



# PIO-D24/D56 系列

包含 PIO-D24, PIO-D24U, PEX-D24U  
PIO-D56, PIO-D56U, PEX-D56U

## 用户手册

简体中文版  
版本: 2.7  
日期: 2015年3月

### 承诺

---

郑重承诺：凡泓格科技股份有限公司产品从购买即日起一年内无任何材料性缺损。

### 免责声明

---

凡使用本系列产品除产品质量所造成的损害，泓格科技股份有限公司不承担任何法律责任。泓格科技股份有限公司有义务提供本系列产品可靠而详尽资料，但保留修订权利，且不承担使用者非法利用资料对第三方所造成侵害构成的法律责任。

### 版权

---

版权所有 © 2011 泓格科技股份有限公司，保留所有权力。

### 商标

---

手册中所涉及所有公司商标，商标名称及产品名称分别属于该商标或名称的拥有者所有。

### 许可

---

用户可以使用、修改和备份单片机上的软件部份，但无权复制，转移或发布该软件的全部或部份拷贝。

# 目录

---

<b>1</b>	<b>绪论</b> .....	<b>4</b>
1.1	规格.....	5
1.1.1	PIO-D24/D24U/D56/D56U.....	5
1.1.2	PEX-D24/D56.....	6
1.2	特点.....	7
1.3	检查产品清单.....	7
<b>2</b>	<b>硬件结构</b> .....	<b>8</b>
2.1	板卡布局.....	8
2.2	I/O 口位置 .....	9
2.3	CARD ID 开关 .....	11
2.4	引脚分配.....	12
2.5	启动 I/O 运行 .....	15
2.5.1	DIO 口结构(CON3).....	16
2.5.2	DI 口结构(CON2) .....	17
2.5.3	DO 口结构(CON1).....	18
2.6	中断运行.....	19
2.7	端子板.....	20
2.7.1	DB-37.....	20
2.7.2	DN-37 及 DN-50.....	20
2.7.3	DB-8125.....	21
2.7.4	ADP-20/PCI.....	21
2.7.5	DB-24P, DB-24PD 光电隔离输入端子板.....	22
2.7.6	DB-24R, DB-24RD 继电器输入端子板.....	23
2.7.7	DB-24PR, DB-24POR, DB-24C .....	24
2.7.8	端子板对照表.....	25
<b>3</b>	<b>I/O 控制寄存器</b> .....	<b>26</b>
3.1	如何找到 I/O 地址 .....	26
3.2	分配 I/O 地址 .....	29

3.3	I/O 地址映像 .....	31
3.3.1	RESET\ 控制寄存器.....	32
3.3.2	AUX 控制寄存器.....	32
3.3.3	AUX 数据寄存器.....	32
3.3.4	INT Mask 控制寄存器.....	33
3.3.5	Aux 状态寄存器.....	33
3.3.6	Read Card ID 寄存器.....	33
3.3.7	I/O 选择控制寄存器.....	34
3.3.8	中断极性控制寄存器.....	34
3.3.9	读/写 8-bit 数据寄存器.....	35
<b>4</b>	<b>软件安装向导.....</b>	<b>36</b>
4.1	软件驱动程序安装.....	36
4.2	PNP 驱动程序安装.....	38
4.3	确认板卡安装成功 .....	39
<b>5</b>	<b>示例程序.....</b>	<b>40</b>
5.1	WINDOWS DEMO 程序 .....	40
5.2	DOS DEMO 程序.....	41
	<b>附录.....</b>	<b>42</b>
A1.	DOS LIB 函式.....	42

# 1 绪论



PIO-D24U/D56U 及 PEX-D24/D56 板卡是泓格新上市并符合 RoHS 环保规范的产品，新的 PIO-D24U/D56U 及 PEX-D24/D56 的设计可直接兼容于 PIO-D24/D56。

PIO-D24U/D56U 支持 3.3 V / 5 V PCI bus 接口，PEX-D24/D56 支持 PCI Express 接口。他们都提供有 24 个符合 5V/TTL 规范的 Digital I/O 通道，并且仿真 8255 Programmable Peripheral Interface (PPI) 芯片的 mode 0 (Basic Input/Output)。每个 PPI 由三组 8 位的双向 I/O 端口所组成，这三个埠分别叫作 Port A(PA)、Port B(PB)、Port C(PC)，每个 Port 的初始设定皆为 Input 模式。PIO-D56U/PEX-D56 比 PIO-D24U/PEX-D24 多提供了 16 个数字输出信道以及 16 个数字输入信道，这 16 个数字输出信道与 16 个数字输入信道输出方向是固定而不能被改变的。

PIO-D24U/D56U 及 PEX-D24/D56 在硬件上新增 Card ID 指拨开关，让使用者可以自由设定每张板卡的识别码。当系统同时使用多张 PIO-D24U/D56U 或 PEX-D24/D56 板卡时，使用者可以迅速而简单区别这些同型号的板卡。

此系列卡支持在 Linux、DOS、Windows 98/NT/2000、32/64-Bit Windows XP/2003/Vista/2008/7/8 等操作系统环境下使用，还提供有动态函数库及 Active X 控件使开发更加容易及简单易懂的各种语言范例程序，如 Turbo C++、Borland C++、Microsoft C++、Visual C++、Borland Delphi、Borland C++ Builder、Visual Basic、Visual C#.NET、Visual Basic.NET 及 LabVIEW 等，让用户能够快速的上手来使用。

# 1.1 规格

## 1.1.1 PIO-D24/D24U/D56/D56U

板卡名称	PIO-D24	PIO-D24U	PIO-D56	PIO-D56U
<b>可编程数字输出入</b>				
通道数	24			
<b>数字输入</b>				
数字输入通道数	-		16	
兼容性	5 V/TTL			
输入电压	Logic 0: 0.8 V max. Logic 1: 2.0 V min.			
响应速度	1 MHz			
<b>数字输出</b>				
数字输出通道数	-		16	
兼容性	5 V/TTL			
输出电压	Logic 0: 0.4 V max. Logic 1: 2.4 V min.			
输出能力	Sink: 64 mA @ 0.8 V Source: 32 mA @ 2.0 V	CON1	Sink: 2.4 mA @ 0.8 V Source: 0.8 mA @ 2.0 V	
		CON3	Sink: 64 mA @ 0.8 V Source: 32 mA @ 2.0 V	
响应速度	1 MHz			
<b>公共</b>				
总线型态	5 V PCI, 32-bit, 33 MHz	3.3 V/5 V Universal PCI, 32-bit, 33 MHz	5 V PCI, 32-bit, 33 MHz	3.3 V/5 V Universal PCI, 32-bit, 33 MHz
数据总线	8-bit			
卡 ID	无	有(4-bit)	无	有(4-bit)
I/O 连接头	DB37 接头 x 1		DB37 接头 x 1 20-pin 接头 x 2	
尺寸(长 x 宽 x 高)	143 mm x 105 mm x 22 mm			
耗电量	420 mA @ +5 V		580 mA @ +5 V	
运行温度	0 ~ 60 °C			
储存温度	-20 ~ 70 °C			
周围环境相对湿度	5 ~ 85% 相对湿度, 非冷凝(non-condensing)			

## 1.1.2 PEX-D24/D56

板卡名称	PEX-D24	PEX-D56	
<b>可编程数字输出</b>			
通道数	24		
<b>数字输入</b>			
数字输入通道数	-	16	
兼容性	5 V/TTL		
输入电压	Logic 0: 0.8 V max. Logic 1: 2.0 V min.		
响应速度	500 kHz		
<b>数字输出</b>			
数字输出通道数	-	16	
兼容性	5 V/TTL		
输出电压	Logic 0: 0.4 V max. Logic 1: 2.4 V min.		
输出能力	Sink: 64 mA @ 0.8 V Source: 32 mA @ 2.0 V	CON1	Sink: 2.4 mA @ 0.8 V Source: 0.8 mA @ 2.0 V
		CON3	Sink: 64 mA @ 0.8 V Source: 32 mA @ 2.0 V
响应速度	500 kHz		
<b>公共</b>			
总线型态	PCI Express x 1		
数据总线	8-bit		
卡 ID	有(4-bit)		
I/O 接头	DB37 接头 x 1	DB37 接头 x 1 20-pin 接头 x 2	
尺寸(长 x 宽 x 高)	143 mm x 105 mm x 22 mm		
耗电量	650 mA @ +3.3 V 0 mA @ +12 V	750 mA @ +3.3 V 0 mA @ +12 V	
运行温度	0 ~ 60 °C		
储存温度	-20 ~ 70 °C		
周围环境相对湿度	5 ~ 85% 相对湿度, 非冷凝(non-condensing)		

## 1.2 特点

---

- PIO-D24/D56 为 PCI bus 接口，支持 +5 V PCI bus 插槽
- PIO-D24U/D56U 为 Universal PCI 接口，支持 +3.3 V 及 +5 V PCI bus 插槽
- PEX-D24/D56 为 PCI Express 接口，支持 PCI Express x 1 插槽
- 最多可提供 56/24 个 (PIO-D56/PIO-D24) 数字输出/输入信道
- 双向 I/O 信道可以软件方式设定为 Input/Output 端口
- 内建 I/O line 缓冲区
- 三个 8-bit 埠 (共 24-bit) 可分别规化为 Input/Output
- 可直接连接 DB-24PR, DB-24PD, DB-24RD, DB-24PRD, DB-16P8R, DB-24POR, DB-24SSR, DB-24C 或者其它兼容 OPTO-22 规格的端子板
- 四个中断源: PC0, PC1, PC2, PC3
- 此系列板卡提供有一个 DB37 接头，而 PIO-D56(U)/PEX-D56 多提供了二个 20-pin headers 接头
- 支持 Plug&Play，不需手动设定 I/O 地址以及 IRQ 信道
- 短卡设计

