headers, and the 8253s have 16 pin headers.

In the ECI driver software, the global constant BASE is created and defined to be 280H. This constant is used when transferring data through the pre-defined, low-level DOS Turbo C function "outportb()". In keeping with good coding practice, the offsets for ports A, B, C & control (P) are also defined as constants. If these values have to be altered, they can be found quickly and the change must be made in only one place.

8255 IC Architecture:

The 8255 (8255 is used generically to refer to all variants, including the 8255A and 82C55A) is a general-purpose, programmable I/O device, which was designed for use with all Intel and most other manufacturers' microprocessors. It is used to connect peripheral equipment to the microcomputer system bus. Because it is programmable, normally no additional logic is necessary to interface external devices or components. On the D-MAX 54, the 8255 links the internal PC expansion bus with external peripheral devices.

The 8255 has an 8-bit bidirectional data bus for communication with its associated system microprocessor. Address, read and write signals are used to control the reception and transmission of data and commands to or from the IC. The input/output pins are grouped into 3 8-bit entities, which are called Ports A, B, & C. There is also a control register which is used to program (configure) the 8255. Each of these four internal locations is accessed by setting the two address pins during a data bus cycle. Not surprisingly, the chip addresses (offsets) given in Table 1 are the internal IC addresses for the ports and control register.

The 8255 can be configured in many ways. These setups are grouped into three major classes: Mode 0 - Basic (Simple) I/O; Mode 1 - Strobed I/O; Mode 3 - Strobed Bidi Bus I/O. In this note, only the Mode 0, the most popular, is discussed. Within Mode 0, there are 16 different selectable combinations of I/O pin assignments. Not only can ports A, B, & C be programmed as either inputs or outputs (8 combinations), but Port C is split into the upper (CU) and lower (CL) nibbles and these can be chosen separately (8 more combinations). Only two setups within Mode 0 are used for this example.

Software Driver Design:

The sample code is intended to illustrate basic methods of transmitting digital positions to a scanner controller. There is a shell (main) program which calls the driver routines and creates a slowly-changing triangle wave ramp. The main/shell program is useful primarily as an outline of how the low-level programs act or interact with other portions of a program. The parts of the code which are of greatest significance are the 3 driver functions contained in the edd_io.c file: startup, shutdown, and galvo_send.

During power-up, the D-MAX 54 applies the PC RESET signal to the 8255s and 8253s. RESET clears all of the bits in the control register, resets all of the port latches, and leaves the 8255 with all Port pins as INPUTs (fairly high impedance) and in Mode 0 - word 15. The D-MAX 54 has a pullup resistor (10 KΩ) connected to each Port pin. This ensures that any pin set as an input (or with a 1 value as an output) is read as a logic "1" by an other device.

In the main function, the startup routine is called before any other commands are issued to the 8255. Startup would normally be called when the 8255 set for all inputs, as when it has been RESET. However, startup can re-configure the 8255 from any other setup. Startup establishes a consistent starting point for the software: the 8255 is set to Mode 0 - word 0 (all Port pins as OUTPUTs = "0"). Startup also sets the STROBE bit (Port C pin 7) to 1, which is the inactive state of the ECI STB- signal.

