

# PC-6312 模入模出接口卡技术说明书

## 1. 概述:

PC-6312 模入模出接口卡适用于具有 ISA 总线的 PC 系列微机,有很好的兼容性。CPU 从目前广泛使用的 64 位处理器直到早期的 16 位处理器均可适用,操作系统可选用经典的 MS-DOS,目前流行的 Windows 系列,高稳定性的 Unix 等多种操作系统以及专业数据采集分析系统 LabVIEW 等软件环境。在硬件的安装上也非常简单,使用时只需将接口卡插入机内任何一个 ISA 总线插槽中,信号电缆从机箱外部直接接入。也可插入我所研制的 PC 扩展箱内使用。

本接口卡采用了高性能的程控仪用放大器,具有极高的输入阻抗和共模抑制比,并具有最高可达 1000 倍的放大增益,可直接配接各种传感器,以完成对不同信号的放大处理,用户可根据使用需要选择不同的输入输出方式和数据码制。

本接口卡出厂配装 $\times 1$ ;  $\times 10$ ;  $\times 100$ ;  $\times 1000$ ; 的程控放大器,如需配装 $\times 1$ ;  $\times 2$ ;  $\times 4$ ;  $\times 8$  的程控放大器,应在定货时加以说明。

## 2. 主要技术参数:

### 2.1 模入部分:

#### 2.1.1 输入通道数:

单端 16 路; \* (标\*为出厂标准状态,下同)双端 8 路

#### 2.1.2 输入信号范围: $0V \sim 10V^*$ ; $-5V \sim +5V$ ; $\pm 10V$

#### 2.1.3 输入阻抗: $\geq 10M\Omega$

#### 2.1.4 A/D 转换分辨率: 12 位

#### 2.1.5 共模抑制比(典型值):

$90dB (G=1)$ ;  $110dB (G=10)$ ;  $130dB (G \geq 100)$

#### 2.1.6 A/D 转换速度: $10\mu S$

#### 2.1.7 A/D 启动方式: 程序启动

#### 2.1.8 A/D 转换结束识别: 程序查询 / 中断方式

#### 2.1.9 A/D 转换非线性误差: $\pm 1LSB$

#### 2.1.10 A/D 转换输出码制: 单极性原码\* / 双极性偏移码

#### 2.1.11 通道切换时间:

模拟开关导通时间  $3.5\mu S$  + 放大器建立时间

( $G=1$ ,  $23\mu S$ ;  $G=10$ ,  $28\mu S$ ;  $G=100$ ,  $140\mu S$ ;  $G=1000$ ,  $1300\mu S$ )

#### 2.1.12 系统综合误差: $\leq 0.1\% F.S$ ( $\times 1$ 倍时)

### 2.2 模出部分:

#### 2.2.1 输出通道数:

2 路 (互相独立,可同时或分别输出,具有上电自动清零功能。)

#### 2.2.2 输出范围:

电压方式:  $0 \sim 5V$ ;  $0 \sim 10V^*$ ;  $1 \sim 5V$ ;  $\pm 5V$ ;  $\pm 10V$ ;  $\pm 2.5V$ ;

电流方式:  $0 \sim 10mA$ ;  $4 \sim 20mA$

#### 2.2.3 输出阻抗: $\leq 2\Omega$ (电压方式)

#### 2.2.4 D/A 转换器件: AD7948

#### 2.2.5 D/A 转换分辨率: 12 位

#### 2.2.6 D/A 转换输入码制:

二进制原码(单极性输出方式时)\*;

二进制偏移码(双极性电压输出方式时)

#### 2.2.7 D/A 转换综合建立时间: $\leq 2\mu S$

#### 2.2.8 D/A 转换综合误差:

电压方式:  $\leq 0.2\% F.S$ ;

电流方式:  $\leq 1\% F.S$

#### 2.2.9 电流输出方式负载电阻范围:

使用机内  $+12V$  电源时:  $0 \sim 250\Omega$ ;

外加  $+24V$  电源时:  $0 \sim 750\Omega$

### 2.3 电源功耗:

$+5V (\pm 10\%) \leq 400mA$ ;

## 2.4 使用环境要求:

相对湿度: 40%~80%;

