# Servo To Go, Inc.

# ISA Bus Servo I/O Card
# Model 2 Hardware Manual

## Motion Control for 2, 4, 6, or 8 Servo Motors,
### Plus 32 Bits of Digital I/O, 8 Channels of Analog Input and Timer/Interrupt Generator


Servo To Go, Inc.

Phone: 317-257-1655

Fax: 317-251-3958

E-mail: servotogo@msn.com

Internet world wide web URL: http://www.servotogo.com

8117 Groton Ln. Indianapolis, IN 46260-2821

# Table of Contents



# 1.0 Summary

The hardware described in this manual is a low cost, general purpose, motion control input/output board which can control up to eight motors simultaneously from an ISA-bus based computer such as an IBM compatible PC. The following is a summary of the hardware functionality:

- **Encoder Input**
  - Up to 8 channels of encoder input
  - A, B, and I (sometimes called 'marker') input
  - 24 bit counters
  - Single-ended or differential (RS422 compatible) input signals
- **Analog Output**
  - Up to 8 channels of analog output
  - + 10 V to - 10 V span.
  - 13 bit resolution
- **Digital Input and Output**
  - 32 bits, configurable in various input and output combinations
  - Opto-22 compatible
- **Analog Input**
  - 8 channels of analog input
  - 13 bit resolution
  - Configurable as +/-10V or +/-5V spans.
- **Interval Timers**
  - Capable of interrupting the PC
  - Timer interval is programmable to 10 minutes in 25 microsecond increments
- **Battery Backup Input**
  - Used to maintain encoder counting capability in event of a power failure.
- **Watchdog Timer**

Sample applications include:

- Robotics
- Machine tools
- Motion picture camera control
- Specialty machine control
- Controls design education
- Automated test equipment
- Medical instrumentation
- Virtual reality “rides”

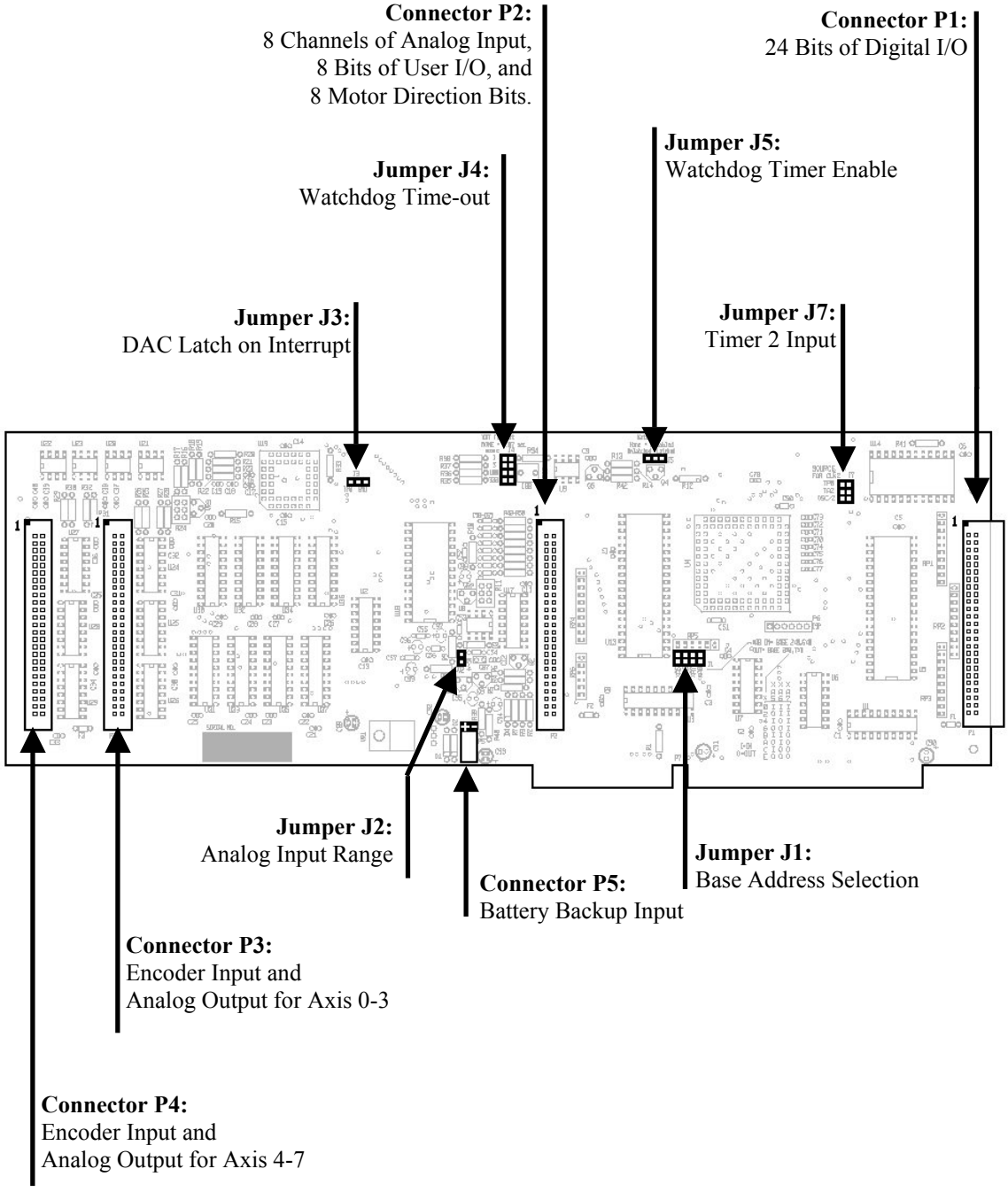
Although the board is typically used to perform servo motor control, it can also be used for specialized I/O. For example, in encoder position monitoring or any other application where encoder input as well as analog and digital I/O are required.

The board is simply and efficiently accessed by the use of a set of registers located in the I/O space of the PC. Connection to the outside is accomplished through four 50-pin connectors.

# 2.0 Setup and Installation

## *2.1 Jumper and connector locations*



## 2.2 Base Address - Jumper J1

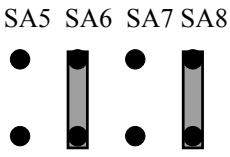
Insert jumpers to select the base address of the board.

| SA5 | SA6 | SA7 | SA8 | Addresses Selected | | |
|---|---|---|---|---|---|---|
| IN | IN | IN | IN | 200-1F | & | 600-1F |
| OUT | IN | IN | IN | 220-3F | & | 620-3F |
| IN | OUT | IN | IN | 240-5F | & | 640-5F |
| OUT | OUT | IN | IN | 260-7F | & | 660-7F |
| IN | IN | OUT | IN | 280-9F | & | 680-9F |
| OUT | IN | OUT | IN | 2A0-BF | & | 6A0-BF |
| IN | OUT | OUT | IN | 2C0-DF | & | 6C0-DF |
| OUT | OUT | OUT | IN | 2E0-FF | & | 6E0-FF |
| IN | IN | IN | OUT | 300-1F | & | 700-1F |
| OUT | IN | IN | OUT | 320-3F | & | 720-3F |
| IN | OUT | IN | OUT | 340-5F | & | 740-5F |
| OUT | OUT | IN | OUT | 360-7F | & | 760-7F |
| IN | IN | OUT | OUT | 380-9F | & | 780-9F |
| OUT | IN | OUT | OUT | 3A0-BF | & | 7A0-BF |
| IN | OUT | OUT | OUT | 3C0-DF | & | 7C0-DF |
| OUT | OUT | OUT | OUT | 3E0-FF | & | 7E0-FF |

Established I/O address for many common devices appear in appendix B.

