**Robotpen SDK for Windows**

目录

[简介 1](#_Toc497152593)

[功能 1](#_Toc497152594)

[所需库 1](#_Toc497152595)

[API参考 2](#_Toc497152596)

# 简介

Robotpen SDK for Windows 提供与Robotpen设备的数据交互, 且在程序生成时直接与应用程序建立连接。本文主要介绍如何使用Robotpen SDK for Windows。

# 功能

Robotpen SDK for Windows 目前主要提供以下C++类:

RobotPenController类提供目前所有的SDK功能。

IRobotEventHandler接口类用于向应用程序发送回调通知。

Robotpen SDK for Windows可以提供两种方式收数据：事件和回调，用户可以通过调用ConnectInitialize选择使用方式

搭建项目环境要求：

Microsoft Visual C++ 2010或以上版本

Microsoft Visual Studio(推荐) 

Microsoft Windows 7 或以上版本

### 所需库

Robotpen SDK for Windows不依赖任何其他库，安装及使用 Robotpen SDK for Windows需要以下操作：

 将 SDK/include目录添加到项目的 INCLUDE目录下。

 将 SDK/lib目录放入项目的 LIB目录下，并确保 RobotUsbWrapper.lib与项目 有连接。

 将 SDK/dll下的 dll文件复制到你的可执行文件所在的目录下。

# API参考

RobotPenController接口类是 Robotpen SDK的基本接口，通过生成 RobotPenController 对象，调用该对象的方法使用 Robotpen SDK的功能。

创建 RobotPenController对象(GetInstance)

RobotPenController\* GetInstance();

创建 RobotPenController对象。RobotPenController对象提供的接口方法，非空即代表获取对象成功，其他为失败。

销毁RobotPenController对象(DestroyInstance)

void DestroyInstance(RobotPenController\* pController);

销毁RobotPenController对象。

初始化(ConnectInitialize)(回调)

void ConnectInitialize(eDeviceType nDeviceType,UsbDataCallback\_t pCallback = NULL, void \*pContext = NULL)

该方法用来初始化初始化设备信息，必须先调用该方法进行初始化，才能使用其他方法。

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| nDeviceType | 初始化设备的类型，参见eDeviceType |
| pCallBack | 接收到设备数据后的回调接口，用于通知客户 |
| pContext | 上下文 |

初始化(ConnectInitialize)(事件)

void ConnectInitialize(eDeviceType nDeviceType,IRobotEventHandler \*pEventHander = NULL)

该方法用来初始化初始化设备信息，必须先调用该方法进行初始化，才能使用其他方法。

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| nDeviceType | 初始化设备的类型，参见eDeviceType |
| pEventHander | 接收到设备数据后的回调接口，用于通知客户 |

连接设备(ConnectOpen)

int ConnectOpen()

该方法用来连接指定设备，设备状态通过事件或回调返回。

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| Return Value | 0: 方法调用成功  !0: 方法调用失败 |

断开设备(ConnectDispose)

void ConnectDispose()

该方法用来断开当前设备

是否连接(IsConnected)

bool IsConnected()

该方法用来打开指定设备

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| Return Value | true: 已连接  false: 未连接 |

发送命令(Send)

void Send(int nCmd)

该方法用来向设备发送命令，设备状态通过事件或回调返回。

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| nCmd | 命令类型，详见eRbtType |

升级(Update)

bool Update(const char \*fileMcu,const char \*fileBle,eDeviceType type = Unknow)

该方法用来升级固件，升级状态通过事件或回调返回。

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| fileMcu | Mcu文件路径，没有传“” |
| fileBle | Ble文件路径 ，没有传“” |
| eDeviceType | 设备的类型，参见eDeviceType |

设置(SetConfig)(此方法仅适用于支持2.4G的设备)

void SetConfig(int nCostumNum,int nClassNum,int nDeviceNum)

该方法用来设置设备的客户号、班级号、设备号，重启后生效。

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| nCostumNum | 客户号 |
| nClassNum | 班级号 |
| nDeviceNum | 设备号 |

