此文档为笔记性质 ，随笔类。 主要记录在USB学习与处理问题过程中的重要知识点的随笔记录 。以供自己后期对该部分知识的回顾 。好记性不如乱笔头 。

1. USB学习的准备和使用的基础

在学习了解USB过程中 ， 参考资料除官网和网站搜索资料之外，可以选取一本好的入门资料来学习 ，在分析USB数据过程中可以使用BUSHOUND 软件来辅助跟踪 ，使用这款软件查看和分析通讯过程中的数据 ,我们对协议的了解会更加深刻 。

这里使用的资料为《圈圈教你玩USB》 【R】，工具为BUSHOUND【B】 。

二，USB【串行总线Universal Serial Bus】 概述与协议基础

2.1 ，USB 概念

2.2 ，USB 的特点

2.3 ，USB的 拓扑结构

2.4 ，USB电气特性

2.5 ，USB 的线缆，插头与插座

2.6 ，USB 的 插入检测机制

2.7 ，USB 的描述符及其之间的关系

2.8 ，USB设备的枚举过程

2.9 ，USB的包结构及传输过程

2．10，USB 的四种传输类型

2.1

2.2

2.3

2.4

2.5

2.6

2.7

2.8 USB设备的枚举过程

USB主设备对从设备识别的过程。

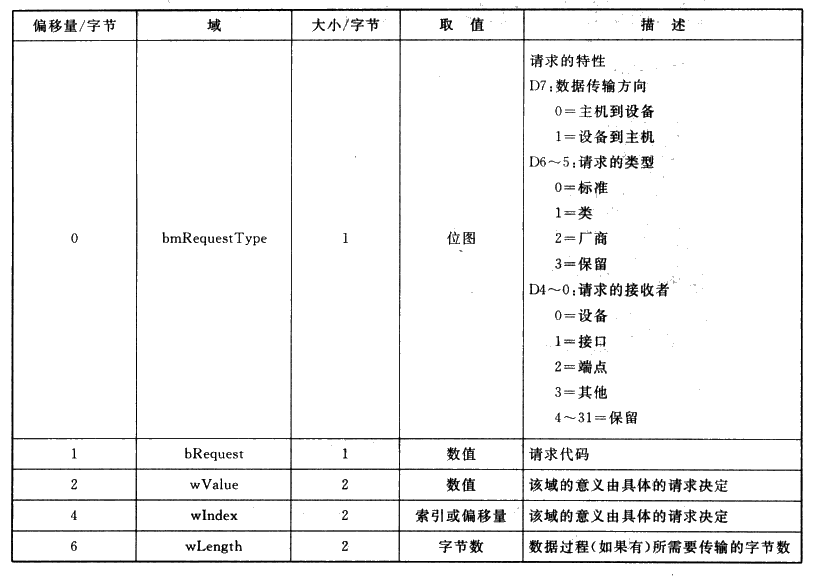
1. 为什么要枚举，枚举的作用？

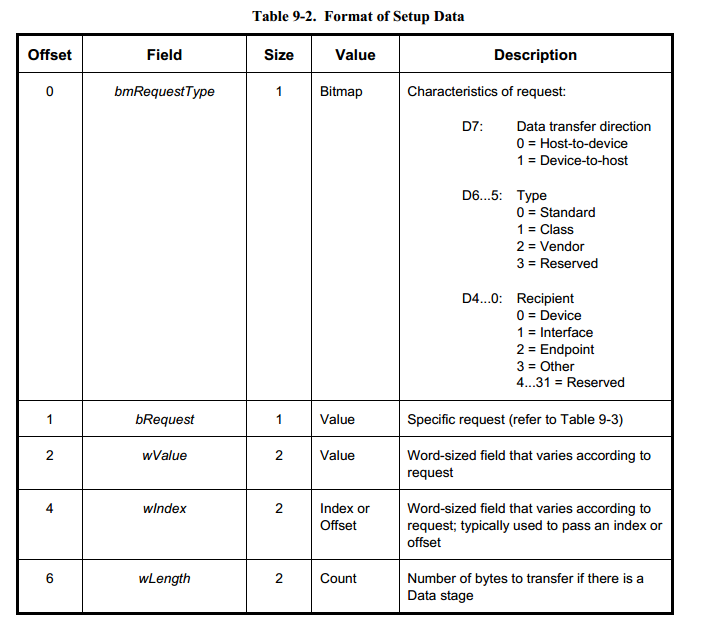
枚举就是主机从从设备读取各种描述符信息，从而主机根据这些信息来加载合适的驱动程序，从而知道设备是什么样的设备，如何进行通信。【调试USB设备，很重要的一点，只要枚举成功了，剩下的工作就不多了 –摘自【R】 】

