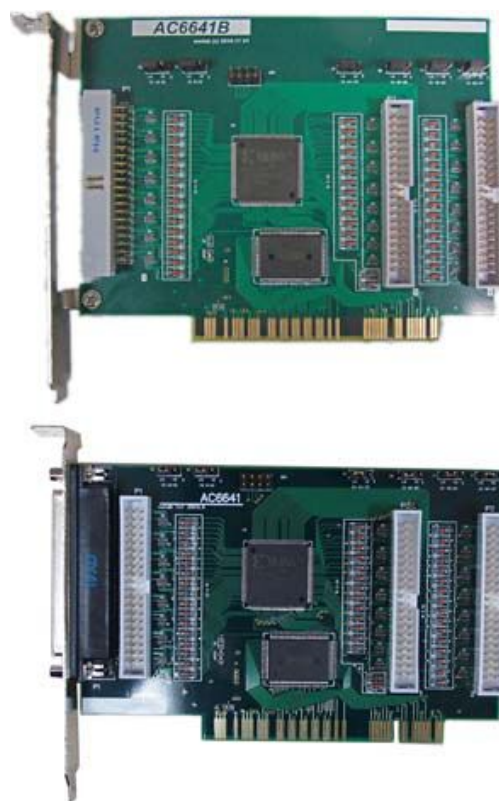


AC6641、AC6641B

使用手册

- ❑ PCI 总线开关量板
- ❑ 96 路可编程 DIO



在开始使用前请仔细阅读下面说明

检查

打开包装请查验如下：

- ✧ AC6641卡一个
- ✧ 手册及光盘
- ✧ DB37插头一套（AC6641B不需要）
- ✧ 40PIN转接电缆

安装

关掉 PC 机电源，将 AC6641 插入主机的任何一个 PCI 插槽中并将外部的输入、输出线连好。如果主机有多套 AC 系列 PCI 插卡，请每次只安装一个插卡。软件启动安装请查看第 3 章说明。

保修

本产品自售出之日起一年内，用户遵守储存、运输和使用要求，而产品质量不合要求，凭保修单免费维修。因违反操作规定和要求而造成损坏的，需缴纳器件费和维修费及相应的运输费用，如果板卡有明显烧毁、烧糊情况原则上不予维修。如果板卡开箱测试有问题，可以免费维修（限购买板卡 10 天内）。

软件支持服务

自销售之日起，提供 6 个月的免费开发咨询。

目录

一、AC6641 说明.....	5
1.1 AC6641 简介.....	5
1.2 主要特点、性能.....	5
1.3 原理说明.....	5
二、安装与连接.....	7
2.1 安装.....	7
2.2 连接器插座定义.....	7
2.3 配套端子板.....	9
2.4 常用硬件应用.....	10
利用开关量输出驱动继电器.....	10
开关量输出驱动光藕.....	10
开关量输入隔离.....	11
三、软件开发.....	12
3.1 软件安装与说明.....	12
软件说明.....	12
驱动安装.....	13
3.2 接口函数说明.....	14
变量约定.....	14
设备函数.....	14
开关量函数.....	15
开关量输入.....	16
开关量输出.....	16
设置开关量输入、输出状态.....	17
读入设置的开关量输入、输出状态.....	18
3.3 VC 程序编程说明.....	18
3.4 VB 程序编程说明.....	19
四、附录.....	21
AC6641 示意图.....	21
IO 译码地址表.....	21
AC142 端子板说明.....	22
AC145N 端子板说明.....	23
连接器说明及针脚定义：.....	23
P1、P2 40 线扁平电缆连接器：.....	23
端子连接器说明：.....	24
AC140E 端子板说明.....	25

AC175 端子板说明.....	26
P6654 端子板说明.....	27

一、AC6641 说明

1.1 AC6641 简介

AC6641B 与 AC6641 功能完全兼容，不同的是 AC6641B 去掉了 DB37 插头，具体连线请参考“配套端子板”部分。以下介绍均以 AC6641 为例。

AC6641 是一款通用 IO 板，96 路（12 个 8 位端口）可编程开关量。采用 PCI 总线支持即插即用、无需地址跳线。采用大规模可编程门阵列设计，提高可靠性。输入兼容 5 伏电平，输出为 3.3V 电平。

主要应用：开关量输入、输出。

相关产品：

AC6651：32 路可编程，3 路脉冲，计数器。（PCI 板卡）

MP451：32 路可编程，3 路脉冲。（USB 板卡）

1.2 主要特点、性能

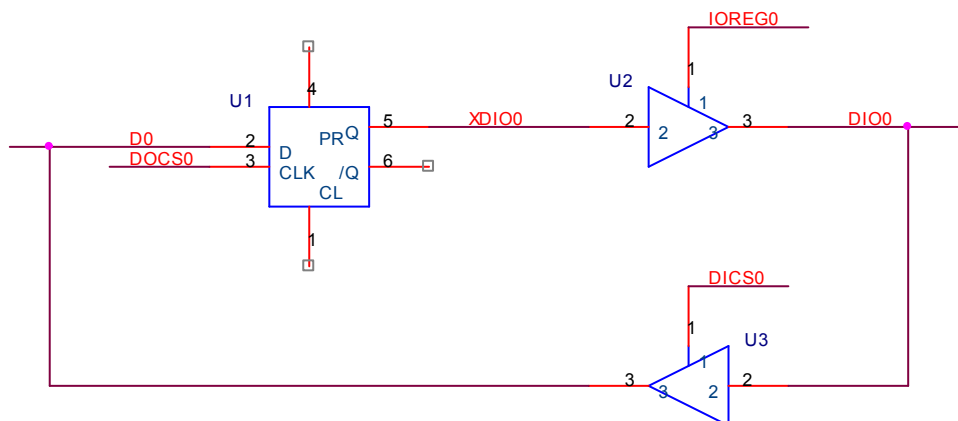
- 12 组可编程 IO 端口，每组端口可以软件设置为输入或输出。
- 开关量复位后为输入状态。
- 输入、输出电平为 TTL 电平（5V 电平）。
- 输入、输出电压范围：0~5V。
- 输入高电平：大于 2.3V（小于 5V）。
- 输入低电平：小于 0.8V（大于 0V）。
- 输出高电平：大于 2.5V，电流大于 6 毫安。
- 输出低电平：小于 0.8V，电流大于 6 毫安。
- 96 路开关量部分内置 4.7K 电阻，可以上拉，也可以下拉。
- PCI 总线，符合 PCI V1.1 标准。
- AC6641 占用 256 个 I/O 选通空间（自动分配）。
- 尺寸：13.5 X 9 厘米

