1. General Information

The DAQ-801/802 is a cost-effective data acquisition board that plugs into one of the expansion slots in the IBM PC or compatible personal computers. The DAQ801/802 board, shown in Figure 1.1, has capabilities for analog input/output (I/O), digital I/O, and timer/counter functions.

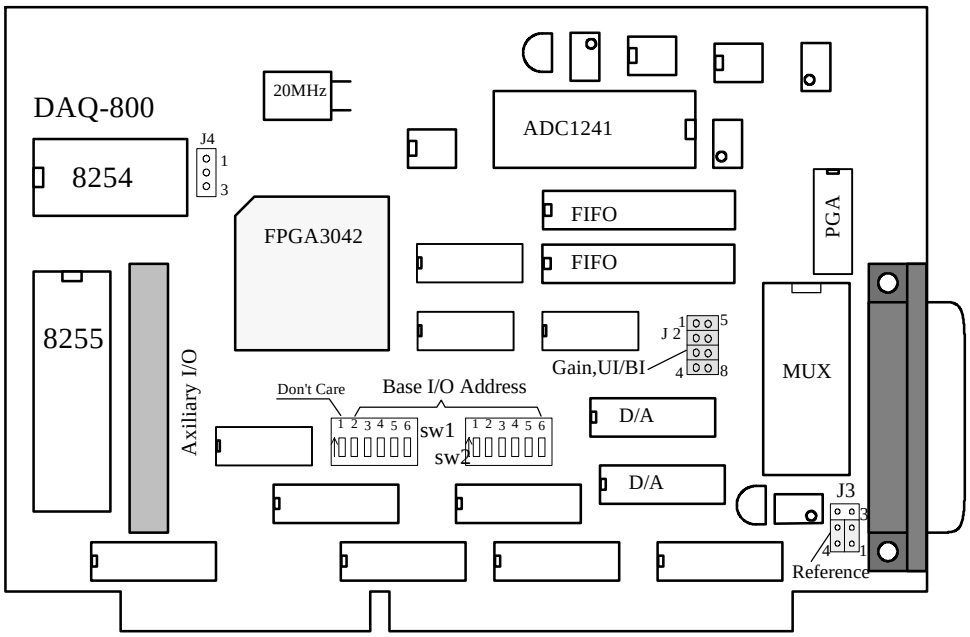

Figure1.1 DAQ-801/802 Board Layout Diagram

The functional block diagram is shown in Figure 1.2.

The analog inputs and outputs, external trigger signal, external reference signal, external clock signal and eight digital inputs and outputs are connected through a 37-pin D-type connector on the board. An auxiliary D-37 connector is employed to support an additional 24-bit digital I/O.

The pin diagram of the two D-37 connectors is shown in Figure 1.3.

The difference between DAQ-801 and DAQ-802 is that DAQ-801 is equipped with programmable gains of 1, 10, 100, and 1000, while DAQ-802 has selectable gains of 1, 2, 4 and 8.

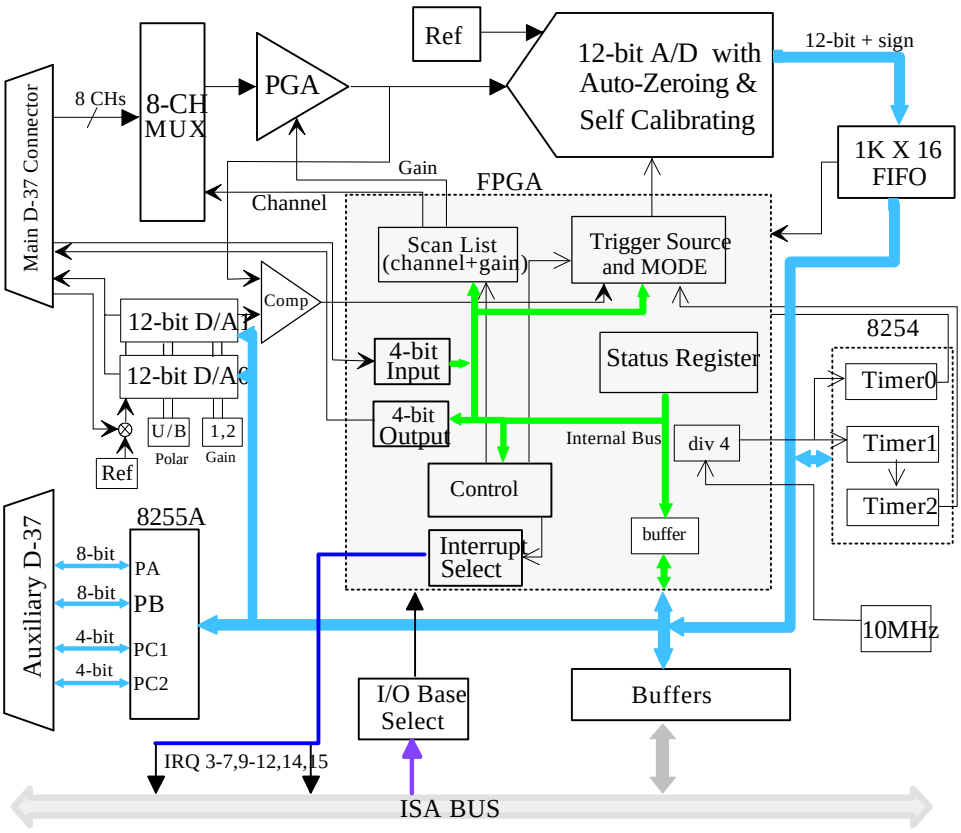

Figure 1.2 DAQ-801/802 Block Diagram

1.1 Analog Input Features

The DAQ-801/802 contains one +/- 12-bit analog-to-digital converter (ADC), and it supports the measurement of 8 differential input signals. Each differential analog input channel has +/- connection lines. When the - line is connected to ground, then the input signal becomes single ended. Therefore each of the 8 channel inputs can be configured as either differential or single ended input. The converter can be configured to receive analog input voltages within the ranges of 0 to +5V, or ±5V. The analog I/O connections are made via a 37-pin "D" connector on the board. The connector is DAS-1600 compatible.

The DAQ-801 provides gains of 1, 10, 100 and 1000, whereas DAQ-802 has gains of 1, 2, 4, 8. Table below shows the uni-polar input signal range.

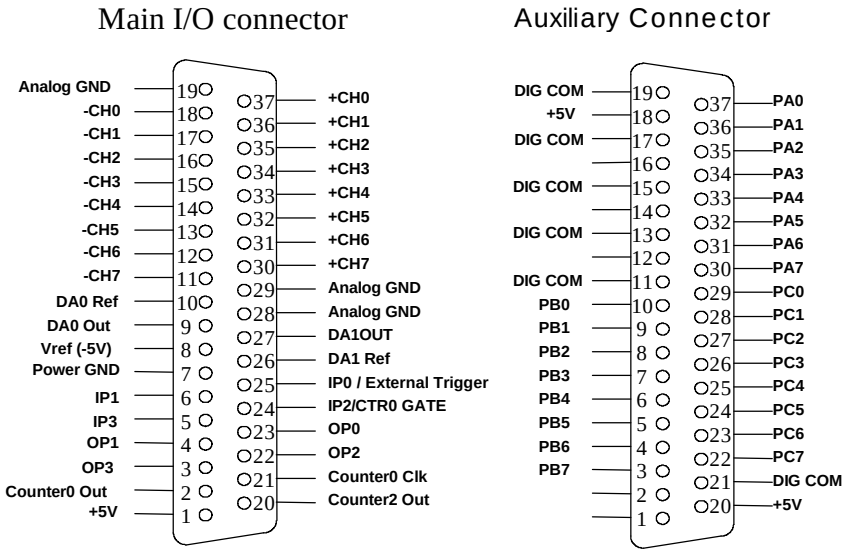

Fig.1.3 D-37 Pin Diagram

DAQ-801		DAQ-802	
Input Range	Gain	Input Range	Gain
0 to 5V	1	0 to 5V	1
0 to 500mV	10	0 to 2.5V	2
0 to 50mV	100	0 to 1.25V	4
0 to 5mV	1,000	0 to 0.625V	8

For a gain setting of 1, the 12-bit resolution (4096 counts) provides a least significant bit (LSB) value of 1.22mV in the 0 to +5V range. Table below shows the bi-polar input signal range.

DAQ-801		DAQ-802	
Input Range	Gain	Input Range	Gain
-5V to +5V	1	-5V to +5V	1
-500mV to +500mV	10	-2.5V to +2.5V	2
-50mV to +50mV	100	-1.25V to +1.25V	4
-5mV to +5mV	1,000	-0.625V to +0.625V	8

For a gain setting of 1, the 12-bit resolution (±4096 counts) provides a least significant bit (LSB) value of ±1.22mV in the ±5V range.

Trigger Source and trigger mode for Analog-to-Digital Conversions

When the DAQ-801/802 is powered up, it is in the idle mode and no conversion is performed. The conversion will begin upon receiving a command or a signal from the user. There are two ways to trigger the process: one is internal trigger; another is external trigger. The internal trigger is performed by a software program, whereas external trigger is

done by hardware. The external triggers supported are TTL trigger and analog trigger. The TTL trigger comes from pin 25 of the main connector. The analog trigger channel is the start channel in the scanning list, and it's trigger voltage is set by D/A channel 1 (see page 36). As soon as the board is triggered, A/D will immediately convert the analog signal into digital data, and will be stored into the data FIFO. In addition to internal /external triggering, there is also a trigger mode available. The trigger mode is used to decide whether the conversion is done once or multiple times. This same trigger mode also applies to scanning multiple channels, whether the scanning is done once or multiple times. The trigger source and the trigger mode are software selectable.

The following summarizes the triggering functions:

(a) Trigger source

- Software trigger
- External TTL trigger with falling or rising edge
- External analog trigger with low to high or high to low transition.

(b) Trigger mode .

- Single mode -- one scan for one trigger
- Continuous mode -- Continuous scanning for one trigger

Scan List

One of the functions provides by the DAQ-801/802 board is the scan list function. The scan list function enables you to digitize and collect the data from several analog input channels almost at the same time. This function performs A/D conversion from channel to channel at the highest possible speed. The user must specify the start and stop channel. As soon as the scan list function is called, the board will initiate the A/D conversion from the start channel. Right after the conversion is done and the converted data entered into the data FIFO, the control logic on the board selects the next channel and repeats the same task. This process continues until it reaches the stop channel. Each channel can have its own gain setting. Before this function is called, you must write into the registers in the FPGA describing the start channel , stop channel and their corresponding gains. The maximum number of scanning is eight. The scanning sequence is sequential and incremental by one. If you specify the start channel number as being the same as the stop channel number,

then it becomes a single channel acquisition mode, otherwise, it will scan from the start to the end channel. What will happen if you state the start channel to be 6 and stop channel to be 2? The scan sequence will proceed in the following order: 6,7,0,1,2. For a single trigger mode, it will scan once and stop. For a continuous mode, it will continue scanning at a speed set by the sampling rate, until it reaches the desired numbers of scans.

