PCI 14 Series

PCI PnP Digital I/O Board User's Manual

Revision 1.3

Eagle Technology – Cape Town, South Africa

www.eagle.co.za

Digital I/O Boards

Data Acquisition and Process Control

© Eagle Technology
31-35 Hout Street • Cape Town • South Africa
Phone +27 21 423 4943 • Fax +27 21 424 4637
Email eagle@eagle.co.za

Copyright

Product Warranty

Eagle Technology, South Africa, warrants its products from defect in material and workmanship from confirmed date of purchase for a period of one year if the conditions listed below are met. The product warranty will call the Eagle Technology Data Acquisition Device short as **ETDAQD**.

- The warranty does not apply to an **ETDAQD** that has been previously repaired, altered, extended by any other company or individual outside the premises of Eagle Technology.
- That a qualified person configure and install the **ETDAQD**, and damages caused to a device during installation shall make the warranty void and null.
- The warranty will not apply to conditions where the **ETDAQD** has been operated in a manner exceeding its specifications.

Eagle Technology, South Africa, does not take responsibility or liability of consequential damages, project delays, damaging of equipment or capital loss as a result of its products.

Eagle Technology, South Africa, holds the option and final decision to repair or replace any **ETDAQD.** Proof of purchase must be supplied when requesting a repair.

TABLE OF CONTENTS

Table of Figures

Table of Tables

1

1. Introduction

The PCI14 series are 32-bit PCI bus architecture data acquisition boards. They are available in two basic models, the B and C. The PCI14 contains digital input and output ports and onboard counters. The PCI14 is a multi-purpose digital board that can be used in many applications. It also contains features for digital input and output protection. The counter modes are compatible with the Intel 8254.

Features

The PCI14 does have some very unique features and are short listed below:

- 32-bit PCI bus Revision 2.2 compliant at 33MHz.
- PCI Bus 3.3V compatible.
- 6 counters for B version and 2 for C version.
- Counter modes are compatible with Intel 8254.
- Fully configurable counter sub-system; includes internal clock source (40MHz) or external, internal or external gates.
- 24 x digital input lines and 24 x digital output lines.
- 50 mA sink, 20 mA source digital outputs with short circuit protection. High impedance on power up or loss.
- Overvoltage protection on all digital inputs.
- Overvoltage protection on digital outputs (B version only).
- Fully programmable interrupt system: interrupts generated from counters or 8 x digital inputs.

Feature	PCI14B	PCI14C
Number of digital input channels	24	24
Number of digital output channels	24	24
Number of 24-bit counters	6	2
High voltage digital input protection	Yes	Yes
High voltage digital output protection	Yes	No

Table 1-1 PCI14 Versions

Applications

The PCI14 can be used in the following applications:

- Digital control applications.
- Digital monitoring applications.
- Frequency measurement.
- Pulse generation.
- Timing applications.
- Pulse counting.

Key Specifications

- 24-bit digital input port.
- 24-bit digital output port.
- 24-bit counters.

Software Support

The PCI14 is supported by EDR Enhanced and comes with an extensive range of examples. The software will help you to get your hardware going very quickly. It also makes it easy to develop complicated control applications quickly. All operating system drivers, utility and test software are supplied on the EDR Enhanced CD-Rom. The latest drivers can also be downloaded from the Eagle Technology website. For further support information see the Contact Details section.

Contact Details

Below are the contact details of Eagle Technology.

Eagle Technology
PO Box 4376
Cape Town
8000
South Africa
Telephone +27 (021) 423 4943
Fax +27 (021) 424 4637
E-Mail eagle@eagle.co.za
Website http://www.eagle.co.za

2

2. Installation

This chapter describes how to install and configure the PCI14 for the first time. Minimal configuration is necessary; almost all settings are done through software. The PCI BIOS or operating system will take care of all resource assignments.

Package

PCI14 package will contain the following:

- PCI14 PCI board
- EDR Enhanced Software Development Kit CD-Rom

Hardware Installation

This section will describe how to install your PCI14 into your computer.

- Switch off the computer and disconnect from power socket.

Failure to disconnect all power cables can result in hazardous conditions, as there may be dangerous voltage levels present in externally connected cables.

- Remove the cover of the PC.
- Choose any open PCI slot and insert PCI14 board
- Insert bracket screw and ensure that the board sits firmly in the PCI socket.
- Replace the cover of the PC.
- Reconnect all power cables and switch the power on.
- The hardware installation is now completed.

Software Installation

Windows 98/2000/ME

Installing the Windows 98/2000 device driver is a very straightforward task. Because it is plug and play Windows will auto detect the PCI14 as soon as it is installed. No setup is necessary. You simply have to supply Windows with a device driver.

Wait until Windows detects the new hardware

Figure 2-1 Add New Hardware Wizard Step1

Select Next

Figure 2-2 Add New Hardware Wizard Step2

Select default option, search for best driver and select next

Figure 2-3 Add New Hardware Wizard Step3

Select *specify a location* and enter the directory location of the driver on the Eagle CD Rom. The driver should be located in the *<CDROM>:\EDRE\Drivers\WDM\PCI14B* directory.
Select Next to proceed

Figure 2-4 Add New Hardware Wizard Step4

Windows should have detected the proper driver and ready to install it. Select Next to proceed.

Figure 2-5 Add New Hardware Wizard Step5

Click on the finish button to complete the installation.
Click Yes to restart your computer.

Figure 2-6 Restart Your Computer

Post installation

After your installation was complete there is a few steps that can be followed to check that your installation was successful.

- First make sure that the driver is working properly by opening the *system folder* in the control panel.
- Check under the Eagle Data Acquisition list if your board is listed and working properly. See picture below.

