AT-MIO-16D
User Manual

Multifunction I/O Board for the PC AT

March 1995 Edition

Part Number 320489-01

National Instruments Corporate Headquarters
6504 Bridge Point Parkway
Austin, TX 78730-5039
(512) 794-0100
Technical support fax: (800) 328-2203
(512) 794-5678

Branch Offices:
Australia (03) 879 9422, Austria (0662) 435986, Belgium 02/757.00.20, Canada (Ontario) (519) 622-9310,
Canada (Québec) (514) 694-8521, Denmark 45 76 26 00, Finland (90) 527 2321, France (1) 48 14 24 24,
Germany 089/741 31 30, Italy 02/48301892, Japan (03) 3788-1921, Mexico 95 800 010 0793,
Netherlands 03480-33466, Norway 32-84 84 00, Singapore 2265886, Spain (91) 640 0085, Sweden 08-730 49 70,
Switzerland 056/20 51 51, Taiwan 02 377 1200, U.K. 0635 523545

Limited Warranty

The AT-MIO-16D is warranted against defects in materials and workmanship for a period of one year from the date of shipment, as evidenced by receipts or other documentation. National Instruments will, at its option, repair or replace equipment that proves to be defective during the warranty period. This warranty includes parts and labor.

The media on which you receive National Instruments software are warranted not to fail to execute programming instructions, due to defects in materials and workmanship, for a period of 90 days from date of shipment, as evidenced by receipts or other documentation. National Instruments will, at its option, repair or replace software media that do not execute programming instructions if National Instruments receives notice of such defects during the warranty period. National Instruments does not warrant that the operation of the software shall be uninterrupted or error free.

A Return Material Authorization (RMA) number must be obtained from the factory and clearly marked on the outside of the package before any equipment will be accepted for warranty work. National Instruments will pay the shipping costs of returning to the owner parts which are covered by warranty.

National Instruments believes that the information in this manual is accurate. The document has been carefully reviewed for technical accuracy. In the event that technical or typographical errors exist, National Instruments reserves the right to make changes to subsequent editions of this document without prior notice to holders of this edition. The reader should consult National Instruments if errors are suspected. In no event shall National Instruments be liable for any damages arising out of or related to this document or the information contained in it.

EXCEPT AS SPECIFIED HEREIN, NATIONAL INSTRUMENTS MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AND SPECIFICALLY DISCLAIMS ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. CUSTOMER'S RIGHT TO RECOVER DAMAGES CAUSED BY FAULT OR NEGLIGENCE ON THE PART OF NATIONAL INSTRUMENTS SHALL BE LIMITED TO THE AMOUNT THERETOFORE PAID BY THE CUSTOMER. NATIONAL INSTRUMENTS WILL NOT BE LIABLE FOR DAMAGES RESULTING FROM LOSS OF DATA, PROFITS, USE OF PRODUCTS, OR INCIDENTAL OR CONSEQUENTIAL DAMAGES, EVEN IF ADVISED OF THE POSSIBILITY THEREOF. This limitation of the liability of National Instruments will apply regardless of the form of action, whether in contract or tort, including negligence. Any action against National Instruments must be brought within one year after the cause of action accrues. National Instruments shall not be liable for any delay in performance due to causes beyond its reasonable control. The warranty provided herein does not cover damages, defects, malfunctions, or service failures caused by owner's failure to follow the National Instruments installation, operation, or maintenance instructions; owner's modification of the product; owner's abuse, misuse, or negligent acts; and power failure or surges, fire, flood, accident, actions of third parties, or other events outside reasonable control.

Copyright

Trademarks

LabVIEW®, NI-DAQ®, and RTSI® are trademarks of National Instruments Corporation.

Product and company names listed are trademarks or trade names of their respective companies.

WARNING REGARDING MEDICAL AND CLINICAL USE OF NATIONAL INSTRUMENTS PRODUCTS

National Instruments products are not designed with components and testing intended to ensure a level of reliability suitable for use in treatment and diagnosis of humans. Applications of National Instruments products involving medical or clinical treatment can create a potential for accidental injury caused by product failure, or by errors on the part of the user or application designer. Any use or application of National Instruments products for or involving medical or clinical treatment must be performed by properly trained and qualified medical personnel, and all traditional medical safeguards, equipment, and procedures that are appropriate in the particular situation to prevent serious injury or death should always continue to be used when National Instruments products are being used. National Instruments products are NOT intended to be a substitute for any form of established process, procedure, or equipment used to monitor or safeguard human health and safety in medical or clinical treatment.

Preface

This manual describes the electrical and mechanical aspects of the AT-MIO-16D and contains information concerning its operation and programming. The AT-MIO-16D, a member of the National Instruments AT Series of expansion boards for the IBM PC AT and compatible computers, combines the functionality of two popular National Instruments boards, the AT-MIO-16 and the PC-DIO-24. The AT-MIO-16D contains two logical sections–the MIO-16 circuitry and the DIO-24 circuitry. The MIO-16 circuitry contains a 12-bit ADC with up to 16 analog inputs, two 12-bit DACs with voltage outputs, eight lines of transistor-transistor logic (TTL) compatible digital I/O, and three 16-bit counter/timer channels for timing I/O. The DIO-24 circuitry is a 24-bit parallel, digital I/O interface based on an 82C55A programmable peripheral interface (PPI). If you require signal conditioning or additional analog inputs, you can use the SCXI signal conditioning modules, the SCXI multiplexer products, or the AMUX-64T multiplexer board.

Organization of This Manual

The *AT-MIO-16D User Manual* is organized as follows:

- Chapter 1, *Introduction*, describes the AT-MIO-16D; lists the contents of your AT-MIO-16D kit, the optional software, and optional equipment; and explains how to unpack the AT-MIO-16D.
- Chapter 2, *Configuration and Installation*, describes the AT-MIO-16D jumper configuration, installation of the AT-MIO-16D board into the PC, signal connections to the AT-MIO-16D board, cable wiring, and handshake timing diagrams for the DIO-24 circuitry of the AT-MIO-16D.
- Chapter 3, *Theory of Operation*, contains a functional overview of the AT-MIO-16D and explains the operation of each functional unit making up the AT-MIO-16D.
- Chapter 4, *Programming*, discusses the programming of the AT-MIO-16D. Included in this chapter are the AT-MIO-16D register address map, a detailed register description, and a functional programming description.
- Chapter 5, *Calibration Procedures*, discusses the calibration procedures for the AT-MIO-16D analog input and analog output circuitry.
- Appendix A, *Specifications*, lists the specifications for the AT-MIO-16D.
- Appendix B, *MIO-16 I/O Connector*, describes the pinout and signal names for the MIO-16 50-pin I/O connector of the AT-MIO-16D.
- Appendix C, *DIO-24 I/O Connector*, describes the pinout and signal names for the DIO-24 50-pin I/O connector of the AT-MIO-16D.
- Appendix D, *AT-MIO-16D I/O Connector*, describes the pinout and signal names for the AT-MIO-16D 100-pin I/O connector.

- Appendix E, *AMD Am9513A Data Sheet*, contains the manufacturer data sheet for the Am9513A System Controller integrated circuit (Advanced Micro Devices, Inc.). This device is used on the AT-MIO-16D.
- Appendix F, *Oki MSM82C55A Data Sheet*, contains the manufacturer data sheet for the MSM82C55A CMOS Programmable Peripheral Interface (Oki Semiconductor). This device is used on the AT-MIO-16D.
- Appendix G, *Customer Communication*, contains forms for you to complete to facilitate communication with National Instruments concerning our products.
- The *Index* alphabetically lists topics covered in this manual, including the page where the topic can be found.

Conventions Used in This Manual

The following conventions are used to distinguish elements of text throughout this manual:

italic	Italic text denotes emphasis, a cross reference, or an introduction to a key concept.
PC	PC refers to the IBM PC AT and compatible computers.

Abbreviations

The following metric system prefixes are used with abbreviations for units of measure in this manual:

Prefix	Meaning	Value
p-	pico-	10^{-12}
n-	nano-	10^{-9}
µ-	micro-	10^{-6}
m-	milli-	10^{-3}
k-	kilo-	10^{3}
M-	mega-	10^{6}
G-	giga-	10^{9}

The following abbreviations are used in this manual:

A	amperes
dB	decibels
ft	feet
hex	hexadecimal
Hz	hertz
kbytes	1,000 bytes
ksamples	1,000 samples
M	megabytes of memory

Abbreviations (continued)

m	meters
Ω	ohms
ppm	parts per million
sec	seconds
V	volts
Vrms	volts, root mean square

Acronyms

The following acronyms are used in this manual:

AC	alternating current
A/D	analog-to-digital
ADC	A/D converter
D/A	digital-to-analog
DAC	D/A converter
DIP	dual inline package
DMA	direct memory access
FIFO	first-in-first-out
I/O	input/output
LS	low-power Schottky
LSB	least significant bit
MSB	most significant bit
PPI	programmable peripheral interface
RTSI	Real-Time System Integration
SSR	solid-state relays
TTL	transistor-transistor logic
VDC	volts direct current

Related Documentation

The following document contains information that you may find helpful as you read this manual:

- *IBM Personal Computer AT Technical Reference* manual

You may also want to consult the following Advanced Micro Devices manual if you plan to program the Am9513A Counter/Timer used on the AT-MIO-16D:

- *Am9513A/Am9513 System Timing Controller* technical manual

Customer Communication

National Instruments wants to receive your comments on our products and manuals. We are interested in the applications you develop with our products, and we want to help if you have problems with them. To make it easy for you to contact us, this manual contains comment and configuration forms for you to complete. These forms are in Appendix G, *Customer Communication*, at the end of this manual.

Contents

Figures

Tables

Chapter 1
Introduction

This chapter describes the AT-MIO-16D; lists the contents of your AT-MIO-16D kit, the optional software, and optional equipment; and explains how to unpack the AT-MIO-16D.

The AT-MIO-16D combines the functionality of two popular National Instruments boards, the AT-MIO-16 and the PC-DIO-24. The AT-MIO-16D contains two logical sections–the MIO-16 circuitry and the DIO-24 circuitry. Henceforth, we will refer to the entire board as the AT-MIO-16D, and to a particular logical part of the board as either the MIO-16 or DIO-24 circuitry. The MIO-16 circuitry contains a 12-bit ADC with up to 16 analog inputs, two 12-bit DACs with voltage outputs, eight lines of transistor-transistor logic (TTL) compatible digital I/O, and three 16-bit counter/timer channels for timing I/O. The DIO-24 circuitry is a 24-bit parallel, digital I/O interface based on an 82C55A programmable peripheral interface (PPI).

The MIO-16 circuitry of the AT-MIO-16D is a high-performance multifunction analog, digital, and timing I/O circuit for the PC. The AT-MIO-16D has a fast 12-bit ADC, 16 single-ended or eight differential channels (expandable with SCXI and the AMUX-64T), and programmable gains of 1, 10, 100, and 500 or 1, 2, 4, and 8. The AT-MIO-16D has a 9-µsec converter, guaranteed transfer rates of up to 100 ksamples/sec, and a 512-word A/D FIFO buffer to obtain the highest possible data acquisition rate. The AT-MIO-16D has internal or external A/D timing, two double-buffered multiplying 12-bit DACs, unipolar or bipolar voltage output, and an onboard DAC reference voltage of 10 V. The AT-MIO-16D also has onboard timers for waveform generation, eight digital I/O lines that can sink up to 24 mA of current, and three independent 16-bit counter/timers for frequency counting, event counting, and pulse output applications. The AT-MIO-16D has timer-generated interrupts, a high-performance RTSI bus interface with four triggers for system-level timing, and full PC AT I/O channel DMA capability.

The DIO-24 circuitry of the AT-MIO-16D is a 24-bit parallel, digital I/O interface for the PC. An 82C55A PPI controls the 24 bits of digital I/O. The 82C55A is very flexible and powerful when interfacing with peripheral equipment, can operate in either a unidirectional or bidirectional mode, and can generate interrupt request outputs. You can program the 82C55A for almost any 8-bit or 16-bit digital I/O application. The 100-pin connector of the AT-MIO-16D breaks out into two standard 50-pin female connectors via a cable assembly. The pin assignments for these connectors are compatible with standard 24-channel digital I/O applications.

Figure 1-1 shows the AT-MIO-16D interface board.

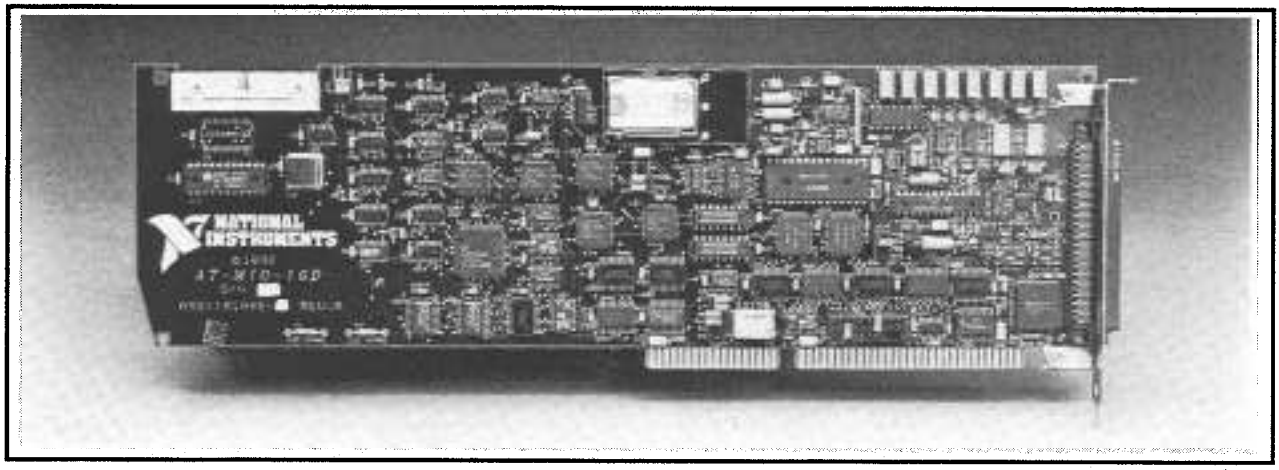


Figure 1-1. AT-MIO-16D Interface Board

You can use the AT-MIO-16D, with its multifunction analog, digital, and timing I/O, in many applications, including the automation of machine and process control, level monitoring and control, instrumentation, electronic testing, and many others. You can use the multichannel analog input for signal and transient analysis, data logging, and chromatography. The two analog output channels are useful for machine and process control, analog function generation, 12-bit resolution voltage source, and programmable signal attenuation. You can use the eight TTL-compatible digital I/O lines for machine and process control, intermachine communication, and relay switching control. The three 16-bit counter/timers can be used for such functions as pulse and clock generation, timed control of laboratory equipment, and frequency, event, and pulse-width measurement. With all these functions on one board, you can automatically monitor and control laboratory processes.

The additional digital I/O of the AT-MIO-16D DIO-24 circuitry is useful for a wide range of digital I/O applications. With the DIO-24 circuitry and I/O connector, you can interface a PC to any of the following:

- Other computers
  - Another PC with a National Instruments PC-DIO-24, AT-DIO-32F, or AT-MIO-16D
  - IBM Personal System/2 with a National Instruments MC-DIO-24 or MC-DIO-32F
  - Apple Macintosh II or Quadra with a National Instruments NB-DIO-24 or NB-DIO-32F
  - Any other computer with an 8-bit or 16-bit parallel interface
- Centronics-compatible printers and plotters
- Panel meters
- Instruments and test equipment with BCD readouts and/or controls
- Opto-isolated solid-state relays (SSRs) and I/O module mounting racks

With the AT-MIO-16D, the PC can serve as a digital I/O system controller for laboratory testing, production testing, and industrial process monitoring and control.

The AT-MIO-16D is interfaced to the National Instruments RTSI bus. With this bus, National Instruments AT Series boards can send timing signals to each other. The AT-MIO-16D can send signals from the onboard counter/timer to another board, or another board can control single and multiple A/D conversions on the AT-MIO-16D.

The AT-MIO-16D is available in two gain ranges. The AT-MIO-16DL-9 has software-programmable gain settings of 1, 10, 100, and 500 for low-level analog input signals. The AT-MIO-16DH-9 has software-programmable gain settings of 1, 2, 4, and 8 for high-level analog input signals. The AT-MIO-16D contains an ADC with a 9-µsec conversion time, and is capable of data acquisition rates of up to 100 kbytes/sec.

Detailed specifications for the AT-MIO-16D are listed in Appendix A, *Specifications*.

What Your Kit Should Contain

Each version of the AT-MIO-16D board has a different part number and kit part number, listed as follows.

Kit Name	Kit Part Number	Kit Component	Board Part Number
AT-MIO-16DL-9	776646-01	AT-MIO-16DL-9 board	181965-01
AT-MIO-16DH-9	776646-11	AT-MIO-16DH-9 board	181965-11

The board part number is printed on your board along the top edge on the component side. You can identify which version of the AT-MIO-16D board you have by looking up the part number in the preceding table.

In addition to the board, each version of the AT-MIO-16D kit contains the following components.

Kit Component	Part Number
AT-MIO-16D User Manual	320489-01
NI-DAQ software for DOS/Windows/LabWindows, with manuals	776250-01
NI-DAQ Software Reference Manual for DOS/Windows/LabWindows	320498-01
NI-DAQ Function Reference Manual for DOS/Windows/LabWindows	320499-01

If your kit is missing any of the components or if you received the wrong version, contact National Instruments.

Your AT-MIO-16D is shipped with the NI-DAQ software for DOS/Windows/LabWindows. NI-DAQ has a library of functions that can be called from your application programming environment. These functions include routines for analog input (A/D conversion), buffered data acquisition (high-speed A/D conversion), analog output (D/A conversion), waveform generation, digital I/O, counter/timer, SCXI, RTSI, and self-calibration. NI-DAQ maintains a consistent software interface among its different versions so you can switch between platforms with minimal modifications to your code. NI-DAQ comes with language interfaces for Professional BASIC, Turbo Pascal, Turbo C, Turbo C++, Borland C++, and Microsoft C for DOS; and Visual Basic, Turbo Pascal, Microsoft C with SDK, and Borland C++ for Windows. NI-DAQ software is on high-density 5.25 in. and 3.5 in. diskettes.

Optional Software

This manual contains complete instructions for directly programming the AT-MIO-16D. Normally, however, you should not need to read the low-level programming details in the user manual because the NI-DAQ software package for controlling the AT-MIO-16D is included with the board. Using NI-DAQ is quicker and easier than and as flexible as using the low-level programming described in Chapter 4, *Programming*.

You can use the AT-MIO-16D with LabVIEW for Windows or LabWindows for DOS. LabVIEW and LabWindows are innovative program development software packages for data acquisistion and control applications. LabVIEW uses graphical programming, whereas LabWindows enhances Microsoft C and QuickBASIC. Both packages include extensive libraries for data acquisition, instrument control, data analysis, and graphical data presentation.

Part numbers for these software packages are listed in the following table.

Software	Part Number
LabVIEW for Windows	776670-01
LabWindows	
Standard package	776473-01
Advanced Analysis Library	776474-01
Standard package with the Advanced Analysis Library	776475-01

Optional Equipment

Equipment		Part Number
CB-100 I/O connector block		
0.5-m cable		776455-01
1.0-m cable		776455-02
Type NB5 100-conductor ribbon cable		
0.5-m cable		181304-05
1.0-m cable		181304-10
SCXI signal conditioning modules		
SCXI-1100 32-channel differential multiplexer/amplifier		776572-00
SCXI-1120 8-channel isolated analog input		776572-20
SCXI-1121 4-channel isolated transducer amplifier with excitation		776572-21
SCXI-1140 8-channel simultaneously sampling differential amplifier		776572-40
SCXI-1180 feedthrough panel		776572-80
SCXI-1181 breadboard		776572-81
AMUX-64T analog multiplexer board without cable		776366-90
with 0.2-m ribbon cable		776366-02
with 0.5-m ribbon cable		776366-05
with 1.0-m ribbon cable		776366-10
with 2.0-m ribbon cable		776366-20
AT Series RTSI bus cables for		
2 boards		776249-02
3 boards		776249-03
4 boards		776249-04
5 boards		776249-05
Cable adapter board for signal conditioning		
SC-2050 without cable		776335-90
SC-2051 without cable		776335-91
SC-2060 optically isolated digital input board with conductor cable	0.2 m	776336-00
	0.4 m	776336-10
SC-2061 optically isolated digital output board		
with 26-conductor cable	0.2 m	776336-01
	0.4 m	776336-11
SC-2062 electromechanical relay digital control board		
with 26-conductor cable	0.2 m	776336-02
	0.4 m	776336-12
General-purpose termination breadboard		
SC-2070 without cable		776358-90
SC-2072 without cable		776358-92
SC-2072D without cable		776358-192
BNC-2080 BNC adapter board without cable		776579-90
Digital signal conditioning modules		
SSR Series mounting rack and 1.0 m cable		
24-channel without cable		776290-924
16-channel without cable		776290-916
8-channel without cable		776290-908
8-channel with SC-205X cable		776290-18

Custom Cables

The AT-MIO-16D I/O connector is a 100-pin male ribbon cable header. The manufacturer part number National Instruments uses for this header is as follows:

- Robinson Nugent (part number P50E-100P1-SR1-TG)

The mating connector for the board is a 100-position, polarized, ribbon socket connector. This connector breaks out into two 50-pin female connectors with 50-conductor ribbon cables via a cable assembly. National Instruments uses a keyed connector to prevent inadvertent upside-down connection to the board. The recommended manufacturer part number for this mating connector is as follows:

- Robinson Nugent (part number P50E-100S-TG)

Figure 1-2 shows the AT-MIO-16D cable assembly.

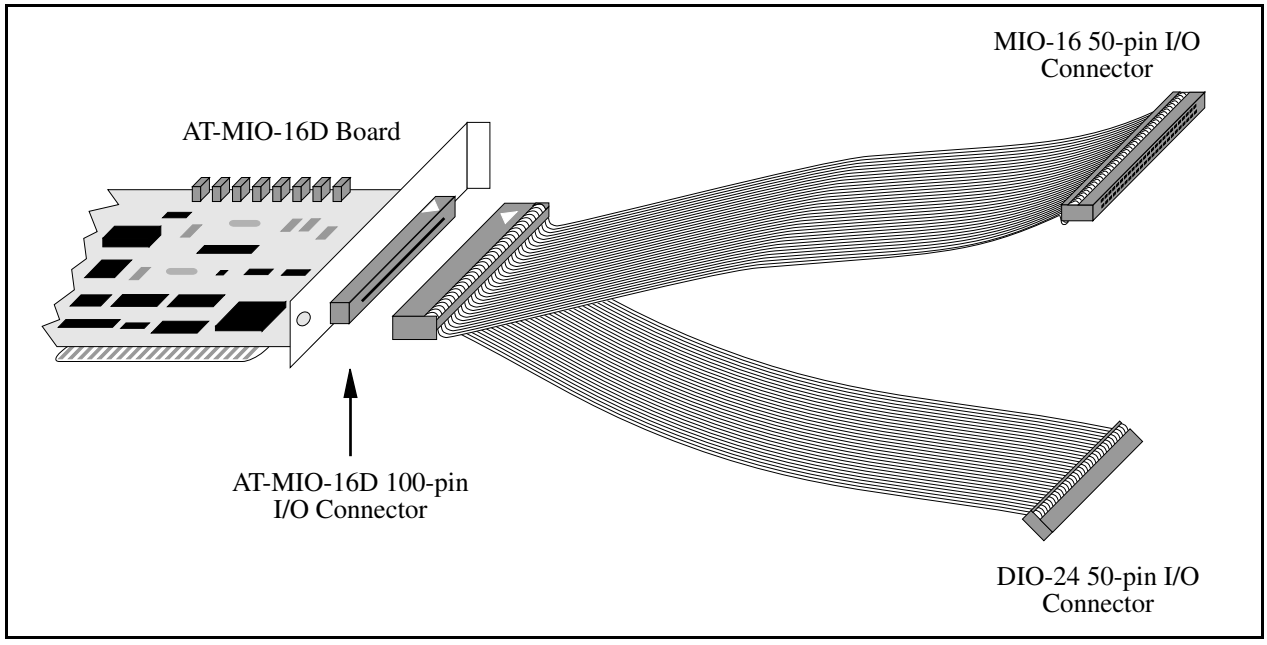


Figure 1-2. AT-MIO-16D Cable Assembly

Recommended manufacturer part numbers for standard ribbon cable (50-conductor, 28 AWG, stranded) that can be used with the mating connector are as follows:

- Electronic Products Division/3M (part number 3365/50)
- T&B/Ansley Corporation (part number 171-50)

Recommended manufacturer part numbers for the 50-pin edge connector for connecting to a module rack with an edge connector are as follows:

- Electronic Products Division/3M (part number 3415-0001)
- T&B Ansley Corporation (part number 609-5015M)

You can plug a polarizing key into these edge connectors to prevent inadvertent upside-down connection to the I/O module rack. The location of this key varies from rack to rack. Consult the specification for the rack you intend to use for the location of any polarizing key. The recommended manufacturer part numbers for this polarizing key are as follows:

- Electronic Products Division/3M (part number 3439-2)
- T&B Ansley Corporation (part number 609-0005)

Unpacking

Your AT-MIO-16D board is shipped in an antistatic plastic bag to prevent electrostatic damage to the board. Several components on the board can be damaged by electrostatic discharge. To avoid such damage in handling the board, take the following precautions:

- Touch the plastic bag to a metal part of your PC chassis before removing the board from the bag.
- Remove the board from the bag and inspect the board for loose components or any other sign of damage. Notify National Instruments if the board appears damaged in any way. *Do not* install a damaged board into your computer.

Chapter 2
Configuration and Installation

This chapter describes the AT-MIO-16D jumper configuration, installation of the AT-MIO-16D board into the PC, signal connections to the AT-MIO-16D board, cable wiring, and handshake timing diagrams for the DIO-24 circuitry of the AT-MIO-16D.

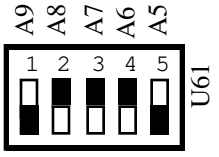
Board Configuration

The AT-MIO-16D contains 14 jumpers and one dual inline package (DIP) switch to configure the AT bus interface and analog input/output (I/O) settings. The DIP switch is used to set the base I/O address. Three jumpers are used as interrupt and direct memory access (DMA) selectors. The remaining 11 jumpers are used to change the analog input and analog output circuitry. The jumpers are shown in the parts locator diagram in Figure 2-1. Jumpers W1, W4, W6, and W9 configure the analog input circuitry. Jumpers W2, W3, W7, W8, W10, and W11 configure the analog output circuitry. Jumper W5 selects the clock signal used by the Am9513A Counter/Timer and the clock pin on the Real-Time System Integration (RTSI) bus. Jumpers W12 and W13 select the DMA channel and the interrupt level, respectively. Jumper W14 selects the DIO-24 circuitry interrupt enable line.

AT Bus Interface

The AT-MIO-16D is configured at the factory to a base I/O address of hex 220, to use DMA channels 6 and 7, to use interrupt level 10 for the MIO-16 circuitry, and to use interrupt enable line PC4 with interrupt level 5 for the DIO-24 circuitry. These settings, as shown in Table 2-1, are suitable for most systems. However, if your system has other hardware at this base I/O address, DMA channel, or interrupt level, you will need to change these settings on the other hardware or on the AT-MIO-16D as described in the following pages.

Table 2-1. AT Bus Interface Factory Settings

Base I/O Address	Hex 220	 (The shaded portion indicates the side of the base address switch that is pressed down.)
DMA Channel	DMA 1 = DMA Channel 6 DMA 2 = DMA Channel 7	W12**:** R6: A-B A6: A-B W12**:** R7: B-C A7: B-C
Interrupt Level	Interrupt levels 5 and 10 selected	W13**:** IRQ 10 (MIO-16) IRQ 5 (DIO-24)
DIO Interrupt Enable Line	PC4	W14: Row PC4

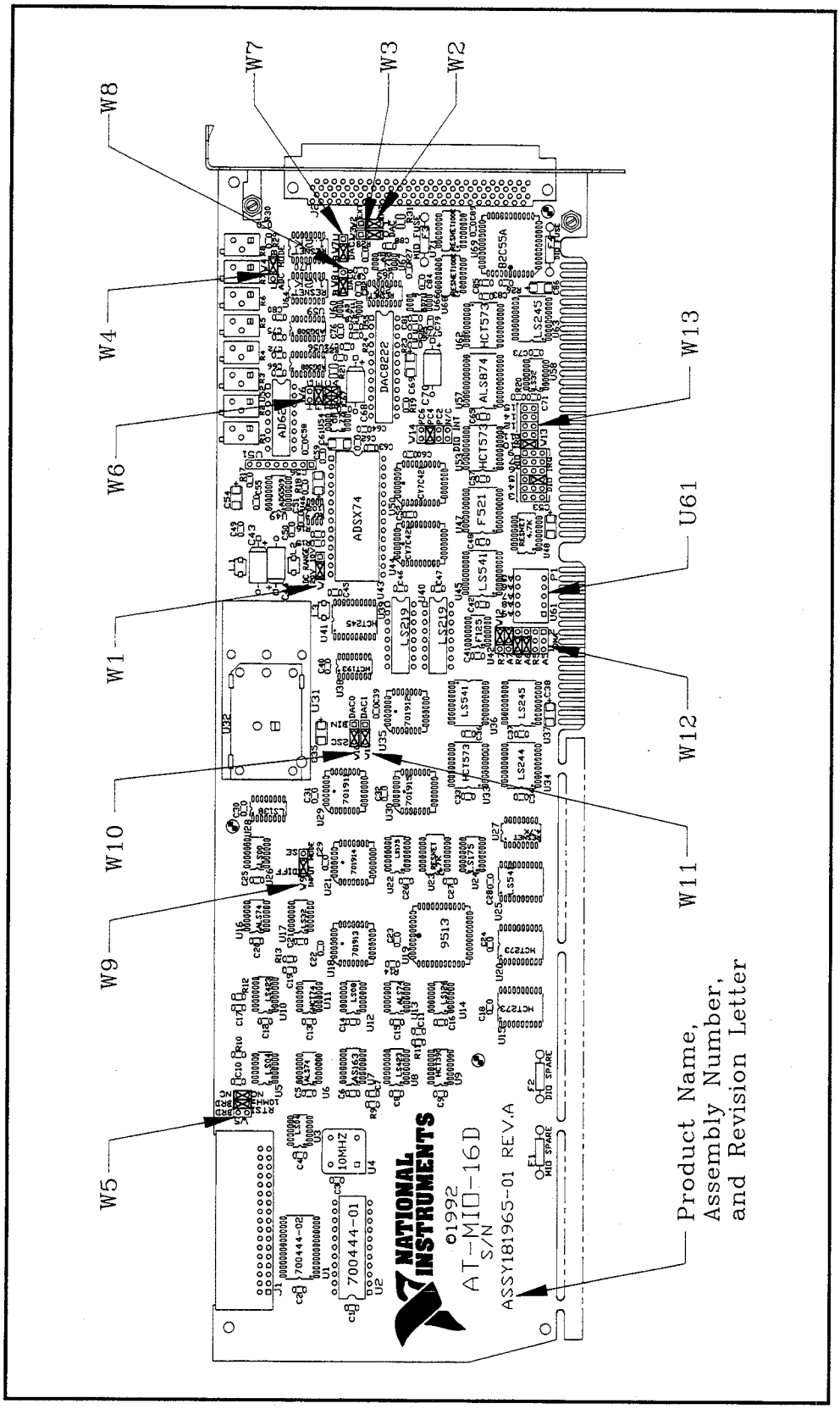


Figure 2-1. Parts Locator Diagram

Base I/O Address Selection

The base I/O address for the AT-MIO-16D is determined by the switches at position U61 as shown in Figure 2-1. The switches are set at the factory for the base I/O address hex 220. This factory setting is used as the default base I/O address value by National Instruments software packages for use with the AT-MIO-16D. The AT-MIO-16D uses the base I/O address space hex 220 through 23F with the factory setting.

Note: Verify that this space is not already used by other equipment installed in your computer. If any equipment in your computer uses this base I/O address space, you must change the base I/O address of the AT-MIO-16D or of the other device. If you change the AT-MIO-16D base I/O address, you must make a corresponding change to any software packages you use with the AT-MIO-16D. Table 2-2 lists the default settings of other National Instruments products for the PC AT. For more information about the I/O address of your PC AT, refer to the technical reference manual for your computer.

Table 2-2. Default Settings of Other National Instruments Products for the PC

Board	DMA Channel	Interrupt Level	Base I/O Address
AT-A2150	None*	None*	120 hex
AT-AO-6/10	Channel 5	Lines 11, 12	1C0 hex
AT-DIO-32F	Channels 5, 6	Lines 11, 12	240 hex
AT-DSP2200	None*	None*	120 hex
AT-GPIB	Channel 5	Line 11	2C0 hex
AT-MIO-16	Channels 6, 7	Line 10	220 hex
AT-MIO-16D	Channels 6, 7	Line 5, 10	220 hex
AT-MIO-16F-5	Channels 6, 7	Line 10	220 hex
AT-MIO-16X	None*	None*	220 hex
AT-MIO-64F-5	None*	None*	220 hex
GPIB-PCII	Channel 1	Line 7	2B8 hex
GPIB-PCIIA	Channel 1	Line 7	02E1 hex
GPIB-PCIII	Channel 1	Line 7	280 hex
Lab-PC	Channel 3	Line 5	260 hex
PC-DIO-24	None	Line 5	210 hex
PC-DIO-96	None	Line 5	180 hex
PC-LPM-16	None	Line 5	260 hex
PC-TIO-10	None	Line 5	1A0 hex

*These settings are software configurable and are set to default at startup time.

Each switch in U61 corresponds to one of the address lines A9 through A5. Press the side marked OFF to select a binary value of 1 for the corresponding address bit. Press the other side of the switch to select a binary value of 0 for the corresponding address bit. Figure 2-2 shows two possible switch settings. The shaded portion indicates the side of the switch that is pressed down.

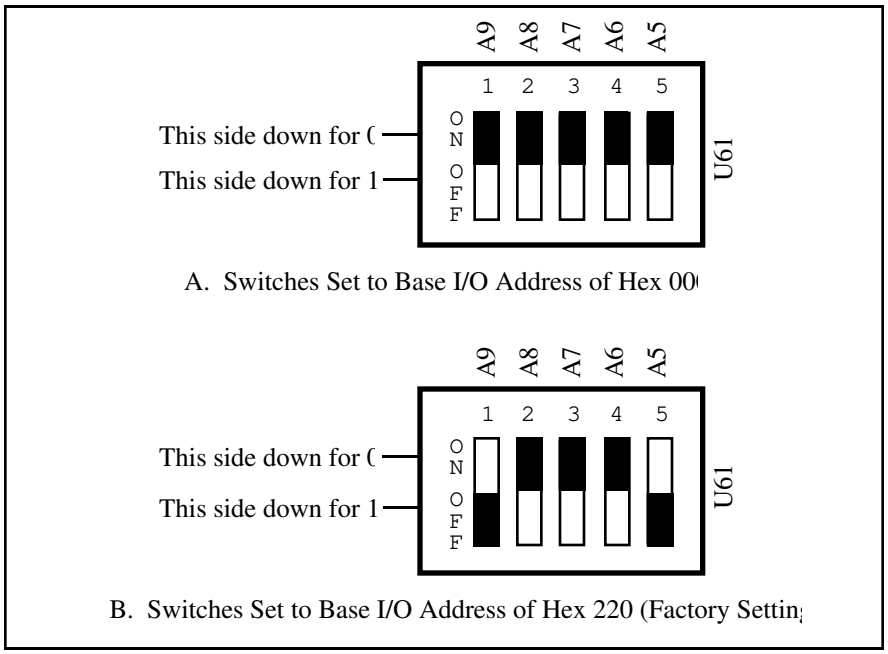


Figure 2-2. Example Base I/O Address Switch Settings

The five least significant bits (LSBs) of the address (A4 through A0) are decoded by the AT-MIO-16D to select the appropriate AT-MIO-16D register. To change the base I/O address, remove the plastic cover on U61; press each switch to the desired position; check each switch to make sure the switch is pressed down all the way; and replace the plastic cover. Make a note of the new AT-MIO-16D base I/O address for use when configuring the AT-MIO-16D software (a form is provided for you in Appendix G, *Customer Communication*). Table 2-3 lists the possible switch settings, the corresponding base I/O address, and the base I/O address space used for that setting.

Table 2-3. Switch Settings with Corresponding Base I/O Address and Base I/O Address Space

Switch Setting A9	A8	A7	A6	A5	Base I/O Address (hex)	Base I/O Address Space Used (hex)
0	0	*X*	*X*	*X*	000-0E0	Reserved
0	1	0	0	0	100	100 - 11F
0	1	0	0	1	120	120 - 13F
0	1	0	1	0	140	140 - 15F
0	1	0	1	1	160	160 - 17F
0	1	1	0	0	180	180 - 19F
0	1	1	0	1	1A0	1A0 - 1BF
0	1	1	1	0	1C0	1C0 - 1DF
0	1	1	1	1	1E0	1E0 - 1FF
1	0	0	0	0	200	200 - 21F
1	0	0	0	1*	220*	220 - 23F*
1	0	0	1	0	240	240 - 25F
1	0	0	1	1	260	260 - 27F
1	0	1	0	0	280	280 - 29F
1	0	1	0	1	2A0	2A0 - 2BF
1	0	1	1	0	2C0	2C0 - 2DF
1	0	1	1	1	2E0	2E0 - 2FF
1	1	0	0	0	300	300 - 31F
1	1	0	0	1	320	320 - 33F
1	1	0	1	0	340	340 - 35F
1	1	0	1	1	360	360 - 37F
1	1	1	0	0	380	380 - 39F
1	1	1	0	1	3A0	3A0 - 3BF
1	1	1	1	0	3C0	3C0 - 3DF
1	1	1	1	1	3E0	3E0 - 3FF

*This setting is the factory default setting.

DMA Channel Selection

The DMA channel used by the AT-MIO-16D is selected by jumpers on W12 as shown in Figure 2-1. The AT-MIO-16D is set at the factory to use DMA channels 6 and 7 for dual DMA mode. These are the default DMA channels used by the AT-MIO-16D software handler. Verify that these DMA channels are not also used by equipment already installed in your computer. If any device uses DMA channel 6 and/or channel 7, change the DMA channel used by either the AT-MIO-16D or the other device. The DMA channels supported by the AT-MIO-16D hardware are channels 5, 6, and 7. Notice that these are the three 16-bit channels on the PC AT I/O channel. The AT-MIO-16D *does not* use and *cannot* be configured to use the 8-bit DMA channels on the PC AT I/O channel.

Each DMA channel consists of two signal lines as shown in Table 2-4.

Table 2-4. DMA Channels for the AT-MIO-16D

DMA Channel	DMA Acknowledge	DMA Request
5	DACK5 (A5)	DRQ5 (R5)
6	DACK6 (A6)	DRQ6 (R6)
7	DACK7 (A7)	DRQ7 (R7)

Two jumpers must be installed to select a DMA channel. The DMA Acknowledge and DMA Request lines selected must have the same number suffix for proper operation. When you use dual DMA mode, the left two rows of W12 are used for DMA 1 and the right two rows of W12 are used for DMA 2. Figure 2-3 displays the jumper positions for selecting DMA channels 6 and 7. In this setting, DMA 1 uses DMA channel 6 and DMA 2 uses DMA channel 7.

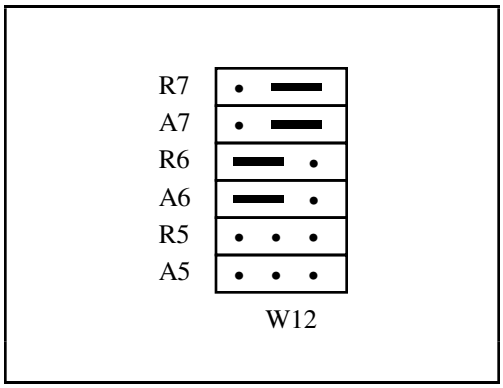


Figure 2-3. DMA Jumper Settings for DMA Channels 6 and 7 (Factory Setting)

If you want to use only one DMA channel, then place the configuration jumpers on W12 in the position shown in Figure 2-4.

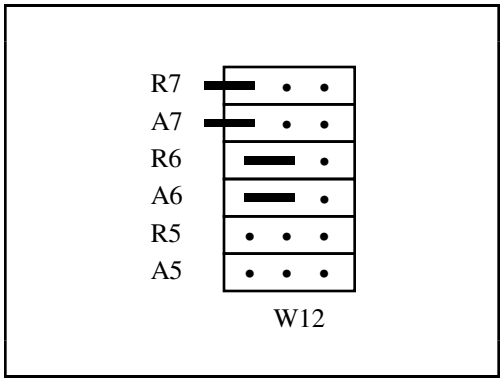


Figure 2-4. DMA Jumper Settings for DMA Channel 6 Only

If you do not want to use DMA for AT-MIO-16D transfers, then place the configuration jumpers on W12 in the position shown in Figure 2-5.

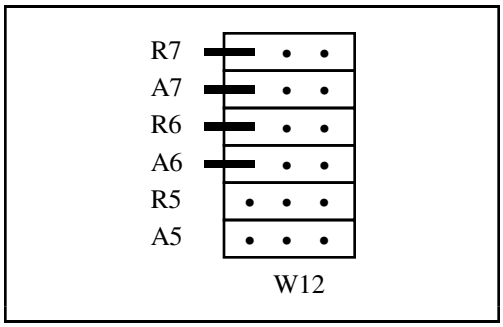


Figure 2-5. DMA Jumper Settings for Disabling DMA Transfers

Interrupt Selection

The AT-MIO-16D board can connect to any of the 11 interrupt lines of the PC AT I/O channel. The interrupt lines for the MIO-16 and DIO-24 circuitry are selected by jumpers on one of the rows of pins located above the I/O slot edge connector on the AT-MIO-16D (refer to Figure 2-1). To use the interrupt capability of the AT-MIO-16D, you must select an interrupt line and place the jumper in the appropriate position to enable that particular interrupt line.

The AT-MIO-16D can share interrupt lines with other devices by using a tristate driver to drive its selected interrupt line. The interrupt lines supported by the AT-MIO-16D hardware for the MIO-16 circuitry are IRQ3, IRQ4, IRQ5, IRQ6, IRQ7, IRQ9, IRQ10, IRQ11, IRQ12, IRQ14, and IRQ15. The interrupt lines supported by the AT-MIO-16D hardware for the DIO-24 circuitry are IRQ3, IRQ4, IRQ5, IRQ6, IRQ7, IRQ9.

Note: *Do not* use interrupt line 6 or interrupt line 14. Interrupt line 6 is used by the diskette drive controller, and interrupt line 14 is used by the hard disk controller on most IBM PC ATs and compatibles.

Once you have selected an interrupt level, place the interrupt jumper on the appropriate pins to enable the interrupt line.

The interrupt jumper set is W13. The default interrupt lines are IRQ10 for the MIO-16 circuitry and IRQ5 for the DIO-24 circuitry, which are selected by placing the jumpers on the pins in rows 5 and 10. Figure 2-6 shows the default interrupt jumper settings IRQ5 and IRQ10. To change to another line, remove the jumper from IRQ5 or IRQ10 and place it on the new pins.

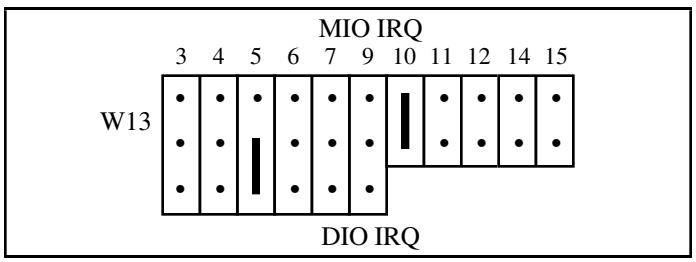


Figure 2-6. Factory Interrupt Jumper Settings IRQ5 (DIO-24) and IRQ10 (MIO-16)

If you do not want to use interrupts, place the jumpers on W13 in the position shown in Figure 2-7. This setting disables the AT-MIO-16D from asserting an interrupt line on the PC AT I/O channel.

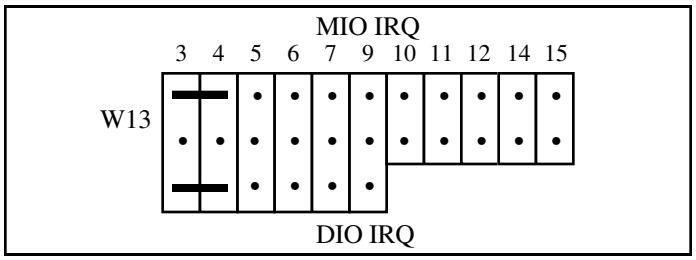


Figure 2-7. Interrupt Jumper Setting for Disabling Interrupts

DIO-24 Circuitry Interrupt Enable Settings

To enable interrupt requests from the DIO-24 circuitry, you must set jumper W14 to select PC2, PC4, or PC6 as the active low interrupt enable line. When the interrupt enable line is logic low, interrupts are enabled from the DIO-24 circuitry of the AT-MIO-16D board. Refer to Chapter 4, *Programming*, for the suggested interrupt enable line setting for each digital I/O mode of operation. If W14 is set to N/C, all interrupt requests from the DIO-24 circuitry are disabled. Figure 2-8 shows the possible jumper settings for W14. The board is shipped with this jumper set to PC4; therefore, interrupt requests from the board are enabled and controlled by PC4.

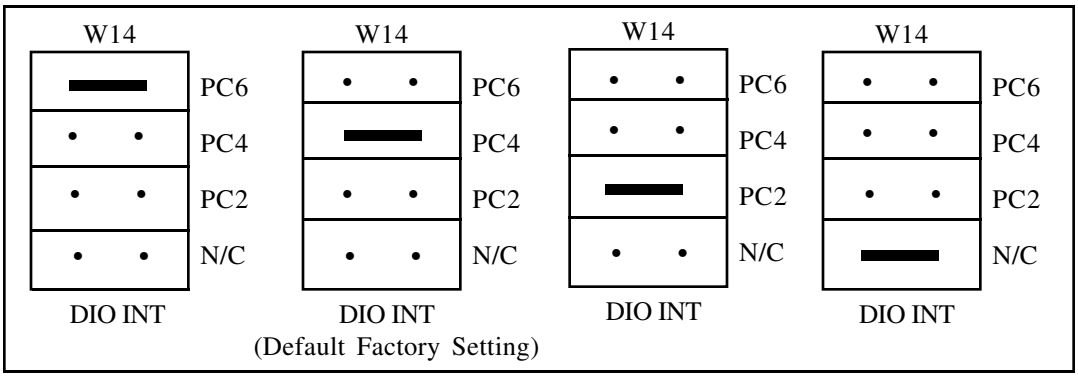


Figure 2-8. Jumper Settings–PC6, PC4, PC2, and N/C

Analog I/O Jumper Settings

The AT-MIO-16D is shipped from the factory with the following configuration:

- Differential analog input (eight channels)
- Bipolar analog input
- ±10 V input range

- ±10 V output range with internal reference selected
- Two's complement digital-to-analog converter (DAC) input modes
- AT-MIO-16D clock signal set to 10 MHz

Table 2-5 lists all the available analog I/O jumper configurations for the AT-MIO-16D with the factory settings noted.

Table 2-5. Analog I/O Jumper Settings

	Configuration	Jumper Settings
ADC Input Range	Unipolar 0 V to +10 V Bipolar ±5 V Bipolar ±10 V (factory setting)	W1: B-C W4: A-B W1: B-C W4: B-C W1: A-B W4: B-C
ADC Input Mode	Differential (DIFF) (factory setting) Nonreferenced single-ended (NRSE) Referenced single-ended (RSE)	W6: A-C, B-D, E-F W9: A-B W6: A-B, C-E, G-H W9: B-C W6: A-B, C-D, G-H W9: B-C
Am9513A & RTSI Bus Clock Select	AT-MIO-16D clock signal = 10 MHz (factory setting) AT-MIO-16D clock signal = RTSI clock signal AT-MIO-16D & RTSI clock signals both = 10 MHz	W5: C-D, E-F W5: A-B, E-F W5: A-B, C-D
DAC0 Reference	Internal (factory setting) External	W3: B-C W3: A-B
DAC1 Reference	Internal (factory setting) External	W2: B-C W2: A-B
DAC0 Output Polarity – Digital Format	Unipolar – Straight binary mode Bipolar – Two's complement mode (factory setting)	W8: B-C W10: B-C W8: A-B W10: A-B
DAC1 Output Polarity – Digital Format	Unipolar – Straight binary mode Bipolar – Two's complement mode (factory setting)	W7: B-C W11: B-C W7: A-B W11: A-B

Analog Input Configuration

You can select different analog input configurations by using the jumper settings shown in Table 2-5. The following paragraphs describe in detail each of the analog input categories. In the configuration illustrations throughout this chapter, the black bars show where to place jumpers.

Input Mode

The AT-MIO-16D offers three different analog input modes–nonreferenced single-ended (NRSE) input, referenced single-ended (RSE) input, and differential (DIFF) input. The single-ended input configurations use 16 channels. The DIFF input configuration uses eight channels. These configurations are described in Table 2-6.

Table 2-6. Input Configurations Available for the AT-MIO-16D

Configuration	Description
DIFF	Differential configuration Provides eight differential inputs with the negative (-) input of the instrumentation amplifier tied to the multiplexer output of channels 8 through 15
RSE	Referenced Single-Ended configuration Provides 16 single-ended inputs with the negative (-) input of the instrumentation amplifier referenced to analog ground
NRSE	Nonreferenced Single-Ended configuration Provides 16 single-ended inputs with the negative (-) input of the instrumentation amplifier tied to AI SENSE and *not* connected to ground

While reading the following paragraphs, you may find it helpful to refer to the *Analog Input Signal Connections* section later in this chapter, which contains diagrams showing the signal paths for the three configurations.

DIFF Analog Input (Eight Channels, Factory Setting)

DIFF input means that each input signal has its own reference, and the difference between each signal and its reference is measured. The signal and its reference are each assigned an input channel. With this input configuration, the AT-MIO-16D can monitor eight different analog input signals. You select the DIFF analog input configuration by setting jumpers W6 and W9 as follows:

W6 :

- A - C Jumper is placed in standby position. Jumper can be discarded.
- B - D AI SENSE is tied to the instrumentation amplifier output ground point.
- E - F Channels 0 through 7 are tied to the positive (+) input of the instrumentation amplifier. Channels 8 through 15 are tied to the negative (-) input of the instrumentation amplifier.

W9 :

A - B Multiplexer is configured to control eight input channels.

This configuration is shown in Figure 2-9.

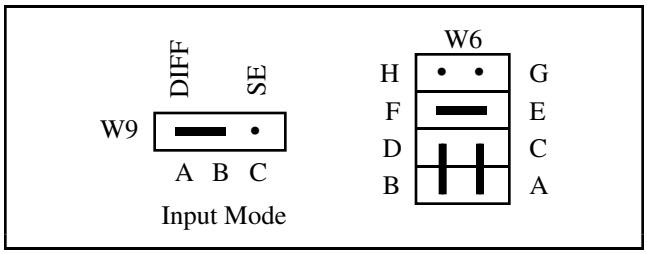


Figure 2-9. DIFF Analog Input Configuration (Factory Setting)

Considerations in using the DIFF analog input configuration are discussed in the *Signal Connections* section later in this chapter. Figure 2-26 shows a schematic diagram of this configuration.

RSE Analog Input (16 Channels)

RSE input means that all input signals are referenced to a common ground point that is also tied to the analog input ground of the AT-MIO-16D board. The negative (-) input of the differential input amplifier is tied to the analog ground. This configuration is useful when measuring floating signal sources. See the *Types of Signal Sources* section later in this chapter for more information. With this input configuration, the AT-MIO-16D can monitor 16 different analog input signals. You select the RSE analog input configuration by setting jumpers W6 and W9 as follows:

W6 :

A - B AI SENSE is tied to the negative (-) input of the instrumentation amplifier.

C - D The negative (-) input of the instrumentation amplifier is tied to the instrumentation amplifier signal ground.

G - H Multiplexer outputs are tied together into the positive (+) input of the instrumentation amplifier.

W9 :

B - C Multiplexer control is configured to control 16 input channels.

This configuration is shown in Figure 2-10.

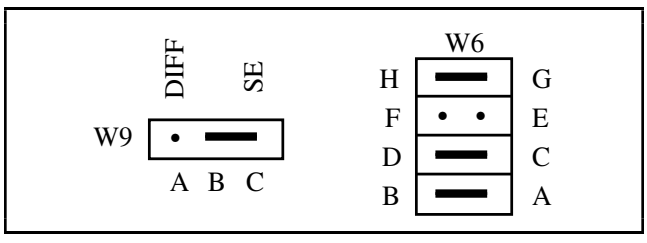


Figure 2-10. RSE Analog Input Configuration

Considerations in using the ground-referenced single-ended analog configuration are discussed in the *Signal Connections* section later in this chapter. Figure 2-28 shows a schematic diagram of this configuration.

NRSE Analog Input (16 Channels)

NRSE analog input means that all input signals are referenced to the same common mode voltage, but that this common mode voltage is allowed to float with respect to the analog ground of the AT-MIO-16D board. This common mode voltage is subsequently subtracted by the input instrumentation amplifier. This configuration is useful when measuring ground-referenced signal sources. See the *Types of Signal Sources* section later in this chapter for more information. With this input configuration, the AT-MIO-16D can measure 16 different analog input signals. You select the NRSE analog input configuration by setting jumpers W6 and W9 as follows:

W6 :

- A - B AI SENSE is tied into the negative (-) input of the instrumentation amplifier.
- C - E Jumper is placed in standby position. Jumper can be discarded.
- G - H Multiplexer outputs are tied together into the positive (+) input of the instrumentation amplifier.

W9:

- B - C Multiplexer control is configured for 16 input channels.

This configuration is shown in Figure 2-11.

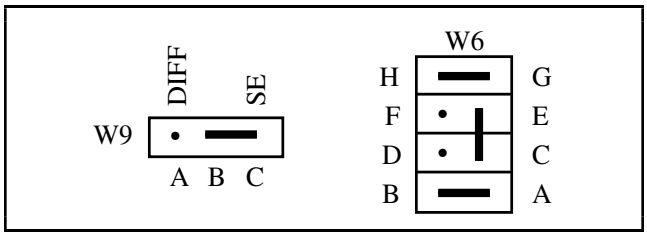


Figure 2-11. NRSE Analog Input Configuration

Considerations in using the NRSE configuration are discussed under the *Signal Connections* section later in this chapter. Figure 2-29 shows a schematic diagram of this configuration.

Analog Input Polarity and Range

The AT-MIO-16D offers two analog input polarities–unipolar input and bipolar input. Unipolar input means that the analog input voltage range is between 0 and V_{ref} where V_{ref} is some positive reference voltage. Bipolar input means that the analog input voltage range is between $-V_{ref}$ and $+V_{ref}$. The AT-MIO-16D also has two input ranges–10 V input range and a 20 V input range. The selection of input polarity and range are combined into three possible configurations as shown in Table 2-7.

Table 2-7. Configurations for Input Range and Input Polarity

Input Range	Input Polarity	Jumper Settings W1	W4
0 to +10 V (10 V range)	Unipolar	B-C	A-B
-5 to +5 V (10 V range)	Bipolar	B-C	B-C
-10 to +10 V (20 V range)	Bipolar	A-B	B-C (factory setting)

Figures 2-12, 2-13, and 2-14 show the jumper positions for the 0 to +10 V, -5 to +5 V, and -10 to +10 V input polarity/range configurations, respectively.

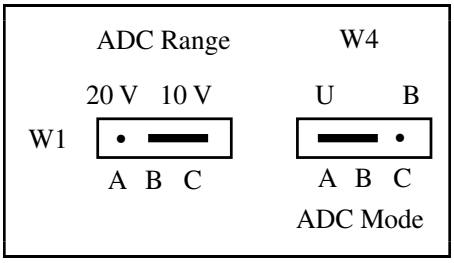


Figure 2-12. 0 to +10 V Input Configuration

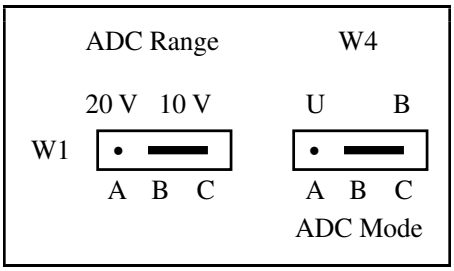


Figure 2-13. -5 to +5 V Input Configuration

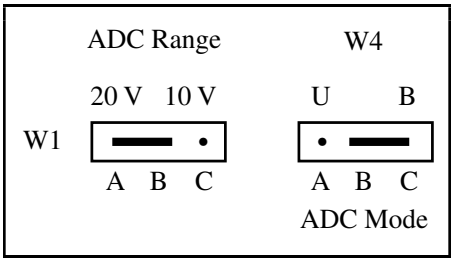


Figure 2-14. Factory -10 to +10 V Analog Input Configuration

Considerations for Selecting Analog Input Ranges

Analog input polarity/range selection depends on the expected input range of the incoming signal. A large input range can accommodate a large signal variation but sacrifices voltage resolution. Choosing a smaller input range increases voltage resolution but may result in the input signal going out of range. For best results, the input range should be matched as closely as possible to the expected range of the input signal. For example, if the input signal is guaranteed to never go negative (below 0 V), a unipolar input is best. However, if the signal does go negative, inaccurate readings will occur.

Software-programmable gain on the AT-MIO-16D increases overall flexibility by matching input signal ranges to those accommodated by the AT-MIO-16D analog-to-digital converter (ADC). The AT-MIO-16DH board has gains of 1, 2, 4, and 8 and is suited for high-level signals near the range of the ADC. The AT-MIO-16DL board is designed to measure low-level signals and has gains of 1, 10, 100, and 500. With the proper gain setting, the full resolution of the ADC can be used to measure the input signal. Table 2-8 shows the overall input range and precision according to the input range configuration and gain used.

Table 2-8. Actual Range and Measurement Precision Versus Input Range Selection and Gain

Range Configuration	Gain	Actual Input Range	Precision*
0 to +10 V	1	0 to +10 V	2.44 mV
	2	0 to +5 V	1.22 mV
	4	0 to +2.5 V	610 µV
	8	0 to +1.25 V	305 µV
	10	0 to +1 V	244 µV
	100	0 to +0.1 V	24.4 µV
	500	0 mV to +20 mV	4.88 µV
-5 to +5 V	1	-5 to +5 V	2.44 mV
	2	-2.5 to +2.5 V	1.22 mV
	4	-1.25 to +1.25 V	610 µV
	8	-0.625 to +0.625 V	305 µV
	10	-0.5 to +0.5 V	244 µV
	100	-50 mV to +50 mV	24.4 µV
	500	-10 mV to +10 mV	4.88 µV
-10 to +10 V	1	-10 to +10 V	4.88 mV
	2	-5 to +5 V	2.44 mV
	4	-2.5 to +2.5 V	1.22 mV
	8	-1.25 to +1.25 V	610 µV
	10	-1 to +1 V	488 µV
	100	-0.1 to +0.1 V	48.8 µV
	500	-20 mV to +20 mV	9.76 µV

* The value of 1 LSB of the 12-bit ADC, that is, the voltage increment corresponding to a change of 1 count in the ADC 12-bit count.

Analog Output Configuration

You can select different analog output configurations by using the jumper settings shown in Table 2-5. The following paragraphs describe in detail each of the analog output configurations.

Analog Output Reference Selection

Each DAC can be connected to the AT-MIO-16D internal reference of 10 V or to the external reference signal connected to the EXTREF pin on the I/O connector. This signal applied to EXTREF must be between -10 V and +10 V. Both channels need not be configured the same way.

External Reference Selection

You select the external reference signal for each analog output channel by setting the following jumpers:

Analog Output Channel 0:	W3	A - B	External reference signal connected to DAC 0 reference input.
Analog Output Channel 1:	W2	A - B	External reference signal connected to DAC 1 reference input.

This configuration is shown in Figure 2-15.

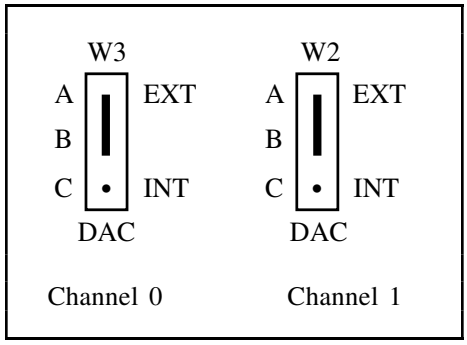


Figure 2-15. External Reference Configuration

Internal Reference Selection (Factory Setting)

You select the onboard 10 V reference for each analog output channel by setting the following jumpers:

Analog Output Channel 0:	W3	B - C	10 V onboard reference connected to DAC 0 reference input.
Analog Output Channel 1:	W2	B - C	10 V onboard reference connected to DAC 1 reference input.

This configuration is shown in Figure 2-16.

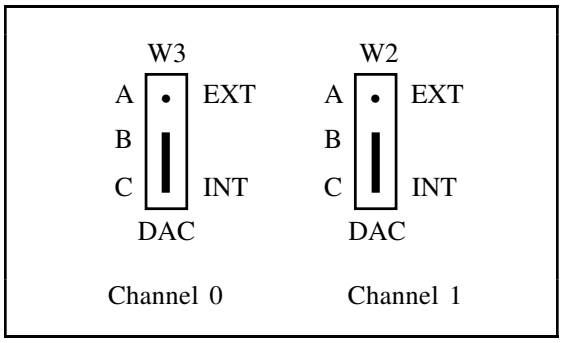


Figure 2-16. Factory Internal Reference Configuration

Analog Output Polarity Selection

Each analog output channel can be configured for either unipolar or bipolar output. A unipolar configuration has a range of 0 to V_{ref} at the analog output. A bipolar configuration has a range of $-V_{ref}$ to $+V_{ref}$ at the analog output. V_{ref} is the voltage reference used by the DACs in the analog output circuitry and can either be the 10 V onboard reference or an externally supplied reference between -10 V and +10 V. Both channels need not be configured the same way; however, at the factory both channels are configured for bipolar output.

Bipolar Output Selection (Factory Setting)

You select the bipolar output configuration for each analog output channel by setting the following jumpers:

Analog Output Channel 0: W8 A - B

Analog Output Channel 1: W7 A - B

This configuration is shown in Figure 2-17.

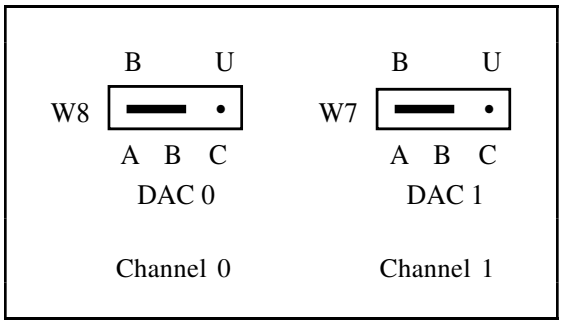


Figure 2-17. Factory Bipolar Output Configuration

When you use the bipolar configuration, you need to select whether to write straight binary or two's complement to the DAC. In straight binary mode, data values written to the analog output channel range from 0 to 4,095 decimal (0 to 0FFF hex). In two's complement mode, data values written to the the analog output channel range from -2,048 to +2,047 decimal (F800 to 07FF hex).

Straight Binary Mode

The data value written to each analog output channel is interpreted as a straight binary number when the following jumpers are set:

Analog Output Straight Binary for Channel 0: W10 B - C

Analog Output Straight Binary for Channel 1: W11 B - C

This configuration is shown in Figure 2-18.

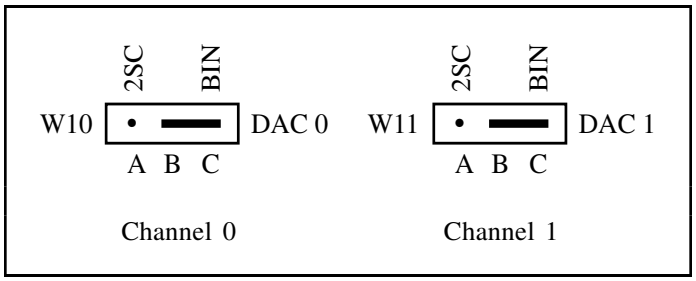


Figure 2-18. Straight Binary Mode

Two's Complement Mode (Factory Setting)

The data value written to each analog output channel is interpreted as a two's complement number when the following jumpers are set:

Analog Output Two's Complement for Channel 0: W10 A - B

Analog Output Two's Complement for Channel 1: W11 A - B

This configuration is shown in Figure 2-19.

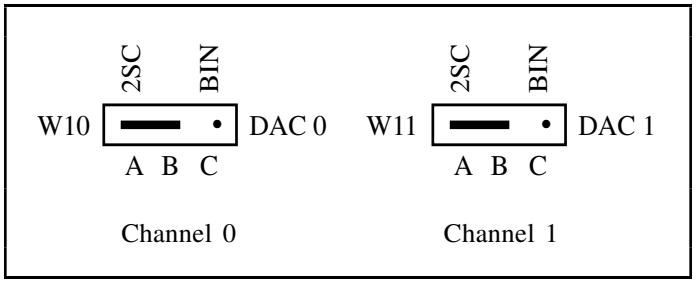


Figure 2-19. Two's Complement Mode (Factory Setting)

Unipolar Output Selection

You select the unipolar output configuration for each analog output channel by setting the following jumpers:

Analog Output Channel 0:	W8	B - C
Analog Output Straight Binary for Channel 0:	W10	B - C
Analog Output Channel 1:	W7	B - C
Analog Output Straight Binary for Channel 1:	W11	B - C

Notice that the straight binary format must be used when in unipolar output mode.

This configuration is shown in Figure 2-20.

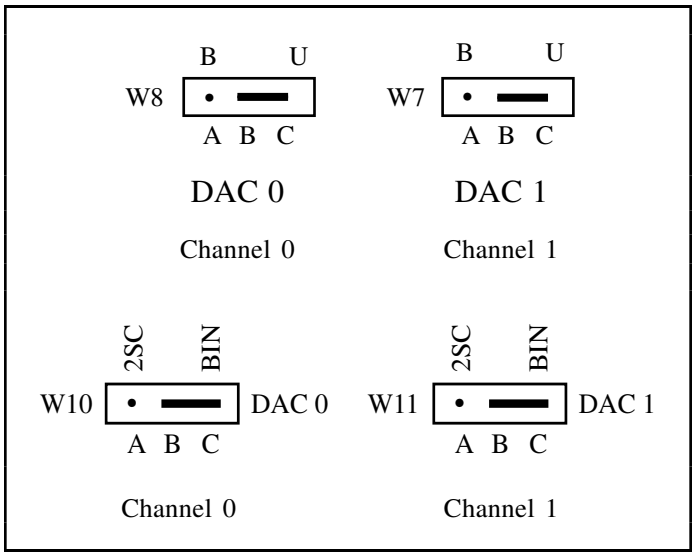


Figure 2-20. Unipolar Output Configuration

Note: If you are using a software package such as LabWindows or NI-DAQ, you may need to reconfigure your software to reflect any changes in jumper or switch settings.

RTSI Bus Clock Selection

When multiple AT Series boards are connected via the RTSI bus, you may want to have all the boards use the same 10-MHz clock. This arrangement is useful for applications that require counter/timer synchronization between boards. Each AT Series board with a RTSI bus interface has an onboard 10-MHz oscillator. Thus, one board can drive the RTSI bus clock signal, and the other boards can receive this signal or disconnect from it.

The configuration for jumper W5 specifies whether a board is to drive the onboard 10-MHz oscillator onto the RTSI bus, receive the RTSI bus clock, or disconnect from the RTSI bus clock. This clock source, whether local or RTSI signal, is then divided by 10 and used as the Am9513A frequency source.

The jumper selections are listed in Table 2-9.

Table 2-9. Configurations for RTSI Bus Clock Selection

Configuration	W5
Disconnect board from RTSI bus clock; use local oscillator	C - D, E - F (factory setting)
Receive RTSI bus clock signal	A - B, E - F
Drive RTSI bus clock signal with local oscillator	A - B, C - D

Figures 2-21, 2-22, and 2-23 show the jumper positions for each of the configurations described above.

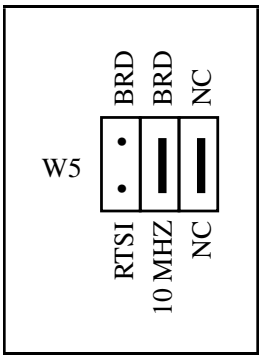


Figure 2-21. Disconnect from RTSI Bus Clock; Use Onboard Oscillator (Factory Setting)

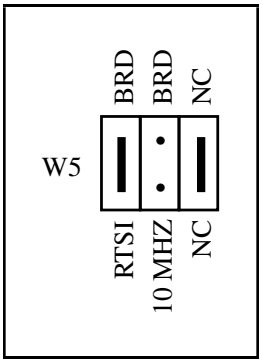


Figure 2-22. Receive RTSI Bus Clock Signal

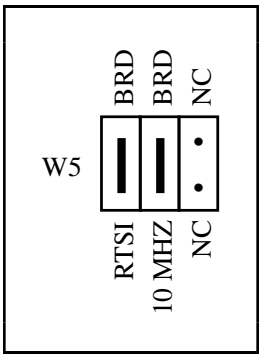


Figure 2-23. Drive RTSI Bus Clock Signal with Onboard Oscillator

Hardware Installation

The AT-MIO-16D can be installed in any available 16-bit expansion slot (AT style) in your computer. The AT-MIO-16D *does not* work if installed in an eight-bit expansion slot (PC style). After you have changed (if needed), verified, and recorded the switches and jumper settings, you are ready to install the AT-MIO-16D. The following are general installation instructions, but consult the user manual or technical reference manual of your PC AT for specific instructions and warnings.

1. Turn off your computer.
2. Remove the top cover or access port to the I/O channel.
3. Remove the expansion slot cover on the back panel of the computer.
4. Insert the AT-MIO-16D into a 16-bit slot. It may be a tight fit, but do not force the board into place.
5. Screw the mounting bracket of the AT-MIO-16D to the back panel rail of the computer.
6. Check the installation.
7. Replace the cover.

The AT-MIO-16D board is installed and ready for operation.

Signal Connections

This section describes input and output signal connections to the AT-MIO-16D board via the AT-MIO-16D I/O connector. This section includes specifications and connection instructions for the signals given on the AT-MIO-16D I/O connector. The I/O connector contains 100 pins that can be split into two standard 50-pin connectors via a cable assembly such as a Type NB5 ribbon cable (see Figure 1-2). One 50-pin connector contains signals associated with the MIO-16 circuitry, while the other 50-pin connector contains signals for the DIO-24 circuitry.

Warning: Connections that exceed any of the maximum ratings of input or output signals on the AT-MIO-16D can result in damage to the AT-MIO-16D board and to the PC AT. Maximum input ratings for each signal are given in this chapter under the discussion of that signal. National Instruments is *not* liable for any damages resulting from any such signal connections.

AT-MIO-16D I/O Connector Pin Description

Figure 2-24 shows the pin assignments for the AT-MIO-16D I/O connector. Refer to *MIO-16 Signal Connection Descriptions* and *DIO-24 Signal Connection Descriptions* later in this chapter for descriptions of the AT-MIO-16D signal connections.

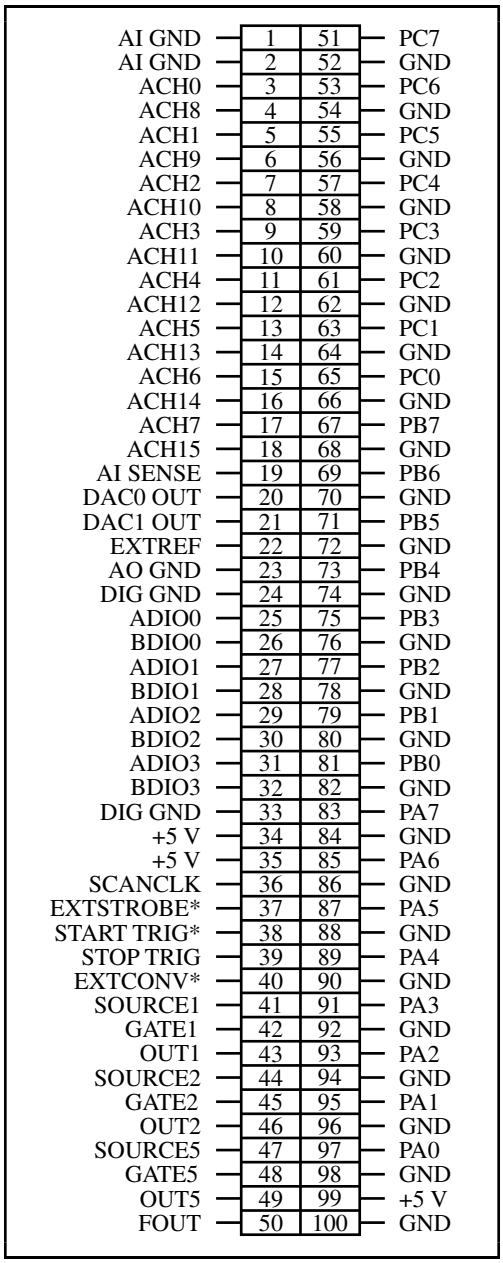


Figure 2-24. AT-MIO-16D I/O Connector Pin Assignments

MIO-16 I/O Connector Pin Description

Figure 2-25 shows the pin assignments for the MIO-16 I/O connector of the AT-MIO-16D.

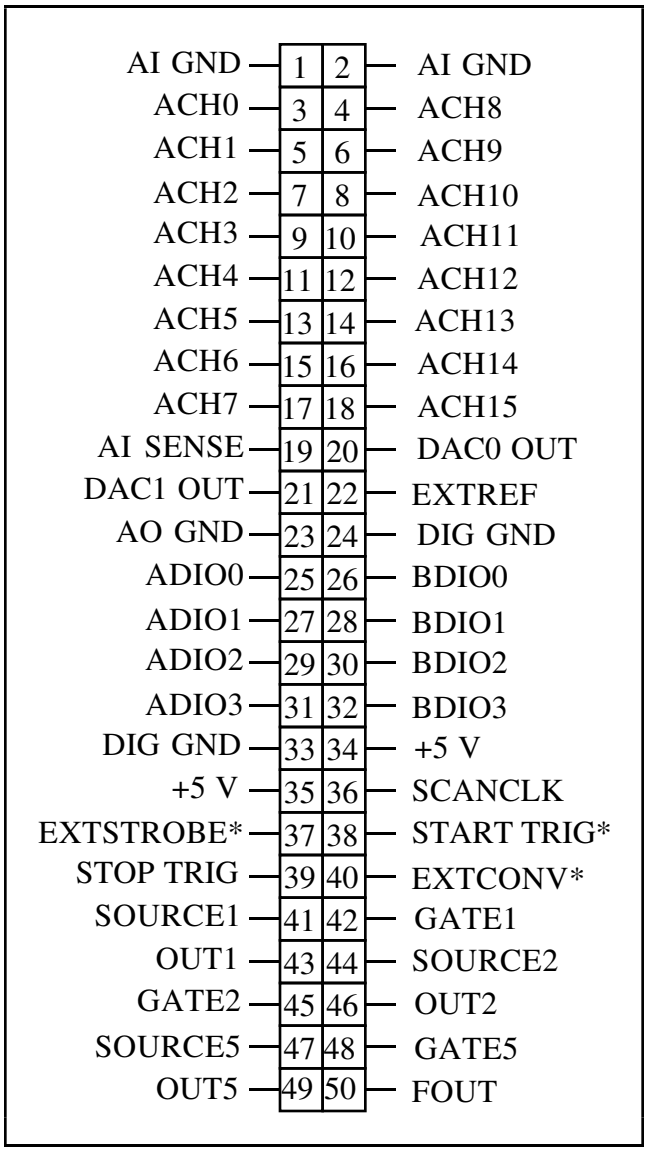


Figure 2-25. MIO-16 I/O Connector Pin Assignments

MIO-16 Signal Connection Descriptions

Pin	Signal Name	Reference	Description
1-2	AI GND	N/A	Analog Input Ground – These pins are the reference point for single-ended measurements and the bias current return point for differential measurements.
3-18	ACH<0..15>	AIGND	Analog Input Channels 0 through 15 – In differential mode, the input is configured for up to eight channels. In single-ended mode, the input is configured for up to 16 channels.
19	AI SENSE	AIGND	Analog Input Sense – This pin serves as the reference node when the board is in NRSE configuration. If desired, this signal can be programmed to be driven by the board analog input ground.
20	DAC0 OUT	AOGND	Analog Channel 0 Output – This pin supplies the voltage output of analog output channel 0.
21	DAC1 OUT	AOGND	Analog Channel 1 Output – This pin supplies the voltage output of analog output channel 1.
22	EXTREF	AOGND	External Reference – This is the external reference input for the analog output circuitry.
23	AO GND	N/A	Analog Output Ground – The analog output voltages are referenced to this node.
24,33	DIG GND	N/A	Digital Ground – This pin supplies the reference for the digital signals at the I/O connector as well as the +5 VDC supply.
25, 27, 29, 31	ADIO<0..3>	DIGGND	Digital I/O port A signals.
26, 28, 30, 32	BDIO<0..3>	DIGGND	Digital I/O port B signals.
34-35	+5 V	DIGGND	+5 VDC Source – This pin is fused for up to 1 A of +5 V supply.
36	SCANCLK	DIGGND	Scan Clock – This pin pulses once for each A/D conversion in the scanning modes. The low-to-high edge indicates when the input signal can be removed from the input or switched to another signal.
37	EXTSTROBE*	DIGGND	External Strobe – Writing to the EXTSTROBE* Register results in a minimum 200 nsec low pulse on this pin.

