

# ***DM7420***

*Analog Data Acquisition Board for PC/104plus systems*

*RTD USA BME Laboratories*

*Hardware Manual*



## 1. Overview

The DM7420 boards turn your PC/104plus system (with PCI bus) into a high-speed, high-performance data acquisition system with these features.

- 8 differential, 16 single-ended analog input channels,
- 12-bit, 600 kHz throughput analog-to-digital converter ,
- Programmable input ranges: +/-5, +/-10, or 0 to +10 volts,
- Programmable gains of 1, 2, 4, 8, 16 & 32,
- 1024 x 24 channel-gain scan memory with skip bit,
- Software, pacer clock and external trigger modes,
- Scan, burst and multiburst using the channel-gain table,
- 16-bit programmable high speed sample counter and 16-bit delay counter,
- 1024 sample A/D buffer for gap-free high speed sampling under Windows™ and DOS
- Pre-, post- and about-trigger modes,
- 8-bit High-Speed Digital Input with 1K FIFO,
- 8 bit programmable digital I/O lines with Advanced Digital Interrupt modes,
- 8 bit port programmable digital I/O lines,
- Nine 16-bit timer/counters (three software configurable available to user) and on-board 8 MHz clock,
- Built-in interrupt priority controller for several simultaneous interrupt source handling,
- PCI target Interface with 16MSample/s Burst mode data transfer

The following paragraphs briefly describe the major functions of the DM7420. A detailed discussion of board functions is included in subsequent chapters. The Figure 1.1 shows the simplified block diagram of the board.



multiburst modes are supported by using the channel-gain scan memory. A first in, first out (FIFO) sample buffer helps your computer manage the high throughput rate of the A/D converter by acting as an elastic storage bin for the converted data. Even if the computer does not read the data as fast as conversions are performed, conversions can continue until the FIFO is full.

The converted data can be transferred using the programmed I/O mode or the interrupt mode.

## **1.2. 8254 Timer/Counters**

Three 8254 programmable interval timers provide nine (three each) 16-bit, 8 MHz timer/counters to support a wide range of board operations and user timing and counting functions. Six of the 16-bit timer/counters are used for board operation. Two are used for the pacer clock, one is used for the burst clock, one is used for the A/D sample counter, one is used for the A/D delay counter and one is used for the A/D about counter. The three remaining timer/counters are available for user functions.

## **1.4. Digital I/O**

The DM7420 has 16 buffered TTL/CMOS digital I/O lines with eight independent, bit programmable lines at Port 0, and an 8-bit programmable port at Port 1. The bit programmable lines support RTD's two Advanced Digital Interrupt modes. An interrupt can be generated when any bit changes value (event interrupt), or when the lines match a programmed value (match interrupt). For either mode, masking can be used to monitor selected lines. These lines are pulled up by 10KOhm resistors.

## **1.5. High-Speed Digital Input**

The DM7420 has 8 TTL/CMOS high-speed digital input lines with a 1KW FIFO buffer. These lines are shared with the Digital I/O P0 port and are pulled up by 10 KOhm resistors.

The sampling signal can be selected through software, the FIFO status can be monitored, and the number of samples can be counted by the User TC1.

## **1.6. What Comes With Your Board**

You receive the following items in your board package:

- DM7420 DAQ board with mounting hardware
- Example programs
- User's manual

If any item is missing or damaged, please call Real Time Devices' Customer Service Department at (814) 234-8087. If you require service outside the U.S., contact your local distributor.

## **1.7. Board Accessories**

In addition to the items included in your DM7420 package, Real Time Devices offers a full line of software and hardware accessories. Call your local distributor or our main office for more information about these accessories and for help in choosing the best items to support your board's application.

## **1.8. Hardware Accessories**

Hardware accessories for the DM7420 include the TB68 terminal board for easy signal connection and XD68 flat ribbon cable assembly for external interfacing.

## **1.9. Using This Manual**

This manual is intended to help you install your new board and get it running quickly, while also providing enough detail about the board and its functions so that you can enjoy maximum use of its features even in the most complex applications. We assume that you already have an understanding of data acquisition principles and that you can customize the example software or write your own application programs.

## **1.10. When You Need Help**

This manual and the example programs in the software package included with your board provide enough information to properly use all of the board's features. If you have any problems installing or using this board, contact our Technical Support Department, (814) 234-8087, during regular business hours, eastern standard time or eastern daylight time, or send a FAX requesting assistance to (814) 234-5218. When sending a FAX request, please include your company's name and address, your name, your telephone number, and a brief description of the problem. You can also contact us through our E-mail address [techsupport@rtdusa.com](mailto:techsupport@rtdusa.com).

## **2. Board Installation**

The DM7420 is easy to install in your PC/104plus stack. This chapter tells you step-by-step how to install and connect the board.

After you have installed the board and made all of your connections, you can turn your system on and run the board diagnostics program included on your example software disk to verify that your board is working.

### **2.1. Board Installation**

Keep the board in its anti-static bag until you are ready to install it in your computer. When removing it from the bag, hold the board at the edges and do not touch the components or connectors.

## 2.2. External I/O Connections

AIN 9 / AIN1-	2	1	AIN 1 / AIN1+
AIN 10 / AIN2-	4	3	AIN 2 / AIN2+
AIN 11 / AIN3-	6	5	AIN 3 / AIN3+
AIN 12 / AIN4-	8	7	AIN 4 / AIN4+
AGND	10	9	AINSENSE
AIN 13 / AIN5-	12	11	AIN 5 / AIN5+
AIN 14 / AIN6-	14	13	AIN 6 / AIN6+
AIN 15 / AIN7-	16	15	AIN 7 / AIN7+
AIN 16 / AIN8-	18	17	AIN 8 / AIN8+
AGND	20	19	AGND
Reserved	22	21	Reserved
Reserved	24	23	Reserved
AGND	26	25	Reserved
AGND	28	27	AGND
Reserved	30	29	Reserved
P1.7 / DIG TABLE 7	32	31	HIGH SPEED INPUT 7 / P0.7
P1.6 / DIG TABLE 6	34	33	HIGH SPEED INPUT 6 / P0.6
P1.5 / DIG TABLE 5	36	35	HIGH SPEED INPUT 5 / P0.5
P1.4 / DIG TABLE 4	38	37	HIGH SPEED INPUT 4 / P0.4
P1.3 / DIG TABLE 3	40	39	HIGH SPEED INPUT 3 / P0.3
P1.2 / DIG TABLE 2	42	41	HIGH SPEED INPUT 2 / P0.2
P1.1 / DIG TABLE 1	44	43	HIGH SPEED INPUT 1 / P0.1
P1.0 / DIG TABLE 0	46	45	HIGH SPEED INPUT 0 / P0.0
DGND	48	47	TRIGGER INPUT
RESET	50	49	EXTERNAL PACER CLOCK INPUT
DGND	52	51	EXTERNAL INTERRUPT INPUT
USER INPUT 1	54	53	USER INPUT 0
USER OUTPUT 1	56	55	USER OUTPUT 0
DGND	58	57	TC OUT 0
EXTERNAL TC GATE 1	60	59	EXTERNAL TC CLOCK 1
TC OUT 2	62	61	TC OUT 1
EXTERNAL TC GATE 2	64	63	EXTERNAL TC CLOCK 2
DGND	66	65	+5 VOLTS
DGND	68	67	+5 VOLTS

Table 2.2.1

AINx / AINx+ / AINx-	Signal Type	SE Analog input high sides / DIFF analog input high sides / DIFF analog inputs low sides.
AGND	Analog	Analog ground.
AINSENSE	Analog Input	Reference Signal in Non ground referenced Single Ended (NRSE) input mode.
Reserved	-	Not connected pins.
HIGH SPEED INPUT x / P0.x / Digital Input Data Markers	Digital Input / Output	High-speed inputs to digital input FIFO / Bit programmable P0 lines from digital I/O Chip.
P1.x / DIG TABLE x	Digital Input / Output	Port programmable lines from digital I/O Chip. Outputs from digital part of channel gain table.
DGND	Digital	Digital ground.
TRIGGER INPUT	Digital Input	External trigger input to trigger A/D pacer clock. (LS TTL)
EXTERNAL PACER CLOCK INPUT	Digital Input	External pacer clock to clock A/D. (LS TTL)
EXTERNAL INTERRUPT INPUT	Digital Input	Programmable rising or falling edge external Interrupt source. (LS TTL)
USER INPUT x	Digital Input	User Input 0 and User Input 1 can be read by the LAS0+04h I/O read instruction. . (LS TTL)
USER OUTPUT x	Digital Output	The source of these buffered lines can be programmed. (LS TTL)
TC OUT x	Digital Output	Buffered outputs from the user timer/counters. (LS TTL)
EXTERNAL TC CLOCK x	Digital Input	External clock signals that go to the software programmable clock source select circuit for the user timer/counters. (LS TTL)
EXT GATE x	Digital Input	External gate signals that go to the software programmable clock source select circuit for the user timer/counters. (LS TTL)
RESET	Digital Output	Active low reset output line asserted when the host PC is in hardware reset, or the Board Clear Command is active. (LS TTL)
+5 VOLTS	Power	+5 Volts from the computer power supply to power front-end boards. (Max. 2A)

Table 2.2.2 - The I/O Connector Signal Description

## 2.3. Connecting the Analog Input Pins

The DM7420 provides flexible input connection capabilities to accommodate a wide range of sensors. You can mix several *input* modes:

- Ground Referenced Single-Ended (GRSE),
- Non Referenced Single-Ended (NRSE),
- Differential (DIFF).

All of the three modes are software selectable. For differential mode operation, it is recommended that you connect a 10 kOhm resistor from the negative input to Analog ground.

The following figures show a simplified block diagram of the analog input section of the board. The NRSEH, ADCDIFFH, INSTGNDH, AINSENSEH are logic state variables for controlling the analog input operation.

### 2.3.1. Ground Referenced Single-Ended (GRSE) input mode

*This mode is suggested only for floating signal sources to avoid ground loops.* To configure the GRSE analog input, connect the high side of the input signal to the selected analog input channel, AIN1 through AIN16, and connect the low side to any of the ANALOG GND pins available at the connector.

In figure 2.3.2 you can see the switch states of this mode. The NRSEH bit is low, which means that this is not NRSE mode. ADCDIFFH bit is low because this is not a differential mode. The INSTGNDH bit is high controlling the connection of the low side of the Instrumentation Amplifier to Analog Ground (AGND). The AINSENSEH bit is low because the reference signal of the Instrumentation Amplifier is Analog Ground.

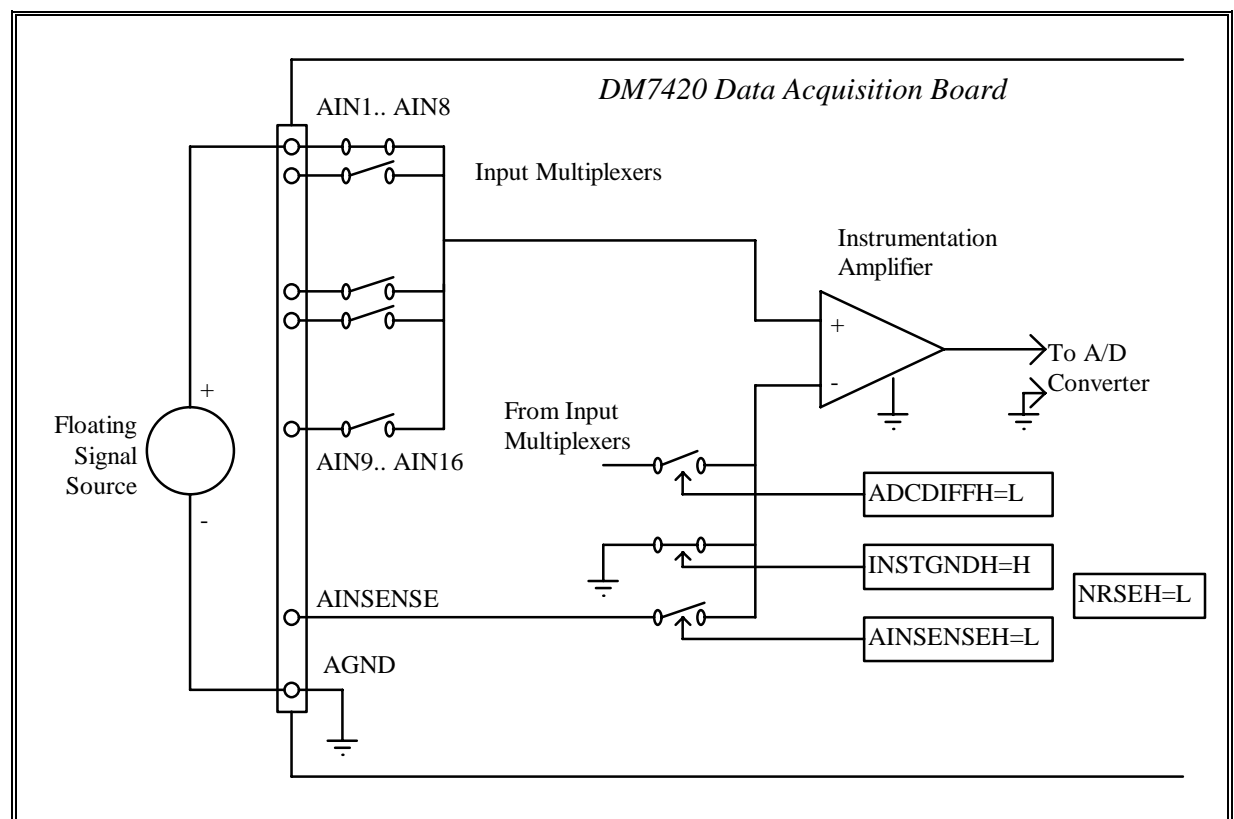


Figure. 2.3.2 Ground Referenced Single Ended input mode

### 2.3.2. Non Referenced Single-Ended (NRSE) input mode

This mode can be used for both grounded signal sources (Figure 2.3.3) and for floating signal sources. For floating signal sources an external 10 kOhm resistor should be connected between the AINSENSE pin and Analog ground. To configure the NRSE analog input, connect the high side of the input signal to the selected analog input channel, AIN1 through AIN16, and connect the low side to the AINSENSE pin at the connector.

In figure 2.3.3 you can see the switch states of this mode. The NRSEH bit is high, which means that this is the NRSE mode. ADCDIFFH bit is low because this is not a differential mode. The AINSENSEH bit is high controlling the connection of the low side of the Instrumentation Amplifier to the AINSENSE signal. The INSTGNDH bit is low because the reference signal of the Instrumentation Amplifier is the AINSENSE signal.

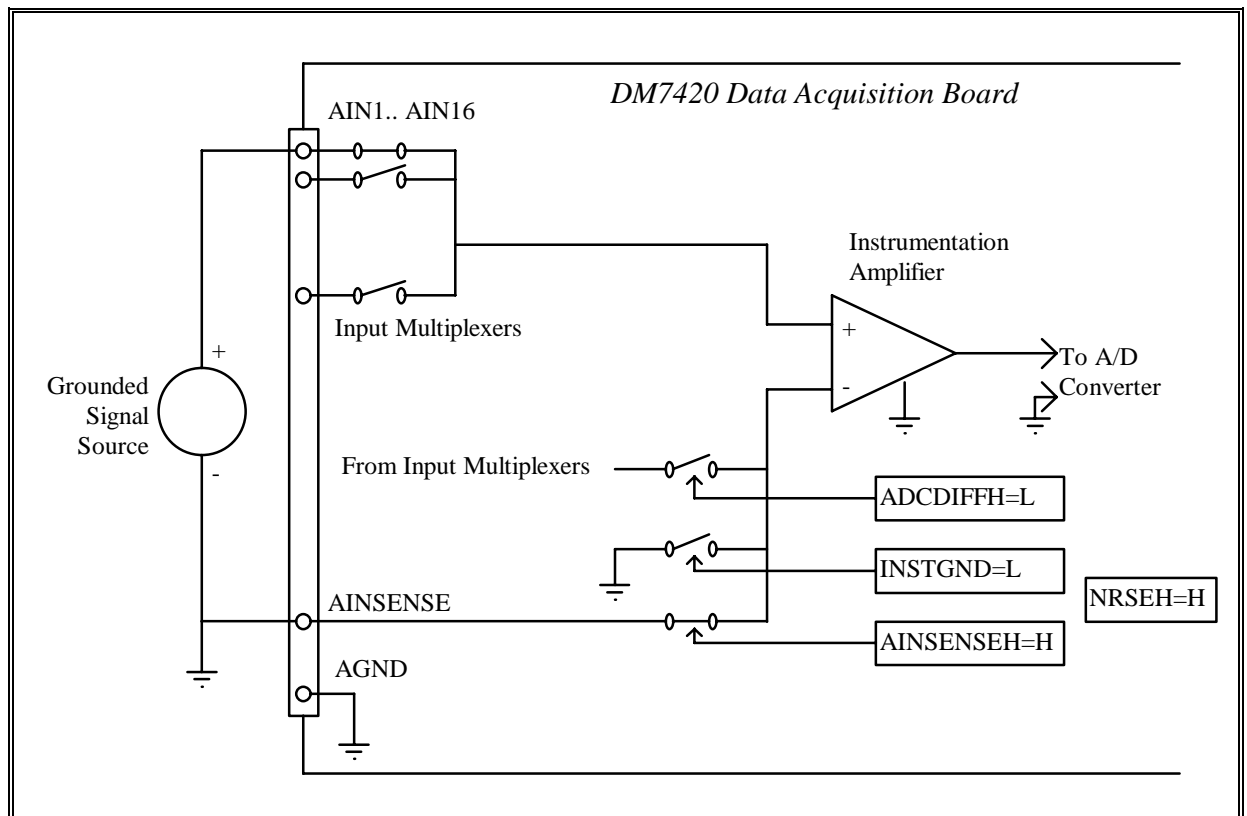


Figure. 2.3.3 Non Referenced Single-Ended input mode

### 2.3.3. Differential (DIFF) input mode

For differential inputs, your signal source may or may not have a separate ground reference. Connect the high side of the analog input to the selected analog input channel, AIN1+ through AIN8+, and connect the low side to the corresponding AIN- pin. For differential signal sources that do not have a separate ground connection, a 10 kOhm resistor should be connected between the AIN- pin and Analog Ground.

In the Figure 2.3.4 you can see the switch states of this mode. The state of the NRSEH bit does not matter in DIFF mode. The ADCDIFFH bit is high controlling the connection of the low side of the Instrumentation Amplifier to the AIN- signal. The INSTGNDH bit is low because the reference signal of the Instrumentation Amplifier is the AIN- signal. The AINSENSEH bit is low because the reference signal of the Instrumentation Amplifier is the AIN- signal in DIFF mode.

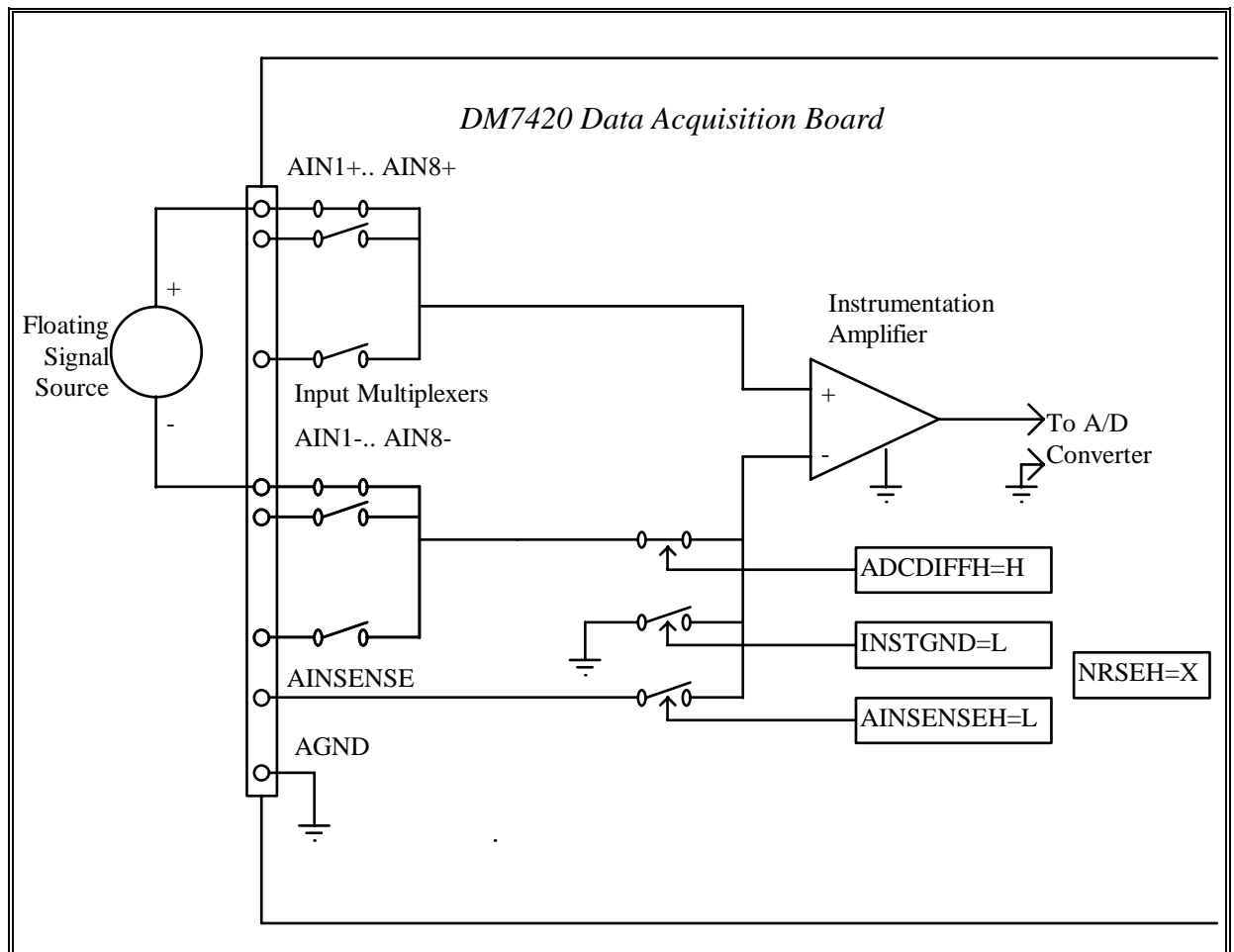


Figure 2.3.4. Differential input mode

## 2.4. Connecting the Timer/Counters and Digital I/O

For all of these connections, the high side of an external signal source or destination device is connected to the appropriate signal pin on the I/O connector, and the low side is connected to any DIGITAL GND.

