DAQ

AT/PCI-DIO-32HS
User Manual

High-Speed Digital I/O Boards for

AT, EISA, or PCI Bus Computers

February 1997 Edition
Part Number 321464A-01

Internet Support

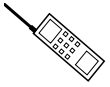
support@natinst.com
E-mail: info@natinst.com
FTP Site: ftp.natinst.com
Web Address: http://www.natinst.com

Bulletin Board Support

BBS United States: (512) 794-5422
BBS United Kingdom: 01635 551422
BBS France: 01 48 65 15 59

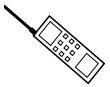
Fax-on-Demand Support

(512) 418-1111



Telephone Support (U.S.)

Tel: (512) 795-8248
Fax: (512) 794-5678



International Offices

Australia 03 9879 5166, Austria 0662 45 79 90 0, Belgium 02 757 00 20,
Canada (Ontario) 905 785 0085, Canada (Québec) 514 694 8521, Denmark 45 76 26 00,
Finland 09 527 2321, France 01 48 14 24 24, Germany 089 741 31 30, Hong Kong 2645 3186,
Israel 03 5734815, Italy 02 413091, Japan 03 5472 2970, Korea 02 596 7456,
Mexico 5 520 2635, Netherlands 0348 433466, Norway 32 84 84 00, Singapore 2265886,
Spain 91 640 0085, Sweden 08 730 49 70, Switzerland 056 200 51 51, Taiwan 02 377 1200,
U.K. 01635 523545

National Instruments Corporate Headquarters

6504 Bridge Point Parkway Austin, TX 78730-5039 Tel: (512) 794-0100

Important Information

Warranty

The AT-DIO-32HS and PCI-DIO-32HS are 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

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.

Table of Contents

About This Manual

Chapter 1
Introduction

Chapter 2
Installation and Configuration

Chapter 3
Hardware Overview

Chapter 4
Signal Connections

Chapter 5
Signal Timing

Figures

Table

About This Manual

This manual describes the electrical and mechanical aspects of the DIO-32HS boards and contains information concerning their operation and programming. Unless otherwise noted, text applies to all boards in the DIO-32HS Series.

The DIO-32HS Series includes the following boards:

- AT-DIO-32HS
- PCI-DIO-32HS

Organization of This Manual

The *AT/PCI-DIO-32HS User Manual* is organized as follows:

- Chapter 1, *Introduction*, describes the DIO-32HS boards, lists what you need to get started, describes optional equipment, and explains how to unpack your board.
- Chapter 2, *Installation and Configuration*, explains how to install and configure your DIO-32HS board.
- Chapter 3, *Hardware Overview*, provides an overview of the hardware functions of your AT-DIO-32HS or PCI-DIO-32HS board.
- Chapter 4, *Signal Connections*, describes how to make input and output signal connections to your DIO-32HS via the board I/O connector and RTSI connector.
- Chapter 5, *Signal Timing*, gives detailed timing specifications for pattern generation and for the various full, two-way handshaking modes.
- Appendix A, *Specifications*, lists the specifications for the AT-DIO-32HS and PCI-DIO-32HS boards.
- Appendix B, *Optional Adapter Description*, describes the optional 68-to-50-pin DIO-32HS adapter.

- Appendix C, *Customer Communication*, contains forms you can use to request help from National Instruments or to comment on our products.
- The *Glossary* contains an alphabetical list and descriptions of terms used in this manual, including acronyms, abbreviations, definitions, metric prefixes, mnemonics, and symbols.
- The *Index* alphabetically lists topics covered in this manual, including the page where you can find the topic.

Conventions Used in This Manual

The following conventions are used in this manual:

< >	Angle brackets containing numbers separated by an ellipsis represent a range of values associated with a bit or signal name (for example, DIOB<3..0>).
♦	The ♦ symbol indicates that the text following it applies only to a specific product, a specific operating system, or a specific software version.
bold italic	Bold italic text denotes a note, caution, or warning.
DIO-32HS	DIO-32HS refers to both the AT-DIO-32HS and the PCI-DIO-32HS boards, unless otherwise noted.
italic	Italic text denotes emphasis, a cross reference, or an introduction to a key concept. This font also denotes text from which you supply the appropriate word or value, as in Windows 3.*x*. The *Glossary* lists abbreviations, acronyms, definitions, metric prefixes, mnemonics, symbols, and terms.

National Instruments Documentation

The *AT/PCI-DIO-32HS User Manual* is one piece of the documentation set for your DAQ system. You could have any of several types of documents depending on the hardware and software in your system. Use the documentation you have as follows:

- Your DAQ hardware documentation—This documentation has detailed information about the DAQ hardware that plugs into or is connected to your computer. Use this documentation for hardware

installation and configuration instructions, specification information about your DAQ hardware, and application hints.

- Software documentation—You may have both application software and NI-DAQ software documentation. National Instruments application software includes LabVIEW, LabWindows®/CVI, and ComponentWorks. After you set up your hardware system, use either your application software documentation or the NI-DAQ documentation to help you write your application. If you have a large, complicated system, it is worthwhile to look through the software documentation before you configure your hardware.
- Accessory installation guides or manuals—If you are using accessory products, read the terminal block and cable assembly installation guides. They explain how to physically connect the relevant pieces of the system. Consult these guides when you are making your connections.

Related Documentation

The following documents contain information that you may find helpful as you read this manual:

- Your computer's technical reference manual
- *PCI Local Bus Specification*, Revision 2.0

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 C, *Customer Communication*, at the end of this manual.

Chapter 1

Introduction

This chapter describes the DIO-32HS boards, lists what you need to get started, describes optional equipment, and explains how to unpack your board.

About the DIO-32HS Board

Thank you for buying a National Instruments AT-DIO-32HS or PCI-DIO-32HS board. The DIO-32HS board is a 32-bit, parallel digital I/O interface for PC-compatible computers. The DIO-32HS offers digital data acquisition, digital waveform generation, and high-speed, flexible handshaking.

The AT-DIO-32HS is a completely switchless, jumperless DAQ board for AT (16-bit ISA) buses. The AT-DIO-32HS implements the Plug and Play ISA Specification so that software can configure all DMA channels, interrupts, and base I/O addresses. You can easily change board configurations without removing the board from your computer. The AT-DIO-32HS offers dual DMA, with channel switching, for uninterrupted, high-speed data transfer.

The PCI-DIO-32HS is a completely switchless, jumperless DAQ board for PCI buses. The PCI-DIO-32HS contains the National Instruments MITE PCI interface. The MITE offers bus-master operation, PCI burst transfers, and high-speed DMA controllers for continuous, scatter-gather DMA without requiring DMA resources from your computer.

Each DIO-32HS board contains the National Instruments DAQ-DIO chip, providing two independent channels of digital input and output, pattern generation, and handshaking. Each channel offers the following functions:

- selectable data path width (8, 16, or 32 bits)
- 16-sample-deep FIFO buffer
- 16-bit and 32-bit counters for timebase and interval generation, with a maximum timing resolution of 50 ns
- a handshaking controller implementing six flexible timing protocols
- start and stop trigger detection and digital pattern detection
- 24 mA standard or wired-OR outputs
- hysteresis and diode-based line termination on all inputs

With the DIO-32HS, you can use your computer as a digital I/O tester, logic analyzer, or system controller for laboratory testing, production testing, and industrial process monitoring and control.

For detailed DIO-32HS specifications, see Appendix A, *Specifications*.

What You Need to Get Started

To set up and use your DIO-32HS board, you will need the following:

- ❑ One of the following boards:
  AT-DIO-32HS
  PCI-DIO-32HS
- ❑ *AT/PCI-DIO-32HS User Manual*
- ❑ One of the following software packages and documentation:
  NI-DAQ for PC compatibles
  LabVIEW for Windows
  LabWindows/CVI
  ComponentWorks
- ❑ Your computer

Software Programming Choices

There are several options to choose from when programming your National Instruments DAQ hardware. You can use LabVIEW, LabWindows/CVI, ComponentWorks, or NI-DAQ.

National Instruments Application Software

LabVIEW features interactive graphics, a state-of-the art user interface, and a powerful graphical programming language. The LabVIEW Data Acquisition VI Library, a series of VIs for using LabVIEW with National Instruments DAQ hardware, is included with LabVIEW. The LabVIEW Data Acquisition VI Library is functionally equivalent to the NI-DAQ software.

LabWindows/CVI features interactive graphics, state-of-the-art user interface, and uses the ANSI standard C programming language. The LabWindows/CVI Data Acquisition Library, a series of functions for using National Instruments DAQ hardware, is included with LabWindows/CVI. The LabWindows/CVI Data Acquisition Library is functionally equivalent to the NI-DAQ software.

ComponentWorks contains tools for data acquisition and instrument control built on NI-DAQ driver software. ComponentWorks provides a higher-level programming interface for building virtual instruments through standard OLE controls and DLLs. With ComponentWorks, you can use all of the configuration tools, resource management utilities, and interactive control utilities included with NI-DAQ.

Using LabVIEW, LabWindows/CVI, or ComponentWorks software will greatly reduce the development time for your data acquisition and control application.

NI-DAQ Driver Software

The NI-DAQ driver software is included at no charge with all National Instruments DAQ hardware. NI-DAQ is not packaged with SCXI or accessory products, except for the SCXI-1200. NI-DAQ has an extensive library of functions that you can call 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 (timed D/A conversion), digital I/O, counter/timer operations, SCXI, RTSI, calibration, messaging, and acquiring data to extended memory.

NI-DAQ has both high-level DAQ I/O functions for maximum ease of use and low-level DAQ I/O functions for maximum flexibility and performance. Examples for high-level functions are streaming data to disk or acquiring a certain number of data points. An example of a low-level function is writing directly to registers on the DAQ device. NI-DAQ does not sacrifice performance of National Instruments DAQ devices because it lets multiple devices operate at their peak performance, even simultaneously.

NI-DAQ also internally addresses many of the complex issues between the computer and the DAQ hardware such as programming interrupts and DMA controllers. NI-DAQ maintains a consistent software interface so that you can change platforms with minimal modifications to your code. Whether you are using conventional programming languages or NI-DAQ software, your application uses the NI-DAQ driver software, as illustrated in Figure 1-1.

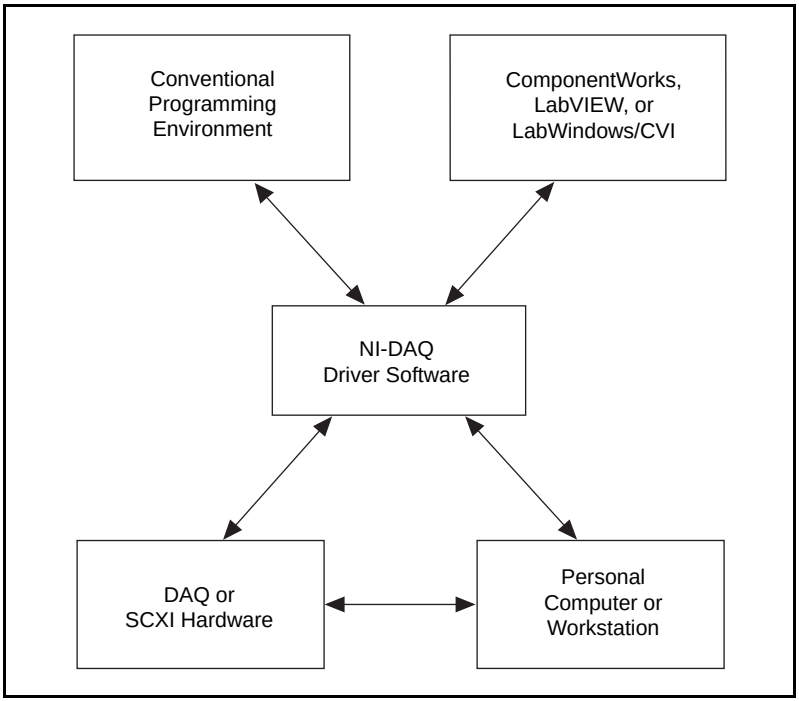


Figure 1-1. The Relationship Between the Programming Environment, NI-DAQ, and Your Hardware

You can use your DIO-32HS board, together with other AT (16-bit ISA), PCI, PC, EISA, DAQCard, and DAQPad Series DAQ hardware, with NI-DAQ software for PC compatibles version 5.0 or later.

Register-Level Programming

The final option for programming any National Instruments DAQ hardware is to write register-level software. Writing register-level programming software can be very time-consuming and inefficient and is not recommended for most users.

Even if you are an experienced register-level programmer, consider using NI-DAQ, LabVIEW, or LabWindows/CVI to program your National Instruments DAQ hardware. Using NI-DAQ, LabVIEW, or LabWindows/CVI software is as easy and as flexible as register-level programming and can save weeks of development time.

Optional Equipment

National Instruments offers a variety of products to use with your DIO-32HS board, including cables, connector blocks, and other accessories, as follows:

- Cables and cable assemblies, shielded and ribbon
- Connector blocks, shielded and unshielded 50 and 68-pin screw terminals
- Real Time System Integration (RTSI) bus cables
- Low channel count signal conditioning modules, boards, and accessories, including relays and optical isolation.

Some cables and accessories require use of the 68 to 50-pin DIO-32HS adaptor, detailed in Appendix B, *Optional Adapter Description*.

For more specific information about these products, refer to your National Instruments catalogue or web site, or call the office nearest you.

Unpacking

Your DIO-32HS board is shipped in an antistatic package to prevent electrostatic damage to the board. Electrostatic discharge can damage several components on the board. To avoid such damage in handling the board, take the following precautions:

- Ground yourself via a grounding strap or by holding a grounded object
- Touch the antistatic package to a metal part of your computer chassis before removing the board from the package.
- Remove the board from the package and inspect the board for loose components or any sign of damage. Notify National Instruments if the board appears damaged in any way. *Do not* install a damaged board into your computer.
- *Never* touch the exposed pins of connectors
- Store your DIO-32HS board in the antistatic envelope when not in use.

Installation and Configuration

Chapter 2

This chapter explains how to install and configure your DIO-32HS board.

Software Installation

If you are using NI-DAQ, LabVIEW, LabWindows/CVI, or ComponentWorks, refer to the release notes for your software. Find the installation and system configuration section for your operating system and follow the instructions given there.

Hardware Installation

You can install a PCI-DIO-32HS in any unused 5 V PCI expansion slot in your computer. You can install an AT-DIO-32HS in any unused AT or EISA expansion slot.

The following are general installation instructions, but consult your computer user manual or technical reference manual for specific instructions and warnings.

1. Write down the board serial number in the *AT/PCI-DIO-32HS Hardware and Software Configuration Form* in Appendix C of this manual. If you have more than one AT-DIO-32HS board, your software uses the serial numbers to distinguish them.
2. Turn off and unplug your computer.
3. Remove the top cover or access port to the expansion slots.
4. Remove the expansion slot cover on the back panel of the computer.
5. Insert the board into the system:
   - PCI-DIO-32HS — Insert the PCI-DIO-32HS board into a 5 V PCI slot. It may be a tight fit, but *do not force* the board into place.
   - AT-DIO-32HS — Insert the AT-DIO-32HS board into an AT (16-bit ISA) or EISA slot. It may be a tight fit, but *do not force* the board into place.

6. Screw the mounting bracket of the DIO-32HS board to the back panel rail of the computer.
7. Check the installation.
8. Replace the top cover of the computer.
9. Plug in and turn on your computer.

The DIO-32HS board is installed. Your board is now ready for software configuration.

PCI Board Configuration

The PCI-DIO-32HS is completely software configurable and is fully compliant with the *PCI Local Bus Specification,* Revision 2.0. The system software automatically allocates all board resources, including base memory address and interrupt level. The PCI-DIO-32HS has its own bus-master DMA logic and does not require DMA resources from your computer.

AT Board Configuration

The plug and play feature of the AT-DIO-32HS makes it completely software configurable. You can use software to configure the base I/O address, DMA channels, and interrupt channels.

Bus Interface

The AT-DIO-32HS works in either a Plug and Play mode or a switchless mode. These modes dictate how system resources are determined and assigned to the board.

Plug and Play Mode

The AT-DIO-32HS is fully compatible with the industry-standard Intel/Microsoft Plug and Play Specification version 1.0a. A Plug and Play system arbitrates and assigns system resources through software, freeing you from manually setting switches and jumpers. These resources include the board base I/O address, DMA channels, and interrupt channels. The AT-DIO-32HS is configured at the factory to request these resources from the Plug and Play Configuration Manager.

The Configuration Manager receives all of the resource requests at startup, compares the available resources to those requested, and assigns the available resources as efficiently as possible to the Plug and Play boards. Application software can query the Configuration Manager to determine the resources assigned to each board without your involvement. The Plug and Play software is installed as a device driver or as an integral component of the computer BIOS.

Switchless Data Acquisition

You can use your AT-DIO-32HS board in a non-Plug and Play system as a switchless DAQ board. A non-Plug and Play system is a system in which the Configuration Manager has not been installed and which does not contain any non-National Instruments Plug and Play products. Use a configuration utility, such as the NI-PnP or Intel configuration utilities, to enter the base address, DMA, and interrupt selections, and the application software assigns them to the board.

☞ **Note:** ***Avoid resource conflicts with non-National Instruments boards. For example, do not configure two boards to have the same base address.***

Base I/O Address Selection

The AT-DIO-32HS board can be configured to use a base address in the range of 100 to 3E0 hex. The AT-DIO-32HS occupies 16 bytes of address space and must be located on a 16-byte boundary. Therefore, valid addresses include 100, 110, 120, ..., 3D0, 3E0 hex. This selection is software configured and does not require you to manually change any settings on the board.

DMA Channel Selection

The AT-DIO-32HS can achieve high transfer rates by using up to two 16-bit DMA channels. The AT-DIO-32HS can use only 16-bit DMA channels, which correspond to channels 5, 6, and 7 in an AT (16-bit ISA) computer and channels 0, 1, 2, 3, 5, 6, and 7 in an EISA computer. These selections are all software configured and do not require you to manually change any settings on the board.

Interrupt Channel Selection

The AT-DIO-32HS increases bus efficiency by using an interrupt channel for event notification. The AT-DIO-32HS can use interrupt channel 3, 4, 5, 6, 7, 9, 10, 11, 12, 14, or 15. This selection is software-

configured and does not require you to manually change any settings on the board.

Tables 2-1, 2-2, and 2-3 provide information concerning possible conflicts in base address, DMA channel, and interrupt channel assignment when configuring your AT-DIO-32HS board.