Pin	Signal Name	Reference	Description (continued)
38	START TRIG*	DIGGND	External Trigger – In posttrigger data acquisition sequences, a low-to-high edge on START TRIG* initiates the sequence. In pretrigger applications, the low-to-high edge of START TRIG* initiates pretrigger conversions while the STOP TRIG signal initiates the posttrigger sequence.
39	STOP TRIG	DIGGND	Stop Trigger – In pretrigger data acquisition, the high-to-low edge of STOP TRIG initiates the posttrigger sequence.
40	EXTCONV*	DIGGND	External Convert – A high-to-low edge on EXTCONV* causes an A/D conversion to occur. If EXTGATE* or EXTCONV* is low, conversions are inhibited.
41	SOURCE1	DIGGND	SOURCE1 – This pin is from the Am9513A Counter 1 signal.
42	GATE1	DIGGND	GATE1 – This pin is from the Am9513A Counter 1 signal.
43	OUT1	DIGGND	OUTPUT1 – This pin is from the Am9513A Counter 1 signal.
44	SOURCE2	DIGGND	SOURCE2 – SOURCE5 – This pin is from the Am9513A Counter 2 signal.
45	GATE2	DIGGND	GATE2 – This pin is from the Am9513A Counter 2 signal.
46	OUT2	DIGGND	OUTPUT2 – This pin is from the Am9513A Counter 2 signal.
47	SOURCE5	DIGGND	SOURCE5 – This pin is from the Am9513A Counter 5 signal.
48	GATE5	DIGGND	GATE5 – This pin is from the Am9513A Counter 5 signal.
49	OUT5	DIGGND	OUT5 – This pin is from the Am9513A Counter 5 signal.
50	FOUT	DIGGND	Frequency Output – This pin is from the Am9513A FOUT signal.

The signals on the connector can be classified as analog input signals, analog output signals, digital I/O signals, digital power connections, or timing I/O signals. Signal connection guidelines for each of these groups are given as follows.

Analog Input Signal Connections

Pins 1 through 19 of the MIO-16 I/O connector are analog input signal pins. Pins 1 and 2 are AI GND signal pins. AI GND is an analog input common signal that is routed directly to the ground tie point on the AT-MIO-16D. These pins can be used for a general analog power ground tie point to the AT-MIO-16D if necessary. Pin 19 is the AI SENSE pin. In single-ended mode, this pin is connected internally to the negative (-) input of the AT-MIO-16D instrumentation amplifier. In DIFF mode, this signal is connected to the reference ground at the output of the instrumentation amplifier.

Pins 3 through 18 are ACH<15..0> signal pins. These pins are tied to the 16 analog input channels of the AT-MIO-16D. In single-ended mode, signals connected to ACH<15..0> are routed to the positive (+) input of the AT-MIO-16D instrumentation amplifier. In DIFF mode, signals connected to ACH<7..0> are routed to the positive (+) input of the AT-MIO-16D instrumentation amplifier, and signals connected to ACH<15..8> are routed to the negative (-) input of the AT-MIO-16D instrumentation amplifier.

The following input ranges and maximum ratings apply to inputs ACH<15..0>:

Differential input range	±10 V
Common-mode input range	±7 V with respect to AT-MIO-16D AGND
Input range	±12 V with respect to AT-MIO-16D AGND
Maximum input voltage rating	±20 V for AT-MIO-16D board powered off ±35 V for AT-MIO-16D board powered on

Warning: Exceeding the differential and common-mode input ranges will result in distorted input signals. Exceeding the maximum input voltage rating may result in damage to the AT-MIO-16D board and to the PC AT. National Instruments is *not* liable for any damages resulting from any such signal connections.

Connection of analog input signals to the AT-MIO-16D depends on the configuration of the AT-MIO-16D analog input circuitry and the type of input signal source. The different AT-MIO-16D configurations allow the AT-MIO-16D instrumentation amplifier to be used in different ways. Figure 2-26 shows a diagram of the AT-MIO-16D instrumentation amplifier.

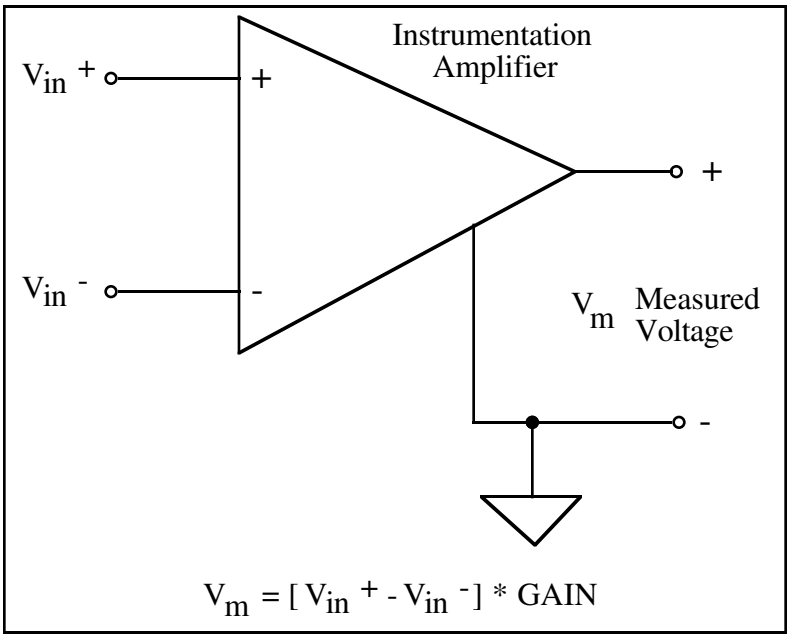


Figure 2-26. AT-MIO-16D Instrumentation Amplifier

The AT-MIO-16D instrumentation amplifier applies gain, common-mode voltage rejection, and high-input impedance to the analog input signals connected to the AT-MIO-16D board. Signals are routed to the positive (+) and negative (-) inputs of the instrumentation amplifier through input multiplexers on the AT-MIO-16D. The instrumentation amplifier converts two input signals to a signal that is the difference between the two input signals multiplied by the gain setting of the amplifier. The amplifier output voltage is referenced to the AT-MIO-16D ground. The AT-MIO-16D ADC measures this output voltage when it performs A/D conversions.

All signals must be referenced to ground somewhere, either at the source device or at the AT-MIO-16D. If you have a floating source, you must use a ground-referenced input connection at the AT-MIO-16D. If you have a grounded source, you must use a nonreferenced input connection at the AT-MIO-16D.

Types of Signal Sources

When configuring the input mode of the AT-MIO-16D and making signal connections, you must first determine whether the signal source is floating or ground-referenced. These two types of signals are described in the following sections.

Floating Signal Sources

A floating signal source is one that is not connected in any way to the building ground system but rather has an isolated ground reference point. Some examples of floating signal sources are outputs of transformers, thermocouples, battery-powered devices, optical isolator outputs, and isolation amplifiers. The ground reference of a floating signal must be tied to the AT-MIO-16D analog input ground in order to establish a local or onboard reference for the signal. Otherwise, the measured input signal varies or appears to float. An instrument or device that provides an isolated output falls into the floating signal source category.

Ground-Referenced Signal Sources

A ground-referenced signal source is one that is connected in some way to the building system ground and is therefore already connected to a common ground point with respect to the AT-MIO-16D board, assuming that the PC AT is plugged into the same power system. Nonisolated outputs of instruments and devices that plug into the building power system fall into this category.

The difference in ground potential between two instruments connected to the same building power system is typically between 1 mV and 100 mV but can be much higher if power distribution circuits are not properly connected. If the grounded signal source is measured improperly, this difference may show up as an error in the measurement. The connection instructions for grounded signal sources below are designed to eliminate this ground potential difference from the measured signal.

Input Configurations

The AT-MIO-16D can be configured for one of three input modes–NRSE, RSE, or DIFF. The following sections discuss the use of single-ended and differential measurements, and considerations for measuring both floating and ground-referenced signal sources. Table 2-10 summarizes the recommended input configuration for both types of signal sources.

Table 2-10. Recommended Input Configurations for Ground-Referenced and Floating Signal Sources

Type of Signal	Recommended Input Configuration
Ground-Referenced (nonisolated outputs, plug-in instruments)	DIFF NRSE
Floating (batteries, thermocouples, isolated outputs)	DIFF with bias resistors RSE

Differential Connection Considerations (DIFF Configuration)

Differential connections are those in which each AT-MIO-16D analog input signal has its own reference signal or signal return path. These connections are available when the AT-MIO-16D is configured in the DIFF mode. Each input signal is tied to the positive (+) input of the instrumentation amplifier; and its reference signal, or return, is tied to the negative (-) input of the instrumentation amplifier.

When the AT-MIO-16D is configured for DIFF input, each signal uses two of the multiplexer inputs–one for the signal and one for its reference signal. Therefore, only eight analog input channels are available when using the DIFF configuration. The DIFF input configuration should be used when any of the following conditions are present:

1. Input signals are low level (less than 1 V).
2. Leads connecting the signals to the AT-MIO-16D are greater than 15 ft.
3. Any of the input signals requires a separate ground reference point or return signal.
4. The signal leads travel through noisy environments.

Differential signal connections reduce picked-up noise and increase common-mode signal and noise rejection. They also allow input signals to float within the common-mode limits of the input instrumentation amplifier.

Differential Connections for Grounded Signal Sources

Figure 2-27 shows how to connect a ground-referenced signal source to an AT-MIO-16D board configured for DIFF input. Configuration instructions are included under the *Analog Input Configuration* section earlier in this chapter.

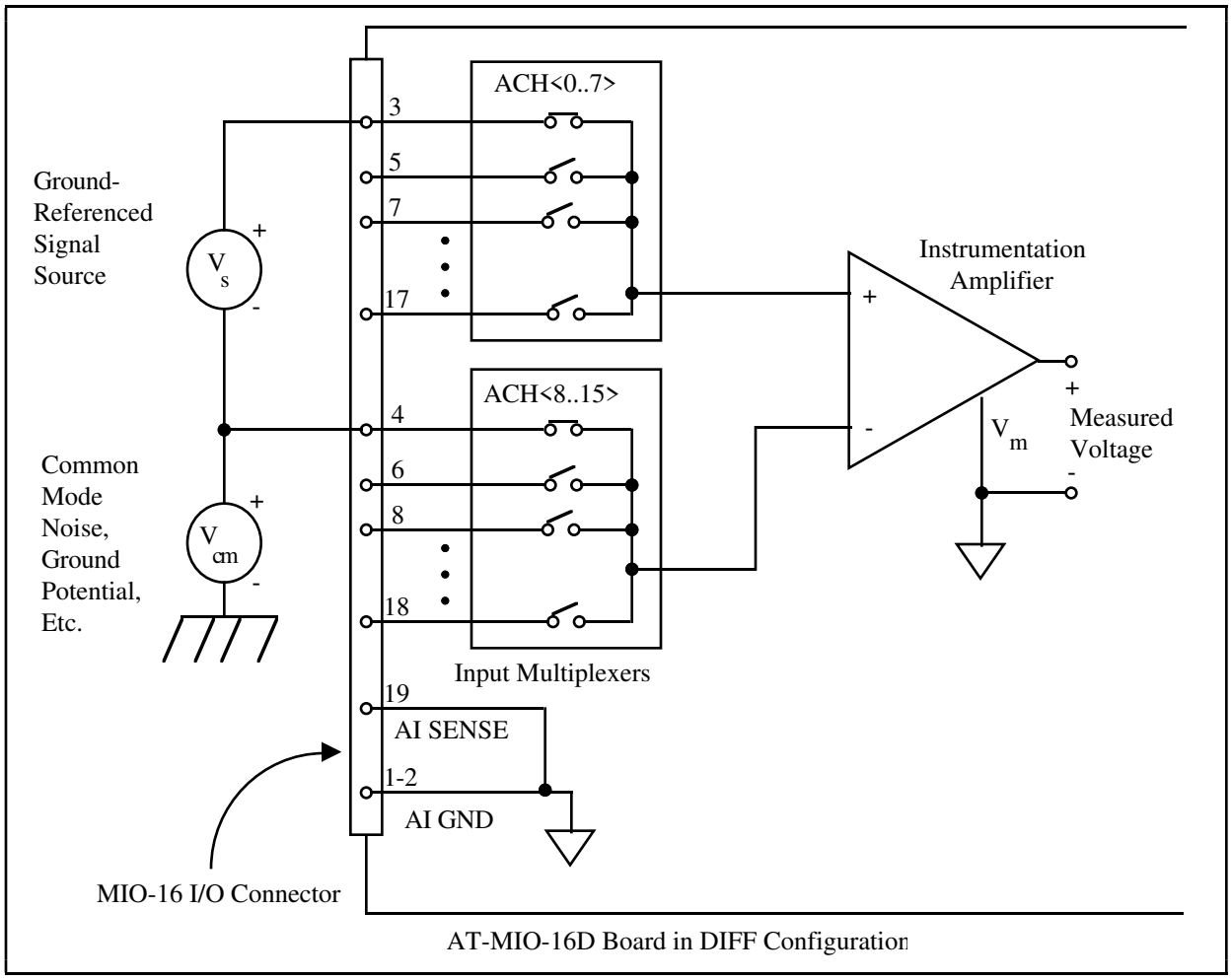


Figure 2-27. Differential Input Connections for Grounded Signal Sources

With this type of connection, the instrumentation amplifier rejects both the common-mode noise in the signal and the ground potential difference between the signal source and the AT-MIO-16D ground (shown as V_{cm} in Figure 2-27).

Differential Connections for Floating Signal Sources

Figure 2-28 shows how to connect a floating signal source to an AT-MIO-16D board configured for DIFF input. Configuration instructions are included under the *Analog Input Configuration* section earlier in this chapter.

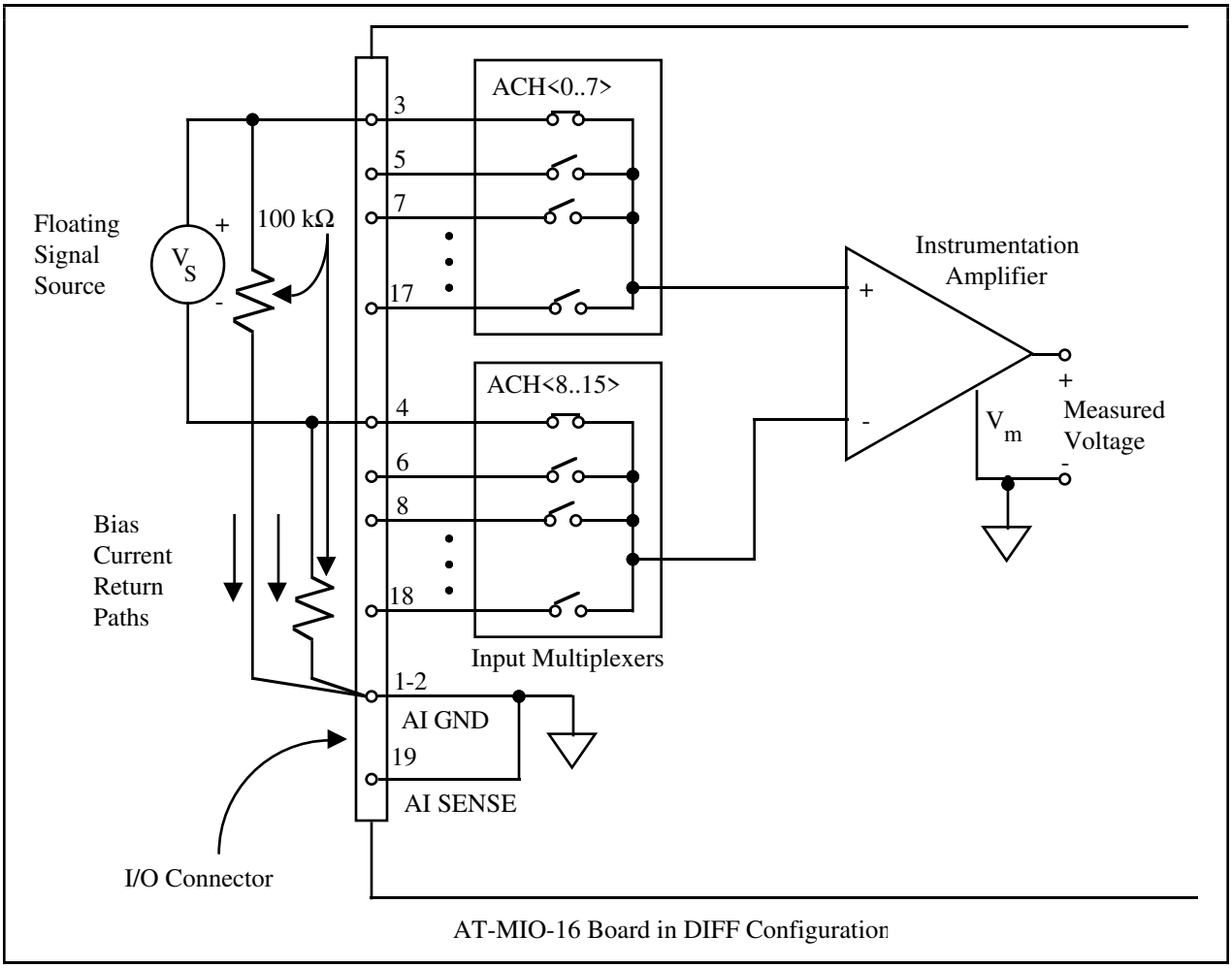


Figure 2-28. Differential Input Connections for Floating Sources

The 100-kΩ resistors shown in Figure 2-28 create a return path to ground for the bias currents of the instrumentation amplifier. If a return path is not provided, the instrumentation amplifier bias currents charge up stray capacitances, resulting in uncontrollable drift and possible saturation in the amplifier. Typically, values from 10 kΩ to 100 kΩ are used.

A resistor from each input to ground, as shown in Figure 2-28, provides bias current return paths for an AC-coupled input signal. This solution, although necessary for AC-coupled signals, lowers the input impedance of the analog input channel. In addition, the input offset current of the instrumentation amplifier contributes a DC offset voltage at the input. The amplifier has a maximum input offset current of ±15 nA and a typical offset current drift of ±20 pA/°C. Multiplied by the 100-kΩ resistor, this current contributes a maximum offset voltage of 1.5 mV and a typical offset voltage drift of 2 µV/°C at the input. Keep this in mind when you observe DC offsets with AC-coupled inputs.

If the input signal is DC-coupled, then you only need the resistor connecting the negative (-) signal input to ground. This connection does not lower the input impedance of the analog input channel.

Single-Ended Connection Considerations

Single-ended connections are those in which all AT-MIO-16D analog input signals are referenced to one common ground. The input signals are tied to the positive (+) input of the instrumentation amplifier, and their common ground point is tied to the negative (-) input of the instrumentation amplifier.

When the AT-MIO-16D is configured for single-ended input (NRSE or RSE), 16 analog input channels are available. You can use single-ended input connections when the following criteria are met by all input signals:

1. Input signals are high-level (greater than 1 V).

2. Leads connecting the signals to the AT-MIO-16D are less than 15 ft.

3. All input signals share a common reference signal (at the source).

If any of the above criteria is not met, using DIFF input configuration is recommended.

You can jumper configure the AT-MIO-16D for two different types of single-ended connections–RSE configuration and NRSE configuration. Use the RSE configuration for floating signal sources; in this case, the AT-MIO-16D provides the reference ground point for the external signal. Use the NRSE configuration for ground-referenced signal sources; in this case, the external signal supplies its own reference ground point and the AT-MIO-16D should not supply one.

Single-Ended Connections for Floating Signal Sources (RSE Configuration)

Figure 2-29 shows how to connect a floating signal source to an AT-MIO-16D board configured for single-ended input. You must configure the AT-MIO-16D analog input circuitry for RSE input to make these types of connections. Configuration instructions are included under the *Analog Input Configuration* section earlier in this chapter.

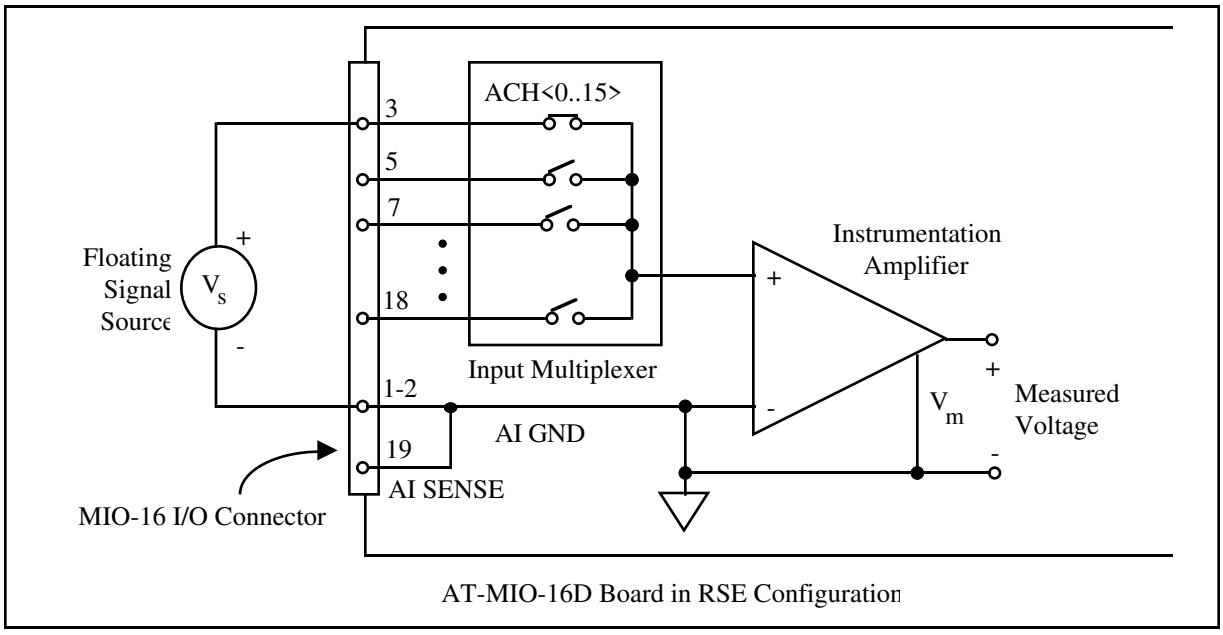


Figure 2-29. Single-Ended Input Connections for Floating Signal Sources

Single-Ended Connections for Grounded Signal Sources (NRSE Configuration)

If a grounded signal source is to be measured with a single-ended configuration, then you must configure the AT-MIO-16D in the NRSE input configuration. Connect the signal to the positive (+) input of the AT-MIO-16D instrumentation amplifier and connect the signal local ground reference to the negative (-) input of the AT-MIO-16D instrumentation amplifier. The ground point of the signal should therefore be connected to the AI SENSE pin. Any potential difference between the AT-MIO-16D ground and the signal ground appears as a common-mode signal at both the positive (+) and negative (-) inputs of the instrumentation amplifier and this difference is rejected by the amplifier. On the other hand, if the input circuitry of the AT-MIO-16D is referenced to ground, such as in the RSE configuration, this difference in ground potentials appears as an error in the measured voltage.

Figure 2-30 shows how to connect a grounded signal source to an AT-MIO-16D board configured in the NRSE configuration. Configuration instructions are included under the *Analog Input Configuration* section earlier in this chapter.

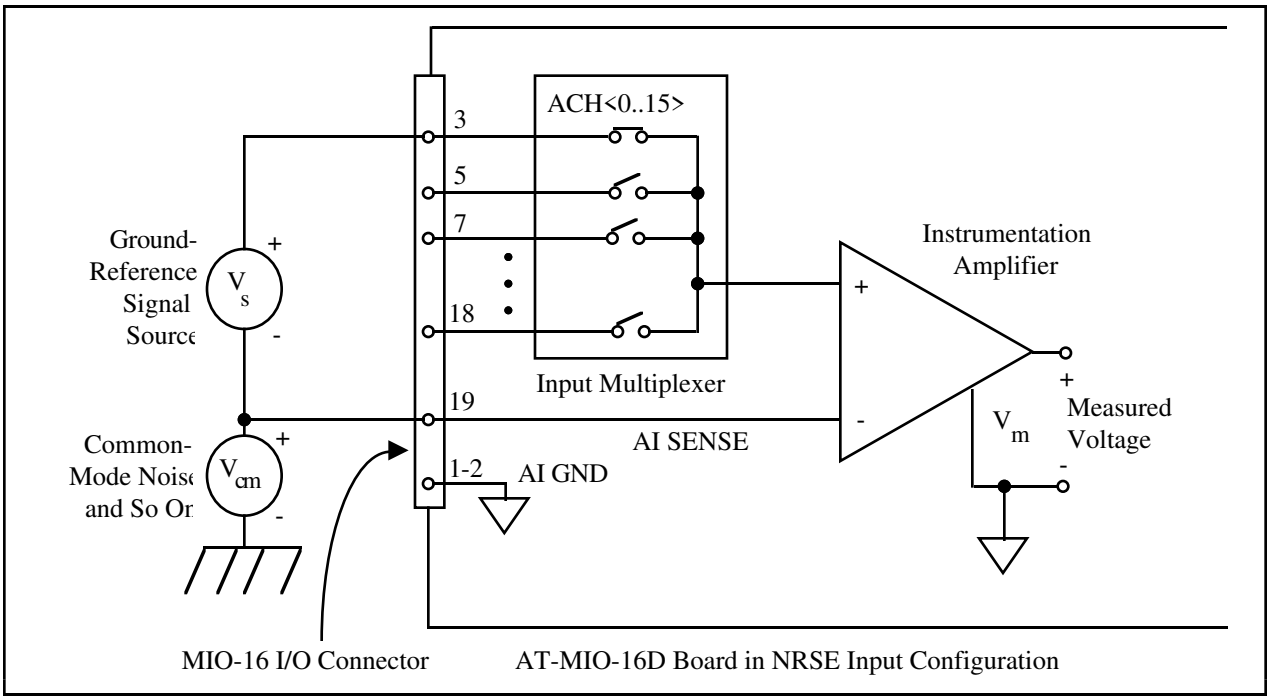


Figure 2-30. Single-Ended Input Connections for Grounded Signal Sources

Common-Mode Signal Rejection Considerations

Figures 2-27 and 2-30, located earlier in this chapter, show connections for signal sources that are already referenced to some ground point with respect to the AT-MIO-16D. In these cases, the instrumentation amplifier can reject any voltage due to ground potential differences between the signal source and the AT-MIO-16D. In addition, with differential input connections, the instrumentation amplifier can reject common-mode noise pickup in the leads connecting the signal sources to the AT-MIO-16D.

The common-mode input range of the AT-MIO-16D instrumentation amplifier is defined as the magnitude of the greatest common-mode signal that can be rejected.

The common-mode input range for the AT-MIO-16D depends on the size of the differential input signal ($V_{diff} = V^+_{in} - V^-_{in}$) and the gain setting of the instrumentation amplifier. The exact formula for the allowed common-mode input range is as follows:

$$V_{cm\text{-}max} = \pm \left(12\ V - \frac{V_{diff} * Gain}{2}\right)$$

where the maximum value for V_{diff} is as follows:

±10 V range	$V_{diff\text{-}max} = \pm 10$ V
0 to +10 V range	$V_{diff\text{-}max} = 10$ V
±5 V range	$V_{diff\text{-}max} = \pm 5$ V

For example, for a differential voltage as large as 20 mV and a gain of 500, the largest common mode voltage that can be rejected is ±7 V. However, if the differential signal is 10 mV with a gain of 500, ±9.5-V common-mode voltage can be rejected.

The common-mode voltage is measured with respect to the AT-MIO-16D ground and can be calculated by the following formula:

$$V_{cm\text{-}actual} = \frac{V^{+}_{in} + V^{-}_{in}}{2}$$

where V^{+}_{in} is the signal at the positive (+) input of the instrumentation amplifier and V^{-}_{in} is the signal at the negative (-) input of the instrumentation amplifier.

If the input signal common-mode range exceeds ±7 V with respect to the AT-MIO-16D ground, you need to limit the amount of floating that occurs between the signal ground and the AT-MIO-16D ground.

Analog Output Signal Connections

Pins 20 through 23 of the MIO-16 I/O connector are analog output signal pins.

Pins 20 and 21 are the DAC0 OUT and DAC1 OUT signal pins. DAC0 OUT is the voltage output signal for analog output channel 0. DAC1 OUT is the voltage output signal for analog output channel 1.

Pin 22, EXTREF, is the external reference input for both analog output channels. You must configure each analog output channel individually for external reference selection in order for the signal applied at the external reference input to be used by that channel. Analog output configuration instructions are included under the *Analog Output Configuration* section earlier in this chapter.

The following ranges and ratings apply to the EXTREF input:

Useful input voltage range:	±10 V peak with respect to AO GND
Absolute maximum ratings:	±25 V peak with respect to AO GND

Pin 23, AO GND, is the ground reference point for both analog output channels and for the external reference signal.

Figure 2-31 shows how to make analog output connections and the external reference input connection to the AT-MIO-16D board. If neither channel is configured to use an external reference signal, do not connect anything to the EXTREF pin.

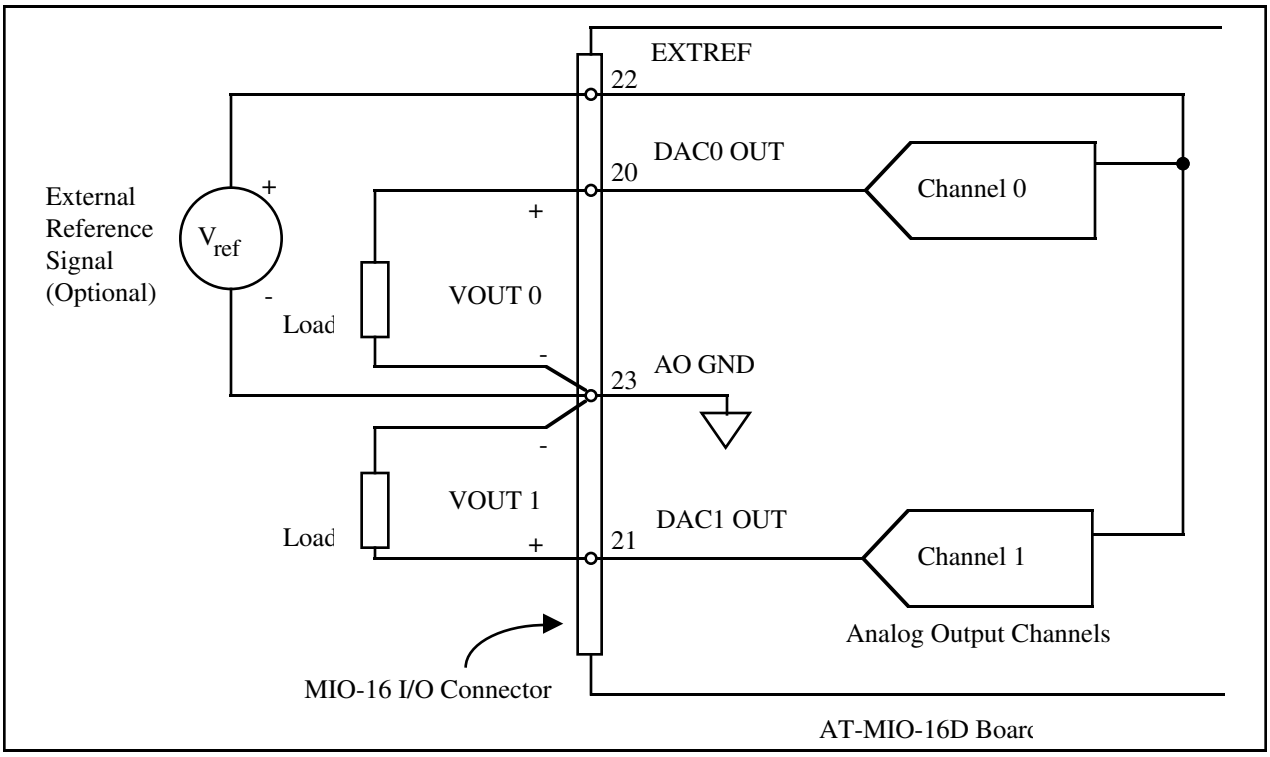


Figure 2-31. Analog Output Connections

The external reference signal can be either a DC or an AC signal. This reference signal is multiplied by the DAC code to generate the output voltage. The DACs in the analog output channels are rated for -82 dB total harmonic distortion with a 1 kHz, 6-Vrms sine wave reference signal and with the DACs set at their maximum (full-scale) digital value.

Digital I/O Signal Connections

Pins 24 through 32 of the MIO-16 I/O connector are digital I/O signal pins associated with the MIO-16 circuitry of the AT-MIO-16D board.

Pins 25, 27, 29, and 31 are connected to the digital lines ADIO<3..0> for digital I/O port A. Pins 26, 28, 30, and 32 are connected to the digital lines BDIO<3..0> for digital I/O port B. Pin 24, DIG GND, is the digital ground pin for both digital I/O ports. Ports A and B can be programmed individually to be inputs or outputs.

The following specifications and ratings apply to the MIO-16 digital I/O lines.

Absolute maximum voltage input rating 5.5 V with respect to DIG GND

Digital input specifications (referenced to DIG GND):

V_{IH} input logic high voltage	2 V minimum
V_{IL} input logic low voltage	0.8 V maximum

I_{IH} input current load,
logic high input voltage 40 µA maximum

I_{IL} input current load,
logic low input voltage -120 µA maximum

Digital output specifications (referenced to DIG GND):

V_{OH} output logic high voltage 2.4 V minimum
V_{OL} output logic low voltage 0.5 V maximum

I_{OH} output source current, logic high 2.6 mA maximum

I_{OH} output sink current, logic low 24 mA maximum

With these specifications, each digital output line can drive 11 standard TTL loads and over 50 LS TTL loads. The MIO-16 circuitry digital I/O lines are pulled up through 100-kΩ resistors to +5 V.

Figure 2-32 depicts signal connections for three typical digital I/O applications.

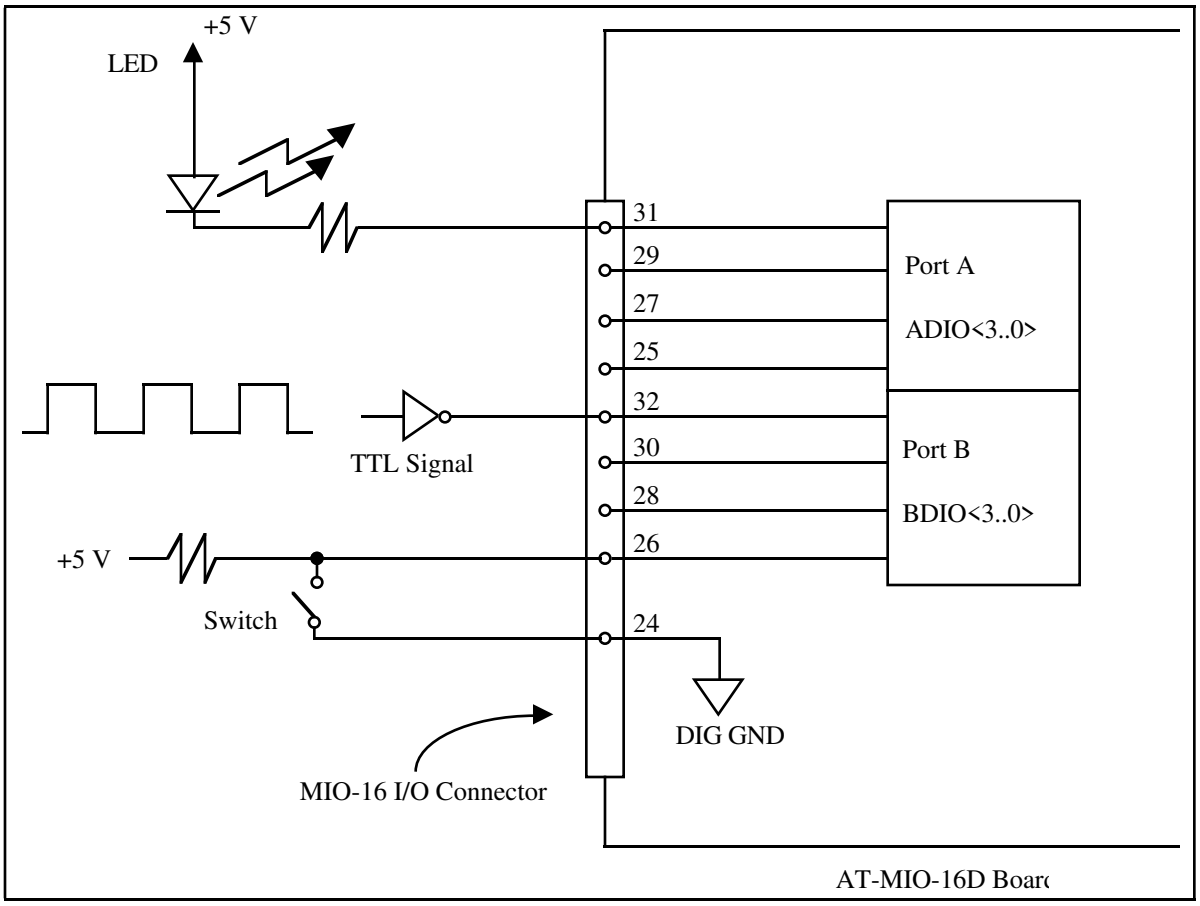


Figure 2-32. Digital I/O Connections

In Figure 2-32, port A is configured for digital output, and port B is configured for digital input. Digital input applications include receiving TTL signals and sensing external device states such as the state of the switch in Figure 2-32. Digital output applications include sending TTL signals and driving external devices such as the LED shown in Figure 2-32.

Power Connections

Pins 34 and 35 of the MIO-16 I/O connector provide +5 V from the PC AT power supply. These pins are referenced to DIG GND and can be used to power external digital circuitry.

Power rating: 1 A at +5 V ± 10%

Warning: These +5 V power pins *should not* be directly connected to analog or digital ground or to any other voltage source on the AT-MIO-16D or any other device. Doing so can damage the AT-MIO-16D and the PC AT. National Instruments is *not* liable for damages resulting from such a connection. A spare MIO-16 fuse is provided in case the power rating is inadvertently exceeded. You should use this fuse only after the cause of the initial problem is known, so as not to blow the spare fuse as well.

Timing Connections

Pins 36 through 50 of the MIO-16 I/O connector are connections for timing I/O signals. Pins 36 through 40 carry signals used for data acquisition timing. These signals are explained under the *Data Acquisition Timing Connections* section later in this chapter. Pins 41 through 50 carry general-purpose timing signals provided by the onboard Am9513A Counter/Timer. These signals are explained under the *General-Purpose Timing Signal Connections* section later in this chapter.

Data Acquisition Timing Connections

The data acquisition timing signals are SCANCLK, EXTSTROBE*, START TRIG*, STOP TRIG, and EXTCONV*.

SCANCLK is an output signal that generates a high-to-low edge whenever an A/D conversion begins. SCANCLK pulses only when scanning is enabled on the AT-MIO-16D. SCANCLK is normally high and pulses low for approximately 1 µsec after the A/D conversion begins. The low-to-high edge signals that the input signal has been acquired. This signal can be used to clock external analog input multiplexers. The SCANCLK signal is driven by one LS TTL gate.

A low pulse is generated on the EXTSTROBE* pin when the External Strobe Register is loaded (see the *External Strobe Register* section in Chapter 4, *Programming*). Figure 2-33 shows the timing for the EXTSTROBE* signal.

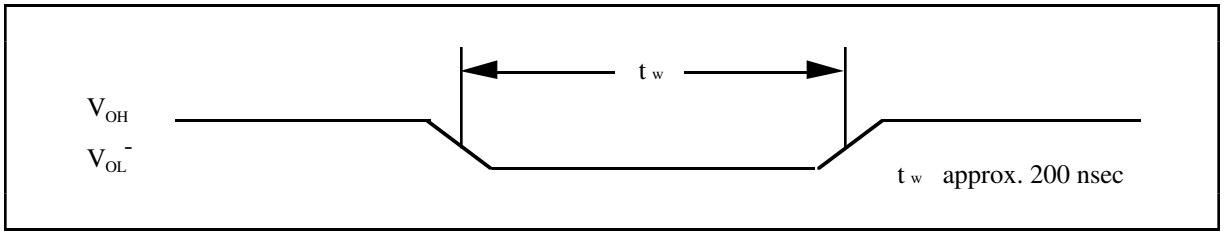


Figure 2-33. EXTSTROBE* Signal Timing

The pulse is typically 200 nsec in width. The EXTSTROBE* signal can be used by an external device to latch signals or trigger events. The EXTSTROBE* signal is an LS TTL signal.

A/D conversions can be externally triggered with the EXTCONV* pin. Applying an active low pulse to the EXTCONV* signal initiates an A/D conversion. The A/D conversion is initiated by the low-to-high edge of the applied pulse. Figure 2-34 shows the timing requirements for the EXTCONV* signal.

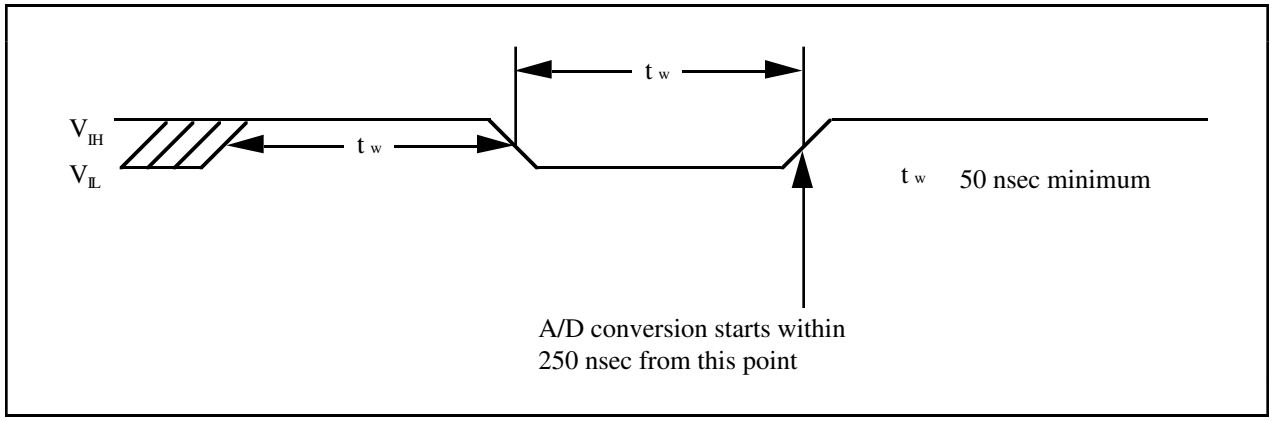


Figure 2-34. EXTCONV* Signal Timing

The minimum allowed pulse width is 50 nsec. An A/D conversion starts within 250 nsec of the low-to-high edge. There is no maximum pulse width limitation. EXTCONV* should be high for at least 50 nsec before going low. The EXTCONV* signal is one LS TTL load and is pulled up to +5 V through a 4.7-kΩ resistor.

Note: EXTCONV* is also driven by the output of Counter 3 of the Am9513A Counter/Timer. This counter is also referred to as the sample-interval counter. The output of Counter 3 must be disabled to a high-impedance state if A/D conversions are to be controlled by pulses applied to the EXTCONV* pin. If Counter 3 is used to control A/D conversions, its output signal can be monitored at the EXTCONV* pin.

You can initiate any data acquisition sequence controlled by the onboard sample-interval and sample counters by an external trigger applied to the START TRIG* pin. If conversions are generated by the EXTCONV* signal, START TRIG* does not affect the acquisition timing. Once the two counters are initialized and armed, applying a falling edge to the START TRIG* pin starts the counters, thereby initiating a data acquisition sequence.

The data acquisition operation is initiated by the high-to-low edge of the applied pulse. Figure 2-35 shows the timing requirements for the START TRIG* signal.

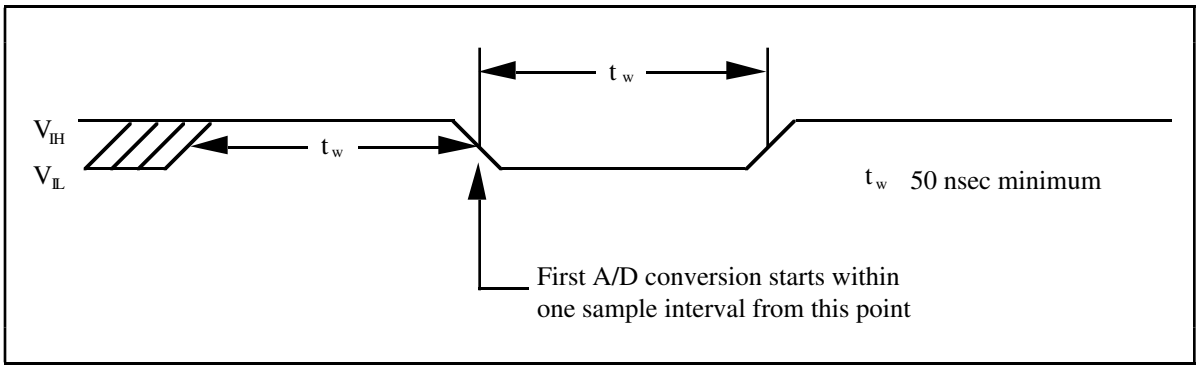


Figure 2-35. START TRIG* Signal Timing

The minimum allowed pulse width is 50 nsec. The first A/D conversion starts within one sample interval from the high-to-low edge. The sample interval is controlled by Counter 3.

There is no maximum pulse width limitation; however, START TRIG* should be high for at least 50 nsec before going low. The START TRIG* signal is one LS TTL load and is pulled up to +5 V through a 4.7-kΩ resistor.

The STOP TRIG pin is used during AT-MIO-16D pretriggered data acquisition operations. In pretriggered mode, data is acquired but no sample counting occurs until a rising edge is applied to the STOP TRIG pin. This causes the sample counter to then start counting conversions. The acquisition then completes when the sample counter decrements to zero. This mode acquires data both before and after a hardware trigger is received. Figure 2-36 shows the timing requirements for the STOP TRIG signal.

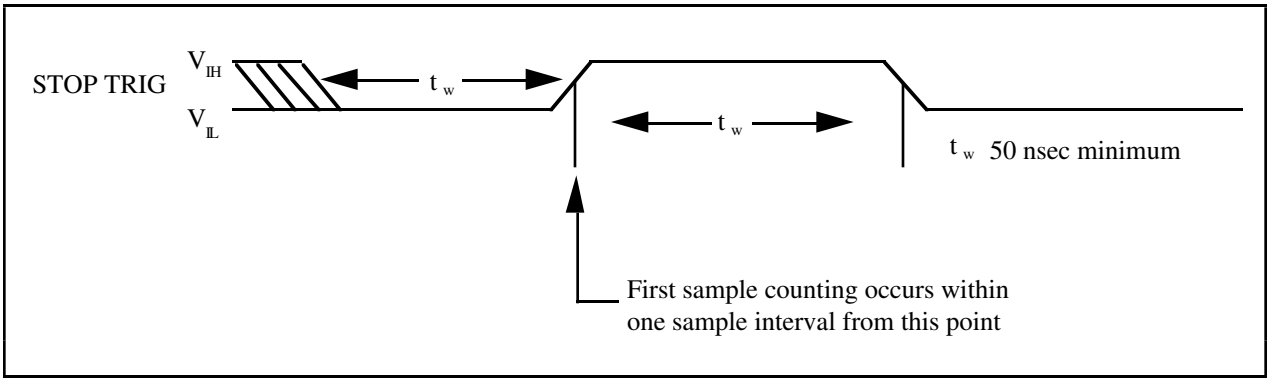


Figure 2-36. STOP TRIG Signal Timing

The STOP TRIG signal is one LS TTL load and is pulled up to +5 V through a 4.7-kΩ resistor.

General-Purpose Timing Signal Connections

The general-purpose timing signals include the GATE, SOURCE, and OUT signals for the Am9513A Counters 1, 2, and 5, and the FOUT signal generated by the Am9513A. Counters 1, 2,

and 5 of the Am9513A Counter/Timer can be used for general-purpose applications, such as pulse and square wave generation, event counting, and pulse-width, time-lapse, and frequency measurements. For these applications, SOURCE and GATE signals can be directly applied to the counters from the I/O connector, and the counters are programmed for various operations.

The Am9513A Counter/Timer is described briefly in Chapter 3, *Theory of Operation*. For detailed programming information, consult Appendix E, *Am9513A Data Sheet.* For detailed applications information, consult the *Am9513A/Am9513 System Timing Controller* technical manual published by Advanced Micro Devices, Inc.

You can produce pulses and square waves by programming Counter 1, 2, or 5 to generate a pulse signal at its OUT output pin or to toggle the OUT signal each time the counter reaches the terminal count.

For event counting, program one of the counters to count rising or falling edges applied to any of the Am9513A SOURCE inputs. The counter value can then be read to determine the number of edges that have occurred. You can gate counter operation on and off during event counting.

Figure 2-37 shows connections for a typical event-counting operation where a switch is used to gate the counter on and off.

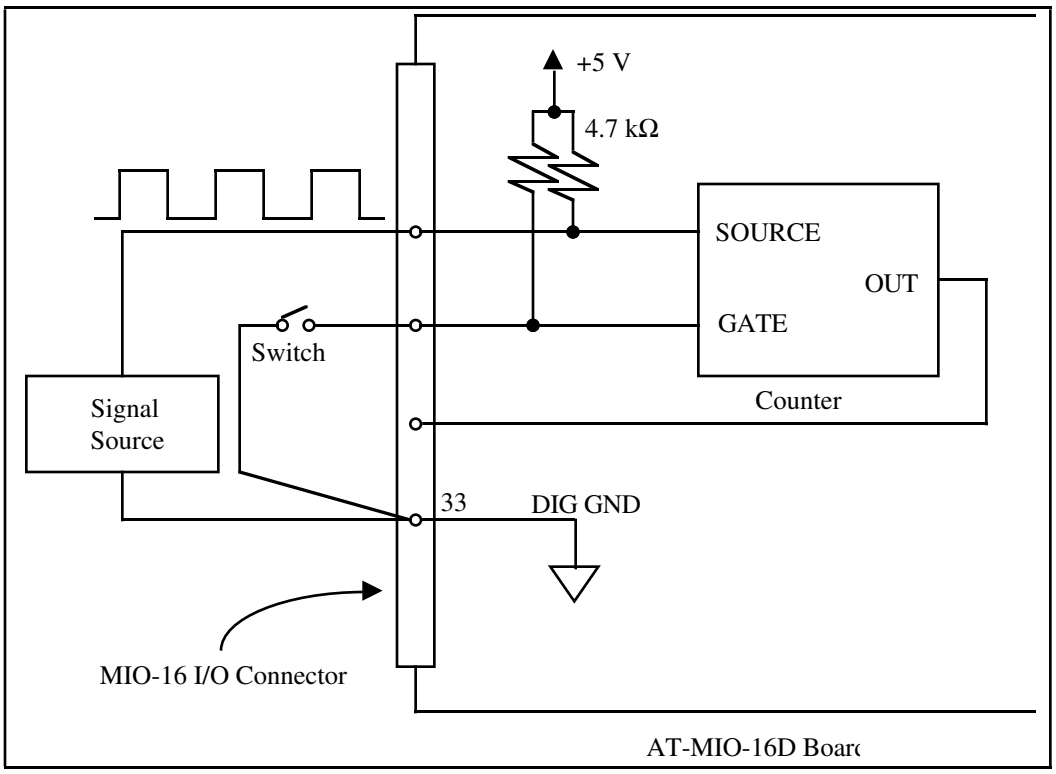


Figure 2-37. Event-Counting Application with External Switch Gating

To perform pulse-width measurement, program a counter to be level gated. The pulse to be measured is applied to the counter GATE input. Program the counter to count while the signal at the GATE input is either high or low. If the counter is programmed to count an internal timebase, then the pulse width is equal to the counter value multiplied by the timebase period.

For time-lapse measurement, program a counter to be edge gated. Apply an edge to the counter GATE input to start the counter. You can program the counter to start counting after receiving either a high-to-low edge or a low-to-high edge. If the counter is programmed to count an internal timebase, then the time lapse since receiving the edge is equal to the counter value multiplied by the timebase period.

To measure frequency, program a counter to be level gated and the rising or falling edges are counted in a signal applied to a SOURCE input. The gate signal applied to the counter GATE input is of some known duration. In this case, program the counter to count either rising or falling edges at the SOURCE input while the gate is applied. The frequency of the input signal is then the count value divided by the known gate period. Figure 2-38 shows the connections for a frequency measurement application. You could use a second counter to generate the gate signal in this application.

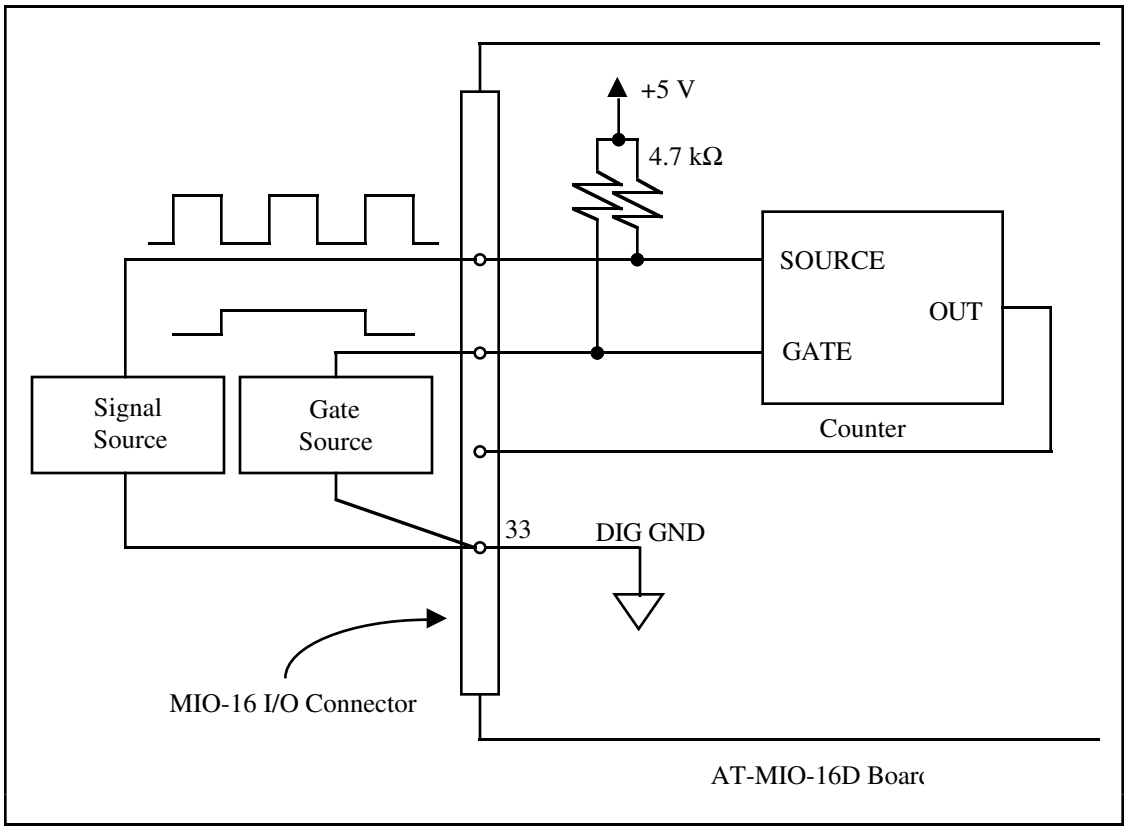


Figure 2-38. Frequency Measurement Application

Two or more counters can be concatenated by tying the OUT signal from one counter to the SOURCE signal of another counter. You can then treat the counters as one 32-bit or 48-bit counter for most counting applications.

The GATE, SOURCE, and OUT signals for Counters 1, 2, and 5, and the FOUT output signal are tied directly from the Am9513A input and output pins to the I/O connector. In addition, the GATE, SOURCE, and OUT1 pins are pulled up to +5 V through a 4.7-kΩ resistor. The input and output ratings and timing specifications for the Am9513A signals are given below.

The following specifications and ratings apply to the Am9513A I/O signals:

Absolute maximum voltage input rating -0.5 V to +7.0 V with respect to DIG GND

Am9513A digital input specifications (referenced to DIG GND):

V_{IH} input logic high voltage	2.2 V minimum
V_{IL} input logic low voltage	0.8 V maximum
Input load current	±10 µA maximum

Am9513A digital output specifications (referenced to DIG GND):

V_{OH} output logic high voltage	2.4 V minimum
V_{OL} output logic low voltage	0.4 V maximum
I_{OH} output source current at V_{OH}	200 µA maximum
I_{OL} output sink current at V_{OL}	3.2 mA maximum
Output current, high-impedance state	±25 µA maximum

Figure 2-39 shows the timing requirements for the GATE and SOURCE input signals and the timing specifications for the OUT output signals of the Am9513A.

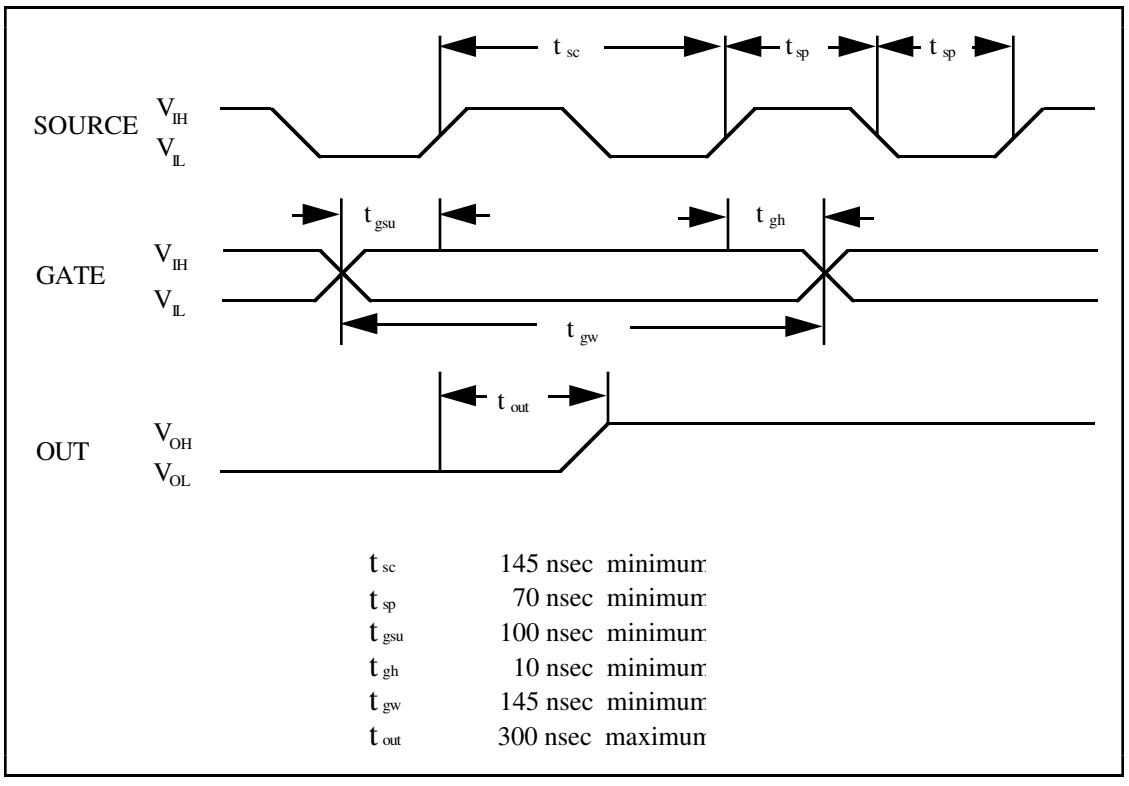


Figure 2-39. General-Purpose Timing Signals

The GATE and OUT signal transitions in Figure 2-39 are referenced to the rising edge of the SOURCE signal. This timing diagram assumes that the counters are programmed to count rising edges. The same timing diagram, with the source signal inverted and referenced to the falling edge of the source signal, applies to the case in which the counter is programmed to count falling edges.

The signal applied at a SOURCE input can be used as a clock source by any of the Am9513A counter/timers and by the Am9513A frequency division output FOUT. The signal applied to a SOURCE input must not exceed a frequency of 6 MHz for proper operation of the Am9513A. The Am9513A counters can be individually programmed to count rising or falling edges of signals applied at any of the Am9513A SOURCE or GATE input pins.

In addition to the signals applied to the SOURCE and GATE inputs, the Am9513A generates five internal timebase clocks from the clock signal supplied by the AT-MIO-16D. This clock signal is selected by the W5 jumper and then divided by 10. The factory default value is 1 MHz into the Am9513A (10-MHz clock signal on the AT-MIO-16D). The five internal timebase clocks can be used as counting sources, and these clocks have a maximum skew of 75 nsec between them. The SOURCE signal shown in Figure 2-38 represents any of the signals applied at the SOURCE inputs, GATE inputs, or internal timebase clocks. See Appendix E, *Am9513A Data Sheet*, for further details.

Specifications for signals at the GATE input are referenced to the signal at the SOURCE input or one of the Am9513A internally generated signals. Figure 2-39 shows the GATE signal referenced to the rising edge of a source signal. The gate must be valid (either high or low) at least 100 nsec before the rising or falling edge of a source signal for the gate to take effect at that source edge (as shown by t_{gsu} and t_{gh} in Figure 2-39). Similarly, the gate signal must be held for at least 10 nsec after the rising or falling edge of a source signal for the gate to take effect at that source edge. The gate high or low period must be at least 145 nsec in duration. If an internal timebase clock is used, the gate signal cannot be synchronized with the clock. In this case, gates applied close to a source edge take effect either on that source edge or on the next one. This arrangement provides an uncertainty of one source clock period with respect to unsynchronized gating sources.

Signals generated at the OUT output are referenced to the signal at the SOURCE input or to one of the Am9513A internally generated clock signals. Figure 2-39 shows the OUT signal referenced to the rising edge of a source signal. Any OUT signal state changes occur within 300 nsec after the source signal rising or falling edge.

DIO-24 I/O Connector Pin Description

The I/O connector contains 100 pins that can be split into two standard 50-pin connectors via a cable assembly such as a Type NB5 ribbon cable (see Figure 1-2). One 50-pin connector contains signals associated with the MIO-16 circuitry, while the other 50-pin connector contains signals for the DIO-24 circuitry.

Figure 2-40 shows the pin assignments for the DIO-24 circuitry I/O connector.

Warning: Connections that exceed any of the maximum ratings of input or output signals on the AT-MIO-16D may result in damage to the AT-MIO-16D board and to the PC. Maximum ratings for each signal are given in this chapter under the discussion of that signal. National Instruments is not liable for any damages resulting from any such signal connections.

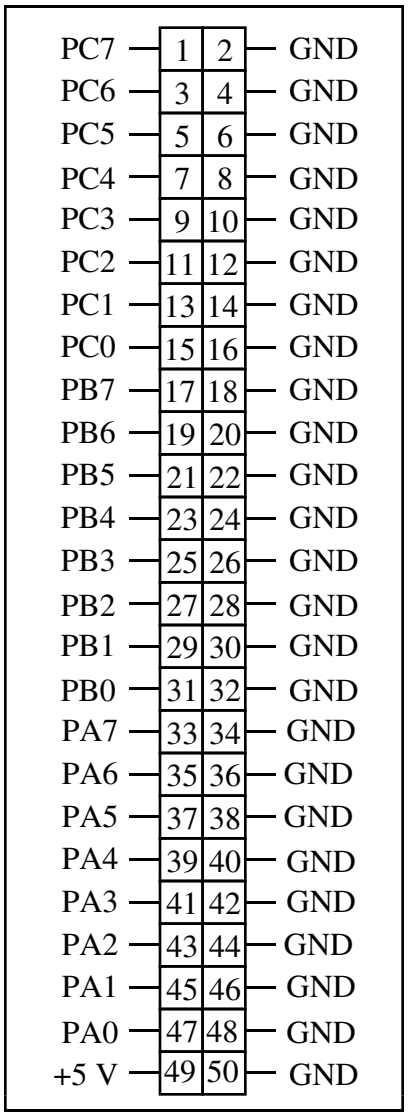


Figure 2-40. DIO-24 I/O Connector Pin Assignments

DIO-24 Signal Connection Descriptions

Pin	Signal Name	Reference	Description
1, 3, 5, 7, 9, 11, 13, 15	PC7 through PC0	DIGGND	Bidirectional data lines for Port C. PC7 is the MSB, PC0 the LSB.
17, 19, 21, 23, 25, 27, 29, 31	PB7 through PB0	DIGGND	Bidirectional data lines for Port B. PB7 is the MSB, PB0 the LSB.
33, 35, 37, 39, 41, 43, 45, 47	PA7 through PA0	DIGGND	Bidirectional data lines for Port A. PA7 is the MSB, PA0 the LSB.
49	+5 V	DIGGND	This pin provides +5 VDC.
All even-numbered pins	DIGGND		These signals are connected to the PC ground signal.

The absolute maximum voltage input rating is -0.5 to +7.0 V with respect to GND.

Power Connections

Pin 49 of the DIO-24 I/O connector provides +5 V from the PC AT power supply. This pin is referenced to DIG GND and can be used to power external digital circuitry.

Power rating: 1 A at +5 V ± 10%

Warning: This +5-V power pin *should not* be directly connected to analog or digital ground or to any other voltage source on the AT-MIO-16D or any other device. Doing so can damage the AT-MIO-16D and the PC AT. National Instruments is *not* liable for damages resulting from such a connection. A spare DIO-24 fuse is provided in case the power rating is inadvertently exceeded. You should use this fuse only after the cause of the initial problem is known, so as not to blow the spare fuse as well.

Port C Pin Assignments

The signals assigned to Port C depend on the mode in which the 82C55A is programmed. In Mode 0, Port C is considered two 4-bit I/O ports. In Modes 1 and 2, Port C is used for status and handshaking signals with two or three I/O bits mixed in. Table 2-11 summarizes the signal assignments of Port C for each programmable mode. See Chapter 4, *Programming*, for programming information.

Table 2-11. Port C Signal Assignments

Programming Mode	Group A					Group B		
	PC7	**PC6**	**PC5**	**PC4**	**PC3**	**PC2**	**PC1**	**PC0**
Mode 0	I/O	I/O	I/O	I/O	I/O	I/O	I/O	I/O
Mode 1 Input	I/O	I/O	IBF_A	STB_A*	$INTR_A$	STB_B*	$IBFB_B$	$INTR_B$
Mode 1 Output	OBF_A*	ACK_A*	I/O	I/O	$INTR_A$	ACK_B*	OBF_B*	$INTR_B$
Mode 2	OBF_A*	ACK_A*	IBF_A	STB_A*	$INTR_A$	I/O	I/O	I/O
* Indicates that the signal is active low.								

Timing Specifications

This section lists the timing specifications for handshaking with the DIO-24 circuitry. The handshaking lines STB* and IBF synchronize input transfers. The handshaking lines OBF* and ACK* synchronize output transfers.

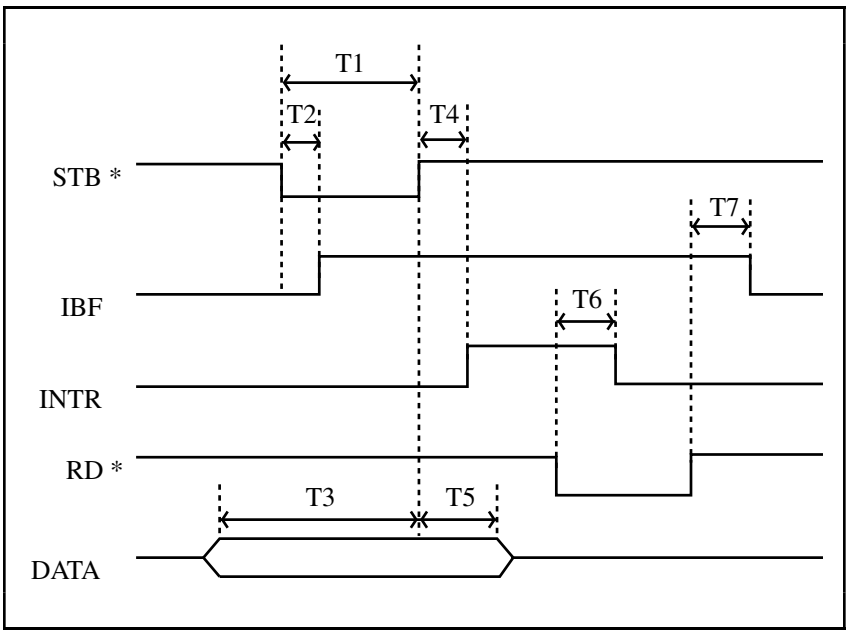
The following signals are used in the timing diagrams that follow.

Name	Type	Description
STB*	input	Strobe Input A low signal on this handshaking line loads data into the input latch.
IBF	output	Input Buffer Full A high signal on this handshaking line indicates that data has been loaded into the input latch. This is an input acknowledge signal.
ACK*	input	Acknowledge Input A low signal on this handshaking line indicates that the data written from the selected port has been accepted. This signal is a response from the external device that it has received the data from the AT-MIO-16D.
OBF*	output	Output Buffer Full A low signal on this handshaking line indicates that data has been written from the selected port.
INTR	output	Interrupt Request This signal becomes high when the 82C55A is requesting service during a data transfer. The appropriate DIO interrupt enable bits must be set to generate this signal.

Name	Type	Description (continued)
RD*	internal	Read Signal This signal is the read signal generated from the control lines of the PC.
WR*	internal	Write Signal This signal is the write signal generated from the control lines of the PC.
DATA	bidirectional	Data Lines at the Selected Port This signal indicates when the data on the data lines at a selected port is or should be available.

DIO-24 Mode 1 Input Timing

The following are the timing specifications for an input transfer in Mode 1:

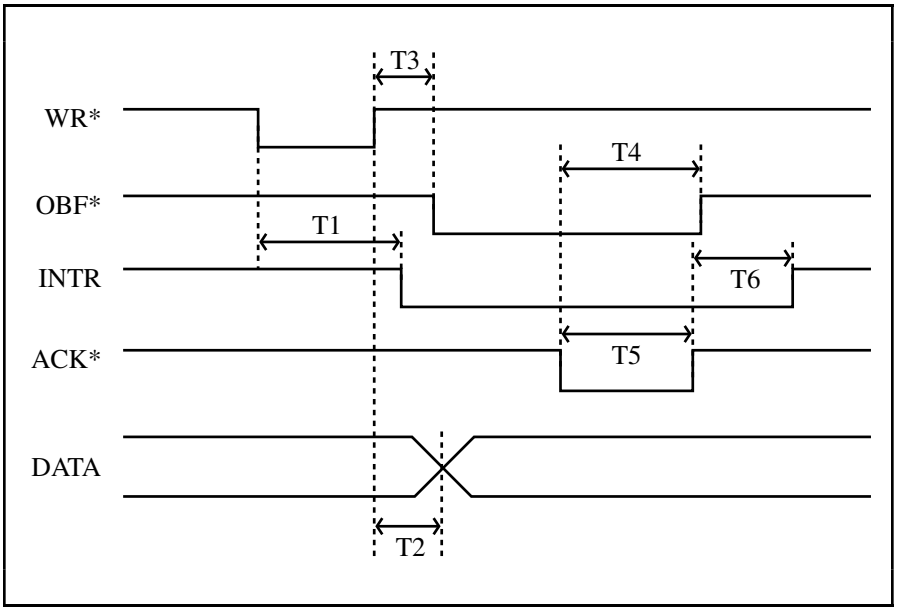


Name	Description	Minimum	Maximum
T1	STB* Pulse Width	500	–
T2	STB* = 0 to IBF = 1	–	300
T3	Data before STB*= 1	0	–
T4	STB* = 1 to INTR = 1	–	300
T5	Data after STB* = 1	180	–
T6	RD* = 0 to INTR = 0	–	400
T7	RD* = 1 to IBF = 0	–	300

All timing values are in nanoseconds.

DIO-24 Mode 1 Output Timing

The following are the timing specifications for an output transfer in Mode 1:

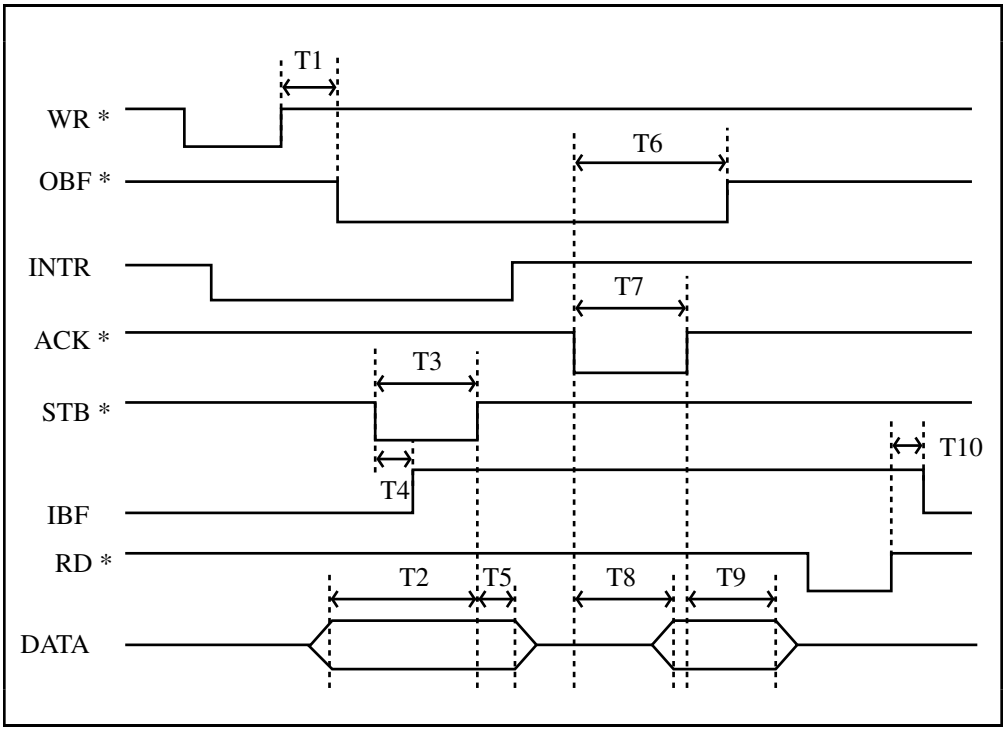


Name	Description	Minimum	Maximum
T1	WR* = 0 to INTR = 0	–	450
T2	WR* = 1 to Output	–	350
T3	WR* = 1 to OBF* = 0	–	650
T4	ACK* = 0 to OBF* = 1	–	350
T5	ACK* Pulse Width	300	–
T6	ACK* = 1 to INTR = 1	–	350

All timing values are in nanoseconds.

DIO-24 Mode 2 Bidirectional Timing

The following are the timing specifications for bidirectional transfers in Mode 2:



Name	Description	Minimum	Maximum
T1	WR* = 1 to OBF* = 0	–	650
T2	Data before STB*= 1	0	–
T3	STB* Pulse Width	500	–
T4	STB* = 0 to IBF = 1	–	300
T5	Data after STB* = 1	180	–
T6	ACK* = 0 to OBF = 1	–	350
T7	ACK* Pulse Width	300	–
T8	ACK* = 0 to Output	–	300
T9	ACK* = 1 to Output Float	20	250
T10	RD* = 1 to IBF = 0	–	300

All timing values are in nanoseconds.

Cabling and Field Wiring

This section discusses cabling and field wiring guidelines for the AT-MIO-16D board.

Field Wiring Considerations

Accuracy of measurements made with the AT-MIO-16D can be seriously affected by environmental noise if proper considerations are not taken into account when running signal wires between signal sources and the AT-MIO-16D board. The following recommendations mainly apply to analog input signal routing to the AT-MIO-16D board, though they are applicable for signal routing in general.

You can minimize noise pickup and maximize measurement accuracy by doing the following:

- Use individually shielded, twisted-pair wires to connect analog input signals to the AT-MIO-16D. With this type of wire, the signals attached to the CH+ and CH- inputs are twisted together and then covered with a shield. This shield is then connected only at one point to the signal source ground. This kind of connection is required for signals traveling through areas with large magnetic fields or high electromagnetic interference.
- Use differential analog input connections to reject common-mode noise.

The following recommendations apply for all signal connections to the AT-MIO-16D:

- Physically separate AT-MIO-16D signal lines from high-current or high-voltage lines. These lines are capable of inducing currents in or voltages on the AT-MIO-16D signal lines if they run in parallel paths at a close distance. Reduce the magnetic coupling between lines by separating them by a reasonable distance if they run in parallel, or by running the lines at right angles to each other.
- Do not run AT-MIO-16D signal lines through conduits that also contain power lines.
- Protect AT-MIO-16D signal lines from magnetic fields caused by electric motors, welding equipment, breakers, or transformers by running the AT-MIO-16D signal lines through special metal conduits.