All input lines have a 10 kOhm pull-up resistor. All output lines have a 10 Ohm series resistor.

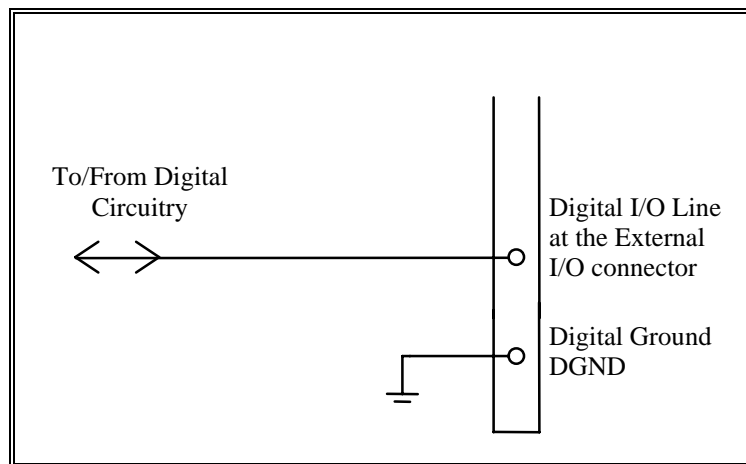


Figure 2.5.1.

## 2.5. RSW1 – PCI clock selection switch

The rotary switch (RSW1) near the bus PC/104 plus connector is used to select the proper PCI bus clock for your PC/104 plus add-in card. The switch setting depends on where the add-in card is stacked in relation to the CPU card. If the add-in card is stacked next to the CPU, the switch should be set for 0. If there is one card between the DM7420 and the CPU, the switch should be set for 1. If there are two cards between the DM7420 and the CPU, the switch should be set for 2. If there are three cards between the DM7420 and the CPU, the switch should be set to 3. You cannot have more than 4 PC/104 plus add-in cards stacked on the CPU.

## 2.6. Running the Diagnostics Program

Now that your board is ready to use, you will want to try it out. An easy-to-use diagnostics program, 7420diag.exe, is included with your example software to help verify the board's operation.



### 3. Hardware Description

This chapter describes the features of the DM7420 hardware. The major circuits are the A/D, the timer/counters, and the digital I/O lines. This chapter describes the hardware, which makes up these major circuits.

#### 3.1. The Operation of the Analog Input and High-Speed Digital Input Section

##### 3.1.1. Overall Description

The Figure 3.1.1 shows the structure of the Analog and High-Speed Digital Input Section of the Board. The Board has 16 Se. (ground referenced or non-ground referenced) or 8 Diff. inputs which are multiplexed. The input voltage range is software programmable for -5 to +5 volts, -10 to +10 volts, or 0 to +10 volts. Overvoltage protection to  $\pm 12$  volts is provided at the inputs. The multiplexed signal can be amplified with programmable gain. Software programmable binary gains of 1, 2, 4, 8, 16 and 32 let you amplify lower level signals to more closely match the board's input ranges. The multiplexed and amplified analog signal is converted by the A/D converter. The converted data is written to the 1KW FIFO. The High-Speed Digital Input lines can be simultaneously sampled with the analog lines. This mode is the Input Data Marker Mode, which can be used to sample digital data synchronized with analog signals. The Sampling Signal for the High-Speed Digital Inputs can be programmed to several sources. The sampled digital data is written to the High-Speed Digital Input FIFO. The channel type, the channel gain and other control bits can come from the channel gain latch or from the channel gain table. The Channel Gain Latch is used in single-channel operation mode, and the Channel Gain Table can be used in multi-channel operation mode.

##### 3.1.2 Channel Gain Latch (CGL) and Channel Gain Table (CGT)

In the case of single-channel operation the Channel Gain Latch mode must be set by appropriate software instruction. Then the Channel Gain Latch must be written by appropriate software instruction. This mode assures the highest sampling rate at the highest accuracy. This mode can be used for *analog trigger* function. You can use one of the input channels as *Analog Trigger Input*. Set the Channel input type, the number and gain according to the signal source from software. Reading the converted data from the input channel the analog trigger event can be detected. When the trigger event has been detected, the multichannel -Channel Gain Table mode can be started.

The Channel-Gain Table lets you sample channels in any order, at high speeds, with a different gain on each channel. This 1024 x 24-bit memory supports complex channel-gain scan sequences, including digital output control. Using the digital output control feature, you can control external input expansion boards such as the TMX32 to expand channel capacity to up to 512 channels. When used, these control lines are output on Port 1. When the digital lines are not used for this feature, they are available for other digital control functions.

A skip bit is provided in the channel-gain data word to support different sampling rates on different channels. When this bit is set, an A/D conversion is performed on the selected channel but not stored in the FOFO.

In the case of multichannel operation the Channel Gain Table must be enabled by appropriate software instruction. Then the Channel Gain Table must be cleared and filled with the appropriate entries by the appropriate software instruction. After this setup the read pointer of the Channel Gain Table points to the first entry. The first A/D conversion works according to the first entry of CGT. After an active Conversion Signal (See 2.3 The A/D Conversion Signal) the A/D Converter asserts the End of Conversion Signal. This signal increases the read pointer of the Channel Gain Table and writes the converted data to the A/D FIFO and the sampled High-Speed Digital Input lines to the FIFO if the High Speed Digital Input is in Data Marker Mode. The next conversion works according to the second entry of CGT etc. After reading the last entry, the read pointer automatically returns to the first entry of the CGT. This returning can be activated by Reset Channel Gain Table software instruction.

The Channel Gain Table assures the possibility of independent programming of the channel type (GRSE, NRSE or DIFF), the channel gain (1..128) and the input range (+-5V, +-10V or 0..10V). Therefore CGT assures the possibility of simultaneous update the D/A1 and D/A2 with the appropriate input channels. These functions can be reached via the bits CGT entries.

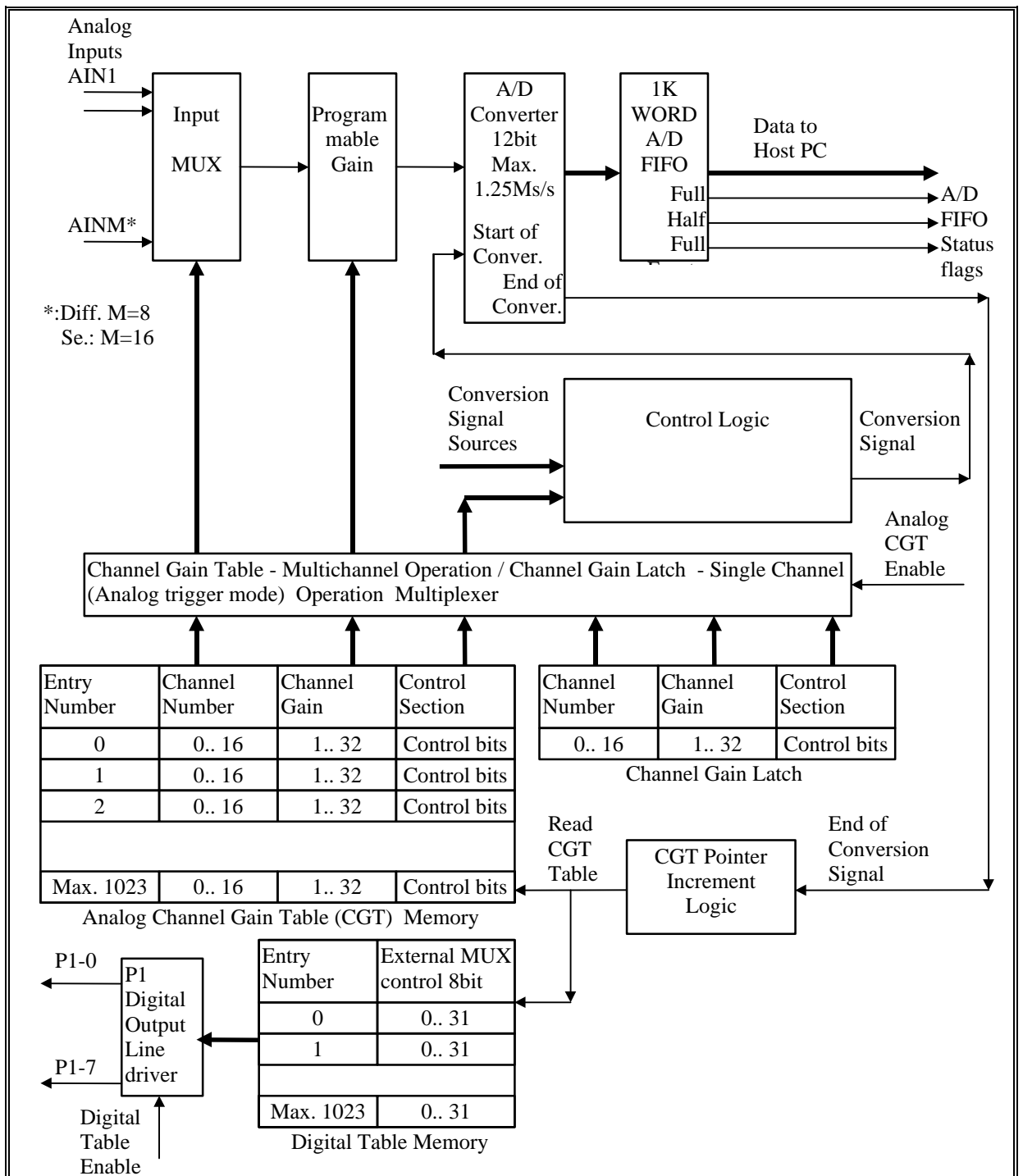


Figure 3.1.1.

### 3.1.3. A/D Converter

The 12-bit successive approximation A/D converter accurately digitizes dynamic input voltages in 0.8 microseconds, for a maximum throughput rate of 600 kHz. The converter IC contains a

sample-and-hold amplifier, a 12-bit A/D converter, a 2.5-volt reference, a clock, and a digital interface to provide a complete A/D conversion function on a single chip. Its low power CMOS logic combined with a high precision, low noise design gives you accurate results.

Conversions are controlled by software command, by pacer clock, by using triggers to start and stop sampling, or by the sample counter to acquire a specified number of samples. An on-board or external pacer clock can be used to control the conversion rate. Conversion modes are described in Chapter 5, A/D Conversions.

### 3.1.4. A/D FIFO - Sample Buffer

A first in, first out (FIFO) 1024 sample buffer helps your computer manage the high throughput rate of the A/D converter by providing an elastic storage bin for the converted data. Even if the computer does not read the data as fast as conversions are performed, conversions will continue until a FIFO full flag is sent to stop the converter.

The sample buffer does not need to be addressed when you are writing to or reading from it; internal addressing makes sure that the data is properly stored and retrieved. All data accumulated in the sample buffer is stored intact until the PC is able to complete the data transfer. Its asynchronous operation means that data can be written to or read from it at any time, at any rate. When a transfer does begin, the data first placed in the FIFO is the first data out.

### 3.1.5. Data Transfer

The converted data can be transferred to PC memory in one of two ways. Data can be transferred using the programmed I/O mode or the interrupt mode.

### 3.1.6. High-Speed Digital Input section

The Figure 3.1.2 shows the block diagram of High-Speed Digital Input section. The sampling signal is software selectable.

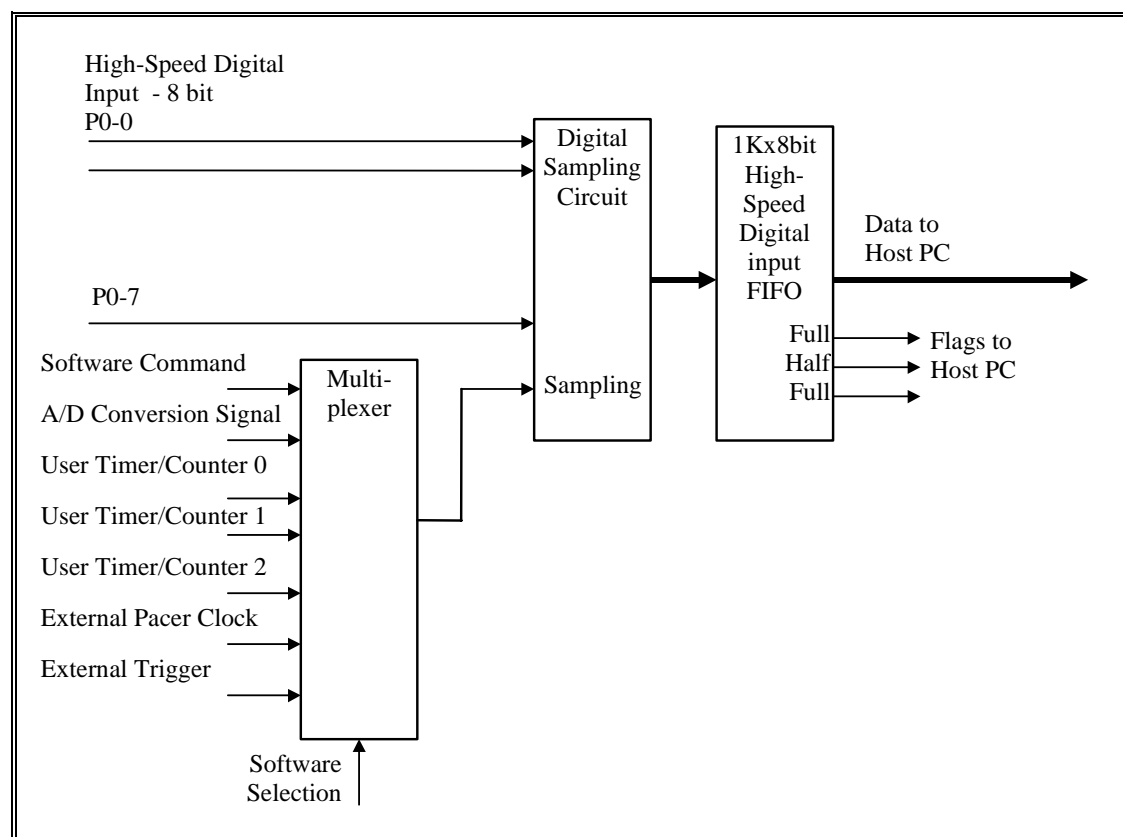


Figure 3.1.2.

The Sampled data is written automatically to the High-Speed Digital Input FIFO. Data can be transferred to PC memory in one of two ways. Data can be transferred using the programmed I/O mode or the interrupt mode. The number of samples in the High-Speed Digital Input FIFO can be counted by User TC1. User TC1 can be an interrupt Source.

### 3.2. Timer/ Counters

Three 8254 programmable interval timers provide nine 16-bit, 8-MHz timer/counters to support a wide range of timing and counting functions.

The 8254 at U23 is the Clock TC. Two of its 16-bit timer/counters, Counter 0 and Counter 1, are cascaded and reserved for the *Pacer Clock*. The pacer clock is described in Chapter 5. The third timer/counter in the Clock TC, Counter 2, is the *Burst Clock*.

The 8254 at U25 is the Sample Counter TC. Counter 2 is the *A/D sample counter*, Counter 1 is the *About Counter*, and Counter 0 is the *Delay Counter*.

The 8254 at U26 is the User TC. All three counters on this chip are available for user functions.

Each 16-bit timer/counter has two inputs, CLK in and GATE in, and one output, timer/counter OUT. The sources of User TC clock and gate inputs can be programmed. (See Chapter4.) Each TC can be programmed as binary or BCD down counters by writing the appropriate data to the command word, as described in Chapter 4. The command word also lets you set up the mode of operation. The six programmable modes are:

- Mode 0 Event Counter (Interrupt on Terminal Count)
- Mode 1 Hardware-Retriggerable One-Shot
- Mode 2 Rate Generator
- Mode 3 Square Wave Mode
- Mode 4 Software-Triggered Strobe
- Mode 5 Hardware Triggered Strobe (Retriggerable)

These modes are detailed in the 8254 Data Sheet, reprinted from Intel in Appendix C.

### 3.3. Digital I/O

The 16 digital I/O lines can be used to transfer data between the computer and external devices. Eight lines are bit programmable and eight lines are byte, or port, programmable.

Port 0 provides eight bit programmable lines which can be independently set for input or output. These ports support RTD's two Advanced Digital Interrupt modes. An interrupt can be generated when the lines match a programmed value or when any bit changes its current state. A Mask Register lets you monitor selected lines for interrupt generation.

Port 1 can be programmed as an 8-bit input or output port.

Chapter 10 details digital I/O operations and Chapter 7 explains digital interrupts.

## 4. Configuration Registers and Register Address Spaces of DM7420

The DM7420 is a PCI bus board with a PCI bus target interface. The board has two configuration register areas and three operation register areas.

The configuration Registers are the PCI Configuration Register and the Local Configuration Register.

The PCI Configuration Registers are automatically filled by the hardware after power up.

The Local Configuration Registers control the operation of the local bus on the board. This register area is filled automatically by the hardware after power up.

The PCI Configuration Registers and the Local Configuration Registers are filled from an EEPROM on the board.

The most interesting areas for the user are the register address spaces of the board. There are three register address spaces, the Local Address Space.. 2 (LAS0, LAS1 and LAS2). These spaces can be accessed by I/O instructions. The base addresses of these spaces can be read from the PCI configuration area.

LAS0 is a 32-byte long 16-bit wide register area, which contains the Function Select/Argument registers, status, command registers, and the registers of built-in priority interrupt controller. The Function Select and Argument Registers are the tools for setting up the board.

LAS1 is an 8-byte long 16-bit wide register area for transferring data to/from the board.

LAS2 is a 32-byte long 8-bit wide register area for the Timer/Counter chips and the Digital I/O Chip.

### 4.1. PCI Configuration Registers

The PCI configuration registers are shown in Table 4.1. The contents of the PCI configuration registers are valid after booting up the system and can be accessed by PCI BIOS calls. The description of the PCI configuration registers are shown in Table 4.2.

Configuration Address Offset	PCI Writable	Byte3	Byte2	Byte1	Byte0
00h	No	Device Identification		Vendor Identification	
04h	Yes	Status		Command	
08h	No	Class Code			Revision
0Ch	Yes (7..0)	BIST	Header Type	PCI Latency Timer	Cache Line Size
10h	Yes	PCI Base Address 0 for Memory Mapped Local Configuration Registers			
14h	Yes	PCI Base Address 1 for I/O Mapped Local Configuration Registers			
18h	Yes	PCI Base Address 2 for Local Address Space 0			
1Ch	Yes	PCI Base Address 3 for Local Address Space 1			
20h	Yes	PCI Base Address 4 for Local Address Space 2			
24h	Yes	Reserved			
28h	No	Reserved			
2Ch	No	Subsystem ID		Subsystem Vendor ID	
30h	Yes	PCI Base Address for Local Expansion ROM			
34h	No	Reserved			
38h	No	Reserved			
3Ch	Yes(7..0)	Max_Lat	Min_Gnt	Interrupt Pin	Interrupt Line

Table 4.1.

