

ART2002 数据采集卡

硬件使用说明书



北京阿尔泰科技发展有限公司

产品研发部修订



目 录

目 录	1
第一章 功能概述	2
第一节、应用	2
第二节、技术指标	2
一、模拟信号输入部分	2
二、A/D 转换电路部分	2
三、中断部分	2
四、DMA 部分	2
第二章 元件布局图及简要说明	3
第一节、主要元件布局图	3
第二节、关于模拟输入引脚 37 芯插头 XS1 的管脚定义	3
第三节、短路套设置及数据格式	6
一、板基地址选择	6
二、板内地址分配	8
三、模拟信号输入方式选择	10
四、模拟输入信号的连接方式	10
第三章 工作原理	12
第一节、A/D 工作模式	12
第二节、A/D 转换启动控制	12
第三节、板上转换定时器	12
第四章 编程举例	14
第一节、查询方式	14
第二节、中断方式	16
第三节、DMA 方式	19
第五章 产品的应用注意事项、校准、保修	22
第一节、使用注意事项	22
第二节、A/D 的校准	22
第三节、保修	22

第一章 功能概述

ART2002是一款基于PC104总线而开发的高性能A/D数据采集卡，板上装有16位100K采样速率的A/D转换器；提供了单端32路/双端16路的AD模拟信号输入通道；配有先进先出FIFO存储器；开关量输入、输出各16路，并具有二种采样模式，伪同步采集模式和分频采集模式。

ART2002 板支持内触发和外触发，支持软件查询、DMA、中断三种数据传输方式。

ART2002A 除板上装有 16 位 200K 采样速率的 A/D 转换器外，其余同 ART2002。

第一节、应用

- ◆野外测控
- ◆信号采集
- ◆医疗设备
- ◆伺服控制
- ◆电子产品质量检测

第二节、技术指标

一、模拟信号输入部分

- ◆模拟通道输入数： 32路单端/16路双端模拟信号输入
- ◆模拟电压输入范围： $\pm 10V$
- ◆模拟输入阻抗： $100M\Omega$
- ◆模拟输入共模电压范围： $>\pm 2V$
- ◆放大器建立时间： $2\mu S$

二、A/D转换电路部分

- ◆A/D分辨率： 16Bit(65536)
- ◆非线性误差： $\pm 1LSB$ (最大)
- ◆转换时间： $2.5\mu S$
- ◆系统测量精度(满量程)： 0.1%
- ◆采样速率： 400K

三、中断部分

- ◆中断申请通道数： 1路(软件选择)
- ◆中断申请级别： IRQ5、IRQ6、IRQ7
- ◆中断信号有效电平： 高电平有效
- ◆中断申请信号电平特性： TTL兼容

四、DMA部分

- ◆DMA申请和应答通道数： 1路(软件选择)
- ◆DMA申请和应答申请级别： DMA5、DMA6、DMA7
- ◆DMA申请有效电平： 高电平有效
- ◆DMA应答有效电平： 低电平有效
- ◆DMA申请和应答电平特性： TTL兼容

注： 以上技术指标未注明者均为典型值。

第二章 元件布局图及简要说明

第一节、主要元件布局图

图2.1和图2.2分别为ART2002板的两个不同版本（它们的区别：一个版本为D型头，另一个版本为40芯双排带锁弯针，其它所有功能不变）的主要元件位置图，此元件位置图上的开关和跳线设置为出厂标准设置。设置为：板基地址=768（300H），单端输入方式，模拟输入范围±10V。