## 2.5 外型尺寸:

### 3. 工作原理:

3.1 工作原理框图: PC-6312 模入模出接口卡工作原理框图见图 1。



### 3.2 模入部分:

### 3.3 模出部分:

#### 4. 安装及使用注意:

为保证安全及采集精度，应确保系统地线（计算机及外接仪器机壳）接地良好。特别是使用双端输入方式时，为防止外界较大的共模干扰，应注意对信号线进行屏蔽处理。

### 5. 使用与操作:

5.1 主要可调整元件位置见图2。

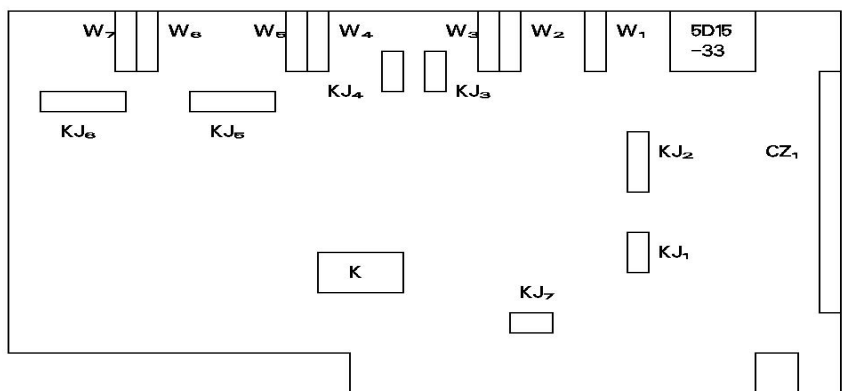


图 2 主要可调整元件位置图

### 5.2 I / O 基地址选择:

I / O 基地址的选择是通过开关 K 进行的, 开关拨至“ON”处为 0, 反之为 1。初始地址的选择范围一般为 0100H~0378 之间。用户应根据主机硬件手册给出的可用范围以及是否插入其它功能卡来决定本卡的 I / O 基地址。出厂时本卡的基地址设为 0100H, 并从基地址开始占用连续 8 个地址。现举例说明见图 3。

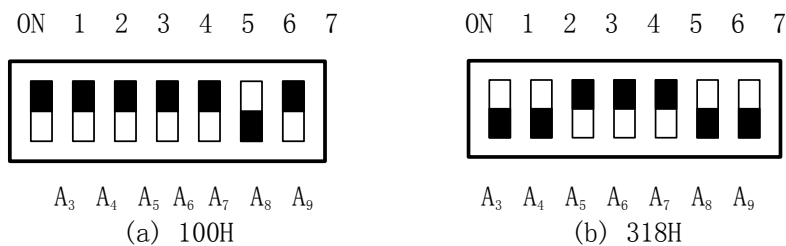


图 3 I / O 基地址选择举例

### 5.3 输入输出插座接口定义:

输入输出插座接口定义(括号内表示双端输入通道组成)CZ<sub>1</sub>见表 1。

表 1 输入输出插座 CZ<sub>1</sub>接口定义

| 插座引脚号 | 信号定义                                  | 插座引脚号 | 信号定义                                 |
|-------|---------------------------------------|-------|--------------------------------------|
| 1     | 模拟地 1                                 | 20    | 模拟地 1                                |
| 2     | CH <sub>1</sub> (CH <sub>1</sub> +) ) | 21    | CH <sub>17</sub> (CH <sub>1</sub> -) |
| 3     | CH <sub>2</sub> (CH <sub>2</sub> +) ) | 22    | CH <sub>18</sub> (CH <sub>2</sub> -) |
| 4     | CH <sub>3</sub> (CH <sub>3</sub> +) ) | 23    | CH <sub>19</sub> (CH <sub>3</sub> -) |
| 5     | CH <sub>4</sub> (CH <sub>4</sub> +) ) | 24    | CH <sub>20</sub> (CH <sub>4</sub> -) |
| 6     | CH <sub>5</sub> (CH <sub>5</sub> +) ) | 25    | CH <sub>21</sub> (CH <sub>5</sub> -) |
| 7     | CH <sub>6</sub> (CH <sub>6</sub> +) ) | 26    | CH <sub>22</sub> (CH <sub>6</sub> -) |
| 8     | CH <sub>7</sub> (CH <sub>7</sub> +) ) | 27    | CH <sub>23</sub> (CH <sub>7</sub> -) |
| 9     | CH <sub>8</sub> (CH <sub>8</sub> +) ) | 28    | CH <sub>24</sub> (CH <sub>8</sub> -) |
| 10    | 模拟地 1                                 | 29    | 模拟地 1                                |
| 11    | NC                                    | 30    | NC                                   |
| 12    | D / A <sub>1</sub> 电压端                | 31    | D / A <sub>2</sub> 电压端               |
| 13    | 模拟地 2                                 | 32    | 模拟地 2                                |
| 14    | D / A <sub>1</sub> 电流端                | 33    | D / A <sub>2</sub> 电流端               |
| 15    | +12V 输出                               | 34    | +12V 输出                              |
| 16    | -12V 输出                               | 35    | -12V 输出                              |
| 17    | +5V 输出                                | 36    | +5V 输出                               |
| 18    | 电源地                                   | 37    | 电源地                                  |
| 19    | 电源地                                   |       |                                      |