Field	Contents	Comment
Vendor Identification	1435h	Value Assigned to Real Time Devices Inc. by the PCI Special Interest Group
Device Identification	4400h	Type number of the Board
Class Code	FF0000h	No Applicable Class Code
Cache Line Size	00h	has no effect
PCI Latency Timer	00h	Not Supported
Header Type	00h	Single Function PCI Device
BIST	00h	The Built In Self Test is not Supported
PCI Base Address 0 for Memory Mapped Local Configuration Registers	Assigned by the PCI BIOS	Controls the operation of local system
PCI Base Address 1 for I/O Mapped Local Configuration Registers	Assigned by the PCI BIOS	Controls the operation of local system
PCI Base Address 2 for Local Address Space 0 (LAS0)	Assigned by the PCI BIOS	LAS0 is the base address of the configuration/setup area of DM7420
PCI Base Address 3 for Local Address Space 1 (LAS1)	Assigned by the PCI BIOS	LAS0 is the base address of the A/D, D/A, and High-Speed Digital Input data transfer area of DM7420
PCI Base Address 4 for Local Address Space 2 (LAS2)	Assigned by the PCI BIOS	LAS0 is the base address of the Timer/Counter and the Digital I/O chip of DM7420
Subsystem ID	00h	No subsystem
Subsystem Vendor ID	00h	No subsystem
PCI Base Address for Local Expansion ROM	00000000h	No external BIOS
Interrupt Line	0xh	Interrupt Line Assigned by the BIOS
Interrupt Pin	01h	INTA# Interrupt
Min_Gnt	00h	Bus mastering not supported
Max_Lat	00h	Bus mastering not supported

Table 4.2.

## 4.2. Local Address Space 0 (LAS0)

This I/O address space can be used to setup, configure and control the operation of the DM7420 board. It can be accessed by word (16 bit) wide I/O instructions. This area cannot be burst read or written. Range size is 32 bytes.

Read Function	Write Function	Local Address Space 0 Offset
-	Function Select	00h (16-bit)
-	Function Argument	02h (16-bit)
Read User Inputs	Write User Outputs	04h (16-bit)
-	-	06h (16-bit)
Read FIFO Status	Software A/D Start	08h (16-bit)
-	-	0Ah (16-bit)
-	-	0Ch (8-bit)
-	-	0Eh (16-bit)
-	-	10h (16-bit)
-	-	12h (16-bit)
Software Pacer Start	Software Pacer Stop	14h (16-bit)
Read Timer Counters Status	Software high-speed digital input sample	16h (16-bit)
Read Interrupt Status	Write Interrupt Enable Mask Register	18h (16-bit)
Clear Interrupt set by the Clear Mask	Set Interrupt Clear Mask	1Ah (16-bit)
Read Interrupt Overrun Register	Clear Interrupt Overrun Register	1Ch (16-bit)
-	-	1Eh (16-bit)

Table 4.1.1.

### 4.2.1. Local Address Space 0 + 0h: Function Select register (Write only)


#### Write operation (16-bit)

A write selects one of the following functions to be programmed via the Function Argument - Local Address Space 0 + 2 address. Simple functions without arguments also need a write to Local Address Space 0 + 2 but the data is irrelevant. See tables on the following pages.

### 4.2.2. Local Address Space 0 + 2h: Function argument register (Write only)

#### Write operation (16-bit)

A write loads the value in the selected register via the Function Select Register - Local Address Space 0 + 0. If the function selected by Local Address Space 0 + 0 has no argument, a write to this location is needed to terminate the function selection. In this case the data is irrelevant. See the various functions and arguments in the tables on the following pages.

After power up the registers of the DM7420 are in initial state. This initial state can be reached also by Software Reset. The initial state is signed by .

The usage rules of the Function Select/Argument are very simple:

1. Write a 16-bit word Function Code into the Function Select Register to select the desired function.
2. Write the appropriate 16-bit word into the Function Argument Register. If the Function Argument parameter is not needed, the data written is irrelevant.

The following Tables and texts describe the different Functions.

#### Function select register and arguments:

Function group	Function name	Function code (hex)	Function argument
Board Control	Software Reset of the board	0x000F	-

Table 4.1.2.

#### Function: Software Reset of the board - 0x000F

Software Reset of the board (See Table 4.1.1.) resets all inside logic variables of the board, equivalently with the power-up states. (☞).

#### Function: A/D Conversion Signal Select - 0x0200

Table 4.1.3 shows the A/D conversion and the High-Speed Digital Input Control Functions.

The A/D Conversion signal select Function is used to choose the A/D Sampling signal:

- 0x0 means that conversions are controlled by a write to LAS0+8h (data is irrelevant).
- 0x1 means that conversions are controlled by the internal or an external pacer clock (CN2-49).
- 0x2 means that conversions are controlled by the burst clock.
- 0x3 means that conversions are controlled by the Digital I/O chip digital interrupt output.

#### Function: Burst Clock start trigger select - 0x0201

If you want to use the burst clock as the conversion signal source, the start trigger must be set by the Burst Clock Start Trigger Select Function. The stop trigger of the burst clock is generated automatically by the CGT reset signal, which occurs at the end of the CGT cycle.

- 0x0 means that burst clock will be started by a write to LAS0+8h (data is irrelevant).
- 0x1 means that burst clock will be started by the internal or an external pacer clock (CN2-49).
- 0x2 means that burst clock will be started by the External Trigger Input (CN2-47).
- 0x3 means that burst clock will be started by the Digital I/O chip digital interrupt output.

Function group	Function name	Function code (hex)	Function argument
A/D Conversion and High Speed Digital Input Control	A/D Conversion Signal Select	0x0200	0x0 = Software A/D Start <sup>Ⓢ</sup> (WR_LAS0+8h) 0x1 = Pacer Clock (Ext. Int. see Func.509) 0x2 = Burst Clock 0x3 = Digital Interrupt
	Burst Clock start trigger select	0x0201	0x0 = Software A/D Start <sup>Ⓢ</sup> (WR_LAS0+8h) 0x1 = Pacer Clock 0x2 = External Trigger 0x3 = Digital Interrupt
	Pacer Clock start trigger select	0x0202	0x0 = Software Pacer Start <sup>Ⓢ</sup> (RD_LAS0+14h) 0x1 = External trigger 0x2 = Digital interrupt 0x3 = User TC 2 out 0x4 = Reserved 0x5 = Reserved 0x6 = Reserved 0x7 = Reserved 0x8 = Delayed Software Pacer Start 0x9 = Delayed external trigger 0xA = Delayed digital interrupt 0xB = Delayed User TC 2 out 0xC = Reserved 0xD = Reserved 0xE = Reserved 0xF = External Trigger Gated controlled mode
	Pacer Clock Stop Trigger select	0x0203	0x0 = Software Pacer Stop <sup>Ⓢ</sup> (WR_LAS0+14h) 0x1 = External Trigger 0x2 = Digital Interrupt 0x3 = About Counter 0x4 = User TC2 out 0x5 = Reserved 0x6 = Reserved 0x7 = Reserved 0x8 = About Software Pacer Stop 0x9 = About External Trigger 0xA = About Digital Interrupt 0xB = Reserved 0xC = About User TC2 out 0xD = Reserved 0xE = Reserved 0xF = Reserved

Table4.1.3a.

About Counter Stop Enable	0x0204	0 = Stop enabled 1 = Stop disabled
Pacer Start Trigger Mode select	0x0205	0x0 = Single Cycle Mode – trigger circuit needs re-armed after trigger signal 0x1 = Trigger Repeat Mode - trigger circuit does not need to be re-armed after trigger signal
Sampling Signal for High Speed Digital Input Select	0x0206	0x0 = Software (Write LAS0+16h) 0x1 = A/D Conversion Signal 0x2 = User TC out 0x3 = User TC out 1 0x4 = User TC out 2 0x5 = External Pacer Clock 0x6 = External Trigger
Clear High Speed Digital Input FIFO	0x020E	-
Clear A/D FIFO	0x020F	-

Table 4.1.3b.

**Function: Pacer Clock start trigger select - 0x0202**

If you want to use the Pacer Clock you must specify the start and stop conditions. The Pacer Clock Start Trigger Function selects the start signal of the Pacer Clock:

- 0x0 means that the pacer clock is started by a read from LAS0+14h.
- 0x1 means that the pacer clock is started by an External Trigger Input signal (CN2-47).
- 0x2 means that the pacer clock is started by a digital interrupt.
- 0x3 means that the pacer clock is started when the output of User TC Counter 2 pulses.

The following start trigger sources provide delayed triggering. When the trigger is issued, the A/D Delay Counter- U25 Counter 0, counts down and conversions are started when the A/D Delay Counter reaches 0. The A/D Delay Counter counts at the pacer clock rate.

- 0x8 means that the pacer clock is started by a read from LAS0+14h.
- 0x9 means that the pacer clock is started by an External Trigger Input signal (CN2-47).
- 0xA means that the pacer clock is started by a digital interrupt.
- 0xB means that the pacer clock is started when the output of User TC Counter 2 pulses.
- 0xF means the Gate Mode: the pacer clock runs as long as the External Trigger Input line (CN2-47) is held high or low, depending on the trigger polarity. This mode does not use a stop trigger.

**Function: Pacer Clock Stop Trigger select- 0x0203**

The Pacer Clock Stop Trigger Function selects the stop signal of the Pacer Clock:

- 0x0 means that the pacer clock is stopped by a write to LAS0+14h.
- 0x1 means that the pacer clock is stopped by an External Trigger signal (CN2-47).
- 0x2 means that the pacer clock is stopped by a digital interrupt.
- 0x3 means that the pacer clock is stopped by the About Counter. The About Counter is clocked by the A/D FIFO write signal. In this mode a desired number of samples can be acquired.
- 0x4 means that the pacer clock is stopped by User TC2 output.

The following stop trigger sources provide about triggering, where data is acquired from the time the start trigger is received, and continues for a specified number of samples *after* the stop trigger is received. The number of samples taken after the stop trigger is received is set by the About Counter, which is clocked by the A/D FIFO write signal.

- 0x8 means that the pacer clock is stopped a specified number of samples after a write to LAS0+14h.
- 0x9 means that the pacer clock is stopped a specified number of samples after an External Trigger signal (CN2-47).
- 0xA means that the pacer clock is stopped a specified number of samples after a digital interrupt occurs.
- 0xC means that the pacer clock is stopped a specified number of samples after the User TC2 output pulses.

**Function: About Counter Stop Enable - 0x0204**

When enabled (set to 0), the A/D sample counter counts down once and stops the pacer clock. When disabled (set to 1), the A/D sample counter repeats the countdown until you enable the stop bit (set this bit to 0). This can be used for sample counts greater than 65,536 (the size of the 16-bit About counter).

**Function: Pacer Start Trigger Mode select - 0x0205**

When set to single cycle, a trigger will initiate one conversion cycle and then stop, regardless of whether the trigger line is pulsed more than once; when set to repeat, a new cycle will start each time a trigger is received, and the current cycle has been completed. Triggers received while a cycle is in progress will be ignored.

**Function: Sampling Signal for High Speed Digital Input Select- 0x0206**

The sampling signal for the High-Speed Digital Inputs can be selected by this function. If you select the A/D conversion signal, the 8-bit digital input lines are simultaneously sampled with the analog signals.

**Function: Clear High Speed Digital Input FIFO- 0x020E**

This Function clears the High-Speed Digital Input FIFO.

**Function: Clear A/D FIFO- 0x020F**

This Function clears the A/D FIFO.

Table 4.1.4 shows the *Channel Gain Table Control Functions*. Figure 4.1.1 shows the bits of the Channel Gain Latch (single-channel mode) and the bits of the Channel Gain Table (multichannel mode).

**Function: Write ADC channel gain table - 0x0300**

For high-speed multi-channel operation the Channel Gain Table must be used. Before writing the channel gain table entries use the 0x30F function to clear the table.

**Using the pause bit:** The pause bit of the channel-gain word is set to 1 if you want to stop at an entry in the table and wait for the next trigger to resume conversions. In burst mode, the pause bit is ignored.

**Using the skip bit:** The skip bit of the channel-gain word is set to 1 if you want to skip an entry in the table. This feature allows you to sample multiple channels at different rates on each channel.

Function group	Function name	Function code (hex)	Function argument
Channel Gain / Digital Table Control	Write Channel Gain Table (Multi-channel mode)	0x0300	See Figure 2.
	Write Channel Gain Latch (Single-channel mode)	0x0301	See Figure 2.
	Write Digital Table (To control external MUX)	0x0302	See Figure 3.
	Enable Channel Gain Table	0x0303	0x0 = Channel Gain Table disabled Channel Gain Latch enabled 0x1 = Channel Gain Table enabled Channel Gain Latch disabled
	Enable Digital Table	0x0304	0x0 = Digital Table disabled Digital I/O P1 port enabled 0x1 = Digital Table enabled Digital I/O P1 port disabled
	Table Pause enable	0x0305	0x0 = Table Pause disabled 0x1 = Table Pause enabled
	Reset Channel Gain Table	0x030E	-
	Clear Channel Gain Table	0x030F	-

Table 4.1.4.

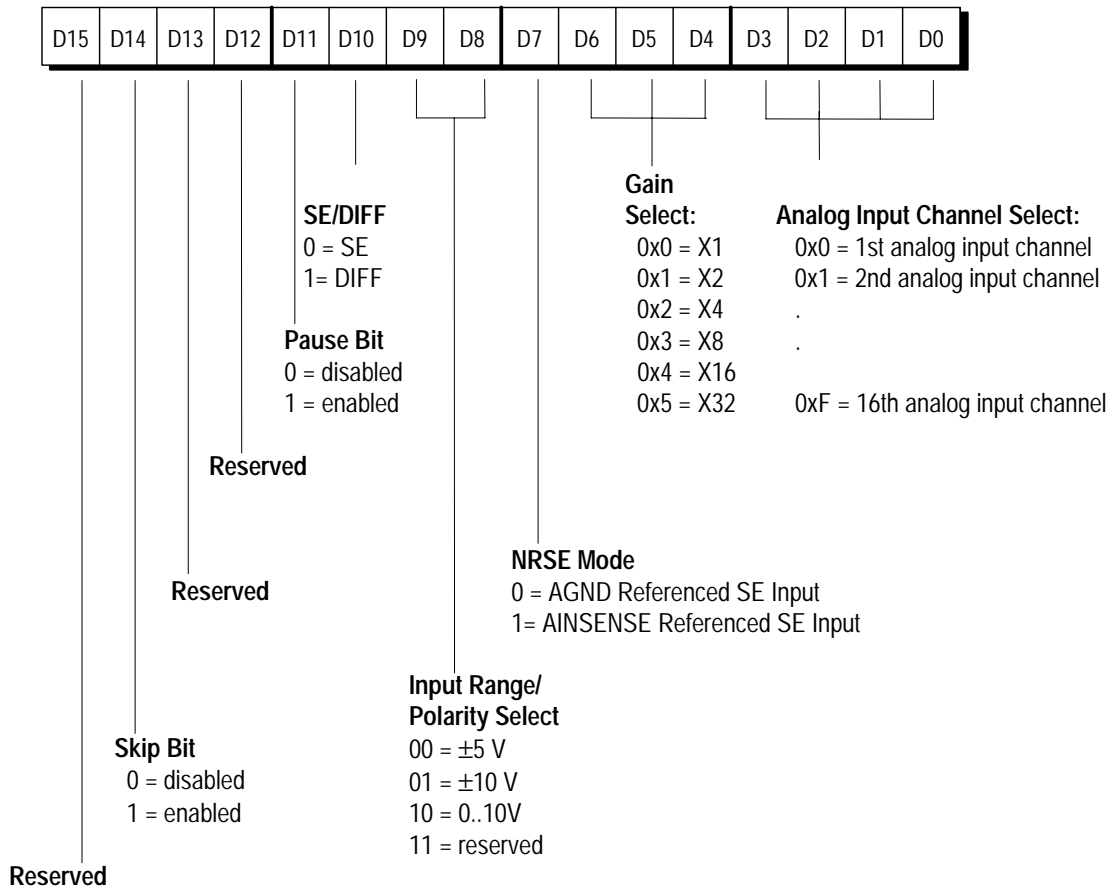


Figure 4.1.1. Channel-gain select latch/ Channel-gain table entry

**Function: Write ADC channel gain latch - 0x0301**

For single-channel operation the Channel Gain Latch must be used. The data structure is the same as in the Channel Gain Table but the skip bit and the Pause Bit are not used. These bits must be zero.

**Function: Write Digital table - 0x0302**

The Digital Table is part of the Channel Gain Table, and can be used to control external devices. Using this function you can fill the 8-bit wide Digital Table. Reading of the Digital Table is simultaneous with reading the Channel Gain Table.

Figure 4.1.2 shows the bit structure of Digital Table.

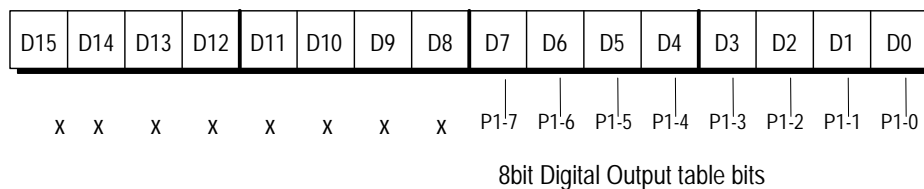


Figure 4.1.2. Digital Output Table bits

The Digital Output Table bits use the same lines as the Digital I/O Chip Port 1 I/O port. If you are using the P1 digital I/O lines the Digital Table bits cannot be used. The Enable Digital Table Function (0x0304) can be used to select between the Digital I/O P1 port and the Digital Output Table bits.

The digital portion of the channel-gain table provides 8-bits to control devices such as external expansion boards. For example, if you have connected one of your input channels on the DM7420 to RTD's TMX32 input expansion board, you can use the bottom 5 bits in this byte to control the TMX32 board channel selection. To load digital information into this portion of the channel, use this function.

This information will be output on the Port 1 lines when you run through the table. The format shown above is for controlling the TMX32's channel selection (32 single-ended or 16 differential). The first load operation will be in the first entry slot of the table (lining up with the first entry in the A/D table), and each load thereafter fills the next position in the channel-gain table. Note that when you are using the digital table, all 8 bits are used and controlled by the table, regardless of the number of bits you may actually need for your digital control application.

**Function: Enable Channel Gain Table- 0x0303**

Using this function you can select the Channel Gain Latch or the Channel Gain Table controlled operation.

**Function: Enable Digital Table - 0x0304**

Using this function you can select the P1 port of Digital I/O chip or the output of the Digital Table on the pin 32..46 of External I/O connector CN2.

**Function: Table Pause enable - 0x0305**

The pause bit of the Channel Gain Table is set to 1 if you want to stop at an entry in the table and wait for the next trigger to resume conversions. In burst mode, the pause bit is ignored.

**Function: Reset A/D Channel Gain Table - 0x030E**

This function sets the read pointer of the Channel Gain Table to the beginning of the Table. The write pointer of the Table does not change.

**Function: Clear A/D Channel Gain Table - 0x030F**

This function sets the read *and* the write pointer of the Channel Gain Table to the beginning of the Table.

The following section shows the *Timer/Counter Control Function Group*.




Function group	Function name	Function code (hex)	Function argument
Timer / Counter Control	A/D Sample Counter Source Select	0x0500	0x0 = Reset Channel Gain Table  0x1 = A/D FIFO write
	A/D Sample Counter Load	0x0501	-
	Pacer Clock Size Select	0x0508	0x0 = 16 bit wide divider -Pacer Clock 0  0x1 = 32 bit wide divider -Pacer Clock 1 (cascaded with Pacer Clock 0)
	Pacer Clock Select	0x0509	0x0 = External Pacer Clock  0x1 = Internal Pacer Clock
	Delay Counter Load	0x050A	-
	About Counter Load	0x050B	-

Table 4.1.6.

**Function: A/D Sample Counter Source Select- 0x0500**

Using this function the A/D Sample Counter Clock can be selected. If you want to count all of the sampled analog data select the A/D FIFO write argument. If you want to count the CGT periods select the Reset Channel Gain Table argument.

**Function: A/D Sample Counter Load- 0x0501**

This function provides a software trigger so that the A/D Sample Counter can be loaded with the correct value. This software correction is used as an easy means to compensate for the operating structure of the 8254. Two pulses of the counter are required to actually load the desired count and prepare the counter to count down correctly (this can be looked at as the initialization procedure for the A/D sample counter). A pulse is sent to the A/D Sample Counter (U25 Counter 2) each time you use this function. Without this correction, the initial count sequence will be off by two pulses. Once the

counter is properly loaded and starts, any subsequent countdowns of this count will be accurate. Note that the A/D sample counter must be programmed for Mode 2 operation. The output of the A/D Sample Counter can be an interrupt source so when the countdown process reaches the zero value an interrupt may be generated. See Figure 4.1.3.

**Function: Pacer Clock Size Select - 0x0508**

Selects a 16-bit or 32-bit on-board pacer clock (U23 TC Counter 0 or 1 output). When a trigger is used to start the pacer clock, there is some delay between the time the trigger occurs and the time the next pacer clock pulse starts an A/D conversion. For a 16-bit clock, this jitter is 250 nanoseconds maximum. However, a 32-bit clock's jitter is dependent on the value programmed into the first divider and can be much greater than 250 nanoseconds. (See Chapter 5.)