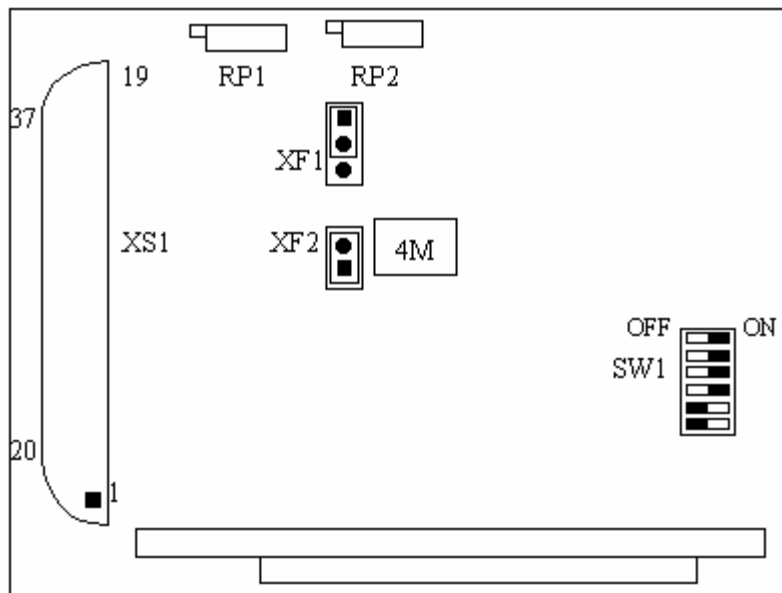


图2.1 ART2002板的主要元件位置图

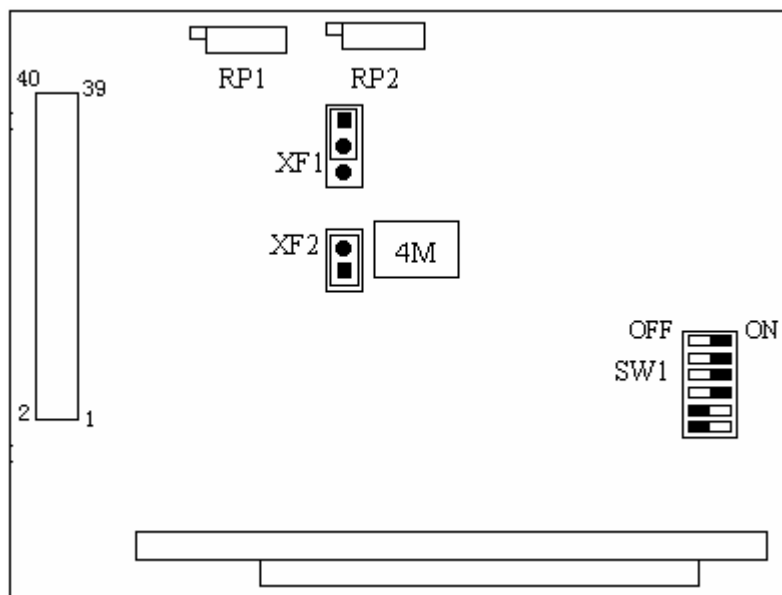
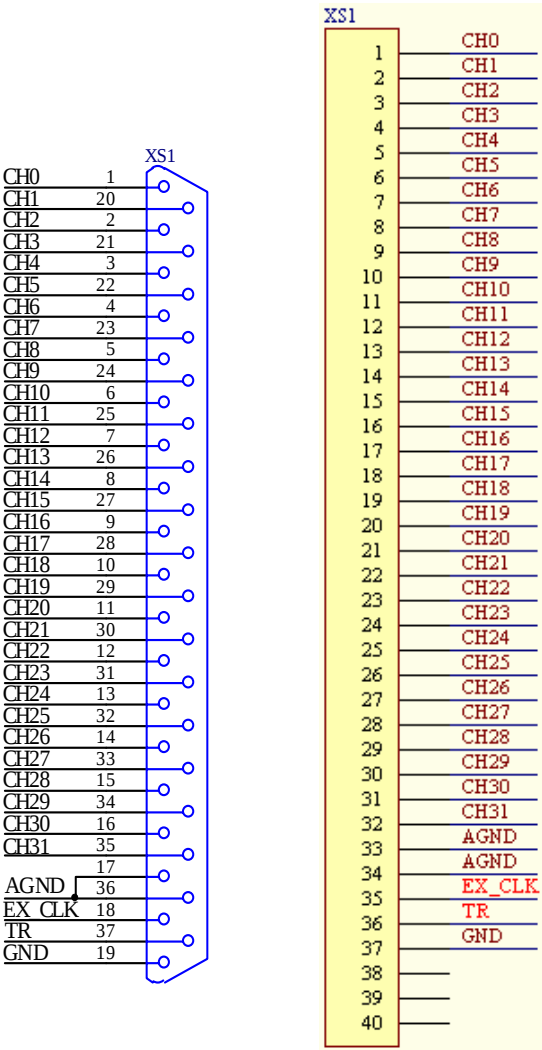


图2.2 ART2002板的主要元件位置图

第二节、关于模拟输入引脚 37 芯插头 XS1 的管脚定义

（左：第一版；右：第二版）



第一版本表格说明：

管脚号	名称	管脚功能	特性	管脚号	名称	管脚功能	特性
1	CH0	模拟信号输入通道0	IN	20	CH1	模拟信号输入通道1	IN
2	CH2	模拟信号输入通道2	IN	21	CH3	模拟信号输入通道3	IN
3	CH4	模拟信号输入通道4	IN	22	CH5	模拟信号输入通道5	IN
4	CH6	模拟信号输入通道6	IN	23	CH7	模拟信号输入通道7	IN
5	CH8	模拟信号输入通道8	IN	24	CH9	模拟信号输入通道9	IN
6	CH10	模拟信号输入通道10	IN	25	CH11	模拟信号输入通道11	IN
7	CH12	模拟信号输入通道12	IN	26	CH13	模拟信号输入通道13	IN
8	CH14	模拟信号输入通道14	IN	27	CH15	模拟信号输入通道15	IN
9	CH16	模拟信号输入通道16	IN	28	CH17	模拟信号输入通道17	IN
10	CH18	模拟信号输入通道18	IN	29	CH19	模拟信号输入通道19	IN
11	CH20	模拟信号输入通道20	IN	30	CH21	模拟信号输入通道21	IN
12	CH22	模拟信号输入通道22	IN	31	CH23	模拟信号输入通道23	IN
13	CH24	模拟信号输入通道24	IN	32	CH25	模拟信号输入通道25	IN
14	CH26	模拟信号输入通道26	IN	33	CH27	模拟信号输入通道27	IN
15	CH28	模拟信号输入通道28	IN	34	CH29	模拟信号输入通道29	IN



16	CH30	模拟信号输入通道30	IN	35	CH31	模拟信号输入通道31	IN
17	AGND	模拟地	IN	36	AGND	模拟地	IN
18	EX_CLK	外部时钟信号输入	IN	37	TR	外触发信号输入	IN
19	GND	数字地	IN				

第二版本表格说明:

管脚号	名称	管脚功能	特性	管脚号	名称	管脚功能	特性
1	CH0	模拟信号输入通道0	IN	2	CH1	模拟信号输入通道1	IN
3	CH2	模拟信号输入通道2	IN	4	CH3	模拟信号输入通道3	IN
5	CH4	模拟信号输入通道4	IN	6	CH5	模拟信号输入通道5	IN
7	CH6	模拟信号输入通道6	IN	8	CH7	模拟信号输入通道7	IN
9	CH8	模拟信号输入通道8	IN	10	CH9	模拟信号输入通道9	IN
11	CH10	模拟信号输入通道10	IN	12	CH11	模拟信号输入通道11	IN
13	CH12	模拟信号输入通道12	IN	14	CH13	模拟信号输入通道13	IN
15	CH14	模拟信号输入通道14	IN	16	CH15	模拟信号输入通道15	IN
17	CH16	模拟信号输入通道16	IN	18	CH17	模拟信号输入通道17	IN
19	CH18	模拟信号输入通道18	IN	20	CH19	模拟信号输入通道19	IN
21	CH20	模拟信号输入通道20	IN	22	CH21	模拟信号输入通道21	IN
23	CH22	模拟信号输入通道22	IN	24	CH23	模拟信号输入通道23	IN
25	CH24	模拟信号输入通道24	IN	26	CH25	模拟信号输入通道25	IN
27	CH26	模拟信号输入通道26	IN	28	CH27	模拟信号输入通道27	IN
29	CH28	模拟信号输入通道28	IN	30	CH29	模拟信号输入通道29	IN
31	CH30	模拟信号输入通道30	IN	32	CH31	模拟信号输入通道31	IN
33	AGND	模拟地	IN	34	AGND	模拟地	IN
35	EX_CLK	外部时钟信号输入	IN	36	TR	外触发信号输入	IN
37	GND	数字地	IN	38		空	
39		空		40		空	