注 1: 模拟地 1 为 A / D 输入参考地, 模拟地 2 为 D / A 输出参考地, 电源地为各种电源输出的参考地。

注 2: 各种电源输出应注意保护, 严禁短路。否则将造成主机电源损坏, 使用中应特别小心。

### 5.4 跨接插座的使用法:

#### 5.4.1 输入单 / 双端方式选择:

KJ<sub>1</sub>、KJ<sub>2</sub>为单 / 双端输入方式选择, 其使用方法见图 4。

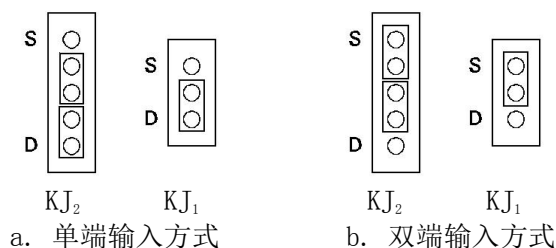


图4 单 / 双端输入方式选择

#### 5.4.2 转换码制选择:

KJ<sub>3</sub>为转换码制选择插座。码制的定义参见 5.6 节。用户应根据输入信号的极性进行选择，选择方法见图 5。

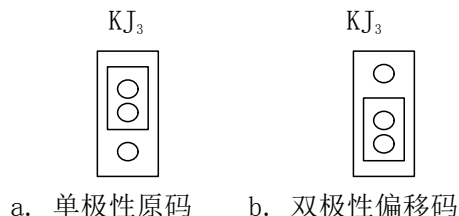


图5 转换码制选择

#### 5.4.3 A / D 量程选择:

KJ<sub>4</sub>为 A / D 量程选择插座，其使用方法见图 6。

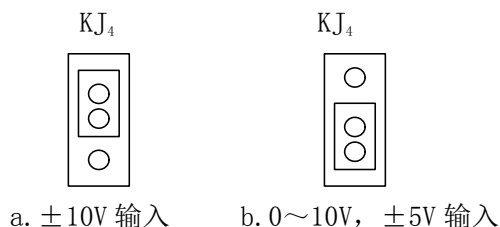


图6 A / D 量程选择

#### 5.4.4 D / A 输出方式及范围选择:

KJ<sub>5</sub>、KJ<sub>6</sub> 为 D / A 输出量程选择插座，其中 KJ<sub>5</sub>对应 D / A<sub>1</sub>，KJ<sub>6</sub>对应 D / A<sub>2</sub>。2 路 D / A 可以选择相同或不同的输出方式和范围，互不影响。各组插座的使用方法见图 7。

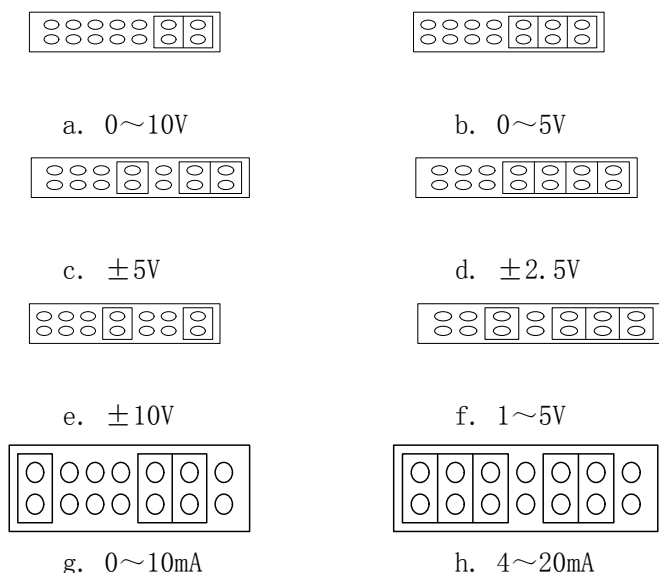


图7 D / A 输出方式及范围选择

#### 5.4.5 中断方式及中断源选择:

KJ<sub>7</sub>为中断有效及中断源选择插座。该插座全部开路时为非中断方式，中断源的选择见图 8。

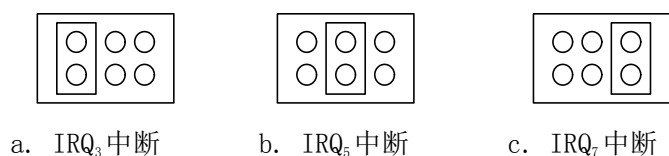


图8 中断源的选择

## 5.5 控制端口地址与有关数据格式:

## 5.5.1 各个控制端的操作地址与功能见表2:

表2 端口地址与功能

| 端口操作地址 | 操作命令 | 功 能                          |
|--------|------|------------------------------|
| 基地址+0  | 写    | 写程控增益代码和通道代码, 选通道。           |
| 基地址+0  | 读    | 启动 D / A 转换                  |
| 基地址+1  | 写    | 启动 A / D 转换                  |
| 基地址+2  | 读    | 查询 A / D 转换状态, 读高 4 位转换结果    |
| 基地址+3  | 读    | 读低 8 位结果, 清 A / D 转换状态及中断标志  |
| 基地址+4  | 写    | 写 D / A <sub>1</sub> 高 8 位数据 |
| 基地址+5  | 写    | 写 D / A <sub>1</sub> 低 4 位数据 |
| 基地址+6  | 写    | 写 D / A <sub>2</sub> 高 8 位数据 |
| 基地址+7  | 写    | 写 D / A <sub>2</sub> 低 4 位数据 |

## 5.5.2 程控增益代码格式: (括号内为×1; ×2; ×4; ×8 器件增益)

| D <sub>7</sub> D <sub>6</sub> | D <sub>5</sub> D <sub>4</sub> | D <sub>3</sub> D <sub>2</sub> D <sub>1</sub> D <sub>0</sub> | 增 益      |
|-------------------------------|-------------------------------|-------------------------------------------------------------|----------|
| X X                           | 0 0                           | 通道代码                                                        | 1 (1)    |
| X X                           | 0 1                           | 通道代码                                                        | 10 (2)   |
| X X                           | 1 0                           | 通道代码                                                        | 100 (4)  |
| X X                           | 1 1                           | 通道代码                                                        | 1000 (8) |