**Function: Pacer Clock Select- 0x0509**

Selects the internal pacer clock, which is the output of U23 Counter 0 or 1, or an external pacer clock routed onto the board through External I/O connector. The maximum pacer clock rate supported by the board is 600 kHz.

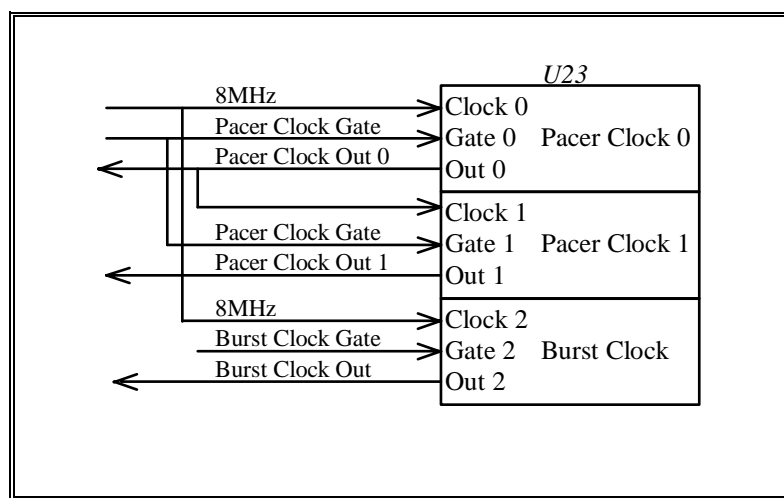


Figure 4.1.4. Pacer and Burst clock TC section

**Function: Delay Counter Load- 0x050A**

This function provides a software trigger so that the A/D delay counter can be loaded with the correct value. This software correction is used as an easy means to compensate for the operating structure of the 8254. Two pulses of the counter are required to actually load the desired count and prepare the counter to count down correctly (this can be looked at as the initialization procedure for the A/D delay counter). A pulse is sent to the A/D delay counter (U25 Counter 0) each time you use this function. Without this correction, the initial count sequence will be off by two pulses. Once the counter is properly loaded and starts, any subsequent countdowns of this count will be accurate. Note that the A/D delay counter must be programmed for Mode 2 operation. See Figure 4.1.4.

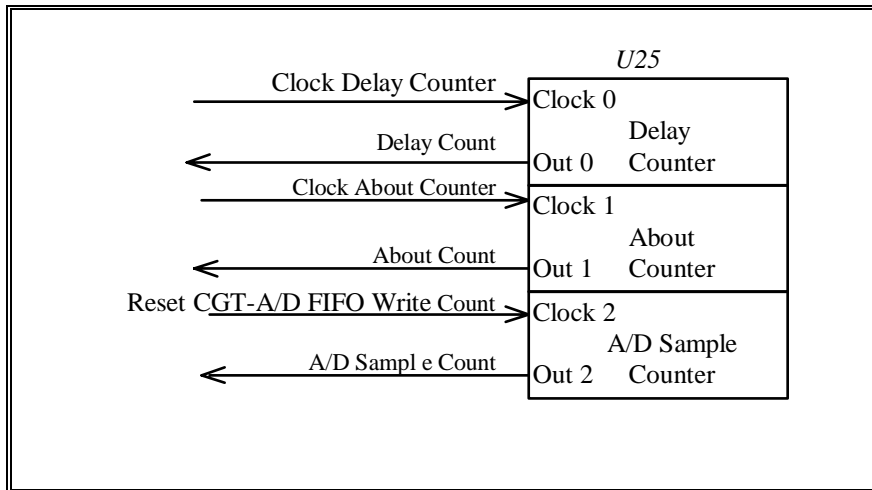


Figure 4.1.5. Delay Counter TC section

**Function: About Counter Load- 0x050B**

This function provides a software trigger so that the About counter can be loaded with the correct value. This software correction is used as an easy means to compensate for the operating structure of the 8254. Two pulses of the counter are required to actually load the desired count and prepare the counter to count down correctly (this can be looked at as the initialization procedure for the About counter). A pulse is sent to the About counter (U25 Counter 1) each time you use this function. Without this correction, the initial count sequence will be off by two pulses. Once the counter is properly loaded and starts, any subsequent countdowns of this count will be accurate. Note that the About counter must be programmed for Mode 2 operation. See Figure 4.1.4.

The following section shows the *External Trigger and External Interrupt Configuration Function Group*.



External Trigger and External Interrupt Configuration	External Trigger polarity select	0x0601	0x0 = positive edge  0x1 = negative edge
	External Interrupt polarity select	0x0602	0x0 = positive edge  0x1 = negative edge

Table 4.1.8.

**Function: External Trigger polarity select- 0x0601**

This function selects the active polarity of External Trigger signal. The External Trigger signal comes from the External I/O connector CN2-47.

**Function: External Interrupt polarity select - 0x0602**

This function selects the active polarity of External Interrupt signal. The External Interrupt signal comes from the External I/O connector CN2-51

The following section shows the *User TC Configuration Function Group*. User TC is an 8254 chip (U26) with three timers, which can be used by the user. (See Figure 4.1.7.) The clock gate sources can be programmed by this Function Group.







Function group	Function name	Function code	Function argument
User Timer-Counter Control	User Timer/Counter 0 Clock Select	0x0700	0x0 = 8MHz  0x1 = Ext. TC Clock 1 0x2 = Ext. TC Clock 2 0x3 = Ext. Pacer Clock
	User Timer/Counter 0 Gate Select	0x0701	0x0 = Not gated  0x1 = Gated 0x2 = Ext. TC Gate 1 0x3 = Ext. TC Gate 2
	User Timer/Counter 1 Clock Select	0x0702	0x0 = 8MHz  0x1 = Ext. TC Clock 1 0x2 = Ext. TC Clock 2 0x3 = Ext. Pacer Clock 0x4 = User Timer/Counter 0 out 0x5 = High-Speed Digital Input Sampling signal
	User Timer/Counter 1 Gate Select	0x0703	0x0 = Not gated  0x1 = Gated 0x2 = Ext. TC Gate 1 0x3 = Ext. TC Gate 2 0x4 = User Timer/Counter 0 out
	User Timer/Counter 2 Clock Select	0x0704	0x0 = 8MHz  0x1 = Ext. TC Clock 1 0x2 = Ext. TC Clock 2 0x3 = Ext. Pacer Clock 0x4 = User Timer/Counter 1 out
	User Timer/Counter 2 Gate Select	0x0705	0x0 = Not gated  0x1 = Gated 0x2 = Ext. TC Gate 1 0x3 = Ext. TC Gate 2 0x4 = User Timer/Counter 1 out

Table 4.1.9.

**Function: User Timer/Counter 0 Clock Select- 0x0700**

This function selects the source of the User TC 0 clock signal. The source of the clock may be the internal 8MHz clock signal, the *External TC Clock x*, or the *External Pacer Clock* signal from the External I/O connector.

**Function: User Timer/Counter 0 Gate Select- 0x0701**

This function selects the source of the User TC 0 gate signal. The source of the gate may be a fixed logic high (not gated, free running mode), fixed logic low (gated shut-down mode) or *External TC Gate x* from the External I/O connector.

**Function: User Timer/Counter 1 Clock Select- 0x0702**

This function selects the source of the User TC 1 clock signal. The source of the clock may be the internal 8MHz clock signal, the *External TC Clock x*, the *External Pacer Clock*, *User Timer/Counter 0 Out* signal or the high-speed digital input FIFO write signal.

**Function: User Timer/Counter 1 Gate Select- 0x0703**

This function selects the source of the User TC 1 gate signal. The source of the gate may be a fixed logic high (not gated, free running mode), fixed logic low (gated shut-down mode), the *External TC Gate x* signal from the External I/O connector or *User Timer/Counter 0 Out* signal.

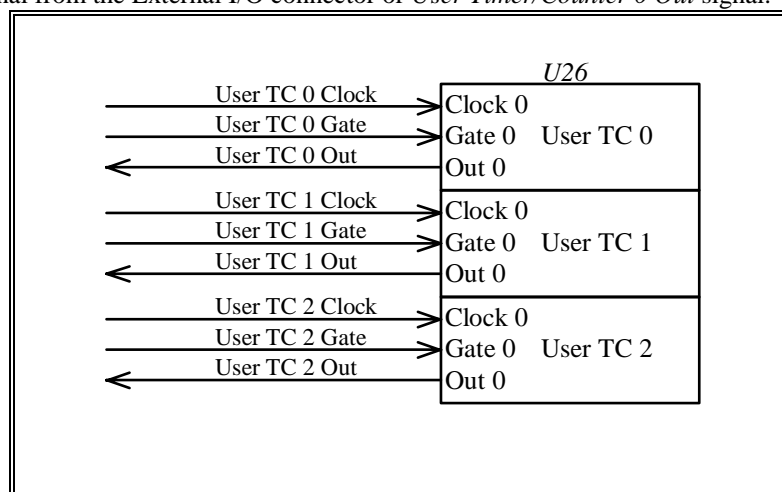


Figure 4.1.7. User TC section

**Function: User Timer/Counter 2 Clock Select- 0x0704**

This function selects the source of the User TC 2 clock signal. The source of the clock may be the internal 8MHz clock signal, the *External TC Clock x*, the *External Pacer Clock* signal or the *User Timer/Counter 1 out* signal.

**Function: User Timer/Counter 2 Gate Select- 0x0705**

This function selects the source of the User TC 2 gate signal. The source of the gate may be a fixed logic high (not gated, free running mode), fixed logic low (gated shut-down mode), the *External TC Gate x* signal from the External I/O connector or *User Timer/Counter 1 Out* signal.

The following section shows the *User Output Configuration Function Group*. The source of the *User Out 0* and *User Out 1* can be selected with this function. The selected sources are buffered and connected to the External I/O connector.



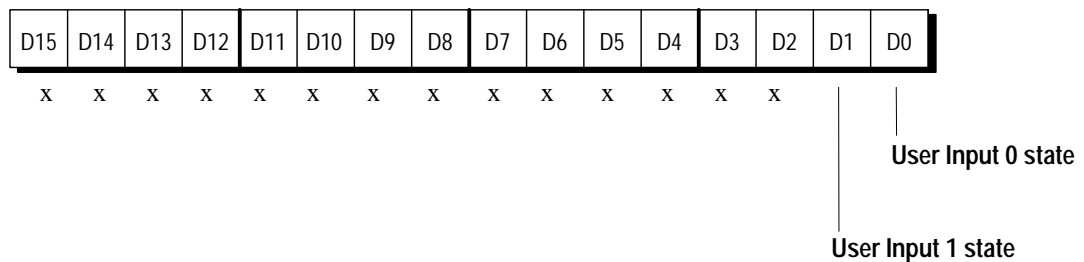
Function group	Function name	Function code	Function argument
User Output Signal Control	User Output Signal 0 select	0x070E	0x0 = A/D Conversion Signal  0x1 = RESERVED 0x2 = RESERVED 0x3 = Software Programmable by LAS 0+4h
	User Output Signal 1 select	0x070F	0x0 = A/D Conversion Signal  0x1 = RESERVED 0x2 = RESERVED 0x3 = Software Programmable by LAS 0+4h

Table 4.1.10.

### 4.2.3. Local Address Space 0 +4h: User Input read / User Output Write (Read/write)

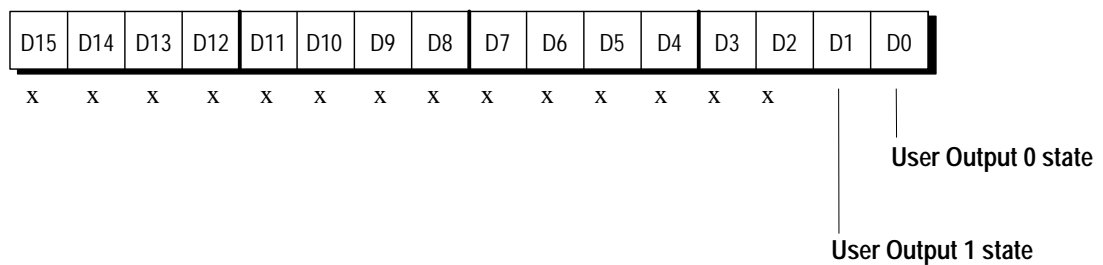
#### Read operation (16-bit)

A read provides the User Input 0 and User Input 1 bits as below. These digital input lines come from the External I/O connector. The User Input bits are sampled by the read instruction.



#### Write operation (16-bit)

These bits go to the External I/O connector of the board. If the source of the User Output x is set to the Software Programmable state by the 0x070E and 0x070F Functions, the state of the User Output bits can be programmed by this write operation.





### 4.2.8. Local Address Space 0 +14h: Pacer Clock Software Start trigger / Pacer Clock Software Stop trigger

#### Read operation (16-bit)

A read generates a software start trigger for the Pacer Clock, if the start trigger source for the pacer clock is software trigger.

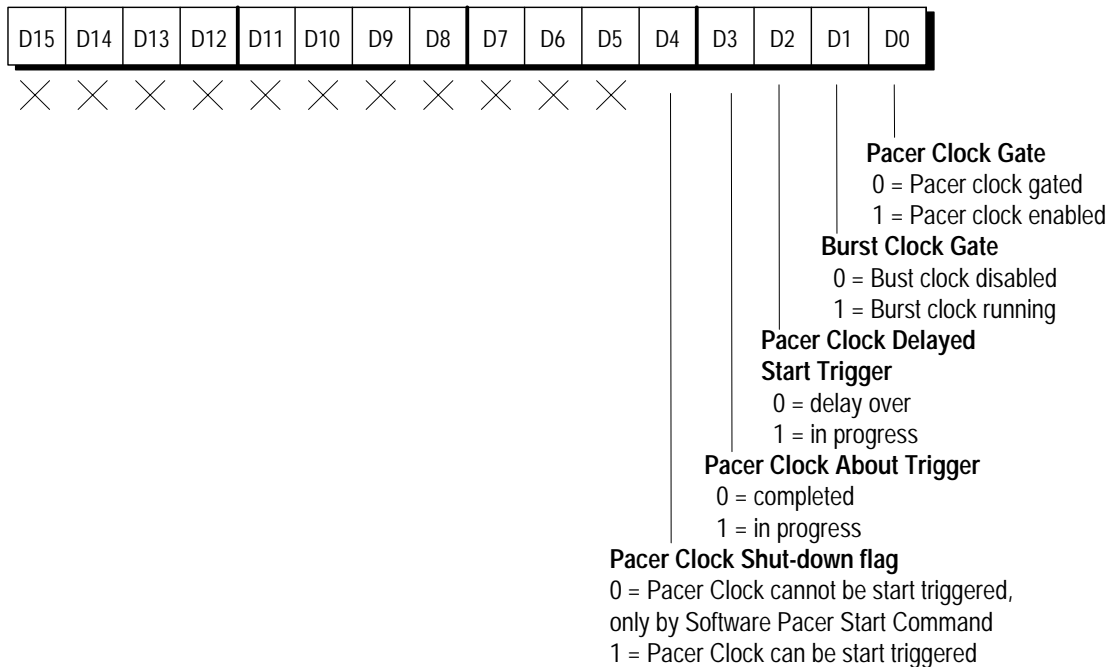
#### Write operation (16-bit)

A write generates a software stop trigger for the Pacer Clock, if the stop trigger source for the pacer clock is software trigger. The written data is irrelevant.

### 4.2.9. Local Address Space 0 +16h: Timer Counter Status Register (Read/Write)

#### Read operation (16-bit)

A read provides the status of the gate of the Timer Counter circuits.



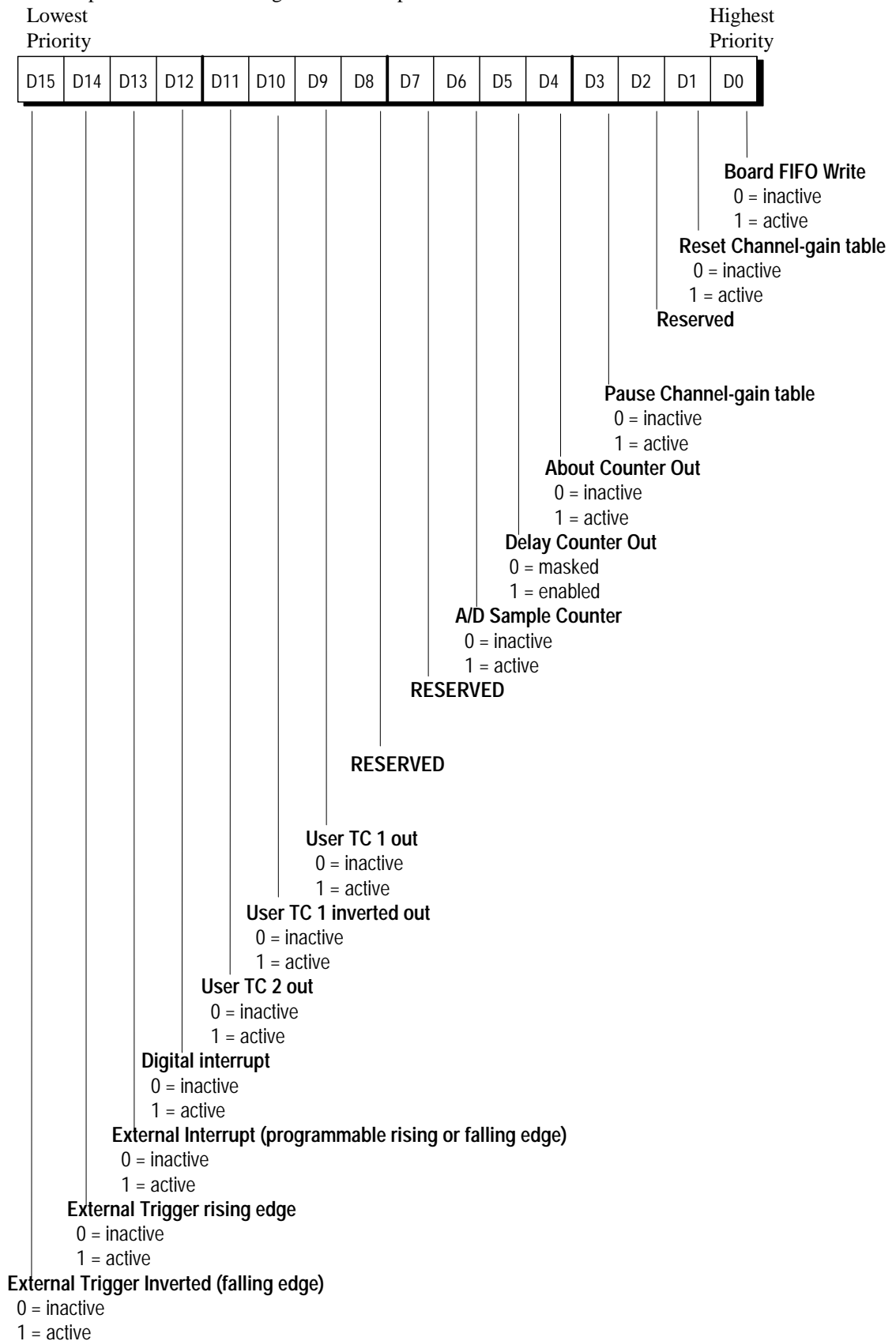
#### Write operation (16-bit)

A write generates a software stop trigger for the High-Speed Digital Input FIFO. The written data is irrelevant.

### 4.2.10. Local Address Space 0 +18h: Interrupt Status/Mask Register Read/write)

#### Read operation (16-bit)

A read provides the status flag of the interrupt.