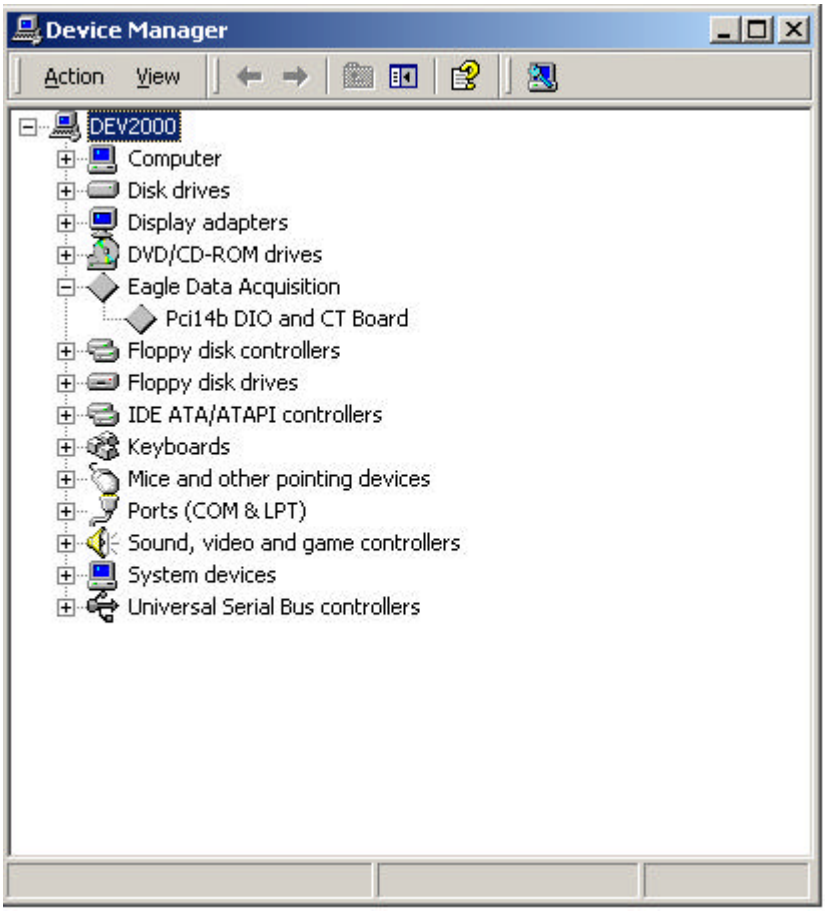


Figure 2-7 System Properties

- Clearly you can see that the PCI14 is listed and working properly.

- Further open the control panel and then the *EagleDAQ* folder. This dialog should list all installed hardware. Verify your board's properties on this dialog. See picture below

Figure 2-8 EagleDAQ

Now the first part of your installation has been completed and ready to install the EDR Enhanced Software Development Kit.

- Run ***setup.exe*** found on the EDR Enhanced SDK CD-Rom and follow the on screen instructions

Windows NT

Windows NT does not require any special setup procedure. The Windows NT driver does not support plug and play. If Windows 2000 detects a new device simply install a default driver, or so called placeholder.

To install the Windows NT drivers simply run ***EDREWinnt.exe*** on the Eagle CD-Rom. This will automatically install the device drivers. Restart your computer when done. Open the *EagleDAQ* folder in the control panel to check if your installation was successful. Figure 2-8 shows a successful installation.

Testing your board

To test your board, run the application supplied on the Eagle CD-Rom. Install the following program <*EAGLECD*>:*EDRE\APPS\PCI14B\pci14b.exe.* Run it to test all features on your board.

Accessories

The PCI14 does have a wide variety of accessories that it can be connected too. See the Eagle Technology catalog for more information.

3

3. Interconnections

The PCI14 has got one external connector that includes both digital I/O and counter signals. All connections are made through this connector situated on the card's bracket.

External Connectors

The PCI14 has a SCSI female centronics 68-way connector. As part of the accessories a cable and adapter can be ordered.

Pin Assignments

The table below shows the pin assignments for the PCI14.

Pin	Name	Pin	Name
1	+5V	35	+5V
2	DI2	36	DI1
3	DI4	37	DI3
4	DI6	38	DI5
5	DI8	39	DI7
6	DI10	40	DI9
7	DI12	41	DI11
8	Di14	42	DI13
9	DI16	43	DI15
10	DI18	44	DI17
11	DI20	45	DI19
12	DI22	46	DI21
13	DI24	47	DI23
14	DGND	48	DGND
15	CNTR0_CLK	49	CNTR0_GATE
16	CNTR1_CLK	50	CNTR1_GATE
17	CNTR2_CLK	51	CNTR2_GATE
18	CNTR3_CLK	52	CNTR3_GATE
19	CNTR4_CLK	53	CNTR4_GATE
20	CNTR5_CLK	54	CNTR5_GATE
21	DGND	55	DGND
22	DO1/CNTR0_TC	56	DO2/CNTR1_TC

23	DO3/CNTR2_TC	57	DO4/CNTR3_TC
24	DO5/CNTR4_TC	58	DO6/CNTR5_TC
25	DO7	59	DO8
26	DO9	60	DO10
27	DO11	61	DO12
28	DO13	62	DO14
29	DO15	63	DO16
30	DO17	64	DO18
31	DO19	65	DO20
32	DO21	66	DO22
33	DO23	67	DO24
34	DGND	68	DGND

Table 3-1 External Connector – SCSI 68F CENT

Signal Definitions

This sections deal with all the signals abbreviations.

