

# DIO1A

## Digital Input/Output Module

---

The DIO1A Digital Input Output Module provides 32 digital input or output channels at TTL levels. These channels are grouped into four 8-bit ports which can be individually programmed for input or output by software commands. The inputs are read directly by the host computer, and the outputs are latched. A power on reset circuit automatically configures all ports as inputs when the DIO1A module is powered up.

I/O connections are made to on-card quick disconnect screw terminal blocks, which accept 16-24 gauge wire stripped 3/16". In addition to the screw terminals, a mass termination connector for ribbon cable is provided on the forward edge of the module.

### CAUTION

**Always turn off the data acquisition system power supply before installing or removing modules. To avoid the possibility of EMI radiation, never operate the system with the top cover open or removed.**

The DIO1A module can be installed in any available slot in any Keithley/DAC Data Acquisition System. However, for best performance any analog modules should be plugged into the lowest-numbered available slots, leaving the higher slots for digital modules. To install the module in a Model 500 or Model 556, remove the top cover and insert the module in any available slot with the component side of the module facing the power supply. To install the module in a Model 570 or a Model 575, open the clamshell cover and insert the module into the option slot with the component side of the module facing up. See your data acquisition system manual for details on opening the system and installing modules.

The DIO1A module can be identified by a Model 575, 500A, 500P, or 556 mainframe through the use of a Self-ID resistor which is mounted on the module. The value of this resistor is measured by the mainframe, and an ID code is generated from the value. The DIO1A module uses the same ID code 52 (4530 ohms) as the older DIO1

module, and is functionally compatible with the older module. The older DIO1 module has hardware switches for port configuration, while this function is performed through software for the DIO1A.

## User Connections

User connections can be made to either the quick disconnect terminal blocks at the top of the module, or to the mass termination connector located at one end of the board (through the supplied ribbon cable).

Table 1. User-configured Components

Component	Designation	Function
Screw Terminals	J1-J14	Input/Output Connections
I/O Connector	J6	Quick Disconnect Mass Termination for Ribbon Cable

Screw terminals for making input/output connections are located on J1, J2, J3, and J4 across the top edge of the module board. The channel numbers (0 to 31) are marked across the top edge of the board and are also shown in Figure 1.

Two connections each for the module ground (GND) and +5 volts (+5) are also included on the top edge of the board. The terminals are intended to be used with 16 to 24 gauge wire stripped 3/16", and the terminal blocks can be pulled off of the strip to facilitate connections. A typical input/output connection for the screw terminals is shown in Figure 2.

In situations requiring rapid connection and disconnection of signal lines, J6 can be used as an alternate means of signal connection. Figure 3 shows the pin diagram for J6, which is designed for use with a matching optional rib-

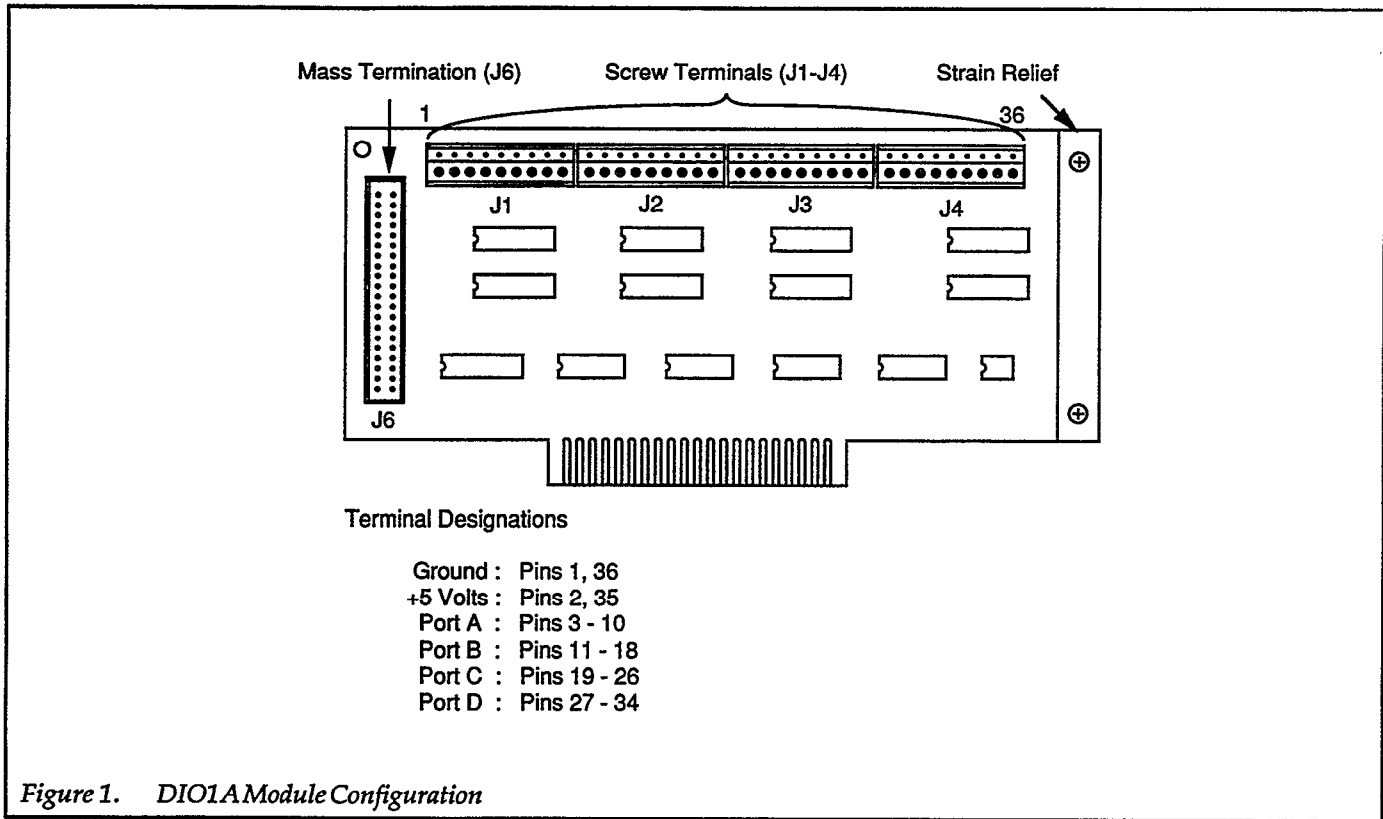


Figure 1. DIO1A Module Configuration

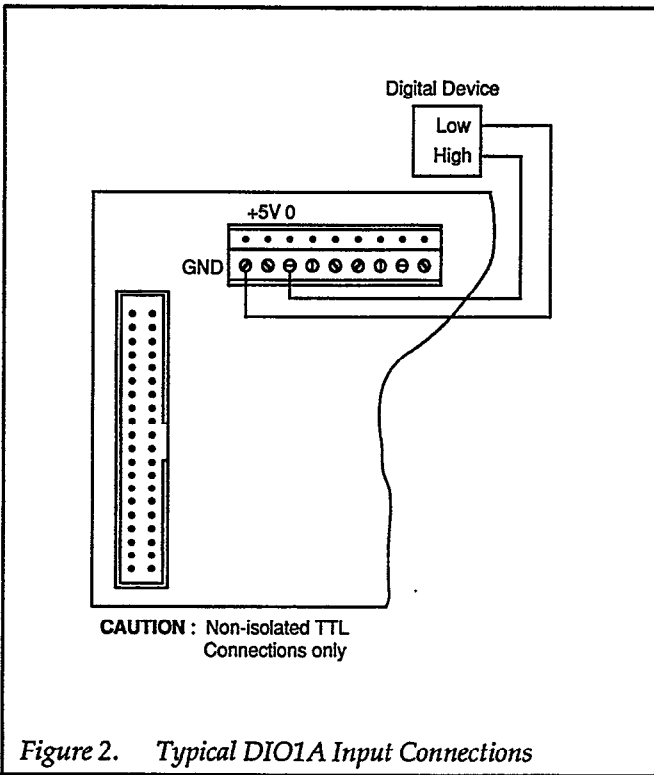


Figure 2. Typical DIO1A Input Connections

bon cable available from Keithley. When connecting signals in this manner, the screw terminals should not be used.

