NuDAQ®
PCI-8133
3 Channels Encoder Counters and PWM Output Card
User's Guide

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

@Copyright 1997~2000 ADLINK Technology Inc.
All Rights Reserved.

Manual Rev. 1.10: September 7, 2000

The information in this document is subject to change without prior notice in order to improve reliability, design and function and does not represent a commitment on the part of the manufacturer.

In no event will the manufacturer be liable for direct, indirect, special, incidental, or consequential damages arising out of the use or inability to use the product or documentation, even if advised of the possibility of such damages.

This document contains proprietary information protected by copyright. All rights are reserved. No part of this manual may be reproduced by any mechanical, electronic, or other means in any form without prior written permission of the manufacturer.

Trademarks
NuDAQ is a registered trademark of ADLINK Technology Inc.,

Other product names mentioned herein are used for identification purposes only and may be trademarks and/or registered trademarks of their respective companies.

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Getting service from ADLINK

- **Customer Satisfaction is always the most important thing for ADLINK Tech Inc. If you need any help or service, please contact us and get it.**

ADLINK Technology Inc.			
Web Site	http://www.adlink.com.tw http://www.adlinktechnology.com		
Sales & Service	service@Adlink.com.tw		
Technical Support	NuDAQ	nudaq@adlink.com.tw	
	NuDAM	nudam@adlink.com.tw	
	NuIPC	nuipc@adlink.com.tw	
	NuPRO	nupro@adlink.com.tw	
	Software	sw@adlink.com.tw	
TEL	+886-2-82265877	FAX	+886-2-82265717
Address	9F, No. 166, Jian Yi Road, Chungho City, Taipei, 235 Taiwan, R.O.C.		

- **Please inform or FAX us of your detailed information for a prompt, satisfactory and constant service.**

Detailed Company Information			
Company/Organization			
Contact Person			
E-mail Address			
Address			
Country			
TEL		FAX	
Web Site			
Questions			
Product Model			
Environment to Use	☐OS: _____ ☐Computer Brand: _____ ☐M/B: ☐CPU: ☐Chipset: ☐Bios: ☐Video Card: ☐Network Interface Card: ☐Other:		
Challenge Description			
Suggestions for ADLINK			

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Table of Contents

How to Use This Manual iii

Chatper 1 Introduction 1

1.1 Features 1
1.2 Applications 2
1.3 Specifications 2
1.4 Software Supporting 3

Chatper 2 Installation 4

2.1 What You Have 4
2.2 Unpacking 4
2.3 Device Installation for Windows 95/98/NT 5
2.4 PCI-8133's Layout 6
2.5 PCI Configuration 7

Chatper 3 Signal Connections 8

3.1 Connectors Pin Assignment 8
3.3 Daughter Board Connection 10
3.3.1 Connect with ACLD-9137 10
3.3.2 Connect with ACLD-9188 10
3.4 UP/DOWN Counters 10
3.4.1. Differential Input & Isolation 11
3.4.2. 3 Stages Digital Filter 11
3.4.3. Quadrature Decoder 11
3.4.4. Position Counter and Data Latch 11
3.4.5. Special Counter Operation Mode 12
3.5. Programmable Interrupt Counter 12
3.6. Index Latch Register 13
3.7. PWM Signal Generator 14
3.8 Digital Inputs and Outputs 15
3.8.1 Isolation Digital Outputs 15
3.8.2 Isolation Digital Inputs 15
3.8.3 Extra 3 Isolation Digital Inputs 15
3.9. Interrupt Control 16
3.9.1 System Architecture 16
3.9.2 INT0 Timer 16
3.9.3 INT1 Timer 16
3.9.4. IRQ Level Setting 17

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

3.9.5 Dual Interrupt System 17

Chatper 4 Registers Structure 18

4.1 I/O Port Address 18
4.2 Counter Registers 19
4.3 Index Registers 19
4.4 Status Register 20
4.5 Digital Output Register 21
4.6 Control Mode Register 21
4.7 Interrupt #0 Period Register 22
4.8 Interrupt #1 Period Register 22
4.9 PWM Output Registers 23
4.10 Digital Input Register 23
4.11 Interrupt Clear Registers 24

Chatper 5 C/C++ Software Library 25

5.1 Installation 25
5.2 C/C++ Programming Library 27
5.3 _8133_Initial 29
5.4 _8133_Software_Reset 29
5.5 _8133_Read_Cnt 30
5.6 _8133_Read_Index 31
5.7 _8133_Read_Status 31
5.8 _8133_CLR_IdxLah 32
5.9 _8133_ModeSelecct 32
5.10 _8133_Set_Int0Perd 33
5.11 _8133_Set_Int1Perd 34
5.12 _8133_Set_PWMPerd 34
5.13 _8133_INT_Control 35
5.14 _8133_CLR_IRQ0 & _8133_CLR_IRQ1 36
5.15 _8133_Get_IRQ_Channel 36
5.16 _8133_Get_IRQ_Status 37
5.17 _8133_DO 37
5.18 _8133_DI 38
5.19 _8133_INT_Enable 39
5.20 _8133_INT_Disable 39

Appendix A 41

Product Warranty/Service 43

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

How to Use This Manual

This manual is designed to help you to use the PCI-8133. The manual describes the versatile functions and the operation theorem of the PCI-8133 card. It is divided into six chapters:

Chapter 1, "Introduction," gives an overview of the product features, applications, and specifications.

Chapter 2, "Installation," describes how to install the PCI-8133. The layout of PCI-8133 is shown, jumper setting for analog input channel configuration.

Chapter 3, "Signal Connection & Operation Theorem," describes the connectors' pin assignment and how to connect the outside signal and devices with the PCI-8133. The operation theorem describes the method that help the user to understand more about the functions of this cards.

Chapter 4, "Registers Structure" describes the details of register format. This information is very important for the programmers who want to control the hardware by low-level programming.

Chapter 5, "C/C++ Software Library" describes high-level programming interface in C/C++ language. It helps programmer to control PCI-8133 in high-level language style.

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

1

Introduction

The PCI-8133 is a 3-channel quadrature encoder counters card for 32-bit PCI PnP bus. This card is suitable for motor control and/or position monitoring for the optical-mechanical system. The outstanding features of PCI-8133 include three 16-bit quadruple AB phase encoder counters, three 12-bits PWM signal output and general purpose 8 isolation DI and 8 isolation DO.

The encoder counters is equipped with digital de-glitch filters for every encoder input signal and on-board 5000V rms isolation circuits. The multi-configurations of input signals can let user fit this to most of the motion control applications.

1.1 Features

The PCI-8133 PCI Bus Advanced Data Acquisition Card provides the following advanced features:

- 32-bit PCI Bus, Plug and Play
- Three quadruple AB phase encoder counters
- 16-bit up down counters
- Digital de-glitch filters for every encoder input signal
- Programmable frequency for digital deglitch filter
- On-board 5000V rms isolation circuits for encoder signals and digital I/O signals
- General purpose isolation 11 DI & 8 DO
- Three 12-bit PWM waveform generators
- Dual interrupt from two programmable timer clock signals
- Compact size–only half-size PCB
- One 37-pin rugged D-type connector for encoder signals,

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

- One 40-pin header connector for digital I/O expansion

1.2 Applications

- Motion control application
- Process monitoring
- Industrial process control

1.3 Specifications

¨ **Counter of Encoder Input**

- Number of channels: 3
- Counter resolution: 16-bit
- Encoder counter Types
  - Up-down counter
  - Digital filter for input signal
  - A, B phase and index input
- Counter 1 input signal types:
  - A, B phase decoder input
  - VCO (CCW+CW pulse) input
- Counter 2 input signal types:
  - A, B phase decoder input
  - Pulse command input
- Counter status read back
- Filter type: 3-order digital filter
- Digital filter frequency: programmable 10MHz, 5MHz, 2.5MHz, 625KHz

¨ **PWM Signal Output**

- Number of channels: 3
- Signal resolution: 12-bit
- Base frequency: 10MHz
- PWM cycle is synchronized with interrupt signal

¨ **Isolation Digital Output/Input**

- Number of input channels: 11 DI
- Number of output channels: 8 DO
- Input voltage: 0-24Vdc
  - Logic H: 3~24V
  - Logic L: 0~1.5V
- Input resistance: 1.2KΩ @ 0.5W

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

- Digital output type: Darlington Transistors, open collector up to 40Vdc
- Sink current: 345mA typical, 500mA maximum per channel
- Isolated voltage: 5000V rms

General Specification

- Connector: one 37-pin D-type female and one 40-pin header connector
- Interrupt sources: (Dual Interrupt)
  Internal Timer Clock 1
  Internal Timer Clock 2
- Operating temperature: 0° ~ 55°C
- Storage temperature: -20° ~ 80°C
- Humidity: 0~95%, non-condensing
- Dimensions: 162 mm x 105 mm
- Power Requirement: +5V @ 580mA(typical)

1.4 Software Supporting

ADLINK provides versatile software drivers and packages for users' different approach to built-up a system. We not only provide programming library such as DLL for many Windows systems. All the software options are included in the ADLINK CD.