Signal	Description
DI1-24	Digital Inputs
DO1-24	Digital outputs
CNTRx_CLK	External clock lines
CNTRx_GATE	External gate lines
+5V	+5V power supply line
DGND	Digital ground

Table 3-2 Signal definitions

Digital Inputs

The PCI14 has got 24 digital input lines. See specification for input protection. Please note when programming the PCI14 the 24 lines are split into 3 ports of 8-bits. This is to be compatible with the ISA PC14B.

Port	Lines
0	1-8
1	9-16
2	17-24

Table 3-2 Input ports

Digital Outputs

The PCI14 has got 24 digital output lines. See specification for output protection. Please note when programming the PCI14 the 24 lines are split into 3 port of 8-bits. This is to be compatible with the ISA PC14B. The outputs can also be configured to be high impedance. Lines 1-6 are the same as the 6 terminal count lines of the counters and must be configured for either.

Port	Lines
0	1-8
1	9-16
2	17-24

Table 3-3 Output ports

Counters

There are 6 counter or 2 counters on the PCI14 depending on the version. The counter modes are fully compatible with the Intel 8254 and can be programmed for internal clock or gate or external clock or gate or a combination of clock and gates. All external clock and gate lines are accessed via the external connector. The terminal count lines are shared with the first 6 digital output lines and have to be configured for either. It has an internal clock of 40MHz.

4

4. Functional Overview

This section provides a functional overview of the PCI14B board.

Device Address Map

The table below defines the address map as per offset of base address 0.

Offset Address (HEX)	Register Name	Description	Read/Write
000	CNT0_MODE	Mode Register	W
004	CNT0_COUNT	Initial count / Current count	W/R
008	CNT0_CONFIG	Configuration	W
010	CNT1_MODE	Mode Register	W
014	CNT1_COUNT	Initial count / Current count	W/R
018	CNT1_CONFIG	Configuration	W
020	CNT2_MODE	Mode Register	W
024	CNT2_COUNT	Initial count / Current count	W/R
028	CNT2_CONFIG	Configuration	W
030	CNT3_MODE	Mode Register	W
034	CNT3_COUNT	Initial count / Current count	W/R
038	CNT3_CONFIG	Configuration	W
040	CNT4_MODE	Mode Register	W
044	CNT4_COUNT	Initial count / Current count	W/R
048	CNT4_CONFIG	Configuration	W
050	CNT5_MODE	Mode Register	W
054	CNT5_COUNT	Initial count / Current count	W/R
058	CNT5_CONFIG	Configuration	W
080	INT_MASK	Interrupt mask	W
084	INT_CONTROL	Interrupt control	W
088	INT_STATUS	Interrupt status / Reset	W/R
08C	OUTPUT_CNTRL	Setup output sources	W
090	OUTPUT_REG	Setting output lines	W
0A0	INPUT_REG	Reading input lines	R
0B0	EEPROM_REG	EEPROM Interface	W/R

Table 4-1 Device Address Map

Counter Mode Register [2..0]

This is a 3-bit register defining the mode of a specific counter. If written to the Terminal Count(TC) will be initialized.

Bits[0..2] [2 | 1 | 0]	Description
[0 : 0 : 0]	Mode 0
[0 : 0 : 1]	Mode 1
[0 : 1 : 0]	Mode 2
[0 : 1 : 1]	Mode 3
[1 : 0 : 0]	Mode 4
[1 : 0 : 1]	Mode 5

Table 4-2 Counter Mode Register

Counter Count Register [23..0]

By writing to the *count* register the initial count of the counter is determined. The value will be loaded and when clocked the counter will start to count down. By reading the *count* register the current count value is returned. This register is 24-bits wide.

Counter Configuration Register [2..0]

The counter configuration register is used to setup the source clock and gate for each counter. Writing to this register always forces the Terminal Count (TC) line to initialize.

Bit[0..2]	Description
Bit 0	Specifies the Clock Source 0 : Clock source is internal – 40 MHz 1 : Clock source is external
Bit [2:1]	00 : Gate is disabled 01 : Gate is enabled 10 : Gate is external

Table 4-3 Counter Configuration Register

Interrupt Mask Register [13..0]

The interrupt mask register is used to enable or disable all 14 interrupt sources. Eight interrupt are derived from the first 8 digital input lines and 6 from the counters. A logic 1 mask an interrupt and a logic 0 enables an interrupt.

Bits [0..13]	Description
Bit 0	Input line 0
Bit 1	Input line 1
Bit 2	Input line 2
Bit 3	Input line 3
Bit 4	Input line 4
Bit 5	Input line 5

Bit 6	Input line 6
Bit 7	Input line 7
Bit 8	Counter terminal count 0
Bit 9	Counter terminal count 1
Bit 10	Counter terminal count 2
Bit 11	Counter terminal count 3
Bit 12	Counter terminal count 4
Bit 13	Counter terminal count 5

Table 4-4 Interrupt Mask Register

Interrupt Control Register [27..0]

The interrupt control register is used to setup each interrupt source. Each source can be programmed for edge/level or polarity trigger.

Bits [0..27]	Description
Bit [1:0]	Input 0: 0 : 0 Rising Edge 0 : 1 Falling Edge 1 : 0 Level '1' 0 : 0 Level '0'
Bit [3:2]	Input 1
Bit [5:4]	Input 2
Bit [7:6]	Input 3
Bit [9:8]	Input 4
Bit [11:10]	Input 5
Bit [13:12]	Input 6
Bit [15:14]	Input 7
Bit [17:16]	Counter 0 TC
Bit [19:18]	Counter 1 TC
Bit [21:20]	Counter 2 TC
Bit [23:22]	Counter 3 TC
Bit [25:24]	Counter 4 TC
Bit [27:26]	Counter 5 TC

Table 4-5 Interrupt Control Register

Interrupt Status Register [13..0]

The *Interrupt Status Register* supply information on current triggered interrupts. The register is 14-bit wide for each interrupt source. If set

Counter Mode 0 – Interrupt on terminal count.