Table 2-1. PC AT I/O Address Map

I/O Address Range (Hex)	Device
100 to 1EF	—
1F0 to 1F8	IBM PC AT Fixed Disk
200 to 20F	PC and PC AT Game Controller, reserved
210 to 213	PC-DIO-24 – default
218 to 21F	—
220 to 23F	Previous generation of AT-MIO boards – default
240 to 25F	AT-DIO-32F – default
260 to 27F	Lab-PC/PC+ – default
278 to 28F	AT Parallel Printer Port 2 (LPT2)
279	Reserved for Plug and Play operation
280 to 29F	WD EtherCard+ – default
2A0 to 2BF	—
2E2 to 2F7	—
2F8 to 2FF	PC, AT Serial Port 2 (COM2)
300 to 30F	3Com EtherLink – default
310 to 31F	—
320 to 32F	IBM PC/XT Fixed Disk Controller
330 to 35F	—
360 to 363	PC Network (low address)
364 to 367	Reserved

Table 2-1. PC AT I/O Address Map (Continued)

I/O Address Range (Hex)	Device
368 to 36B	PC Network (high address)
36C to 36F	Reserved
370 to 366	PC, AT Parallel Printer Port 1 (LPT1)
380 to 38C	SDLC Communications
380 to 389	Bisynchronous (BSC) Communications (alternate)
390 to 393	Cluster Adapter 0
394 to 39F	—
3A0 to 3A9	BSC Communications (primary)
3AA to 3AF	—
3B0 to 3BF	Monochrome Display/Parallel Printer Adapter 0
3C0 to 3CF	Enhanced Graphics Adapter, VGA
3D0 to 3DF	Color/Graphics Monitor Adapter, VGA
3E0 to 3EF	—
3F0 to 3F7	Diskette Controller
3F8 to 3FF	Serial Port 1 (COM1)
A79	Reserved for Plug and Play operation

Table 2-2 shows the PC AT interrupt assignments.

Table 2-2. PC AT Interrupt Assignment Map

IRQ	Device
15	Available
14	Fixed Disk Controller
13	Coprocessor
12	AT-DIO-32F – default
11	AT-DIO-32F – default
10	AT-MIO-16 – default
9	PC Network – default PC Network Alternate – default
8	Real Time Clock
7	Parallel Port 1 (LPT1)
6	Diskette Drive Controller Fixed Disk and Diskette Drive Controller
5	Parallel Port 2 (LPT2) PC-DIO-24 – default Lab-PC/PC+ – default
4	Serial Port 1 (COM1) BSC, BSC Alternate
3	Serial Port 2 (COM2) BSC, BSC Alternate Cluster (primary) PC Network, PC Network Alternate WD EtherCard+ – default 3Com EtherLink – default
2	IRQ 8-15 Chain (from interrupt controller 2)
1	Keyboard Controller Output Buffer Full
0	Timer Channel 0 Output

Table 2-3 shows the PC AT 16-bit DMA channel assignments.

Table 2-3. PC AT 16-Bit DMA Channel Assignment Map

Channel	Device
7	AT-MIO-16 series – default
6	AT-MIO-16 series – default AT-DIO-32F – default
5	AT-DIO-32F – default
4	Cascade for DMA Controller #1 (channels<0..3>)

☞ **Note:** ***EISA computers also have channels<0..3> available as 16-bit DMA channels.***

Chapter 3

Hardware Overview

This chapter provides an overview of the hardware functions of your AT-DIO-32HS or PCI-DIO-32HS board.

Each DIO-32HS board contains the National Instruments DAQ-DIO chip, a 32-bit general-purpose digital I/O interface. The DAQ-DIO chip enables the DIO-32HS to perform single-line and single-point input and output, digital data acquisition, digital waveform generation, and high-speed data transfer using a wide range of handshaking protocols.

Figures 3-1 and 3-2 show the block diagrams for the AT-DIO-32HS and PCI-DIO-32HS.

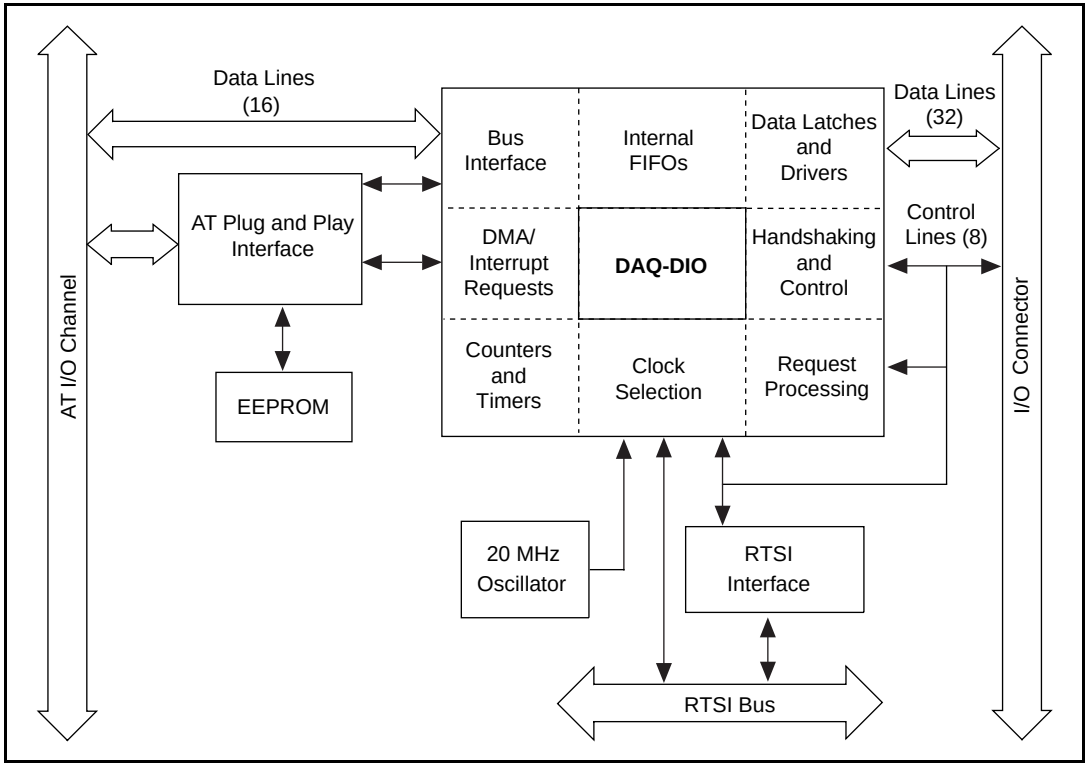


Figure 3-1. AT-DIO-32HS Block Diagram

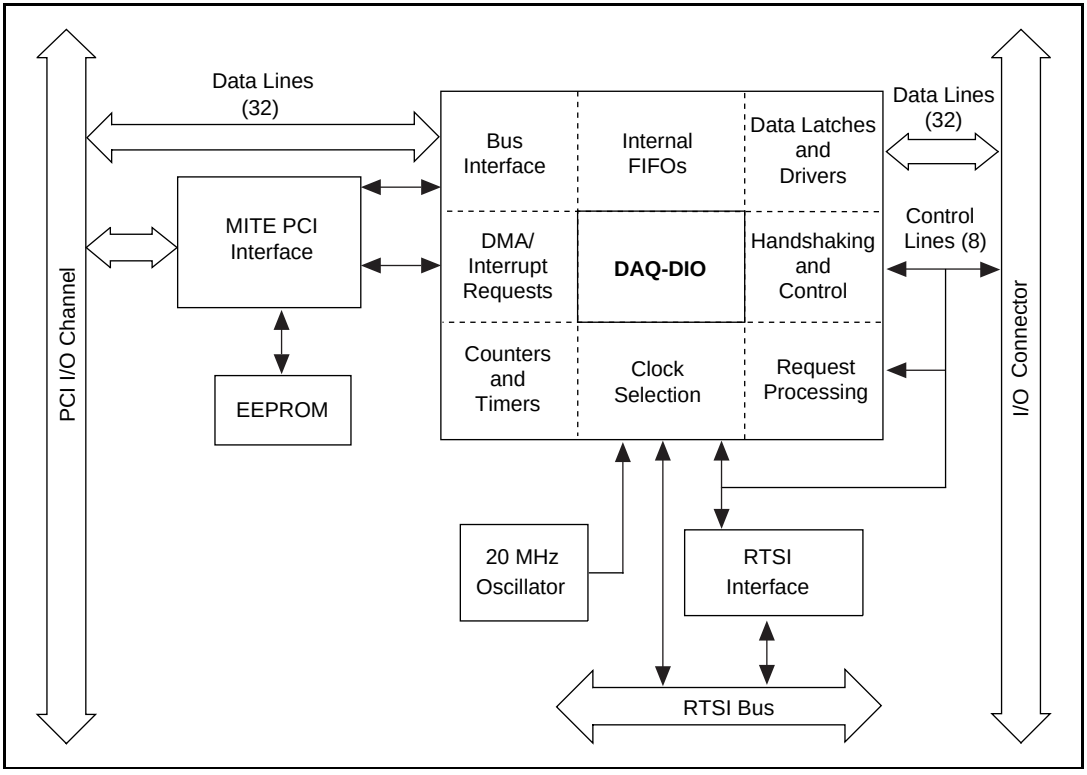


Figure 3-2. PCI-DIO-32HS Block Diagram

Unstrobed I/O

The DIO-32HS can perform unstrobed I/O, which is basic digital I/O that employs no handshaking or hardware-controlled timing. You can write or read data directly to or from the four digital I/O ports of the DIO-32HS. The I/O ports contain eight lines each and are labeled *A*, *B*, *C*, and *D*. You can configure each line individually for either input or output.

When you perform only unstrobed I/O, the DIO-32HS does not require its handshaking control and status signals to carry timing information. Therefore, you can use the REQ and STOPTRIG lines as extra data inputs, and the ACK and PCLK lines as extra data outputs.

Strobed I/O—Pattern Generation and Handshaking

The DIO-32HS can also perform strobed I/O. Strobed I/O is data transfer in which the DIO-32HS hardware regulates timing or performs handshaking functions. The DIO-32HS has two handshaking controllers and can perform up to two strobed operations simultaneously. The operations can be input transfers, output transfers, or one of each.

You select the width of each transfer by allocating the digital I/O ports into two groups for the two controllers. For example, by allocating ports A and B to group 1, you can perform a 16-bit strobed transfer using the group 1 controller. Any port that you do not allocate to a group, you can use for unstrobed I/O.

LabVIEW users should note that the LabVIEW documentation uses the term *group* in another context. LabVIEW groups do not correspond directly to hardware groups.

Each hardware group has its own, independent set of timing control lines. The set consists of ACK (STARTTRIG), REQ, PCLK, and STOPTRIG lines to carry control, status, clocking, and trigger information.

Any external device that the DIO-32HS controls, monitors, tests, or communicates with is referred to as a *peripheral device*.

Strobed operations fall into two categories—*pattern generation* and *full*, or *two-way*, *handshaking* transfer.

In pattern generation, data acquisition applications typically require sampling input data at a predetermined frequency. Similarly, waveform-generation applications require driving output data to specific output patterns at a predetermined frequency. You can regulate the frequency by supplying a timing signal to the REQ line; this signal is an external request. The DIO-32HS can also generate its own REQ pulses, or internal requests. Each group has a 32-bit counter to regulate the period between transfers.

In pattern generation, you can also supply start and stop triggers to begin and end an operation. You can trigger on either the rising edge or the falling edge of a trigger signal. You can also trigger when the DIO-32HS detects a specified digital pattern on its data lines.

In *full*, or *two-way*, *handshaking transfer*, control information passes both to and from the peripheral device. The DIO-32HS and the peripheral device each provides the other with strobe signals as data becomes available or is acquired. By withholding strobe signals, either the DIO-32HS or the peripheral device can slow down the transfer, if necessary. Because of this capability, and because fixed rates are not critical, you can run full-handshaking operations at the highest possible speeds.

Handshaking Protocols

When you perform full, two-way handshaking operations, you can select among several timing protocols offered by the DIO-32HS. The protocol you select determines the timing of the ACK signals that the DIO-32HS sends to the peripheral device and of the REQ signals expected from the peripheral device. One protocol, burst mode, also uses PCLK signals.

The following sections describe the handshaking protocols offered by the DIO-32HS. For timing details, see Chapter 5, *Signal Timing*.

8255 Emulation

This protocol emulates the strobed protocols obeyed by the 8255 and 82C55 PPI chips—chips that are used, for example, on the National Instruments PC-DIO-24 and PC-DIO-96. Because of faster response times, a wider data path, and the use of FIFO buffering, 8255 emulation mode on the DIO-32HS offers much higher data transfer rates than with an actual 8255 chip. As shown in Table 3-1, 8255 emulation offers the highest peak transfer rate of any protocol, other than burst mode.

Level ACK

After each transfer, the DIO-32HS asserts the ACK signal to the peripheral device. Holding the ACK line at the asserted level, the DIO-32HS board does not begin a new transfer until a false-to-true transition on the REQ line from the peripheral device occurs.

Leading-Edge Pulse

After each transfer, the DIO-32HS sends a pulse on the ACK line to the peripheral device. The DIO-32HS then waits for a false-to-true transition on the REQ line, the start of a REQ pulse, before starting a new transfer. You can specify an ACK pulse delay.

Long Pulse

Long-pulse mode is the same as leading-edge pulse mode, except that you can specify a minimum pulse width, instead of an ACK pulse delay.

Trailing-Edge Pulse

After each transfer, the DIO-32HS sends a pulse on the ACK line to the peripheral device. The DIO-32HS waits for a true-to-false transition on the REQ line, the end of a REQ pulse, before starting a new transfer.

Burst Mode

The DIO-32HS sends or receives a clock signal to or from the peripheral device over the PCLK line. Every cycle, the DIO-32HS asserts an ACK signal if ready for a transfer, and the peripheral device, likewise, asserts a REQ signal if ready for a transfer. Each cycle that both the DIO-32HS and the peripheral device indicate that they are ready for a transfer, one data point is latched. Burst mode can transfer data at high rates, particularly over short cables.

Comparing Protocols

Table 3-1 shows similarities and differences among the DIO-32HS handshaking modes. Asynchronous protocols use only the ACK and REQ signals. Burst mode, a synchronous protocol, uses the ACK, REQ, and PCLK signals. The PCLK line shares a clock signal between the DIO-32HS and the peripheral device.

Table 3-1 shows peak handshaking rates for typical cable lengths. The peak rates give an upper limit, deriving from the pulse widths and other timing specifications of the handshaking protocol. Your actual maximum rate depends on many factors; see the *Transfer Rates* section in this chapter.

Table 3-1 also shows whether the ACK and REQ signals are active high, active low, or programmable polarity. The table shows whether the leading or trailing edge of a REQ pulse initiates a data transfer. The table also describes the effect on each protocol of setting a programmable delay. See Chapter 5, *Signal Timing*, for timing details.

The table also shows complementary protocols with which the protocol can communicate, assuming that you choose complementary settings for any options the two protocols offer. For example, a DIO-32HS in

8255 emulation mode can communicate with a DIO-32HS in long pulse mode, if you select ACK and REQ to be active low.

Table 3-1. DIO-32HS Handshaking Protocols

Protocol	Peak Rates (MS/s) at Various Cable Lengths		REQ and ACK Signal	Programmable Delay Location	Complementary Protocols
	1 m	2 or 5 m			
Asynchronous Protocols					
8255 Emulation	5	2.67	Active-low; trailing	Between transfers	Leading-Edge Pulse
Level ACK	3.33	2.5	Programmable; leading	Before ACK and between transfers	Level-ACK
Leading-Edge Pulse	3.33	2.5	Programmable; leading	Before ACK and between transfers	Leading-Edge Pulse
Long Pulse	3.33	2.5	Programmable; leading	For pulse width and between transfers	Long Pulse, 8255-Emulation, PC-DIO-24, PC-DIO-96, 8255, 82C55
Trailing-Edge Pulse	1.8	1.5	Programmable; trailing	For pulse width and between transfers	Trailing-Edge Pulse
Synchronous Protocol					
Burst	20	10*	Active-high, (neither level REQ)	For clock speed	Burst

* Although asynchronous modes can adjust automatically to cable length, for synchronous modes, you must select an appropriate speed for your cable at configuration time. Select a delay of at least the following: 0 for a typical cable up to 1 m, 1 (100 ns) for a typical cable up to 5 m, and 2 (200 ns) for a typical cable up to 15 m long.

Starting a Handshaking Transfer

Starting a handshaking transfer correctly protects against incorrect or missed data when the ACK and REQ lines are changing polarity to active-high or active-low. This is particularly important in burst mode because of the potential to miss a lot of data. You can use either of two startup methods:

- Control the configuration and startup sequence.
- Select compatible line polarities and default line levels.

Controlling the Startup Sequence

One startup method is to follow a prescribed initialization order in which you can make sure the DIO-32HS is configured and is driving a valid ACK value before you enable the transfer on the peripheral device. Similarly, you can make sure the peripheral device is configured and is driving a valid REQ value before you enable the transfer on the DIO-32HS.

To use a prescribed initialization order, perform the following steps:

1. Configure the DIO-32HS for a protocol compatible with your peripheral device.
2. Configure and reset the peripheral device, if appropriate.
3. Enable the input device (DIO-32HS or peripheral device), and begin a transfer.
4. Enable the output device (DIO-32HS or peripheral device), and begin a transfer.

To control the startup order, you must be able to enable and disable the peripheral device, and you must control the order in which the DIO-32HS and the peripheral device are enabled. The DIO-32HS extra input and output lines can be helpful for these purposes.

Controlling Line Polarities

If you cannot control the initialization order of the DIO-32HS and peripheral device, you can still start a transfer reliably if you select the polarities of the ACK and REQ lines so that the power-up, undriven states of the control lines are the inactive states.

By default, the power-up, undriven state of the REQ and ACK lines is low, due to the onboard 2.2 kΩ pull-down resistors. Therefore, you should either select a protocol with active-high REQ and ACK signals

or use pull-up resistors to change the power-up, undriven control-line state to high. See Chapter 4, *Signal Connections*, for information on how to control the DIO-32HS pull-up and pull-down resistors.

Transfer Rates

The maximum average transfer rate that the DIO-32HS can achieve for two-way handshaking applications is the lower of the following two rates:

- the peak handshaking rate from Table 3-1, which can be lowered by the handshaking speed of your peripheral device
- the *average* available bus bandwidth, based on your computer system, the number of other devices generating bus cycles, and your application software

The maximum sustainable transfer rate the DIO-32HS can achieve for pattern generation application is the *minimum* available bus bandwidth, based on your computer system, the number of other devices generating bus cycles, and your application software (this rate is always lower than the peak pattern generation rate).

To achieve the highest possible rates, consider the following information:

- Full, two-way handshaking is faster than pattern generation, because two-way handshaking uses the average rather than the minimum bus bandwidth.
- Burst mode is the fastest handshaking protocol, especially for short cables.
- Your system bus should be as free as possible from unrelated activity. Minimize the number of other I/O cards active in the system.
- The average bus bandwidth is much higher for the PCI-DIO-32HS than for the AT-DIO-32HS.
- For the AT-DIO-32HS, minimize channel reprogramming time by allocating two DMA channels to a single transfer. By allocating two channels, you allow the AT-DIO-32HS software to reprogram one channel while continuing transfers on the other channel. This is particularly important for pattern generation.

Chapter 4

Signal Connections

This chapter describes how to make input and output signal connections to your DIO-32HS via the board I/O connector and RTSI connector.

The I/O connector for the DIO-32HS has 68 pins. You can connect the DIO-32HS to 68-pin accessories through an SH68-68-D1 shielded cable or R6868 ribbon cable. Using an optional 68-to-50 pin DIO-32HS adapter, you can also connect the DIO-32HS to 50-pin accessories through an NB1 ribbon cable.

