A Manual for the UMASS JPL/Stanford Hand Hardware

Miles Thomas Lane

Computer and Information Science Department
University of Massachusetts

COINS Technical Report 88-61

April 1987

A MANUAL FOR THE UMASS JPL/STANFORD HAND HARDWARE

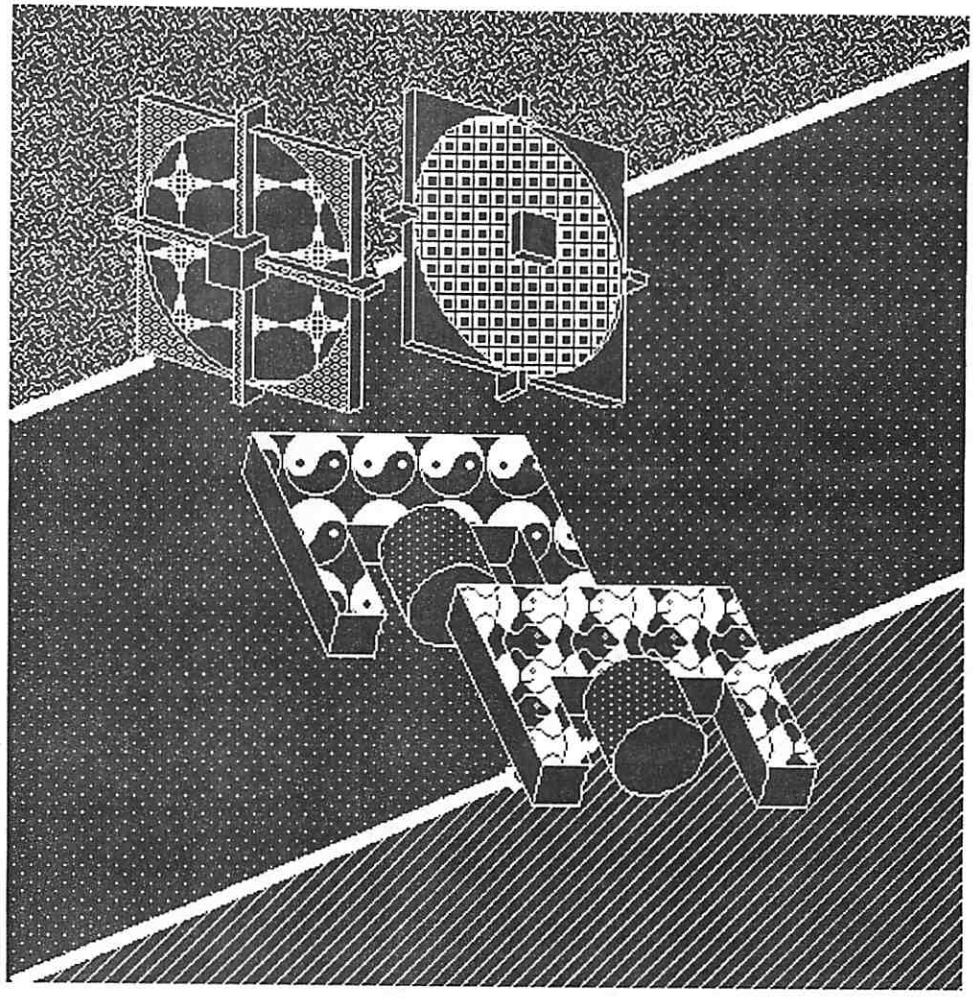
by Miles Thomas Lane

Laboratory for Perceptual Robotics
Department of Computer and Information Science
University of Massachusetts at Amherst, MA

April 1987

Acknowledgments

To my mom and dad, I send a shout of love. They are true and wonderful parents and friends. To my various mentors, scholastic and spiritual, I give thanks.



Contents

List of Figures

Chapter 1

Introduction

1.1 Background Information

This report has been written to provide basic information regarding the JPL/Stanford hand and its I/O hardware. Most importantly, this report contains the information needed to properly understand, troubleshoot, and maintain the hardware associated with running the hand – the great majority of which was designed, assembled and tested in-house through the efforts of a number of LPR (Laboratory for Perceptual Robotics) members, primarily myself, over the last two years – from a PDP-11 or MicroVax-II computer. Also, photocopies of documentation published by the manufacturers of selected hand and I/O components is included in the report by permission of the respective companies. Documentation is provided regarding: the drive motors — PITTMAN LO-COG 7214, the motor position encoders — Hewlett-Packard HEDS-5010-A05, the strain gauges — BLH SR-4, the linear current amplifiers — APEX PA12, the D/A and A/D converters — Analog Devices AD3860 and AD574, and the Q-Bus parallel I/O board — Digital Equipment Corporation DRV11J.

The JPL/Stanford hand was designed by Dr. Ken Salisbury while a student at Stanford University. The hand was purchased by the UMASS-LPR to serve as a test bed for our work in the areas of Distributed Processing, Dynamic Planning, Real-time Control and others. For a picture of the hand, turn to Figure 1.2, and for more detailed images showing a single finger, please see Figures 1.4 and 1.5.

S. T. Venkataraman is in the process of implementing his low-level JPL/Stanford hand

control theories. We pooled our knowledge during the hardware design stages and the results have been fairly satisfactory.

While the hardware is presently configured for control by PDP–11/23 and MicroVax–II computers, only one in–house wire–wrap I/O board must be modified to interface the hardware to new computer systems. This board is responsible for taking the control signals from the computer's parallel I/O ports and translating them into new control signals that can drive the hand hardware.

1.2 A System Overview

The system is composed of the following building blocks: sensory feedback, motor drive, computer I/O and power supply circuitry. Look at the system block diagram in Figure 1.1 for clarification. Hardware debugging is being facilitated by the use of a diagnostic program written in C and running under Digital Equipment Corporation's VAXELN (a real-time operating system).

The hand has two types of sensory feedback: motor position and tendon tension. The positional information originates from optical encoders mounted on the servomotor drive shafts. The computer keeps track of the signals from the encoders through the use of 16–bit decoder/counters. To acquire the tendon tension information, each strain gauge is placed across a wheat–stone bridge and the resulting low–level voltage signal is fed into an operational amplifier. The resulting amplified signal is sent to a 12–bit A/D (analog to digital) converter whose output is available to the computer. To drive the motors, the computer sends numbers to 12–bit DACs (digital to analog converters) whose voltage outputs drive linear current amplifiers. The resulting output currents drive the hand's DC servomotors with desired amounts of torque.

Computer to I/O hardware interfacing is handled primarily by one in–house wire–wrap board. This board translates the control signals provided by three DEC DRV11Js (four–port 16–bit parallel I/O boards) into signals appropriate for driving twelve in–house single–tendon I/O boards. These single–tendon I/O boards each hold an A/D, DAC and a 16–bit counter with additional decoding circuitry.

At the root of the hardware system for the hand are more than fourteen power supplies. These include Vcc for digital logic, ±12V for linear ICs, ±15V for A/D and D/A converters, +7V and −12V to drive the power amps, and others. Housed in a single chassis are the I/O system power supplies, the in–house I/O boards, a bank of fuses, fans and some AC power relays and switches.

1.3 The Hand Mechanism

In discussing the various parts of the hand, terminology illustrated below in Figures 1.6 and 1.7 will be used. These figures also show a finger as it appears when disassembled. Please refer to these figures if you are unsure about the construction of the hand's fingers. In the following paragraph is a verbal description of the hand mechanism's composition.

The JPL/Stanford hand consists of three articulated aluminum fingers mounted together on an aluminum base. The fingers have cylindrical polymer fingertips. Each finger has three joints which are cooperatively actuated by four tendons. The tendons are teflon–coated steel cables. Each tendon is driven by a motor containing a samarium–cobalt field magnet. The samarium-cobalt composition magnet has an extremely strong magnetic field — allowing the hand's small motors to develop large torques. The hand's twelve motors are mounted together in a single housing (see Figure 1.3).

The tendons run through flexible teflon-lined conduits from the motor housing to the base of the hand (see Figure 1.8). Also shown in Figure 1.8 are two of the strain gauges in one finger's knuckle. Each drive tendon passes over a roller attached to one of the strain gauges. Two figures show the layout of the back of a knuckle — Figures 1.9 and 1.10. All the figures mentioned in this paragraph are helpful references when taking apart and reassembling the hand.

To make this report as complete as possible, photocopies of manufacturer documentation of the main JPL/Stanford hand components — the motors, optical encoders and strain gauges — are included after the figures in this section.

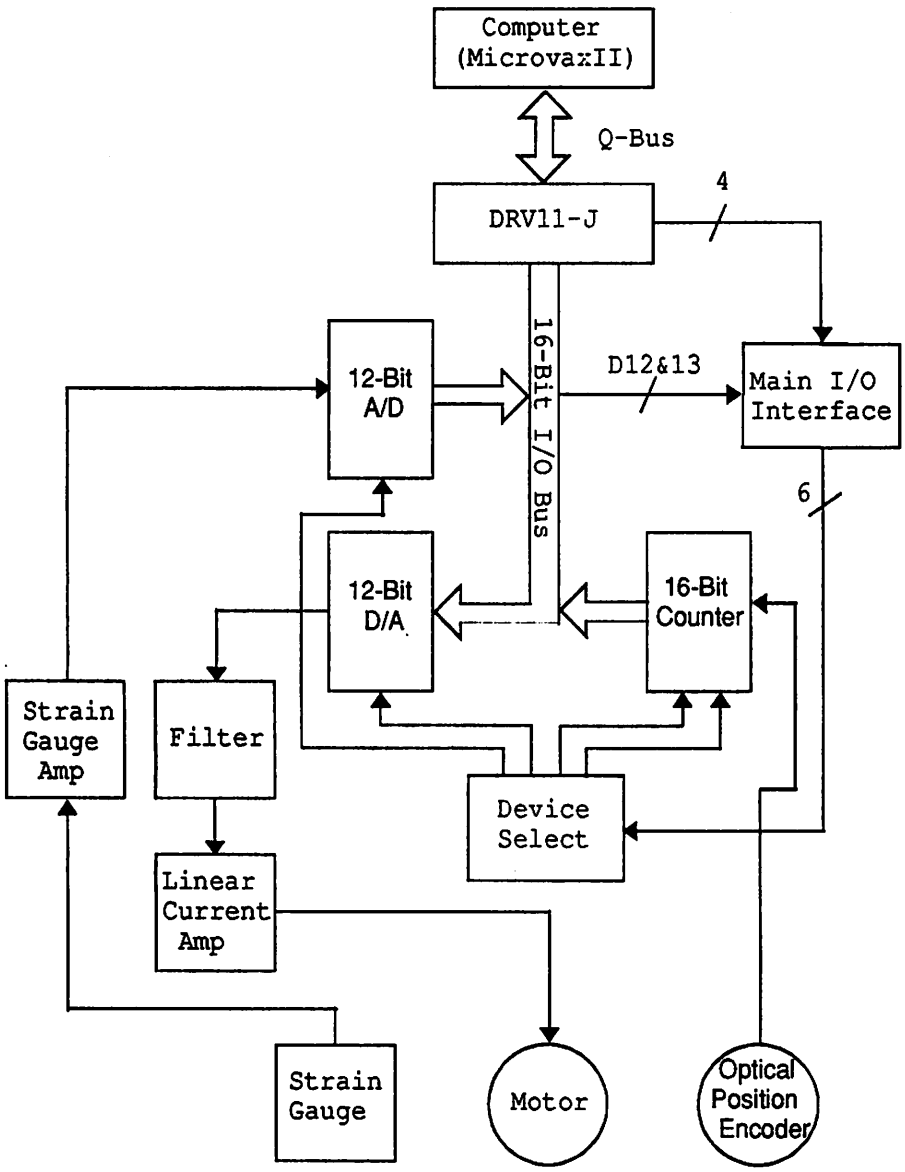


Figure 1.1: A System Block Diagram

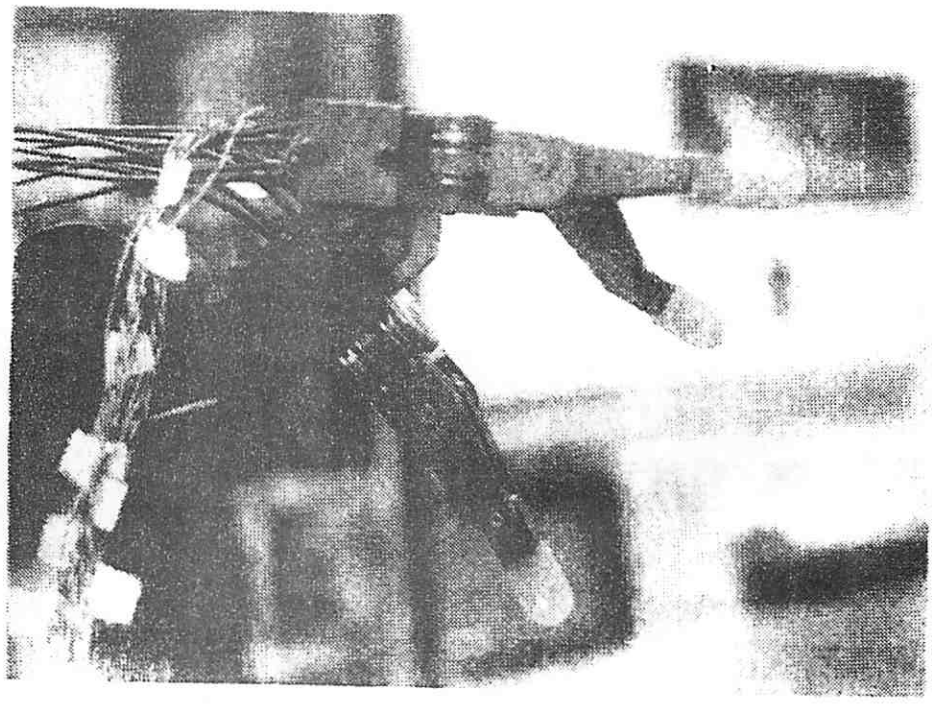


Figure 1.2: A picture of the JPL/Salisbury hand

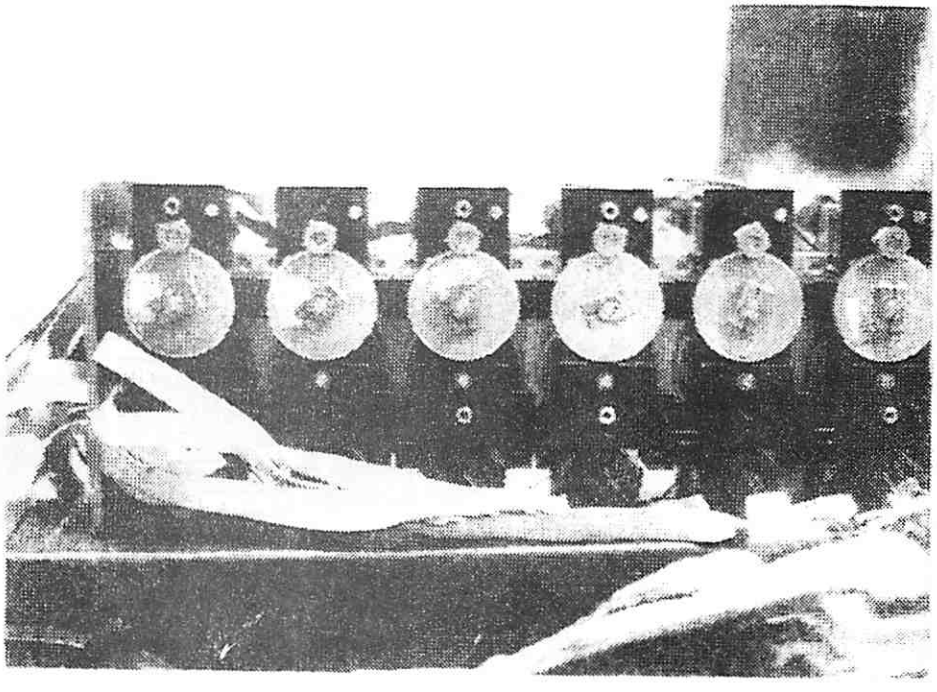


Figure 1.3: A picture of the motor housing

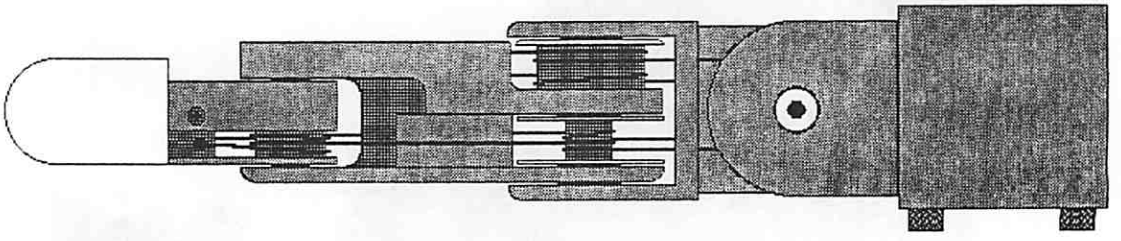


Figure 1.4: A finger assembly – top view

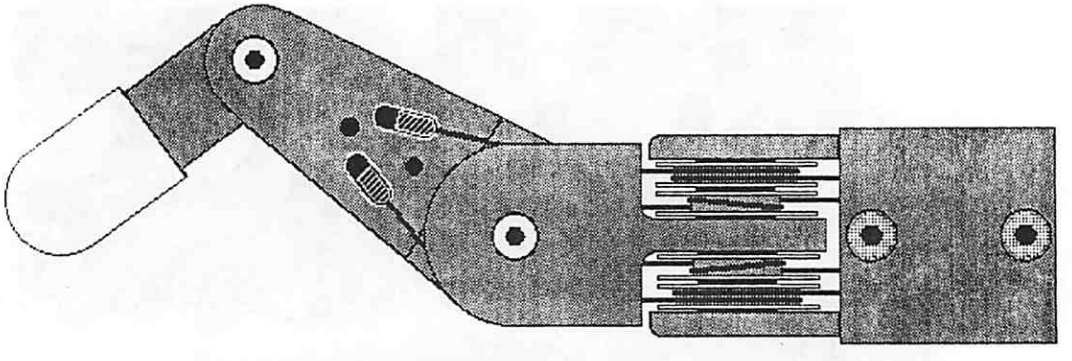


Figure 1.5: A finger assembly – side view

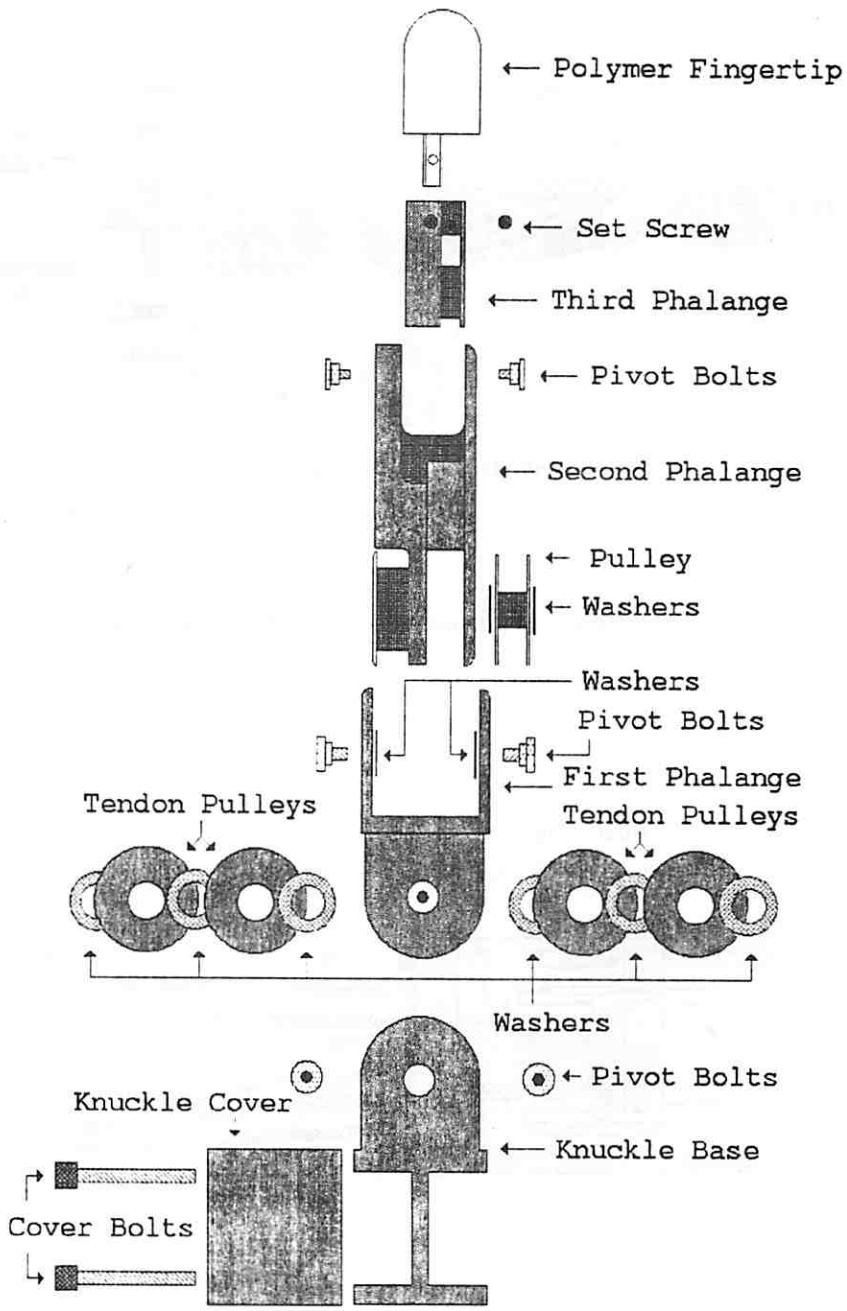


Figure 1.6: A disassembled finger – top view

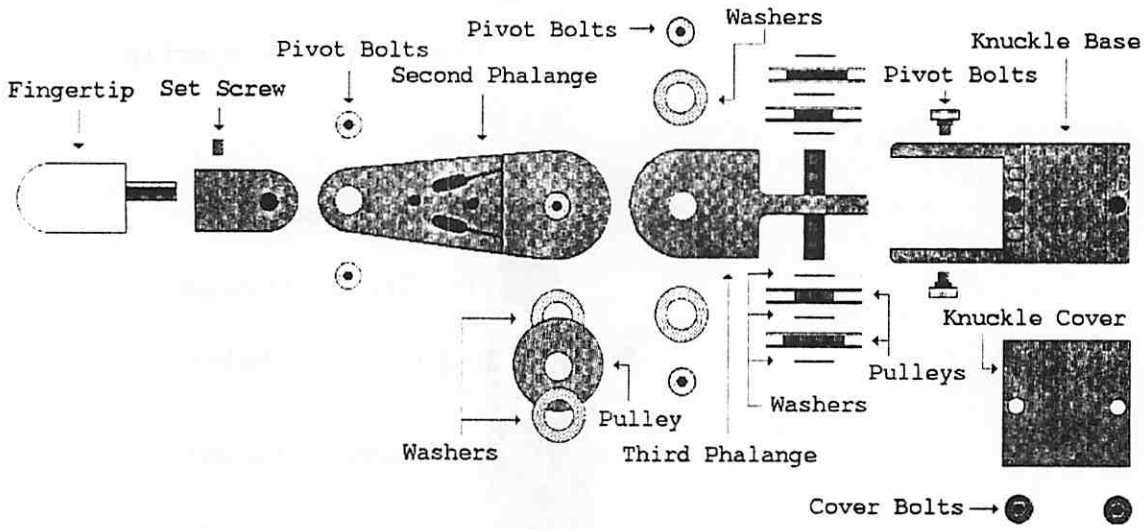


Figure 1.7: A disassembled finger – side view

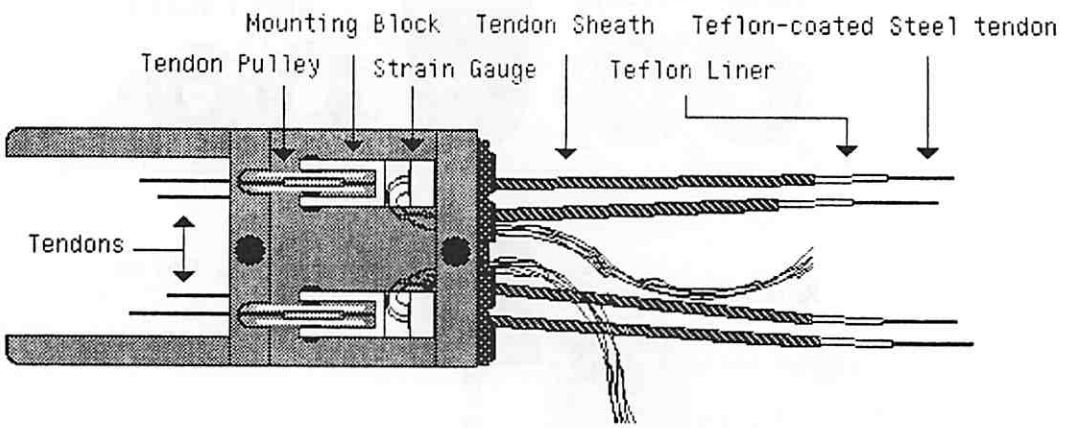


Figure 1.8: Finger base with strain gauge and cable construction shown

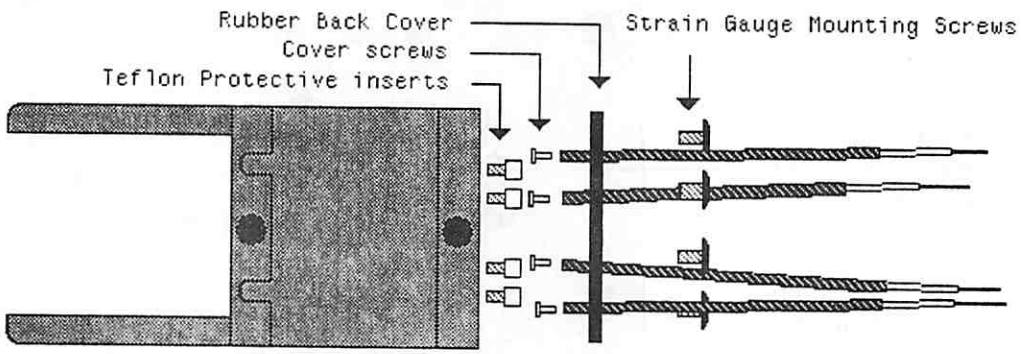


Figure 1.9: The rubber backing and strain gauge mounting screws.

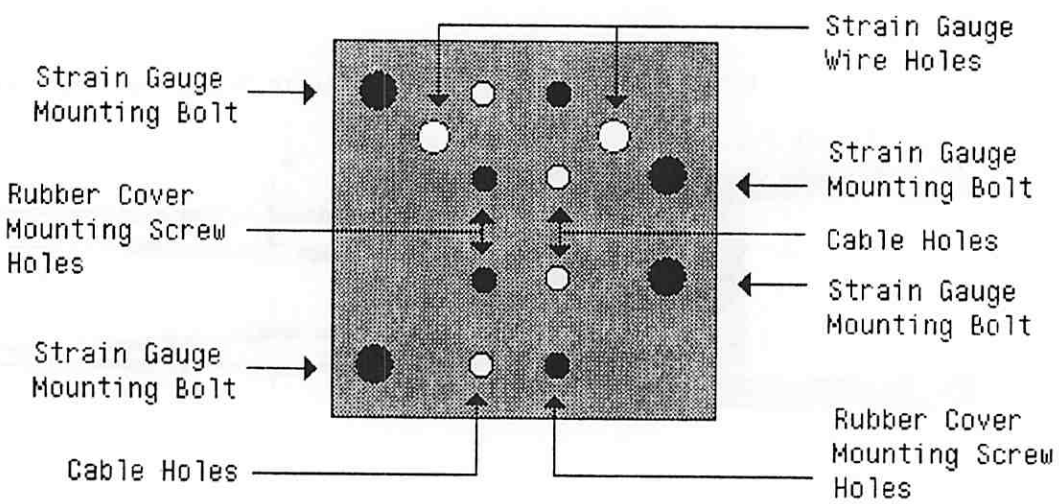


Figure 1.10: The back of a finger base with holes labeled by function

MANUFACTURER'S DOCUMENTATION FOLLOWS.

BULLETIN 7000
Revised July, 1985

LO-COG® D-C SERVO MOTORS

Series 7000 — 1 in. x 1¼ in. cross-section with Stall Torques from 14 to 24 oz.-in.

This family of miniature, four-pole, rare earth field motors features minimal magnetic cogging at low speeds as well as exceptional performance per unit volume. Series 7000 servo motors have been developed, produced, and proved for long, maintenance-free operation. Premium quality materials coupled with the very latest manufacturing and assembly techniques provide highest performance as well as superior reliability. In addition, *every* motor is subjected to complete testing of all critical parameters under both load and no-load conditions in the unique PITTMAN® computerized final test station. A printout of test data is kept on file for any further reference.

Speed, voltage, current and torque characteristics can be varied over a wide range to meet specific needs. Please note that armature winding changes, and any relatively simple modifications that do not require extensive redesign or tooling alterations, may be specified for prototype quantities at only nominal costs.

PRIMARY DESIGN FEATURES OF THE SERIES 7000

PEAK TORQUE (STALL)
from 14.3 to 24.1 oz.-in.

NO-LOAD SPEEDS
from about 4,300 to 5,000 rpm for standard motors at rated voltages.

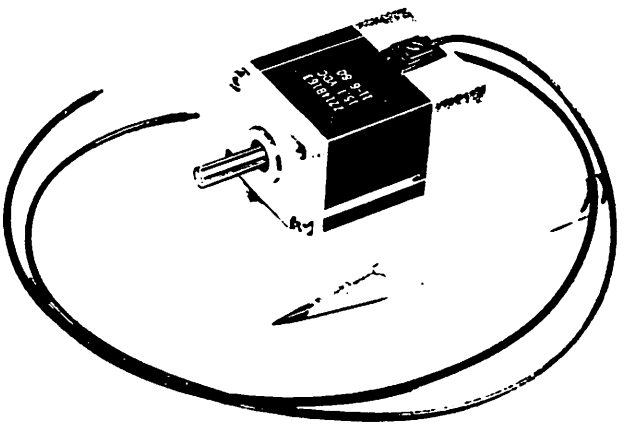
ARMATURES
13-slot design, skewed for optimum reduction of reluctance torque. Laminations are silicon steel, epoxy insulated, with standard windings of film-insulated (class 180°C) magnet wire — impregnated with polyester resin and baked.

COMMUTATORS
diamond turned after armature assembly to ensure optimum concentricity and long brush life.

BRUSHES
silver-graphite standard.

FIELD
Samarium-cobalt magnets in sintered magnetic iron housing. End bells are zinc die castings.

BEARINGS
double-shielded ABEC-3 ball bearings standard.



TELEPHONE (215) 256-6601
FAX (215) 256-1338

PITTMAN®
HARLEYSVILLE, PA 19438-0003 USA
A DIVISION OF PENN ENGINEERING & MANUFACTURING CORP

TELEX 283668
TWX 510-661-8696

MOTOR SIZE CONSTANTS

ITEM	MOTOR SIZE CONSTANTS	UNITS	SYMBOL	7212, 7312 VALUE	7213, 7313 VALUE	7214, 7314 VALUE
1	PEAK TORQUE (STALL)	OZ-IN	TPK	14.3	17.7	24.1
1a	Peak torque (stall)	mN•m	TPK	101	125	170
2	MOTOR CONSTANT	OZ-IN/$\sqrt{W}$	PKO	2.14	2.23	2.52
2a	Motor constant	mN•m/$\sqrt{W}$	PKO	15.1	15.7	17.8
3	POWER FOR PEAK TORQUE	W	PWR	48.0	65.4	94.1
4	DAMPING (ZERO SOURCE IMPED.)	OZ-IN/(rad/s)	DPO	3.22×10^{-2}	3.64×10^{-2}	4.61×10^{-2}
4a	Damping (zero source imped.)	mN•m/(rad/s)	DPO	0.227	0.257	0.326
5	DAMPING (INFINITE SRCE. IMPED.)	OZ-IN/(rad/s)	DPI	2.71×10^{-4}	3.30×10^{-4}	3.88×10^{-4}
5a	Damping (infinite srce. imped.)	mN•m/(rad/s)	DPI	1.9×10^{-3}	2.3×10^{-3}	2.7×10^{3}
6	NO LOAD SPEED	REV/MIN	SNL	4300	4550	5060
6a	No load speed	rad/s	WNL	450	476	530
7	ELECTRICAL TIME CONSTANT	ms	TCE	0.41	0.42	0.45
8	MECHANICAL TIME CONSTANT	ms	TCM	5.0	5.20	4.73
9	FRICTION TORQUE	OZ-IN	TOF	0.50	0.60	0.80
9a	Friction torque	mN•m	TOF	3.5	4.2	5.6
10	ARMATURE INERTIA	OZ-IN-s^2	ERT	1.61×10^{-4}	1.89×10^{-4}	2.18×10^{-4}
10a	Armature inertia	kg•m^2	ERT	1.14×10^{-6}	1.33×10^{-6}	1.54×10^{-6}
11	MOTOR WEIGHT	OZ	WGT	4.76	5.15	5.57
11a	Motor mass	kg	WGT	0.13	0.15	0.16
12	THEORETICAL ACCELERATION	rad/s^2	CEL	88800	93700	110600
13	THERMAL TIME CONSTANT	MIN	TCT	10.3	11.2	12.0
14	ULTIMATE TEMP. RISE/WATT	°C/W	TPR	21.0	20.0	17.2
15	MAXIMUM WINDING TEMPERATURE	°C	TMX	155	155	155

WINDING CONSTANTS (other windings available)

MODEL 7212, 7312

*	WINDING CONSTANTS	UNITS	SYMBOL	WDG #1	WDG #2	WDG #3	WDG #4
16	VOLTAGE	V	VLT	12.0	15.1	19.1	24.0
17	CURRENT (STALL)	A	AMP	4.00	3.23	2.57	2.02
18	TORQUE CONSTANT	OZ-IN/A	TPA	3.70	4.57	5.77	7.29
18a	Torque constant	mN•m/A	TPA	26.1	32.3	40.7	51.5
19	TERMINAL RESISTANCE	OHMS	RTR	3.00	4.67	7.44	11.9
20	BACK EMF	V/(rad/s)	BEF	0.0261	0.0323	0.0407	0.0515
21	INDUCTANCE	mH	DUK	1.23	1.92	3.00	4.92
22	CURRENT (NO LOAD)	A	INL	0.150	0.121	0.096	0.076

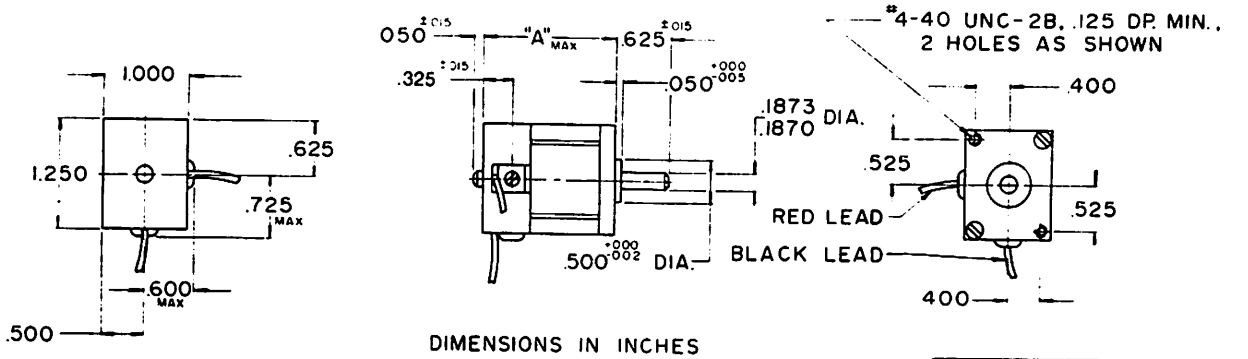
MODEL 7213, 7313

	WINDING CONSTANTS	UNITS	SYMBOL	WDG #1	WDG #2	WDG #3	WDG #4
16	VOLTAGE	V	VLT	12.0	15.1	19.1	24.0
17	CURRENT (STALL)	A	AMP	5.47	4.27	3.42	2.73
18	TORQUE CONSTANT	OZ-IN/A	TPA	3.35	4.29	5.36	6.70
18a	Torque constant	mN•m/A	TPA	23.7	30.3	37.8	47.3
19	TERMINAL RESISTANCE	OHMS	RTR	2.19	3.54	5.58	8.80
20	BACK EMF	V/(rad/s)	BEF	0.0237	0.0303	0.0379	0.0473
21	INDUCTANCE	mH	DUK	0.914	1.50	2.34	3.66
22	CURRENT (NO LOAD)	A	INL	0.202	0.154	0.125	0.099

MODEL 7214, 7314

	WINDING CONSTANTS	UNITS	SYMBOL	WDG #1	WDG #2	WDG #3	WDG #4
16	VOLTAGE	V	VLT	12.0	15.1	19.1	24.0
17	CURRENT (STALL)	A	AMP	7.86	6.28	4.92	3.92
18	TORQUE CONSTANT	OZ-IN/A	TPA	3.16	3.95	5.06	6.32
18a	Torque constant	mN•m/A	TPA	22.3	27.9	35.7	44.6
19	TERMINAL RESISTANCE	OHMS	RTR	1.53	2.41	3.88	6.12
20	BACK EMF	V/(rad/s)	BEF	0.0223	0.0279	0.0357	0.0446
21	INDUCTANCE	mH	DUK	0.69	1.07	1.76	2.75
22	CURRENT (NO LOAD)	A	INL	0.250	0.202	0.154	0.125

SERIES 7200



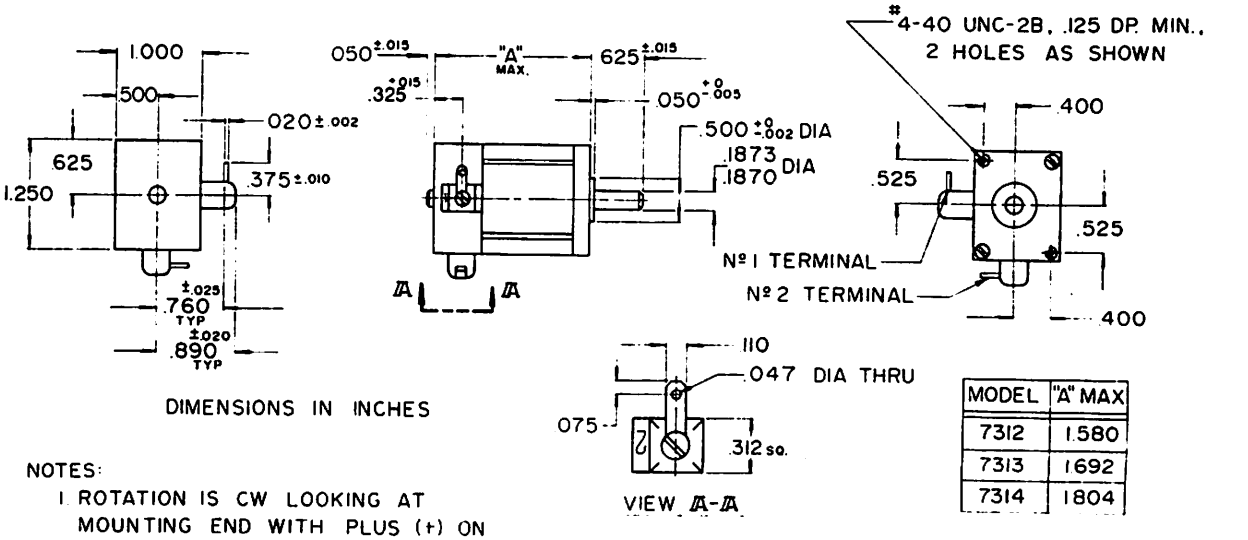
DIMENSIONS IN INCHES

MODEL	"A" MAX
7212	1.580
7213	1.692
7214	1.804

NOTES:

1. ROTATION IS CW LOOKING AT MOUNTING END WITH PLUS (+) ON RED LEAD.
2. LEAD WIRES TO BE 22 GAGE (7×30) PVC WIRE, 12" LONG, STRIPPED AND TINNED 5/32".
3. OPTIONAL MOUNTING HOLE SIZE – M3.5 × 0.6, 3.2 MM DP.
4. OPTIONAL SHAFT DIA. – .1964/1967 ($5^{+.00}_{-.01}$ MM).

SERIES 7300



DIMENSIONS IN INCHES

MODEL	"A" MAX
7312	1.580
7313	1.692
7314	1804

NOTES:

1. ROTATION IS CW LOOKING AT MOUNTING END WITH PLUS (+) ON Nº 1 TERMINAL.
2. TERMINAL WILL MATE WITH AMP, INC OR ETC, INC "110 SERIES" RECEPTACLE.
3. OPTIONAL MOUNTING HOLE SIZE - M3.5 × 0.6, 3.2 MM DP.
4. OPTIONAL SHAFT DIA. – .1964/1967 ($5^{+.00}_{-.01}$ MM).

OTHER PITTMAN PRODUCTS

LO-COG® D-C SERVO MOTORS

SERIES 8000
Nominal 1.2 in. O.D., peak torque (stall) from 5.07 to 11.8 oz.-in.

SERIES 9000
Nominal 1.6 in. O.D., peak torque (stall) from 8.30 to 41.5 oz.-in.

SERIES 13000
Nominal 2 in. O.D., peak torque (stall) from 40.3 to to 128 oz.-in.

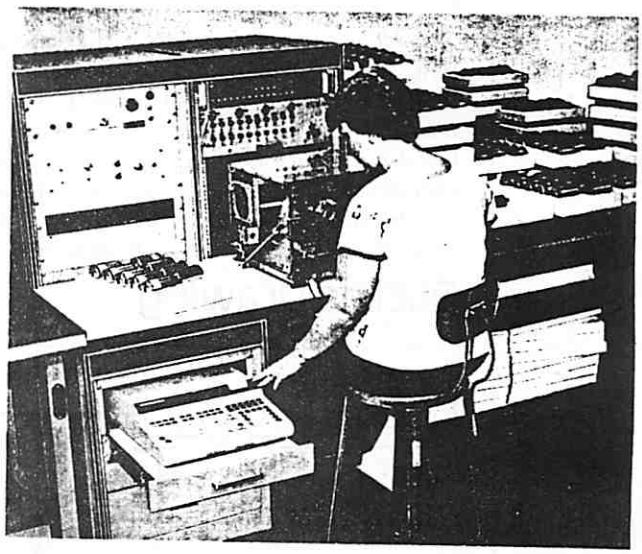
SERIES 14000
Nominal 2.13 in. O.D., peak torque (stall) from 160 to 286 oz.-in.

GEARMOTORS

SERIES GM8700
Nominal 1.38 in O.D. gearbox with standard gear design strength limit of 100 oz.-in.

SERIES GM9400
Nominal 2.00 in O.D. gearbox with standard gear design strength limit of 175 oz.-in.

®elcom PITTMAN® BRUSHLESS D-C SERVOMOTORS



All Pittman products are thoroughly tested just prior to shipment in this computerized testing station (designated PUMA, acronym for Pittman Universal Motor Analyzer). PUMA provides full testing of eight performance parameters and produces a printout with average data and allowable limits. All tapes are stored permanently for any future reference.

TELEPHONE (215) 256-6601
FAX (215) 256-1338

PITMO® PITTMAN®
HARLEYSVILLE, PA 19438-0003 USA
A DIVISION OF PENN ENGINEERING & MANUFACTURING CORP

TELEX 283668
TWX 510-661-8696

Printed in U.S.A.

HEWLETT PACKARD

28 mm DIAMETER TWO AND THREE CHANNEL INCREMENTAL OPTICAL ENCODER KIT

HEDS-5000 SERIES

TECHNICAL DATA JANUARY 1984

Features

- SMALL SIZE — 28 mm DIAMETER
- 100-512 CYCLES/REVOLUTION AVAILABLE
- MANY RESOLUTIONS STANDARD
- LOW INERTIA
- QUICK ASSEMBLY
- 0.25 mm (.010 INCHES) END PLAY ALLOWANCE
- TTL COMPATIBLE DIGITAL OUTPUT
- SINGLE 5V SUPPLY
- WIDE TEMPERATURE RANGE
- INDEX PULSE AVAILABLE

Description

The HEDS-5000 series is a high resolution incremental optical encoder kit emphasizing reliability and ease of assembly. The 28 mm diameter package consists of 3 parts: the encoder body, a metal code wheel, and an emitter end plate. An LED source and lens transmit collimated light from the emitter module through a precision metal code wheel and phase plate into a bifurcated detector lens.

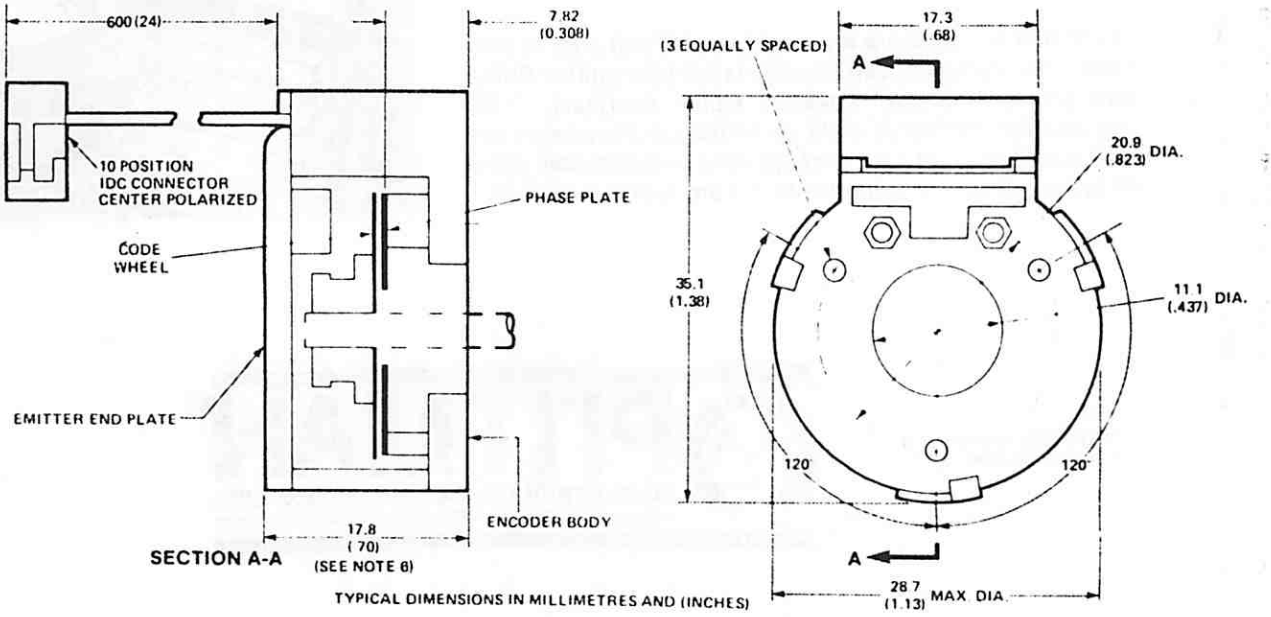
The light is focused onto pairs of closely spaced integrated detectors which output two square wave signals in quadrature and an optional index pulse. Collimated light and a custom photodetector configuration increase long life reliability by reducing sensitivity to shaft end play, shaft eccentricity and LED degradation. The outputs and the 5V supply input of the HEDS-5000 are accessed through a 10 pin connector mounted on a .6 metre ribbon cable.

A standard selection of shaft sizes is available and resolutions between 100 and 512 cycles per revolution are available as options. The part number for the standard 2 channel kit is HEDS-5000, while that for the 3 channel device, with index pulse, is HEDS-5010. See Ordering Information for more details.

Applications

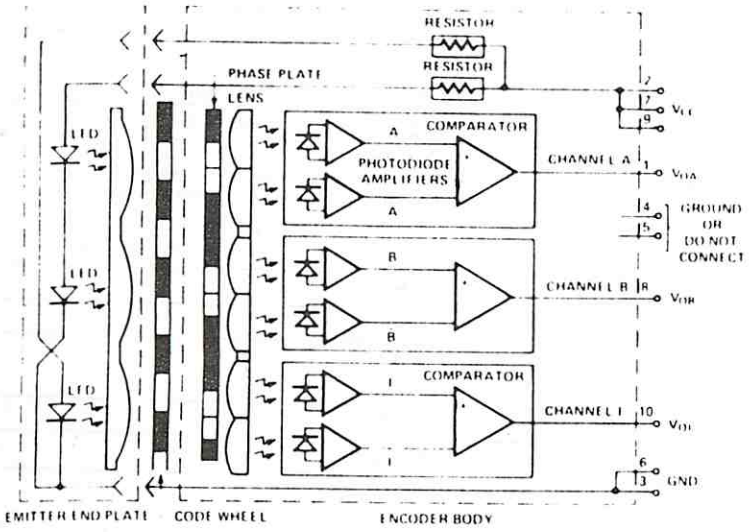
Printers, Plotters, Tape Drives, Positioning Tables, Automatic Handlers, Robots, and any other servo loop where a small high performance encoder is required.

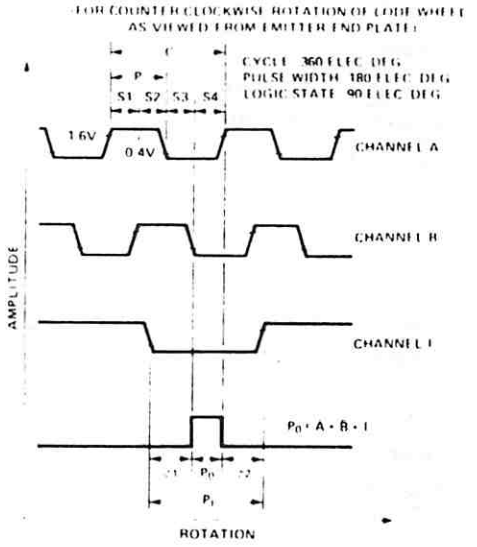
Outline Drawing



TYPICAL DIMENSIONS IN MILLIMETRES AND (INCHES)

Block Diagram and Output Waveforms





Theory of Operation

The incremental shaft encoder operates by translating the rotation of a shaft into interruptions of a light beam which are then output as electrical pulses.

In the HEDS-5XXX the light source is a Light Emitting Diode collimated by a molded lens into a parallel beam of light. The Emitter End Plate contains two or three similar light sources, one for each channel.

The standard Code Wheel is a metal disc which has N equally spaced apertures around its circumference. A matching pattern of apertures is positioned on the stationary phase plate. The light beam is transmitted only when the apertures in the code wheel and the apertures in the phase plate line up; therefore, during a complete shaft revolution, there will be N alternating light and dark periods. A molded lens beneath the phase plate aperture collects the modulated light into a silicon detector.

The Encoder Body contains the phase plate and the detection elements for two or three channels. Each channel consists of an integrated circuit with two photodiodes and amplifiers, a comparator, and output circuitry.

The apertures for the two photodiodes are positioned so that a light period on one detector corresponds to a dark period on the other ("push-pull"). The photodiode signals are amplified and fed to the comparator whose output changes state when the difference of the two photocurrents changes sign. The second channel has a similar configuration but the location of its aperture pair provides an output which is in quadrature to the first channel (phase difference of 90°). Direction of rotation is determined by observing which of the channels is the leading waveform. The outputs are TTL logic level signals.

The optional index channel is similar in optical and electrical configuration to the A and B channels previously described. An index pulse of typically 1 cycle width is generated for each rotation of the code wheel. Using the recommended logic interface, a unique logic state (P_0) can be identified if such accuracy is required.

The three part kit is assembled by attaching the Encoder Body to the mounting surface using three screws. The Code Wheel is set to the correct gap and secured to the shaft. Snapping the cover (Emitter End Plate) on the body completes the assembly. The only adjustment necessary is the encoder centering relative to the shaft. This optimizes quadrature and the optional index pulse outputs.