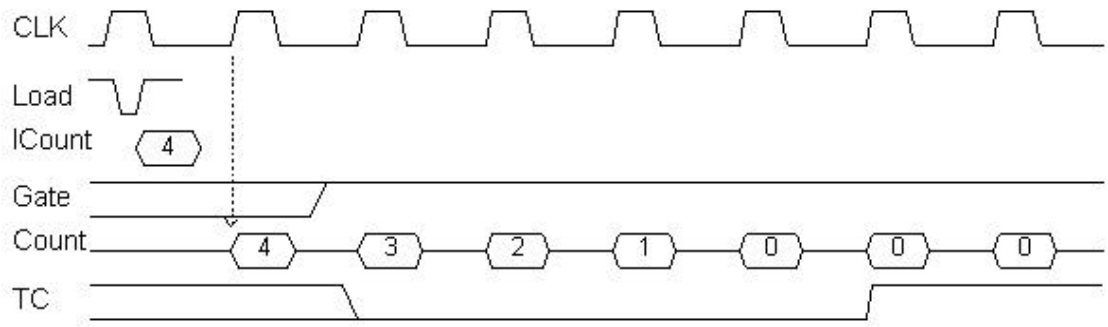


Figure 4-1 Counter Mode 0

TC will be initialized to '0' two clocks after the counter is loaded with an initial count. TC will toggle to a '1' one clock after the counter has reached zero. This state is maintained until the initial count is re-loaded.

Gate ='1' enabled counting. Gate = '0' disable counting.
Only the rising edge of Gate is significant.

Counter Mode 1 – Hardware re-triggerable one shot.

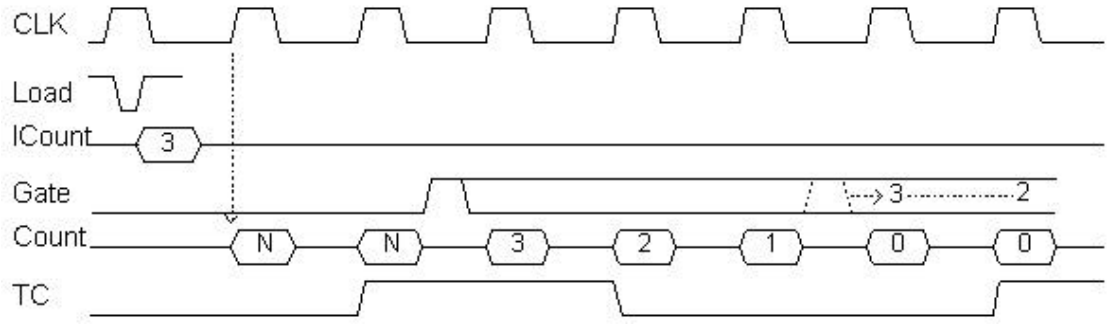


Figure 4-2 Counter Mode 1

The rising edge of GATE reloads the counter with the initial count, initializes TC to '1' and starts the down-count sequence. Further rising edges of GATE will re-initialize this state.

TC will toggle to '0' whenever the counter reaches the count of two. This state is maintained until a new GATE trigger event is detected.

Counter Mode 2 – Rate Generator.

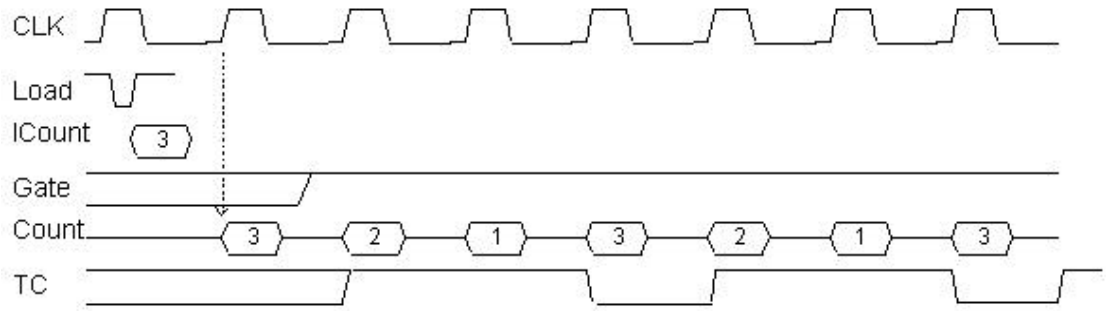


Figure 4-3 Counter Mode 2

TC will be initialized to '1' two clocks after the counter is loaded with an initial count and the counter enabled to count down (GATE ='1'). On the clock following the count of ONE, the counter will be reloaded with the initial count and TC toggled to a '0'. TC will toggle back to a '1' on the next clock. This sequence is maintained.

Gate = '1' enables counting. Gate = '0' disable counting.

Counter Mode 3 – Square Wave Generator.