Programming Library

For customers who are writing their own programs, we provide function libraries for many different operating systems, including:

- **DOS** Library: Borland C/C++, the functions descriptions are included in this user's guide.
- Windows 95/98/NT/98/NT DLL: For VB, VC++, Delphi, BC5, the functions descriptions are included in this user's guide.

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

2

Installation

This chapter describes how to install the PCI-8133. At first, the contents in the package and unpacking information that you should be careful are described.

2.1 What You Have

In addition to this *User's Manual*, the package includes the following items:

- PCI-8133 Enhanced Multi-function Data Acquisition Card
- ADLINK CD

If any of these items is missing or damaged, contact the dealer from whom you purchased the product. Save the shipping materials and carton in case you want to ship or store the product in the future.

2.2 Unpacking

Your PCI-8133 card contains sensitive electronic components that can be easily damaged by static electricity.

The card should be done on a grounded anti-static mat. The operator should be wearing an anti-static wristband, grounded at the same point as the anti-static mat.

Inspect the card module carton for obvious damage. Shipping and handling may cause damage to your module. Be sure there are no shipping and handing damages on the module before processing.

After opening the card module carton, extract the system module and place it only on a grounded anti-static surface component side up. Again inspect the module for damage. Do this only with the module place on a

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

firm flat surface.

Note: DO NOT APPLY POWER TO THE CARD IF IT HAS BEEN DAMAGED.

You are now ready to install your PCI-8133.

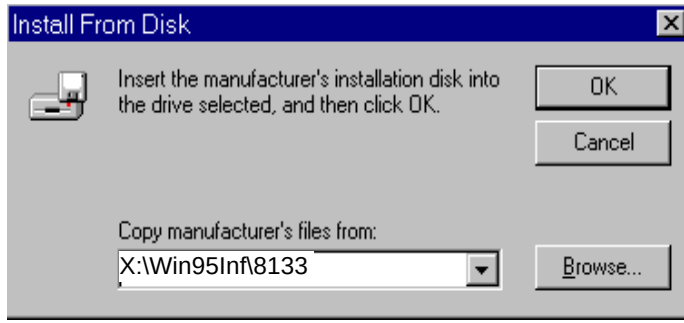
2.3 Device Installation for Windows 95/98

While you first plug PCI-8133 card and enter Windows 95/98, the system will detect this device automatically and show the following dialog box that prompts you to select the device information source.



Choose the default option "*Driver from disk provided by hardware manufacturer*" and then a dialog box is shown to prompt you give the path of installation disk.

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000



Place ADLINK CD into your CD-ROM drive. Browse the ***X:\Win95Inf\8133*** in the input field (X indicates the CD-ROM drive) then click ***OK***. This directory contains PCI-8133 device information file ***Pci8133.inf*** .The system will start the installation of PCI-8133 driver.

2.4 PCI-8133's Layout

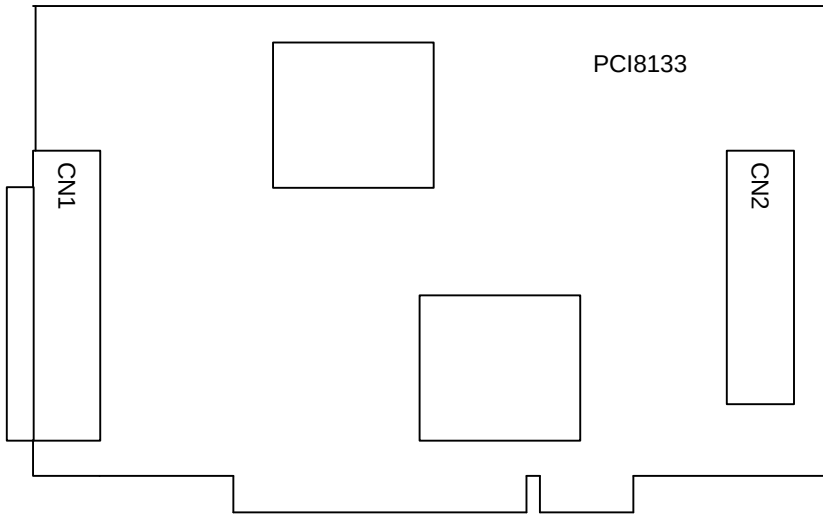


Figure 2.4 PCB Layout of the PCI-8133

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

2.5 PCI Configuration

1. Plug and Play:
As a plug and play component, the board requests an interrupt via a system call. The system BIOS assigns an interrupt level based on the board information and system parameters. The system parameters are assigned by the BIOS and installed drivers.

2. Configurations:
The board configuration is done on a board-by-board basis for all PCI cards on your system. Because the configuration is controlled by BIOS and software, there is no jumpers for setting system parameters like base-address, and interrupt level. The configuration is subject to change with every boot of the system when new boards are added or removed.

3. Trouble shooting:
If your system will not boot or if you experience erratic operation with your PCI board in place, it's likely caused by an interrupt conflict (perhaps because you incorrectly described the ISA setup). In general, the solution, once you determine it is not a simple oversight, is to consult the BIOS documentation that come with your system.

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

3

Signal Connections

This chapter describes the connector of PCI-8133, and the signal connection between the PCI-8133 and external devices, such as daughter boards or other devices. The operation of the functions on PCI-8133 card is also described in this chapter. It can help you to understand how to manipulate or to program the PCI-8133. The main features of PCI-8133 include 3 independent 16-bit up/down counters, 3 index latch registers, 2 programmable interrupt controllers, 3 complementary PWM signal generators, 8 isolated digital inputs and 8 isolated outputs.

3.1 Connectors Pin Assignment

The PCI-8133 comes equipped one 37-pin D-type connector - CN1. The pin assignment of CN1 is illustrated in the Figure 3.1.

- **CN 1: Encoder Input Signals & PWM Output**
- **CN 2: Isolation Digital Input & Output**

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

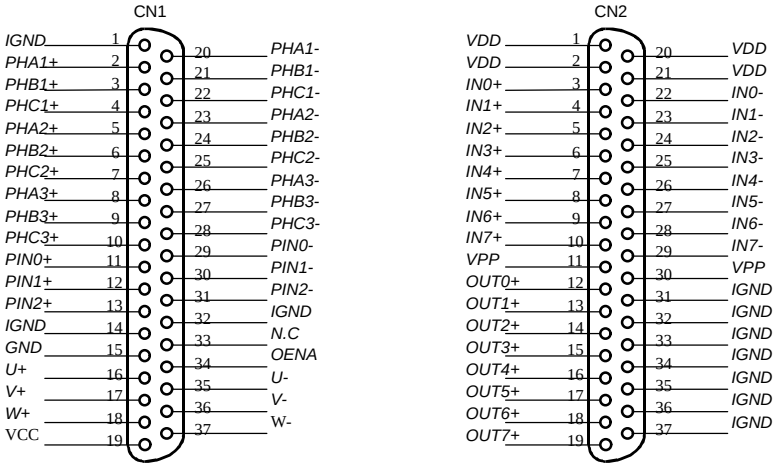


Figure 3.1 Pin Assignment of CN1 & CN2 (CN2 is converted from 40-pin header to 37-pin DSUB connector.)