Index Pulse Considerations

The motion sensing application and encoder interface circuitry will determine the necessary phase relationship of the index pulse to the main data tracks. A unique shaft position can be identified by using the index pulse output only or by logically relating the index pulse to the A and B data channels. The HEDS-5010 allows some adjustment of the index pulse position with respect to the main data channels. The position is easily adjusted during the assembly process as illustrated in the assembly procedures.

Definitions

Electrical degrees:

1 shaft rotation = 360 angular degrees
N electrical cycles

1 cycle = 360 electrical degrees

Position Error:

The angular difference between the actual shaft position and its position as calculated by counting the encoder's cycles.

Cycle Error:

An indication of cycle uniformity. The difference between an observed shaft angle which gives rise to one electrical cycle, and the nominal angular increment of 1/N of a revolution.

Phase:

The angle between the center of Pulse A and the center of Pulse B.

Index Phase:

For counter clockwise rotation as illustrated above, the Index Phase is defined as:

$$\Phi_1 = \frac{\phi_1 - \phi_2}{2}.$$

ϕ_1 is the angle, in electrical degrees between the falling edge of I and falling edge of B. ϕ_2 is the angle, in electrical degrees, between the rising edge of A and the rising edge of I.

Index Phase Error:

The Index Phase Error ($\Delta\Phi_1$) describes the change in the Index Pulse position after assembly with respect to the A and B channels over the recommended operating conditions.

Absolute Maximum Ratings

Parameter	Symbol	Min.	Max.	Units	Notes
Storage Temperature	T_S	-55	100	°Celsius	
Operating Temperature	T_A	-55	100	°Celsius	See Note 1
Vibration			20	g	See Note 1
Shaft Axial Play			.50 (20)	mm(1inch/1000) TIR	
Shaft Eccentricity Plus Radial Play			.1 (4)	mm(1inch/1000) TIR	Movement should be limited even under shock conditions.
Supply Voltage	V_{CC}	-0.5	7	Volts	
Output Voltage	V_O	-0.5	V_{CC}	Volts	
Output Current per Channel	I_O	-1	5	mA	
Velocity			30,000	R.P.M.	
Acceleration	α		250,000	Rad. Sec^{-2}	

Recommended Operating Conditions

Parameter	Symbol	Min.	Max.	Units	Notes
Temperature	T	-20	85	°Celsius	Non-condensing atmos.
Supply Voltage	V_{CC}	4.5	5.5	Volt	Ripple < $100mV_{p-p}$
Code Wheel Gap			1.1 (45)	mm (inch/1000)	Nominal gap = 0.63 mm (.025 in.) when shaft is at minimum gap position.
Shaft Perpendicularity Plus Axial Play			0.25 (10)	mm (inch/1000) TIR	
Shaft Eccentricity Plus Radial Play			0.04 (1.5)	mm (inch/1000) TIR	10 mm (0.4 inch) from mounting surface.
Load Capacitance	C_L		100	pF	

Encoding Characteristics

The specifications below apply within the recommended operating conditions and reflect performance at 500 cycles per revolution (N = 500). Some encoding characteristics improve with decreasing cycles (N). Consult Application Note 1011 or factory for additional details.

Parameter	Symbol	Min.	Typ.	Max.	Units	Notes (See Definitions)
Position Error - Worst Error Full Rotation	$\Delta\theta$		10	40	Minutes of Arc	1 Cycle = 43.2 Minutes See Figure 5.
Cycle Error - Worst Error Full Rotation	ΔC		3	5.5	Electrical deg.	
Max. Count Frequency	f_{MAX}	130,000	200,000		Hertz	f = Velocity (RPM) x N/60
Pulse Width Error - Worst Error Full Rotation	ΔP		16		Electrical deg.	T = 25° C, f = 8 KHz See Note 2
Phase Sensitivity to Eccentricity			520 (13)		Elec. deg./mm (Elec. deg./mil)	mil = inch/1000
Phase Sensitivity to Axial Play			20 (.5)		Elec. deg./mm (Elec. deg./mil)	mil = inch/1000
Logic State Width Error - Worst Error Full Rotation	ΔS		25		Electrical deg.	T = 25° C, f = 8 KHz See Note 2
Index Pulse Width	P_I		360		Electrical deg.	T = 25° C, f = 8 KHz See Note 3
Index Phase Error	$\Delta\Phi_I$		0	17	Electrical deg.	See Notes 4, 5
Index Pulse Phase Adjustment Range		±70	±130		Electrical deg.	See Note 5

Mechanical Characteristics

Parameter	Symbol	Dimension	Tolerance	Units	Notes
Outline Dimensions		See Mech. Dwg.			
Code Wheel Available to Fit the Following Standard Shaft Diameters		2 4 3 5	+.000 −.015	mm	
		5/32	+.0002 −.0005	inches	
		1/8 3/16 1/4	+.0000 −.0007	inches	
Moment of Inertia	J	0.4 (6 x 10^{-6})		gcm^2 ($oz\text{-}in\text{-}s^2$)	
Required Shaft Length		12.8 (.50)	±0.5 (±0.02)	mm (inches)	See Figure 10. Shaft in minimum length position.
Bolt Circle		20.9 (.823)	±0.13 (±.005)	mm (inches)	See Figure 10.
Mounting Screw Size		1.6 x 0.35 x 5 mm DIN 84 or 0-80 x 3/16 Binding Head		mm inches	

Electrical Characteristics When operating within the recommended operating range.

Electrical Characteristics over Recommended Operating Range Typical at 25° C.

Parameter	Symbol	Min.	Typ.	Max.	Units	Notes
Supply Current	I_{CC}		21	40	mA	HEDS-5000 (2 Channel)
			36	60		HEDS-5010 (3 Channel)
High Level Output Voltage	V_{OH}	2.4			V	I_{OH} = -40μA Max.
Low Level Output Voltage	V_{OL}			0.4	V	I_{OL} = 3.2 mA
Rise Time	t_r		0.5		μs	C_L = 25 pF, R_L = 11K Pull-up
Fall Time	t_f		0.2			See Note 7
Cable Capacitance	C_{CO}		12		pF/metres	Output Lead to Ground

NOTES:

1. The structural parts of the HEDS-5000 have been tested to 20g and up to 500 Hz. For use outside this range, operation may be limited at low frequencies by high displacement by cable fatigue and at high frequencies by code wheel resonances. Resonant frequency depends on code wheel material and number of counts per revolution. For temperatures below -20°C the ribbon cable becomes brittle and sensitive to displacements. Maximum operating and storage temperature includes the surface area of the encoder mounting. Consult factory for further information. See Application Note 1011.
2. In a properly assembled lot 99% of the units, when run at 25° C and 8 KHz, should exhibit a pulse width error less than 35 electrical degrees, and a state width error less than 45 electrical degrees. To calculate errors at other speeds and temperatures add the values specified in Figures 1 or 2 to the typical values specified under encoding characteristics or to the maximum 99% values specified in this note.
3. In a properly assembled lot, 99% of the units when run at 25° C and 8 KHz should exhibit an index pulse width greater than 260 electrical degrees and less than 460 electrical degrees. To calculate index pulse widths at other speeds and temperatures add the values specified in Figures 3 or 4 to the typical 360° pulse width or to the maximum 99% values specified in this note.
4. After adjusting index phase at assembly, the index phase error specification $\Delta\phi_1$ indicates the expected shift in index pulse position with respect to channels A and B over the range of recommended operating conditions and up to 50 KHz.
5. When the index pulse is centered on the low-low states of channels A and B as shown on page 2, a unique P_O can be defined once per revolution within the recommended operating conditions and up to 25 KHz. Figure 6 shows how P_O can be derived from A, B, and I outputs. The adjustment range indicates how far from the center of the low-low state that the center of the index pulse may be adjusted.
6. The typical length of an assembled HEDS-5000 encoder is 17.8 (.70 inch). However, it is recommended that room be left to accomodate a length of 21.6 (.85 inch). Future developments may result in an enhanced version of the HEDS-5000 encoder that is slightly longer.
7. The rise time is primarily a fuction of the RC time constant of R_L and C_L. A faster rise time can be achieved with either a lower value of R_L or C_L. Care must be observed not to exceed the recommended value of I_{OL} under worst case conditions.

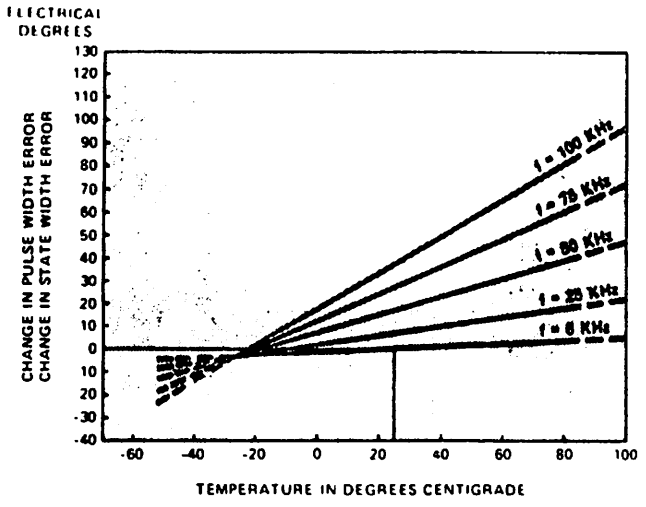


Figure 1. Typical Change in Pulse Width Error or in State Width Error due to Speed and Temperature

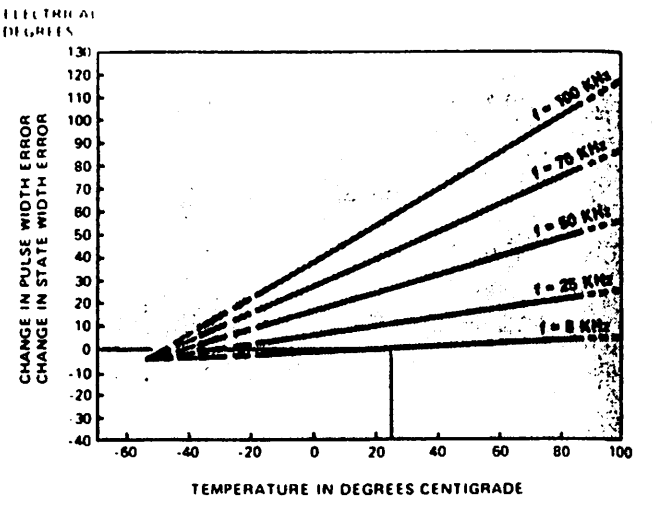


Figure 2. Maximum Change in Pulse Width Error or in State Width Error Due to Speed and Temperature

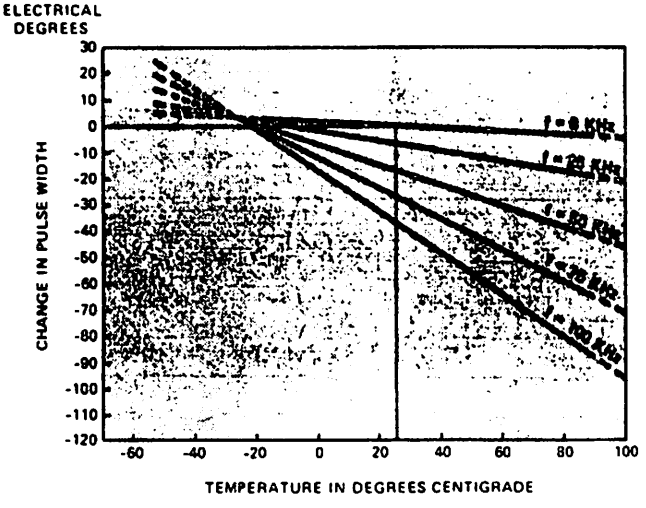


Figure 3. Typical Change in Index Pulse Width Due to Speed and Temperature

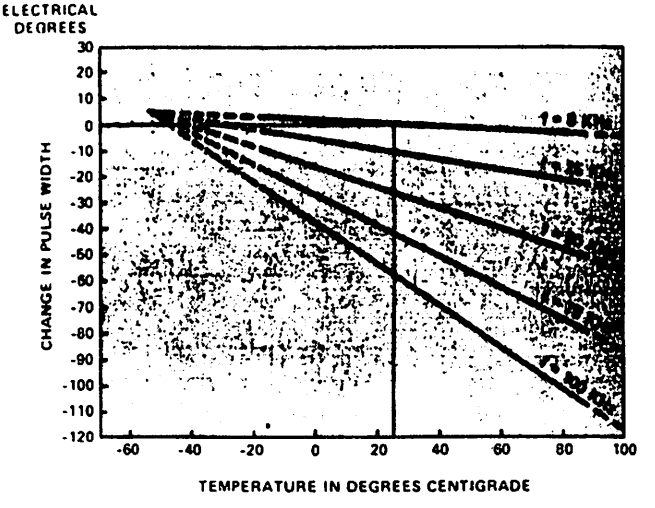


Figure 4. Maximum Change in Index Pulse Width Due to Speed and Temperature

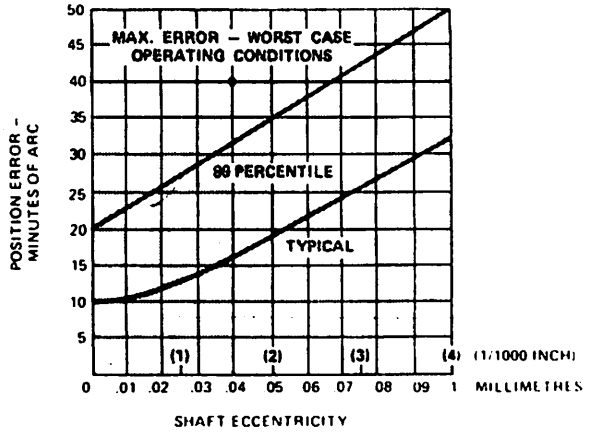
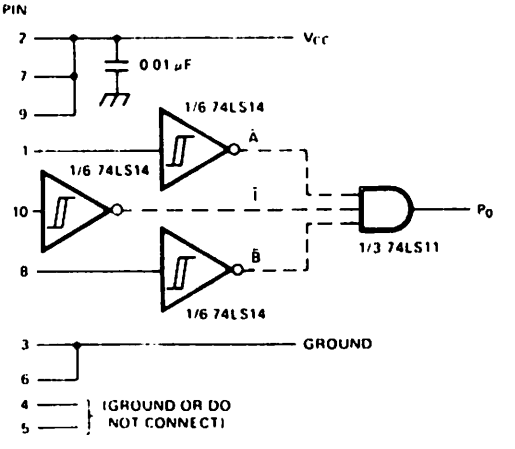


Figure 5. Position Error vs. Shaft Eccentricity



DASHED LINES REPRESENT AN OPTIONAL INDEX SUMMING CIRCUIT
STANDARD 74 SERIES COULD ALSO BE USED TO IMPLEMENT THIS CIRCUIT

Figure 6. Recommended Interface Circuit

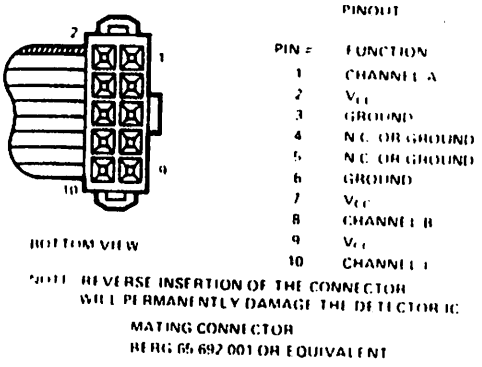


Figure 7. Connector Specifications

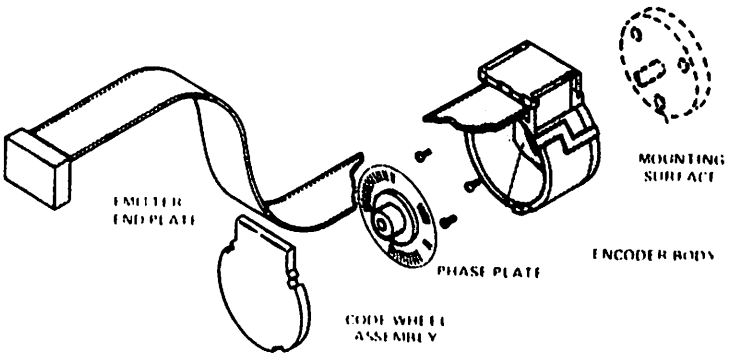


Figure 8. HEDS-5000 Series Encoder Kit

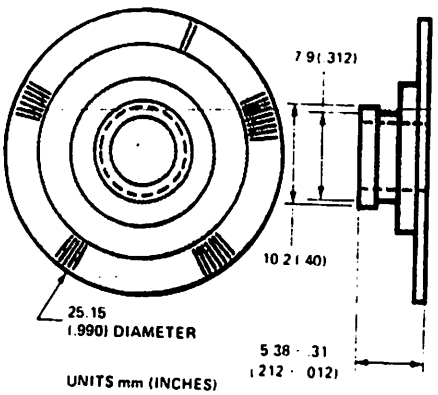


Figure 9. Code Wheel

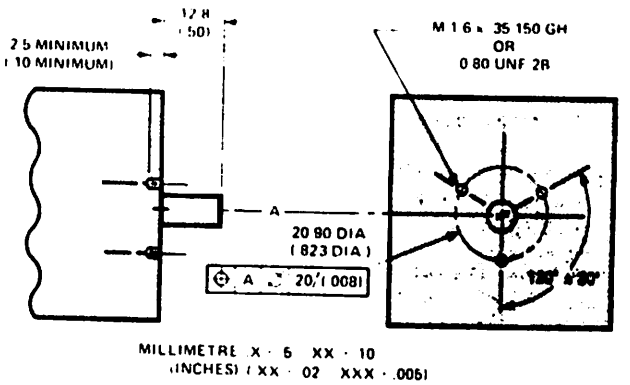
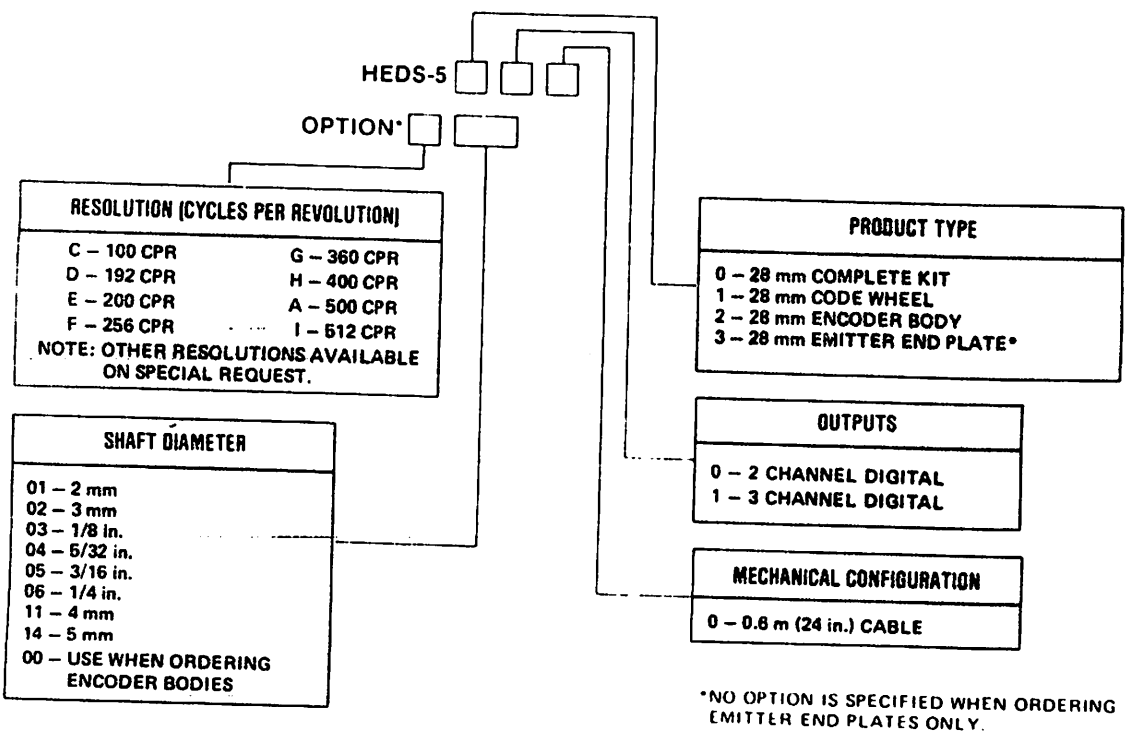
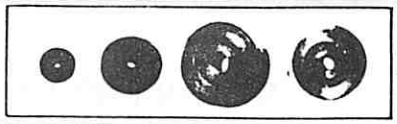


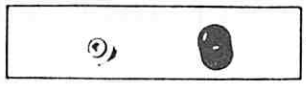
Figure 10. Mounting Requirements

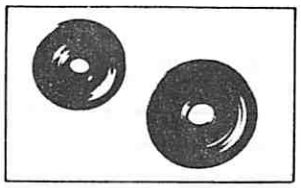
Ordering Information

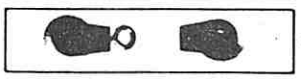


*NO OPTION IS SPECIFIED WHEN ORDERING EMITTER END PLATES ONLY.









CABLE CONSTRUCTION

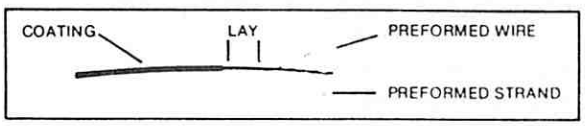
Strand, wire rope or cable, is a uniform helical arrangement of wires concentrically stranded together for a variety of operating conditions. These constructions have different properties which are designed for specific applications.

The STRAND constructions have coarser wires, better abrasion resistance and poor flexibility. In small sizes, they are useful as fishing lines and leaders, light guying and lanyards; in larger sizes, as guy strand, standard rigging on boats and other applications where flexibility of the cable is not essential.

The CABLE constructions have more wires and are consequently more flexible with less abrasion resistance. This type of construction has applications in automotive, appliances, boating, instrumentation, aviation, agriculture, hobbies, medicine, recreation, etc.

BASIC CONSTRUCTIONS

STRAND Two or more wires laid together — the construction is normally referred to as 1x the number of wires involved; i.e., 1x7, 1x19, etc.



1x7

This is the basic strand construction which is used in different combinations to construct another strand or other cables. It is somewhat stiff in larger diameters.

1x19

Stronger than 1x7 and more symmetrical in configuration, affording still better properties than 1x7.

CABLE Three or more strands laid together — the construction is normally referred to as number of strands x number of wires in each strand (for example — 7x7 — see below).

3x7

A combination of three 1x7 strands. Very flexible in small diameters.

7x7

A combination of seven 1x7 strands, affording abrasion resistance and flexibility through a wide range of diameters.

7x19

A combination of seven 1x19 strands producing a fine combination of strength and flexibility in a wide range of sizes.

CABLE APPLICATIONS

Cable and cable assemblies are designed to be a reliable and economical means of transmitting motion or controlling an item mechanically. Typically, their usage falls into two broad applications listed below:

ACTUATING DEVICES

- ADJUSTMENT
- ALIGNMENT
- BRAKING
- CONNECTING
- CONTROLLING
- COUNTERBALANCING
- CYCLING
- DRIVING
- HOISTING
- TRAVERSING

LANYARDS

- CAPTURING
- CONNECTING
- HANGING
- GROUNDING
- RESTRAINING
- SECURING
- SUPPORTING
- STOPPING
- TENSIONING
- TRIPPING

USED IN THESE TYPICAL TYPES OF PRODUCTS

AIRCRAFT • APPLIANCES • AUTOMOTIVE • BICYCLES • BOATS • CARTS • COMPUTERS • COPY MACHINES • DENTAL EQUIPMENT • DRAFTING EQUIPMENT • FARM EQUIPMENT • FLOOR SWEEPERS • HOSPITAL BEDS • LAWN EQUIPMENT • MATERIAL HANDLING • MEDICAL EQUIPMENT • MODELS • OFFICE EQUIPMENT • PLOTTERS • POWER TOOLS • PRINTERS • PUMPS • SCALES • SKI EQUIPMENT • TAPE DRIVES • THROTTLES • TIMERS • TOYS AND GAMES • TYPEWRITERS • VENDING MACHINES • WASHING MACHINES • X-RAY EQUIPMENT

STAINLESS STEEL CABLE
TEFLON COATED (FEP)

PART NO.	UNCOATED NOMINAL DIA. IN.	MM	CONSTRUCTION	MINIMUM BREAKING STRENGTH LBS.	KG.	OUTSIDE DIAMETER OF COATING IN.	MM	APPROX. WEIGHT PER 100 FT. LBS.	KG.
2014-ST	.014	.36	7x7	26	12	.021	.52	.04	.02
2018-ST	.018	.46	7x7	40	18	.026	.66	.06	.03
2027-ST	.027	.69	7x7	90	41	.036	.94	.15	.07
2031-ST	.031	.79	7x7	120	54	.040	.98	.22	.10
2032-ST	.031	.79	7x19	120	54	.040	.98	.22	.10
2036-ST	.036	.92	7x7	160	73	.049	1.24	.26	.12
2037-ST	.036	.92	7x19	160	73	.049	1.24	.26	.12
2048-ST	3/64	1.19	7x7	270	122	.062	1.59	.49	.22
2050-ST	3/64	1.19	7x19	270	122	.062	1.59	.49	.22
2064-ST	1/16	1.59	7x7	480	217	.084	2.15	.85	.40
2065-ST	1/16	1.59	7x19	480	217	.084	2.15	.85	.40

Other diameters and constructions are available on request. "Teflon" fluorocarbon resins, due to their unique combinations of physical, electrical and chemical properties, have found application in nearly every field of modern industrial, technological and scientific endeavor.
Combined with almost total chemical inertness, dielectric stability, and nonflammability, the heat resistance of "Teflon" gives a versatility that is unmatched by other engineering plastics.

"TEFLON" FEP OFFERS THE FOLLOWING EXCELLENT COMBINATION OF PROPERTIES:

- **Nonaging characteristics**
- **Low coefficient of friction**
- **Chemical inertness**
- **Exceptional dielectric properties**
- **Toughness and flexibility**
- **Continuous service temperature — 400°F.**
- **Heat resistance**
- **Nonstick characteristics**
- **Negligible moisture absorption (less than .01%)**
- **Weather resistance**
- **Nonflammable**

SWAGED CABLES

For applications requiring increased wear resistance and lower friction in a given size cable, particularly push-pull types, swaging provides a good answer. The average diameter decrease by swaging without appreciable loss in performance is approximately 10%.

In swaging, the outer wires are flattened to provide greater wear area and smoother surface. The breaking-strength-to-cable-diameter ratio remains constant, without appreciable loss of flexibility.

CUSTOM STRANDING

Our stranding equipment is available for manufacture of cables using your material or a material of your specifications. We have successfully manufactured cables from very unusual materials, some of which were also plastic coated.

CUSTOM PLASTIC COATINGS

SAVA's modern extrusion equipment and expertise lends itself very well to extruding other plastic materials such as polypropylene, polyethylene, polyurethane and many types of nylon. Also we will be pleased to quote on coating more than one cable in a common jacket either in a flat or round shape. Colored plastic is possible in most plastics.

ELECTRIC CUTTING

Electric cutting of uncoated cable is a process by which the ends of the cable are sealed (fused) to prevent fraying and to allow easy application of a fitting. It is particularly desirable to specify electric cutting where only one end of an assembly has a fitting. SAVA is equipped with automatic electric cutting machinery which is also available to cut your bulk cable to desired lengths. Unless you are similarly equipped you will find this service particularly advantageous.

GUIDELINES FOR CABLE SELECTION

TO OPTIMIZE LIFE

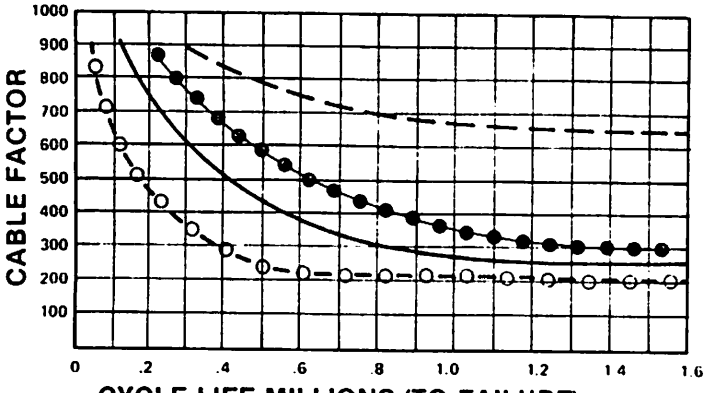
1. Select a cable that has a minimum breaking strength of 10 times the working load.

2. Select a pulley which has a root diameter of 40 times the bare cable diameter (diameter without jacket) or see recommended pulley diameters on page 7.

AFTER you have determined the tension on cable (max.) and selected the cable diameter and pulley for your system, apply the numbers to the formula below to determine the cable load factor. Read anticipated cycle life from chart.

NOTE: A system frequently cannot be designed for optimum conditions. The formula and chart show that a trade off can be made between cable size, pulley size and life.

CABLE FATIGUE LIFE CHART



NOTE: The numbers on this chart were determined under laboratory conditions and are intended to serve as a guide. Since there is considerable variation in application, we recommend actual life tests for your intended use.

FORMULA

(see example)

$$\text{Cable Factor} = \frac{\text{Tension on Cable}}{\text{Bare Cable dia. x pulley root dia.}}$$

7x7 Construction Nylon Coated	Curve for part nos. 2014-SN, 2018-SN, 2027-SN, 2031-SN, 2036-SN, 2048-SN, 3048-GN, 2064-SN, 3064-GN, 3094-GN, 2094-SN.
3x7 Construction Nylon Coated	Curve for parts nos. 2009-SN, 2012-SN, 2030-SN.
7x19 or 7x49 Construction Nylon Coated	Curve for part nos. 2019-SN, 2024-SN, 2032-SN, 2037-SN, 2038-SN, 2046-SN, 2047-SN, 2050-SN, 2054-SN, 2065-SN, 2095-SN, 3065-GN, 3095-GN.
7x7 Construction Uncoated	Curve for parts nos. 2018, 2027, 2031, 2036, 2046, 2048, 2064, 2094, 3048, 3064, 3094.

Example:

Determine the cable size, pulley diameter and cycle life for a counterweight system working with a load of 15 lbs. Anticipated life of cable should be 2,000,000 cycles.

Step 1 Select a cable from page 4 which has a breaking strength of 10 times the load. Cables 2036, 2037 and 2038, .036" diameter, qualify.

Step 2 Multiply cable diameter .036" by 40 for pulley groove diameter.
.036" x 40 = 1.44"

Step 3 Cable Factor = $\frac{15}{.036 \times 1.44} = 289$

Applying this factor to the Cable Fatigue Life Chart, it is seen that a 7 x 7 bare cable will have an expected cycle life of only 400,000 cycles, which is not sufficient. When a nylon coated cable (2036-SN) is selected the Cable Factor line will not intersect the curve on the cable for a 7 x 7 or 7 x 19 coated.
The 7 x 7 coated cable (2036-SN) is less expensive and would be the logical choice.

DATA ON STRETCH OF CABLE AND STRAND

Two kinds of stretch occur in cable — constructional stretch and elastic stretch. They are due to two different causes.

Constructional Stretch

When strand and cable are made, the load at the closing head is light. Therefore there are small clearances between the wires and strands, and between the strand and the core. The application of initial load causes wires and strands to seat properly and a slight overall elongation of the strand or cable accompanies this action.

The amount of constructional stretch is not constant for all cables — it depends on such variables as type of construction, length of lay, and other factors, including the load applied.

Elastic Stretch

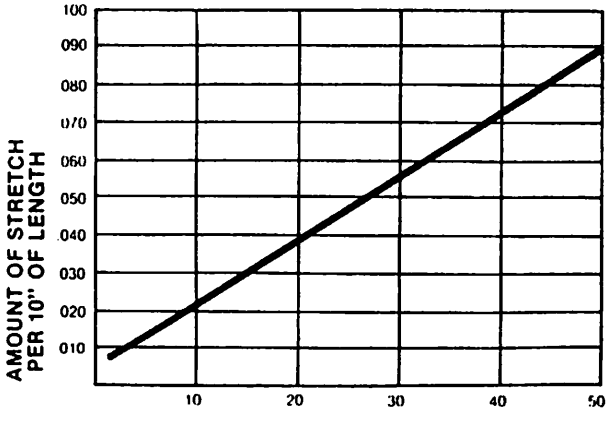
Elastic stretch is the actual elongation of the wires of a strand or a cable. This is caused by the application of a load, up to the yield point of the metal, and the stretch is approximately proportional to the load applied.

When the load is released, strand or cable subjected to elastic stretch returns to its approximate original length, providing the stretch has not reached the yield point of the metal.

Removal of Stretch

Where the elimination of as much stretch as possible is important, the cables or assemblies can be proof loaded to remove most of the constructional stretch. For assemblies, this process also verifies the holding power of the terminals. Proof loading is usually done by applying a 60% load to the cable or assemblies. This load is based on the minimum breaking strength of the cable or fittings, whichever is lower. Handling the cable as little as possible after prestretching helps eliminate putting constructional stretch back in.

ELASTIC STRETCH CHARACTERISTICS OF SAVA CABLE



NOTES: For construction of 7x7, 7x19, 8x19, 7x49 & 7x49 the chart represents close approximates

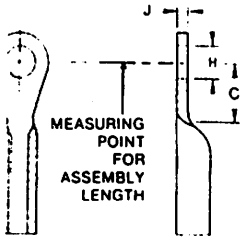
AVAILABLE LOOSE OR FACTORY APPLIED

ın equipment are
ver, we offer most
mped with pliers,
ugh it can be an
ty convenient and
of 25 primarily for

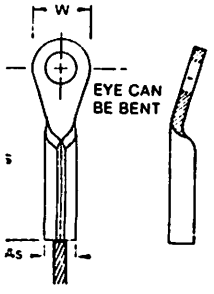
THREADED PLUG (See Notes)

PART NO.			DIMENSIONS (AFTER SWAGING)				
BRASS	STAINLESS STEEL	CARBON STEEL PLATED	BARE CABLE DIA.	THREAD SIZE T	THREAD LGTH B	FLATS F	Ls[7]
455-B	455-S	—	.010 - .048	5 - 40	.500	.105	1.00
457-B	457-S	—	.010 - .048	5 - 40	1.000	.105	1.50
458-B	458-S	—	.010 - .048	6 - 32	.750	.125	1.25
460-B	460-S	—	.028 - .048	8 - 32	.500	.140	1.00
462-B	462-S	—	.028 - .048	8 - 32	1.000	.140	1.50
465-B	465-S	—	3/64 - 1/16	10 - 24	.500	.156	1.12
467-B	467-S	—	3/64 - 1/16	10 - 24	1.000	.156	1.62
468-B	468-S	468-P	3/64 - 1/16	10 - 32	1.250	.156	1.87
470-B	470-S	470-P	1/16 - 3/32	1/4 - 20	.750	.218	1.40
—	475-S-.750	475-P-.750	3/32 - 1/8	1/4 - 20	.750	.218	1.56
—	475-S-1.250	475-P-1.250	3/32 - 1/8	1/4 - 20	1.250	.218	2.06
—	475-S-1.500	475-P-1.500[3]	3/32 - 1/8	1/4 - 20	1.500	.218	2.31
—	476-S-1.375	476-P-1.375	1/16 - 3/32 - 1/8	1/4 - 28	1.375	.218	2.19
—	477-S-1.000	477-P-1.000	5/32	5/16 - 18	1.000	.281	1.94
—	477-S-2.000	477-P-2.000[3]	5/32	5/16 - 18	2.000	.281	2.94
—	478-S-1.500	478-P-1.500	5/32	5/16 - 24	1.500	.281	2.44
—	480-S-1.250	480-P-1.250	3/32 - 1/8 - 3/16	3/8 - 16	1.250	.312	2.25
—	480-S-2.250	480-P-2.250[3]	3/32 - 1/8 - 3/16	3/8 - 16	2.250	.312	3.25
—	481-S-1.500	481-P-1.500	1/8 - 5/32 - 3/16	3/8 - 24	1.500	.312	2.50
—	482-S	482-P	1/4	1/2 - 13	1.500	.437	2.81
—	483-S	483-P	1/4	1/2 - 20	2.000	.437	3.31
—	485-S	485-P	5/16	5/8 - 11	1.750	.562	3.38
—	490-S	490-P	3/8	3/4 - 10	2.000	.687	4.00
—	495-S	495-P	7/16 - 1/2	1 - 8	2.500	.937	5.50

MEASURING POINT FOR ASSEMBLY LENGTH



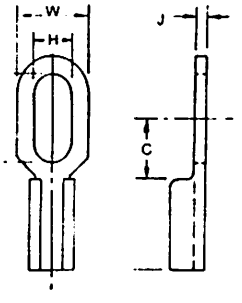
EFORE CRIMPING



AFTER CRIMPING

NOTES:

1. If you order loose the cab be given so hole can be cable you int
2. Unless stated right hand.
3. Available threads to fi turnbuckles See Page 13.
4. Rolled threa available if re
5. Hand tool #T prototype wo 457-B; 458-B 465-B, 467-B
6. Threaded plu rated strengt cables.
7. After swage fore swage th

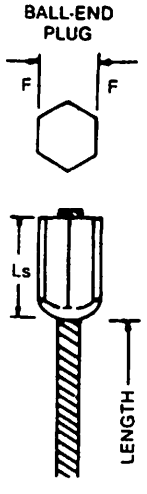


PART NO. 315-PSL-218
PART NO. 315-SSL-218

EFORE CRIMPING

This type can be de-
eloped for other cable
sizes also.

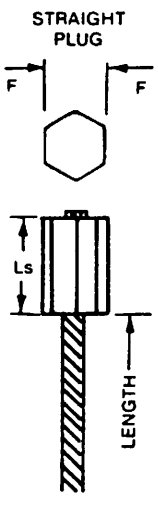
BALL-END PLUG
FOR CABLE SIZES .010 - 3/32



PART NO.	BARE CABLE DIA.	Ls	F[7]	STANDARD MATERIAL
401-1* 401-2 401-3 401-4	.010 - .027 .028 - .037 .038 - .046 3/64	.125	.105	Brass
403-1 403-2 403-3	.010 - .027 .028 - .037 .038 - .046	.250	.105	Brass
405-1 405-2 405-3 405-4	.010 - .027 .028 - .037 .038 - .048 .049 - .065	.395	1/8	Brass
410-1 410-2	3/64 1/16	.500	5/32	Brass
415	3/32	.500	7/32	Brass

*401 fittings hold 50% of cable strength.

STRAIGHT PLUG
FOR CABLE SIZES 1/8 - 3/8



PART NO.	BARE CABLE DIA.	Ls	F[7]	STA MA
420-P 420-S	1/8	.625	.312	Plat Stain
422-P 422-S	5/32	.690	.437	Plat Stain
425-P 425-S	3/16	.750	.437	Plat Stain
430-P 430-S	1/4	1.000	.562	Plat Stain
435-P 435-S	5/16	1.250	.687	Plat Stain
440-P 440-S	3/8	1.560	.875	Plat Stain
445-P 445-S	7/16 - 1/2	2.250	1.00	Plat Stain

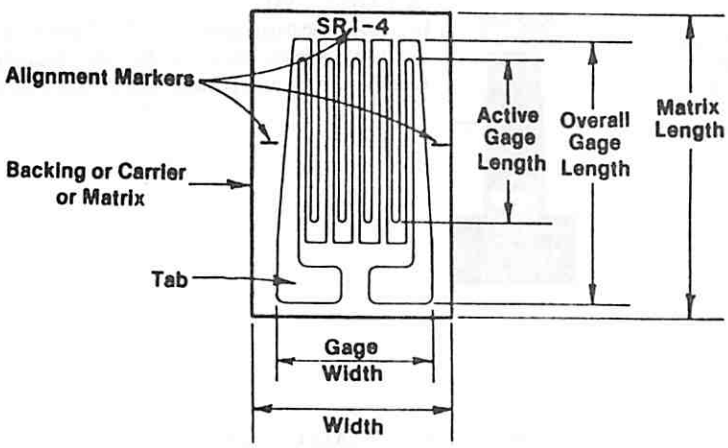
UFACTURE YOUR CABLE ASSEMBLIES

SR-4 APPLICATION INSTRUCTIONS

STRAIN GAGE WORKSHOP NOTES

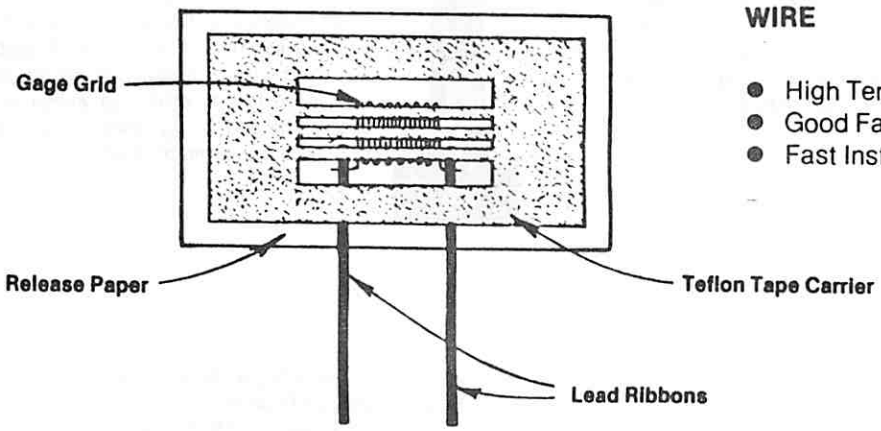
BLH Electronics, 75 Shawmut Road, Canton, Massachusetts 02021
Tel: (617) 821-2000 • TWX: 710-324-0683 • Telex: 92-3413 • Cable: BALIMA

Construction of a Strain Gage



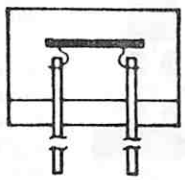
FOIL

- Low Cost
- Multi-Configurations
- Options Available
- Choice of Temperature Compensation
- Transverse Sensitivity Control
- Close Resistance Tolerance
- Low Creep
- Heat Dissipation
- Flexible



WIRE

- High Temperature Operation
- Good Fatigue Characteristics
- Fast Installation

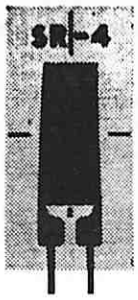


SEMICONDUCTOR

- High Gage Factor
- High Output Voltage
- Small Size
- Low Cost Readout Required

OPTIONS (Standard)

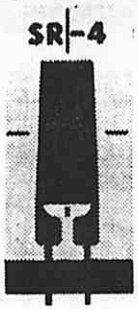
L — LEADS ATTACHED



GAGE TYPES: **FAE, FAQ**

This option adds nickel-clad copper ribbon leads soldered to the FAE gage tabs (nickel-plated beryllium-copper for maximum fatigue life to FDE gages). Lead dimensions are 0.203 mm (0.008 in.) or 0.305 mm (0.012 in.) wide by 0.635 mm (0.0025 in.) thick by 25.40 mm (1.000 in.) long. The leads are soldered to the gage tab with 218C (425F) tin-silver solder, usable to 204C (400F). Gage life of copper is much less than that of the gage sensing element. Routing and dressing the leads to keep them out of the strain area will usually overcome this deficiency.

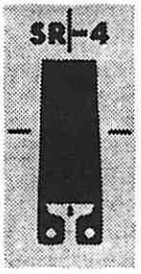
EL — ENCAPSULATED WITH LEADS



GAGE TYPES: **FAE, FSE**

This option combines options E and L providing a gage with both ribbon leads and a 0.0254 mm (0.001 in.) layer of polyimide film.

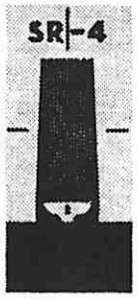
S — SOLDER DOTS



GAGE TYPES: **FAE, FAB, FAQ**

This option simplifies lead attachment by providing an accurately positioned 0.254 mm (0.010 in.), 0.381 mm (0.015 in.), or 0.762 mm (0.030 in.) diameter solder dot on the tab. This allows lead orientation in any position. Solder dot material is tin-silver alloy with a melting point of 218C (425F). The gage retains its flexibility and fatigue life is unaffected.

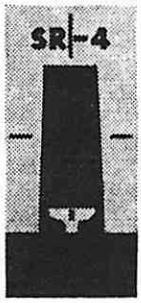
ES — ENCAPSULATED WITH SOLDER DOTS



GAGE TYPES: **FAE**

This option combines options E and S providing a gage with both polyimide film and solder dots. Encapsulation is removed in the area of solder dots for lead attachment.

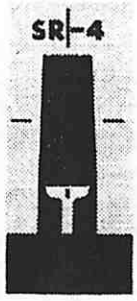
E — ENCAPSULATION ONLY



GAGE TYPES: **FAE, FSE**

This option consists of a 0.0254 mm (0.001 in.) layer of polyimide film that covers all of the gage except that portion of the tab necessary for lead attachment. The polyimide overlay provides protection of the sensing element during installation handling, and promotes better long term stability with the foil grid protected from airborne contaminants or fingerprints.

EG — ENCAPSULATED, GOLD COATED TABS



GAGE TYPES: **FAE**

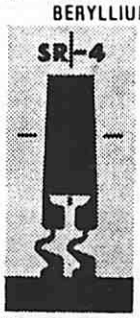
Stress analyzed grid configurations in the FAE Series combined with the superb fatigue resistance of gold in this option to produce an all-new general purpose, fully encapsulated gage. This option eliminates the need for terminal strips and interconnecting leads, by providing oversize gold-coated tabs to which lead wires can be soldered directly. The gold-coating prevents corrosion during curing process for high-temperature adhesives.

WL — WIRE LEADS

GAGE TYPES: **FAE, FAB** (on special order)

This option is the same as option L, except ribbon leads are replaced by 8-mil diameter nickel-plated copper wire. Wire leads are also attached with 218C (425F) solder with the gage useful to 204C (400F). Other wire sizes and types can be substituted on special order.

ET — ENCAPSULATED W/INTEGRAL NICKEL-PLATED BERYLLIUM COPPER TERMINALS



GAGE TYPES: **FAE, FSE, FSM**

The integral beryllium-copper, nickel-plated, etched terminals are mounted on an extension of the polyimide backing, which is then covered with a 0.0254 mm (0.001 in.) layer of polyimide film that encapsulates all but the lead wire attachment area of the terminals. This increases the environmental integrity of the gage, at the same time retaining the flexibility of the gage backing. There is only one solder joint at the gage tab made with 218C (425F) tin-silver alloy solder with a useful paper limit 204C (400F). Only a slight reduction in rated fatigue life occurs. This option is excellent for general purpose gages because installation time is reduced and the more rugged terminal will accept longer lead wires.

TECHNICAL DATA

1. Hooke's Law (Young's Modulus)

$$E = \frac{\sigma}{\epsilon}$$

Where: E is modulus of elasticity expressed in PSI
σ is stress in PSI
ϵ is strain in μ"/"

For many common materials there is a constant ratio between Stress and Strain.

2.

$$G.F. = \frac{\Delta R/R}{\Delta L/L}$$

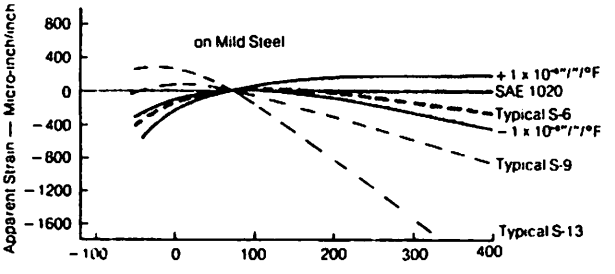
R = Unstrained gage resistance
ΔR = Change in gage resistance in $\mu\Omega/\Omega$
L = Length
ΔL = Change in length in μ"/"

For a G.F. of 2, for every μ"/" change in length, there will be a change of 2 $\mu\Omega/\Omega$.

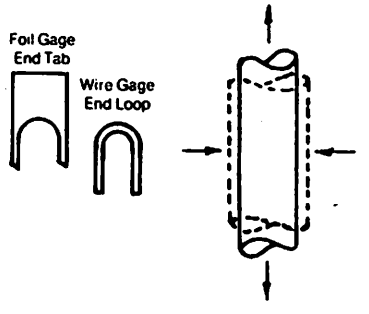
Therefore, a change of 1 μ"/" on 120 Ω represents a resistance change of 240 $\mu\Omega$ or 1000 μ"/" = 0.24 Ω.

1mV/V = 2000 μ"/" with G.F. of 2.

3. Apparent strain is due solely to the change of temperature with no mechanical restraints on the substrate.



4.



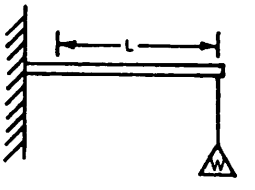
Poisson's Ratio (transverse effect) — is the strain acting perpendicular to the principal strain, and is approximately 30% of that strain on most steels.

For accurate strain measurements, the gage must be designed to sense minimal transverse strain. This can be achieved by eliminating end loop resistance.

5. CANTILEVER BEAM IN BENDING

Beam Material = 1018 Steel

E = Young's Modulus of Elasticity 30.0 x 10^6 PSI
W = Weight in Pounds
L = Distance to Gage in Inches
σ = Unit Stress, PSI
b = Width of Bar, Inches
h = Thickness of Bar, Inches
I = Moment of Inertia of Rectangular Beam
C = Distance of Surface above Neutral Axis
M = Applied Moment
ϵ = Strain in Micro Inches



$$\sigma = \frac{MC}{I} \quad (1)$$

$$M = WL \quad (2)$$

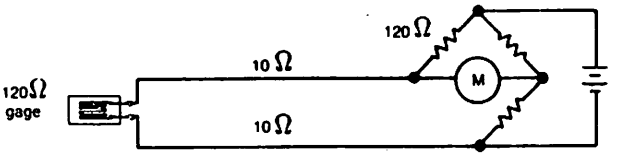
$$\sigma = \epsilon E \quad (3)$$

$$I = \frac{bh^3}{12} \quad (4)$$

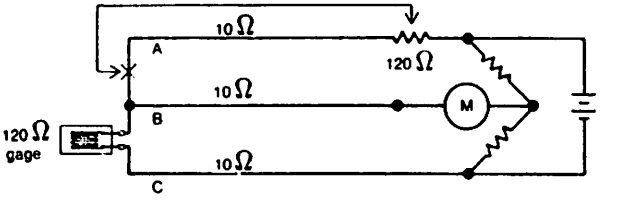
Substitution of Equation 2, 3, & 4 in Equation 1:

$$\epsilon = \frac{6WL}{Ebh^2}$$

6. Two wire — Produces substantial error at temperature, due to TC of copper wire.



Three wire — Temperature effects of copper wire in adjacent arms cancel out. L_3 is in series with meter having internal resistance of 200K Ω; therefore, TC of third lead shows no effect.



GENERAL INFORMATION

Bridge Excitation — The voltage or current used to power a Wheatstone bridge.

Bridge Output — Generally expressed as mV/V (2mV/V = 4000 μ"/")

Creep — The change in gage output vs time under sustained load.

Drift — Change in gage output, when temperature remains constant.

Hysteresis (Thermal) — The maximum deviation between the ascending and descending temperature cycles.

Hysteresis (Mechanical) — The maximum deviation between readings of no load, full load and no load. (Reading taken immediately after no load)

Eastman 910 Solvents: Nitromethene and Dimethyl Formide

Chapter 2

The Interface Hardware

2.1 Introduction

The hardware falls into two major categories: Analog and Digital. The analog boards include linear current amps to drive the motors, filters to limit input signals to the linear current amps, and differential strain gauge amplifiers to convert each gauge's change in resistance – which linearly corresponds to the strain placed on the gauge by the tendon – into a corresponding change in voltage. Figure 2.1 shows the main cabinet in which resides the power supplies and the I/O hardware (excluding the strain gauge amplifiers). As mentioned in the introduction, our JPL/Salisbury hand is being controlled by a MicroVax-II running a real-time operating system called VAXELN. Below is a picture of our reliable computer, "GISKAR" — see Figure 2.2.