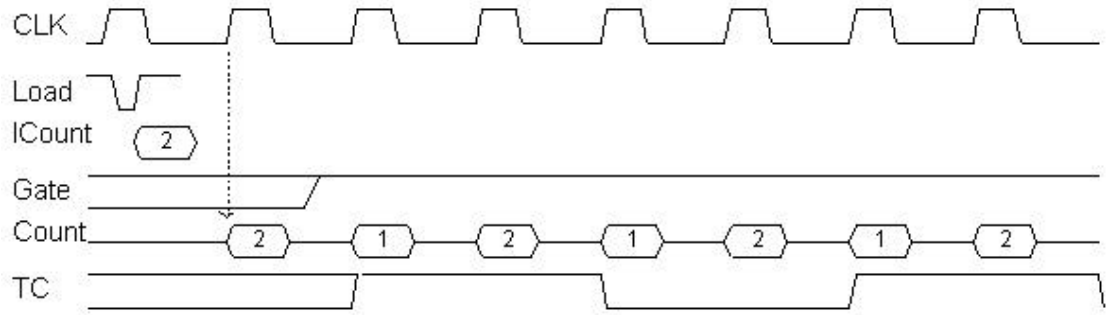


Figure 4-4 Counter Mode 3

This is the similar to mode 2 except that TC has a 50% duty cycle signal.

Counter Mode 4 – Software Triggered Strobe.

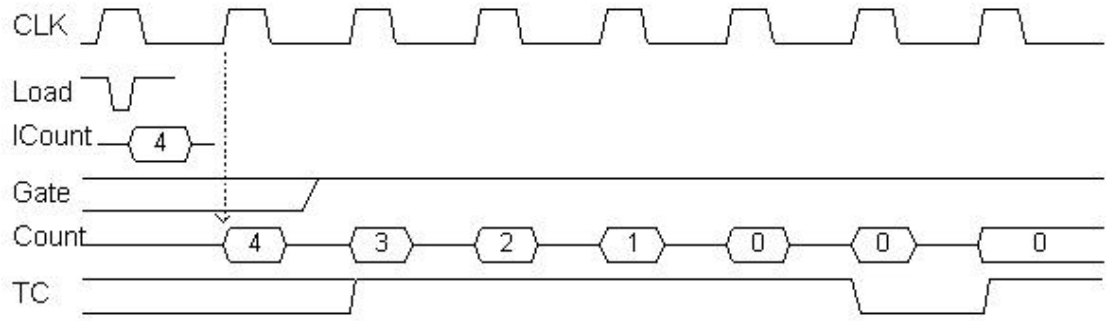


Figure 4-5 Counter Mode 4

TC and the counter are initialized whenever the software writes the initial count register. TC will pulse low for one clock period after the counter has reached zero. Note that is sequence is re-triggerable if the initial count is written before the terminal count condition is reached.

Gate = '1' enables counting. Gate = '0' disables counting.

Counter Mode 5 – Hardware Triggered Strobe

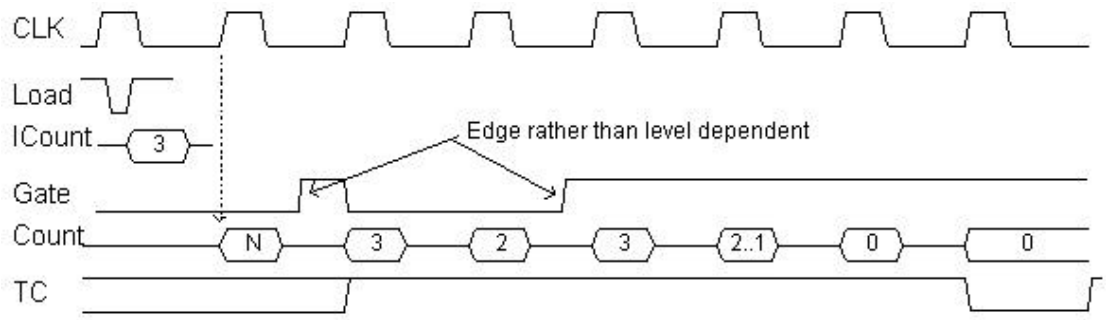


Figure 4-6 Counter Mode 5

Similar to mode four with the exception that the rising edge of GATE is used to re-trigger the initial counter and TC conditions.

Minimum Initial Counter Limits

The minimum initial counter values are mode dependent and defined as in the Table below.

Mode	Minimum initial count value
0	1
1	1
2	2
3	2
4	1
5	1

Table 4-6 Mode and initial count limits

5

5. Programming Guide

The PCI14B/C is supplied with a complete software development kit. EDR Enhanced (EDRE SDK) comes with drivers for many operating systems and a common application program interface (API). The API also serves as a hardware abstraction layer (HAL) between the control application and the hardware. The EDRE API makes it possible to write an application that can be used on all hardware with common sub-systems.

The PCI14B/C can also be programmed at register level, but it is not recommended. A detailed knowledge of the PCI14B/C is needed and some knowledge about programming Plug and Play PCI devices. We recommend that you only make use of the software provided by Eagle Technology.

EDR Enhanced API

The EDR Enhanced SDK comes with both ActiveX controls and a Windows DLL API. Examples are provided in many different languages and serve as tutorials. EDRE is also supplied with a software manual and user's guide.

The EDRE API hides the complexity of the hardware and makes it really easy to program the PCI14. It has got functions for each basic sub-system and is real easy to learn.

Examples

Please refer to the examples directory found in the EDRE folder of the CD-Rom. There are some complete PCI14B examples.

Digital Inputs

Digital inputs are used to read or monitor devices with a digital output capability. The PCI14B/C has got 24 inputs and depending on which version extensive or minimal input protection. The 24 inputs lines are split into 3 x 8-bit ports and assigned in the following configuration; lines 0-7 represents port 0, lines 8-15 represents port 1, lines 16-23 represents port 2.

Reading the Digital Inputs

A single call is necessary to read a digital I/O port.