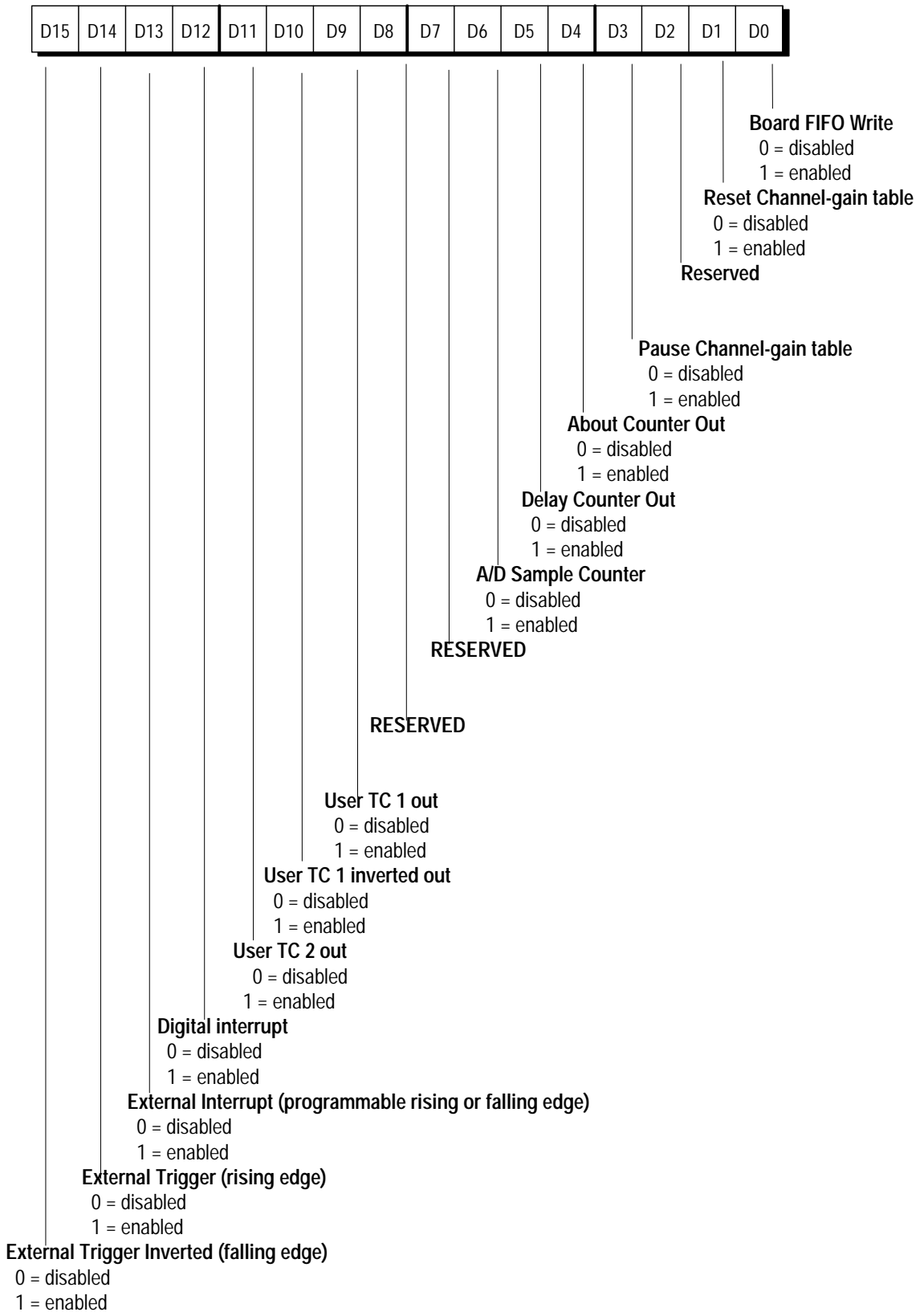
The DM7420 has a built-in Interrupt Controller that allows for multiple interrupts sources ordered by their priority. The following steps illustrate how to use the controller.

1. Set the Interrupt Mask Register (Write LAS0+18h) in your initialization part of the software. Enable the required interrupt sources.
2. The built-in Priority Interrupt Controller orders the interrupt requests and transmits them to the PC. If an interrupt occurs, you can identify the active source by reading the Interrupt Status Register (Read LAS0+18h) in the Interrupt Service Routine. In the Interrupt Status Register always one bit is high indicating the active interrupt source. After identifying the source the request can be serviced.
3. Clear the serviced Interrupt request by the Interrupt Clear register. First write the clear mask, writing the appropriate bit pattern to address LAS0 +1Ah. Then a dummy read from LAS0+1A executes the clear.

If you want to check that during servicing the interrupt a new interrupt has not come yet, after clearing the interrupt request read the Interrupt Overrun Register. Zero bits mean that all interrupts have been serviced correctly. One means that a new interrupt occurred before the previous service was finished. After reading the Interrupt Overrun Register clear it.

**Write operation (16-bit)**

The interrupt mask register:





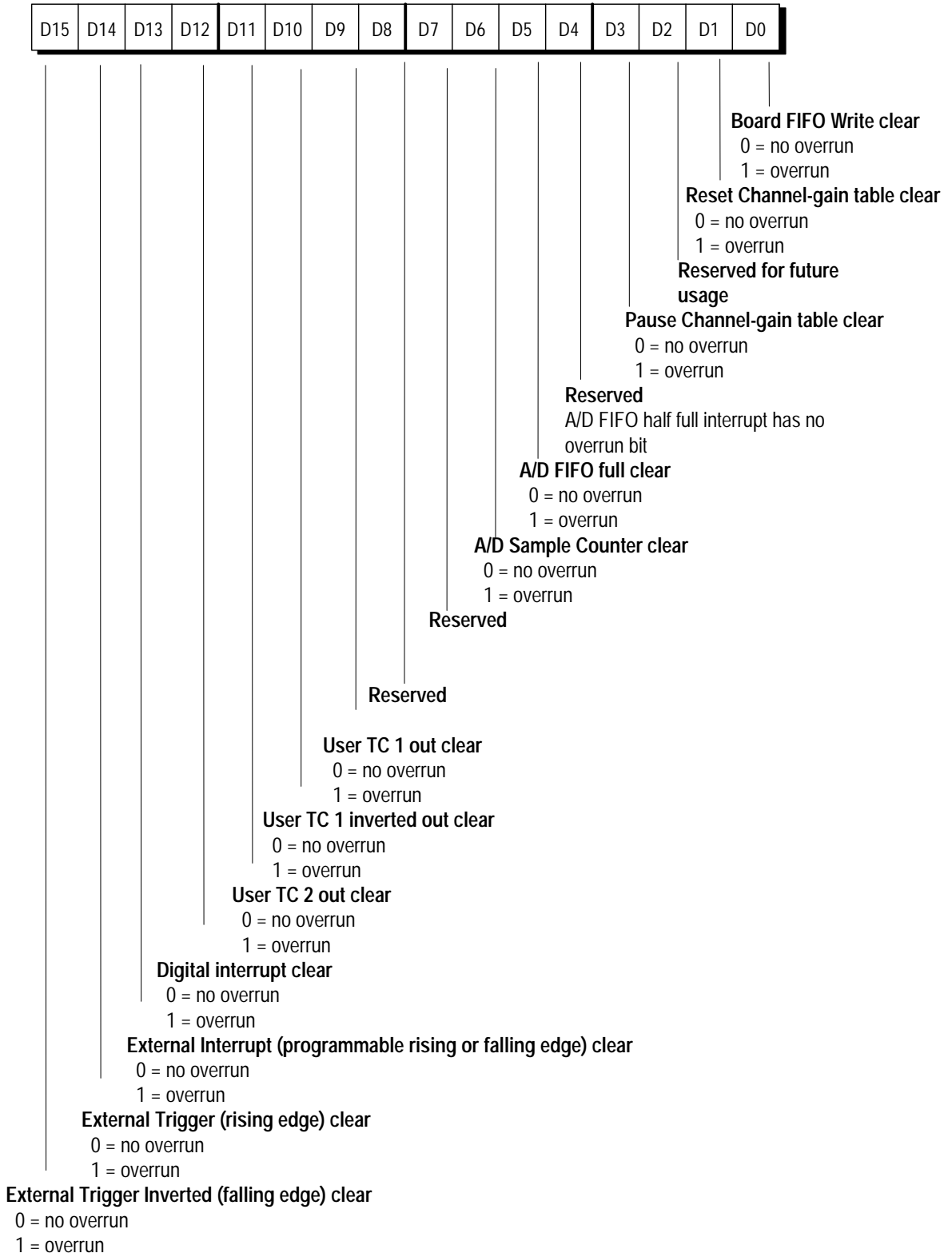
### 4.2.12. Local Address Space 0 +1Ch: Interrupt Overrun Register (Read/Write)

#### Write operation (16-bit)

A write clears all bits of the Interrupt Overrun Register.

#### Read operation (16-bit)

A read provides the Interrupt Overrun Register. If the interrupt is serviced in time all bits will be zero. If a new interrupt request came before the previous has been serviced and the request is cleared, the appropriate overrun bit goes high.



### 4.3. Local Address Space 1 (LAS1)

This I/O address space can be accessed by word wide (16-bit), or double word-wide (32-bit) I/O instructions. This address space can be burst read or written.<sup>1</sup>

Range size is 8 bytes.

This address space is used to transfer data from the A/D input FIFO and the High Speed Digital Input FIFO.

Read Function	Write Function	Local Address Space 1 Offset hex
Read A/D FIFO	-	0h (16-bit)
Read High Speed Digital Input FIFO	-	2h (16-bit)

Table 4.2.1.

#### 4.3.1. Local Address Space 1 +0h: Read A/D FIFO (Read only)

##### Read operation (16-bit)

A read provides the 12-bit A/D converted data as shown below. Bit 15 is the sign bit extension. This sign bit extension gives the opportunity to read the converted data as two's complement number in either unipolar or bipolar mode.

The bottom three bits are the samples of the buffered version of the External I/O connector Port 0 Digital I/O port P0-5, P0-6, P0-7 lines.

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
Sign	Bit	Bit	Bit	Bit	Bit	Bit	Bit	Bit	Bit	Bit	Bit	Bit	P0-7	P0-6	P0-5
	12	11	10	9	8	7	6	5	4	3	2	1			
(MSB)												(LSB)	Digital Input Data Markers		

#### 4.3.2. Local Address Space 1 +2h: Read High Speed Digital Input FIFO (Read only)

##### Read operation (16-bit)

A read provides the 8-bit High Speed Digital Input Data bits which are programmable source sampled. The High Speed Digital Input lines are commonly used with the Digital I/O , bit-programmable P0 port. The upper byte is undefined.

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
X	X	X	X	X	X	X	X	P0-7	P0-6	P0-5	P0-4	P0-3	P0-2	P0-1	P0-0
								DM7	DM6	DM5	DM4	DM3	DM2	DM1	DM0

<sup>1</sup> In Pentium systems you can generate only two word long burst read or write using double word access. There are no any other possibility of the burst access if you use the Pentium itself. If you have an external bus master device (DMA controller), it can be used to transfer data in burst mode.

#### 4.4. Local Address Space 2 (LAS2)

This address space is used to access the 8254 Timer/Counter chips (U23, U25, U26,) and the Digital I/O chip (U24) on the board. This is an 8-bit wide 32 byte long I/O address space and cannot be accessed in burst mode. The Local Address Space 2 can be accessed by byte-wide instructions.

Register Description	Read Function	Write Function	Local Address Space 2 Offset
8254 TC Counter 0 ( <i>U23</i> ) <i>Pacer Clock 0</i>	Read value in TC Counter 0	Load count in TC Counter 0	0h (8-bit)
8254 TC Counter 1 <i>Pacer Clock 1</i>	Read value in TC Counter 1	Load count in TC Counter 1	1h (8-bit)
8254 TC Counter 2 <i>Burst Clock</i>	Read value in TC Counter 2	Load count in TC Counter 2	2h (8-bit)
8254 TC Control Word	Reserved	Program counter mode for TC	3h (8-bit)
8254 TC Counter 0 ( <i>U25</i> ) <i>Delay Counter</i>	Read value in TC Counter 0	Load count in TC Counter 0	4h (8-bit)
8254 TC Counter 1 <i>About Counter</i>	Read value in TC Counter 1	Load count in TC Counter 1	5h (8-bit)
8254 TC Counter 2 <i>A/D Sample counter</i>	Read value in TC Counter 0	Load count in TC Counter 0	6h (8-bit)
8254 TC Control Word	Reserved	Program counter mode for TC	7h (8-bit)
8254 TC Counter 0 ( <i>U26</i> ) <i>User TC 0</i>	Read value in TC Counter 0	Load count in TC Counter 0	8h (8-bit)
8254 TC Counter 1 <i>User TC 1</i>	Read value in TC Counter 1	Load count in TC Counter 1	9h (8-bit)
8254 TC Counter 2 <i>User TC 2</i>	Read value in TC Counter 2	Load count in TC Counter 2	Ah (8-bit)
8254 TC Control Word	Reserved	Program counter mode for TC	Bh (8-bit)
-	-	-	-
-	-	-	-
-	-	-	-
-	-	-	-
Digital I/O Port 0 ( <i>U44</i> )	Read Port 0 digital input lines	Program Port 0 digital output lines	10h (8-bit)
Digital I/O Port 1	Read Port 1 digital input lines	Program Port 1 digital output lines	11h (8-bit)
Port 0 Clear/Direction/ Mask/Compare	Clear digital IRQ status flag/read Port 0 direction, mask or compare register	Clear digital chip/program Port 0 direction, mask or compare register	12h (8-bit)
Read Digital I/O Status/ Set Digital Control Register	Read digital status word	Program digital control register & digital interrupt enable	13h (8-bit)
RESERVED	-	RESERVED	14..1Fh (8-bit)

Table 4.3.1.

### 4.4.1. Local Address Space 2 +0h, 1h, 2h, 4h, 5h, 6h, 8h, 9h, Ah : Timer/Counter 0,1,2

**Read operation (8-bit)**

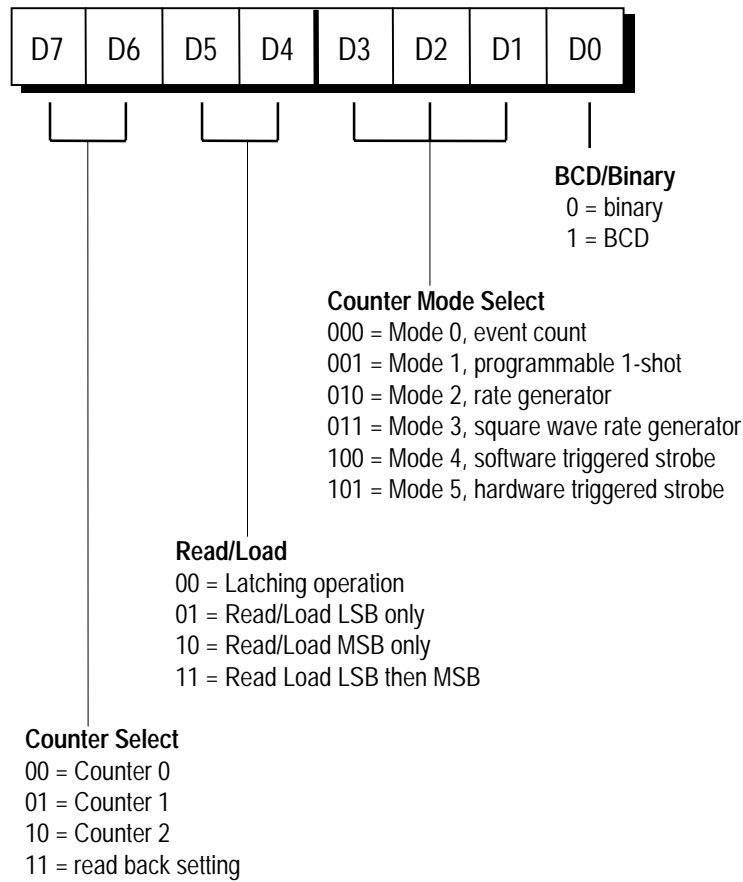
A read shows the count in the first, second and third counter of U23, U25 and U26.

**Write operation (8-bit)**

A write loads the first, second and third counter of U23, U25 or U26 with a new 16-bit value in two 8-bit steps, LSB followed by MSB. The counter must be loaded in two 8-bit steps! Counting begins as soon as the count is loaded.

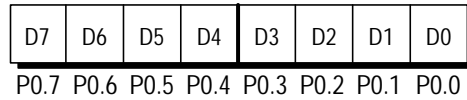
### 4.4.2. Local Address Space 2 +3h, 7h, Bh: Timer/Counter control word

**Read/Write 8-bit Operation.** Accesses the timer/counter's control register to directly control the three 16-bit counters, 0, 1, and 2.



#### 4.4.3. Local Address Space 2 +10h - Digital I/O chip Port 0, Bit Programmable Port (Read/Write)

8-bit operation.

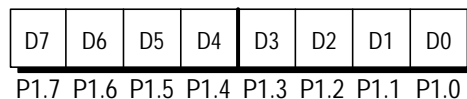


This port transfers the 8-bit Port 0 bit programmable digital input/output data between the board and external devices. The bits are individually programmed as input or output by writing to the Direction Register at LAS2+12h. For all bits set as inputs, a read reads the input values and a write is ignored. For all bits set as outputs, a read reads the last value sent out on the line and a write writes the current loaded value out to the line.

Note that when any reset of the digital circuitry is performed (clear chip or computer reset), all digital lines are reset to inputs and their corresponding output registers are cleared.

#### 4.4.4. Local Address Space 2 +11h - Digital I/O chip Port 1, Byte Programmable Port (Read/Write)

8-bit operation.

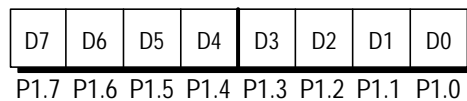


This port transfers the 8-bit Port 1 digital input or digital output byte between the board and an external device. When Port 1 is set as inputs, a read reads the input values and a write is ignored. When Port 1 is set as outputs, a read reads the last value sent out of the port and a write writes the current loaded value out of the port.

Note that when any reset of the digital circuitry is performed (clear chip or computer reset), all digital lines are reset to inputs and their corresponding output registers are cleared.

#### 4.4.5. Local Address Space 2 + 12h - Read/Program Port 0 Direction/ Mask/ Compare Registers (Read/Write)

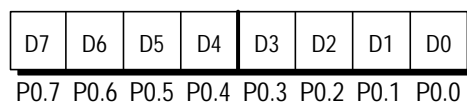
8-bit operation.



A read clears the IRQ status flag or provides the contents of one of digital I/O Port 0's three control registers; and a write clears the digital chip or programs one of the three control registers, depending on the setting of bits 0 and 1 at LAS2+13h. When bits 1 and 0 at LAS2+13h are 00, the read/write operations clear the digital IRQ status flag (read) and the digital chip (write). When these bits are set to any other value, one of the three Port 0 registers is addressed.

##### Direction Register (LAS2+ 13h, bits 1 and 0 = 01):

This register programs the direction, input or output, of each bit at Port 0.

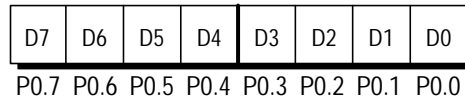


For all bits:

0 = input

1 = output

**Mask Register (LAS2 +13h, bits 1 and 0 = 10):**



For all bits:  
 0 = bit enabled  
 1 = bit masked

In the Advanced Digital Interrupt modes, this register is used to mask out specific bits when monitoring the bit pattern present at Port 0 for interrupt generation. In normal operation where the Advanced Digital Interrupt feature is not being used, any bit which is masked by writing a 1 to that bit will not change state, regardless of the digital data written to Port 0. For example, if you set the state of bit 0 low and then mask this bit, the state will remain low, regardless of what you output at Port 0 (an output of 1 will not change the bit’s state until the bit is unmasked).

**Compare Register (LAS2+13h, bits 1 and 0 = 11):**

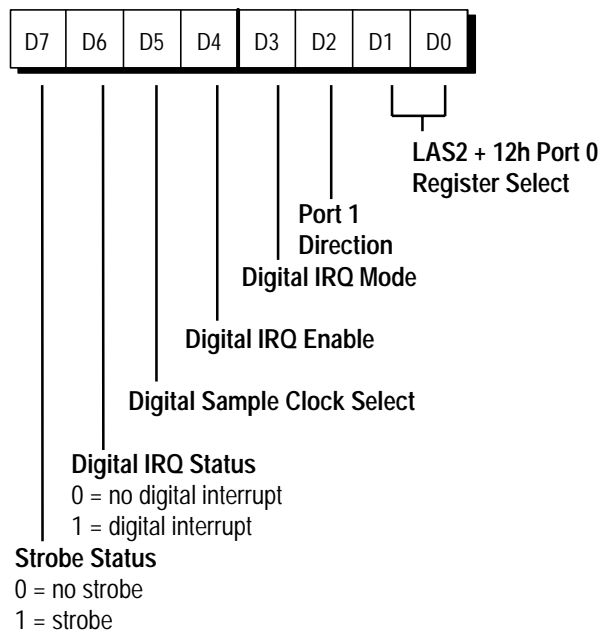
This register is used for the Advanced Digital Interrupt modes. In the match mode where an interrupt is generated when the Port 0 bits match a loaded value, this register is used to load the bit pattern to be matched at Port 0. Bits can be selectively masked so that they are ignored when making a match. NOTE: Make sure that bit 3 at LAS2 + 13h is set to 1, selecting match mode, BEFORE writing the Compare Register value at this address. In the event mode where an interrupt is generated when any Port 0 bit changes its current state, the value which caused the interrupt is latched at this register and can be read from it. Bits can be selectively masked using the Mask Register so a change of state is ignored on these lines in the event mode.