Base address selection example:

| SA5 | SA6 | SA7 | SA8 | Address Selected | | |
|---|---|---|---|---|---|---|
| OUT | IN | OUT | IN | 2A0-BF | & | 6A0-BF |



## 2.3 Analog Input Range - Jumper J2

Insert a jumper in J2 to select a -5 to + 5 volt input range. With no jumper, the range is - 10 to +10 volts.

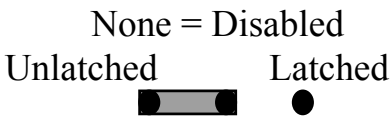

## 2.4 DAC Latch on interrupt - Jumper J3

The board can be configured to generate periodic interrupts to the PC. This is necessary when the board is used in a control algorithm application. Each sample period, the control algorithm needs to read the encoders and write to the DACs. The encoders are automatically latched when the periodic interrupt occur. Using this jumper, the user can select weather or not the DACs are also simultaneously latched at the instant of the period interrupt. If both the encoders and DACs are latched by the hardware, the control algorithm can be performed at anytime during the window. With no change in performance from one sample period to the next. If the jumper is on the right-hand side, the DACs will not be latched by the interrupt, but will be latched when they are written to. If the jumper is on the left, the DACs' output will change when the periodic interrupt occurs (when timer counter 0 times out, please see section 5.3.3 for more information).

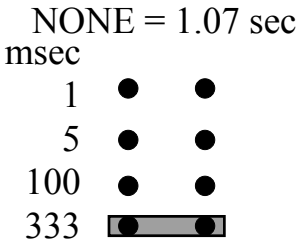
## 2.5 Watchdog - Jumper J5

The board contains a timer so that a "watchdog" will be constantly watching writes to the card. If no writes occur within a specified time, the watchdog will time-out, indicating that something has gone wrong, for example, the processor may not be responding to interrupts as it should. When the watchdog times, out the output of the DACs are set to zero. This jumper determines if the watchdog is disabled, latched or unlatched. The watchdog output is pin 46 of P2, which is low true, i.e. it is normally high, but goes low when the watchdog times out. When the watchdog is disabled, the output is always high. In the unlatched mode, the output will go low when the watchdog times out, but will return high if there is a subsequent write to the DACs. In the latched mode, the output will go low when the watchdog times out, but will remain low, until reset by a system reset, or by writing a 0 to the WDTOUT bit in the CNTRL1 register. When reading CNTRL1 the WDTOUT bit is a latched watchdog output, no matter how jumper J5 is configured.




## 2.6 Watchdog time-out selection - Jumper J4

This jumper selects the amount of time, before the watchdog will time-out. This event can be used to trip a relay in order to take some external action such as applying brakes to the machine that is being controlled.

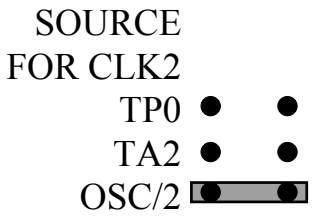



There are 4 possible jumper positions, corresponding to 1, 5, 100 or 333 milliseconds. With no jumper at all, the time-out period is 1.07 second.

## 2.7 Source for CLK2 - Jumper J7

There is a general purpose timer available on the board, called Timer 2 (There are three timers in the chip we use, Timer 0 and Timer 1 are used when generating interrupts). This jumper selects the input (or clock) for Timer 2.




TP0 means the output of Timer 0; usually Timer 0 is producing interrupts and latching the encoders. So, a jumper on the TP0 position would allow you to count interrupts. TA2 selects pin 43, on connector P2 as the input to timer 2. Thus, you could count some external event. OSC/2 selects a 7.159 MHz clock (called OSC/2, because it is derived from a clock on the bus, which is divided by 2).

## 2.8 Battery Backup Connection - Connector P5

The board contains an optional battery connection on connector P5. If power is lost to the card, the encoder counters will not loose their position count and will continue to operate if a battery is connected to this connector. With battery backed-up encoder counters, the controlled machine would not need to be re-calibrated. The supply voltage must be between 7 and 10 volts. The ground connection is on the right side of the connector, with the positive voltage on the left. The battery backed-up portion of the board requires 150 milliamps of current. Of course, a battery or some other uninterruptable power supply should also supply the encoders in order for the counter circuit to have valid quadrature input signals. Connect to P5 with Molex housing 22-0103027, and Molex pins 08-50-0114.

# 3.0 Connections

Connection to the outside world is accomplished through four 50-pin connectors and one 2-pin connector. The following tables lists the pin-out for each of the 50-pin connectors. Connection can be made with standard .1 inch spacing, 50-pin, IDC ribbon cable headers.

| Connector Name | Location* | Pin Count | Description |
|---|---|---|---|
| P1 | Furthest right, has a bracket. | 50 | 24 bits of digital I/O (Ports A, B, and C) which are Opto-22 compatible. |
| P2 | 2nd from right | 50 | 8 bits of user I/O (Port D), 8 motor direction bits, for a total of 16 Opto-22 compatible bits, and 8 channels of analog input. |
| P3 | 3rd from right | 50 | Encoder input and analog output for Axis 0-3 |
| P4 | Left-most | 50 | Encoder input and analog output for Axis 4-7 |
| P5 | near bottom edge | 2 | Battery input, right pin is ground, left pin is +V. |

- Note: These locations are from the perspective of having the board's component side up, with the ISA bus connector at the lower right. Pin 1 of each of the 50-pin connectors is in the upper left corner of the connectors.
- Connector P1 can directly connect to I/O module boards made by OPTO-22 (we've used model G4PB24, but PB24 looks right also), Grayhill (haven't used one, but model 70GRCQ24-HL looks right, from the catalog), and Crydom (again, we haven't used it, but from the catalog, model MS-24H looks right). See also Potter and Brumfield
- Terminal blocks that convert 50 pin cable to screw terminals are available from Phoenix Contact, and Thomas & Betts.

## 8 Channel Encoder Input and Analog Output Connectors

| Connector P3, Motion I/O Axis 0-3 | | | |
|---|---|---|---|
| **Pin** | **Name** | **Pin** | **Name** |
| **1** | Analog Gnd | **2** | DAC 0 |
| **3** | Analog Gnd | **4** | Analog Gnd |
| **5** | DAC 2 | **6** | Analog Gnd |
| **7** | Analog Gnd | **8** | DAC 1 |
| **9** | Analog Gnd | **10** | Analog Gnd |
| **11** | DAC 3 | **12** | Analog Gnd |
| **13** | Gnd | **14** | A 0 + |
| **15** | A 0 - | **16** | Gnd |
| **17** | B 0 + | **18** | B 0 - |
| **19** | Gnd | **20** | I 0 + |
| **21** | I 0 - | **22** | Gnd |
| **23** | A 1 + | **24** | A 1 - |
| **25** | Gnd | **26** | B 1 + |
| **27** | B 1 - | **28** | Gnd |
| **29** | I 1 + | **30** | I 1 - |
| **31** | Gnd | **32** | A 2 + |
| **33** | A 2 - | **34** | Gnd |
| **35** | B 2 + | **36** | B 2 - |
| **37** | Gnd | **38** | I 2 + |
| **39** | I 2 - | **40** | Gnd |
| **41** | A 3 + | **42** | A 3 - |
| **43** | Gnd | **44** | B 3 + |
| **45** | B 3 - | **46** | Gnd |
| **47** | I 3 + | **48** | I 3 - |
| **49** | +5 | **50** | +5 |