获取设备数量(GetDeviceCount)

int GetDeviceCount()

该方法用来获取当前可用设备数量。

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| Return Value | 返回设备数量 |

获取设备信息 (GetDeviceInfo)

bool GetDeviceInfo(int index,USB\_INFO &usbInfo)

该方法用来设置设备的信息，调用此方法前必须先调用GetDeviceCount。

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| index | 获取序号 |
| usbInfo | 设备信息，参见USB\_INFO |
| Return Value | true 成功,false 失败 |

获取设备信息 (Open)

int Open(int nVid,int nPid,bool bAll = true)

该方法用来根据VID和PID打开设备。

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| nVid | 设备VID |
| nPid | 设备PID |
| bAll | 是否全部打开 |
| Return Value | 0: 方法调用成功  !0: 方法调用失败 |

连接蓝牙手写板(ConnectSlave)

void ConnectSlave(int nID)

该方法用来连接手写板，调用此方法前需要通过调用Send(DongleScanStart)来获取手写版信息, 仅限于Dongle设备。

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| nID | 连接设备ID，参见ST\_BLE\_DEVICE |

设置蓝牙手写板名称(SetSlaveName)

virtual void SetSlaveName(const char \*name)

该方法用来设置手写板名称, 仅限于Dongle设备。

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| name | 手写板名称，，名称字能用字母和数字，最大为16字节 |

设置设备类型(SetDeviceType)

void SetDeviceType(eDeviceType nDeviceType)

该方法用来设置设备类型，通过GetDeviceInfo获取设备信息来设置设备类型。

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| nDeviceType | 设备类型，参加eDeviceType |

设置手写板中心偏移(SetOffset)

void SetOffset(int nOffsetX,int nOffsetY)

该方法用来设置手写版中心偏移来获取可用区域。

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| nOffsetX | 水平偏移 |
| nOffsetY | 垂直偏移 |

设置设备类型(SetIsHorizontal)

void SetIsHorizontal(bool bHorizontal)

该方法用来设置设备的横竖屏。

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| bHorizontal | 是否是竖屏 |

获取设备宽(Width)

int Width() = 0;

该方法用来获取设备宽。

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| Return Value | 设备宽 |

获取设备高(Height)

int Height () = 0;

该方法用来获取设备高。

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| Return Value | 设备高 |

旋转设备(Rotate)

void Rotate(int nAngle = 0)

该方法用来旋转设备角度。

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| nAngle | nAngle 设备角度 为90的倍数 |

开始投票(VoteMulit)

void VoteMulit(bool bMulit = true)

该方法用来进行投票，仅限于网关设备。

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| bMulit | 是否是多选 |

设置笔的宽度(SetPenWidth)

void SetPenWidth(float nPenWidth = 0)

该方法用来设置笔的宽度。

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| nPenWidth | 笔的宽度 |

设置画布大小(SetCanvasSize)

void SetCanvasSize(int nWidth,int nHeight)

该方法用来设置画布大小。

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| nPenWidth | 笔的宽度 |

笔记优化(In)

void In(const PEN\_INFO &penInfo)

该方法用来输入原始笔迹信息进行笔记优化。

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| penInfo | 笔迹信息，参见PEN\_INFO |

是否开启压感(SetPressStatus)

void SetPressStatus(bool bPress)

该方法用来开启压感。

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| bPress | 是否开启 |

是否开启笔记优化(SetOptimizeStatus)

void SetOptimizeStatus(bool bOptimize)

该方法用来开启笔记优化。

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| bOptimize | 是否开启 |

是否开启笔记优化(SetKey)

void SetKey(const char \*key)

该方法用来设置不同的key来识别不同型号的设备

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| key | Key,32字节的uuid |

RobotPenController接口类用于 SDK向应用程序发送回调事件通知， 应用程序通过继承该接口类的方法获取 SDK的事件通知。接口类的所有方法都有缺省 （空）实现，应用程序可以根据需要只继承关心的事件。