After the user halts the execution of the triangle wave, the main function "tidies up", by calling the shutdown routine. Shutdown is the complement to startup. The 8255 is reprogrammed into Mode 0 - word 15, as if a RESET was sent. During this operation, the STROBE bit remains a 1, so the scanner does not move from its minimum position (where the triangle wave halted.) As a minor improvement, one could keep track of the current position of the galvo (from any routine). Then, prior to calling shutdown, the galvo could be moved via a smooth motion to the center of the field (or any other desired location) before disabling the 8255.

```
/* edd_io.c */

/***************************************************************************************/
/*                                                                                     */
/*        Output module for EDD servo controllers & ECI Bus Interfacing               */
/*                                                                                     */
/*        The data communication is performed via a programmable IC, the 8255:        */
/*                                                                                     */
/*        This chip has 3 8-bit data ports labelled A, B, & C.  An 8-bit external data */
/*        bus allows the IC to be directly connected to many popular microprocessors   */
/*        or microcontrollers.  Intel designed it as an I/O peripheral device for the 8080- */
/*        8085 µP families.  Although the 8-bit interface requires multiple bus cycles */
/*        to transfer 16-bit or larger quantities, the 8255 is still very useful and common. */
/*        (See mode description in startup() comment block for more information.)      */
/*                                                                                     */
/*        NOTE:  These functions are designed specifically for the D-Max 54 PC I/O     */
/*        Interface Card and will probably require modification for use with any other */
/*        8255 I/O Interface Cards.  The card Addressing most likely will be different, */
/*        but the IC control within the "global" addressing should be similar.         */
/*                                                                                     */
/*        The code is written for Turbo C  version 2.0, although a conscious effort    */
/*        has been made to keep the code as portable as possible.  Towards this goal,  */
/*        no Assembly Language has been used.  The small speed penalty for using       */
/*        only C is insignificant compared to the greater comprehension & readability. */
/*                                                                                     */
/***************************************************************************************/

#include "dos.h"                     /* Turbo C low-level DOS routines                 */

#define   BASE     0x280             /* Hex Address of D-MAX I/O Card in PC memory map */

#define   DEV      0                 /* Chip Number of 8255 on D-Max PCB (1 of 2 ICs)  */

#define   A        0                 /* Offset for Port A within one 8255:  High Byte Data */
#define   B        1                 /* Offset for Port B within one 8255:  Low Byte Data  */
#define   C        2                 /* Offset for Port C within one 8255:  Control lines  */
#define   P        3                 /* Offset for Control (programming) Port within one 8255 */

                                     /* Upper Nibble of Port C = EDD Control Lines     */

#define   STB_BIT  7                 /* Strobe line is MSB of Port C within 8255       */
#define   DIR_BIT  5                 /* Direction of Xfer:  1 = READ, 0 = WRITE (RD/WR-) */

                                     /* Lower Nibble of Port C = EDD Address Bits      */

#define   A4       3                 /* Address Bit #4 = Port C Bit 03                 */
#define   A3       2                 /* Address Bit #3 = Port C Bit 02                 */
#define   A2       1                 /* Address Bit #2 = Port C Bit 01                 */
#define   A1       0                 /* Address Bit #1 = Port C Bit 00                 */
```

```
/********************************************************************************************/

void            startup(void)

/********************************************************************************************/
/* Startup:  Function initializes the 8255 for correct operational mode & sets STROBE line = 1 (HIGH).   */
/********************************************************************************************/
/*                                                                                          */
/* The 8255 has three (3) major modes of operation:                                         */
/*              Mode 0 - Basic (Simple) Input & Output for each of the three ports, greatest flexibility  */
/*              Mode 1 - Strobed Input & Output groups one-half (1/2) of Port C with A & B Ports          */
/*              Mode 2 - Strobed Bi-Directional Bus-Oriented I/O uses only Ports A & C, very structured   */
/*                                                                                          */
/* At power-up, the 8255 receives a RESET pulse from the PC's backplane I/O connector.  A reset pulse  */
/* clears all bits in the control register and sets all Port pins (24 in all) to INPUT (fairly high-impedance)  */
/* with the IC programmed to Mode 0.  The D-Max PCB has 10KΩ pullups on each 8255 pin.      */
/*                                                                                          */
/* In 8255 Mode 0, there are 16 different selectable combinations of I/O pin assignments.  Each of these,  */
/* including Modes 1 & 2 are fully documented in any 8255 or 82C55 data sheet.  Startup configures the    */
/* 8255 for Mode 0, with all pins as OUTPUTS.  This gives the greatest flexibility for bus control and is  */
/* the most appropriate configuration for straightforward custom bus control.  The control word is 0x80H  */
/*                                                                                          */
/********************************************************************************************/

{
int             ctrlport;                       /* variable for addressing Port P(Programming)            */
int             s_stb_h;                        /* variable to Set_STroBe_High                            */
int             mode0_out;                      /* variable for control word SIMPLE MODE 0, all outputs   */