API-CALL

Long EDRE_DioRead(ulng Sn, ulng Port, ulng *Value)

The serial number, port, and a pointer to variable to hold the result must be passed by the calling function. A return code will indicate if any errors occurred.

ACTIVEX CALL

Long EDREDioX.Read(long Port)

Only the port-number needs to be passed and the returned value will either hold an error or the value read. If the value is negative an error did occur.

Digital Outputs

Digital outputs are used to control devices with digital inputs. The PCI14B/C has got 24 inputs and depending on which version extensive or minimal output protection. The 24 inputs lines are split into 3 x 8-bit ports and assigned in the following configuration; lines 0-7 represents port 0, lines 8-15 represents port 1, lines 16-23 represents port 2.

Writing to the Digital Outputs

A single call is necessary to write to a digital I/O port.

API-CALL

Long EDRE_DioWrite(ulng Sn, ulng Port, ulng Value)

The serial number, port, and a value must be passed by the calling function. A return code will indicate if any errors occurred.

ACTIVEX CALL

Long EDREDioX.Write(long Port, ulng Value)

The port number and value to be written needs to be passed and the returned value holds an error or the value read. If the value is negative an error did occur.

Counters

The counter sub-system is supported by functions to Read, Write, Configure and controlling the gate.

Writing the initial counter value

A single call is necessary to write a counter's initial load value.

API-CALL

Long EDRE_CTWrite(ulng Sn, ulng Ct, ulng Value)

The serial number, counter-number, and a value must be passed by the calling function. A return code will indicate if any errors occurred.

ACTIVEX CALL

Long EDRECTX.Write(long Port, ulng Value)

The port number and value to be written needs to be passed and the returned value holds an error or the value read. If the value is negative an error did occur.

Reading a Counter

A single call is necessary to read a counter's current value.

API-CALL

Long EDRE_CTRead(ulng Sn, ulng Ct, ulng *Value)

The serial number, counter-number, and a pointer to variable to hold the result must be passed by the calling function. A return code will indicate if any errors occurred.

ACTIVEX CALL

Long EDRECTX.Read(long Ct)

Only the counter-number needs to be passed and the returned value will either hold an error or the value read. If the value is negative an error did occur.

The counter-number and value to be written needs to be passed and the returned value holds an error or the value read. If the value is negative an error did occur.

Configuring a counter

A single call is necessary to configure a counter.

API-CALL

Long EDRE_CTConfig(ulng Sn, ulng Ct, ulng Mode, ulng Type, ulng ClkSrc, ulng GateSrc)

The serial number, counter-number, mode, type (this parameter is ignored), clock source and gate source is needed to specify a counter's configuration. A return code will indicate if any errors occurred.

ACTIVEX CALL

Long EDRECTX.Configure(long ct, long mode, long type, ulng source, ulng gate)

The counter-number, mode, type (this parameter is ignored), clock source and gate source is needed to specify a counter's configuration. A return code will indicate if any errors occurred.

Controlling the counter gate

A single call is necessary to setup/control a counter's gate

API-CALL

Long EDRE_CTSoftGate(ulng Sn, ulng Ct, ulng Gate)

The serial number, counter-number and gate is needed to control a counter's gate. A return code will indicate if any errors occurred.

ACTIVEX CALL

Long EDRECTX.SoftGate(ulng Ct, ulng Gate)

The counter-number and mode is needed to control a counter's gate. A return code will indicate if any errors occurred.

Configuration Constants

These values are acceptable as a clock source.

Value	Description
0	Internal 40Mhz
1	External

Table 5-1 Clock Configuration

These values are acceptable as a gate source.

Value	Description
0	Gate disabled
1	Gate enabled
2	Gate external

Table 5-2 Gate Configuration

Multi Function I/O

The PCI14 has got some multi I/O lines. These are I/O lines on the connector that is shared by the counter and digital outputs. The output lines can also be configured to be high impedance or enabled. The multi I/O configure function is used to set this up.

Configuring the outputs

A single call is necessary to configure the output lines of the PCI14.

API-CALL

Long EDRE_MioConfig(ulng Sn, ulng Port, ulng Value)

The serial number, Port/Line number, and a value must be passed by the calling function. A return code will indicate if any errors occurred.

ACTIVEX CALL

Long EDREDioX.MioConfigure(long Port, long Value)

The port-number and value to be written needs to be passed and the returned value holds an error or the value read. If the value is negative an error did occur. Please use the table below as reference.

Port	Value
0	0 : Output 0 1 : Counter 0 TC
1	0 : Output 1 1 : Counter 1 TC
2	0 : Output 2

	1 : Counter 2 TC
3	0 : Output 3
	1 : Counter 3 TC
4	0 : Output 4
	1 : Counter 4 TC
5	0 : Output 5
	1 : Counter 6 TC
6	0 : Outputs are high impedance
	1 : Outputs are active

Table 5-3 Multiple I/O Configuration

Programming Interrupts

The PCI14B can generate interrupts from 14 different sources, which include digital inputs and counters. The interrupt sub-system is totally programmable and user-space interrupt-service-routines are triggered through events from the driver. If an event is triggered it also needs to be cleared. The EDRE interrupt system includes functions to configure, enable, disable. The configuration function is used to configure a specific interrupt source and Enable and Disable to globally control interrupts.

WARNING!

Be careful when programming the interrupt sub-system because it is easy to generate interrupts that is faster than what Windows can service. Don't try and generate interrupt faster than 10KHz. This will not work. Remember this is 10KHz in total, and not per source. The PCI14B interrupt service routine will stop servicing interrupts if at any stage it is still busy with an interrupt and the next one is generated.