说明:

CH00~CH31: A/D卡输入通道号(单端方式时)

CH00~CH15: 双端模拟信号输入正端(双端方式时)

CH16~CH31: 双端模拟信号输入负端(双端方式时)

双端方式各通道的对应关系如下表:

原始通道对	对应的采样通道号	原始通道对	对应的采样通道号
CH0,CH16	CH0	CH8,CH24	CH8
CH1,CH17	CH1	CH9,CH25	CH9
CH2,CH18	CH2	CH10,CH26	CH10
CH3,CH19	CH3	CH11,CH27	CH11
CH4,CH20	CH4	CH12,CH28	CH12
CH5,CH21	CH5	CH13,CH29	CH13
CH6,CH22	CH6	CH14,CH30	CH14
CH7,CH23	CH7	CH15,CH31	CH15

EX_CLK : 外部时钟信号输入

TR : 外部触发信号输入

AGND: 模拟地

GND: 数字地

第三节、短路套设置及数据格式

一、板基地址选择

ART2002 的板基地址通过拨码开关 SW1 的设置选择，板基地址可设置成 200H~3F0H 之间可被 16 整除的二进制码，ART2002 将占用基地址起的连续 16 个 I/O 地址，开关的 1、2、3、4、5、6 位分别对应地址 A4、A5、A6、A7、A8、A9。板基地址选择开关 SW1 如下图。

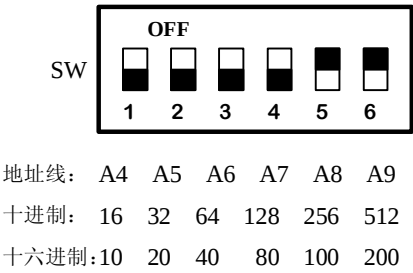


图4.3.1.1 ART2002板基地址选择

板基地址范围512~1008（200H~3F0H），开关置“OFF”为高有效，开关置“ON”为低有效。板基地址等于所有高有效位数之和。如上图是768（300H）。

常用的基地址选择有：

地址	板基地址拨码开关图示	地址	板基地址拨码开关图示
200H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6	210H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6
220H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6	230H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6
240H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6	250H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6
260H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6	270H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6
280H (默认)	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6	290H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6



2A0H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6	2B0H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6
2C0H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6	2D0H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6
2E0H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6	2F0H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6
300H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6	310H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6
320H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6	330H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6
340H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6	350H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6
360H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6	370H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6
380H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6	390H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6
3A0H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6	3B0H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6
3C0H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6	3D0H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6
3E0H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6	3F0H	A4 A5 A6 A7 A8 A9 ON 1 2 3 4 5 6

二、板内地址分配

地址分配表（见下）：

地址	写	读
基地址+0	每通道采样周期设定	可回读所写数据
基地址+2	每通道采样周期辅助设定	可回读所写数据
基地址+4	分组采集组间隔周期设定	可回读所写数据
基地址+6	分组采集组间隔周期辅助设定	可回读所写数据
基地址+8	控制状态寄存器 1	可回读所写数据
基地址+10	控制状态寄存器 2	可回读所写数据
基地址+12	控制状态寄存器 3	获取 FIFO 状态
基地址+14	清 FIFO	FIFO（AD 采样值）

地址分配表说明：

◆ 每通道采样周期设定寄存器(板基地址+0，可读写，16 位)

每通道采样周期设定寄存器通过设置板上时钟的分频系数来设置采样周期，根据所需的采样周期、板上的时钟频率和下面的公式，可以计算出该寄存器的设定值。

$$\text{寄存器设定值} = \frac{\text{时钟频率}}{\text{采样触发频率}} - 1$$

◆ 每通道采样周期设定辅助寄存器(板基地址+2，可读写，16 位)

每通道采样周期辅助设定寄存器可以设置 AD 启动信号的脉宽，这个寄存器的设定值一般建议根据下面的原则设置。首先不能大于采样周期设定寄存器，其次一般是将脉宽设置为大约 0.5us，比如板上时钟是 2MHz，那么该寄存器设置为 1。公式如下。

$$\text{寄存器设定值} = \frac{\text{脉冲宽度}}{\text{时钟周期}}$$

◆ 分组采集组间隔周期设定寄存器(板基地址+4，可读写，16 位)

分组采集组间隔周期设定寄存器通过设置板上时钟的分频系数来设置分组采集组间隔，公式如下。

$$\text{寄存器设定值} = \frac{\text{间隔宽度}}{\text{时钟周期}} - 2$$

◆ 分组采集组间隔周期辅助设定寄存器(板基地址+6，可读写，16 位)

该寄存器现在是对用户透明的，但是必须设置为 1。

◆ AD 通道控制与 FIFO 状态寄存器(板基地址+8，可读写，16 位)

AD 通道控制与 FIFO 状态控制寄存器 1 各位定义如下：

D15	D14	D13	D12	D11-D10	D9-D5	D4-D0
GroudMode	ClockSource	ADMode	TriggerSource	Gains	LastChannel	FirstChannel

FirstChannel: A/D 首通道设置，可读写

LastChannel: A/D 末通道设置，可读写，末通道号应大于等于首通道号。当末通道号等于首通道号时，数据采集将固定在首通道号上；当末通道号大于首通道号时，数据采集顺序将由首通道号开始，每启动一次 A/D 转换，就自动切换到下一通道，直到末通道号，然后再回到首通道号重复进行。

Gains: AD 采样硬件增益。其取值如下表(见下页)：

BD11	BD10	增益 (PGA203)	增益 (PGA202)
------	------	-------------	-------------



0	0	1 倍	1 倍
1	0	2 倍	10 倍
0	1	4 倍	100 倍
1	1	8 倍	1000 倍

TriggerSource: AD 启动源（即触发方式），若等于 0 为内触发，若等于 1 则为外触发。

ADMode: Ad 采集模式，若等于 0 为连续采集（也称为分频采集），若等于 1 则为分组采集（也称为伪同步采集）。

ClockSource: Ad 时钟源，若等于 0 则选择内部时钟，若等于 1 则为外部时钟。

GroundMode: AD 接地方式，若等于 0 则为单端接地方式，若等于 1 则为双端接地方式。

◆中断与 DMA 控制寄存器（板基地址+A，可读写，16 位）

中断与 DMA 控制寄存器的各位定义如下：

D15	D14-D12	D11-D8	D7	D6-D4	D3-D0
EnableInt	IntSignal	IRQNumber	EnableDMA	DMASignal	DMAChannel