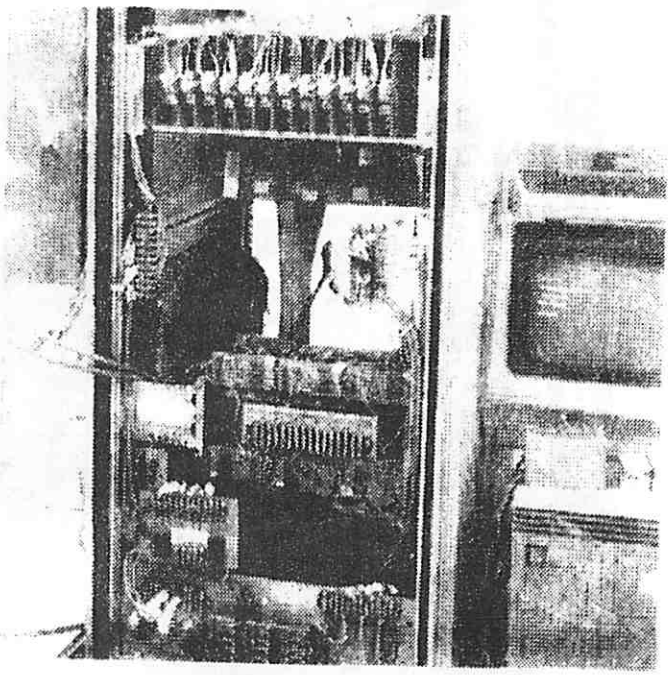


Figure 2.1: A picture of the hand hardware cabinet

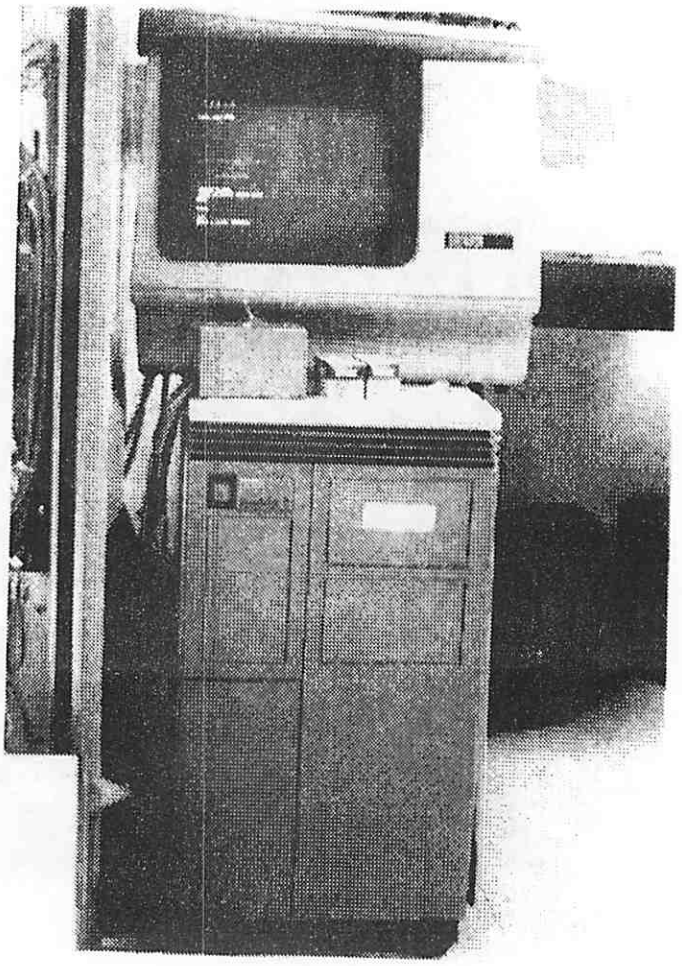


Figure 2.2: A picture of our faithful MicroVax-II

2.2 Analog I/O Boards

2.2.1 Linear Current Amplifiers

Functional description

The linear current amps are designed to drive the servomotors with as much as two amps of current. This maximum amount of current provides enough torque to make the tendons quite taut although not so tight as to be in danger of damaging any tendon cables. These amplifiers exhibit very little ringing at the output. The servomotors look very different to the linear amplifiers depending on whether or not they are spinning quickly. The following paragraphs explain the behavior of the circuitry under both conditions.

When a motor spins, the armature windings quickly cut through the electromagnetic lines of force surrounding the motor's field magnets. This causes a voltage to develop on the motor armature. Therefore, if one of the amplifiers is driving a servomotor to supply torque to a tendon and if the tendon isn't moving quickly, then there will be little or no back–emf voltage developed across the motor armature. In this case, the following rule applies:

$$\text{Torque(oz.in.)} = \text{I(armature)} \times 3.95\text{(our servomotor's constant)}$$

In other words, as long as no back–emf voltage develops, a motor's output torque is directly proportional to the input current from the linear current amps.

However, if a voltage is placed across the armature and the motor is allowed to spin freely, the motor will develop a back–emf voltage of opposite polarity and nearly equal to the armature's input voltage. The higher the armature input voltage (from an amplifier, perhaps) the faster the motor will spin. If the motor is spinning freely, the speed of the spin is proportional to the input voltage. Below is a description of this relationship.

When the amplifier's voltage is applied to the low–resistance armature, a large current flows through the motor armature – causing the motor to spin. When the armature spins, back–emf voltage develops that acts as an opposing voltage source in series with the armature. These forces interplay until a stable point is reached where the input and back–emf voltages are approximately equal and the motor spins at a constant speed. Since the amount of back–emf voltage developed is linearly related to the speed of the motor spin and the

amount of back–emf voltage under these conditions is virtually equal to the input voltage, the speed can be computed as:

$$\text{RPM} = \frac{\text{Voltage}}{\text{Constant x Field density}}$$

The linear current amplifiers are designed to drive a fixed amount of current regardless of the impedance of the load. This makes it possible to set the amount of torque supplied by the motor to the tendon to a constant value. However, when the motors are allowed to spin freely the effective impedance of the motor becomes so great – as a result of back–emf – that the amplifier is no longer able to supply the voltage needed to drive its desired current. Eventually the output voltage will saturate as it approaches one of the supply voltages. Thus, the power supply voltages of the linear current amplifiers effectively set the maximum speed attainable by the hand's servomotors. The positive power supply voltage has purposefully been set as low as possible in order to restrict the upper speed of the motor to within safe limits. At one time both positive and negative supply voltages were set at twenty–six volts. During early testing of the hand, this led to tendons being damaged by the high momentum run–away motors. The present positive power supply voltage is ten volts (the positive supply is the one that supplies power when the motors are being driven to tighten a tendon.)

Related figures and tables

To begin with, check out the design by perusing Figure 2.3. Consulting the manufacturer's documentation for the Apex PA–12 while studying the schematic is recommended. The PA–12 documentation is included at the end of this subsection.

The PA-12 and its auxiliary components have been fit onto a fairly small board (see Figure 2.4). The twelve LCA boards are mounted on a large air–cooled heatsink.

After problems adjusting the linear current amplifiers' gains arose, a gain–adjustment board was added. This board merely holds a set of potentiometers to which the linear current amps are attached via a barrier strip. This board allows us to alter the supply

voltages given to the amplifiers and then fine–tune the gains as needed. For a layout diagram of the gain–adjustment board, check Figure 2.5.

To troubleshoot the linear current amps, it will be necessary to refer to the barrier strip wiring diagram. See Figure 2.6.

You may, at some time, be interested in changing the layout of the linear current amp boards. Also, there may be times during a troubleshooting session when you'd like to trace the runs on the printed circuit board. To facilitate this work, the etching images are included. Take a look at Figure 2.7.

Lastly, you may want to build some of your own boards or replace parts at some time. To help in this situation, a list of all the linear current amplifier components is given in Table 2.1.

Things to Avoid

Some of the bugs that showed up during the debugging stage of this circuit were: problems associated with the PA–12 leads, strange behavior due to oscillations in the output, and various power supply problems.

The leads were a problem for a couple of reasons. Mainly, they had a tendency to short to the heatsink (a fan-cooled heat dissapation unit needed to protect the amps from overheating). To correct this, sheaths of wire insulation were slipped over the leads. Mounting the amps was very difficult until the equipment was arranged properly and made modular. Before this was done, the amps in the back of the heatsink were quite inaccessible.

The second two types of problems – power and oscillation – are intimately connected. Before the power supply voltages were reduced, the amps had a tendency to oscillate.

Other problems that cropped up included: bad powerline connections, and bad output line connections. For more information, see Section 3.4.

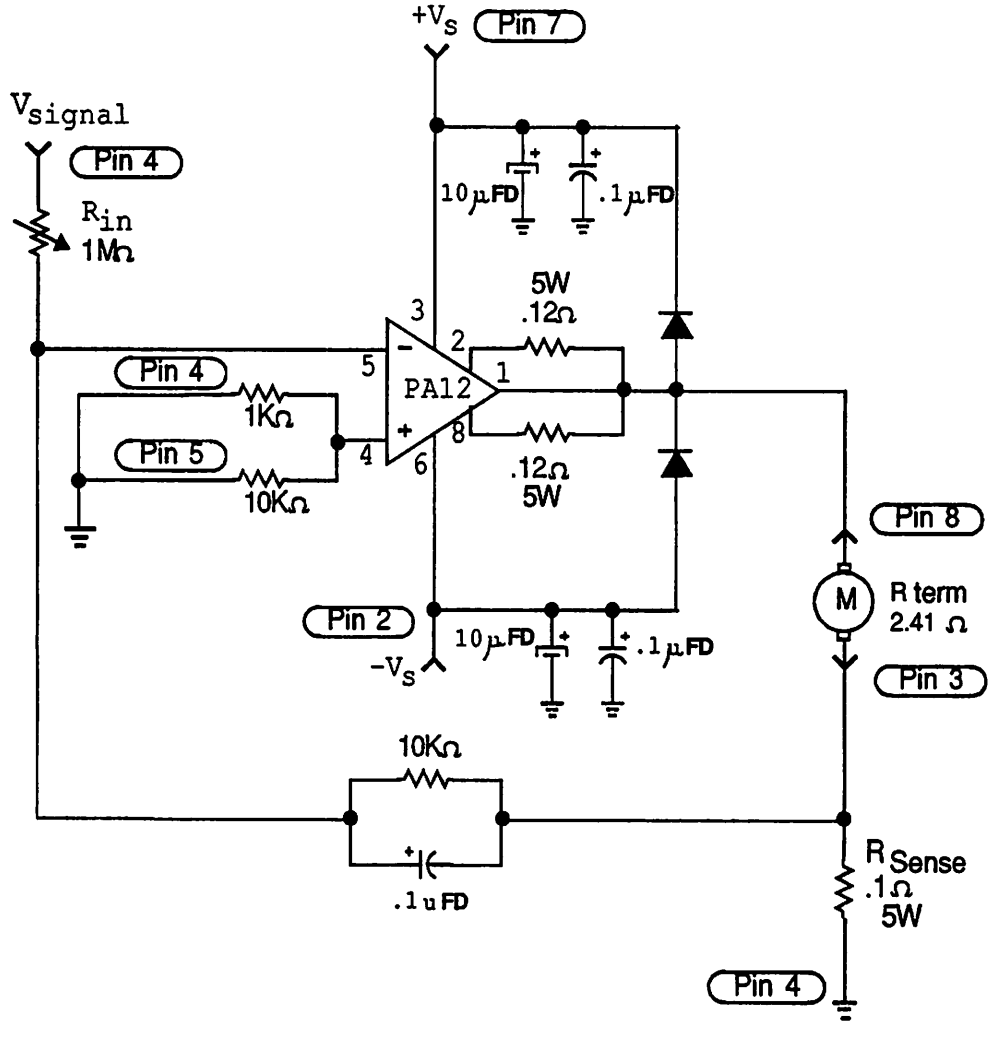


Figure 2.3: Linear Current Amp – Schematic

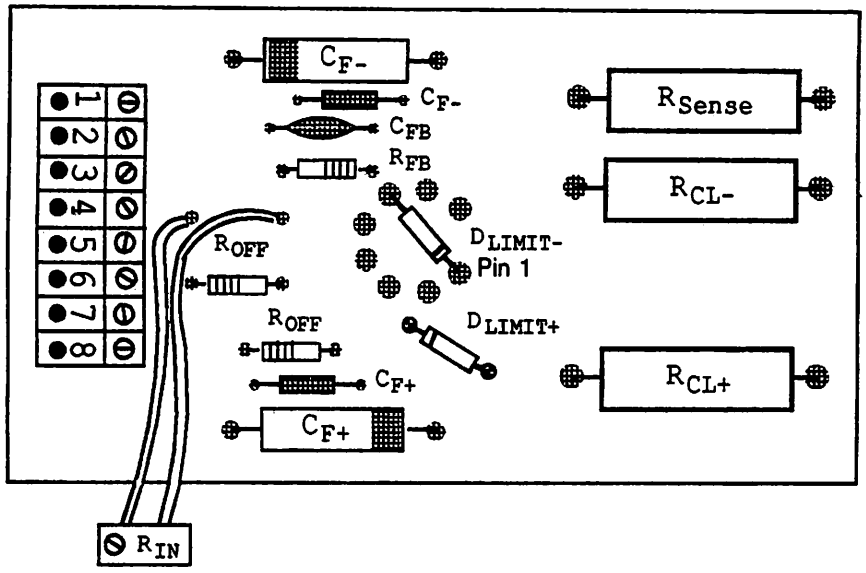
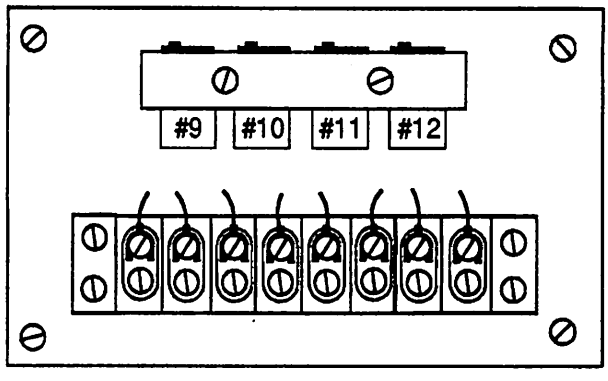


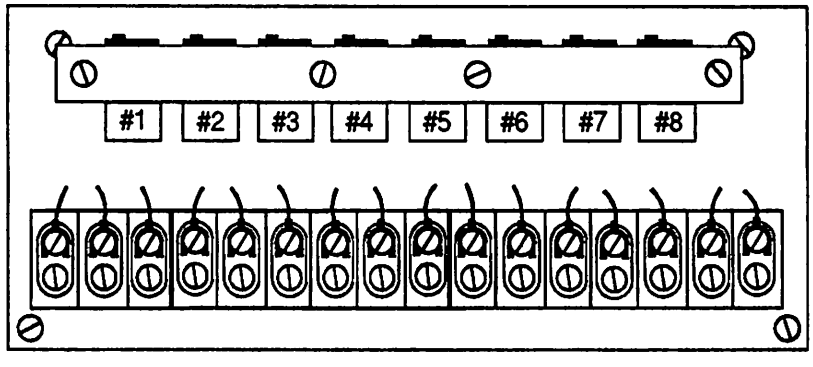
Figure 2.4: LCA – Board layout

Item	*Description*	*Qty.*
Barrier Strip	eight contact	1
Potentiometer	1M ohm, 1/4 watt, twenty–turn	1
Capacitors	10 MFD electrolytic	2
	.1 MFD mica	3
Diodes	power	2
Resistors	1K ohm, 1/4 watt	1
	10K ohm, 1/4 watt	1
	5 watt, $\pm$ 5%, wire-wound, .12 ohm	2
	5 watt, $\pm$ 5%, wire-wound, .1 ohm resistor	1
Op Amp	Apex Corp. – PA–12	1

Table 2.1: LCA – Component List



(Attached to the back of the LCA heatsink)



(Attached to the front of the LCA heatsink)

Figure 2.5: LCA – Gain-adjustment board layout

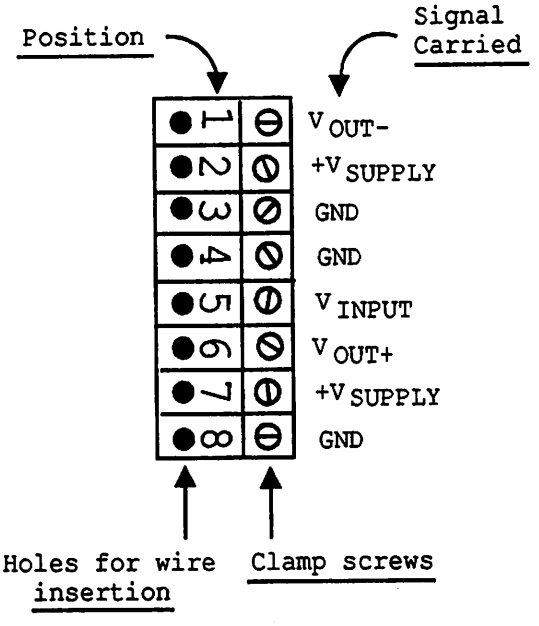


Figure 2.6: LCA - Output signal barrier strip pinouts

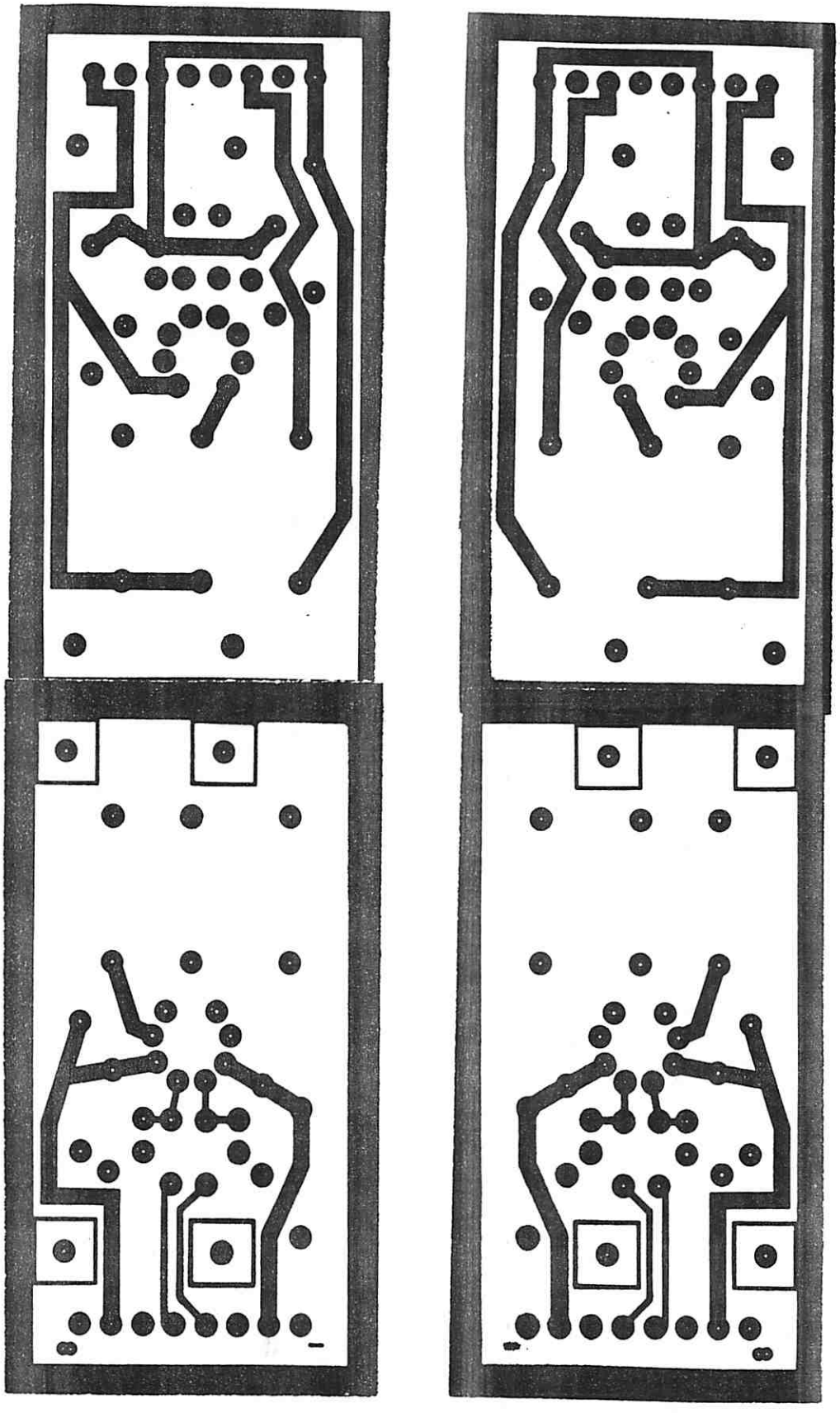


Figure 2.7: LCA – Board etching image

MANUFACTURER'S DOCUMENTATION FOLLOWS.

* Reprinted by permission from Apex

APEX µtech

POWER OPERATIONAL AMPLIFIER

APPLICATIONS NOTE 1

GENERAL OPERATING CONSIDERATIONS

INTERPRETING SPECIFICATIONS

Absolute maximum ratings are stress levels which when applied to the amplifier one at a time will not cause permanent damage. However, proper operation is only guaranteed over the ranges listed in the specifications table. For example, most amplifiers have an absolute maximum case temperature range of −55 to 125°C. If the operating temperature range is less (ie. −25° to 85°) an amplifier may latch to one of its supply rails when above that temperature (>85°C); however, the device will not sustain permanent damage unless the latched condition also violates the safe operating area. Simultaneous application of two or more of these maximum stress levels such as maximum power and temperature may induce permanent damage to the amplifier.

The absolute maximum power dissipation rating used by APEX is the generally accepted industry method which assumes the case temperature of the amplifier is held at 25°C and the junctions are operating at the absolute maximum rating. This standardization provides a yardstick when comparing ratings of various manufacturers. However, this is not a reasonable operating point because it requires an ideal heatsink. Furthermore even with the best heatsink, it would still result in reduced product life due to operation at extreme temperatures. Refer to the heatsink data sheet for information regarding operating junction temperatures and relative product life. APEX generally recommends maximum junction temperatures at 150° or below.

The specifications table should be the prime working document while designing the application. In addition to the minimum/maximum parameters (voltage offset, output capability, etc.) this table contains the guaranteed linear operating ranges (common mode voltage, temperature, power supplies, etc.).

Common mode voltage is another case which points out the difference between absolute maximums and linear operating range. On many amplifiers, the maximums allow inputs up to the power supply rails, while the linear operating range is 5-10V less than the power supply rails. This means inputs exceeding the linear range specification will not damage the part but the amplifier may not achieve the specified rejection ratio, start to distort the signal or may even latch to one of the supply rails.

STABILITY

Oscillations not only destroy signal fidelity but can cause catastrophic failures due to increased power dissipation. Oscillations caused by ground loops, inadequate supply bypassing or high impedance input nodes often are in the megahertz range and can easily be overlooked because the circuit is designed for low frequency operation. When testing the circuit, oscilloscope settings need to be adjusted to detect the presence of any high frequency oscillations. If present, reactive elements of the load can increase current drawn from the amplifier (capacitive load). What is worse, the current is not the phase with the output voltage further increasing the internal power dissipation.

To prevent oscillations, a reasonable phase margin must be maintained. The low pass created by sumpoint (−IN) capacitance and the feedback network may add phase shift and cause instabilities. As a general rule, the sumpoint load resistance (input and feedback resistors in parallel) should be selected so that the low pass breakpoint with the input and stray capacitance in parallel is at least 10 times the small signal bandwidth of the circuit. The external sumpoint stray capacitance to common, $+V_S$ or $-V_S$ should be kept at an absolute minimum.

Low impedance at the noninverting input (+IN) is equally important to prevent pickup of unwanted signals or even positive feedback. Capacitance to common (or a supply rail) is often helpful to eliminate the problem.

IMPROVING STABILITY

In applications where long cables or highly reactive loads are used but the bandwidth is not critical, instabilities can be eliminated by reducing the feedback at the high frequency end. This is done by adding a series RC network across the inputs as shown in Fig. 1. The value of the resistor (R_N) should be selected for a ratio of one tenth (1/10) to one hundredth (1/100) of R_F and the capacitance should be chosen for a corner frequency of less than one tenth (1/10) to one hundredth (1/100) of the small signal bandwidth of the amplifier respectively. Typical values are:

SMALL SIGNAL BW	RATIO	F_C	R_F	R_N	C_N
5 MHz	1/100	50KHz	10K	100	33nf
1 MHz	1/10	100KHz	10K	1K	1.5nf

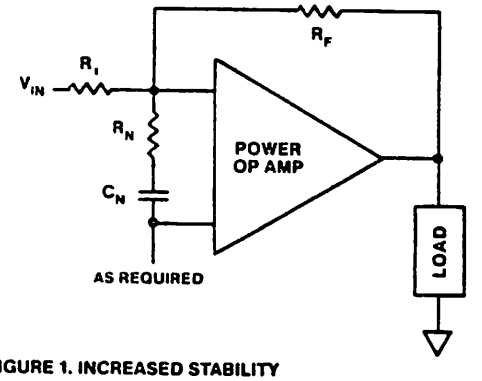


FIGURE 1. INCREASED STABILITY

This RC network can sometimes be used to increase the slew rate of externally compensated high speed Power Op Amps when requiring high speed at a low gain setting because the product data sheet will prescribe an external phase compensation which will reduce slew rate. By applying the RC network above, the AC gain of the amplifier is increased thereby allowing the compensation for a higher gain setting, thus increasing slew rate and bandwidth. Application note 3, Figure 3 illustrates this technique using the PA84 as an electro-static deflection amplifier.

SUPPLY VOLTAGE

The specified voltage ($+/-V_S$) applies for a dual (+/−) supply having equal voltages. A nonsymmetrical (i.e. +50/−10V) or a single supply (i.e. 60V) may be used as long as the total voltage between $+V_S$ and $-V_S$ does not exceed the maximum rating.

SAFE OPERATING AREA (SOA)

The output stage of most power amplifiers has 3 distinct limitations:

1. The current handling capability of the transistor geometry and the wire bonds.
2. The secondary breakdown effect which occurs whenever the simultaneous collector current and collector-emitter voltage exceed specific limits. (Does not apply for MOS output transistors.)
3. The junction temperature of the output transistors.

The SOA curves for the model to be used combine the effect of these limits. The direction and magnitude of the output current must be calculated or measured and checked against these curves. The direction of current flow determines which op amp output transistor is conducting and thus which supply voltage to use when calculating the supply to output differential.

* Reprinted by permission from Apex

The direction of the output current is obvious for purely resistive loads but not for reactive or EMF producing loads which often cause problems. For example, consider a power op amp running on ±28V with a large capacitive load. Assume the output voltage is near the negative supply rail and switches to the positive rail. The output transistor connected to the positive supply rail turns on the supplies current to the capacitor. Initially, the capacitor is charged to the negative output voltage and appears as a near short circuit. This places a voltage stress of twice the supply voltage on the power transistor (nearly 56V in this case).

CURRENT LIMIT ADJUST

Power Op Amps with provisions to adjust current limit externally require two current limit resistors (R_{CL}) which must be connected as shown in the applicable external connection diagram. Since output current flows through these resistors, wattage ratings must be considered. For optimum reliability, the resistor values should be set as high as possible. Each resistor and its power dissipation is calculated as follows:

$$R_{CL}\ (\Omega) = 0.65/I_{LIM}(A) - 0.01^{*} \qquad P(W) = 0.65 \times I_{LIM}$$

*Except for PA12, replace by 0.007

Nonsymmetrical current limiting ($R_{CL+} \neq R_{CL-}$) is permissible. For testing without heatsink, the resistors should be selected to limit power dissipation to less than 3W under short circuit conditions.

The external current limiting for the power op amps is not meant to be a precision current limit. A rule of thumb is to allow ±20% variation at room temperature. Furthermore, the current limit varies over temperature. This variation is generally shown in a typical performance graph in the product data sheet. Specific values of the nominal current at any given temperature may be calculated by modifying the 0.65V term of the above equation with −0.0022V/°C of case temperature rise. For example, at a case temperature of 125°C, this term becomes 0.43V rather than 0.65V. When working with high currents, connection impedances, lead lengths and solder joint resistances must be included when calculating the current limit.

HEATSINK

The amplifier may be operated without a heatsink only if the internal power dissipation is less than 3W at 25°C ambient. To determine the heatsink requirements for any application, follow this procedure:

1. Calculate the maximum DC or instantaneous power dissipation:

$$P = (V_S - V_O)\, I_O - (\,|+V_S| + |-V_S|\,)\, I_Q$$

2. Determine the maximum junction temperature (T_J) and thermal resistance of the Power Op Amp to be used and thermal resistance of the interface between heatsink and case of the amplifier (0.2°C/W for TO-3 packages mounted with thermal grease and without insulating washers). The technical notes section of the Heatsink data sheet discusses relative product life as a function of operating temperature, thermal properties of the mounting interface and mounting recommendations.

3. Calculate the minimum thermal resistance of the heatsink:

$$\Phi_{HS} = \frac{T_J - T_A\ (°C)}{P\ (W)} - \Phi_{JC} - \Phi_{HSC}$$

GROUNDING

Since the case is electrically isolated (floating) with respect to the internal circuits it is recommended to connect it, or the heatsink it is mounted on, to a local signal common. As an alternative, this connection may be AC coupled; i.e. mounting the amplifier side-by-side with a non-insulated hot case regulator (providing the regulator case terminal is well bypassed to common). Single point grounding of the input resistors and signal to common will prevent undesired current feedback, which can cause large errors and/or instabilities.

SINGLE SUPPLY OPERATION

These amplifiers are suitable for operation from a single supply voltage. The operating requirements do however, impose the limitation that the input voltages do not approach closer than 5 to 10V volts to either supply rail. Specific values are determined by the common mode voltage range given in the product data sheet. This is due to the internal operating voltage requirements of the power op amp current sources, the half-dynamic loads and the cascode stage. Thus, single supply operation requires the input signals to be biased at 5 to 10V from either supply rail. Figure 2 illustrates one bias technique to achieve this.

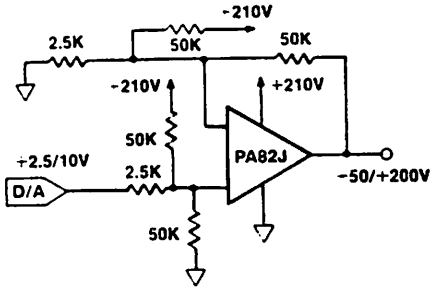


FIGURE 2. TRUE SINGLE SUPPLY OPERATION.

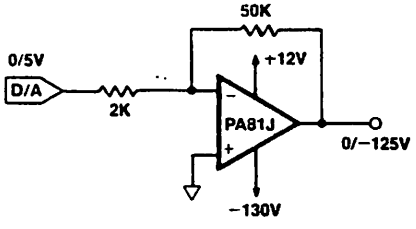


FIGURE 3. NON-SYMMETRIC SUPPLIES.

Figure 3 illustrates a very practical deviation from true single supply operation. The availability of the second low voltage supply still allows ground (common) referenced signals but also maximizes the high voltage capability of the unipolar output. This technique can utilize an existing low voltage system power supply and does not place large current demands on that supply. The 12 volt supply in this case must supply only the quiescent current of the Power Op Amp. If the load is reactive or EMF producing, the low voltage supply must also be able to absorb the reverse currents generated by the load.

BYPASSING OF SUPPLIES

Unless the leads to the power supply are less than 2" long, each supply rail should be bypassed to common with a capacitance of 10μF per Ampere of peak output current physically connected less than 2" from the power supply pins. Capacitors over 0.22μF should be made up of a 0.22μF ceramic capacitor in parallel with additional tantalum capacitors. For operating temperatures above zero °C computer grade aluminum electrolytic capacitors may be adequate.

SYMBOLS USED

C_N	External Rolloff Capacitor
I_{LIM}	Current Limit
I_O	Output Current
I_Q	Quiescent Current
P	Power
R_{CL}	Current Limiting Resistor
R_F	Feedback Resistor
R_I	Input Resistor
R_L	Load Resistor
R_N	External Rolloff Resistor
Φ_{HS}	Thermal Resistance of Heatsink
Φ_{HSC}	Thermal Resistance from Heatsink to Case
T_C	Case Temperature
T_A	Air Temperature (ambient)
T_J	Junction Temperature (Desired Max)
V_{IN}	Input Voltage
V_O	Output Voltage
$V_S - V_O$	Supply to Output Differential Voltage (Internal V-Drop)
V_S	Supply Voltage

* Reprinted by permission from Apex

APEX tech

POWER OPERATIONAL AMPLIFIER

APPLICATIONS NOTE 4

AVOID PREDICTABLE FAILURES

SAVE HOURS OF VALUABLE TIME

This applications note is intended to save you hours (maybe days) of hard work and avoid many frustrating experiences with power circuits. It guides you through your project and points out the common pitfalls in designing and testing power operational amplifier circuits. Many of these pitfalls are theoretically predictable. The majority have been identified in APEX Applications Hotline discussions of actual circuits and range from higher than expected errors to total destruction of the amplifier.

ELECTROSTATIC PROTECTION

In the spring and early summertime, many Arizonans become paranoid of electrostatic discharges (ESD). A walk across carpeting followed by a ¼" blue streak as one reaches for the light switch is a sign of kilovolt charges. Unless you live in a swamp, small geometry transistors used for op amp input stages are vulnerable to ESD. ESD damage causes a wide range of effects from increased voltage offsets or bias currents to total destruction. APEX products are shipped in antistatic foam to help prevent damage. Do not remove amplifiers from the foam for storage or stocking. Assembly procedures should require operators to use static straps. Be sure you touch the actual circuit ground and amplifier case with your hand prior to making contact between the amplifier pins and the circuit.

READ THIS BEFORE YOU APPLY POWER

In the design/prototyping phase of an application many dangers exist which will be eliminated by the time it is ready for production. Pins may be wired in reverse order, connections may be missing or test probes may cause inadvertent momentary shorts. All of these can destroy power amplifiers or other components in short order. Two procedures can be employed to substantially reduce these dangers:

1. Set power supplies to minimum operating levels allowed by the data sheet.
2. Set initial current limit levels to very low levels (2.2 ohms for high current models and 47 ohms for high voltage models). Resist the temptation to use the variable current limit feature of your lab supply. It is much safer to install the initial set of resistors. Even if average power dissipation is well controlled, you must operate within the safe operating area shown on the data sheet to avoid second breakdown of bipolar output stages. This mode of destruction results from simultaneous application of high current and high voltage to the conducting transistor. While each value may be within its maximum rating, a current crowding effect in the base region due to the fields generated by the voltage can destroy the device instantaneously. A low current limit on a commercial lab supply does not afford protection due to the surge current available from the output filter capacitors.
   With low voltage applied and reduced current limits in place, the basic function of the circuit can be verified. Once the circuit is operating as desired, raise the current limit and check worst case operating conditions (i.e. motor reversal, square wave drive of reactive loads or the one half supply voltage point at the output for reactive loads). Only then should you gradually raise the supply voltages to the maximum while checking worst case operation. This procedure not only saves many failures but it also helps to pinpoint problems to specific supply voltages and power levels.

Furthermore, it is suggested to use the largest possible heatsink for your prototype work. This precaution provides the best chance to make thermal measurements on the case of the amplifier during worst case loading conditions without premature failures from thermal overload. Once you verify your calculations you can go back to the smaller heatsink you want to use.

ABOUT CONTACTS AND SWITCHING

As you probably know, connection and disconnection of circuit components and leads should be made only with the circuit de-energized. Avoid all mechanical switches or relays in the high current branches of the power op amp. Unless the entire circuit is de-energized prior to switching, opening these contacts will create inductive kickback spikes because of the instantaneous change of current. The shorter the wires, the faster the risetime of these spikes. Amplitudes often reach several hundred volts, resulting in the immediate destruction of the amplifier. When relays or switch contacts must be used, protect the amplifier with zener diodes and capacitors at all nodes switched plus the supply pins.

These zener diodes at the supply pins may be required even though the power amplifier contains internal diodes to protect the output stages from inductive kickback. As the energy stored in the load is coupled to the power supply through these protection diodes, the power supply must absorb this transient which rises very fast. If the high frequency impedance of the power supply is too high to limit voltage transients to the absolute maximum rating of the amplifier, the short term overvoltage condition will cause the amplifier to break down.

Due to inductive kickback or slow operation, the mechanical switching of feedback elements also causes failures; particularly in circuits featuring high voltage amplifiers. Models without input protection have definite limits and models with protected inputs require limiting input risetimes to less than 1V/ns. Fig. 1A shows a simple gain select method which points out one possible problem with an unprotected input. Since the amplifier's full scale transition time (slew rate = 30V/μs) is faster than the typical relay switching time (milliseconds), the PA08 output may approach 150V while the relay is still switching. Because the 100k feedback resistor has completely discharged its associated rolloff capacitor, the relay will connect 150V directly to the input destroying the amplifier. Fig. 1B shows a protection network to prevent such damage. The diodes should be high speed devices such as 1N4148 and the series impedance should limit instantaneous current to a maximum of 150mA.

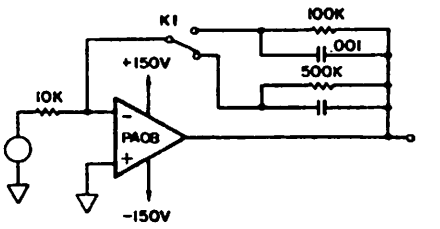


FIG. 1A

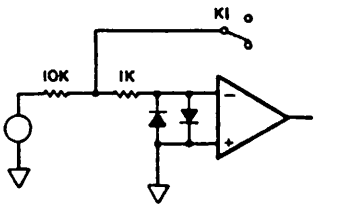


FIG. 1B

* Reprinted by permission from Apex

USING HEATSINKS CORRECTLY

When mounting amplifiers to heatsinks, do not use insulating washers! Direct mounting with thermal joint compound results in a thermal resistance of approximately 0.2°C/watt for the TO-3 package while the use of insulating washers usually increases this figure to 0.8°C/watt. Since the cases of APEX amplifiers are isolated, inclusion of the washer raises operating temperature and lowers reliability. Refer to the APEX heatsink data sheet for hardware recommendations and the effects of junction temperature on reliability.

When mounting power op amps to a heatsink which does not have a predrilled hole pattern, drill #45 holes for each of the eight pins of the TO-3 package. Do not drill a single large hole for all the pins! Nearly all heat is produced inside the pin circle. Single hole mounting dictates a long thermal path through the steel header and between the pins of the package. Because steel is not a good thermal conductor, the amplifier will overheat even on an oversized heatsink. Short term loss of operation, blistered fingers, and total destruction of amplifiers have all been reported when using single hole heatsinks.

Due to the dimensional tolerances of the TO-3 package extreme care must be taken not to let the pins touch the heatsink inside the holes. Do not count on the anodization for insulation. Sleeve at least two pins of the amplifier if you are using a socket or printed circuit board. If you are wiring directly to the pins it is best to sleeve all pins.

CHECK STABILITY

Amplifiers that become oscillators dissipate more power and are not reliable. If the frequency of oscillation is higher than the circuit bandwidth requirement, reactive elements of the load shift the phase. This tends to increase internal power dissipation beyond what is normally expected. Furthermore, the high power signals radiate and will interfere with communication, control, and process operations. Don't invite an FCC investigation!

Applications Note 1, "General Operating Consideration," discusses several techniques to maintain a stable circuit. A review of these suggestions can solve almost any problem. The following is a list of the most common instability situations reported:

1. Ungrounded cases cause oscillation, especially with faster amplifiers. The cases of all Apex amplifiers are isolated to provide mounting flexibility. The case is in close proximity to all the internal nodes of the amplifier and can act as an antenna. Providing a connection to ground prevents noise pickup, cross coupling or positive feedback leading to oscillations.
2. A standard inverting circuit includes an impedance matching resistor in series with the noninverting input to take advantage of the improved input offset current specification. The high impedance input becomes an antenna receiving positive feedback causing oscillation. Calculate the errors without using the resistor (some amplifiers have equal bias and offset currents negating the effect of the resistor). If the resistor is required, bypass it with a ceramic capacitor of at least 10nF.
3. Large electrolytic or tantalum capacitors are installed close to the amplifier pins as recommended but small ceramic bypass capacitors are omitted. The circuit may oscillate because the high frequency impedance of the large capacitors is not low enough to decouple the power supplies.
4. A prototype circuit is checked out and approved. A printed circuit board is built and all modes of operation test OK. A step and repeat technique then uses this same artwork to generate a multiple amplifier board. When tested, every amplifier on the multiple board oscillates. Cross coupling through the supplies and radiation is a major problem in multiple amplifier circuits. Use lots of bypass capacitors, ground the case and include an input RC network as shown in Applications Note 1 if possible.

POWER SUPPLIES CAN BE A PROBLEM

Power op amps often require additional high voltage supplies. The additional supplies may be turned on sequentially or unpredictably. Energizing a control circuit before the power op amp results in an input to a power op amp lacking supply voltage. This violates the maximum input voltage rating. To prevent damage to the power op amp, clamp the input voltages or limit the inupt pin current to 1mA.

If operating power op amps on unregulated supplies make sure high and low line voltages have been allowed for, as well as high voltage transients on the incoming line. Due to the high speed, these spikes will pass right through most transformers and appear across the electrolytic filter capacitors. A good line filter is the first step to a solution. Zener diodes on the DC side can offer absolute protection.

WATCH THE CURRENT LIMIT RESISTORS

The current limit circuits on APEX amplifiers are provided to allow protection within the safe operating area (SOA). The schematic of the PA01 data sheet shows the typical circuit used. This approach is fast and provides a desirable negative temperature coefficient but does not provide a precision limit. If a precision limit is called for use an external circuit. If the external circuit reduces speed it may not protect against second breakdown. Check the SOA graph before implementing the circuit.

The current limit resistors also serve to degenerate (negative feedback) tracking errors between the V_{BE} multiplier bias current and the V_{BE} of the power transistors. You must use a resistor (the large the better) to program no more than the maximum rated output current of the power op amp. Using a jumper will lead to increased bias current in the output stage. This raises power and temperature, while lowering resistance to second breakdown, thereby destroying reliability.

On the other side of the coin, operating the amplifiers without current limit resistors installed (current limit pins left open) can cause failures, especially with inductive loads. This includes even a momentary open circuit while switching current limits with mechanical contacts. For the high current series power op amps, minimum programmed limits should be 20mA, while 10mA is minimum for the high voltage series. Open circuits or limits below these minimums can cause voltage breakdown of the current limit transistors (Q3 and Q5 of the PA01).

HELP IS JUST A PHONE CALL AWAY
(800) 421-1865

The APEX Applications Hotline provides technical support all the way through your project. In many cases, specific failure prevention can be suggested immediately. In some instances we will need the amplifier to be sent to APEX for a failure analysis. The results of the analysis can pinpoint the area of damage which then narrows down the circuit problem.

*Reprinted by permission from Apex

APEX μtech

NEW IMPROVED

PA12 • PA12A

POWER OPERATIONAL AMPLIFIERS

FEATURES

- CURRENT FOLDOVER PROTECTION — NEW
- MIL-STD-883B VERSION — PA12M
- TEMP. RANGE — −55 to +125°C (PA12A)
- EXCELLENT LINEARITY — Class A/B Output
- WIDE SUPPLY RANGE — ±10V to ±50V
- HIGH OUTPUT CURRENT — up to ±15A Peak

APPLICATIONS

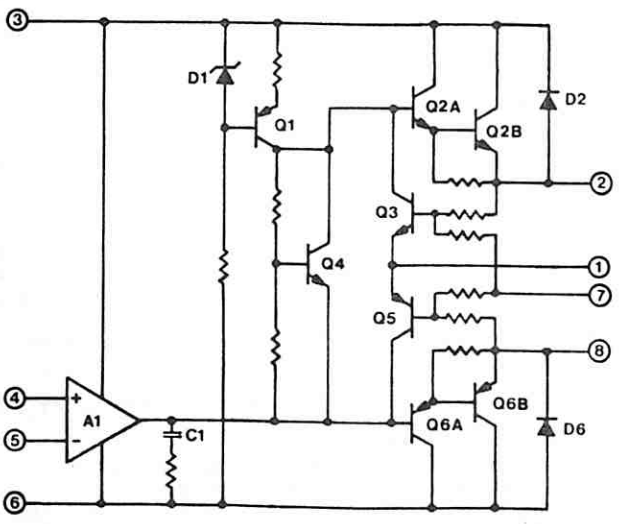
- MOTOR, VALVE AND ACTUATOR CONTROL
- MAGNETIC DEFLECTION CIRCUITS UP TO 10A
- POWER TRANSDUCERS UP TO 100KHz
- TEMPERATURE CONTROL UP TO 360W
- PROGRAMMABLE POWER SUPPLIES UP TO 90V
- AUDIO AMPLIFIERS UP TO 120W RMS

DESCRIPTION

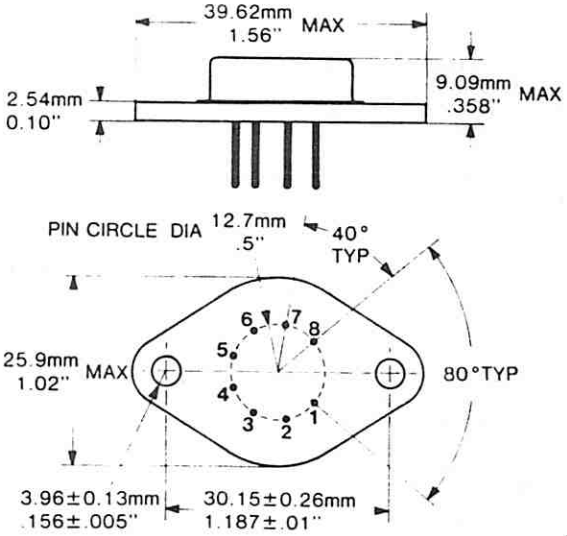
The PA12 is a state of the art high voltage, very high output current operational amplifier designed to drive resistive, inductive and capacitive loads. The complimentary darlington emitter follower output stage is protected against inductive kickback or back EMF. For optimum linearity especially at low levels, the output state is biased for class A/B operation using a thermistor compensated base-emitter voltage multiplier circuit. The safe operating area (SOA) can be observed for all operating conditions by selection of user programmable current limiting resistors. For continuous operation under load, a heat sink of proper rating is recommended. (See AP-Note 1).

A data sheet supplement for the MIL-STD-883B screened devices is available upon request.

These hybrid integrated circuits utilize thick film (cermet) resistors, ceramic capacitors and semiconductor chips to maximize reliability, minimize size and give top performance. Ultrasonically bonded aluminum wires provide reliable interconnections at all operating temperatures. The 8 pin TO-3 package is hermetically sealed by one shot resistance welding.

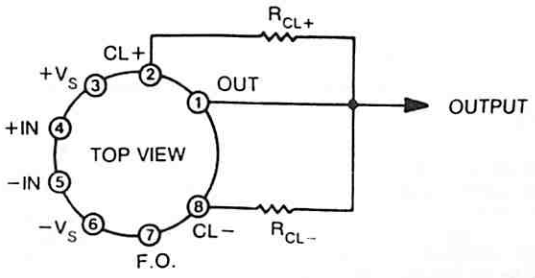


PACKAGE OUTLINE



PIN DIAMETER:	1.01mm or .04"
PIN LENGTH:	10.2mm or .40" MIN
PIN MATERIAL STD:	Nickel plated alloy 52, solderable
PIN MATERIAL MIL:	Gold plated alloy 52, solderable
PACKAGE:	Hermetic, nickel plated steel
ISOLATION:	300VDC any pin to case
SOCKET:	8 PIN TO-3, APEX PN: MS01
HEATSINKS:	8 PIN TO-3, APEX PN: HS01-HS04

EXTERNAL CONNECTIONS



* Reprinted by permission from Apex

ABSOLUTE MAXIMUM RATINGS

	PA12	PA12A
SUPPLY VOLTAGE, $+V_S$ to $-V_S$	100V	100V
OUTPUT CURRENT, source	15A	15A
OUTPUT CURRENT, sink	see SOA	see SOA
POWER DISSIPATION, internal	125W	125W
INPUT VOLTAGE, differential	$\pm V_S$-3V	$\pm V_S$-3V
INPUT VOLTAGE, common mode	$\pm V_S$	$\pm V_S$
TEMPERATURE, pin solder-10s	300°C	300°C
TEMPERATURE, junction[1]	200°C	200°C
TEMPERATURE RANGE, storage	-65 to +150°C	-65 to +150°C
TEMPERATURE RANGE, operating (case)	-55 to +100°C	-55 to +125°C

SPECIFICATIONS

	TEST CONDITIONS	PA12 MIN	PA12 TYP	PA12 MAX	PA12A MIN	PA12A TYP	PA12A MAX	UNITS
INPUT								
OFFSET VOLTAGE, initial	T_C=25°C		±2	±6		±1	±3	*mV*
OFFSET VOLTAGE, vs. temperature	Specified temperature range		±10	±65		±10	±40	µV/°C
OFFSET VOLTAGE, vs. supply	T_C=25°C		±30	±200		±30	±200	µV/V
OFFSET VOLTAGE, vs. power	T_C=25°C		±20			±20		µV/W
BIAS CURRENT, initial	T_C=25°C		12	30		10	20	nA
BIAS CURRENT, vs. temperature	Specified temperature range		±50	400		±50	400	pA/°C
BIAS CURRENT, vs. supply	T_C=25°C		±10			±10		pA/V
OFFSET CURRENT, initial	T_C=25°C		±12	±30		±5	±10	nA
OFFSET CURRENT, vs. temperature	Specified temperature range		±50			±50		pA/°C
INPUT IMPEDANCE, dc	T_C=25°C		200			200		MΩ
INPUT CAPACITANCE	T_C=25°C		3			3		pf
COMMON MODE VOLTAGE RANGE[3]	Specified temperature range	$\pm V_S$-5	$\pm V_S$-3		$\pm V_S$-5	$\pm V_S$-3		V
COMMON MODE REJECTION, dc	Specified temperature range	74	100		74	100		db
GAIN								
OPEN LOOP at 10Hz	T_C=25°C,1KΩ Load		110			110		db
OPEN LOOP at 10Hz	Spec. temp. range,8Ω Load	96	108		96	108		db
GAIN BANDWIDTH PRODUCT at 1MHz	T_C=25°C,8Ω Load		4			4		MHz
POWER BANDWIDTH	T_C=25°C,8Ω Load	13	20		13	20		KHz
PHASE MARGIN	Spec. temp. range,8Ω Load		20			20		°
OUTPUT								
VOLTAGE SWING[3]	T_C=25°C,PA12=10A, PA12A=15A	$\pm V_S$-6			$\pm V_S$-6			V
VOLTAGE SWING[3]	Spec. temp. range,I_O=5A	$\pm V_S$-5			$\pm V_S$-5			V
CURRENT, peak	T_C=25°C	10			15			A
SETTLING TIME to .1%	T_C=25°C,2V Step		2			2		µs
SLEW RATE	T_C=25°C	2.5	4		2.5	4		V/µs
CAPACITIVE LOAD	Specified temp. range,G=1			1.5			1.5	nF
CAPACITIVE LOAD	Specified temp. range,G>10			see SOA			see SOA	
POWER SUPPLY								
VOLTAGE	Specified temperature range	±10	±40	±45	±10	±40	±50	V
CURRENT, quiescent	T_C=25°C		25	50		25	35	mA
THERMAL								
RESISTANCE, AC[4]junction to case	T_C=-55 to +125°C,F>60Hz		.8	.9		.8	.9	°C/W
RESISTANCE, DC junction to case	T_C=-55 to +125°C		1.25	1.4		1.25	1.4	°C/W
RESISTANCE, junction to air	T_C=-55 to +125°C		30			30		°C/W
TEMPERATURE RANGE, specified	T_C	-25		+85	-55		+125	°C

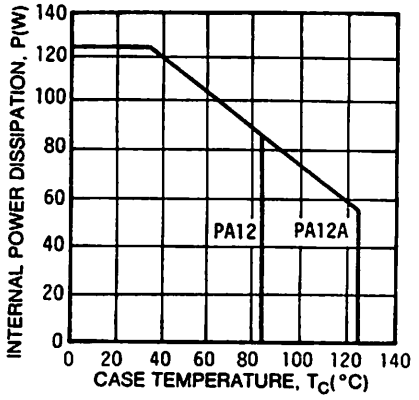
NOTES:

1. Long term operation at the maximum junction temperature will result in reduced product life. Derate internal power dissipation to achieve high MTTF.
2. The power supply voltage for all tests is ±40V unless otherwise specified as a test condition.
3. $+V_S$ and $-V_S$ denote the positive and negative supply rail respectively. Total V_S is measured from $+V_S$ to $-V_S$.
4. Rating applies if the output current alternates between both output transistors at a rate faster than 60Hz.
5. The internal substrate contains beryllia (BeO). Do not break the hermetic seal. If accidentally broken, do not crush, machine or subject to temperatures in excess of 850°C to avoid generating toxic fumes.