## 1.3 检查产品清单

---

硬纸盒包装内包括以下项目：

- 一张 PIO-D24/D56/D24U/D56U 或 PEX-D24/D56 系列板卡
- 一张 软件安装的 PCI 光盘。
- 一张 快速入门指南。

**建议您先阅读快速入门指南。** 此快速入门指南，包括所有必要和重要的信息如下：

- 从哪里获得软件驱动程序、演示程序和其它资源。
- 如何安装软件。
- 如何测试使用板卡。

### 注意：

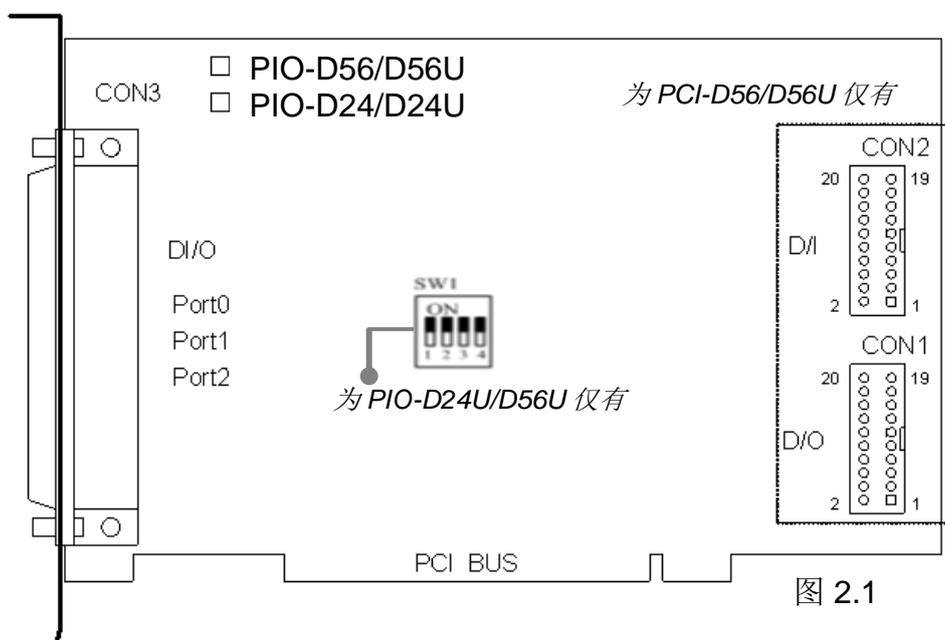
如发现产品包装内的配件有任何损坏或遗失，请保留完整包装盒及配件，尽快联系我们，我们将有专人快速为您服务。

联络方式(E-Mail): [service@icpdas.com](mailto:service@icpdas.com) , [service.icpdas@gmail.com](mailto:service.icpdas@gmail.com)

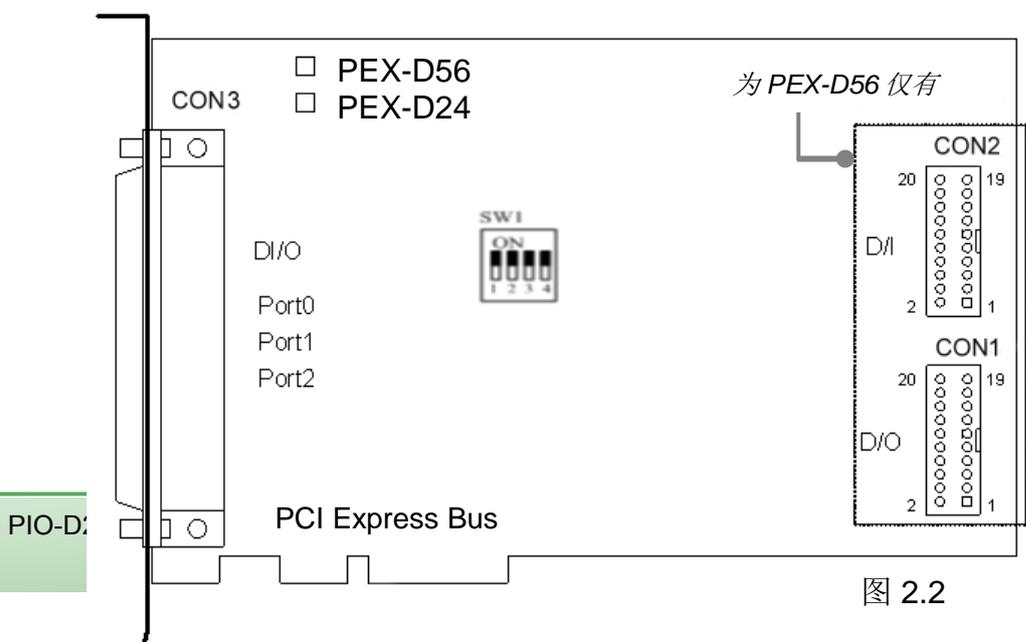
## 2 硬件结构

### 2.1 板卡布局

#### ■ PIO-D24/D24U/D56/D56U 板卡布局



#### ■ PEX-D24/D56 板卡布局



## 2.2 I/O 口位置

PIO-D56/PIO-D24 及 PEX-D24/D56 的 24 位双向口，可以分成 3 个 8 位一组分别为 PA、PB、PC。每个 8 位组可以通过软件设置成 D/I 或 D/O。24 位口当 PC 第一次启动或复位以后所有口被设定为 D/I 口。

I/O 口配置说明：

表 2.1

板卡名称/连接头		PA0 ~ PA7	PB0 ~ PB7	PC0 ~ PC7
PIO-D24(U) PIO-D56(U) PEX-D24 PEX-D56	CON3 (D/I/O)	Port0	Port1	Port2

参考 [章节 2.1](#) 板卡布局和 I/O 口位置

PIO-D56(U)/PEX-D56 还有一个 16 位出入口和一个 16 位输出口。

表 2.2

板卡名称/连接头		Description
PIO-D56(U)	CON1	D/O
PEX-D56	CON2	D/I

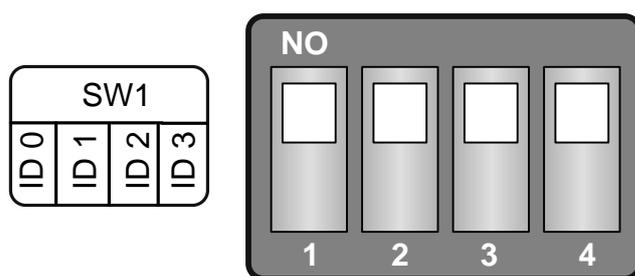
参考 [章节 2.1](#) 板卡布局和 I/O 口位置

※ P2C0, P2C1, P2C2 和 P2C3 能被使用在中断信号源参考 [Sec. 2.5](#)



## 2.3 Card ID 开关

PIO-D24U/D56U 及 PEX-D24/D56 在硬件上新增 Card ID 指拨开关，让使用者可以自由设定每张板卡的识别码。当系统同时使用多张 PIO-D24U/D56U 或 PEX-D24/D56 板卡时，使用者可以迅速而简单区别这些同型号的板卡。出厂预设 Card ID 为 0x0。详细的 SW1 Card ID 设定，请参考至表 2.3。



(预设设定)

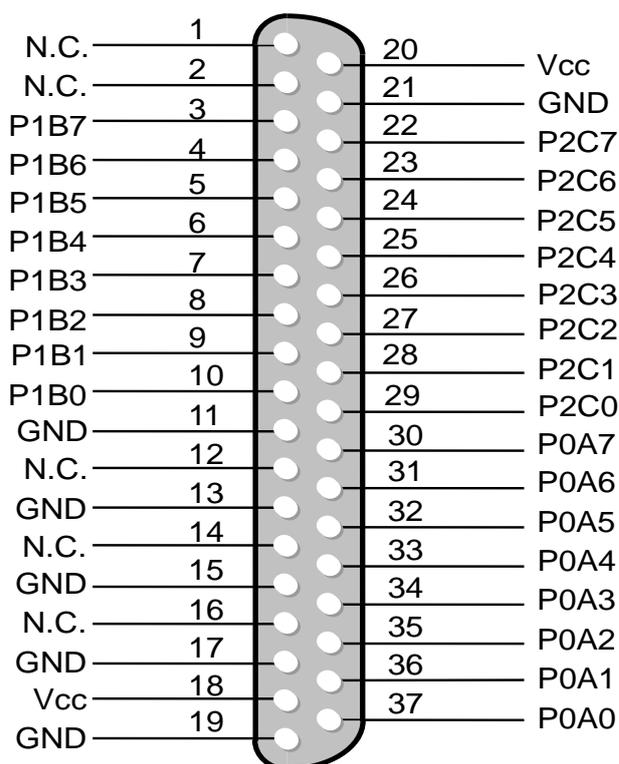
表 2.3 (\*) 预设设定; OFF → 1; ON → 0

Card ID (Hex)	1 ID0	2 ID1	3 ID2	4 ID3
(*) 0x0	ON	ON	ON	ON
0x1	OFF	ON	ON	ON
0x2	ON	OFF	ON	ON
0x3	OFF	OFF	ON	ON
0x4	ON	ON	OFF	ON
0x5	OFF	ON	OFF	ON
0x6	ON	OFF	OFF	ON
0x7	OFF	OFF	OFF	ON
0x8	ON	ON	ON	OFF
0x9	OFF	ON	ON	OFF
0xA	ON	OFF	ON	OFF
0xB	OFF	OFF	ON	OFF
0xC	ON	ON	OFF	OFF
0xD	OFF	ON	OFF	OFF
0xE	ON	OFF	OFF	OFF
0xF	OFF	OFF	OFF	OFF

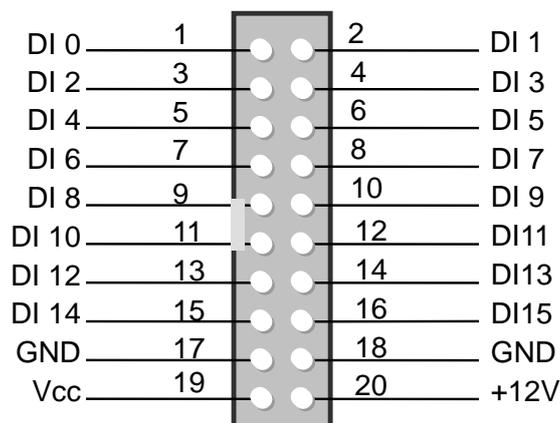
## 2.4 引脚分配

PIO-D56/D24 系列卡引脚分配说明见表 2.4, 2.5, 2.6。每个数字量输入或输出为 TTL 兼容。

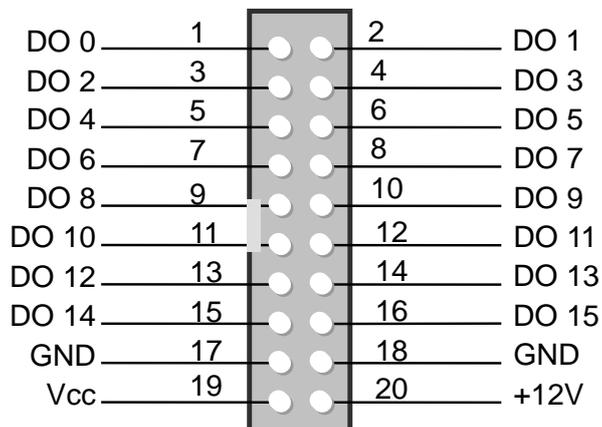
CON3: 37 针 D 型母头连接器



CON2: 20 针公头  
(适用于 PIO-D56(U)/PEX-D56)