DMAChannel: 选择 DMA 通道，同样注意这仅仅是软件的设置，具体哪几个 DMA 通道被硬件支持，见第一章 DMA 部分。如 0101 选择 DMA5（0000 到 1111 分别对应 DMA0 到 DMA15）。

DMASignal: 选择 DMAQ 请求源，与中断一样 100 选择全满，001 选择非空，010 选择半满。

EnableDMA: 为 0 禁止 DMA 传输功能，为 1 使能 DMA 传输功能。

IRQNumber: 选择中断优先级别，根据选择的 PC 机中断级别设置，但是注意这仅仅是软件的设置，具体哪几个中断级别被硬件支持，见第二章技术与性能指标。如 0101 选择中断 5（0000 到 1111 分别对应 INTO 到 INT15）。

IntSignal: 选择中断请求源，100 选择全满中断，001 选择非空中断，010 选择半满中断。

EnableInt: 为 0 禁止中断功能，为 1 使能中断功能。

◆A/D 控制寄存器(板基地址+C，可读写，16 位)

AD 控制寄存器 7 的各位定义如下：

	D15	D14	D13	D12-D2	D1	D0
写	未定义	未定义	未定义	未定义	EnableGlobal	EnableLocal
读	FIFO_FF	FIFO_HF	FIFO_EF	未定义	未定义	未定义

FIFO_EF: FIFO 非空状态查询位，只读，0 表示 FIFO 空，1 表示至少有一个数据。

FIFO_HF: FIFO 半满状态查询位，只读，0 表示 FIFO 半满，1 表示非半满。

FIFO_FF: FIFO 全满状态查询位，只读，0 表示 FIFO 全满，1 表示非全满。

EnableLocal: 0 内触发方式下该位的上升沿将正式启动采集，

1 内触发方式下该位的下升沿将正式启动采集。

EnableGlobal: 0 全局控制位，设置其他寄存器时设为 0，为 0 时 AD 将不会被启动，

1 全局控制位，设置其他寄存器完毕时设为 1。

◆A/D 转换结果寄存器（板基地址+E，可读写，16 位）

ART2002 板对板基地址+E 进行 16 位写操作，将清空 FIFO 内的数据，ART2002 板对板基地址+E 进行 16 位读操作，将读入 FIFO 内的数据（也就是 A/D 转换结果数据），其各位定义如下：

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
AD15	AD14	AD13	AD12	AD11	AD10	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0

其中：AD0~AD15：为 A/D 转换结果的 16 位数据。

ART2002 板为双极性模拟输入（ART2002）时的结果数据格式如下表所示：

输入	A/D结果编码(二进制)	A/D结果编码(十六进制)	A/D结果编码(十进制)
正满度	1111 1111 1111 1111	FFFF	65535
正满度-1LSB	1111 1111 1111 1110	FFFE	65534
中间值+1LSB	1110 0000 0000 0001	8001	32769
中间值(零点)	1000 0000 0000 0000	8000	32768
中间值-1LSB	0111 1111 1111 1111	7FFF	32767
负满度+1LSB	0000 0000 0000 0001	0001	1
负满度	0000 0000 0000 0000	0000	0

ART2002板为单极性模拟输入（ART2002A）时的结果数据格式如下表所示：

输入	A/D结果编码(二进制)	A/D结果编码(十六进制)	A/D结果编码(十进制)
正满度	1111 1111 1111 1111	FFFF	65535
正满度-1LSB	1111 1111 1111 1110	FFFE	65534
零点+1LSB	0000 0000 0000 0001	1	1
零点	0000 0000 0000 0000	0	0

三、模拟信号输入方式选择

本卡模拟信号输入的单、双端选择是由软硬件共同完成的。

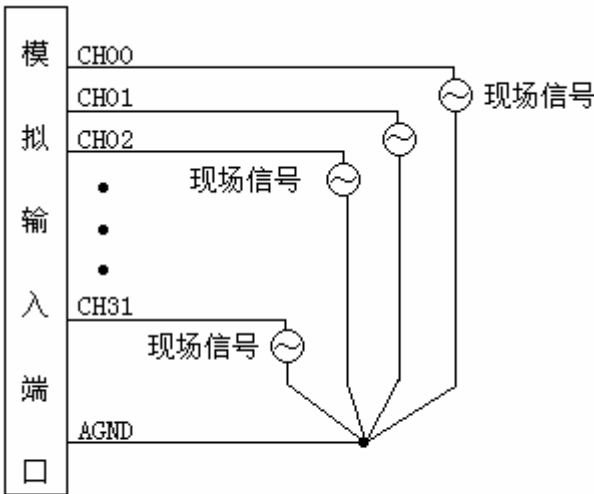
软件设置请看[A/D控制寄存器](#)部分。

单、双端选择	XF1	XF2
单端		
双端		

四、模拟输入信号的连接方式

(1)、单端输入方式

ART2002板均可按图4.1连接成模拟电压单端输入方式，16路模拟输入信号连接到CH00～CH31端，其公共地连接到AGND端。

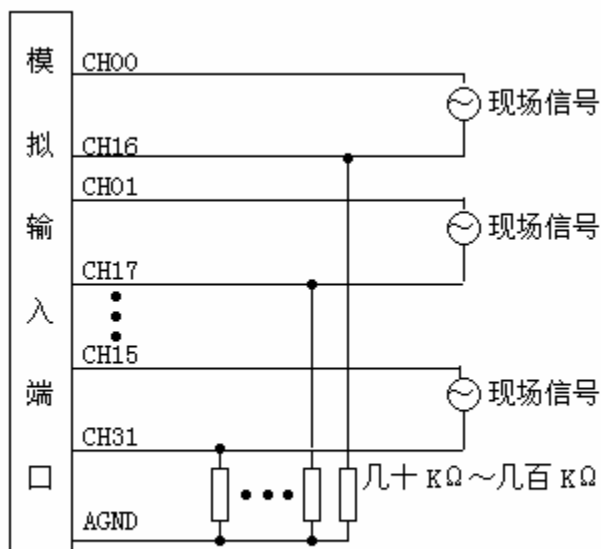


上图是单端输入方式，主要应用在噪声干扰不高的场合

(2)、双端输入方式



ART2002板可按图3.2连接成模拟电压双端输入方式，可以有效抑制共模干扰信号，提高采集精度。8路模拟输入信号正端接到CH00~CH15端，其模拟输入信号负端接到CH16~CH31端，并在距离XS1插座近处，在负端与AGND端各接一只几十K Ω 至几百K Ω 的电阻（当现场信号源内阻小于100 Ω 时，该电阻应为现场信号源内阻的1000倍；当现场信号源内阻大于100 Ω 时，该电阻应为现场信号源内阻的2000倍），为仪表放大器输入电路提供偏置。