I/O Connector

Figure 4-1 shows the pin assignments for the 68-pin DIO-32HS I/O connector. Refer to Appendix B, *Optional Adapter Description*, for the pin assignments for the 68-to-50 pin adapter.

Warning: ***Connections that exceed any of the maximum input or output ratings on the DIO-32HS may damage your board and your computer. See*** *Appendix A, Specifications****, for maximum ratings. This warning includes connecting any power signals to ground and vice versa. National Instruments is NOT liable for any damages resulting from any such signal connections.***

DIOD7	34	68	GND
GND	33	67	DIOD6
DIOD4	32	66	DIOD5
DIOD3	31	65	GND
GND	30	64	DIOD2
DIOD0	29	63	DIOD1
DIOC7	28	62	GND
GND	27	61	DIOC6
DIOC4	26	60	DIOC5
DIOC3	25	59	GND
GND	24	58	DIOC2
DIOC0	23	57	DIOC1
DIOB7	22	56	RGND
DIOB6	21	55	GND
GND	20	54	DIOB5
RGND	19	53	DIOB4
GND	18	52	DIOB3
DIOB1	17	51	DIOB2
DIOB0	16	50	GND
DIOA7	15	49	GND
GND	14	48	DIOA6
DIOA4	13	47	DIOA5
DIOA3	12	46	GND
GND	11	45	DIOA2
DIOA0	10	44	DIOA1
REQ2	9	43	RGND
ACK2	8	42	GND
STOPTRIG2	7	41	GND
PCLK2	6	40	CPULL
PCLK1	5	39	GND
STOPTRIG1	4	38	DPULL
ACK1	3	37	GND
REQ1	2	36	GND
+5 V	1	35	RGND

Figure 4-1. DIO-32HS I/O Connector Pin Assignments

Signal Descriptions

Table 4-1 provides signal descriptions. Each signal on the DIO-32HS is referenced to the GND lines.

Table 4-1. Signal Descriptions

Pins	Signal Name	Signal Type	Description
2, 9	REQ<1..2>	Control	Group 1 and group 2 request lines—In handshaking mode, a group's REQ line carries handshaking status information from the peripheral. In pattern generation mode, REQ carries timing pulses either to or from the peripheral to strobe data into or out of the DIO-32HS. These strobe signals are comparable to the CONVERT* or UPDATE* signals of an analog DAQ board. When not configuring the DIO-32HS for group operations, you can use the REQ<1..2> lines as extra, general-purpose input lines (IN<3..4>).
3, 8	ACK<1..2>	Control	Group 1 and group 2 acknowledge lines—In handshaking mode, a group's ACK line carries handshaking control information to the peripheral. In pattern generation mode, the ACK lines can function as STARTTRIG<1..2> lines. You can use rising or falling edges on these lines to start pattern generation operations. When not configuring the DIO-32HS for group operations, you can use the ACK<1..2> lines as extra, general-purpose output lines (OUT<3..4>).
4, 7	STOPTRIG <1..2>	Control	Group 1 and group 2 stop triggers—You can use rising or falling edges on these lines to end pattern generation operations. When not configuring the DIO-32HS for group operations, you can use the STOPTRIG<1..2> lines as extra, general-purpose input lines (IN<1..2>).

Table 4-1. Signal Descriptions (Continued)

Pins	Signal Name	Signal Type	Description
5–6	PCLK<1..2>	Control	Group 1 and group 2 peripheral clock lines—In handshaking mode, if you select the burst protocol, these lines carry clock signals to the peripheral (during output operations) or from the peripheral device (during input operations). When not configuring the DIO-32HS for group operations, you can use the PCLK<1..2> lines as extra, general-purpose output lines (OUT<1..2>).
10, 44–45, 47–48, 12–13, 15	DIOA<0..7>	Data	Port A bidirectional data lines—Port A is port number 0. DIOA7 is the MSB; DIOA0 is the LSB. When combined in a group with other ports, port A is the least significant port.
16–17, 51–54, 21–22	DIOB<0..7>	Data	Port B bidirectional data lines—Port B is port number 1. DIOB7 is the MSB; DIOB0 is the LSB.
23, 25–26, 57–58, 60–61, 28	DIOC<0..7>	Data	Port C bidirectional data lines—Port C is port number 2. DIOC7 is the MSB; DIOC0 is the LSB.
29, 63–64, 31–32, 66–67, 34	DIOD<0..7>	Data	Port D bidirectional data lines—Port D is port number 3. DIOD7 is the MSB; DIOD0 is the LSB. When combined in a group with other ports, port D is the most significant port.
40	CPULL	Bias Selection	Control pull-up/pull-down selection—This input signal selects whether the DIO-32HS pulls the timing and handshaking control lines (REQ, ACK, PCLK, and STOPTRIG) up or down when undriven. If you connect CPULL to +5 V, the DIO-32HS pulls the control lines up. If you connect CPULL to GND or leave CPULL unconnected, the DIO-32HS pulls the control lines down.

Table 4-1. Signal Descriptions (Continued)

Pins	Signal Name	Signal Type	Description
38	DPULL	Bias Selection	Data pull-up/pull-down selection—This input signal selects whether the DIO-32HS pulls the data lines (DIOA, DIOB, DIOC, and DIOD) up or down when undriven. If you connect DPULL to +5 V, the DIO-32HS pulls the data lines up. If you connect DPULL to GND or leave DPULL unconnected, the DIO-32HS pulls the data lines down.
1	+5 V	Power	5 Volts, output—This line provides a maximum of 1 A of power, regulated by an onboard fuse that can automatically reset itself after current returns to normal.
11, 14, 18, 20, 24, 27, 30, 33, 36–37, 41–42, 46, 49–50, 55, 59, 62, 65, 68	GND	Power	Ground—These lines are the ground reference for all other signals.
19, 35, 43, 56	RGND	Power	Reserved ground—These lines offer additional ground pins that vary in connection from cable to cable. With an R6868 ribbon cable, you can use these lines as additional ground references. With an SH68-68-D1, however, these signals are not connected.

Signal Characteristics

Following is a list of signal characteristics. Characteristics are for all signals, unless otherwise noted. For signal characteristics not given in this section, see Appendix A, *Specifications*.

- Drive current—After being enabled, all lines that can be configured for output sink at least 24 mA at 0.4 V, and source at least 24 mA at 2.4 V.
- Ground reference—All signals are referenced to the GND lines.
- Initial state—At power up, all control and data lines begin at high impedance. With no load attached, the voltage levels of the lines are controlled by the pull-up or pull-down resistors.
- Pull-up/pull-downs
  - Control lines—All timing control lines have 2.2 kΩ pull-up or pull-down resistors, controlled by the CPULL line.
  - Data lines—All timing data lines have 100 kΩ pull-up or pull-down resistors, controlled by the DPULL line.
  - Bias-selection lines—The CPULL and DPULL lines, which select the bias of the control and data lines, are themselves biased low with 20 kΩ pull-down resistors. The default bias of all lines, therefore, is pulled down.
- Polarity
  - Data signals—Active high. A *1* corresponds to a high voltage, and a *0* corresponds to a low voltage.
  - Control signals—Depending on the operating mode and handshaking protocol you select, control signals can be active high or active low.

Control Signal Summary

The direction and function of each group's signal timing and handshaking lines vary, depending on the mode of operation you select for the group. Table 4-2 shows the direction and function of each control signal in each mode.

Table 4-2. Control Signal Summary

Signal Name	Direction in Handshaking Mode	Function in Handshaking Mode	Direction in Pattern Generation	Function in Pattern Generation	Function in Unstrobed Mode
REQ<1..2>	input	request	input or output	request	extra inputs (IN<3..4>)
ACK<1..2>	output	acknowledge	input	start trigger (START–TRIG<1..2>)	extra outputs (OUT<3..4>)
STOPTRIG<1..2>	—	—	input	stop trigger	extra inputs (IN<1..2>)
PCLK<1..2>	input or output	peripheral clock	—	—	extra outputs (OUT<1..2>)

RTSI Bus Interface

The DIO-32HS contains an interface to the National Instruments RTSI bus. The RTSI bus provides seven trigger lines and a system clock line. All National Instruments AT and PCI Series boards that have RTSI bus connectors can be cabled together inside a computer to share these signals.

Board and RTSI Clocks

The DIO-32HS requires a frequency timebase to run the handshaking logic and to generate intervals for pattern generation. The frequency timebase must be 20 MHz.

Either the DIO-32HS can use its internal 20 MHz clock source as the timebase, or you can provide a timebase from another 20 MHz board over the RTSI bus. When using its internal 20 MHz timebase, the DIO-32HS can also drive its internal timebase onto the RTSI bus to another board that uses a 20 MHz clock.

The 20 MHz timebase, whether local or imported from the RTSI bus, serves as the primary frequency source for the DIO-32HS. You can select a clocking configuration through software. By default, the DIO-32HS uses its own internal timebase, without driving the RTSI bus clock line.

RTSI Triggers

The seven RTSI trigger lines on the RTSI bus provide a very flexible interconnection scheme for any AT or PCI board sharing the RTSI bus. Any DIO-32HS control signal can connect to a RTSI bus line. You can drive output control signals onto the bus and receive input control signals from the bus. Figure 4-2 shows the signal connection scheme.

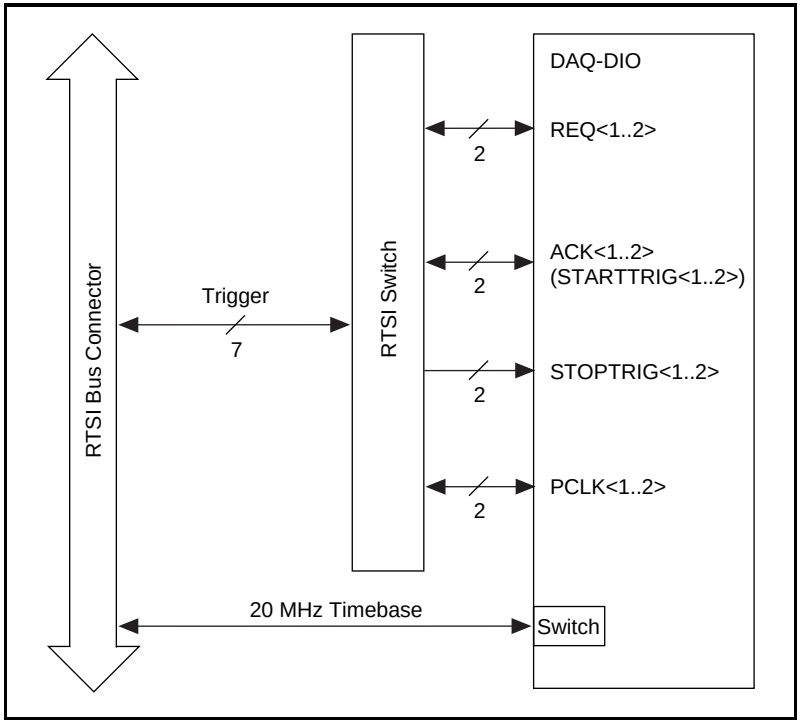


Figure 4-2. RTSI Bus Signal Connection

Data Signal Connections

The digital data signals are DIOA<0..7>, DIOB<0..7>, DIOC<0..7>, and DIOD<0..7>. These data signals are referenced to the GND pins. Ports DIOA, DIOB, DIOC, and DIOD are port numbers 0, 1, 2, and 3, respectively.

Unstrobed I/O

For low-speed, unstrobed operation, you can configure each individual pin for input, standard output, or wired-OR output. Figure 4-3 shows DIOA<0..3> configured for input, DIOA<4..7> configured for standard output, and DIOB<0..3> configured for wired-OR output. Unstrobed input applications include sensing external device states, such as the state of the switch shown in the figure, and receiving low-speed TTL signals. Unstrobed output applications include driving external controls and indicators such as the LED shown in Figure 4-3, and sending low-speed TTL signals.

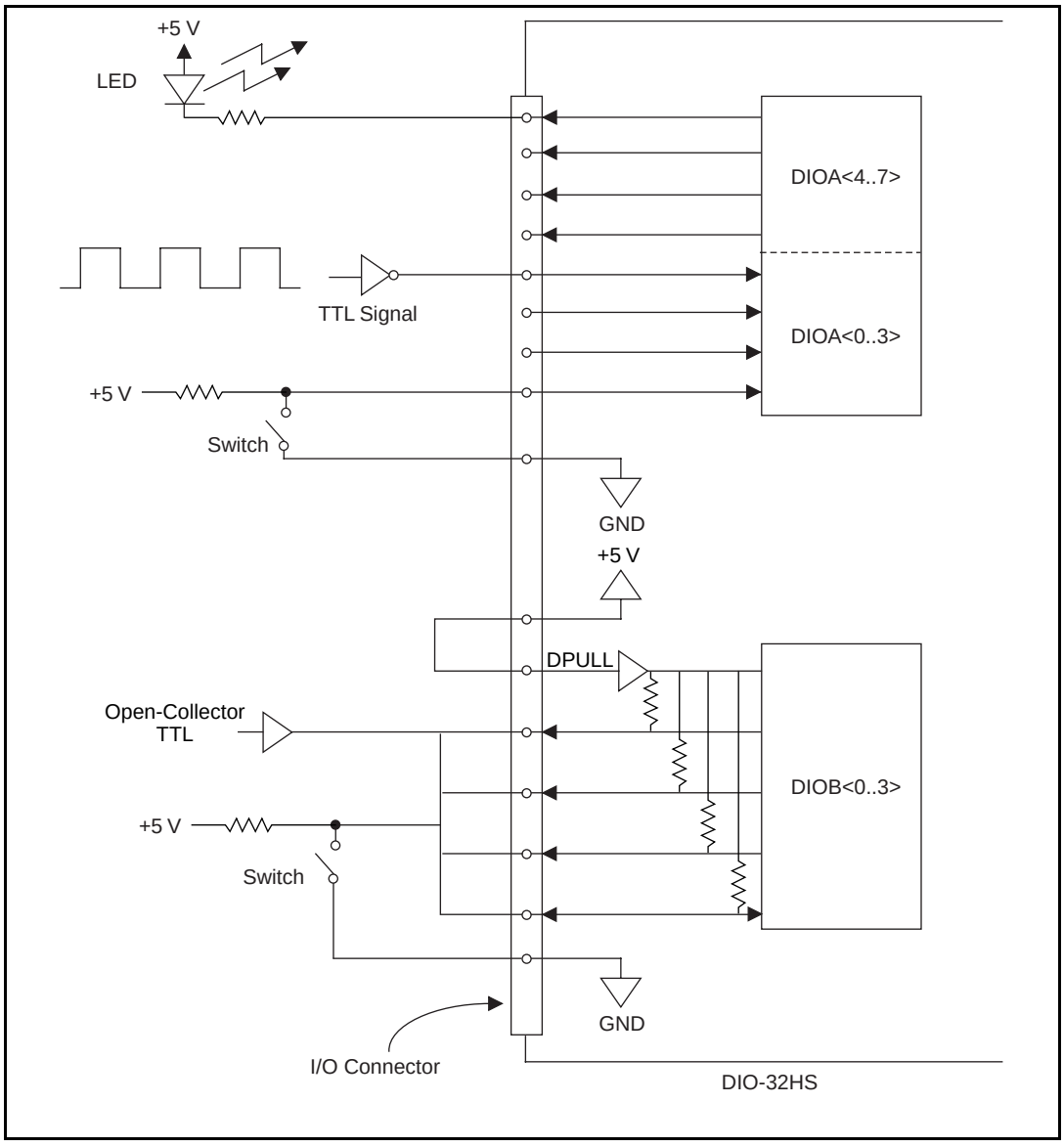


Figure 4-3. Example of Data Signal Connections

For unstrobed operations, you have a choice of two types of output drivers: standard and wired-OR. A standard driver drives its output pin to approximately 0 V for logic low, or +5 V for logic high. A standard driver has several advantages:

- It does not rely on pull-up resistors.
- It is independent of the state of the DPULL line.
- It has high current drive for both its logic high and logic low states.
- It can drive high-speed transitions in both the high-to-low and low-to-high directions.

A wired-OR output driver drives its output pin to 0 V for logic low, but floats (tri-states) the pin for logic high. Therefore, a wired-OR output driver requires a pull-up resistor to pull the pin to +5 V for logic high. To provide a pull-up resistor, you can simply connect the DPULL pin on the I/O connector to the +5 V pin, making the DIO-32HS 100 kΩ pull-down resistors into 100 kΩ pull-up resistors. A wired-OR driver has the following advantages over a standard driver:

- You can connect two or more wired-OR outputs together without damaging the drivers.
- You can connect wired-OR outputs to open-collector drivers to GND signals, or to switches connecting to GND signals, without damaging the drivers.
- You can use wired-OR outputs bidirectionally. For example, after connecting wired-OR outputs together, you can read back the value of one of the pins to determine whether any of the connected outputs is logic low.

Strobed I/O

Strobed operations, such as pattern generation and handshaking, use the same data signal connections as unstrobed operations, with the following exceptions:

- You can configure data signals only on a port-by-port basis, rather than on a pin-by-pin basis. To configure data ports, you must assign them to handshaking groups.
- Strobed output operations use only standard, rather than wired-OR, output drivers.

Strobed applications include digital data acquisition, digital waveform generation, and data transmission to or from an external device.

Timing Connections

Timing connections include the REQ, ACK (STARTTRIG), and STOPTRIG pins for pattern generation, and the REQ, ACK, and PCLK pins for two-way handshaking operation.

The DIO-32HS provides two handshaking groups, each with its own timing connections. To perform pattern generation or handshaking, you must first associate a set of data pins with a group. Do this by assigning data ports to handshaking groups.

Chapter 5, *Signal Timing*, details the connection and timing of each pattern generation and handshaking control signal.

Pull-up and Pull-down Connections

The CPULL and DPULL lines enable you to select the biasing of the control and data signals.

If you drive the CPULL pin low, connect the CPULL pin to a GND pin, or leave the CPULL line disconnected, the DIO-32HS pulls all its control lines down to 0 V with 2.2 kΩ resistors. If you drive the CPULL pin high or connect the CPULL pin to the +5 V pin, the DIO-32HS pulls all its control lines to +5 V with the same 2.2 kΩ resistors.

Similarly, if you drive the DPULL pin low, connect the DPULL pin to a GND pin, or leave the DPULL line disconnected, the DIO-32HS pulls all its data lines down to 0 V with 100 kΩ resistors. If you drive the DPULL pin high or connect the DPULL pin to the +5 V pin, the DIO-32HS pulls all its control lines to +5 V with the same 100 kΩ resistors.

Do not connect CPULL, DPULL, or any other line directly to an external power supply while the DIO-32HS is powered off.

The DIO-32HS drivers power up and reset to high-impedance states. Therefore, the CPULL and DPULL lines control whether you get high or low control and data lines, respectively, when you power up the DIO-32HS or reset its drivers.

You should connect DPULL to +5 V when using any wired-OR output drivers. In other cases, you can use the CPULL and DPULL lines to select a power-up state that is inactive in your application. For example,

if you are using active-low handshaking signals, you can connect the CPULL line to +5 V to place the handshaking lines in the high, inactive state at power up.