设备插拔事件(onDeviceChanged)

void onDeviceChanged(eDeviceStatus status,int pid)

该方法表示相关设备已经插拔。回调参考ROBOT\_DEVICE\_CHANGED

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| status | 插拔状态，参见eDeviceStatus |
| pid | 插拔设备的pid |

设备插拔事件(onDeviceChanged)

void onDeviceChanged(eDeviceStatus status,eDeviceType type)

该方法表示相关设备已经插拔。回调参考ROBOT\_DEVICE\_CHANGED

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| status | 插拔状态，参见eDeviceStatus |
| type | 插拔设备类型 参见eDeviceType |

网关状态事件(onGatewayStatus)

void onGatewayStatus(eGatewayStatus status)

该回调方法表示当前网关状态，当网关状态改变时会触发此方法，仅限于网关设备。回调参考ROBOT\_GATEWAY\_STATUS

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| status | 网关状态，参见eGatewayStatus |

手写板状态事件(onNodeStatus)

void onNodeStatus(const NODE\_STATUS &status)

该回调方法表示当前手写板状态，当手写板状态改变时会触发此方法，仅限于USB连接。回调参考ROBOT\_NODE\_STATUS

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| status | 手写板状态，参见NODE\_STATUS |

设备信息事件(onDeviceInfo)

void onDeviceInfo(const ST\_DEVICE\_INFO &info)

该回调方法表示当前设备的相关信息，当打开设备时会触发此方法。回调参考ROBOT\_GATEWAY\_VERSION

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| info | 设备信息，参见ST\_DEVICE\_INFO |

子设备在线状态事件(onOnlineStatus)

void onOnlineStatus(uint8\_t \*status)

该回调方法表示当前子设备在线状态，当子设备状态改变时时会触发此方法，仅限于网关设备。回调参考ROBOT\_ONLINE\_STATUS

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| status | 60个子设备的在线状态信息，每个字节代表一个设备 |

单选结果事件(onExitVote)

void onExitVote(uint8\_t \*status)

该回调方法表示单选结果,当发送结束单选模式时会触发此方法，仅限于网关设备。回调参考ROBOT\_EXIT\_VOTE

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| status | 60个子设备的结果信息，每个字节代表一个设备 |

多选结果事件(onExitVoteMulit)

void onExitVoteMulit(const ST\_OPTION\_PACKET &packet)

该回调方法表示多选结果,当发送结束多选模式时会触发此方法，仅限于网关设备。回调参考ROBOT\_EXIT\_VOTE

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| packet | 单个设备的结果信息，参见ST\_OPTION\_PACKET |

MassData手写事件(onMassData)

void onMassData(int index,const PEN\_INFO &penInfo)

该回调方法表示子设备正在书写,当开启手写模式时会触发此方法，仅限于网关设备。回调参考ROBOT\_MASS\_DATA

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| index | 设备编号 |
| penInfo | 笔迹信息，参见PEN\_INFO |

多选结果事件(onGatewayError)

void onGatewayError(eNebulaError error)

该回调方法表示网关错误信息,当网关出现异常时会触发此方法，仅限于网关设备。回调参考ROBOT\_GATEWAY\_ERROR

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| error | 错误编号，参见eNebulaError |

MassData手写事件(onMassData)

void onMassData(int index,const PEN\_INFO &penInfo)

该回调方法表示子设备正在书写,当开启手写模式时会触发此方法，仅限于网关设备。回调参考ROBOT\_MASS\_DATA

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| index | 设备编号 |
| penInfo | 笔迹信息，参见PEN\_INFO |

设置设备信息事件(onSetDeviceNum)

void onSetDeviceNum(int result,int customid, int classid, int deviceid)

该回调方法表示设置设备信息，设置完成后会触发此方法，仅限于网关和支持2.4G手写版设备。回调参考ROBOT\_SET\_DEVICE\_NUM

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| result | 结果 0为成功，1为失败 |
| customid | 客户号 |
| classid | 班级号 |
| deviceid | 设备号 |