1.3 原理说明

AC6641 采用 CH 系列 PCI 接口芯片 CH365 及门阵列作为主控芯片。门阵列提供 96 路可编程开关量。

输入、输出原理

以一路 IO 为例说明该输入、输出原理。



上图中，各信号说明如下：

1. D0: 数据线的第 0 位。
2. DIO0: 开关量输入、输出通道 0。
3. DOCS0: 第一组输出选通信号。
4. IOREG0: 输入、输出通道 0 的输出允许信号。
5. DICS0: 第一组输入选通信号。

当板卡进行读入操作时，外部状态 DIO0 由 U3 输入到数据总线上并被 PC 读入。当板卡进行输出操作时，输出数据锁存在触发器 U1 中。若 IOREG0=1，则将 XDIO0 输出到外部链接的 DIO0 上。输出允许信号 IOREG0 由内部 IOMODE 寄存器软件控制。同时，在输出状态时，PC 可以通过读入操作反读输出的数据。

开关量编程操作

在使用开关量之前，先要设置开关量输出=0 或=1，这可以通过将板卡上的 S1~S6 设置为上拉（1）或下拉（0）实现。之后设置输入/输出模式，模式设置好后，就可以进行正常的 IO 操作了。

具体 IO 操作请参考第三章的软件开发部分。

二、安装与连接

2.1 安装

关掉 PC 机电源，将 AC6641 插入主机的任何一个 PCI 插槽中并将外部的输入、输出线连接好。如果主机有多套 AC 系列 PCI 插卡，请逐个安装（详细参考软件部分说明）。请注意“**输出端禁止对正电源短路，否则会烧毁输出级**”。

2.2 连接器插座定义

■ 硬件跳线

S1-S6 六个跳线器提供开关量输入上拉、下拉选择。跳线器的设置如下，以 S1 为例，其余类似。

1. 选择下拉



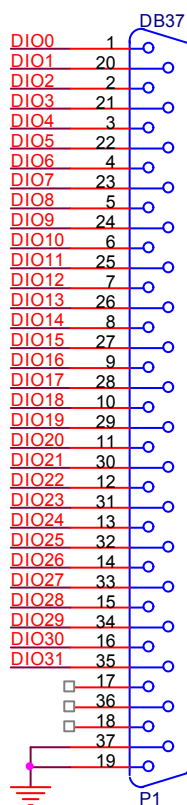
2. 选择上拉



板卡上电之后开关量为输入状态。

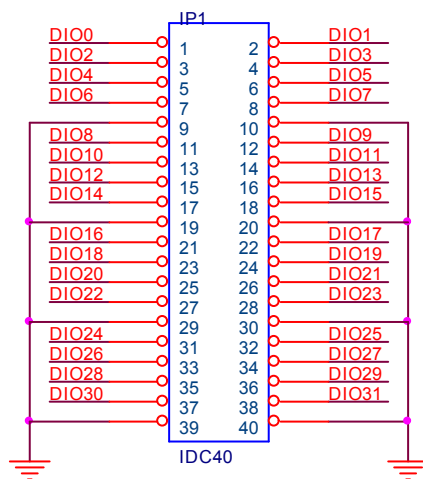
上拉、下拉电阻为 4.7K。通过修改跳线器 S1-S6 来将开关量设置为上拉或者下拉。之后设置输入/输出模式。模式设置好后，就可以进行正常的 IO 操作了。

■ P1: DB37 输出插座



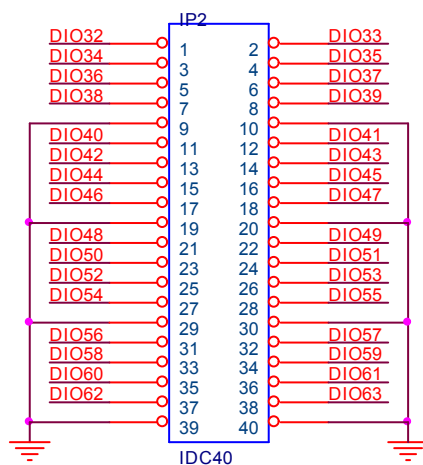
1. DIO0-DIO31: 对应输入、输出通道 0-31 号。
2. 脚 19、37: 地线。

■ IP1: 40 芯扁平电缆输出插座



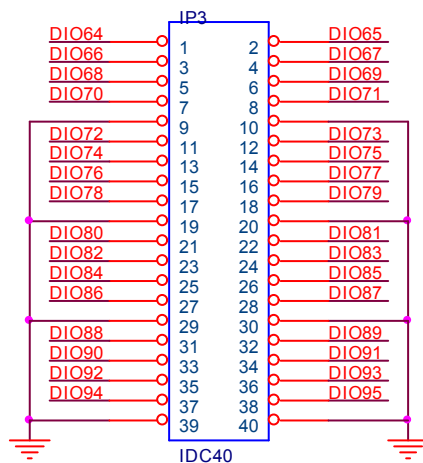
1. DIO0-DIO31: 对应输入、输出通道 0-31 号。
2. 脚 9、10、19、20、29、30、39、40: 地线。