| Connector P4, Motion I/O Axis 4-7 | | | |
|---|---|---|---|
| **Pin** | **Name** | **Pin** | **Name** |
| **1** | Analog Gnd | **2** | DAC 4 |
| **3** | Analog Gnd | **4** | Analog Gnd |
| **5** | DAC 6 | **6** | Analog Gnd |
| **7** | Analog Gnd | **8** | DAC 5 |
| **9** | Analog Gnd | **10** | Analog Gnd |
| **11** | DAC 7 | **12** | Analog Gnd |
| **13** | Gnd | **14** | A 4 + |
| **15** | A 4 - | **16** | Gnd |
| **17** | B 4 + | **18** | B 4 - |
| **19** | Gnd | **20** | I 4 + |
| **21** | I 4 - | **22** | Gnd |
| **23** | A 5 + | **24** | A 5 - |
| **25** | Gnd | **26** | B 5 + |
| **27** | B 5 - | **28** | Gnd |
| **29** | I 5 + | **30** | I 5 - |
| **31** | Gnd | **32** | A 6 + |
| **33** | A 6 - | **34** | Gnd |
| **35** | B 6 + | **36** | B 6 - |
| **37** | Gnd | **38** | I 6 + |
| **39** | I 6 - | **40** | Gnd |
| **41** | A 7 + | **42** | A 7 - |
| **43** | Gnd | **44** | B 7 + |
| **45** | B 7 - | **46** | Gnd |
| **47** | I 7 + | **48** | I 7 - |
| **49** | +5 | **50** | +5 |

## 32 Bit Digital I/O, 8 Channel Analog Input and Sign Bit Output Connectors

| Connector P1, Digital I/O | | | |
|---|---|---|---|
| **Pin** | **Name** | **Pin** | **Name** |
| **1** | Opto-23, C7 | **2** | Gnd |
| **3** | Opto-22, C6 | **4** | Gnd |
| **5** | Opto-21, C5 | **6** | Gnd |
| **7** | Opto-20, C4 | **8** | Gnd |
| **9** | Opto-19, C3 | **10** | Gnd |
| **11** | Opto-18, C2 | **12** | Gnd |
| **13** | Opto-17, C1 | **14** | Gnd |
| **15** | Opto-16, C0 | **16** | Gnd |
| **17** | Opto-15, B7 | **18** | Gnd |
| **19** | Opto-14, B6 | **20** | Gnd |
| **21** | Opto-13, B5 | **22** | Gnd |
| **23** | Opto-12, B4 | **24** | Gnd |
| **25** | Opto-11, B3 | **26** | Gnd |
| **27** | Opto-10, B2 | **28** | Gnd |
| **29** | Opto-9, B1 | **30** | Gnd |
| **31** | Opto-8, B0 | **32** | Gnd |
| **33** | Opto-7, A7 | **34** | Gnd |
| **35** | Opto-6, A6 | **36** | Gnd |
| **37** | Opto-5, A5 | **38** | Gnd |
| **39** | Opto-4, A4 | **40** | Gnd |
| **41** | Opto-3 A3 | **42** | Gnd |
| **43** | Opto-2, A2 | **44** | Gnd |
| **45** | Opto-1, A1 | **46** | Gnd |
| **47** | Opto-0, A0 | **48** | Gnd |
| **49** | +5V | **50** | Gnd |

| Connector P2, Analog & Digital I/O | | | |
|---|---|---|---|
| **Pin** | **Name** | **Pin** | **Name** |
| **1** | ADC Chan 0 | **2** | Analog Gnd |
| **3** | ADC Chan 1 | **4** | Analog Gnd |
| **5** | ADC Chan 2 | **6** | Analog Gnd |
| **7** | ADC Chan 3 | **8** | Analog Gnd |
| **9** | ADC Chan 4 | **10** | Analog Gnd |
| **11** | ADC Chan 5 | **12** | Analog Gnd |
| **13** | ADC Chan 6 | **14** | Analog Gnd |
| **15** | ADC Chan 7 | **16** | Analog Gnd |
| **17** | Opto-15, D7 | **18** | Gnd |
| **19** | Opto-14, D6 | **20** | Gnd |
| **21** | Opto-13, D5 | **22** | Gnd |
| **23** | Opto-12, D4 | **24** | Gnd |
| **25** | Opto-11, D3 | **26** | Gnd |
| **27** | Opto-10, D2 | **28** | Gnd |
| **29** | Opto-9, D1 | **30** | Gnd |
| **31** | Opto-8, D0 | **32** | Gnd |
| **33** | IN2 | **34** | Gnd |
| **35** | IN1 | **36** | Gnd |
| **37** | IN0 | **38** | Gnd |
| **39** | EXLATCH | **40** | Gnd |
| **41** | T2GATE | **42** | Gnd |
| **43** | TA2 | **44** | Gnd |
| **45** | /WATCHDOG | **46** | Gnd |
| **47** | NC | **48** | Gnd |
| **49** | +5V | **50** | Gnd |

Connector P1 can be directly connected to a 24 channel I/O card, such as made by OPTO-22, and other

companies.

Where, for example:

| | | |
|---|---|---|
| **Opto-23, C7** | is | the I/O Port C Bit 7, and pin 23 of an Opto-22 card. |
| **DAC 1** | is | the digital-to-analog converter channel 1. |
| **A 1 +** | is | the encoder "A" signal for channel 1. For differential input it is the more positive "A" signal, and for single-ended input it is the only "A" signal. |
| **A 1 -** | is | the more negative differential input for encoder channel "A". For single-ended mode signals, this pin must be left unconnected (NOT grounded). |
| **ADC Chan 7** | is | the analog-to-digital converter channel 7 |
| **Analog Gnd** | is | analog ground |
| **/WATCHDOG** | is | watch dog timer output |
| **TA2** | is | input to timer 2 (if selected by jumper J7) |
| **T2GATE** | is | gate for timer 2 |
| **EXLATCH** | is | external latch for encoder counters (enabled via software) |
| **IN2** | is | general purpose input 2 (see register BRDTST) |
| **Gnd** | is | digital ground. |
| **NC** | is | not connected. |
| **+5** | is | 5 volt power. |

Notes on using +5: Keep your current usage to less than 500 ma. total for the card. If your current requirements exceed this, use a separate supply. The +5 volt outputs are protected by solid state resettable fuses (Raychem polyswitches). They are reset by removing the offending load. Often, you will be using this power for one side of optical isolation circuit; don't defeat the purpose by connecting grounds together.

# 4.0 Register Descriptions

The board is accessed through the set of registers shown in the table on the next page. Note that some of the registers have different meanings depending on whether a read or a write operation is being performed.

The addresses are divided into two groups, a low 32 byte page when, and a high 32 byte page. A large address set allows faster processing of data to and from the card because extra write operations are not required to specify which data is to be accessed. With a small address space an indirect-access scheme would need to be used; with the larger space, data is accessed directly.

