

PC-6311D 模入模出接口卡技术说明书

1. 概述:

PC-6311D 模入模出接口卡适用于具有 ISA 总线的 PC 系列微机, 具有很好的兼容性, 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。

PC-6311D 模入模出接口卡安装使用方便, 程序编制简单。其模入模出及 I/O 信号均由卡上 37 芯 D 型插头及另配的转换插头与外部信号源和设备连接。对于模入部分, 用户可根据实际需要选择单端或双端输入方式。对于模出部分, 用户可根据控制对象的需要选择电压或电流输出方式以及不同的量程。

2. 主要技术参数:

2.1 模入部分

2.1.1 输入通道数: (标*为出厂标准状态, 下同)

单端 32 路; * / 双端 16 路

2.1.2 输入信号范围:

0V~10V*; / $\pm 5V$

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 A/D 转换速度: $10\mu S$

2.1.6 A/D 启动方式:

程序启动 / 外触发启动

2.1.7 A/D 转换结束识别:

程序查询 / 中断方式

2.1.8 A/D 转换非线性误差: $\pm 1LSB$

2.1.9 A/D 转换输出码制:

单极性原码* / 双极性偏移码

2.2.10 系统综合误差: $\leq 0.2\%$ FSR

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: 0~5V; 0~10V*; $\pm 5V$; $\pm 2.5V$

电流方式: 0~10mA; 4~20mA

2.2.3 输出阻抗: $\leq 2\Omega$ (电压方式)

2.2.4 D/A 转换器件: DAC1210

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: $\leq 0.2\%$ FSR

电流方式: $\leq 1\%$ FSR

2.2.9 电流输出方式负载电阻范围:

使用机内 +12V 电源时: 0~250 Ω

外加 +24V 电源时: 0~750 Ω

2.3 数字量输入输出部分:

2.3.1 DI: 8 路; TTL 标准电平

2.3.2 DO: 8 路; TTL 标准电平; 有输出锁存功能

2.4 电源功耗:

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

+12V(±10%)≤100mA;

−5V(±10%)≤10mA

2.5 使用环境要求:

工作温度: 10℃~40℃;

相对湿度: 40%~80%;

贮存温度: −55℃~+85℃

2.6 外型尺寸: (不含档板)

长×高=182.6mm×106.7 mm (7.2 英寸×4.2 英寸)

3. 工作原理:

PC-6311D 模入模出接口卡主要由模数转换电路、数模转换电路、数字量输入输出电路、接口控制逻辑电路构成。

3.1 工作原理框图: PC-6311D 模入模出接口卡工作原理框图见图 1。

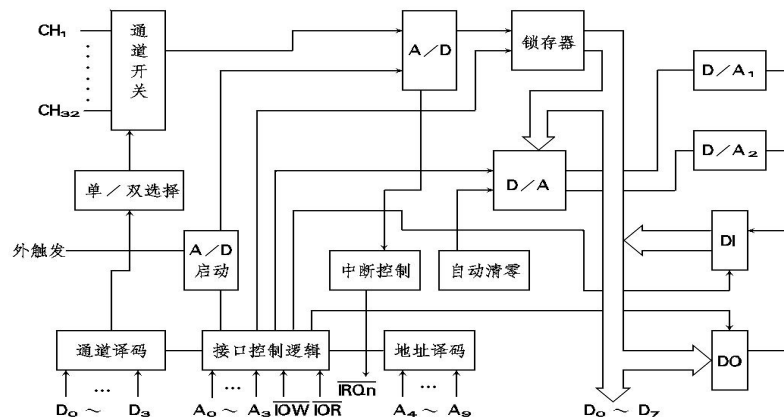


图 1 工作原理框图

3.2 模入部分:

外部模拟信号经多路转换开关选择后送入放大器处理。放大器前后设有单 / 双端输入选择跨接器 KJ_1 、 KJ_2 和转换码制选择跨接器 KJ_3 。处理后的信号送入模数转换器进行转换, 其转换状态和结果可用程序查询和读出。模数转换器的启动也可用外部触发方式启动。转换结束信号也可用中断方式通知 CPU 进行处理。

3.3 模出部分:

模拟量输出部分由 DAC1210 D / A 转换器件和有关的基准源、运放、阻容件和跨接选择器组成。依靠改变跨接套的连接方式, 可分别选择电压或电流输出方式。当采用电流输出方式时, 本卡可直接外接 II、III 型执行器。D / A 部分的各个通道可分别按不同的输出方式和范围由用户自行选择, 并具有加电自动清零功能。

