文档编号：AN2039

上海东软载波微电子有限公司

用户手册

ES–USB-Lab

版权所有©上海东软载波微电子有限公司

修订历史

版本	修订日期	修改概要
V1.0	2021-12-8	初版

地　址：中国上海市徐汇区古美路 1515 号凤凰园 12 号楼 3 楼
E-mail：support@essemi.com
电　话：+86-21-60910333
传　真：+86-21-60914991
网　址：http://www.essemi.com/

版权所有©
上海东软载波微电子有限公司
本资料内容为上海东软载波微电子有限公司在现有数据资料基础上慎重且力求准确无误编制而成，本资料中所记载的实例以正确的使用方法和标准操作为前提，使用方在应用该等实例时请充分考虑外部诸条件，上海东软载波微电子有限公司不担保或确认该等实例在使用方的适用性、适当性或完整性，上海东软载波微电子有限公司亦不对使用方因使用本资料所有内容而可能或已经带来的风险或后果承担任何法律责任。基于使本资料的内容更加完善等原因，上海东软载波微电子有限公司保留未经预告的修改权。使用方如需获得最新的产品信息，请随时用上述联系方式与上海东软载波微电子有限公司联系。

目 录

内容目录

第 1 章 简介 5
1. 1 概述 5
第 2 章 HID 数据收发 6
2. 1 功能说明 6
第 3 章 HID BOOT 7
3. 1 功能说明 7
3. 2 HID BOOT 通信协议 7
3. 2. 1 基本格式 8
3. 2. 2 请求命令格式 8
3. 2. 3 响应命令格式 8
3. 2. 4 数据格式 8
3. 2. 5 命令集 8
3. 2. 5. 1 写数据命令 8
3. 2. 5. 2 发送数据命令 9
3. 2. 6 命令返回代码 9
第 4 章 Bulk 数据收发 10
4. 1 功能说明 10
附录 1 Dll 函数说明 11
附录 1. 1 HID 设备 dll 函数说明 11
附录 1. 1. 1 查找设备函数 11
附录 1. 1. 2 获取设备信息函数 11
附录 1. 1. 3 获取设备名称函数 11
附录 1. 1. 4 获得设备的 vid 函数 11
附录 1. 1. 5 获得设备的 pid 函数 12
附录 1. 1. 6 打开设备函数 12
附录 1. 1. 7 关闭设备函数 12
附录 1. 1. 8 发送数据函数 12
附录 1. 1. 9 接收数据函数 13
附录 1. 1. 10 得到实际写入设备的数据长度函数 13
附录 1. 1. 11 重置写入设备的数据长度函数 13
附录 1. 1. 12 16 进制字符串转 10 进制数函数 13
附录 1. 1. 13 十进制整数转为字符串函数 13
附录 1. 1. 14 获取设备的输入报告长度函数 13
附录 1. 1. 15 获取设备的输出报告长度函数 14
附录 1. 1. 16 获取在读取数据时 ascii 为 0 的数据位置 14
附录 1. 1. 17 获取实际读取到的数据长度函数 14
附录 1. 1. 18 重置实际读取到的数据长度函数 14
附录 1. 1. 19 获取读取到的数据函数 14
附录 1. 2 BULK 设备 dll 函数说明 15
附录 1. 2. 1 设置查找设备使用的 GUID 15
附录 1. 2. 2 查找设备函数 15

版权所有©上海东软载波微电子有限公司 http://www.essemi.com

附录 1. 2. 3 获取设备信息函数 15
附录 1. 2. 4 获取设备名称函数 15
附录 1. 2. 5 获得设备的 vid 函数 16
附录 1. 2. 6 获取设备的 pid 函数 16
附录 1. 2. 7 打开设备函数 16
附录 1. 2. 8 关闭设备函数 16
附录 1. 2. 9 发送数据函数 16
附录 1. 2. 10 接收数据函数 17
附录 1. 2. 11 获取实际写入设备的数据长度函数 17
附录 1. 2. 12 重置写入设备的数据长度 17
附录 1. 2. 13 获取实际读取到的数据长度函数 17
附录 1. 2. 14 转换字符数组格式函数 17
附录 1. 2. 15 获取待发送的字符数组的长度函数 18
附录 1. 2. 16 获取设备的输出端点函数 18
附录 1. 2. 17 获取设备的输入端点函数 18
附录 1. 2. 18 获取在读取数据时 ascii 为 0 的数据位置 18
附录 1. 3 HID_BOOT 设备 dll 函数说明 18
附录 1. 3. 1 解析 bin 文件函数 18
附录 1. 3. 2 获取累加和函数 19