CON2: 20 针公头  
(适用于 PIO-D56(U)/PEX-D56)



所有的信号皆符合 TTL 规范	
TTL 高电位 (1)	2.0 ~ 5.0 V(超过 5.0V 后会有设备毁损的风险)
未定义	2.0 V ~ 0.8 V
TTL 低电位 (0)	低于 0.8 V

表 2.4 CON3: 37 针 D 型母头连接器( Port0, Port1, Port2)。

引脚	说明	引脚	说明
1	N. C.	20	Vcc
2	N. C.	21	GND
3	P1B7	22	P2C7
4	P1B6	23	P2C6
5	P1B5	24	P2C5
6	P1B4	25	P2C4
7	P1B3	26	P2C3
8	P1B2	27	P2C2
9	P1B1	28	P2C1
10	P1B0	29	P2C0
11	GND	30	P0A7
12	N.C.	31	P0A6
13	GND	32	P0A5
14	N.C.	33	P0A4
15	GND	34	P0A3
16	N.C.	35	P0A2
17	GND	36	P0A1
18	VCC	37	P0A0
19	GND	XXXXXXX	This pin not available

**表 2.5 CON1 : 20 针公头 (适用于 PIO-D56(U)/PEX-D56)**

引脚	说明	引脚	说明
1	DI0	2	DI1
3	DI2	4	DI3
5	DI4	6	DI5
7	DI6	8	DI7
9	DI8	10	DI9
11	DI10	12	DI11
13	DI12	14	DI13
15	DI14	16	DI15
17	GND	18	GND
19	Vcc	20	+12 V

**表 2.6 CON2 : 20 针公头(适用于 PIO-D56(U)/PEX-D56)**

引脚	说明	引脚	说明
1	DO0	2	DO1
3	DO2	4	DO3
5	DO4	6	DO5
7	DO6	8	DO7
9	DO8	10	DO9
11	DO10	12	DO11
13	DO12	14	DO13
15	DO14	16	DO15
17	GND	18	GND
19	Vcc	20	+12 V

## 2.5 启动 I/O 运行

---

当 PC 运转时，所有数字量 I/O 通道（CON1，CON2，CON3）为禁用。那么可以通过 RESET 去设定数字量 I/O 通道 enabled/disabled。（具体参考 [章节.3.3.1](#)）

所有 D/I/O 口（CON3）上电后状态说明:

- 每个 D/I/O 口状态为禁用.
- D/I/O 口配置成数字量输入口
- D/O 寄存器输出未定义.

用户在使用数字量 I/O 口之前需要做一些初始化工作。参照以下几个步骤:

步骤 1: 找到 PIO/PISO 板卡的地址映像. (参考 [章节.3.1](#))

步骤 2: 击活所有数字量 I/O. (参考 [章节. 3.3.1](#)).

步骤 3: 配置这 3 个口（CON3）成想要的 D/I/O 状态与发送初始化值到每个 D/O 口 (参考 [章节. 3.3.7](#) 和 [章节.3.3.9](#))

※ 数字量 I/O 初始化程序请参考 DEMO1.C

## 2.5.1 DIO 口结构 (CON3)

接头		说明
PIO-D24/D56 系列 PEX-D24/D56	CON3	可编程数字输出/输入

PIO-D56/D24 系列卡数字量 I/O (CON3) 控制结构见 (图 2.3)。数字量控制运行方法如下:

- RESET\ 为 Low-state → 所有 D/I/O 被禁用
- RESET\ 为 High-state → 所有 D/I/O 被激活.
- 如果 D/I/O 是设置成一个 D/I 口 → D/I= 外部输入信号
- 如果 D/I/O 是设置成一个 D/O 口 → D/I = D/O 读回
- 如果 D/I/O 是设置成一个 D/I 口 → 发送数据到一个数字量输入口将只改变 D/O 闭锁寄存器。当端口设置成数字量输出并且正确激活, 闭锁数据将被输出。

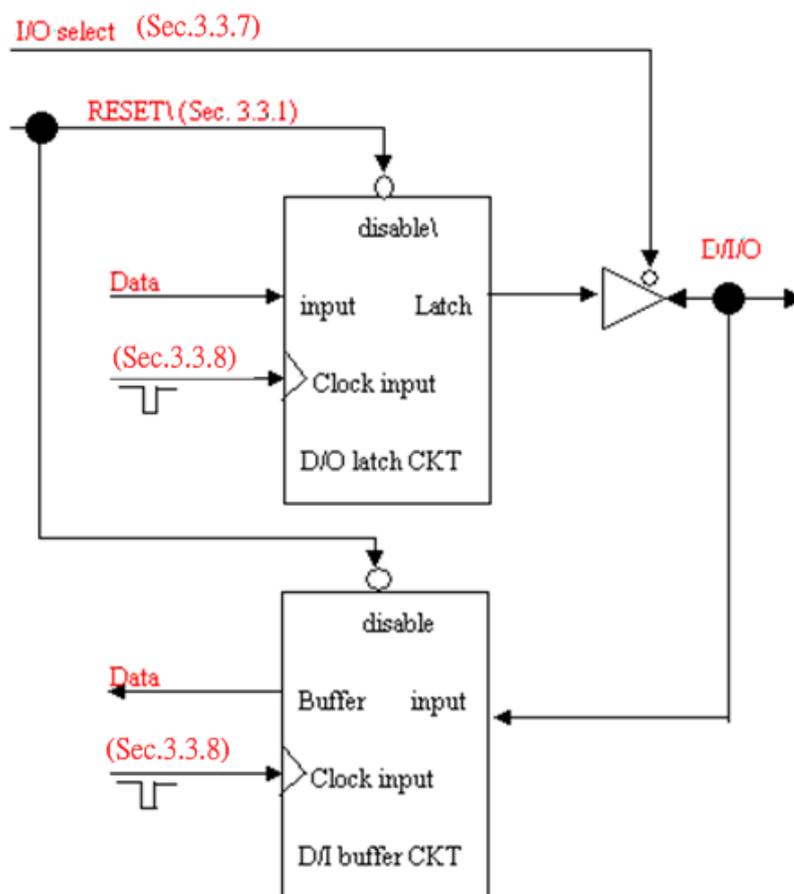


图 2.3 DI/O 口结构(CON3)

## 2.5.2 DI 口结构 (CON2)

接头		说明
PIO-D56/D56U PEX-D56	CON2	数字输入

PIO-D56 系列数字量输入 (CON2) 控制结构见图 2.4。

- RESET $\bar{}$  为 Low-state  $\rightarrow$  所有 DI 为禁用
- RESET $\bar{}$  为 High-state  $\rightarrow$  所有 DI 为激活

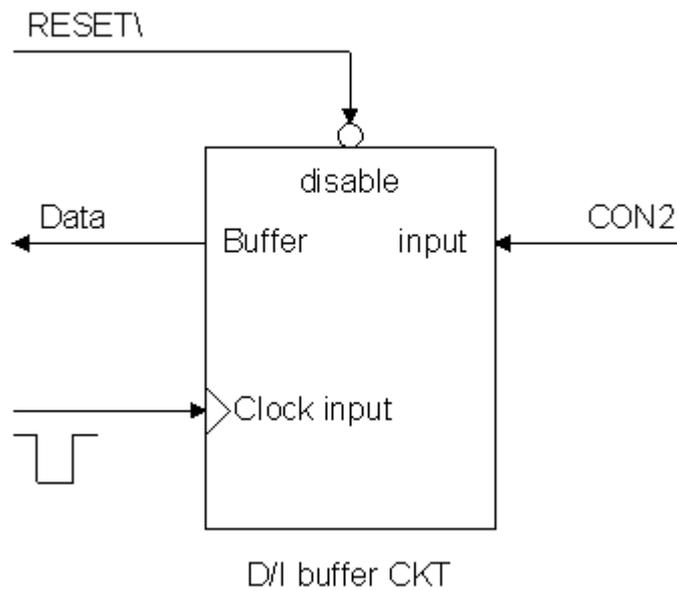


图 2.4 DI 口结构 (CON2)

### 2.5.3 DO 口结构 (CON1)

接头		说明
PIO-D56/D56U PEX-D56	CON1	数字输出

PIO-D56 系列数字量输出（CON1）控制结构见图 2.5。

- RESET $\bar$  为 Low-state  $\rightarrow$  所有 DO 为禁用
- RESET $\bar$  为 High-state  $\rightarrow$  所有 DO 为激活

上电状态如下：

- 所有 DO 运行是禁用
- 所有输出闭锁是清除到 Low-Level

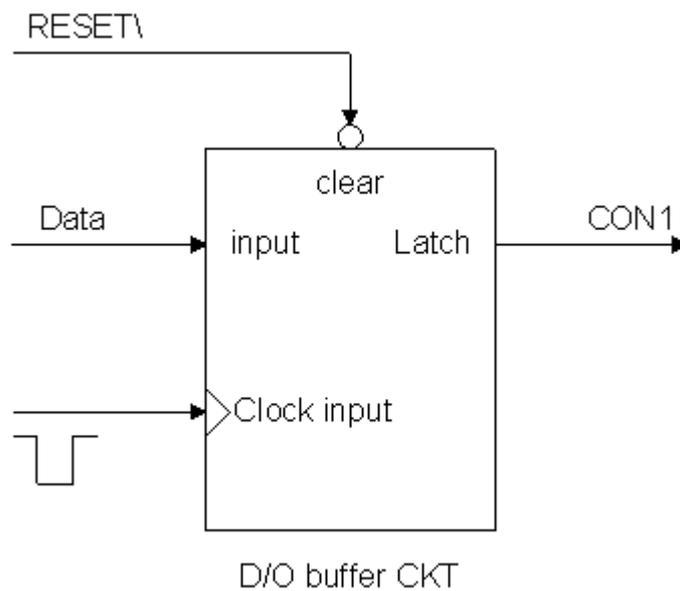


图 2.5 DO 口结构 (CON1)