■ IP2: 40 芯扁平电缆输出插座



1. DIO32-DIO63: 对应输入、输出通道 32-63 号。
2. 脚 9、10、19、20、29、30、39、40: 地线。

■ IP3: 40 芯扁平电缆输出插座



1. DIO64-DIO95: 对应输入、输出通道 64-95 号。
2. 脚 9、10、19、20、29、30、39、40: 地线。

2.3 配套端子板

可以配接 AC142、AC145N、AC140E、P6654、AC175 等端子板。

AC142

- ✧ DB37 和 40 脚扁平电缆插座。
- ✧ 40 路螺丝端子，支持 32 路接线。

AC145N

- ✧ 16 路隔离输入、16 路隔离输出

AC140E

- ✧ 16 路中功率继电器输出
- ✧ 16 路光电隔离输入

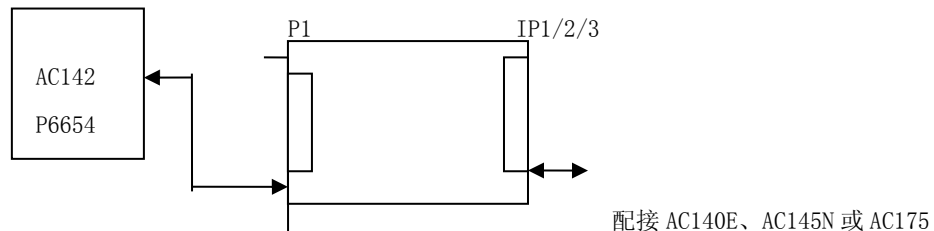
P6654

- ✧ IDC40 扁平电缆转接 DB37

AC175

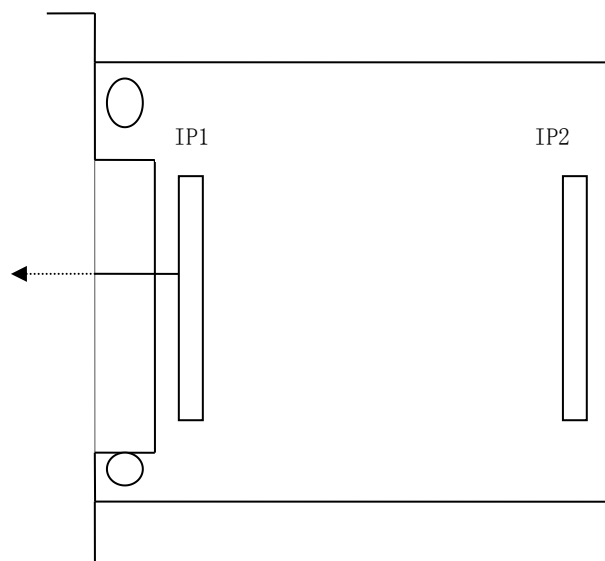
- ✧ 40 脚扁平电缆插座转接 40 脚扁平电缆插座

连接：



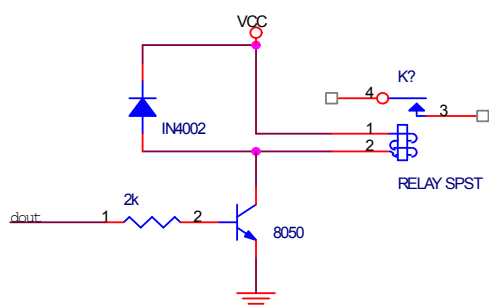
注：AC6641B 的 IP1 端子连接外部板卡示意图如下。

IDC40 扁缆可以从挡片下方接出来，然后与外部板卡连接。

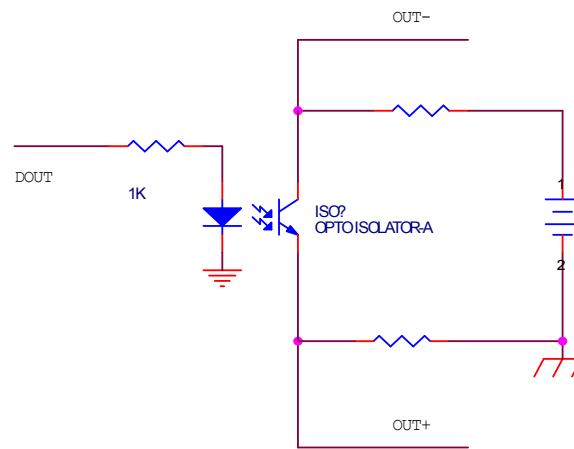


2.4 常用硬件应用

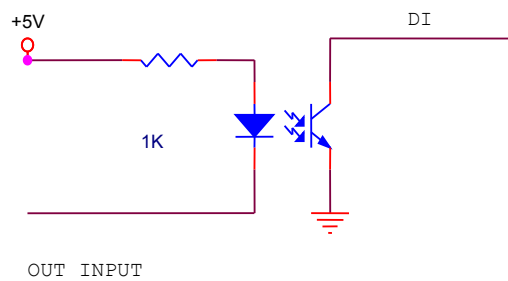
利用开关量输出驱动继电器



开关量输出驱动光藕



开关量输入隔离



三、软件开发

本章介绍驱动的安装、动态链接库函数使用方法以及针对 AC6641 的软件开发指导。请用户在编程前，仔细阅读本手册，了解相关信息。

3.1 软件安装与说明

软件说明

AC6641附带光盘中，提供如下内容：

1. 说明书。
2. 驱动程序，支持win98/win2000/winXP操作系统。
3. Visual C++、Visual Basic编程实例。
4. AC6641测试程序。

注：由于win98、winNT微软已经不提供支持，不建议使用。AC6641卡的驱动不支持win NT。

- 在光盘的\PCI\AC6641\DRIVER目录中包含：AC6641.inf、AC6641.sys、AC6641.dll、AC6641.LIB、AC6641.BAS 5个文件。

- ✧ AC6641.inf：驱动安装文件。
- ✧ AC6641.sys：驱动程序。
- ✧ AC6641.dll：动态链接库。
- ✧ AC6641.LIB：VC的库文件。
- ✧ AC6641.BAS：VB的模块文件。

