IMC-01 Controller

Program Handbook

2018.11

Sanying MotionControl Instruments Ltd.

Content

1、CONSTITUTION OF SDK

PZDll.dll a native C++ Library

PZDll.lib import library of **PZDll.dll**

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

1.1 How to use SDK

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

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

一、static call ：

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

二、dynamic 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.

1.2 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

2、SDK API LIST

Function	Name	description
Communication	Open	Open controller
	OpenByAddress	Open controller
	Close	Close controller
Control to Move	Start	Begin to move
Control to Move	Stop	Stop motion
Zero Position	Zero	Set current position to zero
Settings	SaveConfig	Save current axis settings to flash
Position Set	WritePos	Set absolute position
Encoder Resolution	WriteEncoderResolution	Set encoder resolution
Step Resolution	WriteStepResolution	Set DAC steps per micro resolution
Encoder Polarity	WriteEncoderPolarity	Set encoder polarity
Motor Polarity	WriteMotorPolarity	Set motor polarity
Open Closed Loop	WriteOpenClosedLoopMode	Set open closed loop
KP	WriteKP	Set KP
KI	WriteKI	Set KI
KD	WriteKD	Set KD
Deadband	WriteDeadband	Set deadband for close loop
Timeout	WriteTimeout	Set timeout for close loop
Velocity	WriteVelocity	Set velocity
Acceleration	WriteAcceleration	Set acceleration
Deceleration	WriteDeceleration	Set deceleration
MaxAcceleration	WriteMaxAcceleration	Set max acceleration
JOGAcceleration	WriteJogAcceleration	Set JOG acceleration
Limit Position	WriteTravelNegative	Set negative soft limit position
Limit Position	WriteTravelPositive	Set positive soft limit position
Step Position	ReadStepPos	Get steps position
Encoder Position	ReadEncoderPos	Get encoder position
Encoder Polarity	ReadEncoderPolarity	Get encoder polarity(-1 or 1)
Motor Polarity	ReadMotorPolarity	Get motor polarity(-1 or 1)
Encoder Resolution	ReadEncoderResolution	Get encoder resolution

Step Resolution	ReadStepResolution	Get step resolution
Open Closed Loop	ReadOpenClosedLoopMode	Get open closed loop
KP	ReadKP	Get KP
KI	ReadKI	GetKI
KI	ReadKD	GetKD
Deadband	ReadDeadband	Get deadband
Timeout	ReadTimeout	Get timeout
Velocity	ReadVelocity	Get velocity
Acceleration	ReadAcceleration	Get acceleration
Deceleration	ReadDeceleration	Get deceleration
MaxAcceleration	ReadMaxAcceleration	Get max acceleration
Jog Acceleration	ReadJogAcceleration	Get Jog acceleration
Limit Position	ReadTravelNegative	Get negative soft limit position
Limit Position	ReadTravelPositive	Get positive soft limit position
Status	ReadStatus	Get status(1 means positive soft limit position,2 means negative soft limit position)
Sync configure	SyncConfig	Set sync axis and target
Sync start or stop	SyncControl	Start or stop Sync moving
Sync status	ReadSyncStatus	Read status of sync axis
Home	Home	Go home
Home status	ReadHomeStatus	Read status of home
MoveMode	SetNormalMoveMode	Set Normal Mode
MoveMode	SetHighResolutionMoveMode	Set High Resolution Mode
MoveMode	ReadMoveMode	Read Move Mode

3、SDK API DETAILED INTRODUCTION

int Open(char *com)

description : Open controller
argument : char *com - serial number; default first address is 10;default scan number is 8
return : number of axis
explanation : call this function firstly

VC And C# Example:

```
int number=0;
if(number=Open(“Com1”)>0)
{
    //Open Success
}
```

c# Import

```
using System.Runtime.InteropServices;
[DllImport("PZDll.dll", EntryPoint = "Open", CallingConvention = CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int Open(string com);
```

int OpenByAddress(char *com，int startaddress,int scannumber)

description : Open controller
argument : char *com - serial number; startaddress is first address ;scannumber is scan number
return : number of axis
explanation : call this function firstly