Sampling Rate

When digitizing the analog signal, one of the parameters that must be chosen by the user is the sampling rate. The sampling rate determines how fast the analog signal is digitized. According to sampling theory, the minimum sampling rate must be at least two times the frequency of the input signal. If the sampling rate is two times or higher, the information of the original analog signal can be recovered from the digitized data. The maximum sampling rate of DAQ-801/802 is 40KHz. This sampling rate is derived from the 8254 counter/timer chip available on the board. The 8254 chip has three 16-bit counters/timers. Timer 1 and timer 2 of the 8254 are cascaded to generate the sampling rate pulse. The output of this sampling rate pulse is used for triggering the A/D conversion. The clock input of timer 1 is 2.5MHz. The counting range of Timer 1 and Timer 2 each is from 2 to 65535, and the sampling period is computed as

Sampling period = Timer 1 data $\times$Timer 2 data $\times$ 400 nanoseconds

the sampling rate in Hz is the inverse of the sampling period. To set the highest sampling rate of 40kHz, the sampling period must be $\frac{1}{40}$ms or 25μs.

Calculation of the values to be filled into the Timer 1 and Timer 2 for the highest sampling rate of 40kHz is as follows:

Timer 1 data x Timer 2 data $= \frac{25\mu s}{400ns} = \frac{2500(}{400}$

$= 62.5 = 62$

The Timer 1 and Timer 2 data should be an integer number or ranging from 2 to 65535. Therefore 62.5 should be rounded to 62.

The actual sampling rate becomes $\frac{1}{62\times400}\left(\frac{1}{ns}\right)$, or 40.322 kHz.

Data FIFO

DAQ-801/802 implements a data FIFO (First In First Out) between the output of the A/D converter and ISA bus to buffer the data from the A/D converter output. Unlike the conventional A/D board where the output digitized data is fetched directly to the PC memory, the output data from

the A/D is fed into the FIFO first as a temporary storage. The length of the FIFO is 1024 sampling points. The FIFO circuit provides hardware flags for half full, full and empty signals. Utilizing these signals, the board can generate an interrupt to the PC when the FIFO is half full. When the PC is interrupted, the program in the interrupt routine can use the "STRING" move instruction to move the data in the FIFO directly into

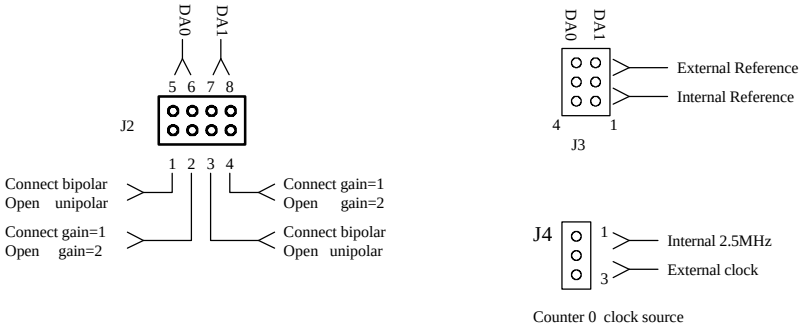

Figure 1.4 Jumper Setting Diagram

the PC memory at a very high speed. In this case, it only interrupts the PC every 512 samples thereby improving the speed of operation. In Windows application, the latency of the interrupt does not affect the integrity of the digitized data because the data was being stored into the FIFO. Interrupt latency and operation of data FIFO is mutually exclusive.

The status register (see Chapter 5, Address map) provides information about FIFO empty, half full and full.

1.2 Analog Output Features

In addition to the analog input channels, the DAQ-801/802 board contains two analog output channels. Each channel has its own 12-bit digital-to-analog converter (DAC). The analog outputs are buffered and the current output is capable of 1 mA. The user can select, through jumper setting, the output voltage range for each channel as 0 to +5V, 0 to +10V (Unipolar), or ±5V, ±10V (bipolar). The 12-bit resolution provides LSB value of 4.88mV on the ±10V range and 2.44mV in the 0 to +10V range.

Jumper settings

Refer to Figure 1.4 jumper setting diagram. The locations of the jumpers on the board are shown in Figure 1.1 DAQ-801/802 Layout Diagram.
Jumper 2 setting

Pins 1 & 5, 2 & 6 are used to set the output voltage range of D/A converter Channel 0 as follows:

Pins 1 & 5	Pins 2 & 6	Gain setting	D/A 0 output voltage
Connect	Connect	1	-5V to +5V
Connect	Open	2	-10V to +10V
Open	Connect	1	0 to +5V
Open	Open	2	0 to +10V

Likewise pins 3 & 7, 4 & 8 are used to set the output voltage range of D/A connector Channel 1 as follows:

Pin 3 & 7	Pins 4 & 8	Gain setting	D/A 1 output voltage
Connect	Connect	1	-5V to +5V
Connect	Open	2	-10V to +10V
Open	Connect	1	0 to +5V
Open	Open	2	0 to +10V

Jumper 3 setting

Connecting pins 1 and 2, pins 4 and 5 provides internal reference voltages for D/A channel 0 and D/A channel 1 respectively. Pins 2 and 3, pins 5 and 6 could be used for connecting external reference voltages for D/A 0 and D/A 1 respectively.

1.3 Digital I/O

DAQ-801/802 has 32 digital I/O lines. Out of 32 lines, 8 of them can be accessed through the main D-37 connector. There are 4 inputs (IP0 - IP3) and 4 outputs (OP0-OP3) as shown in Figure 1.3. The other 24 I/O lines are accessed through a second auxiliary D-37 connector and are generated by a 8255 programmable peripheral interface chip. 8255 has three ports (port A,B,C) and one control register. There are three modes of

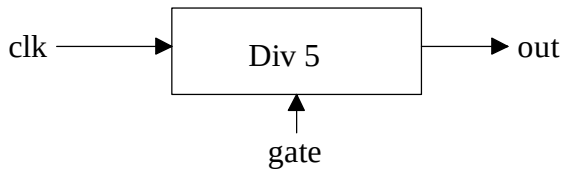

Figure 1.5 Counter Functional Diagram

operations in 8255 determined by the values written into the control register. Mode 0 is for basic input/output configuration where the output port is latched and the input port is not. Any one of the ports can be programmed as input or output. Port C can be further divided into two 4-bit I/O ports. Mode 1 employs Port A or Port B as the data port while using Port C as handshake, interrupt, and digital I/O lines. Mode 2 uses Port A as the bi-directional data port with Port B and C as control and digital I/O applications. For detailed functional description, the reader is referred to the data manual of 8255 by Intel or other manufacturers.

1.4 Counter / Timer

The 8254 counter/timer chip on the DAQ-801/802 provides three 16-bit counter/timer channels for time-related applications. Timer 1 and Timer 2 are cascaded together with an input clock of 2.5 MHz and its output is used for sampling rate application on the board. Timer0/Counter0 is available for the user and connection is provided to the main I/O D-37 connector. Three terminals of counter channel 0 are available through the main I/O D-37 connector.

The three terminals are accessible through Pin 2 (Counter 0 output), Pin 21 (Counter 0 CLK) and Pin 24 (Counter 0 Gate). The functional diagram of the counter is shown in Figure 1.5. The gate terminal should be logic high in order for the counter to function. If gate is held at logic low, the counter is rendered inoperative.

1.5 Interrupt

DAQ-801/802 supports AT style ISA bus Interrupt which includes IRQ2-7, 10-12, 14-15. The selection of interrupt is software programmable. An interrupt conflict can be resolved by moving the selection to the another available line without opening the computer case. This feature greatly enhances the convenience for the user if there is an interrupt conflict in the system.

There are four interrupt sources from DAQ-801/802 and only one is selected at a time to be connected to the ISA bus interrupt. These interrupt sources are:

- End of scan
- Data FIFO Half Full
- Data FIFO Full
- Timer 0

End of scan interrupt is normally used in conjunction with single trigger mode. Right after the scanning list is accomplished, the end of scan generates an interrupt to inform the computer to fetch the data. Data FIFO half full interrupt is for continuous trigger mode where the stream of data is getting into the FIFO and whenever the FIFO is half full, it interrupts the PC to fetch at least 512 sample points. This interrupt is very applicable to the Windows environment because of the latency problem inherent in Windows operating system. Data FIFO full interrupt is not recommended for application unless the interrupt routine is executed promptly before the next data points flow in; Otherwise, the overflow will occur and some data might be lost.

Timer 0 interrupt is used in conjunction with external counter/timer at the main D37 connector. External clock or pulses is connected to clock 0 input and the output of the counter/timer0 can be used for an interrupt source. When the user needs to interrupt the PC in a certain time interval, Time 0 can be programmed to meet your requirement.

1.6 Software Supports

The software drivers provide support for various programming languages like Microsoft C/C++, Borland C/C++, QuickBasic, Visual Basic for DOS and Turbo Pascal. A Dynamic Link Library (DLL) is provided for numerous programming languages under Microsoft Windows, as well as the Visual Basic Controls (VBX). Software support can be summarized into four categories:

1) DAQDRIVE Software driver
2) DAQ-VBX Visual-Basic Control
3) Labtech Notebook™ Data acquisition package
4) Snapmaster™ Data acquisition package

DAQDRIVE is low level driver consisting of a set of commands for the user. Programs written for DAQ-801/802 can be ported to DAQ-1201/1202 in the event that the user decides to upgrade the board in the future. DAQDRIVE is available for Window DLL as well as DOS environment. In the case of Visual Basic application, the software provides optional DAQ-VBX which is the visual basic custom control. This software is very useful for graphic presentation and interaction with the data acquisition boards. Because of its user friendly nature, this software is an amazing tool for creating your own graphic layout.

TestPoint™ is a new alternative to conventional programming languages. It is a tool to create custom test, measurement, and data acquisition applications. With TestPoint™, you can easily control hardware, create

user interface, process and display data and exchange information with other Windows programs by selecting objects that model the components of the application and combing its actions.

The last software presented are turn-key system software by Labtech Notebook and Snapmaster. These are menu driven software where the user does not need to write their application software. By using the Mouse and key stroke, the data acquisition can be obtained and plotted on the monitor and saved in a file. Post processing of frequency spectrum and other facilities are also available by these two packages. These software yield the shortest path to the user to get the system set up and running. No programming is required.