上图是双端输入方式，主要应用在噪声干扰高的场合

第三章 工作原理

ART2002 AD 卡自带时钟和定时器, 在设定首、末通道号后, 模拟量从 37 芯 D 型接口输入后, 经过 8 选一开关选择通道进入放大器, 由放大器输出到 A/D 转换器, 开始数据转换, AD 转换结果数据写入先进先出存储器 FIFO, 最后经 ART2002 卡的 PC104 总线将 AD 数据送入计算机。

在 ART2002 板上用户通过软件编程后由硬件控制来启动 A/D 转换, 当 A/D 转换完成后, 可通过软件查询 A/D 完成位然后读取数据; 或者通过 DMA 和中断方式操作。

ART2002 具有二种采样模式: **伪同步采集模式**、**分频采集模式**

ART2002 板 A/D 转换电路的触发方式分为内部触发和外部信号触发两种方式。内部触发是在通过软件设置后就立刻采集, 而外触发是指当外部引脚出现上升沿时开始采集。

在 ART2002 板 A/D 转换的结果是经过硬件控制后直接写入到 FIFO 中, 因此主机是通过与 FIFO 通信来读取数据, 数据可通过下列方式传送给主机:

1. 软件查询 FIFO 状态位, 然后由主机读入结果数据。
2. 设置 FIFO 状态位启动 DMA 方式进行数据传送。
3. 设置 FIFO 状态位启动中断方式进行数据传送。

第一节、A/D 工作模式

◆ **伪同步采集模式**: 模拟同步采样模式, 也叫分组采集模式, 适合应用在对一组输入通道采样的时差要求尽量小, 但组之间的时间间隔较大的应用场合。当由定时器脉冲或外部时钟有效边沿启动后, 以用户设定的参数开始采样, 从第首通道开始顺序转换到末通道结束, 转换完后进入等待的时间 (时间长短由用户设定), 等待下一个启动信号, 如此循环下去。

◆ **分频器采集模式**: 也叫连续采集模式, 当由定时器脉冲或外部时钟有效边沿启动后, 从首通道开始顺序转换, 到末通道转换完成结束, 然后再重新从首通转换到末通道, 如此循环直到用户结束转换。

因此, 在不同模式下, 启动时钟的最大频率不同。由于 AD 的转换速度是 400KHz, 各种模式下的最大转换速度如下:

◆ **伪同步采集模式**: K 个通道的转换时间 $T = 2.5 * K$ (uS), $K = (\text{末通道} - \text{首通道} + 1)$ 为一次转换的输入通道数量。等效每通道最大采样速度: $400K(\text{Hz}) / (\text{末通道} - \text{首通道} + 1)$ 。

◆ **分频器采集模式**: 每通道最小转换时间: $T_{\min} = 2.5\mu\text{S} / \text{通道}$, 等效每通道最大转换速度 (KHz): $400 / (\text{转换通道个数})$ 。

第二节、A/D 转换启动控制

转换启动可以由软件或外部硬件触发启动整个转换过程, 除非用户终止转换, 否则将一直转换下去。

一旦启动转换, AD 转换将在板上定时器或外部时钟驱动下按用户设置的起始、终止通道逐一通道顺序、循环转换。

第三节、板上转换定时器

ART2002 的 16 位定时器为模拟转换提供精确的定时, 定时器输入基准时钟为 4MHz, 周期为 250 纳秒。定时器为减法计数器, 当由用户设置的初值减到 1 时, 发出启动脉冲并自动将定时器数据重新设置为初值。

16 位数据初值范围: 2-65535, 对应周期 $= 0.25 * N(\mu\text{S})$, N : 设置的 16 位定时器初值。

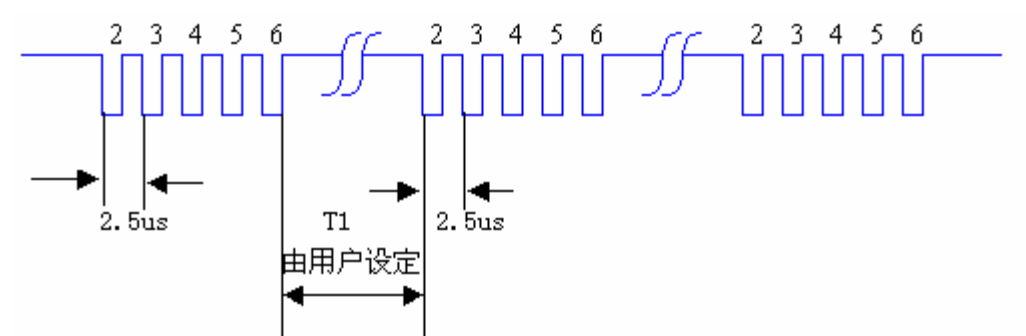
◆ 在分频器采样模式下: 最小转换周期为 $2.5\mu\text{S}$ (初值=10), 由此 ART2002 的转换周期为: $2.5\mu\text{S} \sim 16384\mu\text{S}$ 。

◆ 伪同步步采集, 所谓伪同步采集是指设定的首末通道内以 AD 的最高速率采集 (本卡为 400K, 周期为 $2.5\mu\text{s}$, 由 8253/8254 的 0 通道决定), 而首末通道之间的时间 T_1 可由程序设定, 本板由 8253 的通道的时间常数确定,。