## 5.5.3 通道代码数据格式见表3:

表3 通道代码数据格式

| 通道号 | 十进制<br>代码 | 十六进制<br>代码 | 输入方式  | 通道号 | 十进制<br>代码 | 十六进制<br>代码 | 输入方式 |
|-----|-----------|------------|-------|-----|-----------|------------|------|
| 1   | 0         | 00H        | 单 / 双 | 9   | 8         | 08H        | 单    |
| 2   | 1         | 01H        | 单 / 双 | 10  | 9         | 09H        | 单    |
| 3   | 2         | 02H        | 单 / 双 | 11  | 10        | 0AH        | 单    |
| 4   | 3         | 03H        | 单 / 双 | 12  | 11        | 0BH        | 单    |
| 5   | 4         | 04H        | 单 / 双 | 13  | 12        | 0CH        | 单    |
| 6   | 5         | 05H        | 单 / 双 | 14  | 13        | 0DH        | 单    |
| 7   | 6         | 06H        | 单 / 双 | 15  | 14        | 0EH        | 单    |
| 8   | 7         | 07H        | 单 / 双 | 16  | 15        | 0FH        | 单    |

## 5.5.4 查询 A / D 转换状态数据格式:

查询 A / D 转换状态时的数据格式及意义见表4(端口地址为基地址+2):

表4 A / D 转换状态数据格式(X 表示任意)

| 操作命令 | D <sub>7</sub> | D <sub>6</sub> | D <sub>5</sub> | D <sub>4</sub> | D <sub>3</sub> | D <sub>2</sub> | D <sub>1</sub> | D <sub>0</sub> | A / D 转换状态 |
|------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|------------|
| 读    | 1              | x              | x              | x              | x              | x              | x              | x              | 没有或正在转换    |
| 读    | 0              | x              | x              | x              | x              | x              | x              | x              | 转换结束       |

## 5.5.5 A / D 转换结果数据格式: A / D 转换结果数据格式见表5:

表5 A / D 转换结果数据格式

| 端口地址  | 操作命令 | D <sub>7</sub>  | D <sub>6</sub>  | D <sub>5</sub>  | D <sub>4</sub>  | D <sub>3</sub>   | D <sub>2</sub>   | D <sub>1</sub>  | D <sub>0</sub>  | 意 义     |
|-------|------|-----------------|-----------------|-----------------|-----------------|------------------|------------------|-----------------|-----------------|---------|
| 基地址+1 | 读    | 0               | 0               | 0               | 0               | DB <sub>11</sub> | DB <sub>10</sub> | DB <sub>9</sub> | DB <sub>8</sub> | 高 4 位数据 |
| 基地址+2 | 读    | DB <sub>7</sub> | DB <sub>6</sub> | DB <sub>5</sub> | DB <sub>4</sub> | DB <sub>3</sub>  | DB <sub>2</sub>  | DB <sub>1</sub> | DB <sub>0</sub> | 低 8 位数据 |

5.5.6 D / A 转换数据格式: D / A<sub>1</sub> 转换数据格式见表6, D / A<sub>2</sub> 转换数据格式类同。表6 D / A<sub>1</sub> 转换数据格式 (X 表示任意)

| 端口地址  | 操作命令                    | D <sub>7</sub>   | D <sub>6</sub>   | D <sub>5</sub>  | D <sub>4</sub>  | D <sub>3</sub>  | D <sub>2</sub>  | D <sub>1</sub>  | D <sub>0</sub>  | 意 义     |
|-------|-------------------------|------------------|------------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|---------|
| 基地址+4 | $\overline{\text{IOW}}$ | DB <sub>11</sub> | DB <sub>10</sub> | DB <sub>9</sub> | DB <sub>8</sub> | DB <sub>7</sub> | DB <sub>6</sub> | DB <sub>5</sub> | DB <sub>4</sub> | 高 8 位数据 |
| 基地址+5 | $\overline{\text{IOW}}$ | DB <sub>3</sub>  | DB <sub>2</sub>  | DB <sub>1</sub> | DB <sub>0</sub> | x               | x               | x               | x               | 低 4 位数据 |

## 5.6 模入模出码制以及数据与模拟量的对应关系:

5.6.1 本接口卡在单极性方式工作时，即模入模出的模拟量为 0~10V 时转换后和写出的 12 位数码为二进制原码。此 12 位数码表示一个正数码，其数码与模拟电压值的对应关系为：

$$\text{模拟电压值} = \text{数码}(12 \text{ 位}) \times 10(\text{V}) / 4096 \quad (\text{V})$$

即：1LSB=2.44mV

5.6.2 本接口卡在双极性方式工作时，转换后和写出的 12 位数码为二进制偏移码。此时 12 位数码的最高位(DB<sub>11</sub>)为符号位，“0”表示负，“1”表示正。此时数码与模拟电压值的对应关系为：

模入模出信号为-5~+5V 时：

$$\text{模拟电压值} = \text{数码}(12 \text{ 位}) \times 10(\text{V}) / 4096 - 5 \quad (\text{V})$$