MIO-16 Cabling Considerations

National Instruments has a cable termination accessory–the CB-100–for use with the AT-MIO-16D board. This kit includes two terminated 50-conductor flat ribbon cables and two 50-pin CB-50 connector blocks. You can attach signal I/O leads to screw terminals on the connector block and thereby be connected to the AT-MIO-16D I/O connector.

The CB-100 is useful for prototyping an application or in situations where AT-MIO-16D interconnections are frequently changed. Once you develop a final field wiring scheme, however, you may want to develop your own cable. This section contains information and guidelines for design of such a cable.

The MIO-16 circuitry I/O connector is a 50-pin female ribbon-cable header. The manufacturer part numbers for this header are as follows:

Electronic Products Division/3M	part number 3596-5002
T&B/Ansley Corporation	part number 609-5007

The mating connector for the MIO-16 circuitry is a 50-position ribbon socket connector, polarized, with strain relief. National Instruments uses a polarized (keyed) connector to prevent inadvertent upside-down connection to the AT-MIO-16D. Recommended manufacturer part numbers for this mating connector are as follows:

Electronic Products Division/3M	part number 3425-7650
T&B/Ansley Corporation	part number 609-5041CE

The following is the standard ribbon cable (50-conductor, 28 AWG, stranded) that can be used with these connectors:

Electronic Products Division/3M	part number 3365/50
T&B/Ansley Corporation	part number 171-50

In making your own cabling, you may decide to shield your cables. The following guidelines may help:

- For the analog input signals, shielded twisted-pair wires for each analog input pair yield the best results, assuming that differential inputs are used. Tie the shield for each signal pair to the ground reference at the source.
- The analog lines, pins 1 through 23, should be routed separately from the digital lines, pins 24 through 50.
- When using a cable shield, use separate shields for the analog and digital halves of the cable. Failure to do so will result in noise from switching digital signals coupling into the analog signals.

DIO-24 Cabling Considerations

The DIO-24 circuitry of the AT-MIO-16D can be interfaced to a wide range of printers, plotters, test instruments, I/O racks and modules, screw terminal panels, and almost any device with a parallel interface. The DIO-24 circuitry I/O connector is a standard 50-pin header connector. The pin assignments are compatible with the standard 24-channel I/O module mounting racks (such as those manufactured by Opto 22 and Gordos).

The CB-100 cable termination accessory is available from National Instruments for use with the DIO-24 circuitry of the AT-MIO-16D. This kit includes two 50-conductor flat ribbon cables and two 50-pin CB-50 connector blocks. Signal input and output wires can be attached to screw terminals on the connector block and are therefore connected to the DIO-section I/O connector.

The CB-100 is useful for initial prototyping of an application or in situations where DIO-section interconnections are frequently changed. Once you develop a final field wiring scheme, however, you may want to develop your own cable. This section contains information and guidelines for the design of custom cables.

The DIO-24 circuitry I/O connector is a 50-pin female ribbon-cable header. The manufacturers and the appropriate part numbers for this connector are as follows:

Electronic Products Division/3M	part number 3596-5002
T&B/Ansley Corporation	part number 609-5007

The mating connector for the DIO section is a 50-position, polarized, ribbon socket connector with strain relief. National Instruments uses a polarized (keyed) connector to prevent inadvertent upside-down connection to the DIO section . Recommended manufacturers and the appropriate part numbers for this mating connector are as follows:

Electronic Products Division/3M	part number 3425-7650
T&B/Ansley Corporation	part number 609-5041CE

The standard ribbon cable (50-conductor, 28 AWG, stranded) that can be used with these connectors is as follows:

Electronic Products Division/3M	part number 3365/50
T&B/Ansley Corporation	part number 171-50

If you plan to use the DIO section of the AT-MIO-16D for a communications application, you may need shielded cables to meet FCC requirements. The DIO-section I/O bracket has been designed so that the shield of the I/O cable can be grounded through the computer chassis when a mating connector such as the following is used:

AMP Special Industries	part number 2-746483-2

Many varieties of shielded ribbon cable are available to work with the mating connector listed previously. One type of shielded cable encloses a standard ribbon cable with a shielded jacket. Recommended manufacturers and the appropriate part numbers for this type of cable are as follows:

Belden Electronic Wire and Cable	part number 9L28350
T&B/Ansley Corporation	part number 187-50

Chapter 3
Theory of Operation

This chapter contains a functional overview of the AT-MIO-16D and explains the operation of each functional unit making up the AT-MIO-16D.

MIO-16 Functional Overview

The block diagram in Figure 3-1 is a functional overview of the MIO-16 circuitry of the AT-MIO-16D board.

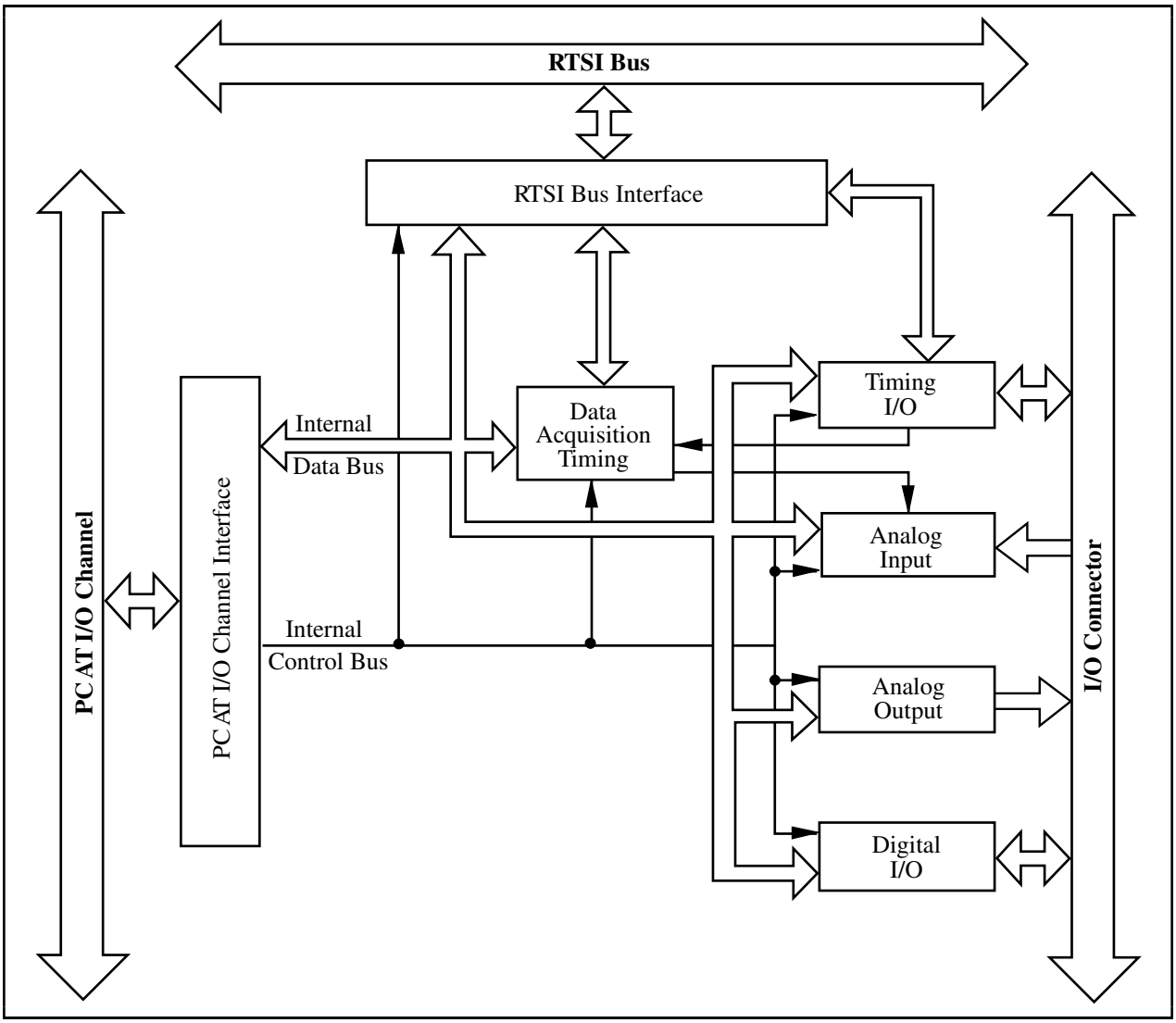


Figure 3-1. AT-MIO-16D MIO-16 Circuitry Block Diagram

The following are the major components making up the MIO-16 section of the AT-MIO-16D board:

- PC AT I/O channel interface circuitry
- Analog input and data acquisition circuitry
- Analog output circuitry
- Digital I/O circuitry
- Timing I/O circuitry
- RTSI bus interface circuitry

The internal data and control buses interconnect the components. The theory of operation of each of these components is explained in the remainder of this chapter.

PC AT I/O Channel Interface Circuitry

The AT-MIO-16D board is a full-size 16-bit PC AT I/O channel adapter. The PC AT I/O channel consists of a 24-bit address bus, a 16-bit data bus, a direct memory access (DMA) arbitration bus, interrupt lines, and several control and support signals. The components making up the AT-MIO-16D PC AT I/O channel interface circuitry are shown in Figure 3-2.

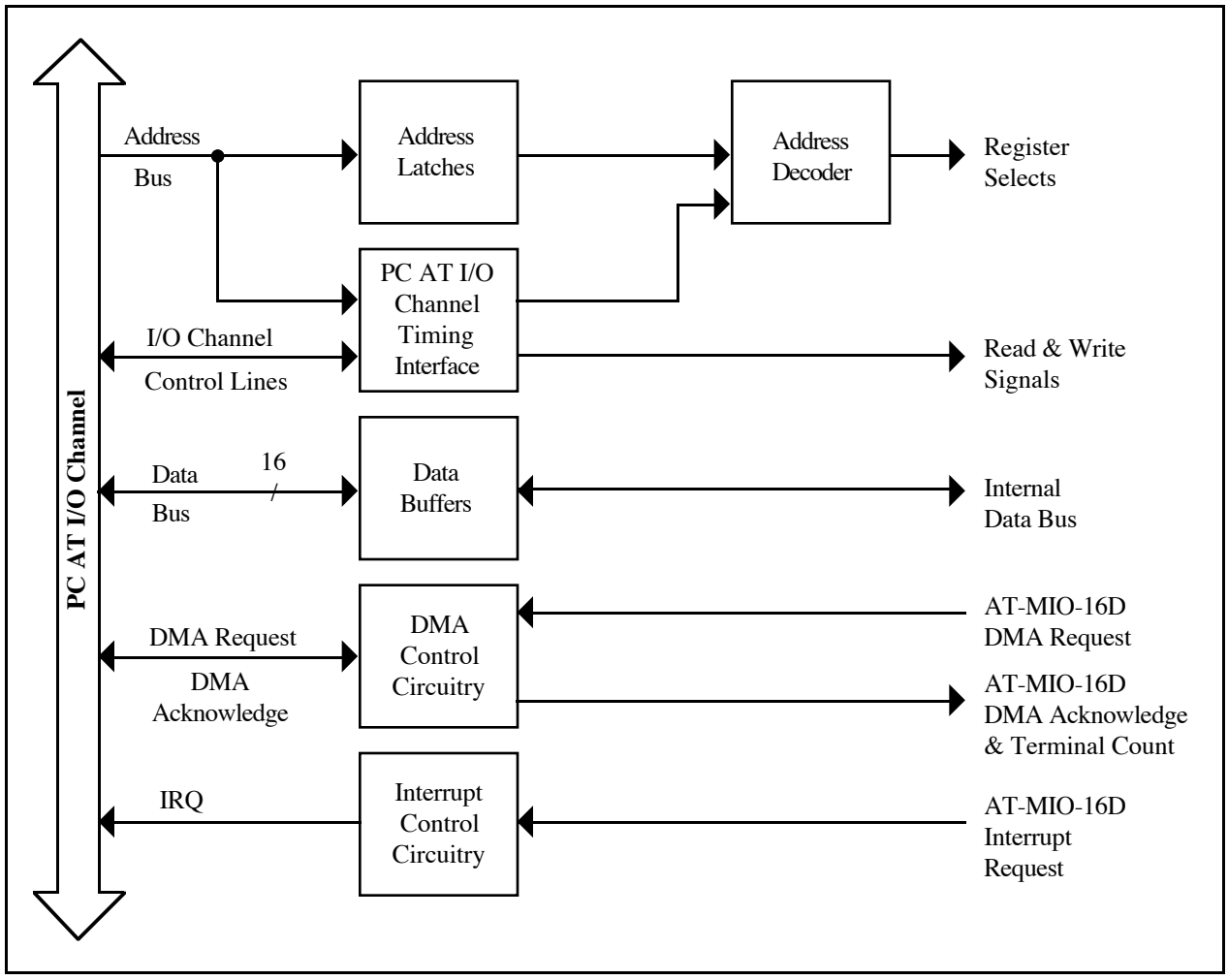


Figure 3-2. PC AT I/O Channel Interface Circuitry Block Diagram

The PC AT I/O channel interface circuitry consists of address latches, address decoder circuitry, data buffers, PC AT I/O channel interface timing signals, interrupt circuitry, and DMA arbitration circuitry. The PC AT I/O channel interface circuitry generates the signals necessary to control and monitor the operation of the AT-MIO-16D multiple function circuitry.

The PC AT I/O channel has 24 address lines; the AT-MIO-16D uses 10 of these lines to decode the board address. Therefore, the board address range is hex 000 to 3FF. SA5 through SA9 are used to generate the board enable signal. SA0 through SA4 are used to select onboard registers. These address lines are latched by the address latches at the beginning of an I/O transfer. The latched address lines send the same address to the address-decoding circuitry during the entire I/O transfer cycle. The address-decoding circuitry generates the register select signals that specify which AT-MIO-16D register is being accessed. The data buffers control the direction of data transfer on the bidirectional data lines based on whether the transfer is a read or a write.

The PC AT I/O channel interface timing signals are used to generate read-and-write signals and to define the transfer cycle. A transfer cycle can be either an 8-bit or a 16-bit data I/O operation. The AT-MIO-16D returns signals to the PC AT I/O channel to indicate when the board has been

accessed, when the board is ready for another transfer, and the data bit size of the current I/O transfer.

The interrupt control circuitry routes any enabled interrupt requests to the selected interrupt request line. The interrupt requests are tristate output signals allowing the AT-MIO-16D board to share the interrupt lines with other devices. Eleven interrupt request lines are available for use by the AT-MIO-16D: IRQ3, IRQ4, IRQ5, IRQ6, IRQ7, IRQ9, IRQ10, IRQ11, IRQ12, IRQ14, and IRQ15. Five different interrupts can be generated by the MIO-16 circuitry of the AT-MIO-16D:

- When an A/D conversion is available to be read from the A/D FIFO memory
- When a data acquisition operation completes
- When a DMA terminal count pulse is received
- When a rising edge signal is detected on the OUT2 pin of the Am9513A Counter/Timer
- When either an OVERFLOW or an OVERRUN error occurs

Each one of these interrupts is individually enabled and cleared. See Chapter 4, *Programming*, for more information about programming with interrupts.

The DMA control circuitry generates DMA requests whenever an A/D measurement is available from the A/D FIFO, if the DMA transfer is enabled. The DMA circuitry supports full PC AT I/O channel 16-bit DMA transfers. DMA channels 5, 6, and 7 of the PC AT I/O channel are available for such transfers. With the DMA circuitry, either single-channel transfer mode or dual-channel transfer mode can be selected for DMA transfer.

Analog Input and Data Acquisition Circuitry

The AT-MIO-16D handles 16 channels of analog input with software-programmable gain and 12-bit A/D conversion. In addition, the AT-MIO-16D contains data acquisition circuitry for automatic timing of multiple A/D conversions and includes advanced options such as external triggering, gating, and clocking. Figure 3-3 shows a block diagram of the analog input and data acquisition circuitry.

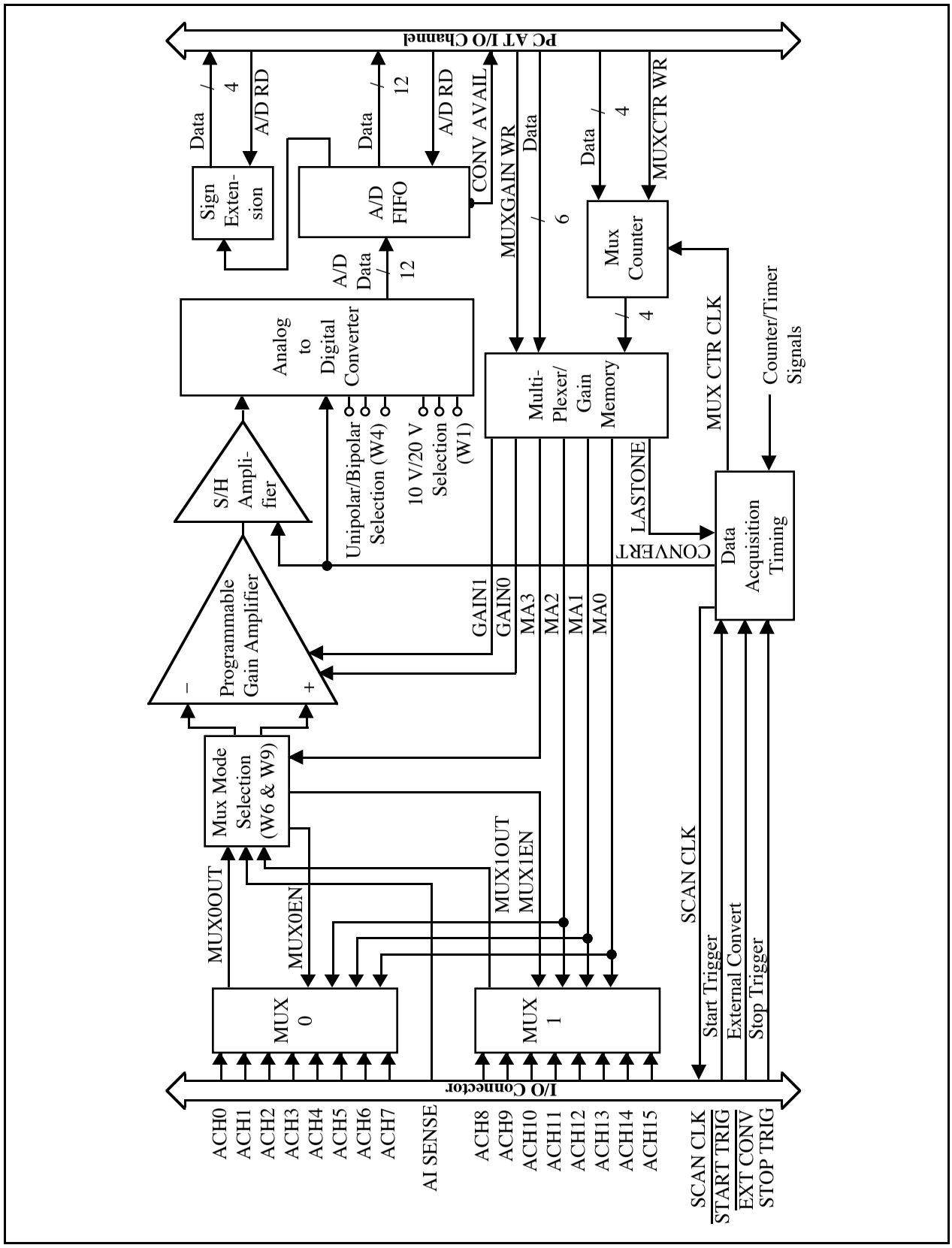


Figure 3-3. Analog Input and Data Acquisition Circuitry Block Diagram

Analog Input Circuitry

The analog input circuitry consists of an input multiplexer, multiplexer-mode selection jumpers, a software-programmable gain instrumentation amplifier, a sample-and-hold amplifier, a 12-bit analog-to-digital converter (ADC), and a 12-bit FIFO with a 16-bit sign extension option.

Analog Input Multiplexers

The input multiplexer consists of two CMOS analog input multiplexers and has 16 analog input channels. Multiplexer MUX0 is connected to analog input channels 0 through 7. Multiplexer MUX1 is connected to analog input channels 8 through 15. The input multiplexers provide input overvoltage protection of ±35 V powered on and ±20 V powered off.

Analog Input Mode Selection

The multiplexer-mode selection jumpers configure the analog input channels as 16 single-ended inputs or 8 differential inputs. When single-ended mode is selected, the outputs of the two multiplexers are tied together and routed to the positive (+) input of the instrumentation amplifier. The negative (-) input of the instrumentation amplifier is tied to the AT-MIO-16D ground for RSE input or to the analog return of the input signals via the AI SENSE input on the I/O connector for NRSE input. When DIFF mode is selected, the output of MUX0 is routed to the positive (+) input of the instrumentation amplifier, and the output of MUX1 is routed to the negative (-) input of the instrumentation amplifier.

The Instrumentation Amplifier

The instrumentation amplifier fulfills two purposes on the AT-MIO-16D board. It converts a differential input signal into a single-ended signal with respect to the AT-MIO-16D ground for a minimum input common-mode rejection ratio of 85 dB. This conversion allows the input analog signal to be extracted from any common-mode voltage or noise before being sampled and converted. The instrumentation amplifier also applies gain to the input signal, allowing an input analog signal to be amplified before being sampled and converted, and thus increasing measurement resolution and accuracy. The gain of the instrumentation amplifier is selected under software control. The AT-MIO-16DL (L stands for low-level signals) provides gains of 1, 10, 100, and 500. The AT-MIO-16DH (H stands for high-level signals) provides gains of 1, 2, 4, and 8.

Channel Selection Circuitry

Selection of the analog input channel and the gain settings is controlled by the mux-gain memory. The mux-gain memory provides two gain control bits to the instrumentation amplifier and four multiplexer address bits to the input multiplexers and multiplexer-mode selection circuitry that select the analog input channels. Operation of the mux-gain memory is explained in more detail in the *Data Acquisition Timing Circuitry* section later in this chapter.

The sample-and-hold amplifier aids the ADC in performing A/D conversions. At the beginning of an A/D conversion, the sample-and-hold amplifier is put in hold mode, which means that it holds its output voltage at a steady value (the value when the hold period started) regardless of voltage changes at its input. This sample-and-hold amplifier provides the ADC with a steady voltage while

it is performing an A/D conversion. Without the sample-and-hold amplifier, the analog input signal could change during a conversion, thereby causing errors during A/D conversion. By isolating the ADC from the analog input signals during conversion, you can change the input multiplexer and allow the instrumentation amplifier to settle to a new value while the ADC is converting the old value. This isolation creates a two-stage pipeline and increases and optimizes the performance of the analog input circuitry during high-speed, multiple A/D conversions.

A/D Converter

The ADC is a 12-bit, successive-approximation ADC with a maximum conversion time of 9 µsec. The 12-bit resolution allows the converter to resolve its input range into 4,096 different steps. This resolution also provides a 12-bit digital word that represents the value of the input voltage level with respect to the converter input range. The ADC supports three input ranges that are jumper-selectable on the AT-MIO-16D board, -10 to +10 V, -5 to +5 V, or 0 to +10 V.

ADC FIFO Buffer

When an A/D conversion is complete, the ADC clocks the result into the A/D FIFO. The A/D FIFO is 12 bits wide and 512 words deep. This FIFO serves as a buffer to the ADC and provides two benefits. Any time an A/D conversion is complete, the value is saved in the A/D FIFO for later reading, and the ADC is free to start a new conversion. Secondly, the A/D FIFO can collect up to 512 A/D conversion values before any information is lost; thus software or DMA has extra time (512 times the sample interval) to catch up with the hardware. If more than 512 values are stored in the A/D FIFO without the A/D FIFO being read from, an error condition called A/D FIFO overflow occurs and A/D conversion information is lost.

The A/D FIFO generates a signal that indicates when it contains A/D conversion data. You can read the state of this signal from the AT-MIO-16D Status Register. You can use this signal to generate a DMA request signal or to generate an interrupt. Sign-extension circuitry at the A/D FIFO output adds four most significant bits (MSBs), bits 15 through 12, to the 12-bit FIFO output (bits 11 through 0) to produce a 16-bit result.

The sign-extension circuitry is software programmable to generate either straight binary numbers or two's complement numbers. In straight binary mode, bits 15 through 12 are always zero and provide a range of 0 to 4,095. In two's complement mode, the MSB of the 12-bit ADC result, bit 11, is inverted and extended to bits 15 through 12, providing a range of -2,048 to +2,047.

Data Acquisition Timing Circuitry

A data acquisition operation refers to the process of taking a sequence of A/D conversions with the sample interval (the time between successive A/D conversions) carefully timed. The data acquisition timing circuitry consists of various clocks and timing signals. Three types of data acquisition are supported by the AT-MIO-16D board–single-channel data acquisition, multiple-channel data acquisition with continuous scanning, and multiple-channel data acquisition with interval scanning.

Scanned data acquisition uses the multiplexer counter and the mux-gain memory to automatically switch between analog input channels during data acquisition. Continuous scanning cycles through the mux-gain memory without any delays between cycles. Interval scanning assigns a

time interval called the *scan interval* to each cycle through the mux-gain memory. The scan interval is basically the time between starts for each cycle through the mux-gain memory.

Data acquisition timing consists of signals that initiate a data acquisition operation, initiate individual A/D conversions, gate the data acquisition operation, and generate scanning clocks. The sources for these signals can be supplied by timers on the AT-MIO-16D board, by signals connected to the AT-MIO-16D I/O connector, or by signals from other AT Series boards connected to the RTSI bus.

Single Conversions

You can initiate single A/D conversions by applying an active low pulse to the EXTCONV* input on the I/O connector or by writing to the Start Convert Register on the AT-MIO-16D board. During data acquisition, the onboard sample-interval counter (Counter 3 of the Am9513A Counter/Timer) generates pulses that initiate A/D conversions. External control of the sample interval is possible by applying a stream of pulses at the EXTCONV* input. In this case, you have complete external control over the sample interval and the number of A/D conversions performed.

Sample-Interval Timer

The sample-interval timer is a 16-bit down counter that can be used with the five internal timebases of the Am9513A to generate sample intervals from 2 µsec to 6 sec (see the *Timing I/O Circuitry* section later in this chapter). The sample-interval timer can also use any of the external clock inputs to the Am9513A as a timebase. During data acquisition, the sample interval counts down at the rate given by the internal timebase or external clock. Each time the sample-interval timer reaches zero, it generates a pulse and reloads with the programmed sample-interval count. This operation continues until data acquisition halts.

Sample Counter

The onboard sample counter can control data acquisition. Load this counter with the number of samples to be taken during a data acquisition operation. The sample counter can be 16-bit for counts up to 65,535 or 32-bit for counts up to $(2^{32} - 1)$. If a 16-bit counter is needed, Counter 4 of the Am9513A Counter/Timer is used. If more than 16 bits are needed, Counter 4 is concatenated with Counter 5 of the Am9513A to form a 32-bit counter. The sample counter decrements its count each time the sample-interval counter generates an A/D conversion pulse, and the sample counter stops the data acquisition process when it counts down to zero.

You can trigger the sample counter externally with the STOP TRIG input on the AT-MIO-16D I/O connector. The counter does not begin counting the A/D conversion pulses until a rising edge signal occurs on STOP TRIG. With this method, A/D conversion samples can be collected both before and after a hardware trigger is received.

You can initiate the data acquisition process by writing to the Start DAQ Register on the AT-MIO-16D board or by applying an active low pulse to the START TRIG* input on the AT-MIO-16D I/O connector. These triggers start the sample-interval and sample counters. The sample-interval counter then manages the data acquisition process until the sample counter reaches zero.

Single-Channel Data Acquisition

During single-channel data acquisition, the mux-gain memory is set up to select the gain and analog input channel before data acquisition is initiated. These gain and multiplexer settings remain constant during the entire data acquisition process; therefore, all A/D conversion data is read from a single channel.

Multiple-Channel (Scanned) Data Acquisition

You perform multiple-channel data acquisition by enabling scanning during data acquisition. You control multiple-channel scanning with the multiplexer counter and the mux-gain memory.

The mux-gain memory consists of 16 words of memory. Each word of memory contains a multiplexer address (4 bits) for input analog channel selection, a gain setting (2 bits), and a bit indicating if the entry is the last in the scan sequence. The mux-gain memory address is controlled by the multiplexer counter. Whenever a mux-gain memory address location is selected, the multiplexer and gain control bits contained in that memory location are applied to the analog input circuitry. For scanning operations, the multiplexer counter steps through successive locations in the mux-gain memory at a rate determined by the scan clock. With the mux-gain memory, therefore, an arbitrary sequence of channels (16 maximum) with a separate gain setting for each channel can be clocked through during a scanning operation.

Both the multiplexer counter and the mux-gain memory can be directly written to through AT-MIO-16D registers. For writing purposes, the multiplexer counter serves as a pointer to the mux-gain memory. The counter can be loaded with any 4-bit value to point to any mux-gain memory location. With this counter, scanning can start at any location in the mux-gain memory.

The SCAN CLK signal is generated from the sample-interval counter. This signal pulses once at the beginning of each A/D conversion and is supplied at the I/O connector. During multiple-channel scanning, the multiplexer counter is incremented repeatedly, thereby sequencing through the mux-gain memory and automatically selecting new channel and gain settings during data acquisition. The MUX CTR CLK signal is generated from the SCAN CLK and provides the pulses that increment the multiplexer counter. MUX CTR CLK can be identical to SCAN CLK, incrementing the multiplexer counter once after every A/D conversion. MUX CTR CLK can also be generated by dividing SCAN CLK by Counter 1 of the Am9513A Counter/Timer. With this method, the multiplexer counter can be incremented once every *N* A/D conversions such that *N* conversions can be performed on a single channel and gain selection before switching to the next channel and gain selection.

Data Acquisition Rates

Data acquisition rates (number of samples per second) are determined by the conversion period of the ADC plus the sample-and-hold acquisition time. During multiple-channel scanning, the data acquisition rates are further limited by the settling time of the input multiplexers and instrumentation amplifier. After the input multiplexers are switched, the instrumentation amplifier should be allowed to settle to the new input signal value before an A/D conversion is performed or else high accuracy will not be achieved. The settling time is determined by the gain selected.

Table 3-1 shows the maximum recommended data acquisition rates for both single-channel and multiple-channel data acquisition. The rates in Table 3-1 refer to typical settling accuracies of 0.5 LSBs of the final value.

Table 3-1. AT-MIO-16D Maximum Recommended Data Acquisition Rates

Data Acquisition Type	Gain	Data Acquisition Rate
Single-channel data acquisition	Any gain setting	100 ksamples/sec
Multiple-channel data acquisition	Gain = 1, 2, 4, 8 Gain = 10 Gain = 100 Gain = 500	100 ksamples/sec 100 ksamples/sec 70 ksamples/sec 20 ksamples/sec

Analog Output Circuitry

The AT-MIO-16D provides two channels of 12-bit digital-to-analog (D/A) output. Each analog output channel provides options such as unipolar or bipolar output and internal or external reference voltage selection. Figure 3-4 shows a block diagram of the analog output circuitry.

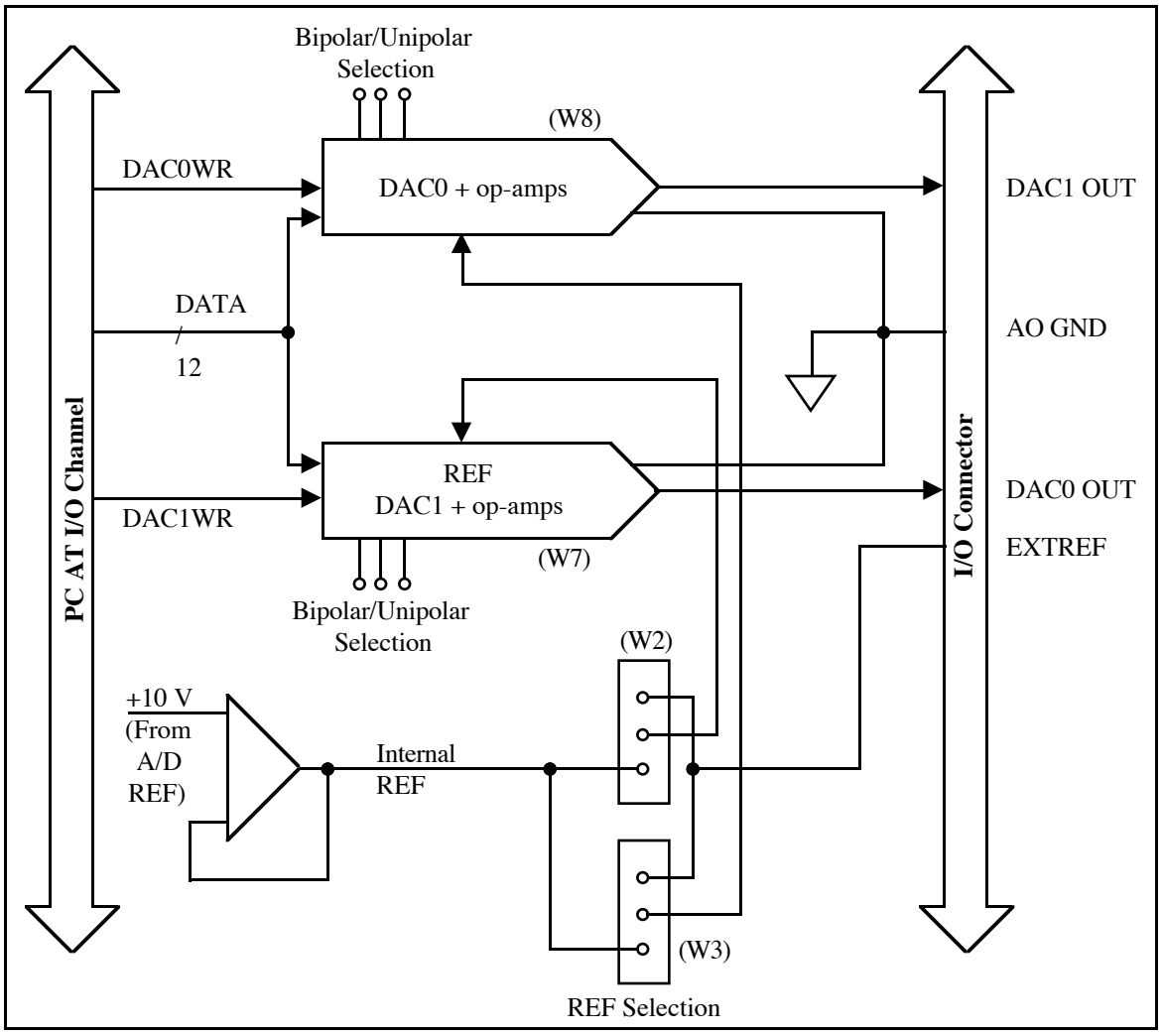


Figure 3-4. Analog Output Circuitry Block Diagram

Each analog output channel contains a 12-bit digital-to-analog converter (DAC), output operational amplifiers (op-amps), reference selection jumpers, and unipolar/bipolar output selection jumpers.

The DAC in each analog output channel generates a current proportional to the input voltage reference (V_{ref}) multiplied by the digital code loaded into the DAC. Each DAC can be loaded with a 12-bit digital code by writing to registers on the AT-MIO-16D board. The output op-amps convert the DAC current output to a voltage output provided at the AT-MIO-16D I/O connector DAC0 OUT and DAC1 OUT pins. The analog output of the DACs is updated to reflect the loaded 12-bit digital code in one of two ways: immediately when the 12-bit code is written to the DACs, or when an active low pulse occurs on the Am9513A OUT2 pin. The update method used is selected by the LDAC bit in Command Register 2.

Analog Output Range

The DAC output op-amps can be jumper configured to provide either a unipolar voltage output or a bipolar voltage output range. A unipolar output has an output voltage range of 0 to $+V_{ref}$ - 1 LSB V. A bipolar output provides an output voltage range of $-V_{ref}$ to $+V_{ref}$ -1 LSB V. For unipolar output, 0 V output corresponds to a digital code word of zero. For bipolar output, the form of the digital code input is jumper selectable. If straight binary form is selected, 0 V output corresponds to a digital code word of 2,048. If two's complement form is selected, 0 V output corresponds to a digital code word of zero. One LSB is the voltage increment corresponding to an LSB change in the digital code word. For unipolar output, 1 LSB = $(V_{ref})/4,096$. For bipolar output, 1 LSB = $(V_{ref})/2,048$.

Analog Output Data Coding

The voltage reference source for each DAC is jumper selectable and can be supplied either externally at the EXTREF input or internally. The external reference can be either a DC or an AC signal. If an AC reference is applied, the analog output channel acts as a signal attenuator, and the AC signal appears at the output attenuated by the digital code divided by 4,096 for unipolar output.

Bipolar output with an AC reference provides four-quadrant multiplication, which means that the signal is inverted for digital codes 0 to 2,047 and not inverted for digital codes 2,049 to 4,095. In straight binary mode, a digital code word of 2,048 attenuates the input signal to 0 V. This attenuation is equivalent to multiplying the signal by (digital code word - 2,048)/+2,048. In two's complement mode, a digital code word of zero attenuates the input signal to 0 V.

The internal voltage reference is a buffered version of the 10 V reference supplied by the ADC. Using the internal reference supplies an output voltage range of 0 to 9.9976 V in steps of 2.44 mV for unipolar output and an output voltage range of -10 V to +9.9951 V in steps of 4.88 mV for bipolar output.

MIO-16 Digital I/O Circuitry

The MIO-16 circuitry of the AT-MIO-16D provides eight digital I/O lines, while the DIO-24 circuitry provides 24 lines of digital I/O (discussed later in this chapter). The eight lines of digital I/O from the MIO-16 circuitry are divided into two ports of four lines each and are located at pins ADIO<3..0> and BDIO<3..0> on the I/O connector. Figure 3-5 shows a block diagram of the digital I/O circuitry.

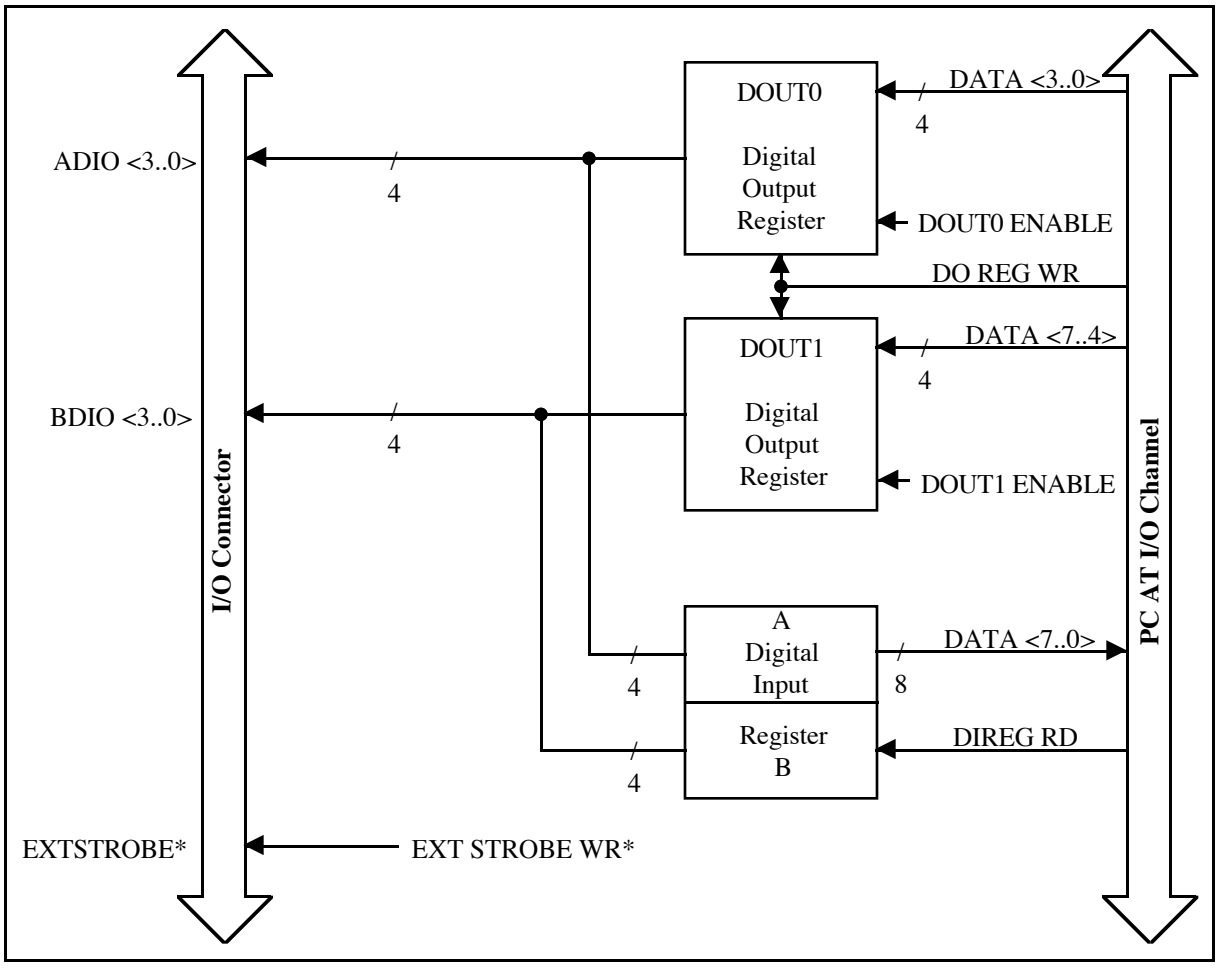


Figure 3-5. Digital I/O Circuitry Block Diagram

The digital I/O lines are controlled by the Digital Output Register and monitored by the Digital Input Register. The Digital Output Register is an 8-bit register that contains the digital output values for both ports 0 and 1. When port 0 is enabled, bits <3..0> in the Digital Output Register are driven onto digital output lines ADIO<3..0>. When port 1 is enabled, bits <7..4> in the Digital Output Register are driven onto digital output lines BDIO<3..0>.

Reading the Digital Input Register returns the state of the digital I/O lines. Digital I/O lines ADIO<3..0> are connected to bits <3..0> of the Digital Input Register. Digital I/O lines BDIO<3..0> are connected to bits <7..4> of the Digital Input Register. When a port is enabled, the Digital Input Register serves as a read-back register, returning the digital output value of the port. When a port is not enabled, reading the Digital Input Register returns the state of the digital I/O lines as driven by an external device.

Both the digital input and output registers are TTL-compatible. The digital output ports, when enabled, are capable of sinking 24 mA of current and sourcing 2.6 mA of current on each digital I/O line. When the ports are not enabled, the digital I/O lines act as high-impedance inputs.

The external strobe signal EXTSTROBE*, shown in Figure 3-5, is a general-purpose strobe signal. Writing to an address location on the AT-MIO-16D board generates an active-low 200-nsec

pulse on this output pin. EXTSTROBE* is not necessarily part of the digital I/O circuitry but is shown here because it can be used to latch digital output from the AT-MIO-16D into an external device.

Timing I/O Circuitry

The AT-MIO-16D uses an Am9513A Counter/Timer for data acquisition timing and for general-purpose timing I/O functions. An onboard oscillator is used to generate the 10-MHz clock. Figure 3-6 shows a block diagram of the timing I/O circuitry.

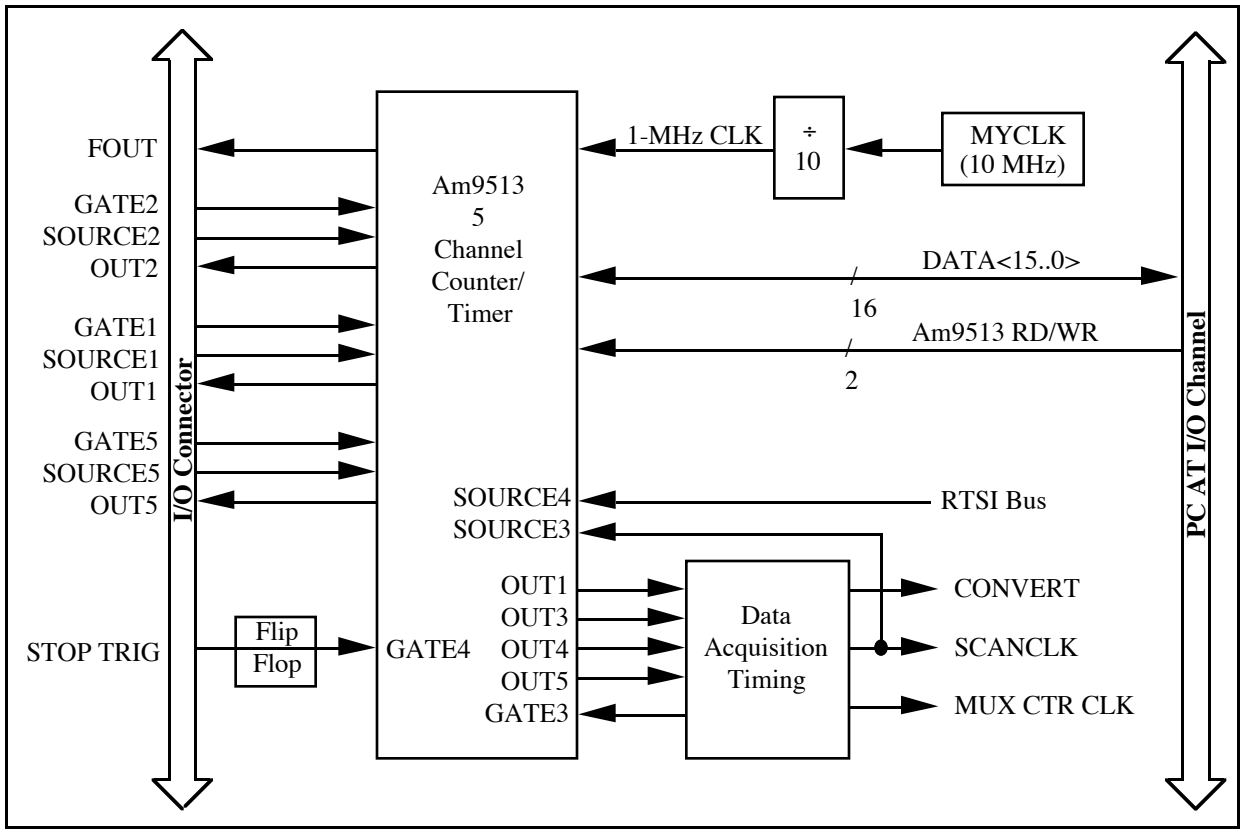


Figure 3-6. Timing I/O Circuitry Block Diagram

The Am9513A contains five independent 16-bit counter/timers, a 4-bit frequency output channel, and five internally generated timebases. The five counter/timers can be programmed to operate in several useful timing modes. The programming and operation of the Am9513A is presented in detail in Appendix E, *Am9513A Data Sheet*.

The Am9513A clock input is one-tenth the MYCLK frequency selected by the W5 jumpers. The factory default for MYCLK is 10 MHz, which generates a 1-MHz clock input to the Am9513A. The Am9513A uses this clock input to generate five internal timebases. These timebases can be used as clocks by the counter/timers and by the frequency output channel. When MYCLK is 10 MHz, the five internal timebases normally used for AT-MIO-16D timing functions are 1 MHz, 100 kHz, 10 kHz, 1 kHz, and 100 Hz.

The 16-bit counters in the Am9513A can be diagrammed as shown in Figure 3-7.

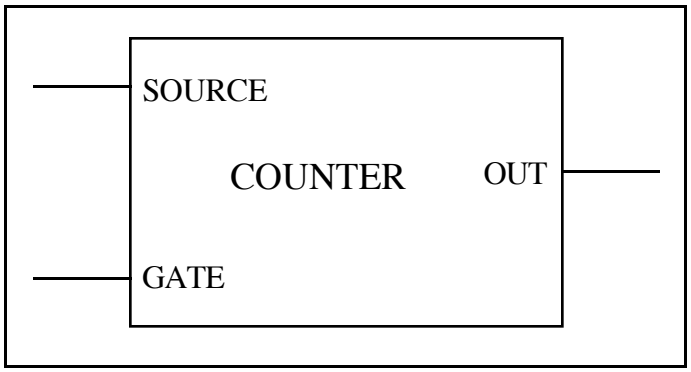


Figure 3-7. Counter Block Diagram

Each counter has a SOURCE input pin, a GATE input pin, and an output pin labeled OUT. The Am9513A counters are numbered 1 through 5, and their GATE, SOURCE, and OUT pins are labeled GATE *N*, SOURCE *N*, and OUT *N*, where *N* is the counter number.

For counting operations, the counters can be programmed to use any of the five internal timebases, any of the five GATE and five SOURCE inputs to the Am9513A, and the output of the previous counter (Counter 4 uses Counter 3 output, and so on). A counter can be configured to count either falling or rising edges of the selected input.

The counter GATE input allows counter operation to be gated. Once a counter is configured for an operation through software, a signal at the GATE input can be used to start and stop counter operation. There are five gating modes supported by the Am9513A: no gating, level gating active high, level gating active low, low-to-high edge gating, and high-to-low edge gating. A counter can also be active high level gated by a signal at GATE *N*+1 and GATE *N*-1, where *N* is the counter number.

The counter generates timing signals at its OUT output pin. The OUT output pin can also be set to a high-impedance state or a grounded-output state. The counters generate two types of output signals during counter operation: terminal count pulse output and terminal count toggle output. Terminal count is often referred to as TC. A counter reaches TC when it counts up or down and rolls over. In many counter applications, the counter reloads from an internal register when it reaches TC. In TC pulse output mode, the counter generates a pulse during the cycle that it reaches TC and reloads. In TC toggle output mode, the counter output changes state after it reaches TC and reloads. In addition, the counters can be configured for positive logic output or negative (inverted) logic output for a total of four possible output signals generated for one timing mode.

The SOURCE, GATE, and OUT pins for Counters 1, 2, and 5 of the onboard Am9513A are located on the AT-MIO-16D I/O connector. A rising edge signal on the STOP TRIG pin of the I/O connector sets the flip-flop output signal connected to the GATE4 input of the Am9513A and can be used as an additional gate input. The flip-flop output connected to GATE4 is cleared when the sample counter reaches TC, when an overflow or overrun occurs, or when the A/D Clear Register is written to.

The Am9513A SOURCE5 pin is connected to the AT-MIO-16D RTSI switch, which means that a signal from the RTSI trigger bus can be used as a counting source for the Am9513A counters.

The Am9513A OUT2 pin can be used in several different ways. If the LDAC bit is set in Command Register 2, an active low pulse on OUT2 updates the analog output on the two DACs. OUT2 can also be used to trigger interrupt requests. If INT2EN bit is set, an interrupt occurs when a rising edge signal is detected on OUT2. This interrupt can be used to update the DACs or to interrupt on an external signal connected to OUT2 through the I/O connector.

Counters 3 and 4 of the Am9513A are dedicated to data acquisition timing and therefore are not made available for general-purpose timing applications. Signals generated at OUT3 and OUT4 are provided to the data acquisition timing circuitry. GATE3 is controlled by the data acquisition timing circuitry.

Counter 5 is sometimes used by the data acquisition timing circuitry and concatenated with Counter 4 to form a 32-bit sample counter. The SCAN CLK signal is connected to the SOURCE3 input of the Am9513A, and OUT1 is provided to the data acquisition timing circuitry. This allows Counter 1 to be used to divide the SCAN CLK signal for generating the MUX CTR CLK signal (see the *Data Acquisition Timing Circuitry* section earlier in this chapter).

Counter 2 is sometimes used by the data acquisition timing circuitry to assign a time interval to each cycle through the scan sequence programmed in the mux-gain memory. This mode is called interval channel scanning. See the *Multiple-Channel (Scanned) Data Acquisition* section earlier in this chapter.

The Am9513A 3-bit programmable frequency output channel is provided at the I/O connector FOUT pin. Any of the five internal timebases and any of the counter SOURCE or GATE inputs can be selected as the frequency output source. The frequency output channel divides the selected source by its 4-bit programmed value and provides the divided down signal at the FOUT pin.

RTSI Bus Interface Circuitry

The AT-MIO-16D is interfaced to the National Instrument RTSI bus. The RTSI bus has seven trigger lines and a system clock line. All National Instruments AT Series boards with RTSI bus connectors can be wired together inside the PC AT and share these signals. A block diagram of the RTSI bus interface circuitry is shown in Figure 3-8.

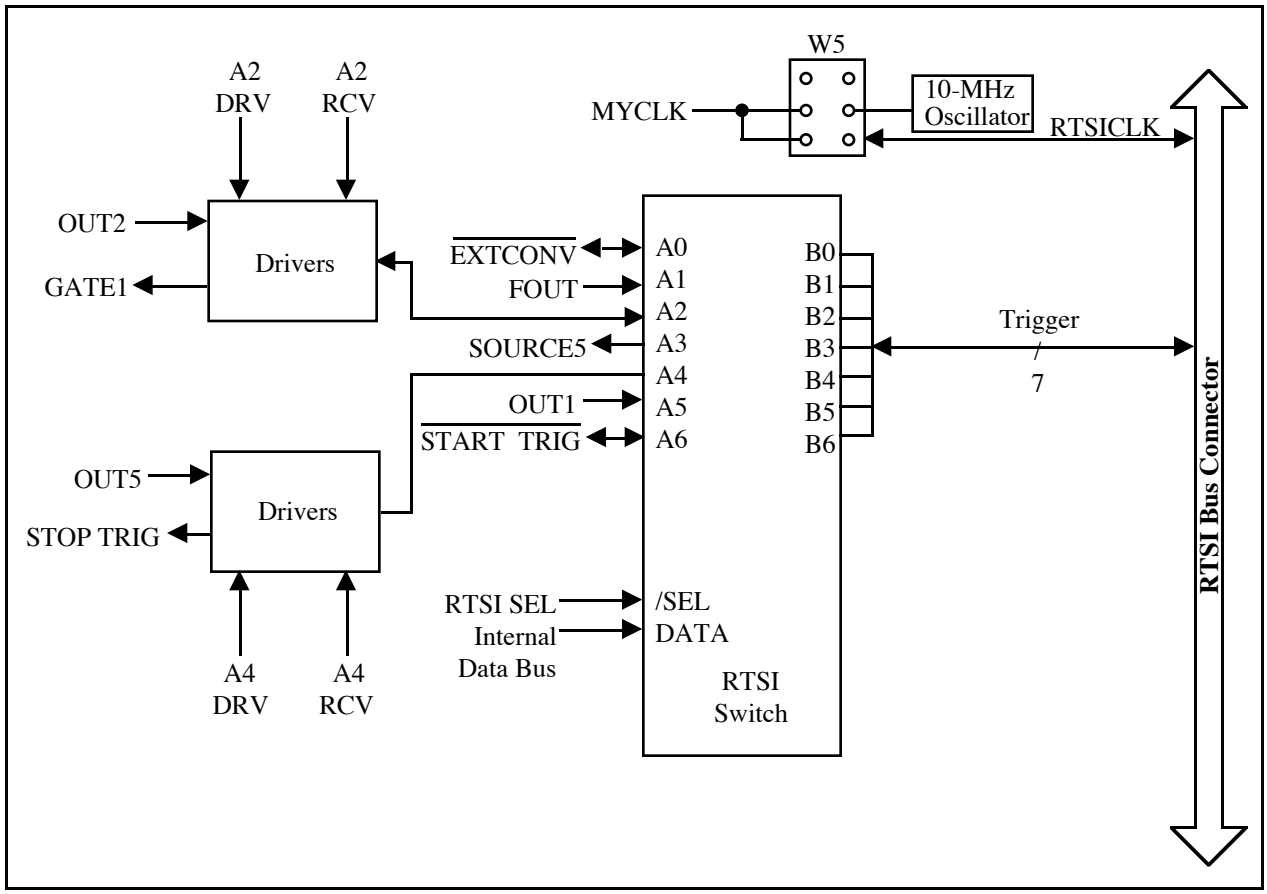


Figure 3-8. RTSI Bus Interface Circuitry Block Diagram

The RTSI CLK line can be used to source a 10-MHz signal across the RTSI bus or to receive another clock signal from another AT board connected to the RTSI bus. MYCLK is the system clock used by the AT-MIO-16D. The W5 jumpers select how these clock signals are routed.

The RTSI switch is a National Instruments custom integrated circuit that acts as a 7x7 crossbar switch. Pins B<6..0> are connected to the seven RTSI bus trigger lines. Pins A<6..0> are connected to seven signals on the board. The RTSI switch can drive any of the signals at pins A<6..0> onto any one or more of the seven RTSI bus trigger lines and can drive any of the seven trigger line signals onto any one or more of the pins A<6..0>. This capability provides a completely flexible signal interconnection scheme for any AT Series board sharing the RTSI bus. The RTSI switch is programmed via its select and data inputs.

On the AT-MIO-16D board, nine signals are connected to pins A<6..0> of the RTSI switch with the aid of additional drivers. The signals GATE1, OUT1, OUT2, OUT5, FOUT, and STOP TRIG are shared with the AT-MIO-16D I/O connector and Am9513A Counter/Timer. The signal SOURCE5 is connected to the Am9513A SOURCE5 pin. The EXTCONV* and START TRIG* signals are shared with the I/O connector and the data acquisition timing circuitry. These onboard interconnections allow AT-MIO-16D general-purpose and data acquisition timing to be controlled over the RTSI bus as well as externally and allow the AT-MIO-16D and the I/O connector to provide timing signals to other AT boards connected to the RTSI bus.

DIO-24 Functional Overview

The block diagram in Figure 3-9 illustrates the key functional components of the AT-MIO-16D DIO-24 circuitry.

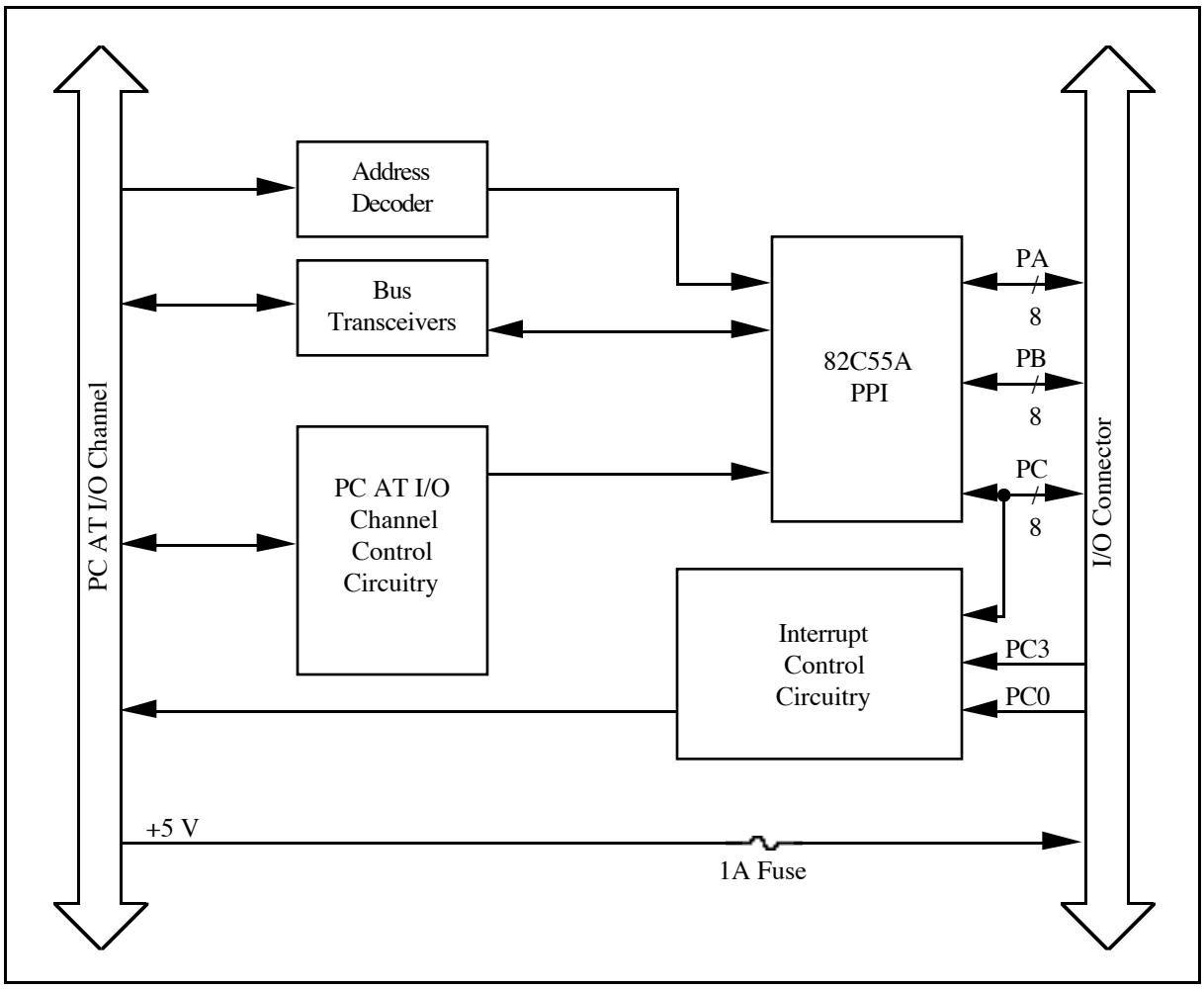


Figure 3-9. AT-MIO-16D DIO-24 Block Diagram

DIO-24 Interrupt Control Circuitry

The interrupt level used by the DIO-24 circuitry of the AT-MIO-16D is selected by the onboard jumper W13. Another onboard jumper, W14, is used to enable interrupts from the DIO-24 circuitry. The setting for W14 selects PC2, PC4, or PC6 as the active low interrupt enable signal. Selecting N/C for W14 disables interrupts from the DIO-24 circuitry. When the onboard jumpers are set to enable interrupts, the 82C55A can be programmed to generate an interrupt request by setting INTRA for Group A or INTRB for Group B. When interrupts are enabled for Group A, an active high signal on the PC3 line generates an interrupt request. When interrupts are enabled for Group B, an active high signal on the PC0 line generates an interrupt request.

DIO-24 Circuitry I/O Connector

All digital I/O is transmitted through a 100-pin male connector. This 100-pin connector is physically divided into two standard 50-pin female connectors using a cable assembly. The pin assignments for the 50-pin DIO-24 I/O connector are compatible with standard 24-channel digital I/O applications. All even pins on the 50-pin DIO-24 connector are attached to logic ground, and pin 49 is connected to +5 V through a protection fuse (F4), which is often required to operate I/O module mounting racks. See Chapter 2, *Configuration and Installation*, for additional information.

82C55A Programmable Peripheral Interface

The 82C55A PPI is the heart of the AT-MIO-16D DIO-24 circuitry. This chip has 24 programmable I/O pins that represent three 8-bit ports: PA, PB, and PC. Each port can be programmed as an input or an output port. The 82C55A has three modes of operation: simple I/O (Mode 0), strobed I/O (Mode 1), and bidirectional I/O (Mode 2). In Modes 1 and 2, the three 8-bit ports are divided into two groups: Group A and Group B (two groups of twelve signals). One 8-bit configuration (or control) word determines the mode of operation for each group. The Group A control bits configure Port A (A0 through A7) and the upper 4 bits (nibble) of Port C (C4 through C7). The Group B control bits configure Port B (B0 through B7) and the lower nibble of Port C (C0 through C3). Modes 1 and 2 use handshaking signals from Port C to synchronize data transfers. Refer to Chapter 4, *Programming*, or to Appendix F, *Oki MSM82C55A Data Sheet,* for more detailed information.

82C55A Modes of Operation

The three basic modes of operation for the 82C55A are as follows:

- Mode 0 – Basic I/O
- Mode 1 – Strobed I/O
- Mode 2 – Bidirectional bus

The 82C55A also has a single bit set/reset feature for Port C. The 8-bit control word also programs this function. For additional information, refer to Appendix F, *Oki MSM82C55A Data Sheet*.

Mode 0

This mode can be used for simple input and output operations for each of the ports. No handshaking is required; data is simply written to or read from a selected port.

Mode 0 has the following features:

- Two 8-bit ports (A and B) and two 4-bit ports (upper and lower nibble of Port C)
- Any port can be input or output
- Outputs are latched, but inputs are not latched

Mode 1

This mode transfers data that is synchronized by handshaking signals. Ports A and B use the eight lines of Port C to generate or receive the handshake signals. This mode divides the ports into two groups (Group A and Group B) and has the following features:

- Each group contains one 8-bit data port (Port A or Port B) and one 4-bit control/data port (upper or lower nibble of Port C).
- The 8-bit data ports can be either input or output, both of which are latched.
- The 4-bit ports are used for control and status of the 8-bit data ports.
- Interrupt generation and enable and/or disable functions are available.

Mode 2

This mode can be used for communication over a bidirectional 8-bit bus. Handshaking signals are used in a manner similar to Mode 1. Interrupt generation and enable and/or disable functions are also available. Other features of this mode include the following:

- Used in Group A only (Port A and upper nibble of Port C)
- One 8-bit bidirectional port (Port A) and a 5-bit control status port (Port C)
- Latched inputs and outputs

Single Bit Set/Reset Feature

Any of the eight bits of Port C can be set or reset with one control word. This feature generates status and control for Port A and Port B when operating in Mode 1 or Mode 2.

Chapter 4
Programming

This chapter discusses the programming of the AT-MIO-16D. Included in this chapter are the AT-MIO-16D register address map, a detailed register description, and a functional programming description.

Note: If you plan to use a programming software package such as NI-DAQ for DOS/Windows or LabWindows with your AT-MIO-16D board, you need not read this chapter. However, you will gain added insight into your AT-MIO-16D board by reading this chapter.

Register Map

The register map for the AT-MIO-16D is shown in Table 4-1. This table gives the register name, the register offset address, the size of the register in bits, and the type of the register (read-only, write-only, or read-and-write). The actual register address is computed by adding the individual offset address to the board base address.

Table 4-1. AT-MIO-16D Register Map

Register Name	OffSet Address (Hex)	Type	Size
Configuration and Status Register Group:			
Command Register 1	0	Write-only	16-bit
Status Register	0	Read-only	16-bit
Command Register 2	2	Write-only	16-bit
Event Strobe Register Group:			
Start Convert Register	8	Write-only	16-bit
Start DAQ Register	A	Write-only	16-bit
A/D Clear Register	C	Write-only	16-bit
External Strobe Register	E	Write-only	16-bit
Analog Output Register Group:			
DAC0 Register	10	Write-only	16-bit
DAC1 Register	12	Write-only	16-bit
INT2CLR Register	14	Write-only	16-bit

(continues)

Table 4-1. AT-MIO-16D Register Map (continued)

Register Name	OffSet Address (Hex)	Type	Size
Analog Input Register Group:			
Mux-Counter Register	4	Write-only	16-bit
Mux-Gain Register	6	Write-only	16-bit
A/D FIFO Register	16	Read-only	16-bit
DMA TC INT Clear Register	16	Write-only	16-bit
Am9513A Counter/Timer Register Group:			
Am9513A Data Register	18	Read-and-write	16-bit
Am9513A Command Register	1A	Write-only	16-bit
Am9513A Status Register	1A	Read-only	16-bit
MIO-16 Digital I/O Register Group:			
MIO-16 Digital Input Register	1C	Read-only	16-bit
MIO-16 Digital Output Register	1C	Write-only	16-bit
RTSI Switch Register Group:			
RTSI Switch Shift Register	1E	Write-only	8-bit
RTSI Switch Strobe Register	1F	Write-only	8-bit
DIO-24 Register Group:			
DIO-24 PORTA Register	0x00	Read-and-write	8-bit
DIO-24 PORTB Register	0x01	Read-and-write	8-bit
DIO-24 PORTC Register	0x02	Read-and-write	8-bit
DIO-24 CNFG Register	0x03	Write-only	8-bit

Register Sizes

The IBM PC AT and compatibles support two different transfer sizes for read-and-write operations: byte (8-bit) and word (16-bit). Table 4-1 shows the size of each AT-MIO-16D register. For example, reading the A/D FIFO Register requires a 16-bit (word) read operation at the specified address, whereas writing to the RTSI Strobe Register requires an 8-bit (byte) write operation at the specified address.

Register Description

Table 4-1 divides the AT-MIO-16D registers into eight different register groups. A bit description of each of the registers making up these groups is included later in this chapter.

The Configuration and Status Register Group controls the overall operation of the AT-MIO-16D hardware. The Event Strobe Group is a group of registers that, when written to, generate some event on the AT-MIO-16D board. The registers in the Analog Output Group access the AT-MIO-16D DACs. The Analog Input Group allows ADC output to be read. The Counter/Timer Group consists of the three registers of the onboard Am9513A Counter/Timer chip. The registers in the Digital I/O Group access the onboard digital input and output lines. The registers in the RTSI Switch Group control the onboard RTSI switch. The DIO-24 Register Group controls all operations and modes of the DIO-24 circuitry on the AT-MIO-16D board.

You may notice that the DIO-24 registers have the same offset as Command Register 1 and Command Register 2. Access to the DIO-24 registers are distinguished by means of performing an 8-bit bus transfer versus a 16-bit bus transfer.

Register Description Format

The remainder of this register description chapter discusses each of the AT-MIO-16D registers in the order shown in Table 4-1. Each register group is introduced, followed by a detailed bit description of each register. The individual register description gives the address, type, word size, and bit map of the register, followed by a description of each bit.

The register bit map shows a diagram of the register with the MSB (bit 15 for a 16-bit register, bit 7 for an 8-bit register) shown on the left, and the LSB (bit 0) shown on the right. A square is used to represent each bit. Each bit is labeled with a name inside its square. An asterisk (*) after the bit name indicates that the bit is inverted (negative logic).

In many of the registers, several bits are labeled with an X, indicating *don't care bits*. When a register is read, these bits may appear set or cleared but should be ignored because they have no significance. When a register is written to, setting or clearing these bit locations has no effect on the AT-MIO-16D hardware.

The bit map field for some write-only registers states *not applicable, no bits used*. Writing to these registers generates a strobe in the AT-MIO-16D. These strobes are used to cause some onboard event to occur. For example, they can be used to clear the analog input circuitry or to start a data acquisition operation. The data is ignored when writing to these registers; therefore, any bit pattern will suffice.

Configuration and Status Register Group

The three registers making up the Configuration and Status Register Group allow general control and monitoring of the AT-MIO-16D hardware. Command Registers 1 and 2 contain bits that control operation of several different pieces of the AT-MIO-16D hardware. The Status Register can be used to read the state of different pieces of the AT-MIO-16D hardware.

Bit descriptions of the three registers making up the Configuration and Status Group are given on the following pages.

Command Register 1

Command Register 1 contains ten bits that control AT-MIO-16D interrupts, direct memory access (DMA), and some analog input and output modes.

Address: Base address + 0 (hex)

Type: Write-only

Word Size: 16-bit

Bit Map:

15	14	13	12	11	10	9	8
X	X	X	X	X	X	DAQSTOPINTEN	TCINTEN

7	6	5	4	3	2	1	0
CONVINTEN	DBDMA	DMAEN	DAQEN	SCANEN	SCANDIV	16*/32CNT	2SCADC*

Bit	Name	Description
15-10	X	Don't care bits.
9	DAQSTOPINTEN	This bit enables and disables the generation of an interrupt when a data acquisition operation is terminated. This termination can be caused by either the normal completion of a data acquisition operation or by an error condition. If an error condition occurs, either OVERFLOW or OVERRUN is set in the Status Register. The interrupt is serviced by writing to the A/D Clear Register. If DAQSTOPINTEN is cleared, no data acquisition termination interrupts are generated.
8	TCINTEN	This bit enables and disables generation of an interrupt when a DMA terminal count pulse is received from the DMA controller in the PC AT. If TCINTEN is set, an interrupt request is generated when the DMA controller transfer count register decrements from 0 to FFFF (hex). The interrupt request is serviced by writing to the DMA TC INT Clear Register. When TCINTEN is cleared, no DMA terminal count interrupts are generated.
7	CONVINTEN	This bit enables and disables the generation of an interrupt when A/D conversion results are available. If CONVINTEN is set, an interrupt is generated whenever an A/D conversion is available to be read from the A/D FIFO. If CONVINTEN is cleared, no interrupt is generated.
6	DBDMA	This bit selects the DMA mode. If DBDMA is cleared and DMAEN is set, a single-channel, single-buffered DMA mode is selected. If DBDMA is set and DMAEN is set, a double-channel, double-buffered DMA mode is selected.

Bit	Name	Description (continued)
5	DMAEN	This bit enables and disables the generation of DMA requests. If DMAEN is set, a DMA request is generated whenever an A/D conversion result is available to be read from the A/D FIFO. If DMAEN is cleared, no DMA request is generated.
4	DAQEN	This bit enables and disables a data acquisition operation that is controlled by the onboard sample-interval and sample counters. If DAQEN is set, a software or start trigger starts the counters (assuming that the counters are programmed and enabled), thereby initiating a data acquisition operation. If DAQEN is cleared, software and start triggers are ignored.
3	SCANEN	This bit enables and disables multiple-channel scanning during data acquisition. If SCANEN is set, alternate analog input channels are sampled during data acquisition under control of the mux-gain memory. If SCANEN is cleared, a single analog input channel is sampled during the entire data acquisition operation.
2	SCANDIV	This bit enables and disables division of the mux-counter clock during data acquisition. The mux-counter clock controls sequencing of the mux-gain memory. If SCANDIV is set, the mux-counter clock is controlled by Counter 1 of the Am9513A Counter/Timer. If SCANDIV is cleared, the mux-counter clock generates one pulse per conversion.
1	16*/32CNT	This bit selects the count resolution for the number of A/D conversions to be performed in a data acquisition operation. If 16*/32CNT is cleared, a 16-bit count mode is selected and Counter 4 of the Am9513A Counter/Timer controls conversion counting. If 16*/32CNT is set, a 32-bit count mode is selected and Counter 4 is concatenated with Counter 5 to control conversion counting. A 16-bit count mode can be used if the number of A/D sample conversions to be performed is less than 65,537. A 32-bit count mode should be used if the number of A/D sample conversions to be performed is greater than or equal to 65,537.
0	2SCADC*	This bit selects the binary format for the 16-bit data word read from the A/D FIFO. If 2SCADC* is set, a straight binary format is used and the data read from the A/D FIFO ranges from 0 to +4,095 decimal (0 to 0FFF hex). This mode is useful if a unipolar input range is used. If 2SCADC* is cleared, a 16-bit two's complement mode is used and the data read from the ADC ranges from -2,048 to +2,047 decimal (F800 to 07FF hex). This mode is useful if a bipolar input range is used.

Status Register

The Status Register contains 16 bits of AT-MIO-16D hardware status information, including interrupt and analog input status.

Address: Base address + 0 (hex)

Type: Read-only

Word Size: 16-bit

Bit Map:

15	14	13	12	11	10	9	8
GINT	DAQSTOPINT	CONVAVAIL	OUT2INT	DAQPROG	DMATCINT	OVERFLOW	OVERRUN

7	6	5	4	3	2	1	0
GAIN1	GAIN0	DMACH	MUX1EN	MUX0EN	MA2	MA1	MA0

Bit	Name	Description
15	GINT	This bit reflects the overall state of interrupts generated by the MIO-16 circuitry on the AT-MIO-16D board. If GINT is set, the AT-MIO-16D is asserting an interrupt request on the MIO-16 interrupt that has not yet been serviced. If GINT is cleared, no MIO-16 interrupt is pending. This bit is normally cleared.
14	DAQSTOPINT	This bit reflects the status of the data acquisition termination interrupt. If DAQSTOPINT is set and either OVERFLOW or OVERRUN is set, the current interrupt is due to an error condition. If DAQSTOPINT is set and neither OVERFLOW nor OVERRUN is set, the current interrupt is due to the completion of the data acquisition operation. DAQSTOPINT is cleared by writing to the A/D Clear Register.
13	CONVAVAIL	This bit reflects the state of the A/D FIFO. If CONVAVAIL is set, one or more A/D conversion results are available to be read from the A/D FIFO. If conversion interrupts are enabled (CONVINTEN is set) and CONVAVAIL is set, the current interrupt indicates that A/D conversion data is available in the A/D FIFO. If CONVAVAIL is cleared, the A/D FIFO is empty and no conversion interrupt request is asserted.
12	OUT2INT	This bit reflects the status of the OUT2INT interrupt. OUT2INT is cleared by writing to the INT2CLR Register. OUT2INT is set whenever a rising edge on OUT2 is detected; this condition generates an interrupt request only if the INT2EN bit in Command Register 2 is set.