$T1 = \text{时间常数} (2-65535) \times 0.25\mu\text{s}$,例如：对 2、3、4、5、6 通道实现伪同步采集如下图：

启动信号



其中：

2.5us 为触发A/D转换的周期（一般为本卡最高速率，即400KHz）。此周期可以通过改变参数来实现。

T1为每组转换完之后与下一组转换开始之前的时间间隔，这个时间可由软件按需求设定。

第四章 编程举例

第一节、查询方式

```
#include<dos.h>
#include<conio.h>
#define BASEA 0x300

main()
{
    int i,j,FirstChannel,LastChannel;
    unsigned HEX[32];
    float VOL[32];
    char ch;
    clrscr();
    do{
        clrscr();
    t:
        gotoxy(20,10);
        clreol();
        printf("Please input FirstChannel(0--31):");
        scanf("%d",&FirstChannel);
        if (FirstChannel>31||FirstChannel<-1) goto t;
    tt:
        gotoxy(20,10);
        clreol();
        printf("Please input LastChannel(0--31):");
        scanf("%d",&LastChannel);
        if (LastChannel>31) goto tt;
        if (FirstChannel>LastChannel)
        {
            gotoxy(20,11);
            printf("Input data error,please repeat.");
            goto t;
        }
        clrscr();
        gotoxy(1,20);
        printf("*****");
        gotoxy(26,21);
        printf("1.Press Pause-break to stop.");
        gotoxy(26,22);
        printf("2.Press any key to continue.");
        gotoxy(26,23);
        printf("3.Press ESC to exit.");
        while(!kbhit())
```



```

{
    outport(BASEA + 0x0c,0x0000);/*禁止AD操作*/
    outport(BASEA + 0x0a,0x0000);/*禁止一切中断和DMA操作*/
    outport(BASEA + 0x00,0x00ff);/*设置分频器周期*/
    /*设置AD启动脉冲的宽度，根据AD芯片和时钟周期设置，具体见文档*/
    outport(BASEA + 0x02,0x0002);
    outport(BASEA + 0x04,0x0200);/*设置组间隔周期*/
    outport(BASEA + 0x06,0x0001);/*设置组间隔周期，但是现在必须设置为1*/
    /*设置起始通道为0结束通道为0，始0结束1为0x0100*/
    outport(BASEA + 0x08,LastChannel*256+FirstChannel);
    outport(BASEA + 0x0e,0x0000);/*清FIFO*/
    /*以下假设增益为1*/
    /*这两句连续的将启动AD自动采集,内触发、分频、单极性、内时钟*/
    outport(BASEA + 0x0c,0x8000); outport(BASEA + 0x0c,0xc000);
    /* outport(BASEA + 0x0c,0x8002); outport(BASEA + 0x0c,0xc002);
    /* outport(BASEA + 0x0c,0x8004); outport(BASEA + 0x0c,0xc004);*/这两句连续的将启动AD自动采
集,内触发、分频、单极性、外时钟*/
    /* outport(BASEA + 0x0c,0x8006); outport(BASEA + 0x0c,0xc006);*/这两句连续的将启动AD自动采
集,内触发、分组、单极性、外时钟*/
    /* outport(BASEA + 0x0c,0x8008); outport(BASEA + 0x0c,0xc008);*/这两句连续的将启动AD自动采
集,内触发、分频、双极性、内时钟*/
    /* outport(BASEA + 0x0c,0x800A); outport(BASEA + 0x0c,0xc00A);*/这两句连续的将启动AD自动采
集,内触发、分组、双极性、内时钟*/
    /* outport(BASEA + 0x0c,0x800C); outport(BASEA + 0x0c,0xc00C);*/这两句连续的将启动AD自动采
集,内触发、分频、双极性、外时钟*/
    /* outport(BASEA + 0x0c,0x800E); outport(BASEA + 0x0c,0xc00E);*/这两句连续的将启动AD自动采
集,内触发、分组、双极性、外时钟*/
    /* outport(BASEA + 0x0c,0x0001); outport(BASEA + 0x0c,0x8001);*/这两句连续的将启动AD自动采
集,外触发、分频、单极性、内时钟*/
    /* outport(BASEA + 0x0c,0x0003); outport(BASEA + 0x0c,0x8003);*/这两句连续的将启动AD自动采
集,外触发、分组、单极性、内时钟*/
    /* outport(BASEA + 0x0c,0x0005); outport(BASEA + 0x0c,0x8005);*/这两句连续的将启动AD自动采
集,外触发、分频、单极性、外时钟*/
    /* outport(BASEA + 0x0c,0x0007); outport(BASEA + 0x0c,0x8007);*/这两句连续的将启动AD自动采
集,外触发、分组、单极性、外时钟*/
    /* outport(BASEA + 0x0c,0x0009); outport(BASEA + 0x0c,0x8009);*/这两句连续的将启动AD自动采
集,外触发、分频、双极性、内时钟*/
    /* outport(BASEA + 0x0c,0x000B); outport(BASEA + 0x0c,0x800B);*/这两句连续的将启动AD自动采
集,外触发、分组、双极性、内时钟*/
    /* outport(BASEA + 0x0c,0x000D); outport(BASEA + 0x0c,0x800D);*/这两句连续的将启动AD自动采
集,外触发、分频、双极性、外时钟*/
    /* outport(BASEA + 0x0c,0x000F); outport(BASEA + 0x0c,0x800F);*/这两句连续的将启动AD自动采
集,外触发、分组、双极性、外时钟*/
do{
    ;

```

```

    }while(!(inport(BASEA + 0x08)&0x2000));/*读取FIFO状态位*/

    j=0;
    for(i=FirstChannel;i<LastChannel+1;i++)
    {
        HEX[i]=(inport(BASEA+0x0e)^0x8000);
        VOL[i]=20.0*HEX[i]/65536-10.0;
        if(i<16)
        {
            gotoxy(8,4+i);
            printf("VOL[%2.2d]=%+2.5fV  HEX[%2.2d]=%4.4x",i,VOL[i],i,HEX[i]);
        }
        if(i>15)
        {
            gotoxy(42,4+j);
            printf("VOL[%2.2d]=%+2.5fV  HEX[%2.2d]=%4.4x",i,VOL[i],i,HEX[i]);
            j++;
        }
    }
}
ch=getch();
if (ch==27) break;
}while(1);
clrscr();
}