If EMI radiation becomes a problem, use shielded cables for input and output connections. Connect the shield of the cable to the baseboard ground ONLY (or one of the GND terminals of the screw terminal connectors), and leave the other end of the shield unconnected. Do not use the shield of the cable as a signal carrying connection.

**CAUTION**

DIO1A inputs and outputs are non-isolated, meaning that the circuit ground is connected to power ground. Any circuits connected to the module must also be referenced to power ground and must not be floating, or the module may be damaged.

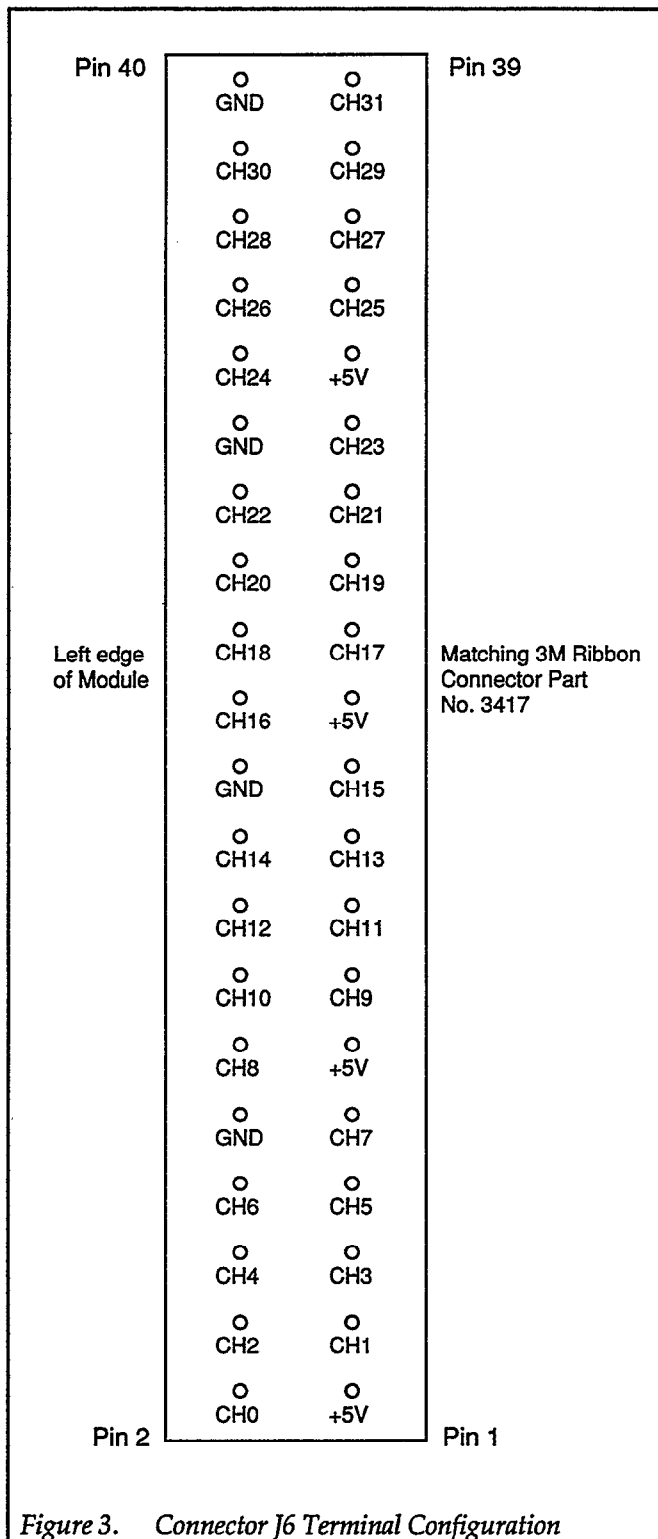


Figure 3. Connector J6 Terminal Configuration

### Input/Output Programming

The 32 digital I/O channels of the DIO1A are grouped into four ports of 8 channels each. Port A consists of channels 0 to 7, Port B of channels 8 to 15, Port C of channels 16 to 23, and Port D of channels 24 to 31. The function of each port (and thus, the 8 channels associated with that port,) can be programmed individually for input or output through the use of software commands discussed below. The data written to or read from any port can be handled as a byte of information, in which eight channels are accessed at one time.

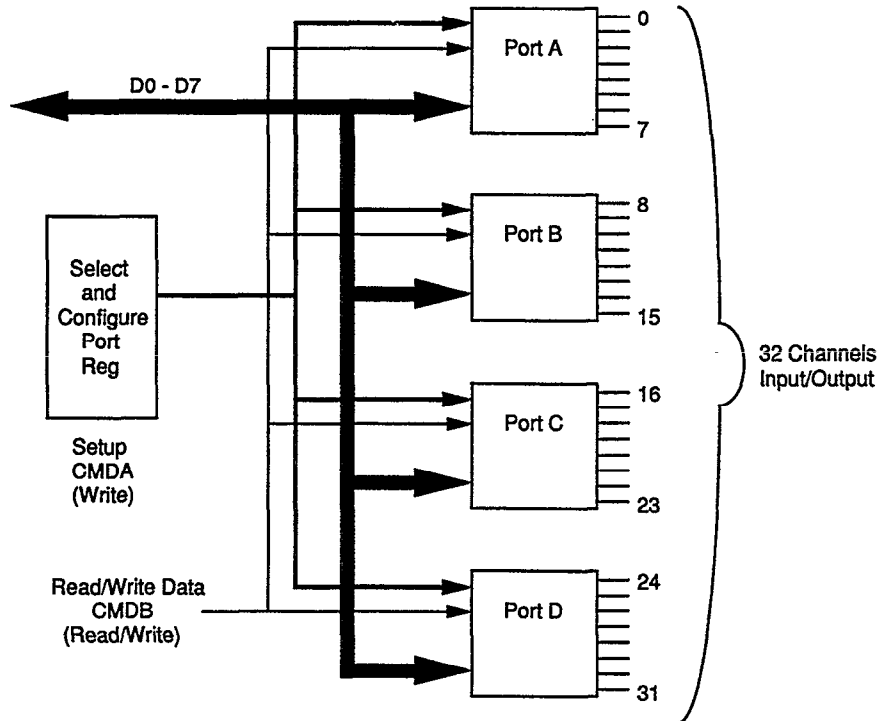
### TTL Levels and Loading Considerations

When configured for input, each channel of the DIO1A represents one LS TTL load to the external circuitry (20µA high, 0.4mA low). Logic low should be between 0 volts and 0.5 volts, and logic high should be between 2 volts and 5 volts (standard TTL levels).