1.7 Power

The DAQ-801/802 is powered directly by the ±5V and ±12V power source provided by the computer bus.

1.8 I/O Terminal Strip Connection

The user can connect one or both I/O connectors on the DAQ-801/802 board to the external UIO-37 screw terminal panel through a cable assembly as shown in Figure 1.6.

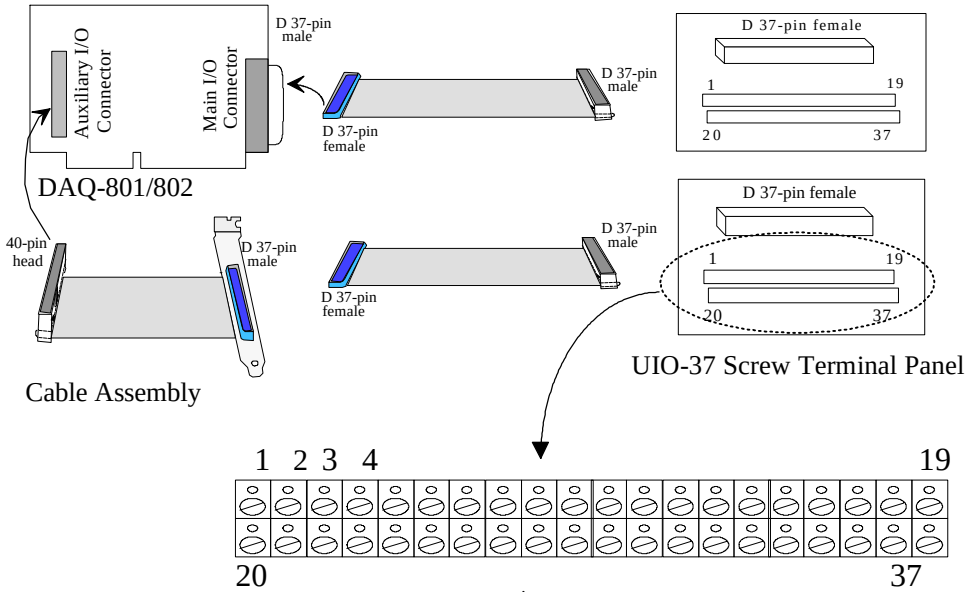

Figure 1.6. I/O Terminal Strip Connector Diagram

The cable assembly contains a cable for connecting the board to the screw termination panel. The UIO-37 screw termination panel contains two barrier strips, one containing 19 standard slot-head screw terminals, and another 18. UIO-37 terminal strip has 37 screw terminals and are labeled from 1 to 37. Each terminal has a number that corresponds to a pin number in D-37 at the main and auxiliary connector. UIO-37 provides easy connection to the external wiring.

1.9 Applications

The DAQ-801/802 performs one or several of the following functions: analog input (A/D), analog output (D/A), digital input, digital output and counter/timer. Typical applications of each function are listed in the following:

Analog Input (A/D)
Analog to digital (A/D) conversion converts analog sensor voltage into digital information, which enables the computer to process or to store the signals.

Typical applications are:

- sensor measurement
- waveform acquisition and analysis
- data logging

Analog Output (D/A)
Digital to analog (D/A) conversion is the opposite of A/D conversion. This operation converts the digital information to analog voltage for control of a process, for generating a waveform. Typical applications are:

- process control
- function generation
- pulse train generation

Digital Input/Output

Digital input function is useful in applications such as

- contact closure monitoring
- switch status monitoring

whereas digital output function is useful in

- relay control
- industrial on/off control

Counter/Timer

A Counter/timer is typically used in applications for

- event counting
- pulse generation
- frequency, period and pulse measurement

1.10 Specifications

Analog Input

Maximum Sampling Rate	40kHz
Channels	8 differential
Input Ranges	-5V to +5V
Output Data Code	Twos complement
Gain Ranges:	
Mode 801	1, 10, 100, 1000
Mode 802	1, 2, 4, 8
Input Impedance	1M ohm
Input Bias Current	50pA
Surge Protection up to	$\pm 20V$
Resolution	12-bit + sign
Scan Time	(from channel to channel)
with auto zeroing	25.6uS
without auto zeroing	15.2uS
Conversion type	Successive Approximation
Size of Scan List	8 Samples
Zero Error	Adjustable to 0
Gain error	Adjustable to 0

Analog Output

Channels	2
Output Ranges	$0-5V, 0-10V, \pm 5V, \pm 10V$
Output Data Coding	Straight Binary
Resolution	12-bit

Digital I/O (8 bits on main D37 Connector)

Output Bits	4
Input Bits	4

Timer/Counter

Number of Counters	3, down
Type	82C54

Digital I/O (24 bits on Auxiliary Connector)

I/O	24 bits
Type	82C55A

Power Requirements

+5V DC	750 mA typ, 900 mA max
-5V DC	15 mA typ, 20 mA max
+12V DC	60 mA typ, 80 mA max
-12V DC	60 mA typ, 80 mA max

Environments

Operating Temperature	$0 - 70^{o}C$
Interrupt Level	3-7, 9-12, 14, 15
Humidity	0-95%
Dimensions	$7in \times 4.8in$

2. Setup

This section describes how to unpack and configure the DAQ-801/802 board.

2.1 Unpacking

The DAQ-801/802 board is packed in an anti-static bag to avoid possible damage to the ICs on the board. Before removing the product from the bag, touch both the bag and the computer's chassis to establish grounding. After grounding has been established, remove the board and inspect it for signs of damage and loose components. If the board appears damaged, contact Omega Engineering at 1-800-872-9436 immediately.

2.2 Configuring the DAQ-801/802

You must decide the appropriate configuration for the DAQ-801/802 board depending on your applications. The DIP switch setting SW1, SW2 and Jumper setting J2, J3 and J4 must be selected before installing the board into the computer. While software programmable configurations

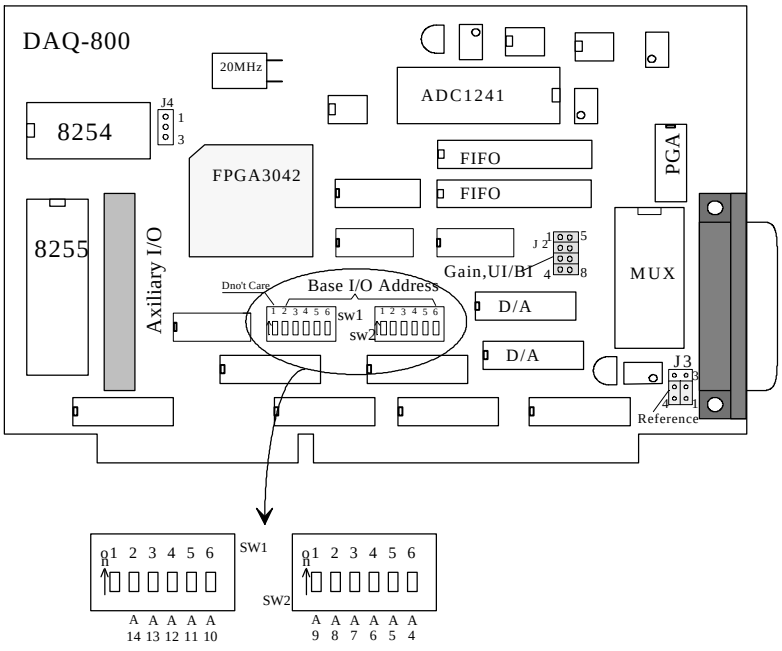

Figure 2.1 Base I/O Address Setting

can be done later, the following items are jumper configurable and must be decided before installing into the PC:

1. Base I/O address selection
2. D/A voltage reference,bipolar/unipolar output range.
3. internal/external clock source for Timer 0

Base I/O Address Selection

Each board inside the PC should have an input/output address. This address is similar to a mail box that the processor of the PC can send the data to or fetch the data from. Each board has its own unique address so that no two or more boards can share the same address. I/O address has 16 address lines. The previous IBM machines employed only the first 10

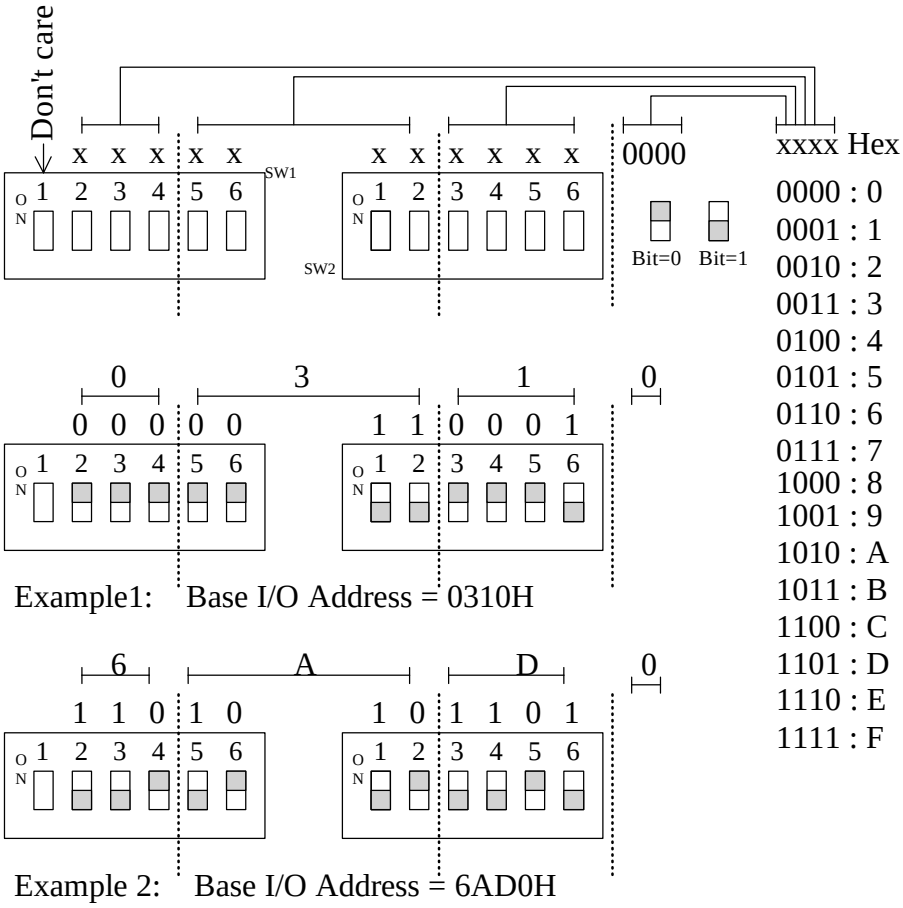

Figure 2.2 Base I/O Address Selection

lines for the address selection. Because the 10 lines address locations are limited, the later machines or add-in boards use more than 10 address lines for address decoding. Some of the I/O locations are pre-assigned and becomes de facto standard locations such as COM1 (3F8H) and COM2 (2F8H). Printer ports location LTP1 and LTP2 are also fixed.