版权所有©上海东软载波微电子有限公司 http://www.essemi.com

第1章 简介

1.1 概述

ES-USB-Lab 上位机软件提供了用户自定义 HID 和 Bulk 设备的实验演示环境。分别支持 HID 数据收发、HID Bootloader、Bulk 数据收发。ES-USB-Lab 除了可视化的 PC 端软件界面，还提供了一套 DLL 动态链接库，用户可用来开发自己专属的 USB 设备上位机软件（见附录）。

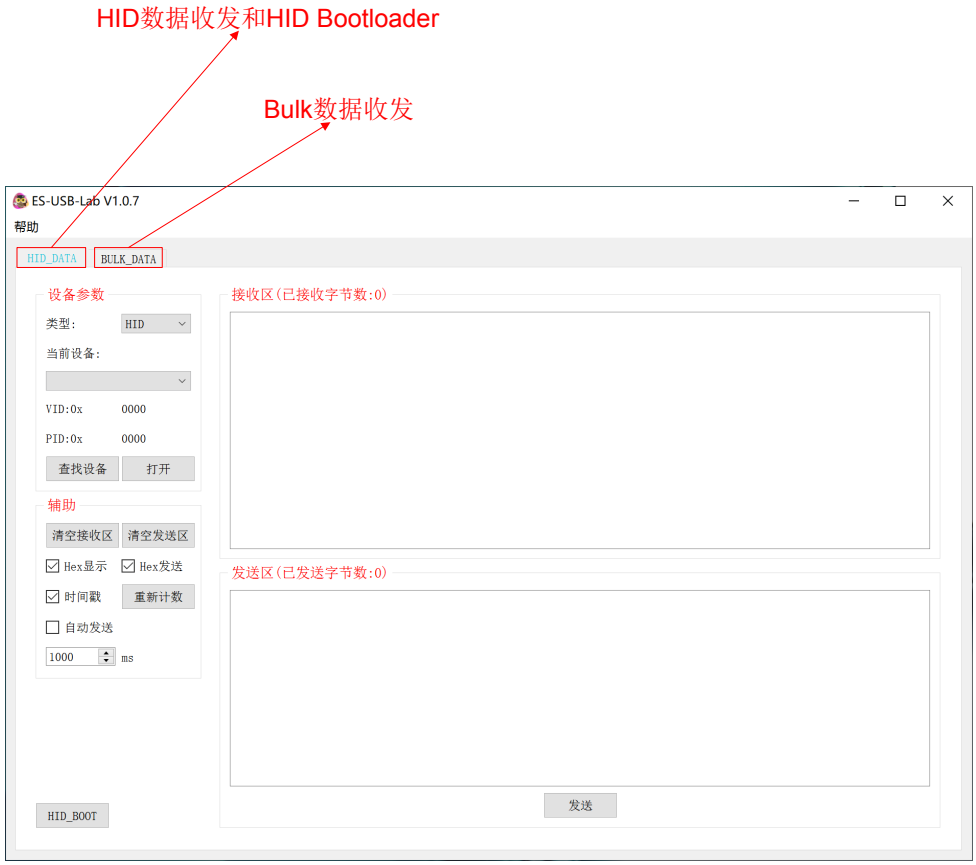


图 1-1 ES-USB-Lab 软件界面

版权所有©上海东软载波微电子有限公司 http://www.essemi.com

第2章　HID数据收发

2. 1　功能说明

选择【HID_DATA】页面，即可使用 HID 数据收发功能。发送数据和接收数据均按照端点描述符的包大小进行收发，数据长度如果不足包大小将补零，数据长度如果超过包大小将进行分帧发送和接收，分帧发送的时间间隔由界面设置。