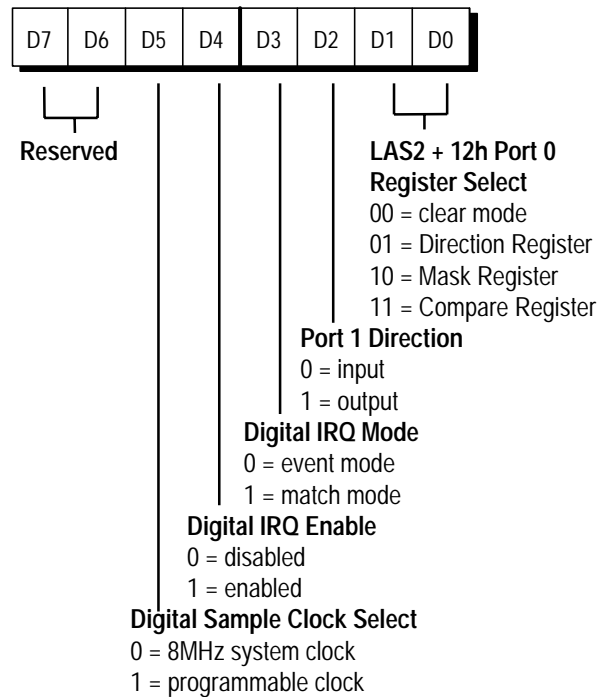
**4.4.6. Local Address Space 2 + 13h - Read Digital IRQ Status/Program Digital Mode (Read/Write)**

**8-bit operation.**

**Digital IRQ/Strobe Status (Read):**

A read shows you whether a digital interrupt has occurred (bit 6), whether a strobe has occurred (bit 7, when using the strobe input as described in Chapter 7), and lets you review the states of bits 0 through 5 in this register. If bit 6 is high, then a digital interrupt has taken place. If bit 7 is high, a strobe has been issued.



**Digital Mode Register (Write):**

Bits 0 and 1 – Select the clear mode initiated by a read/write operation at LAS2 + 12h or the Port 0 control register you talk to at LAS2 + 12h (Direction, Mask, or Compare Register).

Bit 2 – Sets the direction of the Port 1 digital lines.

Bit 3 – Selects the digital interrupt mode: event (any Port 0 bit changes state) or match (Port 0 lines match the value programmed into the Compare Register at LAS2 +12h).

Bit 4 – Disables/enables digital interrupts.

Bit 5 – Sets the clock rate at which the digital lines are sampled when in a digital interrupt mode. Available clock sources are the 8 MHz system clock and the output of User TC Counter 1 (16-bit programmable clock). When a digital input line changes state, it must stay at the new state for two edges of the clock pulse (62.5 nanoseconds when using the 8 MHz clock) before it is recognized and before an interrupt can be generated. This feature eliminates noise glitches that can cause a false state change on an input line and generate an unwanted interrupt. This feature is detailed in Chapter 7.

Bit 6 – Reserved.

Bit 7 – Reserved.



## 5. A/D Conversion

This chapter shows you how to program your DM7420 to perform A/D conversions and read the results. Included in this discussion are instructions on setting up the Channel Gain Table (CGT), the on-board clocks and sample counter, and various conversion and triggering modes. The following paragraphs walk you through the programming steps for performing A/D conversions. Detailed information about the conversion modes and triggering is presented in this section. You can follow these steps in the example programs included with the board.

### 5.1. Before Starting Conversions: Initializing the Board

Regardless of the conversion mode you wish to set up, you should always start your program with a board initialization sequence. This sequence should include:

- Clear Board command
- Clear IRQ command
- Clear Channel Gain Table command
- Clear A/D FIFO command
- Clear Digital I/O chip

This initialization procedure clears all board registers, empties the Channel Gain Table, resets the digital I/O chip and empties the A/D FIFO.

#### 5.1.1. Before Starting Conversions (single-channel mode): Programming Channel, Gain, Input Range and Type using Channel Gain Latch (CGL)

Setting up these things can be done using the Channel Gain Latch (single-channel mode) or using the Channel Gain Table (multi-channel mode). The CGL can be filled up by Function 0x301. The Channel Gain Latch has very similar structure to the Channel Gain Table, so all operations are explained in the next sections of CGT.

#### 5.1.2. Before Starting Conversions (multi-channel mode): Programming the Channel-Gain Table (CGT)

The Channel Gain Table can be programmed with 1024 24-bit entries in tabular format. Sixteen bits contain the A/D channel-gain data (A/D Table), and 8 bits contain digital control data (Digital Table) to support complex channel-gain sequences. To load a new Channel Gain Table, first clear the Channel Gain Table by Function 0x030F (see Table 4.1.4.). To add entries to an existing table, simply write to the A/D Table (and Digital Table if used) as described in the following paragraphs. Note that writing beyond the end of the table is ignored.

#### 5.1.3. 16-Bit A/D Table

The A/D portion of the Channel Gain Table with the channel, gain, input range, input type, pause and skip bit information is programmed into the channel-gain scan memory using the Function 0x300. If you have cleared the existing table, the first word written will be placed in the first entry of the table, the second word will be placed in the second entry, and so on. If you are adding to an existing table, the new data written will be added at the end.

### 5.1.4. Channel Select, Gain Select, Input Range and Input Type

The channel number, gain value, input range and input type are entered in the table using bits 0 through 10. Each of these parameters can be set independently for every entry in the table. This allows you to set up a complex array of sampling sequences mixing channels, gains, input ranges and input types. Care must be taken in selecting the proper input type. The board is capable of 16 single-ended inputs or 8 differential inputs. You can select combinations of single-ended and differential but each differential channel actually uses 2 single-ended channels. If you select channel 1 to be a differential channel, you must connect your signal to AIN1+ and AIN1-. Channel 8 now is not available as a single-ended channel. In the case of single ended mode you can choose the Ground Referenced Single Ended (GRSE mode) or the Non Referenced Single Ended Mode (NRSE). See Chapter 2.

### 5.1.5. Pause bit

Bit 11 is used as a pause bit. If this bit is set to a "1" and the Pause function is enabled by Function 0x0305, the A/D conversions will stop at this entry in the table and resume on the next Start Trigger. This is useful if you have 2 different sequences loaded in the table. You can enable and disable this bit's function by Function 0x305. In the case of single channel mode, when the CGL is used this function is meaningless.

NOTE: This bit is ignored in the Burst sampling modes.

### 5.1.7. Skip bit

If bit 14 of the data loaded is set to 1, then the skip bit is enabled and this entry in the channel-gain table will be skipped, meaning an A/D conversion will be performed but the data is not written into the A/D FIFO. This feature provides an easy way to sample multiple channels at different rates without saving unwanted data. A simple example illustrates this bit's function.

In this example, we want to sample channel 1 once each second and channel 4 once every three seconds. First, we must program 6 entries into the channel-gain table. The channel 4 entries with the skip bit set will be skipped when A/D conversions are performed. The table will continue to cycle until a stop trigger is received.

Next, we will set the pacer clock to run at 2 Hz (0.5 seconds). This allows us to sample each channel once per second, the maximum sampling rate required by one of the channels (pacer clock rate = number of different channels sampled x fastest sample rate). The first clock pulse starts an A/D conversion according to the parameters set in the first entry of the channel-gain table, and each successive clock pulse incrementally steps through the table entries. As shown in Figure 5-1 and Figure 5-2, the first clock pulse starts a sample on channel 1. The next pulse looks at the second entry in the channel-gain table and sees that the skip bit is set to 1. No A/D data is stored. The third pulse starts a sample on channel 1 again, the fourth pulse skips the next entry, and the fifth pulse takes our third reading on channel 1. On the sixth pulse, the skip bit is disabled and channel 4 is sampled. Then the sequence starts over again. Samples are not stored when they are not wanted, saving memory and eliminating the need to throw away unwanted data.

### 5.1.8. 8-Bit Digital Table

The digital portion of the channel-gain table can be programmed with digital control information using the Write Digital Table Function 0x0302. If you have cleared the existing table by the CGT clear Function 0x030F, the first byte written will be placed in the first entry of the table, the second byte will be placed in the second entry, and so on. If you are adding to an existing table, the new data written will be added at the end. The first entry made into the Digital Table lines up with the first entry made into the A/D Table, the second entry made into the Digital Table lines up with the second entry made into the A/D Table, and so on. Make sure that, if you add to an existing table and did not program the Digital Table portion when you made your A/D Table entries previously, you fill those entries with

digital data first before entering the desired added data. Since the first digital entry you make always lines up with the first A/D entry made, failure to do this will cause the A/D and digital control data to be misaligned in the table. You cannot turn the digital control lines off for part of a conversion sequence and then turn them on for the remainder of the sequence. Note that the digital data programmed here is sent out on the Port 1 digital I/O lines whenever this portion of the table is enabled by the Function 0x0304.

These lines can be used to control input expansion boards such as the TMX32 analog input expansion board at the same speed as the A/D conversions are performed with no software overhead.

NOTE: If you only need to use the A/D part of the table, you do not have to program the Digital Table. However if you only want to use the Digital part of the table you must program the A/D part of the table.

### 5.1.9. Setting up the Channel Gain Table

Let's look at how the Channel Gain Table is set up for a simple example using both the A/D and Digital Tables. In this example, we have a TMX32 expansion board connected to channel 1 on the DM7420. Load the channel-gain sequence into the A/D Table (Function 0x0300):

```
Entry 1      0000 0000 0000 0000 gain = 1, channel number = 1
Entry 2      0000 0000 0010 0000 gain = 4, channel number = 1
Entry 3      0100 0000 0000 0000 skip sample
Entry 4      0000 0000 0010 0000 gain = 4, channel number = 1
Entry 5      0000 0000 0000 0000 gain = 1, channel number = 1
Entry 6      0000 0000 0010 0000 gain = 4, channel number = 1
```

Load the digital data into the Digital Table by Function 0x0302. The first digital word loaded lines up with the first A/D Table entry, and so on:

```
Entry 1 0000 0000 0000 0000 gain=1 DM7420 channel=1 0000 0000 TMX32 channel=1
Entry 2 0000 0000 0010 0000 gain=4 DM7420 channel=1 0000 0011 TMX32 channel=4
Entry 3 0000 1000 0000 0000 skip sample                0000 0000 TMX32 channel=1 (skip)
Entry 4 0000 0000 0010 0000 gain=4 DM7420 channel=1 0000 0011 TMX32 channel=4
Entry 5 0000 0000 0000 0000 gain=1 DM7420 channel=1 0000 0000 TMX32 channel=1
Entry 6 0000 0000 0010 0000 gain=4 DM7420 channel=1 0000 0011 TMX32 channel=4
```

### 5.1.10. Using the Channel Gain Table for A/D Conversions

After the Channel Gain Table is programmed, it must be enabled in order to be used for A/D conversions by Function 0x0303. The Digital Table can be enabled by Function 0x0304 when the digital control data is stored. You cannot use the Digital Table without enabling the Channel Gain Table. When the Digital Table is enabled, the 8-bit data is sent out on the Port 1 digital I/O lines.

When you are using the channel-gain table to take samples, it is strongly recommended that you do not enable, disable, and then re-enable the table while performing a sequence of conversions. This causes skipping of an entry in the table. In this case you should issue a reset table command by Function 0x030E.

### 5.1.11. Channel-gain Table and Throughput Rates

When using the Channel Gain Table, you should group your entries to maximize the throughput of your module. Low-level input signals and varying gains are likely to drop the throughput rate because low level inputs must drive out high level input residual signals. To maximize throughput:

- Keep channels configured for a certain range grouped together, even if they are out of sequence.
- Use external signal conditioning if you are performing high speed scanning of low level signals. This increases throughput and reduces noise.

- If you have room in the channel-gain table, you can make an entry twice to make sure that sufficient settling time has been allowed and an accurate reading has been taken. Set the skip bit for the first entry so that it is ignored.

## 5.2. A/D Conversion Modes

To support a wide range of sampling requirements, the PCI400 provides several conversion modes with a selection of trigger sources to start and stop a sequence of conversions. Understanding how these modes and sources can be configured to work together is the key to understanding the A/D conversion capabilities of your module.

The following paragraphs describe the conversion and trigger modes.

### 5.2.1. Start A/D Conversion signal

Using the Function 0x0200 one of nine modes can be selected as A/D conversion signal as can be seen on Figure 5.2.1.

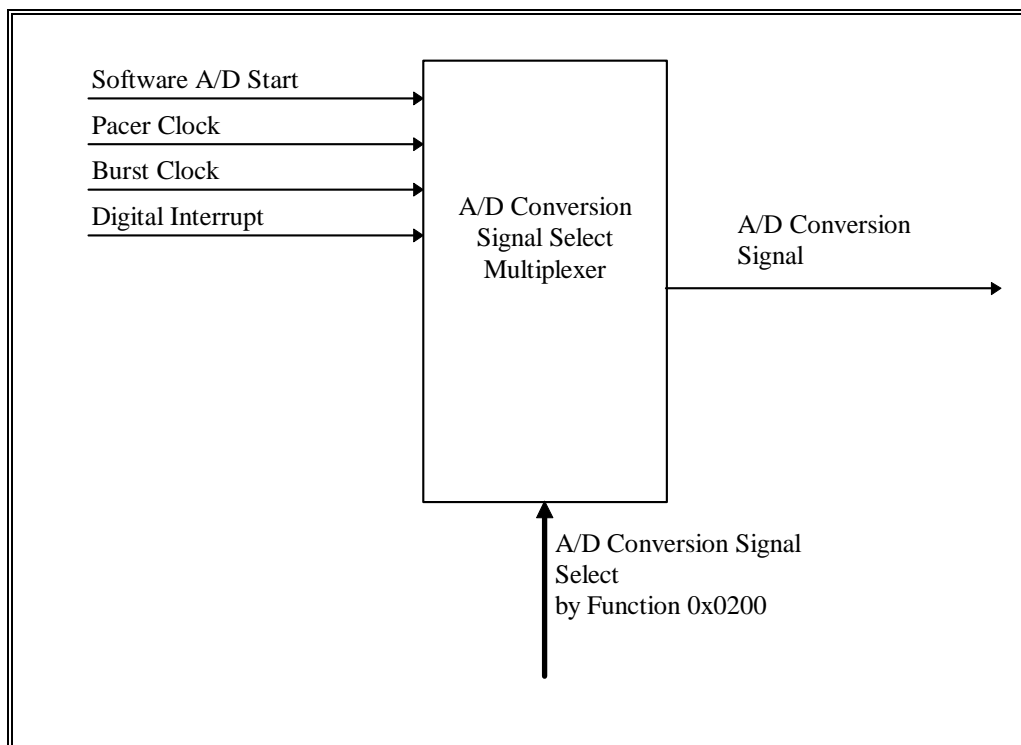


Figure 5.2.1.1.

- *Software A/D Start* (by writing LAS0+8h to initiate a Start Convert)
- *Pacer Clock* (internal TC - see Figure 4.1.4 -or external)
- *Burst Clock* (internal TC - see Figure 4.1.4 )
- *Digital Interrupt* generated by the Advanced Digital Interrupt circuit

### 5.2.2. Pacer Clock Start/Stop Trigger Select.

The Pacer Clock start trigger can be set by the Function 0x0202. The Pacer Clock stop trigger can be set by the Function 0x0203. This functions can be used to turn the pacer clock (internal or

external) on and off. Through these different combinations of start and stop triggers, the DM7420 supports pre-trigger, post-trigger, and about-trigger modes with various trigger sources.

**The Pacer Clock start trigger sources are:**

- *Software Pacer Start:* When selected, a read at LAS0+14h will start the Pacer Clock.
- *External trigger:* When selected, a positive- or negative-going edge (depending on the setting of the trigger polarity, Function 0x0601) on the external TRIGGER INPUT line, will start the pacer clock. The pulse duration should be at least 100 nanoseconds.
- *Digital interrupt:* When selected, a Advanced Digital interrupt -generated by the Digital I/O chip will start the Pacer Clock.
- *User TC 2 out:* When selected, a pulse on the User Timer/Counter 2 - see Figure 4.1.7. - output line (Counter 2's count reaches 0) will start the pacer clock.

The following start trigger sources provide delayed triggering. When the trigger is issued, the A/D delay counter - see Figure 4.1.5. -, counts down and conversions are started when the A/D delay counter reaches 0. The A/D delay counter counts at the pacer clock rate.

- *Delayed Software Pacer Start.* When selected, a read at LAS0+14h will start the delay counter.
- *Delayed external trigger.* When selected, a positive- or negative-going edge (depending on the setting of the trigger polarity, bit 11 in the Control Register) on the external TRIGGER INPUT line, will start the delay counter. The pulse duration should be at least 100 nanoseconds.
- *Delayed digital interrupt.* When selected, a digital interrupt will start the delay counter.
- *Delayed User TC Counter 2 output.* When selected, a pulse on the Counter 2 output line (Counter 2's count reaches 0) will start the delay counter.
- *External Trigger Gated mode.* When selected, the pacer clock runs when the external TRIGGER INPUT line is held high. When this line goes low, conversions stop. This trigger mode does not use a stop trigger. If the trigger polarity bit is set for negative, the pacer clock runs when this line is low and stops when it is taken high.

**The Pacer Clock stop trigger sources are:**

- *Software Pacer Stop triggers:* When selected, a write at LAS0+14h will stop the Pacer Clock.
- *External trigger.* When selected, a positive- or negative-going edge (depending on the setting of the trigger polarity, setting up by Function 0x602) on the external TRIGGER INPUT line, will stop the Pacer Clock. The pulse duration should be at least 100 nanoseconds.
- *Digital interrupt:* When selected, a digital interrupt will stop the pacer clock.
- *About Counter:* When selected, the Pacer Clock stops when the About Counter's count reaches 0. About Counter counts samples which are written into the A/D FIFO
- *User TC2 out:* When selected, the Pacer Clock stops when the User TC 2 counter's count reaches 0.

The next stop trigger sources provide about triggering, where data is acquired from the time the start trigger is received, and continues for a specified number of samples after the stop trigger. The number of samples to acquire after the stop trigger is programmed in the About Counter. About Counter counts samples, which are written into the A/D FIFO

- *About Software Pacer Stop trigger:* When selected, a Software Pacer Stop trigger starts the About counter, and sampling continues until the About Counter's count reaches 0.
- *About external trigger:* When selected, an external trigger starts the About counter, and sampling continues until the sample counter's count reaches 0.
- *About digital interrupt.* When selected, a digital interrupt starts the About Counter, and sampling continues until the About Counter's count reaches 0.
- *About User TC Counter 2 output:* When selected, a pulse on the User Timer Counter 2 output line (Counter 2's count reaches 0) starts the About Counter, and sampling continues until the About Counter's count reaches 0.

Note that the external trigger (TRIGGER INPUT) can be set to occur on a positive-going edge or a negative-going edge, depending on the setting up the Function 0x0602.

**Burst Clock Start Trigger.**

The following paragraph describes the operation when the A/D conversion start signal is selected as Burst Clock. Burst clock is an output of Timer Counter See Figure 4.1.4. The clock signal of the Burst Clock Timer Counter is 8MHz. The gate signal of this TC is used to start and stop the Burst Clock. The start signal can be select from the following list, and the stop is derived from the empty signal of Channel Gain Table. The Burst Clock operation belongs to the multichannel - Channel Gain Table operation.

The start triggers can be set by Function 0x0201:

- *Software A/D start* (by writing LAS0+8h)
- *Pacer Clock* (internal or external )
- *external TRIGGER INPUT*
- *Digital Interrupt*

**Single Cycle Mode, Trigger Repeat mode.**

Using the Pacer Start Mode select Function 0x0205 the Single Cycle mode or the Trigger Repeat Mode can be selected. This function controls the conversion sequence when using a trigger to start the Pacer Clock. When the Function argument is low, the first pulse on the selected Pacer Clock Start Trigger source will start the pacer clock. After the stop trigger has ended the conversion cycle, the triggering circuit is disarmed and must be rearmed before another start trigger can be recognized. To rearm this trigger circuit, you must issue a Software A/D Start command (Write LAS0+8h).

When Function argument is high, the conversion sequence is repeated each time a selected Pacer Clock Start Trigger is received.

**Pacer Clock Source.** The Pacer Clock can be generated from an internal source or an external source using the Function 0x0509.

**5.2.3. Types of Conversions****Single Conversion.**

In this mode, a single specified channel is sampled whenever the Software A/D Start Command is occurred. The active channel is the one specified in the Channel Gain Latch. This is the easiest of all conversions. It can be used in a wide variety of applications, such as sample every time a key is pressed on the keyboard, sample with each iteration of a loop, or watch the system clock and sample every five seconds.

**Multiple Conversions.**

In this mode, conversions are continuously performed at the Pacer Clock rate, or other selected A/D Conversion Signal rate. The pacer clock can be internal or external. The maximum rate supported by the board is 600 kHz. The Pacer Clock can be turned on and off using any of the start and stop triggering modes using the Function 0x202, and 0x203. If you use the internal pacer clock, you must program it to run at the desired rate.

This mode is ideal for filling arrays, acquiring data for a specified period of time, and taking a specified number of samples.

**Random Channel Scan.**

In this mode, the Channel Gain Table is incrementally scanned through, with each selected A/D Conversion Signal pulse starting a conversion at the channel and gain specified in the current table entry. Before starting a conversion sequence Channel Gain Table, you need to load the table with the desired data. Then make sure that the Channel Gain Table is enabled by the Function 0x0303. This enables the A/D portion of the Channel Gain Table. If you are using the Digital Table as well, you must also enable this using the Function 0x0304. Each rising edge of selected A/D Conversion Signal starts a conversion using the current Channel Gain data and then increments to the next position in the table. When the last entry is reached, the next pulse starts the table over again.