ctrlport = (BASE + (DEV << 2) + P);             /* init Port P (control) address & Port C bit manipulation */
s_stb_h = ((STB_BIT << 1) | 1);                 /* init high STROBE for bit manipulation                   */
mode0_out = 0x80;                               /* 1000 0000 bit pattern for 8255 control port             */

outportb(ctrlport, mode0_out);                  /* send control word to 8255 for mode0 output setup        */
outportb(ctrlport, s_stb_h);                    /* force STROBE bit = 1 (HIGH) this is the inactive state  */

} /* end of startup */

/********************************************************************************************/

void            shutdown(void)

/********************************************************************************************/
/* Shutdown:  Function resets the 8255 to mode 0 with all pins as inputs (as if IC was RESET).            */
/********************************************************************************************/

{
int             ctrlport;                       /* variable for addressing Port P(Programming)            */
int             s_stb_l;                        /* variable to Set_STroBe_Low                             */
int             mode0_in;                       /* variable for control word SIMPLE MODE 0, all inputs    */

ctrlport = (BASE + (DEV << 2) + P);             /* init Port P (control) address & Port C bit manipulation */
s_stb_l = (STB_BIT << 1);                       /* init low STROBE for bit manipulation                    */
mode0_in = 0x9b;                                /* 1001 1011 bit pattern for 8255 control port             */

outportb(ctrlport, mode0_in);                   /* program all pins as inputs, set up chip as inactive     */

} /* end of shutdown */
```

```
/******************************************************************************************/

void            galvo_send(int axis, unsigned short gp)

/******************************************************************************************/
/* Galvo_send:  Function transmits a single galvo position (gp) of 16 bits to the driver PCB (an EDD).      */
/* The beam axis is specified as X or Y by the calling routine.  This is converted into ECI signals.        */
/******************************************************************************************/
/*                                                                                                          */
/* Following the ECI Write Cycle Timing Diagram, the Address (A4-A1) and RD/WR- control lines are          */
/* set prior to starting the /STROBE pulse.  Also, the data lines are set while /STROBE is = 0 (TRUE).     */
/* When sufficient processing time exists, the data lines can be set before /STROBE is TRUE.  This will    */
/* eliminate any potential ringing or pickup on the /STROBE line from the switching data lines.  These     */
/* considerations are important when long cable lengths and little shielding are used.  The data lines are not */
/* TRI-stated, but this is unimportant since the EDD is not instructed to drive STATUS onto the bus.       */
/*                                                                                                          */
/******************************************************************************************/

{
int             aport, bport, cport, ctrlport;   /* variables for addressing Ports A, B, C, & P(Programming)  */
int             s_stb_h, s_stb_l;                /* variables to Set_STroBe_High & Set_STroBe_Low             */
int             dir_rd, dir_wr;                  /* variables to set DIR_BIT to ReaD & DIR_BIT to WRite       */
int             x_write, y_write;                /* variables to set ECI address lines & keep STROBE HIGH     */

aport = (BASE + (DEV << 2) + A);                 /* init Port A address                                       */
bport = (BASE + (DEV << 2) + B);                 /* init Port B address                                       */
cport = (BASE + (DEV << 2) + C);                 /* init Port C address                                       */
ctrlport = (BASE + (DEV << 2) + P);              /* init Port P (control) address & Port C bit manipulation   */

s_stb_h = ((STB_BIT << 1) | 1);                  /* init high STROBE for bit manipulation                     */
s_stb_l = (STB_BIT << 1);                        /* init low STROBE for bit manipulation                      */

dir_rd = ((DIR_BIT << 1) | 1);                   /* init high (RD) direction for bit manipulation             */
dir_wr = (DIR_BIT << 1);                         /* init low (WR-) direction for bit manipulation             */

x_write = 0x8a;                                  /* 10001010 bit pattern = X EDD w/inactive STROBE            */
y_write = 0x89;                                  /* 10001001 bit pattern = Y EDD w/inactive STROBE            */

/******************************************************************************************/
/* set the EDD address lines (A4-A1) according to the axis selection input parameter, set the data transfer */
/* direction control (DIR_BIT) line to WRITE-, and keep the STROBE line inactive (HIGH).                   */
/******************************************************************************************/

if (axis == XAMP)                                /* XAMP is globally defined for caller & output routines     */
   outportb(cport, x_write);                     /* bit pattern sets WRITE cycle, STROBE = 1, & A4-A1         */

else if (axis == YAMP)                           /* YAMP is globally defined for caller & output routines     */
   outportb(cport, y_write);                     /* bit pattern sets WRITE cycle, STROBE = 1, & A4-A1         */

/* axis is only defined for XAMP & YAMP, but an error trap could be added here...                           */

/******************************************************************************************/
/* begin the WRITE cycle (on the ECI bus) by setting STROBE- to the active state (0) = LOW                 */
/******************************************************************************************/

outportb(ctrlport, s_stb_l);                     /* force STROBE bit = 0 (LOW), modify only the STROBE        */
                                                 /* bit, using the built-in 8255's Port C bit manipulation    */
```

```
/************************************************************************************/
/* xmission portion of the port access & control routine: send data to the galvo control amplifier!    */