USB 标准请求

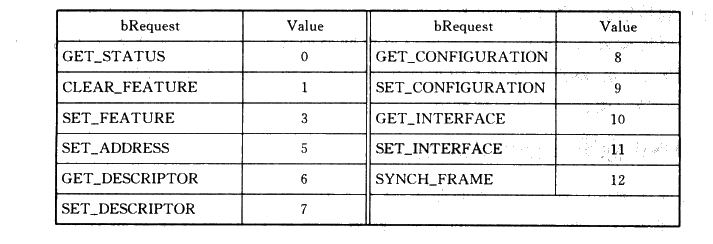
USB协议定义一个8字节的标准设备请求，主要用在设备枚举过程中。这8字节的数据是在控制传输的建立过程中默认通过控制端点0发出的 。在这8字节的数据中， 包含了数据过程所需要传输数据传输方向，长度以及数据类型等信息。正是由于8个字节的标准请求的原因，USB协议规定，端点0的最大包长度至少为8字节。也就是说，任何一个USB设备都能够（而且必须要）接收8字节的标准请求。

USB标准设备请求的结构

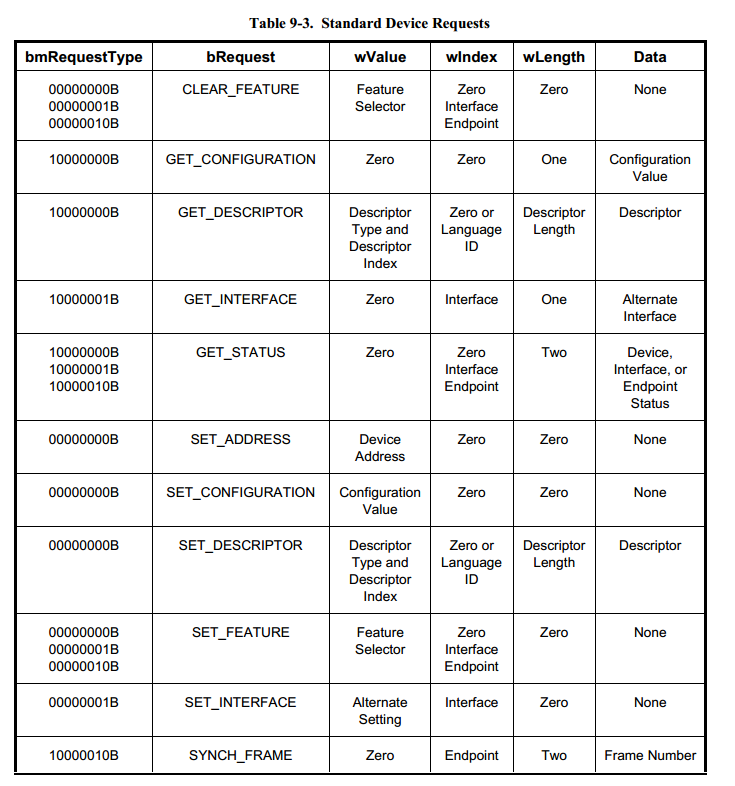




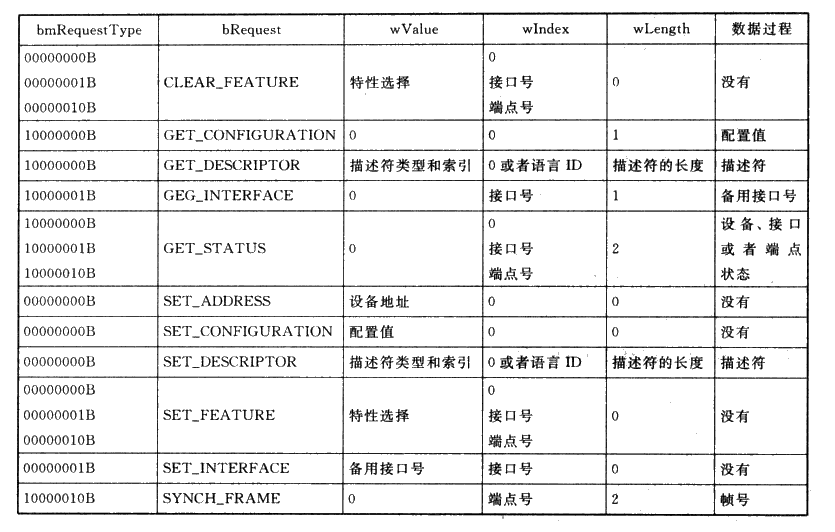
标准请求 ，即bmRequestType 的 BIT 5,6为00 。 下图为USB协议定义的11个标准请求（bRequst）,