Power Connections

The +5 V pin on the I/O connector supplies power from the computer power supply through a self-resetting fuse. The fuse resets automatically within a few seconds after removal of an overcurrent condition. The power pin is referenced to the GND pins and can supply power to external, digital circuitry.

- Power rating: +4.65 to +5.25 VDC at 1 A

You can connect the +5 V pin to the CPULL and DPULL pins to control the bias of the DIO-32HS control and data pins, as described in the *Pull-up and Pull-down Connections* section in this chapter.

Warning: ***Do not connect the +5 V power pin directly to the GND, RGND, or any output pin of the DIO-32HS or any voltage source or output pin on another device. Doing so can damage the board and the computer. National Instruments is NOT liable for damages resulting from such a connection.***

Field Wiring and Termination

Transmission line effects and environmental noise, particularly on clock and control lines, can lead to incorrect data transfers if you do not take proper care when running signal wires to and from the board.

Note: ***Make sure your board and your peripheral device share a common ground reference. Connect one or more DIO-32HS GND lines to the ground reference of your peripheral device.***

Take the following precautions to ensure a uniform transmission line and minimize noise pickup:

- Use twisted-pair wires to connect digital I/O signals to the board. Twist each digital I/O signal with a GND line.
- Place a shield around the wires connecting digital I/O signals to the board.

- Route signals to the board carefully. Keep cabling away from noise sources. The most common noise source in a PC-based system is the video monitor. As much as possible, separate the monitor from any unshielded signal wiring.

For DIO-32HS output signals, it is important to terminate your cable properly to reduce or eliminate signal reflections in the cable. You can use many different methods for terminating transmission lines.

A good method for the DIO-32HS is to connect one fast Schottky diode from +5 V to each signal line, and another from the signal line to ground. The +5 V and ground connections should be low-impedance connections. For example, if you make your +5 V connection through a long wire, back to the +5 V pin of the DIO-32HS, add a capacitor to your termination circuit to stabilize the +5 V connection near the Schottky diodes.

One suitable Shottky diode is the 1N5711, available from several manufacturers. For more specialized use, you may be able to find diodes packaged in higher densities appropriate to your application. For example, the Central Semiconductor CMPSH-35 contains two diodes, suitable for terminating one line. The California Micro Devices PDN001 contains 32 diodes, suitable for terminating 16 lines.

You do not need to terminate the DIO-32HS input signals. The DIO-32HS contains onboard Schottky diode termination. Figure 4-4 shows the recommended transmission line terminations.

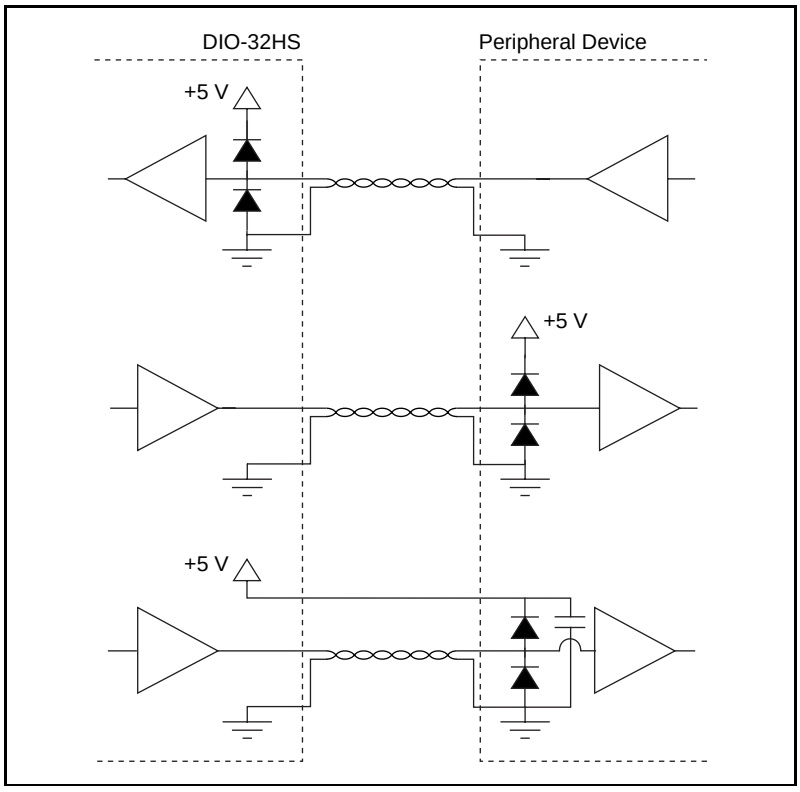


Figure 4-4. Transmission Line Terminations

The following additional recommendations apply for all signal connections to your DIO-32HS board:

- Separate DIO-32HS signal lines from high-current or high-voltage lines. These lines are capable of inducing currents in or voltages on the DIO-32HS signal lines if they run in parallel paths at a close distance. To reduce the magnetic coupling between lines, separate them by a reasonable distance if they run in parallel, or run the lines at right angles to each other.
- Do not run signal lines through conduits that also contain power lines.
- Protect signal lines from magnetic fields caused by electric motors, welding equipment, breakers, or transformers by running them through special metal conduits.

Chapter 5

Signal Timing

This chapter gives detailed timing specifications for pattern generation and for the various full, two-way handshaking modes.

Pattern Generation Timing

Pattern generation timing is similar for digital data acquisition (input) and digital waveform generation (output). Data transfers are timed by request pulses, carried on the REQ pin. The DIO-32HS can generate request pulses internally, or you can provide external pulses. Each request pulse strobes a data point into or out of the DIO-32HS.

You can use up to two additional timing signals, if you select triggered pattern generation: a start trigger and a stop trigger. A start trigger, if used, begins the pattern generation operation. A stop trigger ends the operation. However, you can specify a number of data points to transfer after the stop trigger.

You can substitute a digital pattern for either the start or stop trigger. In this case, the operation begins or ends when the DIO-32HS detects a particular digital pattern on the data lines belonging to the group.

Figure 5-1 shows a pattern generation operation using request pulses, a start trigger, and a stop trigger.

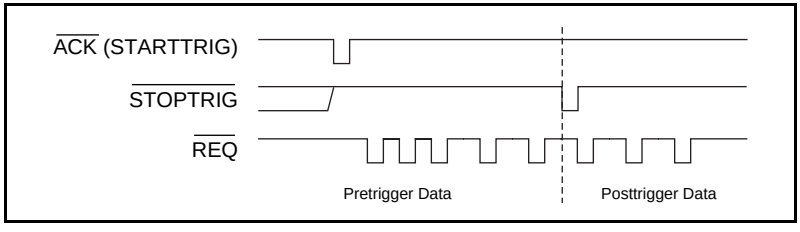
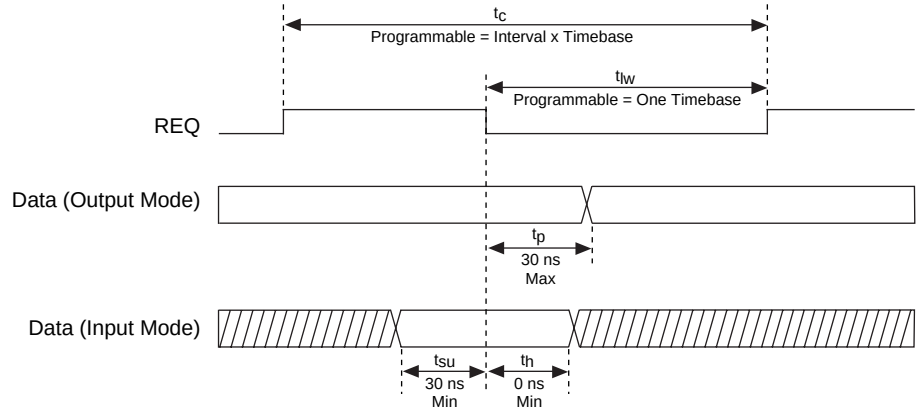


Figure 5-1. Pattern Generation Timing

Request Timing

Internal Requests

Figure 5-2 shows internal request timing. You can select a timebase and an interval. The request pulses low once per data transfer. The duration of the low pulse is equal to one timebase. The period of the request pulse is equal to the interval multiplied by the timebase.

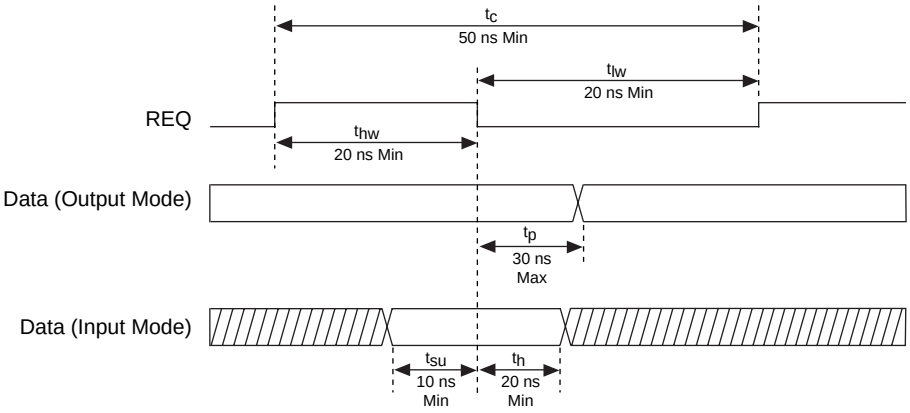


Parameter	Description
t_c	Cycle time
t_{lw}	Width of low pulse
t_p	Propagation time to valid output data
t_{su}	Setup time
t_h	Hold time

Figure 5-2. Internal Request Timing

External Requests

Figure 5-3 shows external request timing. The request signal must pulse low and return high. The request pulse low and high durations must be at least 20 ns each. The minimum period is 50 ns.



Parameter	Description
t_c	Cycle time
t_{lw}	Width of low pulse
t_{hw}	Width of high pulse
t_p	Propagation time to valid output data
t_{su}	Setup time
t_h	Hold time

Figure 5-3. External Request Timing

Trigger Timing

Using pattern generation mode, you can configure the DIO-32HS to accept both start and stop triggers.

The stop trigger is the primary trigger. The DIO-32HS can transfer specified numbers of data both before and after a stop trigger. If you do not enable a stop trigger, the DIO-32HS stops automatically after transferring a number of data points equal to the size of your buffer.

The start trigger is a second trigger that begins a pattern generation operation. If you do not enable a start trigger, the operation starts immediately when you issue a software command to perform a transfer.

Triggers are available for both waveform generation (output mode) and data acquisition (input mode). Acquiring data that occurs before or after a trigger is known as pretrigger or posttrigger data acquisition,

respectively. Using only a start trigger, you can do posttrigger data acquisition. A stop trigger enables you to do pretrigger data acquisition, or combined pretrigger and posttrigger data acquisition. After detecting the stop trigger, the DIO-32HS begins counting the post-stop-trigger portion of the data acquisition. Figure 5-4 shows trigger pulse timing, where t_w is pulse width.

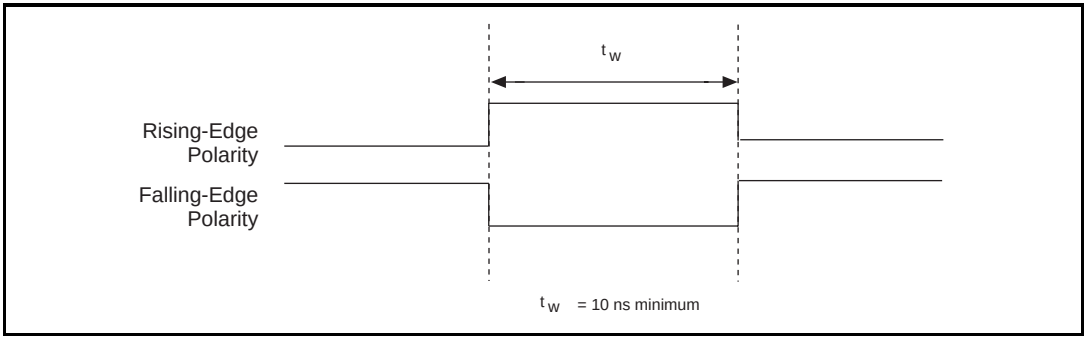


Figure 5-4. Trigger Input Signal Timing

Pattern Detection

Instead of a pulse on the I/O connector, you can also use digital pattern detection as a trigger to start or stop an input operation.

You can specify three parameters to the pattern detection circuit:

- a mask, declaring which data bits you wish to examine
- the pattern value you wish to search for
- whether to trigger when the data matches or mismatches the specified value

Figure 5-5 shows a pattern detection example.

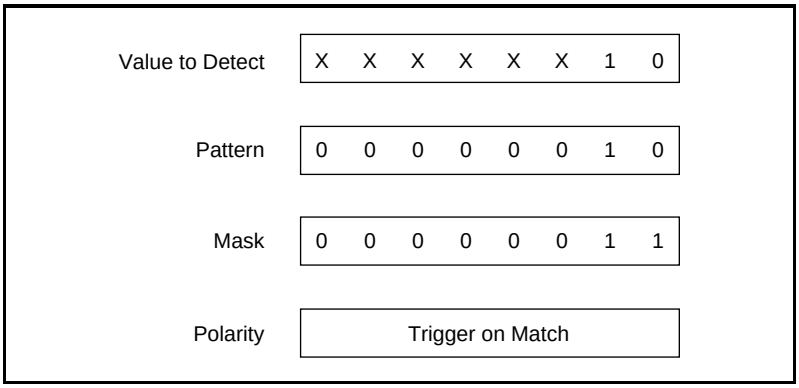


Figure 5-5. Pattern Detection Example

The DIO-32HS provides two types of pattern detection timing:

- Compare all data to the pattern immediately, without waiting for a request pulse (typically used for start triggers).
- Compare acquired data to the pattern, after a request pulse strobes the data in (typically used for stop triggers).

In immediate, unstrobed pattern detection, any occurrence of the pattern, with or without a request strobe, triggers the operation. However, the DIO-32HS filters out very short pattern matches, to ensure that a transient data value that occurs during line switching does not falsely trigger the operation. A glitch must be present for no more than 20 ns to guarantee rejection. A valid pattern must be present for at least 60 ns to guarantee triggering.

In strobed, request-based pattern detection, data is checked as the data is strobed in by request pulses. Strobed pattern detection is typically used to generate stop triggers. You can use strobed pattern detection to generate start triggers too, but only when using an external request source. See the *Request Timing* section earlier in this chapter for the timing of the request pulses that strobe in data.

Handshake Timing

This section describes the DIO-32HS two-way handshaking modes and the timing specifications of each mode.

In handshaking, the ACK signal always conveys information about when the DIO-32HS is ready for a transfer. The REQ signal conveys information about when the peripheral device is ready for a transfer.

☞ **Note:** ***Depending on the protocol and the direction of the transfer, either an ACK or a REQ signal can occur first in the handshaking sequence.***

8255 Emulation

The 8255 emulation mode handshakes in a manner compatible with an 8255 or 82C55 Programmable Peripheral Interface (PPI). The 8255 and 82C55 PPIs are digital I/O chips used on many digital DAQ boards, such as the National Instruments PC-DIO-24 and PC-DIO-96.

DIO-32HS emulation mode is a superset of the 8255 and 82C55 protocols. The PCI-DIO-32HS can handshake with peripheral devices that use 8255 or 82C55 handshaking specifications.

The DIO-32HS can perform back-to-back transfers much faster than a true 8255-based board. If your peripheral device requires more time between transfers, you can configure the DIO-32HS to add a data-settling delay between transfers.

You can use a DIO-32HS in emulation mode with 8, 16, or 32-bit data paths.

Input

☞ **Note:** ***DIO-32HS terminology differs from 8255 terminology. In input mode, the DIO-32HS REQ line carries the 8255 STB input signal, and the DIO-32HS ACK line carries the 8255 IBF output signal. Both lines are active low.***

In input mode, the DIO-32HS asserts the ACK signal low when ready to accept data. The peripheral device can then strobe data into the DIO-32HS by pulsing the REQ line low. The falling REQ signal edge causes the ACK signal to deassert, and the rising REQ signal edge causes the DIO-32HS to latch input data. Afterward, the DIO-32HS reasserts the ACK signal low when ready for another input. Figure 5-6 shows an input transfer in 8255 emulation mode.

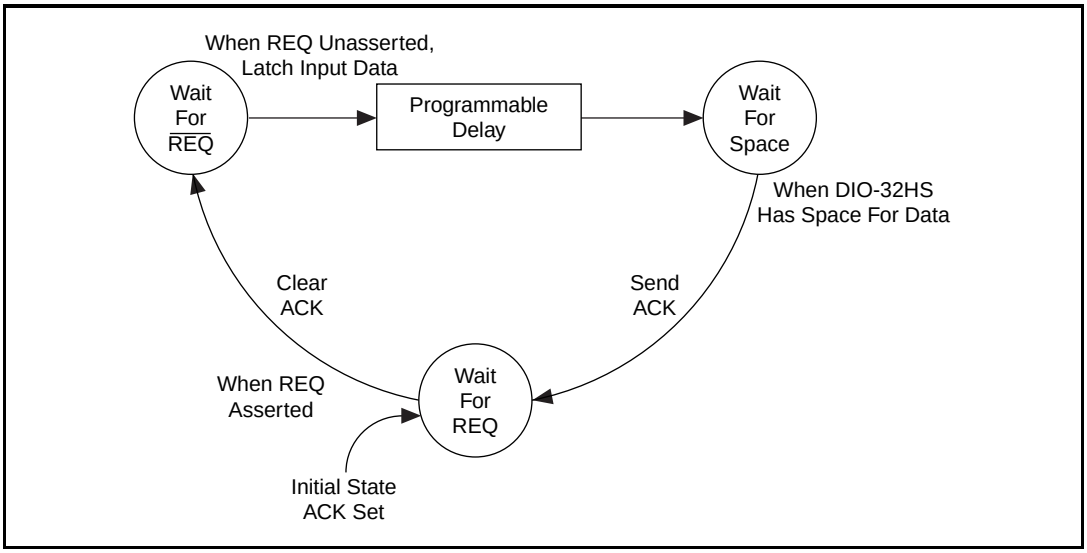


Figure 5-6. 8255 Emulation Mode Input

Output

☞ **Note:** ***DIO-32HS terminology differs from 8255 terminology. In output mode, the DIO-32HS REQ line carries the 8255 ACK input signal, and the DIO-32HS ACK line carries the 8255 OBF output signal. Both lines are active low.***

In output mode, the DIO-32HS asserts the ACK signal low when output data is available. The peripheral device can receive the data on the falling or rising edge of the ACK signal, or any time in between. The peripheral device must respond with an active-low REQ pulse to request additional data. The falling REQ signal edge causes the ACK signal to return to the inactive state, and the rising REQ signal edge enables a new transfer to occur. Therefore, the peripheral device should wait until it has received data before raising the REQ signal. The peripheral device can also wait for the ACK signal to deassert before raising the REQ signal. Figure 5-7 shows an output transfer in 8255 emulation mode.