```

第二节、中断方式

```

#include<dos.h>
#include<conio.h>
#include<stdio.h>
#define BASEA 0x300
int j,dly=0,n, i,FirstChannel,LastChannel;
unsigned HEX[32];
float VOL[32];

void interrupt far intsev(void)  /*中断服务程序*/
{
    for(n=FirstChannel;n<LastChannel+1;n++)
    {
        HEX[n]=inport(BASEA+0x0e)^0x8000;
    }
    dly=1;
    outport(BASEA + 0x0c,0x0000);/*禁止AD操作*/
}

```



```
outport(BASEA + 0x0a,0x0000);/*禁止一切中断和DMA操作*/
```

```
    outp(0x20,0x20); /*关中断*/  
/* outp(0xa0,0x20); /*关中断*/  
/* outp(0xa1,0x20); /*关中断*/  
}
```

```
void main(void)
```

```
{  
    char Exit;  
    clrscr();  
    do{  
        clrscr();  
t:  
        gotoxy(20,10);  
        clreol();  
        printf("Please input FirstChannel(0--31):");  
        scanf("%d",&FirstChannel);  
        if (FirstChannel>31||FirstChannel<=-1) goto t;  
tt:  
        gotoxy(20,10);  
        clreol();  
        printf("Please input LastChannel(0--31):");  
        scanf("%d",&LastChannel);  
        if (LastChannel>31) goto tt;  
        if (FirstChannel>LastChannel)  
        {  
            gotoxy(20,11);  
            printf("Input data error,please repeat.");  
            goto t;  
        }  
        clrscr();  
        gotoxy(1,20);  
        printf("*****");  
        gotoxy(26,21);  
        printf("1.Press Pause-break to stop.");  
        gotoxy(26,22);  
        printf("2.Press any key to continue.");  
        gotoxy(26,23);  
        printf("3.Press ESC to exit.");  
        while(!kbhit())  
        {  
            outport(BASEA + 0x0c,0x0000);/*禁止AD操作*/  
            outport(BASEA + 0x0a,0x0000);/*禁止一切中断和DMA操作*/
```

```
    output(BASEA + 0x00,0x00ff);/*设置分频器周期*/
    output(BASEA + 0x02,0x0002);/*设置AD启动脉冲的宽度，根据AD芯片和时钟周期设置，具体见文
档*/
    output(BASEA + 0x04,0x0200);/*设置组间隔周期*/
    output(BASEA + 0x06,0x0001);/*设置组间隔周期，但是现在必须设置为1*/
    output(BASEA + 0x08,LastChannel*256+FirstChannel);/*设置起始通道为0结束通道为0，始0结束1为
0x0100*/
    output(BASEA + 0x0e,0x0000);/*清FIFO*/
    outp(0x21,0x00); /*设置中断7的屏蔽位*/
    /* outp(0xa1,0); /*清中断*/
    setvect(0x0f,intsev); /*安装中断服务程序*/
    /*以下假设增益为1*/
    output(BASEA + 0x0c,0x8000); output(BASEA + 0x0c,0xc000);/*这两句将启动AD自动采集,
内触发、分频、单极性、内时钟*/
    /* output(BASEA + 0x0c,0x8002); output(BASEA + 0x0c,0xc002);/*这两句将启动AD自动采
集,内触发、分组、单极性、内时钟*/
    /* output(BASEA + 0x0c,0x8004); output(BASEA + 0x0c,0xc004);/*这两句将启动AD自动采
集,内触发、分频、单极性、外时钟*/
    /* output(BASEA + 0x0c,0x8006); output(BASEA + 0x0c,0xc006);/*这两句将启动AD自动采
集,内触发、分组、单极性、外时钟*/
    /* output(BASEA + 0x0c,0x8008); output(BASEA + 0x0c,0xc008);/*这两句将启动AD自动采
集,内触发、分频、双极性、内时钟*/
    /* output(BASEA + 0x0c,0x800A); output(BASEA + 0x0c,0xc00A);/*这两句将启动AD自动采
集,内触发、分组、双极性、内时钟*/
    /* output(BASEA + 0x0c,0x800C); output(BASEA + 0x0c,0xc00C);/*这两句将启动AD自动采
集,内触发、分频、双极性、外时钟*/
    /* output(BASEA + 0x0c,0x800E); output(BASEA + 0x0c,0xc00E);/*这两句将启动AD自动采
集,内触发、分组、双极性、外时钟*/
    /* output(BASEA + 0x0c,0x0001); output(BASEA + 0x0c,0x8001);/*这两句将启动AD自动采
集,外触发、分频、单极性、内时钟*/
    /* output(BASEA + 0x0c,0x0003); output(BASEA + 0x0c,0x8003);/*这两句将启动AD自动采
集,外触发、分组、单极性、内时钟*/
    /* output(BASEA + 0x0c,0x0005); output(BASEA + 0x0c,0x8005);/*这两句将启动AD自动采
集,外触发、分频、单极性、外时钟*/
    /* output(BASEA + 0x0c,0x0007); output(BASEA + 0x0c,0x8007);/*这两句将启动AD自动采
集,外触发、分组、单极性、外时钟*/
    /* output(BASEA + 0x0c,0x0009); output(BASEA + 0x0c,0x8009);/*这两句将启动AD自动采
集,外触发、分频、双极性、内时钟*/
    /* output(BASEA + 0x0c,0x000B); output(BASEA + 0x0c,0x800B);/*这两句将启动AD自动采
集,外触发、分组、双极性、内时钟*/
    /* output(BASEA + 0x0c,0x000D); output(BASEA + 0x0c,0x800D);/*这两句将启动AD自动采
集,外触发、分频、双极性、外时钟*/
    /* output(BASEA + 0x0c,0x000F); output(BASEA + 0x0c,0x800F);/*这两句将启动AD自动采
集,外触发、分组、双极性、外时钟*/
    enable(); /*开中断*/
```



```

        output(BASEA + 0x0a,0xa700);/*中断7半满*/
        /* output(BASEA + 0x0a,0x9700);/*中断7非空*/
        /* output(BASEA + 0x0a,0xc700);/*中断7全满*/

        do{
            ;
        }while(!dly); /*等中断*/
        dly=0;
        j=0;
        for(i=FirstChannel;i<(LastChannel+1);i++)
        {
            /*HEX[i]=(inport(BASEA+0x0e)^0x8000);*/
            VOL[i]=20.0*HEX[i]/65536-10.0;
            if(i<16)
            {
                gotoxy(8,4+i);
                printf("VOL[%2.2d]=%+2.5fV  HEX[%2.2d]=%4.4x",i,VOL[i],HEX[i]);
            }
            if(i>=16)
            {
                gotoxy(42,4+j);
                printf("VOL[%2.2d]=%+2.5fV  HEX[%2.2d]=%4.4x",i,VOL[i],HEX[i]);
                j++;
            }
        }
    }
    Exit=getch();
    if (Exit==27) break;
}while(1);
clrscr();
}