## Register Address Offset Definition Table

| Low Group | | |
|---|---|---|
| **Offset** | **Read** | **Write** |
| 0x00 | CNT0.D | CNT0.D |
| 0x01 | CNT1.D | CNT1.D |
| 0x02 | CNT0.C | CNT0.C |
| 0x03 | CNT1.C | CNT1.C |
| 0x04 | CNT2.D | CNT2.D |
| 0x05 | CNT3.D | CNT3.D |
| 0x06 | CNT2.C | CNT2.C |
| 0x07 | CNT3.C | CNT3.C |
| 0x08 | CNT4.D | CNT4.D |
| 0x09 | CNT5.D | CNT5.D |
| 0x0A | CNT4.C | CNT4.C |
| 0x0B | CNT5.C | CNT5.C |
| 0x0C | CNT6.D | CNT6.D |
| 0x0D | CNT7.D | CNT7.D |
| 0x0E | CNT6.C | CNT6.C |
| 0x0F | CNT7.C | CNT7.C |
| 0x10 | --- | DAC0.L* |
| 0x11 | --- | DAC0.H* |
| 0x12 | --- | DAC1.L* |
| 0x13 | --- | DAC1.H* |
| 0x14 | --- | DAC2.L* |
| 0x15 | --- | DAC2.H* |
| 0x16 | --- | DAC3.L* |
| 0x17 | --- | DAC3.H* |
| 0x18 | --- | DAC4.L* |
| 0x19 | --- | DAC4.H* |
| 0x1A | --- | DAC5.L* |
| 0x1B | --- | DAC5.H* |
| 0x1C | --- | DAC6.L* |
| 0x1D | --- | DAC6.H* |
| 0x1E | --- | DAC7.L* |
| 0x1F | --- | DAC7.H* |

| High Group | | |
|---|---|---|
| **Offset** | **Read** | **Write** |
| 0x400 | PORTA | PORTA |
| 0x401 | CNTRL0 | CNTRL0 |
| 0x402 | PORTB | PORTB |
| 0x403 | BRDTST | --- |
| 0x404 | PORTC | PORTC |
| 0x405 | PORTD | PORTD |
| 0x406 | --- | ABC_DIR |
| 0x407 | --- | D_DIR |
| 0x408 | TIMER_0 | TIMER_0 |
| 0x409 | IDLEN | IDLEN |
| 0x40A | TIMER_1 | TIMER_1 |
| 0x40B | SELDI | SELDI |
| 0x40C | TIMER_2 | TIMER_2 |
| 0x40D | IDL | IDL |
| 0x40E | --- | TMRCMD |
| 0x40F | CNTRL1 | CNTRL1 |
| 0x410 | ADC.L | START CONV |
| 0x411 | ADC.H | START CONV |
| 0x412 | | |
| 0x413 | | |
| 0x414 | | |
| 0x415 | | |
| 0x416 | | |
| 0x417 | | |
| 0x418 | | |
| 0x419 | | |
| 0x41A | | |
| 0x41B | | |
| 0x41C | | |
| 0x41D | | |
| 0x41E | | |
| 0x41F | | |

* Note: these registers (registers at offsets 0x10 to 0x1F) must be accessed as full 16 bit words, not as bytes.

The base address of the board is added to the register offset, shown above, to form the entire address of the register.

# 5.0 Operation and Register Bit Definitions

This section describes the operation and use of each of the registers defined in section 4.0.

## *5.1 Digital I/O Section*

The digital I/O section has 32 I/O lines. We group them into 8 bit units, and refer to them as Port A, B, C or D. Ports A and B can be set to either all input or all output. For Ports C and D, each half of the port can be independently set to input or output.

The signal levels and connectors are Opto-22 compatible for direct connection to industry standard isolation I/O modules. Circuit boards are available from Opto-22, Grayhill, and others, which accept a 50 pin ribbon cable, such as the Opto-22 Rack Model G4PB24. They can be connected directly to connector P1 or P2 on our Servo I/O card. These circuit boards accept I/O modules, which are typically optically isolated. The plug-in I/O modules are available for a variety of AC or DC inputs or outputs. The I/O modules can be powered from our board, if the current needed is less than ¼ Amp.

The signals come from an 82C55 integrated circuit, which will sink or source 5 ma. Each line is pulled up by a 10K resistor to +5 volts; so inputs will default to high if a line is broken or not connected.

During initialization, the direction registers, ABC_DIR and D_DIR, should be set by the software to select which lines are inputs and which are outputs, then the output bits should be initialized by writing to the port registers.

For any port which is configured as an output port, reading will return the current state of the output—what you last wrote to it. If a port is configured as input, writing to it will not do anything.

### 5.1.1 Register PORTA

Offset: 0x400.

This port can be configured as either 8 inputs or 8 outputs by using register ABC_DIR. It can be read or written.

### 5.1.2 Register PORTB

Offset: 0x402.

This port can be configured as either 8 inputs or 8 outputs by using register ABC_DIR.

### 5.1.3 Register PORTC

Offset: 0x404.

For this port, the high 4 bits and low 4 bits can be independently configured as inputs or outputs. The direction is set by register ABC_DIR. For example, the low order 4 bits can be set as inputs and the high order 4 bits as outputs.

### 5.1.4 Register PORTD

Offset: 0x405.

An 8 bit I/O port. Like port C, the high 4 bits and low 4 bits can be independently configured as inputs or outputs. The direction is set by register D_DIR.

### 5.1.5 Register ABC_DIR

Offset: 0x406

This write only register sets the direction, input or output, for Digital I/O Ports A, B and C.

Bits: | 1 | 0 | 0 | A | CH | 0 | B | CL |

where:

A = Port A direction.

B = Port B direction.

CH = Upper 4 bits of port C direction.

CL = Lower 4 bits of port C direction.

1 = Input, and 0 = Output

The other bits should be written as shown. If you want to know what the other bits do, you need to know that you are writing to the Mode Control Register of an 82C55 integrated circuit. The bit configuration shown here sets the general I/O mode. Other modes are available, but would normally not be used for isolated I/O. For more information refer to the 82C55 data sheet.

For example, if the word: 10010001 were written to this address, then port A would be configured as an input port, port B as an output port, the upper 4 bits of port C would be outputs, and the lower 4 bits of port C would be inputs.

### 5.1.6 Register D_DIR

Offset: 0x407.

This write-only register is configures the direction of the top and bottom half of Port D.

Write the byte, shown below, to set the direction of Port D. This byte actually sets the direction of three ports of an 82C55 IC. Besides Port D, the chip implements the BRDTST and CNTRL0 registers. BRDTST is an input, which is the power-on default for the 82C55, so BRDTST works even before you write this byte. CNTRL0 is an output, so you have to write this byte before CNTRL0 will work.

Bits: | 1 | 0 | 0 | 0 | DH | 0 | 1 | DL |

where:

DH = direction of high 4 bits of Port D.

DL = direction of low 4 bits of Port D

For both, 1 = Input, and 0 = Output.