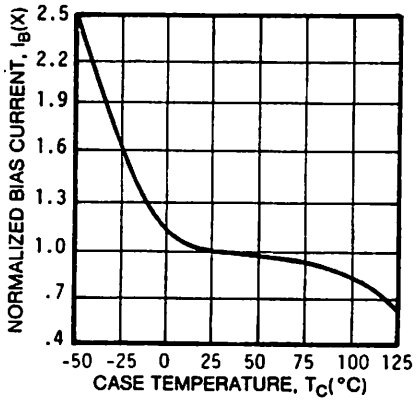
*Reprinted by permission from Apex

TYPICAL PERFORMANCE GRAPHS

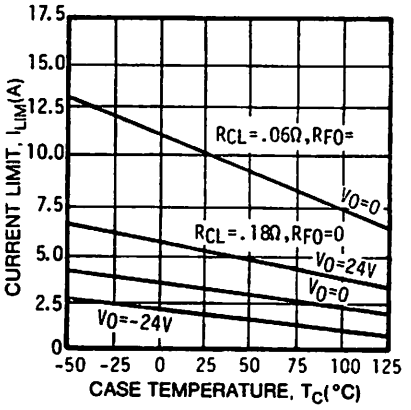
POWER DERATING



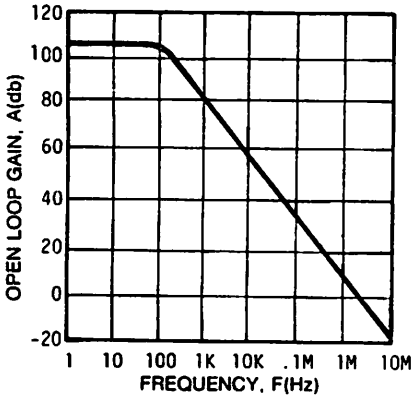
BIAS CURRENT



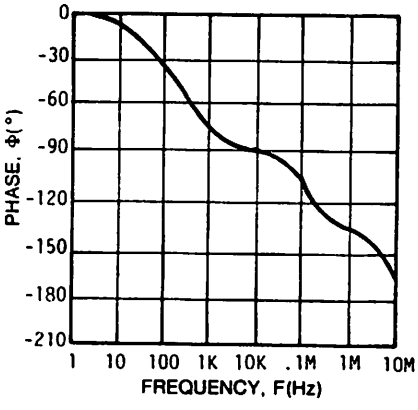
CURRENT LIMIT



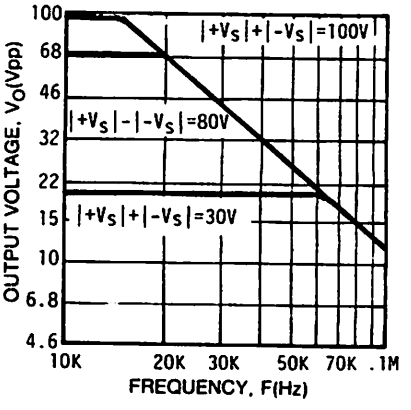
SMALL SIGNAL RESPONSE



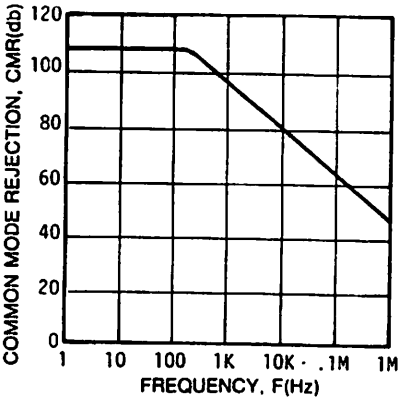
PHASE RESPONSE



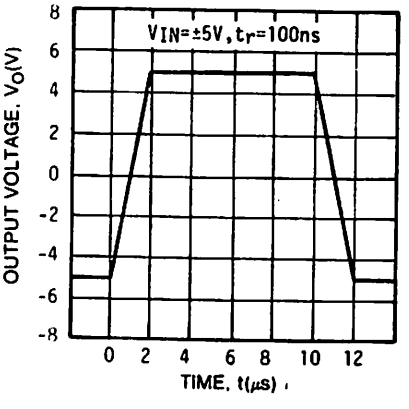
POWER RESPONSE



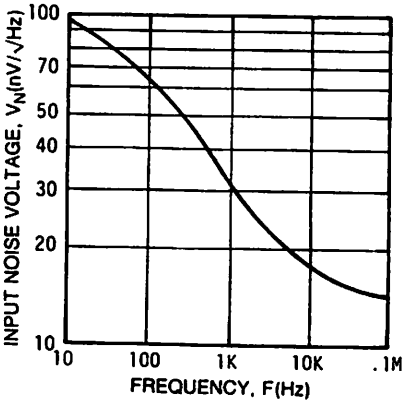
COMMON MODE REJECTION



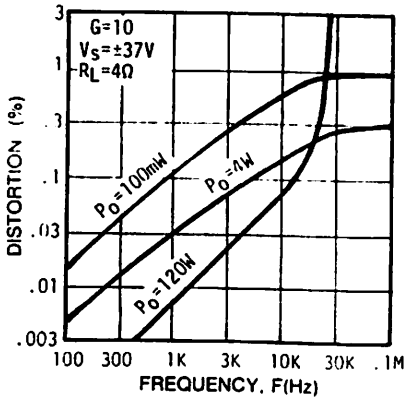
PULSE RESPONSE



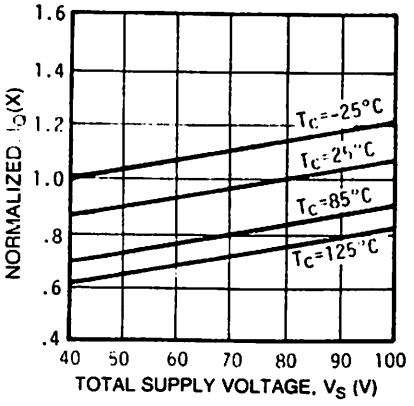
INPUT NOISE



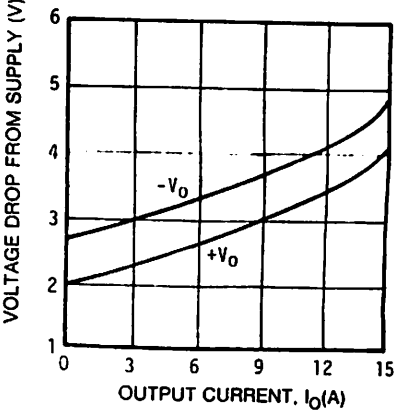
HARMONIC DISTORTION



QUIESCENT CURRENT



OUTPUT VOLTAGE SWING



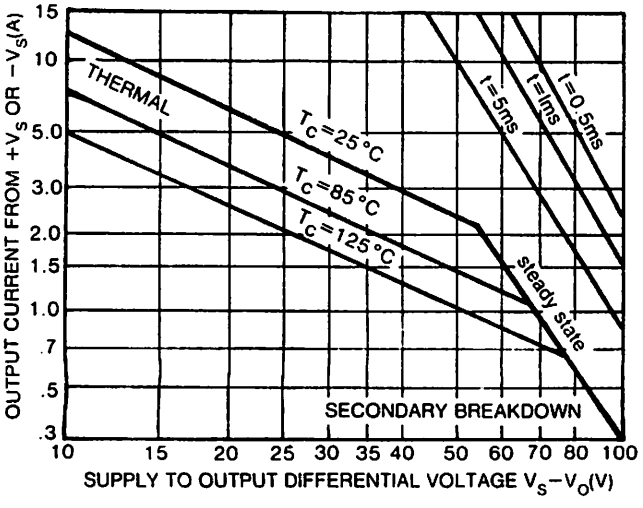
*Reprinted by permission from Apex

OPERATING CONSIDERATIONS

GENERAL

Please consult Power Operational Amplifier Applications Note 1 "General Operating Considerations", which covers stability, supply, heatsink and bypassing requirements. The information given here covers specific considerations for the PA12.

SAFE OPERATING AREA (SOA)



The SOA curves combine the effect of all limits for this Power Op Amp. For a given application, the direction and magnitude of the output current should be calculated or measured and checked against the SOA curves. This is simple for resistive loads but more complex for reactive and EMF generating loads. The following guidelines may save extensive analytical efforts:

1. Capacitive and dynamic* inductive loads up to the following maximums are safe with the current limits set as specified:

	CAPACITIVE LOAD		INDUCTIVE LOAD	
$\pm V_S$	I_{LIM} = 5A	I_{LIM} = 10A	I_{LIM} = 5A	I_{LIM} = 10A
50V	200μF	125μF	5mH	2.0mH
40V	500μF	350μF	15mF	3.0mH
35V	2.0mF	850μF	50mH	5.0mH
30V	7.0mF	2.5mF	150mH	10mH
25V	25mF	10mF	500mH	20mH
20V	60mF	20mF	1,000mH	30mH
15V	150mF	60mF	2,500mH	50mH

*If the inductive load is driven near steady state conditions, allowing the output voltage to drop more than 10V below the supply rail with Ilim = 15A or 25V below the supply rail with Ilim = 5A while the amplifier is current limiting, the inductor should be capacitively coupled or the current limit must be lowered to meet SOA criteria.

2. The amplifier can handle any EMF generating or reactive load and short circuits to the supply rails or shorts to common if the current limits are set as follows at T_C = 25C:

$\pm V_S$	SHORT TO $\pm V_S$, C, L or EMF LOAD	SHORT TO COMMON
50V	.30A	2.4A
40V	.58A	2.9A
35V	.87A	3.7A
30V	1.5A	4.1A
25V	2.4A	4.9A
20V	2.9A	6.3A
15V	4.2A	8.0A

These simplified limits may be exceeded with further analysis using the operating conditions for a specific application.

FOLDOVER PROTECTION

For certain applications, foldover protection allows for increased output current as the output of the Power Op Amp swings close to the supply rail. The new version of this Power Op Amp provides such a programmable current foldover protection function. This function can be activated by connecting pin 7 directly or through a resistor to ground, and controlled by the following equation:

$$I_{LIM} = \frac{.65 + \frac{.28\ V_O}{20 + R_{FO}}}{R_{CL} + .007} \qquad (1)$$

Where:
I_{LIM} is the current limit at a given output voltage V_O.
R_{FO} is the current foldover resistor pin 7 to ground in KΩ.
R_{CL} is the current limit resistors in Ω.
V_O is the instantaneous output voltage in V.

This means that the closer the output swings to the supply rail, the higher the current limit, or inversely; the more voltage is developed across the current carrying output transistors, the lower the current limit.

PROCEDURE

1. Enter the SOA graph x-axis with the supply voltage and find the maximum short circuit output current on the y-axis.

2. Now calculate the required current limiting resistor for a value of output current below the maximum found on the SOA:

$$R_{CL}(\Omega) = .65/I_{LIM}(A) - .007 \qquad (2)$$

3. Find the current limit for the maximum output voltage swing and pin 7 connected to ground/common:

$$I_{LIM} = \frac{.65 + \frac{.28\ V_O}{20}}{R_{CL} + .007} \qquad (3)$$

This is the highest current limit possible at maximum output. It may be decreased without effecting the short circuit current limit by putting a resistor in series with pin 7.

The following equation can be used to calculate R_{FO} (KΩ) using a lower current limit:

$$R_{FO} = \frac{.28\ V_O}{I_{CL}(R_{CL} + .007) - .65} - 20 \qquad (4)$$

4. To calculate the current limit at any output voltage (V_O), use equation "one". If the output transistor has to sink current, assign a negative sign to V_O and check the calculated current against the SOA graph.

5. The power dissipation for R_C is calculated as follows:

$$P(W) = .65\ I_{LIM} \qquad (5)$$

6. The power dissipation for R_{FO} should be ¼ W.

EXAMPLE

1. For Operation with a ±28V Supply at a maximum case temperature of 60°C we interpolate "3A" on the SOA graph.
2. R_{CL} (for 3A) = .22Ω (short circuit protection).
3. Using equation (3) we find I_{LIM} = 4.34A at V_O = 24V and for current sinking I_{LIM} 1.38A at V_O = −24V.

2.2.2 Linear Current Amp Input Filters

Functional description

S. T. Venkataraman is responsible for designing the low–level controller for the hand. His control strategy assumes that the motors can only be driven to tighten the tendons. In other words, when tendons tighten, the opposing tendons should, at most, be relaxed. They should never be pushed loose. This is how the human hand works as well. Our muscles only contract. To make it impossible for the computer system to unwind the tendons, input filters have been placed on the input signals to the linear current amplifiers. These clip any positive voltages to the amplifiers, since it is the positive input voltages that would cause the motors to unwind the tendons. Also, the input filter has a low–pass element that cuts out high frequencies which would cause oscillations to develop in the amplifier/motor circuitry.

Related figures

Take a peek at the simple schematic for this circuit in Figure 2.8.

To troubleshoot the linear current amps, it will be helpful to refer to the barrier strip wiring and board layout diagrams. See Figures 2.11 and 2.9.

In the following subsubsection, the function of this board is described as clipping positive voltage signals sent to the linear current amps. Since diodes have been used in the LCAIF to limit the positive signals, the clipping does not begin until the diodes begin to conduct. In order to be able to set the clipping voltage right at zero volts, an extra board was added which supplies a reference voltage to the cathode of each diode. To check out the layout of this auxiliary board, look at Figure 2.10.

Circuit description

The linear current amp input filter board is located in the front of the chassis on the left–hand side and has a small auxiliary wire–wrap board mounted just above it. The small wire-wrap board holds an operational amplifier that supplies a reference voltage to the main board. The main board uses diodes to clip the input voltages. The cathodes are tied to the output of the operational amplifier's reference voltage and the anodes connect through

a resistor to the input voltage. Thus, when the voltage of an input signal reaches .7 volts above the reference signal, the diode conducts, and the voltage at the anode is held – the resistor from the anode to the input voltage limits the current flow that occurs when the diode conducts. The anode is also connected to a unity-gain operational amplifier buffer that presents a high impedance to the input signal. This keeps the signal from being loaded down. Lastly, the diode has a small capacitor in parallel with it to short high-frequency voltage transitions to the reference voltage. This is a bug. The high frequencies should be shorted to ground, although the present arrangement works.

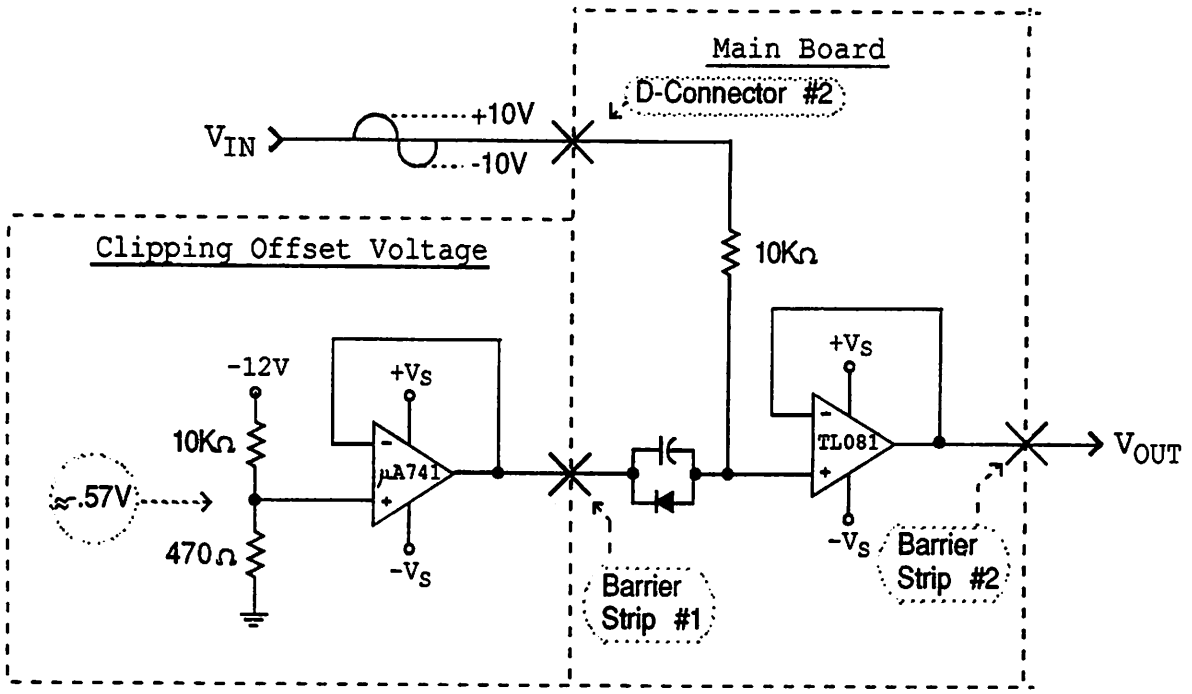


Figure 2.8: Linear Current Amp Input Filter – Schematic

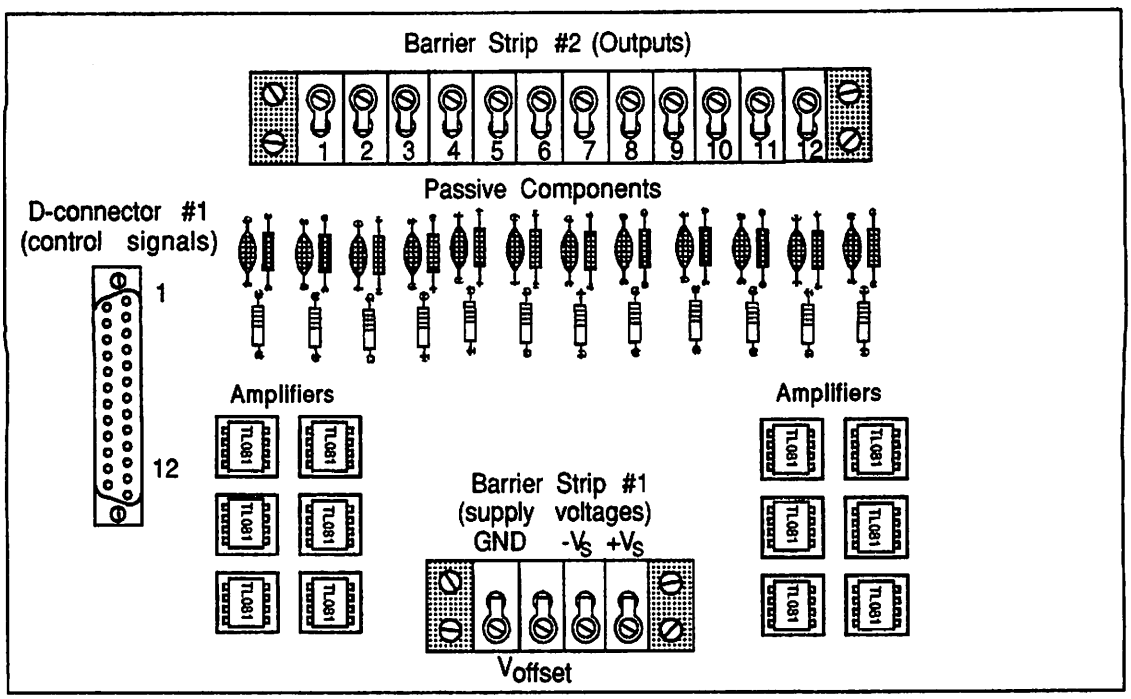


Figure 2.9: LCAIF – Main board layout

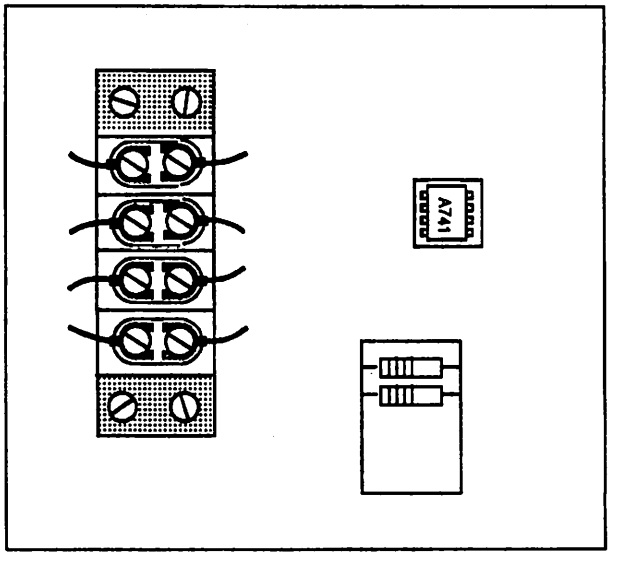


Figure 2.10: LCAIF – Voltage offset board layout

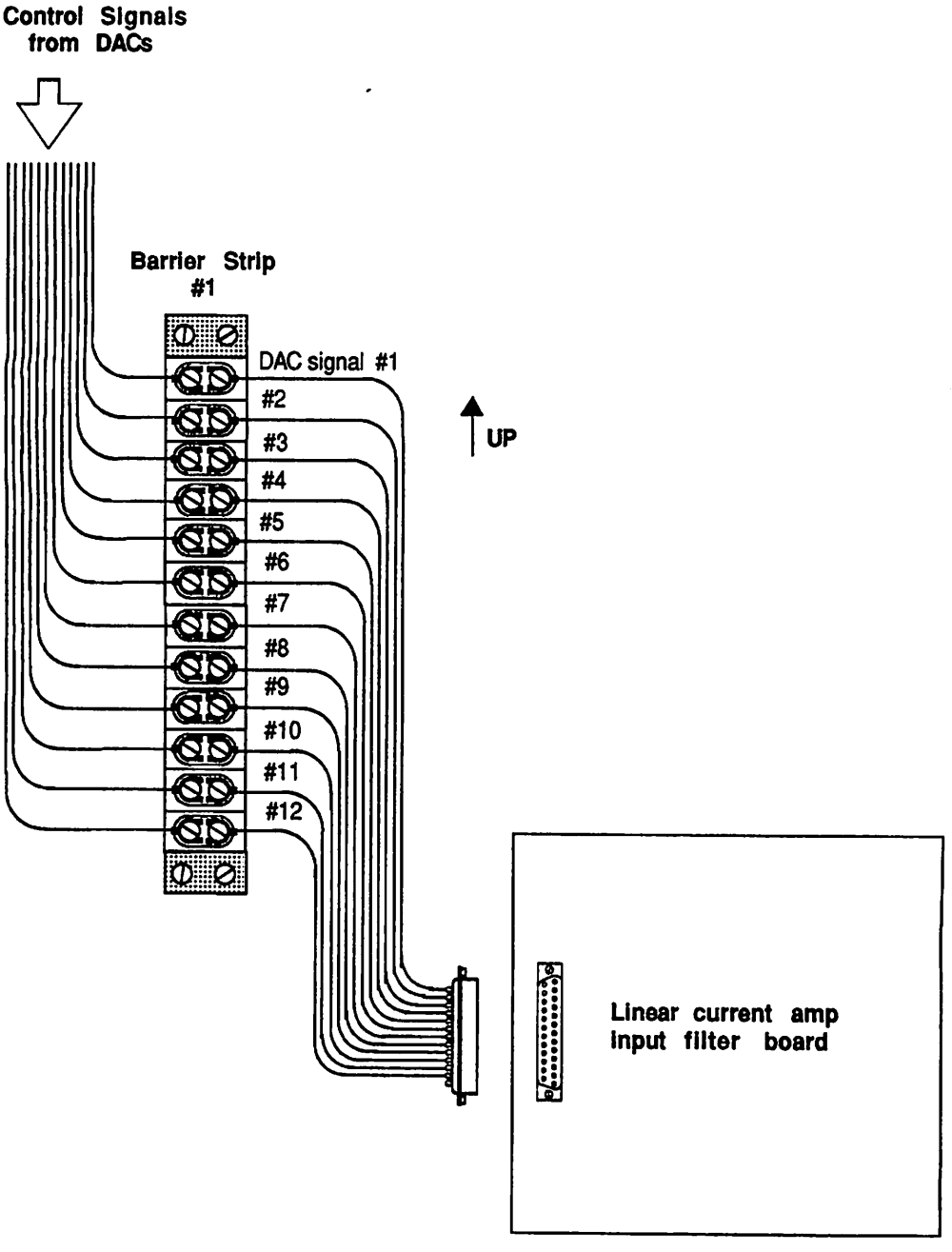


Figure 2.11: LCAIF – Barrier strip pinout

2.2.3 Strain Gauge Amplifiers

Functional description

The purpose of the strain gauge amplifiers is to convert the resistive changes of the strain gauges into linearly corresponding voltage changes. This is accomplished by amplifying the differential voltage signal developed across a wheat–stone bridge which has one of the strain gauges as a component.

Related figures

There are several figures associated with this board. The basic schematic is shown in Figure 2.12.

To assist in troubleshooting, connector pinouts (Figures 2.15, 2.16 and 2.17) and a board layout diagram and picture (Figures 2.13 and 2.14) are included.

Circuit description

Because of the probability of noise developing in the wires leading from the strain gauges to the amplifiers, the amplifiers are mounted in a box near the hand to keep these wires as short as is feasible. When the hand is mounted on an arm, the servomotor enclosure and the strain gauge amplifier box should be installed near each other on the back of the arm. This circuit is really quite simple so no elaboration is needed.

Calibration

Turn on the amps, vary the tendon tension of each of the tendons between loose and high tension, and check the outputs with an oscilloscope or voltmeter. The output voltages should swing between +5V and -5V. Negative five volts corresponds to a loose tendon and positive five volts corresponds to a tight tendon. If the voltage does not swing ten volts, adjust the amplifier gain. Once the output voltage swings over ten volts, adjust the offset potentiometer so that when the tendon is loose the output voltage is slightly above negative five volts. NOTE: Due to the smaller diameter of the pulleys in the first joint for tendons 2, 3, 6, 7, 10 and 11, these tendon's strain gauges feel a smaller percentage of tendon strain.

The greater the pulley diameter, the greater the tendon angle across the gauge and the more force is felt by the gauge. We have compensated for this by using higher gain amplifiers with these gauges.

The A/D converters do not provide absolute tendon tension information because of bugs in the amplifier design. The amplifiers drift due heat variation in the IC package. A much improved design has been developed by Uri Herel [4]

Bugs to avoid

No important bugs have shown up in this circuit. It would be good to redesign the board so that its outputs are a bit more stable over time. As it is, it works well for providing general relative strain information. For acquiring dependable absolute strain values, a new circuit and precision calibration is needed.

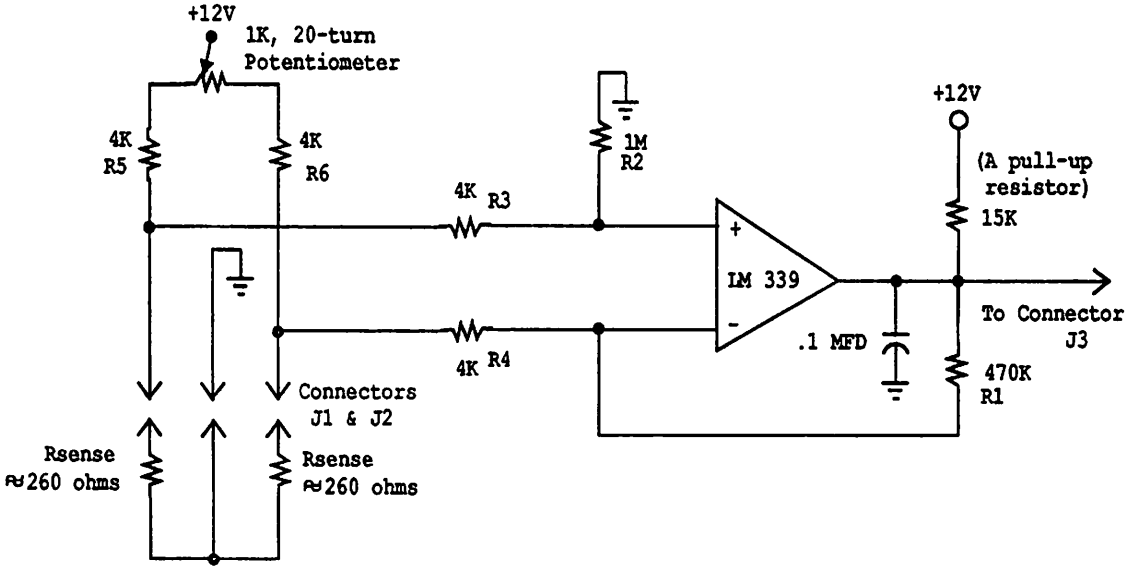


Figure 2.12: Strain Gauge Amp – Schematic

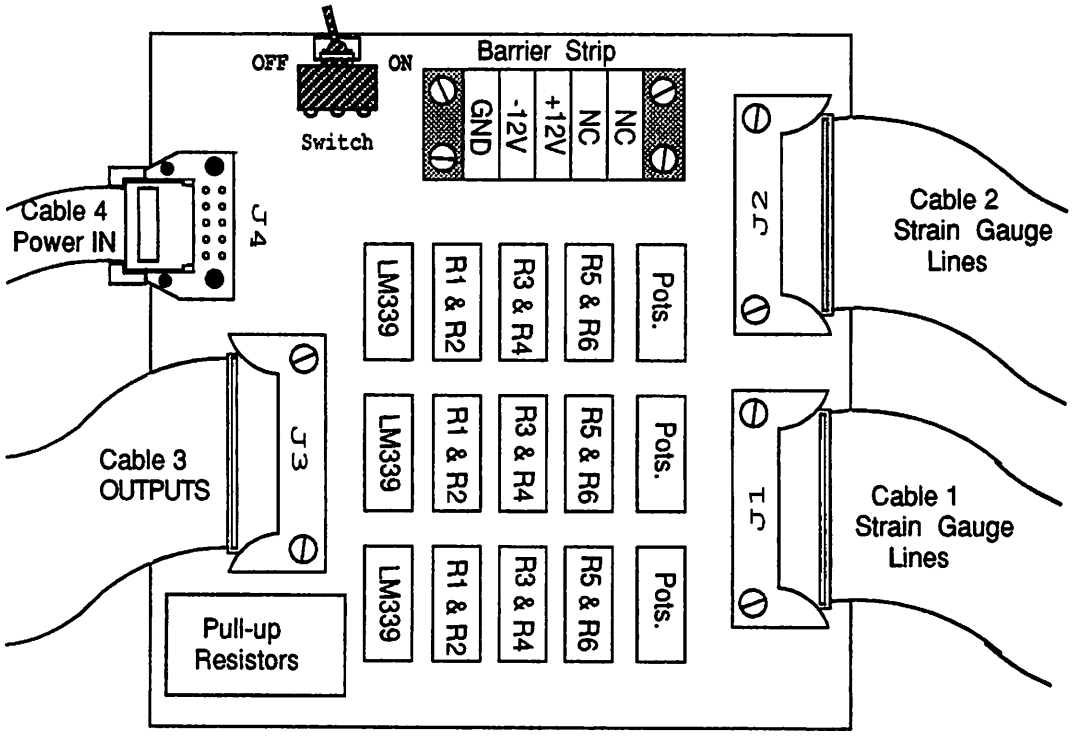


Figure 2.13: SGA - Board layout

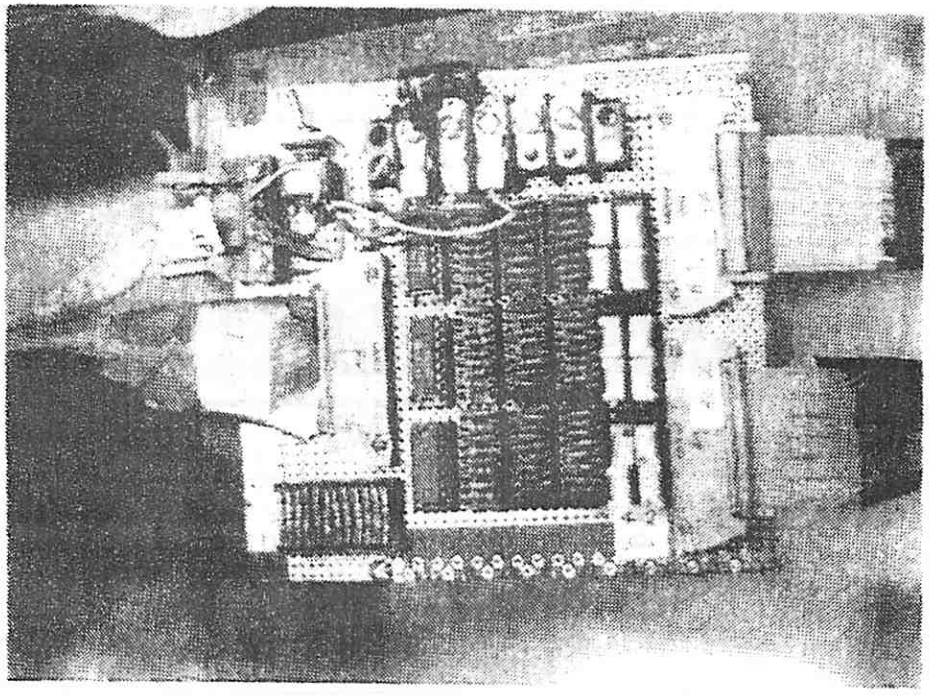


Figure 2.14: A picture of the Strain Gauge Amp Board

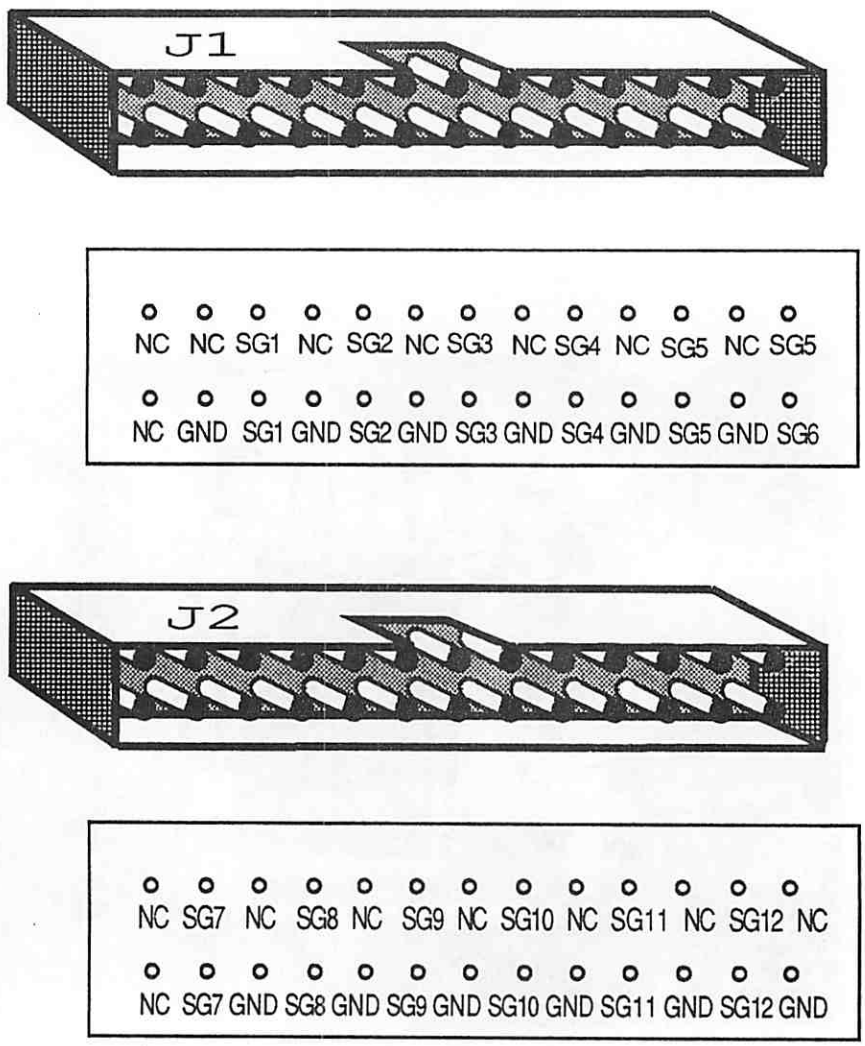


Figure 2.15: SGA – Strain gauge lines pinout

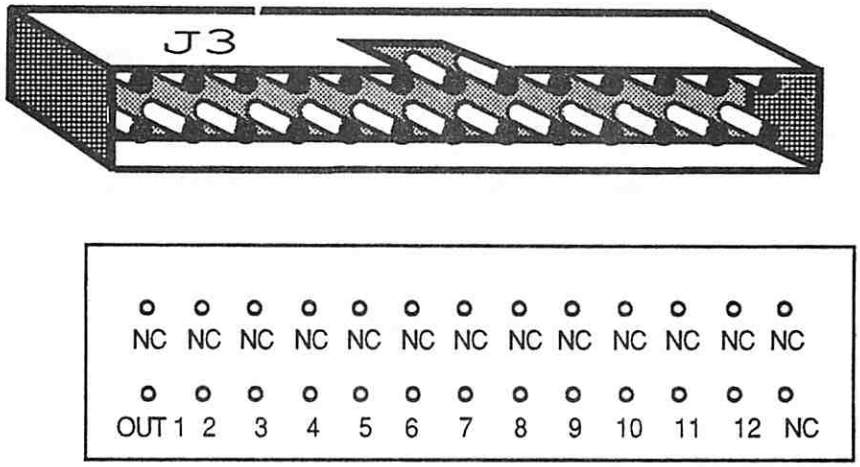


Figure 2.16: SGA – Output lines pinout

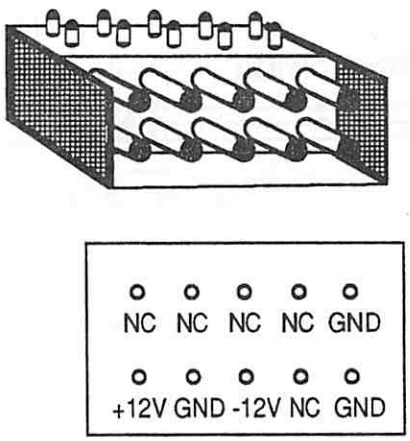


Figure 2.17: SGA – Power lines pinout

Because there are quite a few jumpers that have to be incorporated into the original printed circuit board before things run properly, please refer to Table 2.3 for a list of all the jumper placements.

If you wish to check out the layouts of the original and proposed single–tendon I/O boards, look at Figures 2.19 and 2.20.

There are very different pinouts for the edge–connectors of these two circuit boards. Refer to them in Figure 2.21 and 2.22. The pinout for the D–subminiature connector on these boards is shown in Figure 2.23.

If you want to build some of these boards, check out the component list for the original boards in Table 2.2.

Circuit description

These printed circuit boards are located within the main hardware chassis, just above the center. There are twelve of them plugged into a wire–wrapped backplane, where they are easily accessible. Be very cautious of static when handling these boards. Ground yourself via a static discharge wristband. The A/Ds and DACs will blow–out or suffer parametric damage (performance degradation) if you are careless. Also, there is an anti–static IC remover to use when it is necessary to take out these chips to troubleshoot the board.

Bugs to avoid

One bug that shows up from time to time is broken lines in the parallel I/O cables. If the cables are being disturbed regularly, it is quite likely that the metal strands will break at the point where the connector pins punch through the insulation.

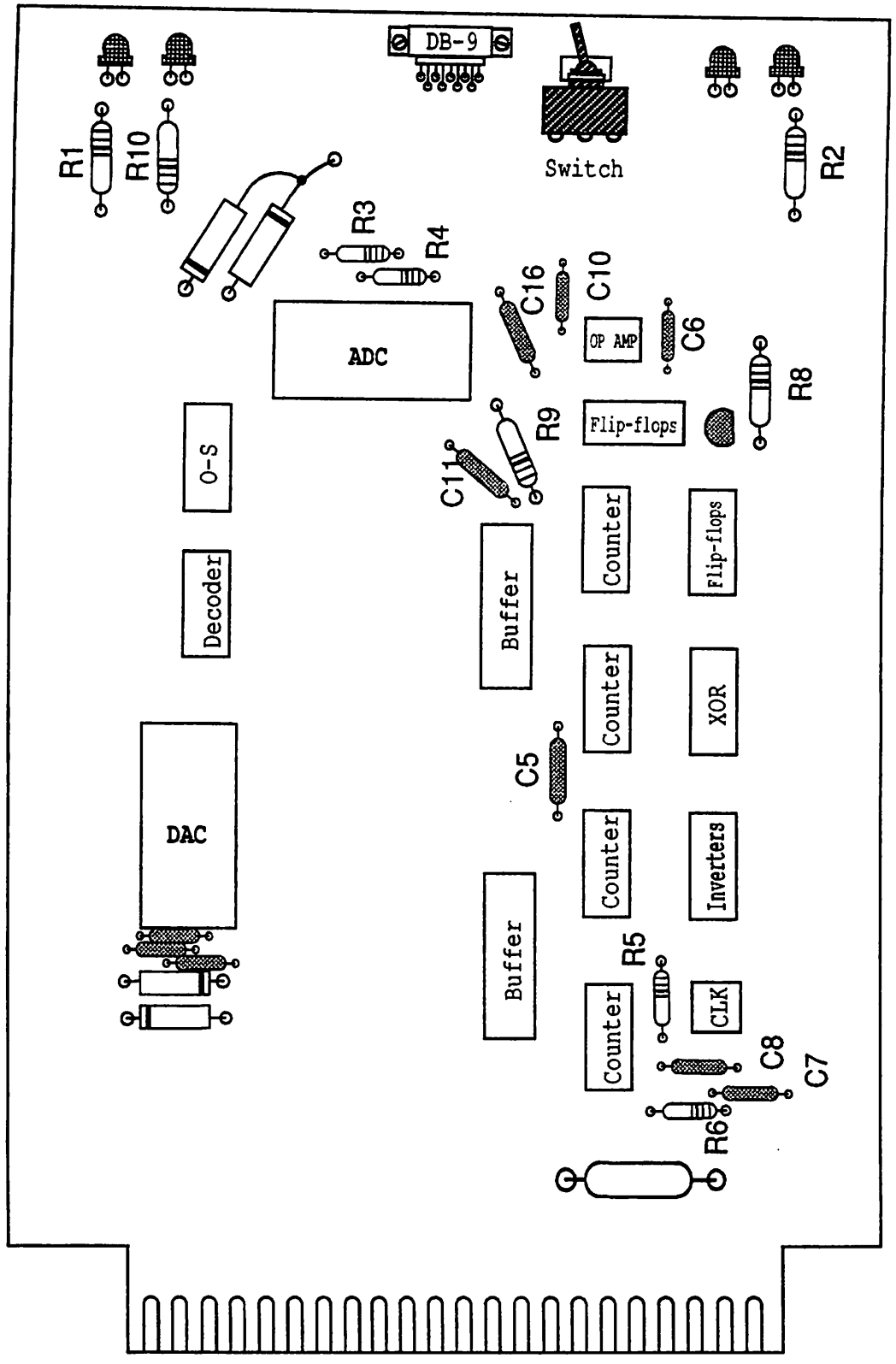


Figure 2.19: STIOB - Original board layout

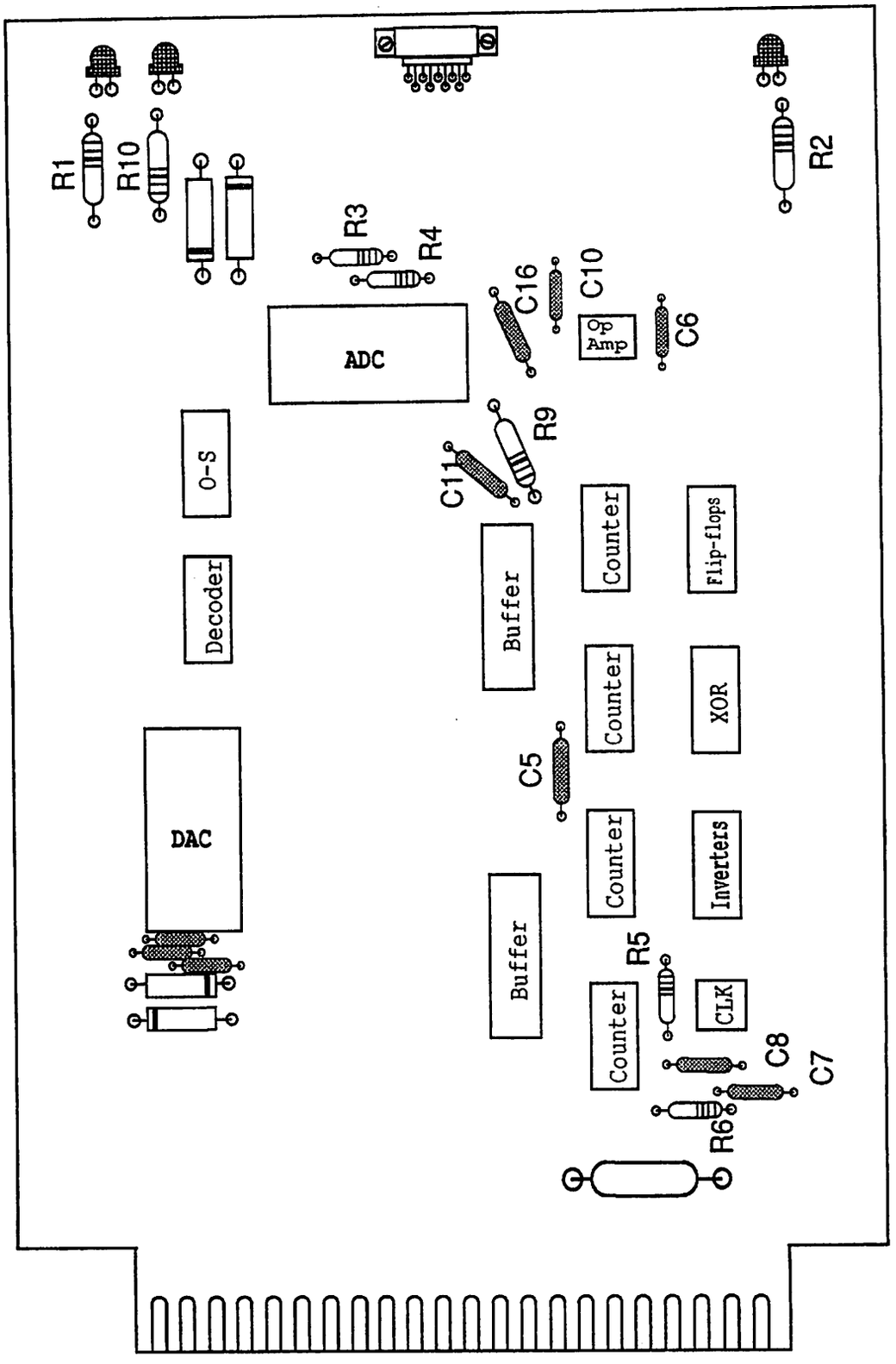
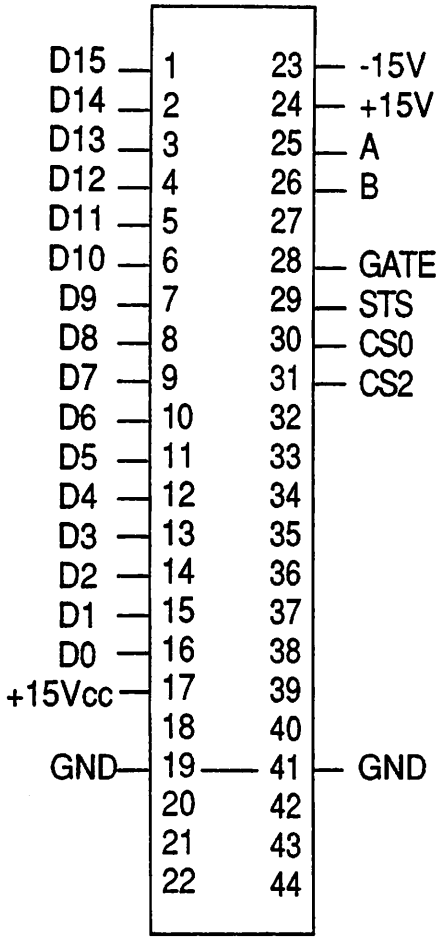
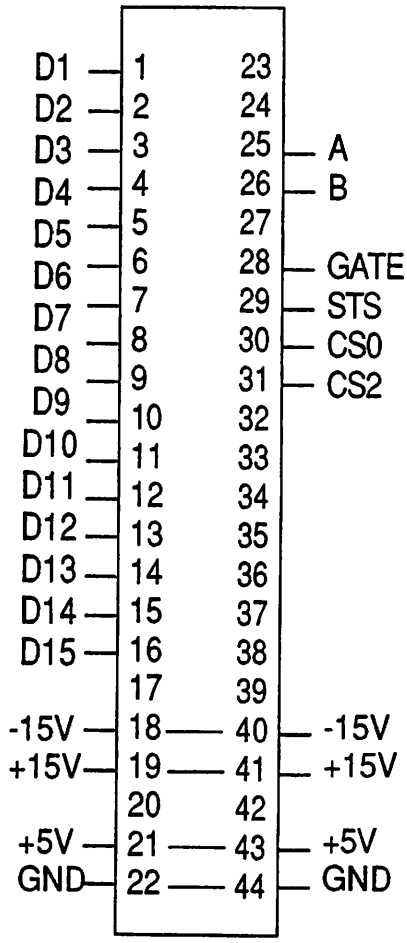


Figure 2.20: STIOB – New board layout



(As seen from wire-wrap pin side)

Figure 2.21: Original STIOB – edge–connector pinout



(As seen from wire-wrap pin side)