3.4 数字量输入输出部分:

数字量输入输出电路由输入缓冲器和输出锁存器及相关电路组成, 可分别输入输出 8 位 TTL 电平信号。

4. 安装及使用注意:

本卡的安装十分简便, 只要将主机机壳打开, 在关电情况下, 将本卡插入主机的任何一个空余扩展槽中, 同时 CZ2 也需要占用一个空余扩展槽, 再将档板固定螺丝压紧即可。

本卡采用的模拟开关是 COMS 电路, 容易因静电击穿或过流造成损坏, 所以在安装或用手触摸本卡时, 应事先将人体所带静电荷对地放掉, 同时应避免直接用手接触器件管脚, 以免损坏器件。

禁止带电插拔本接口卡。本卡跨接选择器较多, 使用中应严格按照说明书进行设置操作。设置接口卡开关、跨接套和安装接口带缆时均应在关电状态下进行。

当模入通道不全部使用时, 应将不使用的通道就近对地短接, 不要使其悬空, 以避免造成通道间串扰和损坏通道。

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。

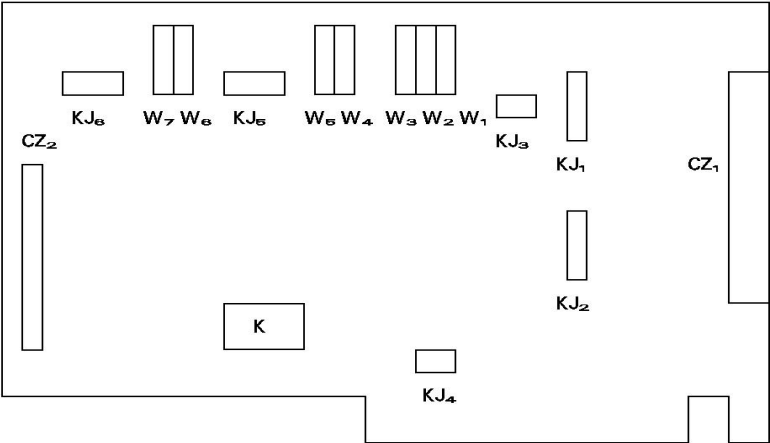


图 2 主要可调整元件位置图

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 100H~1EFH; 210H~2EFH 以及 300H~36FH 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 100H，并从基地址开始占用连续 8 个地址。现举例说明见图 3。

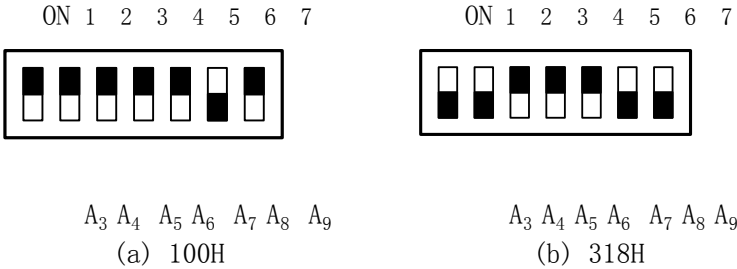


图 3 I / O 基地址选择举例

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方式时通道组成)CZ1 见表 1，CZ2 见表 2。(注：CZ2 需要占用一个 PC 插槽位。)

表 1 输入插座 CZ1 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	20	模拟地
2	CH1 (CH1+)	21	CH17 (CH1-)
3	CH2 (CH2+)	22	CH18 (CH2-)
4	CH3 (CH3+)	23	CH19 (CH3-)
5	CH4 (CH4+)	24	CH20 (CH4-)
6	CH5 (CH5+)	25	CH21 (CH5-)
7	CH6 (CH6+)	26	CH22 (CH6-)
8	CH7 (CH7+)	27	CH23 (CH7-)
9	CH8 (CH8+)	28	CH24 (CH8-)
10	CH9 (CH9+)	29	CH25 (CH9-)
11	CH10 (CH10+)	30	CH26 (CH10-)
12	CH11 (CH11+)	31	CH27 (CH11-)
13	CH12 (CH12+)	32	CH28 (CH12-)
14	CH13 (CH13+)	33	CH29 (CH13-)
15	CH14 (CH14+)	34	CH30 (CH14-)
16	CH15 (CH15+)	35	CH31 (CH15-)
17	CH16 (CH16+)	36	CH32 (CH16-)
18	+5V 输出	37	外触发 E. T
19	模拟地		

