

ASG Programming Manual

编程手册

保证和声明

软件版本

软件升级可能会增加产品功能，请联系国仪量子（合肥）技术有限公司升级软件，必要时我司会主动与您联系。

声明

- 本公司产品受中国及其他国家和地区的专利（包括已取得和正在申请的专利）保护。
- 本公司拥有改变产品规格及价格的权利。
- 本手册提供的信息取代以往出版的任何资料。
- 未经我司事先书面许可，不得影印、复制或改变本手册的任何部分。
- 用户一旦使用产品，即视为对本声明的全部内容认可和接受。

联系我们

- 电子邮箱：sale@cqtek.com, service@cqtek.com
- 电话：4000606976, 0551-63367168
- 企业官网：www.cqtek.com

目录

保证和声明	2
1、接口程序调用	4
1.1C++DLL 接口程序.....	4
1.2C++调用示例.....	9
1.3PYTHON 调用示例	11

1、接口程序调用

1.1C++ + DLL 接口程序

我司为用户提供了 C++ 的动态链接库，各开发语言都能方便的进行调用。用户可调用接口程序中的函数自己编程生成方波序列，并将其下载到硬件中然后控制产品播放生成的方波序列。现将用户所需的函数列出，并作出解释。

函数 int open()

此函数用来连接设备，返回值为 1 时表示连接成功，返回值为 0 时表示连接失败。此函数无参数。

函数 char* getDllInformation ()

参数：无

返回值：字符串

函数功能：获取 DLL 的基本信息。

函数 int setCallbackFunc (DATA_CALLBACK callback)

参数：DATA_CALLBACK 类型的函数指针[typedef void(*DATA_CALLBACK)(int type, char* data);]

返回值：0 设置失败,1 设置成功

函数功能：用于设置回调函数，主要返回 dll 信息和字符串提示。

**函数 int setCallbackFunc_int (DATA_CALLBACK_INT
callback)**

参数：DATA_CALLBACK 类型的函数指针[typedef
void(*DATA_CALLBACK_INT)(int type, int len, unsigned int* data);]

返回值：0 设置失败,1 设置成功

函数功能：用于设置回调函数，主要返回 count 采集到的数据，连续采集
返回一个 unsigned int 型数据，用 count 使能采集时返回 unsigned int 型数
组，数组维数即采集的个数，每个数值代表对应的采集区间采集的个数。

函数 int pulse_download(double **pulses, int length[8])

此函数用来下载脉冲序列，函数返回值为 int 类型，当返回为 1 时表示
下载成功，当返回为 0 时表示下载失败。函数参数说明如下:

参数	参数描述
double **pulses	参数 “pulses” 是一个 double 类 型的二维数组， 数组长度为 8，表 示 8 个通道的脉冲数据。每 个通道 的脉冲序列数据保存在一维数组中， 其中一维数组的第 0、2、4... 个元 素表示高 电平持续时间(单位为

	ns), 第 1、3、5... 个元素表示低电平持续时间。其中, 每一组高低电平时间组成一个脉冲数据, 高低电平时 间不能同时为 0。例 如:[[0,20,20,20],[0,20], [0,20], [0,20], [0,20], [0,20], [0,20], [0,20]] 表 示通道 1 的脉冲序列为 1 个 20ns 低电平 + 1 个 20ns 高电平和 20ns 低电平组合的两个脉 冲 , 其他 2-8 通道的脉冲均为一个 20ns 低电 平脉冲。
int length[8]	参数 “length” 是一个 int 类型的一维数组, 参 数为 8。8 个元素分别表示 8 个通道的脉冲 序列数据的数组长度。

函数 int counter_download (int*pulses, int length)

参数:

参数	参数描述
pulses	波形数据
length	数据的长度

返回值 : 0 下载失败,1 下载成功

函数功能：下载 count 数据，例

count_data=[20,20,20,20,20,20,20,20,500000000],length_count=8 , 下载长度为 8 的 count 数据。数据是高电平+低电平为一组 count , count_data 依次为高低高低... , 数据应为 5 的整数倍。

函数 int set_counter_repeat (int repeat)

参数：

参数	参数描述
repeat	重复次数(偶数)

返回值：0 设置失败,1 设置成功

函数功能：配置 count 采集的重复此时，表示重复 count 序列重复采集 repeat 次调用回调函数回复一次数据。

函数 int isCountContinue (int iIsContinue)

参数：

参数	参数描述
0	关闭
1	开启

返回值：0 设置失败,1 设置成功

函数功能：配置 count 连续采集功能

函数 int countTimeStep (int timeInterval)

参数：

参数	参数描述
timeInterval	连续采集时间间隔

返回值：0 设置失败,1 设置成功

函数功能：配置 count 连续采集时间间隔，每隔 timeInterval ns 回复一次数据。

函数 int countConfig (int iAsgChannelCtrl,int iCount)

参数:

参数	参数描述
iAsgChannelCtrl	asg 通道开关
iCount	count 功能开关

返回值：0 设置失败,1 设置成功

函数功能：配置 asg 和 count 功能的使能，iAsgChannelCtrl：高 8bit 表示 ASG 输出常高电平控制，每一 bit 对应一个通道，为 1 时，高电平输出，0 时，低电平输出。低 8bit 表示 ASG 通道开关。iCount：Bit[0] 表示 counter 计数功能开关。

函数 int start_download()

参数：无

返回值：0 失败，1 成功

函数功能：开始播放 asg 序列，同时开始采集 count。

函数 int stop_download()

参数：无

返回值：0 失败，1 成功

函数功能:向硬件发送指令，使仪器停止播放 asg 序列和停止 count 采集。

函数 int close_usb()

参数：无

返回值：0 失败，1 成功

函数功能：用来断开软件和仪器的连接。

1.2C++调用示例

```
#include<iostream>
#include "asg_dll.h"

void onASGCallback(int type, char* data)
{
    printf("type = %d info = %s",type,data);
}

int dataList[4096]{};
void onCountCallback(int type, int len, unsigned int* data) {
    if(type == 0)
    {
        for(int i=0;i<len;i++)
```

```

    {
        dataList[i] = data[i];
        printf("%d ",dataList[i]);
    }
}
else if(type == 3)
{
    printf("连续采集 = %d",*data);
}
}

int main()
{
    open();//连接

    printf("%s",getDllInfomation());//获取 dll 信息

    setCallbackFunc(onASGCallback);//设置字符串回调函数

    setCallbackFunc_int(onCountCallback);//设置 count 数据采集回调函数

    double channelData[10]={10,10,10,10,10,10,10,10,10,10};//单个 asg 序列
    int length[8]={10,10,10,10,10,10,10,10};//asg 各通道的数据长度
    double *asgData[8];//asg 波形序列
    for(int i=0;i<8;i++)
    {
        asgData[i] = channelData;
    }
    pulse_download(asgData,length);//下载 asg 波形

    int countData[]={20,20,20,100000000};
    int countLength = 4;
    counter_download(countData,countLength);

    countTimeStep(50000000);//50000000 * 20 = 1000000000ns = 1s

    set_counter_repeat(2);//count 重复次数 偶数

    isCountContinue(1);//开启连续计数

    countConfig(0xff,1);//配置 asg8 个通道开启，开启 count 功能

    start_download(1);//开始播放 asg，开始采集 count

```

```
stop_download();//停止 asg 播放，停止 count 采集  
}
```

1.3 Python 调用示例

```
import sys  
from ASG8005_PythonSDK import *  
from ctypes import *  
import ctypes  
import threading  
import time  
  
m_CountCount = 1  
  
@CFUNCTYPE(None, c_int, c_char_p) # 设置字符串回调函数，要返回固件信息和提示  
def status_callback(type, c_char_buff):  
    print(type)  
    print(c_char_buff)  
    return  
  
@CFUNCTYPE(None, c_int, c_int, POINTER(c_uint32)) # 设置 count 回调函数，  
dll 有数据时调用 python 处理  
def count_callback(type, len, c_int_buff):  
    dataList = []  
    for i in range(len):  
        dataList.append(c_int_buff[i])  
    typestr:str  
    if type == 0:  
        if m_CountCount != len:  
            print("数据错误")  
        typestr = 'count 计数: '  
    elif type == 3:  
        typestr = '连续计数 : '  
    print(typestr,"datList :",dataList)  
    return  
  
#实例化 asg 对象  
asg = ASG8005()  
  
#设置回调函数
```

[illegible]

```
print("asg.ASG_counter_download : ",asg.ASG_counter_download(count_data
,length_count))

#配置循环次数
counter_repeat = 1# 1x2 = 4
print("asg.ASG_set_counter_repeat :
",asg.ASG_set_counter_repeat(counter_repeat))

#配置连续采集时间间隔
count_timeStep = 50000000 # 50000000 x 20 = 1000000000ns = 1s
print("asg.ASG_countTimeStep() :
",asg.ASG_countTimeStep(count_timeStep))

#开启连续计数功能
bIsContinue = 1
print("asg.ASG_isCountContinue: ",asg.ASG_isCountContinue(bIsContinue))

#配置 asg 和 count 功能 有两个参数第一个是开启 count 功能，第二个是开启 asg 功
能，第二个默认 asg 全开
print("asg.ASG_countConfig: ",asg.ASG_countConfig(bIsContinue))

#开始播放
#input("press Enter to start")
print("asg.start(): ",asg.start())

#停止播放
input("press Enter to stop")
print("asg.stop(): ",asg.stop())

#关闭连接
print("asg.close_device(): ",asg.close_device())
```



扫码关注 了解更多

用量子技术感知世界
FEEL THE WORLD IN A QUANTUM WAY

国仪量子(合肥)技术有限公司

地址：合肥市高新区创新产业园二期E2楼

无锡量子感知研究所

地址：无锡市惠山区惠山站区站前路2号

国仪量子(上海)测量技术有限公司

地址：上海市虹口区北外滩峨眉路315号8405室

☎ 400-0606-976

🌐 www.ciqtek.com

✉ gylz@ciqtek.com