Bit	Name	Description (continued)
11	DAQPROG	This bit indicates whether a data acquisition operation is in progress. If DAQPROG is set, a data acquisition operation is in progress. If DAQPROG is cleared, a data acquisition operation is not in progress.
10	DMATCINT	This bit reflects the status of the DMA terminal count interrupt. If DMATCINT is set, and if TCINTEN is set in Command Register 1, then the current interrupt is due to the detection of a DMA terminal count pulse. DMATCINT is cleared by writing to the DMA TC Clear Register.
9	OVERFLOW	This bit indicates whether the A/D FIFO has overflowed during a sample run. OVERFLOW is an error condition that occurs if the FIFO fills up with A/D conversion data and A/D conversions continue. If OVERFLOW is set, A/D conversion data has been lost because of FIFO overflow. If OVERFLOW is cleared, no overflow has occurred. If OVERFLOW occurs during a data acquisition operation, the data acquisition is terminated immediately. This bit can be reset by writing to the A/D Clear Register.
8	OVERRUN	This bit indicates whether an A/D conversion was initiated before the previous A/D conversion was complete. OVERRUN is an error condition that may occur if the data acquisition sample interval is too small (sample rate is too high). If OVERRUN is set, one or more conversions were skipped. If OVERRUN is cleared, no overrun condition has occurred. If OVERRUN occurs during a data acquisition operation, the data acquisition is terminated immediately. This bit can be reset by writing to the A/D Clear Register.
7-6	GAIN<1..0>	These two bits show the current gain setting for the programmable gain amplifier (see *Mux-Gain Register* later in this chapter).
5	DMACH	This bit indicates the current DMA channel. If DBDMA in Command Register 1 is set, dual DMA mode is selected. In this mode, DMA transfers switch between two DMA channels. DMACH indicates which DMA channel is currently in use for DMA operation. If DMACH is cleared, then DMA 1 is in use. If DMACH is set, then DMA 2 is in use. In single DMA mode, only DMA 1 is used.
4	MUX1EN	This bit indicates the state of multiplexer 1. Multiplexer 1 controls analog input channels 8 through 15. If this bit is set, multiplexer 1 is currently enabled. If this bit is cleared, multiplexer 1 is currently disabled. In single-ended mode, multiplexer 1 is enabled only when one of the input channels 8 through 15 is selected. In this mode, the output of multiplexer 1 is connected to the positive (+) input of the instrumentation amplifier. In DIFF mode, multiplexer 1 is always enabled. In this mode, the output of multiplexer 1 is connected to the negative (-) input of the instrumentation amplifier.

Bit	Name	Description (continued)
3	MUX0EN	This bit indicates the state of multiplexer 0. Multiplexer 0 controls analog input channels 0 through 7. If this bit is set, multiplexer 0 is currently enabled. If this bit is cleared, multiplexer 0 is currently disabled. In single-ended mode, multiplexer 0 is enabled only when one of the input channels 0 through 7 is selected. In DIFF mode, multiplexer 0 is always enabled. The output of multiplexer 0 is always connected to the positive (+) input of the instrumentation amplifier.
2-0	MA<2..0>	MA<2..0> give the low-order three bits of the analog input channel address. MA stands for multiplexer address. These three bits, in conjunction with the MUX1EN and MUX0EN bits, indicate which analog input channel is currently selected. In single-ended mode, the analog input channel selected is determined by the value of MA<2..0> if MUX0EN is set and by the value of MA<2..0> + 8 if MUX1EN is set. In DIFF mode, two analog input channels are selected simultaneously. The two channels are MA<2..0> and MA<2..0> + 8.

Command Register 2

Command Register 2 contains 10 bits that control AT-MIO-16D interrupts, digital output drivers, and scan modes used by the data acquisition circuitry.

Address: Base address + 2 (hex)

Type: Write-only

Word Size: 16-bit

Bit Map:

15	14	13	12	11	10	9	8
X	X	X	X	X	X	DOUTEN1	DOUTEN0

7	6	5	4	3	2	1	0
INTEN	INT2EN	LDAC	SCN2	A4RCV	A4DRV	A2RCV	A2DRV

Bit	Name	Description
15-10	X	Don't care bits.
9	DOUTEN1	This bit enables and disables driving of the 4-bit MIO-16 digital output port 1 by the MIO-16 Digital Output Register. If DOUTEN1 is set, the MIO-16 Digital Output Register drives the digital lines. If DOUTEN1 is cleared, the MIO-16 Digital Output Register drivers are set to a high-impedance state, thereby allowing an external device to drive the digital lines.
8	DOUTEN0	This bit enables and disables driving of the 4-bit MIO-16 digital output port 0 by the MIO-16 Digital Output Register. If DOUTEN0 is set, the MIO-16 Digital Output Register drives the digital lines. If DOUTEN0 is cleared, the MIO-16 Digital Output Register drivers are set to a high-impedance state, thereby allowing an external device to drive the digital lines.
7	INTEN	This bit enables and disables any interrupt generated from the MIO-16 circuitry of the AT-MIO-16D board. If this bit is cleared, all MIO-16 interrupts are disabled from the AT-MIO-16D board. To generate a specific interrupt, INTEN and a specific interrupt enable bit must be set.
6	INT2EN	This bit enables and disables the generation of an interrupt on the OUT2 signal of the Am9513A Counter/Timer. If INT2EN is set, interrupt requests are generated when a rising edge on OUT2 is detected. The interrupt is cleared by writing to the INT2CLR Register. When INT2EN is cleared, OUT2 interrupts are not generated. This interrupt is useful for waveform generation or interrupt generation on an external signal connected to OUT2.

Bit	Name	Description (continued)
5	LDAC	This bit selects the update method for the DAC outputs. When LDAC is cleared, both DAC0 and DAC1 are updated when either DAC is written to. If LDAC is set, both DACs are updated when an active low pulse is detected on the OUT2 signal of the Am9513A Counter/Timer.
4	SCN2	This bit selects the data acquisition scanning mode used when scanning multiple A/D channels. If SCN2 is cleared, continuous channel scanning is used. In this mode, scan sequences are repeated with no delays between cycles. If SCN2 is set, interval channel scanning is used. In this mode, scan sequences occur during a programmed time interval, called a *scan interval*. One cycle of the scan sequence occurs during each scan interval.
3	A4 RCV	This bit controls a driver that allows the STOP TRIG signal to be driven from pin A4 of the RTSI switch. This bit allows a signal to be received from one of the RTSI bus trigger lines and driven onto the STOP TRIG line. If A4RCV is set, pin A4 of the RTSI switch drives the STOP TRIG signal. If A4RCV is cleared, the STOP TRIG signal is not driven by the RTSI switch.
2	A4 DRV	This bit controls a driver that allows the OUT5 signal to drive pin A4 of the RTSI switch. This driver allows the OUT5 signal to be driven onto one of the RTSI bus trigger lines. If A4DRV is set, pin A4 of the RTSI switch is driven by OUT5. If A4DRV is cleared, pin A4 is not driven.
1	A2 RCV	This bit controls a driver that allows the GATE1 signal to be driven from pin A2 of the RTSI switch. This driver allows a signal to be received from one of the RTSI bus trigger lines and driven onto the GATE1 line. If A2RCV is set, pin A2 of the RTSI switch drives the GATE1 signal. If A2RCV is cleared, the GATE1 signal is not driven by the RTSI switch.
0	A2 DRV	This bit controls a driver that allows the OUT2 signal to drive pin A2 of the RTSI switch. This driver allows the OUT2 signal to be driven onto one of the RTSI bus trigger lines. If A2DRV is set, pin A2 of the RTSI switch is driven by OUT2. If A2DRV is cleared, pin A2 is not driven.

The Event Strobe Register Group

The Event Strobe Register Group consists of four registers that, when written to, cause the occurrence of certain events on the AT-MIO-16D board, such as clearing flags and starting A/D conversions.

Descriptions of the four registers making up the Event Strobe Register Group are given on the following pages.

Start Convert Register

Writing to the Start Convert Register location initiates an A/D conversion.

Address: Base address + 8 (hex)

Type: Write-only

Word Size: 16-bit

Bit Map: Not applicable, no bits used

Note: A/D conversions can be initiated in one of two ways: by writing to the Start Convert Register or by detecting an active low signal on the EXTCONV* signal. The EXTCONV* signal is connected to pin 40 on the MIO-16 I/O connector, to OUT3 of the Am9513A, and to the A0 pin of the RTSI bus switch. If EXTCONV* is driven low by any one of these sources, it prevents the Start Convert Register from initiating an A/D conversion. If the Start Convert Register is to initiate A/D conversions, the OUT3 signal should be initialized to a high-impedance state, any signal connected to pin 40 of the I/O connector should be in a high-impedance or high state, and the A0 pin of the RTSI bus switch should be configured as an input pin.

Start DAQ Register

Writing to the Start DAQ Register location initiates a multiple A/D conversion data acquisition operation.

Note: Several other pieces of AT-MIO-16D circuitry must be set up before a data acquisition run can occur. See the *Programming Multiple A/D Conversions on a Single Input Channel* section later in this chapter.

Address: Base address + A (hex)

Type: Write-only

Word Size: 16-bit

Bit Map: Not applicable, no bits used

Note: Multiple A/D conversion data acquisition operations can be initiated in one of two ways: by writing to the Start DAQ Register or by detecting an active low signal on the START TRIG* signal. The START TRIG* signal is connected to pin 38 on the MIO-16 I/O connector and to the A6 pin of the RTSI bus switch. If START TRIG* is driven low by either of these sources, it prevents the Start DAQ Register from initiating a multiple A/D conversion data acquisition operation. If the Start DAQ Register is to initiate multiple A/D conversions, any signal connected to pin 38 of the I/O connector should be in a high-impedance or high state and the A6 pin of the RTSI bus switch should not be driven low.

A/D Clear Register

Writing to the A/D Clear Register location clears the data acquisition circuitry. The following specific events occur:

- Any data acquisition operation in progress is canceled.
- The A/D FIFO is emptied.
- The overrun flag is cleared.
- The overflow flag is cleared.
- Any pending CONV interrupt is cleared.
- Any pending DAQSTOP interrupt is cleared.
- Any pending DMATCINT interrupt is cleared.
- Any pending DMA request is cleared.

Address: Base address + C (hex)

Type: Write-only

Word Size: 16-bit

Bit Map: Not applicable, no bits used

External Strobe Register

Writing to the External Strobe Register location generates an active-low, approximately 200-nsec strobe pulse at the EXTSTROBE output at the MIO-16 I/O connector. This pulse may be useful for several applications, including generating external general-purpose triggers and latching data into external devices (from the digital output port, for example).

Address: Base address + E (hex)

Type: Write-only

Word Size: 16-bit

Bit Map: Not applicable, no bits used

Analog Output Register Group

Two of the three registers making up the Analog Output Register Group load the two analog output channels. DAC0 controls analog output channel 0. DAC1 controls analog output channel 1. These DACs are written to individually, and the analog output can be updated immediately or each time an active low pulse is detected on the OUT2 bit of the Am9513A Counter/Timer. The update method is selected with the LDAC bit in Command Register 2.

The third register in the Analog Output Register Group is the INT2CLR Register. The AT-MIO-16D can be programmed to interrupt when it detects a rising edge signal on the OUT2 pin of the Am9513A Counter/Timer. This interrupt can be cleared by writing to the INT2CLR Register.

Descriptions of the three registers making up the Analog Output Register Group are given on the following pages.

DAC0 Register

Writing to DAC0 loads the corresponding analog output channel DAC. The voltages generated by the analog output channels are updated either immediately or when an active low pulse occurs on OUT2. The update method is selected by the LDAC bit in Command Register 2.

Address: Base address + 10 (hex)

Type: Write-only

Word Size: 16-bit

Bit Map:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
X	X	X	X	D11	D10	D9	D6	D7	D6	D5	D4	D3	D2	D1	D0
				MSB											LSB

Bit	Name	Description
15-12	X	Don't care bits.
11-0	D<11..0>	These 12 bits are loaded into the DAC and update the voltage generated by the analog output channel in one of two ways, immediately or upon an OUT2 pulse. See the *Programming the Analog Output Circuitry* section later in this chapter for Table 4-5 and 4-6, both of which map digital values to output voltage.

DAC1 Register

Writing to DAC1 loads the corresponding analog output channel DAC. The voltages generated by the analog output channels are updated either immediately or when an active low pulse occurs on OUT2. The update method is selected by the LDAC bit in Command Register 2.

Address: Base address + 12 (hex)

Type: Write-only

Word Size: 16-bit

Bit Map:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
X	X	X	X	D11	D10	D9	D6	D7	D6	D5	D4	D3	D2	D1	D0
				MSB											LSB

Bit	Name	Description
15-12	X	Don't care bits.
11-0	D<11..0>	These twelve bits are loaded into the DAC and update the voltage generated by the analog output channel in one of two ways, immediately or upon an OUT2 pulse. See the *Programming the Analog Output Circuitry* section later in this chapter for Tables 4-5 and 4-6, both of which map digital values to output voltage.

INT2CLR Register

Writing to INT2CLR clears the interrupt request asserted when an OUT2 pulse is detected.

Address: Base address + 14 (hex)

Type: Write-only

Word Size: 16-bit

Bit Map: Not applicable, no bits used

Analog Input Register Group

The four registers making up the Analog Input Register Group control the analog input circuitry and allow the A/D FIFO to be read from. The Mux-Counter Register generates addresses for the mux-gain memory. The Mux-Gain Register controls the current multiplexer and gain settings and allows the mux-gain memory to be written to. Reading the A/D FIFO Register returns stored A/D conversion results. Writing to the DMA TC INT Clear Register clears the interrupt request generated by a DMA terminal count pulse.

Bit descriptions for the registers making up the Analog Input Register Group are given on the following pages.

Mux-Counter Register

The Mux-Counter Register loads the counter that sequences through the mux-gain memory.

Address: Base address + 4 (hex)

Type: Write-only

Word Size: 16-bit

Bit Map:

15	14	13	12	11	10	9	8
X	X	X	X	X	X	X	X

7	6	5	4	3	2	1	0
X	X	X	X	MC3	MC2	MC1	MC0

Bit	Name	Description
15-4	X	Don't care bits.
3-0	MC<3..0>	These four bits are loaded into the multiplexer counter by writing to the Mux-Counter Register. The multiplexer counter generates addresses for the mux-gain memory; therefore, writing to the Mux-Counter Register allows a specific location in the mux-gain memory to be addressed. The mux-gain memory contains a sequence of multiplexer addresses and gain settings. For example, writing 0004 hex to the Mux-Counter Register loads the multiplexer counter with the value 4 and addresses mux-gain memory location 4. The analog circuitry is then controlled by the multiplexer address and gain settings in mux-gain memory location 4 (see the Mux-Gain Register description later in this chapter).

Mux-Gain Register

The Mux-Gain Register controls the multiplexer and gain settings, and, when used in conjunction with the Mux-Counter Register, allows a scan sequence to load into the mux-gain memory.

Address: Base address + 6 (hex)

Type: Write-only

Word Size: 16-bit

Bit Map:

15	14	13	12	11	10	9	8
X	X	X	X	X	X	X	X

7	6	5	4	3	2	1	0
GAIN1	GAIN0	X	LASTONE	MA3	MA2	MA1	MA0

Bit **Name** **Description**

15-8 X Don't care bits.

7-6 GAIN<1..0> This 2-bit field controls the gain setting of the input instrumentation amplifier. The actual amplifier gains depend on the type of
AT-MIO-16D board. The following gains can be selected on the AT-MIO-16DH board:

GAIN<1..0>	Actual Gain
00	1
01	2
10	4
11	8

The following gains can be selected on the AT-MIO-16DL board:

GAIN<1..0>	Actual Gain
00	1
01	10
10	100
11	500

5 X Don't care bit.

Bit	Name	Description (continued)
4	LASTONE	This bit should be set in the last entry of the scan sequence loaded into the mux-gain memory.
3-0	MA<3..0>	This 4-bit field controls the multiplexer address setting of the input multiplexers, thereby allowing the analog input channel to be selected. In single-ended mode (NRSE or RSE), only one analog input channel is selected. In DIFF mode, two analog input channels are selected. A table showing the analog input channel selected for either mode is given below.

MA<3..0>	Selected Analog Input Channels	
	Single-Ended	DIFF
		(+) (-)
0000	0	0 & 8
0001	1	1 & 9
0010	2	2 & 10
0011	3	3 & 11
0100	4	4 & 12
0101	5	5 & 13
0110	6	6 & 14
0111	7	7 & 15
1000	8	0 & 8
1001	9	1 & 9
1010	10	2 & 10
1011	11	3 & 11
1100	12	4 & 12
1101	13	5 & 13
1110	14	6 & 14
1111	15	7 & 15

Writing to the Mux-Gain Register updates the current analog input channel selection and the current gain setting. The mux-gain memory is also loaded by writing to the Mux-Gain Register. The multiplexer counter is written to in order to address a specific location in the mux-gain memory. Any subsequent value written to the Mux-Gain Register is then stored in that memory location as well as applied to the analog input multiplexer and gain circuitry.

A/D FIFO Register

Reading the A/D FIFO Register returns the oldest A/D conversion value stored in the A/D FIFO. Whenever the A/D FIFO is read, the value read is removed from the A/D FIFO, thereby leaving space for another A/D conversion value to be stored. Values are stored into the A/D FIFO by the ADC whenever an A/D conversion is complete.

The A/D FIFO is emptied when all values it contains are read. The Status Register should be read before the A/D FIFO Register is read. If the A/D FIFO contains one or more A/D conversion values, the CONVAVAIL bit is set in the Status Register and the A/D FIFO Register can be read to retrieve a value. If the CONVAVAIL bit is cleared, the A/D FIFO is empty, in which case reading the A/D FIFO Register returns meaningless information.

The values returned by reading the A/D FIFO Register are available in two different binary formats: straight binary, which generates only positive numbers, or two's complement binary, which generates both positive and negative numbers. The binary format used is selected by the 2SCADC* bit in Command Register 1. The bit pattern returned for either format is given below.

Address: Base address + 16 (hex)

Type: Read-only

Word Size: 16-bit

Bit Map: Straight binary mode

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
MSB															LSB

Bit	Name	Description
15-0	D<15..0>	These bits are the straight binary result of a 12-bit A/D conversion. The most significant four bits are set to 0 in order to return a 16-bit result. Values read, therefore, range from 0 to 4,095 decimal (0000 to 0FFF hex). Straight binary mode is useful for unipolar analog input readings because all values that are read reflect a positive polarity input signal.

Bit Map: Two's complement binary mode

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
D11*	D11*	D11*	D11*	D11*	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
MSB															LSB

Bit	Name	Description
15-0	D<15..0>	These bits are the two's complement result of a 12-bit A/D conversion. Bit D11 is inverted and extended out to bits D12 through D15. Values read, therefore, range from -2,048 to +2,047 decimal (F800 to 7FF hex). Two's complement mode is useful for bipolar analog input readings because the values read reflect the polarity of the input signal.

DMA TC INT Clear Register

Writing to the DMA TC INT Clear Register clears the interrupt request asserted when a DMA terminal count pulse is detected.

Address: Base address + 16 (hex)

Type: Write-only

Word Size: 16-bit

Bit Map: Not applicable, no bits used

Am9513A Counter/Timer Register Group

The three registers making up the Am9513A Counter/Timer Register Group access the onboard Am9513A Counter/Timer. The Am9513A controls onboard data acquisition timing as well as general-purpose timing for the user.

The Am9513A registers described here are the Am9513A Data Register, the Am9513A Command Register, and the Am9513A Status Register. The Am9513A contains 18 additional internal registers. These internal registers are accessed through the Am9513A Data Register. A detailed register description of all Am9513A registers is included in Appendix E, *Am9513A Data Sheet*.

Bit descriptions for the Am9513A Counter/Timer Register Group registers are given in the following pages.

Am9513A Data Register

The Am9513A Data Register allows any of the 18 internal registers of the Am9513A to be written to or read from. The Am9513A Command Register must be written to in order to select the register to be accessed by the Am9513A Data Register. The internal registers accessed by the Am9513A Data Register are as follows:

- Counter Mode Registers for Counters 1, 2, 3, 4, and 5
- Counter Load Registers for Counters 1, 2, 3, 4, and 5
- Counter Hold Registers for Counters 1, 2, 3, 4, and 5
- The Master Mode Register
- The Compare Registers for Counters 1 and 2

All these registers are 16-bit registers. Bit descriptions for each of these registers are included in Appendix E, *Am9513A Data Sheet*.

Address: Base address + 18 (hex)

Type: Read-and-write

Word Size: 16-bit

Bit Map:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0

Bit	Name	Description
15-0	D<15..0>	These 16 bits are loaded into the Am9513A Internal Register currently selected. See Appendix E, *Am9513A Data Sheet*, for the detailed bit descriptions of the 18 registers accessed through the Am9513A Data Register.

Am9513A Command Register

The Am9513A Command Register controls the overall operation of the Am9513A Counter/Timer and controls selection of the internal registers accessed through the Am9513A Data Register.

Address: Base address + 1A (hex)

Type: Write-only

Word Size: 16-bit

Bit Map:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	1	1	1	1	1	1	C7	C6	C5	C4	C3	C2	C1	C0

Bit	Name	Description
15-8	1	These bits must always be set when writing to the Am9513A Command Register.
7-0	C<7..0>	These eight bits are loaded into the Am9513A Command Register. See Appendix E, *Am9513A Data Sheet*, for the detailed bit description of the Am9513A Command Register.

Am9513A Status Register

The Am9513A Status Register provides information about the output pin status of each counter in the Am9513A.

Address: Base address + 1A (hex)

Type: Read-only

Word Size: 16-bit

Bit Map:

15	14	13	12	11	10	9	8
X	X	X	X	X	X	X	X

7	6	5	4	3	2	1	0
X	X	OUT5	OUT4	OUT3	OUT2	OUT1	BYTEPTR

Bit	Name	Description
15-6	X	Don't care bits.
5-1	OUT<5..1>	Each of these five bits returns the logic state of the associated counter output pin. For example, if OUT4 is set, then the output pin of Counter 4 is at a logic high state.
0	BYTEPTR	This bit represents the state of the Am9513A Byte Pointer Flip-Flop. This bit has no significance for AT-MIO-16D operation because the Am9513A should always be used in 16-bit mode on the AT-MIO-16D.

MIO-16 Digital I/O Register Group

The two registers making up the MIO-16 Digital I/O Register Group monitor and control the AT-MIO-16D digital I/O lines. The MIO-16 Digital Input Register returns the digital state of the eight digital I/O lines. A pattern written to the MIO-16 Digital Output Register is driven onto the digital I/O lines when the digital output drivers are enabled (see the description for Command Register 2).

Bit descriptions for the registers making up the MIO-16 Digital I/O Register Group are given on the following pages.

MIO-16 Digital Input Register

The MIO-16 Digital Input Register, when read, returns the logic state of the eight AT-MIO-16D digital I/O lines at the MIO-16 I/O connector.

Address: Base address + 1C (hex)

Type: Read-only

Word Size: 16-bit

Bit Map:

15	14	13	12	11	10	9	8
X	X	X	X	X	X	X	X

7	6	5	4	3	2	1	0
BDIO3	BDIO2	BDIO1	BDIO0	ADIO3	ADIO2	ADIO1	ADIO0

Bit	Name	Description
15-8	X	Don't care bits.
7-4	BDIO<3..0>	These four bits represent the logic state of the digital lines BDIO<3..0> from the MIO-16 I/O connector.
3-0	ADIO<3..0>	These four bits represent the logic state of the digital lines ADIO<3..0> from the MIO-16 I/O connector.

MIO-16 Digital Output Register

Writing to the MIO-16 Digital Output Register controls the eight AT-MIO-16D digital I/O lines. The MIO-16 Digital Output Register controls both ports A and B. When either digital port is enabled, the pattern contained in the MIO-16 Digital Output Register is driven onto the lines of the digital port at the MIO-16 I/O connector.

Address: Base address + 1C (hex)

Type: Write-only

Word Size: 16-bit

Bit Map:

15	14	13	12	11	10	9	8
X	X	X	X	X	X	X	X

7	6	5	4	3	2	1	0
BDIO3	BDIO2	BDIO1	BDIO0	ADIO3	ADIO2	ADIO1	ADIO0

Bit	Name	Description
15-8	X	Don't care bits.
7-4	BDIO<3..0>	These four bits control the digital lines BDIO<3..0> at the MIO-16 I/O connector. The bit DOUT1EN in Command Register 2 must be set for BDO<3..0> to be driven onto the digital lines BDIO<3..0>.
3 - 0	ADIO<3..0>	These four bits control the digital lines ADIO<3..0> at the MIO-16 I/O connector. The bit DOUT0EN in Command Register 2 must be set for ADIO<3..0> to be driven onto the digital lines ADIO<3..0>.

The RTSI Switch Register Group

The two registers making up the RTSI Switch Register Group allow the AT-MIO-16D RTSI switch to be programmed for routing of signals on the RTSI bus trigger lines to and from several AT-MIO-16D signal lines. The RTSI switch is programmed by shifting a 56-bit routing pattern into the RTSI switch and then loading the internal RTSI Switch Control Register. The routing pattern is shifted into the RTSI switch by writing one bit at a time to the RTSI Switch Shift Register. The RTSI Switch Control Register is then loaded by writing to the RTSI Switch Strobe Register.

Bit descriptions for the registers making up the RTSI Switch Register Group are given on the following pages.

RTSI Switch Shift Register

The RTSI Switch Shift Register is written to in order to load the RTSI switch internal 56-bit Control Register with routing information for switching signals to and from the RTSI bus trigger lines. The RTSI Switch Shift Register is a 1-bit register and must be written to 56 times to shift the 56 bits into the internal register.

Address: Base address + 1E (hex)

Type: Write-only

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
X	X	X	X	X	X	X	RSI

Bit	Name	Description
7-1	X	Don't care bits.
0	RSI	The name of this bit stands for RTSI switch serial input. This bit is the serial input to the RTSI switch. Each time the RTSI Switch Shift Register is written to, the value of this bit is shifted into the RTSI switch. See the *Programming the RTSI Switch* section later in this chapter for more information.

RTSI Switch Strobe Register

The RTSI Switch Strobe Register is written to in order to load the contents of the RTSI Switch Shift Register into the RTSI Switch Control Register, thereby updating the RTSI switch routing pattern. The RTSI Switch Strobe Register is written to after shifting the 56-bit routing pattern into the RTSI Switch Shift Register.

Address: Base address + 1F (hex)

Type: Write-only

Word Size: 8-bit

Bit Map: Not applicable, no bits used

DIO-24 Register Group

The DIO-24 circuitry uses an 82C55A integrated circuit. The 82C55A is a general-purpose PPI containing 24 programmable I/O pins. These pins represent the three 8-bit I/O ports (A, B, and C) of the 82C55A. These ports can be programmed as two groups of 12 signals or as three individual 8-bit ports.

The DIO-24 Register Group contains the following four registers: DIO-24 PORTA Register, DIO-24 PORTB Register, DIO-24 PORTC Register, and DIO-24 CNFG Register. Bit descriptions for the registers in the DIO-24 Register Group are given on the following pages.

DIO-24 PORTA Register

Reading the DIO-24 PORTA Register returns the logic state of the eight digital I/O lines constituting Port A of the DIO-24 circuitry, that is, PA<7..0>. If Port A is configured for output, the DIO-24 PORTA Register can be written to in order to control the eight digital I/O lines constituting Port A. See *DIO-24 Circuitry Programming Considerations* later in this chapter for information on how to configure Port A for input or output.

Address: Base address + 0x00 (hex)

Type: Read-and-write

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

Bit	Name	Description
7-0	D<7..0>	These eight bits are written to or read from Port A.

DIO-24 PORTB Register

Reading the DIO-24 PORTB Register returns the logic state of the eight digital I/O lines constituting Port B of the DIO-24 circuitry, that is, PB<7..0>. If Port B is configured for output, the DIO-24 PORTB Register can be written to in order to control the eight digital I/O lines constituting Port B. See *82C55A Programming Considerations* later in this chapter for information on how to configure Port B for input or output.

Address: Base address + 0x01 (hex)

Type: Read-and-write

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

Bit	Name	Description
7-0	D<7..0>	These eight bits are written to or read from Port B.

DIO-24 PORTC Register

Port C of the DIO-24 circuitry is special in the sense that it can be used as an 8-bit I/O port like Port A and Port B if neither Port A nor Port B is used in handshaking (latched) mode. If either Port A or Port B is configured for latched I/O, some of the bits in Port C are used for handshaking signals. See *DIO-24 Circuitry Programming Considerations* later in this chapter for a description of the individual bits in the DIO-24 PORTC Register.

Address: Base address + 0x02 (hex)

Type: Read-and-write

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
D7	D6	D5	D4	D3	D2	D1	D0

Bit	Name	Description
7-0	D<7..0>	These eight bits are written to or read from Port C.

DIO-24 CNFG Register

The DIO-24 CNFG Register can be used to configure Port A, Port B, and Port C of the DIO-24 circuitry as inputs or outputs as well as selecting simple mode (basic I/O) or handshaking mode (strobed I/O) for transfers. See *DIO-24 Circuitry Programming Considerations* later in this chapter for a description of the individual bits in the DIO-24 CNFG Register.

Address: Base address + 0x03 (hex)

Type: Write-only

Word Size: 8-bit

Bit Map:

7	6	5	4	3	2	1	0
CW7	CW6	CW5	CW4	CW3	CW2	CW1	CW0

Bit	Name	Description
7-0	CW<7..0>	These eight bits are written to or read from the DIO-24 CNFG Register.

MIO-16 Programming Considerations

This chapter contains programming instructions for operating the MIO-16 circuitry on the AT-MIO-16D board. Programming the AT-MIO-16D involves writing to and reading from the various registers on the board. The programming instructions list the sequence of steps to take. The instructions are language independent; that is, they instruct you to write a value to a given register, to set or clear a bit in a given register, or to detect whether a given bit is set or cleared without presenting the actual code.

Register Programming Considerations

Several write-only registers on the AT-MIO-16D contain bits that control several independent pieces of the onboard circuitry. In the instructions for setting or clearing bits, specific register bits should be set or cleared without changing the current state of the remaining bits in the register. However, writing to these registers simultaneously affects all register bits. You cannot read these registers to determine which bits have been set or cleared in the past; therefore, you should maintain a software copy of the write-only registers. This software copy can then be read to determine the status of the write-only registers. To change the state of a single bit without disturbing the remaining bits, set or clear the bit in the software copy and write the software copy to the register.

Initializing the MIO-16 Circuitry of the AT-MIO-16D Board

The MIO-16 hardware must be initialized for the circuitry to operate properly. To initialize the MIO-16 hardware, complete these steps:

1. Write 0 to Command Register 1 (16-bit write).
2. Write 0 to Command Register 2 (16-bit write).
3. Write 0 to the Mux-Gain Register.
4. Initialize the Am9513A (described below).
5. Write 0 to the A/D Clear Register.
6. Write 0 to the INT2CLR Register (16-bit write).

This sequence leaves the AT-MIO-16D circuitry in the following state:

- DMA disabled.
- All interrupts disabled.
- Outputs of counter/timers in high-impedance state.
- Analog input circuitry initialized.
- A/D FIFO cleared.
- Analog input channel 0 selected.

- Gain of 1 selected.
- All pins on the RTSI switch configured as input pins.

Initializing the Am9513A

Follow the sequence below to initialize the Am9513A Counter/Timer. All writes are 16-bit operations. All values are given in hexadecimal.

1. Issue a master reset by writing FFFF to the Am9513A Command Register.
2. Set up Am9513A 16-bit mode by writing FFEF to the Am9513A Command Register.
3. Point to the Am9513A Master Mode Register by writing FF17 to the Am9513A Command Register.
4. Load the master mode value into the Am9513A Master Mode Register by writing F000 to the Am9513A Data Register.
5. To initialize all five counters – for *ctr* = 1 to 5, follow these steps:
   a. Write FF00 + *ctr* to the Am9513A Command Register to select the Counter Mode Register.
   b. Write 0004 to the Am9513A Data Register to store the counter mode value.
   c. Write FF08 + *ctr* to the Am9513A Command Register to select the Counter Load Register.
   d. Write 3 to the Am9513A Data Register to store an inactive count value in the Counter Load Register.
6. Load all counters with their Counter Load Register values by writing FF5F to the Am9513A Command Register.

After this sequence of writes, the Am9513A Counter/Timer is in the following state:

- 16-bit mode is enabled.
- BCD scalar division is selected.
- The FOUT signal is turned off.
- All counter OUT output pins are set to high-impedance output state.
- All counters are loaded with a non-terminal count value.

For additional details concerning the Am9513A Counter/Timer, see Appendix E, *Am9513A Data Sheet*.

Note: If a data acquisition operation is to be executed *and* Counter 4 of Am9513A is not to be used, then write 0000 to the Am9513A Data Register (instead of 0004) when *ctr* = 4. Writing 0000 to the Am9513A Data Register causes the output of Counter 4 to be low and therefore prevents improper termination of the data acquisition operation.

Initializing the Analog Output Circuitry

The AT-MIO-16D powers up with the analog output circuitry at an unknown voltage. For most applications, the analog output circuitry should be initialized to 0 V.

If the analog output channel is configured for unipolar operation, write 0 to the DAC Register (16-bit write) for that channel.

If the analog output channel is configured for two's complement bipolar output, write 0 to the DAC Register (16-bit write) for that channel.

Programming the Analog Input Circuitry

Programming the analog input circuitry to obtain a single A/D conversion involves the following sequence of steps:

1. Select the analog input channel and gain.

2. Initiate an A/D conversion.

3. Read the A/D conversion result.

In addition, you can program the binary format of the A/D conversion result and you can reset the analog input circuitry.

1. Select analog input channel and gain.

The analog input channel and gain are selected by writing to the Mux-Gain Register. Bits 7 and 6 control the gain, and bits 3 through 0 select the analog input channel. See the Mux-Gain Register bit description earlier in this chapter for gain and analog input channel bit patterns. Set up the bits as given in the Mux-Gain Register bit description and write to the Mux-Gain Register.

Once the Mux-Gain Register is set up, it needs to be written to only when you need to change the analog input channel or gain setting.

2. Initiate an A/D conversion.

An A/D conversion can be initiated in one of two ways–a software-generated pulse or a hardware pulse.

To initiate an A/D conversion through software, write 0 to the A/D Start Convert Register.

To initiate an A/D conversion through hardware, apply an active low pulse to the EXTCONV* pin on the AT-MIO-16D I/O connector. See the *Data Acquisition Timing Connections* section in Chapter 2, *Configuration and Installation*, for EXTCONV* signal specifications.

Once an A/D conversion is initiated, the ADC automatically stores the result in the A/D FIFO at the end of its conversion cycle.

3. Read the A/D conversion result.

A/D conversion results are obtained by reading the A/D FIFO Register. Before reading the A/D FIFO, however, the Status Register must be read to determine whether the A/D FIFO contains any results.

To read the A/D conversion result, do the following:

a. Read the Status Register (16-bit read).

b. If the CONVAVAIL bit is set (bit 13), then read the A/D FIFO Register to obtain the result.

Reading the A/D FIFO Register removes the A/D conversion result from the A/D FIFO. The binary modes of the A/D FIFO output are explained below.

The CONVAVAIL bit indicates whether one or more A/D conversion results are stored in the A/D FIFO. If the CONVAVAIL bit is not set, the A/D FIFO is empty and reading the A/D FIFO Register returns meaningless data. Once an A/D conversion is initiated, the CONVAVAIL bit is set within 10 µsec, indicating that the data conversion result can be read from the FIFO.

An A/D FIFO overflow condition occurs if more than 512 conversions are initiated and stored in the A/D FIFO before the A/D FIFO Register is read. If this condition occurs, the OVERFLOW bit is set in the Status Register to alert you that one or more A/D conversion results have been lost because of FIFO overflow. Writing to the A/D Clear Register clears this error flag and empties the A/D FIFO.

A/D FIFO Output Binary Formats

The A/D conversion result can be returned from the A/D FIFO as a two's complement or straight binary value by setting or clearing the 2SCADC* bit in Command Register 1. If the analog input circuitry is configured for the input range 0 to +10 V, straight binary format is recommended (set the 2SCADC* bit). Straight binary format returns numbers between 0 and 4,095 (decimal) when the A/D FIFO Register is read. If the analog input circuitry is configured for the input ranges -5 to +5 V or -10 to +10 V, two's complement format is recommended (clear the 2SCADC* bit). Two's complement format returns numbers between -2,048 and +2,047 (decimal) when the A/D FIFO Register is read.

The factory default setting is the input range -10 to +10 V. Table 4-2 shows input voltage versus A/D conversion value for straight binary format and 0 to +10 V input range. Table 4-3 shows input voltage versus A/D conversion value for two's complement format for both -5 to +5 V and -10 to +10 V input ranges.

Table 4-2. Straight Binary Mode A/D Conversion Values

Input Voltage (Gain = 1)	**A/D Conversion Result Range: 0 to 10 V**	
	Decimal	**Hex**
0	0	0000
2.5	1,024	0400
5.0	2,048	0800
7.5	3,072	0C00
9.9976	4,095	0FFF

To convert from the A/D FIFO value to the input voltage measured, use the following formula:

$$V = \frac{\text{A/D Count}}{4{,}096} * \frac{10\ V}{\text{Gain}}$$

Table 4-3. Two's Complement Mode A/D Conversion Values

Input Voltage	**A/D Conversion Result**			
(Gain = 1)	**Range: -5 to +5 V**		**Range: -10 to +10 V**	
	Decimal	**Hex**	**Decimal**	**Hex**
-10.0	—	—	-2,048	F800
-5.0	-2,048	F800	-1,024	FC00
-2.5	-1,024	FC00	-512	FE00
0	0	0000	0	0000
2.5	1,024	0400	512	0200
4.9976	2,047	07FF	—	—
5.0	—	—	1,024	0400
9.9951	—	—	2,047	07FF

To convert from the A/D FIFO value to the input voltage measured, use the appropriate formula as follows:

$$\pm 5\,V \text{ Range} = \frac{\text{A/D Count}}{2{,}048} * \frac{5\ V}{\text{Gain}}$$

$$\pm 10\,V \text{ Range} = \frac{\text{A/D Count}}{2{,}048} * \frac{10\ V}{\text{Gain}}$$

Clearing the Analog Input Circuitry

The analog input circuitry can be cleared by writing to the A/D Clear Register. This operation leaves the analog input circuitry in the following state:

- Analog input error flags OVERFLOW and OVERRUN are cleared.
- Pending interrupt requests are cleared.
- A/D FIFO is emptied.

Empty the A/D FIFO before starting any A/D conversions. This action guarantees that the A/D conversion results read from the A/D FIFO are the results from the initiated conversions, not results left over from previous conversions.

To clear the analog input circuitry and the A/D FIFO, write 0 to the A/D Clear Register.

Programming Multiple A/D Conversions on a Single Input Channel

The AT-MIO-16D board can be programmed to execute a multiple A/D conversion sequence with the following options:

- A/D conversions can be initiated either by pulses generated by the onboard sample-interval counter or by pulses applied to the EXTCONV* input. These pulses control the conversion rate.
- The entire conversion sequence can be started by a software write operation to the board or by a signal applied to the START TRIG* input.
- You can select either posttrigger or pretrigger operation. In posttrigger operation, the sample counter begins decrementing with each conversion pulse once the conversion sequence is started. When the sample counter reaches zero, the conversion sequence terminates. Thus, all acquired data was received after the trigger or software start. In pretrigger operation, the sample counter does not decrement until a trigger signal is applied to the STOP TRIG input. When the conversion sequence terminates, some of the acquired data has been received before the trigger signal and some has been received after this signal.

The most commonly used configuration is for the onboard sample-interval and sample counters to control the entire data acquisition operation. Programming this configuration is explained here. The other timing configurations are explained in the *External Timing Considerations for Multiple A/D Conversions* section later in this chapter. Multiple-channel scanning is discussed in the *Programming Multiple A/D Conversions with Channel Scanning* section later in this chapter.

The following programming sequences for sample counts less than 65,537 allow the data acquisition circuitry to be retriggered. The sample-interval and sample counters are reloaded at the end of the data acquisition to prepare for another data acquisition operation. The counters do not need reprogramming, and the next data acquisition operation starts when a trigger is received.

Programming multiple A/D conversions on a single channel requires the following programming steps:

1. Select analog input channel and gain.
2. Program the sample-interval counter.
3. Program the sample counter.

4. Clear the A/D circuitry.
5. Enable the data acquisition operation.
6. Apply a trigger.
7. Service the data acquisition operation.

Each of these programming steps is explained below.

1. Select analog input channel and gain.

The analog input channel and gain are selected by writing to the Mux-Gain Register. Bits 7 and 6 control the gain, and bits 3 through 0 select the analog input channel. See the Mux-Gain Register bit description earlier in this chapter for gain and analog input channel bit patterns. Set up the bits as given in the Mux-Gain Register bit description, and write to the Mux-Gain Register.

The Mux-Gain Register needs to be written to only when you need to change the analog input channel or gain setting.

2. Program the sample-interval counter.

Use Counter 3 of the Am9513A Counter/Timer as the sample-interval counter. Counter 3 can be programmed to generate a pulse once every *N* counts. *N* is referred to as the sample interval, that is, the time between successive A/D conversions. *N* can be between 2 and 65,536. One count is equal to the period of the timebase clock used by the counter. The following clocks are available internal to the Am9513A: 1 MHz, 100 kHz, 10 kHz, 1 kHz, and 100 Hz. In addition, the sample-interval timer can use signals connected to any of the Am9513A SOURCE input pins.

To program the sample-interval counter, use the following programming sequence. All writes are 16-bit operations. All values given are hexadecimal.

a. Write FF03 to the Am9513A Command Register to select the Counter 3 Mode Register.

b. Write the mode value to the Am9513A Data Register to store the Counter 3 mode value. Use one of the following mode values:

   8B25 – Selects 1-MHz clock
   8C25 – Selects 100-kHz clock
   8D25 – Selects 10-kHz clock
   8E25 – Selects 1-kHz clock
   8F25 – Selects 100-Hz clock
   8525 – Selects signal at SOURCE5 input as clock (counts the rising edge of the signal, 6 MHz maximum)

c. Write FF0B to the Am9513A Command Register to select the Counter 3 Load Register.

d. Write 2 to the Am9513A Data Register to store the Counter 3 load value.

e. Write FF44 to the Am9513A Command Register to load Counter 3.

f. Write FFF3 to the Am9513A Command Register to step Counter 3 down to 1.

g. Write the desired sample interval to the Am9513A Data Register to store the Counter 3 load value:

   - If the sample interval is between 2 and FFFF (65,535 decimal), write the sample interval to the Am9513A Data Register.
   - If the sample interval is 10000 (65,536 decimal), write 0 to the Am9513A Data Register.

h. Write FF24 to the Am9513A Command Register to arm Counter 3.

After you complete this programming sequence, Counter 3 is configured to generate A/D conversion pulses as soon as application of a trigger causes it to be enabled.

3. Program the sample counter.

Use Counters 4 and 5 of the Am9513A Counter/Timer as the sample counter. The sample counter tallies the number of A/D conversions initiated by Counter 3 and stops Counter 3 when the desired sample count is reached. If the desired sample count is 65,536 or less, use only Counter 4, making Counter 5 available for general-purpose timing applications. If the desired sample count is greater than 65,536, use both Counters 4 and 5.

Sample Counts 2 through 65,536

To program the sample counter for sample counts up to 65,536, use the following programming sequence. The minimum permitted sample count is two. All writes are 16-bit operations. All values given are hexadecimal.

a. Write FF04 to the Am9513A Command Register to select the Counter 4 Mode Register.

b. Write 1025 to the Am9513A Data Register to store the Counter 4 mode value.

c. Write FF0C to the Am9513A Command Register to select the Counter 4 Load Register.

d. Write the sample count value to the Am9513A Data Register to store the Counter 4 load value:

   - If the sample count is between 2 and FFFF (65,535 decimal), write the sample count to the Am9513A Data Register.
   - If the sample count is 10000 (65,536 decimal), write 0 to the Am9513A Data Register.

e. Write FF48 to the Am9513A Command Register to load Counter 4.

f. Write FFF4 to the Am9513A Command Register to decrement Counter 4.

g. Write FF28 to the Am9513A Command Register to arm Counter 4.

h. Clear the 16*/32 CNT bit in Command Register 1 to notify the hardware that only Counter 4 will be used as the sample counter.

After you complete this programming sequence, Counter 4 is configured to count A/D conversion pulses generated by Counter 3 and turns off the data acquisition operation when Counter 4 decrements to zero.

Sample Counts Greater Than 65,536

To program the sample counter for sample counts greater than 65,536, use the following programming sequence. The lower 16 bits of the sample count are stored in Counter 4, and the upper 16 bits of the sample count are stored in Counter 5. All writes are 16-bit operations. All values given are hexadecimal.

a. Write FF04 to the Am9513A Command Register to select the Counter 4 Mode Register.

b. Write 1025 to the Am9513A Data Register to store the Counter 4 mode value.

c. Write FF0C to the Am9513A Command Register to select the Counter 4 Load Register.

d. Write the least significant 16 bits of the sample count value minus 1 to the Am9513A Data Register to store the Counter 4 load value.

   - If the 16 LSBs are all 0, write FFFF.

e. Write FF48 to the Am9513A Command Register to load Counter 4.

f. Write 0 to the Am9513A Data Register to store 0 into the Load Register for Counter 4 reloading.

g. Write FF28 to the Am9513A Command Register to arm Counter 4.

h. Write FF05 to the Am9513A Command Register to select the Counter 5 Mode Register.

i. Write 25 to the Am9513A Data Register to store the Counter 5 mode value.

j. Write FF0D to the Am9513A Command Register to select the Counter 5 Load Register.

k. Take the most significant 16 bits of the sample count and do the following:

   - If the least significant 16 bits of the sample count are all 0 or all 0 except for a 1 in the least significant bit, write the most significant 16 bits to the Am9513A Data Register to store the Counter 5 load value.

   - Otherwise, add 1 to the most significant 16 bits of the sample count and write that value to the Am9513A Data Register to store the Counter 5 load value.

l. Write FF70 to the Am9513A Command Register to load and arm Counter 5.

m. Set the 16*/32 CNT bit in Command Register 1 to notify the hardware that both Counters 4 and 5 will be used as the sample counter.

After you complete this programming sequence, Counter 4 is configured to count A/D conversion pulses generated by Counter 3, and Counter 5 decrements every time Counter 4 reaches zero. The data acquisition operation is terminated when both Counters 4 and 5 reach zero.

4. Clear the A/D circuitry.

Before you start the data acquisition operation, you must empty the A/D FIFO to clear out any old A/D conversion results. This emptying must be done after the counters are programmed in

case any spurious edges were caused while programming the counters. Write 0 to the A/D Clear Register to empty the FIFO.

5. Enable the data acquisition operation.

To enable the data acquisition operation such that A/D conversions begin when a trigger is received, set the DAQEN bit in Command Register 1.

6. Apply a trigger.

Once set up by the preceding steps, the data acquisition operation is initiated when a trigger is received. A trigger can be provided in one of two ways–through software or through hardware.

To initiate the data acquisition operation through software, write 0 to the Start DAQ Register (16-bit write).

To initiate the data acquisition operation through hardware, apply an active low pulse to the START TRIG* pin on the AT-MIO-16D I/O connector. See the *Data Acquisition Timing Connections* section in Chapter 2, *Configuration and Installation*, for START TRIG* signal specifications.

Once the trigger is applied, Counter 3 generates pulses initiating A/D conversions once every sample interval until the sample counter reaches zero.

7. Service the data acquisition operation.

Once the data acquisition operation is started by application of a trigger, the operation must be serviced by reading the A/D FIFO Register every time an A/D conversion result becomes available. To do this, perform the following sequence until the desired number of conversion results have been read:

a. Read the Status Register (16-bit read).

b. If the CONVAVAIL bit is set (bit 13), read the A/D FIFO Register to obtain the result.

You can also use interrupts or DMA to service the data acquisition operation. These topics are discussed later in this chapter.

Two error conditions may occur during a data acquisition operation–an overflow error or an overrun error. These error conditions are reported through the Status Register and should be checked every time the Status Register is read to check the CONVAVAIL bit. If either of these error conditions occurs, the data acquisition operation stops.

An overflow condition occurs if more than 512 A/D conversions have been stored in the A/D FIFO without the A/D FIFO being read; that is, the A/D FIFO is full and cannot accept any more data. This condition occurs if the software loop reading the A/D FIFO Register is not fast enough to keep up with the A/D conversion rate. When an overflow occurs, at least one A/D conversion result is lost. An overflow condition has occurred if the OVERFLOW bit in the Status Register is set.

An overrun condition occurs if an A/D conversion is initiated before the previous conversion is finished. This condition may result in one or more missing A/D conversions. This condition occurs if the sample interval is too small (sample rate is too high). An overrun condition has occurred if the OVERRUN bit in the Status Register is set. The maximum recommended single-channel data acquisition rate for the AT-MIO-16D is 100 ksamples/sec.

Both the OVERFLOW and OVERRUN bits in the Status Register are cleared by writing to the A/D Clear Register.

External Timing Considerations for Multiple A/D Conversions

The case of controlled data acquisition operations using the onboard sample-interval and sample counters was described above. The two external timing cases are described here–using the STOP TRIG input to control the sample counter, and applying pulses to the EXTCONV* input.

Pretriggering with the STOP TRIG Signal

In this case, the sample-interval counter starts counting when a rising edge signal is applied to the STOP TRIG input on the AT-MIO-16D I/O connector. You program the sample counter for active high level gating on Gate 4. The data acquisition operation is initiated by writing to the Start DAQ Register or by a pulse on the START TRIG* input. The sample count register does not begin counting samples until a rising edge is applied to STOP TRIG. To perform this operation, complete these steps:

1. Select analog input channel and gain.
2. Program the sample-interval counter.
3. Program the sample counter.
4. Clear the A/D circuitry.
5. Apply a trigger.
6. Service the data acquisition operation.

1. Select analog input channel and gain.

The analog input channel and gain are selected by writing to the Mux-Gain Register. Bits 7 and 6 control the gain, and bits 3 through 0 select the analog input channel. See the Mux-Gain Register bit description earlier in this chapter for gain and analog input channel bit patterns. Set up the bits as given in the Mux-Gain Register bit description, and write to the Mux-Gain Register.

Once set up with an initial value, the Mux-Gain Register needs to be written to only when you need to change the analog input channel or gain setting.

2. Program the sample-interval counter.

Use Counter 3 of the Am9513A Counter/Timer as the sample-interval counter. Counter 3 can be programmed to generate a pulse once every *N* counts. *N* is referred to as the sample interval, that is, the time between successive A/D conversions. *N* can be between 2 and 65,536. One count is equal to the period of the timebase clock used by the counter. The following clocks are available internal to the Am9513A: 1 MHz, 100 kHz, 10 kHz, 1 kHz, and 100 Hz. In addition, the sample-interval timer can use signals connected to any of the Am9513A SOURCE input pins.

To program the sample-interval counter, use the following programming sequence. All writes are 16-bit operations. All values given are hexadecimal.

a. Write FF03 to the Am9513A Command Register to select the Counter 3 Mode Register.

b. Write the mode value to the Am9513A Data Register to store the Counter 3 mode value. Use one of the following mode values:

   8B25 – Selects 1-MHz clock
   8C25 – Selects 100-kHz clock
   8D25 – Selects 10-kHz clock
   8E25 – Selects 1-kHz clock
   8F25 – Selects 100-Hz clock
   8525 – Selects signal at SOURCE5 input as clock (counts the rising edge of the signal, 6 MHz maximum)

c. Write FF0B to the Am9513A Command Register to select the Counter 3 Load Register.

d. Write 2 to the Am9513A Data Register to store the Counter 3 load value.

e. Write FF44 to the Am9513A Command Register to load Counter 3.

f. Write FFF3 to the Am9513A Command Register to step Counter 3 down to one.

g. Write the desired sample interval to the Am9513A Data Register to store the Counter 3 load value:

   - If the sample interval is between 2 and FFFF (65,535 decimal) inclusive, write the sample interval to the Am9513A Data Register.

   - If the sample interval is 10000 (65,536 decimal), write 0 to the Am9513A Data Register.

h. Write FF24 to the Am9513A Command Register to arm Counter 3.

After you complete this programming sequence, Counter 3 is configured to generate A/D conversion pulses as soon as you enable it by applying a trigger.

3. Program the sample counter.

Counters 4 and 5 of the Am9513A Counter/Timer are used as the sample counter. The sample counter tallies the number of A/D conversions initiated by Counter 3 and stops Counter 3 when the desired sample count is reached. If the desired sample count is 65,536 or less, only Counter 4 needs to be used, making Counter 5 available for general-purpose timing applications. If the desired sample count is greater than 65,536, both Counters 4 and 5 must be used.

Sample Counts 2 through 65,536. To program the sample counter for sample counts up to 65,536, use the following programming sequence. The minimum permitted sample count is two. All writes are 16-bit operations. All values given are hexadecimal.

a. Write FF04 to the Am9513A Command Register to select the Counter 4 Mode Register.

b. Write 9025 to the Am9513A Data Register to store the Counter 4 mode value.

c. Write FF0C to the Am9513A Command Register to select the Counter 4 Load Register.

d. Write the sample count value to the Am9513A Data Register to store the Counter 4 load value:

   - If the sample count is between 2 and FFFF (65,535 decimal), write the sample count to the Am9513A Data Register.

   - If the sample count is 10000 (65,536 decimal), write 0 to the Am9513A Data Register.

e. Write FF48 to the Am9513A Command Register to load Counter 4.

f. Write FFF4 to the Am9513A Command Register to decrement Counter 4.

g. Write FF28 to the Am9513A Command Register to arm Counter 4.

h. Clear the 16*/32 CNT bit in Command Register 1 to notify the hardware that only Counter 4 will be used as the sample counter.

After you complete this programming sequence, Counter 4 is configured to count A/D conversion pulses generated by Counter 3 and turns off the data acquisition operation when Counter 4 decrements to zero. Counter 4 begins counting A/D conversion pulses when a rising edge signal is received on the STOP TRIG input. A/D conversion data stored before receipt of the STOP TRIG signal are pretrigger samples.

Sample Counts Greater Than 65,536. To program the sample counter for sample counts greater than 65,536, use the following programming sequence. The lower 16 bits of the sample count are stored in Counter 4, and the upper 16 bits of the sample count are stored in Counter 5. All writes are 16-bit operations. All values given are hexadecimal.

a. Write FF04 to the Am9513A Command Register to select the Counter 4 Mode Register.

b. Write 9025 to the Am9513A Data Register to store the Counter 4 mode value.

c. Write FF0C to the Am9513A Command Register to select the Counter 4 Load Register.

d. Write the least significant 16 bits of the sample count value minus 1 to the Am9513A Data Register to store the Counter 4 load value.

   - If the least significant 16 bits are all zeros, write FFFF.

e. Write FF48 to the Am9513A Command Register to load Counter 4.

f. Write 0 to the Am9513A Data Register to store 0 into the Load Register for Counter 4 reloading.

g. Write FF28 to the Am9513A Command Register to arm Counter 4.

h. Write FF05 to the Am9513A Command Register to select the Counter 5 Mode Register.

i. Write 25 to the Am9513A Data Register to store the Counter 5 mode value.

j. Write FF0D to the Am9513A Command Register to select the Counter 5 Load Register.

k. Take the most significant 16 bits of the sample count and do the following:

- If the least significant 16 bits of the sample count are all zeros or all zeros except for a 1 in the least significant bit, write the most significant 16 bits to the Am9513A Data Register to store the Counter 5 load value.
- Otherwise, add one to the most significant 16 bits of the sample count and write that value to the Am9513A Data Register to store the Counter 5 load value.

l. Write FF70 to the Am9513A Command Register to load and arm Counter 5.

m. Set the 16*/32 CNT bit in Command Register 1 to notify the hardware that both Counters 4 and 5 will be used as the sample counter.

After you complete this programming sequence, Counter 4 is configured to count A/D conversion pulses generated by Counter 3, and Counter 5 decrements every time Counter 4 reaches zero. The data acquisition operation is terminated when both Counters 4 and 5 reach zero. Counters 4 and 5 begin counting A/D conversion pulses when a rising edge signal is received on the STOP TRIG input. A/D conversion data stored before receipt of the STOP TRIG signal are pretrigger samples.

4. Clear the A/D circuitry.

Before you start the data acquisition operation, the A/D FIFO must be emptied to clear out any old A/D conversion results. You must do this emptying after the counters are programmed in case any spurious edges were caused while programming the counters. Write 0 to the A/D Clear Register to empty the FIFO (16-bit write).

5. Apply a trigger.

Once set up by the preceding steps, the data acquisition operation is initiated when a trigger is received. A trigger can be provided in one of two ways–through software or through hardware.

To initiate the data acquisition operation through software, write 0 to the Start DAQ Register (16-bit write).

To initiate the data acquisition operation through hardware, apply an active low pulse to the START TRIG* pin on the AT-MIO-16D I/O connector. See the *Data Acquisition Timing Connections* section in Chapter 2, *Configuration and Installation*, for START TRIG* signal specifications.

Once the trigger is applied, Counter 3 generates pulses initiating A/D conversions once every sample interval until the sample counter reaches 0.

6. Service the data acquisition operation.

Once the data acquisition operation is started by application of a trigger, the operation must be serviced by reading the A/D FIFO Register every time an A/D conversion result becomes available. To do this, perform the following sequence until the desired number of conversion results have been read:

a. Read the Status Register (16-bit read).

b. If the CONVAVAIL bit is set (bit 13), read the A/D FIFO Register to obtain the result.

You can use interrupts or DMA to service the data acquisition operation. These topics are discussed later in this chapter.

Two error conditions may occur during a data acquisition operation–an overflow error or an overrun error. These error conditions are reported through the Status Register and should be checked every time the Status Register is read to check the CONVAVAIL bit. If either of these error conditions occurs, the data acquisition operation stops.

An overflow condition occurs if more than 512 A/D conversions have been stored in the A/D FIFO without the A/D FIFO being read; that is, the A/D FIFO is full and cannot accept any more data. This condition occurs if the software loop reading the A/D FIFO Register is not fast enough to keep up with the A/D conversion rate. When an overflow occurs, at least one A/D conversion result is lost. An overflow condition has occurred if the OVERFLOW bit in the Status Register is set.

An overrun condition occurs if an A/D conversion is initiated before the previous conversion is finished. This condition may result in one or more missing A/D conversions. This condition occurs if the sample interval is too small (sample rate is too high). An overrun condition has occurred if the OVERRUN bit in the Status Register is set. The maximum recommended single-channel data acquisition rate for the AT-MIO-16D is 100 ksamples/sec.

Both the OVERFLOW and OVERRUN bits in the Status Register are cleared by writing to the A/D Clear Register.

Once steps 1 through 5 of this sequence are completed, Counter 3 is armed and begins generating pulses. The sample counter does not begin counting samples until a rising edge signal is detected on the STOP TRIG input. When the sample count decrements to zero, the data acquisition operation is halted. The STOP TRIG signal specifications are given in Chapter 2, *Configuration and Installation*.

Controlling Multiple A/D Conversions with the EXTCONV* Signal

When you use EXTCONV* to control multiple A/D conversions, none of the onboard counters are used. Pulses applied to the EXTCONV* input initiate the A/D conversions. To perform this operation, complete these steps:

1. Select analog input channel and gain.

2. Clear the A/D circuitry.

3. Service the data acquisition operation.

First, make certain that Counter 3 is reset as described in the *Resetting the Hardware after a Data Acquisition Operation* section later in this chapter. If Counter 3 is not reset, it may be driving the EXTCONV* line and therefore prevent another signal from successfully driving the line high or low.

1. Select analog input channel and gain.

The analog input channel and gain are selected by writing to the Mux-Gain Register. Bits 7 and 6 control the gain, and bits 3 through 0 select the analog input channel. See the Mux-Gain Register bit description earlier in this chapter for gain and analog input channel bit patterns. Set up the bits as given in the Mux-Gain Register bit description and write to the Mux-Gain Register.

The Mux-Gain Register needs to be written to only when you need to change the analog input channel or gain setting.

2. Clear the A/D circuitry.

Before starting the data acquisition operation, the A/D FIFO must be emptied to clear out any old A/D conversion results. Write 0 to the A/D Clear Register to empty the FIFO.

3. Service the data acquisition operation.

Once an external trigger starts the data acquisition operation, the operation is serviced by reading the A/D FIFO Register every time an A/D conversion result becomes available. To do this, perform the following sequence until the desired number of conversion results have been read:

a. Read the Status Register (16-bit read).

b. If the CONVAVAIL bit is set (bit 13), read the A/D FIFO Register to obtain the result.

Interrupts or DMA can also be used to service the data acquisition operation. These topics are discussed later in this chapter.

Two error conditions may occur during a data acquisition operation–an overflow error or an overrun error. These error conditions are reported through the Status Register and should be checked every time the Status Register is read to check the CONVAVAIL bit. If either of these error conditions occurs, the data acquisition operation stops.

An overflow condition occurs if more than 512 A/D conversions have been stored in the A/D FIFO without the A/D FIFO being read; that is, the A/D FIFO is full and cannot accept any more data. This condition occurs if the software loop reading the A/D FIFO Register is not fast enough to keep up with the A/D conversion rate. When an overflow occurs, at least one A/D conversion result is lost. An overflow condition has occurred if the OVERFLOW bit in the Status Register is set.

An overrun condition occurs if an A/D conversion is initiated before the previous conversion is finished. This condition may result in one or more missing A/D conversions. This condition occurs if the sample interval is too small (sample rate is too high). An overrun condition has occurred if the OVERRUN bit in the Status Register is set. The maximum recommended single-channel data acquisition rate for the AT-MIO-16D is 100 ksamples/sec.

Both the OVERFLOW and OVERRUN bits in the Status Register are cleared by writing to the A/D Clear Register.

An A/D conversion is initiated and stored in the A/D FIFO every time a low-to-high edge is detected on the EXTCONV* input. See Chapter 2, *Configuration and Installation*, for EXTCONV* signal specifications.

Programming Multiple A/D Conversions with Channel Scanning

The data acquisition programming sequences described earlier program the AT-MIO-16D for multiple A/D conversion on a single input channel. You can also program the AT-MIO-16D for scanning analog input channels and switching gain settings during the data acquisition operation. The sequence of A/D channels and gain settings, called the *scan sequence*, is programmed into the mux-gain memory.

There are two types of multiple A/D conversions with channel scanning–continuous channel scanning and interval channel scanning. *Continuous channel scanning* cycles through the scan sequence in the mux-gain memory and repeats the scan sequence until the sample counter terminates the data acquisition. There is no delay between the cycles of the scan sequence. Continuous channel scanning can be thought of as a *round-robin* approach to scanning multiple channels.

Interval channel scanning gives each scan sequence a programmed time interval, called a *scan interval*. Each cycle of the scan sequence begins at the time interval specified by the scan interval. If the sample-interval counter is programmed for the minimum time required to complete an A/D conversion, interval channel scanning can be thought of as a *pseudo-simultaneous* scanning of multiple channels; that is, all channels in the scan sequence are read as quickly as possible at the beginning of each scan interval.

Multiple A/D Conversions with Continuous Channel Scanning (Round Robin)

Programming continuous scanning of multiple A/D conversions involves the following programming steps:

1. Set up the analog channel and gain selection sequence.
2. Program the sample-interval counter.
3. Program the sample counter.
4. Clear the A/D circuitry and reset the multiplexer counter.
5. Enable the scanning data acquisition operation.
6. Apply a trigger.
7. Service the data acquisition operation.

Setting the SCANEN bit in Command Register 1 enables scanning during multiple A/D conversions. You must set this bit regardless of the type of scanning used; otherwise, only a single channel is scanned. In addition, a channel and gain scan sequence must be stored in the mux-gain memory.

1. Set up the analog channel and gain selection sequence.

During a scanning data acquisition operation, a selected number of locations in the mux-gain memory are clocked through. A new mux-gain memory location is selected after each A/D conversion. The first conversion is performed on the first channel and gain setting in the memory. The second conversion is performed on the second channel and gain setting, and so on. The last entry written to the mux-gain memory must have the LASTONE bit set. This bit marks the end of the scan sequence. After the last conversion is performed, the scan sequence starts over. If there are *X* entries in the mux-gain memory, every *X*th conversion in the data collected is performed on the same channel and gain setting.

Multiple conversions can be performed on each entry in the mux-gain memory before incrementing to the next entry in the scan sequence. If the SCANDIV bit in Command Register 1 is set, the mux-gain memory increments to the next entry when an active low pulse is detected on the Am9513A Counter/Timer OUT1 signal. If the SCANDIV bit is cleared, the mux-gain memory is incremented to the next entry after every conversion.

The mux-gain memory must be loaded with the desired scan sequence before data acquisition begins. To load the mux-gain memory, perform the following write operations where *X* is the number of entries in the scan sequence:

For $i = 0$ to X-1, do the following:

a. Write *i* to the Mux-Counter Register to select the mux-gain memory location.

b. Write the desired analog channel selection and gain setting to the Mux-Gain Register to load the mux-gain memory at location *i* .

c. If $i = X$-1, also set the LASTONE bit when writing to the Mux-Gain Register.

2. Program the sample-interval counter.

Use Counter 3 of the Am9513A Counter/Timer as the sample-interval counter. Counter 3 can be programmed to generate a pulse once every *N* counts. *N* is referred to as the sample interval, that is, the time between successive A/D conversions. *N* can be between 2 and 65,536. One count is equal to the period of the timebase clock used by the counter. The following clocks are available internal to the Am9513A: 1 MHz, 100 kHz, 10 kHz, 1 kHz, and 100 Hz. In addition, the sample-interval timer can use signals connected to any of the Am9513A SOURCE input pins.

To program the sample-interval counter, use the following programming sequence. All writes are 16-bit operations. All values given are hexadecimal.

a. Write FF03 to the Am9513A Command Register to select the Counter 3 Mode Register.

b. Write the mode value to the Am9513A Data Register to store the Counter 3 mode value. Use one of the following mode values:

8B25 – Selects 1-MHz clock
8C25 – Selects 100-kHz clock
8D25 – Selects 10-kHz clock
8E25 – Selects 1-kHz clock
8F25 – Selects 100-Hz clock
8525 – Selects signal at SOURCE5 input as clock (counts the rising edge of the signal, 6 MHz maximum)

c. Write FF0B to the Am9513A Command Register to select the Counter 3 Load Register.

d. Write 2 to the Am9513A Data Register to store the Counter 3 load value.

e. Write FF44 to the Am9513A Command Register to load Counter 3.

f. Write FFF3 to the Am9513A Command Register to step Counter 3 down to one.

g. Write the desired sample interval to the Am9513A Data Register to store the Counter 3 load value:

   - If the sample interval is between 2 and FFFF (65,535 decimal), write the sample interval to the Am9513A Data Register.
   - If the sample interval is 10000 (65,536 decimal), write 0 to the Am9513A Data Register.

h. Write FF24 to the Am9513A Command Register to arm Counter 3.

After you complete this programming sequence, Counter 3 is configured to generate A/D conversion pulses as soon as application of a trigger causes it to be enabled.

3. Program the sample counter.

Use Counters 4 and 5 of the Am9513A Counter/Timer as the sample counter. The sample counter tallies the number of A/D conversions initiated by Counter 3 and stops Counter 3 when the desired sample count is reached. The sample count should be programmed as a multiple of the number of entries in the mux-gain memory. If the desired sample count is 65,536 or less, only Counter 4 needs to be used, making Counter 5 available for general-purpose timing applications. If the desired sample count is greater than 65,536, both Counters 4 and 5 must be used.

Sample Counts 2 through 65,536. To program the sample counter for sample counts up to 65,536, use the following programming sequence. The minimum permitted sample count is 2. All writes are 16-bit operations. All values given are hexadecimal.

a. Write FF04 to the Am9513A Command Register to select the Counter 4 Mode Register.

b. Write 1025 to the Am9513A Data Register to store the Counter 4 mode value.

c. Write FF0C to the Am9513A Command Register to select the Counter 4 Load Register.

d. Write the sample count value to the Am9513A Data Register to store the Counter 4 load value:

   - If the sample count is between 2 and FFFF (65,535 decimal), write the sample count minus 1 to the Am9513A Data Register.
   - If the sample count is 10000 (65,536 decimal), write 0 to the Am9513A Data Register.

e. Write FF48 to the Am9513A Command Register to load Counter 4.

f. Write FFF4 to the Am9513A Command Register to decrement Counter 4.

g. Write FF28 to the Am9513A Command Register to arm Counter 4.

h. Clear the 16*/32 CNT bit in Command Register 1 to notify the hardware that only Counter 4 will be used as the sample counter.

After you complete this programming sequence, Counter 4 is configured to count A/D conversion pulses generated by Counter 3 and turns off the data acquisition operation when Counter 4 reaches zero.

Sample Counts Greater Than 65,536. To program the sample counter for sample counts greater than 65,536, use the following programming sequence. The lower 16 bits of the sample count are stored in Counter 4, and the upper 16 bits of the sample count are stored in Counter 5. All writes are 16-bit operations. All values given are hexadecimal.

a. Write FF04 to the Am9513A Command Register to select the Counter 4 Mode Register.

b. Write 1025 to the Am9513A Data Register to store the Counter 4 mode value.

c. Write FF0C to the Am9513A Command Register to select the Counter 4 Load Register.

d. Write the least significant 16 bits of the sample count value minus 1 to the Am9513A Data Register to store the Counter 4 load value.

   - If the least significant 16 bits are all zero, write FFFF.

e. Write FF48 to the Am9513A Command Register to load and arm Counter 4.

f. Write 0 to the Am9513A Data Register to store 0 into the Load Register for Counter 4 reloading.

g. Write FF28 to the Am9513A Command Register to arm Counter 4.

h. Write FF05 to the Am9513A Command Register to select the Counter 5 Mode Register.

i. Write 25 to the Am9513A Data Register to store the Counter 5 mode value.

j. Write FF0D to the Am9513A Command Register to select the Counter 5 Load Register.

k. Take the most significant 16 bits of the sample count and do the following:

   - If the least significant 16 bits of the sample count are all zeros or all zeros except for a 1 in the least significant bit, write the most significant 16 bits to the Am9513A Data Register to store the Counter 5 load value.

   - Otherwise, add 1 to the most significant 16 bits of the sample count and write that value to the Am9513A Data Register to store the Counter 5 load value.

l. Write FF70 to the Am9513A Command Register to load and arm Counter 5.

m. Set the 16*/32 CNT bit in Command Register 1 to notify the hardware that both Counters 4 and 5 will be used as the sample counter.

After you complete this programming sequence, Counter 4 is configured to count A/D conversion pulses generated by Counter 3, and Counter 5 increments every time Counter 4 reaches zero. The data acquisition operation terminates when both Counters 4 and 5 reach zero and the last entry in the mux-gain memory is served.

4. Clear the A/D circuitry and reset the multiplexer counter.

Before starting the data acquisition operation, the A/D FIFO must be emptied to clear out any old A/D conversion results. You must do this emptying after the counters are programmed in case any spurious edges were caused while programming the counters. Write 0 to the A/D Clear Register to empty the FIFO.

Write 0 to the Mux-Counter Register to set the analog input circuitry to the first channel and gain setting of the scan sequence.

5. Enable the scanning data acquisition operation.

To enable the scanning data acquisition operation such that A/D conversions begin when a trigger is received, set the DAQEN bit and the SCANEN bit in Command Register 1.

6. Apply a trigger.

Once set up by the preceding steps, the data acquisition operation is initiated when a trigger is received. A trigger can be provided in one of two ways–through software or through hardware.

To initiate the data acquisition operation through software, write 0 to the Start DAQ Register.

To initiate the data acquisition operation through hardware, apply an active low pulse to the START TRIG* pin on the AT-MIO-16D I/O connector. See the *Data Acquisition Timing Connections* section in Chapter 2, *Configuration and Installation*, for START TRIG* signal specifications.

Once the trigger is applied, Counter 3 generates pulses initiating A/D conversions once every sample interval until the sample counter reaches zero.

7. Service the data acquisition operation.

Once the data acquisition operation is started by application of a trigger, the operation is serviced by reading the A/D FIFO Register every time an A/D conversion result becomes available. To do this, perform the following sequence until the desired number of conversion results have been read:

a. Read the Status Register (16-bit read).

b. If the CONVAVAIL bit is set (bit 13), read the A/D FIFO Register to obtain the result.

Interrupts or DMA can also be used to service the data acquisition operation. These topics are discussed later in this chapter.

Two error conditions may occur during a data acquisition operation–an overflow error or an overrun error. These error conditions are reported through the Status Register and should be checked every time the Status Register is read to check the CONVAVAIL bit. If either of these error conditions occurs, the data acquisition operation stops.

An overflow condition occurs if more than 512 A/D conversions have been stored in the A/D FIFO without the A/D FIFO being read; that is, the A/D FIFO is full and cannot accept any more

data. This condition occurs if the software loop reading the A/D FIFO Register is not fast enough to keep up with the A/D conversion rate. When an overflow occurs, at least one A/D conversion result is lost. An overflow condition has occurred if the OVERFLOW bit in the Status Register is set.

An overrun condition occurs if a second A/D conversion is initiated before the previous conversion is finished. This condition may result in one or more missing A/D conversions. This condition occurs if the sample interval is too small (the sample rate is too high). An overrun condition has occurred if the OVERRUN bit in the Status Register is set.

Scanned data acquisition requires slower data acquisition rates than single-channel data acquisition because signals must settle each time channels are switched. See Table 4-4 for the maximum recommended multiple-channel data acquisition rates.

Both the OVERFLOW and OVERRUN bits in the Status Register are cleared by writing to the A/D Clear Register.

Multiple A/D Conversions with Interval Channel Scanning (Pseudo-Simultaneous)

Programming scanned multiple A/D conversions with a scan interval involves the following programming steps:

1. Set up the analog channel and gain selection sequence.
2. Program the sample-interval counter.
3. Program the sample counter.
4. Program the scan-interval counter.
5. Clear the A/D circuitry and reset the multiplexer counter.
6. Enable the scanning data acquisition operation.
7. Apply a trigger.
8. Service the data acquisition operation.

Setting the SCANEN bit in Command Register 1 enables scanning during multiple A/D conversions. You must set this bit regardless of the type of scanning used; otherwise, only a single channel is scanned. In addition, a channel and gain scan sequence must be stored in the mux-gain memory.

Setting the SCN2 bit in Command Register 2 enables the use of a scan interval during multiple A/D conversions. The scan-interval counter gives each cycle through the scan sequence a time interval. The scan-interval counter begins counting at the start of the scan sequence programmed into the mux-gain memory. When the scan sequence completes, the next cycle through the scan sequence does not begin until the scan-interval counter has reached its terminal count. Be sure that the scan-interval counter allows enough time for all conversions in a scan sequence to occur so that conversions are not missed.

1. Set up the analog channel and gain selection sequence.

During a scanning data acquisition operation, a selected number of locations in the mux-gain memory are clocked through. A new mux-gain memory location is selected after each A/D conversion. The first conversion is performed on the first channel and gain setting in the memory. The second conversion is performed on the second channel and gain setting, and so on. The last entry written to the mux-gain memory must have the LASTONE bit set. This bit marks the end of the scan sequence. After the last conversion is performed, the scan sequence starts over. If there are *X* entries in the mux-gain memory, every *X*th conversion in the data collected is performed on the same channel and gain setting.

Multiple conversions can be performed on each entry in the mux-gain memory before incrementing to the next entry in the scan sequence. If the SCANDIV bit in Command Register 1 is set, the mux-gain memory increments to the next entry when an active low pulse is detected on the Am9513A Counter/Timer OUT1 signal. If the SCANDIV bit is cleared, the mux-gain memory is incremented to the next entry after every conversion.

The mux-gain memory must be loaded with the desired scan sequence before data acquisition begins. To load the mux-gain memory, perform the following write operations where *X* is the number of entries in the scan sequence

For $i = 0$ to X-1, do the following:

a. Write *i* to the Mux-Counter Register to select the mux-gain memory location.

b. Write the desired analog channel selection and gain setting to the Mux-Gain Register to load the mux-gain memory at location *i* .

c. If $i = X$-1, also set the LASTONE bit when writing to the Mux-Gain Register.

2. Program the sample-interval counter.

Use Counter 3 of the Am9513A Counter/Timer as the sample-interval counter. You can program Counter 3 to generate a pulse once every *N* counts. *N* is referred to as the sample interval, that is, the time between successive A/D conversions. *N* can be between 2 and 65,536. One count is equal to the period of the timebase clock used by the counter. The following clocks are available internal to the Am9513A–1 MHz, 100 kHz, 10 kHz, 1 kHz, and 100 Hz. In addition, the sample-interval timer can use signals connected to any of the Am9513A SOURCE input pins.

To program the sample-interval counter, use the following programming sequence. All writes are 16-bit operations. All values given are hexadecimal.

a. Write FF03 to the Am9513A Command Register to select the Counter 3 Mode Register.

b. Write the mode value to the Am9513A Data Register to store the Counter 3 mode value. Use one of the following mode values:

8B25 – Selects 1-MHz clock
8C25 – Selects 100-kHz clock
8D25 – Selects 10-kHz clock
8E25 – Selects 1-kHz clock
8F25 – Selects 100-Hz clock
8525 – Selects signal at SOURCE5 input as clock (counts the rising edge of the signal, 6 MHz maximum)

c. Write FF0B to the Am9513A Command Register to select the Counter 3 Load Register.

d. Write 2 to the Am9513A Data Register to store the Counter 3 load value.

e. Write FF44 to the Am9513A Command Register to load Counter 3.

f. Write FFF3 to the Am9513A Command Register to step Counter 3 down to one.

g. Write the desired sample interval to the Am9513A Data Register to store the Counter 3 load value:

   - If the sample interval is between 2 and FFFF (65,535 decimal), write the sample interval to the Am9513A Data Register.
   - If the sample interval is 10000 (65,536 decimal), write 0 to the Am9513A Data Register.

h. Write FF24 to the Am9513A Command Register to arm Counter 3.