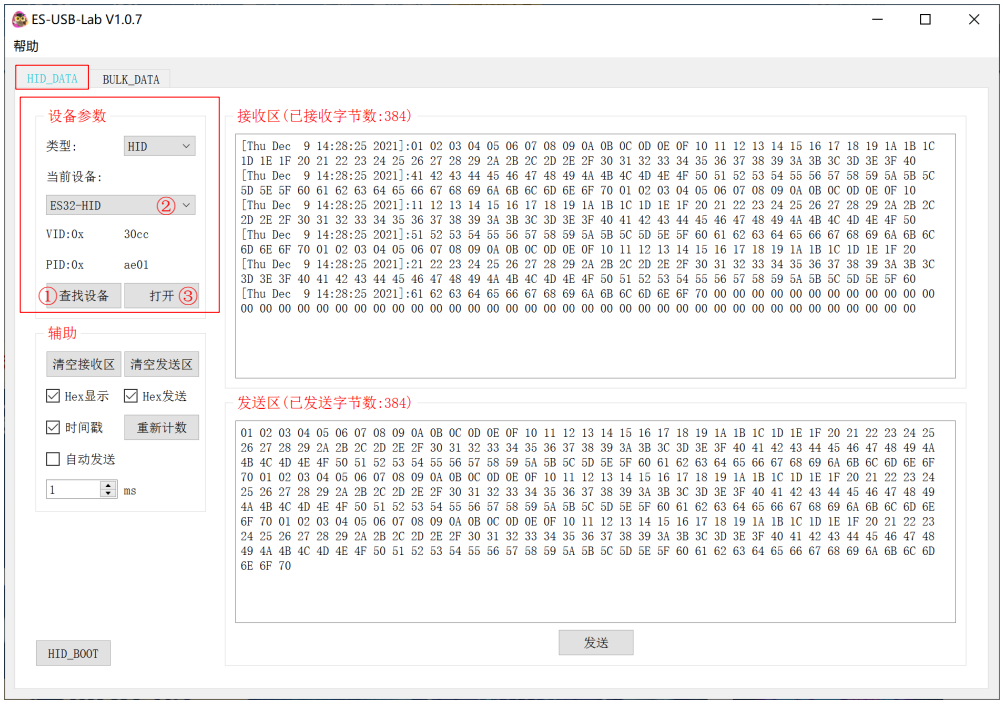


图 2-1　HID_DATA 软件界面

- 参数配置：首先点击【查找设备】，软件会识别当前电脑已经枚举成功的所有 HID 设备；然后在【当前设备】下选择当前需要使用的设备（这里显示的是接口字符串描述符）；最后点击【打开】即可进行数据收发。
- 数据收发：发送数据和接收数据均按照端点描述符的包大小进行收发，数据长度如果不足包大小将补零，数据长度如果超过包大小将进行分帧发送和接收，分帧发送的时间间隔由界面上的【ms】值设置。
- 辅助功能：支持【清空接收区】将接收区窗口清空；支持【清空发送区】将发送区窗口清空；支持【Hex 显示】切换接收区的 HEX 码显示和 ASCII 码显示；支持【Hex 发送】切换发送区的 HEX 码发送和 ASCII 码格式发送；支持【时间戳】控制是否显示每一帧数据的接收时间；支持【重新计数】清零接收区和发送区的收发字节数统计；支持【自动发送】可以无限循环发送数据，发送数据的时间间隔由用户自行填入【ms】值，同时【ms】值还控制分帧发送的时间间隔。

版权所有©上海东软载波微电子有限公司　http://www.essemi.com

第3章 HID BOOT

3. 1 功能说明

对 HID 设备完成【设备参数】配置后，点击【HID_BOOT】按钮即可进入 HID BOOT 功能界面

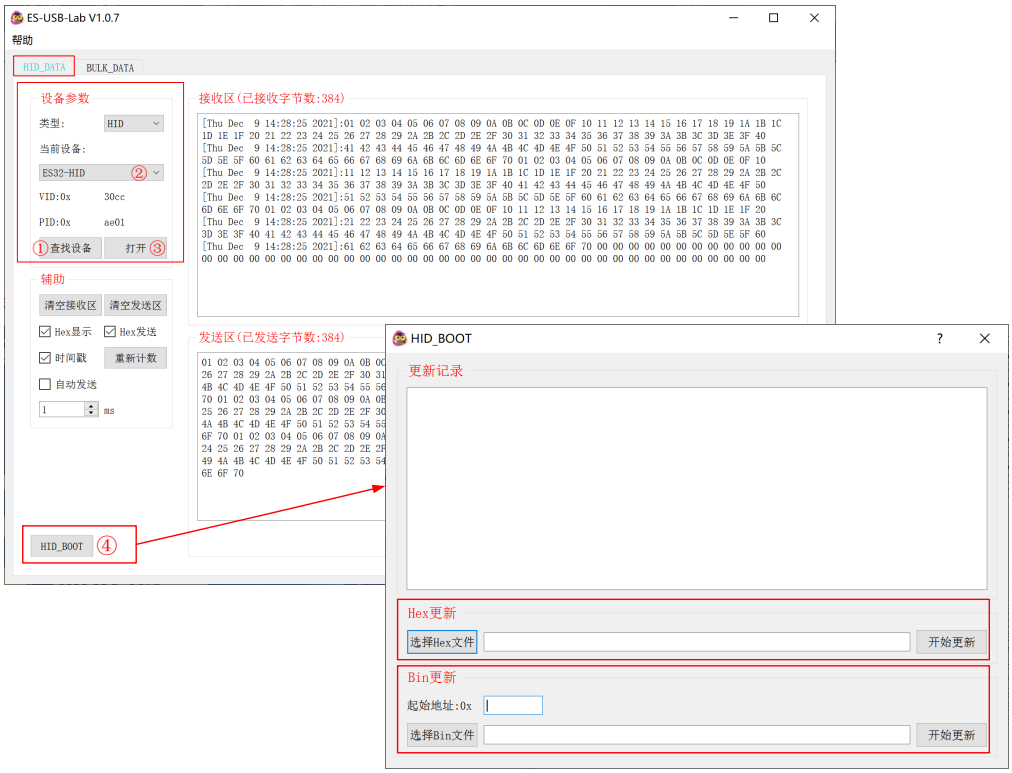


图 3-1 HID_BOOT 软件界面

- Hex 更新：【选择 Hex 文件】后点击【开始更新】，则开始发送 Hex 文件。
- Bin 更新：指定 Bin 文件的【起始地址】，【选择 Bin 文件】后点击【开始更新】，则开始发送 Bin 文件。