When configured as outputs, each channel is capable of driving a maximum of 60 LS TTL loads or 15 standard TTL loads (maximum output is 1.2 mA high and 24 mA low). Logic low will be in the range of 0 volts to 0.5 volts, and logic high will be between 2.7 volts and 5 volts.

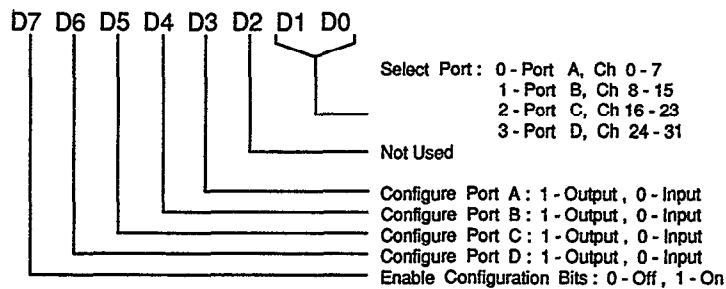
### Commands

The DIO1A uses the slot dependent CMDA and CMDB lines for setup and read/write operations. To select a port for a read or write operation, or to configure the ports for input or output, write to CMDA of the appropriate slot. In order to write or read data, write to or read from CMDB for the appropriate slot. See Table 2 for a list of the slots and memory locations corresponding to the CMDA and CMDB lines. Figure 4 includes details on the DIO1A register usage. Memory locations indicated for these commands assume that the interface card in the host PC is set for a base address of CFF80. Any other setting of the base address will require the appropriate offset of the supplied command locations.



**CMDA (write) Address - Select port, configure port**

Must be issued in order to read or write data to a specified port.



**CMDB (read) Address - Digital Input**

Read data from specified port provided port is configured as an input port.

**CMDB (write) Address - Digital Output**

Write data to specified port provided port is configured as an output port.

Figure 4. DIO1A Block Diagram and Command Register

**Table 2. Locations for Slot Dependent Commands**

Slot	Select/Configure Port (CMDA)	Digital I/O (CMDB)
1	XXX80	XXX81
2	XXX82	XXX83
3	XXX84	XXX85
4	XXX86	XXX87
5	XXX88	XXX89
6	XXX8A	XXX8B
7	XXX8C	XXX8D
8	XXX8E	XXX8F
9	XXX90	XXX91
10	XXX92	XXX93

XXX=1st 3 digits of IBIN address, e.g. "CFF"

### SELECT/CONFIGURE PORT operations

Location: Slot dependent CMDA write

The CMDA write command is used to configure the ports for input or output, and to select the port that data is to be written to or read from. Refer to Figure 4 for the following discussion of the CMDA write data byte.

Bits D3 through D7 are used to configure the ports for input or output. Bit D7 must be set to a logic 1 to enable bits D3 through D6 to set their appropriate port to an input or an output port. When bit D7 is set to a logic 0, bits D3 through D6 are ignored. Bits D3 through D6 are used for PORT A through PORT D, respectively. Setting the appropriate bit to a logic 1 sets that port and all associated channels, to a digital output. Setting the appropriate bit to a logic 0 sets that port, and all associated channels, to a digital input.

#### NOTE

Upon power up, all ports are set to be digital inputs.

Bits D0 and D1 are used to select a particular port for input or output. A value from 0 to 3 will select port A through port D respectively. Port selection will always occur with a CMDA write to the DIO1A module, regardless of the logic value of bit D7.

#### NOTE

Upon power up, a random port will be selected; therefore, port selection should be accomplished prior to read or write operation with CMDB.

#### NOTE

In some applications, you may find it useful to read back the levels previously programmed to a DIO1A port configured for output. The value read back will be the last byte value successfully programmed to the port. This feature can be implemented using BASIC's PEEK command, or the equivalent memory read function of other languages. This WAS NOT a capability of the older DIO1 module, and should not be attempted using the older module.

### DIGITAL INPUT/OUTPUT operations

Location: Slot-dependent CMDB read or write

A CMDB read is used to access data from a selected port when that port is configured as an input port. A CMDB write is used to write data to the selected port when that port is configured as an output port. Note that each of the four ports can be individually programmed as either an input or output port, as discussed above.

When a port is configured as an input port, the CMDB command location is read to access the status of the channels associated with the port. Reading this location will return an 8 bit binary number, with the D0 thru D7 lines representing the input lines for the selected port as indicated in Table 3. Since the inputs are not latched, the data will reflect the condition of the input channels at the instant that the read operation is performed. Note that the channels being read will depend on the port selected with the CMDA write command discussed above.

If you need to know the status of a particular channel, a logical AND instruction can be used to "mask off" the unnecessary bit positions. The mask should contain zeros in the irrelevant bit positions and logic ones in the bit positions representing the channels in question. The result of ANDing the mask with the input data will then be an 8 bit binary number with each bit in an irrelevant position equal to zero. The relevant channels will contain a 1 or a 0 depending on the status of the corresponding bit in the masked byte (a 1 if the object bit is high, a 0 if the object bit is low).

When reading a DIO1A port with the BASIC PEEK command, the returned value will be a decimal number between 0 and 255, depending on the status of the input channels. The status of a particular channel can be determined by ANDing the returned value with the decimal weighting for that particular channel. If the ANDed result is non-zero, the channel is high (logic 1).

**EXAMPLE:** The status of channel 6 on port A is to be determined:

```
100 POKE CMDA,0
110 STATUS6 = PEEK(CMDB) AND 64
```

If the value returned in variable STATUS6 is non-zero, then channel 6 is high.

If a DIO1A port is configured as output, the value written to CMDDB should be the equivalent of an 8 bit binary number, with the status of the 8 bits (1 or 0) representing the status of the 8 channels of the selected port (high or low). The channels affected will depend on the selected port, as outlined in Table 3. Data written to a port will be latched into the outputs and will remain unchanged until the port is written to again, or the port is configured as an input port. From BASIC, the value written to the port using the POKE command will be a decimal value in the range from 0 to 255.

To change the status of one output channel while leaving the others unchanged, you must know the status of all of the channels. To accomplish this, a variable in the controlling program could be assigned to the current status of the port in question (and updated every time the port is written to), or the output on the port could be read di-

rectly from the port using the readback feature (and stored in a variable). When the current status of all the channels on the particular port is known, the value of the variable can be changed to correspond to the desired status of the selected port, and this new value written to the CMDDB location.

In assembly language, this modification can be performed by using logical AND and logical OR instructions. When turning a single channel on (high) the variable representing the current port status should be ORed with a number equivalent to an 8 bit binary value with zeros in the bit positions to remain unchanged and ones in the bit positions to be turned on. When turning a channel off, the port status variable should be ANDed with an 8 bit binary value with ones in the bit positions to remain unchanged and zeros in the positions to be turned off.

From BASIC, the decimal weight of the bit to be turned on or off should be added to (off to on) or subtracted from (on to off) the decimal number that represents the current port status. For example, to turn channel 4 off, it would be necessary to subtract 8 from the decimal value representing the port status. To turn on channel 8, it would be necessary to add 128 to the status.