After you complete this programming sequence, Counter 3 is configured to generate A/D conversion pulses as soon as application of a trigger causes it to be enabled.

3. Program the sample counter.

Use Counters 4 and 5 of the Am9513A Counter/Timer as the sample counter. The sample counter tallies the number of A/D conversions initiated by Counter 3 and stops Counter 3 when the desired sample count is reached. The sample count should be programmed as a multiple of the number of entries in the mux-gain memory. If the desired sample count is 65,536 or less, only Counter 4 needs to be used, making Counter 5 available for general-purpose timing applications. If the desired sample count is greater than 65,536, both Counters 4 and 5 must be used.

Sample Counts 2 through 65,536. To program the sample counter for sample counts up to 65,536, use the following programming sequence. The minimum permitted sample count is two. All writes are 16-bit operations. All values given are hexadecimal.

a. Write FF04 to the Am9513A Command Register to select the Counter 4 Mode Register.

b. Write 1025 to the Am9513A Data Register to store the Counter 4 mode value.

c. Write FF0C to the Am9513A Command Register to select the Counter 4 Load Register.

d. Write the sample count value to the Am9513A Data Register to store the Counter 4 load value:

   - If the sample count is between 2 and FFFF (65,535 decimal), write the sample count minus 1 to the Am9513A Data Register.
   - If the sample count is 10000 (65,536 decimal), write 0 to the Am9513A Data Register.

e. Write FF48 to the Am9513A Command Register to load Counter 4.

f. Write FFF4 to the Am9513A Command Register to decrement Counter 4.

g. Write FF28 to the Am9513A Command Register to arm Counter 4.

h. Clear the 16*/32 CNT bit in Command Register 1 to notify the hardware that only Counter 4 will be used as the sample counter.

After you complete this programming sequence, Counter 4 is configured to count A/D conversion pulses generated by Counter 3 and turns off the data acquisition operation when Counter 4 reaches zero.

Sample Counts Greater Than 65,536. To program the sample counter for sample counts greater than 65,536, use the following programming sequence. The lower 16 bits of the sample count are stored in Counter 4, and the upper 16 bits of the sample count are stored in Counter 5. All writes are 16-bit operations. All values given are hexadecimal.

a. Write FF04 to the Am9513A Command Register to select the Counter 4 Mode Register.

b. Write 1025 to the Am9513A Data Register to store the Counter 4 mode value.

c. Write FF0C to the Am9513A Command Register to select the Counter 4 Load Register.

d. Write the least significant 16 bits of the sample count value minus 1 to the Am9513A Data Register to store the Counter 4 load value.

   - If the least significant 16 bits are all zeros, write FFFF.

e. Write FF48 to the Am9513A Command Register to load Counter 4.

f. Write 0 to the Am9513A Data Register to store 0 into the Load Register for Counter 4 reloading.

g. Write FF28 to the Am9513A Command Register to arm Counter 4.

h. Write FF05 to the Am9513A Command Register to select the Counter 5 Mode Register.

i. Write 25 to the Am9513A Data Register to store the Counter 5 mode value.

j. Write FF0D to the Am9513A Command Register to select the Counter 5 Load Register.

k. Take the most significant 16 bits of the sample count and do the following:

   - If the least significant 16 bits of the sample count are all zeros or all zeros except for a 1 in the least significant bit, write the most significant 16 bits to the Am9513A Data Register to store the Counter 5 load value.

   - Otherwise, add 1 to the most significant 16 bits of the sample count and write that value to the Am9513A Data Register to store the Counter 5 load value.

l. Write FF70 to the Am9513A Command Register to load and arm Counter 5.

m. Set the 16*/32 CNT bit in Command Register 1 to notify the hardware that both Counters 4 and 5 will be used as the sample counter.

After you complete this programming sequence, Counter 4 is configured to count A/D conversion pulses generated by Counter 3, and Counter 5 increments every time Counter 4 reaches zero. The data acquisition operation is terminated when both Counters 4 and 5 reach zero and the last entry in the mux-gain memory is served.

4. Program the scan-interval counter.

Use Counter 2 of the Am9513A Counter/Timer as the scan-interval counter. Counter 2 can be programmed to generate a pulse once every *N* counts. *N* is referred to as the scan interval, that is, the time between successive scan sequences programmed into the mux-gain memory. *N* can be between 2 and 65,536. One count is equal to the period of the timebase clock used by the counter. The following clocks are available internal to the Am9513A–1 MHz, 100 kHz, 10 kHz, 1 kHz, and 100 Hz. In addition, the scan-interval timer can use signals connected to any of the Am9513A SOURCE input pins.

To program the scan-interval counter, use the following programming sequence. All writes are 16-bit operations. All values given are hexadecimal.

a. Write FF02 to the Am9513A Command Register to select the Counter 2 Mode Register.

b. Write the mode value to the Am9513A Data Register to store the Counter 2 mode value. Use one of the following mode values:

   8B25 – Selects 1-MHz clock
   8C25 – Selects 100-kHz clock
   8D25 – Selects 10-kHz clock
   8E25 – Selects 1-kHz clock
   8F25 – Selects 100-Hz clock
   8525 – Selects signal at SOURCE5 input as clock (counts the rising edge of the signal, 6 MHz maximum)

c. Write FF0A to the Am9513A Command Register to select the Counter 2 Load Register.

d. Write 2 to the Am9513A Data Register to store the Counter 2 load value.

e. Write FF42 to the Am9513A Command Register to load Counter 2.

f. Write FFF2 to the Am9513A Command Register to step Counter 2 down to one.

g. Entries stored in the mux-gain memory should be scanned once during a scan interval. The following condition must be satisfied:

   scan interval ≥ sample interval ∗ *x*, where *x* is the number of entries in the scan sequence.

   Write the desired scan interval to the Am9513A Data Register to store the Counter 2 load value:

   - If the scan interval is between 2 and FFFF (65,535 decimal), write the scan interval to the Am9513A Data Register.
   - If the scan interval is 10000 (65,536 decimal), write 0 to the Am9513A Data Register.

h. Write FF22 to the Am9513A Command Register to arm Counter 2.

After you complete this programming sequence, Counter 2 is configured to assign a time interval to scan sequences once the trigger to enable A/D conversions is detected.

5. Clear the A/D circuitry and reset the multiplexer counter.

Before starting the data acquisition operation, the A/D FIFO must be emptied to clear out any old A/D conversion results. This emptying must be done after the counters are programmed in case any spurious edges were caused while programming the counters. Write 0 to the A/D Clear Register to empty the FIFO (16-bit write).

Write 0 to the Mux-Counter Register to set the analog input circuitry to the first channel and gain setting of the scan sequence.

Write 0 to the INT2CLR Register to clear any spurious edge caused by programming Counter 2.

6. Enable the scanning data acquisition operation.

To enable the scanning data acquisition operation such that A/D conversions begin when a trigger is received, set the DAQEN bit and the SCANEN bit in Command Register 1. To enable the scan interval timing, set the SCN2 bit in Command Register 2.

7. Apply a trigger.

Once set up by the preceding steps, the data acquisition operation is initiated when a trigger is received. A trigger can be provided in one of two ways–through software or through hardware.

To initiate the data acquisition operation through software, write 0 to the Start DAQ Register.

To initiate the data acquisition operation through hardware, apply an active low pulse to the START TRIG* pin on the AT-MIO-16D I/O connector. See the *Data Acquisition Timing Connections* section in Chapter 2, *Configuration and Installation*, for START TRIG* signal specifications.

Once the trigger is applied, Counter 3 generates pulses initiating A/D conversions once every sample interval until the sample counter reaches 0 and the last scan cycle is completed. Counter 2 generates a scan interval for each cycle through the scan sequence in the mux-gain memory.

8. Service the data acquisition operation.

Once the data acquisition operation is started by application of a trigger, the operation is serviced by reading the A/D FIFO Register every time an A/D conversion result becomes available. To do this, perform the following sequence until the desired number of conversion results have been read:

a. Read the Status Register (16-bit read).

b. If the CONVAVAIL bit is set (bit 13), read the A/D FIFO Register to obtain the result.

Interrupts or DMA can also be used to service the data acquisition operation. These topics are discussed later in this chapter.

Two error conditions may occur during a data acquisition operation–an overflow error or an overrun error. These error conditions are reported through the Status Register and should be

checked every time the Status Register is read to check the CONVAVAIL bit. If either of these error conditions occurs, the data acquisition operation stops.

An overflow condition occurs if more than 512 A/D conversions have been stored in the A/D FIFO without the A/D FIFO being read; that is, the A/D FIFO is full and cannot accept any more data. This condition occurs if the software loop reading the A/D FIFO Register is not fast enough to keep up with the A/D conversion rate. When an overflow occurs, at least one A/D conversion result is lost. An overflow condition has occurred if the OVERFLOW bit in the Status Register is set.

An overrun condition occurs if a second A/D conversion is initiated before the previous conversion is finished. This condition may result in one or more missing A/D conversions. This condition occurs if the sample interval is too small (sample rate is too high). An overrun condition has occurred if the OVERRUN bit in the Status Register is set.

Scanned data acquisition requires slower acquisition rates than single-channel data acquisition because signals must settle each time channels are switched. See Table 4-4 for the maximum recommended multiple-channel data acquisition rates. The rates in Table 4-4 refer to typical settling accuracies of 0.5 LSBs of the final value.

Table 4-4. Multiple-Channel Data Acquisition Rates

Gain	Data Acquisition Rate
1, 2, 4, 8	100 ksamples/sec
10	100 ksamples/sec
100	70 ksamples/sec
500	20 ksamples/sec

Both the OVERFLOW and OVERRUN bits in the Status Register are cleared by writing to the A/D Clear Register.

External Timing Considerations for Scanned Data Acquisition

After you follow the programming instructions listed previously under *External Timing Considerations for Multiple A/D Conversions*, complete these additional steps:

1. Set up the analog channel and gain sequence as given above.
2. Set the SCANEN bit in Command Register 1.
3. Set the multiplexer counter to 0 before starting the data acquisition operation.

Resetting the Hardware after a Data Acquisition Operation

After a data acquisition operation is complete, if no errors occurred and the sample count was less than or equal to 10000 hex, then the AT-MIO-16D is left in the same state as it was at the beginning of the data acquisition operation. The counters do not need to be reprogrammed; another data acquisition operation begins when a trigger is received. If the next data acquisition

operation requires the counters to be programmed differently, the Am9513A counters that were used must be disarmed and reset.

Resetting Counter 2

To reset Counter 2, use the following programming sequence. All writes are 16-bit operations. All values given are hexadecimal.

1. Write FFC2 to the Am9513A Command Register to disarm Counter 2.
2. Write FF02 to the Am9513A Command Register to select the Counter 2 Mode Register.
3. Write 4 to the Am9513A Data Register to store the Counter 2 mode value such that counter output becomes high-impedance.
4. Write FF0A to the Am9513A Command Register to select the Counter 2 Load Register.
5. Write 3 to the Am9513A Data Register to store nonterminal count value in the Counter 2 Load Register.
6. Write FF42 to the Am9513A Command Register to load Counter 2.
7. Write FF42 to the Am9513A Command Register a second time to load Counter 2 again to guarantee that Counter 2 is not left in a terminal count state.

Resetting Counter 3

To reset Counter 3, use the following programming sequence. All writes are 16-bit operations. All values given are hexadecimal.

1. Write FFC4 to the Am9513A Command Register to disarm Counter 3.
2. Write FF03 to the Am9513A Command Register to select the Counter 3 Mode Register.
3. Write 4 to the Am9513A Data Register to store the Counter 3 mode value such that counter output becomes high-impedance.
4. Write FF0B to the Am9513A Command Register to select the Counter 3 Load Register.
5. Write 3 to the Am9513A Data Register to store nonterminal count value in the Counter 3 Load Register.
6. Write FF44 to the Am9513A Command Register to load Counter 3.
7. Write FF44 to the Am9513A Command Register a second time to load Counter 3 again to guarantee that Counter 3 is not left in a terminal count state.

Resetting Counter 4

To reset Counter 4, use the following programming sequence. All writes are 16-bit operations. All values given are hexadecimal.

1. Write FFC8 to the Am9513A Command Register to disarm Counter 4.
2. Write FF04 to the Am9513A Command Register to select the Counter 4 Mode Register.
3. Write 4 to the Am9513A Data Register to store the Counter 4 mode value such that counter output becomes high-impedance. If Counter 4 is not to be used during the next data acquisition operation, write 0 to the Am9513A Data Register to drive the output low.
4. Write FF0C to the Am9513A Command Register to select the Counter 4 Load Register.
5. Write 3 to the Am9513A Data Register to store non-terminal count value in the Counter 4 Load Register.
6. Write FF48 to the Am9513A Command Register to load Counter 4.
7. Write FF48 to the Am9513A Command Register a second time to load Counter 4 again to guarantee that Counter 4 is not left in a terminal count state.

Resetting Counter 5

To reset Counter 5, use the following programming sequence. All writes are 16-bit operations. All values given are hexadecimal.

1. Write FFD0 to the Am9513A Command Register to disarm Counter 5.
2. Write FF05 to the Am9513A Command Register to select the Counter 5 Mode Register.
3. Write 4 to the Am9513A Data Register to store the Counter 5 mode value such that counter output becomes high-impedance.
4. Write FF0D to the Am9513A Command Register to select the Counter 5 Load Register.
5. Write 3 to the Am9513A Data Register to store non-terminal count value in the Counter 5 Load Register.
6. Write FF50 to the Am9513A Command Register to load Counter 5.
7. Write FF50 to the Am9513A Command Register a second time to load Counter 5 again to guarantee that Counter 5 is not left in a terminal count state.

After resetting the counters, write 0 to the A/D Clear Register to clear all error conditions and to empty the A/D FIFO.

Programming the Analog Output Circuitry

The voltage at the analog output circuitry output pins (pins DAC0 OUT and DAC1 OUT on the AT-MIO-16D MIO-16 I/O connector) is controlled by loading the DAC in the analog output channel with a 12-bit digital code. This DAC is loaded by writing the digital code to the DAC0 and DAC1 Registers. Writing to the DAC0 Register controls the voltage at the DAC0 OUT pin on the MIO-16 I/O connector. Writing to the DAC1 Register controls the voltage at the DAC1 OUT pin. The analog output on pins DAC0 OUT and DAC1 OUT can be updated in one of two ways: immediately when DAC0 or DAC1 is written, or when an active low pulse is detected on the OUT2 pin of the Am9513A Counter/Timer. The LDAC bit in Command Register 2 selects which update method is used.

The output voltage generated from the digital code depends on the configuration, unipolar or bipolar, of the associated analog output channel. This configuration is determined by configuration jumpers on the AT-MIO-16D board. In bipolar mode, configuration jumpers also determine if the digital code written to the DACs is in straight binary form or in a two's complement form. The factory default is the bipolar configuration in two's complement mode. See the *Analog Input Configuration* section in Chapter 2, *Configuration and Installation*, for more information. Table 4-5 shows the output voltage versus digital code for a unipolar analog output configuration. Table 4-6 shows the voltage versus digital code for a bipolar analog output configuration.

The formula for the voltage output versus digital code for a unipolar analog output configuration is as follows:

$$V_{out} = V_{ref} * \frac{\text{(digital code)}}{4{,}096}$$

where V_{ref} is the reference voltage applied to the analog output channel. The digital code in the above formula is a decimal value ranging from 0 to 4,095.

Table 4-5. Analog Output Voltage Versus Digital Code (Unipolar Mode)

Digital Code		**Voltage Output**	
Decimal	**Hex**	$\mathbf{V_{ref = 10\ V}}$	$\mathbf{V_{out}}$
0	0	0	0 V
1	1	$\frac{Vref}{4{,}096}$	2.44 mV
1,024	0400	$\frac{Vref}{4}$	2.5 V
2,048	0800	$\frac{Vref}{2}$	5 V
3,072	0C00	$\frac{Vref * 3}{4}$	7.5 V
4,095	0FFF	$\frac{Vref * 4{,}095}{4{,}096}$	9.9976 V

The formula for the voltage output versus digital code for a bipolar analog output configuration in straight binary form is as follows:

$$V_{out} = V_{ref} * \frac{(\text{digital code} - 2{,}048)}{2{,}048}$$

where V_{ref} is the reference voltage applied to the analog output channel. The digital code in the above formula is a decimal value ranging from 0 to 4,095.

The formula for the voltage output versus digital code for a bipolar analog output configuration in two's complement form is as follows:

$$V_{out} = V_{ref} * \frac{(\text{digital code})}{2{,}048}$$

where V_{ref} is the positive reference voltage applied to the analog output channel. The digital code in the above formula is a decimal value ranging from -2,048 to +2,047.

Table 4-6. Analog Output Voltage Versus Digital Code (Bipolar Mode)

Digital Code				**Voltage Output**	
Straight Binary		**Two's Complement**			
Decimal	**Hex**	**Decimal**	**Hex**	**V_{ref} = 10 V**	**V_{out}**
0	0	-2,048	F800	0	-10 V
1	1	-2,047	F801	$\frac{Vref * (-2{,}047)}{2{,}048}$	-9.9951 V
1,024	0400	-1,024	FC00	$\frac{-Vref}{2}$	-5 V
2,047	07FF	-1	FFFF	$\frac{-Vref}{2{,}048}$	-2.44 mV
2,048	0800	0	0	0	0 V
2,049	0801	1	1	$\frac{Vref}{2{,}048}$	2.44 mV
3,072	0C00	1,024	0400	$\frac{Vref}{2}$	5 V
4,095	0FFF	2,047	07FF	$\frac{Vref * 2{,}047}{2{,}048}$	9.9951 V

Programming the MIO-16 Digital I/O Circuitry

The digital input circuitry is controlled and monitored using the MIO-16 Digital Input Register, the MIO-16 Digital Output Register, and the two bits DOUT0EN and DOUT1EN in Command Register 2. See the register bit descriptions earlier in this chapter for more information.

To enable digital output port 0, set the DOUT0EN bit in Command Register 2. To enable digital output port 1, set the DOUT1EN bit in Command Register 2. When a digital output port is enabled, the contents of the MIO-16 Digital Output Register are driven onto the digital lines corresponding to that port. The digital output for both ports 0 and 1 are updated by writing the desired pattern to the MIO-16 Digital Output Register.

The input ports must be enabled for an external device to drive the MIO-16 digital I/O lines. Clear the DOUT0EN bit in Command Register 2 if an external device is driving digital I/O lines ADIO<3..0>. Clear the DOUT1EN bit in Command Register 2 if an external device is driving digital I/O lines BDIO<3..0>. The MIO-16 Digital Input Register can then be read to monitor the state of the digital I/O lines as driven by the external device.

The logic state of all eight MIO-16 digital I/O lines can be read from the MIO-16 Digital Input Register. If the digital output ports are enabled, the MIO-16 Digital Input Register serves as a read-back register; that is, you can determine how the AT-MIO-16D is driving the digital I/O lines by reading the MIO-16 Digital Input Register.

Programming the Am9513A Counter/Timer

Counters 1, 2, and 5 of the Am9513A Counter/Timer are available for general-purpose timing applications. The programmable frequency output pin FOUT is also available as a timing signal source. These applications and a general description of the Am9513A Counter/Timer are included in the *Data Acquisition Timing Connections* section in Chapter 2, *Configuration and Installation*. The *Timing I/O Circuitry* section in Chapter 3, *Theory of Operation*, explains how the Am9513A is used on the AT-MIO-16D board.

Initialization of the Am9513A as required by the AT-MIO-16D and specific programming requirements for the sample-interval and sample counters are given earlier in this chapter. For general programming details for Counters 1, 2 and 5, and the programmable frequency output, refer to Appendix E, *Am9513A Data Sheet*.

Write-and-read operations to the Am9513A registers require a minimum 1.5-µsec recovery time between operations. If two operations to the Am9513A occur within 1.5 µsec, the second operation is ignored by the AT-MIO-16D. Caution should be taken when writing to the Am9513A registers so that this access recovery time is not violated.

In programming the Master Mode Register, keep the following considerations in mind:

- The Am9513A must be used in 16-bit bus mode.
- The scalar control should be set to BCD division for correct operation of the clocks as described under the *Programming Multiple A/D Programming Conversions on a Single Input Channel* section earlier in this chapter.

RTSI Bus Trigger Line Programming Considerations

The RTSI switch connects signals on the AT-MIO-16D to the seven RTSI bus trigger lines. The RTSI switch has seven pins labeled A<6..0> connected to AT-MIO-16D signals and seven pins labeled B<6..0> connected to the seven RTSI bus trigger lines. Table 4-7 shows the signals connected to each pin.

Table 4-7. RTSI Switch Signal Connections

RTSI Switch Pin	Signal Name	Signal Direction
A Side:		
A0	EXTCONV*	Bidirectional
A1	FOUT	Output
A2	OUT2	Output
A2	GATE1	Input
A3	SOURCE5	Bidirectional
A4	OUT5	Output
A4	STOP TRIG	Input
A5	OUT1	Output
A6	START TRIG*	Bidirectional
B Side:		
B0	TRIGGER0	Bidirectional
B1	TRIGGER1	Bidirectional
B2	TRIGGER2	Bidirectional
B3	TRIGGER3	Bidirectional
B4	TRIGGER4	Bidirectional
B5	TRIGGER5	Bidirectional
B6	TRIGGER6	Bidirectional

Figure 3-8 in Chapter 3, *Theory of Operation*, diagrams the AT-MIO-16D RTSI switch connections.

AT-MIO-16D RTSI Signal Connection Considerations

The AT-MIO-16D board has a total of nine signals connected to the seven A-side pins of the RTSI switch. These same signals also appear at the AT-MIO-16D I/O connector. As shown in Table 4-8, two AT-MIO-16D signals are connected to pin A2, and two signals are connected to pin A4. The routing of these signals is further controlled by the bits A4DRV, A4RCV, A2DRV, and A2RCV in Command Register 2.

- To drive the RTSI switch pin A2 with the signal OUT2, set the A2DRV bit in Command Register 2. Otherwise, clear the A2DRV bit.
- To drive the signal GATE1 from pin A2 of the RTSI switch, set the A2RCV bit in Command Register 2. Otherwise, clear the A2RCV bit.

Note: If both the A2DRV and A2RCV bits are set, the GATE1 signal is driven by the signal OUT2. This arrangement is probably not desirable.

- To drive the RTSI switch pin A4 with the signal OUT5, set the A4DRV bit in Command Register 2. Otherwise, clear the A4DRV bit.
- To drive the signal STOP TRIG from pin A4 of the RTSI switch, set the A4RCV bit in Command Register 2. Otherwise, clear the A4RCV bit.

Note: If both the A4DRV and A4RCV bits are set, the STOP TRIG signal is driven by the signal OUT5. This arrangement is probably not desirable.

Programming the RTSI Switch

The RTSI switch can be programmed to connect any of the signals on the A side to any of the signals on the B side and vice versa. To do this, a 56-bit pattern is shifted into the RTSI switch by writing one bit at a time to the RTSI Switch Shift Register and then writing to the RTSI Switch Strobe Register to load the pattern into the RTSI switch.

The 56-bit pattern is made up of two 28-bit patterns, one for side A and one for side B of the RTSI switch. The low-order 28 bits select the signal sources for the B-side pins. The high-order 28 bits select the signal sources for the A-side pins. Each of the 28-bit patterns are made up of seven 4-bit fields, one for each pin. The 4-bit field selects the signal source and the output enable for the pin. Figure 4-1 shows the bit map of the RTSI switch 56-bit pattern.

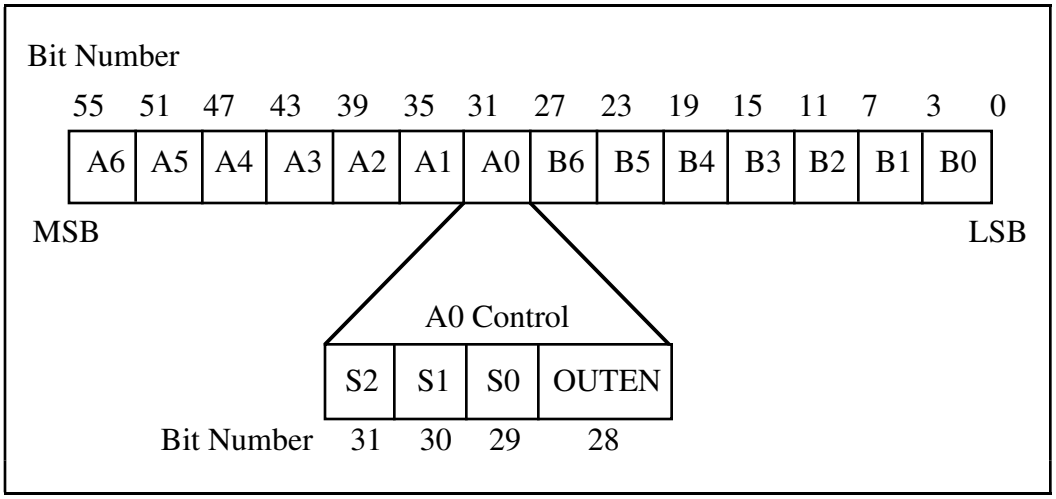


Figure 4-1. RTSI Switch Control Pattern

In Figure 4-1, the fields labeled A6 through A0 and B6 through B0 are the 4-bit control fields for each RTSI switch pin of the same name. The 4-bit control field for pin A0 is shown in Figure 4—1.

The bits labeled S2 through S0 are the signal source selection bits for the pin. One of seven source signals can be selected. Pins A6 through A0 can select any of the pins B6 through B0 as signal sources. Pins B6 through B0 select any of the pins A6 through A0 as signal sources. For example, the pattern 011 for S2 through S0 in the A0 control field selects the signal connected to pin B3 as the signal source for pin A0.

The bit labeled OUTEN is the output enable bit for that pin. If the OUTEN bit is set, the pin is driven by the selected source signal (the pin acts as an output pin). If the OUTEN bit is cleared, the pin is not driven regardless of the source signal selected; instead, the pin can be used as an input pin.

If the A0 control field above contains the pattern 0111, the signal connected to pin B3 (Trigger Line 3) appears at pin A0. On the AT-MIO-16D board, this arrangement allows the EXTCONV* signal to be driven by Trigger Line 3. Conversely, if the B4 control field contains the pattern 1011, the signal connected to pin A5 appears at pin B4. This arrangement allows Trigger Line 4 to be driven by the AT-MIO-16D OUT1 signal. In this way, boards connected via the RTSI bus can send signals to each other over the RTSI bus trigger lines.

To program the RTSI switch, complete these steps:

1. Calculate the 56-bit pattern based on the desired signal routing.
   a. Clear the OUTEN bit for all input pins and for all unused pins.
   b. Specify the signal source pin for all output pins by setting bits S2 through S0 to the source pin number.
   c. Set the OUTEN bit for all output pins.
2. For $i = 0$ to 55, do the following.
   a. Copy bit i of the 56-bit pattern to bit 0 of an 8-bit temporary variable.
   b. Write the temporary variable to the RTSI Switch Shift Register (8-bit write).
3. Write 0 to the RTSI Switch Strobe Register (8-bit write). This operation loads the 56-bit pattern into the RTSI switch. At this point, the new signal routing goes into effect.

Step 2 above can be completed by simply writing the low-order eight bits of the 56-bit pattern to the RTSI Switch Shift Register, then shifting the 56-bit pattern right once, and repeating this two-step operation a total of 56 times. Only bit 0 of the word written to the RTSI Switch Shift Register is used. The higher-order bits are ignored.

Programming DMA Operations

The AT-MIO-16D can be programmed so that the A/D FIFO generates a DMA request signal every time one or more A/D conversion values are stored in the A/D FIFO. There are two DMA modes–single-channel transfer and dual-channel transfer. In single-channel mode, one DMA channel is used. The DMA channel is selected by the onboard jumper. To program the single-channel DMA operation, perform the following steps after the circuitry on the AT-MIO-16D is set up for a data acquisition operation and before the data acquisition operation begins:

1. Set the DMAEN bit in Command Register 1 to enable DMA request generation.
2. Program the DMA controller to service DMA requests from the AT-MIO-16D board. Refer to the *IBM Personal Computer AT Technical Reference* manual for more information on DMA controller programming.
3. If a DMA terminal count is received after the DMA service, write 0 to either the DMATC Clear Register or the A/D Clear Register to clear the DMATC bit in the Status Register.

Once steps 1 and 2 are completed, the DMA controller automatically reads the A/D FIFO Register whenever an A/D conversion result is available and then stores the result in a buffer in memory.

To program the dual-channel DMA operation, perform the following steps:

1. Set the DMAEN and DBDMA bits in Command Register 1.
2. Write 0 to either the DMATC Clear Register or the A/D Clear Register.

3. Program the DMA controller to set up two DMA channels and two memory buffers for each DMA channel data collection.

4. After the DMA service, write 0 to either the DMATC Clear Register or the A/D Clear Register.

During the DMA operation, DMA Channel 1 and Memory Buffer 1 (DMA 1) are served first. When a DMA terminal count is received, the board automatically switches the DMA operation to DMA Channel 2 and Memory Buffer 2 (DMA 2). Therefore, the board can collect data into one buffer and service data in another buffer simultaneously. If the DMA controller is programmed for auto-reinitialize mode, DMA 1 and DMA 2 are continuously served in turn.

Interrupt Programming

Four different interrupts are generated by the AT-MIO-16D board:

- An interrupt whenever a conversion is available to be read from the A/D FIFO
- An interrupt whenever a DMA terminal count is received
- An interrupt whenever a data acquisition operation is completed (either normally or due to an error condition)
- An interrupt whenever a rising edge on the OUT2 pin of the Am9513A is detected.

These four interrupts are enabled individually. To use any one of these interrupts, the overall interrupt enable bit INTEN in Command Register 2 must be set.

To use the conversion interrupt, set the CONVINTEN bit in Command Register 1 and the INTEN bit in Command Register 2. If these bits are set, if an interrupt occurs from the AT-MIO-16D board, and if the CONVAVAIL bit in the Status Register is set, then a conversion interrupt has occurred. Reading from the A/D FIFO Register clears this interrupt condition. Writing to the A/D Clear Register also clears the conversion interrupt.

To use the DMA terminal count interrupt, set the DMAEN and TCINTEN bits in Command Register 1 and the INTEN bit in Command Register 2. If these bits are set, if an interrupt occurs from the AT-MIO-16D board, and if the DMATC bit in the Status Register is set, then a DMA terminal count interrupt has occurred. Writing to the DMA TC INT Clear Register or to the A/D Clear Register clears this interrupt condition.

To use the data acquisition completion interrupt, set the DAQSTOPINTEN bit in Command Register 1 and the INTEN bit in Command Register 2. If these bits are set, if an interrupt occurs from the AT-MIO-16D board, and if the DAQSTOPINT bit in the Status Register is set, then a data acquisition completion interrupt has occurred. If either the OVERFLOW or the OVERRUN bit is set in the Status Register, this interrupt is the result of a data acquisition termination error condition. Otherwise, the normal data acquisition completion interrupt has occurred. Writing to the A/D Clear Register clears the data acquisition completion interrupt and the error condition if any are set.

To use the OUT2 interrupt, set the INTEN and the INT2EN bits in Command Register 2. If these bits are set, if a rising edge occurs on OUT2, and if the OUT2INT bit in the Status Register is set, then an OUT2 interrupt has occurred. Writing to the INT2CLR Register clears the OUT2

interrupt. This interrupt is helpful when using the DACs to implement a waveform generator. This interrupt can also be used to interrupt on an external signal connected to the OUT2 pin. If OUT2 is connected to an external signal, make certain that Counter 2 is reset as described under the *Resetting the Hardware after a Data Acquisition Operation* section earlier in this chapter. If Counter 2 is not reset, it may drive the OUT2 line, preventing the external signal from successfully driving the line high or low.

DIO-24 Circuitry Programming Considerations

The DIO-24 circuitry is designed around the 82C55A integrated circuit. The 82C55A is a general-purpose peripheral interface containing 24 programmable I/O pins. These pins represent the three 8-bit I/O ports (A, B, and C) of the 82C55A. These ports can be programmed as two groups of 12 signals or as three individual 8-bit ports. This section includes programming information for the DIO-24 circuitry, along with program examples written in C.

The three 8-bit ports of the 82C55A are divided into two groups–Group A and Group B (two groups of 12 signals). One 8-bit configuration (or control) word determines the mode of operation for each group. The Group A control bits configure Port A (A0 through A7) and the upper 4 bits (nibble) of Port C (C4 through C7). The Group B control bits configure Port B (B0 through B7) and the lower nibble of Port C (C0 through C3). These configuration bits are defined later in this chapter.

DIO-24 Circuitry Register Descriptions

Figure 4-2 shows the two control-word formats used to completely program the 82C55A. The Control Word Flag determines which control-word format is being programmed. When the Control Word Flag is 1, bits 0 through 6 determine the I/O characteristics of the 82C55A ports and the mode in which they are operating (that is, Mode 0, Mode 1, or Mode 2). When the Control Word Flag is 0, bits 3 through 0 determine the bit set/reset format of Port C.

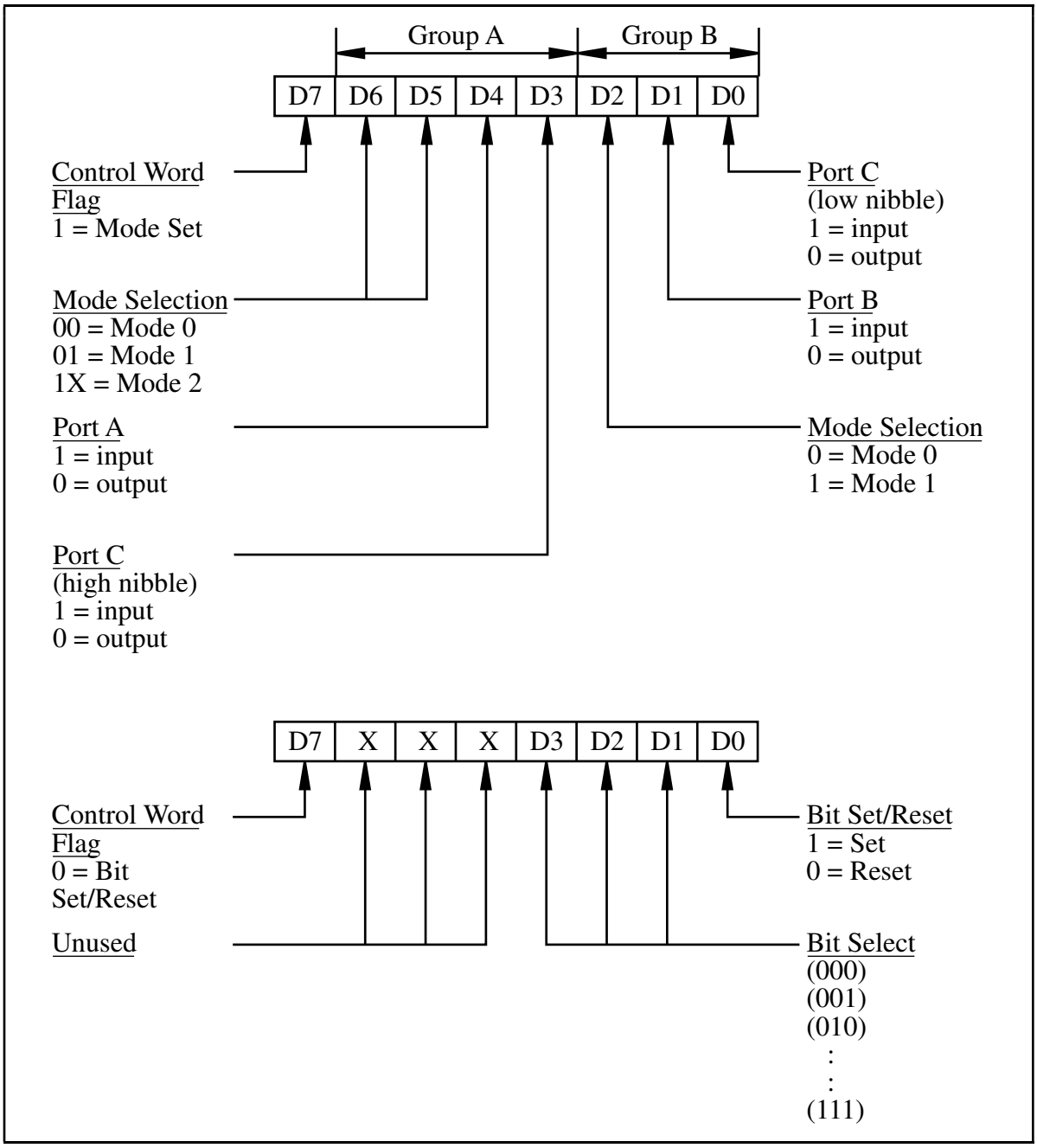


Figure 4-2. Control-Word Formats

Table 4-8 shows the control words for setting or resetting each bit in Port C. Notice that bit 7 of the control word is cleared when programming the set/reset option for the bits of Port C.

Table 4-8. Port C Set/Reset Control Words

Number	Bit Set Control Word	Bit Reset Control Word	Bit Set or Reset in Port C
0	0xxx0001	0xxx0000	xxxxxxxn
1	0xxx0011	0xxx0010	xxxxxxnx
2	0xxx0101	0xxx0100	xxxxxnxx
3	0xxx0111	0xxx0110	xxxxnxxx
4	0xxx1001	0xxx1000	xxxnxxxx
5	0xxx1011	0xxx1010	xxnxxxxx
6	0xxx1101	0xxx1100	xnxxxxxx
7	0xxx1111	0xxx1110	nxxxxxxx

82C55A Modes of Operation

The three basic modes of operation for the 82C55A are as follows:

- Mode 0 – Basic I/O
- Mode 1 – Strobed I/O
- Mode 2 – Bidirectional bus

The 82C55A also has a single bit set/reset feature for Port C. The 8-bit control word also programs this function. For additional information, refer to Appendix F, *Oki MSM82C55A Data Sheet*.

Mode 0–Basic I/O

Mode 0 can be used for simple input and output operations for each of the ports. No handshaking is required; data is simply written to or read from a selected port.

Mode 0 has the following features:

- Two 8-bit ports (A and B) and two 4-bit ports (upper and lower nibble of Port C).
- Any port can be input or output.
- Outputs are latched, but inputs are not latched.

The 16 possible Mode 0 I/O configurations are shown in Table 4-9. Notice that bit 7 of the control word is set when programming the mode of operation for each port.

Table 4-9. Mode 0 I/O Configurations

Control Word		Group A		Group B	
Number	**Bit 76543210**	**Port A**	**Port C[1]**	**Port B**	**Port C[2]**
0	10000000	Output	Output	Output	Output
1	10000001	Output	Output	Output	Input
2	10000010	Output	Output	Input	Output
3	10000011	Output	Output	Input	Input
4	10001000	Output	Input	Output	Output
5	10001001	Output	Input	Output	Input
6	10001010	Output	Input	Input	Output
7	10001011	Output	Input	Input	Input
8	10010000	Input	Output	Output	Output
9	10010001	Input	Output	Output	Input
10	10010010	Input	Output	Input	Output
11	10010011	Input	Output	Input	Input
12	10011000	Input	Input	Output	Output
13	10011001	Input	Input	Output	Input
14	10011010	Input	Input	Input	Output
15	10011011	Input	Input	Input	Input

1 – Upper nibble of Port C
2 – Lower nibble of Port C

Mode 0 Programming Example

```
Main() {

#define BASE_ADDRESS       0x220     /* Board located at address 220 */
#define PORTAoffset        0x00      /* Offset for Port A */
#define PORTBoffset        0x01      /* Offset for Port B */
#define PORTCoffset        0x02      /* Offset for Port C */
#define CNFGoffset         0x03      /* Offset for CNFG   */

register unsigned int porta, portb, portc, cnfg;
char valread;                        /* Variable to store data read from a
                                        port */

/*  Calculate register addresses. */
porta = BASE_ADDRESS + PORTAoffset;
portb = BASE_ADDRESS + PORTBoffset;
portc = BASE_ADDRESS + PORTCoffset;
cnfg  = BASE_ADDRESS + CNFGoffset;
```

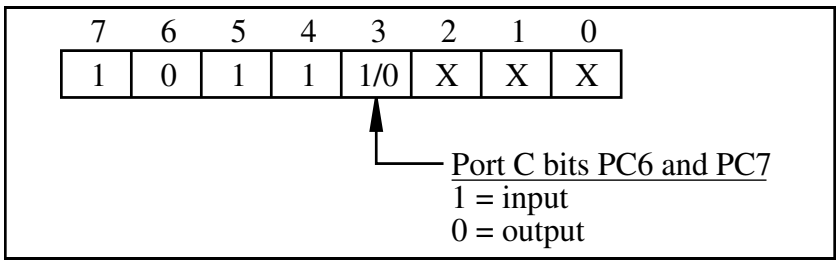
```
/* EXAMPLE 1*/
outp(cnfg,0x80);                    /* Ports A, B, and C are outputs. */
outp(porta,0x12);                   /* Write data to Port A. */
outp(portb,0x34);                   /* Write data to Port B. */
outp(portc,0x56);                   /* Write data to Port C. */

/* EXAMPLE 2*/
outp(cnfg,0x90);                    /* Port A is input; Ports B and C are
                                       outputs. */
outp(portb,0x22);                   /* Write data to Port B. */
outp(portc,0x55);                   /* Write data to Port C. */
valread = inp(porta);               /* Read data from Port A. */

/* EXAMPLE 3 */
outp(cnfg,0x82);                    /* Ports A and C are outputs; Port B
                                       is an input. */

/* EXAMPLE 4 */
outp(cnfg,0x89);                    /* Ports A and B are outputs; Port C
                                       is an input. */

}
```

Mode 1–Strobed Input

Mode 1 transfers data that is synchronized by handshaking signals. Ports A and B use the eight lines of Port C to generate or receive the handshake signals. This mode divides the ports into two groups (Group A and Group B):

- Each group contains one 8-bit data port (Port A or Port B) and one 4-bit control/data port (upper or lower nibble of Port C).
- The 8-bit data ports can be either input or output, both of which are latched.
- The 4-bit ports are used for control and status of the 8-bit data ports.
- The transfer of data is synchronized by handshaking signals in the 4-bit port.
- Interrupt generation and enable and/or disable functions are available.

The control word written to the DIO-24 CNFG Register to configure Port A for input in Mode 1 is shown as follows. Bits PC6 and PC7 of Port C can be used as extra input or output lines.



The control word written to the DIO-24 CNFG Register to configure Port B for input in Mode 1 is shown as follows. Notice that Port B does not have extra input or output lines from Port C.

7	6	5	4	3	2	1	0
1	X	X	X	X	1	1	X

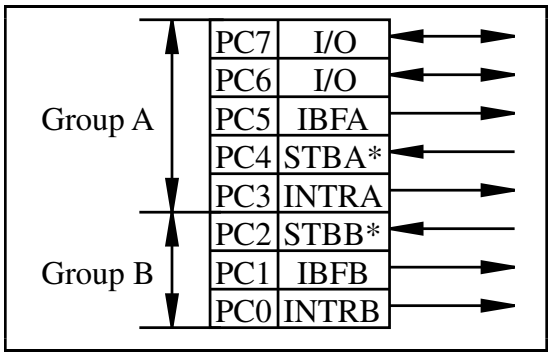
During a Mode 1 data read transfer, the status of the handshaking lines and interrupt signals can be obtained by reading Port C. The Port C status-word bit definitions for an input transfer are shown as follows.

The following are the Port C status-word bit definitions for input (Port A and Port B):

7	6	5	4	3	2	1	0
I/O	I/O	IBFA	INTEA	INTRA	INTEB	IBFB	INTRB

Bit	**Name**	**Description**
7-6	I/O	Extra I/O status lines when Port A is in Mode 1 input.
5	IBFA	Input buffer full for Port A. High indicates that data has been loaded into the input latch for Port A.
4	INTEA	Interrupt enable bit for Port A. Enables DIO interrupts from the 82C55A for Port A. Controlled by bit set/reset of PC4.
3	INTRA	Interrupt request status for Port A. When INTEA is high and IBFA is high, this bit is high, indicating that a DIO interrupt request is asserted.
2	INTEB	Interrupt enable bit for Port B. Enables DIO interrupts from the 82C55A for Port B. Controlled by bit set/reset of PC2.
1	IBFB	Input buffer full for Port B. High indicates that data has been loaded into the input latch for Port B.
0	INTRB	Interrupt request status for Port B. When INTEB is high and IBFB is high, this bit is high, indicating that a DIO interrupt request is asserted.

At the digital I/O connector, Port C has the following pin assignments when in Mode 1 input. Notice that the status of STBA* and STBB* are not included in the Port C status word.

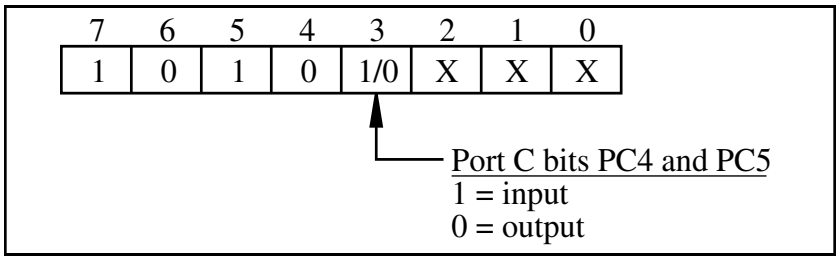


Mode 1 Input Programming Example

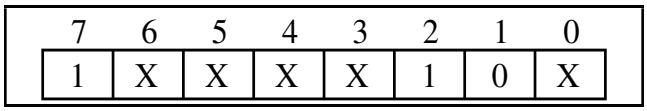
```
Main() {

#define BASE_ADDRESS       0x210     /* Board located at address 210. */
#define PORTAoffset        0x00      /* Offset for Port A */
#define PORTBoffset        0x01      /* Offset for Port B */
#define PORTCoffset        0x02      /* Offset for Port C */
#define CNFGoffset         0x03      /* Offset for CNFG   */

register unsigned int porta, portb, portc, cnfg;
char valread;                         /* Variable to store data read from a
                                         port */

/*  Calculate register addresses. */

porta = BASE_ADDRESS + PORTAoffset;
portb = BASE_ADDRESS + PORTBoffset;
portc = BASE_ADDRESS + PORTCoffset;
cnfg  = BASE_ADDRESS + CNFGoffset;

/* EXAMPLE 1-Port A input */

outp(cnfg,0xB0);                    /* Port A is an input in Mode 1. */
while (!(inp(portc) & 0x20));       /* Wait until IBFA is set, indicating that
                                       data has been loaded in Port A. */
valread = inp(porta);               /* Read the data from Port A. */

/* EXAMPLE 2-Port B input */

outp(cnfg,0x86);                    /* Port B is an input in Mode 1. */
while (!(inp(portc) & 0x02));       /* Wait until IBFB is set, indicating that
                                       data has been loaded in Port B. */
valread = inp(portb);
}
```

Mode 1–Strobed Output

The control word written to the DIO-24 CNFG Register to configure Port A for output in Mode 1 is shown as follows. Bits PC4 and PC5 of Port C can be used as extra input or output lines when Port A uses Mode 1 output.



The control word written to the DIO-24 CNFG Register to configure Port B for output in Mode 1 is shown as follows. Notice that Port B does not have extra input or output lines from Port C.



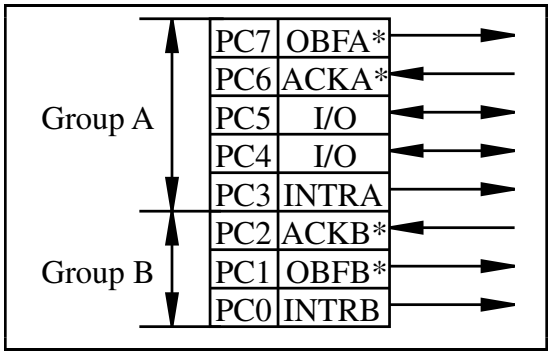
During a Mode 1 data write transfer, the status of the handshaking lines and interrupt signals can be obtained by reading Port C. Notice that the bit definitions are different for a write and a read transfer.

The following are the Port C status-word bit definitions for output (Port A and Port B):

7	6	5	4	3	2	1	0
OBFA*	INTEA	I/O	I/O	INTRA	INTEB	OBFB*	INTRB

Bit	Name	Description
7	OBFA*	Output buffer full for Port A. Low indicates that the CPU has written data to Port A.
6	INTEA	Interrupt enable bit for Port A. If this bit is high, DIO interrupts are enabled from the 82C55A for Port A. Controlled by bit set/reset of PC6.
5, 4	I/O	Extra I/O status line when Port A is in Mode 1 output.
3	INTRA	Interrupt request status for Port A. When INTEA is high and OBFA* is high, this bit is high, indicating that a DIO interrupt request is asserted.
2	INTEB	Interrupt enable bit for Port B. If this bit is high, interrupts are enabled from the 82C55A for Port B. Controlled by bit set/reset of PC2.
1	OBFB*	Output buffer full for Port B. Low indicates that the CPU has written data out to Port B.
0	INTRB	Interrupt request status for Port B. When INTEB is high and OBFB* is high, this bit is high, indicating that a DIO interrupt request is asserted.

At the digital I/O connector, Port C has the following pin assignments when in Mode 1 output. Notice that the status of ACKA* and ACKB* is not included when Port C is read.



Mode 1 Output Programming Example

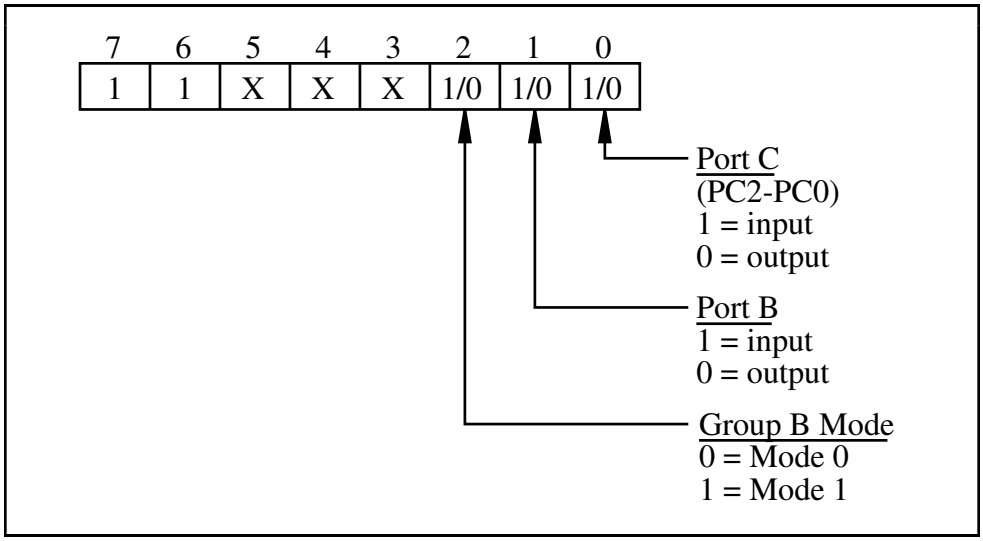
```
Main() {

#define BASE_ADDRESS       0x210      /* Board located at address 210. */
#define PORTAoffset        0x00       /* Offset for Port A */
#define PORTBoffset        0x01       /* Offset for Port B */
#define PORTCoffset        0x02       /* Offset for Port C */
#define CNFGoffset         0x03       /* Offset for CNFG   */

register unsigned int porta, portb, portc, cnfg;
char valread;                          /* Variable to store data read from a
                                          port */

/*  Calculate register addresses. */
porta = BASE_ADDRESS + PORTAoffset;
portb = BASE_ADDRESS + PORTBoffset;
portc = BASE_ADDRESS + PORTCoffset;
cnfg  = BASE_ADDRESS + CNFGoffset;

/* EXAMPLE 1-Port A output */

outp(cnfg,0xA0);                     /* Port A is an output in Mode 1.*/
while (!(inp(portc) & 0x80));        /* Wait until OBFA* is set, indicating
                                        that the data last written to Port A
                                        has been read.*/
outp(porta,0x12);                    /* Write data to Port A. */

/* EXAMPLE 2-Port B output */

outp(cnfg,0x84);                     /* Port B is an output in Mode 1.*/
while (!(inp(portc) & 0x02));        /* Wait until OBFB* is set, indicating
                                        that the data last written to Port B
                                        has been read.*/
outp(portb,0x34);                    /* Write the data to Port B. */
}
```

Mode 2–Bidirectional Bus

Mode 2 has a bidirectional 8-bit bus that can transfer both input and output without changing the configuration. The data transfers are synchronized with handshaking lines in Port C in a manner similar to that of Mode 1. This mode uses only Port A; however, Port B can be used in either Mode 0 or Mode 1 while Port A is configured for Mode 2. Interrupt generation and enable and/or disable functions are also available. Other features of this mode include the following:

- Used in Group A only (Port A and upper nibble of Port C).
- One 8-bit bidirectional port (Port A) and a 5-bit control status port (Port C).
- Latched inputs and outputs.

The control word written to the DIO-24 CNFG Register to configure Port A as a bidirectional data bus in Mode 2 is shown as follows. If Port B is configured for Mode 0, then PC2, PC1, and PC0 of Port C can be used as extra input or output lines.



During a Mode 2 data transfer, the status of the handshaking lines and interrupt signals can be obtained by reading Port C. The Port C status-word bit definitions for a Mode 2 transfer are shown as follows.

The following are the Port C status-word bit definitions for bidirectional data path (Port A only):

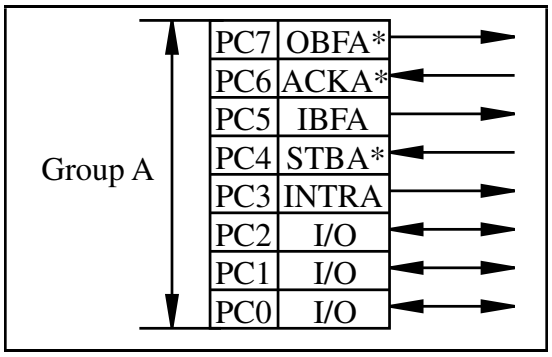
7	6	5	4	3	2	1	0
OBFA*	INTE1	IBFA	INTE2	INTRA	I/O	I/O	I/O

Bit	Name	Description
7	OBFA*	Output buffer full. Low indicates that the CPU has written data to Port A.

(continues)

Bit	Name	Description (continued)
6	INTE1	Interrupt enable bit for output. If this bit is set, DIO interrupts are enabled from the 82C55A for OBFA*. Controlled by bit set/reset of PC6.
5	IBFA	Input buffer full. High indicates that data has been loaded into the input latch of Port A.
4	INTE2	Interrupt enable bit for input. If this bit is set, DIO interrupts are enabled from the 82C55A for IBFA. Controlled by bit set/reset of PC4.
3	INTRA	Interrupt request status. If INTE1 is high and IBFA is high, this bit is high, indicating that a DIO interrupt request is asserted for input transfers. If INTE2 is high and OBFA* is high, this bit is high, indicating that a DIO interrupt request is asserted for output transfers.
2, 1, 0	I/O	Extra I/O status lines available if Port B is not configured for Mode 1.

At the DIO-24 I/O connector, Port C has the following pin assignments when in Mode 2.



Mode 2 Programming Example

```
Main() {

#define BASE_ADDRESS       0x210     /* Board located at address 210. */
#define PORTAoffset        0x00      /* Offset for Port A */
#define PORTBoffset        0x01      /* Offset for Port B */
#define PORTCoffset        0x02      /* Offset for Port C */
#define CNFGoffset         0x03      /* Offset for CNFG */

register unsigned int porta, portb, portc, cnfg;
char valread;                        /* Variable to store data read from a
                                        port */

/*  Calculate register addresses. */
porta = BASE_ADDRESS + PORTAoffset;
portb = BASE_ADDRESS + PORTBoffset;
portc = BASE_ADDRESS + PORTCoffset;
cnfg  = BASE_ADDRESS + CNFGoffset;
```

```
/* EXAMPLE 1*/

outp(cnfg,0xC0);                        /* Port A is in Mode 2. */
while (!(inp(portc) & 0x80));           /* Wait until OBFA* is set,
                                           indicating that the data last
                                           written to Port A has been
                                           read.*/
outp(porta,0x67);                       /* Write the data to Port A. */
while (!(inp(portc) & 0x20));           /* Wait until IBFA is set, indicating
                                           that data is available in Port A to
                                           be read. */
valread = inp(porta);                   /* Read data from Port A. */
}
```

Single Bit Set/Reset Feature

You can set or reset any of the eight bits of Port C with one control word. This feature generates status and control for Port A and Port B when operating in Mode 1 or Mode 2.

Interrupt Programming Examples

The following examples show the process required to enable interrupts for several different operating modes. The interrupt handling routines and interrupt installation routines are not included. See the *IBM Personal Computer AT Technical Reference* manual for additional information.

```
Main() {

#define BASE_ADDRESS       0x210     /* Board located at address 210. */
#define PORTAoffset        0x00      /* Offset for Port A */
#define PORTBoffset        0x01      /* Offset for Port B */
#define PORTCoffset        0x02      /* Offset for Port C */
#define CNFGoffset         0x03      /* Offset for CNFG */

register unsigned int porta, portb, portc, cnfg;
char valread;                        /* Variable to store data read from a
                                        port */

/* Calculate register addresses. */
porta = BASE_ADDRESS + PORTAoffset;
portb = BASE_ADDRESS + PORTBoffset;
portc = BASE_ADDRESS + PORTCoffset;
cnfg  = BASE_ADDRESS + CNFGoffset;

/* EXAMPLE 1-Set up interrupts for Mode 1 input for Port A.  Select PC6 as the
interrupt enable bit. */

outp(cnfg,0xB0);                  /* Port A is an input in Mode 1. */
outp(cnfg,0x09);                  /* Set PC4 to enable interrupts from
                                     82C55A. */
outp(cnfg,0x0C);                  /* Clear PC6 to enable interrupts. */
```

```
/* EXAMPLE 2-Set up interrupts for Mode 1 input for Port B.  Select PC6 as the
interrupt enable bit. */

outp(cnfg,0x86);                  /* Port B is an input in Mode 1. */
outp(cnfg,0x05);                  /* Set PC2 to enable interrupts from
                                     82C55A. */
outp(cnfg,0x0C);                  /* Clear PC6 to enable interrupts. */

/* EXAMPLE 3-Set up interrupts for Mode 1 output for Port A.  Select PC4 as
the interrupt enable bit. */

outp(cnfg,0xA0);                  /* Port A is an output in Mode 1. */
outp(cnfg,0x0D);                  /* Set PC6 to enable interrupts from
                                     82C55A. */
outp(cnfg,0x0C);                  /* Clear PC4 to enable interrupts. */

/* EXAMPLE 4-Set up interrupts for Mode 1 output for Port B.  Select PC4 as
the interrupt enable bit. */

outp(cnfg,0x84);                  /* Port B is an output in Mode 1. */
outp(cnfg,0x05);                  /* Set PC2 to enable interrupts from
                                     82C55A. */
outp(cnfg,0x08);                  /* Clear PC4 to enable interrupts. */

/* EXAMPLE 5-Set up interrupts for Mode 2 output transfers.  Select PC2 as the
interrupt enable bit. */

outp(cnfg,0xC0);                  /* Mode 2 output */
outp(cnfg,0x0D);                  /* Set PC6 to enable interrupts from
                                     82C55A. */
outp(cnfg,0x04);                  /* Clear PC2 to enable interrupts. */

/* EXAMPLE 6-Set up interrupts for Mode 2 input transfers.  Select PC2 as the
interrupt enable bit. */

outp(cnfg,0xD0);                  /* Mode 2 input */
outp(cnfg,0x09);                  /* Set PC4 to enable interrupts from
                                     82C55A. */
outp(cnfg,0x04);                  /* Clear PC2 to enable interrupts. */
}
```

DIO-24 Interrupt Handling

A jumper setting on the AT-MIO-16D selects the signal that is used for the DIO-24 interrupt enable signal. If jumper W14 is set to N/C, interrupts are disabled. Jumper W14 can be used to select PC2, PC4, or PC6 as the active low interrupt enable signal. For example, if PC2 is selected, interrupts are enabled if PC2 is logic low. If PC2 is logic high, interrupts from the DIO-24 circuitry are disabled. Table 4-10 summarizes which signal should be used as the interrupt enable for all mode combinations.

Table 4-10. DIO-24 Interrupt Enable Signals for All Mode Combinations

Port A Mode 2 Output	Port A Mode 2 Input	Port B Mode 1 Output	Port B Mode 1 Input	Port A Mode 1 Output	Port A Mode 1 Input	Port B Mode 0	Port A Mode 0	Interrupt Enable Bit
No	No	No	No	No	No	Yes	Yes	N/C
No	No	No	No	No	Yes	Yes	No	PC6, PC2
No	No	No	No	Yes	No	Yes	No	PC4, PC2
No	No	No	Yes	No	No	No	Yes	PC6, PC4
No	No	No	Yes	No	Yes	No	No	PC6
No	No	No	Yes	Yes	No	No	No	PC4
No	No	Yes	No	No	No	No	Yes	PC6, PC4
No	No	Yes	No	No	Yes	No	No	PC6
No	No	Yes	No	Yes	No	No	No	PC4
No	Yes	No	No	No	No	Yes	No	PC2
No	Yes	No	Yes	No	No	No	No	N/C
No	Yes	Yes	No	No	No	No	No	N/C
Yes	No	No	No	No	No	Yes	No	PC2
Yes	No	No	Yes	No	No	No	No	N/C
Yes	No	Yes	No	No	No	No	No	N/C

The recommended jumper settings for W14 are as follows:

- PC6 – If Port A is in Mode 1 input.
- PC4 – If Port A is in Mode 1 output.
- PC2 – If Port A is in Mode 2 (Port B is not in Mode 1).

To enable interrupts from the DIO-24 circuitry of the AT-MIO-16D board, select PC2, PC4, or PC6 as the active low interrupt enable signal. Initially, set the selected bit high to disable unwanted interrupts.

Program the DIO-24 circuitry for the I/O mode you want. To enable interrupts from the 82C55A, set either the INTEA or the INTEB bit to enable interrupts from Port A or Port B, respectively. In Mode 2, set either INTE1 or INTE2 for interrupts on input or output transfers. After interrupts have been enabled from the 82C55A, clear the selected interrupt enable bit to enable interrupts from the DIO-24 circuitry.

An external signal can be used to interrupt the AT-MIO-16D when Port A or Port B is in Mode 0. Select PC2, PC4, or PC6 as the interrupt enable bit and clear the selected bit to enable interrupts. Connect the external signal that should trigger an interrupt to either PC3 or PC0. When the external signal becomes logic high, an interrupt request occurs. To disable the external signal interrupt, set the selected interrupt enable bit to logic high.

Chapter 5
Calibration Procedures

This chapter discusses the calibration procedures for the AT-MIO-16D analog input and analog output circuitry.

The AT-MIO-16D is calibrated at the factory before shipment. To maintain the 12-bit accuracy of the AT-MIO-16D analog input and analog output circuitry, recalibration at six-month intervals is recommended.

Factory calibration is performed with the AT-MIO-16D in its default factory configuration:

- DIFF analog input mode
- -10 to +10 V analog input range (bipolar)
- -10 to +10 V analog output range (bipolar with internal reference selected)

Recalibration of your AT-MIO-16D board is recommended any time you change your board configuration.

Calibration Equipment Requirements

For best measurement results, the AT-MIO-16D analog input circuitry needs to be calibrated so that its measurement accuracy is within ±0.012% of its input range (±1/2 LSB). According to standard practice, the equipment used to calibrate the AT-MIO-16D should be 10 times as accurate, that is, have ±0.001% rated accuracy. Practically speaking, calibration equipment with four times the accuracy of the item under calibration is generally considered acceptable. Four times the accuracy of the AT-MIO-16D is 0.003%. To calibrate the AT-MIO-16D board you need the following equipment.

For analog input calibration, a precision variable DC voltage source (usually a calibrator):

Accuracy: ±0.001% standard
±0.003% sufficient

Range: greater than ±10 V

Resolution: 100 µV in ±10 V range (5 1/2 digits)

For analog output calibration, a voltmeter:

Accuracy: ±0.001% standard
±0.003% sufficient

Range: greater than ±10 V

Resolution: 100 µV in ±10 V range (5 1/2 digits)

Calibration Trimpots

There are eight trimpots on the AT-MIO-16D for calibration. The location of these trimpots on the AT-MIO-16D board is shown in the partial diagram of the board below in Figure 5-1.

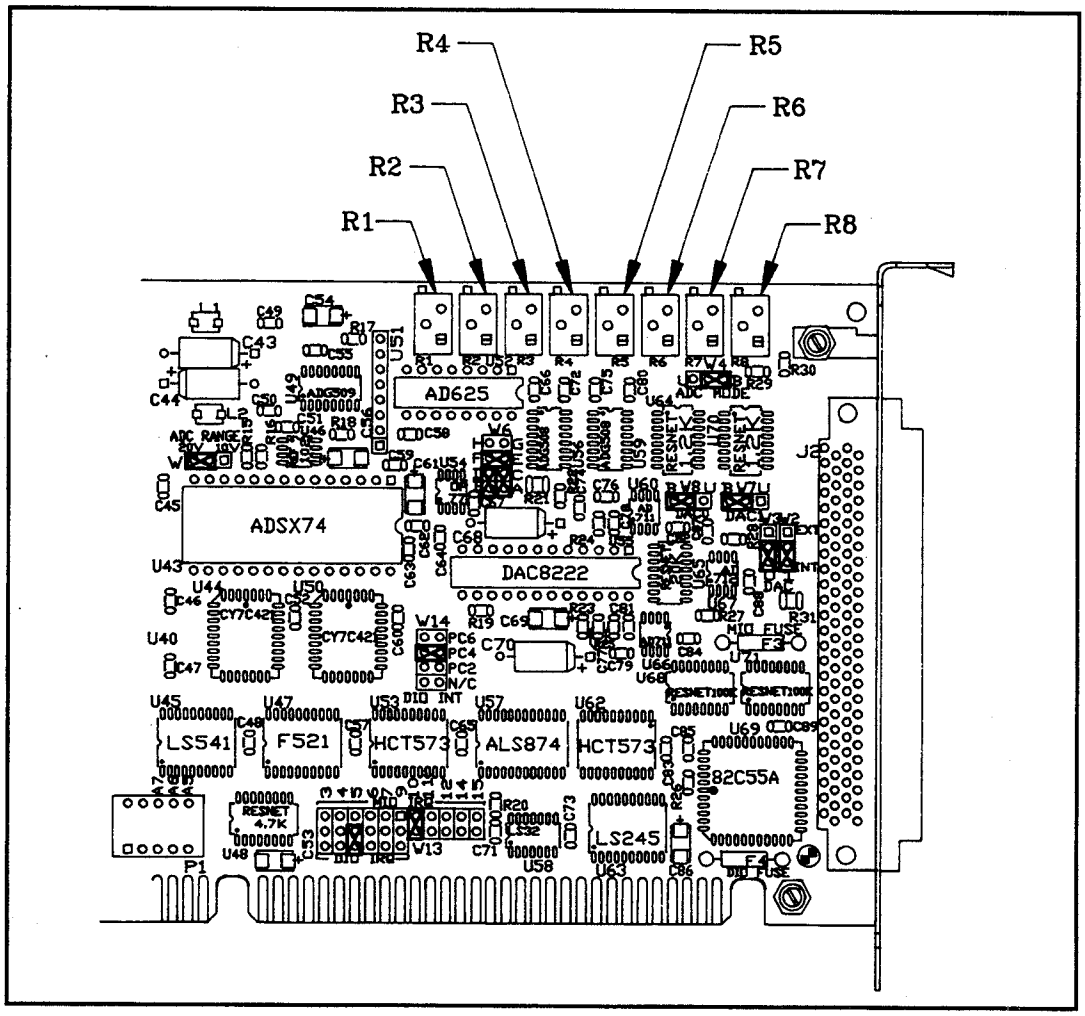


Figure 5-1. Calibration Trimpot Location Diagram

The following trimpots are used to calibrate the analog input circuitry:

R1 – Gain trim, analog input
R6 – Bipolar offset trim, analog input
R8 – Unipolar offset trim, analog input
R2 – Instrumentation amplifier input offset trims

The following trimpots are used to calibrate the analog output circuitry:

R5 – Gain trim, analog output channel 0
R4 – Gain trim, analog output channel 1
R7 – Offset trim, analog output channel 0
R3 – Offset trim, analog output channel 1

Analog Input Calibration

To null out error sources that compromise the quality of measurements, you must calibrate the analog input circuitry by adjusting the following potential sources of error:

- Offset error at the input of the instrumentation amplifier
- Offset error at the input of the ADC
- Gain error of analog input circuitry

Offsets at the input to the instrumentation amplifier contribute gain-dependent error to the analog input system. This offset is multiplied by the gain of the instrumentation amplifier. To calibrate this offset, you must ground the analog input, read it at two different gain settings, and adjust a trimpot until the readings match at the two different gain settings.

Offset error at the input of the ADC is the total of the voltage offsets contributed by the circuitry from the output of the instrumentation amplifier to the ADC input (including the offsets of the ADC). Offset errors appear as a voltage added to the input voltage being measured. To calibrate this offset, you must apply V_{-fs} + 1/2 LSB to the analog input circuitry and adjust a trimpot until the ADC returns readings that flicker between its most negative count and the most negative count plus one. The voltages corresponding to V_{-fs} and 1 LSB are given in the following table.

All the stages up to and including the input of the ADC contribute to the gain error of the analog input circuitry. With the instrumentation amplifier set to a gain of 1, the gain of analog input circuitry is ideally one. The gain error is the deviation of the gain from 1 and appears as a multiplication of the input voltage being measured. To calibrate this offset, you must apply V_{+fs} - 3/2 LSB to the analog input circuitry and adjust a potentiometer until the ADC returns readings that flicker between its most positive count and the most positive count minus one. The voltages corresponding to V_{+fs} and 1 LSB are given below.

The voltages corresponding to V_{-fs}, which is the most negative voltage that the ADC can read, V_{+fs} - 1, which is the most positive voltage the ADC can read, and 1 LSB, which is the voltage corresponding to one count of the ADC, depend on the input range selected. The value of these voltages for each input range is given in this table:

Input Range	V_{-fs}	V_{+fs} - 1	1 LSB	1/2 LSB
-10 to +10 V	-10 V	+9.99512 V	4.88 mV	2.44 mV
-5 to +5 V	-5 V	+4.99756 V	2.44 mV	1.22 mV
0 to 10 V	0 V	+9.99756 V	2.44 mV	1.22 mV

Board Configuration

Calibration procedure differs depending on input ranges and input configuration modes selected. Two analog input calibration procedures are given below: one for the two bipolar input configurations (-10 to +10 V and -5 to +5 V), and one for the unipolar input configuration (0 to +10 V).

The calibration procedures presented here assume that your AT-MIO-16D board is configured for DIFF input. If necessary, reconfigure your board for DIFF input before using the following calibration procedures.

To calibrate your board with a nondifferential input setting, the procedure is similar to the procedures outlined below with the following exception: the procedures given below apply the input calibration voltages across the positive (+) and negative (-) inputs for DIFF channel 0. For single-ended input, apply your calibration voltages between the channel 0 positive (+) input and whichever ground system you are using (refer to Chapter 2, *Configuration and Installation*, for instructions on using single-ended input connections).

Bipolar Input Calibration Procedure

If your board is configured for bipolar input, which provides the ranges -5 to +5 V or -10 to +10 V, then complete the following procedure in the order given. This procedure assumes that ADC readings are in the range -2,048 to +2,047.

1. Adjust the amplifier input offset.

To adjust the amplifier input offset, follow these steps:

a. Connect both ACH0 (pin 3 on the I/O connector) and ACH8 (pin 4) to AI SENSE (pin 19).

b. Take analog input readings from channel 0 at the following gains:

both 1 and 500 for the AT-MIO-16DL

both 1 and 8 for the AT-MIO-16DH

c. Adjust trimpot R2 until the readings match to within one count at both gain settings.

2. Adjust the ADC input offset.

Adjust the ADC input offset by applying an input voltage across ACH0 and ACH8. This input voltage is V_{-fs} + 1/2 LSB and depends on the input range selected:

Input Range	Calibration Voltage
-10 to +10 V -5 to +5 V	-9.99756 V -4.99878 V

a. Connect the calibration voltage across ACH0 (pin 3 on the I/O connector) and ACH8 (pin 4). Connect the ground point on the calibration voltage source to AI SENSE (pin 19).

b. Take analog input readings from channel 0 at a gain of 1, and adjust trimpot R6 until the ADC readings flicker evenly between -2,048 and -2,047.

3. Adjust the analog input gain.

Adjust the analog input gain by applying an input voltage across ACH0 and ACH8. This input voltage is V_{+fs} - 3/2 LSB and depends on the input range selected:

Input Range	Calibration Voltage
-10 to +10 V -5 to +5 V	+9.99268 V +4.99634 V

a. Connect the calibration voltage across ACH0 (pin 3 on the I/O connector) and ACH8 (pin 4). Connect the ground point on the calibration voltage source to AI SENSE (pin 19).

b. Take analog input readings from channel 0 at a gain of 1, and adjust trimpot R1 until the ADC readings flicker evenly between 2,046 and 2,047.

Unipolar Input Calibration Procedure

If your board is configured for unipolar input, which provides an input range of 0 to +10 V, then complete the following procedure in the order given. This procedure assumes that ADC readings are in the range 0 to +4,095.

1. Adjust the amplifier input offset.

To adjust the amplifier input offset, follow these steps:

a. Connect both ACH0 (pin 3 on the I/O connector) and ACH8 (pin 4) to AI SENSE (pin 19).

b. Take analog input readings from channel 0 at a gain of 1, and adjust trimpot R8 until a reading of roughly two counts is returned.

c. Take analog input readings from channel 0 at the following gains:

   both 1 and 500 for the AT-MIO-16DL

   both 1 and 8 for the AT-MIO-16DH

d. Adjust trimpot R2 until the readings at each gain setting match to within one count of each other.

2. Adjust the ADC input offset.

Adjust the ADC input offset by applying an input voltage across ACH0 and ACH8. This input voltage is 1.22 mV, or 0 V + 1/2 LSB.

a. Connect the calibration voltage (1.22 mV) across ACH0 (pin 3 on the I/O connector) and ACH8 (pin 4). Connect the ground point on the calibration voltage source to AI SENSE (pin 19).

b. Take analog input readings from channel 0 at a gain of 1, and adjust trimpot R8 until the ADC readings flicker evenly between zero and one.

3. Adjust the analog input gain.

Adjust the analog input gain by applying an input voltage across ACH0 and ACH8. This input voltage is +9.99634 V, or V_{+fs} - 3/2 LSB.

a. Connect the calibration voltage (+9.99634 V) across ACH0 (pin 3 on the I/O connector) and ACH8 (pin 4). Connect the ground point on the calibration voltage source to AI SENSE (pin 19).

b. Take analog input readings from Channel 0 at a gain of 1, and adjust trimpot R1 until the ADC readings flicker evenly between 4,094 and 4,095.

Analog Output Calibration

To null out error sources that affect the accuracy of the output voltages generated, you must calibrate the analog output circuitry by adjusting the following potential sources of error:

- Analog output offset error
- Analog output gain error

Offset error in the analog output circuitry is the total of the voltage offsets contributed by each component in the circuitry. This error appears as a voltage difference between the desired voltage and the actual output voltage generated and is independent of D/A setting. To correct this offset gain error, set the D/A to negative full scale and adjust a trimpot until the output voltage is the negative full-scale value ±1/2 LSB.

Gain error in the analog output circuitry is the product of the gains contributed by each component in the circuitry. This error appears as a voltage difference between the desired voltage and the actual output voltage generated, which depends on the D/A setting. This gain error is corrected by setting the D/A to positive full scale and adjusting a trimpot until the output voltage corresponds to the positive full-scale value ±1/2 LSB.

Board Configuration

The calibration procedure differs if you select either bipolar or unipolar output configuration. A procedure for each configuration is given below.

The calibration procedures presented in this chapter assume that the internal voltage reference +10 V is selected for the analog output channel to be calibrated.

To calibrate your board to an external reference input (DC only), you must recalculate the desired output voltages to calibrate to.

- For bipolar output:

  1 LSB = V_{extref} /2,048 (therefore, 1/2 LSB = V_{extref} /4,096)
  $V_{-fs} = -V_{extref}$
  $V_{+fs} = V_{extref}$ - 1 LSB

- For unipolar output:

  1 LSB = V_{extref} /4,096 (therefore, 1/2 LSB = V_{extref} /8,192)
  V_{-fs} = 0 V
  $V_{+fs} = V_{extref}$ - 1 LSB

In calibrating to your own external reference, you should write your own procedures using the following procedures as a guide. Substitute your calculated voltages for those given.

Bipolar Output Calibration Procedure

If your board is configured for bipolar output and two's complement mode, which provides an output range of -10 to +10 V, then complete the following procedure in the order given.

