UA307A 型 A/D 采集器使用说明(下载版)

第一章. 概述

UA307A 型 A/ D 采集器是我公司开发的 USB 总线数据采集产品, 可与带 USB 接口的各种台式计算机, 笔记本机, 工控机连接构成高性能的数据采集测量系统。该产品采用美国新型 16位高速 A/D 转换芯片, 设计讲究, 测量精度高, 速度快, 编程简便, 且具有 USB 设备体积小巧, 连接方便, 无需外接电源, 即插即用, 可带电拔插等特有优点。可广泛应用于科学实验, 工业测量控制领域。

UA307A 型 A/ D 采集器是我公司 UA307 型 A/ D 采集器的改型产品，它在 UA307 型的基础上增加了两项新功能：1. 准同步采样功能。 2. 新的外同步触发功能。

准同步采样：在多通道采集时, 许多应用场合希望各通道的数据是同时刻的, 通道间无相位差或相位差尽量小, 这就是所谓的同时刻采集或同步采集。一般的 A/D 板多采用一个 A/D 转换器+多路开关的方式进行多路采集且采样间隔是固定的, 这就造成了通道间的相位差且采样频率越低相位差越大。 解决这一问题的方法有三种: 1. 采用多 A/D 并行采集 2. 加装多路同时刻采样保持器 3. 由硬件设计实现与采样频率无关的通道间最快速度扫描采集。 以上 1,2 种方法可实现理想的无相差采集, 但都有电路复杂, 成本高, 功耗大的缺点. 第三种方法即准同步采集, 该方法不能实现完全的无相差但可以实现相差尽量小, 仍可满足一般使用要求。这种方法的原理是采用变采样间隔的方法, 通道间采用 A/D 允许的最快速度采集, 而每通道样点的采样间隔(频率)可任意设定。如采用 2uS 的 A/D 转换器 4 通道采集, 通道间固定相差为 2.5uS, 1 到 4 通道最大相差 7.5uS, 每通道的采样频率可以是任意的(如用 1KHz 采样频率, 样点间间隔为 1000uS). 这种方式的优点是: 电路简单, 成本低, 采样通道数任意, 功耗小。其中功耗小是非常重要的, 它意味着 USB 采集器不需外加电源, 而以上 1,2 种方法一般都需要外加电源。

UA307A 型采集器即是用以上第三种方法设计的准同步采集器, 它的准同步采样功能完全由硬件实现, 编程使用非常简单, 它与 UA307 向下兼容, 也可以普通方式进行多通道采集。

新外同步触发功能：UA307A 可以实现外脉冲触发采样，且同样具有准同步采样特性, 每个外脉冲各通道快速采样一次。

一.主要功能及特点

- 分辩率 ：16bit
- 16 或 32 模入通道
- 实用最高采样频率 ：500KHz(连续数据量 200KB); 400KHz(连续采集存盘)
- 选装程控放大器
- 任意设定采样通道数, 多通道自动扫描采集，可实现多通道准同步采集
- 64KB先进先出(FIFO)缓冲存储器, 可实现自动数据块采集
- 软件或定时器触发采样, 可任意设定采样频率
- 可加装数字量 I/O

· 可连续大数据量采集
· 带 DC/DC 隔离电源，精度稳定
· 小盒式，小机箱式，或卡式供应，方便使用
· 丰富的软件支持

二.技术指标

1.A/D 部分
· 分辩率:　16bit
· 精度:　优于 0.015%(满量程)
· 最高实用采样频率: 500KHz(连续数据量 200KB); 400KHz(连续采集存盘)
· 模入通道: 16 单端（选装 32 单端）
· 模入范围: ±10V（选装±5V，0-10V, 0-5V）
· 程控增益：选装 1、2、4、8、16 倍
· 输入阻抗: >100MΩ
· 触发方式: 定时器触发，软件触发
· FIFO 存储器: 64KB
2.数字量 I/O
· 数字量 I/O: 选装 8DI/8DO
· 可编程输入或输出
· TTL 电平兼容
3.定时计数器
· 3 通道可编程定时计数起器(825４)
· 字长: 16bit
· 卡上时钟: 10MHz

第二章.硬件连接说明

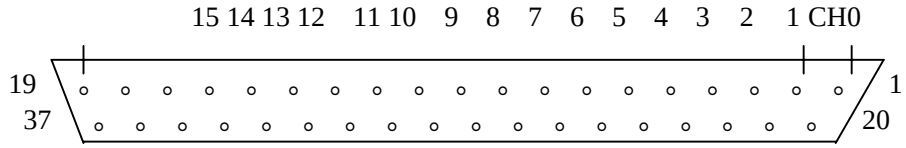
一．A/D 卡的连接

UA306HA 盒式采集器一端装有一只 37 芯 D 型插座，另一端装有方形 USB 插座. 使用时, 37 芯 D 型插座接信号; USB 插座接 USB 电缆，电缆另一端接主机 USB 插口.