Figure 2.22: New STIOB – edge–connector pinout

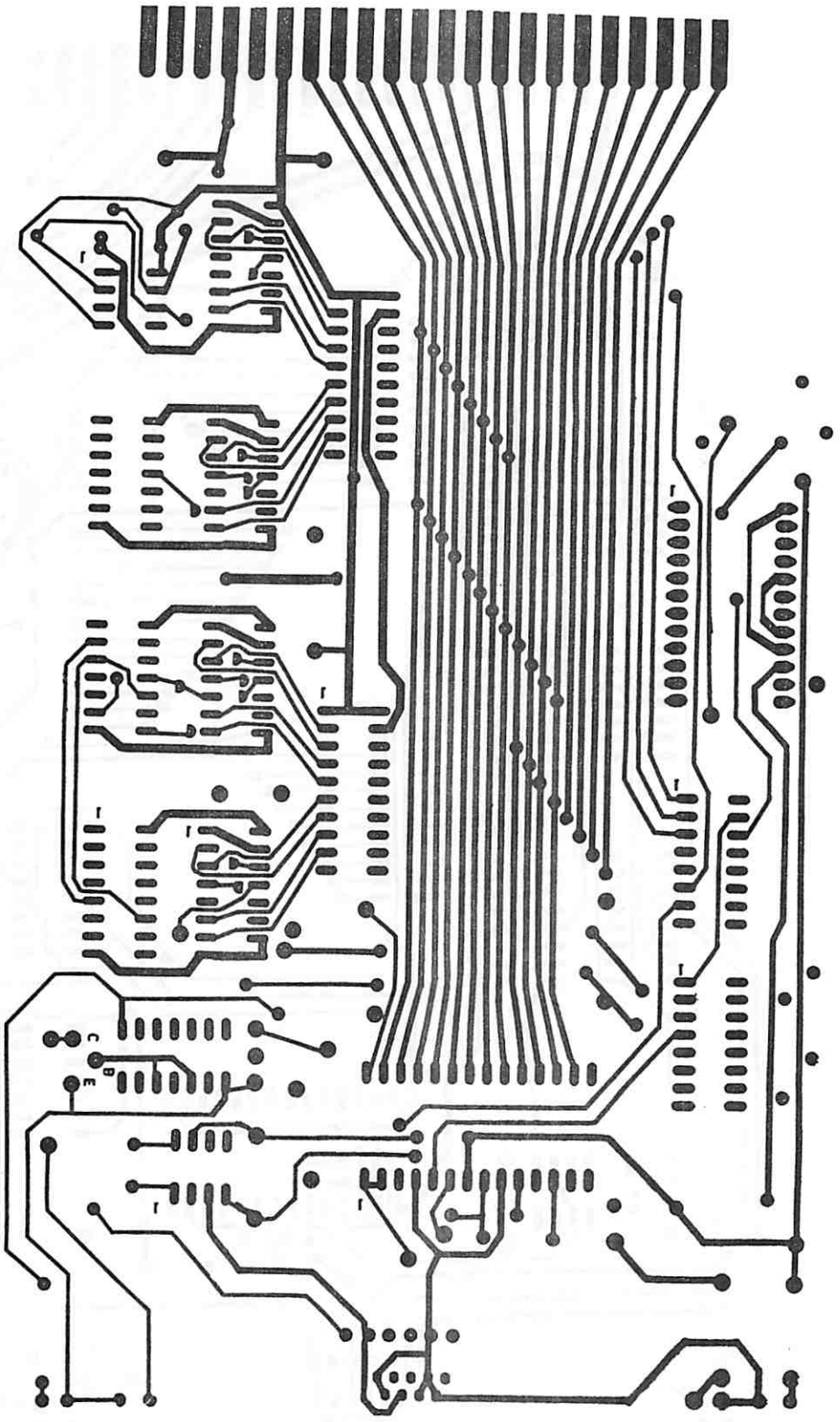


Figure 2.25: Original etching transparency with errors – underside

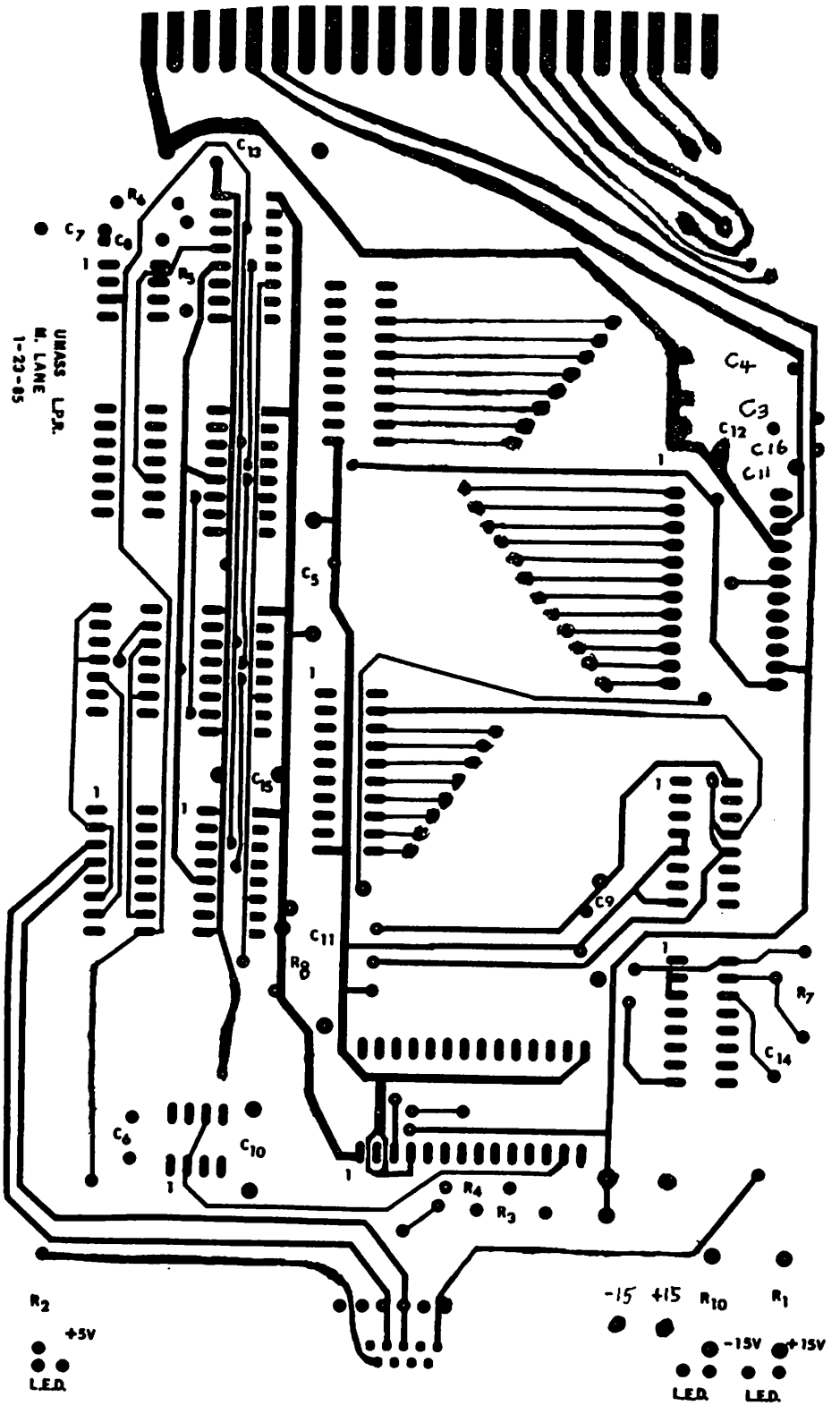


Figure 2.26: New (untested) etching transparency – topside

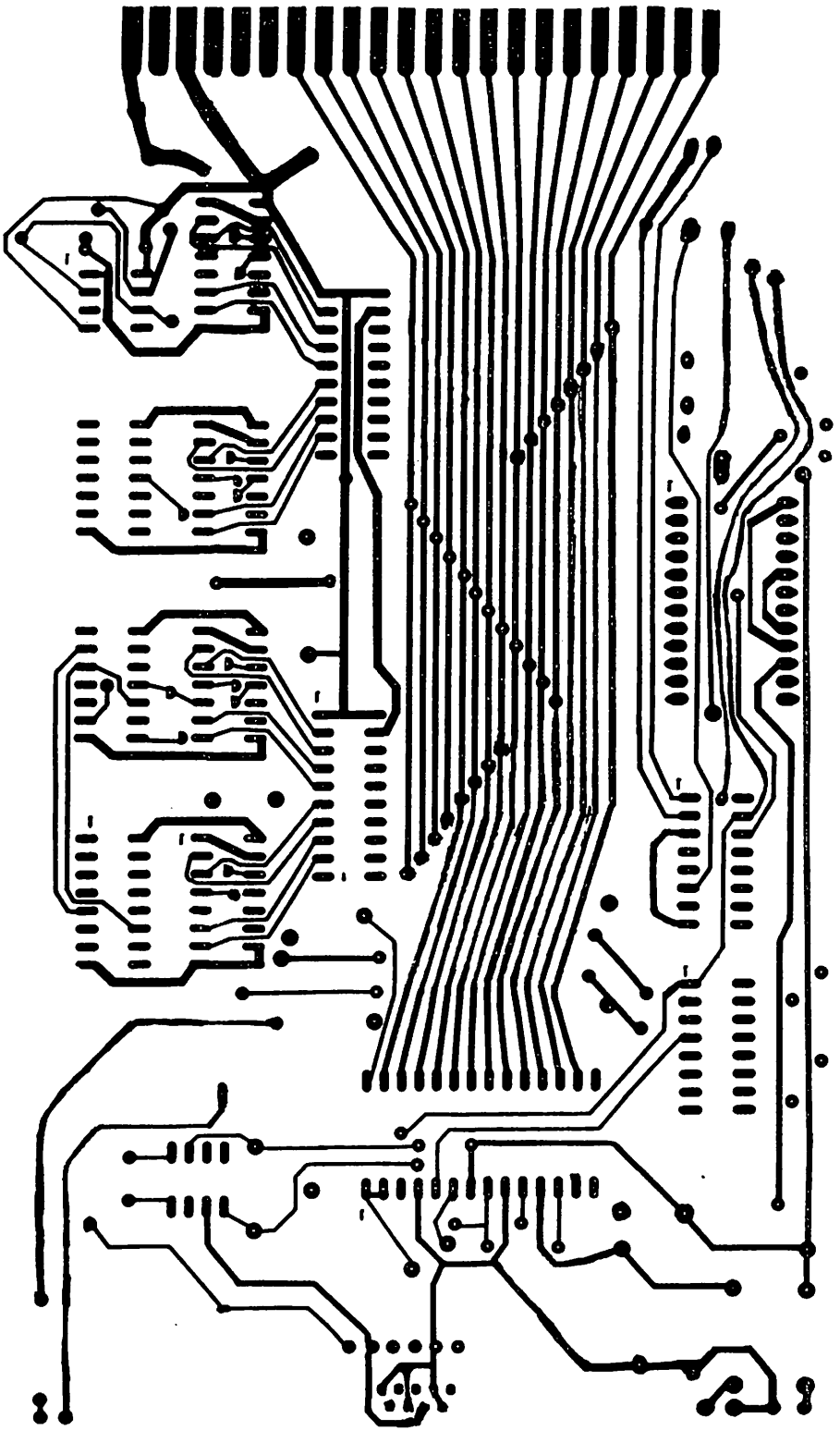


Figure 2.27: New (untested) etching transparency – underside

Item	*Description*	*Qty.*
D/A Converter	Analog Devices - AD3860	1
A/D Converter	Analog Devices - AD574	1
Binary Counters	74LS191	4
One Shot	74LS221	1
1 of 8 Decoder	74LS138	1
8–bit Buffers	74LS245	2
Inverters	74LS04	1
D–Flip-flops	74LS174	1
D–Flip-flops	74LS74	1
Clock Pulse Generator	TLC551	1
Op Amp	Texas Instruments TL081	1
D–Subminiature connector	9–pin, right–angle bracket	1
LEDs	standard	4
Switch	momentary micro SPDT	1
Transistor	NPN, signal	1
±1% DALE RN60D Resistors	49.9 Ohms	2
±5% 1/4 watt Resistors	240 Ohms	2
	270 Ohms	1
	330 Ohms	1
	1K Ohms	1
	1.5K Ohms	2
	2.4K Ohms	1
Capacitors	47 PFD mica	1
	.01 MFD mica	2
	1 MFD mica	7
	3.3 MFD, 10V electrolytic	2
	47 MFD, 25V electrolytic	4

Table 2.2: Original STIOB – Component List

Board Section	*Modifications necessary*
DAC	Cut lines to pins 1 through 12. Jumper: pins 1 through 12 to data I/O lines 11 through 0, respectively.
ADC	Cut lines to pins 3, 14 and 16 through 27. Jumper: pin 3 to pin 1 of the 74LS221 One–Shot, pin 11 to a –15V supply line, pin 13 to pin 6 of the TL081 operational amplifier, and pins 27 through 16 to data I/O lines 0 through 11, respectively.
74LS138	Cut the line to pin 3. Jumper: pin 3 to pin 8 (GND), pin 13 to edge–connector contact 31, and pin 15 to edge–connector contact 30.
DB-9 connector	Cut the lines to pins 6,7 and 8. Jumper: pin 6 to a +5V supply line, pin 7 to DB–9 pin 4, and pin 8 to pin 3 of the TL081 operational amplifier.
LEDs	Cut the ground line between the Wrap–around and +5V indicator LEDs and jumper +5V to the disconnected pin of the Wrap–around LED.
Capacitors	To correct the capacitor placement errors, it is necessary to make some at least two PC board modifications. One must drill holes in a couple of places in order to make connections to the underside of the board. By the DAC, leads are soldered directly onto PC runs. Study the Physical Layout diagram for the original board and the etching diagrams for the new boards to determine how and where the plus and minus 15V filter capacitors are wired–in. The Physical Layout diagram shows how the original board looks after all the design corrections have been made.
Switch	To use the Wrap–around detection circuit, mount the reset switch as shown on the board layout for the origin board. Then wire the leads into the solder holes marked SW1 on the PC board.

Table 2.3: Required modifications of original boards

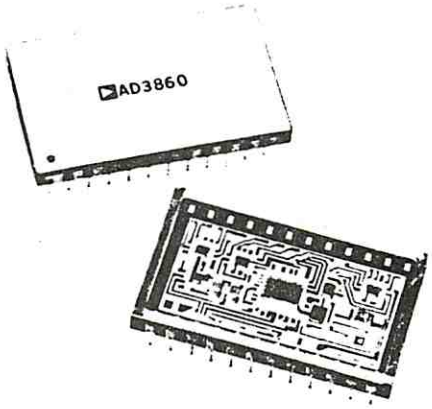
MANUFACTURER'S DOCUMENTATION FOLLOWS.

* Reprinted by permission from Analog Devices

ANALOG DEVICES

Complete, Voltage Output 12-Bit Buffered DAC

FEATURES
Resolution: 12 Bits
Nonlinearity: ±1/2LSB T_{min} to T_{max}
12-Bit Input Register
Small Size: 24 pin DIP
Fast Settling: 5μs to ±0.01%
Internal Reference
Internal Output Amplifier



PRODUCT DESCRIPTION

The AD3860 is a precision 12-bit D/A converter designed for direct interface to microprocessors.

The functional diagram shows that the AD3860 consists of a 12-bit input storage register, a 12-bit DAC, internal reference, and a fast output amplifier. It is TTL compatible and the register enable facilitates deglitching and microprocessor interfacing. The low noise, high stability subsurface zener diode is used to produce a reference voltage with excellent long term stability, external current capability and temperature drift characteristics. The output amplifier gives the user a voltage output and combines with the other features of this circuit to produce a functionally complete digital to analog converter.

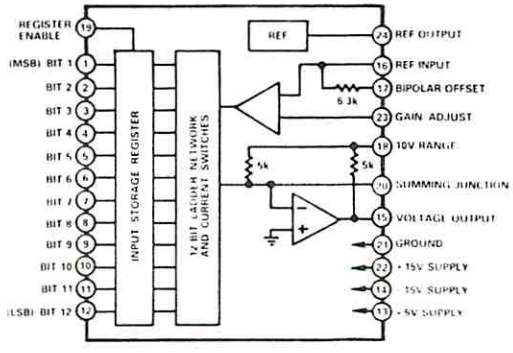
The AD3860 is laser trimmed to achieve ±1/4LSB linearity typical and ±1/2LSB maximum over the full operating temperature range. The low T.C. Binary ladder guarantees that the AD3860 will be monotonic over the specified temperature range.

The AD3860 is available in two versions. The AD3860K is specified for use over 0 to +70°C temperature range. The AD3860S is specified for the −55°C to +125°C temperature range and is especially recommended for high reliability needs in harsh environments. The AD3860S is available processed to MIL-STD-883, Level B. All units are in supplied in 24-pin, hermetically-sealed ceramic DIPs.

PRODUCT HIGHLIGHTS

1. The AD3860 is a functionally complete voltage output DAC with voltage reference, digital latches, and output amplifier in a single hybrid package.
2. The input buffer latches permit interface to microprocessor data buses. All logic inputs are TTL or 5 volt CMOS compatible.
3. Laser trimming the thin-film resistors assures superior linearity and accuracy stability over temperature. Both commercial and military temperature range models have ±1/2LSB linearity maximum guaranteed over the full operating temperature range.
4. Monotonicity is also guaranteed over the full operating temperature range. The typical full scale temperature coefficient is 10ppm/°C.
5. The precision buried zener reference can supply up to 2.5mA for use elsewhere in the application.
6. The fast output amplifier provides a voltage output with a 5μs settling time to 0.01% for a 20 volt step. The AD3860 is designed for military and industrial applications where high speed D/A conversion is required.

AD3860 FUNCTIONAL BLOCK DIAGRAM



P.O. Box 280; Norwood, Massachusetts 02062 U.S.A.
Tel:617/329-4700 Twx: 710/394-6577
Telex: 924491 Cables: ANALOG NORWOODMASS

* Reprinted with permission from Analog Devices

SPECIFICATIONS (typical @ +25°C, rated power supplies unless otherwise noted)

Model	AD3860K	AD3860S[1]
DIGITAL INPUTS		
Resolution	12 Bits	*
Logic Coding: Unipolar Ranges	Complementary Straight Binary	*
Bipolar Ranges	Complementary Offset Binary	*
Logic Levels (TTL Compatible): Logic "1"	+2.0V dc min, +5.5V dc max	*
Logic "0"	0V dc min, 0.8V dc max	*
Input Currents		
Data Inputs: Logic "1"	30μA max	*
Logic "0"	0.6mA max	*
Register Enable: Logic "1"	60μA max	*
Logic "0"	−1.2mA max	*
ANALOG OUTPUT		
Output Impedance	0.5Ω	*
Output Current @ Z_L = 2kΩ‖250pF	±10mA, ±5mA min	*
ACCURACY		
Linearity Error (T_{min} to T_{max})	±1/4LSB[2], ±1/2LSB max	±1/2LSB max
Differential Linearity Error	±1/2LSB, ±1LSB max	±1LSB
Monotonicity	Guaranteed Over Temperature	*
Full Scale Absolute Accuracy Error[3]	±0.05% FSR[4], ±0.1% FSR max	*
T_{min} to T_{max}	±0.15% FSR, ±0.3% FSR max	*
Zero Error	±0.025% FSR, ±0.05% FSR max	*
T_{min} to T_{max}	±0.05% FSR, ±0.1% FSR max	*
Gain Error	±0.1%	*
DRIFT		
Gain	±10ppm/°C	*
Offset	±5ppm/°C	*
DYNAMIC CHARACTERISTICS		
Settling Time to ±0.01% for: 20V Step	5μs, 7μs max	*
10V Step	3μs, 5μs max	*
Output Slew Rate	20V/μs	*
Register Enable[5]		
Pulse Width	60ns min	*
Setup Time Digital Data to Enable	40ns min	*
INTERNAL REFERENCE VOLTAGE		
Voltage	+6.3V	*
Accuracy	±2%	*
External Current	2.5mA max	*
POWER SUPPLIES		
Power Supply Range: +15V Supply	+14.55V min, +15.45V max	*
−15V Supply	−14.55V min, −15.45V max	*
+5V Supply	+4.75V min, +5.25V max	*
Power Supply Rejection: +15V Supply	±0.002% FSR/%V_S ±0.01% FSR/%V_S max	*
−15V Supply	±0.002% FSR/%V_S ±0.004% FSR/%V_S max	*
Current Drain: +15V Supply	10mA, 20mA max	*
−15V Supply	−12mA, −30mA max	*
+5V Supply	30mA, 50mA max	*
Power Consumption	675mW, 1W max	*
TEMPERATURE RANGE		
Operating	0 to +70°C	−55°C to +125°C
Storage	−65°C to +150°C	*

ABSOLUTE MAXIMUM RATINGS

+15 Volt Supply (pin 22) . . . . . +18V
−15 Volt Supply (pin 14) . . . . . −18V
+5 Volt Supply (pin 13) . . −0.5V to +7V
Register Enable (pin 19) . −0.5V to +5.5V
Digital Inputs (pins 1–12) . −0.5V to +5.5V

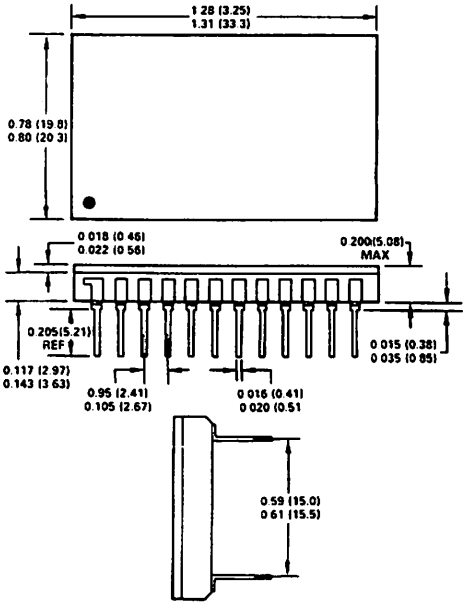
PIN CONFIGURATION

24 LEAD DUAL IN-LINE-PACKAGE

PIN NO.	FUNCTION
1	BIT 1 (MSB)
2	BIT 2
3	BIT 3
4	BIT 4
5	BIT 5
6	BIT 6
7	BIT 7
8	BIT 8
9	BIT 9
10	BIT 10
11	BIT 11
12	BIT 12 (LSB)
13	LOGIC SUPPLY
14	$-V_S$
15	V_{OUT}
16	REF INPUT
17	BIPOLAR OFFSET
18	10V RANGE
19	REGISTER ENABLE
20	SUMMING JUNCTION
21	COMMON
22	$+V_S$
23	GAIN ADJUST
24	6.3V_{REF} OUT

OUTLINE DIMENSIONS

Dimensions shown in inches and (mm).



NOTES:
[1]AD3860S is available processed to MIL-STD-883, Level B.
[2]Least Significant Bit (LSB).
[3]Absolute Accuracy Error includes gain, offset, linearity, noise and all other errors and is specified without adjustment.
[4]FSR is Full Scale Range and is 20 V for ±10 range.
[5]The AD3860's analog output will follow its digital input when register enable is a logic "0". Digital Input date will be latched and analog output voltage constant when register enable is a logic "1".
*Same as AD3860K.
Specifications subject to change without notice.

* Reprinted by permission from Analog Devices

APPLICATIONS INFORMATION

OUTPUT VOLTAGE RANGE SELECTION

Output Range	0 to +10V	±5V	±10V
Pin Connection			
Connect Pin 24 to	16	16	16
Connect Pin 17 to	21	20	20
Connect Pin 15 to	18	18	NC
Connect Pin 20 to	NC	17	17

INPUT LOGIC CODING

Digital Input MSB LSB	Analog Output 0 to +10V	±5V	±10V
0000 0000 0000	+9.9976V	+4.9976V	+9.9951V
0000 0000 0001	+9.9951V	+4.9951V	+9.9902V
0111 1111 1111	+5.0000V	0.0000V	0.0000V
1000 0000 0000	+4.9976V	−0.0024V	−0.0049V
1111 1111 1110	+0.0024V	−4.9976V	−9.9951V
1111 1111 1111	0.0000V	−5.0000V	−10.0000V

CODING NOTES:

1. For unipolar operation, the coding complementary straight binary (CSB).
2. For bipolar operation, the coding complementary offset binary (COB).
3. For FSR = 20V, 1LSB = 4.88mV.
4. For FSR = 10V, 1LSB = 2.44mV.

Layout Considerations

Proper layout and decoupling is necessary to obtain the AD3860's specified accuracy. Ground (pin 21) must be tied to circuit analog ground as close to the package as possible. Grounding through a large ground plane beneath the package is preferred.

Power supplies should be decoupled with electrolytic or tantalum capacitors near the unit. A 1μF capacitor is parallel with a 0.01μF ceramic capacitor on all supplies is recommended, see Figure 1.

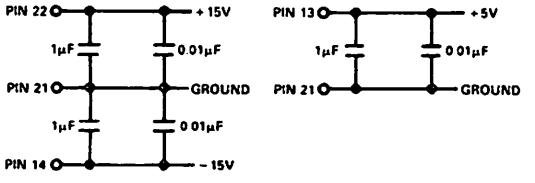


Figure 1. Power Supply Decoupling

Coupling between analog and digital signals should be minimized to avoid noise pick up. Use short jumpers to tie the reference output (pin 24) to the reference input (pin 16) and to tie the bipolar offset (pin 17) to the summing junction (pin 20).

If the external full scale and zero adjustments are used, the series 6.8MΩ resistors should be placed as close to the unit as possible.

Reference Output

The AD3860 is laser trimmed to operate from the internal 6.3 volt voltage reference. The user has the option of supplying an external reference but for specified operation the reference output (pin 24) must be connected to the reference input (pin 16). The internal reference can be used to drive an external load, but it should be buffered if load current will exceed 2.5mA.

Optional Full Scale and Zero (−Full Scale) Adjustments

The AD3860 will operate as specified without adjustment, however, absolute accuracy error can be reduced to ±1LSB by trimming as described below. Adjustments should be made after warmup. As shown in Figures 2 and 3 the zero

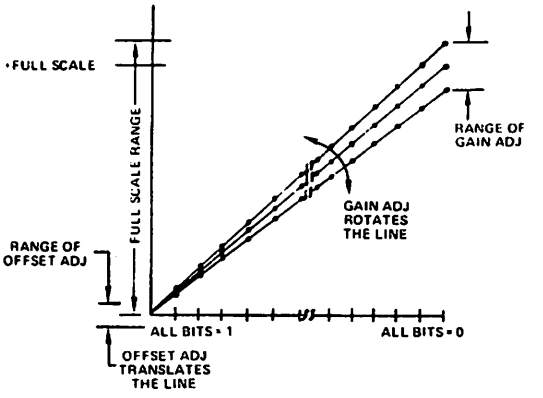


Figure 2. Relationship of OFFSET and GAIN Adjustments for a UNIPOLAR D/A Converter (Input, Horizontal; Output, Vertical)

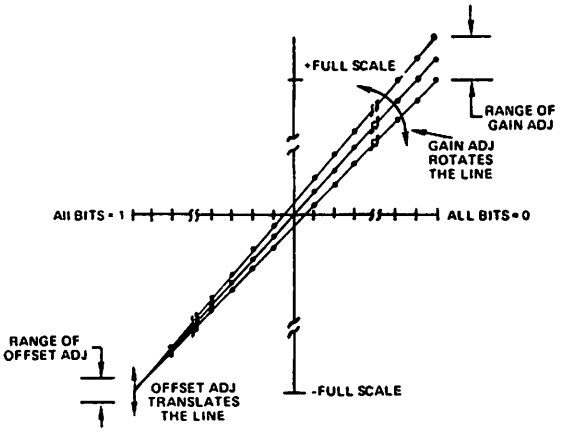
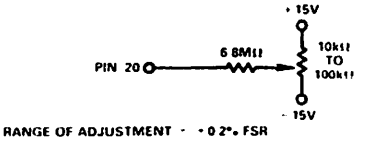


Figure 3. Relationship of OFFSET and GAIN Adjustments for a BIPOLAR D/A Converter (Input, Horizontal; Output, Vertical)

(−full scale) adjustment should be made before the full scale adjustment. We recommend multiturn potentiometers with maximum temperature coefficients of 100ppm/°C. Series resistors can be ±20% carbon composition or better. If these adjustments are not used pins 20 and 23 should not be grounded.

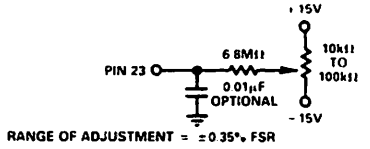
Zero (−Full Scale) Adjustment

Connect the potentiometer as shown and apply all "1s" to the digital inputs. Adjust the potentiometer until the analog output is equal to zero volts for unipolar output ranges and minus full scale for bipolar output ranges.



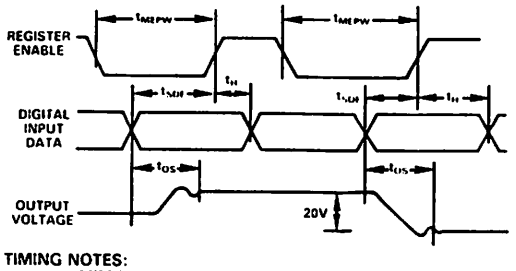
Full Scale Adjustment

Connect the potentiometer as shown and apply all "0s" to the digital inputs. Adjust the potentiometer for maximum chosen analog output.



REGISTER ENABLE

When the register enable (pin 19) is high (hold mode) the digital data in the input register will be latched. When the register enable is low (track mode) the converter's output will follow its input. To latch new digital data into the register, the register enable must go low for a minimum of 60ns and the digital input data must be valid for a minimum of 40ns before the register enable goes high again. See the timing diagram below.



TIMING NOTES:

t_{MEPW} MIMINUM ENABLE PULSE WIDTH IS 60ns.

t_{SDE} MINIMUM SETUP TIME DIGITAL INPUT DATA TO ENABLE IS 40ns.

t_H HOLD TIME IS DEFINED AS THE REQUIRED DELAY BETWEEN THE LEADING EDGE OF REGISTER ENABLE AND THE END OF VALID INPUT DATA. THE HOLD TIME IS ZERO FOR THE AD3860.

t_{OS} OUTPUT SETTLING TIME FOR A 20 VOLT CHANGE TO ±1/2LSB IS 7μs MAX.

Figure 4. Input Register Timing Diagram

8-BIT MICROPROCESSOR INTERFACE

Whenever a 12-bit DAC is loaded from an 8-bit bus, two write cycles are required. The organization most often used is "right justified." Right-justified data calls for the eight least significant bits to occupy one byte, with the four most significant bits residing in the lower half of another byte. This organization simplifies integer arithmetic. Figure 5 shows an addressing

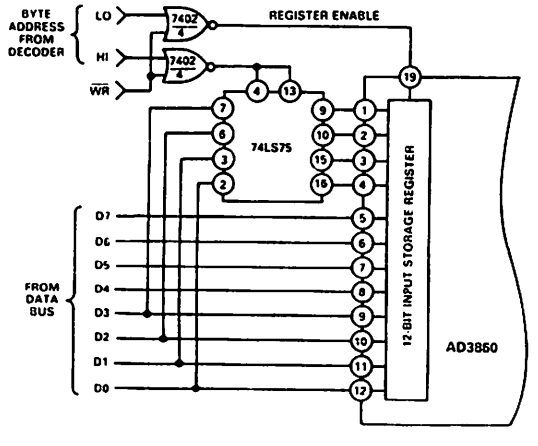


Figure 5. Right-Justified 8-Bit Bus Interface

scheme for the AD3860 set up for right justified data in an 8-bit system. The four MSBs are latched into the 74LS75 latch in the first write cycle. The entire 12-bit word is then loaded into the AD3860's internal input storage register on the next write cycle. An alternate scheme is to use an eight-bit intermediate register, such as the 74LS373, to allow the user to load the lower order bits in the first write cycle.

Left-justified data can be similarly accommodated. The overlapping of data lines is reversed as shown in Figure 6. The AD3860

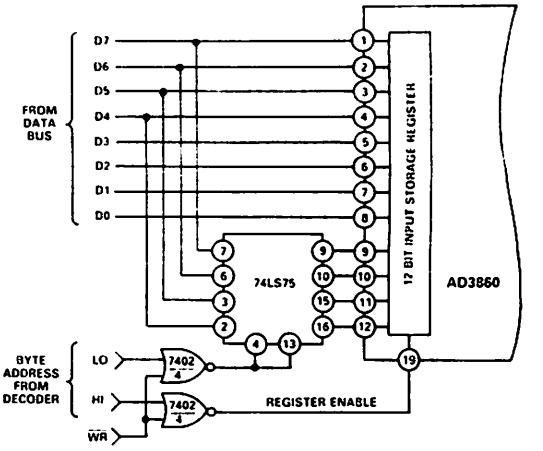


Figure 6. Left-Justified 8-Bit Bus Interface

still occupies two adjacent locations in the processor's memory map. A left-justified format is convenient in applications when the data represents a 12-bit binary fraction (between 0 and $\frac{4095}{4096}$).

Left-justified data has the four least significant bits in the upper half of the first byte and the eight most significant bits in the second byte. The four LSBs on the intermediate latch and the eight MSBs on the data bus are all latched into the AD3860s latch simultaneously. This double buffering technique avoids the analog output slewing to an undesirable state determined by the MSBs of the new digital data and the LSBs of the previous digital data.

Many of the popular microprocessor families include components specifically designed to ease the interface between the microprocessor and a peripheral device such as a converter. These components are called Programmable Peripheral Interface (PPI), Peripheral Interface Adaptor (PIA), Parallel I/O Controller (PIO), or similar names. They typically feature two or more 8-bit wide parallel data ports which can, under program control, be configured as either inputs or outputs. Their control signals are made compatible with the particular processor they serve, and in many systems can provide an attractive alternative to a collection of random logic. For example, the 8255 PPI has two 8-bit and two 4-bit ports which can be used as input, as output, or as a combination of input, output, and control. Each of the 4-bit words can be grouped with one of the 8-bit words so that the interface is split into two 12-bit ports. The ports can be set up as outputs, under program control, for controlling two AD3860s with a single PPI. The 8255 contains two bits of address input. That is, A0 and A1 of the 8255 are driven directly by the address bus, and these bits need not be used by the address decoder. Though the 8255 is an 8080 system component, it is adaptable to other μP systems.

USING THE AD3860 WITH 12- AND 16-BIT BUSES

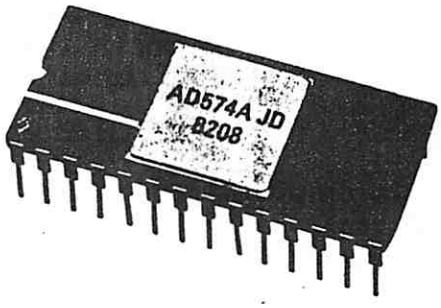
The AD3860 is easily interfaced to 12- and 16-bit data buses. The AD3860's Register Enable signal can usually be derived by NANDing the desired address lines with the processor's MEMORY WRITE or I/O WRITE line. For most processors, valid data remains on the data bus for some time after either the valid address or control signals are removed. Therefore, the data is latched into the AD3860 immediately after one of the address or control signals changes but before valid data goes away. The AD3860 thus occupies a single memory location.

* Reprinted by permission from Analog Devices

Fast, Complete 12-Bit A/D Converter with Microprocessor Interface

FEATURES

- Complete 12-Bit A/D Converter with Reference and Clock
- Full 8- or 16-Bit Microprocessor Bus Interface
- 250ns Bus Access Time
- Guaranteed Linearity Over Temperature
  - 0 to +70°C – AD574AJ, AK, AL
  - −55°C to +125°C – AD574AS, AT, AU
- No Missing Codes Over Temperature
- Fast Successive Approximation Conversion – 25μs
- Buried Zener Reference for Long-Term Stability and Low Gain T.C. 10ppm/°C max AD574AL 12.5ppm/°C max AD574AU
- Low Profile 28-Pin Ceramic DIP
- Low Power: 390mW



PRODUCT DESCRIPTION

The AD574A is a complete 12-bit successive-approximation analog-to-digital converter with 3-state output buffer circuitry for direct interface to an 8-, 12-or 16-bit microprocessor bus. The AD574A design is implemented with two LSI chips each containing both analog and digital circuitry, resulting in the maximum performance and flexibility at the lowest cost.

One chip is the high performance AD565A 12-bit DAC and voltage reference. It contains the high speed current output switching circuitry, laser-trimmed thin film resistor network, low T.C. buried zener reference and the precision input scaling and bipolar offset resistors. This chip is laser-trimmed at the wafer stage (LWT) to adjust ladder network linearity, voltage reference tolerance and temperature coefficient, and the calibration accuracy of input scaling and bipolar offset resistors.

The second chip uses the proven LCI (linear-compatible integrated injection logic) process to provide the low-power I^2L successive-approximation register, converter control circuitry, clock, bus interface, and the high performance latching comparator. The precision, low-drift comparator is adjusted for initial input offset error at the wafer stage by the "zener-zap" technique which trims the comparator input stage to 1/10 LSB typical error. This form of trimming, while cumbersome for complex ladder networks, is an attractive alternative to thin film resistor trimming for a simple offset adjustment and eliminates the need for thin film processing for this portion of the circuitry.

The AD574A is available in six different grades. The AD574AJ, AK, and AL grades are specified for operation over the 0 to +70°C temperature range. The AD574AS, AT, and AU are specified for the −55°C to +125°C range. All grades are packaged in a low-profile, 0.600 inch wide, 28-pin hermetically-sealed ceramic DIP.

PRODUCT HIGHLIGHTS

1. The AD574A interfaces to most popular microprocessors with an 8-, 12-, or 16-bit bus without external buffers or peripheral interface controllers. Multiple-mode three-state output buffers connect directly to the data bus while the read and convert commands are taken from the control bus. The 12-bits of output data can be read either as one 12-bit word or as two 8-bit bytes (one with 8 data bits, the other with 4 data bits and 4 trailing zeros).
2. The precision, laser-trimmed scaling and bipolar offset resistors provide four calibrated ranges, 0 to +10 and 0 to +20 volts unipolar, or −5 to +5 and −10 to +10 volts bipolar. Typical bipolar offset and full scale calibration of ±0.1% can be trimmed to zero with one external component each.
3. The internal buried zener reference is trimmed to 10.00 volts with 1% maximum error and 15ppm/°C typical T.C. The reference is available externally and can drive up to 1.5mA beyond that required for the reference and bipolar offset resistors.

P.O. Box 280; Norwood, Massachusetts 02062 U.S.A.
Tel:617/329-4700 Twx: 710/394-6577
Telex: 924491 Cables: ANALOG NORWOODMASS

* Reprinted with permission from Analog Devices

SPECIFICATIONS (@ = 25°C with V_{CC} = +15V, V or +12V, V_{LOGIC} = +5V, V_{EE} = −15V or −12V unless otherwise indicated)

Model	AD574AJ Min	AD574AJ Typ	AD574AJ Max	AD574AK Min	AD574AK Typ	AD574AK Max	AD574AL Min	AD574AL Typ	AD574AL Max	Units
RESOLUTION			12			12			12	Bits
LINEARITY ERROR										
25°C max)			±1			±1/2			±1/2	LSB
T_{min} to T_{max}			±1			±1/2			±1	LSB
DIFFERENTIAL LINEARITY ERROR										
(Minumum resolution for which no missing codes are guaranteed)										
25°C	11			12			12			Bits
T_{min} to T_{max}	11			12			12			Bits
UNIPOLAR OFFSET (max) (Adjustable to zero)			±2			±2			±2	LSB
BIPOLAR OFFSET (max) (Adjustable to zero)			±10			±4			±4	LSB
FULL SCALE CALIBRATION ERROR (with fixed 50Ω resistor from REF OUT TO REF IN)										
(Adjustable to zero) 25°C (max)			0.25			0.25			0.25	% of F.S.
T_{min} to T_{max} (Without Initial Adjustment)		0.47			0.37			0.30		% of F.S.
(With Initial Adjustment)		0.22			0.12			0.05		% of F.S.
TEMPERATURE RANGE	0		+70	0		+70	0		+70	°C
TEMPERATURE COEFFICIENTS (Using internal reference)										
T_{min} to T_{max}										
Unipolar Offset			±2			±1			±1	LSB
			10			5			5	ppm/°C
Bipolar Offset			±2			±1			±1	LSB
			10			5			5	ppm/°C
Full Scale Calibration			±9			±5			±2	LSB
			50			27			10	ppm/°C
POWER SUPPLY REJECTION										
Max change in Full Scale Calibration										
$+13.5 \le V_{CC} \le +16.5V$ or $+11.4V \le V_{CC} \le +12.6V$			±2			±1			±1	LSB
$+4.5 \le V_{LOGIC} \le +5.5V$			±1/2			±1/2			±1/2	LSB
$-16.5 \le V_{EE} \le -13.5V$ or $-12.6V \le V_{EE} \le -11.4V$			±2			±1			±1	LSB
ANALOG INPUT										
Input Ranges										
Bipolar		−5 to +5			−5 to +5			−5 to +5		Volts
		−10 to +10			−10 to +10			−10 to +10		Volts
Unipolar		0 to +10			0 to +10			0 to +10		Volts
		0 to +20			0 to +20			0 to +20		Volts
Input Impedance										
10 Volt Span	3	5	7	3	5	7	3	5	7	kΩ
20 Volt Span	6	10	14	6	10	14	6	10	14	kΩ
POWER SUPPLIES										
Operating Range										
V_{LOGIC}	+4.5		+5.5	+4.5		+5.5	+4.5		+5.5	Volts
V_{CC}	+11.4		+16.5	+11.4		+16.5	+11.4		+16.5	Volts
V_{EE}	−11.4		−16.5	−11.4		−16.5	−11.4		−16.5	Volts
Operating Current										
I_{LOGIC}		30	40		30	40		30	40	mA
I_{CC}		2	5		2	5		2	5	mA
I_{EE}		18	30		18	30		18	30	mA
POWER DISSIPATION		390	725		390	725		390	725	mW
INTERNAL REFERENCE VOLTAGE	9.9	10.0	10.1	9.9	10.0	10.1	9.9	10.0	10.1	Volts
Output current (available for external loads)			1.5[1]			1.5[1]			1.5[1]	mA
(External load should not change during conversion)										

NOTES

[1]The reference should be buffered for operation on ± 12V supplies.

Specifications subject to change without notice.

Specifications shown in boldface are tested on all production units at final electrical test. Results from those tests are used to calculate outgoing quality levels. All min and max specifications are guaranteed, although only those shown in boldface are tested on all production units.

* Reprinted by permission from Analog Devices

Model	AD574AS Min	AD574AS Typ	AD574AS Max	AD574AT Min	AD574AT Typ	AD574AT Max	AD574AU Min	AD574AU Typ	AD574AU Max	Units
RESOLUTION			12			12			12	Bits
LINEARITY ERROR										
25°C (max)			±1			±1/2			±1/2	LSB
T_{min} to T_{max}			±1			±1/2			±1	LSB
DIFFERENTIAL LINEARITY ERROR (Minumum resolution for which no missing codes are guaranteed)										
25°C	11			12			12			Bits
T_{min} to T_{max}	11			12			12			Bits
UNIPOLAR OFFSET (max) (Adjustable to zero)			±2			±2			±2	LSB
BIPOLAR OFFSET (max) (Adjustable to zero)			±10			±4			±4	LSB
FULL SCALE CALIBRATION ERROR (with fixed 50Ω resistor from REF OUT TO REF IN)										
(Adjustable to zero) 25°C (max)			0.25			0.25			0.25	% of F.S.
T_{min} to T_{max} (Without Initial Adjustment)		0.75			0.5			0.37		% of F.S.
(With Initial Adjustment)		0.5			0.25			0.12		% of F.S.
TEMPERATURE RANGE	−55		+125	−55		+125	−55		+125	°C
TEMPERATURE COEFFICIENTS (Using internal reference)										
T_{min} to T_{max}										
Unipolar Offset			±2			±1			±1	LSB
			5			2.5			2.5	ppm/°C
Bipolar Offset			±4			±2			±1	LSB
			10			5			2.5	ppm/°C
Full Scale Calibration			±20			±10			±5	LSB
			50			25			12.5	ppm/°C
POWER SUPPLY REJECTION										
Max change in Full Scale Calibration										
$+13.5 \le V_{CC} \le +16.5V$ or $+11.4V \le V_{CC} \le +12.6V$			±2			±1			±1	LSB
$+4.5 \le V_{LOGIC} \le +5.5V$			±1/2			±1/2			±1/2	LSB
$-16.5 \le V_{EE} \le -13.5V$ or $-12.6V \le V_{EE} \le -11.4V$			±2			±1			±1	LSB
ANALOG INPUT										
Input Ranges										
Bipolar		−5 to +5			−5 to +5			−5 to +5		Volts
		−10 to +10			−10 to +10			−10 to +10		Volts
Unipolar		0 to +10			0 to +10			0 to +10		Volts
		0 to +20			0 to +20			0 to +20		Volts
Input Impedance										
10 Volt Span	3	5	7	3	5	7	3	5	7	kΩ
20 Volt Span	6	10	14	6	10	14	6	10	14	kΩ
POWER SUPPLIES										
Operating Range										
V_{LOGIC}	+4.5		+5.5	+4.5		+5.5	+4.5		+5.5	Volts
V_{CC}	+11.4		+16.5	+11.4		+16.5	+11.4		+16.5	Volts
V_{EE}	−11.4		−16.5	−11.4		−16.5	−11.4		−16.5	Volts
Operating Current										
I_{LOGIC}		30	40		30	40		30	40	mA
I_{CC}		2	5		2	5		2	5	mA
V_{EE}		18	30		18	30		18	30	mA
POWER DISSIPATION		390	725		390	725		390	725	mW
INTERNAL REFERENCE VOLTAGE	9.9	10.0	10.1	9.9	10.0	10.1	9.9	10.0	10.1	Volts
Output current (available for external loads) (External load should not change during conversion)			1.5[1]			1.5[1]			1.5[1]	mA

NOTES

[1]The reference should be buffered for operation on ± 12V supplies.

Specifications subject to change without notice.

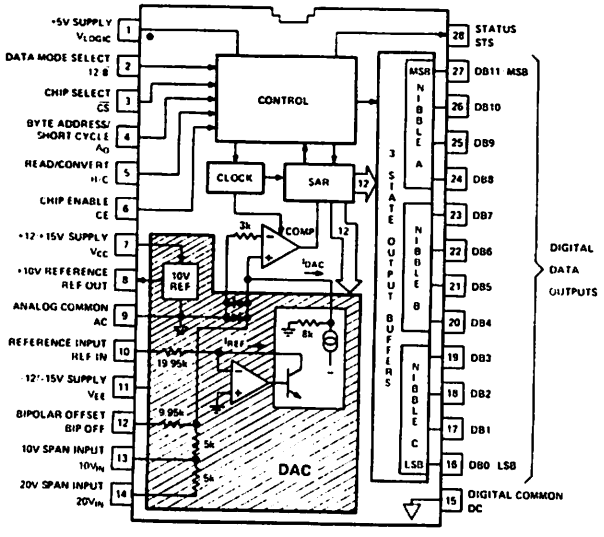
Specifications shown in boldface are tested on all production units at final electrical test. Results from those tests are used to calculate outgoing quality levels. All min and max specifications are guaranteed, although only those shown in boldface are tested on all production units.

* Reprinted by permission of Analog Devices

DIGITAL CHARACTERISTICS[1] (All grades, T_{min} – T_{max})

	Min	Typ	Max	
Logic Inputs[2] (CE, $\overline{CS}$, R/$\overline{C}$, A_0)				
Voltages				
Logic "1"	+2.0V		+5.5V	
Logic "0"	−0.5V		+0.8V	
Current	50μA		+50μA	
Capacitance		5pF		
Logic Outputs (DB11–DB0, STS)				
Logic "0"			+0.4V	$I_{SINK} \leq 1.6mA$
Logic "1"	2.4V			$I_{SOURCE} \leq 500\mu A$
Leakage (When in high-Z state)	−40μA		+40μA	DB11 – DB0 Only
Capacitance		5pF		

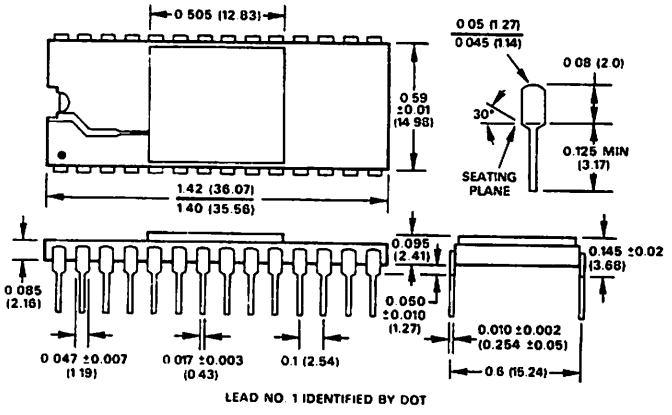
[1]Detailed Timing Specifications appear in the Digital Interface Section.
[2]12/$\overline{8}$ Input is not TTL-compatible and must be hard-wired to V_{LOGIC} or DIGITAL COMMON.



AD574A Block Diagram and Pin Configuration

AD574A PACKAGE OUTLINE

Dimensions shown in inches and (mm).



ABSOLUTE MAXIMUM RATINGS

(Specifications apply to all grades, except where noted)

V_{CC} to Digital Common	0 to +16.5V
V_{EE} to Digital Common	0 to −16.5V
V_{LOGIC} to Digital Common	0 to +7V
Analog Common to Digital Common	±1V
Control Inputs (CE, $\overline{CS}$, A_0, 12/$\overline{8}$, R/$\overline{C}$) to Digital Common	−0.5V to V_{LOGIC} +0.5V
Analog Inputs (REF IN, BIP OFF, $10V_{IN}$) to Analog Common	±16.5V
$20V_{IN}$ to Analog Common	±24V
REF OUT	Indefinite short to common; Momentary short to V_{CC}
Chip Temperature (J, K, L grades)	100°C
(S, T, U grades)	150°C
Power Dissipation	1000mW
Lead Temperature, Soldering	300°C, 10 sec.
Storage Temperature	−65°C to +150°C
Thermal Resistance, θ_{JA}	60°C/W

AD574A ORDERING GUIDE

Model	Temp. Range	Linearity Error Max (T_{min} to T_{max})	Resolution No Missing Codes (T_{min} to T_{max})	Max Full Scale T.C. (ppm/°C)
AD574AJD	0 to +70°C	±1LSB	11 Bits	50.0
AD574AKD	0 to +70°C	±1/2LSB	12 Bits	27.0
AD574ALD	0 to +70°C	±1/2LSB	12 Bits	10.0
AD574ASD	−55°C to +125°C	±1LSB	11 Bits	50.0
AD574ATD	−55°C to +125°C	±1LSB	12 Bits	25.0
AD574AUD	−55°C to +125°C	±1LSB	12 Bits	12.5

* Reprinted with permission from Analog Devices

THE AD574A OFFERS GUARANTEED MAXIMUM LINEARITY ERROR OVER THE FULL OPERATING TEMPERATURE RANGE

DEFINITIONS OF SPECIFICATIONS

LINEARITY ERROR
Linearity error refers to the deviation of each individual code from a line drawn from "zero" through "full scale". The point used as "zero" occurs ½LSB (1.22mV for 10 volt span) before the first code transition (all zeros to only the LSB "on"). "Full scale" is defined as a level 1½LSB beyond the last code transition (to all ones). The deviation of a code from the true straight line is measured from the middle of each particular code.

The AD574AK, AL, AT, and AU grades are guaranteed for maximum nonlinearity of ±½LSB. For these grades, this means that an analog value which falls exactly in the center of a given code width will result in the correct digital output code. Values nearer the upper or lower transition of the code width may produce the next upper or lower digital output code. The AD574AJ and AS grades are guaranteed to ±1LSB max error. For these grades, an analog value which falls within a given code width will result in either the correct code for that region or either adjacent one.

Note that the linearity error is not user-adjustable.

DIFFERENTIAL LINEARITY ERROR (NO MISSING CODES)
A specification which guarantees no missing codes requires that every code combination appear in a monotonic increasing sequence as the analog input level is increased. Thus every code must have a finite width. For the AD574AK, AL, AT, and AU grades, which guarantee no missing codes to 12-bit resolution, all 4096 codes must be present over the entire operating temperature ranges. The AD574AJ and AS grades guarantee no missing codes to 11-bit resolution over temperature; this means that all code combinations of the upper 11 bits must be present; in practice very few of the 12-bit codes are missing.

UNIPOLAR OFFSET
The first transition should occur at a level ½LSB above analog common. Unipolar offset is defined as the deviation of the actual transition from that point. This offset can be adjusted as discussed on the following two pages. The unipolar offset temperature coefficient specifies the maximum change of the transition point over temperature, with or without external adjustment.

BIPOLAR OFFSET
Similarly, in the bipolar mode, the major carry transition (0111 1111 1111 to 1000 0000 0000) should occur for an analog value ½LSB below analog common. The bipolar offset error and temperature coefficient specify the initial deviation and maximum change in the error over temperature.