## 2.6 中断运行

---

P2C0、P2C1、P2C2 和 P2C3 能被使用在中断信号源。参考 [章节.2.1](#)。PIO-D56/PIO-D24 中断是电平触发器和高态有效。中断信号能被编程为正相或反相。设置中断信号源如下：

1. 确定中断源初始成高或低电平
2. 如果初始状态是高电平，请选择反相设定中断信号源。如果初始状态是低电平，请选择正相设定中断信号源([章节. 3.3.6](#))。
3. 允许中断功能([章节. 3.3.4](#))。
4. 如果中断信号是激活，中断服务程序将被启动。

注：DEMO3.C 和 DEMO4.C 演示程序是单个中断源，DEMO5.C 演示程序是 4 个中断源，运行环境为 DOS。如果只有一个中断信号源使用，那么中断服务程序不需要去确定中断源。如果这里是多个或一个中断源，下列方式去确定活跃信号。

1. 读取中断信号源最新状态。
2. 比较新状态与老状态去确定激活信号。
3. 如果 P2C0 是激活的，启用 P2C0 (non-inverter/inverted) 服务程序。
4. 如果 P2C1 是激活的，启用 P2C1 (non-inverter/inverted) 服务程序。
5. 如果 P2C2 是激活的，启用 P2C2 (non-inverter/inverted) 服务程序。
6. 如果 P2C3 是激活的，启用 P2C3 (non-inverter/inverted) 服务程序。
7. 保存当前状态为老状态。

如果中断信号太短，则状态保持原状态不会改变。因此，在中断服务程序被执行之前中断信号需要被激活并维持足够长的时间(Hold time)。不同的操作系统 Hold time 不同，可以从数毫秒到数秒，一般而言 20ms 对大部份操作系统都已足够。

## 2.7 端子板

### 2.7.1 DB-37

DB-37 是一个 37 针的端子板可以很方便的进行接线。

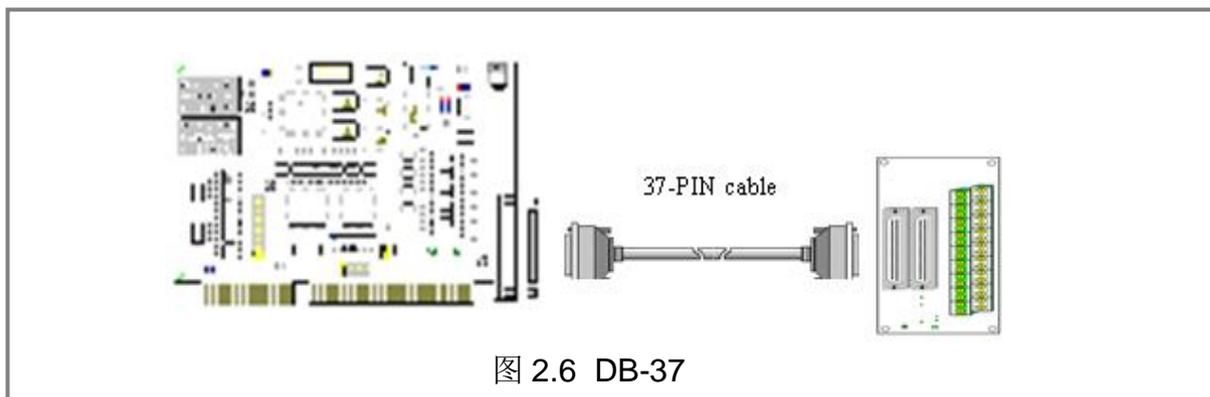


图 2.6 DB-37

### 2.7.2 DN-37 及 DN-50

DN-37 是一个带 DIN 安装导轨和 37 芯 D 型插头的 I/O 接线板。DN-50 是一个带 DIN 安装导轨和 50 针的公头扁平电缆。可以很方便的进行接线。

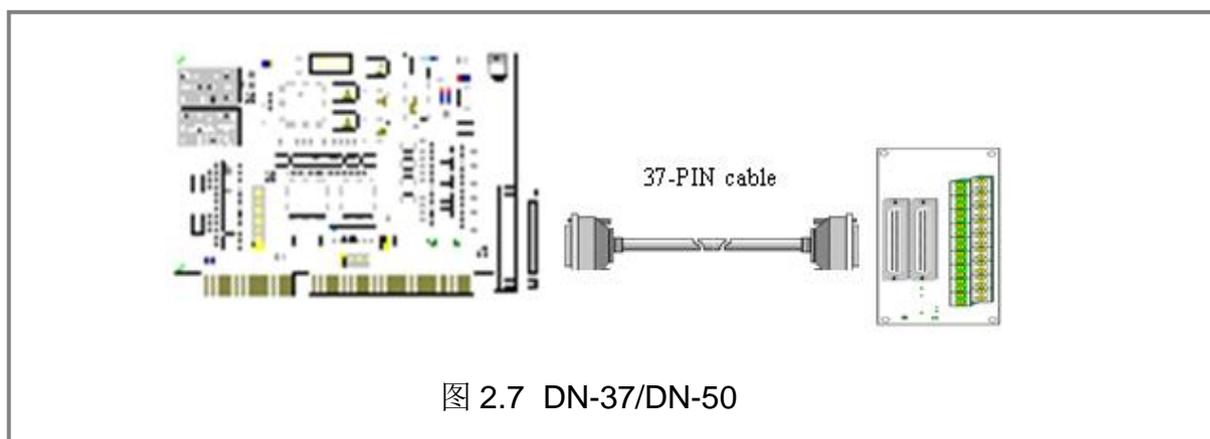
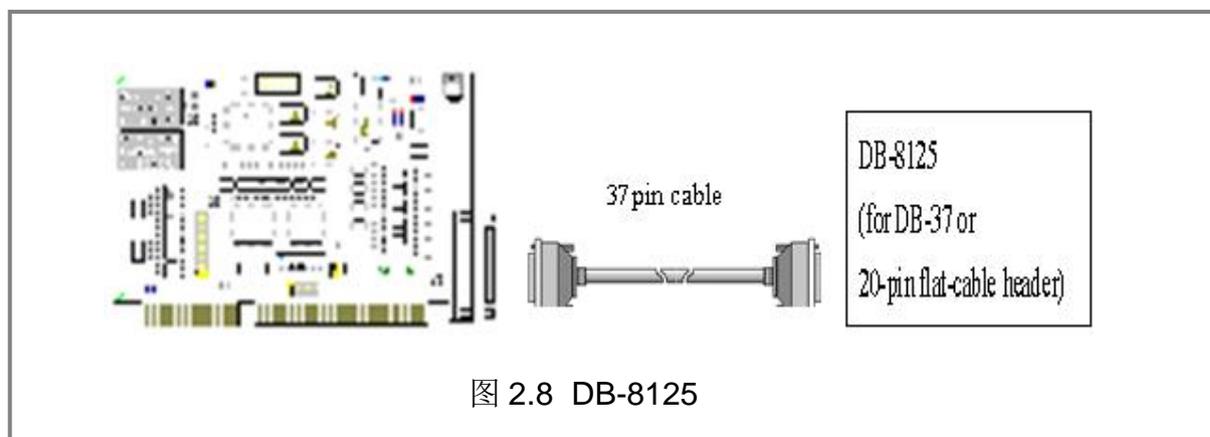


图 2.7 DN-37/DN-50

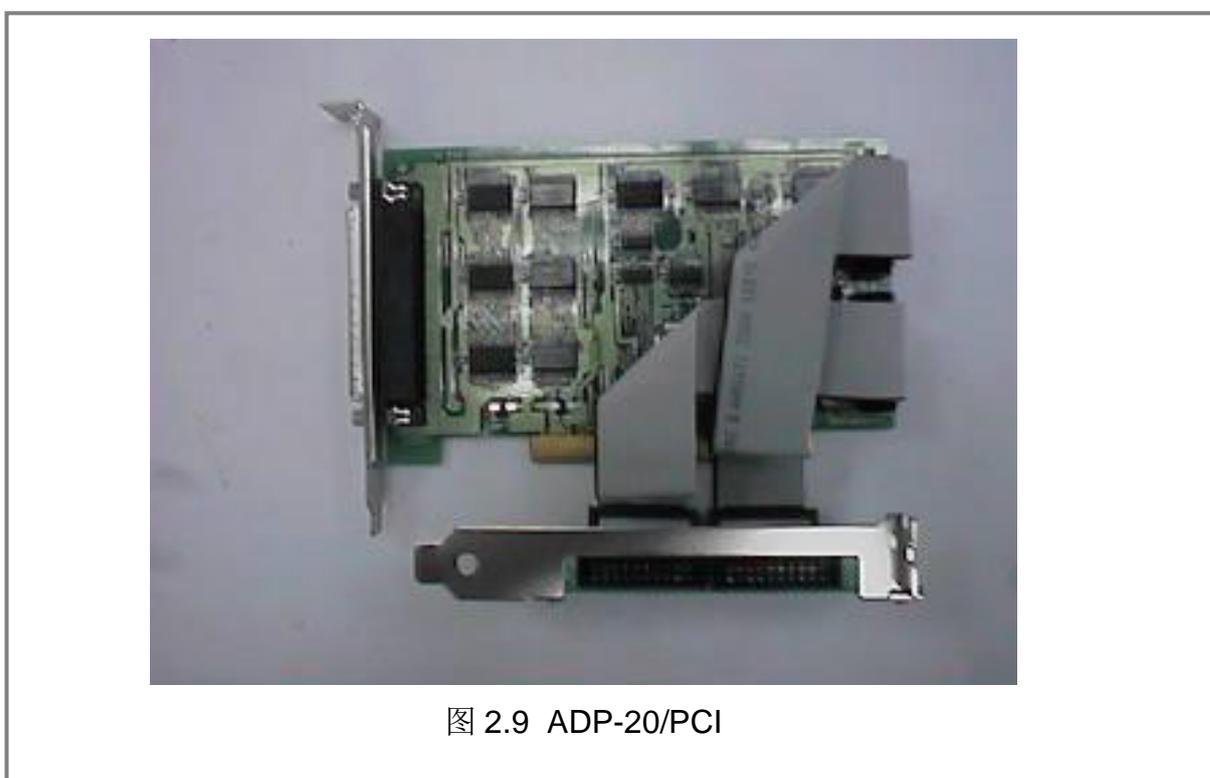
### 2.7.3 DB-8125

DB-8125 是一个带 1 米 D 型接头 37 芯电缆的螺钉端子板。DB8125 由一个 DB-37 和两个 20 芯扁平电缆组成。



### 2.7.4 ADP-20/PCI

ADP-20/PCI 是用于 PCI 总线的 20 针连接器插槽挡板，连接方式见下图：



## 2.7.5 DB-24P, DB-24PD 光电隔离输入端子板

DB-24 是 24 通道隔离数字量输入端子板。DB-24P 的光隔离输入由光电耦合器带一个电流检测用电阻组成。你可以用它来检测从 TTL 电平到 24V 直流信号。也可以用来检测宽范围的交流信号。还可以用此卡来隔离计算机和工业环境中常发生的共模电压，地环流以及电压尖峰。见图 2.10。表 2.7 是用来比较 DB-24P 和 DB-24PD。