37 芯 D 型插座，定义如下图:

其中: CH x 为模拟输入通道

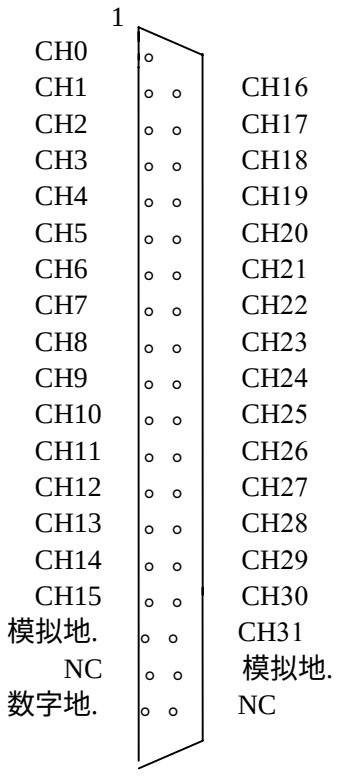
1.16通道模拟输入型信号插座定义（孔型）:



20~35 信号地

1-16:模拟信号输入通道 0-15
20-35：模拟地

2.32通道模拟输入型信号插座定义:



第三章. 安装驱动软件及应用软件

一．安装驱动程序．UA307A 在 WINDOWS 98/2000/ME/XP 下工作, 使用时需要安装设备驱动程序. 一台计算机在第一次连接 UA307A 时需要人工安装设备驱动程序, 以后再连接 UA307A 时设备驱动程序会自动安装. 下面介绍人工安装设备驱动程序的方法.

步骤一:
当计算机在开机状态, WINDOWS 正常运行时, 将 USB 电缆方形端插入 UA307A 采集器(电缆另一端应预先插在计算机任意 USB 插口上)。此时屏幕出现“ 找到新的硬件设备 ”，接着出现如下提示窗口：

(图略)

步骤二：点击下一步，出现如下窗口：

(图略)

步骤三：选择“ 搜索设备的最新驱动程序 ”，点击下一步，出现如下窗口：

步骤四：将 UA307A 驱动光盘插入驱动器，选择“ 搜索光盘驱动器 ”，点击下一步，出现如下窗口：

(图略)

步骤五：点击下一步，出现如下窗口：

(图略)

步骤六：点击完成，驱动程序安装成功了！

二．安装应用软件．UA307A 光盘上还提供了其它软件工具，如：动态连接库，演示应用软件等，应该把它们拷入硬盘以便使用．安装方法是：将盘中文件夹直接拷入硬盘任意文件夹即可．

三．**运行程序**．安装软件后，您就可以运行程序，测试一下采集器了．运行文件夹中 UA306HA.exe 文件即可进入双通道采集示波界面，此时在模拟通道０或１接入信号源或用手触摸输入端便可见到

采集波形显示.

点击“ Fr. ”按钮可改变采集频率.

点击“ CH+ ”按钮可改变通道.

点击“ 数据. ”按钮可显示电压值.

点击“ G+. ”按钮可改变放大倍数.

点击画面可冻结波形，再次点击画面可回复动态采集.

其余可按照界面提示操作.

第四章. UA307A 的编程

UA307A 可使用各种 WINDOWS 编程工具编程, 如: VC++, VB, Delphi, BC++... 等. 为方便使用我们随板提供专用的动态连接库 UA307A. DLL. 在此动态连接库中提供了许多简洁高效的采集和控制函数, 支持 UA307A 采集器的各种功能, 用户可简单方便地调用这些函数完成各种数据采集工作. 本章对这些函数进行详细说明, 并给出调用例子.