Note: An 82C55 is a parallel output IC with three 8-bit ports. This board uses two 82C55s. One of these is used for Ports A, B and C; it is described in the previous section. The second, described in this section, is used for Port D, the CNTRL0 register, and the BRDTST register. This 82C55 must be initialized before you use the CNTRL0 register, therefore, the direction for port D must be set before you use CNTRL0. If you change the direction of port D later, CNTRL0 will be cleared -- instantly redirecting your interrupts. So, if you want to change the direction of Port D after you've set up INT.C and started using interrupts, you should mask interrupts and restore CNTRL0. Here is how you might do it in C:

```
#define CNTRL0          1                   // offset of CNTRL0
#define CNTRL1       0x0f                   // offset of CNTRL1
#define CNTRL1_SLAVE 0x08                   // SLAVE bit in CNTRL1
#define D_DIR        0x07                   // offset of D_DIR
   …
bySaveCntrl0 = fInP(wBaseAddress + CNTRL0); // CNTRL0 needs to be saved, because
                                            // D_DIR reinitializes the 8255 which
                                            // implements the CNTRL0 register.
byHwDir = 0x8b;                             // initialize CNTRL0 as output reg.
                                            // BRDTST to input.
                                            // sets port D, high and low, to input
if (nSwDir & STG_PORT_D_LO)                 // low nibble
    byHwDir &= ~D_LOW_DIR_BIT;
if (nSwDir & STG_PORT_D_HI)                 // high nibble
    byHwDir &= ~D_HI_DIR_BIT;

bySaveCntrl1 = fInP(wBaseAddress+CNTRL1);// save for interrupt enables

    // don't reset any latches; put in slave state;
    // disable interrupts, so the glitch in CTRL0 doesn't
    // cause an interrupt on wrong irq
fOutP(wBaseAddress + CNTRL1, 0xf0);

fOutP(wBaseAddress + D_DIR, byHwDir);    // set port D direction

    // restore CNTRL0, because it was re-initialized, which
    // lost any previous contents.
fOutP(wBaseAddress + CNTRL0, bySaveCntrl0);

    // re-enable interrupts, and restore slave state, don't
    // reset any latches. (1111xxxx)
fOutP(wBaseAddress+CNTRL1, (bySaveCntrl1 & 0x0f) | 0xf0);
```

## 5.2 Board Control Section

### 5.2.1 Register BRDTST Board Test

Offset: 0x403.

This register can be used to confirm or determine the base address of the board through software. It is a read only register. The BRDTST register bit definition is:

| SER | Q2 |Q1 | Q0 | /EOC | IN2 | IN1 | IN0 |

Where SER = Serial data sequence corresponding to Q2-Q0 according to the following table:

| When Q2, Q1, Q0 is: | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| then SER is: (0x74) | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 |

where Q2, Q1, Q0 increment by one on each successive read of this register,

/EOC is the INT pin from the A/D, which signals the end of a conversion. It's high during conversion, low otherwise.

IN0, IN1, and IN0 are general purpose input bits available on P2.

Through the use of this register and some software, a program does not need to be informed of the base address of the board by the user. The user needs only to set the base address which does not conflict with another device and then plug the card in. The software can then read the most significant four bits of all the possible locations where this register might be located in the I/O space until the above bit sequence is detected. Once the bit sequence is detected, the base address of the board is then known. This method may be dangerous, but or DOS example uses it, and we haven't heard of any problems.

For example, if a possible BRDTST register is read, and its Q2, Q1, Q0 bits equal 6, then the SER bit should be a 1. If it is not, then this possible BRDTST location is not a valid one. If the SER bit is equal to one, continue to read the location. On each read, Q2, Q1, Q0 should increment by one and the above serial data sequence should appear in bit 7 of the register. If it does, then the location is valid and the base address is known (board base address equals location tested minus 0x403).

On our Model 1 card, the SER signature was 0x75.

### 5.2.2 Register CNTRL0

Offset: 0x401.

This register is implemented in an 82C55 IC, which it shares with a couple other registers (I/O Port D, and BRDTST). The 82C55 must be initialized before CNTRL0 is usable (the port on the 82C55 must be set to output). It is initialized by the D_DIR register (section 5.1.6).

| AZ | AD2 | AD1 | AD0 | CAL | IA2 | IA1 | IA0 |

Where:

| | |
|---|---|
| AZ | 0 = Autozero, 1 = Don't Autozero. During conversions, the hardware can perform an auto-zero of the ADC before performing a start conversion. This is used to correct errors due to temperature changes. An auto-zero will add 26 clock periods (or about 15 uS) to the conversion time. |
| AD2, AD1, AD0 | Selects the input channel for the ADC. |
| CAL | /CAL for the ADC. Usually high, pulse low to start cal cycle. |
| IA2, IA1, IA0 | Selects the interrupt request number to use. The following table shows the correspondence between IRQ number and the number encoded with bits IA0-IA2. |

| If IA2, IA1, and IA0 is: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| then the IRQ number selected is: | IRQ3 | IRQ15 | IRQ7 | IRQ12 | IRQ5 | IRQ10 | IRQ9 | IRQ11 |

### 5.2.3 Register CNTRL1

Offset 0x40F.

Bits: | WDTOUT | INT-G2 | INT-T2 | INT-T0 | SLAVE | IEN-G2 | IEN-T2 | IEN-T0 |

Where:

| | |
|---|---|
| WDTOUT | Buffered watch dog timer output latched and can be cleared by writing a 0. This also allows the software to determine if a time out has occurred. |
| INT_xx | Where xx is T0, T2 or G2. The interrupt output, latched. Can be cleared by writing a 0. Writing a 1 has no effect. |
| SLAVE | Slave mode is selected when set to 0. In slave mode the interrupt feedback latches the counters, instead of TP0. IA0, IA1 and IA2 selects the input interrupt line for slave mode. |
| IEN_xx | Where xx is T0, T2 or G2. Enables interrupts for the corresponding input when high. |

Notes:

Reset initializes this port to 0. T0 is timer 0, T2 is timer 2, and G2 is timer 2 gate input from connector P2, pin 41.

### 5.2.4 Register IDLEN

Offset: 0x409.

| E7 | E6 | E5 | E4 | E3 | E2 | E1 | E0 |

**I**ndex pulse or **D**igital input **L**atch **EN**able. Writing a 0 to the corresponding bit disables the index pulse, or DIO (as selected) from latching the counter. Writing a 1 enables it.

### 5.2.5 Register SELDI

Offset: 0x40B.

| DI7 | DI6 | DI5 | DI4 | DI3 | DI2 | DI1 | DI0 |

**SEL**ect **D**igital input or **I**ndex pulse. Writing a 0 to the corresponding bit selects the index pulse, or a 1 selects EXLATCH (connector P2 pin 39) to latch the counter, if enabled by the corresponding IDLEN bit.

### 5.2.6 Register IDL

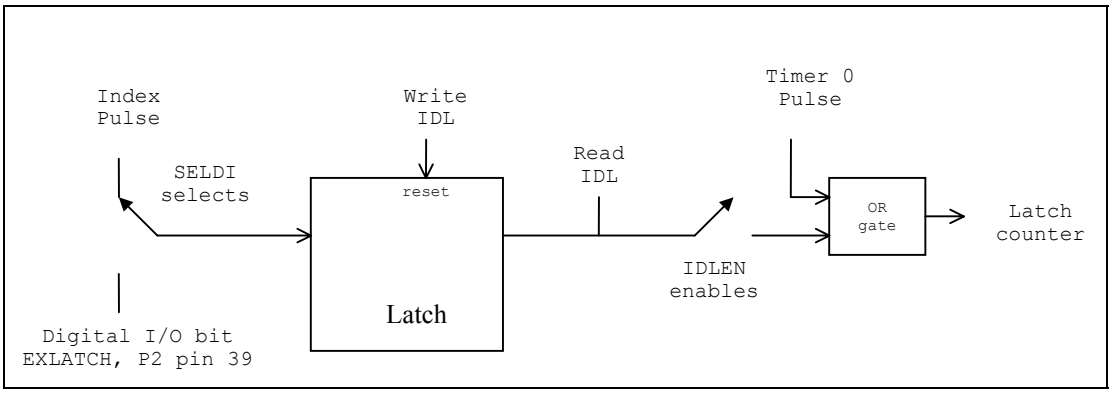
Offset: 0x40D.

| IDL7 | IDL6 | IDL5 | IDL4 | IDL3 | IDL2 | IDL1 | IDL0 |

**I**ndex pulse or **D**igital input **L**atch. Returns the current status of the IDL (before it is combined with TP0, i.e. the state of the index, or DIO latch) for the corresponding bit. This can be cleared by writing a zero the corresponding bit. Writing a one will have no effect.