即：1LSB=2.44mV

5.7 中断工作方式：

本卡的 A/D 转换结束信号可以采用中断方式通知 CPU 进行处理。改变 KJ<sub>7</sub> 的位置可以选用 IRQ<sub>3</sub>、IRQ<sub>5</sub> 或 IRQ<sub>7</sub> 中断。用户在使用中断方式时，应对主机系统的 8259 中断管理器进行初始化并编制中断处理程序。并在 8259 中断允许之前，先清除本卡的中断标志。当 A/D 转换结束时，本卡会向 8259 中断管理器发出一个高电平的中断申请，CPU 接到中断请求后转向中断处理程序运行读数操作。当读取低 8 位转换结果时，会自动清除中断标志。

5.8 电流输出方式的使用与扩展：

本卡模出部分可选择 0~10mA 或 4~20mA 电流输出方式以直接驱动 II、III 型执行仪表。采用电流输出方式时，供电电源可以使用本卡提供的 +12V。也可扩展使用机外 +24V 电源。其连接使用方法见图 9。

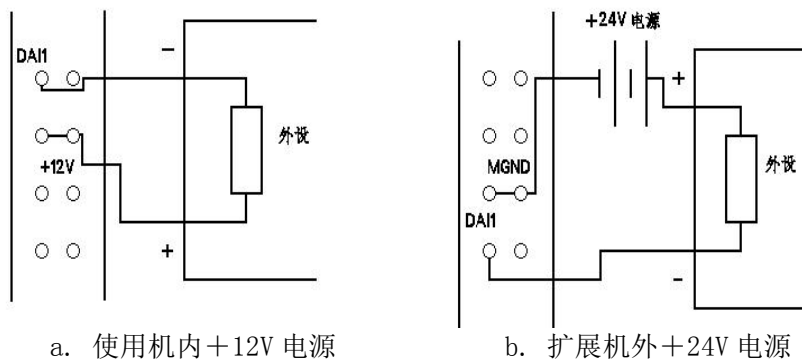


图 9 电流输出方式使用方法

5.9 调整与校准：

5.9.1 产品出厂前，本卡的模入模出部分均已按照单极性 0~10V 调整好，一般情况下用户不需进行调节，如果用户改变了工作模式及范围，可按本节所述方法进行调整。调整时应开机预热 20 分钟以上后进行，并准备一块 4 位半以上的数字万用表。

5.9.2 各电位器功能说明：

W<sub>1</sub> 为输入放大器零点调节。

W<sub>2</sub> 为 A/D 转换器满度调节。

W<sub>3</sub> 为 A/D 转换器双极性偏移调节。

W<sub>4</sub> 为 D/A<sub>1</sub> 零点调节。

W<sub>5</sub> 为 D/A<sub>1</sub> 满度调节。

W<sub>6</sub> 为 D/A<sub>2</sub> 零点调节。

W<sub>7</sub> 为 D/A<sub>2</sub> 满度调节。

5.9.3 模入部分调整：

凡改变模入工作方式，如果采样结果偏差大于 20mV 以上的，需要对模入部分进行调整。

①零点调整：使任一通道与模拟地短接，并按实际需要设置好程控增益和通道代码运行程序对该通道采样。调整 W<sub>1</sub> 使 A/D 转换读数等于零。

②A/D 转换满度调整：在任一通道接入一接近正满度的电压信号，运行程序对该通道采样。调整 W<sub>2</sub> 使 A/D 转换读数等于或接近外信号电压。

③A/D 转换双极性偏移调整：在单极性方式时，W<sub>3</sub> 可用于零点辅助调整。在双极性方式时，如果误差较大，可在外端口分别加上正负电压信号，调整 W<sub>3</sub> 使其对称。

5.9.4 模出部分调整：

凡改变模出部分的方式和量程后，如果输出结果误差较大，需要对模出部分进行调整。

①零点调整：在单极性方式时调整 W<sub>4</sub> (D/A<sub>1</sub>) 或 W<sub>6</sub> (D/A<sub>2</sub>) 使其偏差最小。

②满度调整：在零点调整正常情况下，如果满度偏差较大，可通过调整 W<sub>5</sub> (D/A<sub>1</sub>) 或 W<sub>7</sub> (D/A<sub>2</sub>) 使满度符合要

求。

## 6. 驱动程序简介：

PC-6000 系列演示程序及驱动程序是为 PC-6000 系列多功能工控采集板配制的工作在中西文 Windows 95/ 98/ NT 环境下的一组驱动程序以及使用该驱动程序组建的一个演示程序，可以方便地使用用户在中西文 Windows 环境下检测硬件的工作状态以及帮助软件开发人员在常用的 C\C++, Visual Basic, Delphi, Borland C++ Builder, Borland Pascal for windows 等开发环境中使用 PC-6000 系列工控采集板进行数据采集和过程控制等工作. 驱动程序是一个标准动态链接库 (DLL 文件)。它的输出函数可以被其它应用程序在运行时直接调用。用户的应用程序可以用任何一种可以使用 DLL 链接库的编程工具来编写。 每种板卡依据其自身功能的不同具有不同的输出函数和参数定义。

### 驱动程序输出函数定义：

所列函数的说明格式为 VC++6.0 环境下 PC6000.D11 库函数的原函数格式，无论使用哪一种开发工具，务必请注意数据格式的匹配及函数的返回类型，本说明中所使用的数据类型定义如下：

short     ~   16 位带符号数

double    ~   8 字节浮点数

### 模拟量输入部分：

\* 函数： double APIENTRY AI6312ASingle(short nAdd, short nCha, short nAmp, short AIMode)