Legend :

PHXn+	: Positive arm of differential photo encoder input
PHXn-	: Negative arm of differential photo encoder input [X=A,B,C, n=1~3]
PINk+	: General purpose differential D/I signals
PINk-	: General purpose differential D/I signals [k=0~2]
IGND	: Isolated Signal Ground
U+/-	: PWM Output, U channel positive / negative
V+/-	: PWM Output, V channel positive / negative
W+/-	: PWM Output, W channel positive / negative
GND	: Non-isolated ground for PWM signals
VDD	: Isolated power supply output (un-regulated 5V)
INm+/-	: Differential Digital Input CH-m positive / negative
OUTm	: Isolated Digital Output CH-m
VPP	: Fly wheel power line for Isolation Digital Output

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

3.3 Daughter Board Connection

The PCI-8133 can be connected with several different daughter boards, ACLD-9137 and ACLD-9188. The functionality and connections are specified as follows.

3.3.1 Connect with ACLD-9137

The ACLD-9137 is a direct connector for the card that is equipped with 37-pin D-sub connector. This board provides a simple way for connection. It is very suitable for the simple applications that do not need complex signal condition before the A/D conversion is performed.

3.3.2 Connect with ACLD-9188

ACLD-9188 is a general purpose terminal board for all the card which comes equipped with 37-pin D-sub connector. One ACL-9188, each signal is connected to a screw terminal.

3.4 UP/DOWN Counters

There are 3 independent 16-bit up/down counters, which are usually used for speed and position monitoring in motion control system. These counters, preceded by digital filter circuits, can count pulses from A/B phase encoders or other type of position sensors. The flowchart of signal processing is shown in Figure 5.1.

The counter 3 is dedicated for A/B phase signal input, however, the Counter 1 and Counter 2 have different operation modes. Note that the digital filter in the block diagram is only valid for A/B phase input mode.

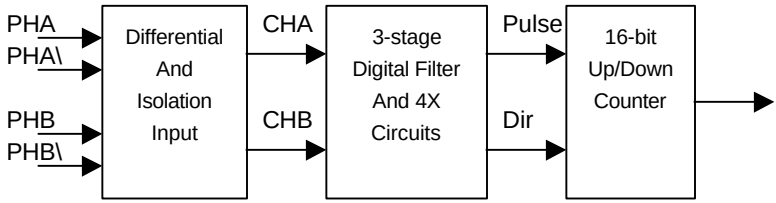


Fig 5.1 Encoder Interface Block Diagram

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

3.4.1. Differential Input & Isolation

The input signal of the encoder input is in differential type. The differential signals are transferred into single-ended by differential driver. The loading of each differential input signal is 220 Ohm. The single-ended signal is isolated from the host power and ground by photo-coupler whose isolation voltage is 5000V rms.

3.4.2. 3 Stages Digital Filter

The output signals from photo-couplers are passed to a 3-stage digital filter. This circuits can reject noise spikes that typically occur in motor system application. Utilizing the filters, the user can improve the accuracy of the system.

The signal on each channel is sampled on rising clock edge. A time history of the signal is stored in the four-bit shift register, any change on the input is tested for a stable level for three consecutive rising clock edges. Therefore, the filtered output waveform can change only after an input level has the same value for three consecutive rising clock edges. Thus filter out the in-coming noise.

The sampling period of the filters can be set by software. Two bits (FTS1, FTS0) are used to select the operation frequency. Please refer to section 4.6 for the definition of these bits.

3.4.3. Quadrature Decoder

The quadrature decoder decodes the incoming filtered signals into pulse for counting. The circuitry multiplies the position resolution of the input signals by a factor of four (4X decoding). When using an encoder for motor position sensing, the increased resolution can provide better system control precision. For example: for an A/B phase encoder with 2000 pulses per revolution, it can get the resolution of 8000 pulses per revolution.

The quadrature decoder samples the outputs of the CHA and CHB filters. Based on the past binary state of the two signals and the present state, it outputs a pulse signal and a direction signal to the internal position counter.

3.4.4. Position Counter and Data Latch

The 16-bit binary up/down counter which counts on rising clock edges. The 16 bits of data are passed to the position data latch after counter value changing. The counter value is from 0 to 65535. When the position (counter) range of a system exceeds 65535, the user should use an interrupt service routine (ISR) to carry out a position monitor task without

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

loss any position information. The ISR techniques will be explained in section 3.5.

The position data latch is a 16-bit latch which captures the position counter output data on each rising clock edge, except when its inputs are disabled by the inhibit logic section during data read operation. The output data is passed to local data bus. When active, a signal from the inhibit logic section prevents new data from being captured by the latch, keeping the data stable while read operation is made through the bus interface. The latch is automatically re-enabled at the end of read operation. The latch is cleared to 0 asynchronously by _8133_Software_Reset() function.

3.4.5. Special Counter Operation Mode

PCI-8133 card can also accept up/down pulse or pulse/direction type signal inputs depending on the sensors used. The Fig.3.4 shows these two types of pulse signals. Counter 1 can accept two types of signal input and Counter 2 can accept four types. The _8133_ModeSelect() function is used to set these modes. Bit 1 MS is for Counter 1 and (C2M1, C2M0) is for Counter 2. Please refer the Section 4.6 for the detail of setting.

3.5. Programmable Interrupt Counter

There are two programmable interrupt sources, INT0 and INT1, in PCI-8133 card. The interrupt signals can help user to calculate motor speed or/and monitor motor position.

User can get position information from encoder every time when an interrupt is generated, for example PLS(n) is the pulse we read at n-th sampling time. PLS (n-1) is the pulse of (n-1)-th sampling time. Then we can get speed information by minus these two values:

VEL(n)=K[PLS(n)-PLS(n-1)],

Where K represents the unit transform coefficient. Let

dPLS(n)=PLS(n)-PLS(n-1),

If we integrate this information at every ISR, then we can get position information with an unlimited range. That is,

POS(n)=POS(n-1)+dPLS(n),

Where POS(n) represents the position information at n-th sampling time. A simple program example in the Appendix demonstrates how to calculate speed and position monitoring with ISR. Please refer to Appendix A.

Please note to choose a suitable interrupting period. The example in the Appendix shows how to set the interrupting period. The principle is not to

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

let dPLS(n) exceed ± 65536 (the maximum counter range). So user has to calculate the maximum input pulse frequency and choose the suitable interrupting period.

For example, a encoder with 5000 pulses per round for each A/B phase, and if the maximum velocity of this motor is 3000 rpm (revolution per minute), the maximum input pulse frequency will be F_max.

F_max = 5000 x 4 x (3000/60) = 1000,000(Hz).

If the period we choose for INT0 is 1ms, then we will have dPLS(n) = 1000. It's the maximum pulse difference we will have between two interrupts. The shorter the interrupt period we set, we can have better dynamic response for sensing, but the resolution and CPU run time effort will be poorer with it. The period of INT0 can be set by _8133_Set_INT0Perd() function, the possible range is from 0.1us to 6.5ms. The period of INT1 can be set by _8133_Set_INT1Perd() function. The period of INT1 can only be n times of INT0's (0 < n < 256). Using this method, we can extend the range of INT1 from 0.1us to about 1.7sec.

These two interrupts can be accepted by host CPU only when they are enabled individually in _8133_Set_Int_Control() function. When entering an ISR, _8133_CLR_IRQ0() or _8133_CLR_IRQ1() function must be excused to clear the interrupt request to host CPU.

3.6. Index Latch Register

Three index latch registers are provided in PCI-8133 card. The data in the counter will be latched into the register at the rising edge of each corresponding index signal (PHCx, x=1, 2, 3). These registers can be read by _8133_Read_Index() function. At the same time, the latching status of the register will be recorded in the status register and can be read by _8133_Read_Status() function.

Three bits (IDL1, IDL2, IDL3) are used to show the status. They are all zeros when reset, become 1 when a rising edge of index signal of PHCx is detected, where IDLx is corresponding to each channel's index signal PHCx. Users can see if an Index position is reached by polling the status of these three bits continuously. IDLx will be 1 once an index signal is reached, thus users will have to clear IDLx by using _8133_CLR_IdxLah() function before index signal enter. This function will be benefieial for user when performing the "Origin Return" function.