表 2 输入输出插座 CZ2 接口定义 (NC 为空脚)

插座引脚号	信号定义	插座引脚号	信号定义
1	D / A ₁ 电压端	20	模拟地
2	D / A ₂ 电压端	21	模拟地
3	NC	22	NC
4	D / A ₁ 电流端	23	+12V 输出
5	D / A ₂ 电流端	24	+12V 输出
6	NC	25	NC
7	+5V 输出	26	+5V 输出
8	DI0	27	DI1
9	DI2	28	DI3
10	DI4	29	DI5
11	DI6	30	DI7
12	NC	31	NC
13	D00	32	D01
14	D02	33	D03
15	D04	34	D05
16	D06	35	D07
17	数字地	36	数字地
18	NC	37	NC
19	NC		

5.4 跨接插座的使用法：

5.4.1 输入单 / 双端方式选择：

KJ₁、KJ₂为单 / 双端输入方式选择，其使用方法见图 4

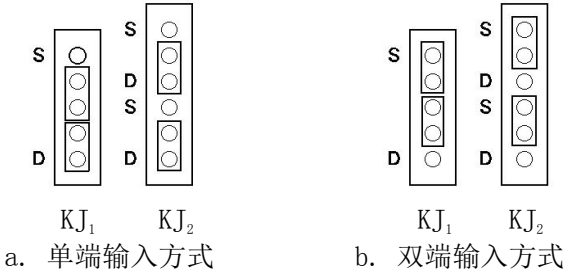


图 4 单 / 双端输入方式选择

5.4.2 转换码制选择：

KJ₃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。

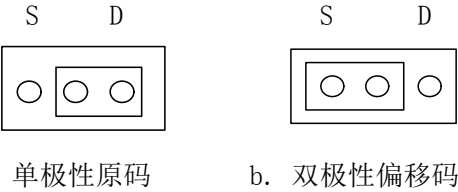


图 5 转换码制选择

5.4.3 D/A 输出方式及范围选择：

KJ₅、KJ₆为 D/A 输出量程选择插座，其中 KJ₅对应 D/A₁，KJ₆对应 D/A₂。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 6。

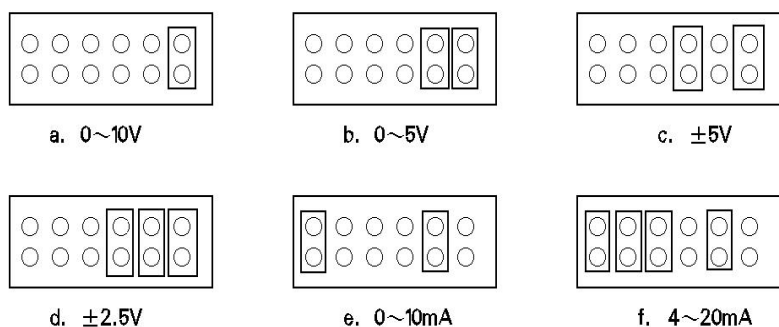


图 6 D / A 输出方式及范围选择

5.4.4 中断方式及中断源选择:

KJ₄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 7。

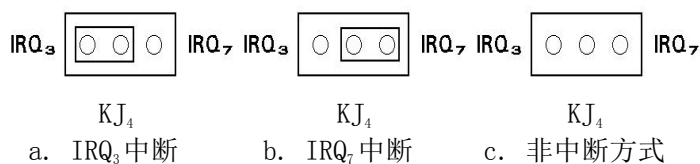


图 7 中断源的选择

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 3:

表 3 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+1	读	查询 A / D 转换状态, 读高 4 位转换结果
基地址+2	读	读低 8 位转换结果, 清除转换状态及中断标志
基地址+3	写	写 I / O 输出数据
基地址+3	读	读 I / O 输入数据
基地址+4	写	写 D / A ₁ 高 8 位数据
基地址+5	写	写 D / A ₁ 低 4 位数据
基地址+6	写	写 D / A ₂ 高 8 位数据
基地址+7	写	写 D / A ₂ 低 4 位数据

5.5.2 通道代码数据格式见表 4:

表 4 通道代码数据格式

通道号	十进制 代码	十六进制 代码	输入方式	通道号	十进制 代码	十六进制 代码	输入方式
1	0	00H	单 / 双	17	16	10H	单
2	1	01H	单 / 双	18	17	11H	单
3	2	02H	单 / 双	19	18	12H	单
4	3	03H	单 / 双	20	19	13H	单
5	4	04H	单 / 双	21	20	14H	单
6	5	05H	单 / 双	22	21	15H	单
7	6	06H	单 / 双	23	22	16H	单
8	7	07H	单 / 双	24	23	17H	单
9	8	08H	单 / 双	25	24	18H	单
10	9	09H	单 / 双	26	25	19H	单
11	10	0AH	单 / 双	27	26	1AH	单
12	11	0BH	单 / 双	28	27	1BH	单
13	12	0CH	单 / 双	29	28	1CH	单

14	13	0DH	单 / 双	30	29	1DH	单
15	14	0EH	单 / 双	31	30	1EH	单
16	15	0FH	单 / 双	32	31	1FH	单

5.5.3 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 5(端口地址为基地址+1):

表 5 A / D 转换状态数据格式(X 表示任意)

操作命令	D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀	A / D 转换状态
读	1	x	x	x	x	x	x	x	正在转换中
读	0	x	x	x	x	x	x	x	转换结束

5.5.4 A / D 转换结果数据格式:

A / D 转换结果数据格式见表 6:

表 6 A / D 转换结果数据格式

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

5.5.5 D / A 转换数据格式:

D / A₁转换数据格式见表 7, D / A₂转换数据格式类同。

表 7 D / A₁转换数据格式 (X 表示任意)

端口地址	操作命令	D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀	意 义
基地址+4	写	DB ₁₁	DB ₁₀	DB ₉	DB ₈	DB ₇	DB ₆	DB ₅	DB ₄	高 8 位数据
基地址+5	写	DB ₃	DB ₂	DB ₁	DB ₀	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:

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

$$\text{即: } 1\text{LSB} = 2.44\text{mV}$$

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(DB₁₁)为符号位,“0”表示负,“1”表示正。

偏移码与补码仅在符号位上定义不同,可以先求出补码再将符号位取反就可得到偏移码。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:

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

$$\text{即: } 1\text{LSB} = 2.44\text{mV}$$

5.7 外触发信号 E.T 的要求:

本卡的模入部分可以在外触发方式下工作。每当 E.T 有一个低电平时, A / D 就启动转换一次。使用该方式时,应注意 E.T 信号必须符合 TTL 电平标准,其波形参见图 8。

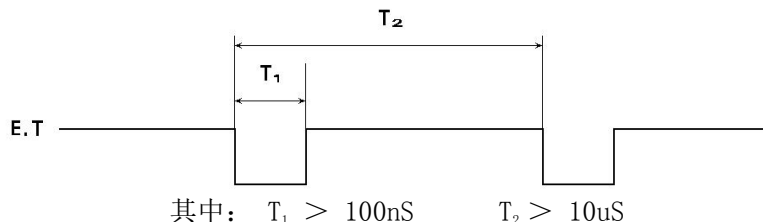


图 8 E.T 信号波形图

5.8 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 KJ₄ 的位置可以选用 IRQ₃ 中断或 IRQ₇ 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。

5.9 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。

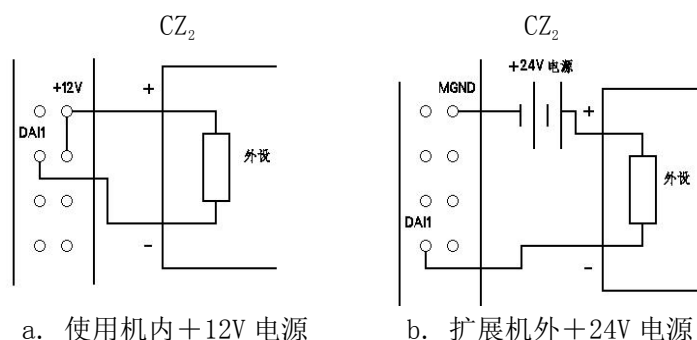


图 9 电流输出方式使用方法

5.10 调整与校准:

5.10.1 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。

5.10.2 各电位器功能说明:

- W_1 为输入放大器零点调节。
- W_2 为 A / D 转换器满度调节。
- W_3 为 A / D 转换器双极性偏移调节。
- W_4 为 D / A_1 零点调节。
- W_5 为 D / A_1 满度调节。
- W_6 为 D / A_2 零点调节。
- W_7 为 D / A_2 满度调节。

5.10.3 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通道代码运行程序对该通道采样。用电压表测量 OP-37 运放的第 6 脚, 调整 W_1 使电压表读数小于 100 μ V。

② A / 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_2 使 A / D 转换读数值等于或接近外信号电压。

③ A / D 转换双极性偏移调整: 在单极性方式时, W_3 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_3 使其对称。

5.10.4 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_4 (D / A_1) 或 W_6 (D / A_2) 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_5 (D / A_1) 或 W_7 (D / A_2) 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
unsigned char ~ 8 位无符号数

模拟量输入部分:

- * 函数: short APIENTRY AI6311Single(short nAdd, short nCha, 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 - 31(单端), 0 - 15(双端)
 AImode 输入方式: 0 -- 原码值
 1 -- 0, 10v
 2 -- -5v, +5v

* 函数: void APIENTRY AI6311AllSingle(short nAdd, short AImode, short *p)

功能: 单端输入方式下, 全部 32 通道的模拟量数据采集。

参数: nAdd 基地址
 AImode 输入方式: 0 -- 原码值
 1 -- 0, 10v
 2 -- -5v, +5v
 p 指向 32 个通道的采集结果的起始地址

* 函数: void APIENTRY AI6311AllDouble(short nAdd, short AImode, short *p)

功能: 双端输入方式下, 全部 16 通道的模拟量数据采集。

参数: nAdd 基地址
 AImode 输入方式: 0 -- 原码值
 1 -- 0, 10v
 2 -- -5v, +5v
 p 指向 16 个通道的采集结果的起始地址

模拟量输出部分:

* 函数: void APIENTRY AO6311Single(short nAdd, short nCha, short nValue, short DAMode)

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 -- 0, 5v
 3 -- -5, 5v
 4 -- -2.5, 2.5v
 5 -- 0, 10mA
 6 -- 4, 20mA

返回: 无返回值。

数字量输入/输出部分:

* 函数: unsigned char APIENTRY DI6311Bit(short nAdd, short nBit)

功能: 采集某一位数字量输入信号的状态。

参数: nAdd 基地址
 nBit 通道号: 0-7

* 函数: unsigned char APIENTRY DI6311All(short nAdd)

功能: 采集全部通道(8 路) 数字量输入信号的状态。

参数: nAdd 基地址

返回: 返回值为8个输入信号的状态

* 函数: void APIENTRY DO6311Bit(short nAdd, short nBit, unsigned char nState)

功能: 进行某一个通道的数字量数据输出操作。

参数: nAdd 基地址
 nBit 通道号: 0-7
 nState 1表示将输出高电平, 0表示将输出低电平。

返回: 无返回值

* 函数: void APIENTRY DO6311All(short nAdd, unsigned char nState)

功能: 同时进行所有8个通道的数字量数据输出操作。

参数: nAdd 基地址

nState 各位输出状态, nState的D0代表Bit0, D3代表Bit3。
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+2) ; 清转换及中断标志
40 CH=0 ; 对通道 1 采样
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+1)>=128 THEN 80 ; 查询 A / D 转换状态
90 H=INP(ADD+1) ; 转换结束, 读高 4 位结果
100 L=INP(ADD+2) ; 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+2) /* 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+1)>=128);
        dh=inportb(base+1); /* 转换结束, 读高 4 位结果 */
        dl=inportb(base+2); /* 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 32 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[32]; /* 定义数组变量 */
    int dl,dh,i,base; /* 定义过程变量 */
    clrscr(); /* 清屏 */
    for(ch=0;ch<=31;ch++) { /* 定义循环通道数 */
        base=0x300; /* 设板基地址=300H */
        dl=inportb(base+2) /* 清转换及中断标志 */
        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+1)>=128);
        dh=inportb(base+1); /* 转换结束, 读高 4 位结果 */
        dl=inportb(base+2); /* 读低 8 位结果 */
        value[ch]=(dh*256+dl)*10.0/4096.0-5.0; /* 转换结果 */
    } /* 下一个通道 */
    for(ch=0;ch<=31;ch++) /* 显示结果 */
        printf("%f", value[ch]);
}
```

7.3 对通道 1 外触发启动，程序查询取数，采样 100 组。