功能： 进行某一通道的模拟量数据采集。

参数： nAdd     基地址

        nCha     通道号： 0 - 15(单端), 0 - 7(双端)

        nAmp     设置程控放大倍数：    0 -- \*1  
                                          1 -- \*10  
                                          2 -- \*100  
                                          3 -- \*1000

        AIMode    输入方式：  0 -- 原码值  
                                  1 -- 0, 10v  
                                  2 -- -5v, +5v  
                                  3 -- -10v, +10v

\* 函数： void APIENTRY AI6312AA11Single(short nAdd, short nAmp, short AIMode, double \*p)

功能： 单端输入方式下，全部 16 通道的模拟量数据采集。

参数： nAdd     基地址

        nAmp     设置程控放大倍数：    0 -- \*1  
                                          1 -- \*10  
                                          2 -- \*100  
                                          3 -- \*1000

        AIMode    输入方式：  0 -- 原码值  
                                  1 -- 0, 10v  
                                  2 -- -5v, +5v  
                                  3 -- -10v, +10v

        p         指向 16 个通道的采集结果的起始地址

\* 函数： void APIENTRY AI6312AA11Double(short nAdd, short nAmp, short AIMode, double \*p)

功能： 双端输入方式下，全部 8 通道的模拟量数据采集。

参数： nAdd     基地址

        nAmp     设置程控放大倍数：    0 -- \*1  
                                          1 -- \*10  
                                          2 -- \*100  
                                          3 -- \*1000

        AIMode    输入方式：  0 -- 原码值  
                                  1 -- 0, 10v  
                                  2 -- -5v, +5v  
                                  3 -- -10v, +10v

        p         指向 8 个通道的采集结果的起始地址

\* 函数: double APIENTRY AI6312BSingle(short nAdd, short nCha, short nAmp, short AIMode)

参数: nAdd 基地址

nAmp 设置程控放大倍数: 0 -- \*1

2 -- \*4

3 -- \*8

AI Mode 输入方式: 0 -- 原码值

$$2 \text{ --- } -5\text{V, +}$$

3 -- -10V, +10V

\* 函数: void APIENTRY AI6312BAllSingle(short nAdd, short nAmp, short AIMode, double \*p)

参数: nAdd 基地址

1 -- \*2

2 -- \*4

3 -- \*8

AI Mode 输入方式: 0 -- 原码值

2 --  $-5v$ , +!

3 --- -10V, +10V

道的采集结果的趋

1. *Journal of the American Medical Association*, 2000; 284: 2689-2695.

\* 函数: void APIENTRY AI6312BA11Double(short nAdd, short nAmp, short AIMode, double \*p)

参数: nAdd 基地址

1 -- \*2

2 -- \*4

3 -- \*8

原码值

$$1 - 0,10v$$
$$3 \rightarrow -10v,$$

道的采集结果的

部分:

模拟量输出部分:

功能：进行某一通道的模拟量数据输出操作。

参数: nAdd      基地址

nCha 通道号: 0-1

nValue 输出数据： 单位为毫伏(电压方式) 或微安(电流方式)。

DAMode 输出方式: 0 -- 原码值

1 -- 0, 10v

$$2 \rightarrow 0,5v$$

3 -- 1,5v

4 -- -5, 5v

5 -- -10, +10v

6 -- -2.5, 2.5v

7 — 0,10mA

8 -- 4, 20mA

返回：无返回值。

如有需要使用 Windows 系列及 LabVIEW 驱动程序的用户可向本公司索取, 请注明所使用的操作系统和开发软件。

## 7. 编程举例:

7.1 对通道 1 连续采样 100 次程序启动和查询。本程序可用于 A / D 部分调校。

### 7.1.1 BASIC 语言:

```
10 CLS ; 清屏
20 ADD=&H300 ; 板基地址设为 0300H
30 A=INP(ADD+3) ; 清转换及中断标志
40 CH=0 ; 对通道 1 采样, 程控增益为 1 倍。
50 OUT(ADD+0), CH ; 送通道代码
60 FOR T=0 TO 99 ; 设采样次数
70 OUT(ADD+1), 0 ; 启动 A / D, 所送数值无关
80 IF INP(ADD+2)>=128 THEN 80 ; 查询 A / D 转换状态
90 H=INP(ADD+2) ; 转换结束, 读高 4 位结果
100 L=INP(ADD+3) ; 读低 8 位结果
110 V=(H×256+L)×10000/4096 ; 将结果转换为十进制数据
120 PRINT V; "mV" ; 显示结果, 用 "mV" 表示
130 NEXT T ; 循环 100 次
140 END
```

注: 如果是双极性信号, 则 110 句改为:

$V=(H \times 256 + L) \times 10000 / 4096 - 5000$

### 7.1.2 C 语言:

```
#include "stdio.h"
#include "dos.h"
#include "conio.h"
main()
{
    int ch; /* 定义通道变量 */
    float value[100]; /* 定义数组变量 */
    int dl,dh,i,j,base; /* 定义过程变量 */
    clrscr(); /* 清屏 */
    base=0x300; /* 设板基地址=300H */
    dl=inportb(base+3) /* 清转换及中断标志 */
    printf("Input channle number:"); /* 输入通道号 */
    scanf("%d",&ch);
    outportb(base, ch); /* 送通道代码 */
    for(j=0;j<100;j++){ /* 设采样次数 */
        for(i=0;i<100;i++){ /* 延时, 常数由机型和*/
            /* 程控增益倍数决定 */
            outportb(base+1, 0); /*启动 A / D, 所送数值无关*/
            do{ /* 查询 A / D 转换状态 */
                ;
            }while(inportb(base+2)>=128);
            dh=inportb(base+2); /*转换结束, 读高 4 位结果 */
            dl=inportb(base+3); /* 读低 8 位结果 */
            value[j]=(dh*256+dl)*10.0/4096.0-5.0; /* 将结果转换为电压值*/
        }
        for(j=0;j<100;j++) /* 显示结果 */
            printf("%f ", value[j]);
    }
}
```

7.2 循环采集 A / D 16 通道, 程序启动和查询。

C 语言程序:

```

#include "stdio.h"
#include "dos.h"
#include "conio.h"

main()
{
    int ch;                /* 定义通道变量 */
    float value[16];       /* 定义数组变量 */
    int dl, dh, i, base;   /* 定义过程变量 */
    clrscr();              /* 清屏 */
    base=0x300;             /* 设板基地址=300H */
    dl=inportb(base+3)     /* 清转换及中断标志 */
    for(ch=0;ch<=15;ch++) { /* 定义循环通道数 */
        outportb(base, ch); /* 送通道代码 */
        for(i=0;i<100;i++); /* 延时,常数由机型决定 */
        outportb(base+1, 0); /* 启动 A / D, 所送数值无关 */
        do {                /* 查询 A / D 转换状态 */
            ;
        } while (inportb(base+2)>=128);
        dh=inportb(base+2); /* 转换结束, 读高 4 位结果 */
        dl=inportb(base+3); /* 读低 8 位结果 */
        value[ch]=(dh*256+dl)*10.0/4096.0-5.0; /* 将结果转换为电压值 */
    }                        /* 下一个通道 */
    for(ch=0;ch<=15;ch++)   /* 显示结果 */
        printf("%f ", value[ch]);
}

```

7.3 使 D/A<sub>1</sub>、D/A<sub>2</sub> 分别输出 0V; 10V; 3.333V; 6.666V; 2.000V; 8.000V。D/A 工作方式为单极性 0~10V。本程序可用于 D/A 部分的调校。

```

10 CLS                      ; 清屏
20 ADD=&H110                 ; 板基地址设为 0110H
30 DAH=ADD+4;
    DAL=ADD+5                 ; 设 D/A1 高、低字节端口地址
40 FOR DA=1 TO 2             ; 2 路 D/A
50 FOR I=1 TO 6              ; 设数据指针长度
60 READ A                    ; 取低字节数据
70 OUT DAH, A                ; 送出
80 READ A                    ; 取高字节数据
90 OUT DAL, A                ; 送出
100 READ B                   ; 取显示用数据
110 A=INP(ADD+0)              ; 启动 D/A 转换, 读出的数据无关
120 PRINT "TEST D/A"; DA;    ;
    "OUT"; B; "V"            ; 显示
130 IF INKEY $ =" " THEN 130 ; 等待, 按任一键继续
140 NEXT I                   ; 循环送 6 组数据
150 DAH=DAH+2; DAL=DAL+2     ; 设下一路 D/A 高低字节端口地址
160 RESTORE                  ; 恢复数据指针
170 NEXT DA                  ; 循环送完 2 路 D/A
180 DATA 0, 0, 255, 240, 10.000,
    85, 80, 3.333, 170, 160, 6.666,
    51, 48, 2.000, 204, 192, 8.000
190 END

```

7.4 在 Windows 95/98 环境下, 使用 MicroSoft Visual Basic 6.0 开发环境, 采取调用驱动程序的输出函数的方法定时循环采集多个 A/D 通道, 并对 D/A 进行操作。

注意: 在 VB 6.0 中, 数据类型 Integer 为 16 位带符号整数; double 为 8 字节浮点数。

首先创建一个窗口, 名为 Form1。设置一个定时器, 名为 Timer1; 一个 Text1; 一个 Text2 数组, Text2[]。

’ 模入部分

```
Private Declare Function AI6312ASingle Lib "pc6000.dll" (ByVal nAdd As Integer, ByVal nCha As Integer, _
    ByVal nAmp As Integer, ByVal AIMode As Integer) As Double
Private Declare Sub AI6312AA11Single Lib "pc6000.dll" (ByVal nAdd As Integer, ByVal nAmp As Integer, _
    ByVal AIMode As Integer, ByRef p As Double)
Private Declare Sub AI6312AA11Double Lib "pc6000.dll" (ByVal nAdd As Integer, ByVal nAmp As Integer, _
    ByVal AIMode As Integer, ByRef p As Double)
Private Declare Function AI6312BSingle Lib "pc6000.dll" (ByVal nAdd As Integer, ByVal nCha As Integer, _
    ByVal nAmp As Integer, ByVal AIMode As Integer) As Double
Private Declare Sub AI6312BA11Single Lib "pc6000.dll" (ByVal nAdd As Integer, ByVal nAmp As Integer, _
    ByVal AIMode As Integer, ByRef p As Double)
Private Declare Sub AI6312BA11Double Lib "pc6000.dll" (ByVal nAdd As Integer, ByVal nAmp As Integer, _
    ByVal AIMode As Integer, ByRef p As Double)
```