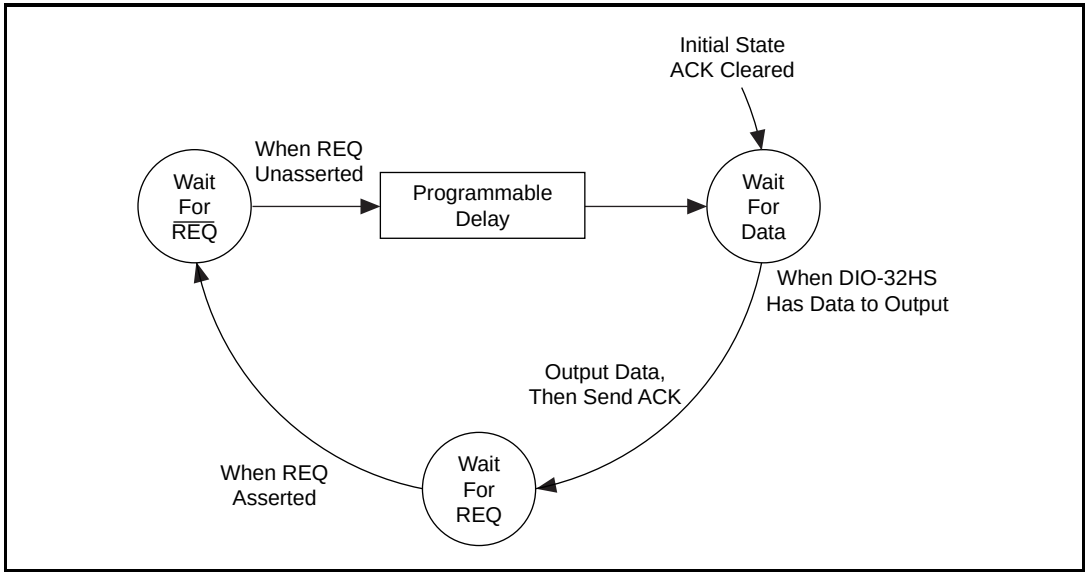
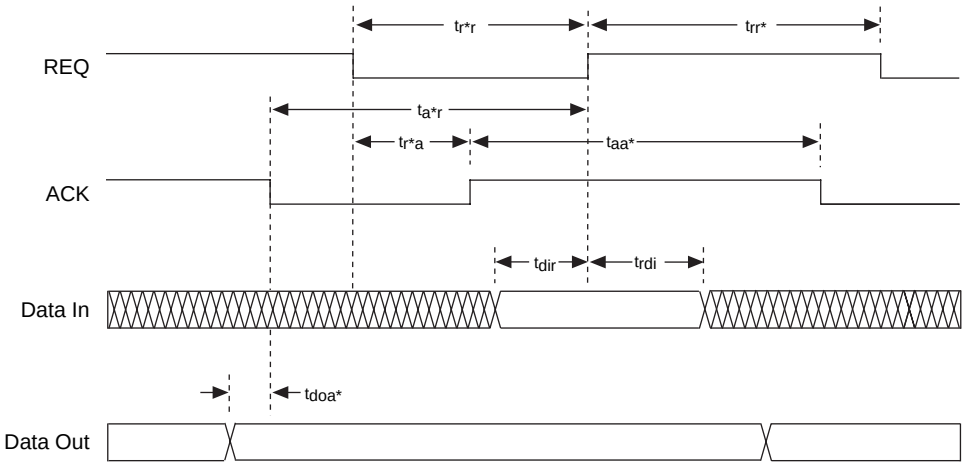


Figure 5-7. 8255 Emulation Mode Output

8255 Emulation Mode Timing Specifications

Figure 5-8 shows the timing diagram for 8255 emulation mode.



Parameter	Description	Minimum	Maximum
Input Parameters			
t_{r*r}	REQ low duration	75	—
t_{rr*}	REQ high duration	75	—
t_{a*r}	ACK falling edge to REQ rising edge	0	—
t_{dir}	Input data valid to REQ rising edge	0	—
t_{rdi}	REQ rising edge to input data invalid	10	—
Output Parameters			
t_{aa*}	ACK high duration	100	—
t_{r*a}	REQ falling edge to ACK rising edge	—	150
t_{doa*}	Output data valid to ACK falling edge	25	—
All timing values are in nanoseconds.			

Figure 5-8. 8255 Emulation Timing

Other Asynchronous Modes

Besides 8255-compatible mode, the DIO-32HS supports several other asynchronous handshaking protocols: level-ACK mode, leading-edge mode, long-pulse mode, and trailing-edge mode. These handshaking modes are compatible with the handshaking modes of the National Instruments AT-DIO-32F board.

Each of these modes offers some options:

- Polarity of the ACK and REQ signals. The diagrams show active-high signals.
- A programmable delay, from 0 to 700 ns, programmable in increments of 100 ns. You can use the programmable delay to reduce handshaking speed for slow peripheral devices. A delay increases the duration of each transfer. The location of the delay in the handshaking sequence differs from protocol to protocol. In addition, a delay increases the minimum spacing between consecutive transfers.
- Request-edge latching. With request-edge latching enabled, in input mode, the DIO-32HS latches data in from the I/O connector on the REQ edge before reading the data. In output mode, after writing the data, the DIO-32HS latches data out of the I/O connector on the REQ edge. Which edge of REQ is used (rising or falling) depends on the handshaking mode and the REQ polarity.

Level-ACK Mode

In level-ACK mode, the DIO-32HS asserts the ACK signal when ready for a transfer and holds the ACK signal level until an active-going edge occurs on the REQ line. After the REQ edge occurs, the DIO-32HS deasserts the ACK signal until ready for another transfer.

Input

In input mode, the DIO-32HS asserts the ACK signal when ready to accept data. The peripheral device can then strobe data into the DIO-32HS by asserting the REQ signal. The active-going REQ signal edge deasserts the ACK signal and causes the DIO-32HS to latch input data. Afterward, the DIO-32HS reasserts the ACK signal when ready for another input.

To slow down the handshake, you can specify a data-settling delay to occur before the ACK signal.

Output

In output mode, the DIO-32HS raises the ACK signal after driving output data to indicate new, valid output data. The peripheral device can latch the data on the falling or rising edge of the ACK signal, or at any time before returning a REQ pulse. The peripheral device must respond with an active-going REQ signal edge to deassert the ACK signal and request additional data.

To slow down the handshake, you can specify a data-settling delay to occur before the ACK signal. This delay increases the setup time from valid output data to the ACK signal.

Figure 5-9 shows an input transfer in level-ACK mode.

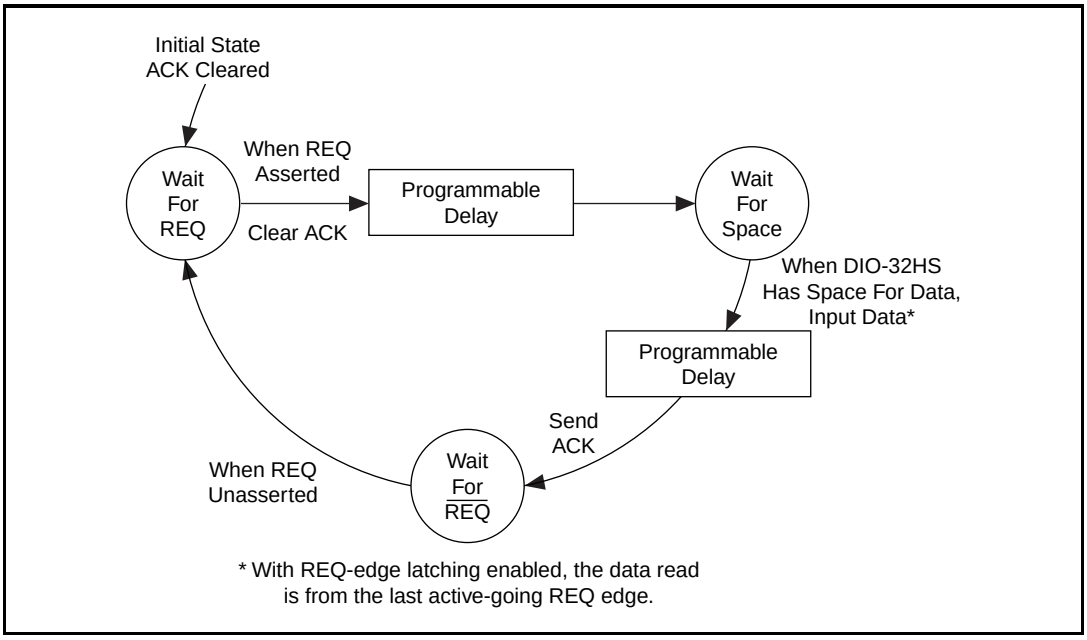


Figure 5-9. Level-ACK Mode Input

Figure 5-10 shows an output transfer in level-ACK mode.

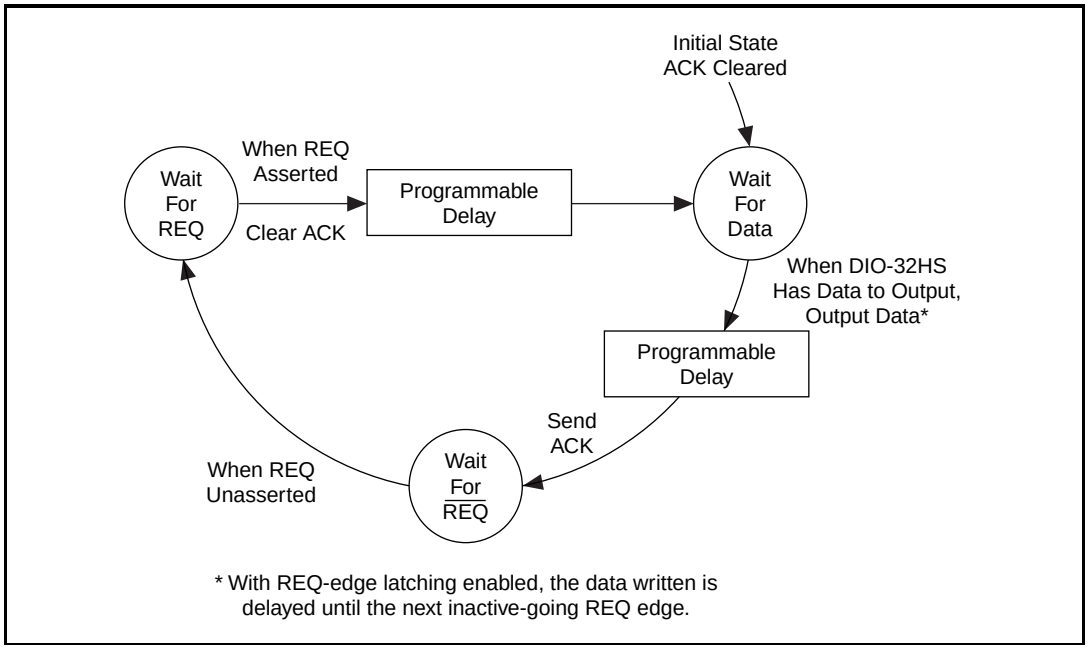
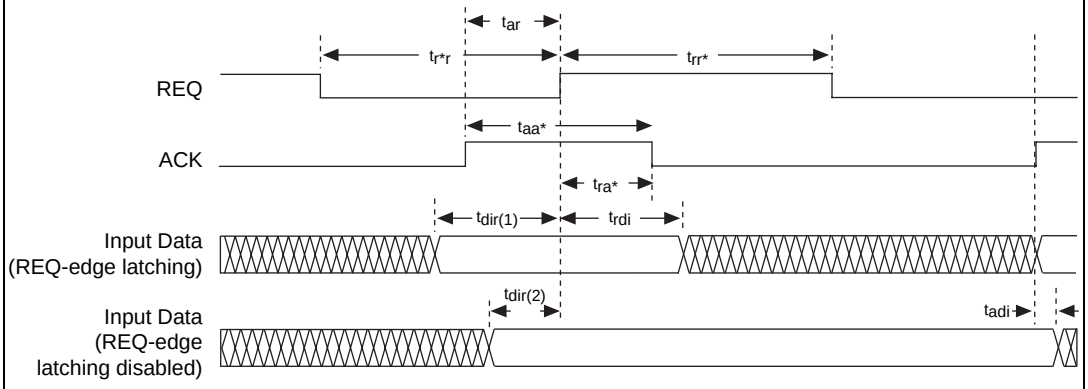


Figure 5-10. Level-ACK Mode Output

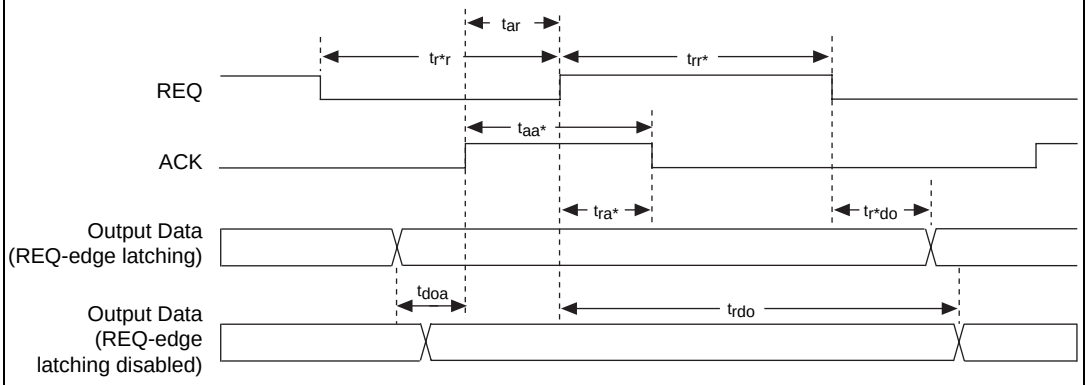
Level-ACK Mode Timing Specifications

Figures 5-11 and 5-12 show the timing diagrams for level-ACK mode.



Parameter	Description	Minimum	Maximum
Input Parameters			
t_{rr*}	REQ pulse width	75	—
t_{r*r}	REQ inactive duration	75	—
t_{ar}	ACK to next REQ	0	—
$t_{dir(1)}$	Input data setup to REQ active (with REQ-edge latching)	0	—
t_{rdi}	Input data hold from REQ active (with REQ-edge latching)	10	—
$t_{dir(2)}$	Input data setup to REQ (with REQ-edge latching disabled)	0	—
t_{adi}	Input data hold from ACK (with REQ-edge latching disabled)	0	—
Output Parameters			
t_{aa*}	ACK pulse width	225	—
t_{ra*}	REQ to ACK inactive	100	200
All timing values are in nanoseconds.			

Figure 5-11. Level-ACK Mode Input Timing



Parameter	Description	Minimum	Maximum
Input Parameters			
t_{rr*}	REQ pulse width	75	—
t_{r*r}	REQ inactive duration	75	—
t_{ar}	ACK to next REQ	0	—
Output Parameters			
t_{aa*}	ACK pulse width	225	—
t_{ra*}	REQ to ACK inactive	100	200
t_{r*do}	REQ inactive to new output data (with REQ-edge latching)	0	50
t_{rdo}	REQ to new output data (with REQ-edge latching disabled)	0	—
t_{doa}	Output data valid to ACK (with REQ-edge latching disabled)	25[1]	—

[1] t_{doa} (min) = 25 + programmable delay

All timing values are in nanoseconds.

Figure 5-12. Level-ACK Mode Output Timing

Leading-Edge Mode

In leading-edge mode, the DIO-32HS and the peripheral device send each other pulses on the ACK and REQ lines. The leading edge of the ACK or REQ pulse indicates that the DIO-32HS or peripheral device is ready for a transfer.

Input

In input mode, the DIO-32HS sends an ACK pulse when ready to receive data. The ACK pulse width is fixed, assuming the peripheral device has deasserted the REQ signal. Otherwise, the ACK signal remains asserted until the REQ signal deasserts. After receiving at least the leading edge of the ACK pulse, the peripheral device can strobe data into the DIO-32HS by asserting the REQ signal. The DIO-32HS sends another ACK pulse when ready for another input.

To slow down the handshake, you can specify a data-settling delay to occur before the ACK signal.

Output

In output mode, the DIO-32HS sends an ACK pulse after driving output data to indicate new, valid output data. The ACK pulse width is fixed, assuming the peripheral device has deasserted the REQ signal. Otherwise, the ACK signal remains until the peripheral device deasserts the REQ signal. The peripheral device can latch the data on the falling or rising edge of the ACK signal, or at any time before returning a REQ pulse. The peripheral device must respond with an active-going REQ signal edge to deassert the ACK signal and request additional data.

To slow down the handshake, you can specify a data-settling delay to occur before the ACK signal. This delay increases the setup time from valid output data to the ACK signal.

Figure 5-13 shows an input transfer in leading-edge mode.

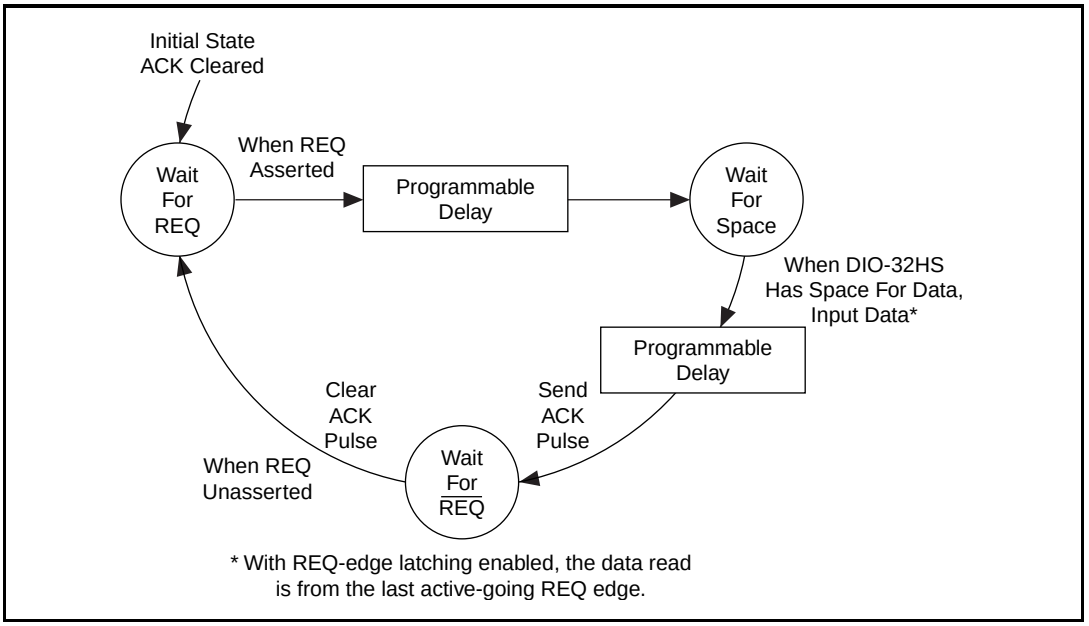


Figure 5-13. Leading-Edge Mode Input

Figure 5-14 shows an output transfer in leading-edge mode.