### 5.2.7 Relation between SELDI, IDL and IDLEN

For each channel:



## 5.3 Timer Section

This section provides a periodic real-time interrupt. The input for this section is a 7.15909 MHz clock, divided down from the 14.31818 MHz bus oscillator. This is fed into two timers - timers 1 and 2. The output of timer 1 is then fed into timer 0. Although the timers can be setup in many different modes, the following modes are recommended for normal operation. Other modes can be found in an 82C54 data sheet.

### 5.3.1 Register TMRCMD Timer Command Register

Offset: 0x40E.

This write-only register is used to set the mode and other parameters defined further in section 5.3.2 and section 5.3.3. The timer counter values can be read from, or loaded to the associated timer register, after first writing this control word to specify the appropriate function.

The bit definitions are as follows:

| SC1 | SC0 | RL1 | RL0 | M2 | M1 | M0 | BCD |

where:

| SC1 | SC0 | Selects |
|---|---|---|
| 0 | 0 | Counter 0 |
| 0 | 1 | Counter 1 |
| 1 | 0 | Counter 2 |
| 1 | 1 | Read Back |

| RL1 | RL0 | Selects |
|---|---|---|
| 0 | 0 | Counter Latch |
| 0 | 1 | Read/Load LSB |
| 1 | 0 | Read/Load MSB |
| 1 | 1 | Read/Load LSB followed by MSB |

| M2 | M1 | M0 | Selects |
|---|---|---|---|
| 0 | 0 | 0 | Mode 0 Interrupt On Terminal Count |
| 0 | 0 | 1 | Mode 1 Programmable One-Shot |
| X | 1 | 0 | Mode 2 Real-time Interrupt |
| X | 1 | 1 | Mode 3 Square Wave Generator |
| 1 | 0 | 0 | Mode 4 Software triggered Strobe |
| 1 | 0 | 1 | Mode 5 Hardware Triggered Strobe |

| BCD | Selects |
|---|---|
| 0 | Binary - 16 Bits |
| 1 | BCD - 4 Decades |

### 5.3.2 Register TIMER_2 Timer Counter Value

Offset: 0x40C.

This timer can be used for general purpose timer/interrupt functions. The input, or clock is selected by Jumper J7. The gate is pulled high, but available on Connector P2, pin 41; you might use the gate to time an event. The timers are in an 82C54 integrated circuit.

### 5.3.3 Registers TIMER_0 and TIMER_1 Timer Counter 0 and 1

TIMER0 offset: 0x40A. TIMER1 offset: 0x408.

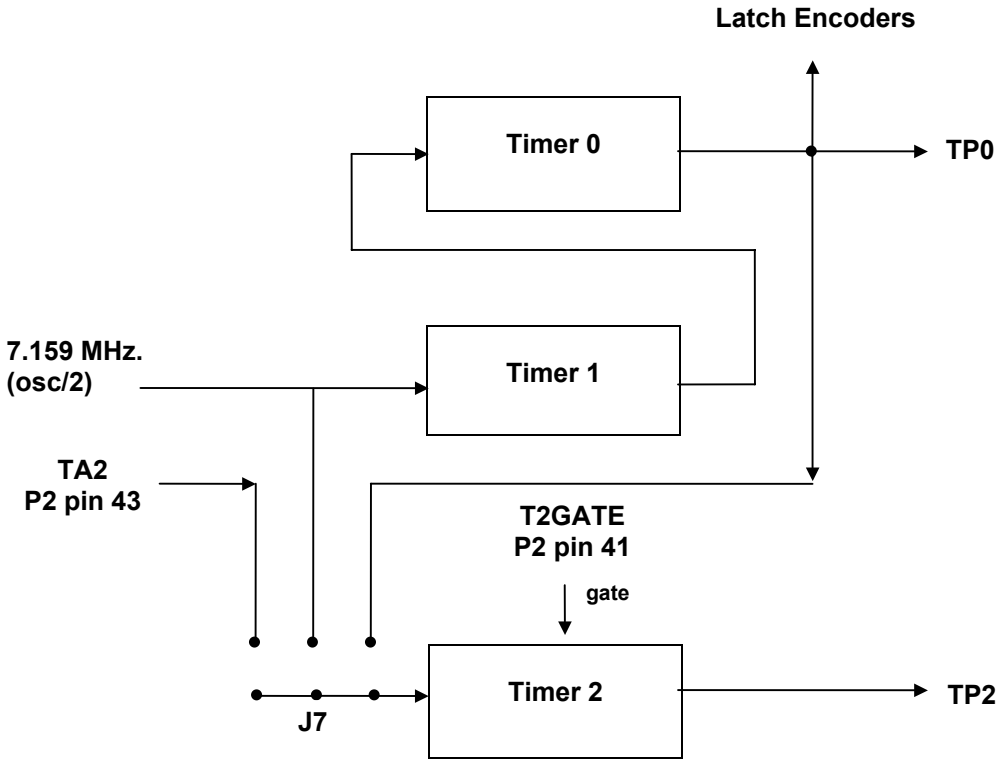
These timers are used to generate the real-time interrupt request from several microseconds up to 10 minutes. Timer 1 should be set for the square wave mode (Mode 3) with a value of 180 decimal (range: 2-65535). This will provide about a 25 microsecond time interval for the input of timer 0. Timer 0 should be set for real-time interrupts (Mode 2). Its value can then be set (range: 2-65535 ) to obtain an interval of 50 microseconds to 1.638 seconds in 25 microsecond increments. For example, if the interval for timer 1 is set to 180, then timer 0 should be set to 40 to obtain a 1mS interrupt interval.

The hardware will also latch the encoder counts into their respective latch registers on the rising edge of TIMER0's output. This saves 8 extra write operations when reading the encoders in the interrupt service routine.

The resolution of timers one and two is 139.68 nanoseconds.

The program sequence is to set the control word (TMRCMD register) then set the count value (TIMER.X register). On the next clock, loading is performed and then counting starts.

Note: these timers can be configured to generate interrupts at a rate faster than the computer can possibly service the interrupt. For example, setting the interrupt rate at 10 microseconds would probably result in the computer constantly servicing the interrupt, with no time left over for any other processing, thereby "locking up" the computer.



## 5.4 Encoder Input Section

These counters accept the quadrature encoder inputs "A" and "B". From this input stream they generate a 24 bit position count. The hardware can be configured so that the count is multiplied by 1, 2, or 4. As a convenience, the count value is automatically latched to all counters for reading by the real-time interrupt.

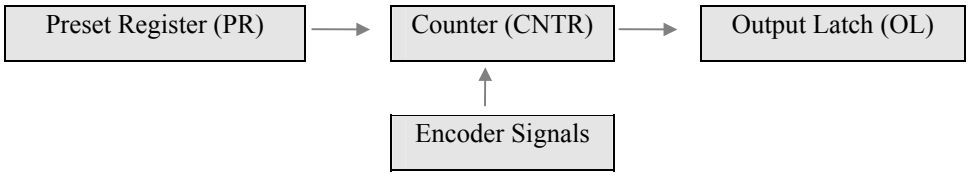
If one of the encoder inputs becomes disconnected, the counter will increment then decrement on each edge of the other pulse. This may be used to determine if the inputs are valid by detecting a count change of at least two. The maximum encoder input rate is 1.2 MHz.