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

3.7. PWM Signal Generator

This function is used to generate 3 complementary PWM signals (UU,UD), (VU,VD), (WU,WD). These PWM signals can be used either for 3 phase power transistor control or used as a simple Digital to Analog converter with a low pass filter. Note that the output of these six PWM signals is open collector, and they can sink current up to 20mA. So they can interface with photo-couplers directly. External pull up resistor is necessary when used as simple D/A converter.

The carrier frequency of PWM signals is synchronized with INT0, and frequency is designed to be half of INT0's frequency. For example: if the INT0 frequency to be 10KHz, then the frequency of PWM signal is automatically 5KHz.

The length of each PWM signal can be set by _8133_PWMPerd() function and it must be less than INT0 period. For example: if the value we set in _8133_Set_INT0Perd() is 1000 for 10KHz frequency, and the value in _8133_set_PWMPerd() function is 500, then the on duration of one INT0 cycle is 100us and the duration of one PWM cycle is 200us (5KHz).

Users can write in the value of PWM period at any time, and the on duration will be changed at next INT0 period. When used in power transistor control application, dead time is necessary to be given to prevent short circuit between upper and lower transistor for example, UU &UD. A Dead Time Generator circuit is provided and the _8133_SET_DT() function is used for setup the length of dead time. The relationship between PWM signals, INT0 and dead time is illustrated in the following figure.

Lower byte of the register value is used to set the dead time of 3 PWM complementary signals (UU, UD), (VU,VD), (WU,WD). The formula of setting this period is as following,

$$TDT = 0.75 * (m + 1); (us) \quad (0<m<256)$$

where TDT is the physical dead time generated in unit of micro second.

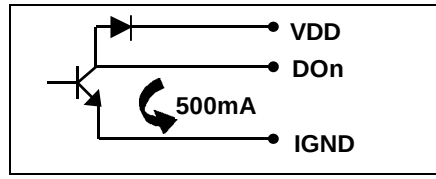
LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

3.8 Digital Inputs and Outputs

8 Isolated digital inputs and 8 isolated digital outputs are provided in PCI-8133. These I/O ports are for general purpose.

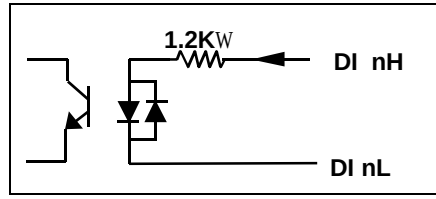
3.8.1 Isolation Digital Outputs

The isolated digital output is an open collector transistor output. The connection of isolated-digital output is shown as following diagram. When the isolated digital output goes to high, the sink current will be from DO channel n.



3.8.2 Isolation Digital Inputs

The isolation digital input accept voltage form 0V to 24V and input resister is 1.2K Ω. The connection between outside signal and PCI-8133 is shown below.



3.8.3 Extra 3 Isolation Digital Inputs

There are 3 extra isolated D/I signals on the 37-pin D-type connectors. The input circuit of these extra signals are the same as which used for the encoder input signals. They are differential input with 220 Ohm resistance loading, and with 5000V rms isolation voltage too.

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

3.9. Interrupt Control

3.9.1 System Architecture

The PCI-8133's interrupt system is a powerful and flexible system that is suitable for motion control applications. The system is a ***Dual Interrupt System***. The dual interrupt means the hardware can generate two interrupt request signals in the same time and the software can service these two request signals by ISR. Note that the dual interrupt does not mean the card occupies two IRQ levels.

The two interrupt request signals (INT0 and INT1) come from the on-board timers. Fig 5.2.1 shows the interrupt system.

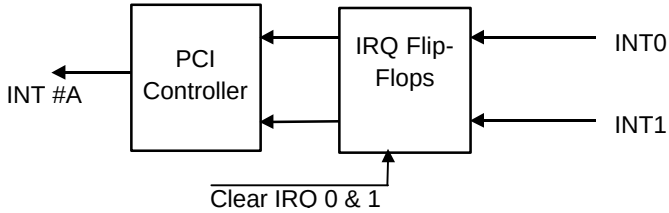


Fig 3.9.1 Dual Interrupt System of PCI-8133

3.9.2 INT0 Timer

The INT0 Timer can generate INT0 interrupt signal under the 10MHz timer base. It is a 16-bit timer. The formula for setting the interrupt period of INT0 is as following:

TINT0 = 0.1(us) x Timer Value (0 < Timer Value < 65536)

3.9.3 INT1 Timer

The INT1 Timer generate INT1 interrupt signal. The period of INT1 is (n+1) times of INT0 period (0 < n < 256). Using INT1, the interrupting period can be adjusted flexibly and be extended to very long time. The timer is 8 bits. The formula for setting INT1 interrupt period is as following,

TINT1 = TINT0 * (n+1) (us); (0<n<256)

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

3.9.4. IRQ Level Setting

There is only one IRQ level used by this card, although it is a dual interrupt system. This card uses INT #A interrupt request signal to PCI bus. The motherboard circuits will transfer INT #A to one of the AT bus IRQ levels. The IRQ level is set by the PCI plug and play BIOS and saved in the PCI controller. It is not necessary for users to set the IRQ level.

3.9.5 Dual Interrupt System

The PCI controller of PCI-8133 can receive two hardware IRQ sources. However, a PCI controller can generate only one IRQ to PCI bus, the two IRQ sources should be distinguished by ISR of the application software if the two IRQ are all used.

The application software can use the " _8133_Get_Irq_Status" function to distinguish which interrupt is inserted. After servicing an IRQ signal, users should check if another IRQ is also asserted and then clear current IRQ to allow the next IRQ occurring. The two IRQs are INT0 and INT1 that come from the interrupt generator.

Note that even you disable all the two IRQ sources without changing the initial condition of the PCI controller, the PCI BIOS still assigns an IRQ level to the PCI card and it will occupy the PC resource. It is not suggested to re-design the initial condition of the PCI card by users' own application software. If users want to disable the IRQ level, please use the ADLINK's software utility to change the power on interrupt setting.

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

4

Registers Format

The detailed descriptions of the register format of the PCI-8133 are specified in this chapter. This information is quite useful for the programmers who wish to handle the card by low-level programming.

4.1 I/O Port Address

The PCI-8133 functions as a 32-bit PCI target device to any master on the PCI bus. There are three types of registers on the PCI-8133: PCI Configuration Registers (PCR), Local Configuration Registers (LCR) and PCI-8133 registers.

The PCRs which compliant the PCI-bus specifications are initialized and controlled by the system plug & play PCI BIOS. Users can study the PCI BIOS specifications to understand the operation of the PCR. The PCR can only be read through by PCI BIOS function call.

The LCRs are specified by the PCI bus controller "PLX-9050". It is not necessary for users to understand the details of the LCR if you use the software library. The base address of the LCR is assigned by the PCI p&p BIOS. The assigned address is located at offset 14h of PCR.

The PCI-8133 registers are shown in the Table 4.1. The base address of the PCI-8133 registers is also assigned by the PCI p&p BIOS. The assigned base address is located at offset 18h of PCR. Note that most of the PCI-8133 registers are 16 bits. The users can access these registers by 16 bits I/O instructions.

There is one 32 bits register on PCI-8133. The 32 bits register occupied another LCR address space, that is, base address #2. The base address is allocated by PCI BIOS and is stored at offset 1Ch of PCR.

Users can read the PCR to get the LCR base address and the PCI-8133

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

base address by using the PCI BIOS function call.