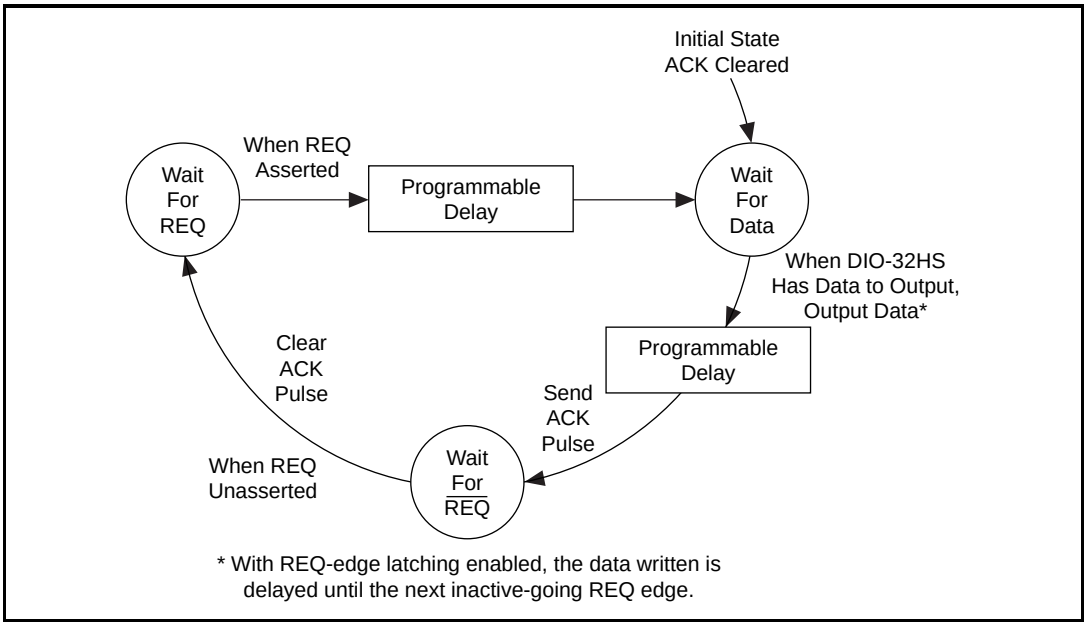
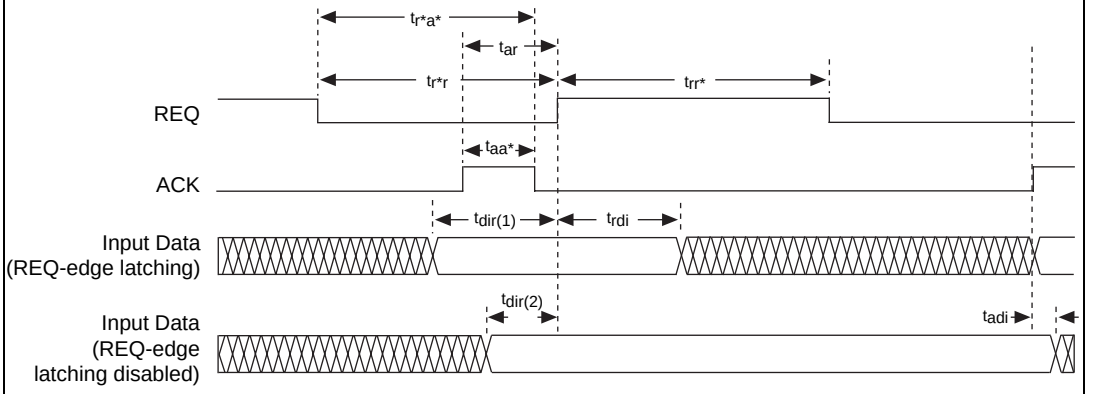


Figure 5-14. Leading-Edge Mode Output

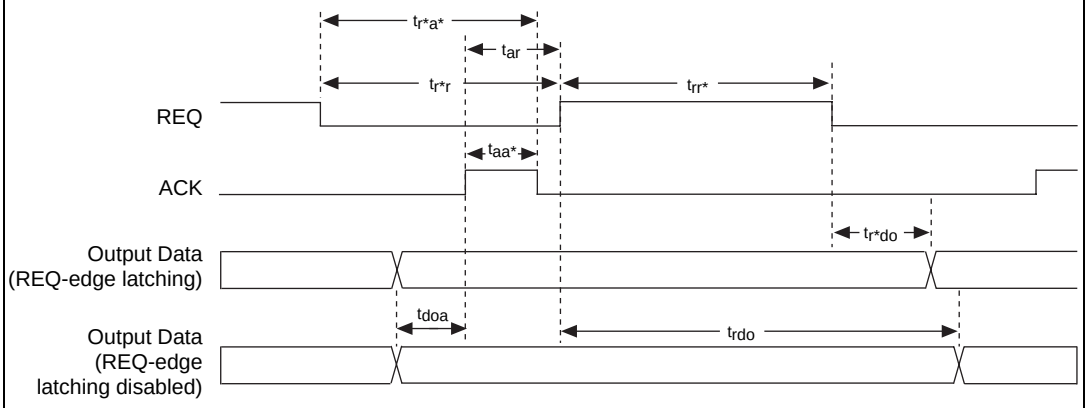
Leading-Edge Mode Timing Specifications

Figures 5-15 and 5-16 show the timing diagrams for leading-edge mode.



Parameter	Description	Minimum	Maximum
Input Parameters			
t_{rr*}	REQ pulse width	75	—
t_{r*r}	REQ inactive duration	75	—
t_{ar}	ACK to next REQ	0	—
$t_{dir(1)}$	Input data setup to REQ active (with REQ-edge latching)	0	—
t_{rdi}	Input data hold from REQ active (with REQ-edge latching)	10	—
$t_{dir(2)}$	Input data setup to REQ (with REQ-edge latching disabled)	0	—
t_{adi}	Input data hold from ACK (with REQ-edge latching disabled)	0	—
Output Parameters			
t_{aa*}	ACK pulse width	125	—
t_{r*a*}	REQ inactive to ACK inactive	150	—
All timing values are in nanoseconds.			

Figure 5-15. Leading-Edge Mode Input Timing



Parameter	Description	Minimum	Maximum
Input Parameters			
t_{rr*}	REQ pulse width	75	—
t_{r*r}	REQ inactive duration	75	—
t_{ar}	ACK to next REQ	0	—
Output Parameters			
t_{aa*}	ACK pulse width	125	—
t_{r*a*}	REQ inactive to ACK inactive	150	—
t_{r*do}	REQ inactive to new output data (with REQ-edge latching)	0	50
t_{rdo}	REQ to new output data (with REQ-edge latching disabled)	0	—
t_{doa}	Output data valid to ACK (with REQ-edge latching disabled)	25[1]	—

[1] t_{doa} (min) = 25 + programmable delay All timing values are in nanoseconds.

Figure 5-16. Leading-Edge Mode Output Timing

Long-Pulse Mode

Long-pulse mode is a variant of leading-edge mode. The only difference is the effect of a data-settling delay, if used. In long-pulse mode, a programmable delay, rather than delaying the ACK pulse, increases the minimum width of the pulse.

Long-pulse mode enables you to handshake with a peripheral device that requires a large minimum pulse width.

Long-pulse mode also enables you to handshake with 8255 emulation mode, if you set the ACK and REQ signals to active low. If you want to use long-pulse mode to handshake with an actual 8255 or 82C55 PPI, make sure you select an adequate minimum pulse width for your 8255 or 82C55. A data-settling delay of 500 ns is sufficient for any current 8255 or 82C55 PPI. Figures 5-17 and 5-18 show long-pulse mode input and output diagrams, respectively.

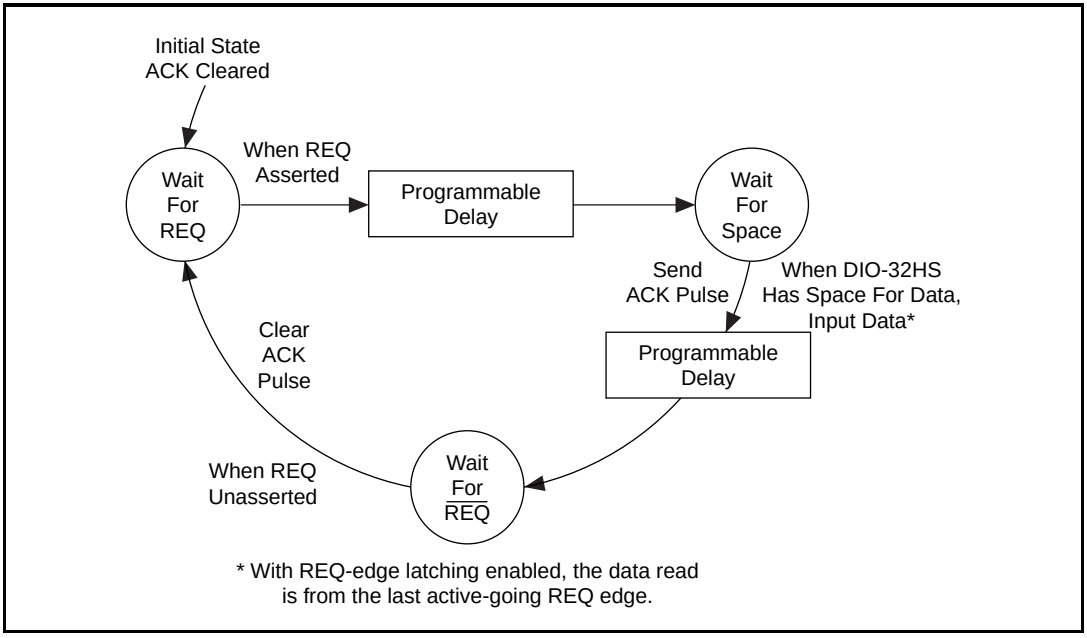


Figure 5-17. Long-Pulse Mode Input

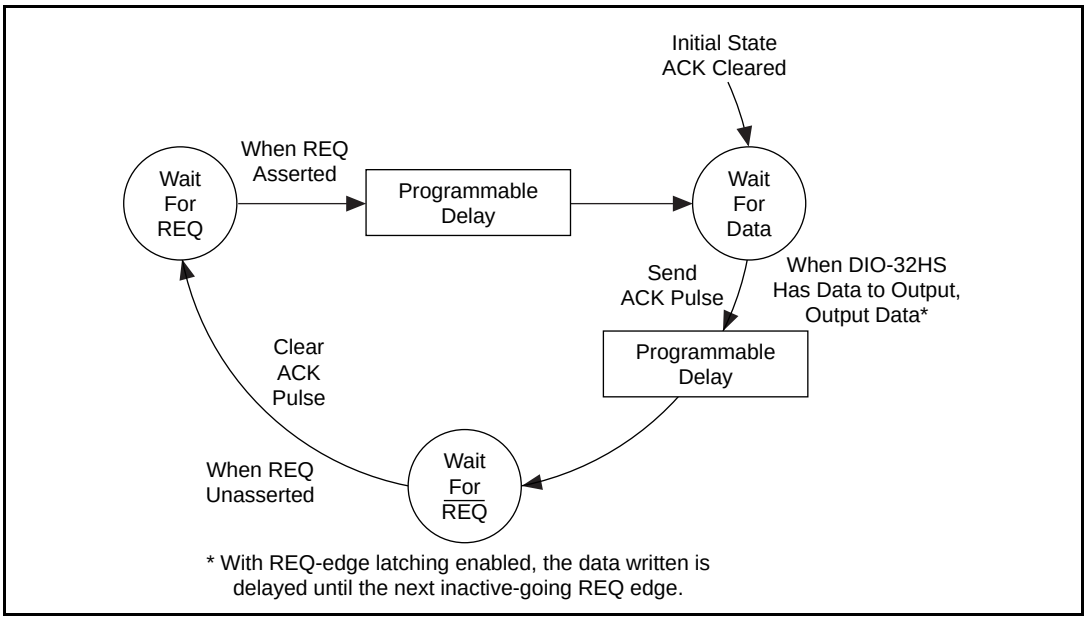
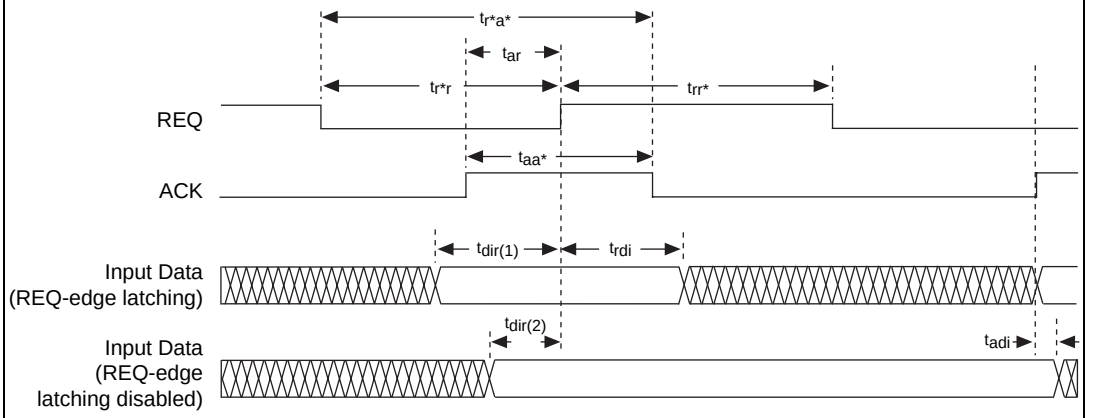


Figure 5-18. Long-Pulse Mode Output

Long-Pulse Mode Timing Specifications

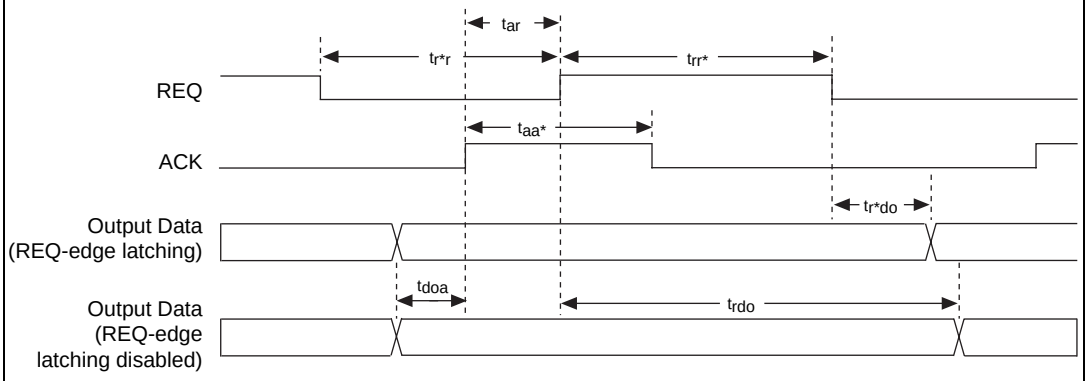
Figures 5-19 and 5-20 show the timing diagrams for long-pulse mode.



Parameter	Description	Minimum	Maximum
Input Parameters			
t_{rr*}	REQ pulse width	75	—
t_{r*r}	REQ inactive duration	75	—
t_{ar}	ACK to next REQ	0	—
$t_{dir(1)}$	Input data setup to REQ active (with REQ-edge latching)	0	—
t_{rdi}	Input data hold from REQ active (with REQ-edge latching)	10	—
$t_{dir(2)}$	Input data setup to REQ (with REQ-edge latching disabled)	0	—
t_{adi}	Input data hold from ACK (with REQ-edge latching disabled)	0	—
Output Parameters			
t_{aa*}	ACK pulse width	125[1]	—
t_{r*a*}	REQ inactive to ACK inactive	150	—

[1] t_{aa*}(min) = 125 + programmable delay All timing values are in nanoseconds.

Figure 5-19. Long-Pulse Mode Input Timing



Parameter	Description	Minimum	Maximum
Input Parameters			
t_{rr*}	REQ pulse width	75	—
t_{r*r}	REQ inactive duration	75	—
t_{ar}	ACK to next REQ	0	—
Output Parameters			
t_{aa*}	ACK pulse width	125[1]	—
t_{r*do}	REQ inactive to new output data (with REQ-edge latching)	0	50
t_{rdo}	REQ to new output data (with REQ-edge latching disabled)	0	—
t_{doa}	Output data valid to ACK (with REQ-edge latching disabled)	25	—

[1] t_{aa*} (min) = 125 + programmable delay All timing values are in nanoseconds.

Figure 5-20. Long-Pulse Mode Output Timing

Trailing-Edge Mode

In trailing-edge mode, the DIO-32HS and the peripheral device send each other pulses on the ACK and REQ lines. The trailing edge of the ACK or REQ pulse indicates that the DIO-32HS or peripheral device is ready for a transfer.

Input

In input mode, the DIO-32HS sends an ACK pulse of programmable width when ready to receive data. After receiving the trailing edge of the ACK pulse, the peripheral device can strobe data into the DIO-32HS by deasserting the REQ signal. The DIO-32HS sends another ACK pulse when ready for another input.

To slow down the handshake, you can specify a data-settling delay to increase the ACK pulse width.

Output

In output mode, the DIO-32HS sends an ACK pulse of programmable width after driving output data to indicate new, valid output data. The peripheral device can latch the data on the falling or rising edge of the ACK signal, or at any time before ending the REQ pulse. The peripheral device must respond with a REQ pulse, the trailing edge of which deasserts the ACK signal and requests additional data.

To slow down the handshake, you can specify a data-settling delay to increase the ACK pulse width and, therefore, the setup time from valid output data to the trailing edge of the ACK signal.

Figure 5-21 shows an input transfer in trailing-edge mode.

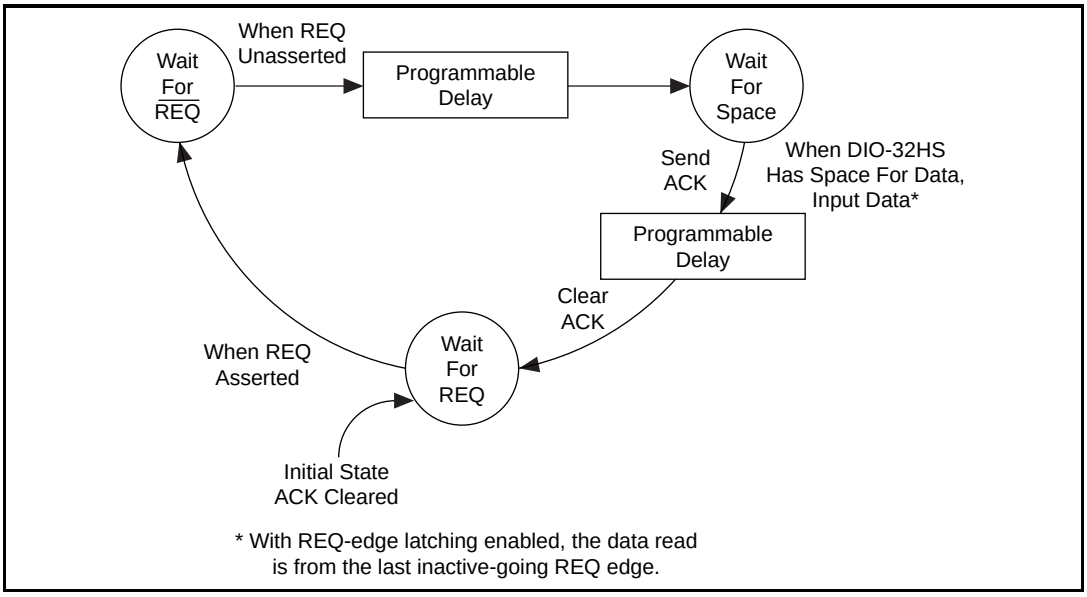


Figure 5-21. Trailing-Edge Mode Input

Figure 5-22 shows a write transfer in trailing edge mode.