1. Adjust the analog output offset.

Adjust the analog output offset by measuring the output voltage generated with the DAC set at negative full-scale (0). This output voltage should be V_{-fs} ±1/2 LSB. For bipolar output, V_{-fs} = -10 V, and 1/2 LSB = 2.44 mV.

- For analog output channel 0:

  a. Connect the voltmeter between DAC0 OUT (pin 20 on the I/O connector) and AOGND (pin 23).

  b. Set the analog output channel to -10 V by writing -2,048 to the DAC.

c. Adjust trimpot R7 until the output voltage read is -10 V ±2.44 mV, that is, between -10.00244 and -9.99756 V.

- For analog output channel 1:

a. Connect the voltmeter between DAC1 OUT (pin 21 on the I/O connector) and AOGND (pin 23).

b. Set the analog output channel to -10 V by writing -2,048 to the DAC.

c. Adjust trimpot R3 until the output voltage read is -10V ± 2.44 mV, that is, between -10.00244 and -9.99756 V.

2. Adjust the analog output gain.

Adjust the analog output gain by measuring the output voltage generated with the DAC set at positive full scale (2,047). This output voltage should be V_{+fs} ±1/2 LSB. For bipolar output, V_{+fs} = +9.99512 V, and 1/2 LSB = 2.44 mV.

- For analog output channel 0:

a. Connect the voltmeter between DAC0 OUT (pin 20 on the I/O connector) and AOGND (pin 23).

b. Set the analog output channel to +9.99512 V by writing 2,047 to the DAC.

c. Adjust trimpot R5 until the output voltage read is +9.99512 V ±2.44 mV, that is, between 9.99268 and 9.99756 V.

- For analog output channel 1:

a. Connect the voltmeter between DAC1 OUT (pin 21 on the I/O connector) and AOGND (pin 23).

b. Set the analog output channel to +9.99512 V by writing 2,047 to the DAC.

c. Adjust trimpot R4 until the output voltage read is +9.99512 V ±2.44 mV, that is, between 9.99756 and 9.99268 V.

Unipolar Output Calibration Procedure

If your analog output channel is configured for unipolar output, which provides an output range of 0 to +10 V, then calibrate your board by performing the following procedure.

1. Adjust the analog output offset.

Adjust the analog output offset by measuring the output voltage generated with the DAC set at zero. This output voltage should be V_{-fs} ±1/2 LSB. For unipolar output, V_{-fs} = 0 V, and 1/2 LSB = 1.22 mV.

- For analog output channel 0:

  a. Connect the voltmeter between DAC0 OUT (pin 20 on the I/O connector) and AO GND (pin 23).

  b. Set the analog output channel to 0 V by writing 0 to the DAC.

  c. Adjust trimpot R7 until the output voltage read is 0 V ±1.22 mV.

- For analog output channel 1:

  a. Connect the voltmeter between DAC1 OUT (pin 21 on the I/O connector) and AO GND (pin 23).

  b. Set the analog output channel to 0 V by writing 0 to the DAC.

  c. Adjust trimpot R3 until the output voltage read is 0 V ±1.22 mV.

2. Adjust the analog output gain.

Adjust the analog output gain by measuring the output voltage generated with the DAC set at positive full scale (4,095). This output voltage should be V_{+fs} ±1/2 LSB. For unipolar output, V_{+fs} = +9.99756 V, and 1/2 LSB = 1.22 mV.

- For analog output channel 0:

  a. Connect the voltmeter between DAC0 OUT (pin 20 on the I/O connector) and AO GND (pin 23).

  b. Set the analog output channel to +9.99756 V by writing 4,095 to the DAC.

  c. Adjust trimpot R5 until the output voltage read is +9.99756 V ±1.22 mV, that is, between 9.99634 and 9.99878 V.

- For analog output channel 1:

  a. Connect the voltmeter between DAC1 OUT (pin 21 on the I/O connector) and AO GND (pin 23).

  b. Set the analog output channel to +9.99756 V by writing 4,095 to the DAC.

  c. Adjust trimpot R4 until the output voltage read is +9.99756 V ±1.22 mV, that is, between 9.99634 and 9.99878 V.

Appendix A
Specifications

This appendix lists the specifications for the AT-MIO-16D. These are typical at 25° C unless otherwise stated. The operating temperature range is 0° to 70° C.

MIO-16 Circuitry Specifications

Analog Input

Number of input channels	16 single-ended, 8 differential
Analog resolution	12-bit, 1 in 4,096
Relative accuracy (nonlinearity + quantization error, see explanation of specifications)	±1.5 LSB maximum ±0.75 LSB typical
Integral nonlinearity	±0.5 LSB maximum
Differential nonlinearity	±1 LSB maximum (no missing codes over temperature); ±0.5 LSB typical worst-case codes
Differential analog input ranges	±10 V, ±5 V, or 0 to +10 V, jumper-selectable
Analog input range	±12 V
Common mode range	±7 V for ±10 V differential analog input range ±9.5 V for ±5 V differential analog input range ±7 V for 0 to +10 V differential analog input range
Instrumentation amplifier	
Common mode rejection ratio	75 dB minimum, DC through 100 Hz
Input bias current	±25 nA maximum
Input offset current	±15 nA maximum
Input impedance	1 GΩ
Gain ranges	1, 2, 4, and 8 for AT-MIO-16DH, 1, 10, 100, and 500 for AT-MIO-16DL, software-selectable
Gain accuracy (includes pot adjustment range)	
gain = 1	±0.83% of full-scale voltage, adjustable to zero
gain > 1	±0.85% of full-scale voltage, ±0.08% of full-scale voltage maximum when gain error adjusted to zero at gain = 1
Temperature drift	36 ppm/° C

Analog Input (continued)

Input offset voltage (includes pot adjustment range)	±50 mV for gain 1, adjustable to zero ±25 mV for gain 2 ±15 mV for gain 4 ±10 mV for gain 8 ±5 mV for gain 10 ±2 mV for gain 100 ±1.5 mV for gain 500
Temperature drift	160 µV/° C + 6 µV/° C + gain
Other system offset voltage (includes pot adjustment range)	±85 mV for ±10 V range, adjustable to zero ±45 mV for ±5 V range ±30 mV for 0 to +10 V range
System noise (figures are for 20-V range; multiply by 2 for 10-V range)	0.15 LSB rms for gains 1 to 10 0.25 LSB rms for gain 100 0.5 LSB rms for gain 500

Explanation of Analog Input Specifications

Relative accuracy is a measure of the linearity of an ADC. However, relative accuracy is a tighter specification than a *nonlinearity* specification. Relative accuracy indicates the maximum deviation from a straight line for the analog-input-to-digital-output transfer curve. If an ADC has been calibrated perfectly, then this straight line is the ideal transfer function, and the relative accuracy specification indicates the worst deviation from the ideal that the ADC permits.

A relative accuracy specification of ±1 LSB is roughly equivalent to (but not the same as) a ±1/2 LSB nonlinearity or integral nonlinearity specification because relative accuracy encompasses both nonlinearity and variable quantization uncertainty, a quantity often mistakenly assumed to be exactly ±1/2 LSB. Although quantization uncertainty is ideally ±1/2 LSB, it can be different for each possible digital code and is actually the analog width of each code. Thus, it is more specific to use relative accuracy as a measure of linearity than it is to use what is normally called nonlinearity, because relative accuracy ensures that the *sum* of quantization uncertainty and analog-to-digital (A/D) conversion error does not exceed a given amount.

Integral nonlinearity in an ADC is an often ill-defined specification that is supposed to indicate overall A/D transfer linearity of a converter. The manufacturers of the ADC chips used by National Instruments specify their integral nonlinearity by stating that the analog center of any code will not deviate from a straight line by more than ±1/2 LSB. This specification is misleading because although the center of a particularly wide code can be found within ±1/2 LSB of the ideal, one of its edges may be well beyond ±1 LSB; thus, the ADC would have a relative accuracy of that amount. National Instruments tests its boards to ensure that they meet all three linearity specifications defined in this appendix; specifications for integral nonlinearity are included primarily to maintain compatibility with a convention of specifications used by other board manufacturers. Relative accuracy, however, is much more useful.

Differential nonlinearity is a measure of deviation of code widths from their theoretical value of 1 LSB. The width of a given code is the size of the range of analog values that can be input to produce that code, ideally 1 LSB. A specification of ±1 LSB differential nonlinearity ensures that no code has a width of 0 LSBs (that is, no missing codes) and that no code width exceeds 2 LSBs.

System noise is the amount of noise seen by the ADC when there is no signal present at the input of the board. The amount of noise that is reported directly (without any analysis) by the ADC is not necessarily the amount of real noise present in the system, unless the noise is ≥ 0.5 LSB root mean square (rms). Noise that is less than this magnitude produces varying amounts of flicker, and the amount of flicker seen is a function of how near the real mean of the noise is to a code transition. If the mean is near or at a transition between codes, the ADC flickers evenly between the two codes, and the noise is seen as very nearly 0.5 LSB. If the mean is near the center of a code and the noise is relatively small, very little or no flicker is seen, and the noise is reported by the ADC as nearly 0 LSB. From the relationship between the mean of the noise and the measured rms magnitude of the noise, the character of the noise can be determined. National Instruments has determined that the character of the noise in the AT-MIO-16D is fairly Gaussian, and so the noise specifications given are the amounts of pure Gaussian noise required to produce our readings.

Analog Data Acquisition Rates

Single-Channel Acquisition Rates

The maximum data acquisition rate for the AT-MIO-16D is 100 ksamples/sec. Permissible data acquisition rates are determined by the minimum A/D conversion time of the system. This minimum conversion time is the sum of the conversion time of the ADC and the settling time of the analog input front end. When data acquisition is performed on a single analog input channel, the time required for the input sample-and-hold amplifier to acquire the input signal and settle to 12-bit accuracy (0.01%) is added to the conversion time of the ADC. The sample-and-hold amplifier in the AT-MIO-16D takes 1 µsec typical and 1.5 µsec maximum to settle to 0.01% for a 10 V step. The data acquisition rates shown in the preceding table are the best rates for single-channel acquisition. These rates take into account the specified typical and maximum (worst-case) conversion times of the ADC plus 2 µsec to allow for sample-and-hold settling time.

Multiple-Channel Scanning Acquisition Rates

The following are the maximum multiple-channel scan rates recommended for the AT-MIO-16D:

Gain	Data Acquisition Rate
1, 2, 4, 8	100 ksamples/sec
10	100 ksamples/sec
100	70 ksamples/sec
500	20 ksamples/sec

Recommended multiple-channel scanning rates are slower than single-channel acquisition rates for higher gains, because as gain is increased, the AT-MIO-16D circuitry takes longer to settle from one channel voltage to the next. The recommended settling time for gains of 1 through 10 is 10 µsec; for a gain of 100, 14 µsec is recommended; and 50 µsec is recommended for a gain of 500. For these settling times, the circuitry on the AT-MIO-16D boards will settle to 0.5 LSBs of the final value, or to 0.01%, for a full-scale step.

A full-scale difference between input channels is the worst-case switching condition for channel scanning settling time, with one channel at the positive end of the full-scale range and the other channel at the negative end of the full-scale range. The lower the analog input source impedance, the better the settling time performance.

Analog Output

Number of output channels	Two
Type of digital-to-analog converter	12-bit, multiplying
Relative accuracy (nonlinearity)	±0.5 LSB maximum ±0.25 LSB typical
Differential nonlinearity	±1 LSB maximum (monotonic over temperature) ±0.2 LSB typical
Gain error (includes pot adjustment range, but excludes reference)	±0.77% of full-scale voltage, adjustable to zero
Voltage offset (includes pot adjustment range)	±64 mV bipolar mode, ±32 mV unipolar mode, adjustable to zero
Internal voltage reference	10 V (±10 mV maximum); 10 ppm/°C drift
Output voltage ranges (jumper-selectable)	0 to 10 V, unipolar mode; ±10 V, bipolar mode
Current drive capability	±2 mA
Output settling time to 0.01%	4 µsec for 20 V step
Output slew rate	30 V/µsec
Output noise	1 mV rms, DC to 1 MHz
Output impedance	≤ 0.2 Ω

Explanation of Analog Output Specifications

Relative accuracy in a digital-to-analog (D/A) system is the same as nonlinearity because no uncertainty is added due to code width. Unlike an ADC, every digital code in a D/A system represents a specific analog value rather than a range of values. The relative accuracy of the system is therefore limited to the worst-case deviation from the ideal correspondence (a straight line), excepting noise. If a D/A system has been calibrated perfectly, then the relative accuracy specification reflects its worst-case absolute error.

Differential nonlinearity in a D/A system is a measure of deviation of code width from 1 LSB. In this case, code width is the difference between the analog values produced by consecutive digital codes. A specification of ±1 LSB differential nonlinearity ensures that the code width is always greater than 0 LSBs (guaranteeing monotonicity) and is always less than 2 LSBs.

Digital I/O (MIO-16 I/O Connector only)

Compatibility	Transistor-transistor logic (TTL) compatible
Output current source capability	Can source 2.6 mA and maintain V_{OH} at 2.4 V
Output current sink capability	Can sink 24 mA and maintain V_{OL} at 0.5 V

Timing I/O

Number of channels	Four: three counter/timers and one frequency output
Resolution	16-bit for 3 counter/timers, 4-bit for frequency output channel
Base clock available	1 MHz, 100 kHz, 10 kHz, 1 kHz, 100 Hz
Base clock accuracy	±0.01%
Compatibility	TTL-compatible inputs and outputs. Counter gate and source inputs are pulled up with 4.7-kΩ resistors onboard.
Counter input frequency	6.9 MHz maximum (145 nsec period) with a minimum pulse width of 70 nsec
Current source capability	200 µA
Current sink capability	3.2 mA

DIO-24 Circuitry Specifications

I/O Signals Rating

Absolute maximum voltage input rating	-0.5 to +7.0 V with respect to GND

Input Signal Specifications	Minimum	Maximum
Input logic–high voltage	2.0 V	5.25 V
Input logic–low voltage	0.0 V	0.8 V
Input current ($0 < I_{in} < 5$ V)	-10.0 µA	10.0 µA

Output Signal Specifications

Pin 49 (at +5 V) 1.0 A typical (fused)

	Minimum	Maximum
Output logic–high voltage at I_{out} = -200 µA	2.4 V	5.0 V
Output logic–low voltage at I_{out} = 1.7 mA	0.0 V	0.45 V
Darlington drive current (R_{EXT} = 750 Ω, V_{EXT} = 1.5 V)	-1.0 mA	-4.0 mA

Transfer Rates

Maximum	500 kbytes/sec
Typical	300 kbytes/sec

Note: The transfer rate depends on both the computer speed and the software speed. The maximum transfer rate shown previously is the result of running an assembly program, which continuously writes a constant to an output port, on an 8-MHz PC AT compatible. The typical transfer rate is the result of running an assembly program, which continuously reads data from memory and writes to an output port, on an 8-MHz PC AT compatible.

Power Requirement (from PC AT I/O Channel)

Power consumption 1.6 A typical at +5 VDC

Physical

Board dimensions 13.3 in. by 3.9 in.

I/O connector 100-pin male, separable into two 50-pin female ribbon-cable connectors

Operating Environment

Component temperature 0° to 70° C

Relative humidity 5% to 90% noncondensing

Storage Environment

Temperature -55° to 150° C

Relative humidity 5% to 90% noncondensing

Appendix B
MIO-16 I/O Connector

This appendix describes the pinout and signal names for the MIO-16 50-pin I/O connector of the AT-MIO-16D.

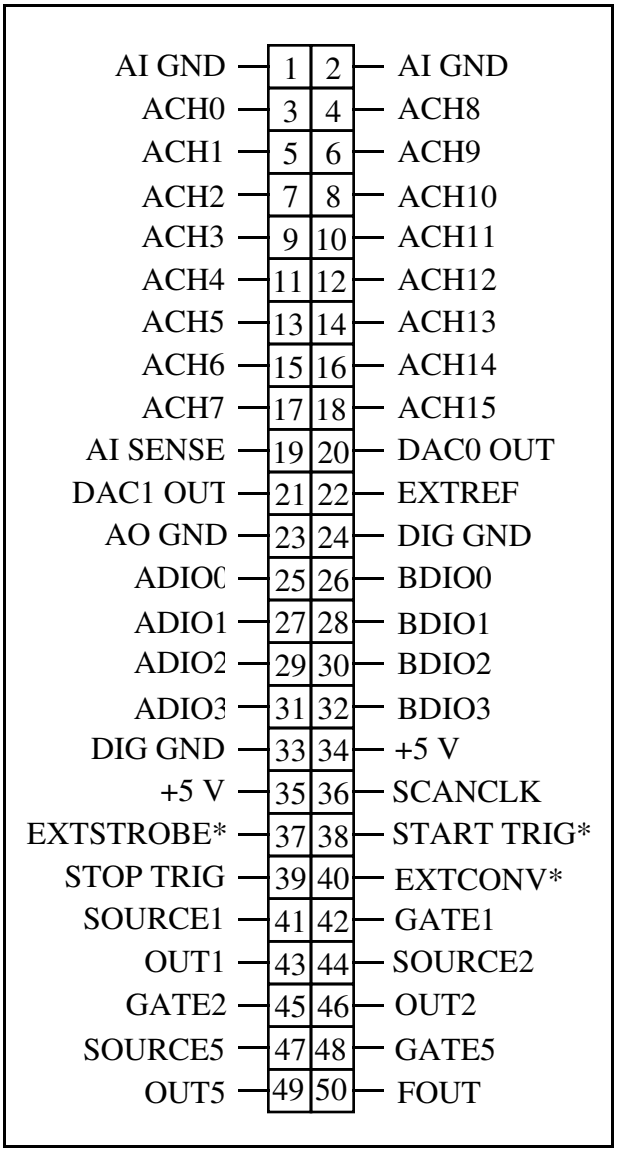


Figure B-1. AT-MIO-16D MIO-16 I/O Connector

Pin	Signal Name	Reference	Description
1-2	AI GND	N/A	Analog Input Ground – These pins are the reference point for single-ended measurements and the bias current return point for differential measurements.
3-18	ACH<0..15>	AIGND	Analog Input Channels 0 through 15 – In differential mode, the input is configured for up to eight channels. In single-ended mode, the input is configured for up to 16 channels.
19	AI SENSE	AIGND	Analog Input Sense – This pin serves as the reference node when the board is in NRSE configuration. If desired, this signal can be programmed to be driven by the board analog input ground.
20	DAC0 OUT	AOGND	Analog Channel 0 Output – This pin supplies the voltage output of analog output channel 0.
21	DAC1 OUT	AOGND	Analog Channel 1 Output – This pin supplies the voltage output of analog output channel 1.
22	EXTREF	AOGND	External Reference – This is the external reference input for the analog output circuitry.
23	AO GND	N/A	Analog Output Ground – The analog output voltages are referenced to this node.
24,33	DIG GND	N/A	Digital Ground – This pin supplies the reference for the digital signals at the I/O connector as well as the +5 VDC supply.
25, 27, 29, 31	ADIO<0..3>	DIGGND	Digital I/O port A signals.
26, 28, 30, 32	BDIO<0..3>	DIGGND	Digital I/O port B signals.
34-35	+5 V	DIGGND	+5 VDC Source – This pin is fused for up to 1 A of +5 V supply.
36	SCANCLK	DIGGND	Scan Clock – This pin pulses once for each A/D conversion in the scanning modes. The low-to-high edge indicates when the input signal can be removed from the input or switched to another signal.
37	EXTSTROBE*	DIGGND	External Strobe – Writing to the EXTSTROBE* Register results in a minimum 200 nsec low pulse on this pin.

Pin	Signal Name	Reference	Description (continued)
38	START TRIG*	DIGGND	External Trigger – In posttrigger data acquisition sequences, a high-to-low edge on START TRIG* initiates the sequence. In pretrigger applications, the high-to-low edge of START TRIG* initiates pretrigger conversions while the STOP TRIG signal initiates the posttrigger sequence.
39	STOP TRIG	DIGGND	Stop Trigger – In pretrigger data acquisition, the high-to-low edge of STOP TRIG initiates the posttrigger sequence.
40	EXTCONV*	DIGGND	External Convert – A high-to-low edge on EXTCONV* causes an A/D conversion to occur. If EXTGATE* or EXTCONV* is low, conversions are inhibited.
41	SOURCE1	DIGGND	SOURCE1 – This pin is from the Am9513A Counter 1 signal.
42	GATE1	DIGGND	GATE1 – This pin is from the Am9513A Counter 1 signal.
43	OUT1	DIGGND	OUTPUT1 – This pin is from the Am9513A Counter 1 signal.
44	SOURCE2	DIGGND	SOURCE2 – SOURCE5 – This pin is from the Am9513A Counter 2 signal.
45	GATE2	DIGGND	GATE2 – This pin is from the Am9513A Counter 2 signal.
46	OUT2	DIGGND	OUTPUT2 – This pin is from the Am9513A Counter 2 signal.
47	SOURCE5	DIGGND	SOURCE5 – This pin is from the Am9513A Counter 5 signal.
48	GATE5	DIGGND	GATE5 – This pin is from the Am9513A Counter 5 signal.
49	OUT5	DIGGND	OUT5 – This pin is from the Am9513A Counter 5 signal.
50	FOUT	DIGGND	Frequency Output – This pin is from the Am9513A FOUT signal.

Appendix C
DIO-24 I/O Connector

This appendix describes the pinout and signal names for the DIO-24 50-pin I/O connector of the AT-MIO-16D.

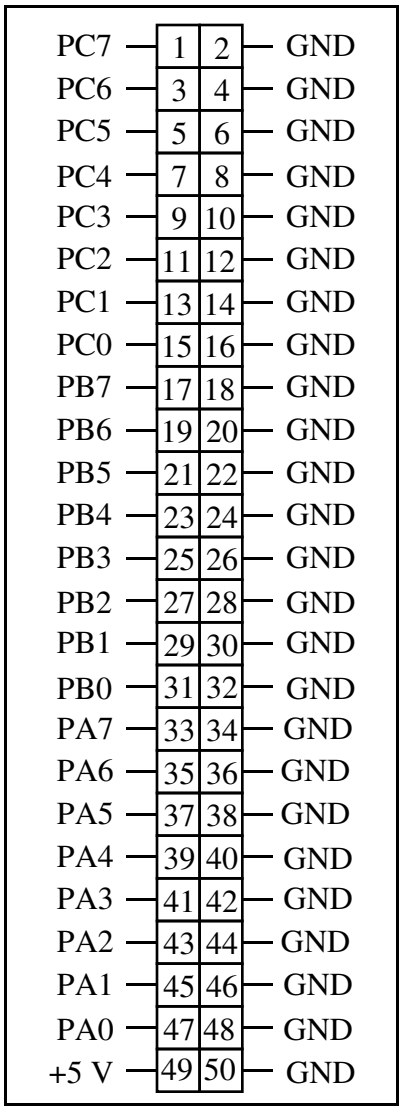


Figure C-1. AT-MIO-16D DIO-24 I/O Connector

Pin	Signal Name	Reference	Description
1, 3, 5, 7, 9, 11, 13, 15	PC7 through PC0	DIGGND	Bidirectional data lines for Port C. PC7 is the MSB, PC0 the LSB.
17, 19, 21, 23, 25, 27, 29, 31	PB7 through PB0	DIGGND	Bidirectional data lines for Port B. PB7 is the MSB, PB0 the LSB.
33, 35, 37, 39, 41, 43, 45, 47	PA7 through PA0	DIGGND	Bidirectional data lines for Port A. PA7 is the MSB, PA0 the LSB.
49	+5 V	DIGGND	This pin provides +5 VDC.
All even-numbered pins	DIGGND		These signals are connected to the PC ground signal.

Appendix D
AT-MIO-16D I/O Connector

This appendix describes the pinout and signal names for the AT-MIO-16D 100-pin I/O connector.

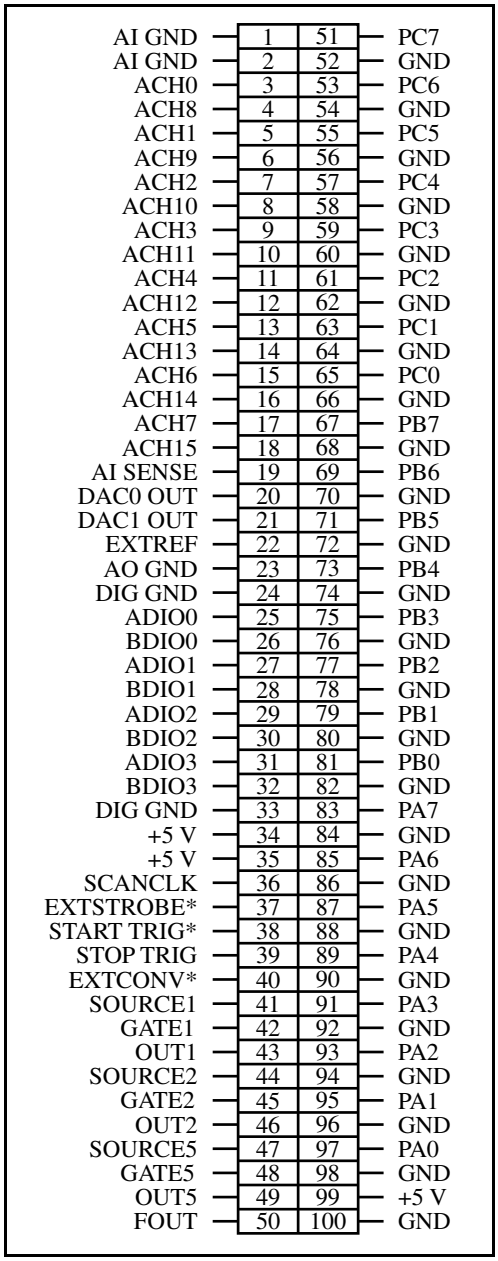


Figure D-1. AT-MIO-16D I/O Connector

Detailed signal specifications are included in Chapter 2, *Configuration and Installation*.

Appendix E
AMD Am9513A Data Sheet*

This appendix contains the manufacturer data sheet for the Am9513A System Controller integrated circuit (Advanced Micro Devices, Inc.). This device is used on the AT-MIO-16D.

Am9513A

System Timing Controller

FINAL

DISTINCTIVE CHARACTERISTICS

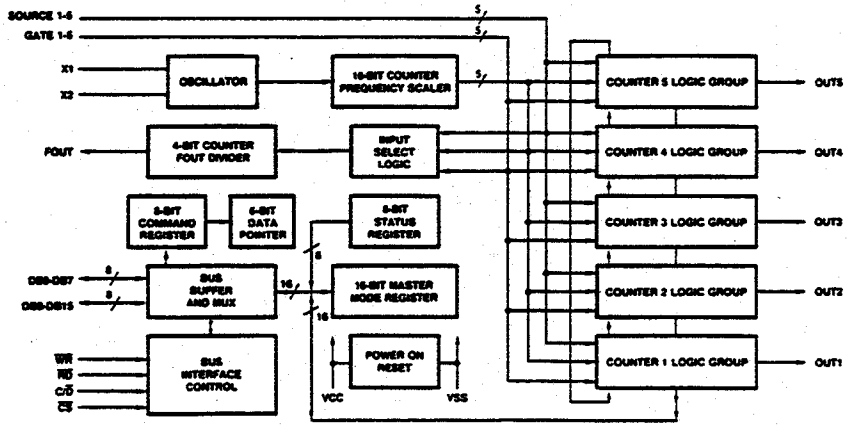
- Five independent 16-bit counters
- High speed counting rates
- Up/down and binary/BCD counting
- Internal oscillator frequency source
- Tapped frequency scaler
- Programmable frequency output
- 8-bit or 16-bit bus interface
- Time-of-day option
- Alarm comparators on counters 1 and 2
- Complex duty cycle outputs
- One-shot or continuous outputs
- Programmable count/gate source selection
- Programmable input and output polarities
- Programmable gating functions
- Retriggering capability
- +5 volt power supply
- Standard 40-pin package
- SMD/DESC qualified

GENERAL DESCRIPTION

The Am9513A System Timing Controller is an LSI circuit designed to service many types of counting, sequencing and timing applications. It provides the capability for programmable frequency synthesis, high resolution programmable duty cycle waveforms, retriggerable digital one-shots, time-of-day clocking, coincidence alarms, complex pulse generation, high resolution baud rate generation, frequency shift keying, stop-watching timing, event count accumulation, waveform analysis, etc. A variety of programmable operating modes and control features allows the Am9513A to be personalized for particular applications as well as dynamically reconfigured under program control.

The STC includes five general-purpose 16-bit counters. A variety of internal frequency sources and external pins may be selected as inputs for individual counters with software selectable active-high or active-low input polarity. Both hardware and software gating of each counter is available. Three-state outputs for each counter provide pulses or levels and can be active-high or active-low. The counters can be programmed to count up or down in either binary or BCD. The host processor may read an accumulated count at any time without disturbing the counting process. Any of the counters may be internally concatenated to form any effective counter length up to 80 bits.

BLOCK DIAGRAM

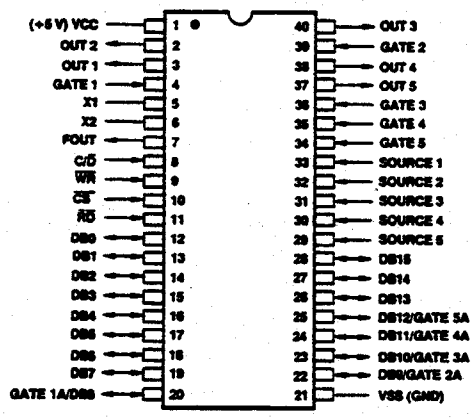


BD003381

Publication # 01731 | Rev. F | Amendment /0
Issue Date: August 1989

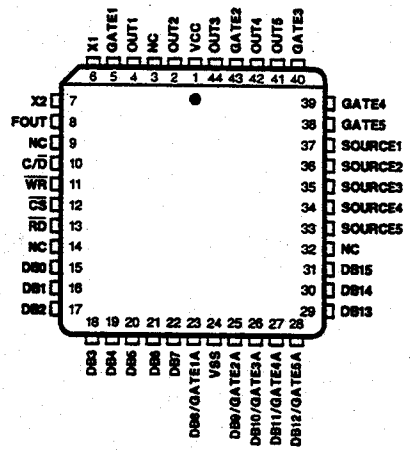
CONNECTION DIAGRAMS
Top View

DIP



CD005212

LCC & PLCC



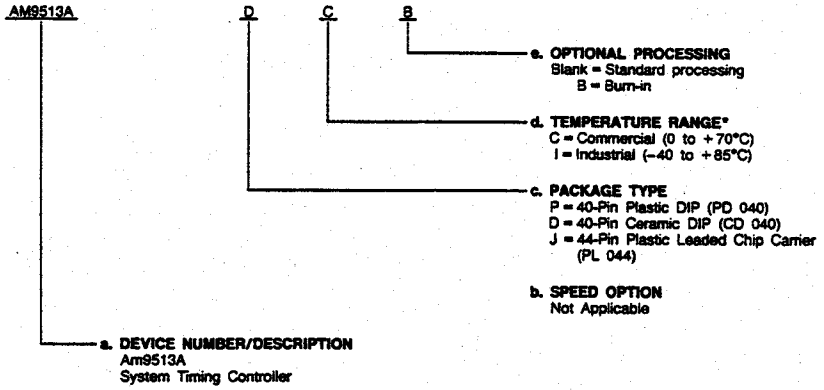
CD009912

Note: Pin 1 is marked for orientation.

2

ORDERING INFORMATION
Standard Products

AMD standard products are available in several packages and operating ranges. The order number (Valid Combination) is formed by a combination of: a. **Device Number**
b. **Speed Option** (if applicable)
c. **Package Type**
d. **Temperature Range**
e. **Optional Processing**



Valid Combinations	
AM9513A	PC, DC, DCB, DIB, JC

*This device is also available in Military temperature range.

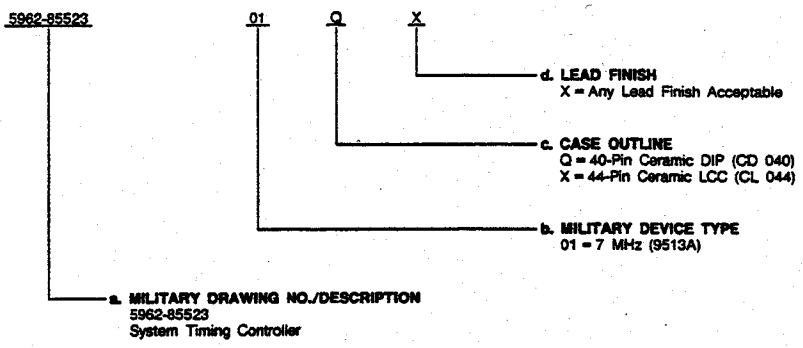
Valid Combinations

Valid Combinations list configurations planned to be supported in volume for this device. Consult the local AMD sales office to confirm availability of specific valid combinations, to check on newly released combinations, and to obtain additional data on AMD's standard military grade products.

ORDERING INFORMATION (continued)

Standard Military Drawing (SMD)/DESC Products

AMD products for Aerospace and Defense applications are available in several packages and operating ranges. Standard Military Drawing (SMD)/DESC products are fully compliant with MIL-STD-883C requirements. The order number (Valid Combination) for SMD/DESC products is formed by a combination of: **a. Military Drawing Part Number**
b. Device Type
c. Case Outline
d. Lead Finish



Valid Combinations	
5962-8552301	QX, XX

Valid Combinations

Valid Combinations list configurations planned to be supported in volume for this device. Consult the local AMD sales office to confirm availability of specific valid combinations or to check for newly released valid combinations.

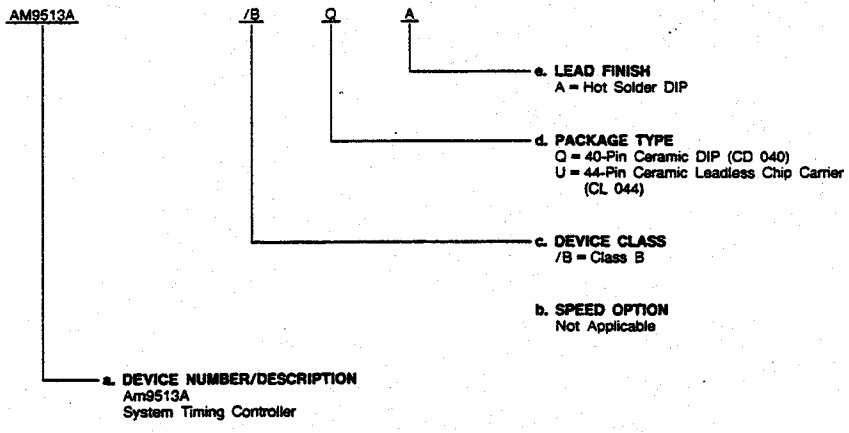
Group A Tests

Group A tests consist of Subgroups 1, 2, 3, 7, 8, 9, 10, 11.

ORDERING INFORMATION (continued)

APL Products

AMD products for Aerospace and Defense applications are available in several packages and operating ranges. APL (Approved Products List) products are fully compliant with MIL-STD-883C requirements. The order number (Valid Combination) for APL products is formed by a combination of: **a. Device Number**
b. Speed Option (if applicable)
c. Device Class
d. Package Type
e. Lead Finish



Valid Combinations	
AM9513A	/BQA, /BUA

Valid Combinations

Valid Combinations list configurations planned to be supported in volume for this device. Consult the local AMD sales office to confirm availability of specific valid combinations or to check for newly released valid combinations.

Group A Tests

Group A tests consist of
Subgroups 1, 2, 3, 7, 8, 9, 10, 11.

2

PIN DESCRIPTION

Pin No.	Name	I/O	Description
1	VCC		+5 V Power Supply.
21	VSS		Ground.
5, 6	X1, X2	O, I	(Crystal). X1 and X2 are the connections for an external crystal used to determine the frequency of the internal oscillator. The crystal should be a parallel-resonant, fundamental-mode type. An RC or LC or other reactive network may be used instead of a crystal. For driving from an external frequency source, X1 should be left open and X2 should be connected to a TTL source and a pull-up resistor.
7	FOUT	O	(Frequency Out). The FOUT output is derived from a 4-bit counter that may be programmed to divide its input by any integer value from 1 through 16 inclusive. The input to the counter is selected from any of 15 sources, including the internal scaled oscillator frequencies. FOUT may be gated on and off under software control and when off will exhibit a low impedance to ground. Control over the various FOUT options resides in the Master Mode register. After power-up, FOUT provides a frequency that is 1/16 that of the oscillator. The input source on power-up is F1.
4, 39, 36 – 34	GATE1 – GATE5	I	(Gate). The Gate inputs may be used to control the operations of individual counters by determining when counting may proceed. The same Gate input may control up to three counters. Gate pins may also be selected as count sources for any of the counters and for the FOUT divider. The active polarity for a selected Gate input is programmed at each counter. Gating function options allow level-sensitive gating or edge-initiated gating. Other gating modes are available including one that allows the Gate input to select between two counter output frequencies. All gating functions may also be disabled. The active Gate input is conditioned by an auxiliary input when the unit is operating with an external 8-bit data bus. See Data Bus description. Schmitt-trigger circuitry on the GATE inputs allows slow transition times to be used.
33 – 29	SRC1 – SRC5	I	(Source).The Source inputs provide external signals that may be counted by any of the counters. Any Source line may be routed to any or all of the counters and the FOUT divider. The active polarity for a selected SRC input is programmed at each counter. Any duty cycle waveform will be accepted as long as the minimum pulse width is at least half the period of the maximum specified counting frequency for the part. Schmitt-trigger circuitry on the SRC inputs allows slow transition times to be used.
3, 2, 40, 38, 37	OUT1 – OUT5	O	(Counter). Each 3-state OUT signal is directly associated with a corresponding individual counter. Depending on the counter configuration, the OUT signal may be a pulse, a square wave, or a complex duty cycle waveform. OUT pulse polarities are individually programmable. The output circuitry detects the counter state that would have been all bits zero in the absence of a reinitialization. That information is used to generate the selected waveform type. An optional output mode for Counters 1 and 2 overrides the normal output mode and provides a true OUT signal when the counter contents match the contents of an Alarm register.
12 – 19, 20, 22 – 28	DB0 – DB7, DB8 – DB15	I/O	(Data Bus). The 16 bidirectional Data Bus lines are used for information exchanges with the host processor. HIGH on a Data Bus line corresponds to one and LOW corresponds to zero. These lines act as inputs when $\overline{WR}$ and $\overline{CS}$ are active and as outputs when $\overline{RD}$ and $\overline{CS}$ are active. When $\overline{CS}$ is inactive, these pins are placed in a high-impedance state. After power-up or reset, the data bus will be configured for 8-bit width and will use only DB0 through DB7. DB0 is the least significant and DB7 is the most significant bit position. The data bus may be reconfigured for 16-bit width by changing a control bit in the Master Mode register. This is accomplished by writing an 8-bit command into the low-order DB lines while holding the DB13 – DB15 lines at a logic high level. Thereafter, all 16 lines can be used, with DB0 as the least significant and DB15 as the most significant bit position. When operating in the 8-bit data bus environment, DB8 – DB15 will never be driven active by the Am9513A. DB8 through DB12 may optionally be used as additional Gate inputs (see Figure 2). If unused, they should be held HIGH. When pulled LOW, a GATENA signal will disable the action of the corresponding counter N gating. DB13 – DB15 should be held HIGH in 8-bit bus mode whenever $\overline{CS}$ and $\overline{WR}$ are simultaneously active.
10	$\overline{CS}$	I	(Chip Select). The active-low Chip Select input enables Read and Write operations on the data bus. When Chip Select is HIGH, the Read and Write inputs are ignored. The first Chip Select signal after power-up is used to clear the power-on reset circuitry. If Chip Select is tied to ground permanently, the power-on reset circuitry may not function. In such a configuration, the software reset command must be issued following power-up to reset the Am9513A.
11	$\overline{RD}$	I	(Read). The active-low Read signal is conditioned by Chip Select and indicates that internal information is to be transferred to the data bus. The source will be determined by the port being addressed and, for Data Port reads, by the contents of the Data Pointer register. $\overline{WR}$ and $\overline{RD}$ should be mutually exclusive.
9	$\overline{WR}$	I	(Write). The active-low Write signal is conditioned by Chip Select and indicates that data bus information is to be transferred to an internal location. The destination will be determined by the port being addressed and, for Data Port writes, by the contents of the Data Pointer register. $\overline{WR}$ and $\overline{RD}$ should be mutually exclusive.
8	C/$\overline{D}$	I	(Control/Data). The Control/Data signal selects source and destination locations for Read and Write operations on the data bus. Control Write operations load the Command register and the Data Pointer. Control Read operations output the Status register. Data Read and Data Write transfers communicate with all other internal registers. Indirect addressing at the data port is controlled internally by the Data Pointer register.

Signal	Abbreviation	Type	Pins
+5 Volts	VCC	Power	1
Ground	VSS	Power	1
Crystal	X1, X2	O, I	2
Read	$\overline{RD}$	Input	1
Write	$\overline{WR}$	Input	1
Chip Select	$\overline{CS}$	Input	1
Control/Data	C/$\overline{D}$	Input	1
Source N	SRC	Input	5
Gate N	GATE	Input	5
Data Bus	DB	I/O	16
Frequency Out	FOUT	Output	1
Out N	OUT	Output	5

Figure 1. Interface Signal Summary

Figure 1 summarizes the interface signals and their abbreviations for the STC.

Package Pin	Data Bus Width (MM14)	
	16 Bits	8 Bits
12	DB0	DB0
13	DB1	DB1
14	DB2	DB2
15	DB3	DB3
16	DB4	DB4
17	DB5	DB5
18	DB6	DB6
19	DB7	DB7
20	DB8	GATE 1A
22	DB9	GATE 2A
23	DB10	GATE 3A
24	DB11	GATE 4A
25	DB12	GATE 5A
26	DB13	(VIH)
27	DB14	(VIH)
28	DB15	(VIH)

Figure 2. Data Bus Assignments

Interface Considerations

All of the input and output signals for the Am9513A are specified with logic levels compatible with those of standard TTL circuits. In addition to providing TTL compatible voltage levels, other output conditions are specified to help configure non-standard interface circuitry. The logic level specifications take into account all worst-case combinations of the three variables that affect the logic level thresholds: ambient temperature, supply voltage and processing parameters. A change in any of these toward nominal values will improve the actual operating margins and will increase noise immunity.

Unprotected open gate inputs of high quality MOS transistors exhibit very high resistances on the order of perhaps 10^{14} ohms. It is easy, therefore, in some circumstances, for charge to enter the gate node of such an input faster than it can be discharged and consequently, for the gate voltage to rise high enough to break down the oxides and destroy the transistor.

All inputs to the Am9513A include protection networks to help prevent damaging accumulations of static charge. The protection circuitry is designed to slow the transitions of incoming current surges and to provide low-impedance discharge paths for voltages beyond the normal operating levels. Note, however, that input energy levels can nonetheless be too high to be successfully absorbed. Conventional design, storage, and handling precautions should be observed so that the protection networks themselves are not overstressed.

Within the limits of normal operation, the input protection circuitry is inactive and may be modeled as a lumped series RC as shown in Figure 3(a). The functionally active input connection during normal operation is the gate of a MOS transistor. No active sources or drains are connected to the inputs so that neither transient nor steady-state currents are impressed on the driving signals other than the charging or discharging of the input capacitance and the accumulated leakage associated with the protection network and the input circuit.

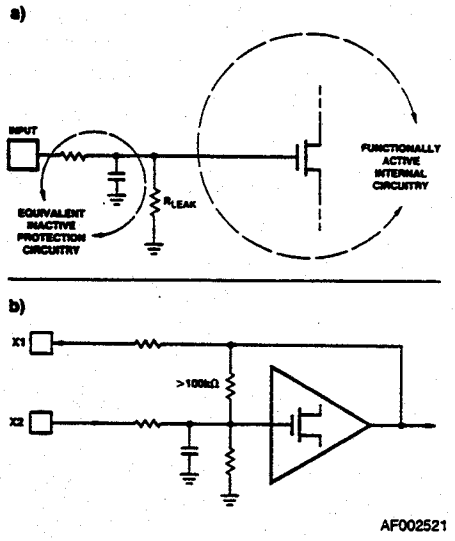


Figure 3. Input Circuitry

The only exception to the purely capacitive input case is the X2 crystal input. As shown in Figure 3(b) an internal resistor connects X1 and X2 in addition to the protection network. The resistor is a modestly high value of more than 100kohms.

Fanout from the driving circuitry into the Am9513A inputs will generally be limited by transition time considerations rather than DC current limitations when the loading is dominated by conventional MOS circuits. In an operating environment, all inputs should be terminated so they do not float and therefore will not accumulate stray static charges. Unused inputs should be tied directly to Ground or VCC, as appropriate. An input in use will have some type of logic output driving it, and termination during operation will not be a problem. Where inputs are driven from logic external to the card containing this chip, however, on-board termination should be provided to protect the chip when the board is unplugged (the input would otherwise float). A pull-up resistor or a simple inverter or gate will suffice.

DETAILED DESCRIPTION

The Am9513A System Timing Controller (STC) is a support device for processor oriented systems that is designed to enhance the available capability with respect to counting and timing operations. It provides the capability for programmable frequency synthesis, high resolution programmable duty cycle waveforms, retriggerable digital timing functions, time-of-day clocking, coincidence alarms, complex pulse generation, high resolution baud rate generation, frequency shift keying, stop-watching timing, event count accumulation, waveform analysis and many more. A variety of programmable operating modes and control features allow the Am9513A to be personalized for particular applications as well as dynamically reconfigured under program control.

The STC includes five general-purpose 16-bit counters. A variety of internal frequency sources and external pins may be selected as inputs for individual counters with software selectable active-high or active-low input polarity. Both hardware and software gating of each counter is available. Three-state outputs for each counter provide either pulses or levels. The counters can be programmed to count up or down in either binary or BCD. The accumulated count may be read without disturbing the counting process. Any of the counters may be internally concatenated to form an effective counter length of up to 80 bits.

The Am9513A block diagrams indicate the interface signals and the basic flow of information. Internal control lines and the internal data bus have been omitted. The control and data registers are all connected to a common internal 16-bit bus. The external bus may be 8- or 16-bits wide; in the 8-bit mode, the internal 16-bit information is multiplexed to the low order data bus pins DB0 through DB7.

An internal oscillator provides a convenient source of frequencies for use as counter inputs. The oscillator's frequency is controlled at the X1 and X2 interface pins by an external reactive network such as a crystal. The oscillator output is divided by the Frequency Scaler to provide several sub-frequencies. One of the scaled frequencies (or one of ten input signals) may be selected as an input to the FOUT divider and then comes out of the chip at the FOUT interface pin.

The STC is addressed by the external system as two locations: a Control port and a Data port. The Control port provides direct access to the Status and Command registers, as well as allowing the user to update the Data Pointer register. The Data port is used to communicate with all other addressable internal locations. The Data Pointer register controls the Data port addressing.

Among the registers accessible through the Data port are the Master Mode register and five Counter Mode registers, one for each counter. The Master Mode register controls the programmable options that are not controlled by the Counter Mode registers.

Each of the five general-purpose counters is 16-bits long and is independently controlled by its Counter Mode register. Through this register, a user can software select one of 16 sources as the counter input, a variety of gating and repetition modes, up or down counting in binary or BCD and active-high or active-low input and output polarities.

Associated with each counter are a Load register and a Hold register, both accessible through the Data port. The Load register is used to automatically reload the counter to any predefined value, thus controlling the effective count period. The Hold register is used to save count values without disturbing the count process, permitting the host processor to read intermediate counts. In addition, the Hold register may be used as a second Load register to generate a number of complex output waveforms.

All five counters have the same basic control logic and control registers. Counters 1 and 2 have additional Alarm registers and comparators associated with them, plus the extra logic necessary for operating in a 24-hour time-of-day mode. For real-time operation, the time-of-day logic will accept 50Hz, 60Hz or 100Hz input frequencies.

Each general counter has a single dedicated output pin. It may be turned off when the output is not of interest or may be configured in a variety of ways to drive interrupt controllers, Darlington buffers, bus drivers, etc. The counter inputs, on the other hand, are specifically not dedicated to any given interface line. Considerable versatility is available for configuring both the input and the gating of individual counters. This not only permits dynamic reassignment of inputs under software control, but also allows multiple counters to use a single input and a single gate pin to control more than one counter. Indeed, a single pin can be the gate for one counter and, at the same time, the count source for another.

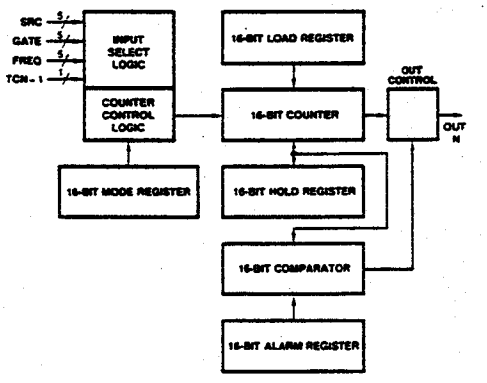


Figure 4. Counter Logic Groups 1 and 2

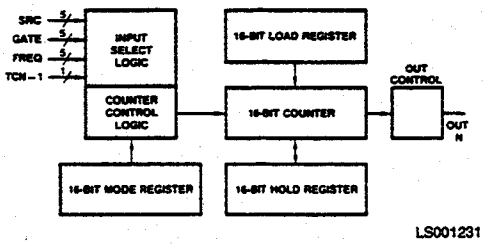


Figure 5. Counter Logic Groups 3, 4 and 5

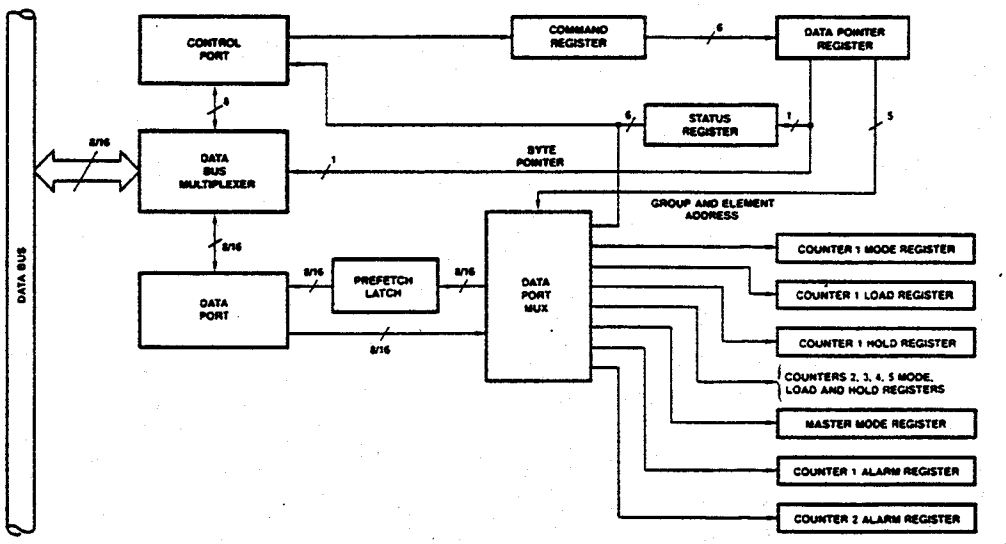


Figure 6. Am9513A Register Access

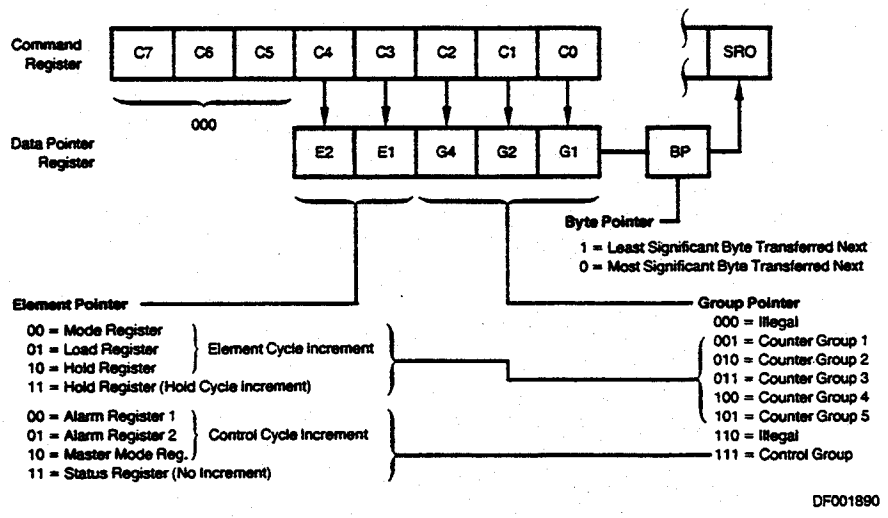


Figure 7. Data Pointer Register

2

	Element Cycle			Hold Cycle
	Mode Register	Load Register	Hold Register	Hold Register
Counter 1	FF01	FF09	FF11	FF19
Counter 2	FF02	FF0A	FF12	FF1A
Counter 3	FF03	FF0B	FF13	FF1B
Counter 4	FF04	FF0C	FF14	FF1C
Counter 5	FF05	FF0D	FF15	FF1D
Master Mode Register = FF17				
Alarm 1 Register = FF07				
Alarm 2 Register = FF0F				
Status Register = FF1F				

Notes:
1. All codes are in hex.
2. When used with an 8-bit bus, only the two low order hex digits should be written to the command port; the "FF" prefix should be used only for a 16-bit data bus interface.

Figure 8. Load Data Pointer Commands

Sequencing is enabled by clearing Master Mode bit 14 (MM14) to zero. As shown in Figure 9 several types of sequencing are available depending on the data bus width being used and the initial Data Pointer value entered by command.

When E1 = 0 or E2 = 0 and G4, G2, G1 points to a Counter Group, the Data Pointer will proceed through the Element cycle. The Element field will automatically sequence through the three values 00, 01 and 10 starting with the value entered. When the transition from 10 to 00 occurs, the Group field will also be incremented by one. Note that the Element field in this case does not sequence to a value of 11. The Group field circulates only within the five Counter Group codes.

If E2, E1 = 11 and a Counter Group are selected, then only the Group field is sequenced. This is the Hold cycle. It allows the Hold registers to be sequentially accessed while bypassing the Mode and Load registers. The third type of sequencing is the Control cycle. If G4, G2, G1 = 111 and E2, E1 ≠ 11, the Element Pointer will be incremented through the values 00, 01 and 10, with no change to the Group Pointer.

When G4, G2, G1 = 111 and E2, E1 = 11, no incrementing takes place and only the Status register will be available through the Data port. Note that the Status register can also always be read directly through the Control port.

For all these auto-sequencing modes, if an 8-bit data bus is used, the Byte pointer will toggle after every data transfer to allow the least and most significant bytes to be transferred before the Element or Group fields are incremented.

Prefetch Circuit

To minimize the read access time to internal Am9513A registers, a prefetch circuit is used for all read operations through the Data port. Following each read or write operation through the Data port, the Data Pointer register is updated to point to the next register to be accessed. Immediately following this update, the new register data is transferred to a special prefetch latch at the interface pad logic. When the user performs a subsequent read of the Data port, the data bus drivers are enabled, outputting the prefetched data on the bus. Since the internal data register is accessed prior to the start of the read operation, its access time is transparent to the user. To keep the prefetched data consistent with the Data Pointer, prefetches are also performed after each write to the Data port and after execution of the "Load Data Pointer" command. The following rules should be kept in mind regarding Data port Transfers.

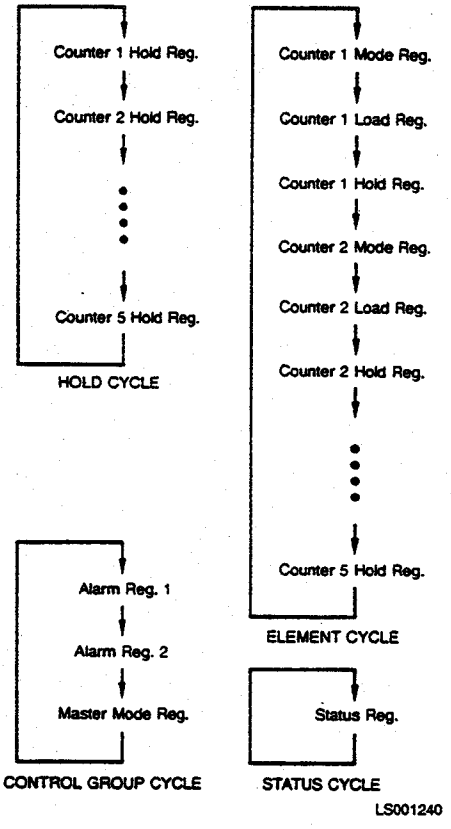


Figure 9. Data Pointer Sequencing

1. The Data Pointer register should always be reloaded before reading from the Data port if a command, other than "Load Data Pointer," was issued to the Am9513A following the last Data port read or write. The Data Pointer does not have to be loaded again if the first Data port transaction after a command entry is a write, since the Data port write will automatically cause a new prefetch to occur.
2. Operating modes N, O, Q, R and X allow the user to save the counter contents in the Hold register by applying an active-going gate edge. If the Data Pointer register had been pointing to the Hold register in question, the prefetched value will not correspond to the new value saved in the Hold Register. To avoid reading an incorrect value, a new "Load Data Pointer" command should be issued before attempting to read the saved data. A Data port write (to another register) will also initiate a prefetch; subsequent reads will access the recently saved Hold register data. Many systems will use the "saving" gate edge to interrupt the host CPU. In systems such as this, the interrupt service routine should issue a "Load Data Pointer" command prior to reading the saved data.

Status Register

The 8-bit read-only Status register indicates the state of the Byte Pointer bit in the Data Pointer register and the state of the

OUT signal for each of the general counters. See Figures 10 and 17. The OUT signals reported are those internal to the chip after the polarity-select logic and just before the three-state interface buffer circuitry. Bits SR6 and SR7 may be 0 or 1.

The Status register OUT bit reflects an active-high or active-low TC output or a TC Toggled output, as programmed in the Output Control Field of the Counter Mode register. That is, it reflects the exact state of the OUT pin. When the low-impedance to Ground Output option (CM2-CM0 = 000) is selected, the Status register will reflect an active-high TC Output. When a high-impedance Output option (CM2-CM0 = 100) is selected, the Status register will reflect an active-low TC output.

For Counters 1 and 2, the OUT pin will reflect the comparator output if the comparators are enabled. The Status register bit and OUT pin are active-high if CM2 = 0 and active-low if CM2 = 1. When the high-impedance option is selected and the comparator is enabled, the status register bit will reflect an active-high comparator output. When the low-impedance to Ground option is selected and the comparator is enabled, the status register bit will reflect an active-low comparator output.

The Status register is normally accessed by reading the Control port (see Figure 6) but may also be read via the Data port as part of the Control Group.

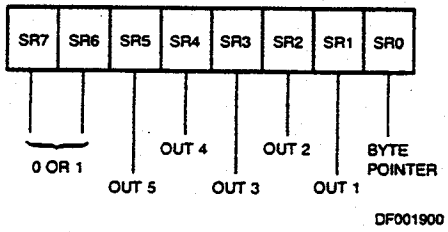


Figure 10. Status Register Bit Assignments

DATA PORT REGISTERS

Counter Logic Groups

As shown in Figures 4 and 5, each of the five Counter Logic Groups consists of a 16-bit general counter with associated control and output logic, a 16-bit Load register, a 16-bit Hold register and a 16-bit Mode register. In addition, Counter Groups 1 and 2 also include 16-bit Comparators and 16-bit Alarm registers. The comparator/alarm functions are controlled by the Master Mode register. The operation of the Counter Mode registers is the same for all five counters. The host CPU has both read and write access to all registers in the Counter Logic Groups through the Data port. The counter itself is never directly accessed.

Load Register

The 16-bit read/write Load register is used to control the effective length of the general counter. Any 16-bit value may be written into the Load register. That value can then be transferred into the counter each time the Terminal Count (TC) occurs. "Terminal Count" is defined as that period of time when the counter contents would have been zero if an external value had not been transferred into the counter. Thus, the terminal count frequency can be the input frequency divided by the value in the Load register. In all operating modes, either the Load or Hold register will be transferred into the counter when TC occurs. In cases where values are being accumulated in the counter, the Load register action can become transparent by filling the Load register with all zeros.

Hold Register

The 16-bit read/write Hold register is dual-purpose. It can be used in the same way as the Load register, thus offering an alternate source for module definition for the counter. The Hold register may also be used to store accumulated counter values for later transfer to the host processor. This allows the count to be sampled while the counting process proceeds without interruption. Transfer of the counter contents into the Hold register is accomplished by the hardware interface in some operating modes or by software commands at any time.

Counter Mode Register

The 16-bit read/write Counter Mode register controls the gating, counting, output and source select functions within each Counter Logic Group. The "Counter Mode Control Options" section of this document describes the detailed control options available. Figure 16 shows the bit assignments for the Counter Mode registers.

Alarm Registers and Comparators

Added functions are available in the Counter Logic Groups for Counters 1 and 2 (see Figure 4). Each contains a 16-bit Alarm register and a 16-bit Comparator. When the value in the counter reaches the value in the Alarm register, the Comparator output will go true. The Master Mode register contains control bits to individually enable/disable the comparators. When enabled, the comparator output appears on the OUT pin of the associated counter in place of the normal counter output. The output will remain true as long as the comparison is true, that is, until the next input causes the count to change. The polarity of the Comparator output will be active-high if the Output Control field of the Counter Mode register is 001 or 010 and active-low if the Output Control field is 101.

MASTER MODE CONTROL OPTIONS

The 16-bit Master Mode (MM) register is used to control those internal activities that are not controlled by the individual Counter Mode registers. This includes frequency control, Time-of-Day operation, comparator controls, data bus width and data pointer sequencing. Figure 11 shows the bit assignments for the Master Mode register. This section describes the use of each control field.

Master Mode register bits MM12, MM13 and MM14 can be individually set and reset using commands issued to the Command register. In addition, they can all be changed by writing directly to the Master Mode register.

After power-on reset or a Master Reset command, the Master Mode register is cleared to an all zero condition. This results in the following configuration:

Time-of-Day disabled
Both Comparators disabled
FOUT Source is frequency F1
FOUT Divider set for divide-by-16
FOUT gated on
Data Bus 8 bits wide
Data Pointer Sequencing enabled
Frequency Scaler divides in binary

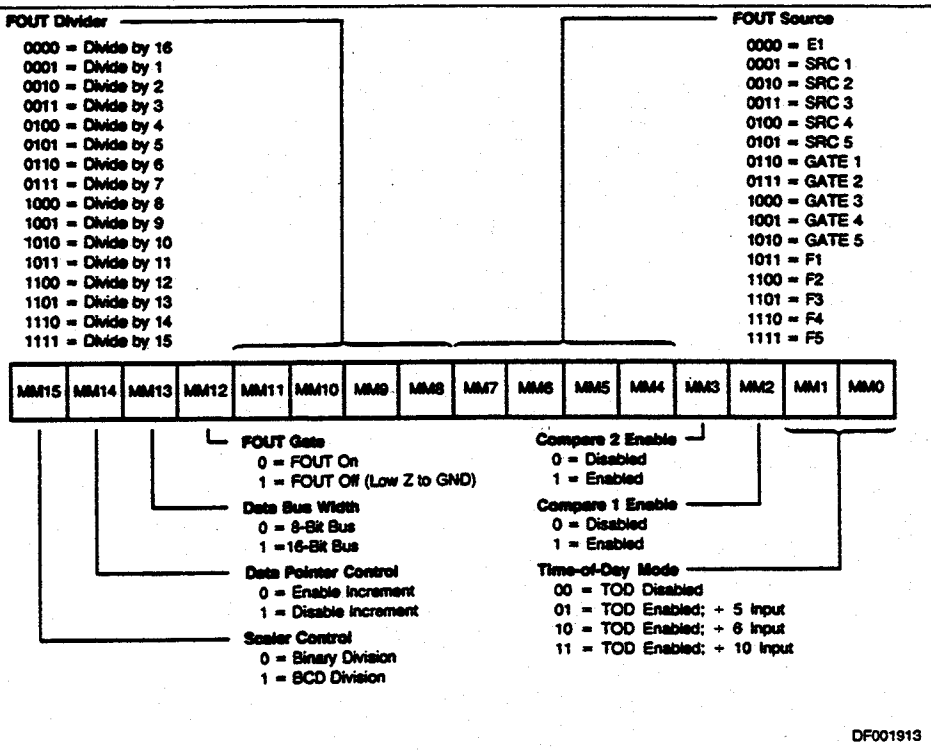


Figure 11. Master Mode Register Bit Assignments

Time-of-Day

Bits MM0 and MM1 of the Master Mode register specify the Time-of-Day (TOD) options. When MM0 = 0 and MM1 = 0, the special logic used to implement TOD is disabled, and Counters 1 and 2 will operate in exactly the same way as Counters 3, 4 and 5. When MM0 = 1 or MM1 = 1, additional counter decoding and control logic is enabled on Counters 1 and 2, which causes their decades to turn over at the counts that generate appropriate 24-hour TOD accumulations. For additional information, see the Time-of-Day chapter in the 9513A System timing controller technical manual.

Comparator Enable

Bits MM2 and MM3 control the Comparators associated with Counters 1 and 2. When a Comparator is enabled, its output is substituted for the normal counter output on the associated OUT1 or OUT2 pin. The comparator output will be active-high if the output control field of the Counter Mode register is 001 or 010 and active-low for a code of 101. Once the compare output is true, it will remain so until the count changes and the comparison therefore goes false.

The two Comparators can always be used individually in any operating mode. One special case occurs when the Time-of-Day option is revoked and both Comparators are enabled. The operation of Comparator 2 will then be conditioned by Comparator 1 so that a full 32-bit compare must be true in order to generate a true signal on OUT2. OUT1 will continue, as usual, to reflect the state of the 16-bit comparison between Alarm 1 and Counter 1.

FOUT Source

Master Mode bits MM4 through MM7 specify the source input for the FOUT divider. Fifteen inputs are available for selection, and they include the five Source pins, the five Gate pins and the five internal frequencies derived from the oscillator. The 16th combination of the four control bits (all zeros) is used to assure that an active frequency is available at the input to the FOUT divider following reset.

FOUT Divider

Bits MM8 through MM11 specify the dividing ratio for the FOUT Divider. The FOUT source (selected by bits MM4 through MM7) is divided by an integer value between 1 and 16, inclusive, and is then passed to the FOUT output buffer. After power-on or reset, the FOUT divider is set to divide-by-16.

FOUT Gate

Master Mode bit MM12 provides a software gating capability for the FOUT signal. When MM12 = 1, FOUT is off and in a low-impedance state to ground. MM12 may be set or cleared in conjunction with the loading of the other bits in the Master Mode register; alternatively, there are commands that allow MM12 to be individually set or cleared directly without changing any other Master Mode bits. After power-up or reset, FOUT is gated on.

When changing the FOUT divider ratio or FOUT source, transient pulses as short as half the period of the FOUT source may appear on the FOUT pin. Turning the FOUT gate on or off can also generate a transient. This should be considered when using FOUT as a system clock source.

Bus Width

Bit MM13 controls the multiplexer at the data bus interface in order to configure the part for an 8-bit or 16-bit external bus. The internal bus is always 16-bits wide. When MM13 = 1, 16-bit data is transferred directly between the internal bus and all 16 of the external bus lines. In this configuration, the Byte Pointer bit in the Data Pointer register remains set at all times. When MM13 = 0, 16-bit internal data is transferred a byte at a time to and from the eight low-order external data bus lines. The Byte Pointer bit toggles with each byte transfer in this mode.

When the Am9513A is set to operate with an 8-bit data bus width, pins DB8 through DB15 are not used for the data bus and are available for other functions. Pins DB13 through DB15 should be tied high. Pins DB8 through DB12 are used as auxiliary gating inputs and are labeled GATE1A through GATE5A respectively. The auxiliary gate pin, GATENA, is logically ANDed with the gate input to Counter N, as shown in Figure 12. The output of the AND gate is then used as the gating signal for Counter N.

Data Pointer Sequencing

Bit MM14 controls the Data Pointer logic to enable or disable the automatic sequencing functions. When MM14 = 1, the contents of the Data Pointer can be changed only directly by entering a command. When MM14 = 0, several types of automatic sequencing of the Data Pointer are available. These are described in the Data Pointer register section of this document.

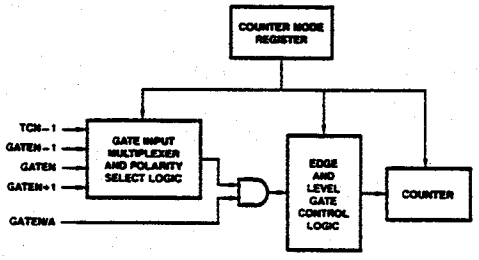
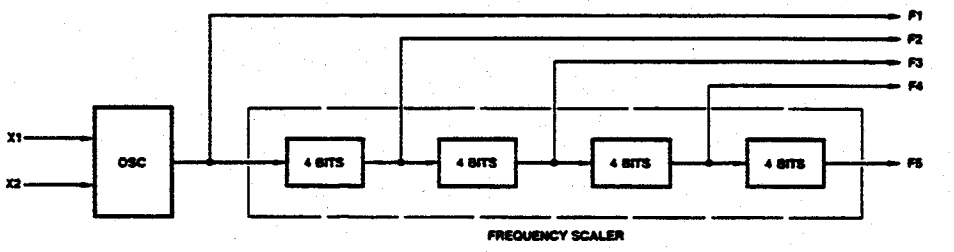


Figure 12. Gating Control

Thus, the host processor, by controlling MM14, may repetitively read/write a single internal location, or may sequentially read/write groups of locations. Bit MM14 can be loaded by writing to the Master Mode register or can be set or cleared by software command.

Scaler Ratios

Master Mode bit MM15 controls the counting configuration of the Frequency Scaler counter. When MM15 = 0, the Scaler divides the oscillator frequency in binary steps so that each subfrequency is 1/16 of the preceding frequency. When MM15 = 1, the Scaler divides in BCD steps so that adjacent frequencies are related by ratios of 10 instead of 16 (see Figure 13).



Frequency	BCD Scaling MM15 = 1	Binary Scaling MM15 = 0
F1	OSC	OSC
F2	F1 ÷ 10	F1 ÷ 16
F3	F1 ÷ 100	F1 ÷ 256
F4	F1 ÷ 1,000	F1 ÷ 4,096
F5	F1 ÷ 10,000	F1 ÷ 65,536

Figure 13. Frequency Scaler Ratios

Counter Mode	A	B	C	D	E	F	G	H	I	J	K	L
Special Gate (CM7)	0	0	0	0	0	0	0	0	0	0	0	0
Reload Source (CM6)	0	0	0	0	0	0	1	1	1	1	1	1
Repetition (CM5)	0	0	0	1	1	1	0	0	0	1	1	1
Gate Control (CM15–CM13)	000	LEVEL	EDGE	000	LEVEL	EDGE	000	LEVEL	EDGE	000	LEVEL	EDGE
Count to TC once, then disarm	X	X	X									
Count to TC twice, then disarm							X	X	X			
Count to TC repeatedly without disarming				X	X	X				X	X	X
Gate input does not gate counter input	X			X			X			X		
Count only during active gate level		X			X			X			X	
Start count on active gate edge and stop count on next TC			X			X						
Start count on active gate edge and stop count on second TC									X			X
No hardware retriggering	X	X	X	X	X	X	X	X	X	X	X	X
Reload counter from Load register on TC	X	X	X	X	X	X						
Reload counter on each TC, alternating reload source between Load and Hold registers							X	X	X	X	X	X
Transfer Load register into counter on each TC that gate is LOW, transfer Hold register into counter on each TC that gate is HIGH.												
On active gate edge transfer counter into Hold register and then reload counter from Load register												

Counter Mode	M	N	O	P	Q	R	S	T	U	V	W	X
Special Gate (CM7)	1	1	1	1	1	1	1	1	1	1	1	1
Reload Source (CM6)	0	0	0	0	0	0	1	1	1	1	1	1
Repetition (CM5)	0	0	0	1	1	1	0	0	0	1	1	1
Gate Control (CM15–CM13)	000	LEVEL	EDGE	000	LEVEL	EDGE	000	LEVEL	EDGE	000	LEVEL	EDGE
Count to TC once, then disarm		X	X									
Count to TC twice, then disarm							X					
Count to TC repeatedly without disarming					X	X				X		X
Gate input does not gate counter input							X			X		
Count only during active gate level		X			X							
Start count on active gate edge and stop count on next TC			X			X						X
Start count on active gate edge and stop count on second TC												
No hardware retriggering							X			X		X
Reload counter from Load register on TC		X	X		X	X						X
Reload counter on each TC, alternating reload source between Load and Hold registers.												
Transfer Load register into counter on each TC that gate is LOW, transfer Hold register into counter on each TC that gate is HIGH.							X			X		
On active gate edge transfer counter into Hold register and then reload counter from Load register		X	X		X	X						
On active gate edge transfer counter into Hold register, but counting continues												X

Notes: 1. Counter modes M, P, T, U and W are reserved and should not be used.
2. Mode X is available for Am9513A only.

Figure 14. Counter Mode Operating Summary

COUNTER MODE DESCRIPTIONS

Counter Mode register bits CM15–CM13 and CM7–CM5 select the operating mode for each counter (see Figure 14). To simplify references to a particular mode, each mode is assigned a letter from A through X. Representative waveforms for the counter modes are illustrated in Figures 15a through 15v. (Because the letter suffix in the figure number is keyed to the mode, Figures 15m, 15p, 15t, 15u and 15w do not exist.) The figures assume down counting on rising source edges. Those modes which automatically disarm the counter (CM5 = 0) are shown with the $\overline{WR}$ plus entering the required ARM command; for modes which count repetitively (CM5 = 1), the ARM command is omitted. The retriggering modes (N, O, Q and R) are shown with one retrigger operation. Both a TC output waveform and a TC Toggled output waveform are shown for each mode. The symbols L and H are used to represent count values equal to the Load and Hold register contents, respectively. The symbols K and N represent arbitrary count values. For each mode, the required bit pattern in the Counter Mode register is shown; "don't care" bits are marked "X." These figures are designed to clarify the mode descriptions; the Am9513A Electrical Specification should be used as the authoritative reference for timing relationships between signals.

To keep the following mode descriptions concise and to the point, the phrase "source edges" is used to refer to active-going source edges only, not to inactive-going edges. Similarly, the phrase "gate edges" refers only to active-going gate edges. Also, again to avoid verbosity and euphuism, the descriptions of some modes state that a counter is stopped or disarmed "on a TC, inhibiting further counting." As is fully explained in the TC section of this document, for these modes the counter is actually stopped or disarmed following the active-going source edge which drives the counter out of TC. In other words, since a counter in the TC state always counts, irrespective of its gating or arming status, the stopping or disarming of the count sequence is delayed until TC is terminated.

MODE A

Software-Triggered Strobe with No Hardware Gating

CM15	CM14	CM13	CM12	CM11	CM10	CM9	CM8
0	0	0	X	X	X	X	X

CM7	CM6	CM5	CM4	CM3	CM2	CM1	CM0
0	0	0	X	X	X	X	X

Mode A, shown in Figure 15a, is one of the simplest operating modes. The counter will be available for counting source edges when it is issued an ARM command. On each TC, the counter will reload from the Load register and automatically disarm itself, inhibiting further counting. Counting will resume when a new ARM command is issued.

MODE B

Software-Triggered Strobe with Level Gating

CM15	CM14	CM13	CM12	CM11	CM10	CM9	CM8
LEVEL			X	X	X	X	X

CM7	CM6	CM5	CM4	CM3	CM2	CM1	CM0
0	0	0	X	X	X	X	X

Mode B, shown in Figure 15b, is identical to Mode A except that source edges are counted only when the assigned Gate is active. The counter must be armed before counting can occur. Once armed, the counter will count all source edges which occur while the Gate is active and disregard those edges which occur while the Gate is inactive. This permits the Gate to turn the count process on and off. On each TC the counter will reload from the Load register and automatically disarm itself, inhibiting further counting until a new ARM command is issued.