I/O address of most other boards are flexible and can be selected anywhere in the I/O space as long as it is not occupied.

The I/O base address of DAQ-801/802 is set using the two DIP switches SW1, and SW2 shown in Figure 2.1. When setting the address selection switches, a switch bit in the "ON" position specifies that the corresponding address line is logic 0. Similarly, a switch in the "OFF" position specifies the corresponding address line to be a logic 1. The I/O base address can be selected from 0000H to 7FF0H with 0010H interval. The upper limit of 7FF0H implies that the address lines involves only 15 lines and the most significant bit A15 is always 0. Switches SW1 and SW2 select address lines A14 through A4. Since the board encompasses 16 register locations requiring 4 address lines A3, A2, A1, A0, only A14 through A4 address lines are used for address decoding. Switch SW1 position 2-6 and SW2 position 1-6 are for A14 through A4 address setting. Figure 2.2 shows how the switches on the DAQ-801/802 represent different address values.

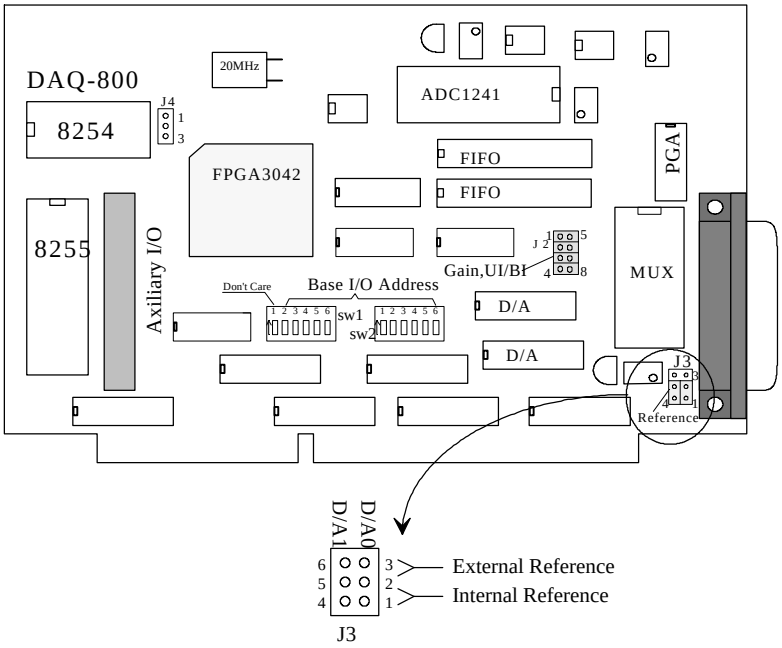

Figure 2.3 D/A Voltage Reference Selection

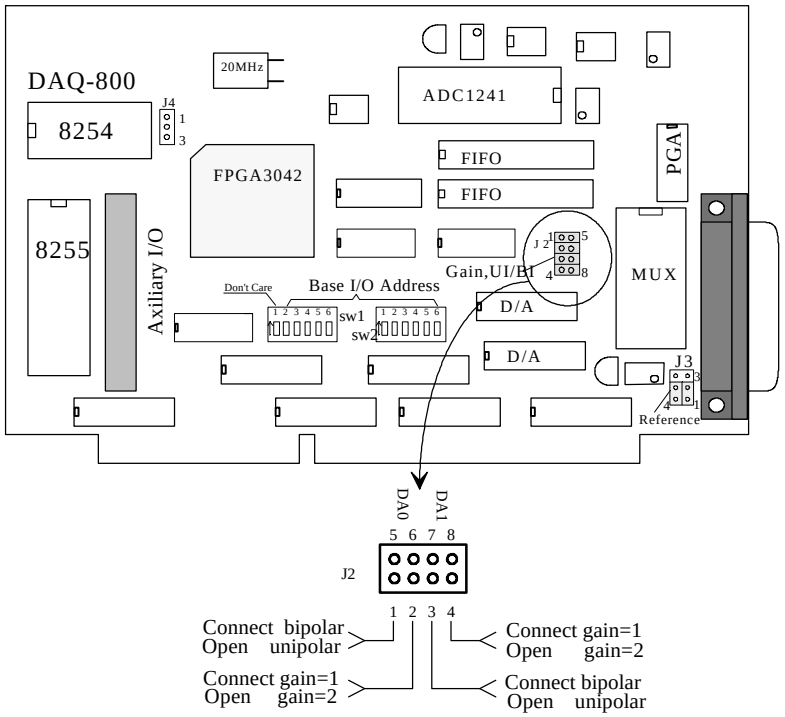

Figure 2.4 D/A Voltage Range and Polar Setting

The default address is set at 300H when shipped from the factory. If you desire to change the factory configuration, use SW1 and SW2 for base I/O address setting. An address of 310H, the SW1 and SW2 setting is shown in Figure 2.2 as Example 1.

D/A Voltage Reference, Bipolar/Unipolar Output Range Selection

Digital to analog converter uses AD7545 multiplying DAC in DAQ-801/802. Multiplying DAC requires reference voltage connecting to the chips. There are two reference voltages available on the board: one is internal 5 volt reference voltage and another from external inputs. Figure 2.3 illustrates internal/external reference voltage selection for DAC0 and DAC1. For internal /external reference voltage connection for both DAC0 and DAC1, place jumper at J3:

	DAC0	DAC1
Internal reference	Pins 1 & 2	Pins 4 & 5
External reference	Pins 2 & 3	Pins 5 & 6

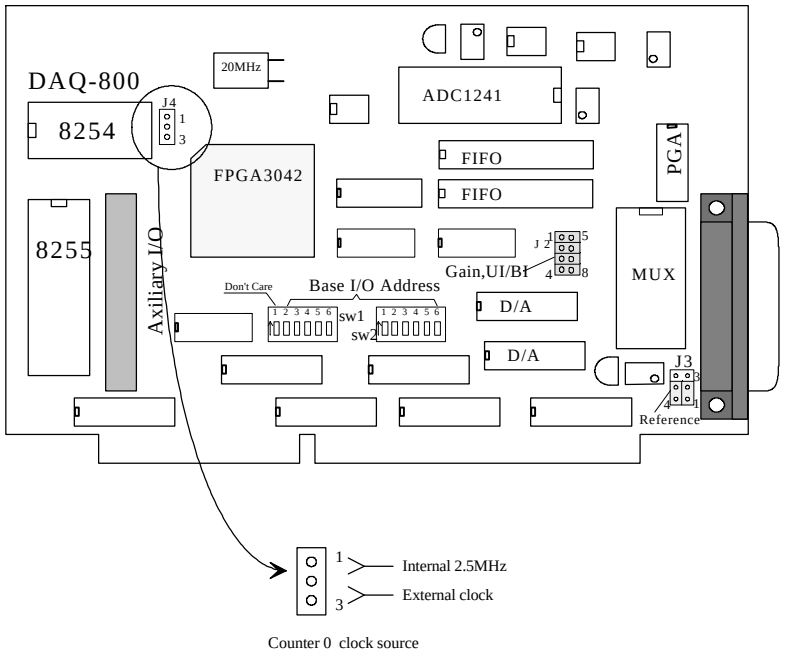

Figure 2.5 Internal/External Clock Selection

Two channel D/A outputs offer either Unipolar output or Bipolar output. In addition, there is a choice for the output voltage range. The user can choose 5V range output or 10V range output. 5V range is selected if the gain selection of 1 is made, whereas 10V range for the gain selection of 2 as shown at Jumper 2 header (Figure 2.4). Unipolar or bipolar output selection is made by simply placing the jumper or not placing the jumper at its corresponding location. Tables below demonstrate the combination of the jumper placement for different requirements.

Pins 1 & 5	Pins 2 & 6	D/A 0 output voltage
Connect	Connect	-5V to +5V, bipolar
Connect	Open	-10V to +10V, bipolar
Open	Connect	0 to +5V, unipolar
Open	Open	0 to +10V, unipolar

Pins 3 & 7	Pins 4 & 8	D/A 1 output voltage
Connect	Connect	-5V to +5V, bipolar
Connect	Open	-10V to +10V, bipolar
Open	Connect	0 to +5V, unipolar
Open	Open	0 to +10V, unipolar

Internal/External Clock Selection For Timer 0

Timer 0 which is made available at the main D37 connector is used for timing application or counting application to the outside circuit. If there is no need to do pulse counting or timing in your application, you can skip this section.

Jumper 4 (Figure 2.5) is used to select internal or external clock. This clock is connected to the clock input of Timer 0. Shorting pins 1 and 2 provides internal 2.5MHz to counter 0, where pins 2 and 3 can be connected to external clock provided by the user. When configured as an external input, it can be used as pulse counting for external event.

3. Cabling and Field Wiring

Before doing any connection or installation, make sure the computer is off and the power on your external circuit is also off.

Follow the instructions provided by your computer, install the DAQ-801 or DAQ-802 board into the computer after finishing the setup procedure in Chapter 2. As shown in Figure 3.1, there are two cables connected from the back of the PC to the UIO-37 terminal strips. As mentioned in Chapter 2, one of the D-37 connectors from the back of the PC is the main connector on DAQ-801/802. The main connector contains all the analog

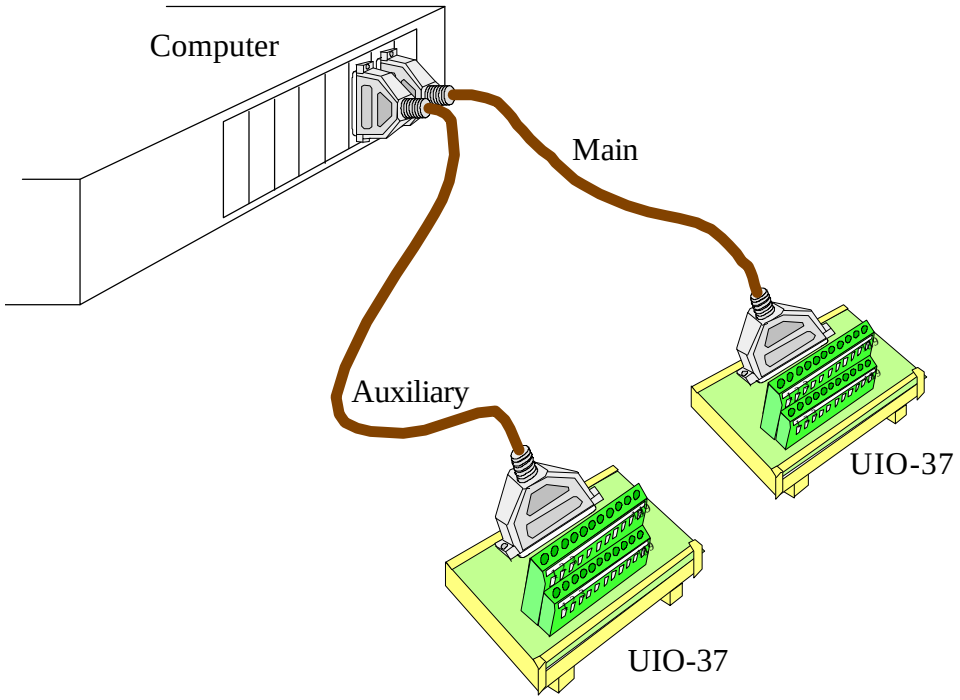

Figure 3.1 Cabling between DAQ-801/802 and UIO-37 Terminal Strip

input and output signals. The auxiliary connector which is also an D-37 connector in the back of the PC is used for digital I/O connection. Both of the connectors and cabling are DAS-1600™ compatible.