I/O Base Address	Write	Read
Base + 00h	--	16-bit Counter Value 1
Base + 02h	--	16-bit Counter Value 2
Base + 04h	--	16-bit Counter Value 3
Base + 06h	Clear Index Register 1	Counter Index for Counter 1
Base + 08h	Clear Index Register 2	Counter Index for Counter 2
Base + 0Ah	Clear Index Register 3	Counter Index for Counter 3
Base + 0Ch	--	Status Register
Base + 10h	Digital Output Register	--
Base + 12h	Control Register	--
Base + 14h	Reserved	--
Base + 16h	16-bit INT0 Period Register	--
Base + 18h	8-bit INT1 Register	--
Base + 1Ah	12-bit PWM U Channel	--
Base + 1Ch	12-bit PWM V Channel	--
Base + 1Eh	12-bit PWM W Channel	--
Base + 40h	--	Digital Input Channels
Base + 80h	Clear H/W INT0	--
Base + 90h	Clear H/W INT1	--

Table 4.1 I/O Address MAP

4.2 Counter Registers

There are 3 independent unsigned 16-bit UP/DOWN counters. The counter register stores the value of the counter. Please refer Section 3.6 for the detail of the counter operation.

Address : BASE + 0 / BASE + 2 / BASE +4
Attribute : read only
Data Format:

Bit	7	6	5	4	3	2	1	0
BASE+0/2/4	CB7	CB6	CB5	CB4	CB3	CB2	CB1	CB0
BASE+1/3/5	CB15	CB14	CB13	CB12	CB11	CB10	CB9	CB8

CB15 ..CB0 : Counter value, CB15 is the Most Significant Bit (MSB), CB0 is the Least Significant Bit(LSB).

4.3 Index Registers

When the PHCn signal is triggered, the index register is used to store the counter value and set the IDLn bit in the status register. Please refer to Section

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

3.7 for the details. To read these registers can get the value of the "index" of a physical system. To write these registers can clear the index status flags in the status register.

Address : BASE + 6 / BASE + 8 / BASE +A
Attribute : read
Data Format:

Bit	7	6	5	4	3	2	1	0
BASE+6/8/A	CB7	CB6	CB5	CB4	CB3	CB2	CB1	CB0
BASE+7/9/B	CB15	CB14	CB13	CB12	CB11	CB10	CB9	CB8

CB15 ..CB0 : Counter value, CB15 is the Most Significant Bit (MSB), CB0 is the Least Significant Bit(LSB).

Address : BASE + 6 / BASE + 8 / BASE +A
Attribute : write
Data Format:

Bit	7	6	5	4	3	2	1	0
BASE+6/8/A	X	X	X	X	X	X	X	X
BASE+7/9/B	X	X	X	X	X	X	X	X

4.4 Status Register

The register shows the status of the photo-isolated pins and the index latch registers.

Address : BASE + 0x0Ch
Attribute : read only
Data Format:

Bit	7	6	5	4	3	2	1	0
BASE+ 0x0C	1	IDL1	IDL2	IDL3	1	IN2	IN1	IN0
BASE+ 0x0D	0	0	0	0	0	0	0	DIR

IN2~IN0: The inputs from 3 differential photo-isolated pins.

IDL3~IDL1: The status of index latch register. The initial value of these bits are zero. The status bits are set to 1 at the rising edge of the index signals (PHC3~PHC1), and are reset to zero by writing the "index register".

DIR: This bit shows the counting direction of the counter 3, DIR="1" means up counting.

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

4.5 Digital Output Register

The register set the isolated digital output pins.

Address : BASE + 0x10h

Attribute : read only

Data Format:

Bit	7	6	5	4	3	2	1	0
BASE+ 0x10	DO7	DO6	DO5	DO4	DO3	DO2	DO1	DO0
BASE+ 0x11	X	X	X	X	X	X	X	X

DO7~DO0: Bit 7~ bit 0 of the isolated digital input.

4.6 Control Mode Register

The register controls the counter operation modes and the PWM output waveform.

Address : BASE + 0x12h

Attribute : write only

Data Format:

Bit	7	6	5	4	3	2	1	0
BASE+ 0x0E	PE	X	INDEX	FTS1	FTS0	C2M1	C2M0	C1MS
BASE+ 0x0F	X	X	X	X	X	X	X	X

C1MS: the mode select bits of counter 1. The input signals of counter 1 are from PHA1, PHA1/, PHB1 and PHB1/.

C1MS	Operation Modes of counter 1
0	Input Signal is (CCW+CW) PHA1: Up count pulse PHB1: Down count pulse
1 (default)	Input Signal is A,B phase encoder

C2M1, C2M0: the mode control bits of counter 2. The input signals of counter 2 are from PHA2, PHA12, PHB2 and PHB2. There are four operating modes for counter 2.

C2M1	C2M0	Operation Modes of counter 2
0	0	PHA2: Input pulse for counting PHB2: Counting Direction
0	1	PHA2: Up count pulse PHB2: Down count pulse
1	0	A,B phase encoder input (No pass digital filter)
1	1	A,B phase encoder input (signals pass through digital filter)

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

FTS1, FTS2: are used to select the time base of digital filters. Three 3-stage digital filters are used to filter out the noise for the three UD/DOWN counters. Users should set the values according to different operating conditions. For example: (FTS1, FTS0) = (1, 0), then the signal level transition time which less than 1.2 μs will be filtered and ignored for the counter. The setting of the time base is as following:

FTS1	FTS0	Digital Filter Time Period
0	0	300 ns
0	1	600 ns
1	0	1.2 μs
1	1	4.8 μs

PE: PWM output enable control bit. Six PWM waveforms output are enabled when this bit is set to "1", otherwise they are all zero voltage output when PE=0. Default value is 0.

4.7 Interrupt #0 Period Register

The register value is used to set the period for generating INT0 interrupt signals. Please refer to section 3.9.2 for the operation.

Address : BASE + 0x16h
Attribute : write only
Data Format:

Bit	7	6	5	4	3	2	1	0
BASE+0x16	CV7	CV6	CV5	CV4	CV3	CV2	CV1	CV0
BASE+0x17	CV15	CV14	CV13	CV12	CV11	CV10	CV9	CV8

CV15 ..CV0: Timer value, CV15 is the MSB, CV0 is the LSB.

4.8 Interrupt #1 Period Register

The register value is used to set the period for generating INT1 interrupt signals and the dead time of PWM output signals. The register value is divided into two parts. Upper byte is used to set INT1 period, whereas lower byte is for setting PWM dead time.

Please refer to section 3.9.3 and section of 3.6 for details of controlling INT1 period and PWM dead time respectively.

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Address : BASE + 0x18h

Attribute : write only

Data Format:

Bit	7	6	5	4	3	2	1	0
BASE+0x18	N7	N6	N5	N4	N3	N2	N1	N0
BASE+0x19	M7	M6	M5	M4	M3	M2	M1	M0

N7~N0: INT1 control value

M7~M0: PWM Dead Time control value

4.9 PWM Output Registers

There are three PWM output channels in PCI-8133. The register value is used to set the time period of the PWM signals: (UU,UD), (VU,VD), (WU,WD). The resolution of the time period is 12-bit. Please refer to section 3.7 for detail of the PWM setting.

Address : BASE + 0x1Ah / +0x1Ch / 0x1Eh

Attribute : write only

Data Format:

Bit	7	6	5	4	3	2	1	0
BASE+0x1A	PM7	PM6	PM6	PM4	PM3	PM2	PM1	PM0
BASE+0x1B	X	X	X	X	PM11	PM10	PM9	PM8

4.10 Digital Input Register

This register value shows the value of isolated digital input channels. There are 8 DI channels.

Address : BASE + 0x40h

Attribute : read only

Data Format:

Bit	7	6	5	4	3	2	1	0
BASE+0x40	DI7	DI6	DI5	DI4	DI3	DI2	DI1	DI0
BASE+0x41	X	X	X	X	X	X	X	X

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

4.11 Interrupt Clear Registers

There are two interrupt clear registers. After processing the interrupt by software ISR, these registers must be written to clear the interrupt flags, so that enable the next interrupt trigger. Address 0x80h and 0x90h are used to clear INT0 and INT1 respectively.