- 在光盘的\PCI\AC6641\VC目录中包含以下文件。

- ✧ VC的编程例子。
- ✧ 编程需要的include文件。

- 在光盘的\PCI\AC6641\VB目录中包含以下文件。

- ✧ VB的编程例子。

◇ VB编程需要的声明模块程序。

■ 在光盘的\PCI\AC6641\MFC目录中包含MFC的编程例子。

■ AC6641.exe: 测试程序。

驱动安装

安装方法:

1. 关闭计算机电源，将AC6641插入一个PCI插槽。如果有多个AC6641插卡，请每一次安装一个AC6641插卡。第一次安装的插卡的设备号为“0”，第二次安装的插卡的设备号为“1”，……，依此类推。
2. 打开计算机电源，启动Windows。
3. Windows将会显示找到新硬件，可按照“找到新硬件向导”进行下一步。
4. 选择搜索适用我的设备的驱动程序，进行下一步。
5. 选择驱动所在目录，进行安装。（目录：\PCI\AC6641\driver）。
6. 按照找到新硬件向导的提示进行下一步。
7. Windows将显示完成添加/删除硬件向导，单击完成即可完成安装过程。
8. 完成后如果安装第二个AC6641，请关闭计算机电源，插入第二块AC6641插卡，重复上述安装过程。

安装后，程序自动将 AC6641.dll 动态链接库程序拷贝到 windows 系统的 system32 目录中，用户也可以自己将 AC6641.dll 拷贝到当前工作目录中。

驱动安装完毕后，在“控制面板→系统→硬件→设备管理器”中可以找到 AC6641 卡，察看“属性→资源”，如果出现 AC6641 的 IO 地址，表明驱动安装正确。

1. 安装完毕后将在设备管理器中出现一个其他设备（若其他设备是问号，并不表示设备有问题，只是表示系统不知道AC6641板卡是何种设备）。
2. 如果需要更新设备驱动，请在硬件设备管理器目录下选择“AC6641卡→按鼠标右键选择属性→选择驱动程序 →选择更新驱动程序”。
3. 当 Visual C++/Visual Basic 例程从 CD-ROM 复制到硬盘时，属性仍将保持为只读属性，这将影响用户调试程序。请**将属性改为文档属性**，这样就可以进行正常的编译、调试工作了。

3.2 接口函数说明

本卡以 DLL-动态链接库的方式封装了用户在 win98/win2000/winXP 环境下编程需要的函数。动态链接库可以被 windows 环境下的多数编程语言调用，用户只要正确使用调用格式就能正确调用函数。本手册只提供了 VC、VB 的调用例子，有关其他语言调用的方法，用户可以参考其他书籍或直接在网上查找。

变量约定

- HANDLE 操作句柄，等效 32 位有符号数。
- int32 32 位有符号数。
- char, unsigned char 8 位有符号、无符号数。

设备函数

- 打开一个 AC6641 设备

函数: HANDLE AC6641_OpenDevice(__int32 DeviceNum)

参数:

- ✧ DeviceNum: 入口参数，AC6641 设备号，=0、1、2.....，表示第一个、第二个、第三个.....AC6641 插卡。设备号的定义参考驱动安装部分。
- ✧ 函数返回值: 卡的操作句柄。

注: VC 中如果句柄不等于 INVALID_HANDLE_VALUE，表示正确。VB 中如果句柄不等于 &HFFFFFFF，表示正确。

- 关闭一个 AC6641 设备

功能: 关闭以 hDevice 打开的 AC6641 卡。

函数: __int32 AC6641_CloseDevice(HANDLE hDevice)

参数:

- ✧ hDevice: 入口参数，卡的操作句柄。
- ✧ 函数返回数值: 0: 成功; -1: 失败。

- 读入型号

功能：读入 AC6641 的型号。

函数：__int32 AC6641_SN (HANDLE hDevice)

参数：

✧ hDevice: 入口参数，卡的操作句柄。

✧ 函数返回：卡的型号（6641）。

开关量函数

AC6641 具有 96 路可编程输入、输出开关量，分别为 PORT0-PORT11 等 12 个 8 位端口。

PORT0: 对应输入、输出通道 0-7 号。

PORT1: 对应输入、输出通道 8-15 号。

PORT2: 对应输入、输出通道 16-23 号。

PORT3: 对应输入、输出通道 24-31 号。

PORT4: 对应输入、输出通道 32-39 号。

PORT5: 对应输入、输出通道 40-47 号。

PORT6: 对应输入、输出通道 48-55 号。

PORT7: 对应输入、输出通道 56-63 号。

PORT8: 对应输入、输出通道 64-71 号。

PORT9: 对应输入、输出通道 72-79 号。

PORT10: 对应输入、输出通道 80-87 号。

PORT11: 对应输入、输出通道 88-95 号。

每一组 IO 口可以利用函数 AC6641_SetIoMode() 设置为输入、输出状态，同时也可以利用函数 AC6641_GetIoMode() 函数读入设置的状态数据。对应读、写数据与 8 位口的对应关系如下。

数据	D7	D6	D5	D4	D3	D2	D1	D0
PORT0	DIO7	DIO6	DIO5	DIO4	DIO3	DIO2	DIO1	DIO0
PORT1	DIO15	DIO14	DIO13	DIO12	DIO11	DIO10	DIO9	DIO8
PORT2	DIO23	DIO22	DIO21	DIO20	DIO19	DIO18	DIO17	DIO16
PORT3	DIO31	DIO30	DIO29	DIO28	DIO27	DIO26	DIO25	DIO24
PORT4	DIO39	DIO38	DIO37	DIO36	DIO35	DIO34	DIO33	DIO32
PORT5	DIO47	DIO46	DIO45	DIO44	DIO43	DIO42	DIO41	DIO40
PORT6	DIO55	DIO54	DIO53	DIO52	DIO51	DIO50	DIO49	DIO48
PORT7	DIO63	DIO62	DIO61	DIO60	DIO59	DIO58	DIO57	DIO56
PORT8	DIO71	DIO70	DIO69	DIO68	DIO67	DIO66	DIO65	DIO64
PORT9	DIO79	DIO78	DIO77	DIO76	DIO75	DIO74	DIO73	DIO72