下图为USB SPEC的截图 ：

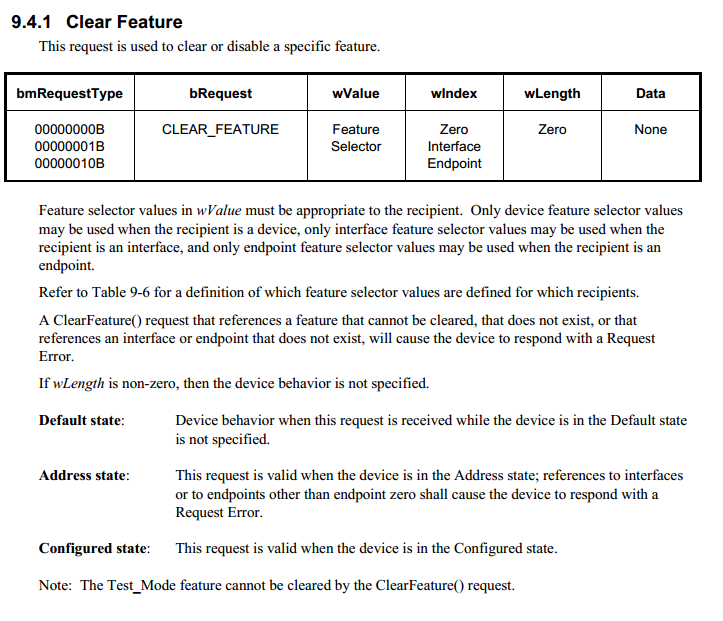


其中文解释为 ：

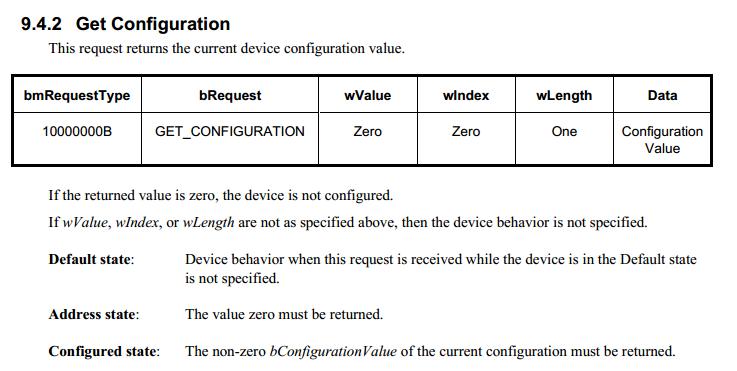


下面分别解释一下每一个标准请求 ：

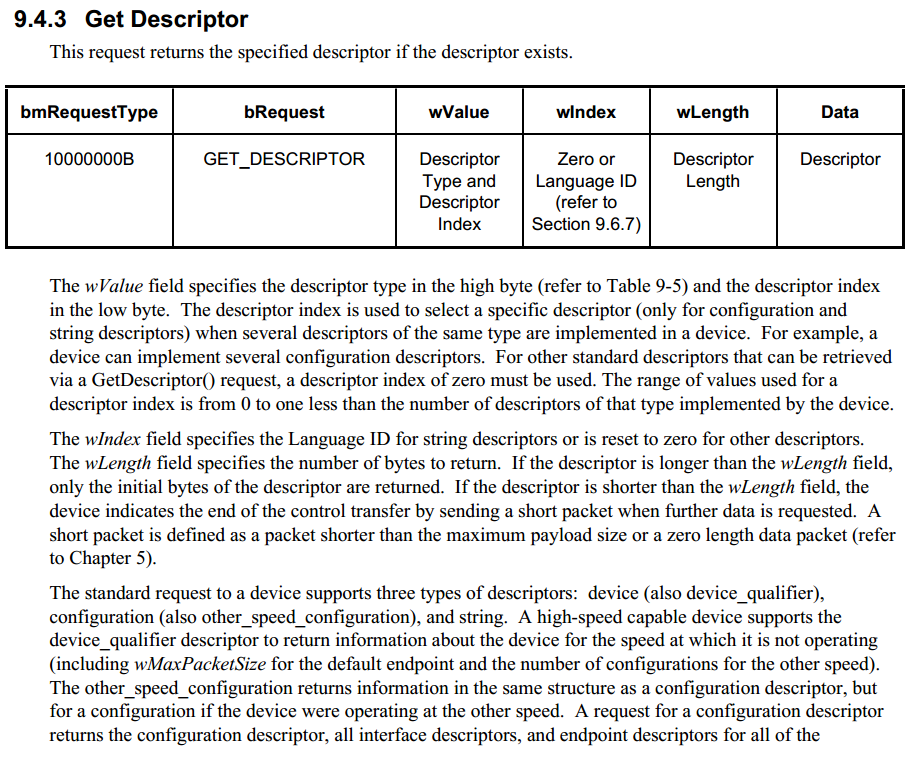
1. **Clear Featrue**

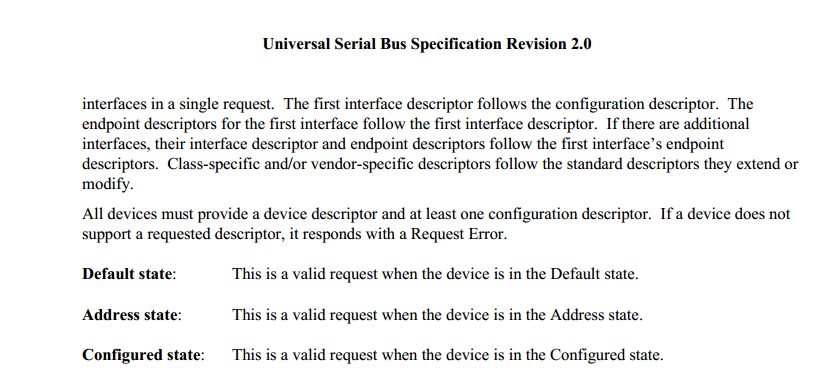


1. **Get Configuration**

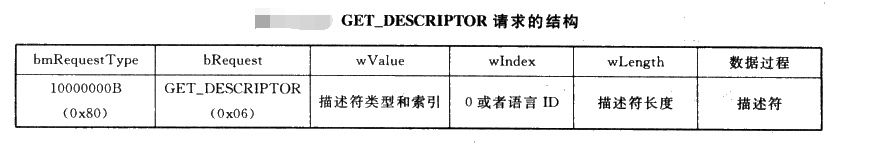