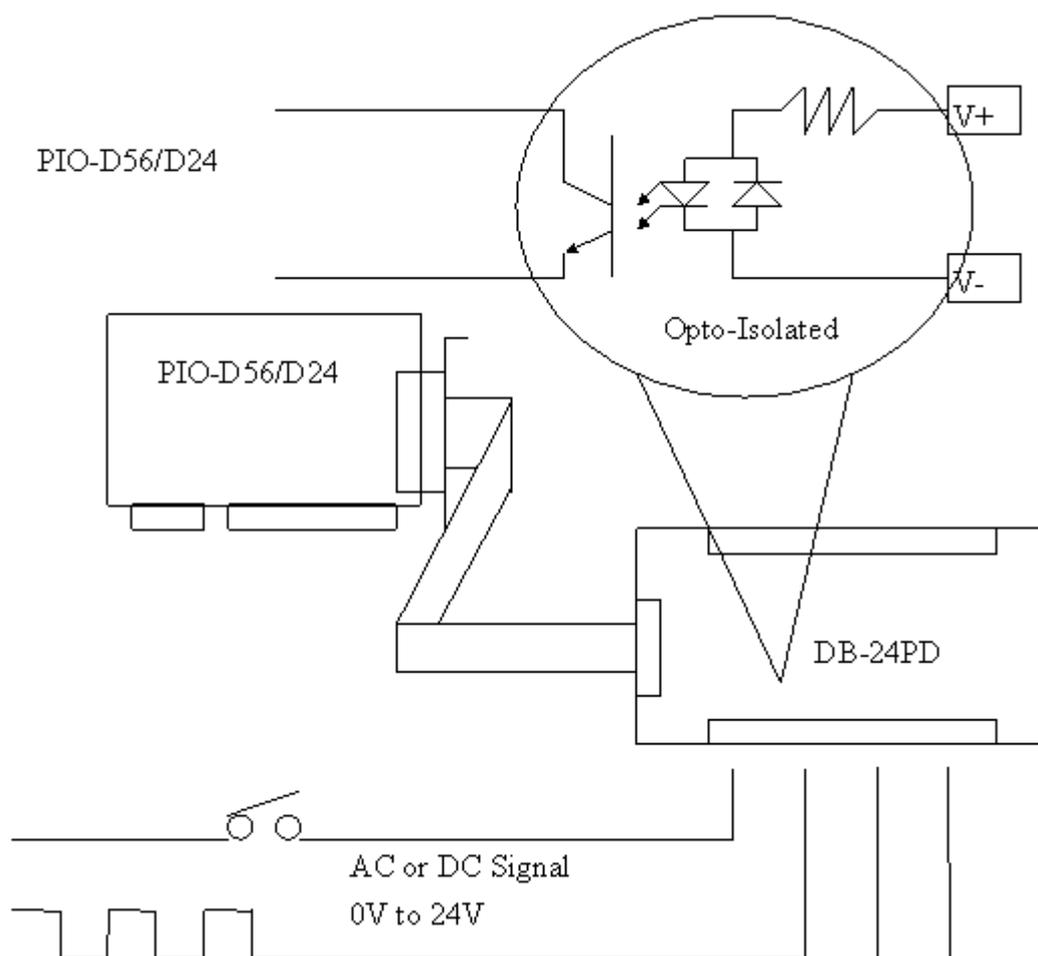


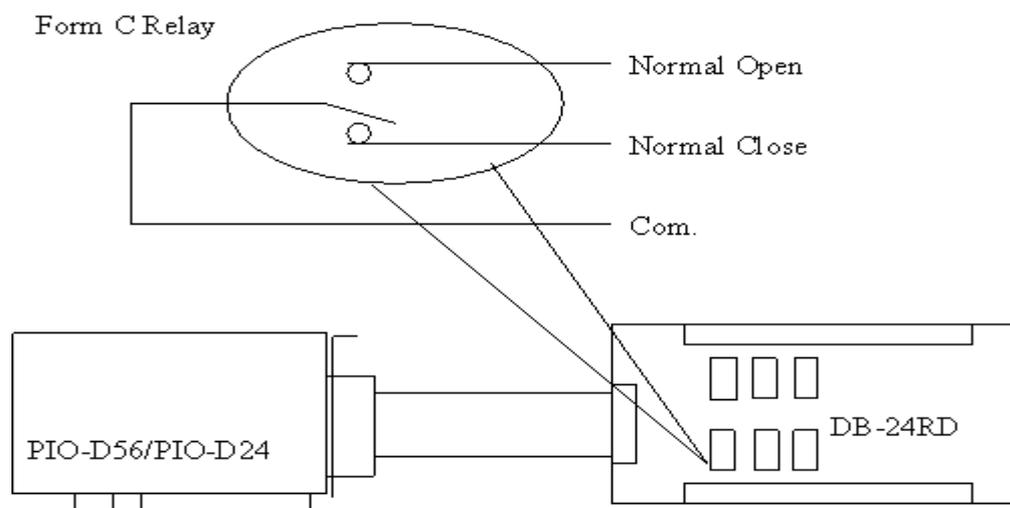
图 2.10

表 2.7

	DB-24P	DB-24PD
50-pin flat-cable header	Yes	Yes
D-sub 37-pin header	No	Yes
Other specifications	Same	

## 2.7.6 DB-24R, DB-24RD 继电器输入端子板

DB-24R 有 24 个 C 型继电器，通过可编程使得机电式继电器可直接通断负载。每路继电器可控制一个 0.5A/110V 或 1A/24Vdc 的负载。通过 50 针与 OPTO-22 兼容的连接器或 20 针扁平电缆线连接器，给对应通道的功率继电器加上 5V 电压信号来激活其工作。每个通道一个 LED，共有 24 个高亮度 LED，当与之关联的继电器接通时发亮。为避免你的 PC 过载，本板卡需要一个+12Vdc 或+24Vdc 的外界电源供电。见图 2.11。



Note:

Channel : 24 Form C Relay

Relay : Switch up to 0.5A at 110ACV  
or 1A at 24DCV

图 2.11

表 2.8

	DB-24R	DB-24RD
50-pin flat-cable header	Yes	Yes
D-sub 37-pin header	No	Yes
Other specifications	Same	

表 2.9

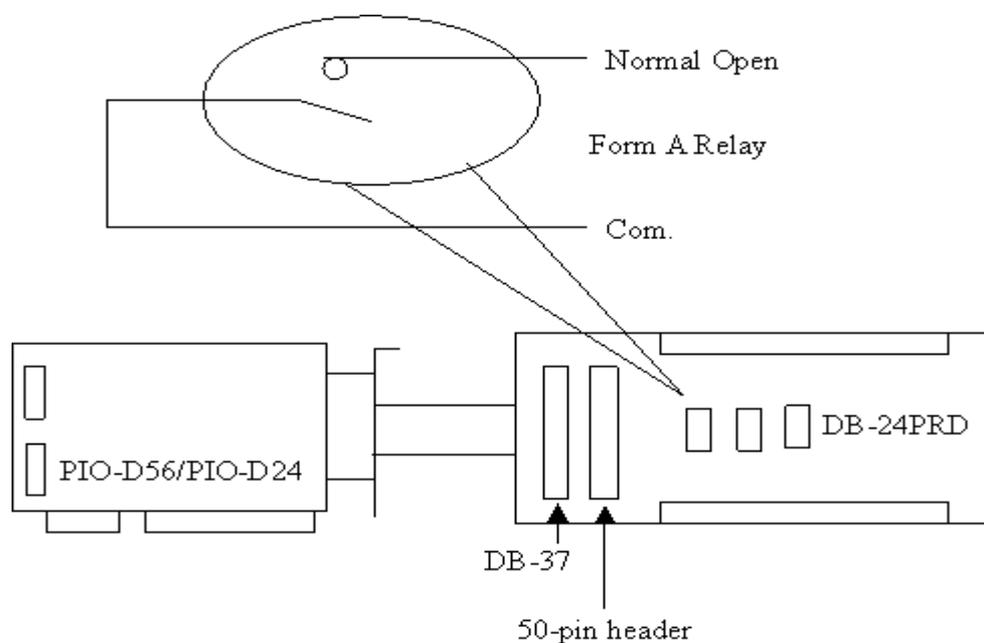
DB-24R, DB-24RD	24 × Relay (120 V, 0.5 A)
DB-24PR, DB-24PRD	24 × Power Relay (250 V, 5 A)
DB-24POR	24 × Photo MOS Relay (350 V, 0.1 A)
DB-24SSR	24 × SSR (250 V <sub>AC</sub> , 4 A)
DB-24C	24 × O.C. (30 V, 100 mA)
DB-16P8R	16 × Relay (120 V, 0.5 A) + 8 × isolated input

## 2.7.7 DB-24PR, DB-24POR, DB-24C

表 2.10

DB-24PR	24 × Power relay, 5 A/250 V
DB-24POR	24 × Photo MOS relay, 0.1 A/350 V <sub>AC</sub>
DB-24C	24 × Open Collector, 100 mA per channel, 30 V max.

DB-24PR 是 24 通道功率继电器板，它有可编程控制的 8 个 C 型继电器和 16 个 A 型继电器，用来直接通断负载。每个继电器可以控制一个 250V<sub>AC</sub> 或 30V<sub>DC</sub> 的 5A 负载。通过 50 针与 OPTO-22 兼容的连接器或 20 针扁平电缆线连接器，给适当通道的功率继电器加上 5V 电压信号来激活。每个通道具有一个 LED 指示，共有 24 个高亮度的 LED，当与之关联的继电器接通时发亮。为避免 PC 过载，本板卡需要一个 +12V<sub>DC</sub> 或 +24V<sub>DC</sub> 的电源供应。见图 2.12。



见图 2.12

### 注意:

1. 50 针连接器（OPTO-22 兼容）有 DIO-24, DIO-48, DIO-144, PIO-D144, PIO-D96, PIO-D56, PIO-D48, PIO-D24, PIO-D168
2. 16 通道数字量输出 20 针连接器有 A-82X, A-62X, DIO-64, ISO-DA16/DA8。
3. 通道: 16 通道 A 型继电器, 8 通道 C 型继电器。
4. 继电器: switching up to 5A at 110V<sub>AC</sub> / 5A at 30V<sub>DC</sub>.