UIO-37 is the terminal strip which has number labeled on each position. The numerical number labeled on the terminal strip corresponds to the

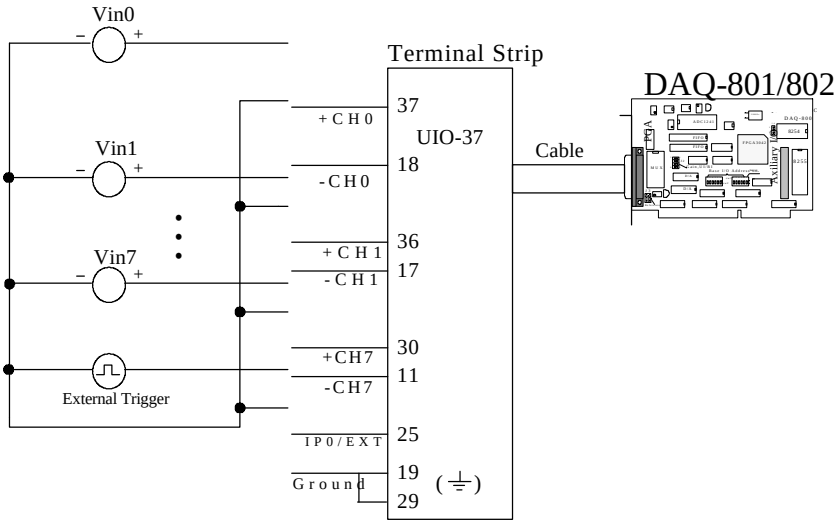

Figure 3.2 Field Wiring for Analog Input with Single ended Configuration

pin number on D-37 and is one to one correspondence. The wire size recommended for screw terminal connection is gage 16 - 28. After the installation, first turn on the computer, then the power on your external circuitry.

3.1 Analog Input Field Wiring

The analog input and output signals reside on the main D-37 connector. The analog signal can have single ended input or differential inputs. Figure 3.2 illustrates the field wiring for the single ended inputs. By looking at the main D-37 connector, there are 8 analog input channel with each channel marked with +CHx and -CHx. Single ended signal usually consists of two wires with one input signal wire (+) and another (-) as ground wire. When you do the wiring, connect signal wire with + to the +CHx input, and connect the other wire to ground. Pins 19 and 29 are both analog ground. You can connect all the - wires together and connect to pins 19 and 29 as shown in Fig 3.2.

If a high electrical noise environment exists, individual shielded wiring is recommended. Try to separate the power line and signal lines during installation and never put signal cables and high current or high voltage cable in the same harness. Electromagnetic field or high electric field intensity might deteriorate or interfere with the actual signal that you are measuring.

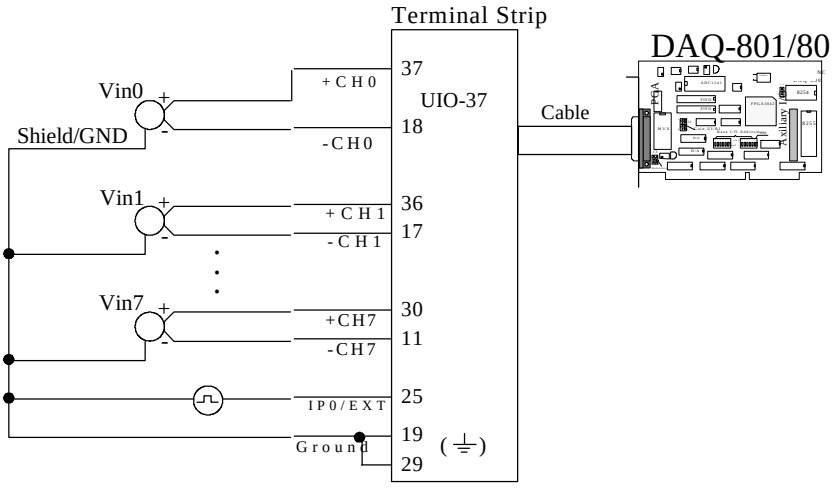

Figure 3.3 Field Wiring for Analog Input with Differential Input Configurat

For differential input configuration, the input signals normally have three wires: +signal, -signal, and shielded or ground wire. To do differential input field wiring, connect +signal to +CHx and connect -signal to -CHx at the terminal strip (UIO-37). All the shielded or ground wires will be tied together and connected to analog ground of pin 19 and pin 20 at the UIO-37. The advantage of having differential input wiring is that the noise picking up along the +signal and -signal lines will be canceled out at the instrumentation amplifier on the board and leave only the pure signal at the input of the A/D converter. This configuration is assumed to be better than the single ended analog input. As long as the noise level is greater than 1 or 2 LSB, the differential configuration will definitely improve the accuracy of the input signal.

If a sensor output has only two wires with +signal and -signal, and no shielded cable is provided, can you connect these two signal wires into differential input field wiring? The answer is yes. You may connect +signal to +CHx and -signal to -CHx. In doing so, the noise induced along the lines will be canceled at the instrumentation amplifier.

Any signal returning wire should not be mixed up with power returning wires. Especially for single ended configuration, signal ground and power ground should not be the same point because the power ground carries current and it might give you noise or offset voltage at the input of the multiplexer on DAQ-801/802.

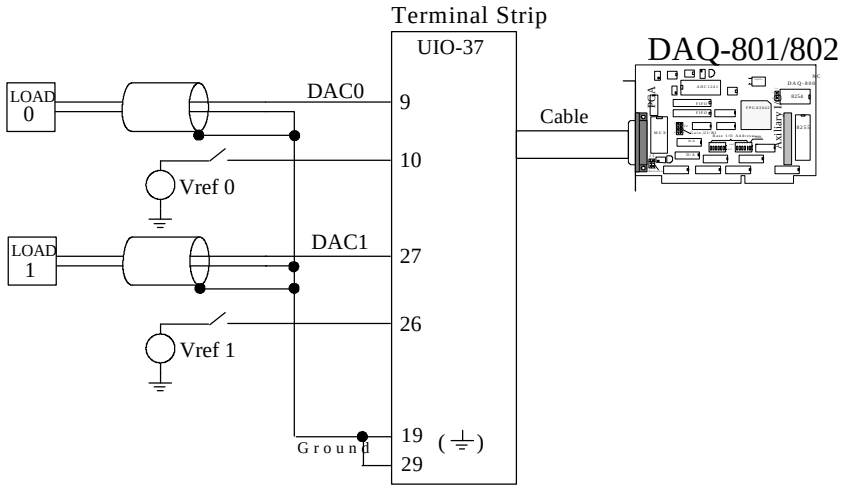

Figure 3.4 Analog Output Field Wiring

3.2 Analog Output Field Wiring

Typical analog output field wiring is shown in Figure 3.4. In this case, two conductor cable with shield is recommended. The positive output is connected to positive terminal which is pin 9 for DAC0 and pin 27 for DAC1. The other terminal and shield is tied together and connected to analog ground of pin 19, pin 29. As mentioned earlier in Chapter 2 the D/A output utilizes internal reference or external reference. If it is an internal reference configuration, no connection is required at pin 10 and pin 26 of their respective reference voltage inputs. However, if the board is configured as an external reference for D/A output, then the external reference voltage must be supplied to it at the terminal strip. These external reference voltages can be a fixed reference voltage or other time varying signal. Since the D/A chip is a multiplying DAC, the output voltage actually is the result of multiplying the D/A output with the reference voltage input. If the reference voltage is fixed, or non time varying, then the reference voltage only affects the magnitude of the output voltage. If the reference signal is time varying signal, then the output D/A signal could become an amplitude modulated signal.

3.3 Counter/Timer Field Wiring

There is a 16-bit counter/timer(Intel 8254 or equivalent) available for the user. 8254 counter/timer chip has three 16-bit counter/timers. Timer 1

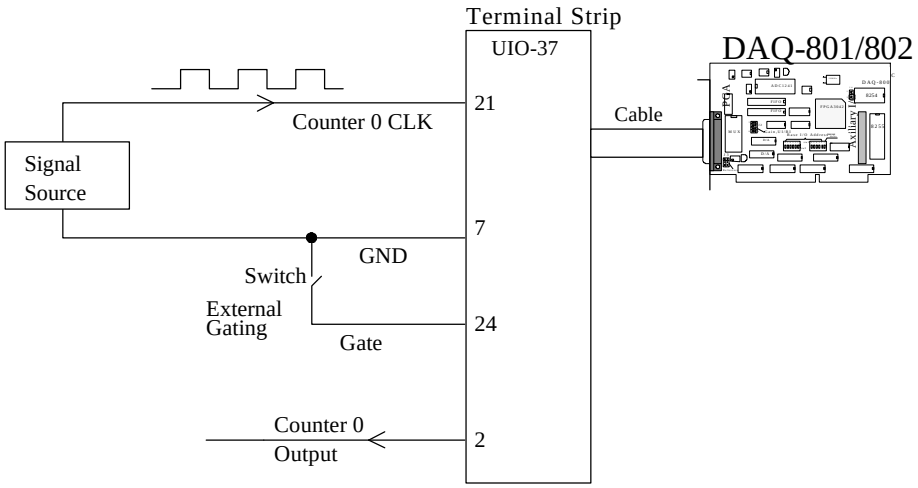

Figure 3.5 Timer/Counter Field Wiring

and Timer 2 are cascaded together using 2.5 Mhz input clock to generate the pacer clock for the A/D function. Only timer 0 is available for the user. All the control signals, input clock and output clock are made available at the terminal strip or main D37 connector. This 16-bit counter/timer becomes a timer if input clock is connected to the internal 2.5 Mhz clock provided on the board (see the jumper configuration on Chapter 2. J4). The output of this timer is taken out at Pin 2. The frequency of this output signal will depend on how the counter is programmed. The gate signal at Pin 24 controls the output signal. When the gate is opened, the output at Pin 2 becomes active. When the gate is closed or connected to ground, the output becomes 0. Therefore you can employ this gate signal to control your output clock.