3. 2 HID BOOT通信协议

HID BOOT 下发 Hex 或 Bin 文件遵守如下图所示的主从协议，MCU 为 USB 设备，MCU 接收 USB 主机的请求报文后，再将响应报文回复给 USB 从机。

版权所有©上海东软载波微电子有限公司 http://www.essemi.com

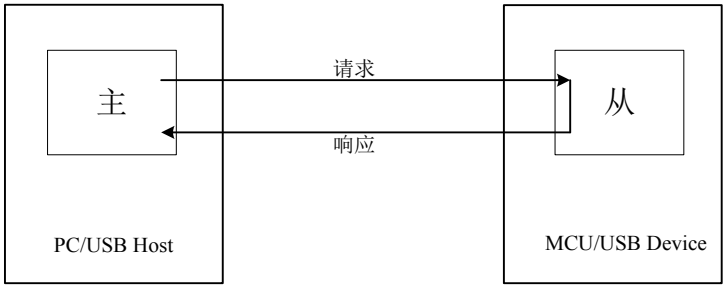


图 3-2 HID_BOOT 主从通信示意图

3.2.1 基本格式

所有命令都以单个 ASCII 字符串形式发送。命令以“？”或“！”开始，“？”表示请求命令，“！”表示响应命令。“？”或“！”紧接着的一个 ASCII 码为命令码。一条命令中的的多个参数码或返回码用空格间隔。参数码和返回码均用 ASCII 字符串表示，数据是以十六进制编码格式表示。所有命令响应都是以<CR><LF>结束的，<CR><LF>后跟随命令帧校验码（校验码为 1byte，是帧头“？”或“！”到<LF>的累加和）。如果为数据命令，后面还将跟随“HEX 数据+累加和”（累加和为 1byte，是全部 HEX 数据的累加和）。多余的 <CR> 和 <LF> 将被忽略。

3.2.2 请求命令格式

“？命令码□参数码 0□参数码 1□...□参数码 n<CR><LF>**累加和**”“HEX 数据+累加和”（数据只适用于数据命令）。

3.2.3 响应命令格式

“！命令码 □返回码 0□返回码 1□...□返回码 n<CR><LF>**累加和**”“HEX 数据+累加和”（数据只适用于数据命令）。

3.2.4 数据格式

数据流采用 HEX 编码格式。每次传送的长度都不应超过 64 个字节，接收器应当将该校验和与接收字节的校验和进行比较。如果校验和匹配，接收器响应“!OK<CR><LF>**累加和**”来继续下一次发送。如果校验和不匹配，接收器响应“!RESEND<CR><LF>**累加和**”。作为响应，发送器应当重新发送那些字节。

3.2.5 命令集

下面的命令是主机发送的命令。每个命令都支持具体的返回代码。当从机接收到未定义命令时，从机会发送返回代码 INVALID_COMMAND。命令和返回代码为 ASCII 格式。只有当接收到的命令执行完毕时，从机才会发送 OK，这时主机才能发送新的命令。但“”、“”、“”和“”命令除外。

3.2.5.1 写数据命令

◆ 命令格式

命令	W
输入参数	参数 0（起始写入地址）□参数 1（数据总长度）
返回代码	OK|PARAM_ERROR
描述	固件更新时使用，写命令发送地址和长度后再通过发送数据命

版权所有©上海东软载波微电子有限公司 http://www.essemi.com

	令进行数据传送

3. 2. 5. 2 发送数据命令

◆ 命令格式

命令	D
输入参数	参数 0: 序号/地址(ASCII) □参数_1: 长度(ASCII)<CR><LF>累加和(1byte HEX)数据(nbyte HEX) 累加和(1byte HEX)
返回代码	参数 0: 序号/地址(ASCII) □参数_1: OK/ERR/OVER
描述	发送数据命令 参数 0: 目前保持为 0，

3. 2. 6 命令返回代码

◆ 返回代码总览

返回字符串	描述
OK	执行成功
CMD_INVALID	无效的命令
PARAM_ERROR	参数错误
COUNT_ERROR	字节数错误
ADDR_ERROR	地址错误
BUSY	硬件忙
*	有后续帧
CONFIG_FAIL	配置失败
CHECK_ERROR	数据校验错误
SLAVE_NACK	从机无应答
SLAVE_BUSY	从机忙