****

**3, Get Descriptor \*\*\*\*\*\*\***

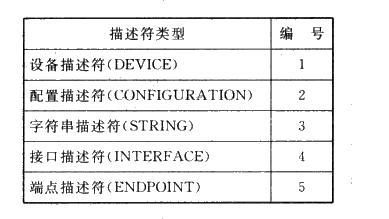
****

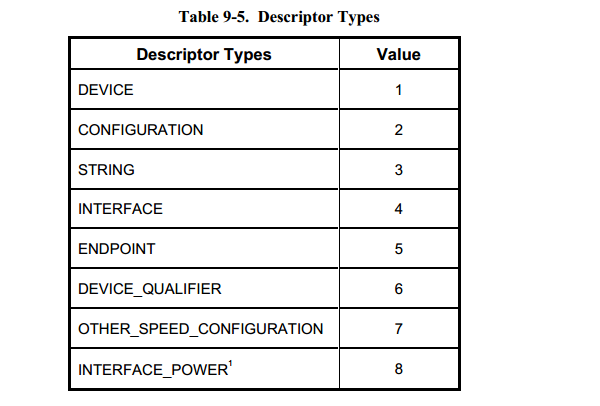
****

**GET\_DESCRIPTOR (获取描述符)请求是在枚举过程中用的最多的一个请求 。主机通过发送获取描述符请求读取设备的各种描述符 ，从而可以获取设备类型，端点情况等众多重要信息 。获取描述符的接受者只能是设备，从bmRequestType 的第7位可以看出，它是请求数据输入的 ，bRequest 值为0x06（GET\_DESCRIPTOR）.**