When the jumper configuration at J4 is for external clock, then this 16-bit counter/timer becomes a counter. A counter can be used to count pulses and the result of the counting can be read by the program and into the computer. The signal to be counted is connected to Pin 21. The counting can be enabled or disabled by controlling the gate signal at Pin 24. If the switch is open, the counter is enabled and any pulses coming in at Pin 21 will be counted. The result will be read by software. If the switch is closed, the counter is disabled. No counting occurs even though there are pulses coming in at Pin 21. When the counter overflows, a pulse will be generated at Pin 2 at the output clock.

3.4 Where To Go From Here - Programming

The DAQ-801/802 is now configured and ready for use. Depending on the type of application software to be used, the user may wish to review one or more of the followings:

1. Chapter 4 of this document provides a basic theory of operation of the adapter for users who wish to learn the technical details about the operation of the DAQ-801/802.
2. For users who want to program the adapter with direct I/O transfers to the DAQ-801/802s register set, Chapter 5 provides an address map and a detailed description of each I/O register.
3. Users who would like to write custom application software without programming the DAQ-801/802 directly should consult the DAQDRIVE software reference manual. DAQDRIVE provides a library of data acquisition subroutines and is included free of charge with the DAQ-801/802.
4. For turn-key data acquisition software (i.e. LabTech Notebook™, SnapMaster™, or TestPoint™) information please consult the Omega Data Acquisition catalog.

4. Theory of Operation

4.1 Signal Flow

The block diagram of the signal flow for DAQ-801/802 is shown in Figure 4.1. The heart of the architecture of this data acquisition board is contained in Field Programmable Gate Array (FPGA). This FPGA controls all the timing required for A/D conversion, D/A output and

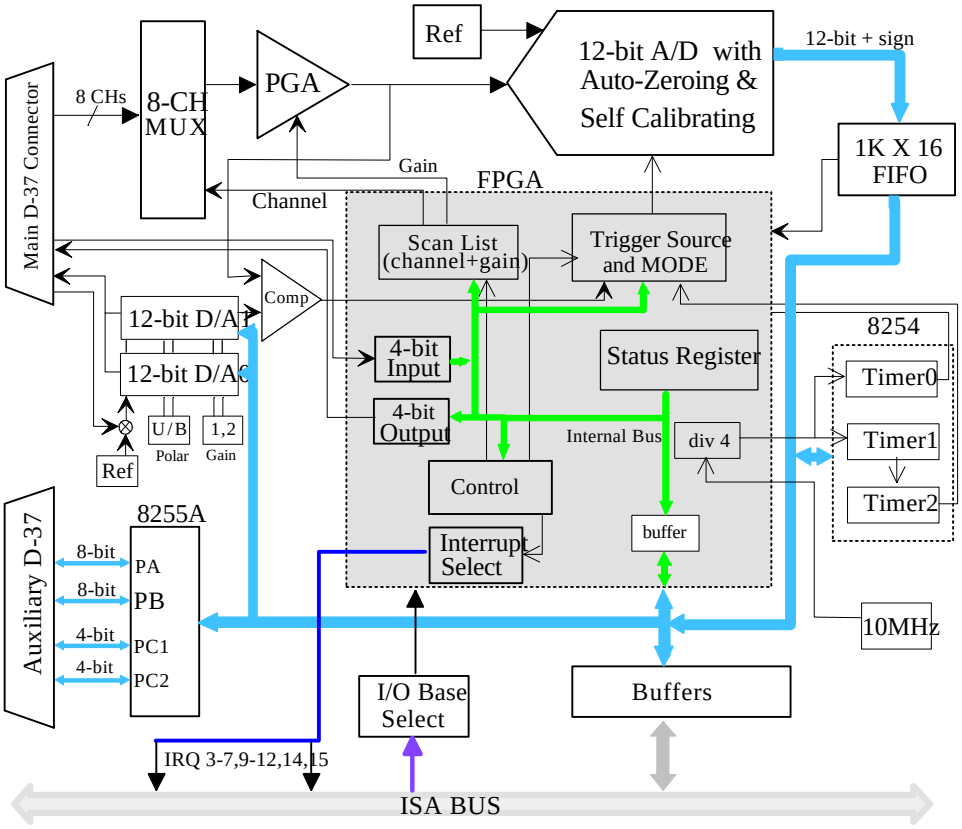

Figure 4.1 DAQ-801/802 Block Diagram

digital I/O. There is a common bus on the board which carries the binary data generated by A/D conversion or binary data sent to D/A converter. The data coming in from the digital input port or going out to the digital output port also passes this common bus. The common bus is separated from the ISA bus by a 16-bit data bus buffer.

At the outset, FPGA sends out all the necessary information to select the multiplexer, the gain of each channel, and sampling rate . This is done

by the CPU sending the appropriate data to the registers residing inside FPGA. When these settings are finished, the input signal comes through the multiplexer and to the A/D converter via Programmable Gain Amplifier (PGA). The A/D converter initiates the conversion by the command of a signal from FPGA . The output of the A/D converter is a 12-bit plus sign binary data and its data format is in 2's complement form. The stream of the A/D output is now fed into a First In First Out 1K X 16 FIFO. The CPU then reads the data from the FIFO. The CPU can read the data after each A/D conversion or CPU can wait until FIFO is half full, then read the data all at once using MOVE STRING operation. Since end of conversion by A/D or FIFO half full , FIFO full and FIFO empty can generate interrupt to the processor, the efficient background data acquisition can be achieved without using polling technique. The FIFO is located at Base +0 address, any read from this location will yield 16-bit data to the CPU. This 16-bit data consists of 12-bit plus sign bit and upper few bits are either 0 or 1 depending on whether it is positive or negative.

The commencement of A/D conversion is initiated by a trigger from FPGA. The trigger source can be from the software trigger or external trigger. The software trigger is generated by writing an output command to register Index 2. Each time a trigger occurs, a conversion happens and data is generated. One trigger stimulating one A/D conversion is called single trigger mode. For a continuous mode, the system will be continuously doing A/D conversion at a certain rate after it is triggered. This rate is referred to as sampling rate. The sampling rate is constructed by cascading timer1 and timer2 together with a 2.5 Mhz clock connecting to the clock input of timer1. The rate is determined by how the timer1 and timer2 is programmed. The maximum sampling rate is 40 kHz.

If the analog signal is coming from only one single channel, then the analog multiplexer is fixed. All the converted digitized data will belong to an analog signal. To observe several channels almost simultaneously, scan operation must be performed. Scan operation requires specifying the start channel and the end channel. This is done by writing a byte consisting of start channel and end channel to a register inside FPGA. When the scan sequence convenes, the trigger signal will start the sampling A/D and at the same time send a signal to switch the multiplexer to the next channel. While waiting for A/D conversion to finish, the next selected analog input signal is settled down at the instrumentation amplifier and no time is wasted. When the end of conversion pulse EOC appears indicating A/D conversion is done, the converted data will be written into the FIFO and the multiplexer is also switched to the next channel. This process continues until the end of the

channel is reached. For a single trigger mode, the controlled logic in FPGA will scan once and stop. For a continuous mode, the controlled logic will scan from the start channel to the end channel, and then wait for the next triggering signal which is the same as the sampling clock. The sampled data is sequentially stored in the FIFO and read by the processor. You must sort out the data for each channel if you wish to plot the curves.

D/A operation is simply performed by sending a 12-bit data through the data buffer to the selected D/A channel. The strobe signal necessary for latching the 12-bit to the register inside D/A converter is generated by FPGA. The data sends to Base+8 will go to channel 0 and the data sends to Base+A will go to channel 1.

The board implements 82C54 counter/timer chip to generate timing signal for the sampling rate. 82C54 has three 16-bit counter/timer. Two of them are used for sampling rate and another is available for the user at the main connector. The board is also augmented with a 82C55 to provide additional digital I/O. Programming of 82C54 counter/timer and 82C55 programmable peripheral interface chip are byte oriented operations. The locations of these two chips are specified in the address map. The data path to and from the processor utilizes half of the internal bus on the board. All the control signals for these two chips are generated by FPGA.

4.2 Analog Input

DAQ-801/802 provides 8 differential analog input channels. Single ended analog signal can be realized by wiring -channel input to analog ground. HI-507A multiplexer has input over-voltage protection and consequently power turn on sequence is immaterial. Nonetheless it is preferred to have computer power on first and the power to the external circuit next. The output of the multiplexer is connected to the programmable gain amplifier (PGA) which has gains of 1,10,100,1000 for DAQ-801 and gains of 1,2,4,8 for DAQ-802. The maximum voltage output of the PGA is limited to +/- 5V. The maximum analog input range at the multiplexer is also +/- 5V. Therefore various gains can be selected to optimize the accuracy of the input signal for data conversion. For instance, if you know your input signal falls within +/- 40 mV, you can use DAQ-801 and set the gain at 100 which will yield the maximum voltage of +/- 4V at the output of PGA. However, if the signal is only +/- 0.625V at the input, then DAQ-802 can be used since a gain of 8 can be chosen to give the maximum accuracy. The output of the PGA is then connected to the sampling A/D converter (ADC1241).

The A/D converter is 12-bit with sign binary output. The format for binary output is in 2's complement form. The converter type is successive approximation and its conversion time is 13.6 us for non auto zero and 24 us for auto zero function. The A/D chip possesses an auto zero function which calibrates the zero reading before converting each point. The time requires for auto zero is 10.4 us when the clock is running at 2.5 Mhz. Therefore auto zero function will slow down the speed of data acquisition system if you select it.

The A/D converter chip also provides full range calibration. The full range calibration needs to be done once at the beginning of the system setup and is not necessary to repeat again. The offset adjustment and full range adjustment are done at the factory .

4.3 Analog Output

The 12-bit data sent by the processor to Base + 8 location using word transfer will travel from ISA bus to internal bus and finally get to the latch at the D/A converter of channel 0. When the latch receives the new data, it will go through digital to analog conversion and analog signal corresponding to the binary value appears. For each binary value, it will get its corresponding analog voltage. The analog output voltage is then buffered through operation amplifier to pin 9 of the main connector. The buffered amplifier is used to increase its output driving capability. Any 12-bit data sent by processor to address Base + A will terminate at the latch of D/A channel 1. The same type of buffered circuit is attached to the output of channel 1 D/A converter.