/************************************************************************************/

                                        /* mask lower eight bits to isolate least significant Byte,     */
outportb(bport, (gp & 0xff));           /* lsB is sent first to minimize "jump" if no STROBE            */

                                        /* mask upper eight bits & shift right eight bits to create     */
outportb(aport, ((gp & 0xff00) >> 8));  /* most significant Byte, msB is sent 2nd for same reasons     */

/************************************************************************************/
/* complete the WRITE cycle by returning the /STROBE line to the inactive state                        */
/************************************************************************************/

outportb(ctrlport, s_stb_h);            /* force STROBE bit = 1 (HIGH), modify only the STROBE  */
                                        /* bit, using the built-in 8255's Port C bit manipulation       */

} /* end of galvo_send */


/************************************************************************************/

/* g_shell.c */

/************************************************************************************/
/* G_shell is a sample program which uses three functions which are written for the D-Max 54 I/O Card.  */
/************************************************************************************/
/*                                                                                                      */
/* The program provides a simple test for the low-level routines and can be easily modified or expanded. */
/* It is intended and expected that these functions and sample usage will be a guideline to the new user. */
/*                                                                                                      */
/* As discussed in the text of the application note, there are many enhancements which can and should be */
/* made to these basic routines for real-world usage. Such modifications would speed up data transmission */
/* to the galvo and make the code more bullet-proof.                                                    */
/*                                                                                                      */
/* NOTE: As it is written, the code assumes that the programmer will observe the DAC OUT signal at     */
/* the SIC's diagnostic connector. This voltage is valid without a scanner connected. Due to a startup */
/* jump, which has not been addressed in these functions, the code should be tested WITHOUT a scanner  */
/* connected to the EDD amplifier. Unless your amplifiers are STEP-TUNED, this initial jump could      */
/* cause the servo control loop to become temporarily unstable and cause galvo/mirror damage.          */
/*                                                                                                      */
/************************************************************************************/


#include    <stdio.h>                   /* standard I/O library routines printf, etc.                   */
#include    <conio.h>                   /* standard console I/O routines ...                            */

#include    "edd_io.c"                  /* low-level EDD-specific routines                              */

/************************************************************************************/
/*                  Constant Declarations                                                              */
/************************************************************************************/

#define     MINPOS      0               /* smallest digital scanner position                            */
#define     MAXPOS      65535           /* largest digital scanner position                             */

#define     XAMP        10              /* lower bits of EDD Address for X-axis (in decimal)            */
#define     YAMP        9               /* lower bits of EDD Address for Y-axis (in decimal)            */
```

```
/*****************************************************************************/
/*                 Main Program Module                                       */
/*****************************************************************************/

main ()                              /* do a simple example of generating scanner positions      */

{
void startup();                      /* declarations of non-integer returning functions          */
void shutdown();
void galvo_send();
void clrscr();

unsigned short position;             /* variable to pass digital scanner positions to output routine */
int axis;                            /* variable to pass which beam axis will be modified        */

axis = XAMP;                         /* set axis for X galvo: only 1 axis is used in this program */

startup();                           /* init/activate first 8255 on I/O card for Mode 0 - all outputs */

clrscr();                            /* erase screen, move cursor to upper left corner           */

printf("\n\n\n");                    /* display a message describing what the program does and   */
                                     /* how the user can stop it when they get bored             */

printf("\tThis program generates a slow-moving, full-field triangle wave ramp\n");
printf("\tfor the X-axis scanner. The execution of the program can be checked by\n");
printf("\tobserving the DAC OUT voltage from the SIC's diagnostic connector.\n\n");
printf("\tTo QUIT the program, press any (a) key.\n");
printf("\tThe program will complete the current full triangle before halting.\n");

while (!kbhit())                     /* loop until any (a) key is depressed by the user: ! = NOT  */
  {
  for (position = MINPOS; position < MAXPOS; position++)   /* waxing part of triangle waveform  */
      galvo_send(axis, position);                          /* loop from MIN to MAX by ones      */
                                                           /* send to X scanner as position     */

  for (position = MAXPOS; position > MINPOS; position--)   /* waning part of triangle waveform  */
      galvo_send(axis, position);                          /* loop from MAX to MIN by ones      */
                                                           /* send to X scanner as position     */

  } /* end of while loop */

shutdown();                          /* deactivate I/O card == set all lines to input state       */

clrscr();                            /* wipe screen                                               */

} /* end of g_shell program */
```