It should be noted that the DIO1A module is command compatible with the older DIO1 module, with the added programmability of the port configuration versus the switch selectability of the older module. Programs written for the older module can be modified for use with the new module by simply issuing a CMDA WRITE to configure the ports before any input or output is accomplished.

**Table 3. Digital I/O Channels and Bit Values**

Bit Position	D7	D6	D5	D4	D3	D2	D1	D0
Weighting	128	64	32	16	8	4	2	1
Port A Chan Nos.	7	6	5	4	3	2	1	0
Port B Chan Nos.	15	14	13	12	11	10	9	8
Port C Chan Nos.	23	22	21	20	19	18	17	16
Port D Chan Nos.	31	30	29	28	27	26	25	24

## Theory of Operation

A schematic drawing of the DIO1A module is located on drawing number 501-206.

The DIO1A circuitry may be divided into six sections: data bus buffering circuitry, port selection circuitry, port configuration circuitry, channel input buffers, channel output latches, and power up reset circuitry.

Data from the baseboard bus circuitry is applied to the octal bus transceiver U13 (74LS245). Bus transmission direction of U13 is controlled by the baseboard Read/Write line, while the CMDB line is used to enable the device when reading or writing data to one of the 4 ports on the module using the DIGITAL I/O command.

The two least significant bits of the data bus (D0 and D1) are also applied to the port selection circuitry made up of a dual D type flip flop U2 (74LS75) and a 3 to 8 line decoder U4 (74LS138). When a CMDA write is issued in the select port mode, the values on the D0, D1, and Read/Write lines are latched into U4. When the CMDB line next goes low, U4 decodes the input from U2 into an enabling signal to go to only 1 of the 4 ports (either the input buffer or the output latch, depending upon the state of the Read/Write line). For example, if 11 (binary) were present at D0 and D1, and the Read/Write line was active (low), executing a CMDA write would latch the data into U4; then the Y7 output of U4 would go low with the next CMDB write, latching the data present on the bus at that time to the output of the port.

Input/Output selection is performed by triple 3 input nor gate U1 (74LS27) and quad latch U3 (74LS175). When a CMDA write is issued (bit D7 high) the data on lines D3 to D6 is latched into the inverting outputs of U3. A low to the output control of the output latches U6, U8, U10, and U12 (74LS374) allows the Read/Write line to clock the data from the bus to the port. When D3 to D6 are low, the output latches tri-state and the select port registers of U4 enable the input buffer for the selected port.

The Self-ID resistor (R40) connected between pins 4 and 41 of the card edge connector allow the mainframe to identify the module by reading the value of the resistor with an A/D conversion circuit and generating a code based on the value of the resistor. The ID resistor has a value of 4.53 kilohms, with a tolerance of 0.1%.

The power up reset circuitry is used to ensure that all outputs power up in a high impedance state by holding the clear line of U3 low until power up is complete, forcing all inverting outputs of the quad latch high.

## Troubleshooting Information

Troubleshoot the DIO1A by using the procedure outlined in Table 4. This process is fairly straightforward, as signal tracing is simply a matter of checking for the correct digital logic level throughout the module.

If a defective part is found, a replacement can be ordered from Keithley Data Acquisition and Control. Refer to the Service Section of the hardware manual for ordering details. Part numbers for the module are located on the component layout drawing (501-200).

**Table 4. DIO1A Troubleshooting Information**

Step	Component	Required Condition	Remarks
1	DIO1A module	Install in slot 10	turn power on
2	channel 0-31	apply logic high (+5V)	apply to inputs
3	computer	program all ports as inputs and select port A	use CMDA write DATA 11111X00
4	computer	read back 255 from port A	use CMDB read
5	ports B-D	repeat steps 4 and 5	to test ports B-D
6	channel 0-31	apply logic low (0V)	apply to inputs
7	computer	select port A	use CMDA write
8	computer	read back 0 from port A	use CMDB read
9	ports B-D	repeat steps 8 and 9	to test ports B-D
10	computer	program all ports as outputs and select port A	use CMDA write
11	computer	write 255 to port	DATA 10000X00
12	channel 0-7	all outputs high	use CMDB write
13	ports B-D	repeat steps 13 and 14	use logic probe/DMM
14	computer	select port A	to test ports B-D
15	computer	write 0 to port	use CMDA write
16	channel 0-7	all outputs low	use CMDB write
17	ports B-D	repeat steps 17 and 18	use logic probe/DMM to test ports B-D

Note: Slot 10 locations:      CMDA XXX92  
   CMDB XXX93

## DIO1A Specifications

**Channels:** 32, software programmable in groups of 8 for input or output

**Signal connection:** Quick disconnect screw terminal blocks, 40 pin receptacle for ribbon cable assembly

### Input characteristics:

TTL-compatible, high-true  
High-level Input Current: 20 $\mu$ A  
Low-level Input Current: -0.4mA

### Output characteristics:

TTL-compatible, high true  
Drive capacity: 60 LS-TTL loads or 15 standard TTL loads; 24mA sink at 0.5V  
Self-ID: resistor value - 4.53k $\Omega$ ,  $\pm$ 1%. ID code 52

**General:** All inputs and outputs referenced to system ground

All inputs and outputs power up in high impedance state as inputs for safety

**Accessories:** Ribbon cable assembly, 6 foot, 40 conductor; one end mates with 40 pin receptacle on DIO1A, other end is unterminated. Keithley part number 500-CA3.