**Programmable Burst.**

In this mode, a single trigger initiates a scan of the entire Channel-gain table. Before starting a burst of the channel-gain table, you need to load the table with the desired data. Then make sure that the channel-gain table is enabled by setting bit 2 at BA + 2 high. This enables the A/D portion of the channel-gain table. If you are using the Digital Table as well, you must also set bit 3 at BA + 2 high.

Burst is used when you want one sample from a specified number of channels for each trigger. Figure 5-8 shows a timing diagram for burst sampling. As shown, the burst trigger, which is a trigger or pacer clock, triggers the burst and the burst clock initiates each conversion. At high speeds, the burst mode emulates simultaneous sampling of multiple input channels. For time critical simultaneous sampling applications, a simultaneous sample-and-hold board can be used (SS8 eight-channel boards are available from Real Time Devices).

**Programmable Multiscan.**

This mode - when the A/D Conversion Start Signal is the Burst Clock - lets you scan the Channel Gain Table after a Burst Clock Start Signal. When the Channel Gain Table is empty the Burst Clock is stopped, and waiting for a new Start Signal.

As you can see, the DM7420 is designed to support a wide range of conversion requirements. You can set the clocks, triggers, and channel and gain to a number of configurations to perform simple or very complex acquisition schemes where multiple bursts are taken at timed intervals. Remember that the key to configuring the board for your application is to understand what signals can actually control conversions and what signals serve as triggers. The discussions presented in this section and the example programs on the disk should help you to understand how to configure the board.

**Starting an A/D Conversion**

Depending on your conversion and trigger settings, the Software A/D Start command (Write LAS0+8h) has different functions. In any mode that uses the Software A/D Start command, this command will do the appropriate action. In any mode that does not use the Software A/D Start command as the trigger, you will still need to do a write the LAS0+8h to arm (enable) the triggering circuitry. An example of this would be, if you set the Pacer Clock Start Trigger as external trigger, a write to LAS0+8h is required to arm the external trigger circuitry. After you have set all the trigger and conversion registers to the proper values, the last command will need to be Software A/D Start. Any external triggers received before this command will be ignored. It is also a good practice to clear the A/D FIFO just prior to triggering the measurement or arming the trigger. Study the example programs to see this sequence.

**Conversion Status Monitoring**

The A/D conversion status can be monitored through the A/D FIFO empty flag in the FIFO status word read at LAS0+8h. Typically, you will want to monitor the Empty flag (active low) for a transition from low to high. This tells you that a conversion is complete and data has been placed in the sample buffer.

**Halting Conversions**

In single convert modes, a single conversion is performed and the module waits for another Software A/D Start command. In multi-convert modes, conversions are halted by one of two methods: when a stop trigger has been issued to stop the pacer clock, or when the FIFO is full. The Pacer Clock Shut Down Flag, bit 4 of the status word (LAS0+16h), is set when the sample buffer is full, disabling the A/D converter. Even if you've removed data from the sample buffer since the buffer filled up and the FIFO full flag is no longer set, the Pacer Clock Shut Down Flag will confirm that at some point in your conversion sequence, the sample buffer filled and conversions were halted. At this point a clear A/D FIFO command must be issued and a Software A/D Start convert (write at LAS0+8h) to rearm the trigger circuitry.

**5.3. Reading the Converted Data**

Each 12-bit conversion is stored in a 16-bit word in the sample buffer, in the A/D FIFO. The buffer can store 1024 samples. This section explains how to read the data stored in the sample buffer.

The sample buffer -A/D FIFO contains the converted data and 3-bit data marker (if used) in a 16-bit word.

The 12-bit A/D data + sign bit is left justified in a 16-bit word, with the least significant three bits reserved for the data marker. Because of this, the A/D data read must be scaled to obtain a valid A/D reading. The data marker portion should be masked out of the final A/D result. Shifting the word three bits to the right will eliminate the data marker from the data word. If you are using the data marker, then you should preserve these bits someplace in your program.

The output code format is always two's complement. This is true for both bipolar and unipolar signals since the sign bit is added above the 12-bit conversion data. For bipolar conversions, the sign bit will follow the MSB of the 12-bit data. If this bit is a "0", the reading is a positive value. If this bit is a "1", the reading is a negative value. When the input is a unipolar range, the coding is the same except that the sign bit is always a "0" indicating a positive value. The data should always be read from the A/D FIFO as a signed integer.

Voltage values for each bit will vary depending on input range and gain. For example, if the input is set for  $\pm 5$  volts and the gain = 1, the formula for calculating voltage is as follows:

$$\text{Voltage} = ((\text{input range} / \text{Gain}) / 4096) \times \text{Conversion Data}$$

$$\text{Voltage} = ((10 / 1) / 4096) \times \text{Conversion Data}$$

$$\text{Voltage} = 2.44 \text{ mV} \times \text{Conversion Data}$$

Remember that when you change the gain, you are increasing the resolution of the bit value but you are decreasing the input range. In the above example if we change the gain to 4, each bit will now be equal to 610  $\mu\text{V}$  but our input range is decreased from 10 volts to 2.5 volts. The formula would look like this:

$$\text{Voltage} = ((\text{input range} / \text{Gain}) / 4096) \times \text{Conversion Data}$$

$$\text{Voltage} = ((10 / 4) / 4096) \times \text{Conversion Data}$$

$$\text{Voltage} = 610 \text{ uV} \times \text{Conversion Data}$$

If we now change the input range to  $\pm 10$  volts and the gain = 1, the formula would be:

$$\text{Voltage} = ((\text{input range} / \text{Gain}) / 4096) \times \text{Conversion Data}$$

$$\text{Voltage} = ((20 / 1) / 4096) \times \text{Conversion Data}$$

$$\text{Voltage} = 4.88 \text{ mV} \times \text{Conversion Data}$$

## 5.4. Using the A/D Data Markers

For certain applications where you may want to store digital information with the analog data at the same rate the analog data is being acquired, the bottom three bits of the converted data are available for this feature. For example, you may want to tag the acquired data with a marker so that you know when the data was sampled. Three lines are available at I/O connector to send the data marker settings to the sample buffer along with the 12-bit A/D converted data. These lines are P0.5, P0.6 and P0.7.

## 5.5. Programming the Pacer Clock

Two 16-bit timers in the U23 TC counters 0 and 1, are cascaded to form a 16-bit or 32-bit on-board pacer clock, shown in Figure 4.1.4. When you want to use the pacer clock for continuous A/D conversions, you must select a 16-bit or 32-bit clock configuration and program the clock rate.

### 5.5.1. Selecting 16-bit or 32-bit Pacer Clock

The size of the pacer clock, 16-bit or 32-bit, is programmed by the Function 0x0508. When the Function argument is set to 0, a 16-bit Pacer Clock is selected. Whenever possible, it is strongly recommended that the 16-bit pacer clock be used to minimize the delay between the time a trigger occurs and the first conversion is initiated by the pacer clock. When using a 16-bit clock, the first conversion will always start within 250 nanoseconds of the trigger, and subsequent conversions are synchronized to the pacer clock. The 16-bit clock conversion speeds can be set from 600 kHz down to 123 Hz.

Because the 32-bit pacer clock cascades two 16-bit timers, the uncertainty between the time a trigger occurs and the first conversion is initiated can be significantly greater than for the 16-bit clock.

The triggering uncertainty here is based on the value programmed into the first divider and can become unacceptable for certain applications. However, for conversion rates slower than 123 Hz, you must use the 32-bit pacer clock. The 32-bit clock is selected by the Function 0x0508 with Function argument 1. When programming the 32-bit clock, you should always program the smallest possible value in Divider 1 in order to minimize the triggering uncertainty.

### 5.5.2. Programming Steps

The pacer clock is accessed for programming at LAS2+0 and LAS2+1 addresses. To find the value you must load into the clock to produce the desired rate, you first have to calculate the value of Divider 1 (Clock TC Counter 0) for a 16-bit clock, or the value of Divider 1 and Divider 2 (Clock TC Counter 1) for a 32-bit clock (Figure 4.1.4.) The formulas for making this calculation are as follows:

16-bit pacer clock frequency = 8 MHz/(Divider 1)  
 Divider 1 = 8 MHz/16-bit Pacer Clock Frequency

32-bit pacer clock frequency = 8 MHz/(Divider 1 x Divider 2)  
 Divider 1 x Divider 2 = 8 MHz/32-bit Pacer Clock Frequency

To set the 16-bit pacer clock frequency at 500 kHz, this equation becomes:  
 Divider 1 = 8 MHz/500 kHz ---> 16 = 8 MHz/500 kHz

When Divider 1 is greater than 65,536, you will have to select a 32-bit pacer clock and program the clock rate into Dividers1 and 2. When programming the 32-bit clock, divide the result by the least common denominator. The least common denominator is the value that is loaded into Divider1, and the result of the division, the quotient, is loaded into Divider 2. The tables below list some common pacer clock frequencies and the counter settings for a 16-bit and a 32-bit pacer clock.

After you calculate the decimal value of each divider, you can convert the result to a hex value if it is easier for you when loading the count into each 16-bit counter.

16-Bit Pacer Clock	Divider 1 decimal / (hex)
500 kHz	16 / (0010)
100 kHz	80 / (0050)
50 kHz	160 / (00A0)
10 kHz	800 / (0320)
1 kHz	8000 / (1F40)

32-Bit Pacer Clock	Divider 1 decimal / (hex)	Divider 2 decimal / (hex)
100 Hz	2 / (0002)	40000 / (9C40)
10 Hz	16 / (0010)	50000 / (C350)

To set up the 16-bit pacer clock, follow these steps:

1. Set pacer clock size to 16 bits (Pacer Clock Size Select Function - 0x0508 with argument 0).
2. Use LAS2+0.. 3 area to access the TC of Pacer Clock
3. Program Counter 0 for Mode 2 operation.
4. Load Divider 1 LSB by byte wide operation.
5. Load Divider 1 MSB by byte wide operation..

To set up the 32-bit pacer clock, follow these steps:

1. Set pacer clock size to 32 bits (Pacer Clock Size Select Function - 0x0508 with argument 1).
2. Use LAS2+0.. 3 area to access the TC of Pacer Clock
3. Program Counter 0 for Mode 2 operation.
4. Program Counter 1 for Mode 2 operation.

5. Load Divider 1 LSB by byte wide operation.
6. Load Divider 1 MSB by byte wide operation.
7. Load Divider 2 LSB by byte wide operation.
8. Load Divider 2 MSB by byte wide operation.

Depending on your conversion mode, the counters start their countdown and the pacer clock starts running when a trigger occurs.

### **5.6. Programming the Burst Clock**

The third 16-bit timer in the U23 TC, Counter 2, is the on-board Burst Clock. When you want to use the Burst Clock for performing A/D conversions in the burst mode, you must program the clock rate. To find the value you must load into the clock to produce the desired rate, make the following calculation:

$$\text{Burst clock frequency} = 8 \text{ MHz} / \text{Counter 2 Divider}$$

To set the burst clock frequency at 100 kHz using the on-board 8 MHz clock source, this equation becomes:

$$\text{Burst clock frequency} = 8 \text{ MHz} / 100 \text{ kHz} \rightarrow 80 = 8 \text{ MHz} / 100 \text{ kHz}$$

After you determine the value that will result in the desired clock frequency, load it into Counter 2. In this case, decimal 80 (hex 0050) is loaded into the counter.

To set up the burst clock, follow these steps:

1. Use LAS2+2.. 3 area to access the TC of Burst Clock
2. Program Counter 2 for Mode 2 operation.
3. Load Divider LSB.
4. Load Divider MSB.

Depending on your conversion mode, the counter start its countdown and the burst clock starts running when a trigger occurs.

### **5.7. Programming the About Counter**

The About Counter lets you program the DM7420 to take a certain number of samples and then halt conversions. (Select A/D Conversion Signal to Pacer Clock, select the Pacer Clock Stop Trigger to About Counter) The number of samples to be taken is loaded into the 16-bit sample counter, Counter1 TC U25 See Figure 4.1.5 Recall that because of the operating structure of the 8254, the count loaded initially is not the count which is counted down during the first cycle. A software correction is used as an easy means to compensate for this. Two pulses of the counter are required to actually load the desired count and prepare the counter to count down correctly (this can be looked at as the initialization procedure for the sample counter). A pulse is sent to the 8254 sample counter each time you use Function About Counter Load, x050B. Without this correction, the initial count sequence will be off by two pulses. Note that once the counter is properly loaded and starts, any subsequent countdowns of this count will be accurate.

After you determine the desired number of samples, load the count into Counter1.

To set up the About Counter, follow these steps:

1. Use LAS2+9h, +Bh address to access the TC of Pacer Clock
2. Program Counter 0 for Mode 2 operation.
3. Load Count LSB.
4. Load Count MSB.
5. Use the About Counter Load Function 0x050B two times so that the loaded count matches the desired count.

### 5.7.1. Using the About Counter to Create Large Data Arrays

The 16-bit About Counter allows you to take up to 65,535 samples before the count reaches 0 and sampling is halted. Suppose, however, you want to take 100,000 samples and stop. The DM7420 provides a Function, About Counter Stop Enable 0x0204 that allows you to use the About counter to take more than 65,535 samples in a conversion sequence.

The About Counter stop enable bit can be set to 1 to allow the sample counter to continuously cycle through the loaded count until the stop enable bit is set to 0, which then causes the sample counter to stop at the end of the current cycle. Let's look back at our example where we want to take 100,000 readings. First, we must divide 100,000 by a whole number that gives a result of less than 65,535. In our example, we can divide as follows:

$$\text{Sample Counter Count} = 100,000 / 2 = 50,000$$

To use the sample counter to take 100,000 samples, we will load a value of 50,000 into the counter and cycle the counter two times. After the value is loaded, make sure that the Stop Bit is set to 1 so that the sample counter will cycle. Then, set up the sample counter so that it generates an interrupt when the count reaches 0. Initialize the sample counter as described in the preceding section and start the conversion sequence. When the sample counter interrupt occurs telling you that the count has reached 0 and the cycle is starting again, set the Stop Bit to 0 to stop the sample counter after the second cycle is completed. The result: the sample counter runs through the count two times and 100,000 samples are taken.



## 6. Interrupts

This chapter explains the priority Interrupt Controller of the board.

The DM7420 uses the INTA# interrupt line which are assigned to one of the free IRQ channels by the PCI BIOS.

Because of the several interrupt sources on the board a Priority Interrupt Controller was built on the board. This controller assures even usage all of the interrupt sources on the board.

The Priority Interrupt Controller register area is at the LAS0+28.. LAS0+1C.

### 6.1. The Operation of Priority Interrupt Controller

After power-up all of the interrupt sources are disabled on the board. In this state place your Interrupt Service Routine which will be used in the case of an interrupt generated by the board.

The initialization process of the controller is:

- Set all bits to 1 in the Interrupt Clear Mask Register.
- Read a dummy data from Clear Interrupt set by Clear Mask address. These two steps means that all Interrupt requests are cleared
- Write Interrupt mask register. If an interrupt source must be used, that position in the register must be set to 1. (See Chapter 4.2.10.)

After this initialization process the Interrupt Controller receives the interrupt requests, and according to their priority order transmits them to the PC. One time one request.

In the Interrupt service routine you must identify the current interrupt source reading by Interrupt Status Register. In this register there is a position where the bit is high, signaling the active interrupt source. All of the other bits are zero. Identifying the source it can be serviced. After servicing the request must be cleared by accessing the Interrupt clear mask and the Clear Interrupt set by Clear Mask registers.

In the normal operation, the next interrupt request comes later than clearing of the previous. If this is override, it can be detected by the Interrupt Overrun register (See Chapter 4.2.12.). If the interrupts serviced in time all bits are zeros in the overrun register. If a new interrupt request comes before the previous has been serviced and the request is cleared, the appropriate overrun bit goes into high signaling the faulty - too slow interrupt service - operation.

### 6.2. Advanced Digital Interrupts

The bit programmable digital I/O circuitry (P0 Port) supports two Advanced Digital Interrupt modes, event mode or match mode. These modes are used to monitor digital input lines (P0..7) for state changes. The mode is selected at LAS2+13h, D3 and enabled at LAS2+13h, D4.

#### 6.2.1. Event Mode

When enabled, this mode samples the Port 0 input lines at a specified clock rate (using the 8 MHz system clock or a programmable clock in User TC Counter 1), looking for a change in state in any one of the eight bits. When a change of state occurs, an interrupt is generated and the input pattern is latched into the Compare Register. You can read the contents of this register at LAS2+12h to see which bit caused the interrupt to occur. Bits can be masked and their state changes ignored by programming the Mask Register with the mask at LAS2+12h.

#### 6.2.2. Match Mode

When enabled, this mode samples the Port 0 input lines at a specified clock rate (using the 8 MHz system clock or a programmable clock in User TC Counter 1) and compares all input states to the value programmed in the Compare Register at LAS2+12. When the states of all of the lines match the value in the Compare Register, an interrupt is generated. Bits can be masked and their states ignored by programming the Mask Register with the mask at LAS2+12h.

### **6.2.3. Sampling Digital Lines for Change of State**

In the Advanced Digital Interrupt modes, the digital lines are sampled at a rate set by the 8 MHz system clock or the clock programmed in User TC Counter 1. With each clock pulse, the digital circuitry looks at the state of the next Port 0 bits. To provide noise rejection and prevent erroneous interrupt generation because of noise spikes on the digital lines, a change in the state of any bit must be seen for two edges of a clock pulse to be recognized by the circuit.

## 7. Timer/Counters

Three 8254 programmable interval timers, U23, U25, U26, each provide three 16-bit, 8-MHz timers for timing and counting functions such as frequency measurement, event counting, and interrupts. Two of the timers in the (U23) are cascaded and used for the on-board pacer clock, described in Chapter 5. The third timer is the burst clock, also discussed in Chapter 5. See Figure 8.1.

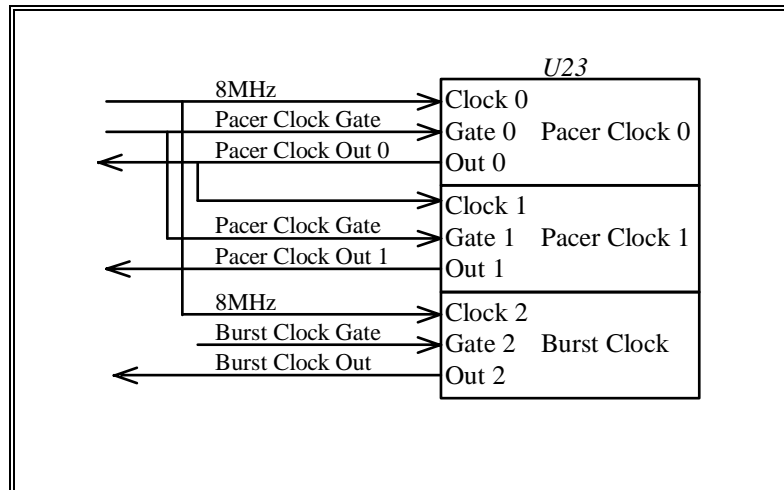


Figure 8.1.

The Counter 2 of 8254 at U25 is the A/D sample counter. The A/D sample counter is discussed in Chapter 5.

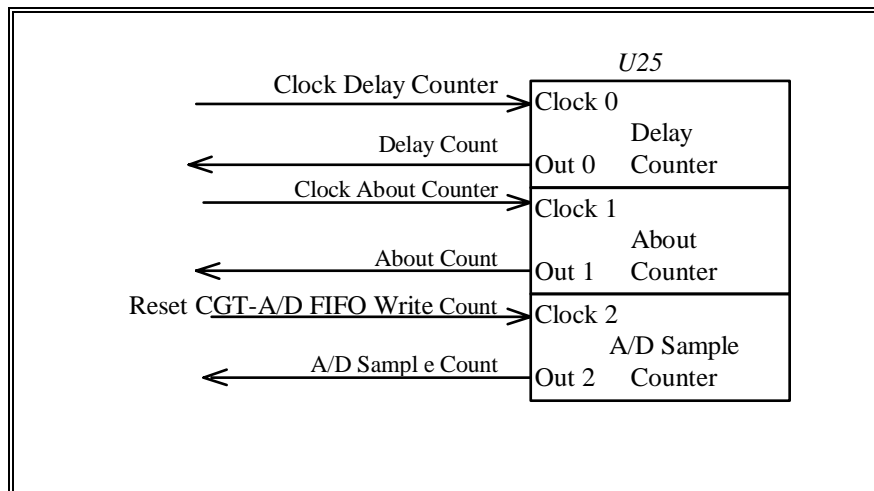


Figure 8.2.

The Counter 0 of 8254 at U25 is the Delay Counter, Counter 1 is the About Counter. The Delay and About Counter are discussed in chapter 5.

The 8254 at U26 is the User TC. All three counters on this chip are available for user functions. For details on the programming modes of the 8254, see the data sheet in Appendix.

Each timer/counter has two inputs, CLK in and GATE in, and one output, timer/counter OUT. They can be programmed as binary or BCD down counters by writing the appropriate data to the command word, as described in the I/O map discussion in Chapter 4. See Figure 8.4. The sources of the user TC clocks and gates can be programmed by User Timer Counter Function group, 0x0700.. 0x0705 Functions.