The actual inputs are buffered using a RS-422 compatible receiver which accepts either differential or single ended inputs. For differential inputs, both inputs need to be connected for each channel. For single ended operation only the positive input (for example A+) is to be connected.

The counter specific registers are arranged in memory so that two registers for two axis can be read or written simultaneously by performing word wide read and write operations. Reading two encoders at a time saves a considerable amount of processing time when reading all eight encoders in the interrupt service routine. They can also be accessed individually, if desired, by performing byte wide operations.

Another time-saving measure is that the hardware is configured so that the timer-terminal-count event automatically latches all of the counters to their respective output latches. This occurs even if the timer interrupt is disabled. This saves a write operation, during the interrupt service routine, for each encoder input. If the timer is setup in mode 5, it will be effectively disabled and the timer terminal count event will not occur. If the timer itself is disabled (as opposed to the timer *interrupt*), the encoder counters can still be latched to their output latches through the use of a register command (see below).

The following diagram illustrates the operation of the counters.



### 5.4.1 Register CNTX.D Counter Data Register for Channel X

These registers are used to access the count portion of the counters, they are located in the low page. On reads, the counter register is referred to as the Output Latch, and on writes it is referred to as the Preset Register. The 24 bit data in an individual internal counter data register is accessed by a series of three 8 bit reads to the Output Latch. Two encoder output latches appear next to each other in the I/O space, therefore, with three 16 bit reads, the 24 bit counter data for each of two axis can be obtained. A pointer, internal to the counter hardware, points to one of the three bytes which is next in line to be accessed. This pointer can be reset using the RADR bit in the Master Control Register (see 5.4.2b).

**A. Output Latch (OL)**

This read-only register returns the byte of the latched count pointed to by the internal counter address pointer. The 3 byte sequence is from least significant byte to the most significant byte. As each byte is read, the internal address pointer is incremented to the next byte. The count register is copied (latched) to this register when the real-time interrupt occurs. The count register can also be latched to this register by software (see the counter master control register).

**B. Preset Register (PR)**

This write-only register is used to set the count to a specified, or preset, value. Write the byte of the preset value pointed to by the address pointer. The three byte sequence is from least significant byte to the most significant byte. This value is then used to load the counter and for comparison to the counter value.

### 5.4.2 Register CNTX.C Counter Command Register for Channel X

This register, is used to access the command portion of the counter. On reads, it is referred to as the Output Status register. On writes it is referred to as one of the following:

| Name | D7 | D6 |
|---|---|---|
| Master Control register | 0 | 0 |
| Input Control register | 0 | 1 |
| Output Control register | 1 | 0 |
| Quadrature register | 1 | 1 |

**A. Output Status Register**

This read-only register may be used to return the status information about the currently selected register.

| X | X | X | UP | SIGN | CMP | CRY | BRW |

where:

| Bit Name | Bit Description |
|---|---|
| UP | Reading a 1 means the counter is counting up, reading a 0 means it is counting down. |
| SIGN | Reading a 1 means the counter has overflowed and a 0 means it has underflowed. |
| CMP | This bit toggles every time the counter preset (PR) equals the count (CNTR). |
| CRY | This bit toggles every time the counter overflows. |
| BRW | This bit toggles every time the counter underflows |
| X | not used |

**B. Master Control Register**

This write-only register is written to the counter command register with bits 7 and 6 set to 0. It is used to clear or transfer the count value and associated flags.

| 0 | 0 | MRST | RCMP | TPR | RCNT | TOL | RADR |

where:

| Bit Name | Bit Description |
|---|---|
| MRST | Master Reset - Reset all control registers, flags, and the internal address pointer. Also, set SIGN and set Preset Register (PR) to 0xFFFFFF. |
| RCMP | Resets the Output Status Register's CMP bit. |
| TPR | Transfer Preset Register (PR) to Counter (CNTR), a 24 Bit operation. |
| RCNT | Reset CNTR, BRW, and CRY. Set SIGN. |
| TOL | Transfer Counter (CNTR) to the Output Latch (OL), a 24 Bit operation. |
| RADR | Reset the address pointer. |

**C. Input Control Register**

This write-only register is written to the counter command register with bit 7 set to 0 and bit 6 set to 1. It is used to set the counter operating mode (recommended setting is 0x68, in bold below).

| 0 | 1 | P3 | P4 | ENA/B | DCR | INC | MDE |

where:

| Bit Name | Bit Description |
|---|---|
| P3 | Pin 3 Function, **1 = OL Load**, 0 = CNTR Load |
| P4 | Pin 4 Function, 1 = Gate, **0 = Reset Counter** |
| ENA/B | Enable A/B, **1 = Enable Inputs A and B**, 0 = Disable |
| DCR | Decrement, 1 = Decrement counter once, **0 = No operation** |
| INC | Increment, 1 = Increment counter once, **0 = No operation** |
| MDE | 1 = A is clock and B is direction (0=up, 1=down), **0 = A is up count and B is down count** (overridden in quadrature mode) |

**D. Output Control Register**

This read-only register is written to the counter command register with bit 7 set to 1 and bit 6 set to 0. It is used to define the count mode and the function of output pins which are not used. The recommended setting for this register is 0x80.

**E. Quadrature Register**

This register is written to the counter command register with bit 7 and bit 6 set to 1. It is used to select and enable the quadrature mode. It has the following options (recommended setting, 0xC3, is in bold below):

0xC0 - Disable quadrature mode

0xC1 - Enable times 1 (X1) quadrature mode

0xC2 - Enable times 2 (X2) quadrature mode

0xC3 - Enable times 4 (X4) quadrature mode

## 5.5 Analog Output Section

These write-only registers provide access to the digital-to-analog converters (DACs) as 16 bit words. Accessing the registers as bytes is invalid. The registers are located at even offsets beginning at location 0x10 and ending at 0x1E. The digital-to-analog converters have a 13 bit resolution. The output voltage has a -10V to +10V range. Upon a hardware reset, and at power on, the DAC output is set to 0V by the hardware.

Below are some example DAC values:

| Value Written | Output Voltage |
|---|---|
| 0x0000 | -10V |
| 0x1000 | 0V |
| 0x1FFF | +10V |

These values are inverted, compared to the datasheet for the MAX547B because an inverting amplifier buffers the output of the MAX547B. The output buffer is a Texas Instrument's TLE2062, and will drive 20 ma.

An optional potentiometer may be installed to adjust the reference voltage for the analog output circuits. Use a 10 K potentiometer, and place it in the holes marked R24.

Also, the board can be configured so that the DACs are all latched simultaneously by the hardware at the time of the periodic interrupt. Thus, the DACs change output simultaneously, and at a known time, but you have to wait until the next interrupt to do it. Set J3 to TP0 to latch simultaneously, or set it to GND so the DACs will change output voltage immediately as they're written to (the default).