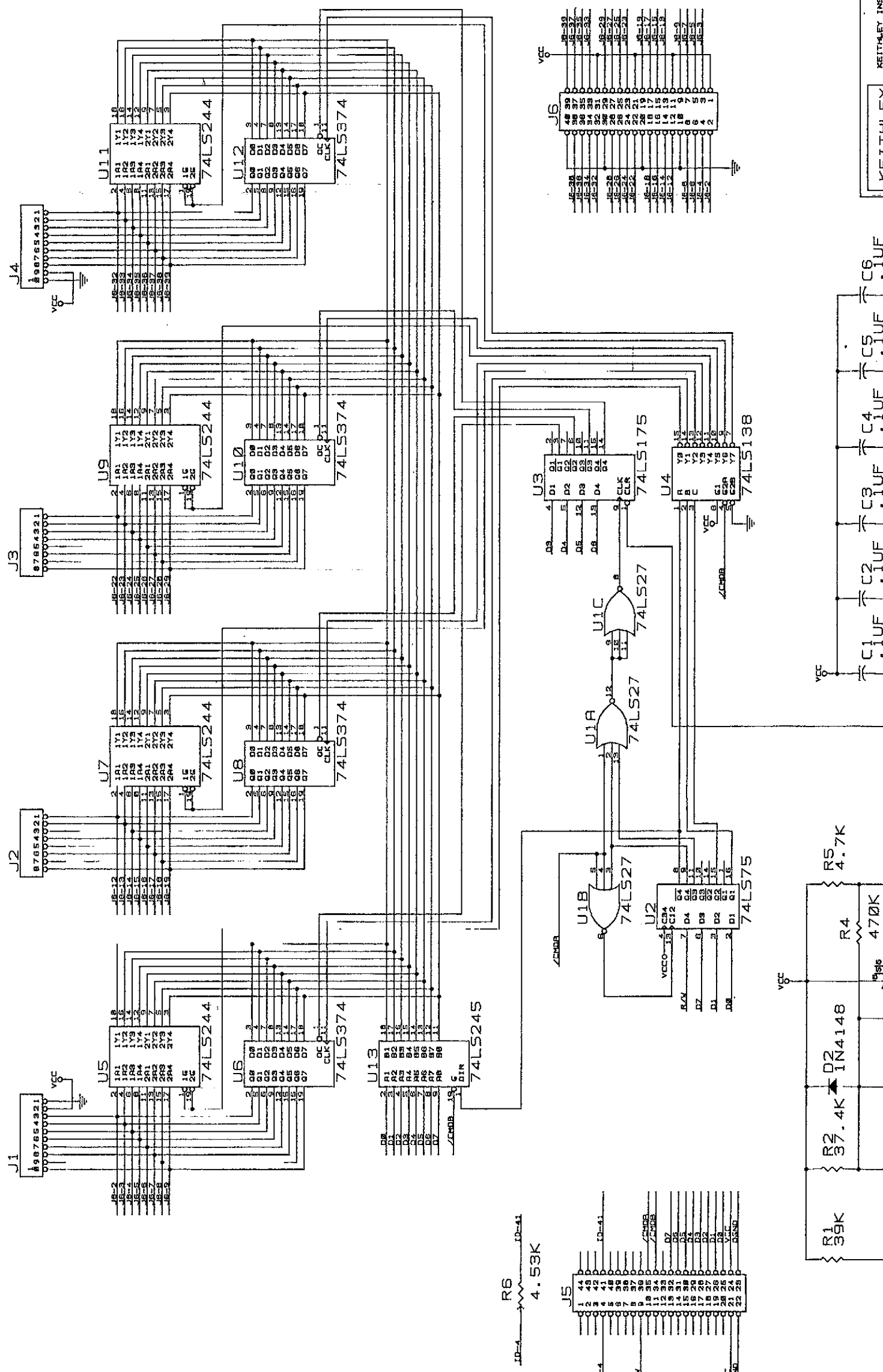
Cable ribbon plug, only - 3M part number 3417.

## Parts List

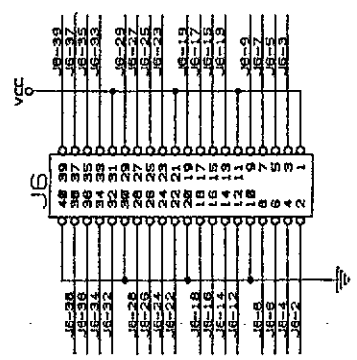
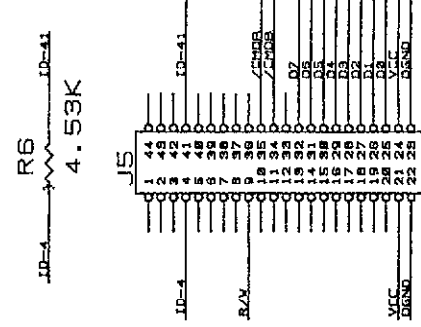
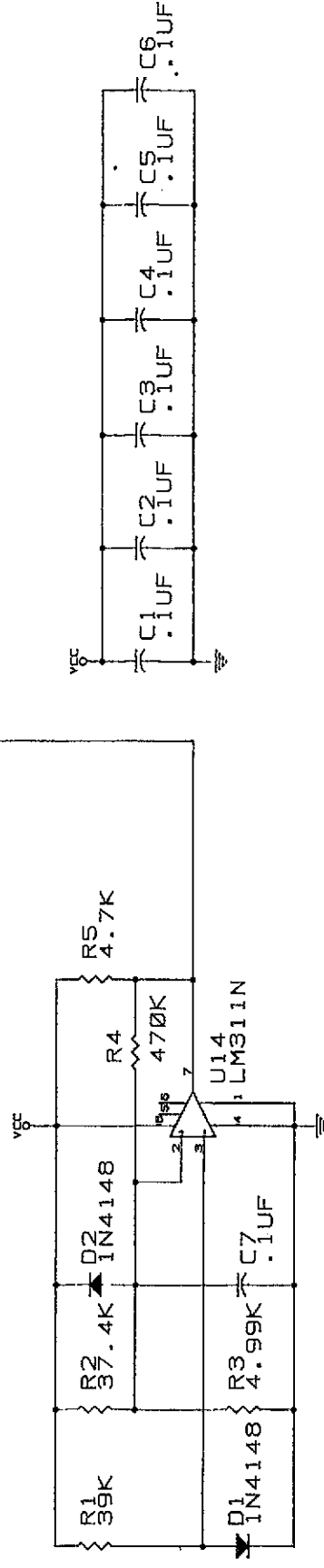
The following components are used in the DIO1A module. Spares can be ordered from the repair department of Keithley Instruments, Inc., 28775 Aurora Road, Cleveland, OH 44139. Please supply the part number, description, and quantity when ordering.



Part No.	Quantity	Description	Designation
C-365-.1	7	Cap, .1 $\mu$ F, 20%, 50V Ceramic	C1-C7
CS-368-40	1	Conn, 40-pin ribbon	J6
CS-521-2	2	Conn, Strip, 10-pin	J1, J4
CS-521-4	2	Conn, Strip, 8-pin	J2, J3
IC-157	1	IC, Quad D Flip-flop, 74LS175	U3
IC-173	1	IC, Comparator, LM311N	U14
IC-182	1	IC, Decoder/DEMUX, 74LS138	U4
IC-230	4	IC, Octal Buffer/Line Driver, 74LS244	U5, U7, U9, U11
IC-233	1	IC, Triple 3-input NOR, 74LS27	U1
IC-242	4	IC, Octal D-type Latch, 74LS374	U6, U8, U10, U12
IC-356	1	IC, Octal Bus Transceiver, 74LS245	U13
IC-366	1	IC, 4-bit Bistable Latch, 74LS75	U2
R-176-4.53K	1	Res, 4.53K, .1%, 1/8W, Metal Film	R2
R-76-39K	1	Res, 39K, 5%, 1/4W, Comp. or Film	R1
R-76-4.7K	1	Res, 4.7K, 5%, 1/4W, Comp. or Film	R5
R-76-470K	1	Res, 470K, 5%, 1/4W, Comp. or Film	R4
R-76-37.4K	1	Res, 37.4K, 1%, 1/8W, Metal Film	R2
R-76-4.99K	1	Res, 4.99K, 1%, 1/8W, Metal Film	R3
RF-28	2	Diode, Silicon, 1N4148 (DO-35)	D1, D2
6-32x1/2PPH	2	Screw	
500-323	1	Cable Clamp Assembly	
500-321	1	Cable Clamp	
500-322	1	Strip Rubber	
501-916-01A	1	DIO1A Manual	

