**端点**：

EP0：控制端点，最大传输长度为64BYT

EP1：双向bulk传输，独立FIFO，最大传输64BYTE

EP2：双向ISO端点，独立FIFO，数据接收采用双FIFO机制，最大传输768BYTE

**USB拔插检查：**

WS300内置了一个USB检查模块。可以用检测USB线的插入和拔出，由于USB应用场景的复杂性，USB硬件检查模块不能满足所有要求，因此sdk保留了sof检查作为插入PC检测和一个定时器作为拔出检测。USB检测和初始化一般按照以下流程进行。

1、调用usbd\_sys\_module\_init初始化USB模块

2、调用usb\_detect\_init打开硬件检查模块

3、当检查到USB线插入时会进入on\_usb\_cable\_plus\_in并触发USB\_EVENT\_CABLE\_IN事件，收到该事件后，初始化USB描述符，端点等，务必打开usb\_sof\_isr\_en(TRUE)使能SOF中断，当接入主机后，就会收到sof中断，通过sof中断来判断USB接线是否是PC/手机还是火牛等无效设备

4、当我们要进行模式切换时，应该等到SOF信号确认后再进行模式切换，整个枚举过程会在后台自动完成，只要不关闭USB模块即可。

5、设备拔出检测是定时器来做的，理论上每个毫秒其sof帧号会加1，根据这个特性，我们判断一定时间内其帧号没有变，就说明已经拔出了。通过调用函数get\_sof\_frame来读取。

**程序结构：**

1. **枚举流程**

软件处理USB EP0请求处理流程如下，其中，用户层能够介入的处理如红色部分所示。其中1,2,5 处理模块可以通过**usbd\_register\_setup\_event**函数注册相关回调进行干涉。



1、EP0预处理。通过回调**OnEp0\_pre\_setup**实现，其返回值为true时表示无需继续传递，一个控制请求完成，0时表示继续传递给ROM函数处理。

2、标准描述符预处理。标准描述符预处理。通过回调**OnPreStandardSetupProcess**，其返回值为true时表示无需继续传递，一个控制请求完成，0时表示继续传递给ROM函数处理。

3、EP0响应后处理。通过回调**OnEp0\_setup\_processed**函数进行，相关参数可以通过变量setupReq等进行获取

4、扩展命令处理。该模块只是针对音频控制命令，当控制命令不属于以下命令（SET\_CUR，GET\_CUR,，GET\_MIN，GET\_MAX，SET\_RES，GET\_RES,）时会进入该流程，通过注册**registrer\_usb\_audio\_callback**进行注册，回调**ExternAudioCmd**，

5、扩展数据处理。当接受完成扩展命令的数据，会进入这个流程进行。通过回调**ExternAudioData**。

**二、音频处理**

WS300 SDK已经通过registrer\_usb\_audio\_callback注册相关的回调，实现了音频的基本功能。其各个回调函数说明如下：

1、**AudioSystemParamInit**。

pcmPacketSiz，需要返回最大的PCM接收长度

micPacketSize，需要返回最大的录音数据长度

以上两个参数的返回值会被用于设定FIFO的长度，当支持多个采样率时，使用最大数据量

2、InitAudio

ptrPcmBuf，返回已分配用于接收数据的buffer，并转化成uint32\_t类型

pcmBufLength，返回第一次接收数据的长度。

3、OnPrepareMicData

ptrMicBuf，返回录音数据的有效buffer，转化成uint32\_t

length，返回录音数据的长度

4、OnNewPcmReceiveBuff

newBuf：接收数据的buffer，

length： 接收数据的寄存器计数值，取数据长度时要除以4

5、GetCur

获取音量、静音状态或者采样率的回调

6、SetCur

设置音量，静音状态或者采样率的回调

7、GetMin

获取最小音量的逻辑值。用户需自行对应到实际音量

8、GetMax

获取最大音量的逻辑值。用户需自行对应到实际音量

9、GetRes

获取步进音量值。

10、SetRes

设置步进音量值。

11、OnAlternateSettingChange

当音频接口改变时会触发该事件。

12、ExternAudioCmd

扩展命令。用户可以定义一个音频命令，此回调表明收到了一组需要用户处理的命令

13、ExternAudioData

扩展数据。自定义或者非常用音频命令时需要处理数据时，由此回调处理

**三、音频描述符**

请参考usbd\_desc.c和usbd\_desc\_process.c。 目前已经定义了立体声和DOLBY音乐类型两种，需要修改其它参数，请另外参考USB AUDIO协议。

**四、字符串设置**

通过ROM变量COMP\_Desc中的相关回调来设置，具体参考代码usbd\_audio\_app.c。