Address : BASE + 0x80h / BASE + 0x90h

Attribute : write only

Data Format:

Bit	7	6	5	4	3	2	1	0
BASE+0x80	X	X	X	X	X	X	X	X
BASE+0x90	X	X	X	X	X	X	X	X

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

5

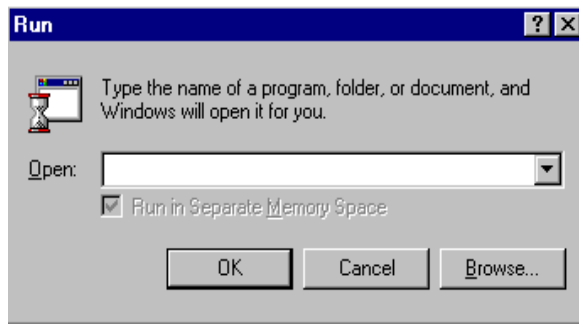
C/C++ Software Library

The library includes all the functions of PCI-8133. There are some sample programs included in this disk to help you to use this library.

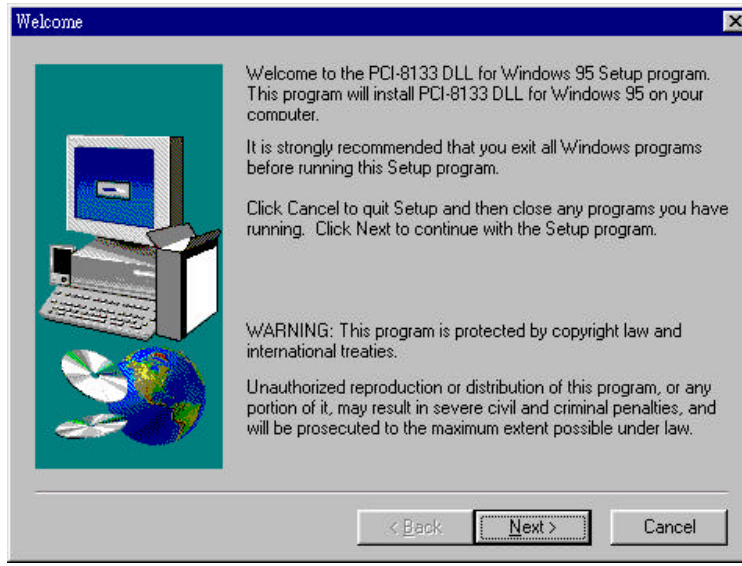
5.1 Installation

- **MS-DOS Software Installation**
  - **step 1.** Place ADLINK's " Manual & Software utility " CD into your CD-ROM drive.
  - **step 2.** Type the command (X indiicates the CD-ROM drive):
    X:\> **CD NuDAQPCI\8133\DOS**
    X:\> **NuDAQPCI\8133\DOS>SETUP**
  - **step 3.** An installation complete message will be shown on the screen.
- **Windows 95/98/NT Software Installation**
  - **step 1.** ADLINK's " Manual & Software utility " CD into your CD-ROM drive.
  - **step 2.** If Windows 95/98/NT is loaded, choose Run from the taskbar.
  - **step 3.** Type **X:\NuDAQPCI\8133\Win95>Setup.exe** in the Run dialog box.

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000



Setup first displays a Welcome dialog box. Please click Next button to go on installation.



LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Setup then prompt the following dialog box for you to specify the destination directory. The default path is C:ADLINK\8133\W95. If you want to install *PCI-8133 DLL for Windows 95/98/NT* in another directory, please click Browse button to change the destination directory.



Then you can click Next to begin installing *PCI-8133 DLL for Windows 95/98/NT*.

After you complete the installation of PCI-8133 Software, PCI-8133's DLL (8133.DLL) is copied to Windows System directory (default is C:\WINDOWS\SYSTEM) and the driver files (W95_8133.VXD and PCIW95.VXD) are also copied to the appropriate directory.

There are two kinds of import library files in <InstallDir>\LIB directory. 8133.lib is Visual C/C++ import library file. 8133_bc.lib is Borland C++ 5.0 import library files.

5.2 C/C++ Programming Library

We defined some data types in Pci_8133.h. These data types are used by PCI-8133 library. We suggest you to use these data types in your application programs. The following table shows the data type names and their range.

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Data Types

Type Name	Description	Range
U8	8-bit ASCII character	0 to 255
I16	16-bit signed integer	-32768 to 32767
U16	16-bit unsigned integer	0 to 65535
I32	32-bit signed integer	-2147483648 to 2147483647
U32	32-bit single-precision floating-point	0 to 4294967295
F32	32-bit single-precision floating-point	-3.402823E38 to 3.402823E38
F64	64-bit double-precision floating-point	-1.797683134862315E308 to 1.797683134862315E309
Boolean	Boolean logic value	TRUE, FALSE

The functions of PCI-8133's software drivers use full-names to represent the functions' real meaning. The naming convention rules are :

In DOS Environment :

_{hardware_model}_{action_name}. e.g. **_8133_Initial()**.

In order to recognize the difference between DOS library and Windows 95/98/NT library, A capital "**W**" is put on the head of each function name of the Windows 95/98/NT DLL driver. e.g. **W_8133_Initial()**.

The detail descriptions of each function are specified in the following sections.

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

5.3 _8133_Initial

@ *Description*

This function is used to initialize PCI-8133 card. Every PCI-8133 card has to be initialized by this function before calling other functions.

@ *Syntax*

C/C++ (DOS)

```
U16 _8133_Initial (U16 *existCards, PCI_INFO *info)
```

C/C++ (Windows 95/98)

```
U16 W_8133_Initial (U16 *existCards, PCI_INFO *info)
```

C/C++ (Windows NT)

```
U16 W_8133_Initial (U16 CardNo)
```

Visual Basic (Windows 95/98)

```
W_8133_Initial (existCards As Integer, info As PCI_INFO) As
Integer
```

Visual Basic (Windows NT)

```
W_8133_Initial (ByVal CardNo As Integer) As Integer
```

@ *Argument*

existCards:	numbers of existing PCI-8133 cards
CardNo:	Assigned card number
info:	relative information of the PCI-8133 cards

@ *Return Code*

```
ERR_NoError
ERR_BoardNoInit
ERR_PCIBiosNotExist
```

5.4 _8133_Software_Reset

@ *Description*

This function is used to reset the I/O port configuration. Note that this function can not re-start the PCI bus and all the hardware setting won't be changed neither.

@ *Syntax*

C/C++ (DOS)

```
void _8133_Software_Reset (U16 cardNo)
```


LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

C/C++ (Windows 95/98/NT)

```
void W_8133_Software_Reset (U16 cardNo)
```

Visual Basic (Windows 95/98/NT)

```
W_8133_Software_Reset (ByVal cardNo As Integer)
```

@ *Argument*

cardNo: card number

@ *Return Code*

ERR_NoError
ERR_BoardNoInit
ERR_PCIBiosNotExist

5.5 _8133_Read_Cnt

@ *Description*

This function is used to read data from 3 independent unsigned 16-bit UP/DOWN counters, whose range is from 0h to 0xFFFFh. The operating mode of these 3 counters is defined by 8133_ModeSelect function.

@ *Syntax*

C/C++ (DOS)

```
U16  _8133_Read_Cnt (U16 cardNo, U16 CntNo, U16 *CntData)
```

C/C++ (Windows 95/98/NT)

```
U16  W_8133_Read_Cnt (U16 cardNo, U16 CntNo, U16 *CntData)
```

Visual Basic (Windows 95/98/NT)

```
W_8133_Read_Cnt (ByVal cardNo As Integer, ByVal CntNo As
Integer, CntData As Integer) As Integer
```

@ *Argument*

cardNo: card number
CntNo: Counter Number (= 1, 2, 3)
CntData: Data read from counter (= 0H .. FFFFh)

@ *Return Code*

ERR_BoardNoInit
ERR_NoError

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

5.6 _8133_Read_Index

@ *Description*

This function is used to read data from 3 index registers associated with 3 UD/DOWN counters. The value of UP/DOWN counters (Counter No.=1, 2, 3) will be latched into index registers (Index No.=1, 2, 3) at the rising edge of its associated index signal: PHC1, PHC2, PHC3. The latching status of every index register can be read from _8133_Read_Status function.