PORT10	DIO87	DIO86	DIO85	DIO84	DIO83	DIO82	DIO81	DIO80
PORT11	DIO95	DIO94	DIO93	DIO92	DIO91	DIO90	DIO89	DIO88

注：

1. D7-D0 为读入 8 位数据，D0 为最低位。
2. DIO0-DIO95 为对应输入、输出通道 0-95 号。

开关量输入

功能：读入 0-95 号口中一个 8 位口输入数据。

函数：__int32 AC6641_DI (HANDLE hDevice, __int32 ionum)

参数：

- ✧ hDevice: 入口参数，卡的操作句柄。
- ✧ ionum: 入口参数，=0、1、2、3、4、5、6、7、8、9、a、b，分别选择读入 PORT0-PORT11。
- ✧ 函数返回：出口参数，返回读入的数据，低 8 位有效。8 位数据（D7-D0）分别对应端口的 8 个 IO 线 7-0 号。

开关量输出

功能：设置 PORT0-PORT11，12 个 8 位口中的一个口的输出数据。

函数：__int32 AC6641_DO (HANDLE hDevice, __int32 ionum, __int32 iodata)

参数：

- ✧ hDevice: 入口参数，卡的操作句柄。
- ✧ ionum: 入口参数，=0、1、2、3、4、5、6、7、8、9、a、b，分别选择读入 PORT0-PORT11。
- ✧ iodata: 入口函数，对应 ionum 号口输出的数据。低 8 位有效。8 位数据（D7-D0）分别对应端口的 8 个 IO 线 7-0 号。
- ✧ 函数返回：出口参数，=0 操作成功，其他失败。

注：开关量的位操作用户可以参考 \wwlab disk\应用程序\DIO 位操作 目录

注：关于开关量的操作

- 输入：如果需要判断 16 位输入的某一个位的状态，可以利用“与逻辑”操作完成。例：判断第 DI7 位的状态，DI7 对应二进制 0000 0000 1000 0000 即：16 进制 0080H，只要将读入数据 didata 进行以下操作：

VC: int32 I;

```
I=didata & 0x0080;
VB: dim I as int32
I=didata AND &H0080
```

判断如果 I=0，表示 DI7=0，否则为 1。

- 输出操作：如果希望对 16 位输出端口的某一个输出置位，可以通过与逻辑操作置 0，或逻辑操作置 1。
例：输出数据存放在变量 I 中。

1. 输出 D07 置 0 操作：同样第 7 位的二进制码=1111 1111 0111 1111(第 7 位=0)，对应 16 进制码 FF7FH，
输出数据 dodata：
VC: dodata=I & 0xff7f;
VB: dodata=I AND &Hff7F
2. 输出 D07 置 1 操作：同样第 7 位的二进制码=0000 0000 1000 0000（第 7 位=1），对应 16 进制码
0080H，输出数据 dodata：
VC: dodata=I | 0x0080;
VB: dodata=I OR &H0080

设置开关量输入、输出状态

功能：设置输入、输出控制字。
函数：__int32 AC6641_SetIoMode (HANDLE hDevice,__int32 iomode0,__int32 iomode1)
参数：

- ✧ hDevice: 入口参数，卡的操作句柄。
- ✧ iomode0、iomode1: 入口数据，分别对应控制 IO 通道 7-0 组和 11-8 组。对应关系如下：

D7	D6	D5	D4	D3	D2	D1	D0
M7	M6	M5	M4	M3	M2	M1	M0
X	X	X	X	M11	M10	M9	M8

注：
iomode0: M7-M0。
iomode1: M11-M8。（低四位有效）
Mn: =1: 设置 PORTn 为输出模式/=0: 设置 PORTn 为输入模式。（n=0-11）

✧ 函数返回：出口函数，=0 操作成功，其他失败。

读入设置的开关量输入、输出状态

功能：读入输入、输出控制字。

函数：__int32 AC6641_GetIoMode (HANDLE hDevice)

参数：

✧ hDevice: 入口参数，卡的操作句柄。

✧ 函数返回：输入、输出控制字。

3.3 VC 程序编程说明

编程前，请将 **AC6641.dll** 动态链接库程序拷贝到用户当前目录中或 windows 系统的 system32 目录中。
将 **AC6641.lib** 及 **AC6641.h** 程序拷贝到用户当前目录中。（需要的文件在 VC 目录中）

VC 编程的基本流程：

1. 利用显式调用加载函数。AC6641.lib、AC6641.h 文件必须在当前工作目录中。

方法：程序的开始处加入如下语句：

```
#pragma comment(lib,"AC6641.lib")
#include"AC6641.h"
```

详细可以参考 VC 目录中的程序，AC6641.h 文件包含了需要的函数的声明过程。

2. 利用 AC6641_OpenDevice 函数获得板卡的操作句柄。
3. 在退出程序时必须执行如下操作：利用 AC6641_CloseDevice 函数关闭操作句柄。

例：

```
//获得所有 AC6641 的操作函数
#pragma comment(lib,"AC6641.lib")
#include"AC6641.h"
```

```
HANDLE hDevice=INVALID_HANDLE_VALUE; //硬件操作句柄
```

```
main()
{
```

```
    //获得 6641 硬件操作句柄
```

```

hDevice=AC6641_OpenDevice(0); //创建设备驱动句柄，设备号为 0

..... //用户程序

//退出
AC6641_CloseDevice(hDevice); //关闭操作句柄
return 0;
}

