



三英精控

SYMC-SC-300(400) Controller

Program Handbook

Sanying MotionControl Instruments Ltd.

All rights reserved by Sanying MotionControl Instruments Ltd.

1. Constitution of SDK

sc300dll.dll a native C++ Library

sc300dll.lib import library of **sc300dll.dll**

sc300lib.h header file of **sc300dll.dll**, constitute of macro definition、struct definition、function declaration.

2. How to use SDK

sc300dll.dll is the encapsulation of all functions, so you can only use this DLL.

There are two methods to use dll in vc++ project:

一、call :

You should link **sc300dll.lib** in your vc++ project, then call API function.

二、call :

- 1、Create a function pointer, the type must meet the function in DLL.
- 2、Call LoadLibrary() to load DLL, return instance handle of DLL.
- 3、Call GetProcAddress() to get function address of DLL, the result assign to the function pointer.
- 4、Call function pointer to use function off DLL finally.
- 5、Call FreeLibrary() to release DLL at last.

VC example refer to \VCDemo directory.

C# example refer to \C#Demo directory.

Labview example refer to \LabviewDemo directory.

3. Technical support

Please connect with us via email if you encounter a problem by using this Dll, we will serve you sincerely.

email: support@symc-tec.com

1. SDK API LIST

Function name	description
SC300_Dev_List	Find All Controller Info
SC300_Open	Open Controller
SC300_Close	Close Controller
SC300_SetServo	Set open loop or close loop
SC300_GetServo	Read open loop or close loop
SC300_GetPos	Read current position
SC300_SetPos	Set target position
SC300_SetDelayTime	Set DelayTime
SC300_GetDelayTime	Read DelayTime
SC300_GetRemainTime	Read RemainTime
SC300_GetDelayOperationStatus	Read status of DelayOperation
SC300_SaveConfig	Save config to FLASH

错误码

-1	通讯连接错误
-2	通道索引号错误
-3	传入的参数错误
-4	接口索引参错误
-5	延时等待时间未到

2. SDK API DETAILED INTRODUCTION

SC300_Dev_List(`int *devInfo`)

description : find all controller interfaces
argument : array of controller interfaces' info
return : return number of controller interfaces(>0)
explanation : call this function firstly; array of devInfo include SN and interface type of every controller(type:0 USB interface;1 ETH interface).

VC code:

```
int devinfo[8];
```

```

int devnum = SC300_Dev_List(devinfo);
if(devnum<1)
{
    return;
}
for(int i=0;i<devnum; i++)
{
    int sn=devinfo[2*i];
    int type=devinfo[2*i+1];
    string stype;
    if(type==0)
    {
        stype="USB";
    }
    else
    {
        stype="ETH";
    }
    printf("Device %d :  sn=%d    type=%s\n", i,sn,stype.c_str());
}
c# code
using System.Runtime.InteropServices;
[DllImport("sc300dll.dll", EntryPoint = "SC300_Dev_List", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
private static extern int SC300_Dev_List(IntPtr devInfo);
int[] devInfo = new int[8] { 0, 0, 0, 0, 0, 0, 0, 0 };
GCHandle hObject = GCHandle.Alloc(devInfo, GCHandleType.Pinned);
IntPtr ptr_devInfo = hObject.AddrOfPinnedObject();
int devnum= SC300_Dev_List(ptr_devInfo);
if(devnum<1)
{
    return;
}
for (int i = 0; i < devnum; i++)
{

```

```

    int sn = devInfo[2 * i];
    int type = devInfo[2 * i + 1];
    string stype;
    if (type == 0)
    {
        stype = "USB";
    }
    else
    {
        stype = "ETH";
    }
    string info;
    info = string.Format("Device {0} : sn={1}    type={2}\n", i, sn, stype);
    Console.WriteLine(info);
}

```

SC300_Open(int index)

description : open controller
argument : index of controller interfaces (from 0 to devnum-1), not SN
return : 1 success, -1 fail
explanation : call this function after SC300_Dev_List;

VC code:

```

if(SC300_Open(0))
{
    //success
}

c# code
using System.Runtime.InteropServices;
[DllImport("sc300dll.dll", EntryPoint =
"SC300_Open", CallingConvention=CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
private static extern int SC300_Open(int index);
if(SC300_Open(0))
{
    //success
}

```

SC300_Close(int index)

description : close control
argument : index of controller interfaces (from 0 to devnum-1), not SN
return : none
explanation : call this function at last;

VC code:

```
SC300_Close(0);
```

c# code

```
[DllImport("sc300d11.dll", EntryPoint =
"SC300_Close", CallingConvention=CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
private static extern void SC300_Close(int index);
SC300_Close(0);
```

SC300_SetServo(int index, int channel, int servo)

description : set open loop or close loop
argument : index=index of controller interfaces (from 0 to devnum-1), not SN; channel=channel index 1~3; servo=close loop 1, openloop 0
return : 1 success , -2 channel invalid,-3 servo invalid

VC code:

```
if(SC300_SetServo(0,1, 1))//interface 0 channel 1 to set close loop
{
```

```
    //success
```

```
}
```

c# code

```
[DllImport("sc300d11.dll", EntryPoint = "SC300_SetServo", CallingConvention =
 CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
private static extern int SC300_SetServo(int index, int channel, int servo);
if(SC300_SetServo(0,1, 1))//device 0 channel 1 to set close loop
{
```

```
    //success
```

```
}
```