## 2.7.8 端子板对照表

表 2.11

	20-pin flat-cable	50-pin flat-cable	D-sub 37-pin
DB-37	No	No	Yes
DN-37	No	No	Yes
ADP-37/PCI	No	Yes	Yes
ADP-50/PCI	No	Yes	No
DB-24P	No	Yes	No
DB-24PD	No	Yes	Yes
DB-16P8R	No	Yes	Yes
DB-24R	No	Yes	No
DB-24RD	No	Yes	Yes
DB-24C	Yes	Yes	Yes
DB-24PRD	No	Yes	Yes
DB-24POR	Yes	Yes	Yes
DB-24SSR	No	Yes	Yes

注意：PIO-D56/PIO-D24 没有 50 针扁平电缆。PIO-D56/PIO-D24 有一个 DB-37 连接器，PIO-D56 还有两个 20 针扁平电缆。

## 3 I/O 控制寄存器



### 3.1 如何找到 I/O 地址

电源开启后即查即用 BIOS 分配适当的一个 I/O 地址到每个 PIO/PISO 板卡，PIO-D56/D24 系列板卡的 ID 如下：

PIO-D24U/D56U 及 PEX-D24/D56	
<b>Rev 1.0 或更新版本</b>	
Vendor ID	0xE159
Device ID	0x0001
Sub-vendor ID	0x8080 0xC080
Sub-device ID	0x01
Sub-aux ID	0x40

PIO-D24			
Rev 1.0~Rev 5.0		Rev 6.0 或更新版本	
Vendor ID	0xE159	Vendor ID	0xE159
Device ID	0x0002	Device ID	0x0001
Sub-vendor ID	0x80	Sub-vendor ID	0x8080 0xC080
Sub-device ID	0x01	Sub-device ID	0x01
Sub-aux ID	0x40	Sub-aux ID	0x40

PIO-D56			
<Rev 1.0 ~ Rev 4.0>		Rev 5.0 或更新版本	
Vendor ID	0xE159	Vendor ID	0xE159
Device ID	0x0002	Device ID	0x0001
Sub-vendor ID	0x80	Sub-vendor ID	0x8080 0xC080
Sub-device ID	0x01	Sub-device ID	0x01
Sub-aux ID	0x40	Sub-aux ID	0x40

程序 PIO\_PISO.EXE 将找到 PC 中安装的 PIO/PISO 板卡并呈现所有的信息。通过 ICPDAS 数据采集卡的 sub\_vender, sub\_device 和 sub\_Aux ID 来确定 PIO 板卡，相关内容对照下图和表 3-1。

PIO\_PISO.exe 工具程序下载位置:

CD:\NAPDOS\PCI\Utility\Win32\PIO\_PISO\

[http://ftp.icpdas.com/pub/cd/iocard/pci/napdos/pci/utility/win32/pio\\_piso/](http://ftp.icpdas.com/pub/cd/iocard/pci/napdos/pci/utility/win32/pio_piso/)

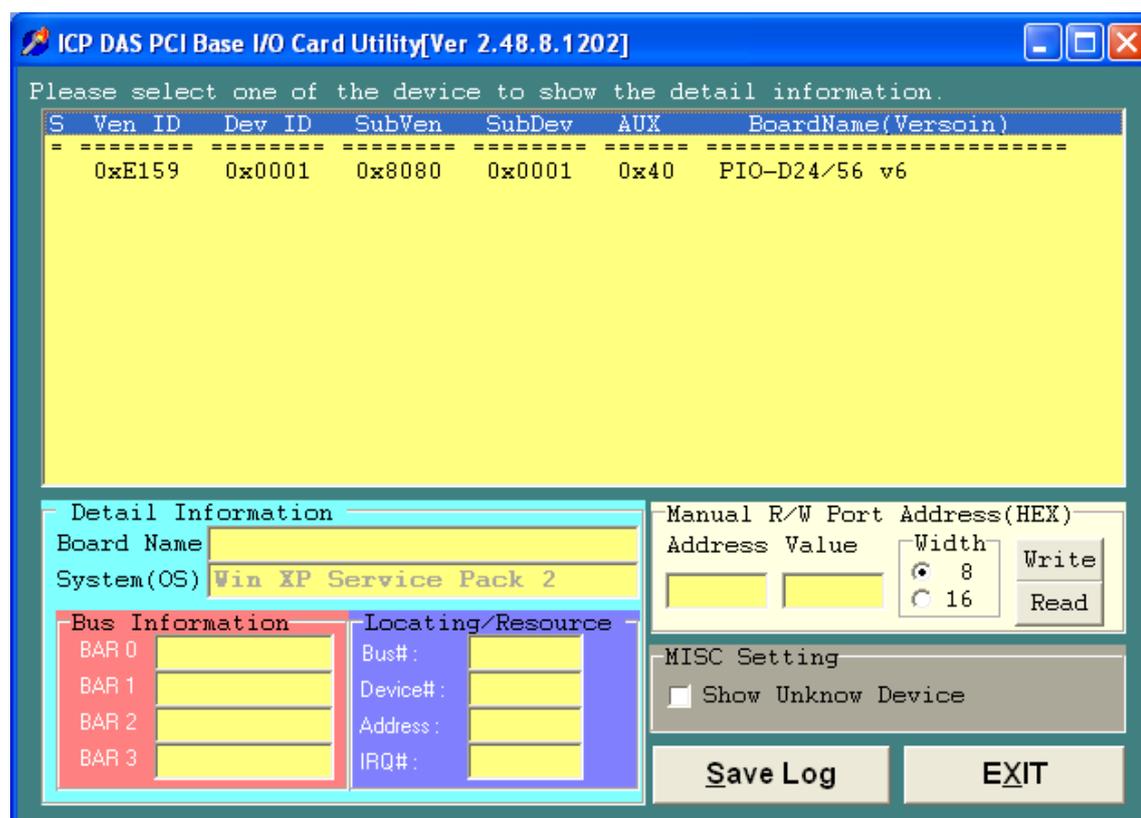


图 3.1

表 3-1

PIO/PISO 系列卡	说明	Sub_Vendor	Sub_Device	Sub_AUX
PIO-D168	168 * DIO	9880	01	50
PIO-D168A	168 * DIO	80	01	50
PIO-D144(REV4.0)	144 * D/I/O	80 (5C80)	01	00
PIO-D96 (REV4.0)	96 * D/I/O	80 (5880)	01	10
PIO-D64 (REV2.0)	64 * D/I/O	80 (4080)	01	20
PIO-D56 (REV5.0)	24 * D/I/O + 16 * D/I+16*D/O	80 (8080)	01	40
PIO-D48 (REV2.0)	48 * D/I/O	80 (0080)	01	30
PIO-D24 (REV6.0)	24 * D/I/O	80 (8080)	01	40
PIO-821	Multi-function	80	03	10
PIO-DA16	16 * D/A	80	04	00
PIO-DA8	8 * D/A	80	04	00
PIO-DA4	4 * D/A	80	04	00
PISO-C64	64 * isolated D/O (Current sinking)	80	08	00
PISO-A64	64 * isolated D/O (Current sourcing)	80	08	50
PISO-P64	64 * isolated D/I	80	08	10
PISO-P32C32	32* isolated D/O (Current sinking) + 32* isolated D/I	80	08	20
PISO-P32A32	32*isolated DO (Current sourcing) + 32* isolated D/I	80	08	70
PISO-P8R8	8* isolated D/I + 8 * 220 V relay	80	08	30
PISO-P8SSR8AC	8* isolated D/I + 8 * SSR /AC	80	08	30
PISO-P8SSR8DC	8* isolated D/I + 8 * SSR /DC	80	08	30
PISO-730	16*DI + 16*D/O + 16* isolated D/I + 16*isolated D/O (Current sinking)	80	08	40
PISO-730A	16*DI + 16*D/O + 16* isolated D/I + 16*isolated D/O (Current sourcing)	80	08	80
PISO-813	32 * isolated A/D	80	0A	00
PISO-DA2	2 * isolated D/A	80	0B	00



注：如果你的板卡版本不同，它可能有不同的 **Sub ID**。可是不会出现使用问题，对于不同版本的板卡我们会提供同样的功能调用。

## 3.2 分配 I/O 地址

---

即插即用 BIOS 将分配 PIO/PISO 板卡适当的 I/O 地址。假如只有一块 PIO/PISO 卡，用户能够确定这块卡为 card\_0。如果有两个 PIO/PISO 卡在系统中，用户将很难找到哪一块卡是 card\_0。软件驱动最多能够支持 16 块卡。所以用户可以安装 16 块 PIO/PISO 板卡在一台 PC 中。下面的方法说明查找和确定 card\_0 和 card\_1:

下面的方式是使用 **wSlotBus & wSlotDevice** 来简单的识别 **card\_0**:

步骤 1: 移除 PC 中所有 PIO-D56/D24 系列卡。

步骤 2: 安装一块 PIO-D56/D24 系列卡到 PC 的第一个 PCI/PCI Express 插槽，运行 PIO\_PISO.EXE。然后记录 wSlotBus1 和 wSlotDevice1 的信息。

步骤 3: 移除 PC 中所有 PIO-D56/D24 系列卡。

步骤 4: 安装一块 PIO-D56/D24 系列在 PC 的第二个插槽并运行 PIO\_PISO.EXE。然后记录 wSlotBus2 和 wSlotDevice2 的信息。

步骤 5: 重复步骤 3 和 4 到每个 PCI/PCI Express 插槽并记录所有 wSlotBus 和 wSlotDevice 信息。

记录的信息也许与下表相似：

表 3-2

PC's PCI slot	wSlotBus	wSlotDevice
Slot_1	0	0x07
Slot_2	0	0x08
Slot_3	0	0x09
Slot_4	0	0x0A
PCI-BRIDGE		
Slot_5	1	0x0A
Slot_6	1	0x08
Slot_7	1	0x09
Slot_8	1	0x07

上面所记录的是在一台 PC 机上的 wSlotBus 和 wSlotDevice 信息。这些值将被映射到 PC 的物理插槽。任何 PIO/PISO 卡的映射将不能被改变。因此，下面三个步骤就能够使用这个信息确定 PIO/PISO 卡：