Configuring the Interrupt sub-system

A single call is necessary to configure the interrupt sub-system.

API-CALL

Long EDRE_IntConfigure(ulng Sn, ulng Src, ulng Mode, ulng Type)

Parameter	Type	Description
Sn	Unsigned long	Board's Serial Number
Src	Unsigned long	Interrupt Source

No	Description
0	DIO Line 0
1	DIO Line 1
2	DIO Line 2
3	DIO Line 3
4	DIO Line 4
5	DIO Line 5
6	DIO Line 6
7	DIO Line 7
8	TC Counter 0
9	TC Counter 1
10	TC Counter 2

		10 TC Counter 2 11 TC Counter 3 12 TC Counter 4 13 TC Counter 5
Mode	Unsigned long	Disable or Enable a source 0 : Disable 1 : Enable
Type	Unsigned long	Set the type of trigger for the interrupt **No** **Description** 0 Rising Edge 1 Falling Edge 2 Level = '1' 3 Level = '0'
RETURN	Long	This parameter contains the error code return. If =0 then no error occurred.

Table 5-4 EDRE_IntConfigure Parameters

ACTIVEX CALL

Long EDREIntX.Configure(long Source, long Mode, long Type)

As the DLL call the ActiveX function takes the same parameters except the serial number.

Enabling Interrupts

A single call is necessary to enable the interrupt sub-system. This will also enable the global interrupt on the PCI14B and connect it to the PCI Bus.

API-CALL

Long EDRE_IntEnable(ulng Sn)

Only a serial number needs to be specified. The return code will contain the status of the call.

ACTIVEX-CALL

Long EDREIntX.Enable

A returned error code will contain the status of the call.

Disabling Interrupts

A single call is necessary to disable the interrupt sub-system.

API-CALL

Long EDRE_IntDisable(ulng Sn)

Only a serial number needs to be specified. The return code will contain the status of the call.

ACTIVEX-CALL

Long EDREIntX.Disable

A returned error code will contain the status of the call.

Programming interrupts by using the EDREIntX OCX

The interrupt OCX makes programming interrupts very easy. You only have to configure the interrupts, Enable/Disable it and implement a trigger. The trigger will also supply you with the source of the interrupt. By simply clicking on the control your programming language will jump to the code for you trigger procedure.

Visual Basic Interrupt Example

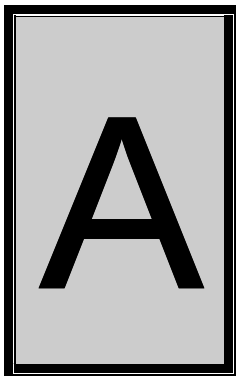
```
Option Explicit
Dim err As Long

Private Sub Command1_Click()
   EDREIntX1.Enable
End Sub

Private Sub Command2_Click()
   EDREIntX1.Disable
End Sub

Private Sub EDREIntX1_Interrupt(ByVal Source As Long)
'this is the interrupt service routine
End Sub

Private Sub Form_Load()
   sn = EDREUtlX1.SelectDialog()              'get a board's serial number
   EDREIntX1.SerialNumber = sn                'set the interrupt OCX serial number
   err = EDREIntX1.Configure(0, 1, 0)         'configure interrupt system
                                    'source = 0, mode = enable, type = 0
End Sub
```

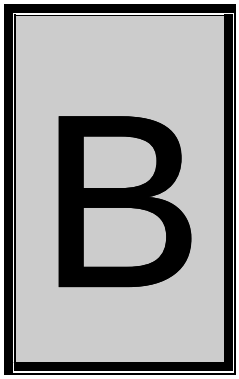


A. I/O Electrical Specifications

SPECIFICATION	PCI-14B	PCI-14C
Digital Inputs		
Minimum input voltage (Relative to DGND) to register a logic '1'	+ 2.5 V	+ 2.5 V
Maximum input voltage (Relative to DGND) to register a logic '0'	+ 1.0 V	+ 1.0 V
Maximum Input Frequency	8 MHz	8 MHz
Maximum Continuous Input Voltage (Relative to DGND)	± 32 V	± 32 V
Maximum Input Voltage Spike < 100 ms (Relative to DGND)	± 80 V	± 80 V
Counter Clock and Gate Control Inputs		
Minimum input voltage (Relative to DGND) to register a logic '1'	+ 2.5 V	+ 2.5V
Maximum input voltage (Relative to DGND) to register a logic '0'	+ 1.0 V	+ 1.0 V
Maximum Input Frequency	8 MHz	8 MHz
Maximum Continuous Input Voltage (Relative to DGND)	± 32 V	± 32 V
Maximum Input Voltage Spike < 50 ms (Relative to DGND)	± 80 V	± 80 V
Digital Outputs & Counter Timer Outputs		
Number of Digital Outputs (Multiplexed with counter TC outputs.)	24	24
Sink current (Output voltage =< 0.5V)	- 48 mA	- 48 mA
Source current (Output voltage >= 2.5V)	+ 18 mA	+ 18 mA
Maximum Output Frequency	8 MHz	8 MHz
Short Circuit protection	- 40 to – 140 mA	-40 to –140 mA
Maximum Continuous Over Voltage Stress (Note 1)	+ 32 V	+ 5.5 V
Maximum Continuous Under Voltage Stress (Note 1)	- 32 V	- 0.8 V
Maximum Output Voltage Spike < 50 ms (Relative to DGND)	± 80 V	-

Table A-1 I/O Electrical Specifications

Note 1. For Option B, the outputs are protected by 100 mA ± 20 mA (at 25°C) polyfuses and 5.6V transorbs. Over or under voltage stress levels longer than 50 ms will activate the fuse protecting the driver. Note also that once activated such fuses have a fairly long recovery time.