QUANTIZATION UNCERTAINTY
Analog-to-digital converters exhibit an inherent quantization uncertainty of ±½LSB. This uncertainty is a fundamental characteristic of the quantization process and cannot be reduced for a converter of given resolution.

LEFT-JUSTIFIED DATA
The data format used in the AD574A is left-justified. This means that the data represents the analog input as a fraction of full-scale, ranging from 0 to $\frac{4095}{4096}$. This implies a binary point to the left of the MSB.

FULL SCALE CALIBRATION ERROR
The last transition (from 1111 1111 1110 to 1111 1111 1111) should occur for an analog value 1 1/2LSB below the nominal full scale (9.9963 volts for 10.000 volts full scale). The full scale calibration error is the deviation of the actual level at the last transition from the ideal level. This error, which is typically 0.05 to 0.1% of full scale, can be trimmed out as shown in Figures 3 and 4. The full scale calibration error over temperature is given with and without the initial error trimmed out. The temperature coefficients for each grade indicate the maximum change in the full scale gain from the initial value using the internal 10 volt reference.

TEMPERATURE COEFFICIENTS
The temperature coefficients for full-scale calibration, unipolar offset, and bipolar offset specify the maximum change from the initial (25°C) value to the value at T_{min} or T_{max}.

POWER SUPPLY REJECTION
The standard specifications for the AD574A assume use of +5.00 and ±15.00 or ±12.00 volt supplies. The only effect of power supply error on the performance of the device will be a small change in the full scale calibration. This will result in a linear change in all lower order codes. The specifications show the maximum change in calibration from the initial value with the supplies at the various limits.

CODE WIDTH
A fundamental quantity for A/D converter specifications is the code width. This is defined as the range of analog input values for which a given digital output code will occur. The nominal value of a code width is equivalent to 1 least significant bit (LSB) of the full scale range or 2.44mV out of 10 volts for a 12-bit ADC.

* Reprinted with permission from Analog Devices

CIRCUIT OPERATION

The AD574A is a complete 12-bit A/D converter which requires no external components to provide the complete successive-approximation analog-to-digital conversion function. A block diagram of the AD574A is shown in Figure 1. The device consists of two chips, one containing the precision 12-bit DAC with voltage reference, the other containing the comparator, successive-approximation register, clock, output buffers and control circuitry.

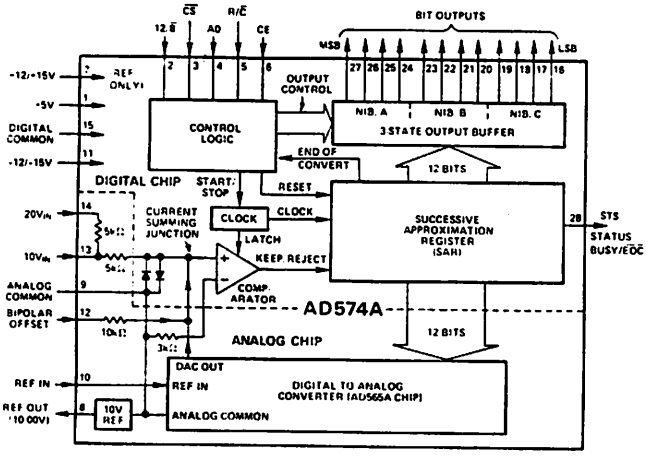


Figure 1. Block Diagram of AD574A 12-Bit A-to-D Converter

When the control section is commanded to initiate a conversion (as described later), it then enables the clock and resets the successive-approximation register (SAR) to all zeros. Once a conversion cycle has begun, it cannot be stopped or re-started and data is not available from the output buffers. The SAR, timed by the clock, will then sequence through the conversion cycle and return an end-of-convert flag to the control section. The control section will then disable the clock, bring the output status flag low, and enable control functions to allow data read functions by external command.

During the conversion cycle, the internal 12-bit current output DAC is sequenced by the SAR from the most-significant-bit (MSB) to least-significant-bit (LSB) to provide an output current which accurately balances the input signal current through the 5kΩ (or 10kΩ) input resistor. The comparator determines whether the addition of each successively-weighted bit current causes the DAC current sum to be greater or less than the input current; if the sum is less, the bit is left on; if more, the bit is turned off. After testing all the bits, the SAR contains a 12-bit binary code which accurately represents the input signal to within ±½LSB.

The temperature-compensated buried Zener reference provides the primary voltage reference to the DAC and guarantees excellent stability with both time and temperature. The reference is trimmed to 10.00 volts ±1%; it can supply up to 1.5mA to an external load in addition to that required to drive the reference input resistor (0.5mA) and bipolar offset resistor (1mA) when the AD574A is powered from ±15V supplies. If the AD574A is used with ±12V supplies, or if external current must be supplied over the full temperature range, an external buffer amplifier is recommended. Any external load on the AD574A reference must remain constant during conversion. The thin film application resistors are trimmed to match the full scale output current of the DAC. There are two 5kΩ input scaling resistors to allow either a 10 volt or 20 volt span. The 10kΩ bipolar offset resistor is grounded for unipolar operation or connected to the 10 volt reference for bipolar operation.

DRIVING THE AD574A ANALOG INPUT

The AD574A is a successive-approximation type analog-to-digital converter. During the conversion cycle, the ADC input current is modulated by the DAC test current at approximately a 500kHz rate. Thus it is important to recognize that the signal source driving the AD574A must be capable of holding a constant output voltage under dynamically-changing load conditions.

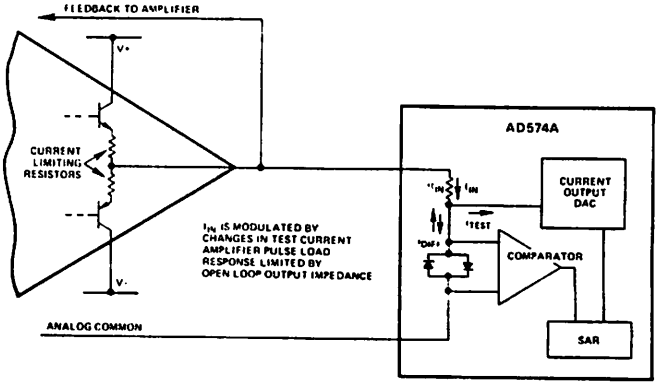


Figure 2. Op Amp – AD574A Interface

The closed loop output impedance of an op amp is equal to the open loop output impedance (usually a few hundred ohms) divided by the loop gain at the frequency of interest. It is often assumed that the loop gain of a follower-connected op amp is sufficiently high to reduce the closed loop output impedance to a negligibly small value, particularly if the signal is low frequency. However, the amplifier driving an AD574A must either have sufficient loop gain at 500kHz to reduce the closed loop output impedance to a low value or have low open loop output impedance.

This can be accomplished either by using a wideband op amp or by placing a discrete-transistor or integrated buffer inside the amplifier's feedback loop.

SUPPLY DECOUPLING AND LAYOUT CONSIDERATIONS

It is critically important that the AD574A power supplies be filtered, well-regulated, and free from high frequency noise. Use of noisy supplies will cause unstable output codes to be generated. Switching power supplies are not recommended for circuits attempting to achieve 12-bit accuracy unless great care is used in filtering any switching spikes present in the output. Remember that a few millivolts of noise represents several counts of error in a 12-bit ADC.

Decoupling capacitors should be used on all power supply pins; the +5V supply decoupling capacitor should be connected directly from pin 1 to pin 15 (digital common) and the $+V_{CC}$ and $-V_{EE}$ pins should be decoupled directly to analog common (pin 9). A suitable decoupling capacitor is a 47µF tantalum type in parallel with a 0.1µF disc ceramic type.

Circuit layout should attempt to locate the AD574A, associated analog input circuitry, and interconnections as far as possible from logic circuitry. For this reason, the use of wire-wrap circuit construction is not recommended. Careful printed-circuit construction is preferred.

*Reprinted with permission from Analog Devices

UNIPOLAR RANGE CONNECTIONS FOR THE AD574A

The AD574A contains all the active components required to perform a complete 12-bit A/D conversion. Thus, for most situations, all that is necessary is connection of the power supplies (+5, −12 +15 and −12/−15 volts), the analog input, and the conversion initiation command, as discussed on the next page. Analog input connections and calibration are easily accomplished; the unipolar operating mode is shown in Figure 4.

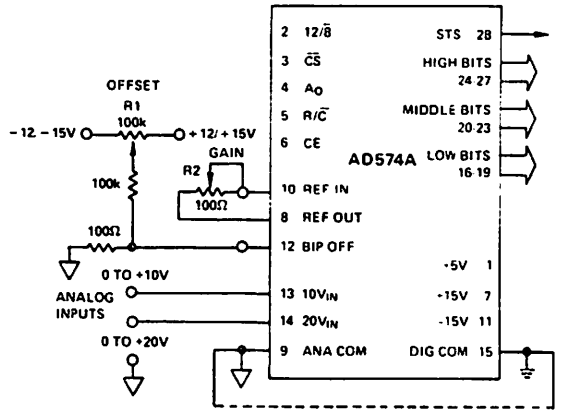


Figure 3. Unipolar Input Connections

All of the thin film application resistors of the AD574A are trimmed for absolute calibration. Therefore, in many applications, no calibration trimming will be required. The absolute accuracy for each grade is given in the specification tables. For example, if no trims are used, the AD574AK guarantees ±2LSB max zero offset error and ±0.25% (10LSB) max full scale error. (Typical full scale error is ±2LSB.) If the offset trim is not required, pin 12 can be connected directly to pin 9; the two resistors and trimmer for pin 12 are then not needed. If the full scale trim is not needed, a 50Ω ±1% metal film resistor should be connected between pin 8 and pin 10.

The analog input is connected between pin 13 and pin 9 for a 0 to +10V input range, between 14 and pin 9 for a 0 to +20V input range. The AD574A easily accommodates an input signal beyond the supplies. For the 10 volt span input, the LSB has a nominal value of 2.44mV, for the 20 volt span, 4.88mV. If a 10.24V range is desired (nominal 2.5mV/bit), the gain trimmer (R2) should be replaced by a 50Ω resistor, and a 200Ω trimmer inserted in series with the analog input to pin 13 (for a full scale range of 20.48V (5mV/bit), use a 500Ω trimmer into pin 14). The gain trim described below is now done with these trimmers. The nominal input impedance into pin 13 is 5kΩ, and 10kΩ into pin 14.

UNIPOLAR CALIBRATION

The AD574A is intended to have a nominal ½LSB offset so that the exact analog input for a given code will be in the middle of that code (halfway between the transitions to the codes above and below it). Thus, when properly calibrated, the first transition (from 0000 0000 0000 to 0000 0000 0001) will occur for an input level of +1/2LSB (1.22mV for 10V range).

If pin 12 is connected to pin 9, the unit typically will behave in this manner, within specifications. If the offset trim (R1) is used, it should be trimmed as above, although a different offset can be set for a particular system requirement. This circuit will give approximately ±15mV of offset trim range.

The full scale trim is done by applying a signal 1 1/2LSB below the nominal full scale (9.9963 for a 10V range). Trim R2 to give the last transition (1111 1111 1110 to 1111 1111 1111).

BIPOLAR OPERATION

The connections for bipolar ranges are shown in Figure 4. Again, as for the unipolar ranges, if the offset and gain specifications are sufficient, one or both of the trimmers shown can be replaced by a 50Ω ±1% fixed resistor. The analog input is applied as for the unipolar ranges. Bipolar calibration is similar to unipolar calibration. First, a signal ½LSB above negative full scale (−4.9988V for the ±5V range) is applied and R1 is trimmed to give the first transition (0000 0000 0000 to 0000 0000 0001). Then a signal 1½LSB below positive full scale (+4.9963V for the ±5V range) is applied and R2 trimmed to give the last transition (1111 1111 1110 to 1111 1111 1111).

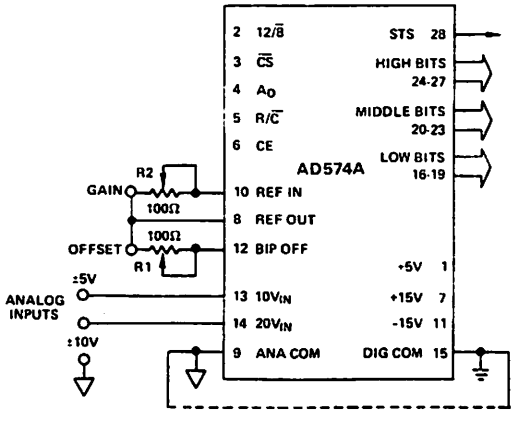


Figure 4. Bipolar Input Connections

GROUNDING CONSIDERATIONS

The analog common at pin 9 is the ground reference point for the internal reference and is thus the "high quality" ground for the AD574A; it should be connected directly to the analog reference point of the system. In order to achieve all of the high accuracy performance available from the AD574A in an environment of high digital noise content, it is required that the analog and digital commons be connected together at the package. In some situations, the digital common at pin 15 can be connected to the most convenient ground reference point; analog power return is preferred.

CONVERSION START/DATA READ CONTROL LOGIC

The AD574A contains on-chip logic to provide conversion initiation and data read operations from signals commonly available in microprocessor systems. Figure 5 shows the internal logic circuitry of the AD574A.

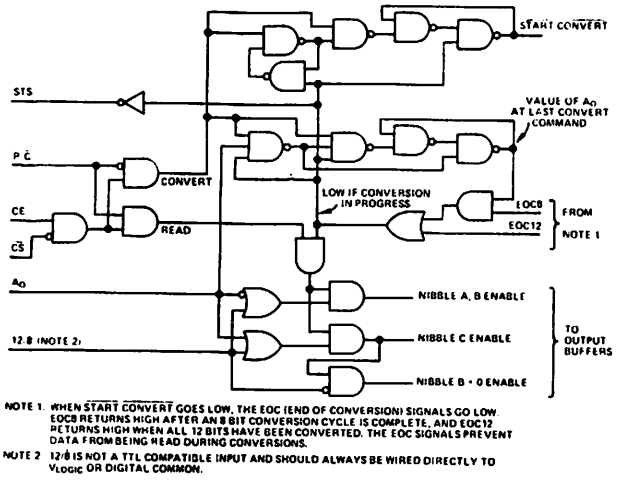


Figure 5. AD574A Control Logic

The control signals CE, $\overline{CS}$, and R/$\overline{C}$ control the operation of the converter. The state of R/$\overline{C}$ when CE and $\overline{CS}$ are both asserted determines whether a data read (R/$\overline{C}$ = 1) or a convert (R/$\overline{C}$ = 0) is in progress. The register control inputs A_0 and 12/$\overline{8}$ control conversion length and data format. The A_0 line is usually tied to the least significant bit of the address bus. If a conversion is started with A_0 low, a full 12-bit conversion cycle is initiated. If A_0 is high during a convert start, a shorter 8-bit conversion cycle results. During data read operations, A_0 determines whether the three-state buffers containing the 8 MSBs of the conversion result (A_0 = 0) or the 4 LSBs (A_0 = 1) are enabled. The 12/$\overline{8}$ pin determines whether the output data is to be organized as two 8-bit words (12/$\overline{8}$ tied to DIGITAL COMMON) or a single 12-bit word (12/$\overline{8}$ tied to VLOGIC). The 12/$\overline{8}$ pin is not TTL-compatible and must be hard-wired to either VLOGIC or DIGITAL COMMON. In the 8-bit mode, the byte addressed when A_0 is high contains the 4 LSBs from the conversion followed by four trailing zeroes. This organization allows the data lines to be overlapped for direct interface to 8-bit buses without the need for external three-state buffers.

It is not recommended that A_0 change state during a data read operation. Asymmetrical enable and disable times of the three-state buffers could cause internal bus contention resulting in potential damage to the AD574A.

An output signal, STS, indicates the status of the converter. STS goes high at the beginning of a conversion and returns low when the conversion cycle is complete.

CE	$\overline{CS}$	R/$\overline{C}$	12/$\overline{8}$	A_0	Operation
0	X	X	X	X	None
X	1	X	X	X	None
1	0	0	X	0	Initiate 12-Bit Conversion
1	0	0	X	1	Initiate 8-Bit Conversion
1	0	1	Pin 1	X	Enable 12-Bit Parallel Output
1	0	1	Pin 15	0	Enable 8 Most Significant Bits
1	0	1	Pin 15	1	Enable 4LSBs + 4 Trailing Zeroes

Table I. AD574A Truth Table

TIMING

The AD574A is easily interfaced to a wide variety of microprocessors and other digital systems. Discussion of the timing requirements of the AD574A control signals will provide the system designer with useful insight into the operation of the device.

Figure 6 shows a complete timing diagram for the AD574A convert start operation. R/$\overline{C}$ should be low before both CE and $\overline{CS}$ are asserted; if R/$\overline{C}$ is high, a read operation will momentarily occur, possibly resulting in system bus contention. Either CE or $\overline{CS}$ may be used to initiate a conversion. As shown in Figure 6,

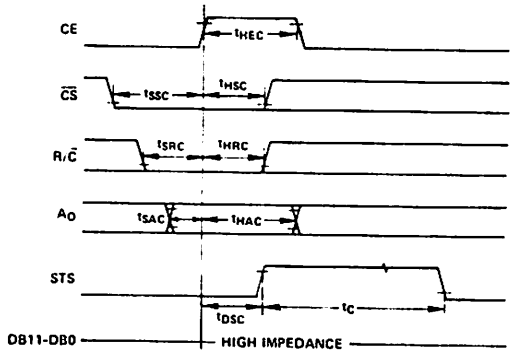


Figure 6. Convert Start Timing

CE is used. If $\overline{CS}$ is used to trigger conversion or if the specified set-up times are not met, appropriately longer pulses are necessary (to provide at least 200ns when R/$\overline{C}$, CE, and $\overline{CS}$ are all valid). Note that CE includes one less propagation delay than $\overline{CS}$ and is therefore the faster input.

Once a conversion is started and the STS line goes high, convert start commands will be ignored until the conversion cycle is complete. The output data buffers cannot be enabled during conversion.

CONVERT START TIMING – FULL CONTROL MODE

Symbol	Parameter	Min	Typ	Max	Units
t_{DSC}	STS Delay from CE			300	ns
t_{HEC}	CE Pulse Width	300			ns
t_{SSC}	$\overline{CS}$ to CE Setup	300			ns
t_{HSC}	$\overline{CS}$ Low During CE High	200			ns
t_{SRC}	R/$\overline{C}$ to CE Setup	250			ns
t_{HRC}	R/$\overline{C}$ Low During CE High	200			ns
t_{SAC}	A_0 to CE Setup	0			ns
t_{HAC}	A_0 Valid During CE High	300			ns
t_C	Conversion Time				
	8-Bit Cycle	10		24	µs
	12-Bit Cycle	15		35	µs

Figure 7 shows the timing for data read operations. The AD574A differs from the original AD574 design in that the three-state output buffers feature faster access time and shorter data latency

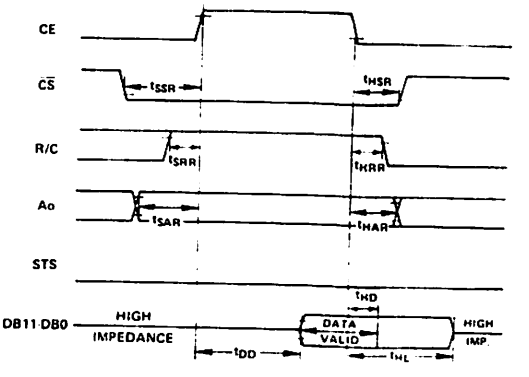


Figure 7. Read Cycle Timing

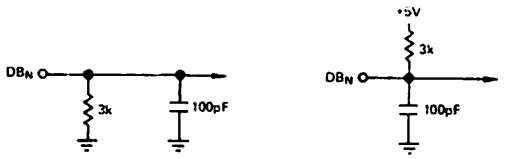
times. This speed improvement simplifies the interface to faster microprocessors. During data read operations, access time is measured from the point where CE and R/$\overline{C}$ both are high (assuming $\overline{CS}$ is already low). If $\overline{CS}$ is used to enable the device, access time is extended by 100ns.

In the 8-bit bus interface mode (12/$\overline{8}$ input wired to DIGITAL COMMON), the address bit, A_0, must be stable at least 150ns prior to $\overline{CE}$ going high and must remain stable during the entire read cycle. If A_0 is allowed to change, damage to the AD574A output buffers may result.

READ TIMING – FULL CONTROL MODE

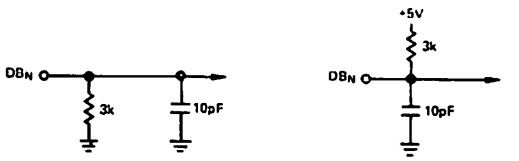
Symbol	Parameter	Min	Typ	Max	Units
t_{DD}[1]	Access Time (from CE)		210	250	ns
t_{HD}	Data Valid after CE Low	25			ns
t_{HL}[2]	Output Float Delay		110	150	ns
t_{SSR}	$\overline{CS}$ to CE Setup	150			ns
t_{SRR}	R/$\overline{C}$ to CE Setup	0			ns
t_{SAR}	A_0 to CE Setup	150			ns
t_{HSR}	$\overline{CS}$ Valid After CE Low	50			ns
t_{HRR}	R/$\overline{C}$ High After CE Low	0			ns
t_{HAR}	A_0 Valid After CE low	50			ns

[1] t_{DD} is measured with the load circuit of Figure 8 and defined as the time required for an output to cross 0.4V or 2.4V.
[2] t_{HL} is defined as the time required for the data lines to change 0.5V when loaded with the circuit of Figure 9.



a. High-Z to Logic 1 *b. High-Z to Logic 0*

Figure 8. Load Circuit for Access Time Test



a. Logic 1 to High-Z *b. Logic 0 to High-Z*

Figure 9. Load Circuit for Output Float Delay Test

"STAND-ALONE" OPERATION

The AD574A can be used in a "stand-alone" mode, which is useful in systems with dedicated input ports available and thus not requiring full bus interface capability.

In this mode, CE and 12/$\overline{8}$ are wired high, $\overline{CS}$ and A_0 are wired low, and conversion is controlled by R/$\overline{C}$. The three-state buffers are enabled when R/$\overline{C}$ is high and a conversion starts when R/$\overline{C}$ goes low. This gives rise to two possible control signals–a high pulse or a low pulse. Operation with a low pulse is shown in Figure 10. In this case, the outputs are forced into the high-impedance state in response to the falling edge of R/$\overline{C}$ and return to valid logic levels after the conversion cycle is completed. The STS line goes high 500ns after R/$\overline{C}$ goes low and returns low 300ns after data is valid.

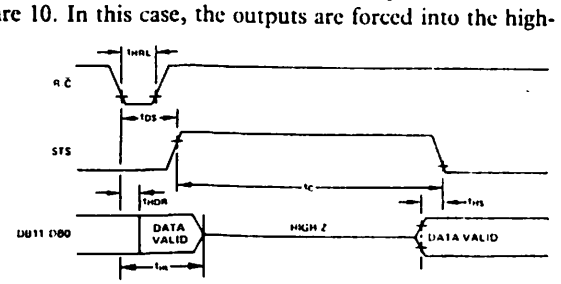


Figure 10. Low Pulse for R/$\overline{C}$ – Outputs Enabled After Conversion

If conversion is initiated by a high pulse as shown in Figure 11, the data lines are enabled during the time when R/$\overline{C}$ is high. The falling edge of R/$\overline{C}$ starts the next conversion and the data lines return to three-state (and remain three-state) until the next high pulse of R/$\overline{C}$.

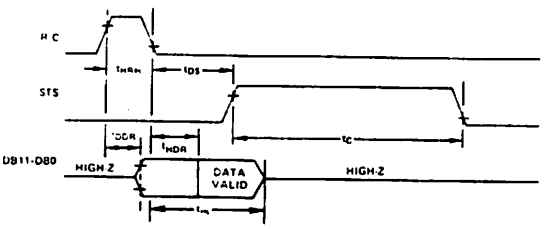


Figure 11. Low Pulse for R/$\overline{C}$ – Outputs Enabled While R/$\overline{C}$ High, Otherwise High-Z

STAND-ALONE MODE TIMING

Symbol	Parameter	Min	Typ	Max	Units
t_{HRL}	Low R/$\overline{C}$ Pulse Width	350			ns
t_{DS}	STS Delay from R/$\overline{C}$			500	ns
t_{HDR}	Data Valid After R/$\overline{C}$ Low	25			ns
t_{HL}	Output Float Delay		110	150	ns
t_{HS}	STS Delay After Data Valid	300		1000	ns
t_{HRH}	High R/$\overline{C}$ Pulse Width	250			ns
t_{DDR}	Data Access Time			250	ns

INTERFACING THE AD574A TO MICROPROCESSORS

The control logic of the AD574A makes direct connection to most microprocessor system buses possible. While it is impossible to describe the details of the interface connections for every microprocessor type, several representative examples will be described here.

GENERAL A/D CONVERTER INTERFACE CONSIDERATIONS

Analog-to-digital converters, like any I/O device, may be interfaced to microprocessors by several methods. These methods include (but are not limited to) direct memory access, isolated or accumulator I/O, and memory-mapped I/O. Direct memory access (DMA) is the fastest, since conversions occur automatically and data updates into memory are transparent to the processor. DMA logic is very processor-dependent and makes use of dedicated specialized hardware.

Memory-mapped and accumulator I/O are more often used and somewhat easier to understand. Memory-mapped I/O assigns the I/O device to one or more locations in the memory space of the microprocessor. This technique has the advantage that the full range of memory reference instructions may be used to operate on the data. The potential disadvantages include limiting the memory space available for program and data memory, somewhat more complex address decoding and more difficult isolation of device select pulses for system debugging. Many processors offer only memory-mapped I/O.

Accumulator I/O uses a set of control signals which are distinct and different from the memory control signals. These control signals, combined with the address bus, serve to define a totally

* Reprinted by permission from Analog Devices

separate I/O address space. This architecture is simpler from a hardware standpoint, since address decoding requirements are less severe and distinct I/O read and write pulses are more easily located for system debugging purposes. However, processors using accumulator I/O generally can only send data to an output device from the accumulator. This can make the software more cumbersome, since processor-controlled transfers of I/O device data to a memory location cannot be accomplished in a single instruction.

A typical A/D converter interface routine involves several operations. First, a write to the ADC address initiates a conversion. The processor must then wait for the conversion cycle to complete, since most integrated circuit ADCs take longer than one instruction cycle to complete a conversion. Valid data can, of course, only be read after the conversion is complete. The AD574A provides an output signal (STS) which indicates when a conversion is in progress. This signal can be polled by the processor by reading it through an external three-state buffer (or other input port). The STS signal can also be used to generate an interrupt upon completion of conversion, if the system timing requirements are critical (bear in mind that the maximum conversion time of the AD574A is only 35 microseconds) and the processor has other tasks to perform during the ADC conversion cycle. Another possible time-out method is to assume that the ADC will take 35 microseconds to convert, and insert a sufficient number of "do-nothing" instructions to ensure that 35 microseconds of processor time is consumed.

Once it is established that the converter is done with its cycle, the data can be read. In the case of an ADC of 8-bit resolution (or less), a single data read operation is sufficient. In the case of converters with more data bits than are available on the bus, a choice of data formats is required, and multiple read operations are needed. The AD574A includes internal logic to permit direct interface to 8-bit or 16-bit data buses, selected by connection of the 12/$\overline{8}$ input. In 16-bit bus applications (12/$\overline{8}$ high) the data lines (DB11 through DB0) may be connected to either the 12 most significant or 12 least significant bits of the data bus. The remaining four bits should be masked in software. The interface to an 8-bit data bus (12/$\overline{8}$ low) is done in a left-justified format. The even address (A0 low) contains the 8MSBs (DB11 through DB4). The odd address (A0 high) contains the 4LSBs (DB3 through DB0) in the upper half of the byte, followed by four trailing zeroes, thus eliminating bit masking instructions.

	D7							D0
XXX0 (EVEN ADDR):	DB11 (MSB)	DB10	DB9	DB8	DB7	DB6	DB5	DB4
XXX1 (ODD ADDR):	DB3	DB2	DB1	DB0 (LSB)	0	0	0	0

Figure 12. AD574A Data Format for 8-Bit Bus

It is not possible to rearrange the AD574A data lines for right-justified 8-bit bus interface.

The AD574A three-state buffers feature access times and data latency times comparable to presently-available memory devices. Therefore, the AD574A can interface directly to many processor buses without the need for wait states or external data buffers.

SPECIFIC PROCESSOR INTERFACE EXAMPLES

6800/6502-Type Systems

The control signals and bus architecture of the 6800 series and 6502 series microprocessors are very similar. In each, the state of the R/$\overline{W}$ signal at the rising edge of the ø2 (or equivalent) clock establishes whether a memory read or write is in progress. The memory address being exercised is signaled by decoding the address bits to (usually) an active low signal.

This control structure is directly compatible with the AD574A. The R/$\overline{W}$ line can be used for R/$\overline{C}$, the active-low decoded base address (the AD574A occupies two memory locations) is applied to $\overline{CS}$, and ø2 is used for CE. The least-significant address line ties to the AD574A A0 input.

In this interface, the processor can write to one address (A0 low) to start a full 12-bit conversion or another address (A0 high) to start a short 8-bit conversion. The contents of the data bus are meaningless during these writes. After sufficient time has passed for the conversion to complete, the processor can read the data in the two memory locations occupied by the AD574A. The even location (A0 low) contains the eight MSBs and the odd location contains the four LSBs and four trailing zeroes.

The AD574A may be used directly with 6800 series processors running at clock speeds up to 1.5MHz.

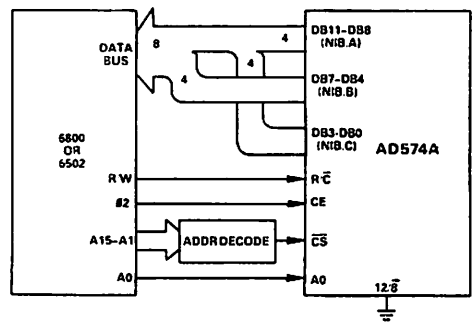


Figure 13. AD574A-6800/6502 Interface Connections

8085A Interface

The 8085A microprocessor uses a multiplexed address/data bus. At the beginning of a machine cycle, this bus contains the low byte of the address being exercised. The ALE output signal is available to strobe a latch to hold the low address byte. For the rest of the machine cycle, this bus carries data to or from the CPU.

The 8085A can use either accumulator I/O or memory-mapping for I/O devices. The system $\overline{RD}$ and $\overline{WR}$ are gated with IO/$\overline{M}$ to provide distinct I/O read and write signals and memory read and write signals. The control signals required for the AD574A are easily derived from the 8085A control bus. $\overline{CS}$ is taken from an address decoder on the high-order address bits. R/$\overline{C}$ can be taken from $\overline{WR}$ (either I/O write or memory write), A0 is tied to the LSB of the address bus, and CE is taken from the output of a NAND gate driven from $\overline{RD}$ and $\overline{WR}$. All bus access and float delay requirements are met for direct bus interface for 8085A clock rates up to 3MHz.

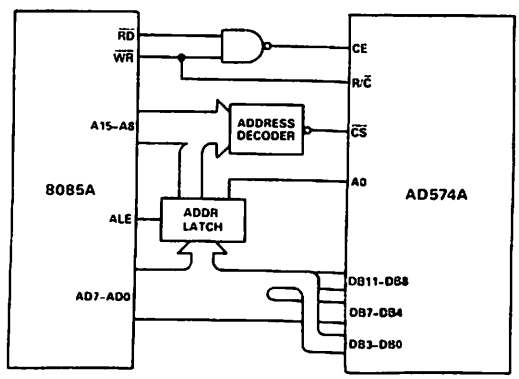


Figure 14. AD574A-8085A Direct Bus Interface

In 8085A systems running at high clock frequencies some external circuitry is required. First, the AD574A delay from CE going low to the data lines going into three-state will cause a bus conflict when the 8085A sends out the low byte of the next instruction address. This conflict will occur if the AD574A data outputs are tied directly to the 8085A bus. In systems where bus transceivers (e.g., 74LS245, 8286, etc.) are used to separate the address and data lines, the conflict is eliminated. The transceivers are disabled at the end of the read cycle and thus isolate the AD574A from the 8085A bus. Since most systems incorporate such buffers, this does not add to system complexity.

A second consideration when interfacing to higher speed 8085A systems is the width of the convert start pulse. The $\overline{WR}$ pulse from a 5MHz 8085A is only guaranteed to be 230 nanoseconds wide and is thus not long enough to initiate a conversion. There are two solutions to this problem. One possibility is to use a dual D-type flip-flop connected as shown in Figure 15 to insert a single wait state in read and write operations directed towards the AD574A. Another solution is to substitute the earlier-occurring S1 and S0 outputs from 8085A for $\overline{RD}$ and $\overline{WR}$ in the circuit of Figure 14 to generate the required control signals. It is important that bus transceivers be employed if S1 and S0 are used for control signals since these signals remain active longer than $\overline{RD}$ and $\overline{WR}$, enabling the AD574A output buffers in read operations for too long, causing potential bus conflicts.

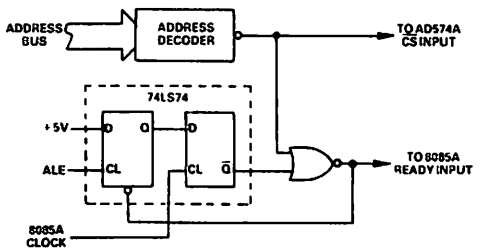


Figure 15. Wait State Generator for 5MHz 8085A Interface

Z-80 System Interface

The Z-80 series of 8-bit microprocessors, like the 8085A, offers both memory-mapped and accumulator I/O capability. While the 8085A only includes two instructions for accumulator I/O (IN and OUT), the Z-80 I/O instruction set is considerably more extensive.

The control signals available on the Z-80 include $\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$, and $\overline{WR}$. The $\overline{RD}$ and $\overline{WR}$ signals indicate direction of data flow while $\overline{MREQ}$ and $\overline{IORQ}$ determine whether the read or write cycle in progress is a memory or I/O cycle. During I/O reads and writes, only 8 address lines are active (as in the 8085A). An interesting feature of the Z-80 is that I/O read and write cycles are automatically extended by one clock cycle (one wait state is inserted) and are thus slower. The Z-80 control signal connections to the AD574A are identical to the 8085A connections.

The AD574A can be interfaced to Z-80 series processors with clock speeds up to 2.5MHz in the memory address space using the $\overline{MWR}$ and $\overline{MRD}$ signals. At higher clock rates (4 and 6MHz), the memory write pulse is not wide enough to properly start a conversion. The extra wait state added during I/O write operations will extend this pulse to a suitable width at clock rates up to 6MHz so that accumulator I/O is possible.

INTERFACING THE AD574A TO THE APPLE II COMPUTER

The AD574A can be used to provide a low-cost precision analog input port for the Apple II microcomputer without the need for additional power supplies or extensive digital interface logic. The AD574A can be mounted on a hobby card designed to plug into an Apple II I/O slot.

Hardware

All required supply voltages and control signals are available on the Apple's peripheral connectors. Each connector contains, on pin 41, a $\overline{DEVICE\ SELECT}$ output which is active when the address bus holds a hexadecimal address between C0n0 and C0nF, where n is equal to the slot number plus 8. This signal can be connected to pin 3 ($\overline{CS}$) of the AD574A. The Φ0 clock on pin 40 of the peripheral connector can be used for the AD574A CE input (pin 6). The AD574A R/$\overline{C}$ input (pin 5) can be driven directly by the R/$\overline{W}$ output available on peripheral connector pin 18. Pin 2 of the peripheral connector, A0, connects directly to the AD574A pin 4. The connections between the peripheral connector and the AD574A are shown in Figure 16.

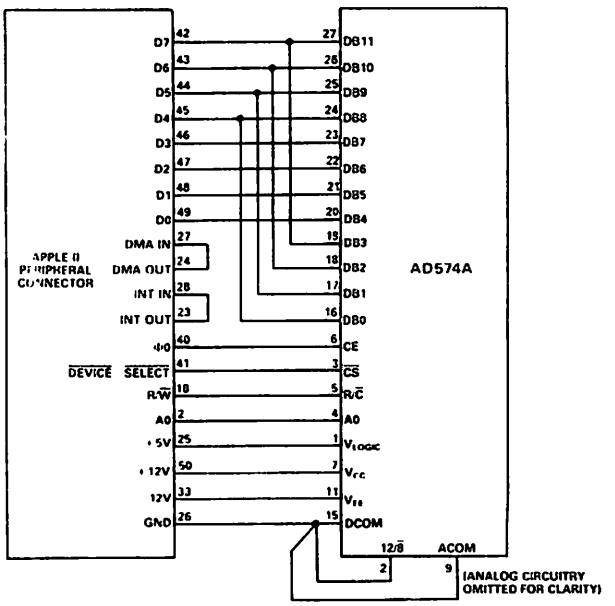


Figure 16. AD574A Connections to Apple II Peripheral Connector

The Apple II represents a relatively hostile electrical environment to the AD574A. The high frequency clocks radiate a large amount of noise which can be inadvertently coupled into analog signal lines. Furthermore, the switching power supply in the Apple is noisy, and this noise will often pollute the analog signals. It is possible, however, by judicious bypassing, decoupling, and ground management, to achieve a data acquisition system with only occasional flicker. A suggested grounding and decoupling scheme is shown in Figure 17.

It is recommended that any signal preamplification used in such a system be physically located outside the Apple cabinet. A full-scale signal range is less susceptible to electromagnetically coupled interference than a smaller signal range would be. Thus, the preferred method is to deliver a buffered, high-level signal to the AD574A through a shielded cable. The ±5V or ±10V

* Reprinted by permission from Analog Devices

range is suggested. Full-scale and offset trims, if desired, are performed as shown on page 7.

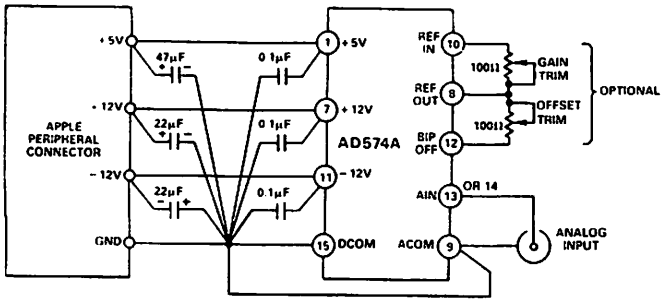


Figure 17. Recommended Grounding Procedure

Software

In this discussion, the AD574A is assumed to be located in I/O slot 2 of the Apple II and be the only device in that slot. The AD574A thus occupies the sixteen locations from $C0A0 through $C0AF, even though only two locations are actually required.

It is possible to operate the AD574A from either machine language or a high-level language. In machine language, the converter is started by writing data to either $C0A0 or $C0A1, using a STA instruction. Writing to $C0A0 will start a full 12-bit conversion cycle; writing to $C0A1 starts an 8-bit cycle. Accumulator contents are unimportant during convert start operations. It is then necessary to wait for the AD574A to finish converting before attempting to read the data. This can be accomplished by loading the accumulator with the value 02 and calling the WAIT subroutine located at $FCA8 in the Apple Monitor.

When data is read, it can be read only 8 bits at a time, as explained on page 10. The sample subroutine below starting at location $4000 performs the control for the AD574A and returns the result in RAM locations $0300 and $0301.

```
4000  A9 02      LDA  #$02
4002  8D A0 C0   STA  $C0A0
4005  20 A8 FC   JSR  $FCA8
4008  AD A0 C0   LDA  $C0A0
400B  8D 00 03   STA  $0300
400E  AD A1 C0   LDA  $C0A1
4012  8D 01 03   STA  $0301
4015  60         RTS
```

Figure 18. Assembly-Language AD574A Control Subroutine

Programs written in Applesoft Basic can also operate the AD574A. Conversion is started by POKEing into location 49312 decimal for a 12-bit conversion (or location 49313 for an 8-bit conversion). Basic executes slowly enough that no delay routines are needed. The output of the AD574A is read by PEEKing into those locations. In order to compute the actual analog voltage, it is necessary to establish the proper weighting for the two bytes read.

The Basic subroutine shown in Figure 19 will accomplish this arithmetic. This routine assumes a ±5V analog signal range and returns the value of actual analog signal voltage in the variable V.

```
100  POKE (49312)
110  A = PEEK (49312): B = PEEK (49313): C = 256
120  A = (A + B/C)/C
130  V = A*10 - 5
140  RETURN
```

Figure 19. Applesoft II Basic Subroutine for AD574A Control

2.3.2 Computer Specific Device Select Board and the DRV11–J

Functional description

The job of the CSDS board (Computer Specific Device Select Board) is to generate the signals necessary to control the functions of the single–tendon I/O boards from the control signals produced by the twelve 16–bit parallel ports of the computer system – provided, in our present case, by Digital Equipment Corporation DRV11–J boards.

Related figures and tables

If you have questions regarding the pinouts of the integrated circuits used in this board, refer to Figures 2.37 and 2.38.

If you aren't working at the University of Massachusetts, you may want to know how we arranged the layout of the board. If so, flip to Figure 2.33.

The wiring diagrams in Figure 2.34 are needed for deciphering the wiring patterns on the wire-wrapped CSDS board.

Circuit description

The CSDS board takes five control signals from each of the system's twelve DRV11–J ports and then translates each set of signals into five new signals that control a corresponding single–tendon I/O board's functions.

The CSDS board performs all its translations in hardwired circuits to achieve high operating speed. Higher speeds would be possible if the DRV11–J made another software–driven control line available.

The STIO boards each carry a 74LS138 3 to 8 decoder. This chip is responsible for controlling the operation of each of the STIO board's functions, and is unable to activate more than one function at a time. This is one way that the CSDS board prevents conflicts on the data I/O bus. A data bus conflict occurs when two or more ICs write out on the same data bus lines simultaneously. A conflict will almost certainly damage the conflicting ICs. The 74LS138 prevents conflicts from occurring between chips residing the on the same STIO board, but a more complicated task is avoiding conflicts between a DRV11–J and its

corresponding STIO board. How this is accomplished is discussed in the explanation that accompanies the CSDS board timing diagrams below. The Analog Devices chips aren't cheap, so if you decide to develop a CSDS board to handle I/O control for another computer system, pay special care to this issue!

A timing diagram explanation

In order to follow the presentation in this subsubsection, study the schematic for this board (Figure 2.28) as well as the timing diagrams for the execution of the various I/O functions of the single–tendon I/O boards (Figures 2.29, 2.30, 2.31 and 2.32). Also, it would be helpful to you to glance at the connector pinouts in Figures 2.35 and 2.36. Lastly, it will be necessary for you to refer to DEC's DRV11–J simplified schematic. This simplified schematic may be found in DEC's DRV11–J manual which is a required companion to this technical report.

Now, let's start analyzing the timing diagrams.

When a large "X" inside a boxed area is present in a signal's timing illustration, it indicates that the state of the signal is undetermined and irrelevant during that period of time.

If you look at the schematic you'll see an OR gate — labeled "C" — whose output drives the GATE input of the 74LS138. This OR gate in turn has two AND gates feeding it. These logic gates check for the presence of sets of conditions under which the 74LS138 should be active. There are four sets of conditions.

The first set of conditions relates to clearing a counter. These conditions are: 1) the CLK input to the latches is high — indicating that the DRV11–J is writing, and 2) the three to eight decoder input signals A and B are both high — which is required for selecting "CS3 NOT". A counter is cleared when "CS3 NOT" is enabled. In order for the CLK input to go high, the DRV11–J DIR bit must be set to one (write) and a write operation must be in progress (causing the DRV11–J RPLY signal to go low). When the CLK signal goes high, the values on data lines D12 and D13 are latched through to Q12 and Q13. Q12 and Q13 then drive signals A and B of the three to eight decoder. If, as mentioned above, the value written has highs in both bits twelve and thirteen, then the output of the OR gate labeled "A" in the schematic will go high. Since the DRV11–J RPLY pulse is long enough, the CLK signal will still be high when OR gate A goes high. Therefore, all the conditions needed to activate OR gate C will have been met. The GATE goes high and the counter is cleared.

The second set of conditions relates to writing to a D/A converter. These conditions are identical to those described for clearing a counter. The only difference is in the value that is written by the DRV11–J. In this case, bit twelve is a one and bit thirteen is a zero, and

bits zero through eleven describe the value to be written into the DAC. These new values for bits twelve and thirteen select "CS1 NOT" which enables the data inputs on the DAC. Once the DAC inputs are active, you may write values out to the DAC as often as you like and the DAC will track the input values.

The third set of conditions relates to reading the position recorded by one of the sixteen–bit counters. These conditions are: 1) three to eight decoder input A is high and input B is low, and DRV11–J DIR is low (read). The exact sequence of events that most quickly lead to fulfilling these conditions are as follows: set the DIR bit to one (write), write out a value with bits twelve and thirteen set as mentioned above, set the DIR bit to zero (read), and then read counter values.

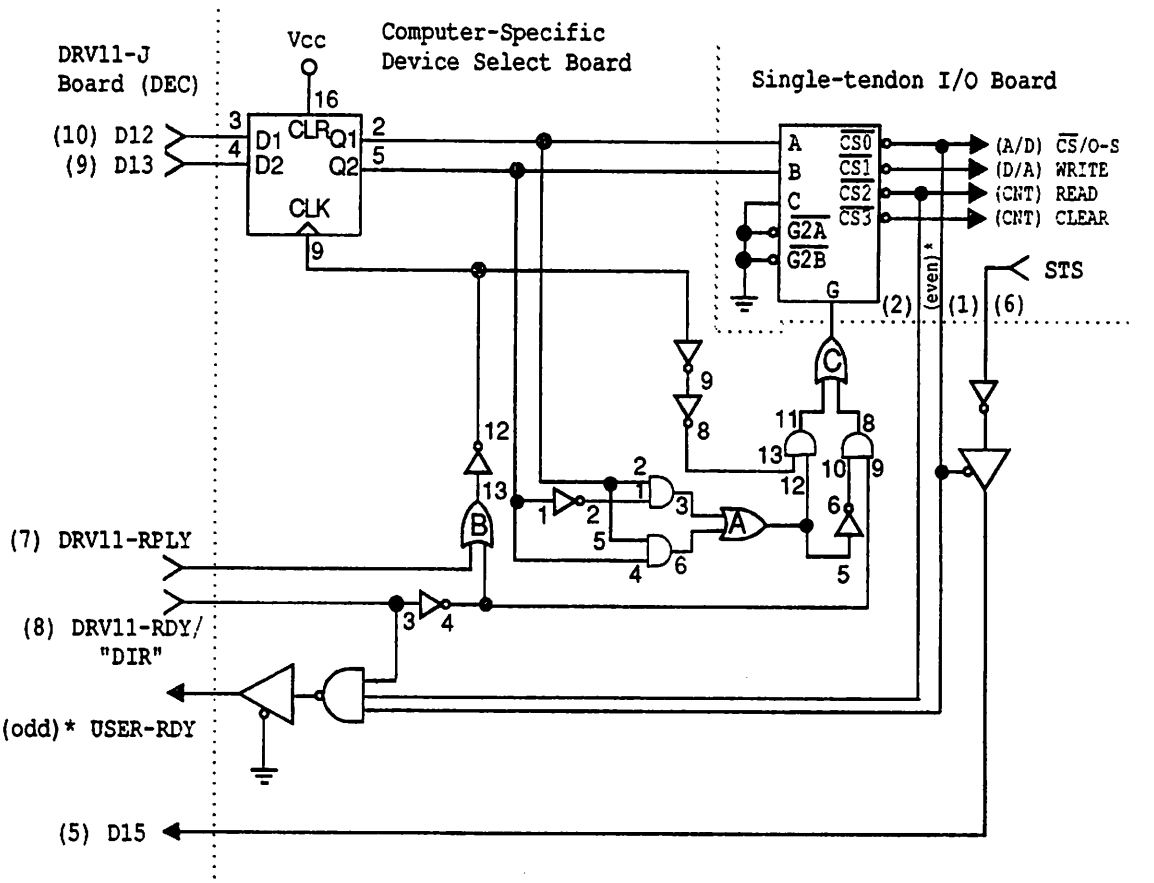
Note: for all the sets of conditions described above, once the device selection signals A and B are latched in, consecutive reads or writes are encouraged since they require less time.

The fourth and last set of conditions relates to reading from an A/D converter. Reading one of the A/Ds is only slightly more complicated than reading a counter. Since the A/Ds don't start a conversion until "CS0 NOT" is enabled, the conversion process may hypothetically be incomplete when a read operation is performed. In order to be sure that the conversion is complete, the STS signal must be checked. STS is brought out to data line 15 through a tristate buffer and inverter. So, if bit 15 is a one, the value read is valid. Consecutive DRV11-J read operations will not yield new values in this case. In order to start a new conversion, DIR must be toggled, and then a new value may be read. A little time may be saved by only writing device select A and B initially when reading an A/D consecutively. However, the time gained is negligible.

CSDSB – Board layout explanation

An explanation of the following two diagrams (Figures 2.33 and 2.34) is in order because these diagrams are rather difficult to interpret. The wire–wrap patterns make good use of the ICs and board space available. Remember that the layout diagram pertains to the board as seen from the back of the board – wire–wrap pin side. And also keep in mind that pin 1 is located in the lower right–hand corner of each wire-wrap socket.

Since there is one '174 – D–type flip–flop package – per I/O channel, the wiring of each



Note: Parentheses indicate pin numbers on control connectors 1, 2 & 3. An asterix indicates the signal passes thru the "extra control connector".