SC300_SetGervo(int index, int channel, int *servo)

description : Read open loop or close loop
argument : index=index of controller interfaces (from 0 to

```

devnum-1), not SN; channel=channel index 1~3; *servo=1(close loop) ,
0(openloop)
return      : 1 success , -2 channel invalid
VC code:
Int servo=-1;
if(SC300_GetServo(0, 1, &servo))//interface 0 channel 1 to Read open loop
or close loop
{
    //success
}
c# code
[DllImport("sc300dll.dll", EntryPoint = "SC300_GetServo", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
private static extern int SC300_GetServo(itn index, int channel, ref int servo);
int servo=-1;
if(SC300_GetServo(0,1, ref servo))//interface 0 channel 1 to Read open
loop or close loop
{
    //success
}

```

```

SC300_SetPos(int index, int channel, float pos)

description : set target position
argument      : index=index of controller interfaces(from 0 to
devnum-1), not SN; channel=channel index 1~3; pos=target position
return        : 1 success, -2 channel invalid,-3 pos out of range

VC code:
if(SC300_SetPos(0, 1, 1))//interface 0 channel 1 to set target position
1
{
    //success
}
c# code
[DllImport("sc300dll.dll", EntryPoint = "SC300_SetPos", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
private static extern int SC300_SetPos(int index, int channel, float pos);

```

```
if(SC300_SetPos(0, 1, 1)) //interface 0 channel 1 to set target position  
1  
{  
    //success  
}
```

SC300_GetPos(int index, int channel, float *pos)
description: read current position
argument : index=index of controller interfaces (from 0 to devnum-1), not SN; channel=channel index 1~3; pos=current position
return : 1 success , -2 channel invalid

VC code:

```
float pos=0;  
if(SC300_GetPos(0, 1, &pos)) //read interface 0 channel 1 current  
position
```

```
{  
    //success  
}
```

c# code

```
[DllImport("sc300d11.dll", EntryPoint = "SC300_GetPos", CallingConvention =  
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
```

```
private static extern int SC300_GetPos(int index, int channel, ref float pos);  
float pos=0;
```

```
if(SC300_GetPos(0, 1, ref pos)) //read interface 0 channel 1 current  
position
```

```
{  
    //success  
}
```

SC300_SetDelayTime(int index, int delaytime)

description: set delay time
argument : index=index of controller interfaces (from 0 to devnum-1), not SN; **delaytime**=delay time

return : 1 success , <0 fail

VC code:

```
if(SC300_SetDelayTime(0, 30))
{
    //success
}
```

c# code:

```
[DllImport("sc300d11.dll", EntryPoint = "SC300_SetDelayTime", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
private static extern int SC300_SetDelayTime(int index, int delaytime);
```

As above.

SC300_GetDelayTime(int index, int *delaytime)

description: read delay time
argument : index=index of controller interfaces (from 0 to devnum-1), not SN; delaytime=current delay time
return : 1 success , <0 fail

VC code:

```
int delaytime=0;
If(SC300_GetDelayTime(0, &delaytime))
{
    //success
}
```

c# code:

```
[DllImport("sc300d11.dll", EntryPoint = "SC300_GetDelayTime", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
private static extern int SC300_GetDelayTime(int index, ref int delaytime);
int delaytime=0;
If(SC300_GetDelayTime(0, ref delaytime))
{
    //success
}
```

SC300_GetRemainTime(int index, int *remainTime)

description: read current remain time
argument : index=index of controller interfaces (from 0 to devnum-1), not

SN; remaintime=current remain time

return : 1 success , <0 fail

VC code:

```
int remaintime=0;  
If(SC300_GetRemainTime(0, &remaintime))  
{  
    //success  
}
```

c# code:

```
[DllImport("sc300dll.dll", EntryPoint = "SC300_GetRemainTime", CallingConvention =  
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]  
private static extern int SC300_GetRemainTime(int index, ref int remaintime);  
int remaintime=0;  
if(SC300_GetRemainTime(0, ref remaintime))  
{  
    //success  
}
```

SC300_GetDelayOperationStatus(int index, int *delaystatus)

description: read current delay status

argument : **index**=index of controller interfaces (from 0 to devnum-1), not

SN; delaystatus=1(end of this delay mode), 0(otherwise)

return : 1 success , <0 fail

VC code:

```
int delaystatus=0;  
if(SC300_GetDelayOperationStatus(0, &delaystatus))  
{  
    //success  
}
```

c# code:

```
[DllImport("sc300dll.dll", EntryPoint = "SC300_GetDelayOperationStatus",  
 CallingConvention = CallingConvention.Cdecl, CharSet = CharSet.Ansi)]  
private static extern int SC300_GetDelayOperationStatus(int index, ref int delaystatus);  
int delaystatus=0;
```

```
if(SC300_GetDelayOperationStatus(0, ref delaystatus))
{
    //success
}

SC300_SaveConfig(int index)

description: save config to FLASH
argument : index=index of controller interfaces (from 0 to devnum-1), not
SN
return : 1 success , <0 fail
```

VC code:

```
if(SC300_SaveConfig(0))
```

```
{
```

```
    //success
```

```
}
```

c# code:

```
[DllImport("sc300dll.dll", EntryPoint = "SC300_SaveConfig", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
private static extern int SC300_SaveConfig(int index);
```

As above.

Company Headquarters : C21 Building,Venture Headquarter Base,
Fuyuan Road,Wuqing Development,Tianjin,China

Zip code: 301700

TEL: +86-022-22977677

FAX: +86-022-29516023

<http://en.symc-tec.com/>