```

详细可以参考光盘上的 AC6641 的 VC 目录下的例子。

在编程时必须注意，硬件操作句柄 **HANDLE** 必须为全局变量或必须传递给有相应硬件操作的函数。硬件句柄只要在程序启动时打开一次即可，不需要每次打开或关闭。

3.4 VB 程序编程说明

编程前，请将 AC6641.dll 动态链接库程序拷贝到用户当前目录中或 windows 系统的 system32 目录中。

VB 编程的基本流程：

1. 在工程菜单中选择添加模块，将 AC6641.bas 模块添加进来（该模块在光盘中的 \PCI\AC6641\vb 目录中，应用时将文件拷贝到当前工作目录），此文件为所有函数的声明文件。
2. 在模块中定义一个硬件操作句柄，为一个 __int32 属性的全局变量，这样可以被用户程序中的所有 form 调用（例：AC6641.bas 中声明的句柄 hd6641）。
3. 利用 AC6641_OpenDevice 函数获得板卡的操作句柄。

在退出程序时必须执行如下操作：

利用 AC6641_CloseDevice 函数关闭句柄。

注：AC6641.bas 模块已经包含了所有必要的 6641 函数的声明语句。

例：

```

DIM hd6641 as __int32
Private Sub Form_Load()
DIM I as __int32
Hd6641 = AC6641_OpenDevice(0) ‘打开设备 0 号，获得驱动句柄
..... ‘其他操作
End Sub
.....

Private Sub Form_Unload(Cancel As Integer)
AC6641_CloseDriver hd6641 ‘关闭驱动

```

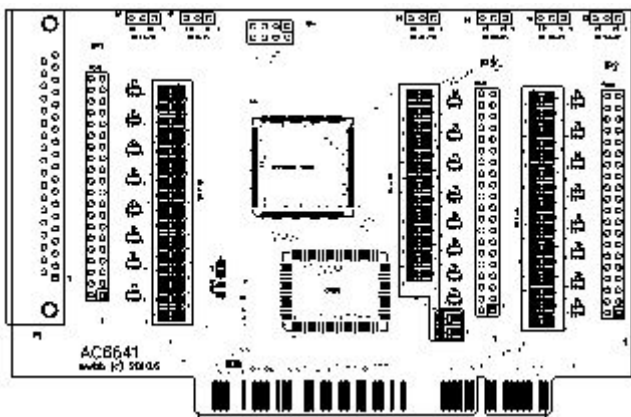
End Sub

有关用户其他方面的应用请参考光盘中的例程。

注：VB 中如果设备操作句柄不等于：&HFFFFFFF 为有效句柄。

四、附录

AC6641 示意图



IO 译码地址表

PCI设备参数:

- 1. VID: 4348H
- 2. PID: 5049H
- 3. SYSSUBID: 66410001H

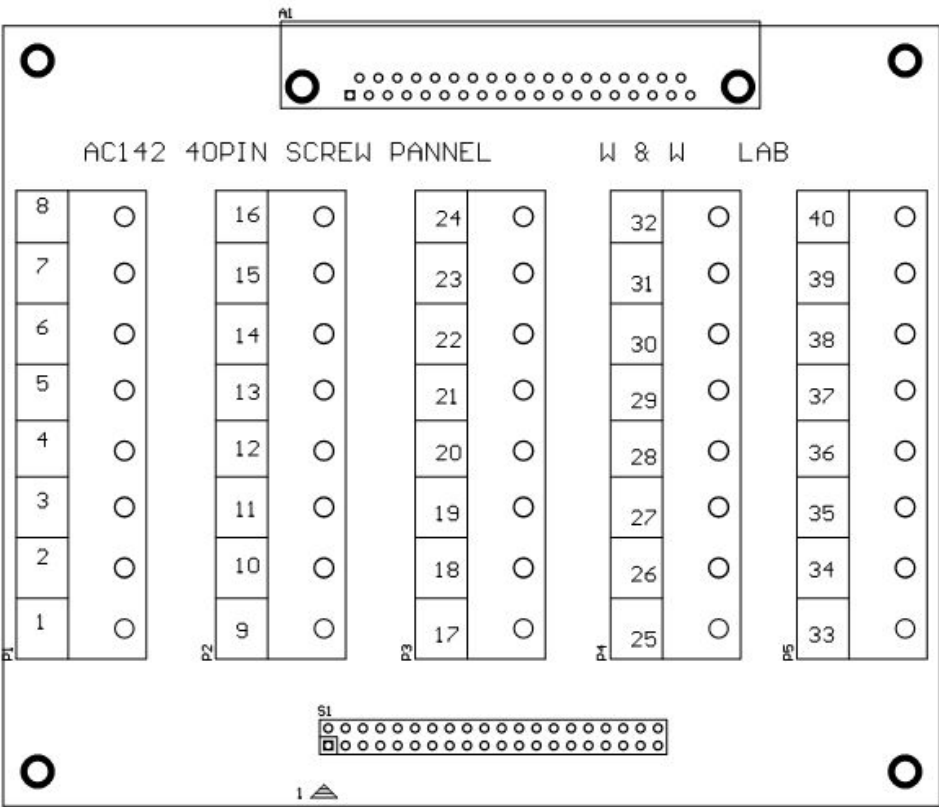
偏移地址分配 (IOBASE0) :

偏移地址 (A0-A3)	读操作 (RD)	写操作 (WR)
0 H	DI PORT0	DO PORT0
1 H	DI PORT1	DO PORT1
2 H	DI PORT2	DO PORT2
3 H	DI PORT3	DO PORT3

4 H	DI PORT4	DO PORT4
5 H	DI PORT5	DO PORT5
6 H	DI PORT6	DO PORT6
7 H	DI PORT7	DO PORT7
8 H	DI PORT8	DO PORT8
9 H	DI PORT9	DO PORT9
AH	DI PORT10	DO PORT10
BH	DI PORT11	DO PORT11
CH	state0	iomode0
DH	state1	iomode1