Figure 2.28: Computer Specific Device Select Board - Schematic

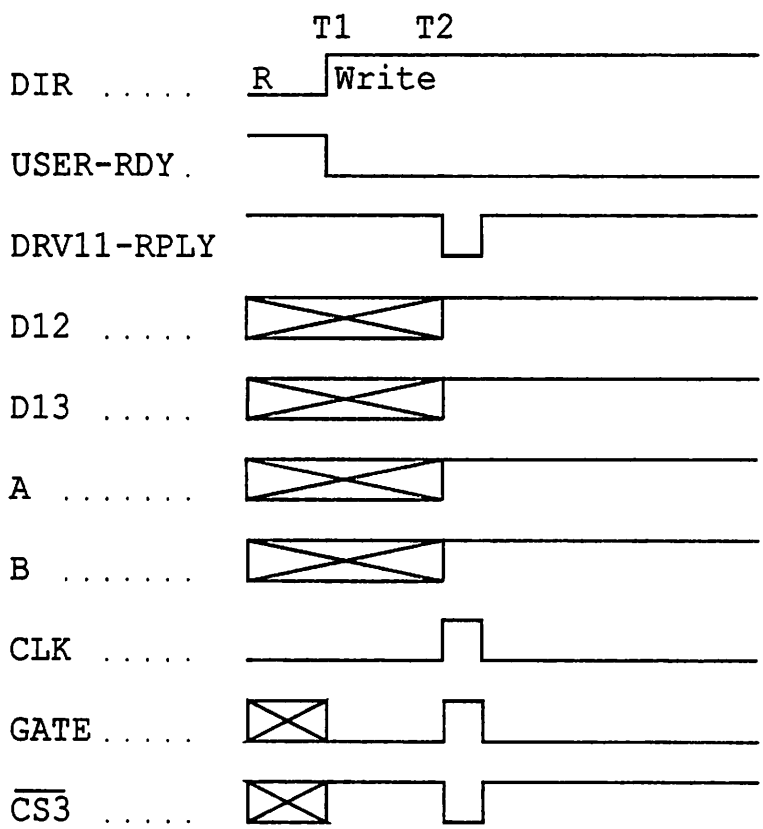


Figure 2.29: Timing diagrams – Clearing a counter

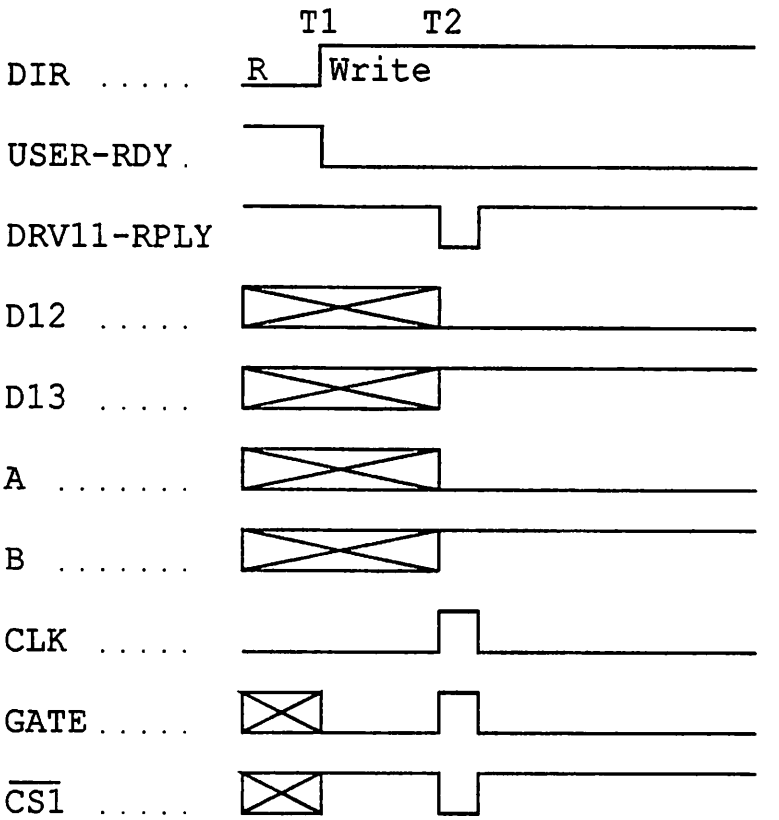
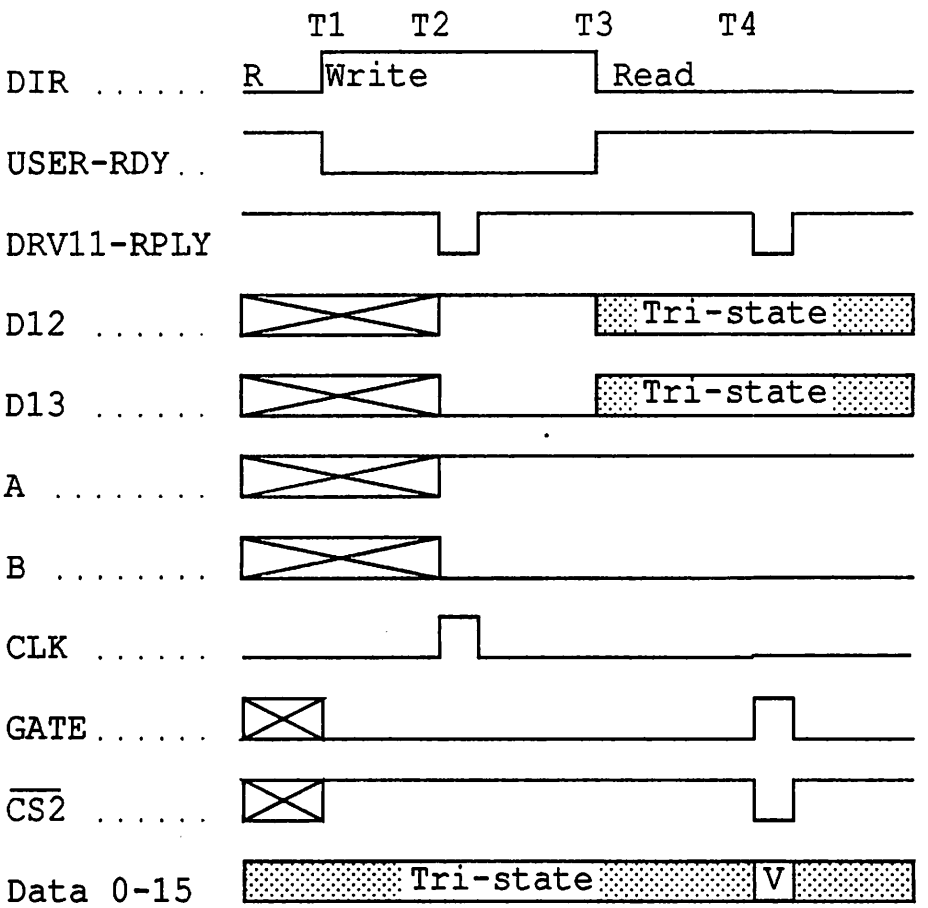


Figure 2.30: Timing diagrams – Writing to a D/A



Note: The "V" on Data 0-15 indicates valid data.

Figure 2.31: Timing diagrams – Reading a counter

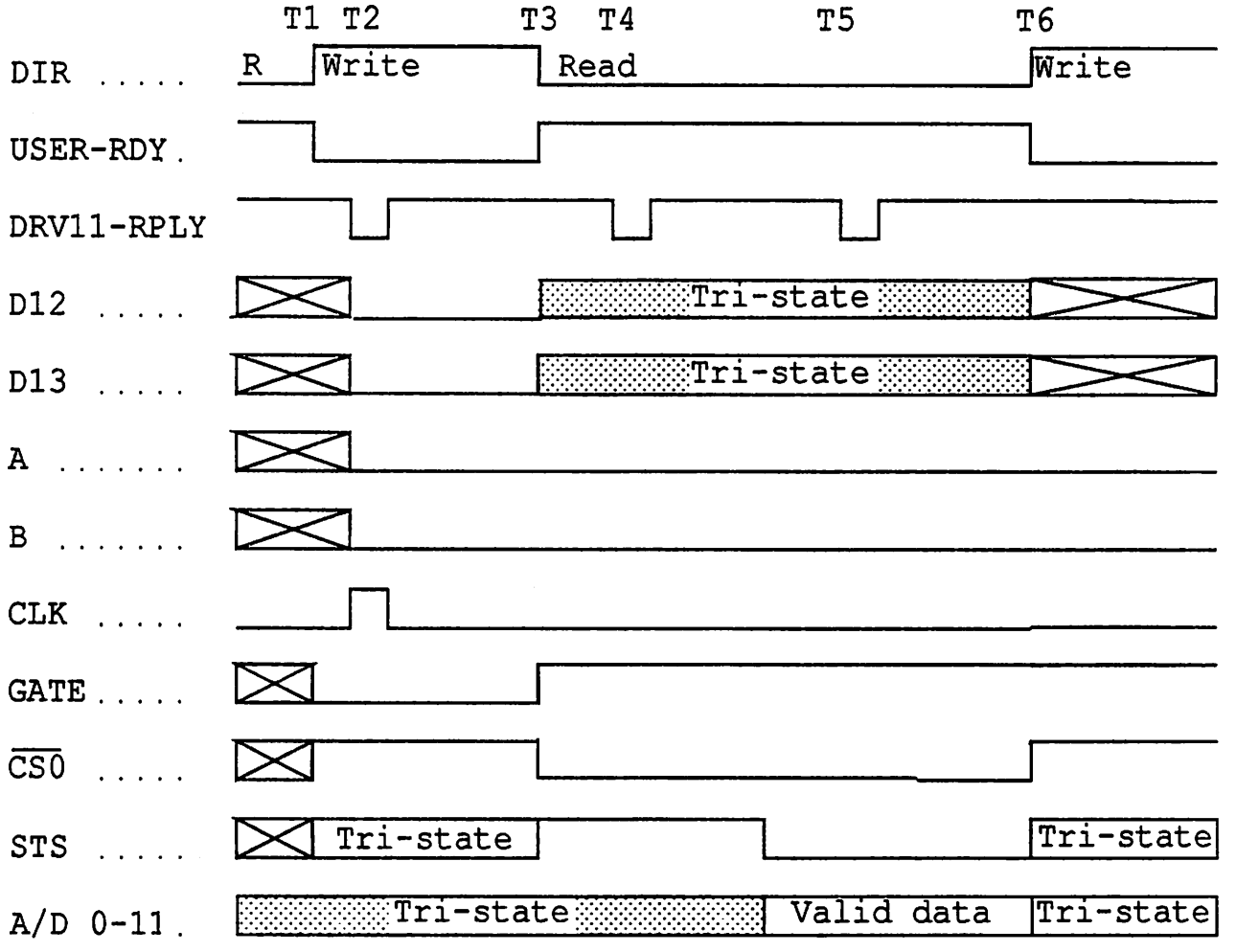


Figure 2.32: Timing diagrams – Reading an A/D

of them is identical. Likewise, there is one main '04 – inverter package – per I/O channel and they are wired identically. Lastly, the same is true of the twelve '08 ICs – AND gates. In other words, each I/O channel uses one package of latched flip–flops, one package of main inverters and one package of AND gates (in addition to other ICs which have more complex and difficult–to–analyze wiring patterns). Note: Each of the ICs in this group are marked with the numbers 1 through 12 in the "Wire–wrap pattern" diagram. The number corresponds to the STIO board served by that IC. To determine the wiring of one of these three sets of IC packages, look at the CSDS board schematic. The pinouts are shown there.

The second bunch of ICs to notice are the OR–gates. As it turns out, each I/O channel requires three OR–gates. Since each OR–gate package has four gates, the most effective way to standardize the wiring is to have one IC hold all the gates that perform a given function for one finger – four I/O channels. In our case we have functions A, B, and C. For each set of finger I/O channels there are three ICs. In the "Wire–wrap pattern" these are labeled 1A, 1B, 1C, 2A, 2B, 2C, 3A, 3B and 3C. The numeric component of the label indicates the associated finger. To ease building and troubleshooting the CSDS board, gates 1, 2, 3 and 4 are dedicated to their respective four channels on each finger. Refer to the "Wire-wrap pattern" diagram to determine the numbering of the gates. For example, let's say that the gates of IC #1A perform function A for the channels of finger one. More specifically, gate 1 (pins 13, 12 and 11) performs function A for channel one of finger one. Then, gate 1 of IC #2A must perform function A for channel one of finger two. NOTE: Channel one of finger two is more accurately referred to as channel five. Lastly, gate 1 of IC #3A must perform function A for channel one of finger three – i.e. channel nine.

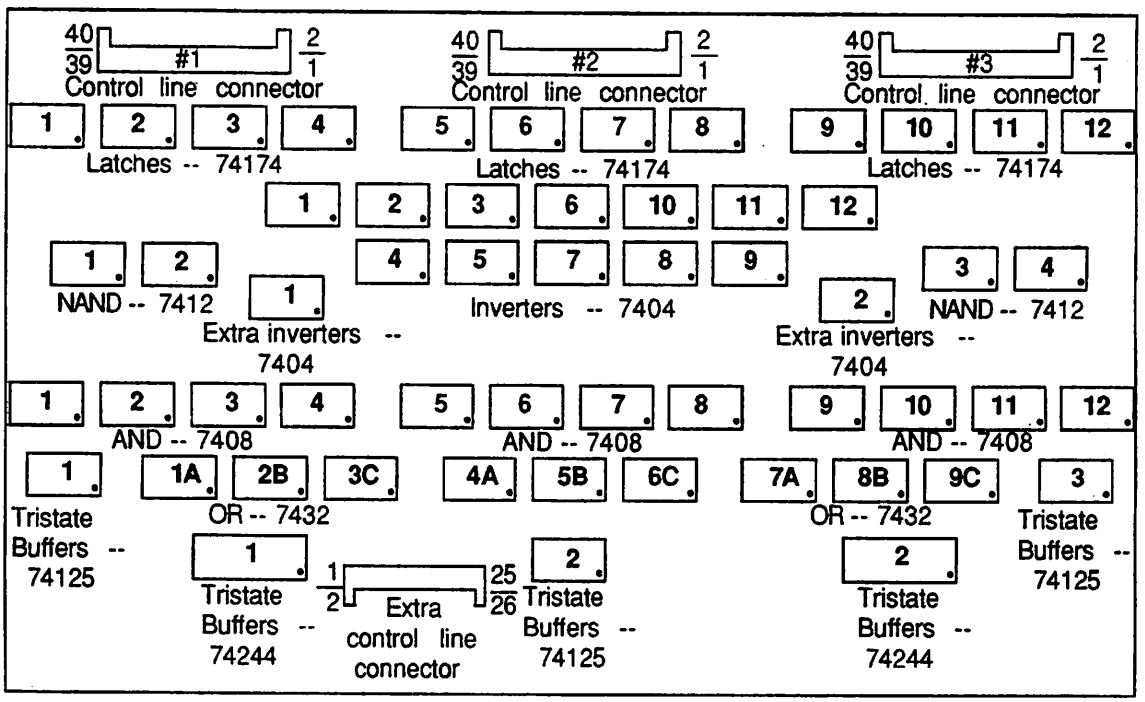


Figure 2.33: CSDSB – Board layout

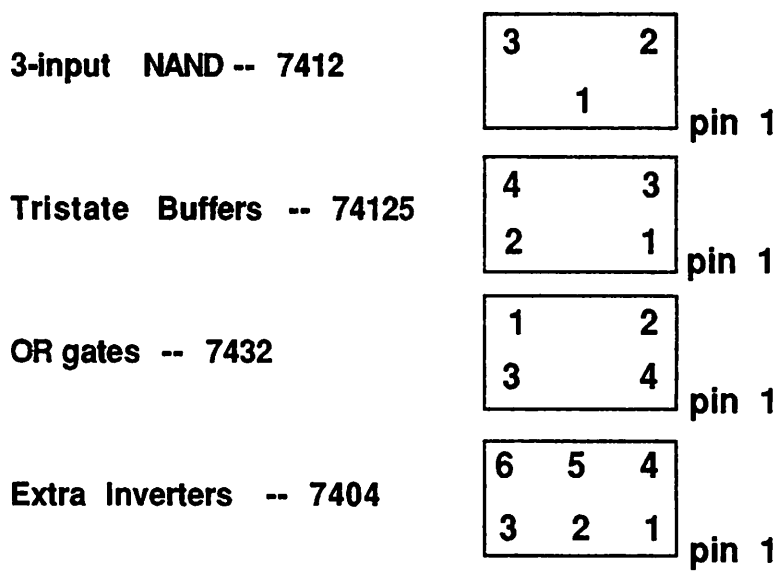


Figure 2.34: Wire-wrap patterns as seen from the wire-wrap pin side

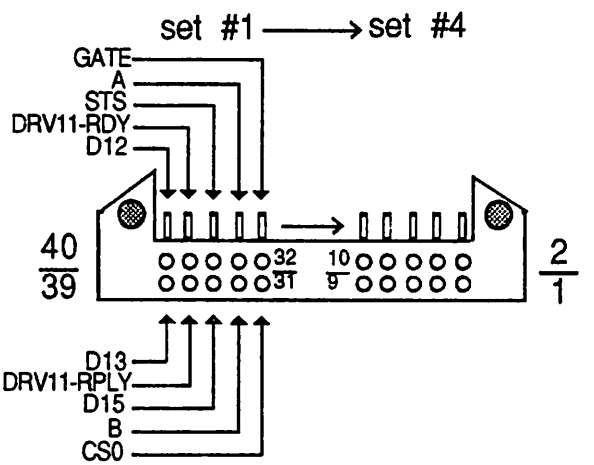


Figure 2.35: CSDSB - Connector pinout

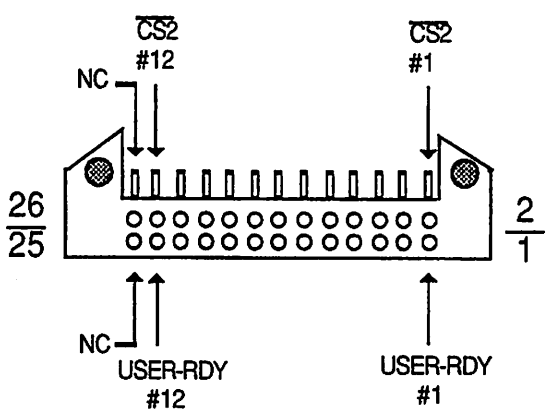
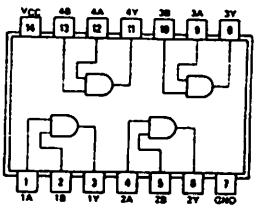


Figure 2.36: CSDSB – Extra connector pinout

QUADRUPLE 2-INPUT
POSITIVE-AND GATES

08

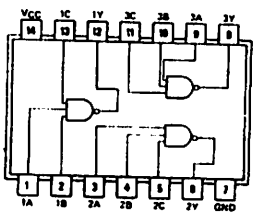
positive logic:
Y = AB



TRIPLE 3-INPUT
POSITIVE-NAND GATES
WITH OPEN-COLLECTOR OUTPUTS

12

positive logic:
$Y = \overline{ABC}$



QUADRUPLE 2-INPUT
POSITIVE-OR GATES

32

positive logic:
Y = A+B

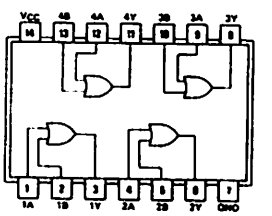
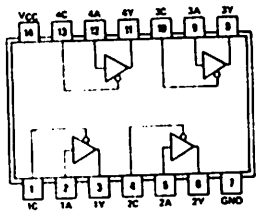


Figure 2.37: CSDSB – IC pinouts

QUADRUPLE BUS BUFFER GATES WITH THREE-STATE OUTPUTS

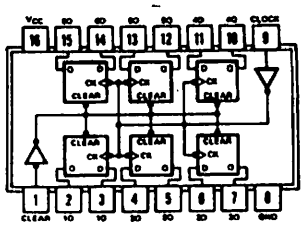
125

positive logic:
Y = A
Output is off (disabled) when C is high.



HEX D-TYPE FLIP-FLOPS

174 SINGLE-RAIL OUTPUTS
COMMON DIRECT CLEAR



OCTAL BUFFERS/LINE DRIVERS/LINE RECEIVERS

244 NONINVERTED 3-STATE OUTPUTS

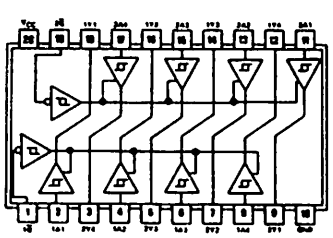
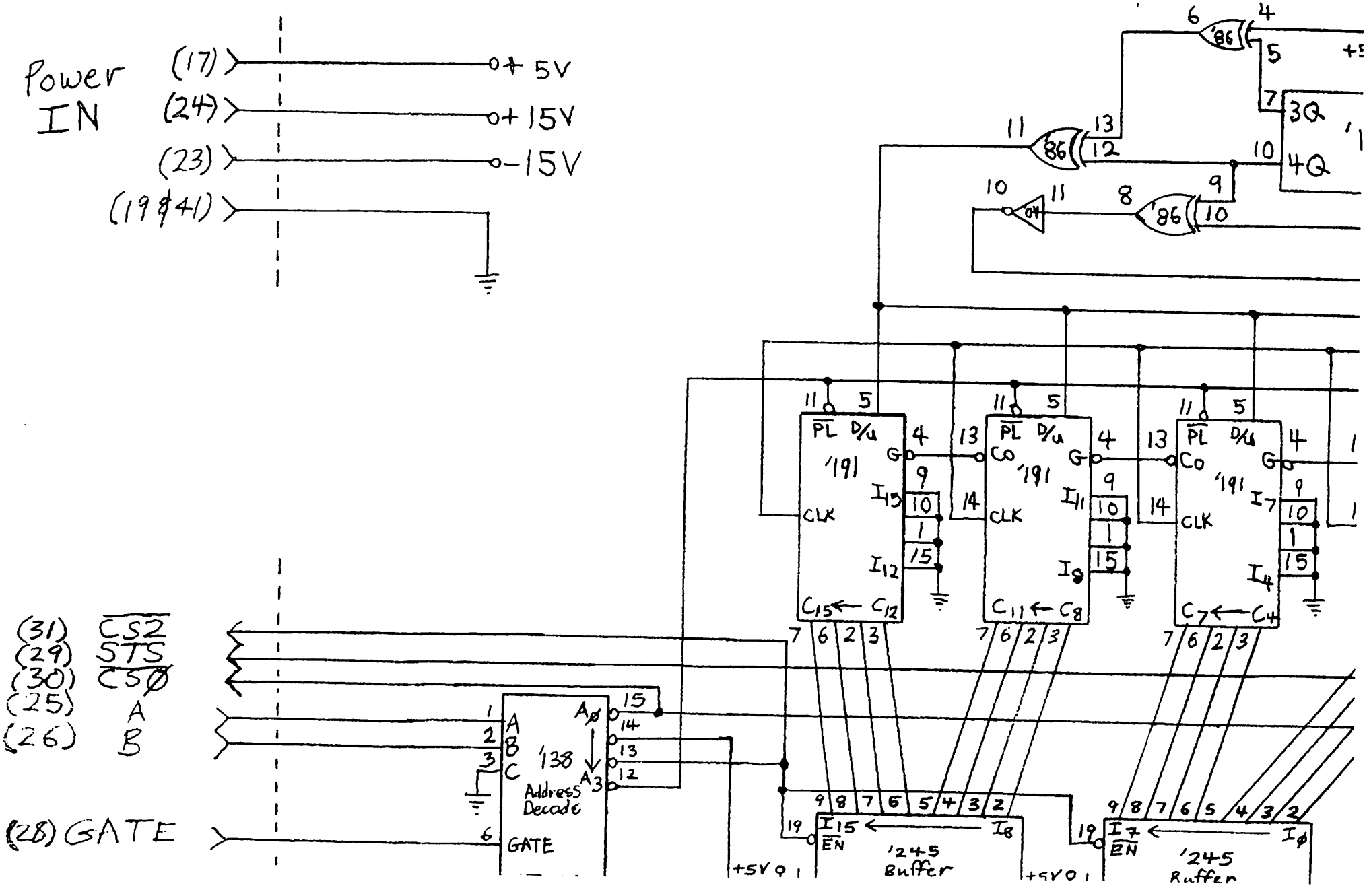


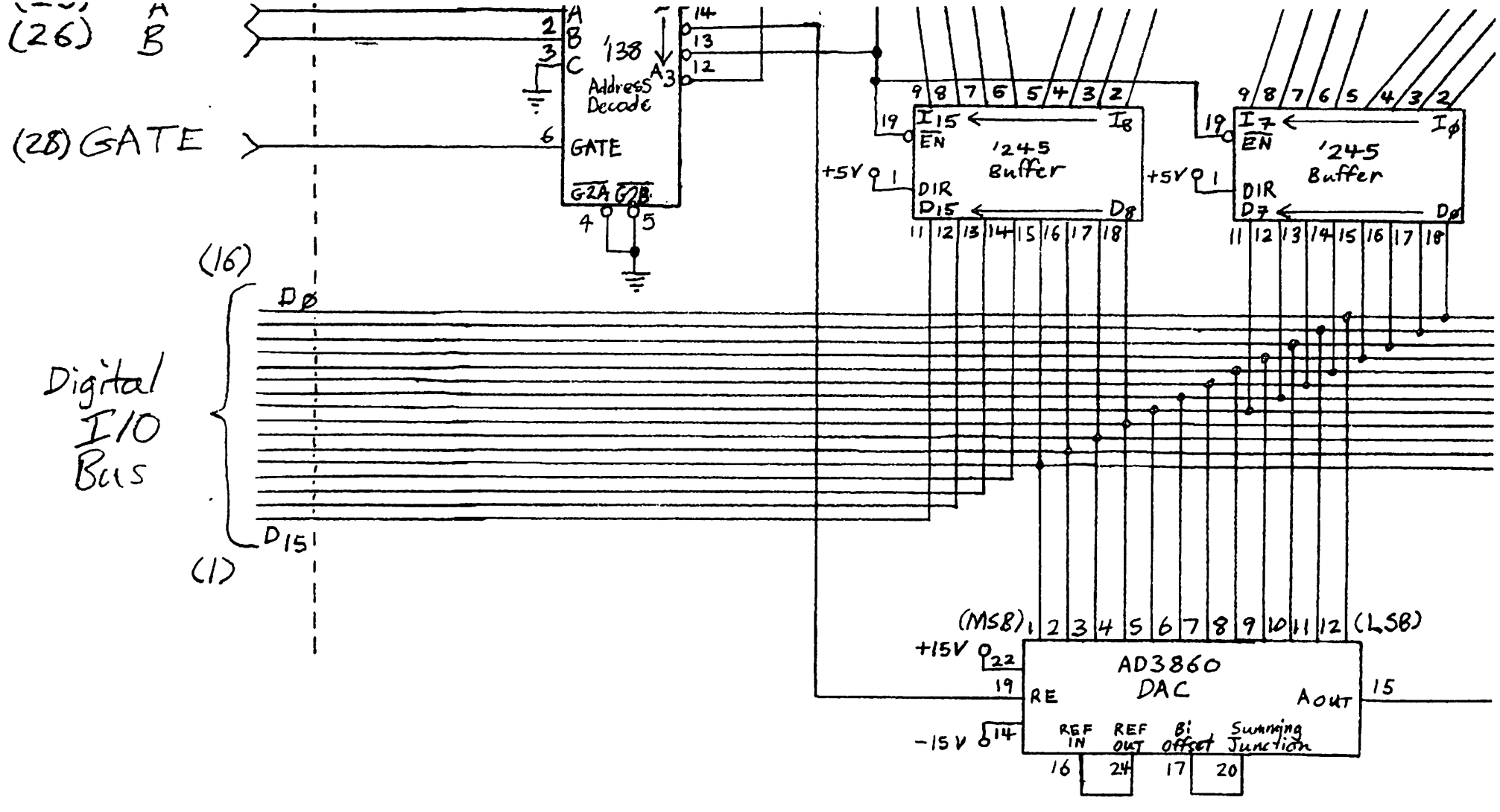
Figure 2.38: CSDSB – IC pinouts (cont.)

Item	*Description*	*Qty.*
TTL ICs	74LS04s	14
	74LS08s	12
	74LS12s	4
	74LS32s	9
	74LS125s	3
	74LS174s	12
	74LS244s	2
Vector Board	12 1/2" x 4 3/4"	1
Connectors (gold–plated, fully–shrouded right–angle FRC headers with wire–wrap pins.)	40–pin	3
	26–pin	1
Wire-wrap sockets	14-pin	42
	16-pin	12
	20-pin	2

Table 2.4: CSDSB – Component List

Edge-connector Pinout



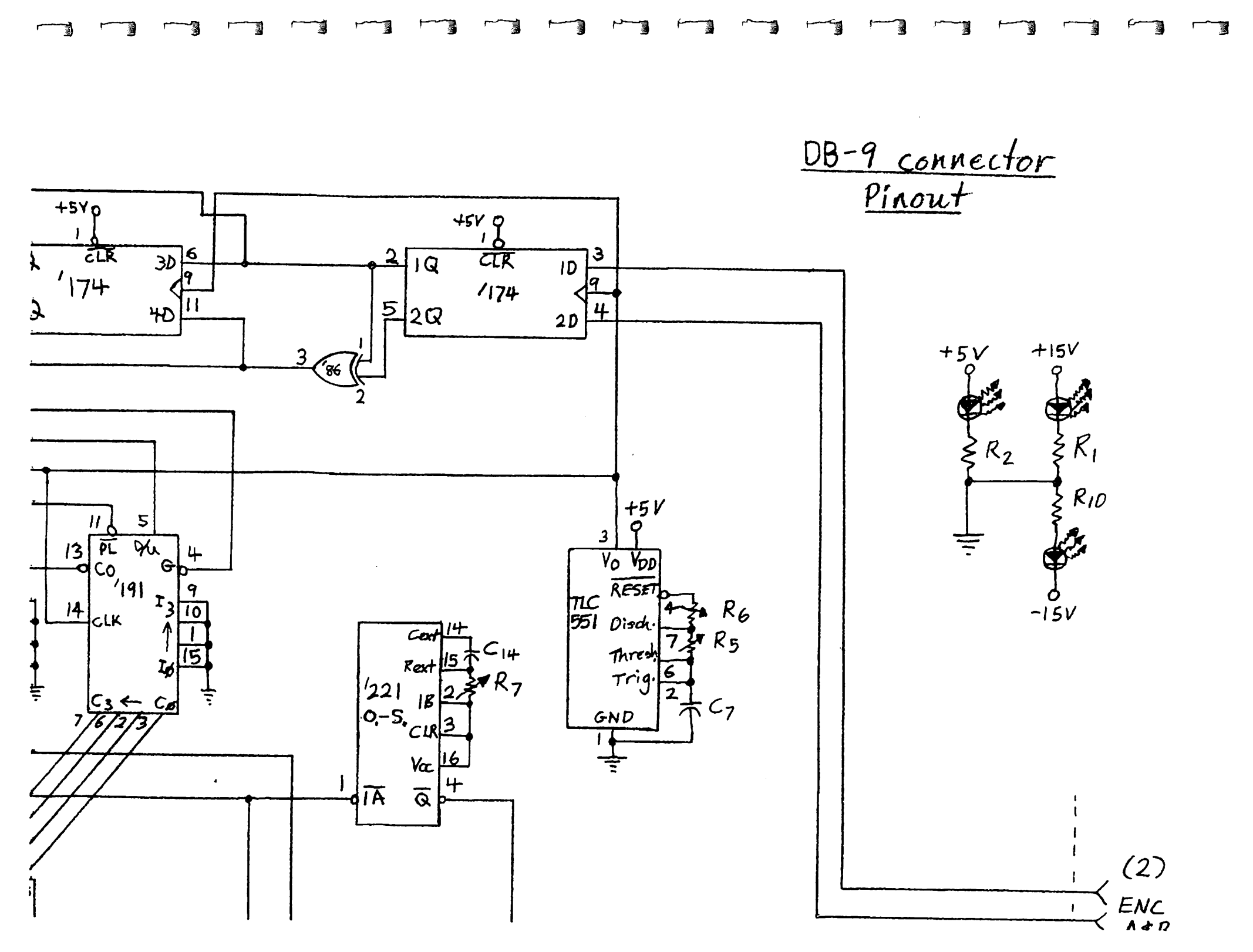


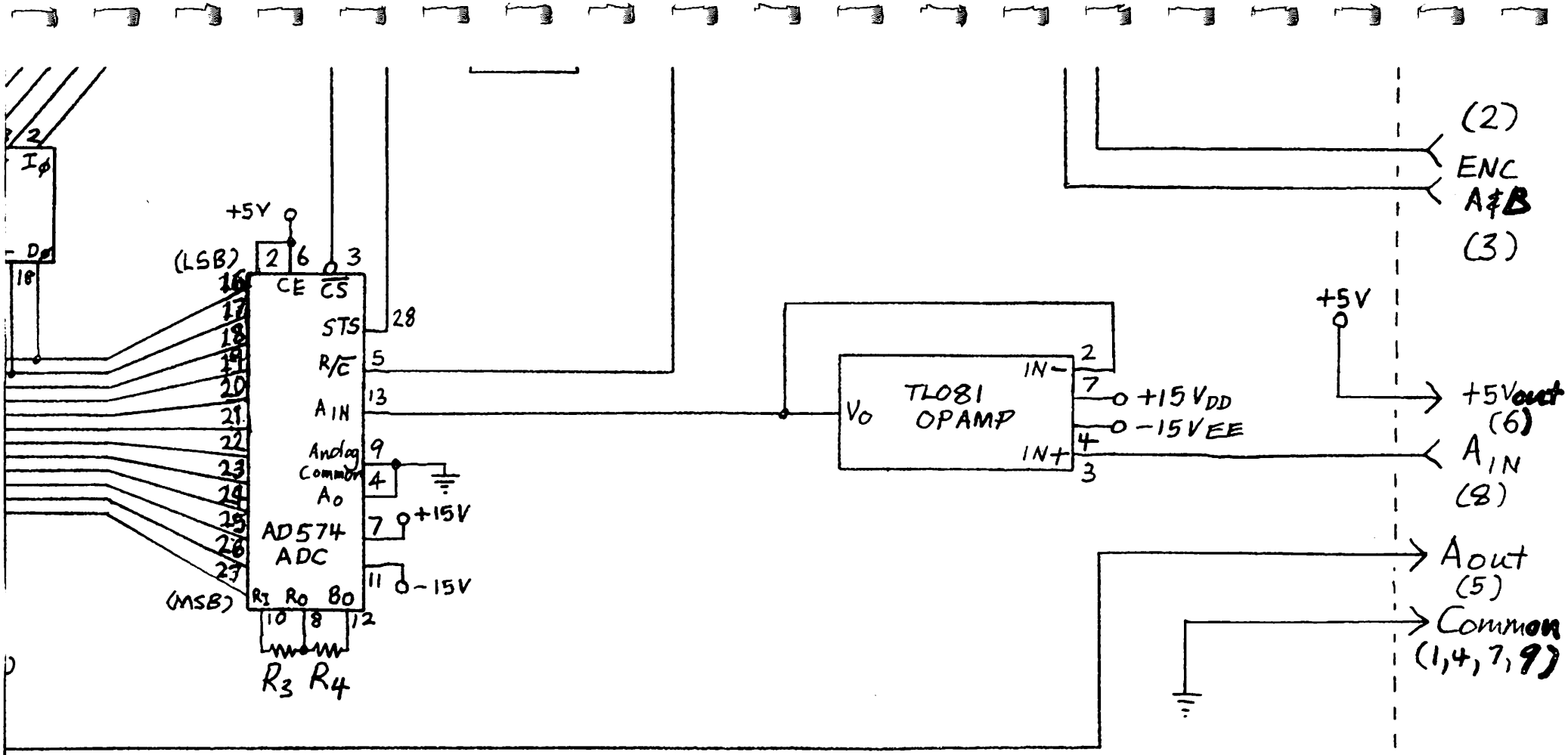
Note 1: This schematic pertains to an original-style board after the nec

Note 2: Overflow indicator schematic is not included because the indi

Note 3: Pinouts for the boards connectors are shown in parentheses.

Note 4: If new-style boards are etched using my suggested layout, connections at the edge connector will be reversed and will have to be reworked.


DB-9 connector
Pinout
+5V
CLR
3D
4D
'174
1Q
2Q
1D
2D
'86
+5V
CLR
'174
+5V
+15V
R2
R1
R10
-15V
PL
D/U
CO
CLK
I3
I0
C3 ← C0
'191
+5V
VO
VDD
RESET
TLC
551
Disch.
Thresh.
Trig.
GND
R6
R5
C7
Cext
Rext
CLR
Vcc
1A
Q
'221
O.-S.
C14
R7
(2)
ENC
A & B



necessary jumpers are installed.
indicator is unimportant,
es.
ut, the order of the data line
nd the wire-wrap backplane

2.3.3 Digital Equipment Corporation's DRV11J – 4–port 64–bit parallel I/O

How we use the DRV11J

DEC's board is capable of generating processor interrupts. However, as of present, the hand control hardware does not require use of the interrupt–generation capability. At this point, the DRV11Js serve simply as software–controlled parallel I/O data paths.

Please refer to Digital Equipment Corporation's documentation for specific information on the operation of this board. It is thorough and well–written. It is also a necessary companion to this technical manual.

Chapter 3

Troubleshooting the I/O hardware

3.1 General information

It has been my experience that, by far, most recurring hardware problems involve the mechanical connections of wires. Getting the best and most durable connectors is crucial. Taking care during the installation of connectors is important as well. Anyone planning to add further sensory capabilities to the system or to modify the I/O hardware make a special effort to modularize their circuitry as much as possible. Not only will this save you grief *when* you need to rearrange the system's layout, your system will also be more amenable to troubleshooting. Liberal use of well–placed, well–made and clearly–marked connectors is helpful. Also, dividing circuitry logically onto as many boards as is necessary makes fixing any hardware far more effective and efficient. Also, take time to work out a physical mounting arrangement that allows easy access to and removal of any module. If you pack the modules tightly you will probably regret it.

3.2 TTL signal problems

Here's an overview of symptoms that are commonly caused by open or shorted cable and connector lines, damaged power lines and the like. Of the problems that occur with TTL signals, there are two main types – problems that cause symptoms in many circuits simultaneously, and problems that only effect a single circuit. First, examples of problems with more widespread repercussions for the hand hardware are presented.

3.2.1 Broad effect TTL signal problems

Here's one example of a problem with many repercussions. If you look at the schematic for the STIO boards, you will notice that there are several control lines indicated. If any of the signals that these lines carry are lost due to a broken wire, a poorly–connected wire–wrap line or incompletely–inserted connector, you will see entire sections of the circuitry malfunction. Let's say the control line to pin A of the three to eight decoder is broken. This would cause signal A to be constantly read as a one. Thus, you would only be able to select two of the board's four functions, and the results of calling the two lost functions would be wrong since the software control sequences would be driving the wrong circuitry. Another example of a control line problem follows. If the DRV11J–RDY signal from one of the DRV11J ports is lost, you will notice that all the functions for the corresponding STIO board will be affected. Here is another problem. If Vcc for a set of STIO boards is loaded down excessively, you will see that the Vcc power indicators on the front of the STIO boards affected will go dim. Do not make the mistake of assuming that the power regulator was damaged. More likely than this is that a short has occurred somewhere and the regulator has gone into current–limiting mode. Disconnect the regulator output from all circuitry and check it. The voltage will probably read between +4.5 and +5 volts. The regulator is undamaged in this case. To trace the short, try reconnecting sections of affected circuitry until you see the voltage drop down again. When you find out which circuit or board causes the power to drop, you will be close to pin-pointing the problem. Another problem that shows up sometimes is a counter or A/D value always reading as if all its bits were ones. This is usually caused by either a control signal problem (perhaps the data lines are being kept tristated because the chip's enable signal has been lost) or a lack of Vcc to the digital logic on some or all of the STIO boards. If a main supply of power for Vcc voltage regulators should blow a fuse because of a shorted output line, a whole bank of Vcc indicator LEDs would go completely dark. Tracing and fixing a problem like this is simple.

3.2.2 Limited effect TTL signal problems

One symptom that is seen from time to time is erratic behavior by one of the counters. Possible causes of this include: a short between two adjacent output data lines, an open data line, or the shorting of a data line to ground. To determine the exact cause, get the diagnostic

software up and running and use the command that reads a counter and gives binary output. Run the command so that you can watch the bits toggle as you move the motor shaft. If one bit always reads high, check for a broken data line connection between the counter buffer and the DRV11J or between the counter and the counter buffer. Remember, an undriven TTL input will float high (read a one). If a bit always reads low, check for a data line shorted to a ground line or to the chassis. This will usually occur either on a board or in a connector. Sometimes two adjacent data lines will short together. A common result is that one of the outputs blows out. This happens when one output is driving a 1 and the other is trying to drive a 0. Therefore, if you do find a short, don't be surprised if you need to replace an IC or two. All of the information above applies to the A/D circuitry as well. As for clearing the counters, it is a trivial task. The only bugs you might see are a bad IC or perhaps an open parallel load line from time to time. There are rarely problems with the D/A converters. A jumpered data line opened up once, but it was fairly straight forward to find and fix.

3.3 Analog signal problems

Below is an overview of symptoms that are often a result of open or shorted cable or connector lines, or by power supply problems.

A bug that occasionally develops is a blown fuse on one of the power supply lines to a linear current amplifier. If this happens, you will see the motor driven by the affected amp begin to spin at a constant speed in some direction – regardless of the amplifier's input signal. Once the motor has spooled up all the loose tendon, the motor will apply some constant torque to the tendon until the amp is shut off. It is possible for an amplifier to be damaged in the above situation, so don't use fuses too near the amplifier's maximum current capacity.

The behavior exhibited by a dead amp is almost identical to the actions described above for an amp with a blown power supply, so beware. If, upon investigation of the supply voltages present *at the amplifier's barrier strip connections,* you find that power is being delivered to the amp, then check the amplifier. Also, before you rush off to change the PA-12 operational amplifier, make sure that the feedback resistor is properly connected. The linear current amp printed circuit board needs reworking. Currently, the boards have been hacked

a bit and are fairly exposed. If they are bumped a bit, a resistor or capacitor can come loose. There is a design trade off; it is difficult to completely protect the amplifier boards and still maintain easy access for repair and heat dissipation purposes.

3.4 Good connectors and bad connectors

As a general rule, connectors are a constant source of trouble. Many connectors tend to leave the ends of wires exposed. If you've got connectors like these in your hardware, suspect them first for causing such problems as shorting to a grounded chassis or to an adjacent wire. Occasionally you will have connectors that are under a fair amount of mechanical stress. These connectors may well develop open electrical connections due to broken wires or poor solder joints.

Several kinds of connectors are used in the hand hardware. These are the most common: D–subminiature, header, card edge connectors and barrier terminal blocks.

A discussion of D–subminiature connectors (sometimes called "delta connectors" because of their shape) follows. D–subminiature connectors are appropriate for cables carrying most any kind of signal including those produced by TTL or tactile sensors. Many of the available D–subminiature connectors can handle in the neighborhood of 5 Amps and around 1000 VAC. There are two main types of D–subminiature connectors: FRC (with mechanical connections) and solder pin. The FRC type can be purchased with some very useful optional hardware – namely, strain relief bars. Strain relief bars protect the mechanical connections from undue wear and tear which often results when cables are moved and pulled during system use. I approve of D–subminiature connectors for flat ribbon cable connections. Using D–subminiature connectors that have solder pins is a bit more tricky. These are used for standard RS–232 serial communication cables and the like. Regarding the assembly of such connectors, it is best to slide a piece of insulation up each wire before your solder it to a pin. This way, when you are finished, you can slide the insulation back down over the solder joint to restrain the wires from touching each other. This will be especially useful should a wire break free due to mechanical wear.

Header connectors are the most commonly used FRC connector used in industry. These connectors tend to pull loose from their mates more easily than do D–subminiature connectors. Several times I've traced bugs to a poorly inserted connector where one or two of

the outside pins have disconnected. Ejector latches are a useful option to get with header connectors since they allow you to extract the female connector without pulling on the cable. Pulling on the cable to disengage header connectors is definitely *not* recommended! It is the fastest way to break wires inside the connector. A more effective approach to separating headers that don't have ejector levers is to slip a thin flat-head screwdriver into the polarization slot and twist it. This will safely push the two connectors apart. If you do find that a wire has broken inside a header connector, you can open it up and carefully work the flat ribbon cable out of the pincers of the connector pins. Once you have gotten the cable loose from the pins and checked the pincers for damage (make sure the pincers are properly spaced and the pins are seated all the way down into the connector) then cut off the used end of the cable and clamp the connector together again on a new section of cable. Then trim any excess cable off and put the strain relief bar back on.

Many of the headers used in the lab are the "right-angle wire-wrap pin" variety. These are used to connect the FRC to circuitry on a lab-made wire-wrapped board (such as the CSDS board). There is nothing particularly different between these headers and the ones discussed earlier except that care should be given to the task of securing the male header to the perforated board. This should be done before the header is wired up.

Currently, the edge connectors used in the hand hardware all have wire-wrap pins. Through these pins, connections are made from the STIO boards to male right-angle wire-wrap header connectors. These connectors mate with the female headers on the cables of the CSDS board and the DRV11J. There is a way to polarize the edge connectors to prevent improper insertion of the STIO boards. There are small slots in the edge side of the connectors between the edge connector contacts. You may notice that a small piece of plastic has been slipped into one of these slots in each connector. These pieces of plastic polarize the connectors, making it impossible to insert the STIO boards upside-down. A corresponding notch has been cut into the edge of each STIO board. Should the piece of plastic fall out of one of the edge connectors, it would be possible to ruin a board by inserting it improperly. Note that all of the STIO boards should have the same physical orientation in the backplane. Here is a safety precaution to follow when troubleshooting the hardware. Remember that whenever you are working around the backplane with the system powered-up, avoid shorting the +15V or -15V power bars to each other or to TTL data pins. The damage caused by

doing so would be serious and expensive to repair!

Most analog signal and power lines pass, at some point, through barrier strip connections. These should be used in most any hardware you design since they are an excellent way to break circuitry into modules that can be tested independently.

3.5 The diagnostic software

Diagnostic code has been written in both C and LPR–FORTH for testing the operation of the JPL/Stanford hand. The program works quite well and its routines may easily be incorporated into user programs. When the C code was written, special care was taken to break the testing problem into low–level functions that could be used to easily construct custom functions for a given troubleshooting need. Some possible extensions to the diagnostic code are a line parser and user determination of command looping time. Below, is a list of the interactive commands made available by the C program. The code is shown in appendix A at the end of this report.

3.5.1 C–code interactive commands

Since a line parser has not been included in the current version of the diagnostic program, the commands listed below must be entered with a carriage return following each command word or parameter.

```
        Diagnostic commands -- names and descriptions
        ---------------------------------------------
         '?' or 'help'  Print command list.
              w <1-12>  Set the control line 'DRV11-J' to write.
              r <1-12>  Set the control line 'DRV11-J' to read,
  wr_csr <1-12><value>  Write an octal value to
                           control/status register.
  wr_bfr <1-12><value>  Write an octal value to an output
                           buffer.
           rd_bfr <1-12>  Print the value currently on the data bus.
         b_rd_bfr <1-12>  Print the value currently on the data bus
```

```
                    in unsigned binary form.
        cc <1-12>   Clear a counter.
       ccf <1-3>    Clear a finger's counters.
             cch    Clear all counters.
        rc <1-12>   Read a counter.
       rcf <1-3>    Read a finger's counters.
             rch    Read all the counters.
       rad <1-12>   Read an A/D.
      radf <1-3>    Read a finger's A/Ds.
              radh  Read all the A/Ds.
     b_rc <1-12>    Read a counter in binary.
    b_rcf <1-3>     Read a finger's counters in binary.
   b_rad <1-12>     Read an A/D in binary.
   b_radf <1-3>     Read a finger's A/Ds in binary.
wda <1-12> <value>  Write a value to a D/A.
          powerupf  Write out 2048 to one finger's
                    DACs for a while.
           powerup  Write out 2048 to all DACs for a
                    few seconds.
             loose  Write 2048 to all DACs for some
                    to protect hand from high currents
                    caused by corruption of DAC values
                    when supply voltage is fed to the
                    linear current amps.
    loosef <1-3>    Write out 2048 to one finger's DACs.
            loose   Write out 2048 to all finger's DACs.
 describe_coms1     Print a description of some commands.
 describe_coms2     Print a description of some commands.
             loop   Execute a command in a loop until ^C.
             exit   Quit the diagnostic code.
```

Chapter 4

How to Maintain the JPL/Stanford Hand Mechanism

First, to work on the hand you'll need some basic tools and supplies. These include a full set of American standard allen wrenches, a few small screwdrivers, some rubbing alcohol, some light-weight oil (Dr. Salisbury recommends a special oil containing beads of teflon to reduce friction effects), some absorbent cloth or paper, teflon-coated steel cable, crimps and a crimping tool. Regarding the crimps and cable, please send inquiries to Dr. Kenneth Salisbury at Massachusetts Institute of Technology, Computer and Information Science Department.

In discussing replacing the tendons of the hand, terminology illustrated below in Figures 4.2 and 4.3 will be used. These figures also show a finger as it appears when disassembled. Please refer to these figures if you are unsure about the construction of the hand's fingers.

4.1 How to take apart a motor/gear/spool assembly

Look at Figure 4.1 for clarification of terms used to describe the various parts of the motor housing. The first step in the process of taking apart the motor is removing the yolk from the large gear and the fastening ring from the small gear on the motor shaft. Then remove both gears and their washers. You now have access to all the four large allen bolts that secure the body of the motor housing. Remove these bolts now. Once the bolts are removed, you should grab the shaft to which the large gear was attached and then carefully lift up the body of the motor housing. Set the housing upsidedown so that the shaft you are holding doesn't

fall out. You will now see the main cable spool with the cable wound about its middle. Lift up the spool and remove its washers. Once that's done you may push the tendon crimp out of its receptacle in the spool and clip the crimp off if you need to replace the tendon. Also, you can lift out the shaft you held onto earlier and retrieve its washer from inside the motor housing. Lastly, the motor itself is held to the housing by two screws. You'll hopefully never need to take a motor out of the housing.

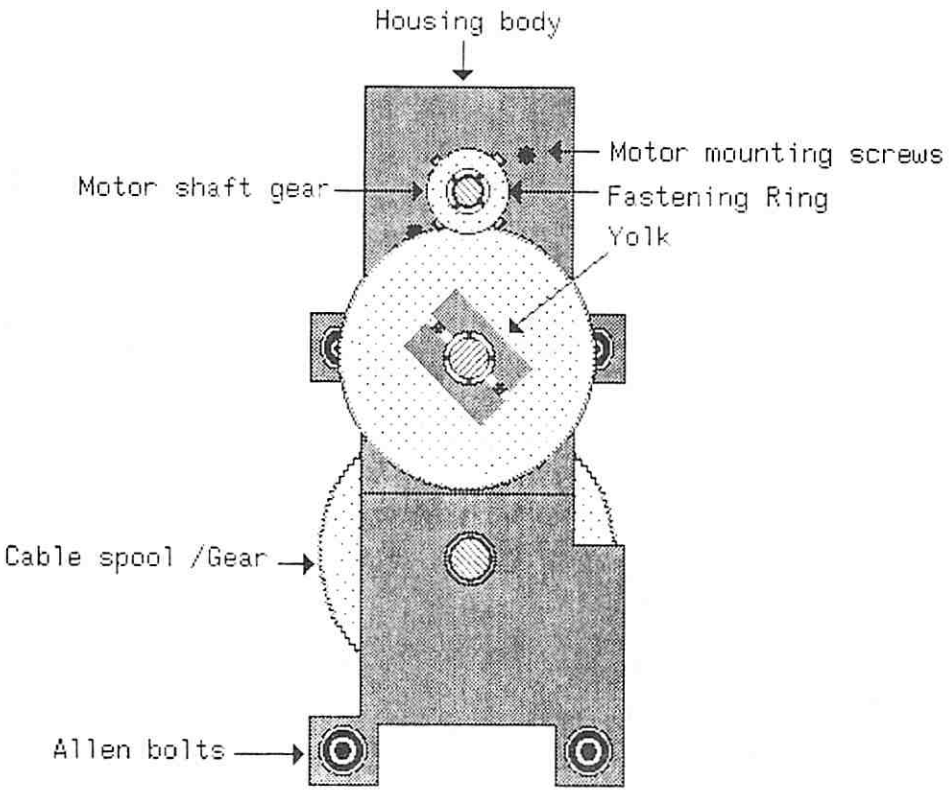


Figure 4.1: A close-up of a motor/gear assembly

4.2 Cleaning the gears and gear housing

To clean the components of the motor gearing assembly, just wipe everything down with rubbing alcohol and then maybe put a drop of oil in the obvious places. You might want to use a brush to clean the gear teeth sometimes. That's it.

4.3 How to replace a tendon

When giving the hand routine maintenance, it is necessary to occasionally replace the steel tendon cables. Dr. Kenneth Salisbury has stated that normal tendon wear does not account for the need to replace tendons as often as we have. It is certainly true that the LPR's hand has seen some indelicate usage at times — mostly during the early days of controller design and installation.

To replace a tendon, you must first take apart the tendon's motor housing. How to do this is described in Section 4.1. Once this is done, push or pull the crimped end of the tendon out of its receptacle in the tendon spool, and then clip the crimp off. This allows you to remove the tendon from the spool. Now free the other crimped end of the cable from its connection to the finger assembly.

The JPL/Stanford hand tendons take one of two paths — terminating either in the third phalange under the fingertip, or in the side of the second phalange. If the tendon terminates in the second phalange, it is very simple to merely slip the crimp out of its slot in the side of the phalange, and then to pull the entire tendon out the hand. The tendon should easily slide through whole hand mechanism. If the tendon you wish to replace terminates at the third phalange, you must remove the yellow polymer fingertip in order to get at the end of the cable. You do this by using a small allen wrench to loosen the fingertip's set screw. After the fingertip is removed, simply pull the appropriate tendon out.

Regarding putting the new tendons in, the first part of the operation is threading the tendon cable through the tendon conduit and around the finger pulleys. The second part is getting the new crimps on the tendons and securing them properly. Both of these operations take a little practice.

The reason it is hard to thread new cables is that there are small teflon inserts that are fitted into the end of the steel conduit. These inserts protect the tendon from scraping

against the ends of the conduit. There is a small space between the end of the teflon conduit liner and the lip of the teflon insert. When you push the cable through the conduit, you'll find that the teflon on the tip of the cable tends to get pushed back very slightly. When the tendon reaches the end of the teflon liner, the steel strands will spread apart a bit, and this will make it difficult to pass the cable into the insert.