## 5.6 Analog Input Section

The 8 analog input channels are multiplexed to an Analog to Digital Converter (ADC). Resolution is 13 bits (12 bits plus sign) To read an analog channel:

1. Set the multiplexer to the desired channel, and Auto Zero as desired. These are set in the CNTRL0 register.

   *Recall from Section 5.2.2, what the CNTRL0 register looks like:*

   *| AZ | AD2 | AD1 | AD0 | CAL | IA2 | IA1 | IA0 |*

   *Where:*

| | |
|---|---|
| *AZ* | *0 = Autozero, 1 = Don't Autozero. During conversions, the hardware can perform an auto-zero of the ADC before performing a start conversion. This is used to correct errors due to temperature changes. An auto-zero will add 26 clock periods (or about 15 uS) to the conversion time.* |
| *AD2, AD1, AD0* | *Selects the input channel for the ADC.* |
| *CAL* | */CAL for the ADC. Usually high, pulse low to start cal cycle.* |
| *IA2, IA1, IA0* | *Selects the interrupt request number to use.* |

   *If you're using interrupts, be sure you don't inadvertently change the interrupt, when you write to CNTRL0.*

2. Wait settling time—4 us. Note that you can interleave steps 1 and 2 with a previous reading. The ADC has sample and hold. Once the ADC has acquired the voltage, and started the conversion, you could set the multiplexer for the next reading.
3. Start conversion, by writing to START CONV. Write a full word (16 bits), any data (such as 0) will do.
4. Wait for conversion to finish. Either wait longer than the conversion time, or poll for the EOC bit in the BRDTST register to go low. Note that EOC does not get reset (to high) until you read from the ADC.
5. Read ADC. A word read (16 bits) returns ADC counts.

| Auto-zero (AZ) Bit in Register CNTRL0 | Conversion time |
|---|---|
| 0 = Auto-zero | 34 µS |
| 1 = Don't Auto-zero | 19 µS |

The ADC counts are in 2's complement. Below are some examples:

| Counts | Input Voltage | |
|---|---|---|
| | **5 Volt range** | **10 Volt range** |
| 0x0FFF | 4.998 V | 9.9975 V |
| 0x0000 | 0 V | 0 V |
| 0x1FFF | -0.005 V | -0.0024 V |
| 0x1000 | -5.000 V | -10.000 V |

The input range is set by Jumper J2.

Auto zeroing the ADC will correct for temperature drifts. It doesn't have to be done very often, when the temperature is constant.

Calibrating the ADC: The ADC has an auto calibration cycle, which is done automatically at powerup, but should be done again after voltages have settled. Pulse CAL, in CNTRL0 low for at least 60 ns to begin the auto calibration cycle. Then wait 800 us, for calibration to complete.

Note: The ADC is a National Semiconductor ADC12441; more information can be found in the manufacturer's datasheet. Also, EOC in CNTRL0 is INT on the chip (the chip has a pin called EOC, but that's not what we use).

A 10 K potentiometer can be installed in the holes marked R11. It will adjust the voltage reference for the ADC, giving you some full scale adjustment.

# 6.0 Appendix A - IBM PC IRQs

IBM PC Hardware Interrupt Table (in order of priority)

| IRQ# | Interrupt | Function |
|---|---|---|
| | | |
| IRQ0 | 0x08 | timer (55ms intervals, 18.2 per second) |
| IRQ1 | 0x09 | keyboard service required |
| IRQ2 | 0x0A | slave 8259 or EGA/VGA vertical retrace |
| IRQ8 | 0x70 | real time clock (AT,XT286,PS50+) |
| IRQ9 | 0x 71 | software redirected to IRQ2 (AT,XT286,PS50+) |
| IRQ10 | 0x 72 | reserved (AT,XT286,PS50+) |
| IRQ11 | 0x 73 | reserved (AT,XT286,PS50+) |
| IRQ12 | 0x 74 | mouse interrupt (PS50+) |
| IRQ13 | 0x 75 | numeric coprocessor error (AT,XT286,PS50+) |
| IRQ14 | 0x 76 | fixed disk controller (AT,XT286,PS50+) |
| IRQ15 | 0x 77 | reserved (AT,XT286,PS50+) |
| IRQ3 | 0x0B | COM2 or COM4 service required, (COM3-COM8 on MCA PS/2) |
| IRQ4 | 0x0C | COM1 or COM3 service required |
| IRQ5 | 0x0D | fixed disk or data request from LPT2 |
| IRQ6 | 0x0E | floppy disk service required |
| IRQ7 | 0x0F | data request from LPT1 (unreliable on IBM mono) |

# 7.0 Appendix B - Common I/O Port Addresses

The following is a table of common I/O port address definitions in the IBM PC.

| Address | Description of Device |
|---|---|
| | |
| 000-0FF | Reserved for DMA, PIC, PIT, PPI, Keyboard, etc. |
| 0F0-0FF | Math coprocessor (AT, PS/2) |
| 100-10F | POS Programmable Option Select (PS/2) |
| 110-1EF | System I/O channel |
| 170-17F | Fixed disk 1 (AT) |
| 1F0-1FF | Fixed disk 0 (AT) |
| 200-20F | Game Adapter |
| 210-217 | Expansion Card Ports (XT) |
| 220-26F | Reserved for I/O channel |
| 270-27F | Third parallel port |
| 280-2AF | Reserved for I/O channel |
| 2A2-2A3 | MSM58321RS clock |
| 2B0-2DF | Alternate EGA, or 3270 PC video (XT, AT) |
| 2E2-2E3 | Data acquisition adapter (AT) |
| 2E8-2EF | COM4 non PS/2 UART (Reserved by IBM) |
| 2F0-2F7 | Reserved |
| 2F8-2FF | COM2 Second Asynchronous Adapter |
| 300-31F | Prototype Experimentation Card (except PCjr) |
| 320-32F | Hard Disk Controller (XT) |
| 330-33F | Reserved for XT/370 |
| 340-35F | Reserved for I/O channel |
| 360-36F | PC Network |
| 370-377 | Floppy disk controller (except PCjr) |
| 378-37F | Second Parallel Printer |
| 380-38F | Secondary Binary Synchronous Data Link Control (SDLC) adapter |
| 390-39F | Cluster Adapter |
| 3A0-3AF | Primary Binary Synchronous Data Link Control (SDLC) adapter |
| 3B0-3BF | Monochrome Display Adapter (write only) |
| 3C0-3CF | EGA/VGA |
| 3D0-3DF | Color Graphics Monitor Adapter (ports 3D0-3DB are write only) |
| 3E8-3EF | COM3 non PS/2 UART (Reserved by IBM) |
| 3F0-3F7 | Floppy disk controller (except PCjr) |
| 3F8-3FF | COM1 Primary Asynchronous Adapter |

Note:

Port addresses are not always constant across PC, AT and PS/2

Many cards designed for the ISA BUS use only the lower 10 bits of the port address but some ISA adapters use addresses beyond 0x3FF. Any address that matches in the lower 10 bits will decode to the same card. It is up to the adapters to resolve or ignore the high bits of the port addresses. An example would be the Cluster adapter that has a port address of 390h. The second cluster adapter has a port address of 790h which resolves to the same port address with the cards determining which one actually gets the data.

## 8.0 Appendix C - Integrated Circuits Used

The following integrated circuits, or their equivalents, are used to implement some of the functionality of the board:

| Function | IC Designation | Manufacturer | Description |
|---|---|---|---|
| Digital I/O | 82C55 | NEC (www.nec.com) | Programmable peripheral interface |
| Timers | 82C54 | Intel (www.intel.com) | Programmable interval timer |
| Analog input | ADC1241 or 12441 | National (www.national.com) | 13-bit ADC |
| Analog output | Max547 | Maxim (www.maxim-ic.com) | Octal, 13-bit DAC |
| Analog output driver | TLE2062 | Texas Instruments (www.ti.com | Operational Amplifier |
| Encoder Counter | LS7166 | US Digital (www.usdigital.com) | 24 bit counter |
| Encoder Input | 26LS32 | Texas Instruments (www.ti.com) | RS-422 receiver |