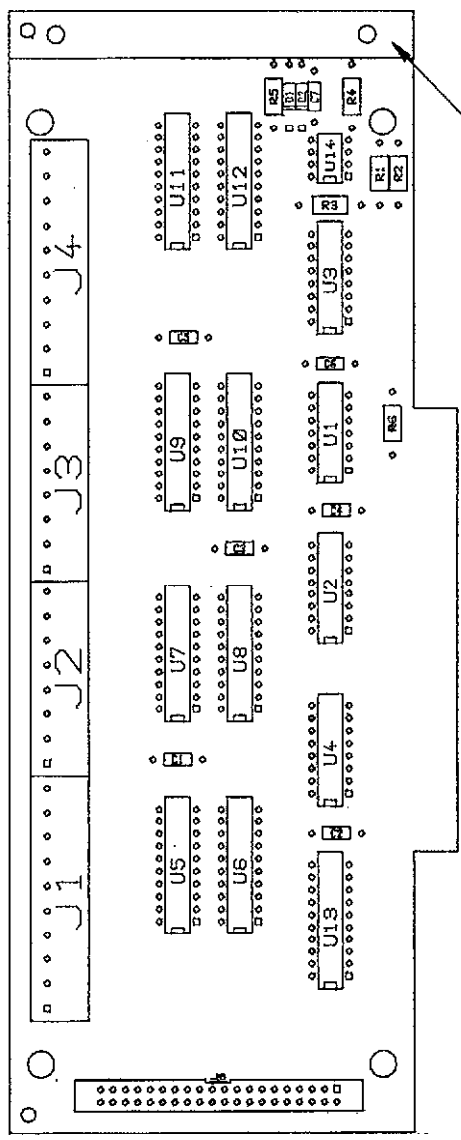


KEITHLEY		KEITHLEY INSTRUMENTS CLEVELAND, OHIO 44150	
TITLE SCHEMATIC, DIGITAL INPUT/OUTPUT MODULE			
SIZE	MODEL	NO.	REV
C	D101A	501-206	1.6638 B
DESIGN	CHKD	APPROVED	DATE
ESS	ESS	ESS	8-13-69



002-105<sub>DN</sub>

LTR	ECO NO.	REVISION	ENG.	DATE
A	13412	RELEASED	SAS	4-7-89
B	13536	REVISE ARTWORK	SAS	6-13-89



MODEL	NEXT ASSEMBLY	QTY	USED ON
500	500-DIDIA		

DO NOT SCALE THIS DRAWING		DIMENSIONS IN PARENTS UNLESS OTHERWISE SPECIFIED		DATE 6-13-89		SCALE 1:1		TITLE	
XX=+/-0.015	ANG=±1/4°	DRN	SAS	ENG	ASPR	COMPONENT LAYOUT, DIGITAL I/O MODULE			
XXX=+/-0.005	FRAC=+/-1/64	MATERIAL		FINISH		C NO. 501-200			
SURFACE MAX. 63		N/A		N/A					

**KEITHLEY**  
Keithley Instruments Inc.  
Cleveland, Ohio 44135

# Using the DIO1A and KDAC500 Software

---

The DIO1A digital I/O module can be used in a variety of control and monitoring applications to source and sense digital logic levels. The module's 32 channels can be configured for input or output in groups of 8 channels. Each group of 8 channels is called a "port". Port A consists of channels 0-7, port B consists of channels 8-15, and so on. The DIO1A channels can be read or written individually, or as ports.

The DIO1A channels are non-isolated, meaning that DIO1A digital ground is connected to earth ground. Therefore, the low side of any circuitry connected to the DIO1A must also be at ground potential or the circuitry, DIO1A, or data acquisition system may be damaged.

DIO1A inputs and outputs are compatible with low-power Schottky transistor-transistor logic (LS-TTL). This defines the voltage and current levels associated with logical high and low. When configured for input, each DIO1A channel presents one LS-TTL load to an external circuit. The external circuit must be able to absorb (sink) at least 0.4mA at 0-0.5 volts for the DIO1A to recognize an input signal as a logic low. The same input must be capable of sourcing at least 20 $\mu$ A at

2-5 volts to drive a DIO1A input high.

A DIO1A output can drive up to 60 LS-TTL inputs, or 15 standard TTL inputs. This corresponds to a current sink capability of 24mA at 0-0.5 volts. Drive current available from a high output is approximately 1mA at 2-5 volts.

Do not connect signals of less than 0V or greater than 5V to the DIO1A or you may damage the module.

## Connection to Digital Logic

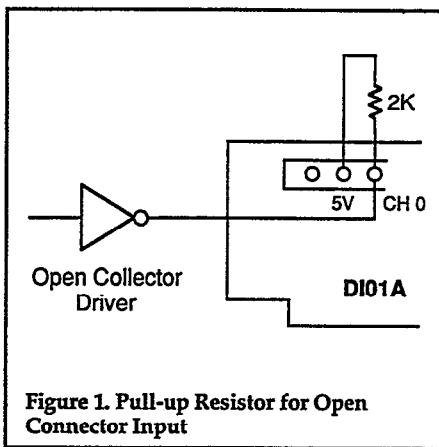
The DIO1A can interface many types of digital circuitry and logic families. However, it is important that any external voltages applied to the DIO1A inputs or outputs be within the range of 0-5V. Any input signal connected to the DIO1A must have sufficient sink and source capability to drive at least one LS-TTL load. Any load connected to a DIO1A output must not cause the DIO1A output to sink more than 24mA. Review the input and output specifications for the type of logic connected to the DIO1A to determine what conditioning, if any, may be required. This information is readily available in IC manufacturers' data books.

# Using the DIO1A and KDAC500 Software

The DIO1A can usually be connected to standard TTL logic without concern for voltage levels. However, the total current loading is important where one TTL output is connected to several DIO1A inputs, or where several external loads are connected to one DIO1A output. A variety of TTL sub-families exist, including low- medium- and high-power types. Each sub-family has different current sink and source specifications associated with input and output. Make sure that a DIO1A channel and the external logic can each sink and source sufficient current to control each other.

Open-collector digital circuitry is another type of TTL logic which requires special consideration. As with standard TTL, an open-collector logic low sinks some current at 0-0.5 volts. However, a logic high simply floats, and cannot actively drive an input. In contrast, a standard TTL gate contains internal circuitry to actively set a high output to at least 2V. To assure that an open-collector output will reliably control a DIO1A input, connect a 2k pull-up resistor between the DIO1A input and +5 volts.

CMOS and other low-power logic families operate at extremely low current levels, with logic high



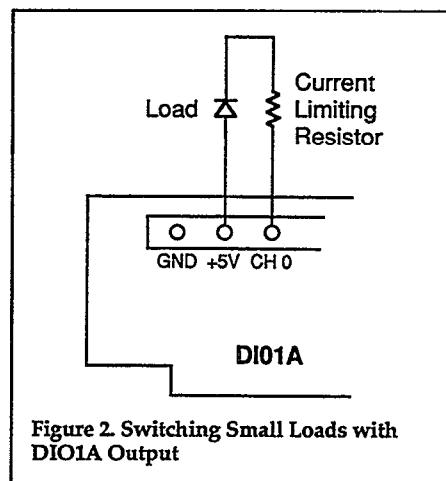
typically being 3-15 volts according to the circuit supply voltage. These devices usually require level translation or buffering to interface DIO1A channels. Integrated circuits such as the 4049 and 4050 buffer ICs are available for CMOS-to-TTL buffering. Buffering circuits based on discrete components can also be used.

**Note:** For reliable operation, a DIO1A input must be actively driven to 2-5 volts when the signal is high. Hardware switches, open-collector logic, and some other types of signals may allow a DIO1A input to "float" then they switch high. A floating DIO1A input may be susceptible to false triggering. A 2k pull-up resistor should be connected between a DIO1A input and +5V if the signal does not actively drive the input high.