There are several approaches to getting around the problem mentioned above. The easiest approach is to wiggle the conduit near the insert a bit while lightly pushing the cable in from the other end.

As for applying new crimps to a cable, the crimps Ken Salisbury told us to use appear to be too small. Several people have told me that our crimps are the appropriate (including people at Sava Industries who make both the crimps and cable), but noone in the Laboratory for Perceptual Robotics has had any success at putting on these crimps. The proper way to apply crimps is to first slide the crimp over the teflon coating, then peel enough of the teflon off of the tip of the cable to accommodate the crimp, and lastly to slip the crimp back onto the exposed steel and squish the crimp into place using a special crimping tool. We have had success following these steps, but only after drilling out a very thin layer of the crimp's hole.

4.4 How to take one of the fingers apart

To break a finger down into its components parts, begin at the fingertip and then work toward the knuckle of the finger. Once you have taken off the fingertip and removed all the tendons, remove the phalanges one by one until you have only the unopened knuckle which contains the strain gauges left. Do not open the knuckles unless truly necessary, since the strain gauges are delicate. If you do open a knuckle, don't disconnect the strain gauges from their mounting position. It is not trivial to reseat them properly. Once in a while you may need to blow the dust out of the knuckle compartment, but that should be it. Obviously, special care should be taken to keep track of all the steel and plastic washers used in the hand. The steel washers are not all the same thickness or diameter, so remember where you find them during this work.

4.5 Cleaning the finger components

Once the finger is completely apart, cleaning is a simple process of wiping the components down with rubbing alcohol. This will take any oil, dust, or powder off. Powdered aluminum sometimes shows up in or around the finger joints. This is because some wear occurs in the finger joints where the surface of a phalange rubs against the head of the pivot bolt. This has been a source of some concern to me, since there is obvious play in one of the joints of one of our hand's fingers.

A couple of individuals have told me that aluminum bearings, such as the ones in the JPL/Stanford hand, are light and exhibit very little friction. In an effort to find ways to reduce the play in the in the worn joint, using copper shims was tried. These were difficult to insert and either caused jamming or simply fell out. One person suggested having a special brass fitting put into the finger. Once the components are clean, reassemble the fingers and then restring all the tendons.

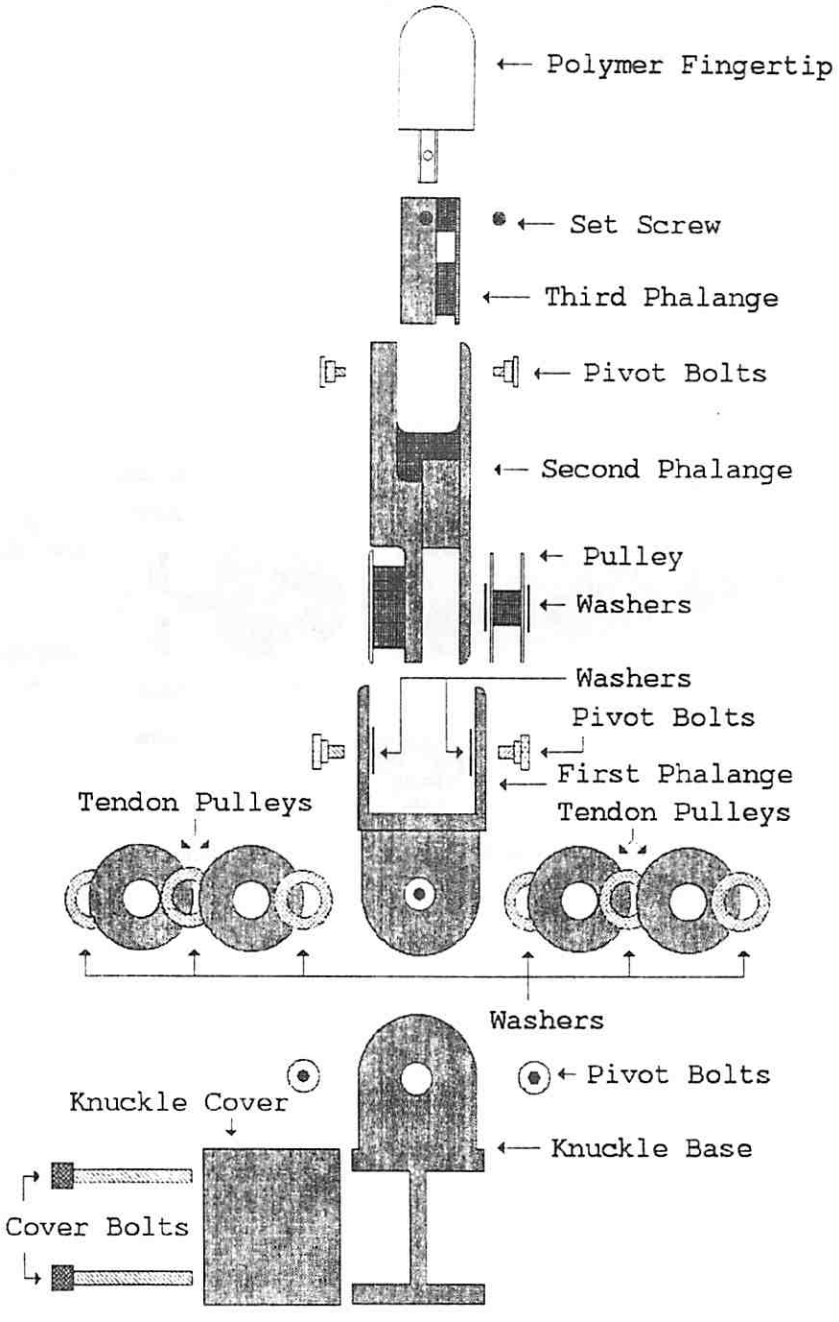


Figure 4.2: A disassembled finger – top view

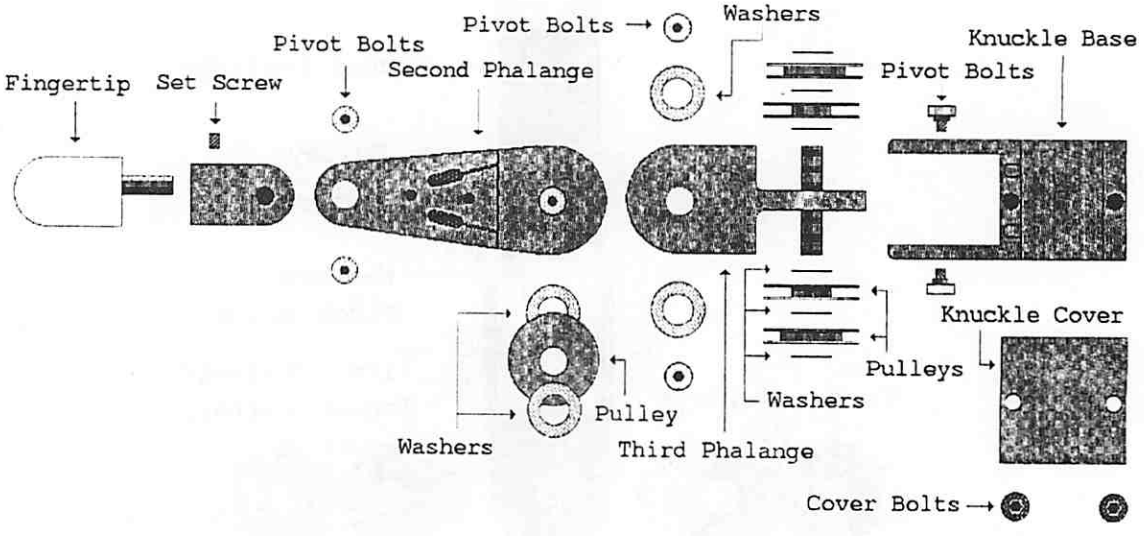


Figure 4.3: A disassembled finger – side view

Chapter 5

The Power Supply Hardware

Please begin your study of the power supplies by taking a good look at the power supply system block diagram in Figure 5.1. Notice that there are seven main power supplies in the present configuration of hardware.

To see how AC voltage is distributed to the seven power supplies, check out Figure 5.2. This relay circuit is fairly straight-forward in design. When the NOP switch is pressed closed, current conducts through the line labeled "brown" in the "Relay Circuitry" section of the figure. The current flows through the "brown" line and thereby through the relay coil. This causes the solenoid to snap down – closing the switches across its three sets of contacts. Now when the NOP switch is released, current continues to flow through the solenoid coil. The coil current is sustained through the relay contacts labeled "brown" and "white" and then on through the NCL switch to the coil. Something that may not seem obvious to the reader is that the relay coil is in parallel with the power supplies. That is, current flowing to the supplies connected to "Plug 2" does not pass through the coil that activates the solenoid. Please study the power distribution circuit until this point becomes clear. Finally, to shut power to the system off, the NCL switch may be opened. This will break the flow of current through the relay coil – releasing the solenoid.

Two of the seven power supplies are able to deliver approximately +12 volts at about 14 amps apiece. A drawing of the physical layout of these supplies is shown in Figure 5.3.

Beside the two high-current supplies, there are three multi-fixed-voltage supplies. These multi-output amps each provide +12 volts, -12 volts and +11 volts. These supplies were taken out of obsolete university equipment. We have been unable to acquire documentation

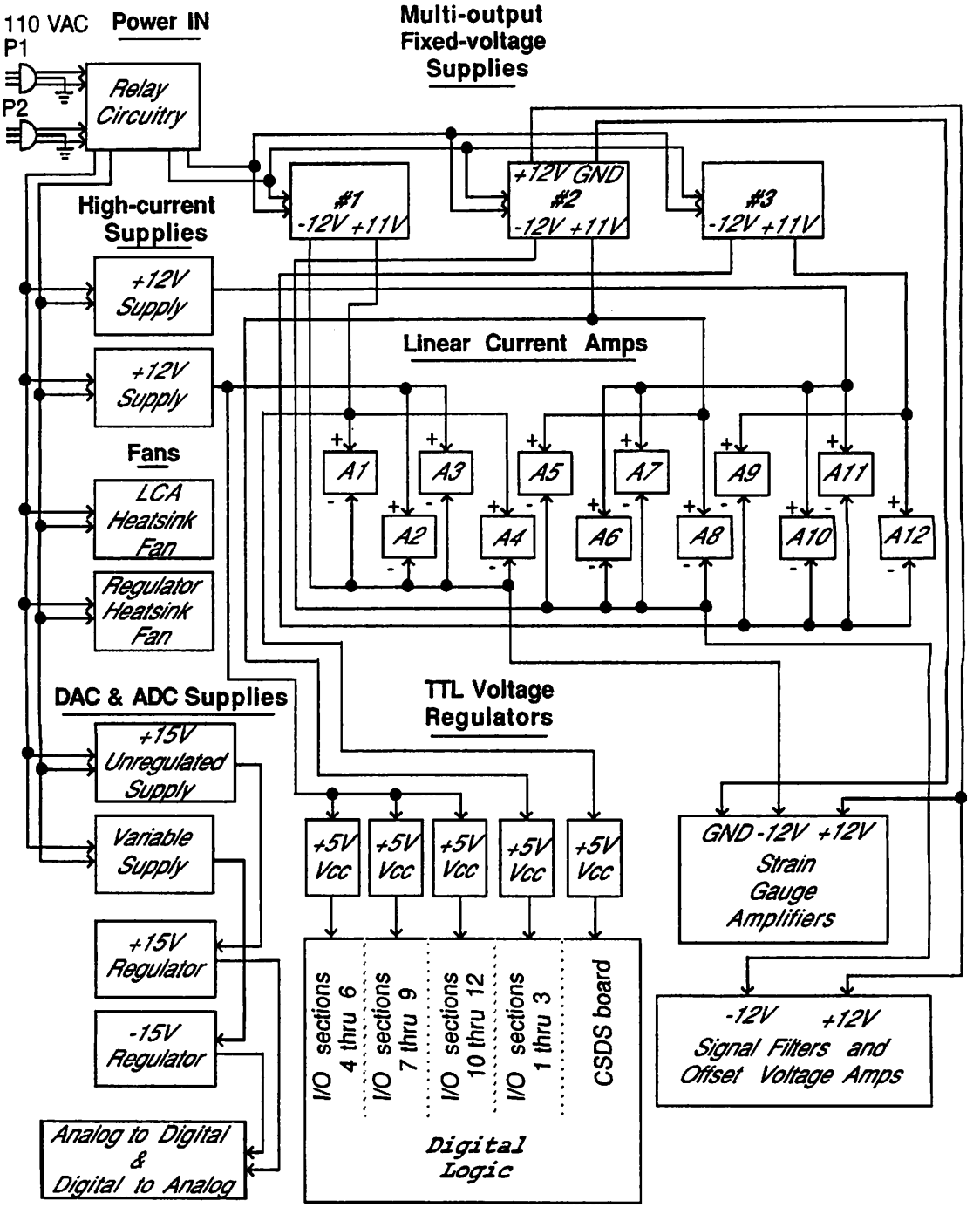


Figure 5.1: A block diagram of the system power supplies

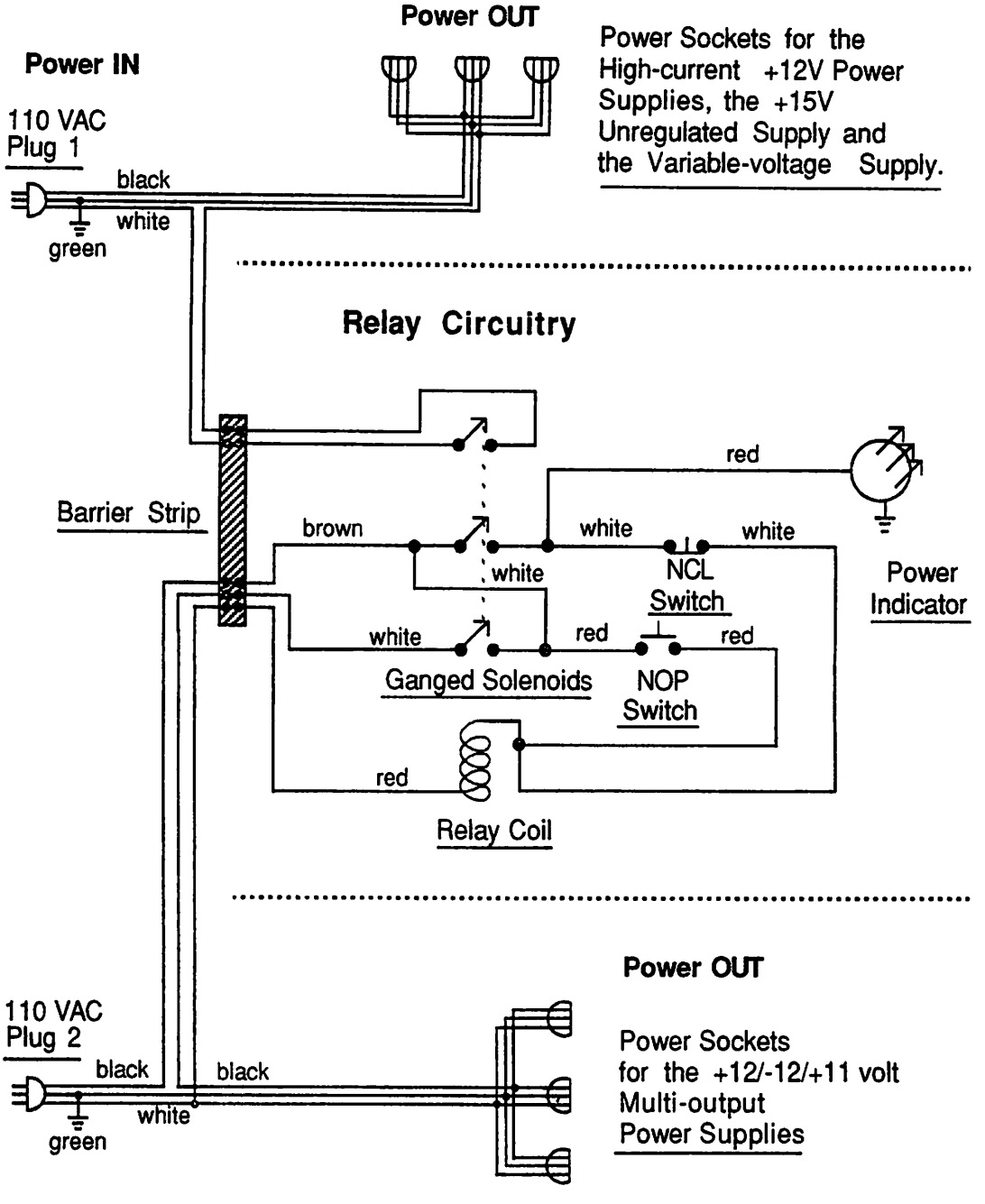


Figure 5.2: A power relay schematic

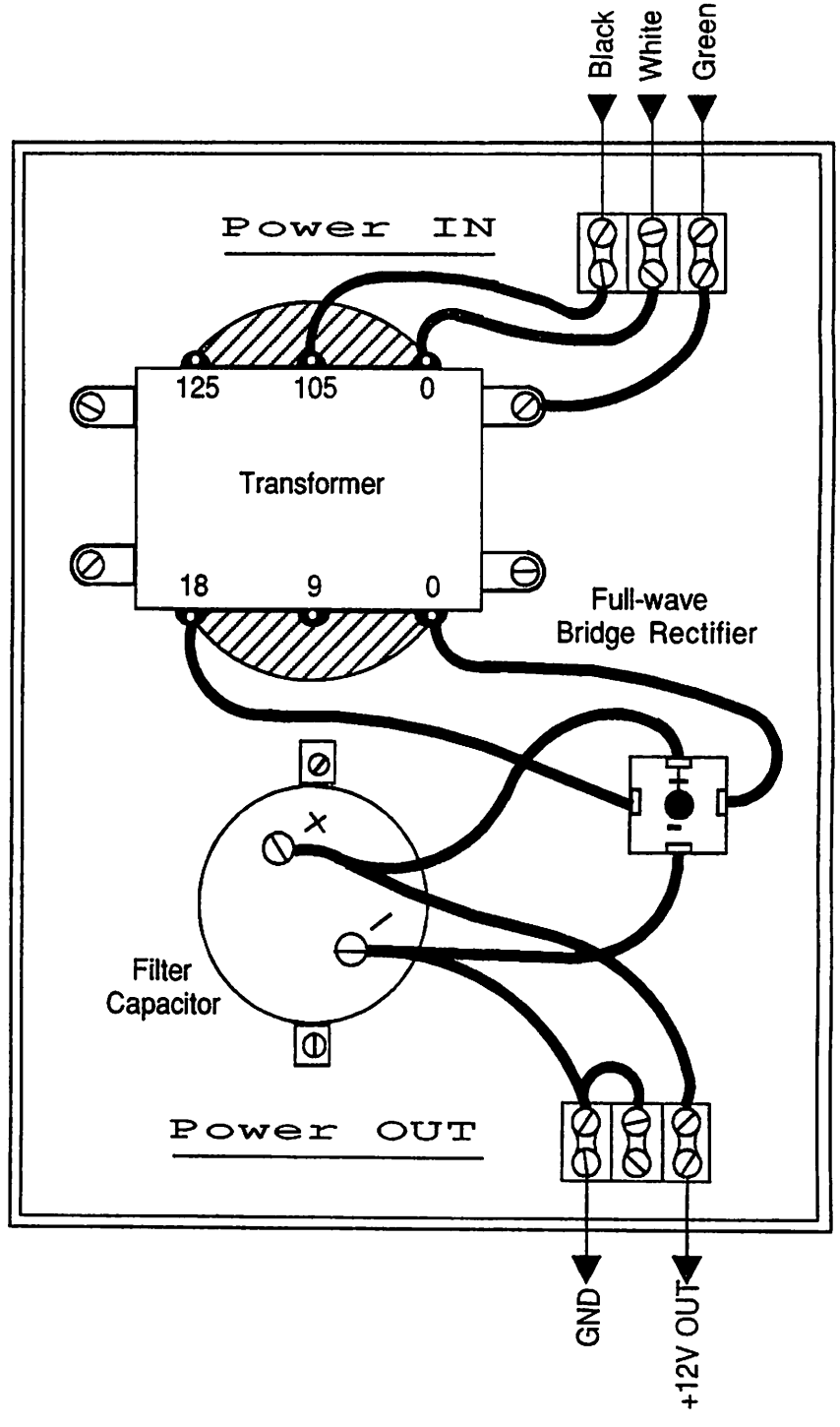


Figure 5.3: The physical layout of the +12 volt supplies

for them other than the voltage and current ratings of the various outputs. Therefore, there is no schematic for them below. The linear current amplifiers (as shown in Figure 5.1) draw all their power from *five* supplies. These are the two +12 volt supplies and the three +12/-12/+11 volt supplies described above.

A schematic showing the +12V supplies, the wiring of the hand's main power switch and the routing of +11 volts from the multi-voltage supplies is presented in Figure 5.4.

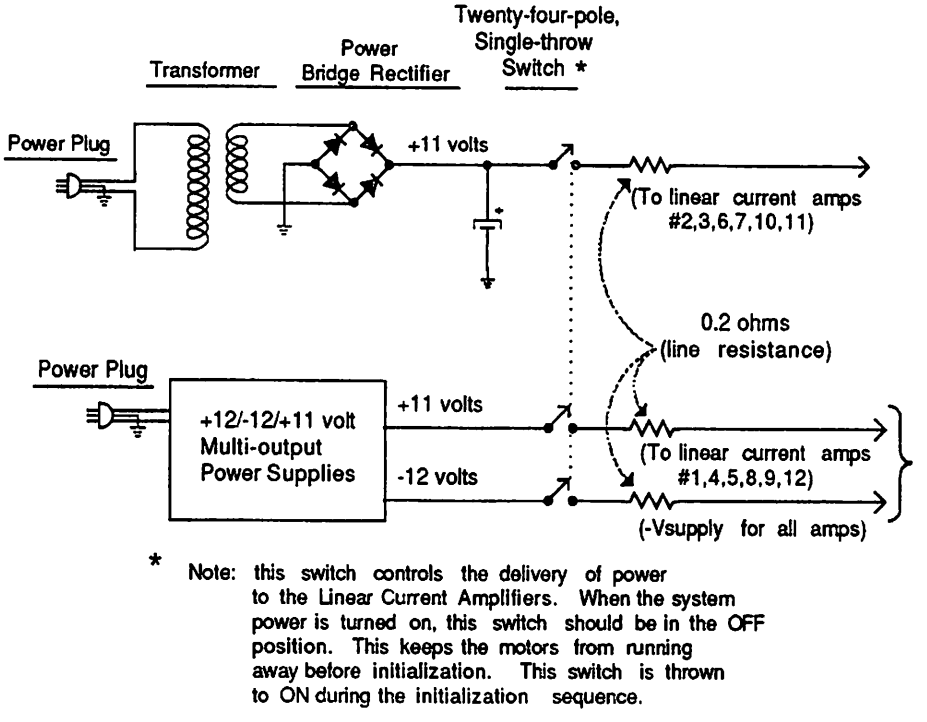


Figure 5.4: A power schematic for the linear current amps

These two supplies were put together in the lab, and have only brute force voltage regulation (big capacitors, in other words). These supplies provide the positive supply voltage to half of the linear current amplifiers. Also, power from one of these supplies is regulated by three +5 volt regulators. The regulated +5 volt output drives a large chunk of the digital TTL circuitry.

One supply is a variable output voltage supply. Its differential output voltage is set to

25 volts, the positive terminal is tied to ground, and its negative terminal output is fed into a -15 volt regulator. The regulator's output is the negative supply voltage for the D/A and A/Ds.

The last supply is a fixed-voltage +15 volt supply. The output is further regulated and then is fed to the A/D and D/A circuitry.

Appendix A

The C diagnostic code

```
#include $vaxelnc
#include descrip
#include signal

#define BUFFER_LENGTH 512

#define YES     1    /*  Flags for controlling 'user_io' execution.     */
#define NO      0
#define INRANGE 1    /*  Flags indicate whether a numeric value entered  */
#define ERROR   0    /*  by user is in or out of range.                 */
#define TRUE    1
#define FALSE   0

 /*
    The struct below is set up to take advantage of the fact that the
    DRV11-J registers are contiguous. These registers are memory-mapped.
 */

struct drv_register_def {
/* board 1 */
  short csr0;
  short buf0;
  short csr1;
  short buf1;
  short csr2;
  short buf2;
  short csr3;
  short buf3;

/* board 2 */
```

```
  short csr4;
  short buf4;
  short csr5;
  short buf5;
  short csr6;
  short buf6;
  short csr7;
  short buf7;

/* board 3 */
  short csr8;
  short buf8;
  short csr9;
  short buf9;
  short csr10;
  short buf10;
  short csr11;
  short buf11;
} *drv_register_ptr;      /*  The '*' causes a register (four bytes) to be
                              created with the name "drv_register_ptr".
                              Eventually, it will point to the base-address
                              of the memory-mapped registers. NOTE: The
                              memory-mapped space is not allocated at this
                              time. */

 /*
      Below is a structure definition whose format is determined
      by the requirements of ELN. The I/O code doesn't make use of these
      variables, but the structure must be passed in order to create our
      own device driver.

      This is true of both 'region' structures.
 */
struct drv_region_def {
     char    buffer[BUFFER_LENGTH];
     int     read_cnt;
     int     buf_ptr;
     BOOLEAN read_in_progress;
     BOOLEAN error;
} *drv_region_ptr;


/***************************  MAIN  *******************************/

main()
{
```

```
  unsigned short int i1, i2;
  VARYING_STRING(32) device_name_string;
  DEVICE drv_device;      /*  This 'DEVICE' line tells the compiler to  */
  int i, j;               /*  prepare for creating a device driver.     */
  int *drv_adapter, *drv_vector, drv_ipl, drv_status;
  char test;


   /*
      This VAXELN function sets up some device default
      stuff that VAXELN needs.
   */

  $DESCRIPTOR(drv_name,"DRV");     /*  the function "$DESCRIPTOR" is    */
                                   /*  included from "descrip" above.  */


  void drv_service_routine();  /*  VOID tells the compiler that these  */
                               /*  won't be returning values.          */
                               /*  This results in faster code.        */

/*
   Here, the device driver is created. Note: many holes intentionally
   left by the compiler. These are filled in during EBUILD, which
   builds the real-time operating system around the compiled
   executable image.
*/
  printf("Creating DRV11 device driver...\n\n");
  ker$create_device(NULL, &drv_name,
      0, drv_service_routine,
      sizeof (struct drv_region_def),
      &drv_region_ptr, &drv_register_ptr,  /* The '&' causes the base-  */
      &drv_adapter, &drv_vector, &drv_ipl, /* address to be loaded into */
      &drv_device, sizeof(DEVICE), NULL);  /* pointer variables at this */
                                           /* time.                     */


/*  User interaction starts here. */

  scanf("%c", &test);      /*  Pause until keystrike.                */
  user_io();               /*  Starts up the user command interface.*/
}


/****************  FUNCTION DEFINITIONS:  ********************/
```

```
/*********************************************************************/
/*                                                                   */
/*  REMEMBER TO WRITE a new SSCANF and SCANF for this program.       */
/*                                                                   */
/*  This is necessary because the current ones don't handle ^Z and CR */
/*  in a way that is best for this program. CRs are seen as null     */
/*  characters until an other character is entered. ^Zs are seen     */
/*  as invalid entries rather than EXIT requests.                    */
/*                                                                   */
/*  The SSCANF needs to be rewritten to handle missing input, etc.   */
/*                                                                   */
/*********************************************************************/


user_io()
{
  int i, found, sw_index, b, port, finger, bit, run, value;
  char string[30], c;
  char *com_input[] = {
      "?", "w", "r", "wr_csr", "wr_bfr", "rd_bfr", "b_rd_bfr", "cc",
      "ccf", "cch", "rc", "rcf", "rch", "rad", "radf", "radh", "b_rc",
      "b_rcf", "b_rad", "b_radf", "wda", "tg", "powerupf",
      "powerup", "tightf", "tight", "loosef", "loose", "describe_coms1",
      "describe_coms2", "loop", "help", "exit"};


  printf("Entering user_io()\n\n");
  question_mark();
  run = YES;
  while(run == YES)
   {
     printf("    Enter command> ");
     scanf("%s", string);
     for(sw_index=0, found=0; (sw_index<34) && (found == 0); ++sw_index)
     {
       if(strchk(string, com_input[sw_index]))
        {
          found = 1;
          c = (sw_index + 48);
        }
     }
     if(found != 0)
       switch(c) {

      case '0':
```

```
    question_mark();
    break;
  case '1':
    if(get_port(&port))
      DIR_to_write(port);
    break;
  case '2':
    if(get_port(&port))
      DIR_to_read(port);
    break;
  case '3':
    if(get_port(&port))
      if(a_get_val(&value))
        wr_csr(port,value);
    break;
  case '4':
    if(get_port(&port))
      if(a_get_val(&value))
        wr_bfr(port, value);
    break;
  case '5':
    if(get_port(&port))
    {
      i_rd_bfr(port);
      putchar('\n');
    }
    break;
  case '6':
    if(get_port(&port))
    {
      b_rd_bfr(port);
      putchar('\n');
    }
    break;
  case '7':
    if(get_port(&port))
      cc(port);
    break;
  case '8':
    if(get_finger(&finger))
      ccf(finger);
    break;
  case '9':
    cch();
    break;
  case ':':
```

```
    if(get_port(&port))
    {
      rcl(port);
      putchar('\n');
    }
    break;
  case ';':
    if(get_finger(&finger))
    {
      rcf(finger);
      putchar('\n');
    }
    break;
  case '<':
    rch();
    break;
  case '=':
    if(get_port(&port))
    {
    rad(port);
      putchar('\n');
    }
    break;
  case '>':
    if(get_finger(&finger))
    {
      radf(finger);
      putchar('\n');
    }
    break;
  case '?':
    radh();
    break;
  case '@':
    if(get_port(&port))
    {
      b_rcl(port);
      putchar('\n');
    }
    break;
  case 'A':
    if(get_finger(&finger))
    {
      b_rcf(finger);
      putchar('\n');
    }
```

```
    break;
  case 'B':
    if(get_port(&port))
    {
      b_rad(port);
      putchar('\n');
    }
    break;
  case 'C':
    if(get_finger(&finger))
      b_radf(finger);
    break;
  case 'D':
    if(get_port(&port))
      if(b_get_val(&value))
        wda(port, value);
    break;
  case 'E':
  if(get_port(&port))
      if(get_bit(&bit))
        toggle(port, bit);
    break;
  case 'F':
  if(get_finger(&finger))
      powerupf(finger);
    break;
  case 'G':
    powerup();
    break;
  case 'H':
    if(get_finger(&finger))
      tightf(finger);
    break;
  case 'I':
    tight();
    break;
  case 'J':
    if(get_finger(&finger))
      loosef(finger);
    break;
  case 'K':
    loose();
    break;
  case 'L':
    a_describe_coms();
    break;
```

```
      case 'M':
        b_describe_coms();
        break;
      case 'N':
        /* loop(); */
        break;
      case 'O':
        question_mark();
        break;
      case 'P':
        run = "no";
        break;
      default:
        printf("\nIt's time to check out the 'switch' statement!\n");
        break;
    }
    else
     if(run == YES)
       printf("\nInvalid command.\n");
 }
}


/*-----------------------------------------------------------*/

strchk(s,t)
char s[], *t;
{
  int i;

  i=0;
  while (s[i] == *(t + i))
   if (s[i++] == '\0')
     return(1);
  return(0);
}


/*-----------------------------------------------------------*/

get_port(n)  /*  Pass POINTERS to this function. */
int *n;
{
  printf("       Select port (1-12) > ");
  scanf("%d", n);
  if((*n >= 1) && (*n <= 12))
```

```
    {
      *n = --(*n) * 2;
      return(INRANGE);
    }
    else
    {
      printf("\n       Value is out of range!\n");
      return(ERROR);
    }
}


/*-------------------------------------------------------*/

get_finger(n)  /*  Pass POINTERS to this function. */
int *n;
{
  printf("       Select finger (1-3) > ");
  scanf("%d", n);
  if((*n >= 1) && (*n <= 3))
  {
    *n = --(*n) * 8;
    return(INRANGE);
  }
  else
  {
    printf("\n       Value is out of range!\n");
    return(ERROR);
  }
}


/*-------------------------------------------------------*/

get_bit(n)  /*  Pass POINTERS to this function. */
int *n;
{
  printf("       Select bit (0-11) > ");
  scanf("%d", n);
  if((*n >= 0) && (*n <= 11))
    return(INRANGE);
  else
  {
    printf("\n       Value is out of range!\n");
    return(ERROR);
  }
```

```
}


/*-------------------------------------------------------*/

a_get_val(n)  /*  Pass POINTERS to this function. */
int *n;       /*  Sixteen bit value is gotten.    */
{
  printf("       Enter octal value (0-177777) > ");
  scanf("%o", n);
  if((*n < 0) || (*n > 0177777))
  {
    printf("\n       Value is out of range!\n");
    return(ERROR);
  }
  else
    return(INRANGE);
}


/*-------------------------------------------------------*/

b_get_val(n)  /*  Pass POINTERS to this function. */
int *n;       /*  Twelve bit value is gotten.     */
{
  printf("       Enter octal value (0-7777) > ");
  scanf("%o", n);
  if((*n < 0) || (*n > 07777))
  {
    printf("\n       Value is out of range!\n");
    return(ERROR);
  }
  else
    return(INRANGE);
}


/*-------------------------------------------------------*/

question_mark()
{
  printf("\n\n");
  printf("?                          w <1-12>              r <1-12>      \n");
  printf("wr_csr <1-12> <value>  wr_bfr <1-12> <value>  rd_bfr <1-12> \n");
  printf("cc <1-12>              ccf <1-3>             cch           \n");
  printf("rc <1-12>              rcf <1-3>             rch           \n");
```

```
  printf("rad <1-12>            radf <1-3>            radh          \n");
  printf("b_rc <1-12>           b_rcf <1-3>                         \n");
  printf("b_rad <1-12>          b_radf <1-3>                        \n");
  printf("wda <1-12> <value>                                        \n");
  printf("tg (toggle) <1-12> <0-11>                                 \n");
  printf("describe_coms1        powerupf              powerup       \n");
  printf("describe_coms2        tightf <1-3>          tight         \n");
  printf("help                  loosef <1-3>          loose         \n");
  printf("loop                  exit                               \n\n");
}


/*-----------------------------------------------------------*/

a_describe_coms()
{
  printf("\n\n");
  printf("       Diagnostic commands -- names and descriptions       \n");
  printf("       -------------------------------------------      \n\n");
  printf("     '?' or 'help'  Print command list.                    \n");
  printf("         w <1-12>  Set the control line 'DRV11-J' to write.\n");
  printf("          r <1-12>  Set the control line 'DRV11-J' to read,\n");
  printf(" wr_csr <1-12><value>  Write an octal value to              \n");
  printf("                                    control/status register.\n");
  printf(" wr_bfr <1-12><value>  Write an octal value to an output   \n");
  printf("                                                  buffer.\n");
  printf("  rd_bfr <1-12>  Print the value currently on the data bus.\n");
  printf(" b_rd_bfr <1-12>  Print the value currently on the data bus\n");
  printf("                                    in unsigned binary form.\n");
  printf("         cc <1-12>  Clear a counter.                      \n");
  printf("         ccf <1-3>  Clear a finger's counters.            \n");
  printf("               cch  Clear all counters.                   \n");
  printf("         rc <1-12>  Read a counter.                       \n");
  printf("         rcf <1-3>  Read a finger's counters.             \n");
  printf("               rch  Read all the counters.                \n");
  printf("        rad <1-12>  Read an A/D.                          \n");
  printf("        radf <1-3>  Read a finger's A/Ds.                 \n");
  printf("              radh  Read all the A/Ds.                   \n\n");
}


/*-----------------------------------------------------------*/

b_describe_coms()
{
  printf("\n\n");
```

```
  printf("          Diagnostic commands -- names and descriptions  \n");
  printf("          ---------------------------------------------\n\n");
  printf("          b_rc <1-12>  Read a counter in binary.        \n");
  printf("         b_rcf <1-3>  Read a finger's counters in binary.\n");
  printf("         b_rad <1-12>  Read an A/D in binary.           \n");
  printf("        b_radf <1-3>  Read a finger's A/Ds in binary.   \n");
  printf("   wda <1-12> <value>  Write a value to a D/A.          \n");
  printf("            powerupf  Write out 2048 to one finger's    \n");
  printf("                        DACs for a while.               \n");
  printf("             powerup  Write out 2048 to all DACs for a  \n");
  printf("                        few seconds.                    \n");
  printf("               loose  Write 2048 to all DACs for some   \n");
  printf("                      to protect hand from high currents\n");
  printf("                      caused by corruption of DAC values\n");
  printf("                      when supply voltage is fed to the \n");
  printf("                      linear current amps.              \n");
  printf("       loosef <1-3>  Write out 2048 to one finger's DACs.\n");
  printf("             loose  Write out 2048 to all finger's DACs.\n");
  printf("    describe_coms1  Print a description of some commands.\n");
  printf("    describe_coms2  Print a description of some commands.\n");
  printf("             loop  Execute a command in a loop until ^C.\n");
  printf("             exit  Quit the diagnostic code.          \n\n");
}


/*-----------------------------------------------------------*/

DIR_to_write(port)  /* The DIR bit drives the DRV11 control */
int port;           /* signal 'RDY'.                        */
{
  write_register(0400, &drv_register_ptr->csr0 + port);
}


/*-----------------------------------------------------------*/

DIR_to_read(port)  /*  Tristates the DRV11J output buffer  */
int port;
{
  write_register(0177377, &drv_register_ptr->csr0 + port);
}


/*-----------------------------------------------------------*/

wr_csr(port, val_a)  /*  Write a value onto the DRV11-J data bus. */
```

```
int port, val_a;
{
  unsigned short val_b;

  val_b = (unsigned short)val_a;
  write_register(val_b, &drv_register_ptr->csr0 + port);
}


/*-------------------------------------------------------*/

short rd_bfr(port)  /*  Read the value on the DRV11-J data bus. */
int port;
{
  return(read_register(&drv_register_ptr->buf0 + port));
}


/*-------------------------------------------------------*/

i_rd_bfr(port)  /*  Read the value on the DRV11-J data bus  */
int port;       /*  interactively.                         */
{
  printf("%d  ",(read_register(&drv_register_ptr->buf0 + port)));
}


/*-------------------------------------------------------*/

b_rd_bfr(port)  /*  Read the value on the DRV11-J data bus  */
int port;       /*  interactively.                         */
{
  sixteen_binary(read_register(&drv_register_ptr->buf0 + port));
}


/*-------------------------------------------------------*/

wr_bfr(port, val_a)  /* Write a value onto the DRV11-J data bus.*/
int port, val_a;
{
  unsigned short val_b;

  val_b = (unsigned short)val_a;
  write_register(val_b, &drv_register_ptr->buf0 + port);
}
```

```
/*-----------------------------------------------------------*/

cc(port)  /*  Clear one counter. */
int port;
{
  DIR_to_write(port);    /* Setting appropriate device select bits   */
  wr_bfr(port, 030000);  /* automatically resets associated counter.*/
}


/*-----------------------------------------------------------*/

ccf(port)  /*  Clear one finger's counters. */
int port;
{
int i;

  for (i=0; i<4; ++i, port+=2)
  {
    DIR_to_write(port);
    wr_bfr(port, 030000);
  }
}


/*-----------------------------------------------------------*/

cch()  /*  Clear all the counters. */
{
int port;

  for (port=0; port<24; port+=2)
  {
    DIR_to_write(port);
    wr_bfr(port, 030000);
  }
}


/*-----------------------------------------------------------*/

/* The following function expects a port value of 0, 2,...or 10.*/

rc(port)  /*  Read one counter  */
```

```
int port;
{
  DIR_to_write(port);
  wr_bfr(port, 020000);              /*  Set device select bits. */
  DIR_to_read(port);
  printf("%d  ", (int) rd_bfr(port));
}


/*-------------------------------------------------------------*/

rc1(port)  /*  A looping 'rc'  */
int port;
{
  int i, b, orig;
  char loop1[30], *loop2 = "yes";

  printf("    Do you want to loop?  (yes or no) > ");
  scanf("%s", loop1);
  if(strchk(loop1, loop2))
  {
    orig = port;
    for(b=0; b<1900; ++b)
    {
      rc(port);
      putchar('\r');
    }
  }
  else
    rc(port);
}


/*-------------------------------------------------------------*/

/* The function below requires 'port' to contain 0, 8 or 16.  */
/* These are the offsets for the first port of each I/O board.*/

rcf(port)  /*  Read the counters of one finger. */
int port;
{
  int i, b, orig;
  char loop1[30], *loop2 = "yes";

  printf("    Do you want to loop?  (yes or no) > ");
  scanf("%s", loop1);
```

```
  if(strchk(loop1, loop2))
  {
    orig = port;
    for(b=0; b<800; ++b)
    {
      for(i=0, port = orig; i<4; ++i, port+=2)
        rc(port);
      putchar('\r');
    }
  }
  else
    for(i=0; i<4; ++i, port+=2)
      rc(port);
}


/*-------------------------------------------------------*/

rch()  /*  Read the counters for the entire hand. */
{
  int port;

  for(port=0; port<24; port+=2)
    rc(port);
}


/*-------------------------------------------------------*/

/*  Expects a port value of 0, 2,...or 10. */

b_rc(port)  /*  Same as 'rc', but with binary output. */
int port;
{
  int i, b;

  DIR_to_write(port);
  wr_bfr(port, 020000);          /* Set device select bits.*/
  DIR_to_read(port);
  sixteen_binary(rd_bfr(port));
}


/*-------------------------------------------------------*/

b_rcl(port)  /*  Same as 'rcl', but binary output. */
```

```
int port;
{
  int i, b, orig;
  char loop1[30], *loop2 = "yes";

  printf("    Do you want to loop?  (yes or no) > ");
  scanf("%s", loop1);
  if(strchk(loop1, loop2))
  {
    orig = port;
    for(b=0; b<1800; ++b)
     {
       b_rc(port);
       putchar('\r');
     }
  }
  else
      b_rc(port);
}


/*-------------------------------------------------------------*/

/*  The function below requires 'port' to contain 0, 8 or 16.  */
/*  These are the offsets for the first port of each I/O board.*/

b_rcf(port)  /*  Same as 'finger_rcs', but binary output. */
int port;
{
  int i, b, orig;
  char loop1[30], *loop2 = "yes";

  printf("    Do you want to loop?  (yes or no) > ");
  scanf("%s", loop1);
  if(strchk(loop1, loop2))
  {
    orig = port;
    for(b=0; b<800; ++b)
     {
       for(i=0, port = orig; i<4; ++i, port+=2)
         b_rc(port);                  /* 'b_rc' gets the desired */
       putchar('\r');                 /* counter values into the */
     }                                /* global array.           */
  }
  else
    for(i=0; i<4; ++i, port+=2)  /* 'b_rc' gets the desired counter */
```

```
      b_rc(port);                /* values into the global array.  */
}


/*-----------------------------------------------------------*/

/* The following function expects a port value of 0, 2,...or 10.*/

rad(port)  /*  Read one A/D. */
int port;
{
  DIR_to_write(port);
  wr_bfr(port, 0);                    /*  Set device select bits. */
  DIR_to_read(port);
  while((rd_bfr(port)&(0100000)) == 0)  /*  Wait for conversion.  */
     ;                                  /*  Null body.            */
  printf("%d  ", (rd_bfr(port)&(07777)));
}


/*-----------------------------------------------------------*/

/*  The function below requires 'port' to contain 0, 8 or 16.  */
/*  These are the offsets for the first port of each I/O board.*/

radf(port)  /*  Read the A/Ds for one finger. */
int port;
{
  int i, b, orig;
  char loop1[30], *loop2 = "yes";

  printf("    Do you want to loop?  (yes or no) > ");
  scanf("%s", loop1);
  if(strchk(loop1, loop2))
  {
    orig = port;
    for(b=0; b<800; ++b)
    {
      for(i=0, port = orig; i<4; ++i, port+=2)
        rad(port);
      putchar('\r');
    }
  }
  else
    for(i=0; i<4; ++i, port+=2)
      rad(port);
```

```
}


/*-----------------------------------------------------------*/

radh()  /*  Read the A/Ds for the whole hand. */
{
  int port;

  for(port=0; port<24; port+=2)
    rad(port);
}


/*-----------------------------------------------------------*/

/* The following function expects a port value of 0, 2,...or 10.*/

b_rad(port)  /*  Same as 'rad', but output in binary. */
int port;
{
  int i;

  DIR_to_write(port);
  wr_bfr(port, 0);                      /* Set device select bits.*/
  DIR_to_read(port);
  while((rd_bfr(port)&(0100000)) == 0)  /* Wait for conversion.   */
     ;                                  /* Null body.             */
  twelve_binary(rd_bfr(port)&(07777));  /* Read value and save it.*/
}


/*-----------------------------------------------------------*/

/*  The function below requires 'port' to contain 0, 8 or 16.  */
/*  These are the offsets for the first port of each I/O board.*/

b_radf(port)  /*  Same as finger_rads, but output in binary. */
int port;
{
  int i, b, orig;
  char loop1[30], *loop2 = "yes";

  printf("    Do you want to loop?  (yes or no) > ");
  scanf("%s", loop1);
```

```
   if(strchk(loop1, loop2))
   {
     orig = port;
     for(b=0; b<800; ++b)
      {
        for(i=0, port = orig; i<4; ++i, port+=2)
          b_rad(port);                /* 'b_rad' gets the desired */
        putchar('\r');                /* A/D values.              */
      }
    }
    else
      for(i=0, port = orig; i<4; ++i, port+=2)
        b_rad(port);                  /* 'b_rad' gets the desired */
}                                     /* A/D values.              */


/*-------------------------------------------------------*/

wda(port,val_a)  /*  Write a new value out to one DAC. */
int port, val_a;
{
  DIR_to_write(port);
  wr_bfr(port, (val_a & 07777) | 010000);
}


/*-------------------------------------------------------*/

/*  The function below requires 'port' to contain 0, 8 or 16.  */
/*  These are the offsets for the first port of each I/O board.*/

tension(port,tension) /* 'tension' is written to all the D/As */
int port;                 /* for the selected finger.             */
unsigned short tension;
{
  int i;

  for(i=0; i<4; ++i, port+=2)
    wda(port, tension);
}


/*-------------------------------------------------------*/

loose() /* Set all DACs to 2048 (unsigned decimal).This corresponds */
{       /*  to all loose tendons.                                    */
```

```
  int port, num;

  num = (int) 04000;
  for(port=0; port<24; port+=2)
    wda(port,num);
}


/*-------------------------------------------------------------*/

loosef(port) /* Set one finger's DACs to 2048 (unsigned decimal).*/
int port;
{
  int i, num;

  num = (int) 04000;
  for(i=0 ; i<4; ++i, port+=2)
    wda(port,num);
}


/*-------------------------------------------------------------*/
tightf(port)  /*  Set all DACs to 4095 -- all tight tendons. */
int port;
{
  int i, num;

  num = (int) 07777;
  for(i=0; i<4; ++i, port+=2)
    wda(port,num);      /* 'wda' gets the desired counter values */
}                       /* into the global array.               */


/*-------------------------------------------------------------*/
tight()  /*  Set all DACs to 4095 -- all tight tendons. */
{
  int port, num;

  num = (int) 07777;
  for(port=0; port<24; port+=2)
    wda(port,num);      /* 'wda' gets the desired counter values */
}                       /* into the global array.               */


/*-------------------------------------------------------------*/
```

```
      /*  This should always be executed inside 'loop'. */

toggle(port,bit)  /*  Flip one D/A bit between 0 and 1. */
int port, bit;          /*  Bit is an integer from 0 to 11. */
{
  int i, shifted, pause;
  shifted = 1 << bit;

  for(i=0; i<30000; ++i)
  {
    wda(port,shifted);
    for(pause=0; pause < 7; ++pause)
        ;
    wda(port,0);
  }
}


/*-------------------------------------------------------------*/

powerupf(finger)/* This function continuously sets the A/D values to a */
int finger;     /* low value for some moments during powerup, since a  */
{               /* glitch in the system tends to cause these values to */
  int j;        /* go high (thus putting the hand under great stress).*/

  for (j=0; j<18000; j++)
    loosef(finger);
}


/*-------------------------------------------------------------*/

powerup()       /* This function continuously sets the A/D values to a */
{               /* low value for some moments during powerup, since a  */
  int j, port;  /* glitch in the system tends to cause these values to */
                /* go high (thus putting the hand under great stress).*/

  for (j=0; j<8000; j++)
    loose();
}


/*-------------------------------------------------------------*/

/*  The following function prints unsigned binary output for  */
/*  a 16 bit value.                                           */
```

```
sixteen_binary(input)
int input;
{
  int mask, i, outa, outb, leading_z, a_zeros_index, b_zeros_index;
  char *zeros[] = {"\0", "0", "00", "000", "0000", "00000", "000000",
                   "0000000", "0000000"};

  leading_z = TRUE;
  a_zeros_index = b_zeros_index = outa = outb = 0;
  for(i=15, mask = 0100000; i>7; --i, mask = mask >> 1)
    if((input & mask) > 0)
    {
      leading_z = FALSE;
      outa += power(i-8);
    }
    else if(leading_z == TRUE)
      ++a_zeros_index;
  leading_z = TRUE;
  for( ; i >= 0; --i, mask = mask >> 1)
    if((input & mask) > 0)
    {
      leading_z = FALSE;
      outb += power(i);
    }
    else if(leading_z == TRUE)
      ++b_zeros_index;
  printf("%s%d%s%d ", zeros[a_zeros_index],
          outa, zeros[b_zeros_index], outb);
}


/*-----------------------------------------------------------*/

/*  The following function prints unsigned binary output for  */
/*  a 12 bit value.                                           */

twelve_binary(input)
int input;
{
  int mask, i, outa, outb, leading_z, a_zeros_index, b_zeros_index;
  char *zeros[] = {"\0", "0", "00", "000", "0000", "00000", "00000"};

  leading_z = TRUE;
  a_zeros_index = b_zeros_index = outa = outb = 0;
  for(i=11, mask = 04000; i>5; --i, mask = mask >> 1)
```

```
    if((input & mask) > 0)
    {
      leading_z = FALSE;
      outa += power(i-6);
    }
    else if(leading_z == TRUE)
      ++a_zeros_index;
  leading_z = TRUE;
  for( ; i >= 0; --i, mask = mask >> 1)
    if((input & mask) > 0)
    {
      leading_z = FALSE;
      outb += power(i);
    }
    else if(leading_z == TRUE)
      ++b_zeros_index;
  printf("%s%d%s%d ", zeros[a_zeros_index],
              outa, zeros[b_zeros_index], outb);
}


/*-------------------------------------------------------------*/

power(n)  /*  Raises 10 to the power of n. */
int n;
{
  int out;

  for(out=1; n>0; --n)
    out = out * 10;
  return(out);
}


/*-------------------------------------------------------------*/

void drv_service_routine(int_registers, int_region)
struct drv_register_def   *int_registers;
struct drv_region_def  *int_region;
{
}
```

Bibliography

[1] Kenneth Salisbury *Kinematic and Force Analysis of Articulated Hands*, Ph.D. Dissertation, May 1982, Stanford University.

[2] S. T. Venkataraman *Task Dependent Dexterous Hand Control*, Ph.D. Dissertation, May 1988, University of Massachusetts at Amherst.

[3] Judy A. Franklin *Computer Interfaces and Operating Instructions for a Prototype Cartesian Robot*, July 1983, Department of Computer and Information Science Technical Report #83-10, University of Massachusetts at Amherst.

[4] Uri Herel *Improvement in the Strain Gages System of UMASS JPL/Stanford Hand*, June 1988, available on request from the COINS Robotics Laboratory, University of Massachusetts at Amherst.