****

**wValue 域的第一个字节（低字节）表示的是索引号，用来选择同一种描述符（如字符串描述符和配置描述符）中具体的描述符。wValue 域的第二字节 ，表示描述符的类型编号。各种描述符类型号如下表 ：**

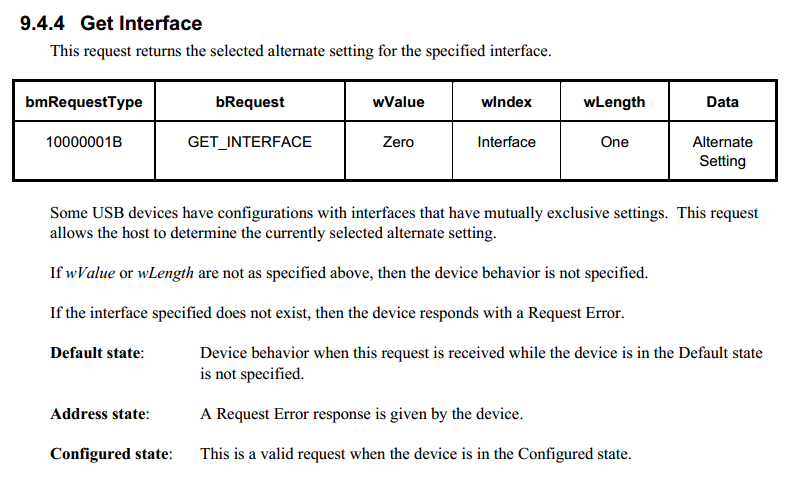
****

****

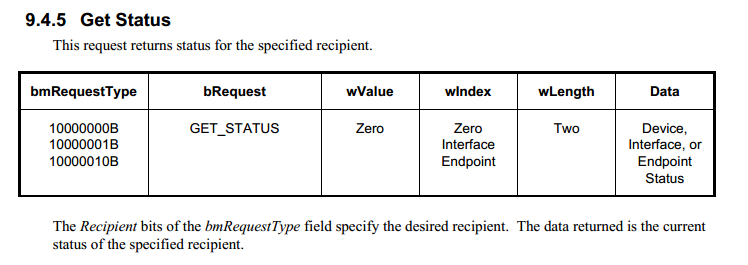
**wIndex 域只有在获取字符串描述符中有用，它表示字符串的语言ID号，获取字符串描述符的其他描述符时，wIndex 的值为0 。wLength 域为请求设备返回数据的字节数 ，设备实际返回的字节数可以比该域的字节数少 。 