AD7545 D/A converter is a multiplying DAC. A reference voltage must be provided to the chip. LM336-5.0 supports this reference voltage and yields a +/- 5V analog output with the buffer stage configured at a gain of 1. When the buffer stage sets the gain by jumper configuration to a gain of 2, the output will have a maximum of +/- 10V. The jumper configuration also enables the user to select unipolar output with 0 to 5V or 0 to 10V.

Reference voltage can come from an external circuit by jumper selection. The reference voltage does not have to be a constant voltage. If the reference voltage is a time varying signal, multiplying this voltage with D/A output will produce a complex signal. Not only the analog value sent by processor changes, the maximum magnitude determined by the reference input also changes. If the reference voltage is not a time varying signal , adjusting the reference voltage will change the D/A output range.

You can customize the D/A output voltage by feeding the appropriate reference voltage.

4.4 Digital I/O

The digital I/O function of DAQ-801/802 provides 4-bit TTL compatible input and 4-bit TTL compatible output. Both are accessed through the main connector on the board. The digital data flow is controlled by FPGA. In addition to this 4 bit digital I/O, there is a 82C55 programmable peripheral interface chip on the board which supplements 24 additional digital I/O lines. 82C55 is located at Base + C and occupies four consecutive I/O addresses. The 24 digital I/O is divided into three 8-bit ports and each port can be configured to be either input or output. There are three modes of operation for 82C55: mode 0 for basic input output, mode 1 for digital I/O with handshake lines and mode 3 for bi-directional data transfer.

The control word located at Base +F determines which mode it will be in. The data transfer for 82C55 is all 8-bit oriented and control signals to manage the data traffic is generated by FPGA. All three ports are wired to a double row header (auxiliary connector). The auxiliary connector is situated at the left hand side of the board. A separate cable provides connection to the second slot, next to the main connector for outside access.

At power up all three ports are configured to be input ports. However, the configuration can be altered by writing a value to the control word register. For detailed discussion of 82C55, refer to Intel catalog covering 82C55 chip.

4.5 Counter / Timer

82C54 has three counter/timers. Timer1 and Timer2 are cascaded to generate pacer clock or sampling clock. Only counter 0/timer 0 is available for the user. The counter/timer has clock input, gate control input and pulse output. When the gate control signal is low, no counting/timing occurs. DAQ-801/802 has an internal pull up at the gate control input. No connection at this gate will enable the counter. When used as a counter, the content of the counts can be read by the processor. The application includes frequency measurement, event counting etc. When used as a timer, the output pulse rate can be programmed by the processor and its applications such as time proportional output, pulse outputs etc. can be realized.

5. Address Map for DAQ-801/802

The address map of DAQ-801/802 occupies 16 I/O locations. It starts from Base+0 and ends with Base + F. The actual addressable registers in DAQ-801/802 are more than 16 locations. This is done by utilizing index register at Base +2; the contents of the index in the index register will address to different sets of locations when you write or read to Base + 3. The following table illustrates the address map of DAQ-801/802.

Base + 0,1	Read 16-bit DATA FIFO Write 8-bit Base + 0 Gain of channel 0-3 Base + 1 Gain of channel 4 -7
Base + 2	Read 8-bit Index Register Write 8-bit Index Register
Base + 3	8-bit Read/Write Index 0 Configuration register 1 Interrupt level selection register 2 Auxiliary control register 3 Interrupt enable register 4-7 82C54 counter/timer
Base + 4	8-bit Read/Write Status register
Base + 5	8-bit Read only Interrupt status register
Base + 6	4-bit Read/Write 4-bit digital I/O
Base + 7	8-bit Read/Write Scan channel register
Base + 8..9	16-bit Write only D/A channel 0
Base + A..B	16-bit Write only D/A channel 1
Base + C..F	82C55 8-bit Read/Write
Base + 8000	Read/Write Disable/Enable DAQ-801/802

Base + 0,1: Read -- 16-bit data will be read from DATA FIFO into the PC

This is a 16-bit data transfer. The data format is a 2's complement number. For positive number the 16-bit data format is: 0000XXXX XXXXXXXX. For negative number the data format is: 1111XXXX XXXXXXXX.

Write is a byte operation. Write to Base+0, and Base+1.

Writing to Base+0 will select the gains for channel 0-3 and writing to Base+1 will select the gains for channel 4-7. Each channel has 4 different gain selections (1,2,4,8 or 1,10,100, 1000) and therefore two bits are required for each channel.

	DAQ-801	DAQ-802
00	1	1
01	10	2
10	100	4
11	1000	8

Base + 0 D7 D6 D5 D4 D3 D2 D1 D0

D1D0: Gain selection for channel 0
D3D2: Gain selection for channel 1
D5D4: Gain selection for channel 2
D7D6: Gain selection for channel 3

Base + 1 D7 D6 D5 D4 D3 D2 D1 D0

D1D0: Gain selection for channel 4
D3D2: Gain selection for channel 5
D5D4: Gain selection for channel 6
D7D6: Gain selection for channel 7

Base + 2: Index register

Write: 00000XXX. Write a byte to this location will set an index in the index register. The last 3-bit represents the number of index ranging from 0 through 7.
Read: 11111XXX. The last 3-bit is the index number written.
Indexing operation occupies two I/O location. First writing an index to the index register at address Base+2. If the index is 0, the next byte written to Base +3 will be going to index 0 register.

Base + 3: After writing to index register at Base + 2, the next write to Base + 3 will direct you to different index registers.

Index 0 : Configuration register XXXXD3D2D1D0

D3: 1 -- digital trigger
0 -- analog trigger *see example at the end of this chapter.
D2: 1 -- single trigger
0 -- continuous
D1: 1 -- internal trigger
0 -- external trigger
D0: 1 -- rising edge trigger
0 -- trailing edge trigger
X : don't care

Index 1: Interrupt level selection register D7D6D5D4XXXX

D7D6D5D4	
0000	disabled
0001	disabled
0010	disabled
0011	IRQ3
0100	IRQ4
0101	IRQ5
0110	IRQ6

0111	IRQ7
1000	disabled
1001	IRQ9
1010	IRQ10
1011	IRQ11
1100	IRQ12
1101	disabled
1110	IRQ14
1111	IRQ15
X	don't care

Index 2: Auxiliary control register , Write only

D7D6D5D4D3D2D1D0

D7: 1 -- software trigger
0 -- no software trigger

D6: reserved

D5: 1-- flushing ADC FIFO-reset the FIFO pointer
0 -- no action

D4: 1-- calibration cycle. This is to perform full scale calibration. During calibration, there is a status bit located at Base+4 status register indicating the completion status. When this bit is set, it takes 5.584 ms to finish calibration. Normally the calibration is done once at the beginning of data collection.
0 -- no calibration.

D3: setting this bit to "1" will stop the scanning operation in continuous mode (trigger mode: continuous). During the middle of scanning operation, the scanning function will not stop immediately when the bit is set to 1. It will continue A/D conversion until the scanning sequence is done and then it will be stopped. After ceasing operation, the state is still in continuous mode waiting for trigger to occur.

D2: not used

D1: not used

D0: not used

Index 3: Interrupt enable register. Write/Read

D7D6D5D4D3D2D1D0

D7: 1 - global enable (must be set for any interrupt)
0 -- disable

D6: not used

D5: not used
D4: 1 -- Timer0/Counter0 interrupt enable
0 -- disable
D3: 1 -- external trigger interrupt enable
0 -- disable
D2: 1-- FIFO full interrupt enable
0 -- disable
D1: 1 -- FIFO half full interrupt enable
0 -- Disable
D0: 1 -- End of scan interrupt enable
0 -- Disable

Index 4..7: Write/Read 8254 counter/timer
Index 4: Write/Read Counter0/Timer0
Index 5: Write/Read Counter1/Timer1
Index 6: Write/Read Counter2/Timer2
Index 7: Write Control word register of 8254

Base + 4: Status register , Read/Write , D7D6D5D4D3D2D1D0
Read operation:
D7: 1 -- EOC end of conversion
0 -- EOC not finished
D6: always 0 bipolar mode.
D5: 1 -- AZ (auto zero
0 -- not AZ
D4: 1 -- FIFO empty
0 -- FIFO not empty
D3: 1 -- FIFO half full
0 -- FIFO not half full
D2: 1 -- FIFO full
0 -- FIFO not full
D1: 1 -- BUSY . This bit is set to 1 during full scale calibration or when the scan sequence is not completed yet.
0 -- not busy
D0: 1 -- A/D enable
0 -- disable
Write: XXD5XXXXD0
X--- don't care
D5: 1 -- AZ (auto zero). Perform auto zero for each A/D conversion. Time required: 10.4us. It will slow down the sampling rate if each point needs auto zero calibration.
0 -- non AZ.

D0: 1 -- arming the A/D conversion waiting for trigger signal to start operation.
0 -- disarm.

Base + 5: Interrupt status register. Read only D7D6D5D4D3D2D1D0
Read: D7 --- not used
D6 --- not used
D5 --- not used
D4 --- Timer0/counter0 interrupt status
D3 --- External trigger interrupt status
D2 --- FIFO full interrupt status
D1 --- FIFO half full interrupt status
D0 --- End of scan interrupt status
Interrupt status bit : 1 for interrupt occurred, 0 not occurred. Must be read to cleare.

Base + 6: 4-bit digital I/O Write/Read XXXXD3D2D1D0
Write: D3D2D1D0 Output 4-bit to output port located at main D37 pin 3, pin 22, pin 4, pin 23.
Read: D3D2D1D0 Read the input port located at main D37 pin 5, pin 24, pin 6, pin 25.

Base + 7: Scan channel register
Write/Read XD6D5D4XD2D1D0
This register is to store the information of the scanning channels. The beginning channel and ending channel are specified in this register.
X --- don't care
D6D5D4 --- specified the beginning channel
D2D1D0 --- specified the ending channel

Base + 8, 9 : DAC0 D/A channel 0 output port
Write only: D15D1 4 -------D4D3D2D1D0
16-bit

Base + A,B: DAC1 D/A channel 1 output port
Write only: D15D14 ---------D4D3D2D1D0
16-bit
Base + C..F: 82C55 Programmable Peripheral Interface chip
Base+ C: PIO Port A
Base +D: PIO Port B
Base +E: PIO Port C
Base +F: PIO Control word

Base + 8000 : Board enable /disable

Read: disable the DAQ-801/802

Any read to Base +8000 will cause DAQ board to be disable.