固件升级进度事件(onFirmwareData)

void onFirmwareData(int progress)

该回调方法表示固件升级进度，当进行固件升级时会触发此方法。回调参考ROBOT\_FIRMWARE\_DATA

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| progress | 进度 范围0-100 |

固件升级结果事件(onRawResult)

void onRawResult(int result)

该回调方法表示升级结果，当进行固件升级完成时会触发此方法。回调参考ROBOT\_RAW\_RESULT

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| result | 结果 0为成功，1为失败 |

网关重启事件(onGatewayReboot)

void onGatewayReboot()

该回调方法表示网关重启，当进行固件升级完成后会触发此方法。回调参考ROBOT\_GATEWAY\_REBOOT

笔迹数据事件(onOriginalPacket)

void onOriginalPacket(const PEN\_INFO &penInfo)

该回调方法表示笔迹数据，当手写板书写时会触发此方法。回调参考ROBOT\_ORIGINAL\_PACKET

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| penInfo | 笔迹数据，参见PEN\_INFO |

手写板模式事件(onNodeMode)

void onNodeMode(eNodeMode mode)

该回调方法表示手写板当前，当手写板模式改变时会触发此方法。回调参考ROBOT\_NODE\_MODE

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| mode | 当前模式，参见eNodeMode |

设置RTC事件(onSetRtc)

void onSetRtc(int result)

该回调方法表示设置RTC，设置RTC后会触发此方法。回调参考ROBOT\_SET\_RTC

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| result | 结果 0为成功，1为失败 |

按键按下事件(onKeyPress)

void onKeyPress(int result)

该回调方法表示按键按下，手写板按键按下后会触发此方法。回调参考ROBOT\_KEY\_PRESS

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| result | 按键类型 |

页面显示事件(onShowPage)

void onShowPage(const PAGE\_INFO &pageInfo)

该回调方法表示页面显示，页面改变后会触发此方法，仅限于支持自动识别页码的手写板。回调参考ROBOT\_SHOW\_PAGE

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| pageInfo | 页码信息 参见PAGE\_INFO |

离线笔记坐标数据事件(onSyncPacket)

void onSyncPacket(const PEN\_INFO &penInfo)

该回调方法表示离线笔记坐标数据，开启同步离线笔记后会触发此方法。回调参考ROBOT\_SYNC\_PACKET

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| penInfo | 笔迹信息 参见PEN\_INFO |

离线笔记开始事件(onSyncBegin)

void onSyncBegin(int leftNum,int noteNum,const ST\_RTC\_INFO &rtcInfo)

该回调方法表示离线笔记坐标开始，开启同步离线笔记后会触发此方法。回调参考ROBOT\_SYNC\_TRANS\_BEGIN

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| leftNum | 剩余笔记数量 |
| noteNum | 笔记编号 |
| rtcInfo | 离线笔记时间，参见ST\_RTC\_INFO |

离线笔记数据结束事件(onSyncEnd)

void onSyncEnd(int result)

该回调方法表示离线笔记坐标数据，开启同步离线笔记后会触发此方法。回调参考ROBOT\_SYNC\_TRANS\_END

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| result | 结束结果，0为完成，1为还有数据 |

MassData模式页面显示事件(onMassShowPage)

void onMassShowPage(int index,const PAGE\_INFO &pageInfo)

该回调方法表示MassData模式页面显示，当页面改变时触发此方法，仅限于网关设备。回调参考ROBOT\_SHOW\_PAGE

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| index, | 子设备编号 |
| pageInfo | 页码信息 参见PAGE\_INFO |

抢答事件(onVoteAnswer)

void onVoteAnswer(int index,int answer)

该回调方法表示抢答结果，当抢答结束时触发此方法，仅限于网关设备。回调参考ROBOT\_VOTE\_ANSWER

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| index, | 子设备编号 |
| answer | 答案 |

X8Mac地址事件(onX8Mac)

void onX8Mac(uint8\_t \*mac)