版权所有©上海东软载波微电子有限公司 http://www.essemi.com

第4章 Bulk 数据收发

4.1 功能说明

选择【BULK_DATA】页面，即可使用 Bulk 数据收发功能。

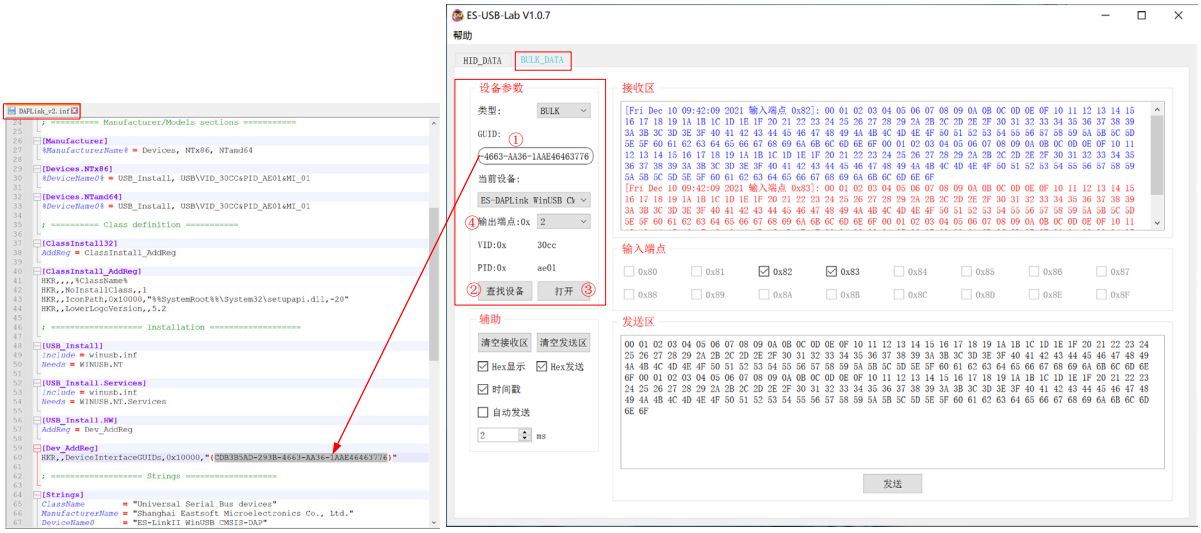


图 4-1 BULK_DATA 软件界面

- 参数配置：首先在【GUID】窗口填入需要连接设备的 GUID，GUID 可以在设备驱动 inf 文件中获取；然后点击【查找设备】，在【当前设备】窗口就会识别出当前电脑已经枚举成功的对应 Bulk 设备（这里显示的是接口字符串描述符）；点击【打开】即可进行数据收发；如果当前 Bulk 设备的接口包含多个输出端点，那么还要在【输出端点】下选择使用哪个端点发送数据；
- 数据收发：发送数据和接收数据大小不可超过当前设备的端口缓冲区大小。如果当前 Bulk 设备的接口包含多个输入端点，可以在【输出端点】下选择接受哪个端点的数据；
- 辅助功能：支持【清空接收区】将接收区窗口清空；支持【清空发送区】将发送区窗口清空；支持【Hex 显示】切换接收区的 HEX 码显示和 ASCII 码显示；支持【Hex 发送】切换发送区的 HEX 码发送和 ASCII 码格式发送；支持【时间戳】控制是否显示每一帧数据的接收时间；支持【自动发送】可以无限循环发送数据，发送数据的时间间隔由用户自行填入【ms】值。

版权所有©上海东软载波微电子有限公司 http://www.essemi.com

附录1 DII函数说明

附录1. 1 HID 设备 dll 函数说明

HIDDLL.dll 主要包括查找 HID 设备、打开(关闭)HID 设备、发送数据与接收数据等功能,便于用户进行二次开发,所在路径与软件安装路径为同一目录,HID 例程也在该目录下.