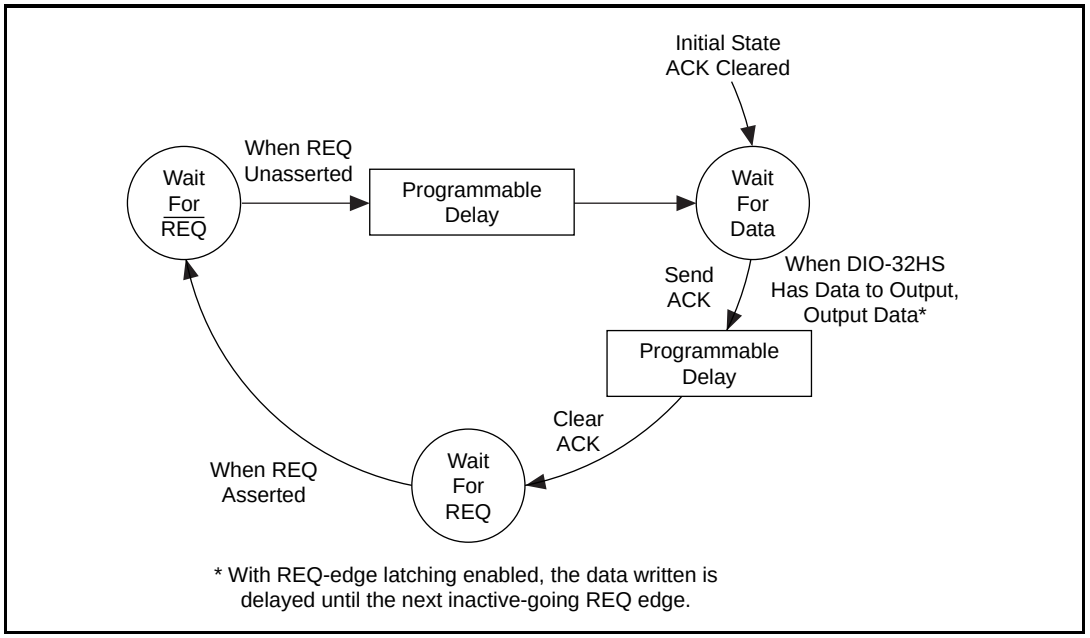
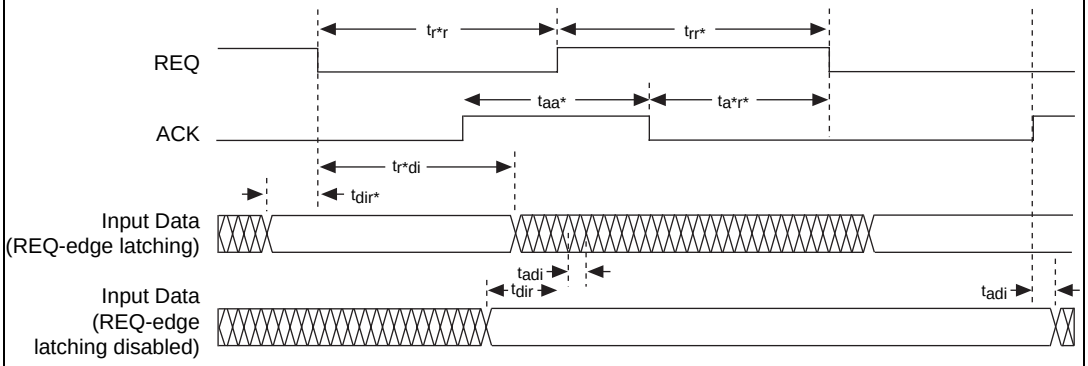


Figure 5-22. Trailing-Edge Mode Output

Trailing-Edge Mode Timing Specifications

Figures 5-23 and 5-24 show the timing diagrams for trailing-edge mode.



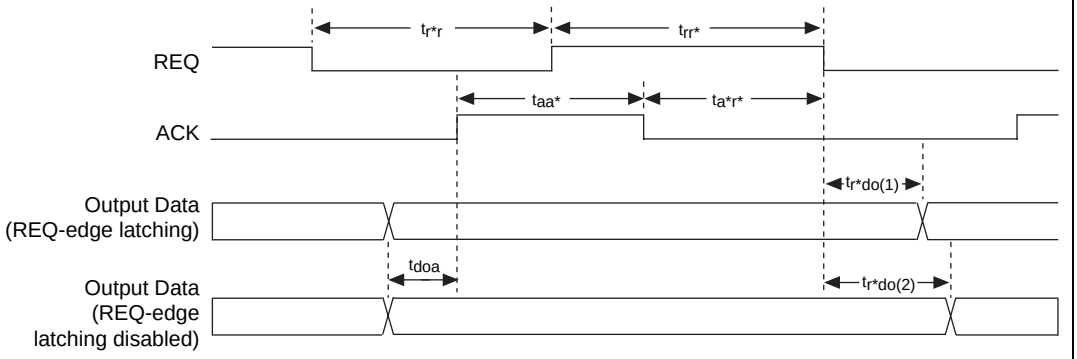
Parameter	Description	Minimum	Maximum
Input Parameters			
t_{rr*}	REQ pulse width	75	—
t_{r*r}	REQ inactive duration	75	—
t_{dir*}	Input data setup to REQ inactive (with REQ-edge latching)	0	—
t_{r*di}	Input data hold from REQ inactive (with REQ-edge latching)	10	—
t_{dir}	Input data setup to REQ (with REQ-edge latching disabled)	0	—
t_{adi}	Input data hold from ACK (with REQ-edge latching disabled)	0	—
Output Parameters			
t_{aa*}	ACK pulse width	225[1]	275[2]
t_{a*r*}	ACK inactive to next REQ inactive	0	—

[1] t_{aa*} (min) = 225 + programmable delay

[2] t_{aa*}(max) = 275 + programmable delay

All timing values are in nanoseconds.

Figure 5-23. Trailing-Edge Mode Input Timing



Parameter	Description	Minimum	Maximum
Input Parameters			
t_{rr*}	REQ pulse width	75	—
t_{r*r}	REQ inactive duration	75	—
t_{a*r*}	ACK inactive to next REQ inactive	0	—
Output Parameters			
t_{aa*}	ACK pulse width	225[1]	275[2]
$t_{r*do(1)}$	REQ inactive to new output data (with REQ-edge latching)	0	50
$t_{r*do(2)}$	REQ inactive to new output data (with REQ-edge latching disabled)	0	—
t_{doa}	Output data valid to ACK (with REQ-edge latching disabled)	25	—

[1] t_{aa*} (min) = 225 + programmable delay

[2] t_{aa*} (max) = 275 + programmable delay

All timing values are in nanoseconds.

Figure 5-24. Trailing-Edge Mode Output Timing

Burst Mode

Burst mode is a synchronous protocol. The data transmitter and receiver share a clock signal driven by the data receiver onto the PCLK line.

In every clock cycle, the DIO-32HS asserts the ACK signal if it is ready to perform a transfer. If the peripheral device also asserts the REQ signal, a transfer occurs on the rising clock edge. Either the DIO-32HS

or the peripheral device can insert wait states into the protocol by deasserting the ACK or REQ signal, respectively.

Burst Mode Timing Specifications

Figure 5-25 shows a burst mode transfer example, where D1 is data point number one, and D2 is data point number 2. The data-out line shows an output example, and the data-in line shows an input example. Figures 5-26 and 5-27 show the burst mode timing diagrams.

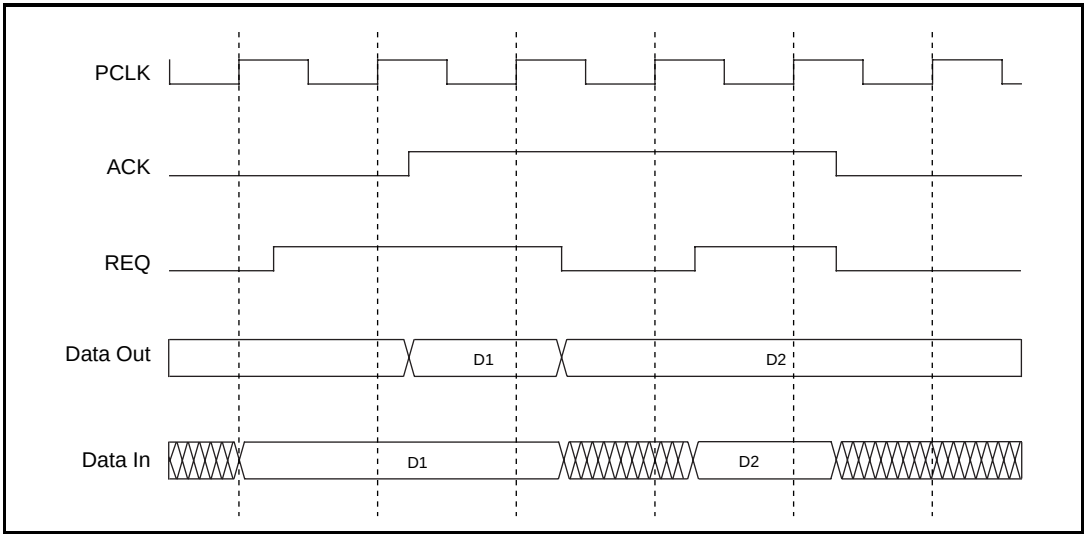
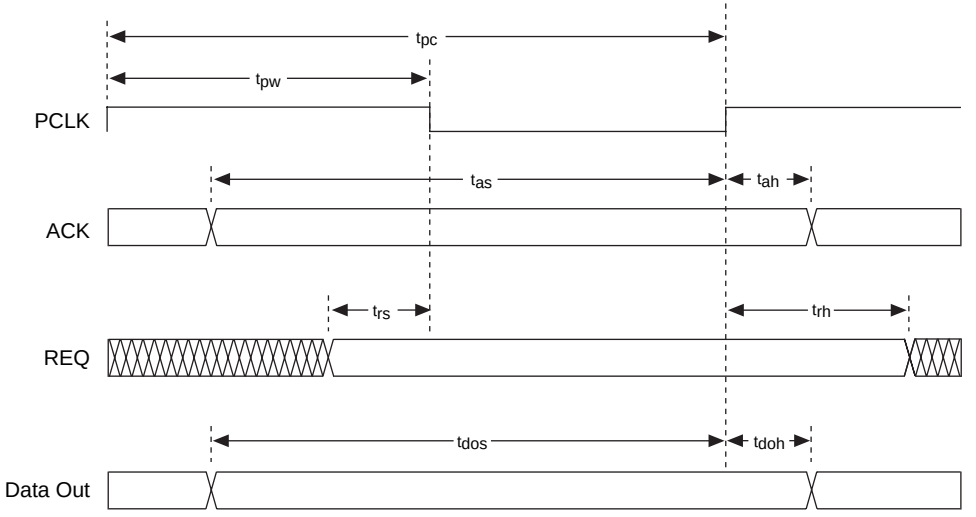
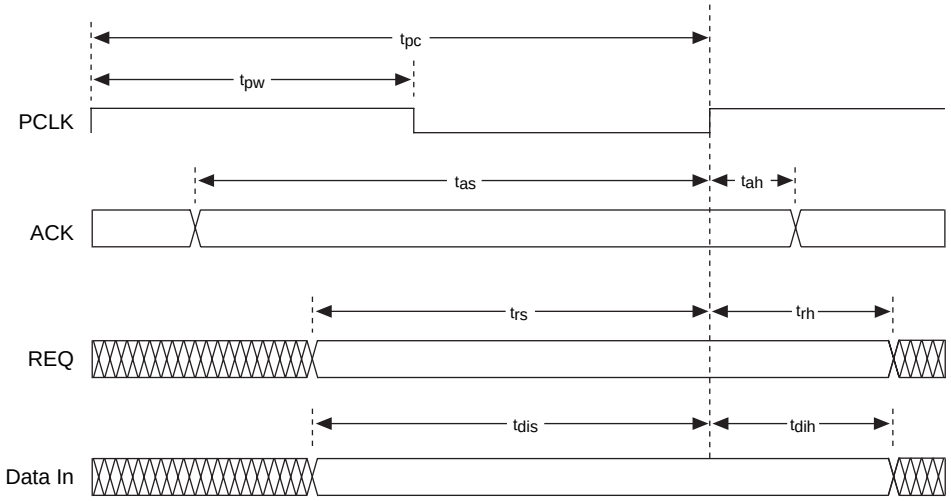


Figure 5-25. Burst Mode Transfer Example



Parameter	Description	Minimum	Maximum
Input Parameters			
t_{pc}	PCLK cycle time	50	—
t_{pw}	PCLK high pulse duration	20	t_{pc} – 20
t_{rs}	Setup time from REQ valid to PCLK falling edge	1	—
t_{rh}	Hold time from PCLK to REQ invalid	0	—
Output Parameters			
t_{as}	Setup time from ACK valid to PCLK	t_{pc} – 12	—
t_{ah}	Hold time from PCLK to ACK invalid	3	—
t_{dos}	Setup time from output data valid to PCLK	t_{pc} – 18	—
t_{doh}	Hold time from PCLK to output data invalid	5	—
All timing values are in nanoseconds.			

Figure 5-26. Burst Mode Output Timing



Parameter	Description	Minimum	Maximum
Input Parameters			
t_{rh}	Hold time from PCLK to REQ invalid	0	—
t_{rs}	Setup time from REQ valid to PCLK	$t_{pc} - 38$	—
t_{dis}	Setup time from input data valid to PCLK	4	—
t_{dih}	Hold time from PCLK to input data invalid	6	—
Output Parameters			
t_{pc}	PCLK cycle time	50	700[1]
t_{pw}	PCLK high pulse duration	$t_{pc}/2 - 5$	$t_{pc}/2 + 5$
t_{as}	Setup time from ACK valid to PCLK	32	—
t_{ah}	Hold time from PCLK to ACK invalid	3	—

[1] t_{pc} = programmable delay from 100 to 700 ns, or 50 ns if programmable delay is 0. Timebase stability for the onboard 20 MHz oscillator is 50 ppm.

All timing values are in nanoseconds.

Figure 5-27. Burst Mode Input Timing

Specifications

Appendix
A

This appendix lists the specifications for the AT-DIO-32HS and PCI-DIO-32HS boards. These specifications are typical at 25°C unless otherwise noted.

AT/PCI-DIO-32HS Boards

Digital I/O

Number of channels .....32 input/output
4 dedicated output and control;
4 dedicated input and status

Compatibility .....TTL/CMOS (standard or wired-OR)

Hysteresis .....500 mV

Digital logic levels

Level	Min	Max
Input low voltage	0 V	0.8 V
Input high voltage	2 V	Vcc
Input low current for data lines (V_{in} = 0.4 V) DPULL high DPULL low	 — —	 -70 µA -10 µA
Input high current for data lines (V_{in} = 2.4 V) DPULL high DPULL low	 — —	 10 µA 40 µA
Input low current for control lines (V_{in} = 0.4 V) CPULL high CPULL low	 — —	 -2.5 mA -200 µA
Input high current for control lines (V_{in} = 2.4 V) CPULL high CPULL low	 — —	 200 µA 1.4 mA
Input low current for CPULL/DPULL (V_{in} = 0.4 V)	—	4 µA
Input high current for CPULL/DPULL (V_{in} = 2.4 V)	—	140 µA

Level	Min	Max
Output low voltage (I_{OL} = 24 mA)	—	0.4 V
Output high voltage* (I_{OH} = 24 mA)	2.4 V	—
* When configured as standard outputs. Drivers configured as wired-OR outputs are tri-stated when logic is high.		

Absolute max input voltage range.....-0.3 V to V_{CC} + 0.3 V

Power-on state for outputs .....Tri-stated, pulled up or down (selectable)

Data transfers.....Programmed I/O, DMA

Strobed I/O

Pattern Generation

Direction.....Input or output

Modes.....Internally or externally timed

Sample rate (max sustainable)..... System dependent[1]

Mode	Triton I Chip Set	Triton II Chip Set	Natoma Chip Set
AT-DIO-32HS	**Rates in kS/s (kbytes/s) on Sample Systems**		
32-bit	350 (1400)	—	—
16-bit	700 (1400)	—	—
8-bit	1400 (1400)	—	—
PCI-DIO-32HS	**Rates in MS/s (MB/s) on Sample Systems**		
32-bit input	2.8 (11.2)	4 (16)	4 (16)
16-bit input	4 (8)	5 (10)	6.67 (13.33)
8-bit input	6.67 (6.67)	10 (10)	10 (10)
32-bit output	1 (4)	2 (8)	3.33 (13.33)
16-bit output	1 (2)	2.5 (5)	3.33 (6.67)
8-bit output	2 (2)	5 (5)	6.67 (6.67)

Sample rate (peak internally timed, for small transfers) ..... 10 MS/s

Sample rate (peak externally timed, for small transfers) ..... 20 MS/s

Sample rate (min internally timed)..... 1 S/10 min.

Sample rate (min externally timed) .... no limit

1. Pattern generation rates depend on your computer, software, and other bus activity. The rates shown were measured for 100 kS transfers on sample Intel Pentium-based computers, using NI-DAQ software, with no other DAQ operations in progress. The PCI Triton I rates were measured on a 100 MHz Pentium computer with the Triton I (430FX) chip set. The PCI Triton II rates were measured on a 166 MHz Pentium computer with the Triton II HX (430HX) chip set. The PCI Natoma rates were measured on a 180 MHz Pentium Pro system with the Natoma (440FX) chip set. The AT-DIO-32HS rates were measured using the dual-DMA transfer method on a 100 MHz Pentium computer with the Triton I (430FX) chip set.

Handshaking

Direction.....Input or output

Modes.....6 (burst, level-ACK, leading-edge pulse, trailing-edge pulse, long pulse, and 8255 emulation)

Transfer rate[1] (max)

AT-DIO-32HS .....up to 1.8 MB/s (450 kS/s) at 32 bits; up to 1.8 MB/s (900 kS/s) at 16 bits; up to 1.8 MB/s (1.8 kS/s) at 8 bits

PCI-DIO-32HS.....up to 76 MB/s (19 MS/s) at 32 bits; up to 38 MB/s (19 MS/s) at 16 bits; up to 19 MB/s (19 MS/s) at 8 bits

Triggers

Start and Stop Triggers

Compatibility.....TTL/CMOS

Trigger types.....Rising or falling edge, or digital pattern

Pulse width for edge triggers (min).....10 ns

Pattern triggers detection capabilities.....Detect pattern match or mismatch on user-selected bits

1. Handshaking rates depend on your computer, software, other bus activity, and handshaking protocol. The rates shown were measured on a sample 100 MHz Intel Pentium-based computer with an Intel 430FX (Triton I) chip set, using NI-DAQ software and the burst-mode handshaking protocol, with no other DAQ operations in progress.