@ *Syntax*

C/C++ (DOS)

```
U16  _8133_Read_Index (U16 cardNo, U16 IndexNo, U16
*IndexData)
```

C/C++ (Windows 95/98/NT)

```
U16  W_8133_Read_Index (U16 cardNo, U16 IndexNo, U16
*IndexData)
```

Visual Basic (Windows 95/98/NT)

```
W_8133_Read_Index (ByVal cardNo As Integer, ByVal IndexNo
As Integer, IndexData As Integer) As Integer
```

@ *Argument*

cardNo: card number
IndexNo: Index Number (= 1, 2, 3)
IndexData: Index Value (= 0H .. FFFFh)

@ *Return Code*

```
ERR_BoardNoInit
ERR_NoError
```

5.7 _8133_Read_Status

@ *Description*

This function is used to read data from status register. The definition of each bit in this register is described in section 5.x.

@ *Syntax*

C/C++ (DOS)

```
U16  _8133_Read_Status (U16 cardNo, U16 *Status)
```

C/C++ (Windows 95/98/NT)

```
U16  W_8133_Read_Status (U16 cardNo, U16 *Status)
```


LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Visual Basic (Windows 95/98/NT)

```
W_8133_Read_Status (ByVal cardNo As Integer, Status As
Integer) As Integer
```

@ *Argument*

cardNo: card number
Status: Value in Status Reg.

@ *Return Code*

ERR_BoardNoInit, ERR_NoError

5.8 _8133_CLR_IdxLah

@ *Description*

The latching status of the index register can be reset by using this function.

@ *Syntax*

C/C++ (DOS)

```
U16  _8133_CLR_IdxLah (U16 cardNo, U16 IndexNo)
```

C/C++ (Windows 95/98/NT)

```
U16  W_8133_CLR_IdxLah (U16 cardNo, U16 IndexNo)
```

Visual Basic (Windows 95/98/NT)

```
W_8133_ CLR_IdxLah (ByVal cardNo As Integer, ByVal IndexNo
As Integer) As Integer
```

@ *Argument*

cardNo: card number
IndexNo: The index number for the latching bit to be cleared

@ *Return Code*

ERR_BoardNoInit
ERR_NoError

5.9 _8133_ModeSelecct

@ *Description*

This function is used to set the control mode resister. The definition of each bit in the control word is described in section 4.6.

@ *Syntax*

C/C++ (DOS)

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

```
U16  _8133_ModeSelect (U16 cardNo, U16 Mode)
```

C/C++ (Windows 95/98/NT)

```
U16  W_8133_ModeSelect (U16 cardNo, U16 Mode)
```

Visual Basic (Windows 95/98/NT)

```
W_8133_ModeSelect (ByVal cardNo As Integer, ByVal Mode As
Integer) As Integer
```

@ *Argument*

cardNo: card number
Mode: Control word to be written

@ *Return Code*

ERR_BoardNoInit
ERR_NoError

5.10 _8133_Set_Int0Perd

@ *Description*

This function is used to write the INT0 period control registers. Please refer to Section 4.X for the detail of this registers.

@ *Syntax*

C/C++ (DOS)

```
U16  _8133_Set_Int0Perd (U16 cardNo, U16 Int0Perd)
```

C/C++ (Windows 95/98/NT)

```
U16  W_8133_Set_Int0Perd (U16 cardNo, U16 Int0Perd)
```

Visual Basic (Windows 95/98/NT)

```
W_8133_Set_Int0Perd (ByVal cardNo As Integer, ByVal
Int0Perd As Integer) As Integer
```

@ *Argument*

cardNo: card number
Int0Perd: Interrupt 0 period to be set

@ *Return Code*

ERR_BoardNoInit
ERR_NoError

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

5.11 _8133_Set_Int1Perd

@ *Description*

This function is used to set the INT1 interrupt period control register. Please refer to section 4.8 for the details.

@ *Syntax*

C/C++ (DOS)

```
U16  _8133_Set_Int1Perd (U16 cardNo, U16 Int1Perd)
```

C/C++ (Windows 95/98/NT)

```
U16  W_8133_Set_Int1Perd (U16 cardNo, U16 Int1Perd)
```

Visual Basic (Windows 95/98/NT)

```
W_8133_Set_Int1Perd (ByVal cardNo As Integer, ByVal
Int1Perd As Integer) As Integer
```

@ *Argument*

cardNo: card number
Int1Perd: Interrupt period to be set

@ *Return Code*

ERR_BoardNoInit
ERR_NoError

5.12 _8133_Set_PWMPerd

@ *Description*

This function is used to set the period for generating 3 complementary PWM signals (UU, UD), (VU,VD), (WU,WD). Please refer to Section 4.9 for the detail of the PWM control.

@ *Syntax*

C/C++ (DOS)

```
U16  _8133_Set_PWMPerd (U16 cardNo, U16 PWMNo, U16 PWMPerd)
```

C/C++ (Windows 95/98/NT)

```
U16  W_8133_Set_PWMPerd (U16 cardNo, U16 PWMNo, U16 PWMPerd)
```

Visual Basic (Windows 95/98/NT)

```
W_8133_Set_PWMPerd (ByVal cardNo As Integer, ByVal PWMNo As
Integer, ByVal PWMPerd As Integer) As Integer
```

@ *Argument*

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

cardNo: card number
PWMNo: PWMNo = 1, 2, 3 for U, V, W arm respectively
PWMPerd: PWM period to be set

@ *Return Code*

ERR_BoardNoInit
ERR_NoError

5.13 _8133_INT_Control

@ *Description*

The PCI-8133 uses the dual interrupts system. The two interrupt sources can be generated and be checked by the software. This function is used to select and control PCI-8133 interrupt sources by writing data to interrupt control register. Please refer to section 5.2 for detailed descriptions of interrupt system.

@ *Syntax*

C/C++ (DOS)

void _8133_INT_Control (U16 cardNo, U16 int0Ctrl , U16 int1Ctrl)

C/C++ (Windows 95/98/NT)

void W_8133_INT_Control (U16 cardNo, U16 int0Ctrl , U16 int1Ctrl)

Visual Basic (Windows 95/98/NT)

```
W_8133_INT_Control (ByVal cardNo As Integer, ByVal int0Ctrl
As Integer, ByVal int1Ctrl As Integer) As Integer
```

@ *Argument*

cardNo: the card number of PCI-8133 card initialized.
Int0Ctrl: 0: INT0 disable
1: INT0 enable
int1Ctrl: 0: INT1 disable
1: INT1 enable

@ *Return Code*

None

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

5.14 _8133_CLR_IRQ0 & _8133_CLR_IRQ1

@ Description

These functions are used to clear interrupt request that generated by PCI-8133.

@ Syntax

C/C++ (DOS)

void _8133_CLR_IRQ0 (U16 cardNo)
void _8133_CLR_IRQ1 (U16 cardNo)

C/C++ (Windows 95/98/NT)

void W_8133_CLR_IRQ0 (U16 cardNo)
void W_8133_CLR_IRQ1 (U16 cardNo)

Visual Basic (Windows 95/98/NT)

```
W_8133_CLR_IRQ0 (ByVal cardNo As Integer)
W_8133_CLR_IRQ1 (ByVal cardNo As Integer)
```

@ Argument

cardNo: the card number of PCI-8133 card initialized.

@ Return Code

None

5.15 _8133_Get_IRQ_Channel

@ Description

This function is used to get the IRQ level of the PCI-8133 card used currently.

@ Syntax

C/C++ (DOS)

void _8133_Get_IRQ_Channel (U16 cardNo, U16 *irq_no)

C/C++ (Windows 95/98/NT)

void W_8133_Get_IRQ_Channel (U16 cardNo, U16 *irq_no)

Visual Basic (Windows 95/98/NT)

```
W_8133_Get_IRQ_Channel (ByVal cardNo As Integer, irq_no As
Integer)
```

@ Argument

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

cardNo: the card number of PCI-8133 card initialized.
Irq_no: the IRQ level used to transfer A/D data for this card

@ *Return Code*

None

5.16 _8133_Get_IRQ_Status

@ *Description*

The PCI-8133 has dval interupt system .Two interupt sources can be generated and be checked by software.This function is used to distinguish which interupt is inserted if both INT1 and INT2 are enabled .