附录1. 1. 1 查找设备函数

类型	描述
函数原型	int GetDeviceList()
功能描述	能够获得电脑上所连接的 HID 设备的个数，能够获得设备的名称与设备相关信息，并将其分别保存在字符数组中
输入参数	无
返回值	Value: 电脑上所连接的 HID 设备的个数
使用示例	

附录1. 1. 2 获取设备信息函数

类型	描述
函数原型	char* GetDescrip()
功能描述	能够获得包含查找到的设备信息的字符数组，设备信息主要包括 vid、pid 等，设备信息以回车符(\n)分割
输入参数	无
返回值	buffer:\\?\hid#vid_0483&pid_5750#7&2b048c5c&0&0000#{4d1e55b2-f16f-11cf-88cb-001111000030} \\?\hid#vid_093a&pid_2510#7&36427222&0&0000#{4d1e55b2-f16f-11cf-88cb-001111000030}
使用示例	

附录1. 1. 3 获取设备名称函数

类型	描述
函数原型	char* GetName()
功能描述	能够获得包含查找到的设备名称的字符数组，设备名以回车符(\n)分割
输入参数	无
返回值	buffer: ES-Bridge HID-compliant mouse
使用示例	

附录1. 1. 4 获得设备的 vid 函数

类型	描述
函数原型	char* GetVid(char* mystr)
功能描述	能够获得设备的 vid
输入参数	mystr:包含 vid 的数组
返回值	buffer: 0483

版权所有©上海东软载波微电子有限公司 http://www.essemi.com

使用示例	GetVid(str),str 为 GetDescrip()函数返回的包含设备信息的数组 ('\\?\hid#vid_0483&pid_5750#7&2b048c5c&0&0000#{4d1e55b2-f16f-11cf-88cb-001111000030}')

附录1. 1. 5 获得设备的 pid 函数

类型	描述
函数原型	char* GetPid(char* mystr)
功能描述	能够获得设备的 pid
输入参数	mystr:包含 pid 的数组
返回值	buffer: 5750
使用示例	GetPid(str),str 为 GetDescrip()函数返回的包含设备信息的数组 ('\\?\hid#vid_0483&pid_5750#7&2b048c5c&0&0000#{4d1e55b2-f16f-11cf-88cb-001111000030}')

附录1. 1. 6 打开设备函数

类型	描述
函数原型	BOOL OpenDevice(char* vid, char* pid)
功能描述	打开设备，是设备与上位机能够通讯
输入参数	vid:通过 GetVid(char* mystr)函数得到 pid:通过 GetPid(char* mystr)函数得到
返回值	成功：True 失败：False
使用示例	OpenDevice(0483,5750)，vid 与 pip 为 16 进制数组

附录1. 1. 7 关闭设备函数

类型	描述
函数原型	BOOL CloseDevice()
功能描述	关闭设备，中断设备与上位机之间的通讯
输入参数	无
返回值	成功：True 失败：False
使用示例	

附录1. 1. 8 发送数据函数

类型	描述
函数原型	BOOL SendData(char* str, int length, char* data, int num)
功能描述	发送命令与数据
输入参数	str:待发送的命令 length:待发送的命令的长度 data:待发送的数据，当待发送数据为空时,此参数为空数组,长度为 0 num:待发送的数据长度
返回值	成功：True 失败：False
使用示例	SendData(str, 10, data, 20)

版权所有©上海东软载波微电子有限公司 http://www.essemi.com

附录1. 1. 9 接收数据函数

类型	描述
函数原型	BOOL ReceiveData()
功能描述	接收从设备返回的数据
输入参数	无
返回值	成功：True 失败：False
使用示例	

附录1. 1. 10 得到实际写入设备的数据长度函数

类型	描述
函数原型	DWORD GetWriteLength()
功能描述	得到实际写入设备的数据长度
输入参数	无
返回值	value：value 不为 0 时，代表有数据写入，可以进行读操作
使用示例	

附录1. 1. 11 重置写入设备的数据长度函数

类型	描述
函数原型	void SetWriteLength()
功能描述	将写入设备的数据长度重置为 0，避免重复读取
输入参数	无
返回值	无
使用示例	

附录1. 1. 12 16 进制字符串转 10 进制数函数

类型	描述
函数原型	int ChangeNum(string str, int length)
功能描述	将 16 进制字符串转为十进制数
输入参数	str:待转换的 16 进制字符串 length: 字符串长度
返回值	value:转换后得到的数值
使用示例	ChangeNum('FF', int 2)

附录1. 1. 13 十进制整数转为字符串函数

类型	描述
函数原型	char* intToA(long long n, int radix)
功能描述	将十进制整数转为字符串
输入参数	n:待转换的整数 radix: 需要转换的进制，16 表示转换为 16 进制数，8 表示转换为 8 进制数
返回值	buffer:转换后得到的数组
使用示例	intToA(12345,16)