一．函数一览:

OpenUA300	打开 UA307A 设备
CloseUA300	关闭 UA307A 设备
GetVer	得到版本号
ssinit	单通道单点采集初始化
ssad	单通道单点采集
minit	单或多通道多点采集初始化(第一种方式)
readdata	单或多通道多点采集(第一种方式)
minitz	多通道多点准同步采集初始化(第一种方式)
readdataz	多通道多点准同步采集(第一种方式)
minit2	单或多通道多点采集初始化(第二种方式)
readdata2	单或多通道多点采集(第二种方式)
endread2	多点采集结束(第二种方式)
startad	可替代 minit2 功能，更适合于连续大数据量采集(第二种方式)
minit2z	多通道多点准同步采集初始化(第二种方式)
startadz	可替代 minit2z 功能，更适合于连续大数据量采集(第二种方式)
dout	数字量输出
din	数字量输入
outb	扩展口输出
inb	扩展口输入

说明: 以上 mint, readdata 为第一组多点采集函数, 适用于一次性读取数据.

Minit2, readdata2, endread2 为第二组多点采集函数，适用于边采集边处理的应用，在执行 minit2 初始化后，可以用 readdata2 函数多次读取数据，并可在读取间隔进行数据处理，只要处理时间不是太长，间隔读取的数据将是连续的. 此种方式采集结束时需用 endread2 函数终止采集.

Minitz, readdataz 为准同步采集时使用的函数，功能与 minit, readdata 相同。

Minit2z 为准同步采集时使用的函数，功能与 minit2 相同。

二．函数的详细说明

1．OpenUA300

函数形式:

HANDLE _stdcall OpenUA300()

说　　明: 该函数用于打开 UA307A 设备，取得设备句柄，对 UA307A 操作时，应首先执行该函数．

输入参数: 无

返 回 值: 返回设备句柄，这个句柄在调用其他函数时需要使用．

２．CloseUA300

函数形式:

HANDLE _stdcall CloseUA300(HANDLE husb)

说　　明: 该函数用于关闭 UA307A，在结束对 UA307A 操作时，应执行该函数关闭设备．

输入参数: 在打开设备时得到的设备句柄．

返 回 值: 可忽略．

3．ssinit

函数形式:

void _stdcall ssinit(HANDLE husb)

说　　明: 该函数用于单通道单点数据采集的初始化（任意通道读一点数据），在单通道单点数据采集时应先执行此函数．

输入参数: husb 在打开设备时得到的设备句柄．

返 回 值: 无．

４．ssad

函数形式: short int _stdcall ssad(HANDLE husb,short int adch, short int gain)

说　　明: 该函数用于单通道读一点数据（任意通道读一点数据）．

输入参数: husb 在打开设备时得到的设备句柄．

adch 输入通道号（０－３１）．

gain 放大倍数代码（0＝１倍（不放大），１＝２倍，２＝４倍，３＝８倍，４＝１６倍．．．）

返 回 值: 整数的AD 数据值，－３２７６８－３２７６７ 代表－１０V－＋１０V 电压（放大倍数为１时）．

5. minit

函数形式: void _stdcall minit(HANDLE husb, short fch,short chn, short gain)

说　　明: 该函数用于多点（单通道或多通道）数据采集的初始化，一次连续采集多点数据时应先执行此函数．此函数和 readdata 函数连用.

输入参数: husb 在打开设备时得到的设备句柄．

fch 首输入通道号（０－３１）

chn 通道数（１－３２）．多通道采集时，采集通道应连续，采集通道由首通道号和通道数决定．单通道采集时，通道数应设为１．

gain 放大倍数代码（0＝１倍（不放大），１＝２倍，２＝４倍，３＝８倍，４＝１６倍．．．）

5a．minitz

函数形式: void _stdcall minit(HANDLE husb, short fch,short chn, short gain, unsigned short fn)

说　　明: 该函数用于多通道多点准同步采集的初始化，进行多通道多点准同步采集数据时应先执行此函数．此函数和 readdataz 函数连用.

输入参数: husb 在打开设备时得到的设备句柄．

fch 首输入通道号（０－３１）

chn 通道数（１－３２）．多通道采集时，采集通道应连续，采集通道由首通道号和通道数决定．单通道采集时，通道数应设为１．

gain 放大倍数代码（0＝１倍（不放大），１＝２倍，２＝４倍，３＝８倍，４＝１６倍．．．）

fn 控制通道间快速扫描频率的频率控制码，fn＝1 0０００００００（Hz）/通道间采样频率（Hz）. 实际上它决定通道间延时，对于 UA307A fn 最小为 22

６．readdata

函数形式: void _stdcall readdata(HANDLE husb, short *addat,unsigned short fcode, long leng)

说　　明: 该函数用于多点连续（单通道或多通道）数据采集．此函数与 minit 连用.