@ *Syntax*

C/C++ (DOS)

void _8133_Get_IRQ_Status (U16 cardNo, U16 *ch1, U16 *ch2)

C/C++ (Windows 95/98/NT)

void W_8133_Get_IRQ_Status (U16 cardNo, U16 *ch1, U16 *ch2)

Visual Basic (Windows 95/98/NT)

```
W_8133_Get_IRQ_Status (ByVal cardNo As Integer, ch1 As
Integer, ch2 As Integer)
```

@ *Argument*

cardNo: the card number of PCI-8133 card initialized.
ch1: the IRQ status of INT 0: interupt is not from INT1, 1: interupt is from INT1 .
ch2: the IRQ status of INT2, 0: interupt is not from INT2, 1: interupt is from INT2 .

@ Return Code

None

5.17 _8133_DO

@ *Description*

This function is used to write data to digital output port. There are 8 digital output channels on PCI_8133.

@ *Syntax*

C/C++ (DOS)

```
U16  _8133_DO (U16 cardNo, U16 DOData)
```

C/C++ (Windows 95/98/NT)

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

```
U16  W_8133_DO (U16 cardNo, U16 DOData)
```

Visual Basic (Windows 95/98/NT)

```
W_8133_DO (ByVal cardNo As Integer, ByVal DOData As Integer)
As Integer
```

@ *Argument*

cardNo: The card number of PCI-8133 card initialized
DOData: The value will be written to digital output port

@ *Return Code*

ERR_NoError

5.18 _8133_DI

@ *Description*

This function is used to read data from digital input ports. There are 8 digital input channels on PCI_8133. The digital input status can be accessed by this function directly.

@ *Syntax*

C/C++ (DOS)

U16 _8133_DI (U16 cardNo, U16 *DIData)

C/C++ (Windows 95/98/NT)

U16 W_8133_DI (U16 cardNo, U16 *DIData)

Visual Basic (Windows 95/98/NT)

```
W_8133_DI (ByVal cardNo As Integer, DIData As Integer) As
Integer
```

@ *Argument*

cardNo: The card number of PCI-8133 card initialized
DIData: The value accessed from digital input port

@ *Return Code*

ERR_NoError

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

5.19 _8133_INT_Enable

@ Description

This function is only available in Windows 95/98/NT driver. It is used to start up the interrupt control. After calling this function, every time an interrupt request signal generated, a software event is signaled. So that in your program, you can use wait operation.Wait For Single Object, to wait for the event. When the event is signaled, it means an interrupt is generated.

@ Syntax

C/C++ (Windows 95/98/NT)

U16 W_8133_INT_Enable (U16 cardNo, HANDLE *hEvent)

Visual Basic (Windows 95/98/NT)

```
W_8133_INT_Enable (ByVal cardNo As Integer, hEvent As Long)
As Integer
```

@ Argument

cardNo: The card number of PCI-8133 card initialized

hEvent: The address of an array of two handles , hEvent[0] and Event[1] are the events for interupt signals INT1 and INT2 respectively.

@ Return Code

ERR_NoError

5.20 _8133_INT_Disable

@ Description

This function is only available in Windows 95/98/NT driver. It is used to disable the interrupt signal generation.

@ Syntax

C/C++ (Windows 95/98/NT)

U16 W_8133_INT_Disable (U16 cardNo)

Visual Basic (Windows 95/98/NT)

```
W_8133_INT_Disable (ByVal cardNo As Integer) As Integer
```

@ Argument

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

cardNo: The card number of PCI-8133 card initialized

@ *Return Code*

ERR_NoError

Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Appendix A

This appendix is a demo program to describe how to write an ISR for speed and position monitoring on DOS environment.

```
#define         IC8259_1        0x20
#define         IC8259_2        0xA0
#define         EOI             0x20
void interrupt far   isr(void);
void (interrupt far *old_isr)();
U16     irq_mask,old_mask;
U16     ISR_irqchn;
int     dPLS;
long    Position=0;
U16     pls_N=0, pls_NM1=0;

void main(void)
{
        U16     vect_no, irq_chn;

        _8133_Software_Reset( cno  );
        delay( 1 );

        /*  Set Control mode for index mode, CNT1 & CNT2 at CW/CCW mode */
        _8133_ModeSelect( cno, 0x42);

        /*  Set Period of INT0 is 200us */
        _8133_Set_Int0Perd( cno, 0x07d0);

        /*  Set Period of INT1 is 5ms, Dead time 3.5 us */
        _8133_Set_Int1Perd( cno, 0x1803);

        /*  Enable only INT1 */
        _8133_Set_INT_Control( cno, 0, 1 );

        _8133_Get_IRQ_Channel( cno, &irq_chn );  /* Get IRQ Channel */
        ISR_irqchn = irq_chn;

        if ( irq_chn==0)   return 1;
        if ( irq_chn < 8)          vect_no = 0x08 + irq_chn ;
        else if (irq_chn == 9)     vect_no = 0x0A ;
        else                       vect_no = 0x70 + irq_chn - 8;

        disable();
        old_isr = getvect( vect_no  );
        setvect( vect_no  , isr );
        enable();
                                                /* Enable ISR */
```


LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

```
        if( irq_chn < 8 )
        {
                irq_mask = inp( IC8259_1 + 1 );
                old_mask = irq_mask ;
                outp( IC8259_1 + 1 , irq_mask & (0xFF^(1<<irq_chn)) );
        }
        else{
                irq_mask = inp( IC8259_1 + 1 );
                outp( IC8259_1 + 1 , irq_mask & 0xFB );
                                                /* IRQ2 : 1111 1011
*/
                irq_mask = inp( IC8259_2 + 1 );
                old_mask = irq_mask ;
                outp( IC8259_2 + 1 , irq_mask & (0xFF^(1<<(irq_chn-8) )) );
        }

        /*  MAIN  PROGRAM  */
        if( irq_chn <8 )   outp( IC8259_1 + 1 , old_mask );
        else               outp( IC8259_2 + 1 , old_mask );
        setvect( vect_no  , old_isr );
        else               outp( IC8259_2 + 1 , old_mask );
        setvect( vect_no  , old_isr );
}

void interrupt far isr(void)
{
        _8133_Read_Cnt( cno, 3, &pls_N);
        _8133_CLR_IRQ1(  cno  );
                                // Clear INT1 Request
        dPLS = (int)pls_N - (int)pls_NM1;
                                // dPLS(n) = pls(n)-pls(n-1)
                                      // Speed = K x dPLS(n)
        Position += dPLS;             // P(n) = P(n-1) + dPLS(n)
        pls_NM1 = pls_N;              // update pls(n-1)

        if( ISR_irqchn > 8 )  outp( IC8259_2, EOI );
        outp( IC8259_1 , EOI );
}
```


LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000

Product Warranty/Service

Seller warrants that equipment furnished will be free form defects in material and workmanship for a period of one year from the confirmed date of purchase of the original buyer and that upon written notice of any such defect, Seller will, at its option, repair or replace the defective item under the terms of this warranty, subject to the provisions and specific exclusions listed herein.

This warranty shall not apply to equipment that has been previously repaired or altered outside our plant in any way as to, in the judgment of the manufacturer, affect its reliability. Nor will it apply if the equipment has been used in a manner exceeding its specifications or if the serial number has been removed.

Seller does not assume any liability for consequential damages as a result from our products uses, and in any event our liability shall not exceed the original selling price of the equipment.

The equipment warranty shall constitute the sole and exclusive remedy of any Buyer of Seller equipment and the sole and exclusive liability of the Seller, its successors or assigns, in connection with equipment purchased and in lieu of all other warranties expressed implied or statutory, including, but not limited to, any implied warranty of merchant ability or fitness and all other obligations or liabilities of seller, its successors or assigns.

The equipment must be returned postage-prepaid. Package it securely and insure it. You will be charged for parts and labor if you lack proof of date of purchase, or if the warranty period is expired.

LRI Distribuída no Brasil pela LRI | vendas@lri.com.br | 51 2129-3000