VC And C# Example:

```
int number=0;
if(number=OpenByAddress(“Com1”,8,4)>0)
{
    //Open Success
}
```

c# Import

```
using System.Runtime.InteropServices;
```

```
[DllImport("PZDll.dll", EntryPoint = "OpenByAddress", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int OpenByAddress(string com);
```

void Close()

```
description  : Close controller
argument     : none
Return       : none
explanation  : call this function to close controller
```

VC And C# Example:

```
Close();
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "Close", CallingConvention = CallingConvention.Cdecl,
CharSet = CharSet.Ansi)]
public static extern void Close();
```

int Start(int chnl)

```
description  : begin to move
argument     : chnl - 0~number-1
Return       : -1 Set fail, 0 Set Success
```

VC And C# Example:

```
if(Start(0)==0)//axis1 begin to move
{
    //Set Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "Start", CallingConvention = CallingConvention.Cdecl,
CharSet = CharSet.Ansi)]
public static extern int Start(int channel);
```

int Stop(int chnl)

```
description  : stop motion
argument     : chnl - 0~number-1
Return       : -1 Set fail, 0 Set Success
```

VC And C# Example:

```
if(Stop(0)==0)//stop motion of axis1
{
    //Set Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint ="Stop"
, CallingConvention = CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int Stop(int channel);
```

int Zero(int chnl)

description : set zero position
argument : chnl - 0~number-1
Return : -1 Set fail, 0 Set Success

VC And C# Example:

```
if(Zero(0)==0)//set current position of axis1 to zero
{
    //Set Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "Zero", CallingConvention = CallingConvention.Cdecl,
CharSet = CharSet.Ansi)]
public static extern int Zero(int channel);
```

int SaveConfig(int chnl)

description : save settings to flash
argument : chnl - 0~number-1
Return : -1 Set fail,0 Set Success

VC And C# Example:

```
if(SaveConfig(0)==0)//save settings of axis1 to flash
{
    //Set Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "SaveConfig", CallingConvention =
```

```
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int SaveConfig(int channel);
```

int WritePos(int chnl,float data)

description : set target position and begin to move,unit is mm
argument : chnl - 0~number-1; data - the value to set
Return : -1 Set fail,0 Set Success
VC And C# Example:

```
if(WritePos(0, 1)==0)//Set value of axis1 to 1
{
    //Set Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "WritePos", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int WritePos(int channel,float value);
```

int WriteEncoderResolution(int chnl,float data)

description : set encoder resolution,unit is um/cnt
argument : chnl - 0~number-1; data - the value to set
Return : -1 Set fail, 0 Set Success

VC And C# Example:

```
if(WriteEncoderResolution(0, 0.05)==0)//Set value of axis1 to 0.05um
{
    //Set Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "WriteEncoderResolution", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int WriteEncoderResolution(int channel, float value);
```

int WriteStepResolution(int chnl,float data)

description : set step resolution,unit is steps/um
argument : chnl - 0~number-1; data - the value to set(steps/um)
Return : -1 Set fail, 0 Set Success

VC And C# Example:

```
if(WriteStepResolution(0, 12500)==0)//Set value of axis1 to 12500
{
    //Set Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "WriteStepResolution", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int WriteStepResolution(int channel, float value);
```

int WriteEncoderPolarity(int chnl,float data)

description : set encoder polarity,1default , -1 reverse
argument : chnl - 0~number-1; data - the value to set (um/cnt)
Return : -1 Set fail,0 Set Success

VC And C# Example:

```
if(WriteEncoderPolarity(0, 1)==0)//Set value of axis1 to 1
{
    //Set Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "WriteEncoderPolarity", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int WriteEncoderPolarity(int channel, float value);
```

int WriteMotorPolarity(int chnl,float data)

description : set motor polarity,1default , -1 reverse
argument : chnl - 0~number-1; data - the value to set
Return : -1 Set fail,0 Set Success