’ 模出部分

```
Private Declare Sub AO6312Single Lib "pc6000.dll" (ByVal nAdd As Integer, ByVal nCha As Integer, _
    ByVal nValue As Integer, ByVal DAMode As Integer)
```

```
Dim a(16) As Double ’ 数组元素个数应>=16
```

```
Private Sub Timer1_Timer()
```

```
    Dim i As Integer
```

```
    ’ 多通道采集 16 通道, 0-15 通道采集结果分别存放在 a(0)-a(15)中
```

```
    Call AI6312AA11Single(256, 0, 1, a(0))
```

```
    For i = 0 To 15
```

```
    ,      Text2(i) = a(i)      ’ 界面显示 16 通道数据
```

```
    Next i
```

```
    ’ 单通道采集第 0 通道
```

```
    Text1 = AI6312ASingle(256, 0, 0, 1)
```

```
    ’ D/A 输出
```

```
    For i = 0 To 1
```

```
        Call AO6312Single(256, i, AoOutValue, 1)
```

```
    Next i
```

```
    If AoOutValue = 10000 Then
```

```
        AoOutValue = 0
```

```
    Else
```

```
        AoOutValue = AoOutValue + 100      ’ 改变 D/A 输出电压值
```

```
    End If
```

```
End Sub
```

## 附 A. 名词注释:

### 1. 单端输入方式:

各路输入信号共用一个参考电位, 即各路输入信号共地, 这是最常用的接线方式。使用单端输入方式时, 地线比较稳定, 抗干扰能力较强, 建议用户尽可能使用此种方式。

## 2. 双端输入方式：

各路输入信号各自使用自己的参考电位，即各路输入信号不共地。如果输入信号来自不同的信号源，而这些信号源的参考电位(地线)略有差异，可考虑使用这种接线方式。使用双端输入方式时，输入信号易受干扰，所以，应加强信号线的抗干扰处理，同时还应确保模拟地以及外接仪器机壳接地良好。而且特别注意的是，所有接入的信号，不论是高电位还是低电位，其电平相对于模拟地电位应不超过 +12V 及 -5V，以避免电压过高造成器件损坏。

## 3. 单极性信号：

输入信号相对于模拟地电位来讲，只偏向一侧，如输入电压为 0~10V。

## 4. 双极性信号：

输入信号相对于模拟地电位来讲，可高可低，如输入电压为 -5V ~ +5V。

## 5. 码制：

模拟量信号转换为数字量后，形成一组由 0 开始的连续数字，每一个数字对应着一个特定的模拟量值，这种对应关系称为编码方法或码制。依据输入信号的不同分为单极性原码与双极性偏移码。单极性输入信号对应着单极性原码，双极性信号对应着双极性偏移码。

## 6. 单极性原码：

以 12 位 A/D 为例，输入单极性信号 0~10V。转换后得到 0 ~ 4095 的数字量，数字量 0 对应的模拟量为 0V，数字量 4095 对应的模拟量为 10V，这种编码方法称为单极性原码，其数字量值与模拟电压值的对应关系可描述为：

$$\text{模拟电压值} = \text{数码(12位)} \times 10(\text{V}) / 4096 \quad (\text{V})$$

$$\text{即：1LSB (1 个数码位)} = 2.44\text{mV}$$

## 7. 双极性偏移码：

以 12 位 A/D 为例，输入双极性信号 -5~+5V。转换后得到 0 ~ 4095 的数字量，数字量 0 对应的模拟量为 -5V，数字量 4095 对应的模拟量为 +5V，这种编码方法称为双极性偏移码，其数字量值与模拟电压值的对应关系可描述为：

$$\text{模拟电压值} = \text{数码(12位)} \times 10(\text{V}) / 4096 - 5 \quad (\text{V})$$

$$\text{即：1LSB (1 个数码位)} = 2.44\text{mV}$$

此时 12 位数码的最高位(DB<sub>11</sub>)为符号位，此位为 0 表示负，1 表示正。偏移码与补码仅在符号位上定义不同，如果反向运算，可以先求出补码再将符号位取反就可得到偏移码。

## 8. A/D 转换速率：

表明 A/D 转换芯片的工作速度。如对 BB774 来讲，完成一次转换所需要的时间是 10 微秒，则它的转换速率为 100 KHz。

## 9. 通过率：

指 A/D 采集卡对某一路信号连续采集时的最高采集速率。

## 10. 初始地址：

使用板卡时，需要对卡上的一组寄存器进行操作，这组寄存器占用数个连续的地址，一般将其中最低的地址值定为此卡的初始地址，这个地址值需要使用卡上的拨码开关来设置。

## 附 B. 产品清单及保修：

产品清单：

1. PC-6312 模入模出接口卡壹块。
2. 37 芯 D 型插头壹套。

本产品自售出之日起一年内，凡用户遵守贮存、运输及使用要求，而产品质量低于技术指标的，凭保修单免费维修。因违反操作规定和要求而造成损坏的，需交纳器件和维修费。