```
10 CLS ; 清屏
20 ADD=&H280 ; 设板基地址为 0280H
30 A=INP(ADD+2) ; 清转换及中断标志
40 CH=0 ; 对通道 1 采样，
50 DIM A1(100) ; 定义数组长度
60 FOR T=0 TO 99 ; 定义循环次数
70 OUT (ADD+0), CH ; 送通道代码
80 IF INP(ADD+1)<128 THEN 80 ; 等待外触发信号
90 IF INP(ADD+1)>=128 THEN 90 ; 等待转换结束
100 H=INP(ADD+1) ; 取高 4 位结果
110 L=INP(ADD+2) ; 取低 8 位结果
120 A1(T)=(H×256+L)×10000/4096 ; 处理后存入数组
130 NEXT T ; 循环
```

7.4 读写数字量：

```
10 ADD=&H300 ; 板基地址设为 0300H
20 DO=XX ; 设数据输出为 XX
30 OUT(ADD+3), DO ; 写出并锁存
40 DI=INP(ADD+3) ; 读入数字量状态
```

```

50 PRINT DI          ; 显示
60 END

```

7.5 使 D/A₁、D/A₂ 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)         ; 设 D/A1 高、低字节
35 DAL=(ADD+5)         ; 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              ; 取高字节数据
70 OUT DAH, A          ; 送出
80 READ A              ; 取低字节数据
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, 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 0,0,0,255,240,10.000,
    85,80,3.333,170,160,6.666,
    51,48,2.000,204,192,8.000
190 END

```

7.6 在 Windows 95/98/NT 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 及数字 I/O 进行操作。

注意: 在 VB 6.0 中, 数据类型 Integer 为 16 位带符号整数; Byte 为 8 位无符号整数。

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 Text3; 一个 Text2 数组, Text2[]。

```

' 模入部分
Private Declare Function AI6311Single Lib "pc6000.dll" (ByVal nAdd As Integer, ByVal nCha As Integer, ByVal _
    AIMode As Integer) As Integer
Private Declare Sub AI6311AllSingle Lib "pc6000.dll" (ByVal nAdd As Integer, ByVal AIMode As Integer, ByRef p _
    As Integer)
Private Declare Sub AI6311AllDouble Lib "pc6000.dll" (ByVal nAdd As Integer, ByVal AIMode As Integer, ByRef p _
    As Integer)

' 模出部分
Private Declare Sub AO6311Single Lib "pc6000.dll" (ByVal nAdd As Integer, ByVal nCha As Integer, ByVal nValue _
    As Integer, ByVal DAMode As Integer)

' 数字量
Private Declare Function DI6311Bit Lib "pc6000.dll" (ByVal nAdd As Integer, ByVal nBit As Integer) As Byte
Private Declare Function DI6311All Lib "pc6000.dll" (ByVal nAdd As Integer) As Byte
Private Declare Sub DO6311Bit Lib "pc6000.dll" (ByVal nAdd As Integer, ByVal nBit As Integer, ByVal nState As Byte)
Private Declare Sub DO6311All Lib "pc6000.dll" (ByVal nAdd As Integer, ByVal nGroup As Byte)

Dim a(32) As Integer ' 数组元素个数应>=32

Private Sub Timer1_Timer()

```

```

Dim i as integer
'多通道采集 32 通道, 0-31 通道采集结果分别存放在 a(0)-a(31)中
Call AI6311A11Single(256, 1, a(0))
For i = 0 To 31
    Text2(i) = a(i)    ' 界面显示 32 通道数据
Next i

'单通道采集第 0 通道
Text1 = AI6311Single(256, 0, 1)

'D/A 输出
For i = 0 To 1
    Call AO6311Single(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

'I/O 输出
Call DO6311A11(256, DoOutValue)
If DoOutValue = 0 Then
    AoOutValue = 255
Else
    AoOutValue = 0    ' 改变数字量输出状态.
End If

'I/O 输入
Text3 = DI6311A11(256)    ' text3 用于显示所有通道状态

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：

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

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

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：

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

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

此时 12 位数码的最高位(DB_{11})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-6311D 模入模出接口卡壹块。

2. 37 芯 D 型插头两套及 40 芯至 37 芯 D 型插头转换线壹套。

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。