```

第三节、DMA 方式

```

#include<dos.h>
#include<conio.h>
#define BASEA 0x300

main()
{
    int k,j,FirstChannel,LastChannel;
    unsigned i,HEX[32];
    float VOL[32];
    char ch;

```

```

clrscr();
do{
    clrscr();
t:
    gotoxy(20,10);
    clreol();
    printf("Please input FirstChannel(0--31):");
    scanf("%d",&FirstChannel);
    if (FirstChannel>31||FirstChannel<-1) goto t;
tt:
    gotoxy(20,10);
    clreol();
    printf("Please input LastChannel(0--31):");
    scanf("%d",&LastChannel);
    if (LastChannel>31) goto tt;
    if (FirstChannel>LastChannel)
    {
        gotoxy(20,11);
        printf("Input data error,please repeat.");
        goto t;
    }
    clrscr();
    gotoxy(1,20);
    printf("*****");
    gotoxy(26,21);
    printf("1.Press Pause-break to stop.");
    gotoxy(26,22);
    printf("2.Press any key to continue.");
    gotoxy(26,23);
    printf("3.Press ESC to exit.");
        while(!kbhit())
        {
outport(BASEA + 0x0c,0x0000);/*禁止AD操作*/
outport(BASEA + 0x0a,0x0000);/*禁止一切中断和 DMA 操作*/
outport(BASEA + 0x00,0xffff);/*设置分频器周期*/
/*设置 AD 启动脉冲的宽度, 根据 AD 芯片和时钟周期设置, 具体见文档*/
outport(BASEA + 0x02,0x0002);
outport(BASEA + 0x04,0x0200);/*设置组间隔周期*/
outport(BASEA + 0x06,0x0001);/*设置组间隔周期, 但是现在必须设置为 1*/
outport(BASEA + 0x08,0x1f00);/*设置起始通道为 0 结束通道为 0 增益为 1*/
outport(BASEA + 0x0e,0x0000);/*清 FIFO*/
/* outport(BASEA + 0x0a,0x00a6);/*DMA6半满*/
outport(BASEA + 0x0a,0x0096);/*DMA6非空*/
/* outport(BASEA + 0x0a,0x00c6);/*DMA6全满*/
/*for(i=0;i<300;i=i+2)

```



```

{
poke(0x4000,i,0);
}*/
outp(0x08,0x04);
outp(0xd0,0x04);
outp(0xd8,0x00);
outp(0xd6,0x46);
outp(0xd6,0xc4);

outp(0xd2,0x02);
outp(0x89,0x04);
outp(0xc8,0x00);
outp(0xc8,0x00);
outp(0xca,0x01);
outp(0xca,0x01);
outp(0xd4,0x02);
outp(0x08,0x00);
outp(0xd0,0x00);
outport(BASEA + 0x0c,0x8000);
outport(BASEA + 0x0c,0xc000);/*这连续的两句将启动AD自动采集*/
j=0;
for(i=FirstChannel*2;i<LastChannel*2+1;i=i+2)
{
    HEX[i]=peek(0x4000,i)^0x8000;
    VOL[i]=20.0*HEX[i]/65536-10.0;
    if(i/2<16)
    {
        gotoxy(8,4+i/2);
        printf("VOL[%2.2d]=%+2.5fV          HEX[%2.2d]=%4.4
x",i/2,VOL[i],i/2,HEX[i]);

    }
    if(i/2>15)
    {
        gotoxy(42,4+j);
        printf("VOL[%2.2d]=%+2.5fV
HEX[%2.2d]=%4.4x",i/2,VOL[i],i/2,HEX[i]);

        j++;
    }
}
}
ch=getch();
if (ch==27) break;
}while(1);
clrscr();
}

```

第五章 产品的应用注意事项、校准、保修

第一节、使用注意事项

在公司售出的产品包装中,用户将会找到这本说明书和ART2002板,同时还有产品质保卡。产品质保卡请用户务必妥善保存,当该产品出现问题需要维修时,请用户将产品质保卡同产品一起,寄回本公司,以便我们能尽快的帮用户解决问题。

在使用ART2002板时,应注意以下问题:

- ◆ ART2002板正面的IC芯片不要用手去摸,防止芯片受到静电的危害。
- ◆ 用户务必注意电源的开关顺序,使用时要求先开主机电源,后开信号源的电源;先关信号源的电源,后关主机电源。

第二节、A/D 的校准

ART2002板出厂时已经校准,只有当用户使用一段时间后,或者用户认为需要时才做校准。下面以 $\pm 5V$ 量程为例,说明校准过程:

准备一块5位半精度以上数字电压表,安装好ART2002,打开主机电源,预热15分钟。

◆ 设定放大器增益为1性况下:选模拟输入的任两个通道,比如CH0、CH1通道,将CH0通道输入接0伏电压,CH1接正满度电压4999.39毫伏,运行ART2002测试程序,调整电位器RP1,使CH0电压为0.000MV,调整电位器RP2,使CH1电压为4999.89MV。重复以上步骤,直到满足要求为止。

◆ 如果放大器增益大于1,在程控增益控制字内键入相应值,CH1通道也要接入相应的满度电压,校准方法同上。

第三节、保修

ART2002自出厂之日起,两年内,凡用户遵守运输,贮存和使用规则,而质量低于产品标准者公司免费修理。