输入参数: husb 在打开设备时得到的设备句柄．

fcode 频率码＝1 0０００００００（Hz）/ 采样频率（Hz）

leng 采样数据的总点数(一点为 16 位字), 注意，需是 64 的倍数．该参数是长整数，意味着只要数据数组开的足够大，可以一次采集几十兆字节以上的数据．

输出参数: *addat 数据数组．采集数据存放于数据数组．

6a. readdataz

函数形式: void _stdcall readdata(HANDLE husb, short *addat,unsigned short fcode, long leng)

说　　明: 该函数用于通道多点连续采集．用于读取数据，此函数与 minitz 连用.

输入参数: husb 在打开设备时得到的设备句柄．

fcode 频率码＝1 0０００００００（Hz）/ 每通道采样频率（Hz）. 注意此处是每通道采样频率, 即单通道采样频率. 与 readdata 函数不同，readdata 此处是总采样频率.

leng　采样数据的总点数(一点为 16 位字), 注意，需是 64 的倍数．该参数是长整数，意味着只要数据数组开的足够大，可以一次采集几十兆字节以上的数据．

输出参数:　*addat　数据数组．采集数据存放于数据数组．

7．minit2

函数形式: void _stdcall minit2(HANDLE husb, short fch, short chn,unsigned short fcode, short gain)

说　　明: 该函数用于第二种方式多点（单通道或多通道）数据采集的初始化，连续采集多点数据时应先执行此函数．此函数执行后可用 readdata2 函数读取数据.

输入参数:　husb　在打开设备时得到的设备句柄．

fch　首输入通道号（０－３１）

chn　通道数（１－３２）．多通道采集时，采集通道应连续，采集通道由首通道号和通道数决定．单通道采集时，通道数应设为１．

fcode　频率码＝1 0０００００００（Hz）/　采样频率（Hz）

gain　放大倍数代码（0 = １倍（不放大），１＝２倍，２＝４倍，３＝８倍，４＝１６倍．．．）

7a．minit2z

函数形式: void _stdcall minit2z(HANDLE husb, short fch, short chn, short gain, unsigned short fn,unsigned short fcode)

说　　明: 该函数用于第二种方式多点多通道准同步数据采集的初始化，连续准同步采集多点数据时应先执行此函数．此函数执行后可用 readdata2 函数读取数据.

输入参数:　husb　在打开设备时得到的设备句柄．

fch　首输入通道号（０－３１）

chn　通道数（１－３２）．多通道采集时，采集通道应连续，采集通道由首通道号和通道数决定．单通道采集时，通道数应设为１．

fn　控制通道间快速扫描频率的频率控制码，fn＝1 0０００００００（Hz）/通道间采样频率（Hz）．实际上它决定通道间延时，对于 UA307A　fn 最小为 22

fcode　频率码＝1 0０００００００（Hz）/　每通道采样频率（Hz）．注意此处是每通道采样频率, 即单通道采样频率.

gain　放大倍数代码（0 = １倍（不放大），１＝２倍，２＝４倍，３＝８倍，４＝１６倍．．．）

7b．startad

函数形式: void _stdcall startad(HANDLE husb, short fch, short chn,unsigned short fcode, short gain)

说　　明: 此函数与 minit2 函数功能相同，在连续大数据量采集用此函数代替 minit2 函数有更高的效率. 此函数执行后可用 readdata2 函数读取数据.

输入参数:　husb　在打开设备时得到的设备句柄．

fch　首输入通道号（０－３１）

chn　通道数（１－３２）．多通道采集时，采集通道应连续，采集通道由首通道号和

通道数决定．单通道采集时，通道数应设为１．

fcode　频率码＝1 0０００００００（Hz）/　采样频率（Hz）

gain　放大倍数代码（0 = １倍（不放大），１＝２倍，２＝４倍，３＝８倍，４＝１６倍．．．）

7c．startadz

函数形式: void _stdcall startadz(HANDLE husb, short fch, short chn, short gain, unsigned short fn,unsigned short fcode)

说　　明: 该函数与 minit2z 函数功能相同，在连续大数据量采集用此函数代替 minit2z 函数有更高的效率. 此函数执行后可用 readdata2 函数读取数据.