## Controlling External Loads with the DIO1A

A DIO1A output can sink 24mA at 0.5V when at logic low. The same output can source about one milli-amp when at logic high. Thus, the DIO1A is better used as a switchable ground point, rather than a power source for controlling external loads. For example, a DIO1A output channel lacks sufficient current to light an LED indicator. For sufficient brightness, the anode of the LED should be connected to an external source of voltage, and the cathode connected to a DIO1A channel configured for output. When the DIO1A output is switched low, current flows through the LED to ground, completing the circuit. A 5V supply terminal is included on the DIO1A connector for such applications.

Applications may require con-



trolling higher-current DC or AC loads which are beyond the DIO1A's capacity. In this case, an AC or DC solid state relay can be used between the load and a DIO1A output. Normally, these relays are energized by switching a control terminal on the relay to ground, although some may require application of a 3-5V control voltage. A typical control current is 5-15 mA, which is well within the sink capability of a DIO1A output. The Keithley/DAC PROTO module can be used for assembling one or two auxiliary relay circuits for use with the DIO1A. If several relays are needed, it may be more cost-effective to use a PCM2 or PCM3 module which can control up to 16 relays.

**Caution:** The DIO1A should not be operated beyond the specified ratings or the module will be damaged.

Where signals with relatively high voltages are involved, or where isolation is needed, an opto-isolator or other conditioning circuitry can be used between the DIO1A and the external circuit. The user must choose the conditioning method according to the application, taking care that the signal ultimately

# Using the DIO1A and KDAC500 Software

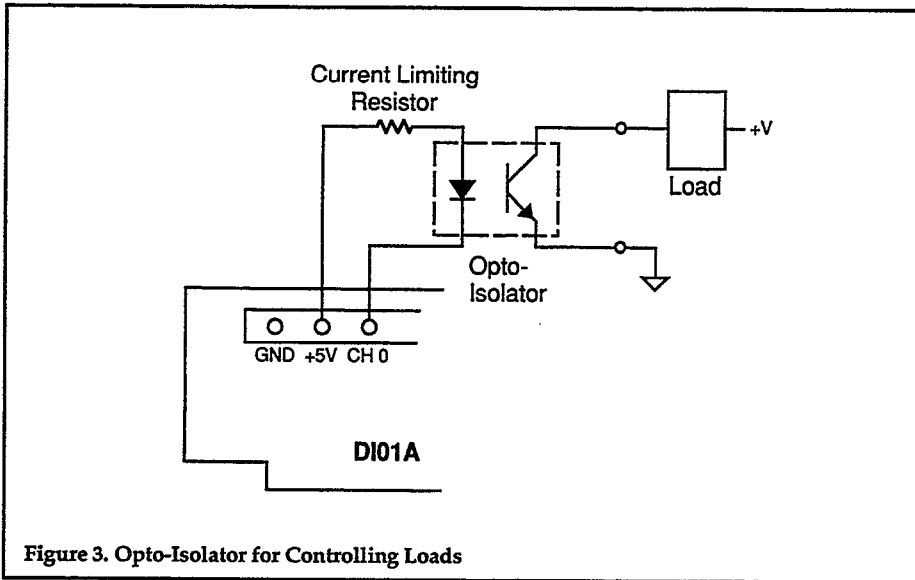


Figure 3. Opto-Isolator for Controlling Loads

applied to the DIO1A is within acceptable limits. Where many channels are involved, Keithley's DIM1 and DOM1 digital I/O modules offer 16 channels each, with optical isolation and the ability to interface voltages up to 28V and loads up to 50mA. Both modules maintain isolation of up to 500V between channels, or from channel to ground.

## Connection to Mechanical Switches

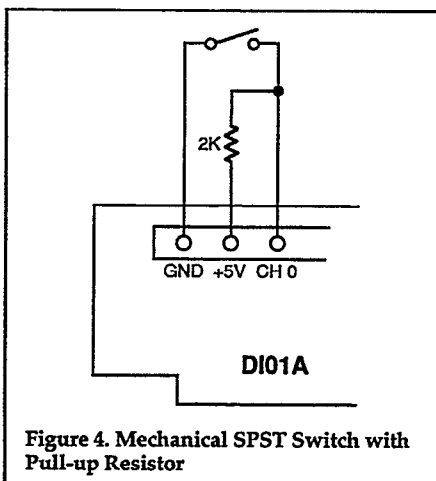


Figure 4. Mechanical SPST Switch with Pull-up Resistor

Monitoring the status of mechanical switches presents two problems. Where an application must detect a contact closure, it is a simple matter to connect the switch

from a DIO1A input to ground. However, this method will allow the DIO1A input to float when the switch is opened. The channel may be susceptible to noise or activity on other channels. The solution is to also connect a 2k pull-up resistor between the DIO1A input channel and +5V (Figure 4). When the switch opens, the channel will be pulled to +5V through the resistor, and be immune to noise and false triggering.

Second, mechanical switch contacts usually exhibit "bounce", in which electrical continuity rapidly makes and breaks several times at the instant the contacts close. In some digital applications, this bounce may be unacceptable. The solution is to use a "debouncing" circuit between the switch and DIO1A input. A flip-flop circuit suitable for debouncing SPDT limit switches is shown below. A second

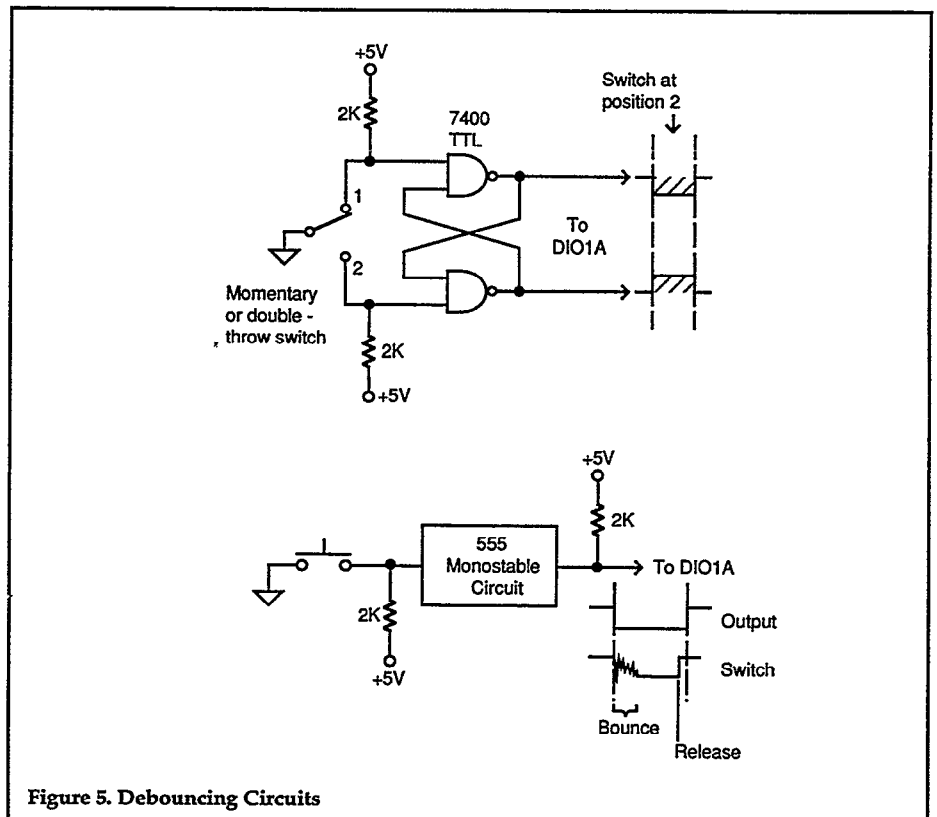


Figure 5. Debouncing Circuits

# Using the DIOA1 and KDAC500 Software

remedy, ideal for debouncing momentary push-button switches, uses a circuit which generates a single pulse when triggered by the switch. The pulse must be longer than the time it takes the contacts to settle (typically a matter of milliseconds). A 555 timer integrated circuit or various types of one-shot ICs can be used for this application. See IC manufacturers' data books for more information.