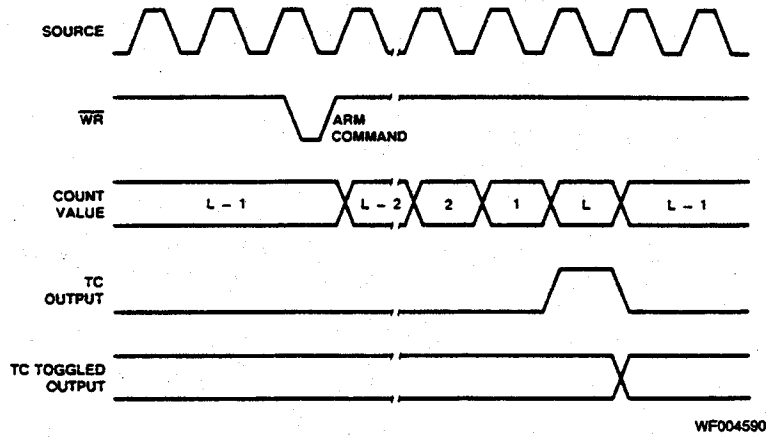


Figure 15a. Mode A Waveforms

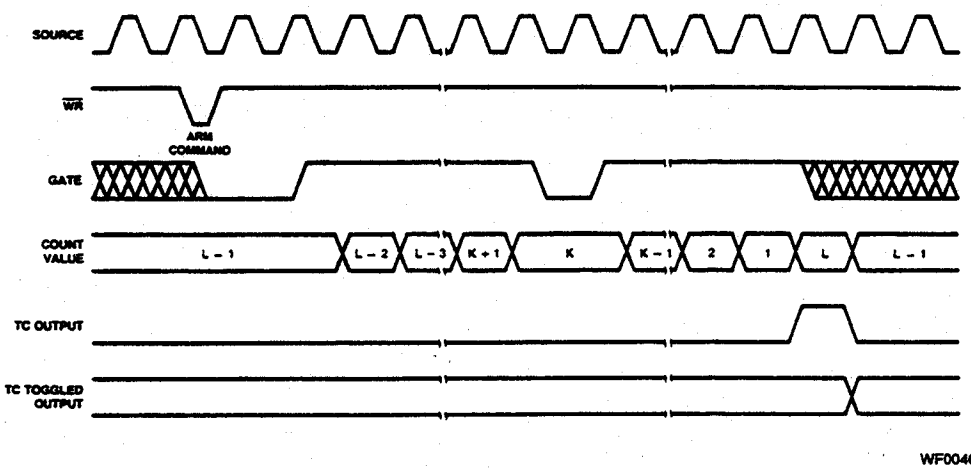


Figure 15b. Mode B Waveforms

MODE C

Hardware-Triggered Strobe

CM15	CM14	CM13	CM12	CM11	CM10	CM9	CM8
EDGE			X	X	X	X	X

CM7	CM6	CM5	CM4	CM3	CM2	CM1	CM0
0	0	0	X	X	X	X	X

Mode C, shown in Figure 15c, is identical to Mode A, except that counting will not begin until a Gate edge is applied to the armed counter. The counter must be armed before application of the triggered Gate edge; Gate edgec applied to a disarmed counter are disregarded. The counter will start counting on the first source edge after the triggering Gate edge and will continue counting until TC. At TC, the counter will reload from the Load register and automatically disarm itself. Counting will then remain inhibited until a new ARM command and a new Gate edge are applied in that order. Note that after application of a triggered Gate edge, the Gate input will be disregarded for the remainder of the count cycle. This differs from Mode B, where the Gate can be modulated throughout the count cycle to stop and start the counter.

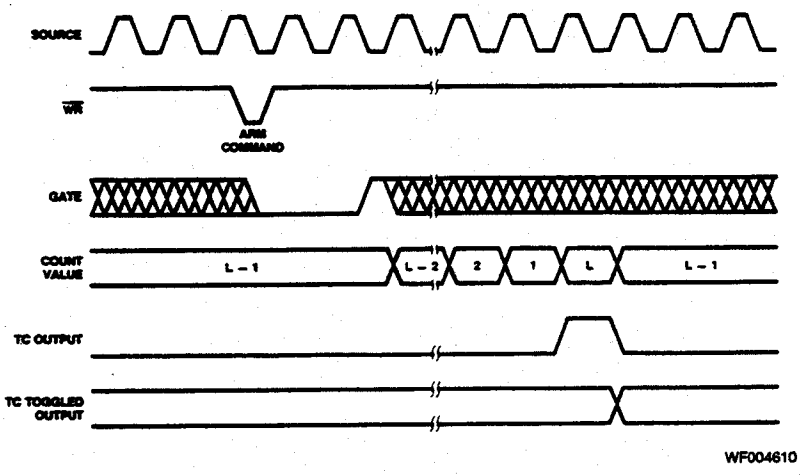


Figure 15c. Mode C Waveforms

MODE D

Rate Generator with No Hardware Gating

CM15	CM14	CM13	CM12	CM11	CM10	CM9	CM8
0	0	0	X	X	X	X	X

CM7	CM6	CM5	CM4	CM3	CM2	CM1	CM0
0	0	1	X	X	X	X	X

Mode D, shown in Figure 15d, is typically used in frequency generation applications. In this mode, the Gate input does not affect counter operation. Once armed, the counter will count to TC repetitively. On each TC, the counter will reload itself from the Load register; hence, the Load register value determines the time between TCs. A square wave rate generator may be obtained by specifying the TC Toggled output mode in the Counter Mode register.

MODE E

Rate Generator with Level Gating

CM15	CM14	CM13	CM12	CM11	CM10	CM9	CM8
LEVEL			X	X	X	X	X

CM7	CM6	CM5	CM4	CM3	CM2	CM1	CM0
0	0	1	X	X	X	X	X

Mode E, shown in Figure 15e, is identical to Mode D, except the counter will only count those source edges which occur while the Gate input is active. This feature allows the counting process to be enabled and disabled under hardware control. A square wave rate generator may be obtained by specifying the TC Toggled output mode.

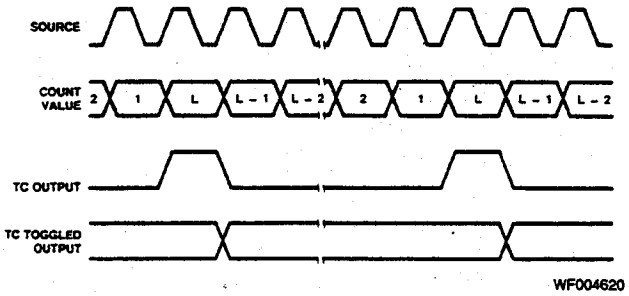


Figure 15d. Mode D Waveforms

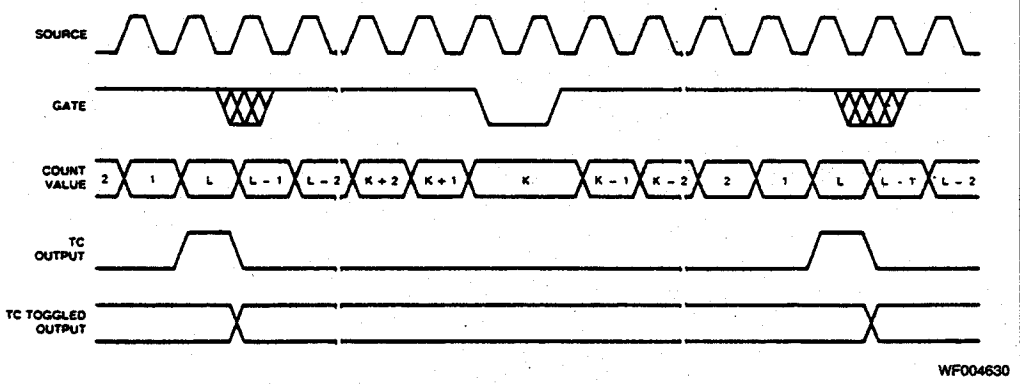


Figure 15e. Mode E Waveforms

MODE F

Non-Retriggerable One-Shot

CM15	CM14	CM13	CM12	CM11	CM10	CM9	CM8
EDGE			X	X	X	X	X

CM7	CM6	CM5	CM4	CM3	CM2	CM1	CM0
0	0	1	X	X	X	X	X

Mode F, shown in Figure 15f, provides a non-retriggerable one-shot timing function. The counter must be armed before it will function. Application of a Gate edge to the armed counter will enable counting. When the counter reaches TC, it will reload itself from the Load register. The counter will then stop counting, awaiting a new Gate edge. Note that unlike Mode C, a new ARM command is not needed after TC, only a new Gate edge. After application of a triggering Gate edge, the Gate input is disregarded until TC.

MODE G

Software-Triggered Delayed Pulse One-Shot

CM15	CM14	CM13	CM12	CM11	CM10	CM9	CM8
0	0	0	X	X	X	X	X

CM7	CM6	CM5	CM4	CM3	CM2	CM1	CM0
0	1	0	X	X	X	X	X

In Mode G, the Gate does not affect the counter's operation. Once armed, the counter will count to TC twice and then automatically disarm itself. For most applications, the counter will initially be loaded from the Load register either by a LOAD command or by the last TC of an earlier timing cycle. Upon counting to the first TC, the counter will reload itself from the Hold register. Counting will proceed until the second TC, when the counter will reload itself from the Load register and automatically disarm itself, inhibiting further counting. Counting can be resumed by issuing a new ARM command. A software-triggered delayed pulse one-shot may be generated by specifying the TC Toggled output mode in the Counter Mode register. The initial counter contents control the delay from the ARM command until the output pulse starts. The Hold register contents control the pulse duration. Mode G is shown in Figure 15g.

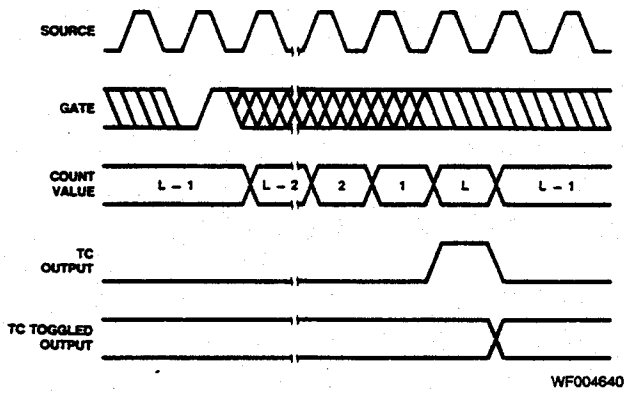


Figure 15f. Mode F Waveforms

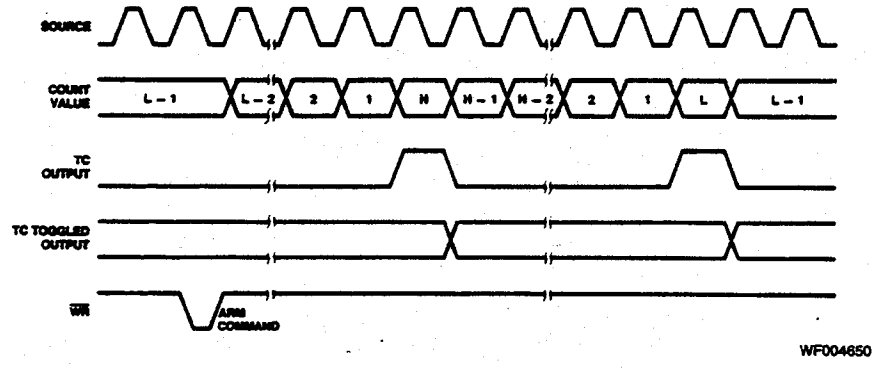


Figure 15g. Mode G Waveforms

MODE H

Software-Triggered Delayed Pulse One-Shot with Hardware Gating

CM15	CM14	CM13	CM12	CM11	CM10	CM9	CM8
LEVEL			X	X	X	X	X

CM7	CM6	CM5	CM4	CM3	CM2	CM1	CM0
0	1	0	X	X	X	X	X

Mode H, shown in Figure 15h, is identical to Mode G except that the Gate input is used to qualify which source edges are to be counted. The counter must be armed for counting to occur. Once armed, the counter will count all source edges that occur while the Gate is active. This permits the Gate to turn the count process on and off. As with Mode G, the counter will be reloaded from the Hold register on the first TC and reloaded from the Load register and disarmed on the second TC. This mode allows the Gate to control the extension of both the initial output delay time and the pulse width.

MODE I

Hardware-Triggered Delayed Pulse Strobe

CM15	CM14	CM13	CM12	CM11	CM10	CM9	CM8
EDGE			X	X	X	X	X

CM7	CM6	CM5	CM4	CM3	CM2	CM1	CM0
0	1	0	X	X	X	X	X

Mode I, shown in Figure 15i, is identical to Mode G, except that counting will not begin until a Gate edge is applied to an armed counter. The counter must be armed before application of the triggering Gate edge; Gate edges applied to a disarmed counter are disregarded. An armed counter will start counting on the first source edge after the triggering Gate edge. Counting will then proceed in the same manner as in Mode G. After the second TC, the counter will disarm itself. An ARM command and Gate edge must be issued in this order to restart counting. Note that after application of a triggering Gate edge, the Gate input will be disregarded until the second TC. This differs from Mode H, where the Gate can be modulated throughout the count cycle to stop and start the counter.

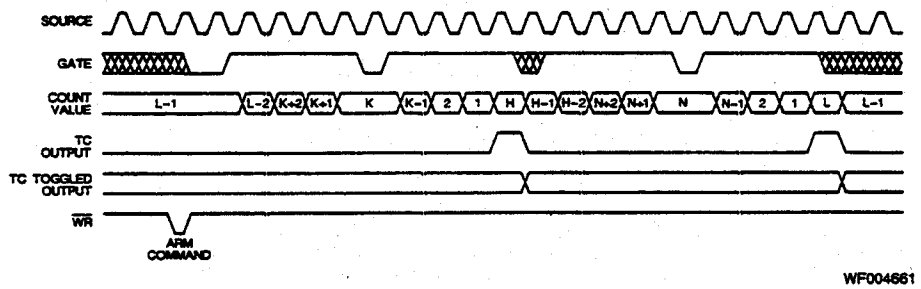


Figure 15h. Mode H Waveforms

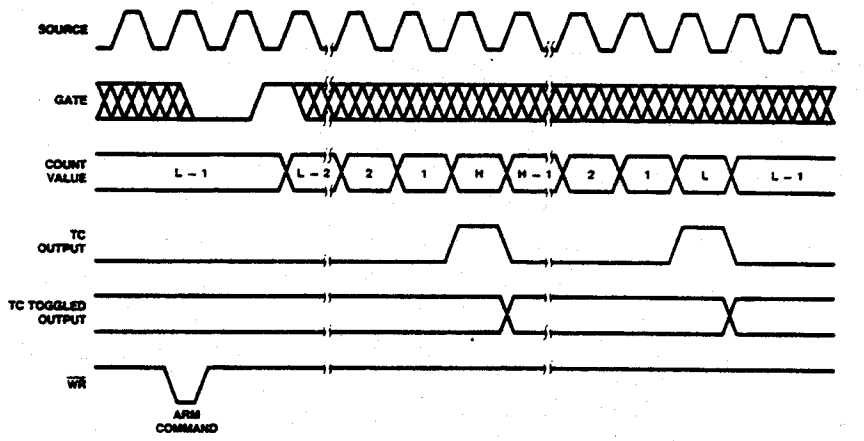


Figure 15i. Mode I Waveforms

MODE J

Variable Duty Cycle Rate Generator with No Hardware Gating

CM15	CM14	CM13	CM12	CM11	CM10	CM9	CM8
0	0	0	X	X	X	X	X

CM7	CM6	CM5	CM4	CM3	CM2	CM1	CM0
0	1	1	X	X	X	X	X

Mode J, shown in Figure 15j, will find the greatest usage in frequency generation applications with variable duty cycle requirements. Once armed, the counter will count continuously until it is issued a DISARM command. On the first TC, the counter will be reloaded from the Hold register. Counting will then proceed until the second TC at which time the counter will be reloaded from the Load register. Counting will continue, with the reload source alternating on each TC, until a DISARM command is issued to the counter. (The third TC reloads from the Hold register, the fourth TC reloads from the Load register, etc.) A variable duty cycle output can be generated by specifying the TC Toggled output in the Counter Mode register. The Load and Hold values then directly control the output duty cycle, with high resolution available when relatively high count values are used.

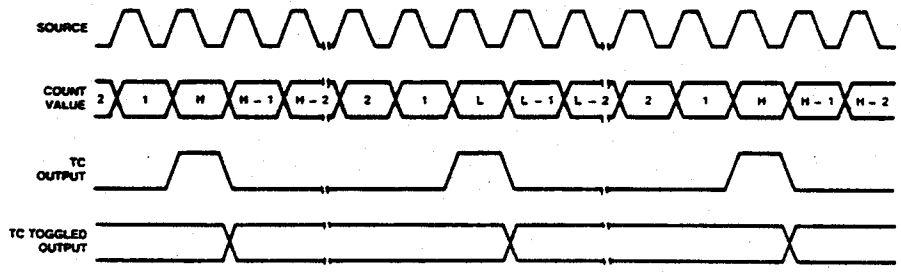
MODE K

Variable Duty Cycle Rate Generator with Level Gating

CM15	CM14	CM13	CM12	CM11	CM10	CM9	CM8
LEVEL			X	X	X	X	X

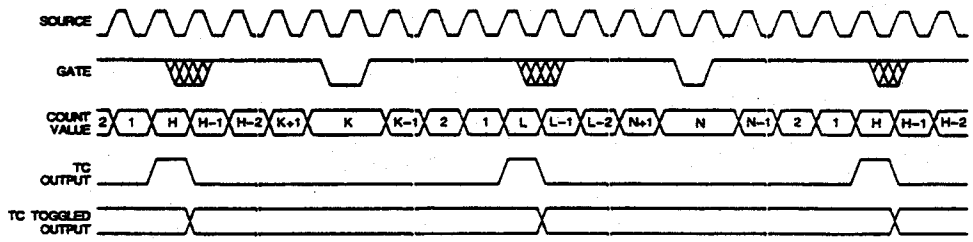
CM7	CM6	CM5	CM4	CM3	CM2	CM1	CM0
0	1	1	X	X	X	X	X

Mode K, shown in Figure 15k, is identical to Mode J, except that source edges are only counted when the Gate is active. The counter must be armed for counting to occur. Once armed, the counter will count all source edges which occur while Gate is active and disregard those source edges which occur while the Gate is inactive. This permits the Gate to turn the count process on and off. As with Mode J, the reload source used will alternate on each TC, starting with the Hold register on the first TC after any ARM command. When the TC Toggled output is used, this mode allows the Gate to modulate the duty cycle of the output waveform. It can affect both the HIGH and LOW portions of the output waveform.



WF004680

Figure 15j. Mode J Waveforms



WF004691

Figure 15k. Mode K Waveforms

2

MODE L

Hardware-Triggered Delayed Pulse One-Shot

CM15	CM14	CM13	CM12	CM11	CM10	CM9	CM8
EDGE			X	X	X	X	X

CM7	CM6	CM5	CM4	CM3	CM2	CM1	CM0
0	1	1	X	X	X	X	X

Mode L, shown in Figure 15l, is similar to Mode J except that counting will not begin until a Gate edge is applied to an armed counter. The counter must be armed before application of the triggering Gate edge; Gate edges applied to a disarmed counter are disregarded. The counter will start counting source edges after the triggering Gate edge, and counting will proceed until the second TC. Note that after application of a triggering Gate edge, the Gate input will be disregarded for the remainder of the count cycle. This differs from Mode K, where the gate can be modulated throughout the count cycle to stop and start the counter. On the first TC after application of the triggering Gate edge, the counter will be reloaded from the Hold register. On the second TC, the counter will be reloaded from the Load register, and counting will stop until a new gate edge is issued to the counter. Note that unlike Mode K, new Gate edges are required after every second TC to continue counting.

MODE N

Software-Triggered Strobe with Level Gating and Hardware Retriggering

CM15	CM14	CM13	CM12	CM11	CM10	CM9	CM8
LEVEL			X	X	X	X	X

CM7	CM6	CM5	CM4	CM3	CM2	CM1	CM0
1	0	0	X	X	X	X	X

Mode N, shown in Figure 15n, provides a software-triggered strobe with level gating that is also hardware retriggerable. The counter must be issued an ARM command before counting can occur. Once armed, the counter will count all source edges which occur while the gate is active and disregard those source edges which occur while the Gate is inactive. This permits the Gate to turn the count process on and off. After the issuance of the ARM command and the application of an active Gate, the counter will count to TC. Upon reaching TC, the counter will reload from the Load register and automatically disarm itself, inhibiting further counting. Counting will resume upon the issuance of a new ARM command. All active-going Gate edges issued to an armed counter will cause a retrigger operation. Upon application of the Gate edge, the counter contents will be saved in the Hold register. On the first qualified source edge after application of the retriggering gate edge, the contents of the Load register will be transferred into the counter. Counting will resume on the second qualified source edge after the retriggering Gate edge. Qualified source edges are active-going edges which occur while the Gate is active.

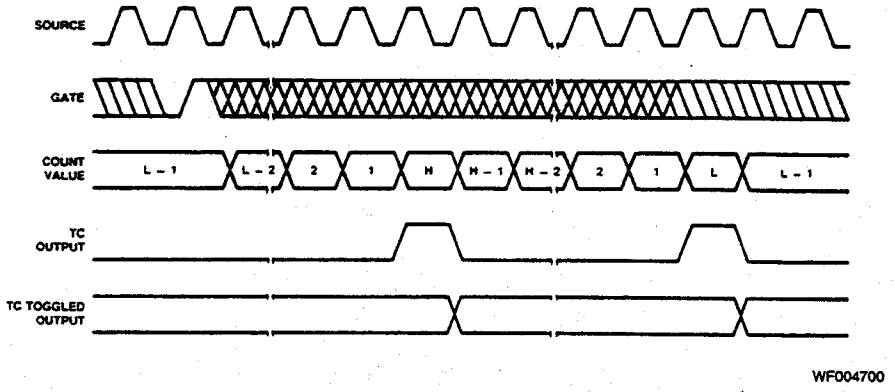


Figure 15l. Mode L Waveforms

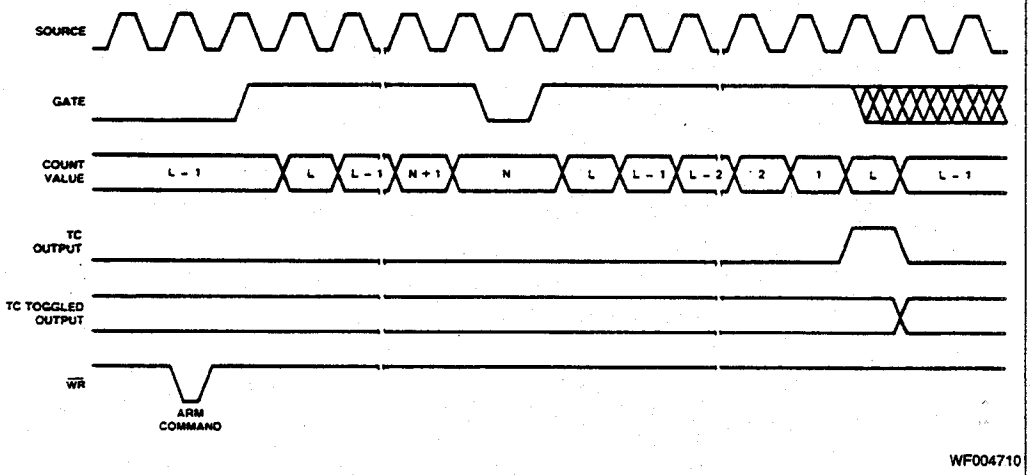


Figure 15n. Mode N Waveforms

MODE O

Software-Triggered Strobe with Edge Gating and Hardware Retriggering

CM15	CM14	CM13	CM12	CM11	CM10	CM9	CM8
EDGE			X	X	X	X	X

CM7	CM6	CM5	CM4	CM3	CM2	CM1	CM0
1	0	0	X	X	X	X	X

Mode O, shown in Figure 15o, is similar Mode N, except that counting will not begin until an active-going Gate edge is applied to an armed counter and the Gate level is not used to modulate counting. The counter must be armed before application of the triggering Gate edge; Gate edges applied to a disarmed counter are disregarded. Irrespective of the Gate level, the counter will count all source edges after the triggering Gate edge until the first TC. On the first TC,the counter will be reloaded from the Load register and disarmed. A new ARM command and a new Gate edge must be applied in that order to initiate a new counting cycle. Unlike Modes C, F, I and L, which disregard the Gate input once counting starts, in Mode O the count process will be retriggered on all active-going Gate edges, including the first Gate edge used to start the counter. On each retriggering Gate edge, the counter contents will be transferred into the Hold register. On the first source edge after the retriggering Gate edge, the Load register contents will be transferred into the counter. Counting will resume on the second-source edge after a retrigger.

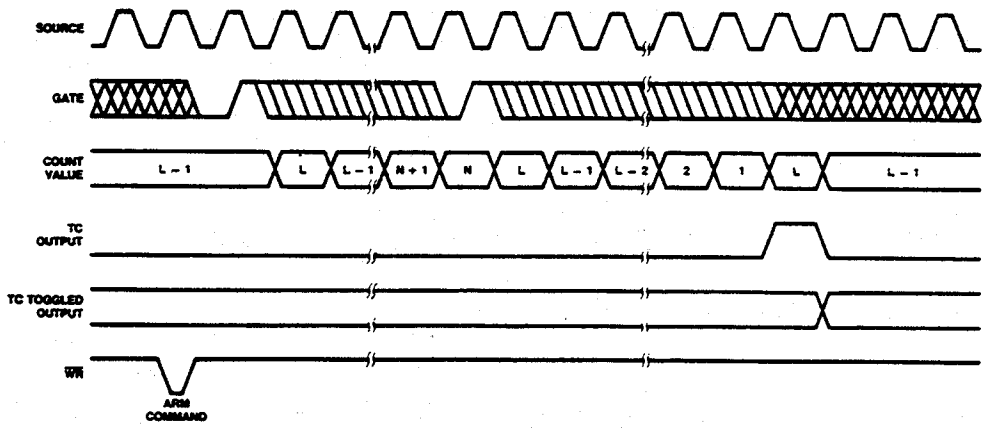


Figure 15o. Mode O Waveforms

MODE Q

Rate Generator with Synchronization (Event Counter with Auto-Read/Reset)

CM15	CM14	CM13	CM12	CM11	CM10	CM9	CM8
LEVEL			X	X	X	X	X

CM7	CM6	CM5	CM4	CM3	CM2	CM1	CM0
1	0	1	X	X	X	X	X

Mode Q, shown in Figure 15q, provides a rate generator with synchronization or an event counter with auto-read/reset. The counter must first be issued an ARM command before counting can occur. Once armed, the counter will count all source edges which occur while the Gate is active and disregard those edges which cccur while the Gate is inactive. This permits the Gate to turn the count process on and off. After the issuance of an ARM command and the application of an active Gate, the counter will count to TC repetitively. On each TC,the counter will reload itself from the Load register. The counter may be retriggered at any time by presenting an active-going Gate edge to the Gate input. The retriggering Gate edge will transfer the contents of the counter into the Hold register. The first qualified source edge after the retriggering Gate edge will transfer the contents of the Load register into the Counter. Counting will resume on the second qualified source edge after the retriggering Gate edge. Qualified source edges are active-going edges which occur while the Gate is active.

MODE R

Retriggerable One-Shot

CM15	CM14	CM13	CM12	CM11	CM10	CM9	CM8
EDGE			X	X	X	X	X

CM7	CM6	CM5	CM4	CM3	CM2	CM1	CM0
1	0	1	X	X	X	X	X

Mode R, shown in Figure 15r, is similar to Mode Q, except that edge gating rather than level gating is used. In other words, rather than use the Gate level to qualify which source edges to count, Gate edges are used to start the counting operation. The counter must be armed before application of the triggering Gate edge; Gate edges applied to a disarmed counter are disregarded. After application of a Gate edge, an armed counter will count all source edges until TC, irrespective of the Gate level. On the first TC, the counter will be reloaded from the Load register and stopped. Subsequent counting will not occur until a new Gate edge is applied. All Gate edges applied to the counter, including the first used to trigger counting, initiate a retrigger operation. Upon application of a Gate edge, the counter contents are saved in the Hold register. On the first source edge after the retriggering Gate edge, the Load register contents will be transferred into the counter. Counting will resume on the second source edge after the retriggering Gate edge.

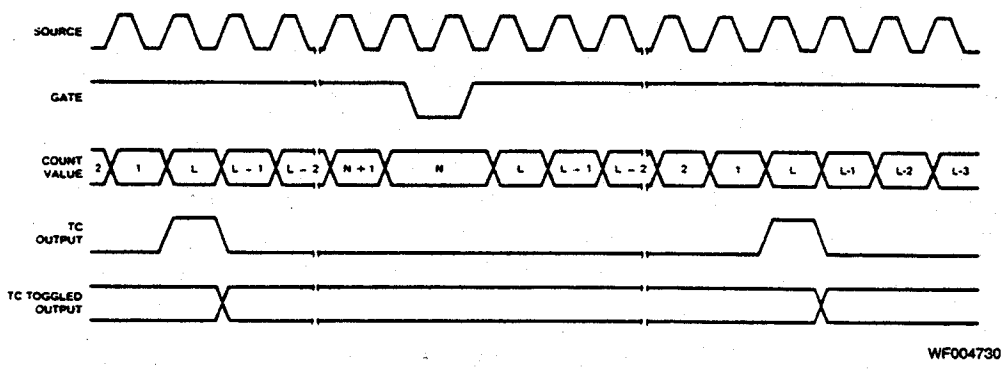


Figure 15q. Mode Q Waveforms

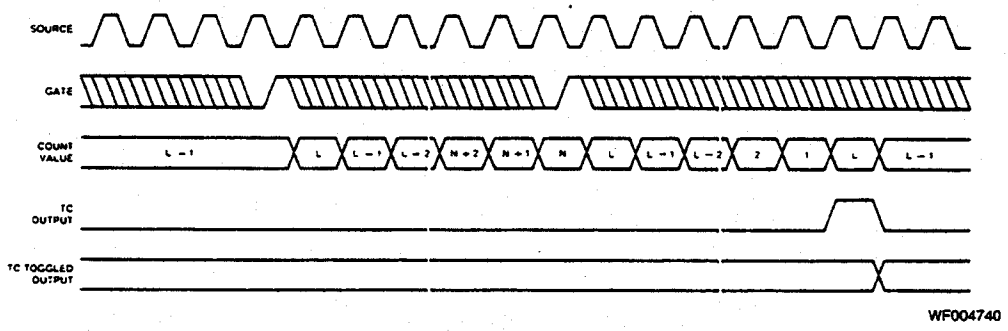


Figure 15r. Mode R Waveforms

MODE S

RELOAD SOURCE

CM15	CM14	CM13	CM12	CM11	CM10	CM9	CM8
0	0	0	X	X	X	X	X

CM7	CM6	CM5	CM4	CM3	CM2	CM1	CM0
1	1	0	X	X	X	X	X

In this mode, the reload source for LOAD commands (irrespective of whether the counter is armed or disarmed) and for TC-initiated reloads is determined by the Gate input. The Gate input in Mode S is used only to select the reload source, not to start or modulate counting. When the Gate is Low, the Load register is used; when the Gate is High, the Hold register is used. Note the Low-Load, High-Hold mnemonic convention. Once armed, the counter will count to TC twice and then disarm itself. On each TC, the counter will be reloaded from the reload source selected by the Gate. Following the second TC, an ARM command is required to start a new counting cycle. Mode S is shown in Figure 15s.

MODE V

Frequency-Shift Keying

CM15	CM14	CM13	CM12	CM11	CM10	CM9	CM8
0	0	0	X	X	X	X	X

CM7	CM6	CM5	CM4	CM3	CM2	CM1	CM0
1	1	1	X	X	X	X	X

Mode V, shown in Figure 15v, provides frequency-shift keying modulation capability. Gate operation in this mode is identical to that in Mode S. If the Gate is Low, a LOAD command or a TC-induced reload will reload the counter from the Load register. If the Gate is HIGH, LOADs and reloads will occur from the Hold register. The polarity of the Gate only selects the reload source; it does not start or modulate counting. Once armed, the counter will count repetitively to TC. On each TC, the counter will reload itself from the register determined by the polarity of the Gate. Counting will continue in this manner until a DISARM command is issued to the counter. Frequency shift keying may be obtained by specifying a TC Toggled output mode in the Counter Mode register. The switching of frequencies is achieved by modulating the Gate.

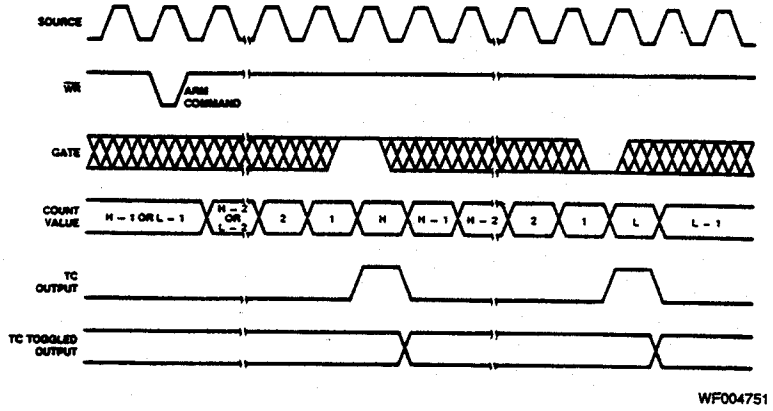


Figure 15s. Mode S Waveforms

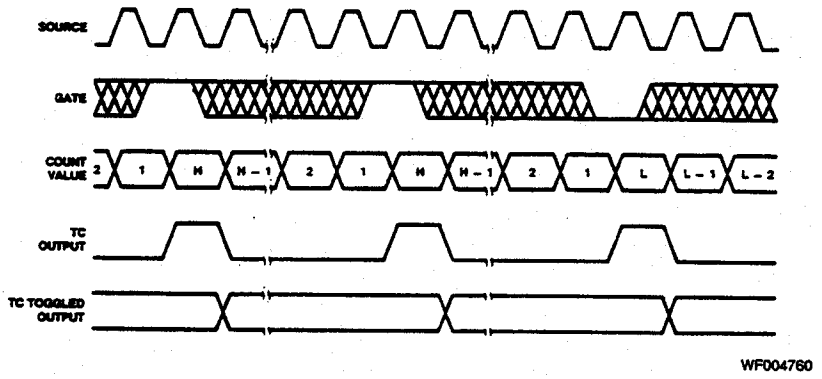


Figure 15v. Mode V Waveforms

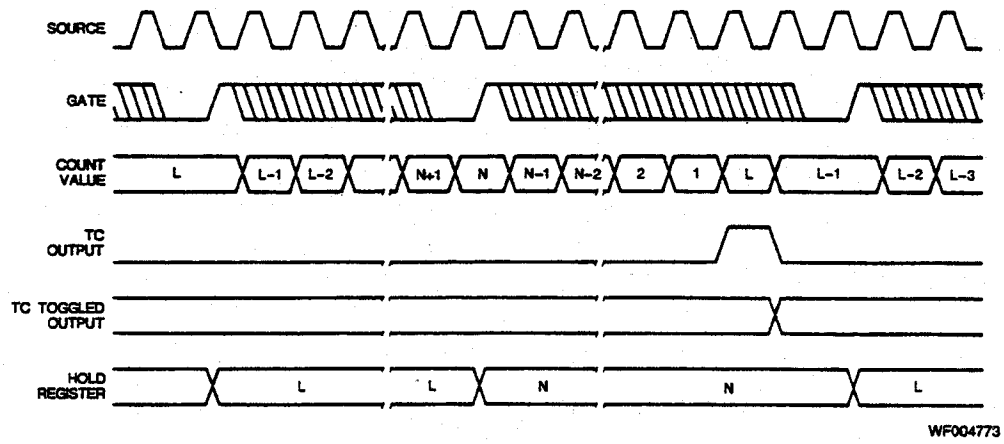


Figure 15x. Mode X Waveforms

MODE X

Hardware Save (available in Am9513A only)

CM15	CM14	CM13	CM12	CM11	CM10	CM9	CM8
Edge			X	X	X	X	X

CM7	CM6	CM5	CM4	CM3	CM2	CM1	CM0
1	1	1	X	X	X	X	X

Mode X, as shown in Figure 15x, provides a hardware sampling of the counter contents without interrupting the count. A LOAD AND ARM command or a LOAD command followed by an ARM command is required to initialize the counter. Once armed, a Gate edge starts the counting operation; Gate edges applied to a disarmed counter are disregarded. After application of the Triggering Gate edge, the counter will count all qualified source edges until the first TC, irrespective of the gate level. All gate edges applied during the counting sequence will store the current count in the Hold register, but they will not interrupt the counting sequence. On each TC, the counter will be reloaded from the Load register and stopped. Subsequent counting requires a new triggering Gate edge; counting resumes on the first source edge following the triggering Gate edge.

Note: Mode X is only available in the Am9513A devices.

COUNTER MODE CONTROL OPTIONS

Each Counter Logic Group includes a 16-bit Counter Mode (CM) register used to control all of the individual options available with its associated general counter. These options include output configuration, count control, count source and gating control. Figure 16 shows the bit assignments for the Counter Mode registers. This section describes the control options in detail. Note that generally each counter is independently configured and does not depend on information outside its Counter Logic Group. The Counter Mode register should be loaded only when the counter is Disarmed. Attempts to load the Counter Mode register when the counter is armed may result in erratic counter operation.

After power-on reset or a Master Reset command, the Counter Mode registers are initialized to a preset condition. The value entered is 0B00 hex and results in the following control configuration:

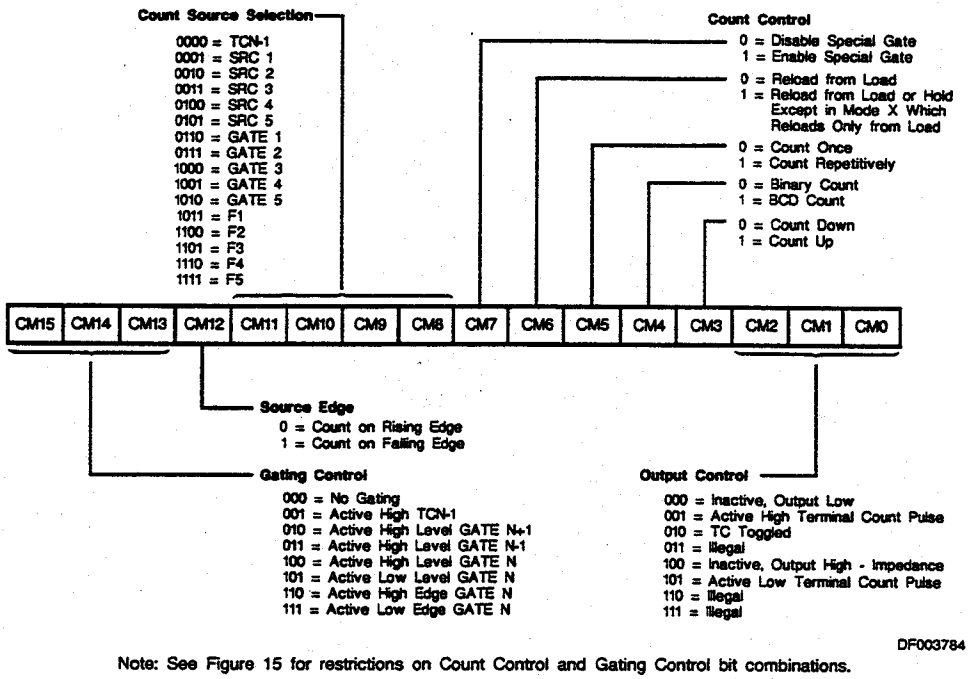
Output low-impedance to ground
Count down
Count binary
Count once
Load register selected
No retriggering
F1 input source selected
Positive-true input polarity
No gating

Output Control

Counter mode bits CM0 through CM2 specify the output control configuration. Figure 17 shows a schematic representation of the output control logic. The OUT pin may be off (a high-impedance state), or it may be inactive with a low-impedance to ground. The three remaining valid combinations represent the active-high, active-low or TC Toggle output waveforms.

One output form available is called Terminal Count (TC) and represents the period in time that the counter reaches an equivalent value of zero. TC will occur on the next count when the counter is at 0001 for down counting, at 9999 (BCD) for BCD up counting or at FFFF (hex) for binary up counting. Figure 18 shows a Terminal Count pulse and an example context that generated it. The TC width is determined by the period of the counting source. Regardless of any gating input or whether the counter is Armed or Disarmed, the terminal count will go active for only one clock cycle. Figure 18 assumes active-high source polarity, counter armed, counter decrementing and an external reload value of K.

The counter will always be loaded from an external location when TC occurs; the user can choose the source location and the value. If a non-zero value is picked, the counter will never really attain a zero state, and TC will indicate the counter state that would have been zero had no parallel transfer occurred.



Note: See Figure 15 for restrictions on Count Control and Gating Control bit combinations.

Figure 16. Counter Mode Register Bit Assignments

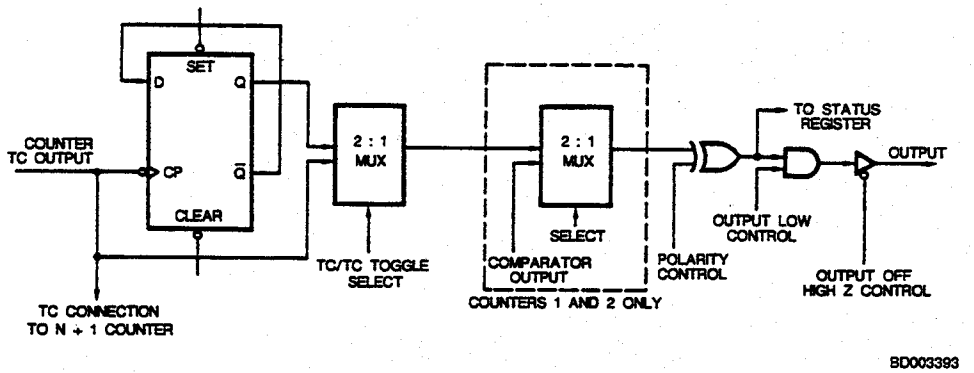


Figure 17. Output Control Logic

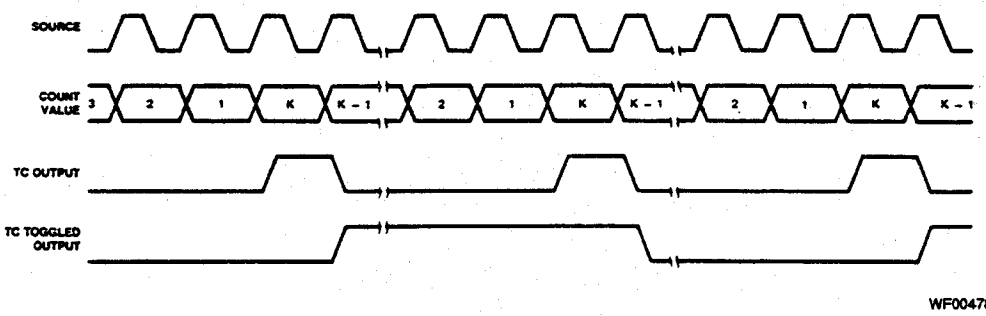


Figure 18. Counter Output Waveforms

The other output form, TC Toggled, uses the trailing edge of TC to toggle a flip-flop to generate an output level instead of a pulse. The toggle output is half the frequency of TC. The TC Toggled output will frequently be used to generate variable duty-cycle square waves in Operating Modes G through K.

In Mode L the TC Toggled output can be used to generate a one-shot function, with the delay to the start of the output pulse and the width of the output pulse separately programmable. With selection of the minimum delay to the start of the pulse, the output will toggle on the second source pulse following application of the triggering Gate edge.

Note that the TC Toggled output form contains no implication about whether the output is active-high or active-low. Unlike the TC output, which generates a transient pulse which can clearly be active-high or active-low, the TC Toggled output waveform only flips the state of the output on each TC. The sole criterion of whether the TC Toggled output is active-high or active-low is the level of the output at the start of the count cycle. This can be controlled by the Set and Clear Output commands. (See Figure 19.)

TC (Terminal Count)

On each Terminal Count (TC), the counter will reload itself from the Load or Hold register. TC is defined as that period of time when the counter contents would have been zero had no reload occurred. Some special conditions apply to counter operation immediately before and during TC.

1. In the clock cycle before TC, an internal signal is generated that commits the counter to go to TC on the next count, and retriggering by a hardware Gate edge (Modes N, O, Q and R) or a software LOAD or LOAD AND ARM command will not extend the time to TC. Note that the "next count" driving the counter to TC can be caused by the application of a count source edge (in level gating modes, the edge must occur while the gate is active, or it will be disregarded), by the application of a LOAD or LOAD AND ARM command (see 2 below) or by the application of a STEP command.
2. If a LOAD or LOAD AND ARM command is executed during the cycle preceding TC, the counter will immediately go to TC. If these commands are issued during TC, the TC state will immediately terminate.
3. When TC is active, the counter will always count the next source edge issued to it, even if it is disarmed or gated off during TC. This means that TC will never be active for longer than one count period and it may, in fact, be shorter if a STEP command or a LOAD or LOAD AND ARM command is applied during TC (see item 2 above). This also means that a counter that is disarmed or stopped on TC is actually disarmed/stopped immediately following TC.

This may cause count sequences different from what a user might expect. Since the counter is always reloaded at the start of TC and since it always counts at the end of TC, the counter contents following TC will differ by one from the reloaded value, irrespective of the operating mode used.

If the reloaded value was 0001 for down counting, 9999 (BCD) for BCD up counting or FFFF (hex) for binary up counting, the count at the end of TC will drive the counter into TC again regardless of whether the counter is gated off or disarmed. As long as these values are reloaded, the TC output will stay active. If a TC Toggled output is selected, it will toggle on each count. Execution of a LOAD, LOAD AND ARM or STEP command with these counter contents will act the same as application of a source pulse, causing TC to remain active and a TC Toggled output to toggle.

Count Control

Counter Mode bits CM3 through CM7 specify the various options available for direct control of the counting process. CM3 and CM4 operate independently of the others and control up/down and BCD/binary counting. They may be combined freely with other control bits to form many types of counting configurations. The other three bits and the Gating Control field interact in complex ways. Bit CM5 controls the repetition of the count process. When CM5 = 1, counting will proceed in the specified mode until the counter is disarmed. When CM5 = 0, the count process will proceed only until one full cycle of operation occurs. This may occur after one or two TC events. The counter is then disarmed automatically. The single or double TC requirement will depend on the state of other control bits. Note that even if the counter is automatically disarmed upon a TC, it always counts the count source edge which generates the trailing TC edge.

When TC occurs, the counter is always reloaded with a value from either the Load register of the Hold register. Bit CM6 specifies the source options for reloading the counter. When CM6 = 0, the contents of the Load register will be transferred into the counter at every occurrence of TC. When CM6 = 1, the counter reload location will be either the Load or Hold Register. The reload location in this case may be controlled externally by using a Gate pin (Modes S and V) or may alternate on each TC (Modes G through L). With alternating sources and with the TC Toggled output selected, the duty cycle of the output waveform is controlled by the relative Load and Hold values and very fine resolution of duty cycles ratios may be achieved.

Bit CM7 controls the special gating functions that allow retriggering and the selection of Load or Hold sources for counter reloading. The use and definition of CM7 will depend

on the status of the Gating Control field and bits CM5 and CM6.

Hardware Retriggering

Whenever hardware retriggering is enabled (Modes N, O, Q, and R), all active-going Gate edges initiate retrigger operations. On application of the Gate edge, the counter contents will be transferred to the Hold register. On the first qualified source edge after application of the retriggering Gate edge, the Load register contents will be transferred into the counter. (Qualified source edges are edges which occur while the counter is gated on and Armed.)

This means that, if level gating is used, the edge occurring on active-going gate transitions will initiate a retrigger. Similarly, when edge gating is enabled, an edge used to start the counter will also initiate a retrigger. The first count source edge applied after the Gate edge will not increment/decrement the counter but retrigger it.

If a LOAD, LOAD AND ARM, or a STEP Command occurs between the retriggering Gate edge and the first qualified source edge, it will be interpreted as a source edge and transfer the Load register contents into the counter. Thereafter, the counter will count all qualified source edges.

When some form of Gating is specified, CM7 controls hardware retriggering. In this case, when CM7 = 0, hardware retriggering does not occur; when CM7 = 1, the counter is retriggered any time an active-going Gate edge occurs. Retriggering causes the counter value to be saved in the Hold register and the Load register contents to be transferred into the counter.

When No Gating is specified, the definition of CM7 changes. In this case, when CM7 = 0, the Gate input has no effect on the counting; when CM7 = 1, the Gate input specifies the source (selecting either the Load or Hold register) used to reload the counter when TC occurs. Figure 14 shows the various available control combinations for these interrelated bits.

Count Source Selection

Counter Mode bits CM8 through CM12 specify the source used as input to the counter and the active edge that is counted. Bit CM12 controls the polarity for all the sources; logic zero counts rising edges and logic one counts falling edges. Bits CM8 through CM11 select 1 of 16 counting sources to route to the counter input. Five of the available inputs are internal frequencies derived from the internal oscillator (see Figure 13 for frequency assignments). Ten of the available inputs are interface pins; five are labeled SRC and five are labeled GATE.

The 16th available input is the TC output from the adjacent lower-numbered counter. (The Counter 5 TC wraps around to the Counter 1 input.) This option allows internal concatenating that permits very long counts to be accumulated. Since all five counters may be concatenated, it is possible to configure a counter that is 80-bits long on one Am9513A chip. When TCN-1 is the source, the count ripples between the connected counters. External connections can also be made, and can use the toggle bit for even longer counts. This is easily accomplished by selecting a TC Toggled output mode and wiring OUTN to one of the SRC inputs.

Gating Control

Counter Mode bits CM15, CM14, CM13 specify the hardware gating options. When "no gating" is selected (000), the counter will proceed unconditionally as long as it is armed. For any other gating mode, the count process is conditioned by the specified gating configuration.

For a code of 100 in this field, counting can proceed only when the pin labeled GATEN associated with Counter N is at a logic high level. When it goes LOW, counting is simply suspended until the Gate goes HIGH again. A code of 101 performs the same function with an opposite active polarity. Codes 010 and 011 offer the same function as 100, but specify alternate input pins as Gating Sources. This allows any of three interface pins to be used as gates for a given counter. On Counter 4, for example, pin 34, pin 35 or pin 36 may be used to perform the gating function. This also allows a single Gate pin to simultaneously control up to three counters. Counters 1 and 5 are considered adjacent when using TCN-1 (001), Gate N + 1 (010) and Gate N – 1 (011) controls.

For codes of 110 or 111 in this field, counting proceeds after the specified active Gate edge until one or two TC events occur. Within this interval, the Gate input is ignored, except for the retriggering option. When repetition is selected, a cycle will be repeated as soon as another Gate edge occurs. With repetition selected, any Gate edge applied after TC goes active will start a new count cycle. Edge gating is useful when implementing a digital single-shot since the gate can serve as a convenient firing trigger.

A 001 code in this field selects the TC (not TOGGLE) output from the adjacent lower-numbered counter as the gate. This is useful for synchronous counting when adjacent counters are concatenated.

COMMAND DESCRIPTIONS

The command set for the Am9513A allows the host processor to customize and manage the operating modes and features for particular applications, to initialize and update both the internal data and control information, and to manipulate operating bits during operation. Commands are entered directly into the 8-bit Command register by writing into the Control port (see Figure 6).

All available commands are described in the following text. Figure 19 summarizes the command codes and includes a brief description of each function. Figure 20 shows all the unused code combinations; unused codes should not be entered into the Command register since undefined activities may occur.

Six of the command types are used for direct software control of the counting process and they each contain a 5-bit S field. In a linear-select fashion, each bit in the S field corresponds to one of five general counters (S1 = Counter 1, S2 = Counter 2, etc.). When an S bit is a one, the specified operation is performed on the counter so designated; when an S bit is a zero, no operation occurs for the corresponding counter. This type of command format has three basic advantages. It saves host software by allowing any combination of counters to be acted on by a single command. It allows simultaneous action on multiple counters where synchronization of commands is important. It allows counter-specific service routines to control individual counters without needing to be aware of the operating context of other counters.

Three of the commands use a 3-bit binary code (N4, N2, N1) to identify the affected counter (a 001 programs counter 1, etc.). Unlike the previously mentioned commands, these commands allow you to program only one counter at a time.

2

Command Code								Command Description
C7	C6	C5	C4	C3	C2	C1	C0	
0	0	0	E2	E1	G4	G2	G1	Load Data Pointer register with contents of E and G fields. (G ≠ 000, G ≠ 110)
0	0	1	S5	S4	S3	S2	S1	Arm counting for all selected counters
0	1	0	S5	S4	S3	S2	S1	Load contents of specified source into all selected counters
0	1	1	S5	S4	S3	S2	S1	Load and Arm all selected counters*
1	0	0	S5	S4	S3	S2	S1	Disarm and Save all selected counters
1	0	1	S5	S4	S3	S2	S1	Save all selected counters in Hold register
1	1	0	S5	S4	S3	S2	S1	Disarm all selected counters
1	1	1	0	1	N4	N2	N1	Set Toggle out (HIGH) for counter N (001 ≤ N ≤ 101)
1	1	1	0	0	N4	N2	N1	Clear Toggle out (LOW) for counter N (001 ≤ N ≤ 101)
1	1	1	1	0	N4	N2	N1	Step counter N (001 ≤ N ≤ 101)
1	1	1	0	1	0	0	0	Set MM14 (Disable Data Pointer Sequencing)
1	1	1	0	1	1	1	0	Set MM12 (Gate off FOUT)
1	1	1	0	1	1	1	1	Set MM13 (Enter 16-bit bus mode)
1	1	1	0	0	0	0	0	Clear MM14 (Enable Data Pointer Sequencing)
1	1	1	0	0	1	1	0	Clear MM12 (Gate on FOUT)
1	1	1	0	0	1	1	1	Clear MM13 (Enter 8-bit bus mode)
1	1	1	1	1	0	0	0	Enable Prefetch for Write operations (Am9513'A' only)
1	1	1	1	1	0	0	1	Disable Prefetch for Write operations (Am9513'A' only)
1	1	1	1	1	1	1	1	Master reset

*Not to be used for asynchronous operations.

Figure 19. Am9513A Command Summary

C7	C6	C5	C4	C3	C2	C1	C0
1	1	1	1	0	0	0	0
1	1	1	1	0	1	1	0
1	1	1	1	0	1	1	1
0	0	0	X	X	1	1	0
0	0	0	X	X	0	0	0
*1	1	1	1	1	X	X	X

*Unused except when XXX = 111, 001 or 000.

Figure 20. Am9513A Unused Command Codes

Arm Counters

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
0	0	1	S5	S4	S3	S2	S1

Description: Any combination of counters, as specified by the S field, will be enabled for counting. A counter must be armed before counting can commerce. Once armed, the counting process may be further enabled or disabled using the hardware gating facilities. This command can only arm or do nothing for a given counter; a zero in the S field does not disarm the counter.

ARM and DISARM commands can be used to gate counter operation on and off under software control. DISARM commands entered while a counter is in the TC state will not take effect until the counter leaves TC. This ensures that the counter never latches up in a TC state. (The counter may leave the TC state because of application of a count source edge, execution of a LOAD or LOAD AND ARM command, or execution of a STEP command.)

In modes which alternate reload sources (Modes G – L), the ARMing operation is used as a reset for the logic which determines which reload source to use on the upcoming TC. Following each ARM or LOAD AND ARM command, a counter in one of these modes will reload from the Hold register on the first TC and alternate reload sources thereafter (reload from the Load register on the second TC, the Hold register on the third, etc.).

Load Counters

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
0	1	0	S5	S4	S3	S2	S1

Description: Any combination of counters, as specified in the S field, will be loaded with previously entered values. The source of information for each counter will be either the associated Load register or the associated Hold register, as determined by the operating configuration in the Mode register. The Load/Hold contents are not changed. This command will cause a transfer independent of any current operating configuration for the counter. It will often be used as a software retrigger or as counter initialization prior to active hardware gating.

If a LOAD or LOAD AND ARM command is executed during the cycle preceding TC, the counter will go immediately to TC. This occurs because the LOAD operation is performed by generating a pseudo-count pulse internal to the Am9513A, and the Am9513A is expecting to go into TC on the next count pulse. The reload source used to reload the counter will be the same as that which would have been used if the TC were generated by a source edge rather than by the LOAD operation.

Execution of a LOAD or LOAD AND ARM command while the counter is in TC will cause the TC to end. For Armed counters in all modes except S or V, the LOAD source used will be that to be used for the upcoming TC. (The LOADing operation will not alter the selection of reload source for the upcoming TC.) For Disarmed counters in modes except S or V, the reload sources used will be the LOAD register. For modes S or V, the reload source will be selected by the GATE input, regardless of whether the counter is Armed or Disarmed.

Special considerations apply when modes with alternating reload sources are used (Modes G – L). If a LOAD command drives the counter to TC in these modes, the reload source for the next TC will be from the opposite reload location. In other words, the LOAD-generated TC will cause the reload sources to alternate just as a TC generated by a source edge would. Note that if a second LOAD command is issued during the LOAD-generated TC (or during any other TC, for that matter), the second LOAD command will terminate the TC and cause a reload from the source designated for use with the next TC. The second LOAD will not alter the reload source for the next TC since the second LOAD does not generate a TC; reload sources alternate on TCs only, not on LOAD commands.

Load and Arm Counters*

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
0	1	1	S5	S4	S3	S2	S1

Description: Any combination of counters, as specified in the S field, will be first loaded and then armed. This command is equivalent to issuing a LOAD command and then an ARM command.

A LOAD AND ARM command which drives a counter to TC generates the same sequence of operations as execution of a LOAD command and then an ARM command. In modes which disarm on TC (Modes A – C and N – O, and Modes G – I and S if the current TC is the second in the cycle), the ARM part of the LOAD AND ARM command will re-enable counting for another cycle. In modes which alternate reload sources (Modes G – L), the ARMing operation will cause the next TC to reload from the HOLD register, irrespective of which reload source the current TC used.

*This command should not be used during asynchronous operations.

Disarm Counters

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
1	1	0	S5	S4	S3	S2	S1

Description: Any combination of counters, as specified by the S field, will be disabled from counting. A disarmed counter will cease all counting independent of other conditions. The only exception to this is that a counter in the TC state will always count once, in order to leave TC, before DISARMing. This count may be generated by a source edge, by a LOAD or LOAD AND ARM command (the LOAD AND ARM command will negate the DISARM command) or by a STEP command. A disarmed counter may be updated using the LOAD command and may be read using the SAVE command. A count process may be resumed using an ARM command. See the ARM command description for further details.

Save Counters

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
1	0	1	S5	S4	S3	S2	S1

Description: Any combination of counters, as specified by S field, will have their contents transferred into their associated Hold register. The transfer takes place without interfering with any counting that may be underway. This command will overwrite any previous Hold register contents. The SAVE command is designed to allow an accumulated count to be preserved so that it can be read by the host CPU at some later time.

Disarm and Save Counters

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
1	0	0	S5	S4	S3	S2	S1

Description: Any combination of counters, as specified by the S field, will be disarmed, and the contents of the counter will be transferred into the associated Hold registers. This command is identical to issuing a DISARM command followed by a SAVE command.

Set TC Toggle Output

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
1	1	1	0	1	N4	N2	N1

(001 ≤ N ≤ 101)

Description: The initial output level for TC Toggle mode is set (HIGH) for counter N selected by N4, N2, N1 = 001 (Counter 1) thru 101 (Counter 5) respectively. This command conditions the TC Toggle flip-flop (see Figure 17) but does not appear at the counter output unless TC Toggle mode (CM2, CM1, CM0 = 010) is selected.

Clear TC Toggle Output

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
1	1	1	0	0	N4	N2	N1

(001 ≤ N ≤ 101)

Description: The initial output level for TC Toggle mode is Cleared (LOW) for counter N selected by N4, N2, N1 = 001 (Counter 1) thru 101 (Counter 5) respectively. This command conditions the TC Toggle flip-flop (see Figure 17) but does not appear at the counter output unless TC Toggle mode (CM2, CM1, CM0 = 010) is selected.

Step Counter

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
1	1	1	1	0	N4	N2	N1

(001 ≤ N ≤ 101)

Description: Counter N is incremented or decremented by one, depending on its operating configuration. If the Counter Mode register associated with the selected counter has its CM3 bit cleared to zero, this command will cause the counter to decrement by one. If CM3 is set to a logic high, this command will increment the counter by one. The STEP command will take effect even on a disarmed counter.

Load Data Pointer Register

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
0	0	0	E2	E1	G4	G2	G1

(G4, G2, G1 ≠ 000, ≠ 110)

Description: Bits in the E and G fields will be transferred into the corresponding Element and Group fields of the Data Pointer register as shown in Figure 7. The Byte Pointer bit in the Data Pointer register is set. Transfers into the Data Pointer only occur for G field values of 001, 010, 011, 100, 101 and 111. Values of 000 and 110 for G should not be used. See the "Setting the Data Pointer Register" section of this document for additional details.

Disable Data Pointer Sequencing

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
1	1	1	0	1	0	0	0

Description: This command sets Master Mode bit 14 without affecting other bits in the Master Mode register. MM14 controls the automatic sequencing of the Data Pointer register. Disabling the sequencing allows repetitive host processor access to a given internal location without repetitive updating of the Data Pointer. MM14 may also be controlled by loading a full word into the Master Mode register.

Enable Data Pointer Sequencing

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
1	1	1	0	0	0	0	0

Description: This command clears Master Mode bit 14 without affecting other bits in the Master Mode register. MM14 controls the automatic sequencing of the Data Pointer register. Enabling the sequencing allows sequential host processor access to several internal locations without repetitive updating of the Data Pointer. MM14 may also be controlled by loading a full word into the Master Mode register. See the "Data Pointer Register" section of this document for additional information on Data Pointer sequencing

Enable 16-Bit Data Bus

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
1	1	1	0	1	1	1	1

Description: This command sets Master Mode bit 13 without affecting other bits in the Master Mode register. MM13 controls the multiplexer in the data bus buffer. When MM13 is set, no multiplexing takes place and all 16 external data bus lines are used to transfer information into and out of the STC. MM13 may also be controlled by loading the full Master Mode register in parallel.

Enable 8-Bit Data Bus

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
1	1	1	0	0	1	1	1

Description: This command clears Master Mode bit 13 without affecting other bits in the Master Mode register. MM13 controls the multiplexer in the data bus buffer. When MM13 is cleared, the multiplexer is enabled and 16-bit internal information is transferred eight bits at a time to the eight low-order external data bus lines. MM13 may also be controlled by loading the full Master Mode register in parallel.

Gate Off FOUT

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
1	1	1	0	1	1	1	0

Description: This command sets Master Mode bit 12 without affecting other bits in the Master Mode register. MM12 controls the output state of the FOUT signal. When gated off, the FOUT line will exhibit a low-impedance to ground. MM12 may also be controlled by loading the full Master Mode register in parallel.

Gate On FOUT

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
1	1	1	0	0	1	1	0

Description: This command clears Master Mode bit 12 without affecting other bits in the Master Mode register. MM12 controls the output status of the FOUT signal. When MM12 is cleared, FOUT will become active and will drive out the selected and divided FOUT signal. MM12 may also be controlled by loading the full Master Mode register in parallel. When FOUT is gated on or off, a transient pulse may be generated on the FOUT signal.

Disable Prefetch for Write Operations

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
1	1	1	1	1	0	0	1

Description: This command disables the prefetch circuitry during Write operations (if does not affect Read operations). This reduces the write recovery time and allows the user to use block move instructions for initialization of the Am9513A registers. Once prefetch is disabled for writing, an Enable Prefetch for Write or a Reset command is necessary to re-enable the prefetch circuitry for writing.
Note: This command is only available in Am9513A devices; it is an illegal command in the "non-A" Am9513 device.

Enable Prefetch for Write Operations

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
1	1	1	1	1	0	0	0

Description: This command re-enables the prefetch circuitry for Write operations. It is used only to terminate the Disable Prefetch Command.
Note: This command is only available in Am9513A devices; it is an illegal command in the "non-A" Am9513 device.

Master Reset

Coding:

C7	C6	C5	C4	C3	C2	C1	C0
1	1	1	1	1	1	1	1

Description: The Master Reset command duplicates the action of the power-on reset circuitry. It disarms all counters, enters 0000 in the Master Mode, Load and Hold registers and enters 0B00 (hex) in the Counter Mode registers.

Following either a power-up or software reset, the LOAD command should be applied to all the counters to clear any that may be in a TC state. The Data Pointer register should also be set to a legal value, since reset does not initialize it. A complete reset operation follows.

1. Using the procedure given in the "Command Initiation" section of this document, enter the FF (hex) command to perform a software reset.
2. Using the "Command Initiation" procedure, enter the LOAD command for all counters, opcode 5F (hex).
3. Using the procedure given in the "Setting the Data Pointer Register" section of this document, set the Data Pointer to a valid code. The legal Data Pointer codes are given in Figure 8.

The Master Mode, Counter Mode, Load and Hold registers can now be initialized to the desired values.

ABSOLUTE MAXIMUM RATINGS

Storage Temperature ..... −65°C to +150°C
VCC with Respect to VSS ..... −0.5 V to +7.0 V
All Signal Voltages
with Respect to VSS ..... −0.5 V to +7.0 V
Power Dissipitation (Package Limitation) ..... 1.5 W

Stresses above those listed under ABSOLUTE MAXIMUM RATINGS may cause permanent device failure. Functionality at or above these limits is not implied. Exposure to absolute maximum ratings for extended periods may affect device reliability.

OPERATING RANGES

Commercial (C) Devices
Temperature (T_A) ..... 0 to +70°C
Supply Voltage (V_{CC}) ..... 5 V ±5%

Industrial (I) Devices
Temperature (T_A) ..... −40 to +85°C
Supply Voltage (V_{CC}) ..... 5 V ±5%

Military (M) Devices
Temperature (T_C) ..... −55 to +125°C
Supply Voltage (V_{CC}) ..... 5 V ±5%

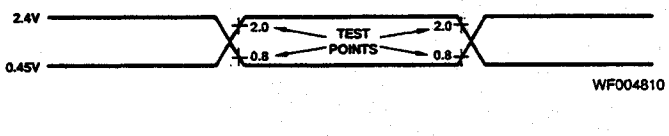
Operating ranges define those limits between which the functionality of the device is guaranteed.

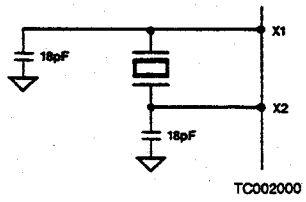
DC CHARACTERISTICS over operating ranges unless otherwise specified.

Parameters	Description		Test Conditions	Min	Max		Units
VIL	Input Low Voltage	All Inputs Except X2		VSS − 0.5	0.8		Volts
		X2 Input		VSS − 0.5	0.8		
VIH	Input High Voltage	All Input Except X2		2.2 V	VCC		Volts
		X2 Input		3.8	VCC		
VITH	Input Hysteresis (SRC and GATE Inputs Only)			0.2			Volts
VOL	Output Low Voltage		IOL = 3.2 mA		0.4		Volts
VOH	Output High Voltage		IOH = −200 µA	2.4			Volts
IIX	Input Load Current (Except X2)		VSS ≤ VIN ≤ VCC		±10		µA
IIX	Input Load Current X2		VSS ≤ VIN ≤ VCC		±100		µA
IOZ	Output Leakage Current (Except X1)		VSS + 0.4 ≤ VOUT ≤ VCC High-Impedance State		±25		µA
ICC	VCC Supply Current (Steady State)				255	275	mA
CIN	Input Capacitance		f = 1 MHz, T_A = +25°C. All pins not under test at 0 V.		10*	20*	pF
COUT	Output Capacitance				15*	20*	
CIO	IN/OUT Capacitance				20*	20*	

* Guaranteed by design.

SWITCHING TEST INPUT/OUTPUT WAVEFORMS





Crystal is fundamental mode parallel resonant 32 pF load capacitance less than 100 Ω ESR C_0 less than 100 pF.

The second and fourth letters designate the reference states of the signals named in the first and third letters respectively, using the following abbreviations.

H = HIGH
L = LOW
V = VALID
X = Unknown or Don't care
Z = High-Impedance

2. Any input transition that occurs before this minimum setup requirement will be reflected in the contents read from the status register.
3. Any input transition that occurs before this minimum setup requirement will act on the counter before the execution of the operation initiated by the write and the counter may be off by one count.
4. Any input transition that occurs after this minimum hold time is guaranteed to not influence the contents read from the status register on the current read operation.
5. Any input transition that occurs after this minimum hold time is guaranteed to be seen by the counter as occurring after the action initiated by the write operation and the counter may be off by one count.
6. This parameter applies to cases where the write operation causes a change in the output bit.
7. The enabled count source is one of F1 – F5, TCN-1 SRC1 – SRC5 or GATE1 – GATE 5, as selected in the applicable Counter Mode register. The timing diagram assumes the counter counts on rising source edges. The timing specifications are the same for falling-edge counting.
8. This parameter applies to edge gating (CM15 – CM13 = 110 or 111) and gating when both CM7 = 1 and CM15 – CM13 ≠ 000. This parameter represents the minimum GATE pulse width needed to ensure that the pulse initiates counting or counter reloading.
9. This parameter applies to both edge and level gating (CM15 – CM13 = 001 through 111 and CM7 = 0). This parameter represents the minimum setup or hold times to ensure that the Gate input is seen at the intended level on the active source edge and the counter may be off by one count.
10. This parameter assumes that the GATENA input is unused (16-bit bus mode) or is tied high. In cases where the GATENA input is used, this timing specification must be met by both the GATE and GATENA inputs.
11. Signals F1 – F5 cannot be directly monitored by the user. The phase difference between these signals will manifest itself by causing counters using two different F signals to count at different times on nominally simultaneous transitions in the F signals. F1 = X2.
12. This timing specification assumes that $\overline{CS}$ is active whenever $\overline{RD}$ or $\overline{WR}$ are active. $\overline{CS}$ may be held active indefinitely.
13. This parameter assumes X2 is driven from an external gate with a square wave.
14. This parameter assumes that the write operation is to the command register.
15. This timing specification applies to single-action commands only (e.g., LOAD, ARM, SAVE, etc.). For double-action commands such as LOAD AND ARM and DISARM AND SAVE, TWHEH minimum = 700 ns.
16. In short data write mode, TWHRL and TWHWL minimum = 1000 ns.
17. This parameter applies to the hardware retrigger/save modes N, O, Q, R, and X (CM7 = 1 and CM15 – CM13 < > 000). This parameter ensures that the gating pulse initiates a hardware retrigger/save operation.
18. This parameter applies to hardware load source select modes S and V (CM7 = 1 and CM15 – CM13 = 000). This parameter represents the minimum hold time to ensure that the GATE input selects the correct load source on the active source edge.

SWITCHING CHARACTERISTICS over **COMMERCIAL** operating ranges unless otherwise specified (Note 1)

Parameters	Description		Figure	Am9513A Min	Am9513A Max	Unit
TAVRL	C/$\overline{D}$ Valid to Read Low		21	25		ns
TAVWH	C/$\overline{D}$ Valid to Write High		21	170		ns
TCHCH	X2 High to X2 High (X2 Period) (Note 13)		22	145		ns
TCHCL	X2 High to X2 Low (X2 High Pulse Width) (Note 13)		22	70		ns
TCLCH	X2 Low to X2 High (X2 Low Pulse Width) (Note 13)		22	70		ns
TDVWH	Data In Valid to Write High		21	80		ns
TEHEH	Count Source High to Count Source High (Source Cycle Time) (Note 7)		22	145		ns
TEHEL TELEH	Count Source Pulse Duration (Note 7)		22	70		ns
TEHFV	Count Source High to FOUT Valid (Note 7)		22		500	ns
TEHGV	Count Source High to Gate Valid (Level Gating Hold Time) (Notes 7, 9, 10)		22	10		ns
TEHRL	Count Source High to Read Low (Set-up Time) (Notes 2, 7)		21	190		ns
TEHWH	Count Source High to Write High (Set-up Time) (Notes 3, 7)		21	-100		ns
TEHYV	Count Source High to Out Valid (Note 7)	TC Output	22		300	ns
		Immediate or Delayed Toggle Output	22		300	
		Comparator Output	22		350	
TFN	FN High to FN + 1 Valid (Note 11)		22		75	ns
TGVEH	Gate Valid to Count Source High (Level Gating Set-up Time) (Notes 7, 9, 10)		22	100		ns
TGVGV	Gate Valid to Gate Valid (Gate Pulse Duration) (Notes 8, 10)		22	145		ns
TGVWH	Gate Valid to Write High (Notes 3, 10)		21	-100		ns
TRHAX	Read High to C/$\overline{D}$ Don't Care		21	0		ns
TRHEH	Read High to Count Source High (Notes 4, 7)		21	0		ns
TRHQX	Read High to Data Out Invalid		21	10		ns
TRHQZ	Read High to Data Out at High-Impedance (Data Bus Release Time)		21		85	ns
TRHRL	Read High to Read Low (Read Recovery Time)		21	1000		ns
TRHSH	Read High to $\overline{CS}$ High (Note 12)		21	0		ns
TRHWL	Read High to Write Low (Read Recovery Time)		21	1000		ns
TRLQV	Read Low to Data Out Valid		21		110	ns
TRLQX	Read Low to Data Bus Driven (Data Bus Drive Time)		21	20		ns
TRLRH	Read Low to Read High (Read Pulse Duration) (Note 12)		21	160		ns
TSLRL	$\overline{CS}$ Low to Read Low (Note 12)		21	20		ns
TSLWH	$\overline{CS}$ Low to Write High (Note 12)		21	170		ns
TWHAX	Write High to C/$\overline{D}$ Don't Care		21	20		ns
TWHDX	Write High to Data In Don't Care		21	20		ns
TWHEH	Write High to Count Source High (Notes 5, 7, 14, 15)		21	550		ns
TWHGV	Write High to Gate Valid (Notes 5, 10, 14)		21	475		ns
TWHRL	Write High to Read Low (Write Recovery Time) (Note 16)		21	1500*		ns
TWHSH	Write High to $\overline{CS}$ High (Note 12)		21	20		ns
TWHWL	Write High to Write Low (Write Recovery Time) (Note 16)		21	1500*		ns
TWHYV	Write High to Out Valid (Notes 6, 14)		21		650	ns
TWLWH	Write Low to Write High (Write Pulse Duration) (Note 12)		21	150		ns
TGVEH2	Gate Valid to Count Source High (Special Gate) (Notes 10, 13, 17)		22	200		ns
TEHGV2	Count Source High to Gate Valid (Special Gate) (Notes 10, 13, 18)		22	80		ns

Notes:

1. Abbreviations used for the switching parameter symbols are given as the letter T followed by four or five characters. The first and third characters represent the signal names on which the measurements start and end. Signal abbreviations used are:

   A (Address) = C/$\overline{D}$
   C (Clock) = X2
   D (Data In) = DB0-DB15
   E (Enabled counter source input) = SRC1 - SRC5, GATE1 - GATE5, F1 - F5,TCN-1
   F = FOUT
   G (Counter gate input) = GATE1 - GATE5, TCN-1
   Q (Data Out) = DB0 - DB15
   R (Read) = $\overline{RD}$
   S (Chip Select) = $\overline{CS}$
   W (Write) = $\overline{WR}$
   Y (Output) = OUT1 - OUT5

2

SWITCHING CHARACTERISTICS over **MILITARY** operating range (for SMD/DESC and APL Products, Group A, Subgroups 9, 10, 11 are tested unless otherwise noted)

Parameter Symbol	Description		Am9513A Min.	Am9513A Max.	Unit
TAVRL	C/$\overline{D}$ Valid to Read Low		25		ns
TAVWH	C/$\overline{D}$ Valid to Write High		170		ns
TCHCH	X2 High to X2 High (X2 Period) (Note 13)		145		ns
TCHCL	X2 High to X2 Low (X2 High Pulse Width) (Note 13)		70		ns
TCLCH	X2 Low to X2 High (X2 Low Pulse Width) (Note 13)		70		ns
TDVWH	Data In Valid to Write High		80		ns
TEHEH	Count Source High to Count Source High (Source Cycle Time) (Note 7)		145		ns
TEHEL TELEH	Count Source Pulse Duration (Note 7)		70		ns
TEHFV	Count Source High to FOUT Valid (Note 7)			500	ns
TEHGV	Count Source High to Gate Valid (Level Gating Hold Time) (Notes 7, 9, 10)		10		ns
TEHRL	Count Source High to Read Low (Set-up Time) (Notes 2, 7)		190		ns
TEHWH	Count Source High to Write High (Set-up Time) (Notes 3, 7)		-100		ns
TEHYV	Count Source High to Out Valid (Note 7)	TC Output		300	ns
		Immediate or Delayed Toggle Output		300	
		Comparator Output		350	
TFN	FN High to FN + 1 Valid (Note 11)			75	ns
TGVEH	Gate Valid to Count Source High (Level Gating Set-up Time) (Notes 7, 9, 10)		100		ns
TGVGV	Gate Valid to Gate Valid (Gate Pulse Duration) (Notes 8, 10)		145		ns
TGVWH	Gate Valid to Write High (Notes 3, 10)		-100		ns
TRHAX	Read High to C/$\overline{D}$ Don't Care		0		ns
TRHEH	Read High to Count Source High (Notes 4, 7)		0		ns
TRHQX	Read High to Data Out Invalid		10		ns
TRHQZ	Read High to Data Out at High-Impedance (Data Bus Release Time)			85	ns
TRHRL	Read High to Read Low (Read Recovery Time)		1000		ns
TRHSH	Read High to $\overline{CS}$ High (Note 12)		0		ns
TRHWL	Read High to Write Low (Read Recovery Time)		1000		ns
TRLQV	Read Low to Data Out Valid			110	ns
TRLQX	Read Low to Data Bus Driven (Data Bus Drive Time)		20		ns
TRLRH	Read Low to Read High (Read Pulse Duration) (Note 12)		160		ns
TSLRL	$\overline{CS}$ Low to Read Low (Note 12)		20		ns
TSLWH	$\overline{CS}$ Low to Write High (Note 12)		170		ns
TWHAX	Write High to C/$\overline{D}$ Don't Care		20		ns
TWHDX	Write High to Data In Don't Care		20		ns
TWHEH	Write High to Count Source High (Notes 5, 7, 14, 15)		550		ns
TWHGV	Write High to Gate Valid (Notes 5, 10, 14)		475		ns
TWHRL	Write High to Read Low (Write Recovery Time) (Note 16)		1500		ns
TWHSH	Write High to $\overline{CS}$ High (Note 12)		20		ns
TWHWL	Write High to Write Low (Write Recovery Time) (Note 16)		1500		ns
TWHYV	Write High to Out Valid (Notes 6, 14)			650	ns
TWLWH	Write Low to Write High (Write Pulse Duration) (Note 12)		150		ns
TGVEH2	Gate Valid to Count Source High (Special Gate) (Notes 10, 13, 17)		200		ns
TEHGV2	Count Source High to Gate Valid (Special Gate) (Notes 10, 13, 18)		80		ns

Notes:

1. Abbreviations used for the switching parameter symbols are given as the letter T followed by four or five characters. The first and third characters represent the signal names on which the measurements start and end. Signal abbreviations used are:

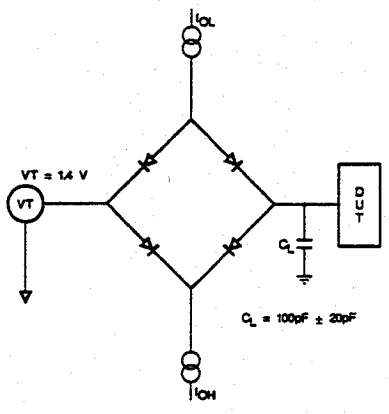
   A (Address) = C/$\overline{D}$
   C (Clock) = X2
   D (Data In) = DB0 - DB15
   E (Enabled counter source input) = SRC1 - SRC5, GATE1 - GATE5, F1 - F5,TCN-1
   F = FOUT
   G (Counter gate input) = GATE1 - GATE5, TCN-1
   Q (Data Out) = DB0 - DB15
   R (Read) = $\overline{RD}$
   S (Chip Select) = $\overline{CS}$
   W (Write) = $\overline{WR}$
   Y (Output) = OUT1 - OUT5

The second and fourth letters designate the reference states of the signals named in the first and third letters respectively, using the following abbreviations.