输入参数:　husb　在打开设备时得到的设备句柄．

fch　首输入通道号（０－３１）

chn　通道数（１－３２）．多通道采集时，采集通道应连续，采集通道由首通道号和通道数决定．单通道采集时，通道数应设为１．

fn　控制通道间快速扫描频率的频率控制码，fn＝1 0０００００００（Hz）/通道间采样频率（Hz）. 实际上它决定通道间延时，对于 UA307A　fn 最小为 22

fcode　频率码＝1 0０００００００（Hz）/　每通道采样频率（Hz）. 注意此处是每通道采样频率, 即单通道采样频率.

gain　放大倍数代码（0 = １倍（不放大），１＝２倍，２＝４倍，３＝８倍，４＝１６倍．．．）

8．readdata2

函数形式: void _stdcall readdata2(HANDLE husb, short *addat, long leng)

说　　明: 该函数用于第二种方式多点连续（单通道或多通道）数据采集．在执行 minit2 或 startad 或 minit2z 或 startadz 后, 可多次调用本函数读取数据, 只要间隔不是太长, 数据将是连续的.

输入参数:　husb　在打开设备时得到的设备句柄．

leng　采样数据的总点数(一点为 16 位字), 注意，需是 64 的倍数．该参数是长整数，意味着只要数据数组开的足够大，可以一次采集几十兆字节以上的数据．

输出参数:　*addat　数据数组．采集数据存放于数据数组．

返 回 值:　无

9. endread2

函数形式: void _stdcall endread2(HANDLE husb)

说　　明: 该函数用于关闭第二种方式多点数据采集, 在第二种方式多点数据采集结束时需执行.

输入参数:　husb　在打开设备时得到的设备句柄．

返 回 值:　无

10．GetVer

函数形式: UCHAR _stdcall GetVer(HANDLE husb)
说　　明: 得到版本函数．可读取 UA307A 的版本号
输入参数:　husb　在打开设备时得到的设备句柄．
返 回 值: 版本号

11．dout
函数形式: void _stdcall dout(HANDLE husb, UCHAR ddat)
说　　明: 在 UA307A 数字量输出口输出 8 位数据
输入参数: husb　在打开设备时得到的设备句柄
　　ddat　输出数据(一个字节)
返 回 值: 无

12．din
函数形式: UCHAR _stdcall din(HANDLE husb)
说　　明: 在 UA306HA 数字量输入口输入 8 位数据
输入参数: husb　在打开设备时得到的设备句柄
返 回 值: 8 位数据

13　outb
函数形式: void _stdcall outb(HANDLE husb, UCHAR addr,UCHAR dat)
说　　明: 在 UA307A 输出扩展口输出一字节
输入参数: husb　在打开设备时得到的设备句柄
　　addr　外部口地址
　　dat　输出的数据字节
返 回 值: 无

14. inb
函数形式: UCHAR _stdcall inb(HANDLE husb, UCHAR addr)
说　　明: 在 UA307A 输入扩展口读一字节
输入参数: husb　在打开设备时得到的设备句柄
　　addr　外部口地址
返 回 值: 8 位数据

以上描述了 UA307a.DLL 中的基本函数，您可以调用这些函数进行各种数据采集工作．以下给出调用例子．

二．函数调用例子(非准同步采样方式)

１．VC++６调用例子

这是一个控制台方式的调用例子．编译时需加上 UA307a.h, UA307a.lib

```
// UA307a.cpp
#include "windows.h"
#include "stdafx.h"
#include "ua307a.h"

HANDLE husb;

int main(int argc, char* argv[])
{
short int ch;
short int i,j;
short int addat[4096];

printf("Hello World!\n");

husb=OpenUA300();                          // 打开设备,得到句柄

printf("test 1 \n");
ssinit(husb);                              // 单点采集初始化

// 显示 0－７通道数据８次
for(i=0;i<8;i++){
for(ch=0;ch<8;ch++){
   printf("%6d    ",(short int)ssad(husb,ch,0));
}
printf("\n");
}

printf("test 2 \n");
minit(husb,0,8,0);                         // 多点采集初始化,0-7 通道,倍数 1
readdata(husb, addat,50,256);              // 多点采集, 总频率 200 KHz, 共 256 点

//   显示数据
j=0;
for(i=0;i<8;i++){
for(ch=0;ch<8;ch++){
printf("%6d    ",addat[j++]);
```

```
}
printf("\n");
}

CloseUA300(husb);                    //   关闭设备
return 0;
}
```