VC And C# Example:

```
if(WriteMotorPolarity(0, 1)==0)//Set value of axis1 to 1
{
    //Set Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "WriteMotorPolarity", CallingConvention =
```

```
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int WriteMotorPolarity(int channel, float value);
```

int WriteOpenClosedLoopMode(int chnl,float data)

```
description   : set open or closed loop
argument      : chnl - 0~number-1; data - the value to set  (0 traditional
open loop, 1 special open loop, 2 closed loop)
Return        : -1 Set fail,0 Set Success
```

VC And C# Example:

```
if(WriteOpenClosedLoopMode(0, 2)==0)//Set value of axis1 to 2 closed loop
{
    //Set Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "WriteOpenClosedLoopMode", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int WriteOpenClosedLoopMode(int channel, float value);
```

int WriteKP(int chnl,float data)

```
description   : set KP
argument      : chnl - 0~number-1; data - the value to set
Return        : -1 Set fail,0 Set Success
```

VC And C# Example:

```
if(WriteKP(0, 0.12)==0)//Set value of axis1 to 0.12
{
    //Set Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "WriteKP", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int WriteKP(int channel, float value);
```

int WriteKI(int chnl,float data)

```
description   : set KI
argument      : chnl - 0~number-1; data - the value to set
```

Return : -1 Set fail,0 Set Success

VC And C# Example:

```
if(WriteKI(0, 0.07)==0)//Set value of axis1 to 0.07
{
    //Set Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "WriteKI", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int WriteKI(int channel, float value);
```

int WriteKD(int chnl,float data)

description : set KD
argument : chnl - 0~number-1; data - the value to set
Return : -1 Set fail,0 Set Success

VC And C# Example:

```
if(WriteKD(0, 0)==0)//Set value of axis1 to 0
{
    //Set Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "WriteKD", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int WriteKD(int channel, float value);
```

int WriteDeadband(int chnl,float data)

description : set deadband,unit is cnt
argument : chnl - 0~number-1; data - the value to set
Return : -1 Set fail,0 Set Success

VC And C# Example:

```
if(WriteDeadband(0, 0)==0)//Set value of axis1 to 0
{
    //Set Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "WriteDeadband", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int WriteDeadband(int channel, float value);
```

int WriteTimeout(int chnl,float data)

```
description   : set timeout,uint is second
argument      : chnl - 0~number-1; data - the value to set
Return        : -1 Set fail,0 Set Success
```

VC And C# Example:

```
if(WriteTimeout(0, 1)==0)//Set value of axis1 to 1
{
    //Set Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "WriteTimeout", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int WriteTimeout(int channel, float value);
```

int WriteVelocity(int chnl,float data)

```
description   : set velocity,unit is mm/s
argument      : chnl - 0~number-1; data - the value to set
Return        : -1 Set fail,0 Set Success
```

VC And C# Example:

```
if(WriteVelocity(0, 0.2)==0)//Set value of axis1 to 0.2
{
    //Set Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "WriteVelocity", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int WriteVelocity(int channel, float value);
```

int WriteAcceleration(int chnl,float data)

```
description   : set acceleration,unit is mm/s²
argument      : chnl - 0~number-1; data - the value to set
```

Return : -1 Set fail,0 Set Success

VC And C# Example:

```
if(WriteAcceleration(0, 1)==0)//Set value of axis1 to 1
{
    //Set Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "WriteAcceleration", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int WriteAcceleration(int channel, float value);
```

int WriteDeceleration(int chnl,float data)

description : set deceleration,unit is mm/s^2
argument : chnl - 0~number-1; data - the value to set
Return : -1 Set fail,0 Set Success

VC And C# Example:

```
if(WriteDeceleration(0, 1)==0)//Set value of axis1 to 1
{
    //Set Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "WriteDeceleration", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int WriteDeceleration(int channel, float value);
```

int WriteMaxAcceleration(int chnl,float data)

description : set max acceleration,unit is mm/s^2
argument : chnl - 0~number-1; data - the value to set
Return : -1 Set fail,0 Set Success