B. Configuration Constants

Query Codes

Name	Value	Description
APIMAJOR	1	Query EDRE API major version number.
APIMINOR	2	Query EDRE API minor version number.
APIBUILD	3	Query EDRE API build version number.
APIOS	4	Query EDRE API OS type.
APINUMDEV	5	Query number of devices installed.
BRDTYPE	10	Query a board's type.
BRDREV	11	Query a board's revision.
BRDYEAR	12	Query a board's manufactured year.
BRDMONTH	13	Query a board's manufactured month.
BRDDAY	14	Query a board's manufactured day.
BRDSERIALNO	15	Query a board's serial number.
DRVMAJOR	20	Query a driver's major version number.
DRVMINOR	21	Query a driver's minor version number.
DRVBUILD	22	Query a driver's build version number.
ADNUMCHAN	100	Query number of ADC channel.
ADNUMSH	101	Query number of samples-and-hold channels.
ADMAXFREQ	102	Query maximum sampling frequency.
ADBUSY	103	Check if ADC system is busy.
ADFIFOSIZE	104	Get ADC hardware FIFO size.
ADFIFOOVER	105	Check for FIFO overrun condition.
ADBUFFSIZE	106	Check software buffer size.
ADBUFFOVER	107	Check for circular buffer overrun.
ADBUFFALLOC	108	Check if software buffer is allocated.
ADUNREAD	109	Get number of samples available.
ADEXTCLK	110	Get status of external clock line – PCI30FG.
ADEXTTRIG	111	Get status of external trigger line – PCI30FG.
ADBURST	112	Check if burst mode is enabled.
ADRANGE	113	Get ADC range.
DANUMCHAN	200	Query number of DAC channels.
DAMAXFREQ	201	Query maximum DAC output frequency.
DABUSY	202	Check if DAC system is busy.
DAFIFOSZ	203	Get DAC FIFO size.
CTNUM	300	Query number of counter-timer channels.
CTBUSY	301	Check if counter-timer system is busy.
DIONUMPORT	400	Query number of digital I/O ports.
DIOQRYPORT	401	Query a specific port for capabilities.
DIOPORTWIDTH	402	Get a specific port's width.
INTNUMSRC	500	Query number of interrupts sources.
INTSTATUS	501	Queries interrupt system's status.
INTBUSCONNECT	502	Connect interrupt system to bus.
INTISAVAILABLE	503	Check if an interrupt is available.
INTNUMTRIG	504	Check number times interrupted

Table B-1 Query Code

Error Codes

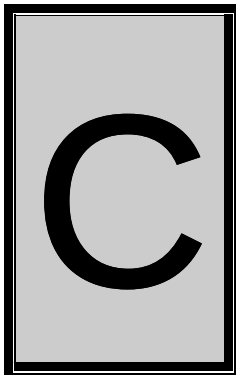
Name	Value	Description
EDRE_OK	0	Function successfully.
EDRE_FAIL	-1	Function call failed.
EDRE_BAD_FN	-2	Invalid function call.
EDRE_BAD_SN	-3	Invalid serial number.
EDRE_BAD_DEVICE	-4	Invalid device.
EDRE_BAD_OS	-5	Function not supported by operating system.
EDRE_EVENT_FAILED	-6	Wait on event failed.
EDRE_EVENT_TIMEOUT	-7	Event timed out.
EDRE_INT_SET	-8	Interrupt in use.
EDRE_DA_BAD_RANGE	-9	DAC value out of range.
EDRE_AD_BAD_CHANLIST	-10	Channel list size out of range.
EDRE_BAD_FREQUECY	-11	Frequency out of range.
EDRE_BAD_BUFFER_SIZE	-12	Data passed by buffer incorrectly sized
EDRE_BAD_PORT	-13	Port value out of range.
EDRE_BAD_PARAMETER	-14	Invalid parameter value specified.
EDRE_BUSY	-15	System busy.
EDRE_IO_FAIL	-16	IO call failed.
EDRE_BAD_ADGAIN	-17	ADC-gain out of range.
EDRE_BAD_QUERY	-18	Query value not supported.
EDRE_BAD_CHAN	-19	Channel number out of range.
EDRE_BAD_VALUE	-20	Configuration value specified out of range.
EDRE_BAD_CT	-21	Counter-timer channel out of range.
EDRE_BAD_CHANLIST	-22	Channel list invalid.
EDRE_BAD_CONFIG	-23	Configuration invalid.
EDRE_BAD_MODE	-24	Mode not valid.
EDRE_HW_ERROR	-25	Hardware error occurred.
EDRE_HW_BUSY	-26	Hardware busy.
EDRE_BAD_BUFFER	-27	Buffer invalid.
EDRE_REG_ERROR	-28	Registry error occurred.
EDRE_OUT_RES	-29	Out of resources.
EDRE_IO_PENDING	-30	Waiting on I/O completion

Table B-2 Error Codes

Digital I/O Codes

Name	Value	Description
DIOOUT	0	Port is an output.
DIOIN	1	Port is an input.
DIOINOROUT	2	Port can be configured as in or out.
DIOINANDOUT	3	Port is an input and an output.

Table B-3 Digital I/O Codes



C. Ordering Information

For ordering information please contact Eagle Technology directly or visit our website www.eagle.co.za. They can also be emailed at eagle@eagle.co.za.

Board	Description
PCI 14B	48 channel digital I/O and 6 counters board.
PCI 14C	48 channel digital I/O and 2 counters board.

Table C-1 Ordering Information