头部文件:

```
//ua307a.h
HANDLE _stdcall OpenUA300();
HANDLE _stdcall CloseUA300(HANDLE husb);
void _stdcall ssinit(HANDLE husb);
short _stdcall ssad(HANDLE husb,short adch,short gain);
UCHAR _stdcall GetVer(HANDLE husb);
void _stdcall minit(HANDLE husb, short fch,short chn, short gain);
void _stdcall readdata(HANDLE husb, short *addat,unsigned short fcode, long leng);
void _stdcall minitz(HANDLE husb, short fch,short chn, short gain,unsigned short fn);
void _stdcall readdataz(HANDLE husb, short *addat,unsigned short fcode, long leng);
void _stdcall minit2(HANDLE husb, short fch, short chn,unsigned short fcode, short gain);
void _stdcall startad(HANDLE husb, short fch, short chn,unsigned short fcode, short gain);
void _stdcall minit2z(HANDLE husb, short fch, short chn, short gain, unsigned short fn,unsigned short
fcode)
void _stdcall startadz(HANDLE husb, short fch, short chn, short gain, unsigned short fn,unsigned short
fcode)
void _stdcall readdata2(HANDLE husb, short *addat, long leng);
void _stdcall endread2(HANDLE husb);
void _stdcall dout(HANDLE husb, UCHAR ddat);
UCHAR _stdcall din(HANDLE husb)
```

2. VB６调用例子

```
Dim addaτ(20000) As Integer
Dim husb As Long
Dim i As Integer

Private Declare Function OpenUA300 Lib "ua307a.dll" () As Long
Private Declare Function CloseUA300 Lib "ua307a.dll" (ByVal Handle As Long) As Long
Private Declare Function ssad Lib "ua307a.dll" (ByVal Handle As Long, ByVal n1%, ByVal n2%) As
```

```
Integer
Private Declare Sub ssinit Lib "ua307a.dll" (ByVal Handle As Long)
Private Declare Function GetVer Lib "ua307a.dll" (ByVal Handle As Long) As Byte
Private Declare Sub minit Lib "ua307a.dll" (ByVal Handle As Long, ByVal n1%, ByVal n2%, ByVal
n3%)
Private Declare Sub readdata Lib "ua307a.dll" (ByVal Handle As Long, dd1 As Integer, ByVal n1%,
ByVal leg As Long)
Private Declare Sub minit2 Lib "ua307a.dll" (ByVal Handle As Long, ByVal n1%, ByVal n2%, ByVal
n3%, ByVal n4%)
Private Declare Sub minit2z Lib "ua307a.dll" (ByVal Handle As Long, ByVal n1%, ByVal n2%, ByVal
n3%, ByVal n4%, ByVal n5%)
Private Declare Sub startad Lib "ua307a.dll" (ByVal Handle As Long, ByVal n1%, ByVal n2%, ByVal
n3%, ByVal n4%)
Private Declare Sub startadz Lib "ua307a.dll" (ByVal Handle As Long, ByVal n1%, ByVal n2%, ByVal
n3%, ByVal n4%, ByVal n5%)
Private Declare Sub readdata2   Lib "ua307a.dll" (ByVal Handle As Long, dd1 As Integer,   ByVal   leg
As Long)
Private Declare Sub endread2   Lib "ua307a.dll" (ByVal Handle As Long)
．．．

husb = OpenUA300()
If husb = 0 Then
      i = MsgBox("  设备打开错误!!! ", 0, "警告!")
End If
ssinit (husb)
dd = ssad(husb, 2, 1)                ’读第二通道一点数据值，增益为２

minit husb, 0, 4, 0                  ’多点采集初始化，采０－３通道，增益为１
readdata husb, addaτ(0), 50, 4096     ’    多点采集，总频率 200KHz，共 4096 点（字）
．．．

’第二种方式多点采集调用例子
minit2   husb, 0, 4, 100, 0          ’多点采集初始化，采０－３通道，100KHz，增益为１
readdata husb, addaτ(0),   4096      ’读 4096 点（字）数据
．．．                               ’可进行数据处理
readdata husb, addaτ(4096),   4096   ’再读 4096 点（字）数据，接着存放，也可从(0)存放
．．．                               ’可进行数据处理
readdata husb, addaτ(8192),   4096   ’再读 4096 点（字）数据，接着存放，也可从(0)存放
．．．                               ’可进行数据处理
```

endread2　husb　　’结束第二种方式多点采集

CloseUA300 (husb)