VC And C# Example:

```
if(WriteMaxAcceleration(0, 1)==0)//Set value of axis1 to 1
{
    //Set Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "WriteMaxAcceleration", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int WriteMaxAcceleration(int channel, float value);
```

int WriteJogAcceleration(int chnl,float data)

description : set Jog acceleration,unit is mm/s^2
argument : chnl - 0~number-1; data - the value to set
Return : -1 Set fail,0 Set Success

VC And C# Example:

```
if(WriteJogAcceleration(0, 1)==0)//Set value of axis1 to 1
{
    //Set Success
}
```

c# Import

```
 [DllImport("PZDll.dll", EntryPoint = "WriteJogAcceleration", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int WriteJogAcceleration(int channel, float value);
```

int WriteTravelNegative(int chnl,float data)

description : set negative limit position,unit is mm
argument : chnl - 0~number-1; data - the value to set
Return : -1 Set fail,0 Set Success

VC And C# Example:

```
if(WriteTravelNegative(0, -12.5)==0)//Set value of axis1 to 12.5
{
    //Set Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "WriteTravelNegative", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int WriteTravelNegative(int channel, float value);
```

int WriteTravelPositive(int chnl,float data)

description : set positive limit position,unit is mm
argument : chnl - 0~number-1; data - the value to set

Return : -1 Set fail,0 Set Success

VC And C# Example:

```
if(WriteTravelPositive(0, 12.5)==0)//Set value of axis1 to 12.5
{
    //Set Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "WriteTravelPositive", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int WriteTravelPositive(int channel, float value);
```

int ReadStepPos(int chnl,float *data)

description : Get step position,unit is mm
argument : chnl - 0~number-1; data - save the read value
Return : -1 Get fail, 0 Get Success

VC Example:

```
float pos=0;
if(ReadStepPos(0, &pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "ReadStepPos", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int ReadStepPos(int channel, ref float value);
```

C# Example:

```
float pos=0;
if(ReadStepPos(0, ref pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

int ReadEncoderPos(int chnl,float *data)

description : Get encoder position,unit is mm
argument : chnl - 0~number-1; data -save the read value

Return : -1 Get fail,0 Get success

VC Example:

```
float pos=0;
if(ReadEncoderPos(0, &pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "ReadEncoderPos", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int ReadEncoderPos(int channel, ref float value);
```

C# Example:

```
float pos=0;
if(ReadEncoderPos(0, ref pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

int ReadEncoderPolarity(int chnl,float *data)

description : Get encoder polarity

argument : chnl - 0~number-1; data - save the read value,1default , -1 reverse

Return : -1 Get fail,0 Get success

VC Example:

```
float pos=0;
if(ReadEncoderPolarity(0, &pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "ReadEncoderPolarity", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int ReadEncoderPolarity(int channel, ref float value);
```

C# Example:

```
float pos=0;
if(ReadEncoderPolarity(0, ref pos)==0)//Get value of axis1 stored in pos
```

```
{
    //Get Success
}
```

int ReadMotorPolarity(int chnl,float *data)

description : Get motor polarity
argument   : chnl - 0~number-1; data - save the read value , 1default , -1 reverse
Return   : -1 Get fail,0 Get success

VC Example:

```
float pos=0;
if(ReadMotorPolarity(0, &pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "ReadMotorPolarity", CallingConvention = CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int ReadMotorPolarity(int channel, ref float value);
```

C# Example:

```
float pos=0;
if(ReadMotorPolarity(0, ref pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

int ReadEncoderResolution(int chnl,float *data)

description : Get encoder resolution,uint is um/cnt

argument   : chnl - 0~number-1; data - save the read value
Return   : -1 Get fail,0 Get success

VC Example:

```
float pos=0;
if(ReadEncoderResolution(0, &pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
```

```
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "ReadEncoderResolution", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int ReadEncoderResolution(int channel, ref float value);
```

C# Example:

```
float pos=0;
if(ReadEncoderResolution(0, ref pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

int ReadStepResolution(int chnl,float *data)

description : Get step resolution,uint is steps/um
argument : chnl - 0~number-1; data - save the read value
Return : -1 Get fail,0 Get success

VC Example:

```
float pos=0;
if(ReadStepResolution(0, &pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "ReadStepResolution", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int ReadStepResolution(int channel, ref float value);
```

C# Example:

```
float pos=0;
if(ReadStepResolution(0, ref pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

int ReadOpenClosedLoopMode(int chnl,float *data)

description : Get status of open or closed loop
argument : chnl - 0~number-1; data - save the read value(0 traditional

open loop, 1 special open loop, 2 closed loop)
Return : -1 Get fail,0 Get success

VC Example:

```
float pos=0;
if(ReadOpenClosedLoopMode(0, &pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "ReadOpenClosedLoopMode", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int ReadOpenClosedLoopMode(int channel, ref float value);
```

C# Example:

```
float pos=0;
if(ReadOpenClosedLoopMode(0, ref pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

int ReadKP(int chnl,float *data)

description : Get KP
argument : chnl - 0~number-1; data - save the read value
Return : -1 Get fail,0 Get success

VC Example:

```
float pos=0;
if(ReadKP(0, &pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "ReadKP", CallingConvention = CallingConvention.Cdecl,
CharSet = CharSet.Ansi)]
public static extern int ReadKP(int channel, ref float value);
```

C# Example:

```
float pos=0;
if(ReadKP(0, ref pos)==0)//Get value of axis1 stored in pos
```

```
{
   //Get Success
}
```

int ReadKI(int chnl,float *data)

description : Get KI
argument   : chnl - 0~number-1; data - save the read value
Return   : -1 Get fail,0 Get success

VC Example:

```
float pos=0;
if(ReadKI(0, &pos)==0)//Get value of axis1 stored in pos
{
   //Get Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "ReadKI", CallingConvention = CallingConvention.Cdecl,
CharSet = CharSet.Ansi)]
public static extern int ReadKI(int channel, ref float value);
```

C# Example:

```
float pos=0;
if(ReadKI(0, ref pos)==0)//Get value of axis1 stored in pos
{
   //Get Success
}
```

int ReadKD(int chnl,float *data)

description : Get KD
argument   : chnl - 0~number-1; data - save the read value
Return   : -1 Get fail,0 Get success

VC Example:

```
float pos=0;
if(ReadKD(0, &pos)==0)//Get value of axis1 stored in pos
{
   //Get Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "ReadKD", CallingConvention = CallingConvention.Cdecl,
CharSet = CharSet.Ansi)]
public static extern int ReadKD(int channel, ref float value);
```

C# Example:

```
float pos=0;
if(ReadKD(0, ref pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

int ReadDeadband(int chnl,float *data)

description : Get deadband,uint is cnt
argument : chnl - 0~number-1; data - save the read value
Return : -1 Get fail,0 Get success

VC Example:

```
float pos=0;
if(ReadDeadband(0, &pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "ReadDeadband", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int ReadDeadband(int channel, ref float value);
```

C# Example:

```
float pos=0;
if(ReadDeadband(0, ref pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

int ReadTimeout(int chnl,float *data)

description : Get timeout,uint is second
argument : chnl - 0~number-1; data - save the read value
Return : -1 Get fail,0 Get success

VC Example:

```
float pos=0;
if(ReadTimeout(0, &pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "ReadTimeout", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int ReadTimeout(int channel, ref float value);
```

C# Example:

```
float pos=0;
if(ReadTimeout(0, ref pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

int ReadVelocity(int chnl,float *data)

description : Get velocity,uint is mm/s
argument : chnl - 0~number-1; data - save the read value
Return : -1 Get fail,0 Get success

VC Example:

```
float pos=0;
if(ReadVelocity(0, &pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "ReadVelocity", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int ReadVelocity(int channel, ref float value);
```

C# Example:

```
float pos=0;
if(ReadVelocity(0, ref pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
```

```
}
```

int ReadAcceleration(int chnl,float *data)

description : Get acceleration,uint is mm/s²
argument : chnl - 0~number-1; data - save the read value
Return : -1 Get fail,0 Get success

VC Example:

```
float pos=0;
if(ReadAcceleration(0, &pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "ReadAcceleration", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int ReadAcceleration(int channel, ref float value);
```

C# Example:

```
float pos=0;
if(ReadAcceleration(0, ref pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

int ReadDeceleration(int chnl,float *data)

description : Get deceleration,uint is mm/s²
argument : chnl - 0~number-1; data - save the read value
Return : -1 Get fail,0 Get success

VC Example:

```
float pos=0;
if(ReadDeceleration(0, &pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

c# Import

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

```
public static extern int ReadDeceleration(int channel, ref float value);
```

C# Example:

```
float pos=0;
if(ReadDeceleration(0, ref pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

int ReadMaxAcceleration(int chnl,float *data)

description : Get max acceleration,uint is mm/s^2

argument : chnl - 0~number-1; data - save the read value

Return : -1 Get fail,0 Get success

VC Example:

```
float pos=0;
if(ReadMaxAcceleration(0, &pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "ReadMaxAcceleration", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int ReadMaxAcceleration(int channel, ref float value);
```

C# Example:

```
float pos=0;
if(ReadMaxAcceleration(0, ref pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

int ReadJogAcceleration(int chnl,float *data)

description : Get Jog acceleration,uint is mm/s^2

argument : chnl - 0~number-1; data - save the read value

Return : -1 Get fail,0 Get success

VC Example:

```
float pos=0;
if(ReadJogAcceleration(0, &pos)==0)//Get value of axis1 stored in pos
```

```
{
    //Get Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "ReadJogAcceleration", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int ReadJogAcceleration(int channel, ref float value);
```

C# Example:

```
float pos=0;
if(ReadJogAcceleration(0, ref pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

int ReadTravelNegative(int chnl,float *data)

description : Get negative limit position,uint is mm

argument : chnl - 0~number-1; data - save the read value

Return : -1 Get fail,0 Get success

VC Example:

```
float pos=0;
if(ReadTravelNegative(0, &pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "ReadTravelNegative", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int ReadTravelNegative(int channel, ref float value);
```

C# Example:

```
float pos=0;
if(ReadTravelNegative(0, ref pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

int ReadTravelPositive(int chnl,float *data)

description : Get positive limit position,uint is mm
argument : chnl - 0~number-1; data - save the read value
Return : -1 Get fail,0 Get success

VC Example:

```
float pos=0;
if(ReadTravelPositive(0, &pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "ReadTravelPositive", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int ReadTravelPositive(int channel, ref float value);
```

C# Example:

```
float pos=0;
if(ReadTravelPositive(0, ref pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

int ReadStatus(int chnl,float *data)

description : Get status
argument : chnl - 0~number-1; data - save the read value(1 positive limit, 2 negative limit)
Return : -1 Get fail,0 Get success

VC Example:

```
float pos=0;
if(ReadStatus(0, &pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

c# Import

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

```
public static extern int ReadStatus(int channel, ref float value);
```

C# Example:

```
float pos=0;
if(ReadStatus(0, ref pos)==0)//Get value of axis1 stored in pos
{
    //Get Success
}
```

int SyncConfig(int channel[],float target[],int number)

description : Set sync axis and target, uint of target is mm
argument : channels – axis to sync; target – target position, number – number of axis
Return : -1 Set fail,0 Get success

VC Example:

```
int channel[2] ={ 0,1};
float target[2] = {1,2};
if(SynConfig(channel, target,2)==0)
{
      //success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "SyncConfig", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int SyncConfig(int []channel, float []target,int number);
```

C# Example:

```
int []channel =new int[2]{ 0,1};
float []target =new float [2]{1,2};
if(SynConfig(channel, target,2)==0)
{
    //success
}
```

int SyncControl(float data)

description : Start or Stop to sync move
argument : data – 0 stop , 2 start
Return : -1 Set fail,0 Get success

C# and VC Example:

```
if(SynControl(0, 2)==0)
{
    //设置成功
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "SyncControl", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int SyncControl(float data);
```

int ReadSyncStatus(int chnl,bool *status)

description : Read status of sync axis,Call SyncControl(0) if all axis sync are completed

argument : chnl－0~number-1；status－true sync complete ，false sync not complete

Return : -1 Get fail,0 Get success

VC Example:

```
bool  status=false;
if(ReadSyncStatus(0, &status)==0)
{
    if(status)
    {
        //Sync Complete
    }
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "ReadSyncStatus", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int ReadSyncStatus(int channel, ref bool status);
```

C# Example:

```
bool  status=false;
if(ReadSyncStatus(0, ref status)==0)
{
    if(status)
    {
        //Sync complete
    }
```

```
}
```

int Home(int chnl,int dir)

description : Move to Home
argument : chnl - 0~number-1; dir - 0 clear status, 1 positive home, 2 negative home
Return : -1 Set fail,0 Get success
C# and VC Example:

```
if(Home(0, 2)==0)
{
    //success
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "Home", CallingConvention = CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int Home(int chnl,int dir);
```

int ReadHomeStatus(int chnl,bool *status)

description : Read status of home,Call Home(chnl,0) if status is true
argument : chnl - 0~number-1 ;status - true home complete, false home not complete
Return : -1 Get fail,0 Get success

VC Example:

```
bool  status=false;
if(ReadHomeStatus(0, &status)==0)//
{
    if(status)
    {
        //Home complete
    }
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "ReadHomeStatus", CallingConvention = CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int ReadHomeStatus(int channel, ref bool status);
```

C# Example:

```
bool  status=false;
if(ReadHomeStatus(0, ref status)==0)
{
    if(status)
    {
        //Home complete
    }
}
```

int SetNormalMoveMode(int chnl)

description : Set Normal Move Mode
argument : chnl - 0~number-1
Return : -1 Set fail,0 Set success

VC Example:

```
if(SetNormalMoveMode(0)==0)//
{
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "SetNormalMoveMode", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int SetNormalMoveMode(int channel);
```

C#Example:

```
if(SetNormalMoveMode(0)==0)//
{
}
```

int SetHighResolutionMoveMode(int chnl)

description : Set High Resolution Move Mode
argument : chnl - 0~number-1
Return : -1 Set fail,0 Set success

VC Example:

```
if(SetHighResolutionMoveMode(0)==0)//
{
}
```

c# Import

```
[DllImport("PZDll.dll", EntryPoint = "SetHighResolutionMoveMode
", CallingConvention = CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int SetHighResolutionMoveMode(int channel);
```

C# Example:

```
if(SetHighResolutionMoveMode(0)==0)//
{
}
```

int ReadMoveMode(int chnl,bool *highResolution)

description : ReadMoveMode

argument : chnl - 0~number-1 ;highResolution- true High Resolution Mode.

Return : -1 Get fail,0 Get success

VC Example:

```
bool mbNormal=false;
int ret = ReadMoveMode(0,&mbNormal);
```

c# Import

```
 [DllImport("PZDll.dll", EntryPoint = "ReadMoveMode", CallingConvention =
CallingConvention.Cdecl, CharSet = CharSet.Ansi)]
public static extern int ReadMoveMode(int channel, ref bool mbMoveMode);
```

C# Example:

```
bool mbhighResolution = false;
ReadMoveMode(0, ref mbhighResolution );
```

ADD: C21 Building, Venture Headquarter Base, Fuyuan Road, Wuqing Development, Tianjing, China 301700
TEL: +86-22-22977677
FAX: +86-22-29516023
E-mail: info@symc-tec.com
http://en.symc-tec.com/