该回调方法表示抢答结果，当抢答结束时触发此方法，仅限于X8设备。回调参考ROBOT\_GET\_X8\_MAC

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| mac, | Mac地址 |

Dongle状态事件(onDongleStatus)

void onDongleStatus(eDongleStatus status,int mode)

该回调方法表示Dongle状态，当Dongle状态改变时触发此方法，仅限于Dongle设备。回调参考ROBOT\_DONGLE\_STATUS

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| status | Dongle状态 参见eDongleStatus |
| mode | 模式 |

Dongle版本事件(onDongleVersion)

void onDongleVersion(const ST\_VERSION &version)

该回调方法表示Dongle版本，当查询Dongle版本时触发此方法，仅限于Dongle设备。回调参考ROBOT\_DONGLE\_VERSION

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| version | Dongle版本 参见ST\_VERSION |

Dongle扫描结果事件(onDongleScanRes)

void onDongleScanRes(const ST\_BLE\_DEVICE &device)

该回调方法表示Dongle扫描结果，当扫描到蓝牙设备时触发此方法，仅限于Dongle设备。回调参考ROBOT\_DONGLE\_SCAN\_RES

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| device | 蓝牙设备信息，参见ST\_BLE\_DEVICE |

蓝牙设备版本事件(onSlaveVersion)

void onSlaveVersion(eDeviceType type,const ST\_VERSION &version)

该回调方法表示蓝牙设备版本，当Dongle查询蓝牙设备版本时触发此方法，仅限于Dongle设备。回调参考ROBOT\_SLAVE\_VERSION

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| type | 蓝牙设备类型 参见eDeviceType |
| version | 蓝牙设备版本ST\_VERSION |

蓝牙设备状态事件(onSlaveStatus)

void onSlaveStatus(const NODE\_STATUS &status)

该回调方法表示当前蓝牙设备状态，当蓝牙设备状态改变时会触发此方法，仅限于Dongle设备。回调参考ROBOT\_SLAVE\_STATUS

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| status | 手写板状态，参见NODE\_STATUS |

设置蓝牙设备名称事件(onSetName)

void onSetName(int result)

该回调方法表示设置蓝牙设备名称结果，当设置蓝牙设备名称时会触发此方法，仅限于Dongle设备。回调参考ROBOT\_SET\_NAME

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| result | 结果 0为成功，1为失败 |

蓝牙设备错误事件(onSlaveError)

void onSlaveError(eSlaveError error)

该回调方法表示蓝牙设备错误，当设置蓝牙设备出现异常时会触发此方法，仅限于Dongle设备。回调参考ROBOT\_SLAVE\_ERROR

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| error | 错误类型 参见eSlaveError |

模组版本事件(onModuleVersion)

void onModuleVersion(const ST\_MODULE\_VERSION &version)

该回调方法表示模组版本，当查询模组版本时会触发此方法。回调参考ROBOT\_MODULE\_VERSION

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| version | 版本号 参见ST\_MODULE\_VERSION |

进入模组校准事件(onAjdustMode)

void onAjdustMode()

该回调方法表示进入模组，当模组进入校准模式时会触发此方法。回调参考ROBOT\_ENTER\_ADJUST\_MODE

模组校准结果事件(onAjdustResult)

void onAjdustResult(int result)

该回调方法表示模组校准结果，当模组校准完成后会触发此方法。回调参考ROBOT\_MODULE\_ADJUST\_RESULT

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| result | 结果 0为成功，1为失败 |

笔记优化输出事件(onOut)

void onOut(float x,float y,float width,float speed,int status)

该回调方法表示笔记优化输出，当开启笔记优化并传入原始坐标后会触发此方法。回调参考ROBOT\_OPTIMIZE\_PACKET

|  |  |
| --- | --- |
| 形参名称 | 描述 |
| x | x坐标 |
| y | y坐标 |
| width | 宽度 |
| speed | 速度 |
| status | 状态 0为结束，16为悬浮 17为正在书写 |