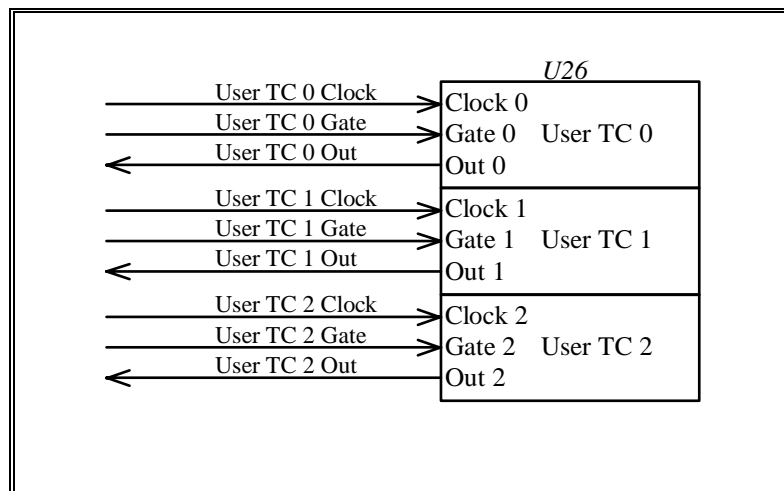


Figure 8.4.

### 7.1. Timer/Counters: Operating Modes

The timers can be programmed to operate in one of six modes, depending on your application. The following paragraphs briefly describe each mode.

**Mode 0, Event Counter (Interrupt on Terminal Count).** This mode is typically used for event counting. While the timer/counter counts down, the output is low, and when the count is complete, it goes high. The output stays high until a new Mode 0 control word is written to the timer/counter.

**Mode 1, Hardware-Retriggerable One-Shot.** The output is initially high and goes low on the clock pulse following a trigger to begin the one-shot pulse. The output remains low until the count reaches 0, and then goes high and remains high until the clock pulse after the next trigger.

**Mode 2, Rate Generator.** This mode functions like a divide-by-N counter and is typically used to generate a real-time clock interrupt. The output is initially high, and when the count decrements to 1, the output goes low for one clock pulse. The output then goes high again, the timer/counter reloads the initial count, and the process is repeated. This sequence continues indefinitely.

**Mode 3, Square Wave Mode.** Similar to Mode 2 except for the duty cycle output, this mode is typically used for baud rate generation. The output is initially high, and when the count decrements to one-half its initial count, the output goes low for the remainder of the count. The timer/counter reloads and the output goes high again. This process repeats indefinitely.

**Mode 4, Software-Triggered Strobe.** The output is initially high. When the initial count expires, the output goes low for one clock pulse and then goes high again. Counting is "triggered" by writing the initial count.

**Mode 5, Hardware Triggered Strobe (Retriggerable).** The output is initially high. Counting is triggered by the rising edge of the gate input. When the initial count has expired, the output goes low for one clock pulse and then goes high again.

## 8. Digital I/O

The DM7420 has digital circuitry to receive and transmit digital data from, or to the external digital world. This chapters describes only the 31.. 46 pins of External I/O connector. These 16 Digital input/output lines are multifunction and assure a flexible connection with the digital world.

The connection of odd numbered pin2 is shown in the Figure 9.1. These lines are monitored by the high-speed digital input circuitry, and the least significant three bits of the A/D FIFO as data markers, therefore these lines can be read or driven by the Port 0 of Digital I/O Chip.

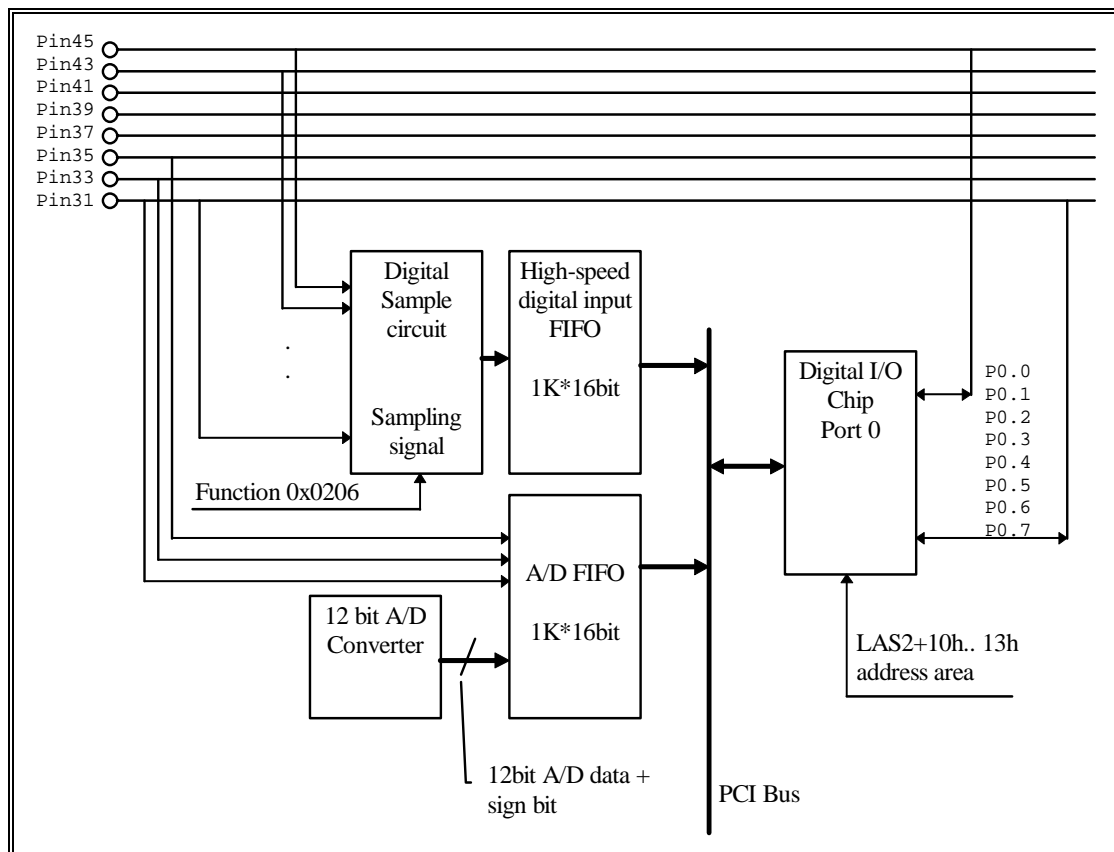


Figure 9.1.

The connection of even numbered pins is shown in the Figure 9.2. These lines can be driven by the digital part of the CGT, therefore these lines can be read or driven by the Port 1 of Digital I/O Chip. The Function 0x0206 enables the Digital Table mode or the Port1 of Digital I/O chip.

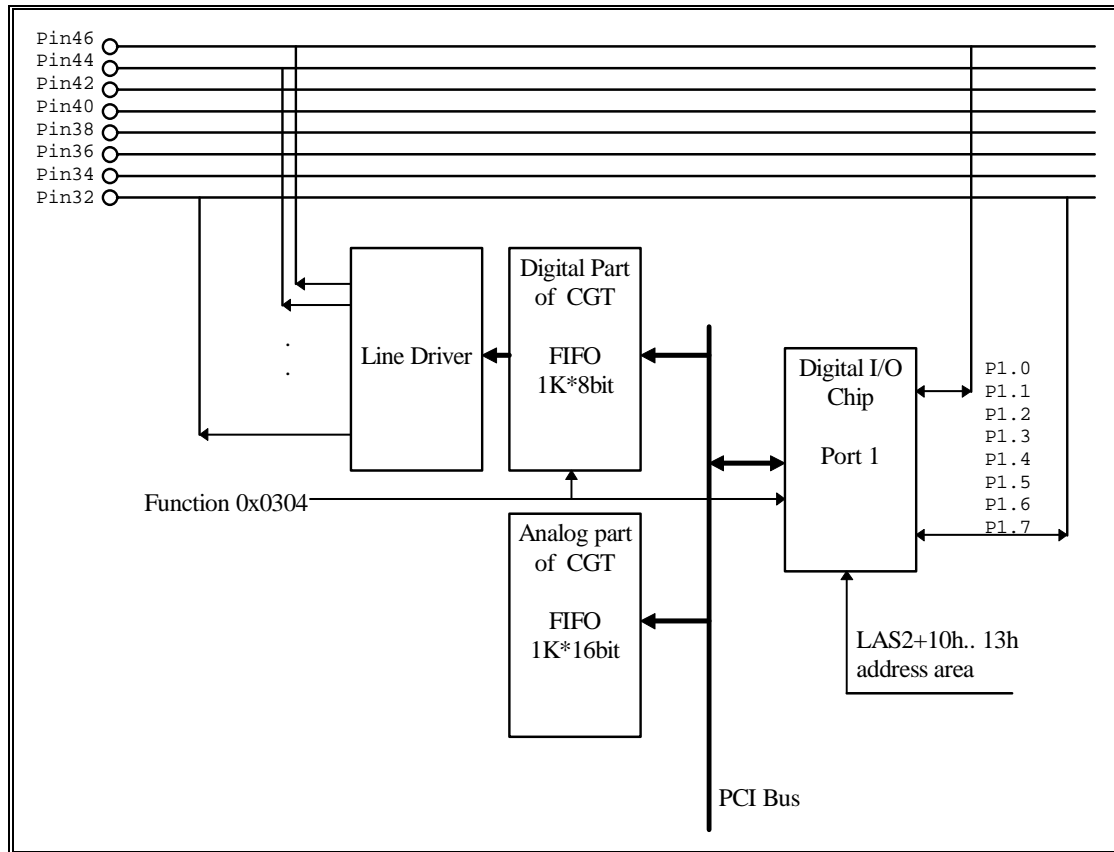


Figure 9.2.

## 8.1. The Digital I/O Chip

The DM7420 has 16 buffered TTL/CMOS digital I/O lines available for digital control applications. These lines are grouped in two 8-bit ports. The sixteen bits in Port 0 can be independently programmed as input or output. Port 1 can be programmed as 8-bit input or output ports. These lines are grouped in digital I/O chip with sixteen lines. The Digital I/O chip is addressed at LAS2+10h.. LAS2+13h.

All digital inputs are pulled up to +5V by 10kOhm resistors. All digital outputs are terminated by series 100Ohm resistors.

### 8.1.1 Port 0, Bit Programmable Digital I/O

The eight Port 0 digital lines are individually set for input or output by writing to the Direction Register at LAS2+12h. The input lines are read and the output lines are written at LAS2+10h.

### 8.1.2. Advanced Digital Interrupts: Mask and Compare Registers

The Port 0 bits support two Advanced Digital Interrupt modes. An interrupt can be generated when the data read at the port matches the value loaded into the Compare Register. This is called a match interrupt. Or, an interrupt can be generated whenever any bit changes state. This is an event interrupt. For either interrupt, bits can be masked by setting the corresponding bit in the Mask Register high. In a digital interrupt mode, this masks out selected bits when monitoring the bit pattern for a match or event. In normal operation where the Advanced Digital Interrupt mode is not activated, the Mask Register can be used to preserve a bit's state, regardless of the digital data written to Port 0.

When using event interrupts, you can determine which bit caused an event interrupt to occur by reading the contents latched into the Compare Register.

### 8.1.3. Port 1, Port Programmable Digital I/O

The direction of the eight bit Port 1 digital lines is programmed at LAS2+13h, bit 2. These lines are configured as all inputs or all outputs, with their states read and written at LAS+11h. In the case lines

### 8.1.4. Resetting the Digital Circuitry

When a digital chip clear (LAS2+13h bits 1 and 0 = 00 followed by a write to LAS2+12h), Software Reset of the board (Function 0x000F), all of the digital I/O lines are set up as inputs.

### 8.1.5. Strobing Data into Port 0

When not in an Advanced Digital Interrupt mode, external data can be strobed into Port 0 by connecting a trigger pulse through the External Pacer Clock pin at the External I/O Connector. This data can be read from the Compare Register at LAS2+12h.

## 8.2. High-Speed Digital Input

As you can see in the Figure 9.1. the Pin 31.. 45 digital pins of external I/O connector can be sampled by high -speed digital input circuitry. The sampling signal can be selected by the Function 0x0206. Samples are written automatically into the high-speed digital input FIFO. The status bits of FIFO can be monitored at the address LAS0+8h.

If you want to get an interrupt at a required number of samples in the FIFO, use User Timer Counter 1 as high speed digital input sample counter. Select the high-speed digital input sampling signal as clock of user TC1 by the Function 0x0702.

All digital inputs are pulled up to +5V by 10kOhm resistors.

## 8.3. Digital Input Data Markers

As you can see in the Figure 9.1. the digital pin 31, 33, 35 can be sampled nearly simultaneously with the analog input signal by the A/D FIFO. The delay time between the analog input sampling and the digital input sampling is app. 800 ns. If you want to sample the digital lines with the analog lines really simultaneously, use the high-speed digital input with A/D Conversion Signal as sampling signal of high-speed digital input.

All digital inputs are pulled up to +5V by 10kOhm resistors.



## 9. Calibration

This chapter tells you how to calibrate the DM7420 using the 7420diag.exe program included in the software package and the trim pots on the board. These trim pots calibrate the A/D converter gain and offset, and the D/A outputs.

This chapter tells you how to calibrate the A/D converter gain and offset, and the D/A output multiplier. The offset and full-scale performance of the board's A/D and D/A converters is factory-calibrated. Any time you suspect inaccurate readings, you can check the accuracy of your conversions using the procedure below, and make adjustments as necessary. Using the 7420diag.exe program is a convenient way to monitor conversions while you calibrate the board.

Calibration is done with the board installed in your system. You can access the trim pots at the top edge of the board. Power up the system and let the board circuitry stabilize for 15 minutes before you start calibrating.

### 9.1. Required Equipment

The following equipment is required for calibration:

- Precision Voltage Source: -10 to +10 volts
- Digital Voltmeter: 5-1/2 digits
- Small Screwdriver (for trimpot adjustment)

While not required, the 7420diag.exe program (included software) is helpful when performing calibrations.

### 9.2. A/D Calibration

Two procedures are used to calibrate the A/D converter for all input voltage ranges. The first procedure calibrates the converter for the bipolar ranges ( $\pm 5$ ,  $\pm 10$  volts), and the second procedure calibrates the unipolar range (0 to +10 volts). Table 12-1 shows the ideal input voltage for each bit weight for the bipolar ranges, and Table 12-2 shows the ideal voltage for each bit weight for the unipolar range.

#### 9.2.1. Bipolar Calibration

##### Bipolar Range Adjustments: -5 to +5 Volts

Two adjustments are made to calibrate the A/D converter for the bipolar range of -5 to +5 volts. One is the offset adjustment, and the other is the full-scale, or gain, adjustment. Trimpot TR4 is used to make the offset adjustment, and trimpot TR5 is used for gain adjustment. Before making these adjustments, make sure that the board is programmed for a range of  $\pm 5$  volts.

Use analog input channel 1 and set it for a gain of 1 while calibrating the board. Connect your precision voltage source to channel 1. Set the voltage source to -1.22070 millivolts, start a conversion, and read the resulting data. Adjust trimpot TR4 until the reading flickers between the values listed in the table below. Next, set the voltage to -4.99878 volts, and repeat the procedure, this time adjusting TR5 until the data flickers between the values in the table.

Data Values for Calibrating Bipolar 10 Volt Range (-5 to +5 volts)		
	Offset (TR4) Input Voltage = -1.22 mV	Converter Gain (TR5) Input Voltage = -4.99878V
A/D Converted Data	0000 0000 0000	1000 0000 0000
	1111 1111 1111	1000 0000 0001

**Bipolar Range Adjustments: -10 to +10 Volts**

To adjust the bipolar 20-volt range (-10 to +10 volts), program the board for  $\pm 10$  volt input range. Then, set the input voltage to +5.0000 volts and adjust TR2 until the output matches the data in the table below.

<b>Data Value for Calibrating Bipolar 20 Volt Range (-10 to +10 volts)</b>	
	<b>TR2 Input Voltage = +5.0000V</b>
<b>A/D Converted Data</b>	0100 0000 0000

Below is a table listing the ideal input voltage for each bit weight for the bipolar ranges.

<b>Table 12-1 A/D Converter Bit Weights, Bipolar</b>			
<b>SIGN</b>	<b>A/D Bit Weight</b>	<b>Ideal Input Voltage (millivolts)</b>	
		<b>-5 to +5 Volts</b>	<b>-10 to +10 Volts</b>
1	1111 1111 1111	-2.44	-4.88
1	1000 0000 0000	-5000.00	-10000.00
0	0100 0000 0000	+2500.00	+5000.00
0	0010 0000 0000	+1250.00	+2500.00
0	0001 0000 0000	+625.00	+1250.00
0	0000 1000 0000	+312.50	+625.00
0	0000 0100 0000	+156.25	+312.50
0	0000 0010 0000	+78.13	+156.25
0	0000 0001 0000	+39.06	+78.13
0	0000 0000 1000	+19.53	+39.06
0	0000 0000 0100	+9.77	+19.53
0	0000 0000 0010	+4.88	+9.77
0	0000 0000 0001	+2.44	+4.88
0	0000 0000 0000	0.00	0.00

### 9.2.3. Unipolar Calibration

One adjustment is made to calibrate the A/D converter for the unipolar range of 0 to +10 volts. Trimpot TR6 is used to make the offset adjustment. This calibration procedure is performed with the module programmed for a 0 to +10 volt input range. Before making these adjustments, make sure that the module is programmed properly and has been calibrated for the bipolar ranges.

Use analog input channel 1 and set it for a gain of 1 while calibrating the board. Connect your precision voltage source to channel 1. Set the voltage source to +1.22070 millivolts, start a conversion, and read the resulting data. Adjust trimpot TR6 until the data flickers between the values listed in the table below.

Data Values for Calibrating Unipolar 10 Volt Range (0 to +10 volts)		
	Offset (TR6) Input Voltage = +1.22070 mV	
<b>A/D Converted Data</b>	0000 0000 0000	0000 0000 0001

Below is a table listing the ideal input voltage for each bit weight for the unipolar range.

Table 12-2 A/D Converter Bit Weights, Unipolar		
SIGN	A/D Bit Weight	Ideal Input Voltage (millivolts)
		0 to +10 Volts
0	1111 1111 1111	+9997.6
0	1000 0000 0000	+5000.0
0	0100 0000 0000	+2500.0
0	0010 0000 0000	+1250.0
0	0001 0000 0000	+625.00
0	0000 1000 0000	+312.50
0	0000 0100 0000	+156.25
0	0000 0010 0000	+78.125
0	0000 0001 0000	+39.063
0	0000 0000 1000	+19.531
0	0000 0000 0100	+9.7656
0	0000 0000 0010	+4.8828
0	0000 0000 0001	+2.4414
0	0000 0000 0000	0

### 9.2.4. Gain Adjustment

Should you find it necessary to check any of the programmable gain settings, the following table will show the proper trimpot to adjust.

<b>Trim pots for Calibrating Gains</b>	
<b>Gain</b>	<b>Trimpot</b>
x2	TR7
x4	TR8
x8	TR9
x16	TR10
x32	TR11

## 10. Specifications

Typical @ 25 C.

### 10.1. Computer Interface

PC/104plus PCI target device

Three I/O mapped address area for configuration and mode control, and data transfer and timer-counter, digital I/O area. The data transfer area is burst accessed for fast data transfer.

The maximum data transfer rate in burst mode is 16Msample/s

### 10.2. Analog Input circuitry

Up to 8 Differential 8 SE with dedicate ground, 16 SE inputs Ground or non ground referenced, software selectable

Input impedance, each channel	>10 MOhm
Gains, Software selectable	1, 2, 4, 8, 16, 32
Gain error	0.05%, typ. max 0.1%
Input ranges	+5, +10, or 0.. 10V
Overvoltage protection	+15V max.
Common mode input voltage	+10V max.
Channel scanning error	see next Table

Scanning Rate (kHz)	Scanning error (% of +-full scale)
10.. 500	0.0000 %
600	0.05% max.

### 10.3. A/D Converter

Type	Successive approximation
Resolution	12bit (2.44mV/4.88mV)
Linearity error	+1 LSB max. +-0.3 LSB typ.
Sampling Rate	1.25MHz max.
-3dB bandwidth (for undersampling application)	20MHz
11 effective bit bandwidth (for undersampling application)	2.5MHz

### 10.4. A/D Sample Buffer

FIFO size	1024x16bits
-----------	-------------

### 10.5. Channel Gain table

Size	1024x24bits
------	-------------

### 10.6. Clocks and Counters

Based on	CMOS 82C54
Pacer Clock period time range	9 minutes to 800 ns
Sample Counter maximum count(1cycle)	65535

### **10.7. Digital I/O**

Input Output type	TTL compatible
Number of lines	8bit bit-programmable, 8bit byte programmable
Isource	-12mA
Isink	24mA
Input termination	10kOhm up to +5V
Output termination	10 Ohm series resistor