附录1. 1. 14 获取设备的输入报告长度函数

类型	描述

版权所有©上海东软载波微电子有限公司 http://www.essemi.com

函数原型	DWORD GetCapInReLength()
功能描述	获取设备的输入报告长度
输入参数	
返回值	无
使用示例	

附录1. 1. 15 获取设备的输出报告长度函数

类型	描述
函数原型	DWORD GetCapOutReLength()
功能描述	获取设备的输入报告长度
输入参数	
返回值	无
使用示例	

附录1. 1. 16 获取在读取数据时 ascii 为 0 的数据位置

类型	描述
函数原型	char* GetErrorValue()
功能描述	当读取到的数据的 ascii 为 0 时,在 python 中获取到的数据会截至在该数据位置,为了保证能够接受完全,将 ascii 为 0 的数据的位置保存到字符串中,每个位置以"\n"分割. 其余不为 0 的数据正常输出,在 python 中作处理时将两个字符串按照位置组合成一个完整的字符串,再作其它用途.
输入参数	
返回值	数据中 ascii 为 0 的位置
使用示例	

附录1. 1. 17 获取实际读取到的数据长度函数

类型	描述
函数原型	DWORD GetReadLength()
功能描述	获取实际读取到的数据长度, 用于将读取到的数据实时添加到 RichEdit 框中
输入参数	无
返回值	value: 实际读取到的数据长度
使用示例	

附录1. 1. 18 重置实际读取到的数据长度函数

类型	描述
函数原型	void SetReadLength()
功能描述	将实际读取到的数据长度重置为 0，避免重复添加
输入参数	无
返回值	无
使用示例	

附录1. 1. 19 获取读取到的数据函数

类型	描述
函数原型	char* GetRecData()
功能描述	得到读取到的 ascii 不为 0 的数据,与 GetErrorValue()函数结合起来一同使用

版权所有©上海东软载波微电子有限公司 http://www.essemi.com

输入参数	无
返回值	buffer:实际读取到的 ascii 不为 0 的数据
使用示例	

附录1. 2 BULK 设备 dll 函数说明

BULKDLL.dll 主要包括查找 BULK 设备、打开(关闭)BULK 设备、发送数据与接收数据等功能,便于用户进行二次开发,所在路径与软件安装路径为同一目录,BULK 例程也在该目录下.

附录1. 2. 1 设置查找设备使用的 GUID

类型	描述
函数原型	void GetGuid(unsigned long data1,unsigned short data2,unsigned short data3,char* data4)
功能描述	在 python 上位机中输入设备的 GUID, 转换为 dll 中 GUID 类型的变量
输入参数	1. unsigned long data1：data1 为 GUID 中的前 4 字节的 10 进制数进制数 2. unsigned short data2: data2 为 GUID 中的第 5-6 字节的 10 进制数进制数 3. unsigned short data3: data3 为 GUID 中的第 7-8 字节的 10 进制数进制数 4. char* data4: data4 为剩余字节的字符数组
返回值	无
使用示例	GUID:{F70242C7-FB25-443B-9E7E-A4260F373982}, F70242C7 的 10 进制数为 data1, FB25 的 10 进制数为 data2, 443B 的 10 进制数为 data3, 9E、7E、A4、26、0F、37、39、82 对应的字符组成字符数组为 data4

附录1. 2. 2 查找设备函数

类型	描述
函数原型	int GetDeviceList1()
功能描述	能够获得电脑上所连接的 BULK 设备的个数，能够获得设备的名称与设备相关信息，并将其分别保存在字符数组中
输入参数	无
返回值	Value: 电脑上所连接的 BULK 设备的个数
使用示例	

附录1. 2. 3 获取设备信息函数

类型	描述
函数原型	char* GetDescrip()
功能描述	能够获得包含查找到的设备信息的字符数组，设备信息主要包括 vid、pid 等，设备信息以回车符(\n)分割
输入参数	无
返回值	buffer:
使用示例	

附录1. 2. 4 获取设备名称函数

类型	描述
函数原型	char* GetName()
功能描述	能够获得包含查找到的设备名称的字符数组，设备名以回车符(\n)分割

版权所有©上海东软载波微电子有限公司 http://www.essemi.com

输入参数	无
返回值	buffer:
使用示例	

附录1. 2. 5 获得设备的 vid 函数

类型	描述
函数原型	char* GetVid1(char* mystr)
功能描述	能够获得设备的 vid
输入参数	mystr:包含 vid 的数组
返回值	buffer:
使用示例	GetVid(str), str 为 GetDescrip()函数返回的包含设备信息的数组

附录1. 2. 6 获取设备的 pid 函数

类型	描述
函数原型	char* GetPid1(char* mystr)
功能描述	能够获得设备的 pid
输入参数	mystr:包含 pid 的数组
返回值	buffer:
使用示例	GetPid1(str), str 为 GetDescrip()函数返回的包含设备信息的数组

附录1. 2. 7 打开设备函数

类型	描述
函数原型	BOOL OpenDevice(char* vid, char* pid)
功能描述	打开设备，是设备与上位机能够通讯
输入参数	vid:通过 GetVid1(char* mystr)函数得到 pid:通过 GetPid1(char* mystr)函数得到
返回值	成功：True 失败：False
使用示例	OpenDevice(vid,pid)， vid 与 pip 为 16 进制数组