步骤 1: 利用表 3-2 wSlotBus 和 wSlotDevice 信息

步骤 2: 输入板卡号到函数 PIO\_GetConfigAddressSpace(...)去获得板卡的详细信息，尤其是 wSlotBus 和 wSlotDevice 信息。

步骤 3: 在步骤 1 和步骤 2 中用户能够确定一个的 PIO/PISO 卡详细信息通过 wSlotBus 及 wSlotDevice 去比较数据。



注：通常 PIO/PISO 卡安装在插槽 0 就是 card0，安装在插槽 1 就是 card1。

## 3.3 I/O 地址映像

通过主板 ROM BIOS 去自动分配 PIO/PISO 板卡的 I/O 地址。用户同样可以再分配 I/O 地址。强烈推荐用户不要去改变 I/O 地址，即插即用 BIOS 将会很好的去分配每个 PIO/PISO 板卡的 I/O 地址。PIO-D56/D24 的 I/O 地址见下表，

表 3-3

地址	Read/读	Write/写
Wbase+0	保留	RESET\ 控制寄存器
Wbase+2	保留	Aux 控制寄存器
Wbase+3	Aux 数据寄存器	Aux 数据寄存器
Wbase+5	保留	INT 屏蔽控制寄存器
Wbase+7	Aux 引脚状态寄存器	保留
Wbase+0x2a	保留	INT 极性控制寄存器
Wbase+0xc0	读 Port0	写 Port0
Wbase+0xc4	读 Port1	写 Port1
Wbase+0xc8	读 Port2	写 Port2
Wbase+0xcc	读 Card ID	配置 Port0~Port2
Wbase+0xd0	读 CON2 Low byte (为 PIO-D56(U)和 PEX-D56 仅有)	写 CON1 Low byte (为 PIO-D56(U)和 PEX-D56 仅有)
Wbase+0xd4	读 CON2 high byte (为 PIO-D56(U)和 PEX-D56 仅有)	写 CON1 high byte (为 PIO-D56(U)和 PEX-D56 仅有)

注: Wbase+0xd0 和 Wbase+0xd4 为 PIO-D56(U) 及 PEX-D56 系列卡仅有。

---

### 3.3.1 RESET\ 控制寄存器

(Write): wBase+0

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Reserved	RESET\						

注. 详细关于 wBase 请参考至 [章节 3.1](#)。

当 PC 的电源第一次开启, RESET\ 信号是低电平状态。这将禁用所有 D/I/O 运行。在所有的 D/I/O 命令应用开始之前用户去设定 RESET\ 信号为高电平状态。

```
outportb (wBase,1);      /* RESET\=High → 所有 D/I/O 激活 */
outportb (wBase,0);      /* RESET\=Low → 所有 D/I/O 禁用 */
```

---

### 3.3.2 AUX 控制寄存器

(Write): wBase+2

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Aux7	Aux6	Aux5	Aux4	Aux3	Aux2	Aux1	Aux0

注. 详细关于 wBase 请参考至 [章节 3.1](#)。

Aux?=0 → 表明 Aux 用作 D/I

Aux?=1 → 表明 Aux 用作 D/O

当 PC 第一次上电启动时, 所有 Aux 信号默认为低电平, 且所有 PIO/PISO 系列板卡 I/O 口为数字量输入状态。

---

### 3.3.3 AUX 数据寄存器

(Read/Write): wBase+3

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Aux7	Aux6	Aux5	Aux4	Aux3	Aux2	Aux1	Aux0

注. 详细关于 wBase 请参考至 [章节 3.1](#)。

当 Aux?用于 D/O 时, 输出状态由寄存器控制。由于寄存器设计结构特点所至, 因此这个寄存器为保留不使用。

### 3.3.4 INT Mask 控制寄存器

(Write): wBase+5

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	EN3	EN2	EN1	EN0

注. 详细关于 wBase 请参考至 [章节 3.1](#)。

EN0=0→ P2C0 中断信号禁止 (默认)

EN0=1→ P2C0 中断信号激活

```
outportb(wBase+5,0);           /* 禁止中断 */
outportb(wBase+5,1);           /* 激活中断 P2C0 */
outportb(wBase+5,0x0f);        /* 激活中断 P2C0,P2C1,P2C2,P2C3 */
```

### 3.3.5 Aux 状态寄存器

(Read): wBase+7

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Aux7	Aux6	Aux5	Aux4	Aux3	Aux2	Aux1	Aux0

注. 详细关于 wBase 请参考至 [章节 3.1](#)。

Aux0=P2C0, Aux1=P2C1, Aux2=P2C2, Aux3=P2C3, Aux7~4=Aux-ID. Aux 0~3 连结到上述四个中断源, 中断服务程序必须读这个寄存器来确定中断源。

更多信息参考 [章节 2.5](#) 与 DEMO5.C。

### 3.3.6 Read Card ID 寄存器

(Read): wBase+0xcc

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	ID3	ID2	ID1	ID0	Port2	Port1	Port0

注. 详细关于 wBase 请参考至 [章节 3.1](#)。

```
wCardID = inportb(wBase+0xcc) >>3; /* 读取 Card ID */
```



注: 仅有 PIO-D24U/D56U 及 PEX-D24/D56 (1.0 版或更新版本) 支持 Card ID 功能。

### 3.3.7 I/O 选择控制寄存器

(Write): wBase+0xcc

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	ID3	ID2	ID1	ID0	Port2	Port1	Port0

注. 详细关于 wBase 请参考至 [章节 3.1](#)。

寄存器提供函数去配置 PIO-D56/D24 系列卡数字量输入/输出口。每个 I/O 口都能被程序设定为 D/I 或 D/O 口。注. 当 PC 第一次运行的时候所有端口为 D/I。

Port? = 1 → 该口为 D/O 口

Port? = 0 → 该口为 D/I 口

```
outputb(wBase+0xcc,0x00); /* 设定 Port0/1/2 为 D/I 口 */
outputb(wBase+0xcc,0x04); /* 设定 Port0/1 为 D/I 口 */
                          /* 设定 Port2 为 D/O 口 */
```

### 3.3.8 中断极性控制寄存器

(Write): wBase+0x2a

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	INV3	INV2	INV1	INV0

注. 详细关于 wBase 请参考至 [章节 3.1](#)。

这个寄存器决定是否要将中断信号源反相处理。下列为应用示例：

INV0=1 → 选择 P2C0 信号不反相处理 (正相输入)

INV0=0 → 选择 P2C0 信号反相处理 (反相输入)

```
outputb(wBase+0x2a,0x0f); /* 选择所有中断源信号正相输入 P2C0/1/2/3 */
outputb(wBase+0x2a,0x00); /* 选择所有中断源信号反相输入 P2C0/1/2/3 */
outputb(wBase+0x2a,0x0e); /* 选择反相输入 P2C0 */
                          /* 选择正相输入 P2C1/2/3 */
outputb(wBase+0x2a,0x0c); /* 选择反相输入 P2C0/1 */
                          /* 选择正相输入 P2C2/3 */
```

更多信息参考 [章节 2.5](#) 与 DEMO5.C。

### 3.3.9 读/写 8-bit 数据寄存器

(Read/Write):wBase+0xc0/0xc4/0xc8/0xd0/0xd4

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

注. 详细关于 wBase 请参考至 [章节 3.1](#)。

PIO-D56/PIO-D24 系列卡有 5/3 8 位 I/O 口。通过访问自己的数据寄存器每个口可以很方便的去读/写。

```

outputb(wBase+0xc0,Val);          /* 写入 port-0          */
Val=inportb(wBase+0xc0);         /* 读取 port-0          */

outputb(wBase+0xcc,0x07);        /* 设定 Port0~Port2 为 DO 口 */
outputb(wBase+0xc0,i1);          /* 写入 Port0            */
outputb(wBase+0xc4,i2);          /* 写入 Port1            */
outputb(wBase+0xc8,i3);          /* 写入 Port2            */

outputb(wBase+0xcc,0x01);        /* 设定 Port0 为 DO 口    */
                                /* Port1~Port2 为 DI 口  */
outputb(wBase+0xc0,i1);          /* 写入 Port0            */
j2=inportb(wBase+0xc4);          /* 读取 Port1            */
j3=inportb(wBase+0xc8);          /* 读取 Port2            */

L=inportb(wBase+0xd0);           /* 读取 CON2 低字节      */
H=inportb(wBase+0xd4);           /* 读取 CON2 高字节      */
Val=(H<<8)+L ;                  /* Val 是 16 位数据      */

outputb(wBase+0xd0,Val);         /* 写入 CON1 低字节      */
outputb(wBase+0xd4,(Val>>8));    /* 写入 CON1 高字节      */

```

注：在读/写数据寄存器(wBase+0xc0/0xc4/0xc8)之前，确定 I/O 口配置(D/I 或 D/O)。(更多信息参考 [章节 3.3.7](#) 和表 3.3)

## 4 软件安装向导

PIO-D56/D24 系列板卡支持在 DOS、Windows 98/NT/2000、32-bit 及 64-bit Windows XP/2003/Vista/2008/7 等操作系统环境下使用。软体安装程序参考本手册中第 4.1 节 ~ 第 4.3 节，或参考快速入门指南(CD:\NAPDOS\PCI\PIO-DIO\Manual\QuickStart\)

<http://ftp.icpdas.com/pub/cd/iocard/pci/napdos/pci/pio-dio/manual/quickstart/>

### 4.1 软件驱动程序安装

1. 依照您的 PC 平台选择适合的驱动程序来安装。

■ UniDAQ SDK 驱动程序 (支持 32-bit/64-bit Windows XP/2003/Vista/7):

**步骤 1:** 将软件安装 PCI 光盘放入 CD-ROM 光驱中，PCI 光盘安装程序将自动启动。如 PCI 光盘安装程序无法自动启动，请以手动方式来启动: 1. 进入 PCI 光盘中的 NAPDOS 文件夹里。2. 双击 NAPDOS 文件夹中的 **AUTO32.EXE** 执行档。



**步骤 2:** 单击“PCI Bus DAQ Card”项目。

**步骤 3:** 单击“UniDAQ”项目。

**步骤 4:** 单击“DLL for Windows 2000 and XP/2003/Vista 32-bit”项目。

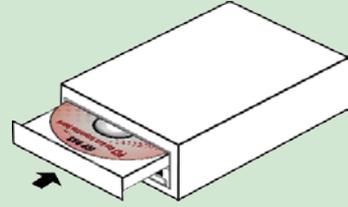


**步骤 5:** 双击 Driver 文件夹中的“UniDAQ\_Win\_Setup\_x.x.x.x.\_xxxx.exe”执行档。



■ **PIO-DIO Series 驱动程序 (支持 Windows 98/NT/2K 及 32-bit Windows XP/2003/Vista/7):**