AC142 端子板说明

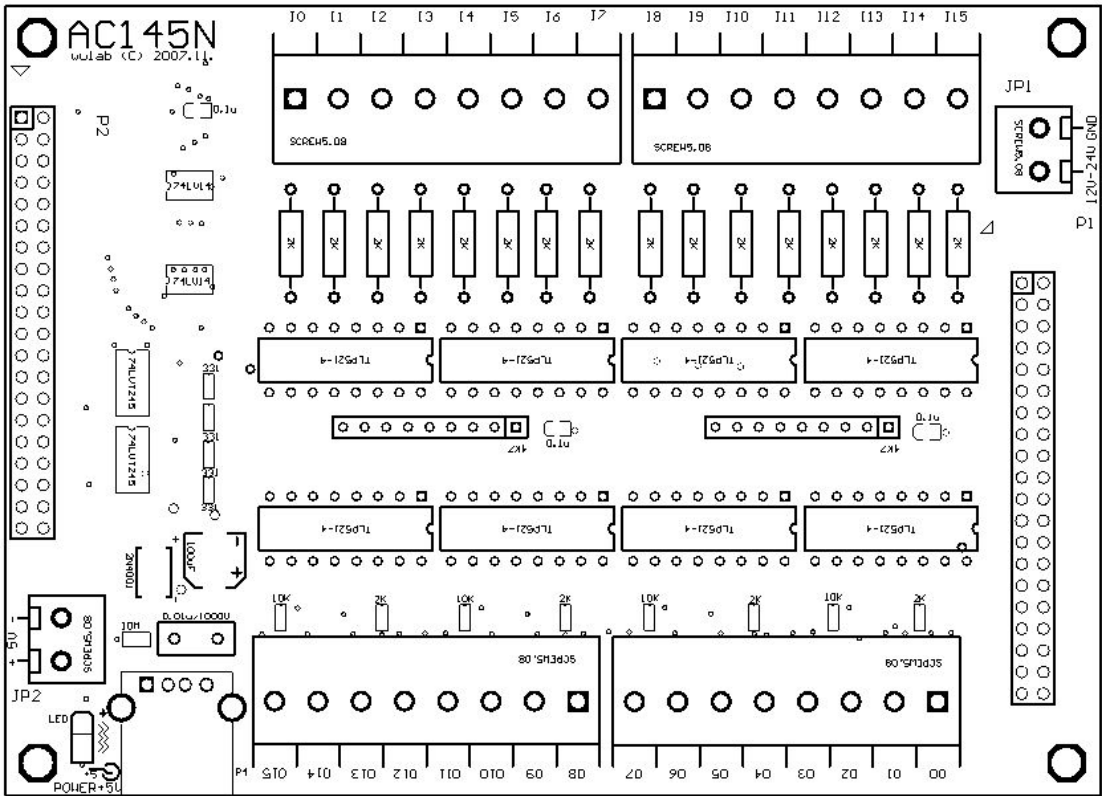
IDC40 与 DB37 转接板。



IDC40 与 DB37 的号码与端子一一对应。

AC145N 端子板说明

板卡示意图：

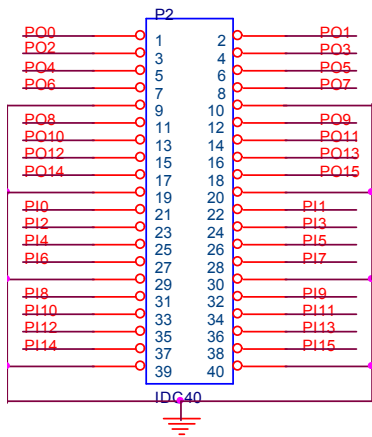


连接器说明及引脚定义：

对外信号输入、输出连接器，提供螺丝端子与扁平电缆二种连接方式。

P1、P2 40 线扁平电缆连接器：

- P1：40 线扁平电缆数据连接器为测试连接插座，不建议用户使用。
- P2：40 线扁平电缆数据连接器，连接 PC 应用板连接器，定义见下图：



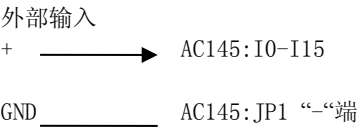
1. PO0-PO15: 端子板隔离器输出，连接到 PC 应用板输入。
2. PI0-PI15: 端子板驱动隔离器输出的输入端，对应连接 PC 到应用板的输出。
3. 9、10、19、20、29、30、39、40: PC 控制端口地线。

端子连接器说明:

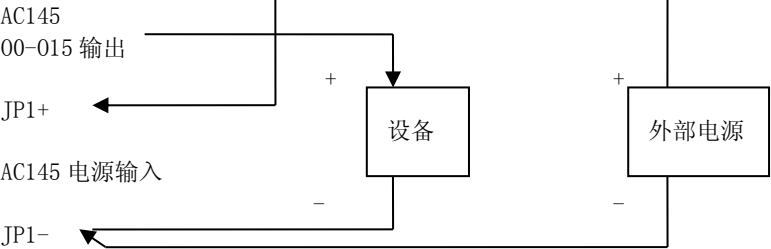
- 端子: I0-I15 对应隔离输入通道: 0-15。输入地线为 JP1 “-”输入
- 端子: O0-O15 对应隔离输出通道: 0-15。输出地线为 JP1 “-”输入
- JP1 端子的+/-端为隔离输出驱动电源的+/-输入，电压: +5-24V。JP1 的地线 (- 端) 也是输出部分的地线。O0-O15 的输出电压=JP1 电压。
- JP2 端子的+/-为 AC145N 自身电源接口，电压 5-12 伏。如果用户应用 USB 供电，这个电源可以不用。这个电源必须与 JP1 电源隔离。

注意: 操作前请认真阅读说明，切勿接反。

输入示意:



输出示意:

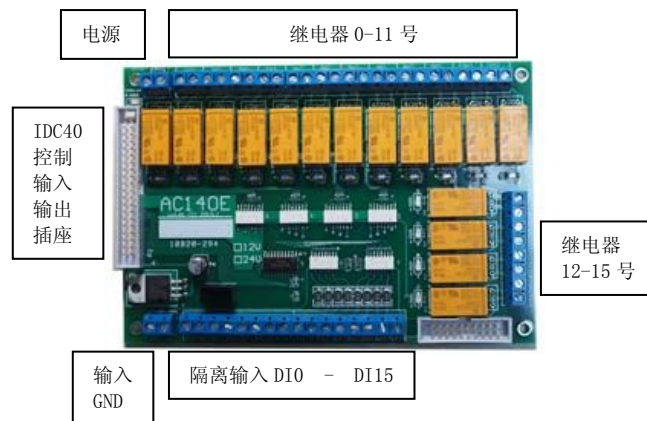


AC140E 端子板说明

使用说明

16 路光耦隔离输入连接

- DI0-DI15: 从端子板下方从左至右分别对应 16 路光电隔离输入 0-15 号。
 - GND: 光电隔离输入地线, 即 DI0-DI15 的公共输入地线, 两个 GND 为同一个地。
 - 注意: 电源负极与光电隔离输入地线 GND 相互隔离。
- 16 路光耦隔离输入示意图如下:



AC140E 示意图

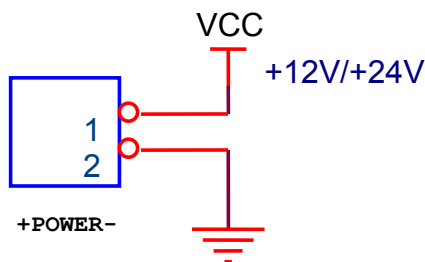
16 路光耦继电器端子连接

CH0-CH15: 分别对应 16 个继电器 0-15 的常开开关接点; 每个 CH 都是二个接线端子, 对应一路的继电器开关接点。每路继电器旁边 RC 为去火花电容焊接位置, 用户可在接感性负载时自行安装焊接。

16 路通道排序示意图见上面 AC140E 示意图:

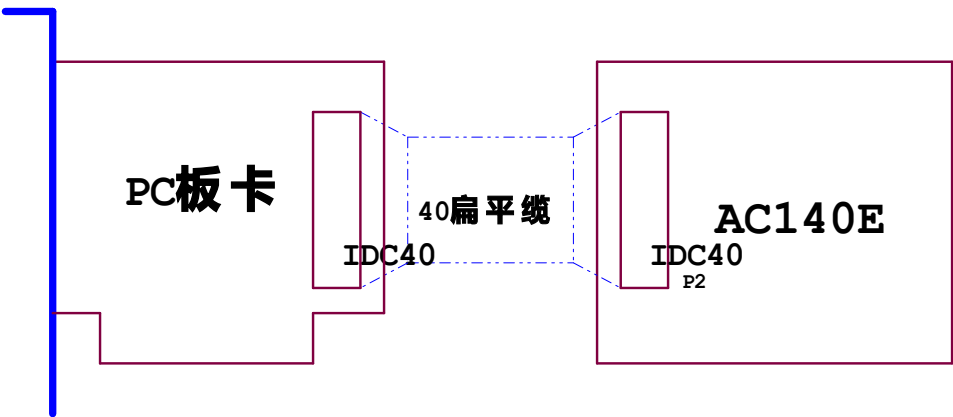
端子板电源连接

供电电源端子(位于端子板左上角) 连接示意图: AC140EV12 使用+12V 1.5A 规格电源供电; AC140EV24 使用+24V 1A 规格电源供电!



端子板与 PC 连接

端子板与 PC 板卡使用 IDC40 扁平电缆连接，示意图如下：



端子板与 PC 连接示意图

AC175 端子板说明

提供安装在 PC 档片位置的 40 脚扁平电缆转接，由 PC 内部转接道外部。用户可以在外部配接 AC142 40 线螺丝端子端子板。

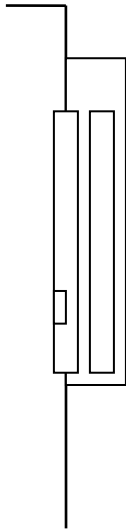
插座定义：



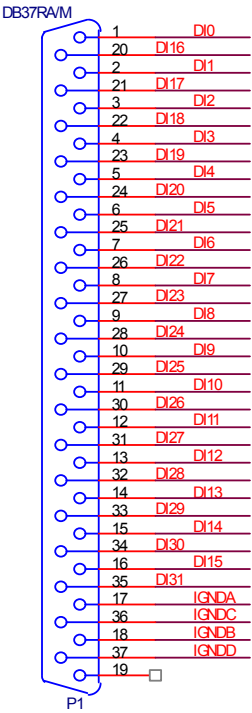
外部插座

内部插座

示意图：

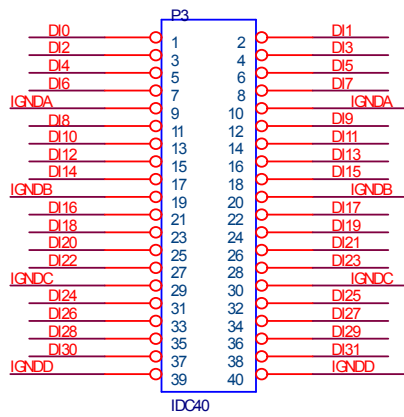


P6654 端子板说明



P6654 将 P3 输入的信号转接到 PC 的挡片的外部输入插座上,DB37 针座。

- P3: 40 芯扁平电缆，通道 0-31 号输入插座



1. DI0-DI31 对应 32 输入通道的 0-31 号输入。
2. 通道输入 0-7 号具有输入脉冲检测功能，详细见上一章“原理说明”
3. IGNDA: 通道 0-7 号输入地线。
4. IGNDB: 通道 8-15 号输入地线。
5. IGND: 通道 16-23 号输入地线。
6. IGND: 通道 24-31 号输入地线。