Write: enable the DAQ-801/802

Any write to Base +8000 will cause DAQ board to be enable.

* Example--analog trigger

If you prefer to write your own code for the data acquisition system in stead of using the DAQDRIVE software driver, the example below illustrates the procedure you have to follow to perform an analog trigger data acquisition:

a) Set the analog threshold voltage (or triggered voltage) by sending its value to DAC1 (analog output channel 1 located at Base+A)

b) Program the index 0 register :

D3: 0 analog trigger
D2: single/continuous
D1: 0 external trigger
D0: rising/trailing edge trigger

c) Program the scan sequence of start channel/end channel at Base +7

d) Program the gain of each channel at Base +0,1

e) Arm the circuit by writing D0=1 to Base +4.

After the above procedures are executed, the board will be waiting for triggering by comparing the threshold voltage with the analog input voltage at the start channel.

6. 8254 COUNTER/TIMER

6.1 INTRODUCTION

Counter/timer uses the industrial standard 8254 which provides three independent 16-bit counters, each capable of handling clock input up to 10 Mhz. Two of the counter/timers are cascaded together to provide on board sampling clock for Analog to Digital converter. Only counter/timer0 is available to the user. When the clock input to the counter0 is originated from the board with 2.5 Mhz clock , the counter becomes a timer. The counter 0 output can be used for timing applications such as real time clock, digital one-shot, programmable rate generator, square wave generator, etc. When the clock input to the counter 0 is external , the counter/timer0 operates as a counter. The counter or timer operation is jumper selectable. The selection is illustrated in Jump 4 of Figure 2.5 .

Jumper 4	**Application**
Connect Pins 1 & 2	Timer application
Connect Pins 2 & 3	Event counting

Timer 0 output (Counter0 Out) and external pulse input (Counter0 Clk) are accessed through the main D-37 connector.

6.2 FUNCTIONAL/OPERATIONAL DESCRIPTION

8254 counter/timer occupies 4 consecutive I/O locations; namely

location for counter 0 (first location)
location for counter 1 (second location)
location for counter 2 (third location)
location for control word (fourth location)

Control word is used to configure the different operations of counter/timer. When the board is powered up, the state of the 8254 is undefined. How the counter operates is determined by how it's programmed. The counter is programmed by writing a control word to the fourth location and then an initial count to the first location for counter 0. The control word for counter 0 is as follows:

D7	D6	D5	D4	D3	D2	D1	D0
0	0	RW1	RW0	M2	M1	M0	BCD

RW is defined as Read/Write

RW1	RW0	
0	0	counter latch command
0	1	Read/Write least significant byte only
1	0	Read/Write most significant byte only
1	1	Read/Write least significant byte first then most significant byte

M is the mode of operation

M2	M1	M0	
0	0	0	mode 0: event counting
0	0	1	mode 1: timing retriggerable one-shot
0	1	0	mode 2 : rate generator
0	1	1	mode3 :squarewave mode
1	0	0	mode 4 : software triggered strobe
1	0	1	mode 5 :hardware triggerable strobe

BCD

0: Binary count
1: BCD count

6.3 WRITE OPERATION

The programming procedure for the 8254 is that the control word must be written before the initial count is written. The initial count must follow the count format specified in the control word. For instance if RW1 RW0 is written as 11, then the initial count format is to write the least significant byte (LSB) first, and then the most significant byte(MSB). The 16-bit counter is always counting down. After writing the initial count to the counter 0, the first pulse that appears at the clock 0 input (Counter0 Clk as shown at D-37 of Main I/O connector in Fig 1.3) will load the value to the counter and then second pulse will initiate the count-down.

6.4 READ OPERATION

The value of the counter 0 can be read without disturbing the count in progress. The read format depends on how RW1 RW0 is written in the control word register. If RW1 RW0 is written as 01, then every time the

counter is read, only the least significant byte is received. However, when RW1 RW0 is 11, then the first read will be the least significant byte, and the second read will be the most significant byte. The control word needs to be written only once unless it's used for latching operation. The 16-bit counter can be read any time in LSB-MSB sequence. It is possible that during LSB-MSB read operation, as soon as LSB is read, a carry bit is generated by LSB counter such that MSB is incremented by one. As the MSB is read, the 16-bit number results in an erroneous number. In order to avoid this undefined result, Gate0 input (CTR0 GATE at the main D37 connector) must be disabled to stop counting or using the latch command. The latch command is used to instantly transfer the 16-bit value from the output of the counter to a latch. After the data is latched, then read the counter.

For example, a control word with the following data is written into the control word register:

D7	D6	D5	D4	D3	D2	D1	D0		
0	0	1	1	0	0	0	0	30H	Counter 0

This control word (30H) is to Read/Write the counter 0 with LSB first and then MSB next with mode 0 for events counting and employing binary count format. After the control word is written, then write an initial count with LSB first and MSB next. Assuming the CTR0 GATE is enabled, the counter will start to count down with each pulse. To read the 16-bit count at anytime, the following latch command can be issued to the control word register:

D7	D6	D5	D4	D3	D2	D1	D0		
0	0	0	0	X	X	X	X	0H	where X is don't care

This latch command will latch the 16-bit count from the output of the counter 0 to a latch. The counter is now ready to be read at the counter 0 location. Two subsequent read operations must be done before the next latch command can be issued. If the data is not read, another latch command will not update the new count.

6.5 MODE DEFINITION

Mode 0: It is used for event counting. Its Counter0 Out at the connector can be used for an interrupt signal at the terminal count. Counter0 Out is low when the control word is written and will remain low until the counter reaches zero. Counter0 Out then goes high and will remain high

until a new count or a mode 0 control word is written. The initial count is loaded to the counter on the next clock pulse and therefore the first clock pulse does not decrement the counter. Thus Counter0 Out will go high at N+1 clock pulses after the initial count is written.

CTR0 Gate=1 ---------- counting
CTR0 Gate=0 ---------- not counting

CTR0 Gate can control the counting after the counter is programmed.

Mode 1: Timing application---hardware retriggerable one-shot.
Clock can be from the on board 2.5 Mhz clock and the time duration of one shot can be programmed by the initial count. Counter0 Out is high and will go low following a trigger at the CTR0 GATE. Counter0 Out will remain low until the counter reaches zero. Counter0 Out will then go high until it is triggered again.

Detailed explanation of Mode 0 through Mode 5 can be found in the manufacturer's 8254 data sheet.

6.6 HOW TO ACCESS COUNTER/TIMER

There are two locations to be accessed for counter/timer 0: one is the control word location (the same location for latching command) and the another is the Read/Write location. The access to these two locations is done by indexed addressing. Indexed addressing requires selecting the content of the index register first before reaching to the actual location. Therefore two instructions must be executed for each access. The first instruction chooses the index number at Base+2 for direction and the next instruction accesses the actual location at Base+3.

Let index_reg represents Base address + 2
port represents Base address+3

Writing a control word or latching command to control word location is done by:

a) outp(index_reg, 7) ------ selecting index register 7
b) outp(port, value) ------ sending control word or latching command. value is a control word or 0H for latching

Writing an initial count to counter 0:

a) outp(index_reg,4) ------ selecting index register 4 where counter 0 is located.

b) outp(port,value0)------ sending the initial count of LSB
c) outp(port,value1) ------ sending the initial count of MSB

Reading the current counts at counter 0:
a) outp(index_reg,4) ------- selecting index register 4 where counter 0 is located.
b) value0=inp(port) ------- reading the LSB
c) value1=inp(port) ------- reading the MSB

6.7 PROGRAMMING EXAMPLES

Listing 1. 8254 Programming Example 1

```
#include <conio.h>
#include <stdio.h>
//=====================================
// base_address+2: index register
// base_address+3: data port address
//
//
//                index register                  register in 8254
//                index4:                         timer/counter 0
//                index5:                         timer/counter 1
//                index6:                         timer/counter 2
//                index7:                         control register
//
// To access one of the register in 8254 need two steps:
//  step 1: write the index number of 8254 register to IO address base+2
//  step 2: read/write base+3 to access the selected register in 8254
// Example: write 30H to control register of 8254
//   step 1:  outp(0x302, 7);              assume base addres is 300h
//                                         index register is 300h+2=302h
//   step 2:  outp(0x303,0x30)   data port address is 300h+3=303h
//                                         write 74H to control register in 8254
//=====================================

void main(){
unsigned base_address = 0x300;
unsigned short byte_LSB;
unsigned short byte_MSB;
outp(base_address+0x8000, 0);         // Enable the DAQ-801/802 board

//=====================================
// This example programs counter 0 of 8254 in mode 0 with initial value
// of 2675H, then reads the value of couter0 using simple read operation
// and display the higher byte and lower byte on the screen.
//=====================================
  outp(base_address+2, 7);             //The 8254 control register address
  outp(base_address+3, 0x30);         // Set control register
  outp(base_address+2, 4);             // Select timer index
  outp(base_address+3, 0x75);          // Low byte
  outp(base_address+3, 0x26);          // High byte

  printf("Press any key to read the counter value, press ESC to quit\n");
```

```
do  {
        byte_LSB = inp(base_address+3);      //Read the least significant byte
        byte_MSB = inp(base_address+3);      //Read the most significant byte
        printf("High byte: %3d,   Low byte: %3d\n",byte_MSB, byte_LSB);
        }
while (getch() != 0x1b);
}
```

Listing 2. 8254 Programming Example 2

```
#include <conio.h>
#include <stdio.h>

void main(){
unsigned base_address = 0x300;
unsigned short byte_LSB;
unsigned short byte_MSB;

outp(base_address+0x8000, 0);         // Enable the DAQ-801/802 board

/*************************************************************/
/* This example programs counter 0 of 8254 in mode 0 with initial value      */
/* of 2675H, then reads the value of couter0 using Counter Latch Command     */
/* and display the higher byte and lower byte on the screen.                 */
/*************************************************************/

  outp(base_address+2, 7);                    //The 8254 control register address
  outp(base_address+3, 0x30);         // Set control register
  outp(base_address+2, 4);                    // Select timer index
  outp(base_address+3, 0x75);                 // Low byte
  outp(base_address+3, 0x26);                 // High byte

  printf("Press any key to read the counter value, press ESC to quit\n");

do  {
        outp(base_address+2, 7);              //Select control register of 8254
        outp(base_address+3, 0);              //Conter Latch Command
        outp(base_address+2, 4);              //Select counter 0 of 8254
        byte_LSB = inp(base_address+3);       //Read the least significant byte
        byte_MSB = inp(base_address+3);       //Read the most significant byte
        printf("High byte: %3d,   Low byte: %3d\n",byte_MSB, byte_LSB);
        }
while (getch() != 0x1b);
}
```