RTSI Triggers

Trigger lines 7

Bus Interfaces

Type (AT-DIO-32HS) ..... AT slave with dual DMA

Type (PCI-DIO-32HS) ..... PCI master and target with onboard linking (scatter-gather) DMA

Power Requirement

+5 VDC (±5%)
(with light output load) ..... 500 mA

Power available at I/O connector..... +4.65 to +5.25 VDC at 1 A

Physical

Dimensions, not including
connectors ..... 17.5 by 10.7 cm (6.9 by 4.2 in.)

I/O connector ..... 68-pin male SCSI-II type

Environment

Operating temperature..... 0 to 55° C

Storage temperature ..... -40 to 125° C

Relative humidity..... 5% to 90% noncondensing

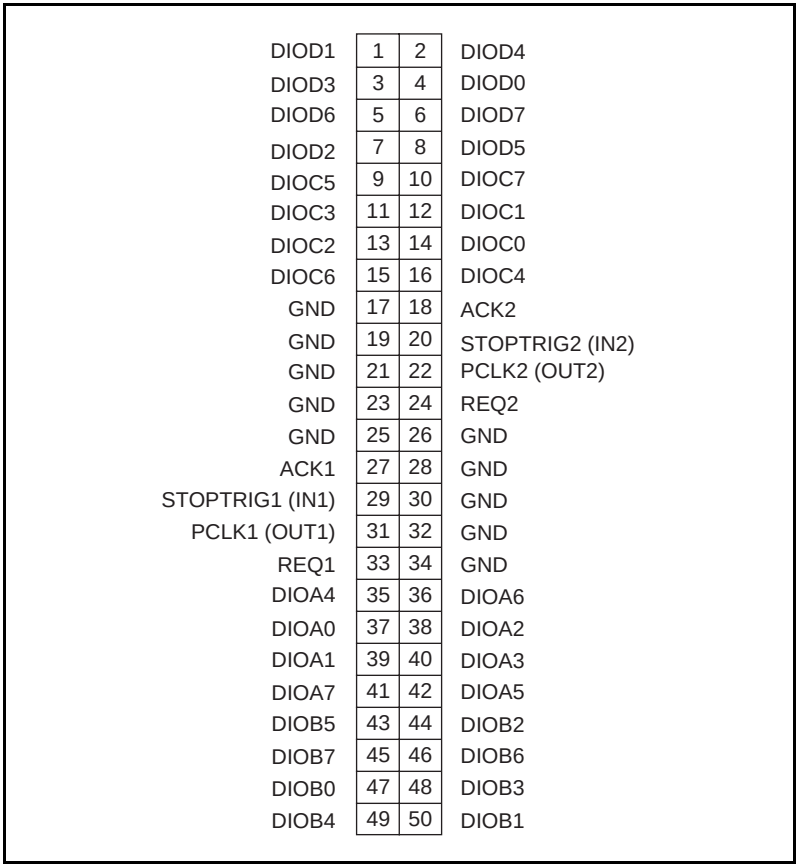
Optional Adapter Description

Appendix
B

This appendix describes the optional 68-to-50-pin DIO-32HS adapter. The adapter changes the pinout of the DIO-32HS to match the pinout of an AT-DIO-32F board. The adapter enables you to use the DIO-32HS with cables, signal conditioning modules, and other accessories that require an AT-DIO-32F pinout.

The female side of the adapter connects directly to the DIO-32HS board. The male side of the adapter provides the pin assignments shown in Figure B-1. See Chapter 4, *Signal Connections*, for a description of each signal.

The 50-pin adapter connector has no +5 V, CPULL, or DPULL pins and has fewer ground pins than the 68-pin DIO-32HS connector. Figure B-1 shows the 50-pin adapter pin assignment.



DIOD1	1	2	DIOD4
DIOD3	3	4	DIOD0
DIOD6	5	6	DIOD7
DIOD2	7	8	DIOD5
DIOC5	9	10	DIOC7
DIOC3	11	12	DIOC1
DIOC2	13	14	DIOC0
DIOC6	15	16	DIOC4
GND	17	18	ACK2
GND	19	20	STOPTRIG2 (IN2)
GND	21	22	PCLK2 (OUT2)
GND	23	24	REQ2
GND	25	26	GND
ACK1	27	28	GND
STOPTRIG1 (IN1)	29	30	GND
PCLK1 (OUT1)	31	32	GND
REQ1	33	34	GND
DIOA4	35	36	DIOA6
DIOA0	37	38	DIOA2
DIOA1	39	40	DIOA3
DIOA7	41	42	DIOA5
DIOB5	43	44	DIOB2
DIOB7	45	46	DIOB6
DIOB0	47	48	DIOB3
DIOB4	49	50	DIOB1

Figure B-1. 68-to-50-Pin Adapter Pin Assignments

Customer Communication

Appendix
C

For your convenience, this appendix contains forms to help you gather the information necessary to help us solve your technical problems and a form you can use to comment on the product documentation. When you contact us, we need the information on the Technical Support Form and the configuration form, if your manual contains one, about your system configuration to answer your questions as quickly as possible.

National Instruments has technical assistance through electronic, fax, and telephone systems to quickly provide the information you need. Our electronic services include a bulletin board service, an FTP site, a Fax-on-Demand system, and e-mail support. If you have a hardware or software problem, first try the electronic support systems. If the information available on these systems does not answer your questions, we offer fax and telephone support through our technical support centers, which are staffed by applications engineers.

Electronic Services

Bulletin Board Support

National Instruments has BBS and FTP sites dedicated for 24-hour support with a collection of files and documents to answer most common customer questions. From these sites, you can also download the latest instrument drivers, updates, and example programs. For recorded instructions on how to use the bulletin board and FTP services and for BBS automated information, call (512) 795-6990. You can access these services at:

United States: (512) 794-5422
Up to 14,400 baud, 8 data bits, 1 stop bit, no parity

United Kingdom: 01635 551422
Up to 9,600 baud, 8 data bits, 1 stop bit, no parity

France: 01 48 65 15 59
Up to 9,600 baud, 8 data bits, 1 stop bit, no parity

FTP Support

To access our FTP site, log on to our Internet host, `ftp.natinst.com`, as `anonymous` and use your Internet address, such as `joesmith@anywhere.com`, as your password. The support files and documents are located in the `/support` directories.

Fax-on-Demand Support

Fax-on-Demand is a 24-hour information retrieval system containing a library of documents on a wide range of technical information. You can access Fax-on-Demand from a touch-tone telephone at (512) 418-1111.

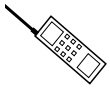
E-Mail Support (currently U.S. only)

You can submit technical support questions to the applications engineering team through e-mail at the Internet address listed below. Remember to include your name, address, and phone number so we can contact you with solutions and suggestions.

`support@natinst.com`

Telephone and Fax Support

National Instruments has branch offices all over the world. Use the list below to find the technical support number for your country. If there is no National Instruments office in your country, contact the source from which you purchased your software to obtain support.



	Telephone	Fax
Australia	03 9879 5166	03 9879 6277
Austria	0662 45 79 90 0	0662 45 79 90 19
Belgium	02 757 00 20	02 757 03 11
Canada (Ontario)	905 785 0085	905 785 0086
Canada (Quebec)	514 694 8521	514 694 4399
Denmark	45 76 26 00	45 76 26 02
Finland	09 527 2321	09 502 2930
France	01 48 14 24 24	01 48 14 24 14
Germany	089 741 31 30	089 714 60 35
Hong Kong	2645 3186	2686 8505
Israel	03 5734815	03 5734816
Italy	02 413091	02 41309215
Japan	03 5472 2970	03 5472 2977
Korea	02 596 7456	02 596 7455
Mexico	5 520 2635	5 520 3282
Netherlands	0348 433466	0348 430673
Norway	32 84 84 00	32 84 86 00
Singapore	2265886	2265887
Spain	91 640 0085	91 640 0533
Sweden	08 730 49 70	08 730 43 70
Switzerland	056 200 51 51	056 200 51 55
Taiwan	02 377 1200	02 737 4644
U.K.	01635 523545	01635 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 (include version number) _____

Clock 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 reproduce the problem:_____

_____

_____

_____

_____

_____

AT/PCI-DIO-32HS 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

DAQ hardware _____

Serial number of hardware _____

Interrupt level of hardware _____

DMA channels of hardware _____

Base I/O address of hardware _____

Programming choice _____

ComponentWorks, NI-DAQ, LabVIEW, LabWindows/CVI, or other version _____

Other boards in system _____

Base I/O address of other boards _____

DMA channels of other boards _____

Interrupt level of other boards _____

Other Products

Computer make and model _____

Microprocessor _____

Clock frequency or speed _____

Type of video board installed _____

Operating system version _____

Operating system mode _____

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/PCI-DIO-32HS User Manual*

Edition Date: February 1997

Part Number: 321464A-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 (___)_____ Fax (___) _____

Mail to: Technical Publications
National Instruments Corporation
6504 Bridge Point Parkway
Austin, TX 78730-5039

Fax to: Technical Publications
National Instruments Corporation
(512) 794-5678

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

Symbols

°	degree
-	negative of, or minus
Ω	ohm
/	per
%	percent
±	plus or minus
+	positive of, or plus

A

A	amperes
AC	alternating current
ACK	acknowledge signal
address	character code that identifies a specific location (or series of locations) in memory
ANSI	American National Standards Institute
asynchronous	hardware—a property of an event that occurs at an arbitrary time, without synchronization to a reference clock

B

b	bit—one binary digit, either 0 or 1
B	byte—eight related bits of data, an eight-bit binary number. Also used to denote the amount of memory required to store one byte of data.
base address	a memory address that serves as the starting address for programmable registers. All other addresses are located by adding to the base address.
bus	the group of conductors that interconnect individual circuitry in a computer. Typically, a bus is the expansion vehicle to which I/O or other devices are connected. Examples of PC buses are the AT, EISA, and PCI bus.

C

C	Celsius
clock	hardware component that controls timing for reading from or writing to groups
CMOS	complementary metal-oxide semiconductor
CPU	central processing unit
CPULL	control pullup/pulldown selection

crosstalk	an unwanted signal on one channel due to an input on a different channel
current drive capability	the amount of current a digital or analog output channel is capable of sourcing or sinking while still operating within voltage range specifications
current sinking	the ability of a DAQ board to dissipate current for analog or digital output signals
current sourcing	the ability of a DAQ board to supply current for analog or digital output signals

D

DAQ	data acquisition—(1) collecting and measuring electrical signals from sensors, transducers, and test probes or fixtures and inputting them to a computer for processing; (2) collecting and measuring the same kinds of electrical signals with A/D and/or DIO boards plugged into a computer, and possibly generating control signals with D/A and/or DIO boards in the same computer
DC	direct current
device	a plug-in data acquisition board, card, or pad that can contain multiple channels and conversion devices. Plug-in boards, PCMCIA cards, and devices such as the DAQPad-1200, which connects to your computer parallel port, are all examples of DAQ devices. SCXI modules are distinct from devices, with the exception of the SCXI-1200, which is a hybrid.
digital input group	a collection of digital input ports. You can associate each group with its own clock rates, handshaking modes, buffer configurations, and so on. A port cannot belong to more than one group.
digital output group	a collection of digital output ports. You can associate each group with its own clock rates, handshaking modes, buffer configurations, and so forth. A port cannot belong to more than one group.
digital port	*See port.*
digital trigger	a TTL level signal having two discrete levels—a high and a low level—that starts or stops an operation

DIO	digital input/output
DLL	dynamic link library—a software module in Microsoft Windows containing executable code and data that can be called or used by Windows applications or other DLLs. Functions and data in a DLL are loaded and linked at run time when they are referenced by a Windows application or other DLLs.
DMA	direct memory access—a method by which data can be transferred to/from computer memory from/to a device or memory on the bus while the processor does something else. DMA is the fastest method of transferring data to/from computer memory.
DPULL	data pullup/pulldown selection
drivers	software that controls a specific hardware device such as a DAQ board

E

EEPROM	electrically erasable programmable read-only memory–ROM that can be erased with an electrical signal and reprogrammed
EISA	extended industry standard architecture
event	the condition or state of an analog or digital signal
external trigger	a voltage pulse from an external source that triggers an event such as A/D conversion

F

FIFO	first-in first-out memory buffer–the first data stored is the first data sent to the acceptor. FIFOs are often used on DAQ devices to temporarily store incoming or outgoing data until that data can be retrieved or output. For example, an analog input FIFO stores the results of A/D conversions until the data can be retrieved into system memory, a process that requires the servicing of interrupts and often the programming of the DMA controller. This process can take several milliseconds in some cases. During this time, data accumulates in the

	FIFO for future retrieval. With a larger FIFO, longer latencies can be tolerated. In the case of analog output, a FIFO permits faster update rates, because the waveform data can be stored on the FIFO ahead of time. This again reduces the effect of latencies associated with getting the data from system memory to the DAQ device.
ft.	feet

G

glitch	a brief, unwanted change, or disturbance, in a signal level
GND	ground

H

h	hour
handshaked digital I/O	a type of strobed digital I/O in which control signals pass both to and from the digital device, timing and confirming each data transfer. Also called *full*, or *two-way* handshaking, to distinguish this type of transfer from pattern generation.
hardware	the physical components of a computer system, such as the circuit boards, plug-in boards, chassis, enclosures, peripherals, cables, and so on
hardware triggering	a form of triggering where you set the start time of an acquisition and gather data at a known position in time relative to a trigger signal
hex	hexadecimal
Hz	hertz–the number of scans read or updates written per second

I

IBM	International Business Machines
IC	integrated circuit
ID	identification

in.	inches
interrupt	a computer signal indicating that the CPU should suspend its current task to service a designated activity
interrupt level	the relative priority at which a device can interrupt
I/O	input/output–the transfer of data to/from a computer system involving communications channels, operator interface devices, and/or data acquisition and control interfaces
I_{OH}	current, output high
I_{OL}	current, output low
IRQ	interrupt request signal
ISA	industry standard architecture

K

kbytes/s	a unit for data transfer that means 1,000, or 10^3, bytes/s
kS	1,000 samples
Kword	1,024 words of memory

L

LabVIEW	laboratory virtual instrument engineering workbench
latched digital I/O	*see strobed digital I/O*
LED	light-emitting diode
LSB	least significant bit

M

m	meters
max	maximum

MB	megabytes of memory
min	minimum
min.	minute
MSB	most significant bit
mux	multiplexer—a switching device with multiple inputs that sequentially connects each of its inputs to its output, typically at high speeds, in order to measure several signals with a single analog input channel

N

NI-DAQ	NI driver software for DAQ hardware
noise	an undesirable electrical signal–Noise comes from external sources such as the AC power line, motors, generators, transformers, fluorescent lights, soldering irons, CRT displays, computers, electrical storms, welders, radio transmitters, and internal sources such as semiconductors, resistors, and capacitors. Noise corrupts signals you are trying to send or receive.
nonlatched digital I/O	*see unstrobed digital I/O*

O

operating system	base-level software that controls a computer, runs programs, interacts with users, and communicates with installed hardware or peripheral devices
optical isolation	the technique of using an optoelectric transmitter and receiver to transfer data without electrical continuity, to eliminate high-potential differences and transients

P

pattern generation	a type of strobed digital I/O in which timing signals pass either to or from the digital device, but not in both directions. Typically, the timing signals occur at a constant rate, resulting in constant-rate digital data acquisition or waveform generation.

PCI	Peripheral Component Interconnect–a high-performance expansion bus architecture originally developed by Intel to replace ISA and EISA. It is achieving widespread acceptance as a standard for PCs and workstations; it offers a theoretical maximum transfer rate of 132 Mbytes/s.
PCLK	peripheral clock signal
PCMCIA	an expansion bus architecture that has found widespread acceptance as a de facto standard in notebook-size computers. It originated as a specification for add-on memory cards written by the Personal Computer Memory Card International Association.
peripheral device	an external device that a board controls, monitors, tests, or communicates with
Plug and Play devices	devices that do not require dip switches or jumpers to configure resources on the devices—also called switchless devices
Plug and Play ISA	a specification prepared by Microsoft, Intel, and other PC-related companies that will result in PCs with plug-in boards that can be fully configured in software, without jumpers or switches on the boards
port	(1) a communications connection on a computer or a remote controller (2) a digital port, consisting of four or eight lines of digital input and/or output
posttrigger acquisition	the technique used on a DAQ board to acquire a programmed number of samples after trigger conditions are met
PPI	programmable peripheral interface
ppm	parts per million
pretrigger acquisition	the technique used on a DAQ board to keep a continuous buffer filled with data, so that when the trigger conditions are met, the sample includes the data leading up to the trigger condition
programmed I/O	the standard method a CPU uses to access an I/O device each byte of data is read or written by the CPU
protocol	the exact sequence of bits, characters, and control codes used to transfer data between computers and peripherals through a communications channel, such as the GPIB

R

RAM	random-access memory
REQ	request signal
ribbon cable	a flat cable in which the conductors are side by side
RGND	reserved ground
rms	root mean square
RTSI Bus	real-time system integration bus—the National Instruments timing bus that connects DAQ boards directly, by means of connectors on top of the boards, for precise synchronization of functions

S

s	seconds
S	samples
settling time	the amount of time required for a voltage to reach its final value within specified limits
S/s	samples per second
STARTTRIG	start trigger signal
STOPTRIG	stop trigger signal
strobed digital I/O	a type of digital input or output in which hardware uses timing signals to regulate the rate of input or output. Types of strobed digital I/O include *handshaking* and *pattern generation*.
switchless device	devices that do not require dip switches or jumpers to configure resources on the devices—also called *Plug and Play* devices
synchronous	hardware—a property of an event that is synchronized to a reference clock

T

transfer rate	the rate, measured in bytes/s, at which data is moved from source to destination after software initialization and set up operations; the maximum rate at which the hardware can operate
trigger	any event that causes, starts, or stops some form of data capture
tri-state	a third output state, other than high or low, in which the output is undriven
TTL	transistor-transistor logic

U

unstrobed digital I/O	a type of digital input or output in which software reads or writes the digital line or port states directly, without using any handshaking or hardware-controlled timing functions. Also called *immediate*, *nonhandshaking*, or *unlatched* digital I/O.
update	the output equivalent of a scan. One or more analog or digital output samples. Typically, the number of output samples in an update is equal to the number of channels in the output group. For example, one pulse from the update clock produces one update which sends one new sample to every analog output channel in the group.

V

V	volts
V_{cc}	the voltage of the power supply from the computer, approximately 5 V
VDC	volts direct current
V_{IH}	volts, input high
V_{IL}	volts, input low
V_{in}	volts in
V_{OH}	volts, output high

V_{OL}	volts, output low
V_{ref}	reference voltage

W

wire	data path between nodes
wired-OR	a type of output driver that provides sink current but little or no source current, and is typically used with a pull-up resistor to provide source current. If you connect two or more wired-OR outputs together, any one of the output drivers can drive the resulting connection low. Also called an *open-collector* or *open-drain driver*.
word	the standard number of bits that a processor or memory manipulates at one time. Microprocessors typically use 8, 16, or 32-bit words.
working voltage	the highest voltage that should be applied to a product in normal use, normally well under the breakdown voltage for safety margin

Index

Numbers

A

B

C

D

E

F

G

H

I

L

N

O

P

R

S

T

U