三．准同步采样方式的编程

准同步采样方式采用 minitz,readdataz 函数，使用方法基本与上面 minit,readdata 相同，不同之处是：1. 在 minitz 中需要给出通道间采样频率.　2. 在 readdataz 中采样频率设定的是每通道的采样频率，而不是总采样频率。

第二种方式多点准同步采集用 minit2z,readdata2 函数或 startadz, readdata2 函数，见以上说明.

另外需注意的是：1.单通道采样时不采用这种方式. 2.本方式总采样频率的限制：对 UA307A 各通道采样频率之和<=400KHz. 以下给出准同步采样方式的编程例子：

１．VC++６调用例子

这是一个控制台方式的调用例子．编译时需加上 UA307a.h,　UA307a.lib

```
// UA307AT.cpp
#include "windows.h"
#include "stdafx.h"
#include "ua307a.h"

HANDLE husb;

int main(int argc, char* argv[])
{
short int ch;
short int i,j;
short int addat[4096];

printf("Hello World!\n");

husb=OpenUA300();                    // 打开设备,得到句柄

printf("test 1 \n");
minitz(husb,0,8,0,25);                // 多点采集初始化,0-7 通道,倍数 1,通道间频率 400KHz
readdataz(husb, addat,500,1024);          // 每通道采集频率  20 KHz,  总数据量 1024 点

//　显示数据
```

```
j=0;
for(i=0;i<8;i++){
for(ch=0;ch<8;ch++){
printf("%6d    ",addat[j++]);
}
printf("\n");
}
CloseUA300(husb);                    //    关闭设备
return 0;
}
```

2. VB６调用例子

```
Dim addaτ(20000) As Integer
Dim husb As Long
Dim i As Integer
Private Declare Function OpenUA300 Lib "ua307a.dll" () As Long
Private Declare Function CloseUA300 Lib "ua307a.dll" (ByVal Handle As Long) As Long
Private Declare Function ssad Lib "ua307a.dll" (ByVal Handle As Long, ByVal n1%, ByVal n2%) As
Integer
Private Declare Sub ssinit Lib "ua307a.dll" (ByVal Handle As Long)
Private Declare Function GetVer Lib "ua307a.dll" (ByVal Handle As Long) As Byte
Private Declare Sub minit Lib "ua307a.dll" (ByVal Handle As Long, ByVal n1%, ByVal n2%, ByVal
n3%)
Private Declare Sub readdata Lib "ua307a.dll" (ByVal Handle As Long, dd1 As Integer, ByVal n1%,
ByVal leg As Long)
Private Declare Sub minit2 Lib "ua307a.dll" (ByVal Handle As Long, ByVal n1%, ByVal n2%, ByVal
n3%, ByVal n4%)
Private Declare Sub minit2z Lib "ua307a.dll" (ByVal Handle As Long, ByVal n1%, ByVal n2%, ByVal
n3%, ByVal n4%, ByVal n5%)
Private Declare Sub startad Lib "ua307a.dll" (ByVal Handle As Long, ByVal n1%, ByVal n2%, ByVal
n3%, ByVal n4%)
Private Declare Sub startadz Lib "ua307a.dll" (ByVal Handle As Long, ByVal n1%, ByVal n2%, ByVal
n3%, ByVal n4%, ByVal n5%)
Private Declare Sub readdata2    Lib "ua307a.dll" (ByVal Handle As Long, dd1 As Integer,    ByVal    leg
As Long)
Private Declare Sub endread2    Lib "ua307a.dll" (ByVal Handle As Long)

. . .

husb = OpenUA300()
```

```
If husb = 0 Then
        i = MsgBox("  设备打开错误!!! ", 0, "警告!")
End If

minitz husb, 0, 4, 0,25                ’准同步采集初始化，采０－３通道，增益为１，通道间频率 100KHz
readdataz husb, addaτ(0), 500, 1024        ’  每通道采集频率  20 KHz,  总数据量 1024 点
．．．

CloseUA300 (husb)
```

更多编程例子见光盘。