附录1. 2. 8 关闭设备函数

类型	描述
函数原型	BOOL CloseDevice1()
功能描述	关闭设备，中断设备与上位机之间的通讯
输入参数	无
返回值	成功：True 失败：False
使用示例	

附录1. 2. 9 发送数据函数

类型	描述
函数原型	BOOL SendData1(char* data, int inport, int num)
功能描述	发送命令与数据
输入参数	data:待发送的数据 inport:输出端点 num:待发送的数据长度

版权所有©上海东软载波微电子有限公司 http://www.essemi.com

返回值	成功：True 失败：False
使用示例	SendData(data, 20)

附录1. 2. 10 接收数据函数

类型	描述
函数原型	char* ReceiveData1(char* recbuffer,int outport)
功能描述	接收从设备返回的数据
输入参数	recbuffer: 返回的命令保存在传入的参数 recbuffer 中 outport: 输入端点
返回值	成功：True 失败：False
使用示例	

附录1. 2. 11 获取实际写入设备的数据长度函数

类型	描述
函数原型	DWORD GetWriteLength1()
功能描述	获取实际写入设备的数据长度
输入参数	无
返回值	value：实际写入设备的数据长度，当 value 不为 0 时，代表有数据写入，可以进行读操作
使用示例	

附录1. 2. 12 重置写入设备的数据长度

类型	描述
函数原型	void SetWriteLength1()
功能描述	重置写入设备的数据长度，将写入设备的数据长度重置为 0，避免重复读取
输入参数	无
返回值	无
使用示例	

附录1. 2. 13 获取实际读取到的数据长度函数

类型	描述
函数原型	DWORD GetReadLength1()
功能描述	得到实际读取到的数据长度
输入参数	无
返回值	value:实际读取到的数据长度
使用示例	

附录1. 2. 14 转换字符数组格式函数

类型	描述
函数原型	char * GetHexNum(char* str, int num)
功能描述	将字符数组改为相应格式去发送，例如 123456->12 34 56->18 52 86。
输入参数	str:待转换的字符数组 num:数组长度
返回值	buffer:修改之后的字符数组

版权所有©上海东软载波微电子有限公司 http://www.essemi.com

使用示例	

附录1. 2. 15 获取待发送的字符数组的长度函数

类型	描述
函数原型	DWORD GetHexLength()
功能描述	获取待发送的字符数组的长度
输入参数	无
返回值	value：待发送的字符数组的长度
使用示例	

附录1. 2. 16 获取设备的输出端点函数

类型	描述
函数原型	char * GetInPort()
功能描述	获取设备的输出端点字符串数组
输入参数	无
返回值	buffer：包含输出端点的字符串数组,输出端点之间以"\n"分割
使用示例	

附录1. 2. 17 获取设备的输入端点函数

类型	描述
函数原型	char * GetOutPort()
功能描述	获取设备的输入端点字符串数组
输入参数	无
返回值	buffer：包含输入端点的字符串数组,输入端点之间以"\n"分割
使用示例	

附录1. 2. 18 获取在读取数据时 ascii 为 0 的数据位置

类型	描述
函数原型	char* GetErrorValue()
功能描述	当读取到的数据的 ascii 为 0 时,在 python 中获取到的数据会截至在该数据位置,为了保证能够接受完全,将 ascii 为 0 的数据的位置保存到字符串中,每个位置以"\n"分割. 其余不为 0 的数据正常输出,在 python 中作处理时将两个字符串按照位置组合成一个完整的字符串,再作其它用途.
输入参数	
返回值	数据中 ascii 为 0 的位置
使用示例	

附录1. 3　HID_BOOT 设备 dll 函数说明

HID_BOOT 模块调用的 dll 为 HIDDLL.dll,在调用 HID 模块中包含的函数外,还有以下函数.HID_BOOT 例程在软件安装路径下.

附录1. 3. 1　解析 bin 文件函数

类型	描述
函数原型	char* GetBintext(char* str)
功能描述	得到 bin 文件的内容
输入参数	str：bin 文件的实际路径

版权所有©上海东软载波微电子有限公司 http://www.essemi.com

返回值	buffer:bin 文件内容
使用示例	

附录1. 3. 2 获取累加和函数

类型	描述
函数原型	long long GetCheckValue(char* buffer, int num)
功能描述	获取数据的累加和
输入参数	buffer：包含数据的数组 num：数组的长度
返回值	value:10 进制整数，在进行校验时，将获取到的累加和转为 16 进制字符串，取其后八位。校验和采用小端模式，例如：校验和为 0x12345678 时，传入的参数应为 78 56 34 12
使用示例	

版权所有©上海东软载波微电子有限公司 http://www.essemi.com