ADDENDUM A:

CONNECTING THE SINGLE AXIS DSC POWER SUPPLY

Single axis Digital Controllers have a different power supply from the one used in dual axis controllers. Please note the following changes:

INPUT SPECIFICATIONS:

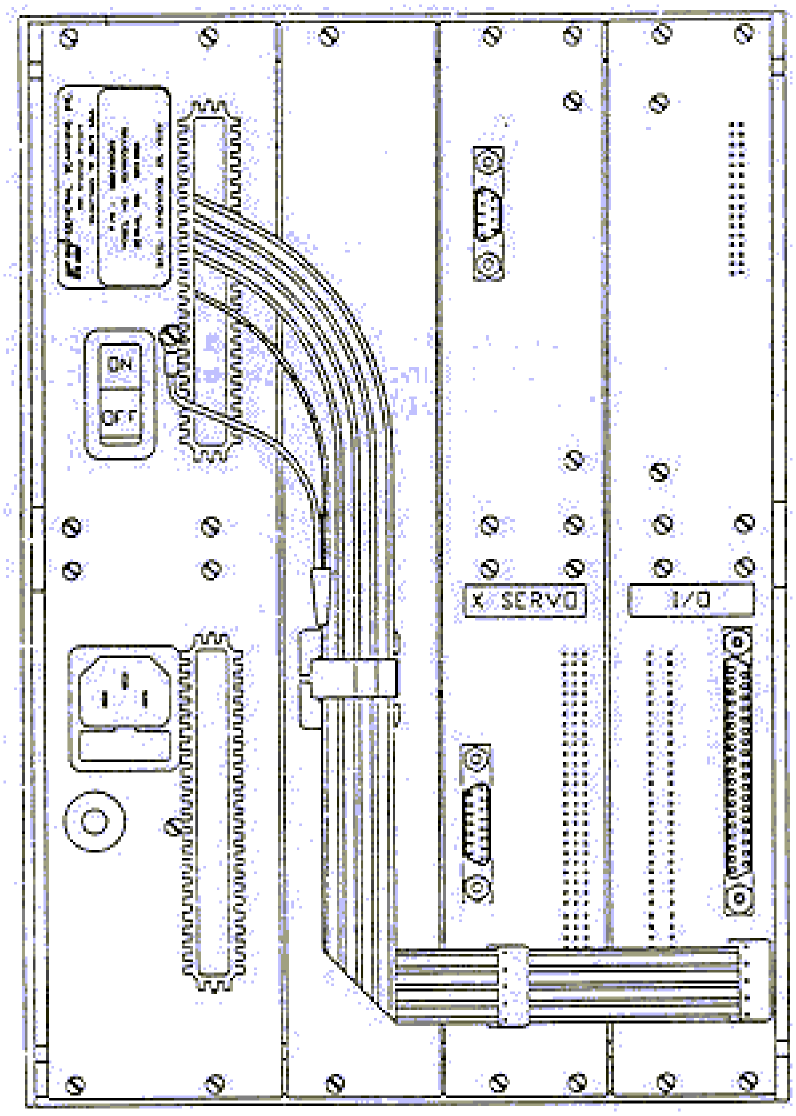
Input voltage: 85-265 VAC Universal

Input frequency: 50/60 Hz

Input current: 1.0A (rms) for 115 VAC
0.6A (rms) for 230 VAC

There are no jumpers to set to use the DSC with either the 115 VAC or the 230 VAC.

Other than having a single servo driver and a different power supply, there are no other differences between the dual axis and single axis Digital Scanning Contollers.



REAR VIEW