## Reading BCD Outputs

Before IEEE-488 and RS-232-C serial interfaces became the preferred method, binary-coded decimal (BCD) outputs were included on some instruments to make digitized data available to computers, calculators, or other digital equipment. In principle, BCD uses four lines to represent the numerals 0-9, but the actual encoding method and number of digits varied among manufacturers and instruments. A multi-digit BCD readout might use four separate lines for each digit, resulting in many lines but relatively simple decoding. Alternately, an instrument might multiplex several digits onto four lines plus a few additional digit select lines, resulting in fewer lines but more complicated decoding.

With attention to signal levels and the BCD encoding method, a DIO1A can be used to read BCD lines. The easiest method is to read the DIO1A inputs as ports. This will return eight data bits at one time. The DIO1A's 32 inputs can thus deal with 6-8 BCD digits. If the digits are multiplexed, the DIO1A can read an unlimited number of digits, but the requisite program will be more complex. The decoding of the signals must be done in software, and may be based upon a look-up table or

some calculation method.

## Software Considerations

The DIO1A powers up with all ports configured for input. Any software used with the DIO1A must be able to configure the module's ports for output if the application requires digital output.

Keithley's KDAC500 software provides complete control of all DIO1A functions. The following suggestions cover programming in a BASIC or QuickBASIC environment, but are equally useful in all KDAC500 versions. Most applications of KDAC500 and the DIO1A are simple in concept, and can be summarized as follows:

1. Where one digital input reading is required, a foreground read (FGREAD) command can be used. FGREAD will return the bit value representing a single input channel, or the byte value representing an input port (8 channels).

Where a port is read, the values of an individual bit within the byte can be extracted by "ANDing" the byte value with the weight of the bit's position. For example, the weight of the most significant bit (MSB) in a byte is 128. To test the value of the MSB in the byte value "BV", one would execute the BASIC statement:

BT = BV AND 128

If BT equals 0, the MSB value is 0, otherwise the MSB value is 1. Channel and bit values are covered in more detail in the DIO1A manual.

2. Where values must be read from a digital input channel at regular intervals, a background read (BGREAD) command should be

used. BGREAD will automatically allocate and store the data in a KDAC500 memory array. If single channels are being read, a KDAC500 bit array should be specified in the BGREAD command. If ports are being read, a KDAC500 byte array should be specified in BGREAD.

The BGREAD command will begin taking data when the KDAC500 INTON command is executed. One data value will be read when each interrupt fires. A BGSTATUS command should be used after the INTON to determine when all the data points have been collected. Afterward, the individual data values can be read from the array by using an ARGET command nested in a FOR-NEXT loop. If the array is a byte array, the values of individual bits in each byte can be extracted by ANDing the byte values with bit weights as described above.

3. Where one digital value must be programmed to a DIO1A output channel or port, a foreground write (FGWRITE) command is most convenient. FGWRITE can write a 0 or 1 to a single channel, or a byte value 0-255 to a port. The value of the byte can be calculated by adding up the weights of all the high bits in the byte. For example, a port is to be programmed with the bit values 11010011:

BYTE WEIGHT							
MSB							LSB
1	1	0	1	0	0	1	1
128	64	32	16	8	4	2	1
128 + 64 + 0 + 16 + 0 + 0 + 2 + 1							
TOTAL							= 211

# Using the DIOA1 and KDAC500 Software

---

The byte value 211 should be written to the port.

4. Where several values must be written to a digital output channel or port, a background write (BGWRITE) command should be used.

Before BGWRITE, a KDAC500 array must be allocated with ARMAKE to hold the output data. If the data values are intended for a single channel, the array must be a bit array, and the data will consist of 1's and 0's. If the data values are intended for a port, the array should be a byte array, and the data will consist of values 0-255. The array must be long enough to hold the desired number of data points.

The data values should be written to the array using an ARPUT command. A FOR-NEXT loop around ARPUT may be helpful in reducing the number of program lines need to fill the array.

If the desired output is a simple digital square wave, a single 1 and 0 can be written with ARPUT to a two-deep, KDAC500 bit array.

If the output involves a more complex pattern of pulses, ARPUT can be used to fill a longer KDAC500 bit array with the necessary order of 1's and 0's.

If the output involves a digital port, a KDAC500 byte array must be used. The values of individual bytes

should be calculated as shown in the weighting example above, and then written to the array with ARPUT.

In all cases for BGWRITE, output will commence when interrupts are turned on with INTON. One data value will be written from the array each time an interrupt fires. Note that one complete cycle through an array may require many interrupts, so the period of square waves and other pulses will probably not equal the interrupt rate. BGWRITE's cycling parameter can be set for a specific number of repeats, or for continuous output of the array. If cycling is not continuous, a BGSTATUS command should be used after INTON to determine when all the data points have been written.



---

## **KEITHLEY DAC**

### **Data Acquisition and Control Division**

**Keithley Instruments, Inc.** • 28775 Aurora Road • Cleveland, Ohio 44139 • (216) 248-0400 • Fax: 349-4569

**WEST GERMANY:** Keithley Instruments GmbH • Heiglhofstr. 5 • München 70 • 089-71002-0 • Telex: 52-12160 • Fax: 089-7100259

**GREAT BRITAIN:** Keithley Instruments, Ltd. • 1 Boulton Road • Reading, Berkshire RG 2 ONL • 0734-861287 • Telex: 847 047 • Fax: 0734-863665

**FRANCE:** Keithley Instruments SARL • 3 Allée des Garays • B.P. 60 • 91124 Palaiseau/Z.I. • 1-6-0115 155 • Telex: 600 933 • Fax: 1-6-0117726

**NETHERLANDS:** Keithley Instruments BV • Avelingen West 49 • 4202 MS Gorinchem • P.O. Box 559 • 4200 AN Gorinchem • 01830-35333 • Telex: 24 684 • Fax: 01830-30821

**SWITZERLAND:** Keithley Instruments SA • Kriesbachstr. 4 • 8600 Dübendorf • 01-821-9444 • Telex: 828 472 • Fax: 0222-315366

**AUSTRIA:** Keithley Instruments GesmbH • Rosenhügelstrasse 12 • A-1120 Vienna • (0222) 84 65 48 • Telex: 131677 • Fax: (0222) 84 35 97

**ITALY:** Keithley Instruments SRL • Viale S. Gimignano 4/A • 20146 Milano • 02-4120360 or 02-4156540 • Fax: 02-4121249