**步骤 1:** 将软件安装 PCI 光盘放入 CD-ROM 光驱中, PCI 光盘安装程序将自动启动。如 PCI 光盘安装程序无法自动启动, 请以手动方式来启动: 1. 进入 PCI 光盘中 NAPDOS 文件夹里。2. 双击 NAPDOS 文件夹中的 AUTO32.EXE 执行档。



**步骤 2:** 单击 “PCI Bus DAQ Card” 项目。

**步骤 3:** 单击 “PIO-DIO” 项目。

**步骤 4:** 单击 “DLL and OCX for Windows 98/NT/2K/XP/2003” 项目。



**步骤 5:** 在 Driver 文件夹中, 双击 “PIO\_DIO\_Win\_vxxx.exe” 执行档来安装。



2. 双击安装执行档后, 将开始安装驱动程序并复制相关文件 (”.DLL”, ”.SYS”, ”.Vxd”) 到指定的目录下及注册驱动程序至您的 PC 上, 目录路径如下所示:

■ **64-bit Windows XP/2003/Vista/7:**

UniDAQ.DLL 文件将被复制到 C:\WINNT\SYSTEM32 文件夹下

NAPWNT.SYS 及 UniDAQ.SYS 文件将被复制到  
C:\WINNT\SYSTEM32\DRIVERS 文件夹下

更多更详细的 UniDAQ.DLL 函式信息, 请参考到 UniDAQ SDK 用户手册。  
(CD:\NAPDOS\PCI\UniDAQ\Manual\).

<http://ftp.icpdas.com/pub/cd/iocard/pci/napdos/pci/unidaq/maunal/>

■ **Windows NT/2K 和 32-bit Windows XP/2003/Vista/7:**

PIODIO.DLL 文件将被复制到 C:\WINNT\SYSTEM32 文件夹下

NAPWNT.SYS 及 PIO.SYS 文件将被复制到

C:\WINNT\SYSTEM32\DRIVERS 文件夹下

■ **Windows 95/98/ME:**

PIODIO.DLL 及 PIODIO.Vxd 文件将被复制到

C:\Windows\SYSTEM 文件夹下

更多更详细的 PIODIO.DLL 函式信息，请参考到 PIO-DIO 系列 DLL 软件手册。

(CD:\NAPDOS\PCI\PIO-DIO\Manual\).

<http://ftp.icpdas.com/pub/cd/iocard/pci/napdos/pci/pio-dio/manual/>

## 4.2 PnP 驱动程序安装

---

以上软件驱动程序安装完成后，请关闭您 PC 的电源，并安装 PIO-D56/D24 系列板卡到 PC 的插槽上。板卡安装完成后，并启动 PC 电源，当进入 Windows 系统后，Plug&Play 驱动程序会自动执行，板卡即可使用。如 Plug&Play 发生问题造成无法自动执行，请以手动方式来安装，请参考到 PnPInstall.pdf 。

## 4.3 确认板卡安装成功

请到装置管理员中来确认您的 PIO-D56/D24 系列板卡已正确的安装到 PC 中，开启装置管理员步骤如下：

步骤 1: 选择「开始」→「设定(S)」→「控制台(C)」后，  
双击控制台窗口中的「系统」。



步骤 2: 在系统内容窗口中，选择「硬件」卷标，并单击「装置管理员(D)」按钮来进入装置管理员中。

步骤 3: 检查 PIO-D56/D24 系列板卡是否正确安装，如已安装完成，装置管理员中将显示 PIO-D56/D24 板卡名称于 DAQCard 项目下，如下图所示：



## 5 示例程序



### 5.1 Windows DEMO 程序

如果 DLL 驱动没有正确安装那么所有 DEMO 程序将不能正常工作。在 DLL 驱动安装过程的时候，安装程序将注册适当的内核驱动到操作系统中，并且拷贝 DLL 驱动和 DEMO 程序到适当的位置，你可以选择(Win98/Me/NT/2000 and 32-bit Win XP/2003/Visa/7)驱动软件包。一次完整的驱动安装，下列出现相关的 DEMO 程序、库文件、声明的头文件在不同的运行环境：

取得 Windows 示例程序位置：

CD:\NAPDOS\PCI\PIO-DIO\DLL\_OCX\Demo\  
[http://ftp.icpdas.com/pub/cd/iocard/pci/napdos/pci/pio-dio/dll\\_ocx/demo/](http://ftp.icpdas.com/pub/cd/iocard/pci/napdos/pci/pio-dio/dll_ocx/demo/)

- BCB 4 → for Borland C++ Builder 4  
PIODIO.H → Header files  
PIODIO.LIB → Linkage library for BCB only

- Delphi4 → for Delphi 4  
PIODIO.PAS → Declaration files

- VB6 → for Visual Basic 6  
PIODIO.BAS → Declaration files

- VC6 → for Visual C++ 6  
PIODIO.H → Header files  
PIODIO.LIB → Linkage library for VC only

- VB.NET2005 → for VB.NET2005  
PIODIO.vb → Visual Basic Source files

- CSharp2005 → for C#.NET2005  
PIODIO.cs → Visual C# Source files

**DEMO 程序清单如下：**

- DIO: 数字输出入程序
- DIO\_2: 数字输出入程序 (为 PIO-D56(U)/PEX-D56 仅有)
- INT: 中断测试程序
- INTAPC: 中断测试程序

## 5.2 DOS DEMO 程序

---

DOS 软件和所有 DEMO 在 CD 中可以找到:

CD:\NAPDOS\PCI\PIO-DIO\dos\

<http://ftp.icpdas.com/pub/cd/iocard/pci/napdos/pci/pio-dio/dos/>

- TC\\*. \* → Turbo C 2.xx 或更新版本  
TC\LIB\\*. \* → TC 库文件  
TC\DEMO\\*. \* → TC Demo 程序  
TC\DIAG\\*. \* → TC 诊断程序  
  
TC\LIB\PIO.H → TC 声明文件  
TC\LIB\TCPIO\_L.LIB → TC 大型模块库文件  
TC\LIB\TCPIO\_H.LIB → TC 巨型模块库文件
- MSC\\*. \* → for MSC 5.xx 或更新版本  
MSC\LIB\PIO.H → MSC 声明文件  
MSC\LIB\MSCPIO\_L.LIB → MSC 大型模块库文件  
MSC\LIB\MSCPIO\_H.LIB → MSC 巨型模块库文件
- BC\\*. \* → for BC 3.xx 或更新版本  
BC\LIB\PIO.H → BC declaration file  
BC\LIB\BCPIO\_L.LIB → BC large model library file  
BC\LIB\BCPIO\_H.LIB → BC huge model library file

**DEMO** 程序清单如下:

DEMO1: D/O for CON1

DEMO2: D/I/O for CON1 ~ CON3

DEMO3: Interrupt of P2C0 (Initial low and active high)

DEMO4: Interrupt of P2C0 (Initial high and active low)

DEMO5: 4 interrupt sources



## A1. DOS LIB 函式

### A1-1. ErrorCode 和 ErrorString Code 描述表

表 A.1

Error Code	Error ID	Error String
0	NoError	OK (No error)
1	Driver HandleError	Error opening the device driver
2	DriverCallError	An error occurred while calling the driver functions
3	FindBoardError	Can't find the board on the system
4	TimeOut	Timeout
5	ExceedBoardNumber	Invalid board number (Valid range: 0 to TotalBoards -1)
6	NotFoundBoard	Can't detect the board on the system

### A1-2. PIO\_DriverInit

#### ■ 描述:

这个函数能检测所有系统中 PIO/PISO 板卡。它是基于 PCI 即插即用装置上。它将找到所有安装在系统中的 PIO/PISO 板卡和保存所有它们的资源在库中。

#### ■ 语法:

```
WORD PIO_DriverInit(WORD *wBoards, WORD wSubVendorID,
                    WORD wSubDeviceID, WORD wSubAuxID)
```

#### ■ 参数:

<b>WBoards</b>	[Output]	PC 中板卡号
<b>wSubVendorID</b>	[Input]	板卡的 SubVendor ID
<b>wSubDeviceID</b>	[Input]	板卡的 SubDevice ID
<b>wSubAuxID</b>	[Input]	板卡的 SubAux

#### ■ 返回值:

请参考至 "表 A.1"

### A1-3. PIO\_GetConfigAddressSpace

- **描述:**  
用户能够使用这个函数去得到安装在系统中所有 PIO/PISO 板卡资源信息。那么应用程序就能够很方便的去调用 PIO/PISO 函数。
- **语法:**  
WORD PIO\_GetConfigAddressSpace(wBoardNo, \*wBase, \*wlrq, wSubVendor, \*wSubDevice, \*wSubAux, \*wSlotBus, \*wSlotDevice)
- **参数:**

wBoardNo	[Input]	板卡号
wBase	[Output]	板卡的基地址
wlrq	[Output]	板卡使用的 IRQ 号
wSubVendor	[Output]	Sub Vendor ID
wSubDevice	[Output]	Sub Device ID
wSubAux	[Output]	Sub Aux ID
wSlotBus	[Output]	Slot Bus number (插槽号)
wSlotDevice	[Output]	Slot Device ID (插槽设备 ID)
- **返回值:**  
请参考至 "表 A.1"

### A1-4. PIO\_GetDriverVersion

- **描述:**  
这个函数将得到 PIODIO 驱动版本号。
- **语法:**  
WORD PIO\_GetDriverVersion(WORD \*wDriverVersion)
- **参数:**

wDriverVersion	[Output]	wDriverVersion 地址
----------------	----------	-------------------
- **返回值:**  
请参考至 "表 A.1"

---

## A1-5. ShowPIOPIISO

- **描述:**

这个函数将显示一个专用的 Sub\_ID 文本字符串.这个文本字符串同 PIO.H 中定义的一样.

- **语法:**

WORD ShowPIOPIISO(wSubVendor, wSubDevice, wSubAux)

- **参数:**

<b>wSubVendor</b>	[Input]	板卡的 SubVendor ID
<b>wSubDevice</b>	[Input]	板卡的 SubDevice ID
<b>wSubAux</b>	[Input]	板卡的 SubAux ID

- **返回值:**

请参考至 "表 A.1"