H = HIGH
L = LOW
V = VALID
X = Unknown or Don't care
Z = High-impedance

2. Any input transition that occurs before this minimum setup requirement will be reflected in the contents read from the status register.
3. Any input transition that occurs before this minimum setup requirement will act on the counter before the execution of the operation initiated by the write and the counter may be off by one count.
4. Any input transition that occurs after this minimum hold time is guaranteed to not influence the contents read from the status register on the current read operation.
5. Any input transition that occurs after this minimum hold time is guaranteed to be seen by the counter as occurring after the action initiated by the write operation and the counter may be off by one count.
6. This parameter applies to cases where the write operation causes a change in the output bit.
7. The enabled count source is one of F1 – F5, TCN-1 SRC1 – SRC5 or GATE1 – GATE 5, as selected in the applicable Counter Mode register. The timing diagram assumes the counter counts on rising source edges. The timing specifications are the same for falling-edge counting.
8. This parameter applies to edge gating (CM15 – CM13 = 110 or 111) and gating when both CM7 = 1 and CM15 – CM13 ≠ 000. This parameter represents the minimum GATE pulse width needed to ensure that the pulse initiates counting or counter reloading.
9. This parameter applies to both edge and level gating (CM15 – CM13 = 001 through 111 and CM7 = 0). This parameter represents the minimum setup or hold times to ensure that the Gate input is seen at the intended level on the active source edge and the counter may be off by one count.
10. This parameter assumes that the GATENA input is unused (16-bit bus mode) or is tied high. In cases where the GATENA input is used, this timing specification must be met by both the GATE and GATENA inputs.
11. Signals F1 – F5 cannot be directly monitored by the user. The phase difference between these signals will manifest itself by causing counters using two different F signals to count at different times on nominally simultaneous transitions in the F signals. F1 = X2.
12. This timing specification assumes that $\overline{CS}$ is active whenever $\overline{RD}$ or $\overline{WR}$ are active. $\overline{CS}$ may be held active indefinitely.
13. This parameter assumes X2 is driven from an external gate with a square wave.
14. This parameter assumes that the write operation is to the command register.
15. This timing specification applies to single-action commands only (e.g., LOAD, ARM, SAVE, etc.). For double-action commands such as LOAD AND ARM and DISARM AND SAVE, TWHEH minimum = 700 ns.
16. In short data write mode, TWHRL and TWHWL minimum = 1000 ns.
17. This parameter applies to the hardware retrigger/save modes N, O, Q, R, and X (CM7 = 1 and CM15 – CM13 < > 000). This parameter ensures that the gating pulse initiates a hardware retrigger/save operation.
18. This parameter applies to hardware load source select modes S and V (CM7 = 1 and CM15 – CM13 = 000). This parameter represents the minimum hold time to ensure that the GATE input selects the correct load source on the active source edge.

SWITCHING TEST CIRCUIT



TC003853

This test circuit is the dynamic load of a Teradyne J941.

SWITCHING WAVEFORMS

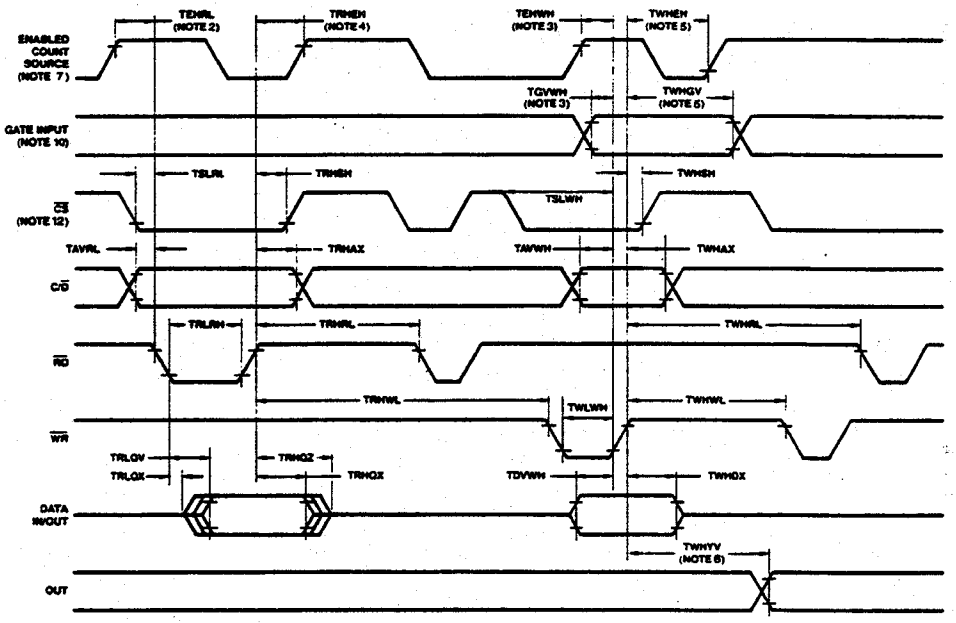


Figure 21. Bus Transfer Switching Waveforms

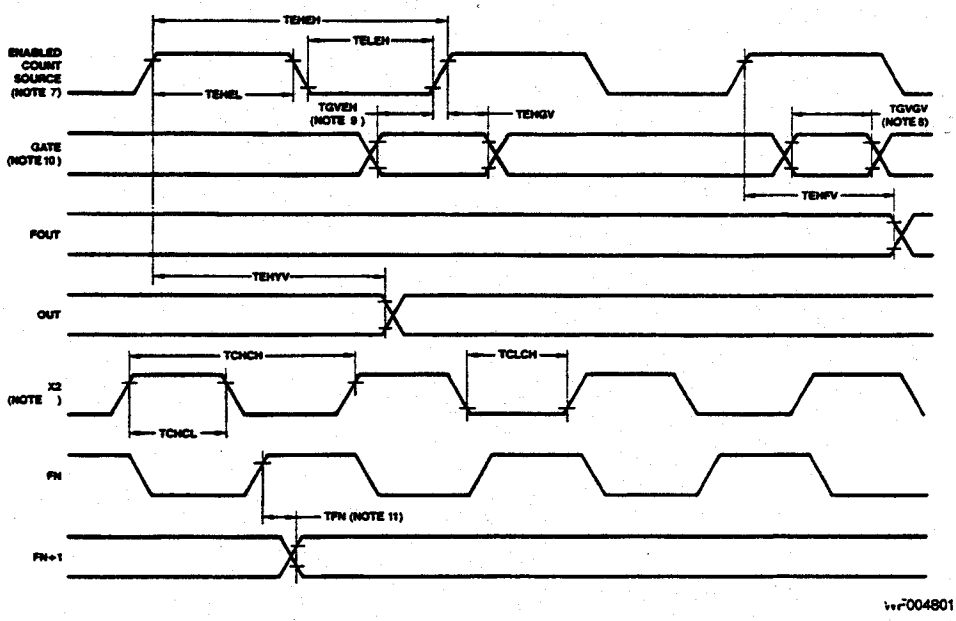


Figure 22. Counter Switching Waveforms

APPENDIX A

Design Hints

1) When a crystal is not being used, X1 and X2 should be connected as shown for TTL input (Figure A1) and no input (Figure A2).

2) Recommended oscillator capacitor values are 18 pF on X1 and X2.

3) Unused inputs should be tied to VCC.

4) The TC output can glitch when the counter is loaded. For this reason this output should not be connected to edge sensitive interrupts. The counter output should be set or cleared after the LOAD command.

5. The two most significant bits of the status register are not specified. They may be zero or one.

6. The mode register should not be modified when the counter is armed.

7. The LOAD and HOLD registers should not be changed during TC.

8. When using the different clocks for different counters be aware that there is a 75 ns skew between F1, F2, F3, F4 and F5.

9. The TC output will remain inactive if programmed to be in the TC TOGGLE mode and the step command is used to increment or decrement the counter. The output will go into TC if programmed to be in the active High or active Low terminal count modes. The only two ways out of TC in this case are:

—Arming the counter and having an active source connected to it.

—Issuing another step command.

10) Timing parameters TEHWH and TGVWH are specified as negative. The diagrams in Figure A3 show the relationship between these signals.

11) In mode X the counter will count all qualified source edges until the second (not the first) TC and then stop.

12) A TC can occur when the counters are loaded if the counter was stopped at $FFFF_H$ or 9999_{10} in the count up mode or at count 0001 when counting down. This is because an internal TC is generated which forces TC to be generated on the next count pulse.

13) In modes that alternate the reload source between the load and the hold registers (e.g., mode J), if the counter is disarmed at 0001_H for down counting or 9999_{10} for BCD up counting or $FFFF_H$ for binary up counting and rearmed, the reload source after the first TC will be the load register instead of the hold register. To avoid this, issue a software "dummy" load to the counter immediately after the disarm command.

14) In the down counting mode of the Am9513A, if a 0001 is loaded into the counter and another LOAD COUNTER command is issued, the TC of that counter will go active. If the load register contents are subsequently changed, and the counter armed, the first clock edge will cause the new load register contents to transfer into the counter and the next clock edge will decrement the counter and make it go out of TC.

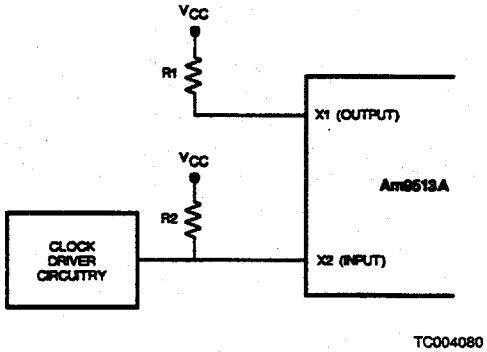
15) Glitches on $\overline{CS}$ just before the $\overline{RD}$ or $\overline{WR}$ pulse may cause the part to behave incorrectly.

16) Timing parameters TGVEH & TEHGV must not be violated; Figure A4 shows a method.

Troubleshooting (Symptom: Solution)

1) Registers not being programmed correctly: Check READ or WRITE recovery time.

2) Setup and hold problems observed in synchronous systems: Try switching from positive edge to negative edge triggering.



R1 = 6.8 kΩ ±10%
R2 is a function of Driver Circuitry to meet
X2 VIH = 3.8 V
X2 VIL = 0.8 V

Figure A1. Crystal Input Configuration

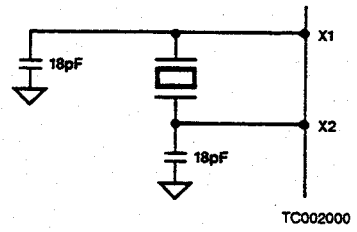


Figure A2. Crystal Input Configuration

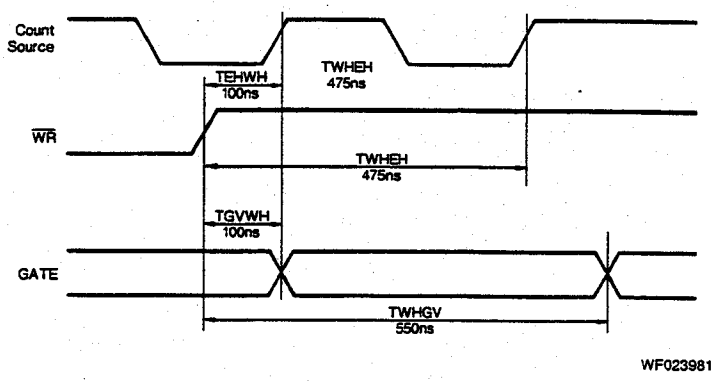


Figure A3. TEHWH/TGVWH Timing Diagram

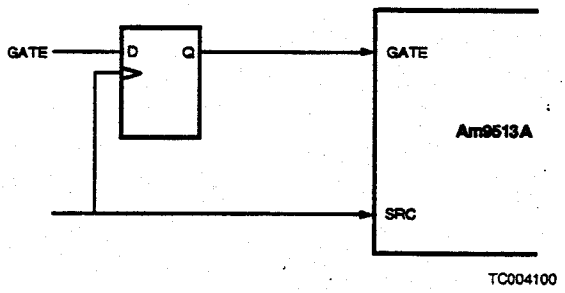


Figure A4. GATE/SRC Configuration Suggestion

Appendix F
Oki MSM82C55A Data Sheet*

This appendix contains the manufacturer data sheet for the MSM82C55A CMOS Programmable Peripheral Interface (Oki Semiconductor). This device is used on the AT-MIO-16D.

OKI semiconductor

MSM82C55A-2RS/GS/VJS

CMOS PROGRAMMABLE PERIPHERAL INTERFACE

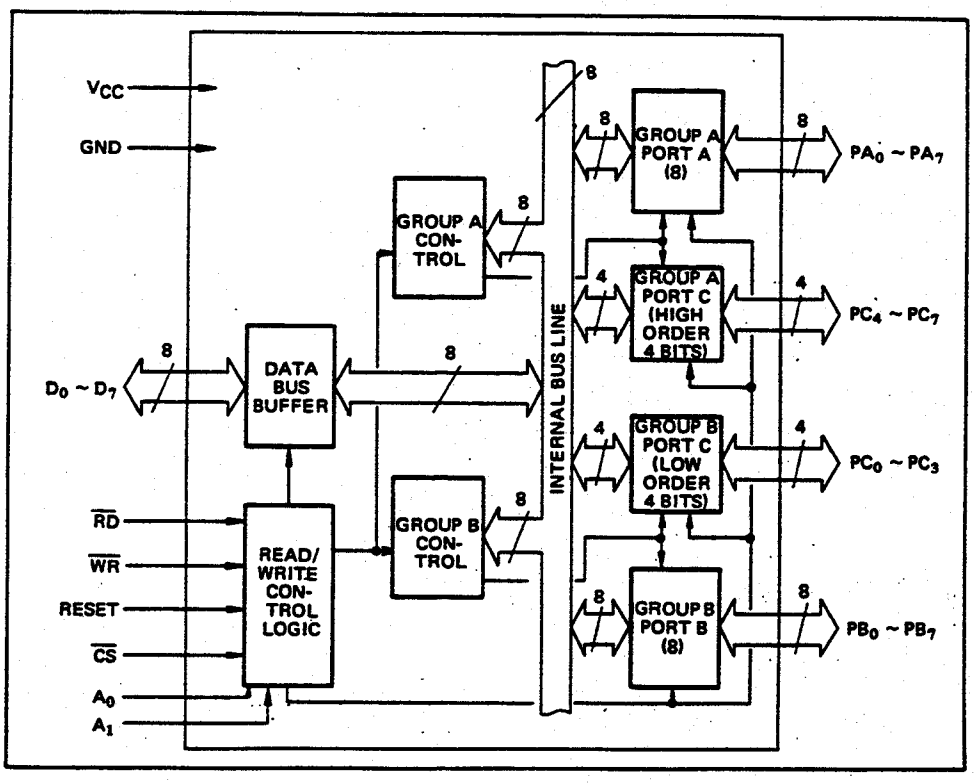
GENERAL DESCRIPTION

The MSM82C55A is a programmable universal I/O interface device which operates as high speed and on low power consumption due to 3 μ silicon gate CMOS technology. It is the best fit as an I/O port in a system which employs the 8-bit parallel processing MSM80C85A CPU. This device has 24-bit I/O pins equivalent to three 8-bit I/O ports and all inputs/outputs are TTL interface compatible.

FEATURES

- High speed and low power consumption due to 3 μ silicon gate CMOS technology
- 3 V to 6 V single power supply
- Full static operation
- Programmable 24-bit I/O ports
- Bidirectional bus operation (Port A)
- Bit set/reset function (Port C)
- TTL compatible
- Compatible with 8255A-5
- 40 pin Plastic DIP (DIP40-P-600)
- 44 pin PLCC (QFJ44-P-S650)
- 44 pin-V Plastic QFP (QFP44-P-910-VK)
- 44 pin-VI Plastic QFP (QFP44-P-910-VIK)

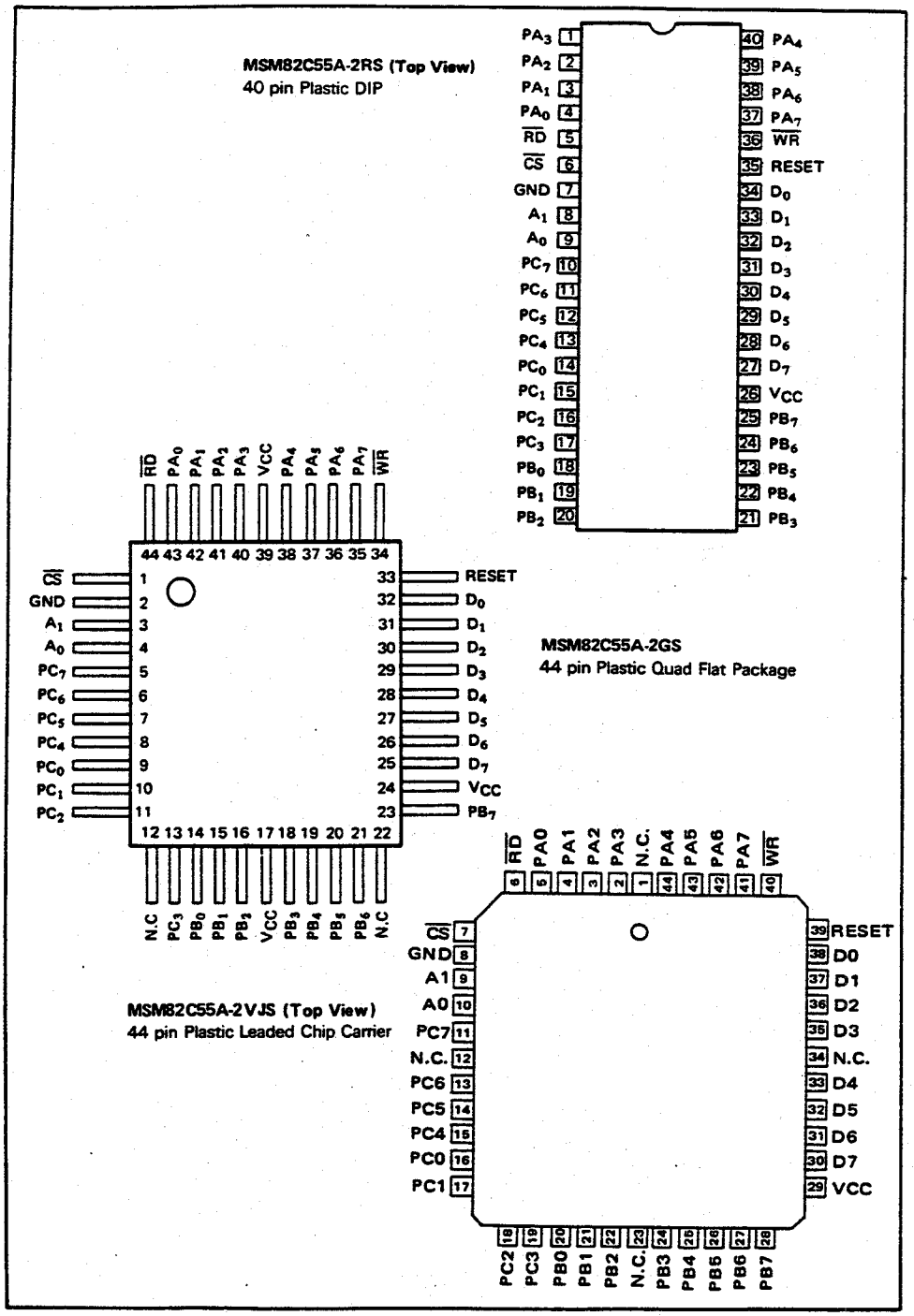
CIRCUIT CONFIGURATION



5

■ I/O-MSM82C55A-2RS/GS/VJS ■

PIN CONFIGURATION



I/O·MSM82C55A-2RS/GS/VJS

ABSOLUTE MAXIMUM RATINGS

Parameter	Symbol	Conditions	Limits			Unit
			MSM82C55A-2RS	MSM82C55A-2GS	MSM82C55A-2VJS	
Ssupply Voltage	V_{CC}	Ta = 25°C with respect to GND	−0.5 to +7			V
Input Voltage	V_{IN}		−0.5 to V_{CC} + 0.5			V
Output Voltage	V_{OUT}		−0.5 to V_{CC} + 0.5			V
Storage Temperature	T_{stg}	—	−55 to +150			°C
Power Dissipation	P_D	Ta = 25°C	1.0	0.7	1.0	W

OPERATING RANGE

Parameter	Symbol	Limits	Unit
Supply Voltage	V_{CC}	3 to 6	V
Operating Temperature	T_{OP}	−40 to 85	°C

RECOMMENDED OPERATING RANGE

Parameter	Symbol	Min.	Typ.	Max.	Unit
Supply Voltage	V_{CC}	4.5	5	5.5	V
Operating Temperature	T_{OP}	−40	+25	+85	°C
"L" Input Voltage	V_{IL}	−0.3		+0.8	V
"H" Input Voltage	V_{IH}	2.2		V_{CC} +0.3	V

5

DC CHARACTERISTICS

Parameter	Symbol	Conditions		MSM82C55A-2			Unit
				Min.	Typ.	Max.	
"L" Output Voltage	V_{OL}	I_{OL} = 2.5 mA	V_{CC} = 4.5V to 5.5V Ta = −40°C to +85°C (C_L = 0pF)			0.4	V
"H" Output Voltage	V_{OH}	I_{OH} = −40 μA		4.2			V
		I_{OH} = −2.5 mA		3.7			V
Input Leak Current	I_{LI}	$0 \leq V_{IN} \leq V_{CC}$		−1		1	μA
Output Leak Current	I_{LO}	$0 \leq V_{OUT} \leq V_{CC}$		−10		10	μA
Supply Current (standby)	I_{CCS}	$\overline{CS} \geqq V_{CC}$ −0.2V $V_{IH} \geqq V_{CC}$ −0.2V $V_{IL} \leqq$ 0.2V			0.1	10	μA
Average Supply Current (active)	I_{CC}	I/O write cycle 82C55A-2 ... 8MHz CPU timing				8	mA

■ I/O·MSM82C55A-2RS/GS/VJS ■

AC CHARACTERISTICS

(V_{CC} = 4.5 to 5.5V, Ta = −40 to +80°C)

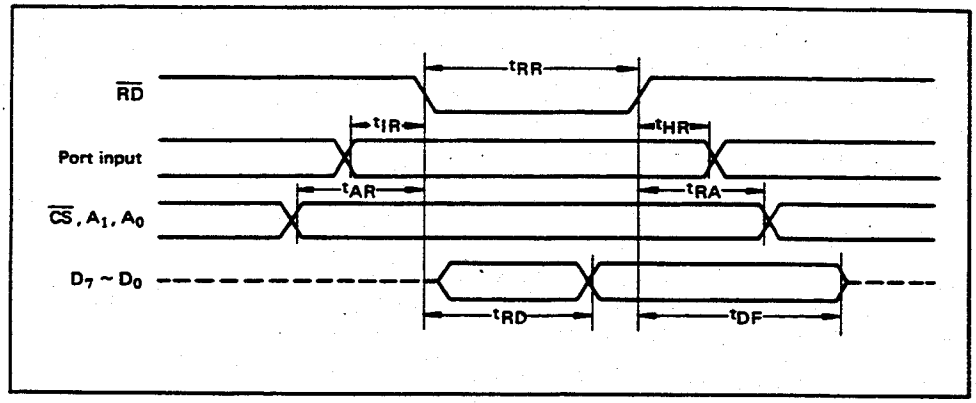
Parameter	Symbol	MSM82C55A-2		Unit	Remarks
		Min.	Max.		
Setup Time of address to the falling edge of $\overline{RD}$	t_{AR}	20		ns	
Hold Time of address to the rising edge of $\overline{RD}$	t_{RA}	0		ns	
$\overline{RD}$ Pulse Width	t_{RR}	100		ns	
Delay Time from the falling edge of $\overline{RD}$ to the output of defined data	t_{RD}		120	ns	
Delay Time from the rising edge of $\overline{RD}$ to the floating of data bus	t_{DF}	10	75	ns	
Time from the rising edge of $\overline{RD}$ or $\overline{WR}$ to the next falling edge of $\overline{RD}$ or $\overline{WR}$	t_{RV}	200		ns	
Setup Time of address before the falling edge of $\overline{WR}$	t_{AW}	0		ns	
Hold Time of address after the rising edge or $\overline{WR}$	t_{WA}	20		ns	
$\overline{WR}$ Pulse Width	t_{WW}	150		ns	
Setup Time of bus data before the rising edge of $\overline{WR}$	t_{DW}	50		ns	
Holt Time of bus data after the rising edge of $\overline{WR}$	t_{WD}	30		ns	
Delay Time from the rising edge of $\overline{WR}$ to the output of defined data	t_{WB}		200	ns	
Setup Time of port data before the falling edge of $\overline{RD}$	t_{IR}	20		ns	
Hold Time of port data after the rising edge of $\overline{RD}$	t_{HR}	10		ns	
$\overline{ACK}$ Pulse Width	t_{AK}	100		ns	
$\overline{STB}$ Pulse Width	t_{ST}	100		ns	Load 150 pF
Setup Time of port data before the rising edge of $\overline{STB}$	t_{PS}	20		ns	
Hold Time of port data after the rising edge of $\overline{STB}$	t_{PH}	50		ns	
Delay Time from the falling edge of $\overline{ACK}$ to the output of defined data	t_{AD}		150	ns	
Delay Time from the rising edge of $\overline{ACK}$ to the floating of port (Port A in mode 2)	t_{KD}	20	250	ns	
Delay Time from the rising edge of WR to the falling edge of $\overline{OBF}$	t_{WOB}		150	ns	
Delay Time from the falling edge of ACK to the rising edge of $\overline{OBF}$	t_{AOB}		150	ns	
Delay Time from the falling edge of STB to the rising edge of IBF	t_{SIB}		150	ns	
Delay Time from the rising edge of $\overline{RD}$ to the falling edge of IBF	t_{RIB}		150	ns	
Delay Time from the falling edge of $\overline{RD}$ to the falling edge of INTR	t_{RIT}		200	ns	
Delay Time from the rising edge of $\overline{STB}$ to the rising edge of INTR	t_{SIT}		150	ns	
Delay Time from the rising edge of $\overline{ACK}$ to the rising edge of INTR	t_{AIT}		150	ns	
Delay Time from the falling edge of $\overline{WR}$ to the falling edge of INTR	t_{WIT}		250	ns	

Note: Timing is measured at V_L = 0.8 V and V_H = 2.2 V for both input and outputs.

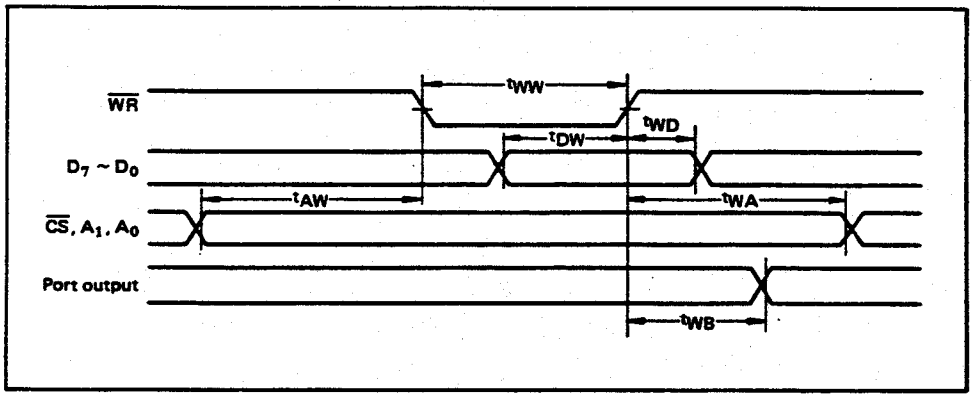
5

■ I/O-MSM82C55A-2RS/GS/VJS ■

Basic Input Operation (Mode 0)

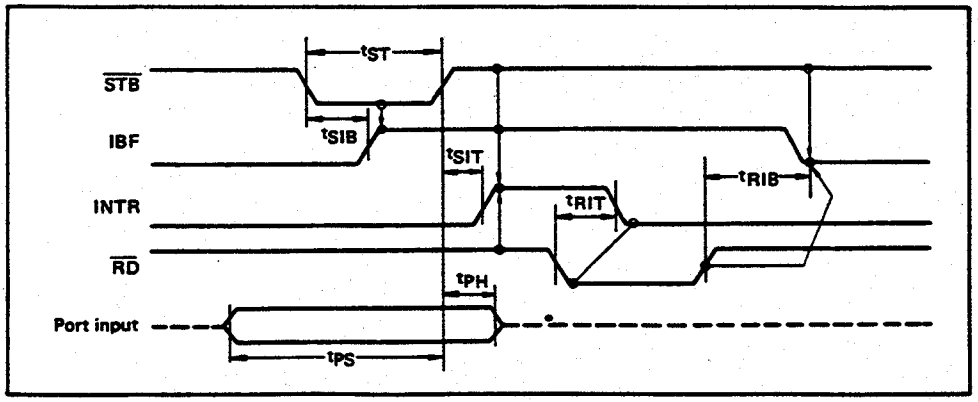


Basic Output Operation (Mode 0)



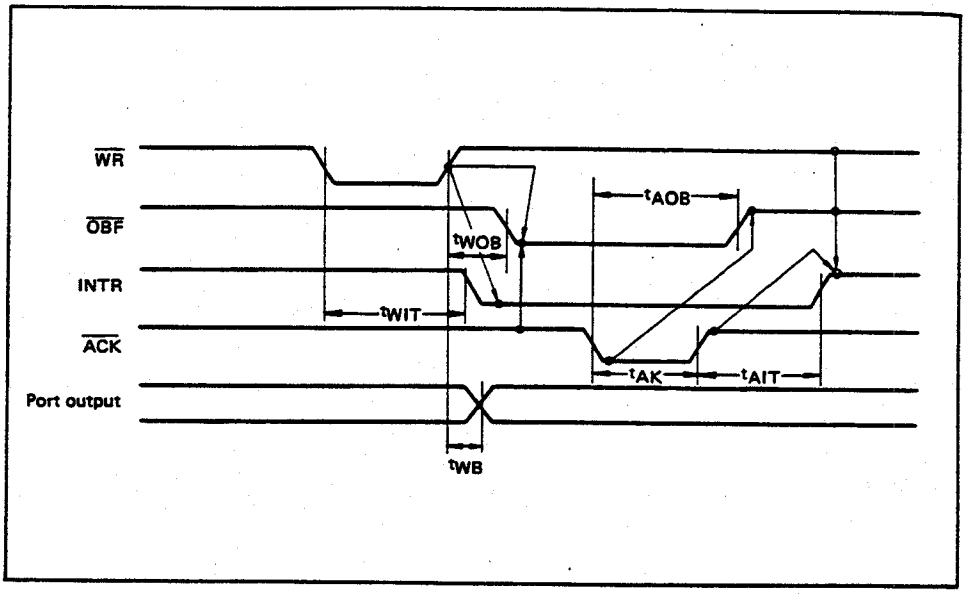
5

Strobe Input Operation (Mode 1)

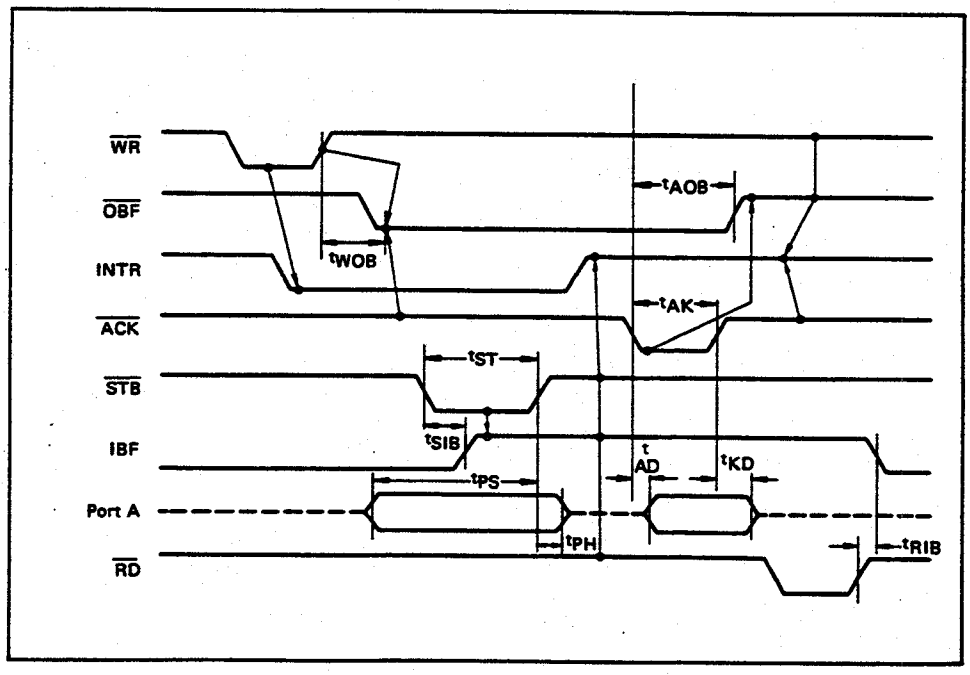


■ I/O-MSM82C55A-2RS/GS/VJS ■

Strobe Output Operation (Mode 1)

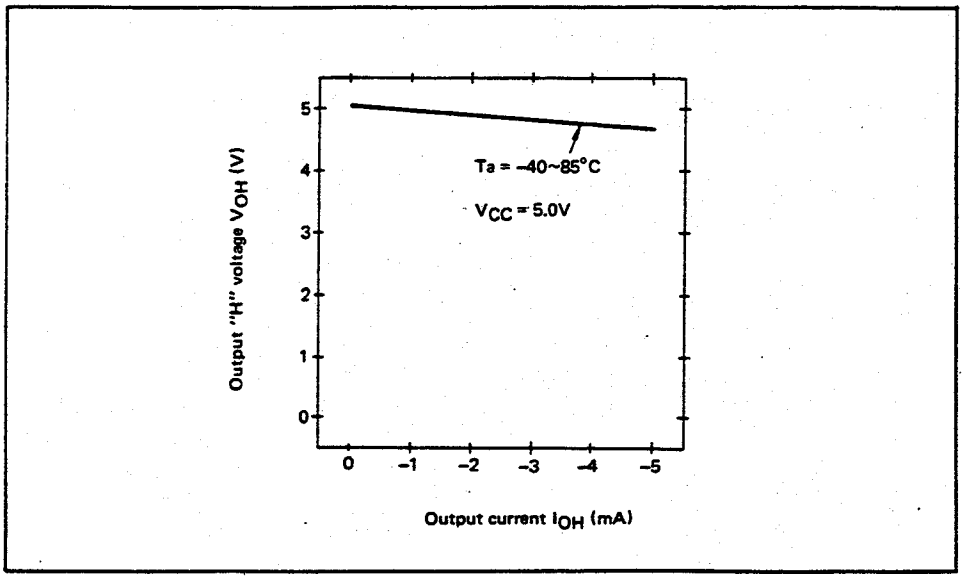


Bidirectional Bus Operation (Mode 2)

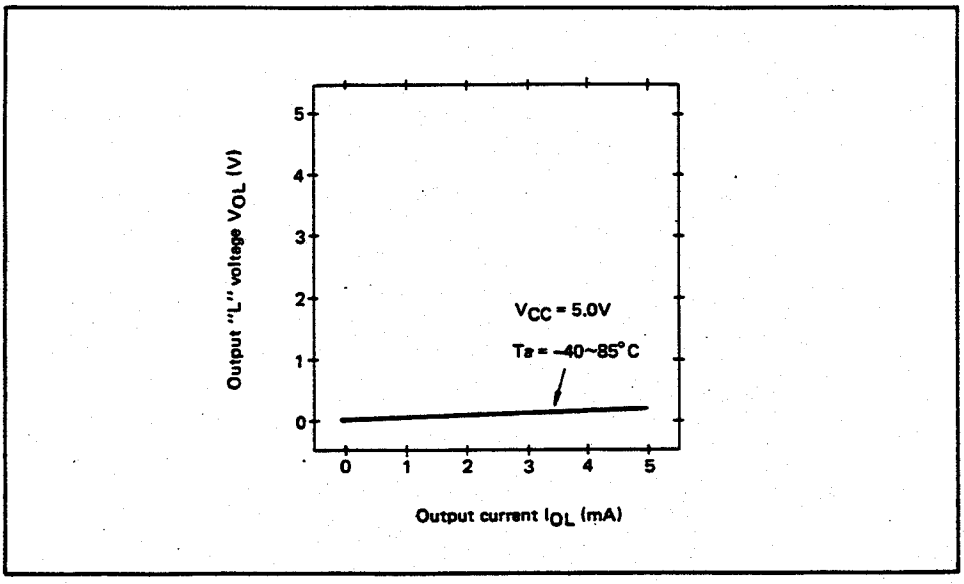


OUTPUT CHARACTERISTICS (REFERENCE VALUE)

1 Output "H" Voltage (V_{OH}) vs. Output Current (I_{OH})



2 Output "L" Voltage (V_{OL}) vs. Output Current (I_{OL})



Note: The direction of flowing into the device is taken as positive for the output current.

5

■ I/O·MSM82C55A-2RS/GS/VJS ■

FUNCTIONAL DESCRIPTION OF PIN

Pin No.	Item	Input/Output	Function
D7 ~ D0	Bidirectional data bus	Input and output	These are three-state 8-bit bidirectional buses used to write and read data upon receipt of the $\overline{WR}$ and $\overline{RD}$ signals from CPU and also used when control words and bit set/reset data are transferred from CPU to MSM82C55A.
RESET	Reset input	Input	This signal is used to reset the control register and all internal registers when it is in high level. At this time, ports are all made into the input mode (high impedance status). all port latches are cleared to 0. and all ports groups are set to mode 0.
$\overline{CS}$	Chip select input	Input	When the $\overline{CS}$ is in low level, data transmission is enabled with CPU. When it is in high level, the data bus is made into the high impedance status where no write nor read operation is performed. Internal registers hold their previous status, however.
$\overline{RD}$	Read input	Input	When RD is in low level, data is transferred from MSM82C55A to CPU.
$\overline{WR}$	Write input	Input	When WR is in low level, data or control words are transferred from CPU to MSM82C55A.
A0, A1	Port select input (address)	Input	By combination of A0 and A1, either one is selected from among port A, port B, port C, and control register. These pins are usually connected to low order 2 bits of the address bus.
PA7 ~ PA0	Port A	Input and output	These are universal 8-bit I/O ports. The direction of inputs/outputs can be determined by writing a control word. Especially, port A can be used as a bidirectional port when it is set to mode 2.
PB7 ~ PB0	Port B	Input and output	These are universal 8-bit I/O ports. The direction of inputs/outputs can be determined by writing a control word.
PC7 ~ PC0	Port C	Input and output	These are universal 8-bit I/O ports. The direction of inputs/outputs can be determined by writing a control word as 2 ports with 4 bits each. When port A or port B is used in mode 1 or mode 2 (port A only), they become control pins. Especially when port C is used as an output port, each bit can be set/reset independently.
V_{CC}			+5 V power supply.
GND			GND

5

BASIC FUNCTIONAL DESCRIPTION

Group A and Group B

When setting a mode to a port having 24 bits, set it by dividing it into two groups of 12 bits each.

- Group A: Port A (8 bits) and high order 4 bits of port C (PC7 ~ PC4)
- Group B: Port B (8 bits) and low order 4 bits of port C (PC3 ~ PC0)

Mode 0, 1, 2

There are 3 types of modes to be set by grouping as follows:

- Mode 0: Basic input operation/output operation (Available for both groups A and B)
- Mode 1: Strobe input operation/output operation (Available for both groups A and B)
- Mode 2: Bidirectional bus operation (Available for group A only)

When used in mode 1 or mode 2, however, port C has bits to be defined as ports for control signal for operation ports (port A for group A and port B for group B) of their respective groups.

Port A, B, C

The internal structure of 3 ports is as follows:

- Port A: One 8-bit data output latch/buffer and one 8-bit data input latch
- Port B: One 8-bit data input/output latch/buffer and one 8-bit data input buffer
- Port C: One 8-bit data output latch/buffer and one 8-bit data input buffer (no latch for input)

Single bit set/reset function for port C

When port C is defined as an output port, it is possible to set (to turn to high level) or reset (to turn to low level) any one of 8 bits individually without affecting other bits.

■ I/O-MSM82C55A-2RS/GS/VJS ■

OPERATIONAL DESCRIPTION

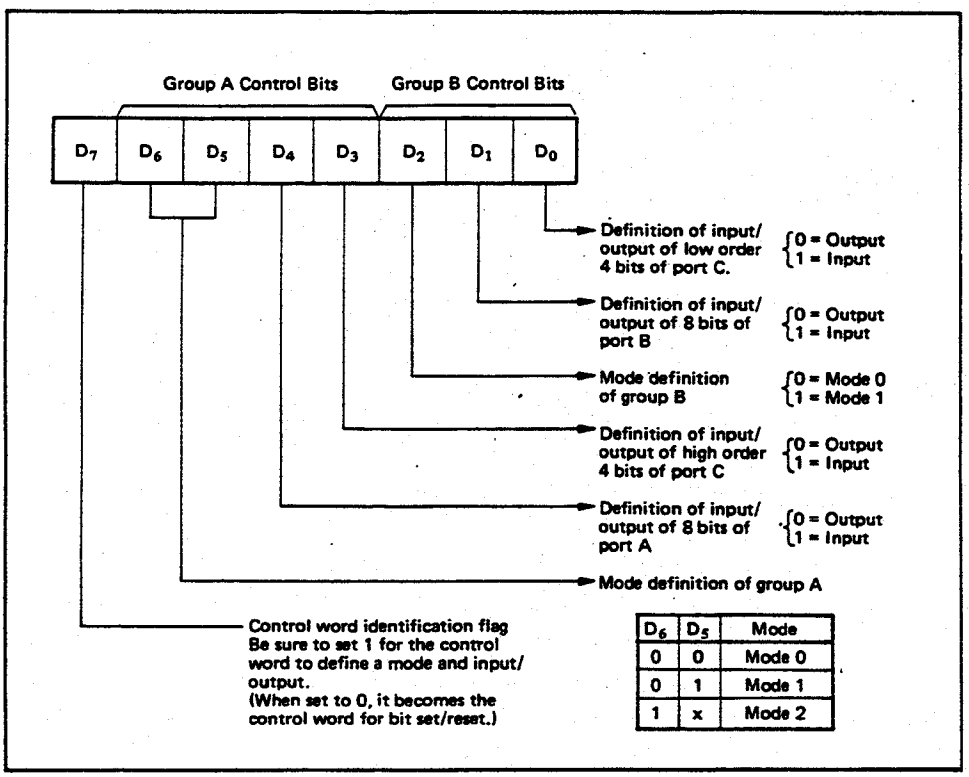
Control Logic

Operations by addresses and control signals, e.g., read and write, etc. are as shown in the table below:

Operation	A1	A0	$\overline{CS}$	$\overline{WR}$	$\overline{RD}$	Operation
Input	0	0	0	1	0	Port A → Data Bus
	0	1	0	1	0	Port B → Data Bus
	1	0	0	1	0	Port C → Data Bus
Output	0	0	0	0	1	Data Bus → Port A
	0	1	0	0	1	Data Bus → Port B
	1	0	0	0	1	Data Bus → Port C
Control	1	1	0	0	1	Data Bus → Control Register
Others	1	1	0	1	0	Illegal Condition
	x	x	1	x	x	Data bus is in the high impedance status.

Setting of Control Word

The control register is composed of 7-bit latch circuit and 1-bit flag as shown below.



D_6	D_5	Mode
0	0	Mode 0
0	1	Mode 1
1	x	Mode 2

Precaution for mode selection

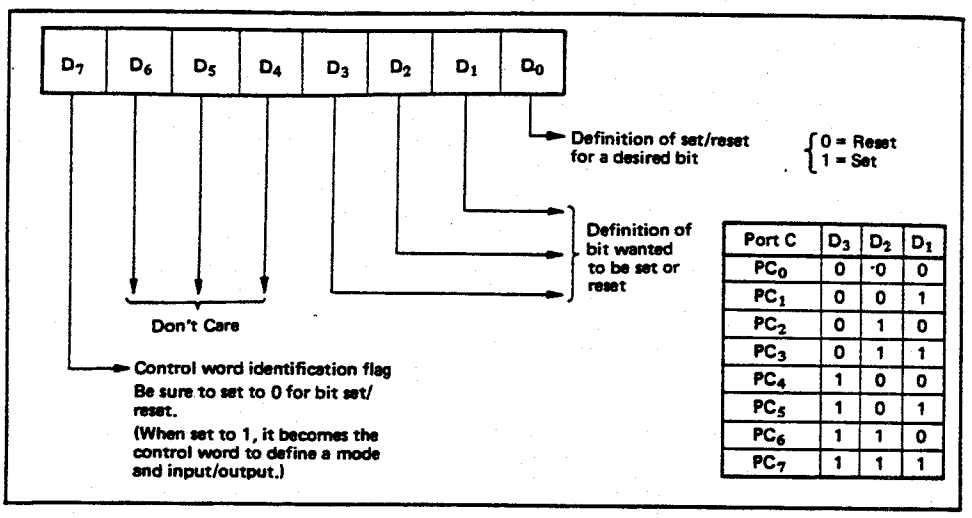
The output registers for ports A and C are cleared to ϕ each time data is written in the command register and the mode is changed, but the port B state is undefined.

Bit Set/Reset Function

When port C is defined as output port, it is possible to set (set output to 1) or reset (set output to 0) any one of 8 bits without affecting other bits as shown next page.

5

■ I/O-MSM82C55A-2RS/GS/VJS ■



Port C	D_3	D_2	D_1
PC_0	0	0	0
PC_1	0	0	1
PC_2	0	1	0
PC_3	0	1	1
PC_4	1	0	0
PC_5	1	0	1
PC_6	1	1	0
PC_7	1	1	1

Interrupt Control Function

When the MSM82C55A is used in mode 1 or mode 2, the interrupt signal for the CPU is provided. The interrupt request signal is output from port C. When the internal flip-flop INTE is set beforehand at this time, the desired interrupt request signal is output. When it is reset beforehand, however, the interrupt request signal is not output. The set/reset of the internal flip-flop is made by the bit set/reset operation for port C virtually.

Bit set → INTE is set → Interrupt allowed
Bit reset → INTE is reset → Interrupt inhibited

Operational Description by Mode

1. Mode 0 (Basic input/output operation)

Mode 0 makes the MSM82C55A operate as a basic input port or output port. No control signals such as interrupt request, etc. are required in this mode. All 24 bits can be used as two-8-bit ports and two 4-bit ports. Sixteen combinations are then possible for inputs/outputs. The inputs are not latched, but the outputs are.

5

Type	Control Word								Group A		Group B	
	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Port A	High Order 4 Bits of Port C	Port B	Low Order 4 Bits of Port C
1	1	0	0	0	0	0	0	0	Output	Output	Output	Output
2	1	0	0	0	0	0	0	1	Output	Output	Output	Input
3	1	0	0	0	0	0	1	0	Output	Output	Input	Output
4	1	0	0	0	0	0	1	1	Output	Output	Input	Input
5	1	0	0	0	1	0	0	0	Output	Input	Output	Output
6	1	0	0	0	1	0	0	1	Output	Input	Output	Input
7	1	0	0	0	1	0	1	0	Output	Input	Input	Output
8	1	0	0	0	1	0	1	1	Output	Input	Input	Input
9	1	0	0	1	0	0	0	0	Input	Output	Output	Output
10	1	0	0	1	0	0	0	1	Input	Output	Output	Input
11	1	0	0	1	0	0	1	0	Input	Output	Input	Output
12	1	0	0	1	0	0	1	1	Input	Output	Input	Input
13	1	0	0	1	1	0	0	0	Input	Input	Output	Output
14	1	0	0	1	1	0	0	1	Input	Input	Output	Input
15	1	0	0	1	1	0	1	0	Input	Input	Input	Output
16	1	0	0	1	1	0	1	1	Input	Input	Input	Input

Note: When used in mode 0 for both groups A and B

2. Mode 1 (Strobe input/output operation)

In mode 1, the strobe, interrupt and other control signals are used when input/output operations are made from a specified port. This mode is available for both groups A and B. In group A at this time, port A is used as the data line and port C as the control signal.

Following is a descrption of the input operation in mode 1.

$\overline{STB}$ (Strobe input)

- When this signal is low level, the data output from terminal to port is fetched into the internal latch of the port. This can be made independent from the CPU, and the data is not output to the data bus until the RD signal arrives from the CPU.

IBF (Input buffer full flag output)

- This is the response signal for the $\overline{STB}$. This signal when turned to high level indicates that data is fetched into the input latch. This signal turns to high level at the falling edge of $\overline{STB}$ and to low level at the rising edge of $\overline{RD}$.

INTR (Interrupt request output)

- This is the interrupt request signal for the CPU of the data fetched into the input latch. It is indicated by high level only when the internal INTE flip-flop is set. This signal turns to high level at the rising edge of the $\overline{STB}$ (IBF = 1 at this time) and low level at the falling edge of the $\overline{RD}$ when the INTE is set.

$INTE_A$ of group A is set when the bit for PC_4 is set, while $INTE_B$ of group B is set when the bit for PC_2 is set.

Following is a description of the output operation of mode 1.

$\overline{OBF}$ (Output buffer full flag output)

- This signal when turned to low level indicates that data is written to the specified port upon receipt of the $\overline{WR}$ signal from the CPU. This signal turns to low level at the rising edge of the WR and high level at the falling edge of the ACK.

$\overline{ACK}$ (Acknowledge input)

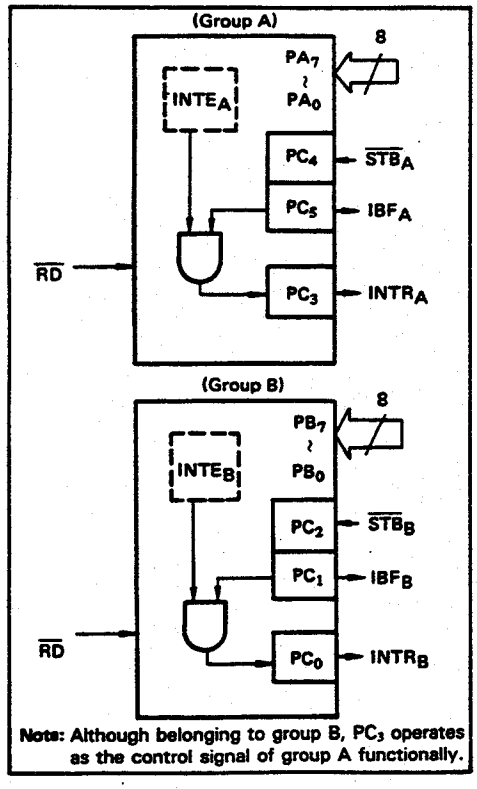
- This signal when turned to low level indicates that the terminal has received data.

INTR (Interrupt request output)

- This is the signal used to interrupt the CPU when a terminal receives data from the CPU via the MSM82C55A-5. It indicates the occurrence of the interrupt in high level only when the internal INTE flip-flop is set. This signal turns to high level at the rising edge of the $\overline{ACK}$ (OBF = 1 at this time) and low level at the falling edge of $\overline{WR}$ when the $INTE_B$ is set.

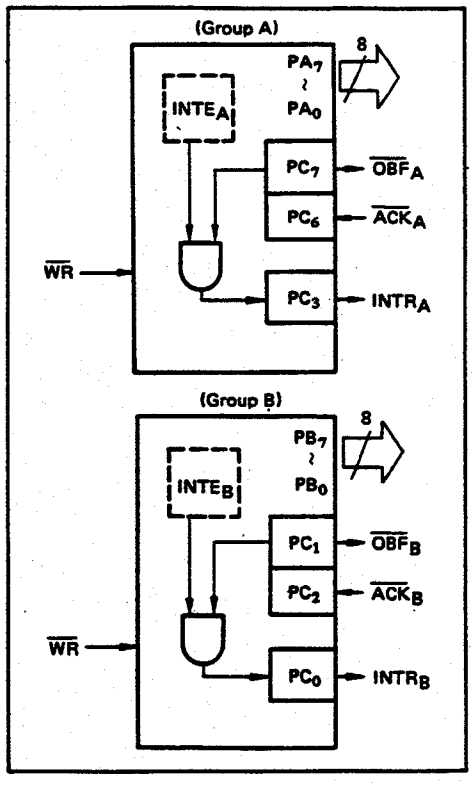
$INTE_A$ of group A is set when the bit for PC_6 is set, while $INTE_B$ of group B is set when the bit for PC_2 is set.

Mode 1 Input



Note: Although belonging to group B, PC_3 operates as the control signal of group A functionally.

Mode 1 output



5

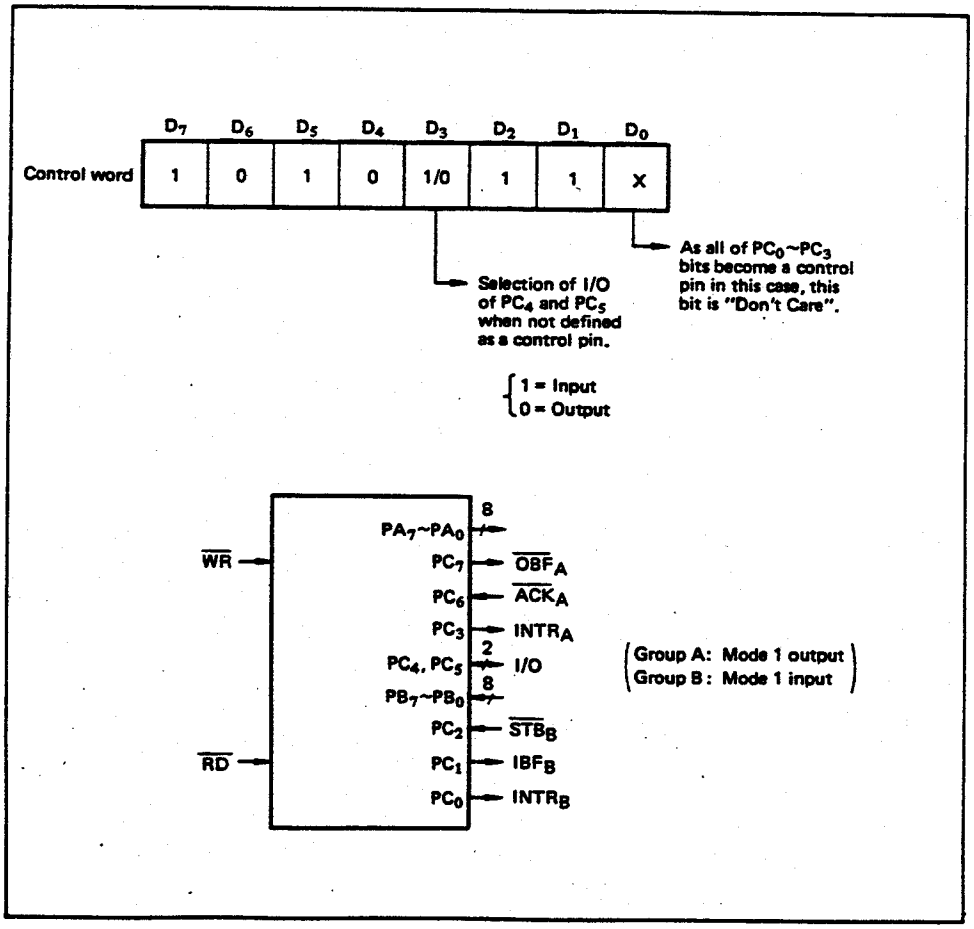
■ I/O-MSM82C55A-2RS/GS/VJS ■

Port C Function Allocation in Mode 1

Port C \ Combination of Input/Output	Group A: Input Group B: Input	Group A: Input Group B: Output	Group A: Output Group B: Input	Group A: Output Group B: Output
PC_0	$INTR_B$	$INTR_B$	$INTR_B$	$INTR_B$
PC_1	IBF_B	$\overline{OBF}_B$	IBF_B	$\overline{OBF}_B$
PC_2	$\overline{STB}_B$	$\overline{ACK}_B$	$\overline{STB}_B$	$\overline{ACK}_B$
PC_3	$INTR_A$	$INTR_A$	$INTR_A$	$INTR_A$
PC_4	$\overline{STB}_A$	$\overline{STB}_A$	I/O	I/O
PC_5	IBF_A	IBF_A	I/O	I/O
PC_6	I/O	I/O	$\overline{ACK}_A$	$\overline{ACK}_A$
PC_7	I/O	I/O	$\overline{OBF}_A$	$\overline{OBF}_A$

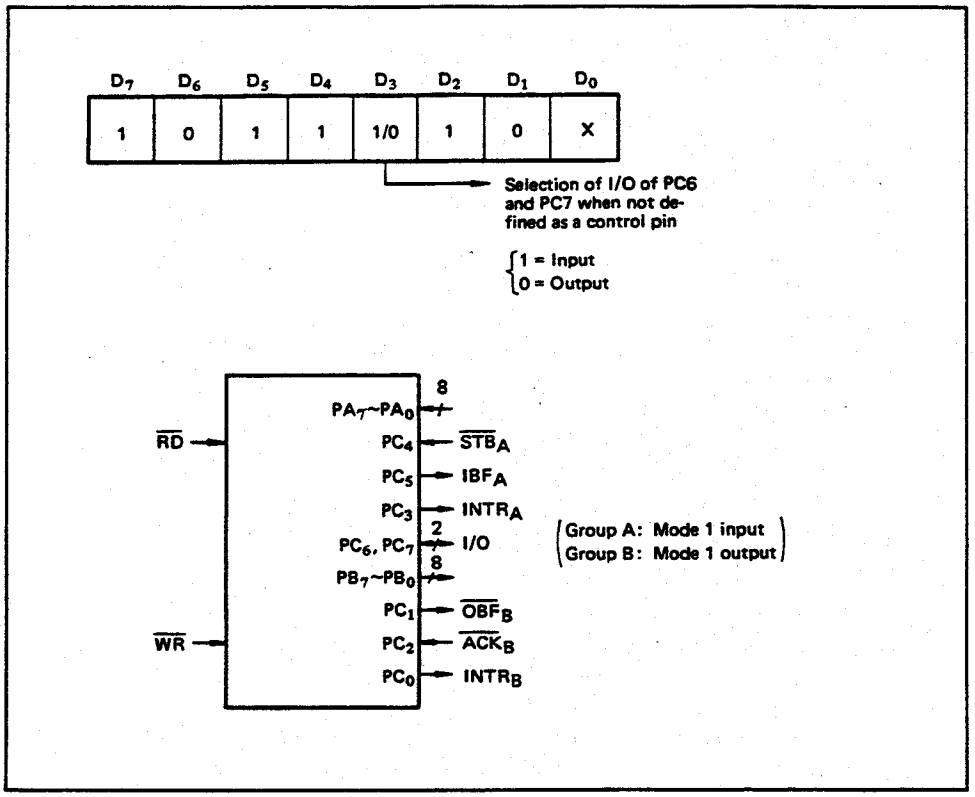
Note: I/O is a bit not used as the control signal, but it is available as a port of mode 0.

Examples of the relation between the control words and pins when used in mode 1 is shown below:
(a) When group A is mode 1 output and group B is mode 1 input.



5

(b) When group A is mode 1 input and group B is mode 1 output.



5

3. Mode 2 (Strobe bidirectional bus I/O operation)

In mode 2, it is possible to transfer data in 2 directions through a single 8-bit port. This operation is akin to a combination between input and output operations. Port C waits for the control signal in this case, too. Mode 2 is available only for group A, however.

Next, a description is made on mode 2.

$\overline{OBF}$ (Output buffer full flag output)

- This signal when turned to low level indicates that data has been written to the internal output latch upon receipt of the $\overline{WR}$ signal from the CPU. At this time, port A is still in the high impedance status and the data is not yet output to the outside. This signal turns to low level at the rising edge of the $\overline{WR}$ and high level at the falling edge of the $\overline{ACK}$.

$\overline{ACK}$ (Acknowledge input)

- When a low level signal is input to this pin, the high impedance status of port A is cleared, the buffer is enabled, and the data written to the internal output latch is output to port A. When the input returns to high level, port A is made into the high impedance status.

$\overline{STB}$ (Strobe input)

- When this signal turns to low level, the data output to the port from the pin is fetched into the internal input latch. The data is output to the data bus upon receipt of the RD signal from the CPU, but it remains in the high impedance status until then.

IBF (Input buffer full flag output)

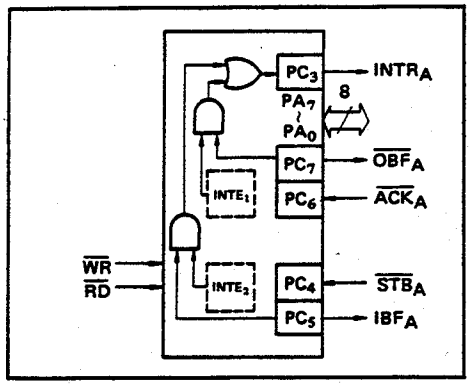
- This signal when turned to high level indicates that data from the pin has been fetched into the input latch. This signal turns to high level at the falling edge of the $\overline{STB}$ and low level at the rising edge of the $\overline{RD}$.

INTR (Interrupt request output)

- This signal is used to interrupt the CPU and its operation in the same as in mode 1. There are two INTE flip-flops internally available for input and output to select either interrupt of input or output operation. The INTE1 is used to control the interrupt request for output operation and it can be reset by the bit set for PC6. INTE2 is used to control the interrupt request for the input operation and it can be set by the bit set for PC4.

■ I/O-MSM82C55A-2RS/GS/VJS ■

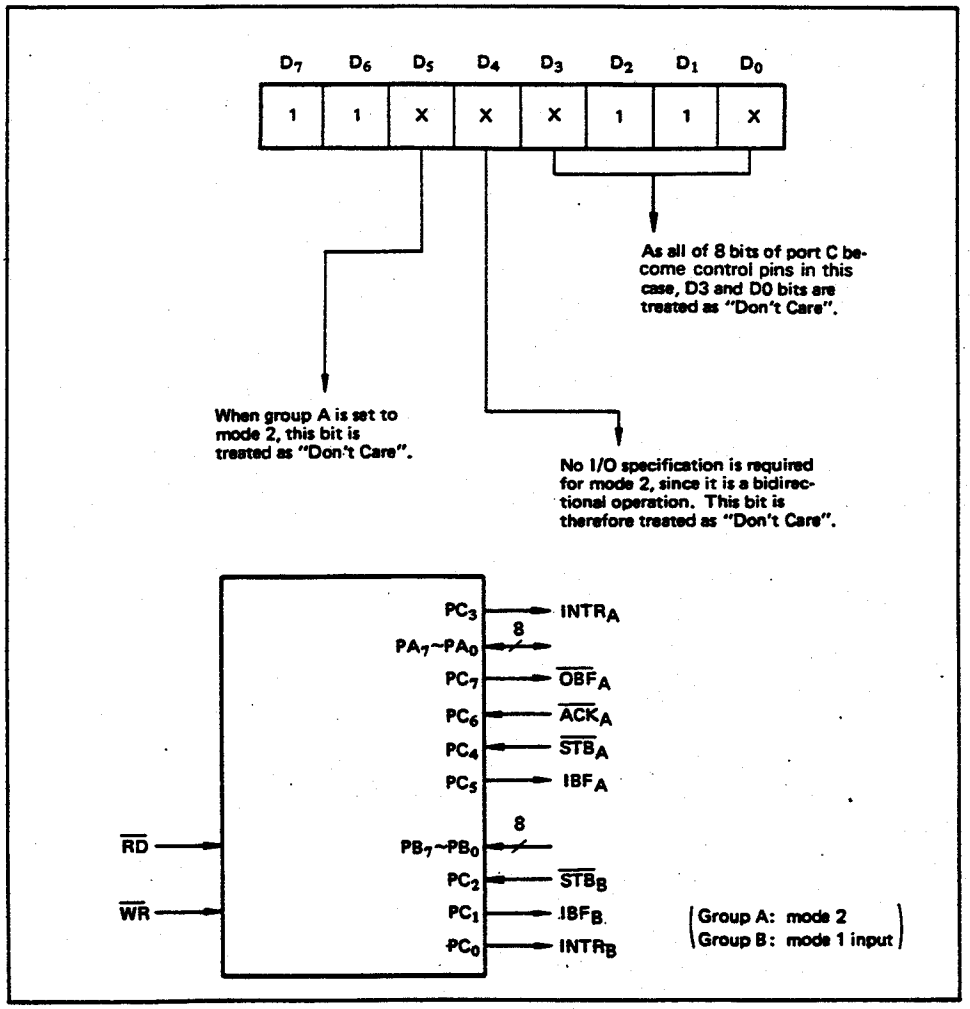
Mode 2 I/O Operation



Port C Function Allocation in Mode 2

Port C	Function
PC_0	Confirmed to the group B mode
PC_1	
PC_2	
PC_3	$INTR_A$
PC_4	$\overline{STB}_A$
PC_5	IBF_A
PC_6	$\overline{ACK}_A$
PC_7	$\overline{OBF}_A$

Following is an example of the relation between the control word and the pin when used in mode 2. When input in mode 2 for group A and in mode 1 for group B.



5

4. When Group A is Different in Mode from Group B

Group A and group B can be used by setting them in different modes each other at the same time. When either group is set to mode1 or mode 2, it is possible to set the one not defined as a control pin in port C to both input and output as a port which operates in mode 0 at the 3rd and 0th bits of the control word.

(Mode combinations that define no control bit at port C)

	Group A	Group B	Port C							
			PC_7	PC_6	PC_5	PC_4	PC_3	PC_2	PC_1	PC_0
1	Mode 1 input	Mode 0	I/O	I/O	IBF_A	$\overline{STB}_A$	$INTR_A$	I/O	I/O	I/O
2	Mode 0 output	Mode 0	$\overline{OBF}_A$	$\overline{ACK}_A$	I/O	I/O	$INTR_A$	I/O	I/O	I/O
3	Mode 0	Mode 1 input	I/O	I/O	I/O	I/O	I/O	$\overline{STB}_B$	IBF_B	$INTR_B$
4	Mode 0	Mode 1 output	I/O	I/O	I/O	I/O	I/O	$\overline{ACK}_B$	$\overline{OBF}_B$	$INTR_B$
5	Mode 1 input	Mode 1 input	I/O	I/O	IBF_A	$\overline{STB}_A$	$INTR_A$	$\overline{STB}_B$	IBF_B	$INTR_B$
6	Mode 1 input	Mode 1 output	I/O	I/O	IBF_A	$\overline{STB}_A$	$INTR_A$	$\overline{ACK}_B$	$\overline{OBF}_B$	$INTR_B$
7	Mode 1 output	Mode 1 input	$\overline{OBF}_A$	$\overline{ACK}_A$	I/O	I/O	$INTR_A$	$\overline{STB}_B$	IBF_B	$INTR_B$
8	Mode 1 output	Mode 1 output	$\overline{OBF}_A$	$\overline{ACK}_A$	I/O	I/O	$INTR_A$	$\overline{ACK}_B$	$\overline{OBF}_B$	$INTR_B$
9	Mode 2	Mode 0	$\overline{OBF}_A$	$\overline{ACK}_A$	IBF_A	$\overline{STB}_A$	$INTR_A$	I/O	I/O	I/O
			Controlled at the 3rd bit (D3) of the control word				Controlled at the 0th bit (D0) of the control word			

When the I/O bit is set to input in this case, it is possible to access data by the normal port C read operation.
When set to output, PC7 – PC4 bits can be accessed by the bit set/reset function only. Meanwhile, 3 bits from PC2 to PC0 can be accessed by normal write operation.

The bit set/reset function can be used for all of PC3 ~ PC0 bits. Note that the status of port C varies according to the combination of modes like this.

5

■ I/O·MSM82C55A-2RS/GS/VJS ■

5. Port C Status Read

When port C is used for the control signal, that is, in either mode 1 or mode 2, each control signal and bus status signal can be read out by reading the content of port C.
The status read out is as follows:

	Group A	Group B	Status read on the data bus							
			D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	Mode 1 input	Mode 0	I/O	I/O	IBF_A	$INTE_A$	$INTR_A$	I/O	I/O	I/O
2	Mode 1 output	Mode 0	$\overline{OBF}_A$	$INTE_A$	I/O	I/O	$INTR_A$	I/O	I/O	I/O
3	Mode 0	Mode 1 input	I/O	I/O	I/O	I/O	I/O	$INTE_B$	IBF_B	$INTR_B$
4	Mode 0	Mode 1 output	I/O	I/O	I/O	I/O	I/O	$INTE_B$	$\overline{OBF}_B$	$INTR_B$
5	Mode 1 input	Mode 1 input	I/O	I/O	IBF_A	$INTE_A$	$INTR_A$	$INTE_B$	IBF_B	$INTR_B$
6	Mode 1 input	Mode 1 output	I/O	I/O	IBF_A	$INTE_A$	$INTR_A$	$INTE_B$	$\overline{OBF}_B$	$INTR_B$
7	Mode 1 output	Mode 1 input	$\overline{OBF}_A$	$INTE_A$	I/O	I/O	$INTR_A$	$INTE_B$	IBF_B	$INTR_B$
8	Mode 1 output	Mode 1 output	$\overline{OBF}_A$	$INTE_A$	I/O	I/O	$INTR_A$	$INTE_B$	$\overline{OBF}_B$	$INTR_B$
9	Mode 2	Mode 0	$\overline{OBF}_A$	$INTE_1$	IBF_A	$INTE_2$	$INTR_A$	I/O	I/O	I/O
10	Mode 2	Mode 1 input	$\overline{OBF}_A$	$INTE_1$	IBF_A	$INTE_2$	$INTR_A$	$INTE_B$	IBF_B	$INTR_B$
11	Mode 2	Mode 1 output	$\overline{OBF}_A$	$INTE_1$	IBF_A	$INTE_2$	$INTR_A$	$INTE_B$	$\overline{OBF}_B$	$INTR_B$

5

6. Reset of MSM82C55A

Be sure to keep the RESET signal at power ON in the high level at least for 50 μs. Subsequently, it becomes the input mode at a high level pulse above 500 ns.

Note:

MSM82C55A-5

After a write command is executed to the command register, the internal latch is cleared in PORTA PORTC. For instance, 00H is output at the beginning of a write command when the output port is assigned. However, if PORTB is not cleared at this time, PORTB is unstable. In other words, PORTB only outputs ineffective data (unstable value according to the device) during the period from after a write command is executed till the first data is written to PORTB.

MSM82C55A-2

After a write command is executed to the command register, the internal latch is cleared in All Ports(PORTA,PORTB,PORTC). 00H is ontput at the beginning of a write command when the output port is assigned.

Appendix G
Customer Communication

For your convenience, this appendix contains forms to help you gather the information necessary to help us solve technical problems you might have as well as a form you can use to comment on the product documentation. Filling out a copy of the *Technical Support Form* before contacting National Instruments helps us help you better and faster.

National Instruments provides comprehensive technical assistance around the world. In the U.S. and Canada, applications engineers are available Monday through Friday from 8:00 a.m. to 6:00 p.m. (central time). In other countries, contact the nearest branch office. You may fax questions to us at any time.

Corporate Headquarters
(512) 795-8248
Technical support fax: (800) 328-2203
(512) 794-5678

Branch Offices	Phone Number	Fax Number
Australia	(03) 879 9422	(03) 879 9179
Austria	(0662) 435986	(0662) 437010-19
Belgium	02/757.00.20	02/757.03.11
Denmark	45 76 26 00	45 76 71 11
Finland	(90) 527 2321	(90) 502 2930
France	(1) 48 14 24 00	(1) 48 14 24 14
Germany	089/741 31 30	089/714 60 35
Italy	02/48301892	02/48301915
Japan	(03) 3788-1921	(03) 3788-1923
Mexico	95 800 010 0793	95 800 010 0793
Netherlands	03480-33466	03480-30673
Norway	32-848400	32-848600
Singapore	2265886	2265887
Spain	(91) 640 0085	(91) 640 0533
Sweden	08-730 49 70	08-730 43 70
Switzerland	056/20 51 51	056/20 51 55
Taiwan	02 377 1200	02 737 4644
U.K.	0635 523545	0635 523154

Technical Support Form

Photocopy this form and update it each time you make changes to your software or hardware, and use the completed copy of this form as a reference for your current configuration. Completing this form accurately before contacting National Instruments for technical support helps our applications engineers answer your questions more efficiently.

If you are using any National Instruments hardware or software products related to this problem, include the configuration forms from their user manuals. Include additional pages if necessary.

Name _____

Company _____

Address _____

_____

Fax (___)_____ Phone (___)_____

Computer brand _____ Model _____ Processor _____

Operating system _____

Speed _____ MHz RAM _____ MB Display adapter _____

Mouse _____ yes _____ no Other adapters installed _____

Hard disk capacity _____ MB Brand _____

Instruments used _____

National Instruments hardware product model _____ Revision _____

Configuration _____

National Instruments software product _____ Version _____

Configuration _____

The problem is _____

_____

_____

_____

_____

List any error messages _____

_____

_____

_____

_____

The following steps will reproduce the problem _____

_____

_____

_____

AT-MIO-16D Hardware and Software Configuration Form

Record the settings and revisions of your hardware and software on the line to the right of each item. Complete a new copy of this form each time you revise your software or hardware configuration, and use this form as a reference for your current configuration. Completing this form accurately before contacting National Instruments for technical support helps our applications engineers answer your questions more efficiently.

National Instruments Products

- AT-MIO-16D Model Number
  (for example, AT-MIO-16DL-9) _____
- MIO-16 Circuitry Interrupt Level of AT-MIO-16D _____
  (Factory Setting: 10)
- DIO-24 Circuitry Interrupt Level of AT-MIO-16D _____
  (Factory Setting: 5)
- DMA Channels of AT-MIO-16D _____
  (Factory Setting: 6 and 7)
- Base I/O Address of AT-MIO-16D _____
  (Factory Setting: hex 0220)
- NI-DAQ Version _____

Other Products

- Computer Make and Model _____
- Microprocessor _____
- Clock Frequency _____
- Type of Video Board Installed _____
- DOS Version _____
- Programming Language _____
- Programming Language Version _____
- Other Boards in System _____
- Base I/O Address of Other Boards _____
- DMA Channels of Other Boards _____
- Interrupt Level of Other Boards _____

Documentation Comment Form

National Instruments encourages you to comment on the documentation supplied with our products. This information helps us provide quality products to meet your needs.

Title: **AT-MIO-16D User Manual**

Edition Date: **March 1995**

Part Number: **320489-01**

Please comment on the completeness, clarity, and organization of the manual.

If you find errors in the manual, please record the page numbers and describe the errors.

Thank you for your help.

Name _____

Title _____

Company _____

Address _____

_____

Phone (_____) _____

Mail to: Technical Publications
National Instruments Corporation
6504 Bridge Point Parkway, MS 53-02
Austin, TX 78730-5039

Fax to: Technical Publications
National Instruments Corporation
MS 53-02
(512) 794-5678

Index

Numbers/Symbols

A

B

C

D

E

F

G

H

I

J

L

M

N

O

P

R

S

T

U