设备在收到设备描述符的请求之后，应该按照所请求的描述符类型编号，在数据返回过程中返回相应的描述符 。另外的接口描述符和端点描述符时跟随配置描述符一并返回的，不能单独请求返回（如果单独返回，主机无法确认他们属于哪个配置）。**

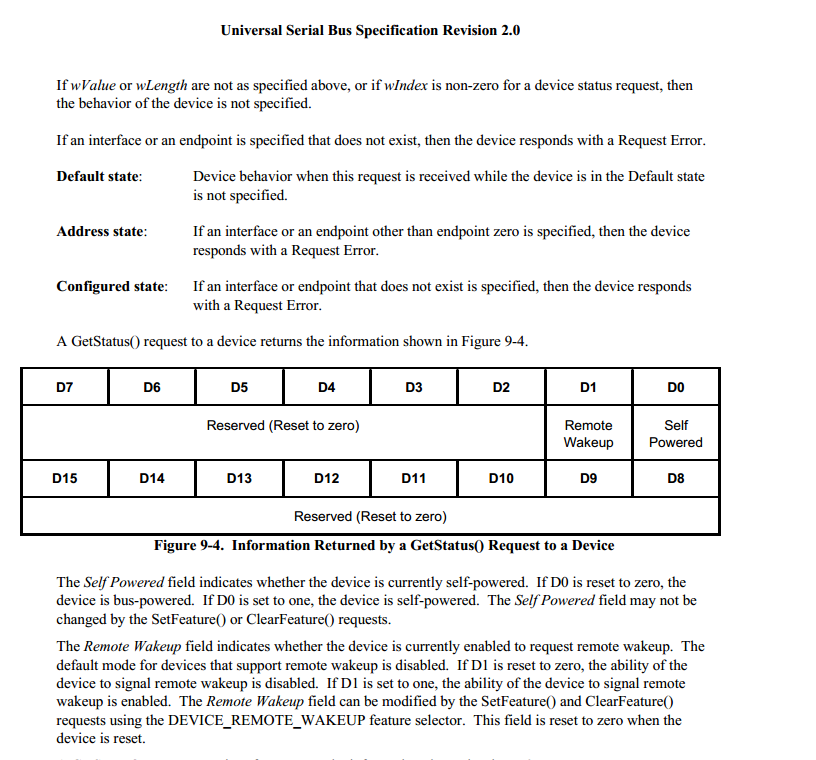
**注意： wValue , wIndex ,wLength 这三个域都是两个字节，在USB写一个中规定，使用的是小端结构，即低字节在前，高字节在后。**

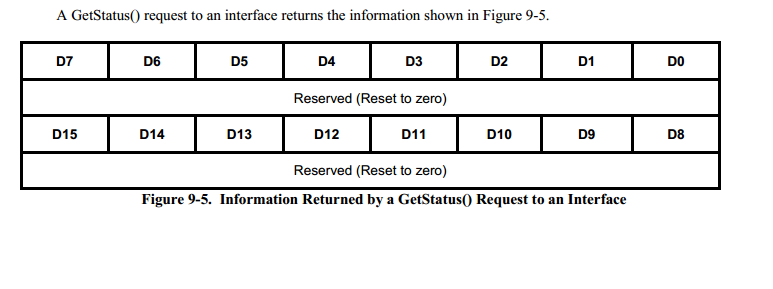
**4,Get Interface**

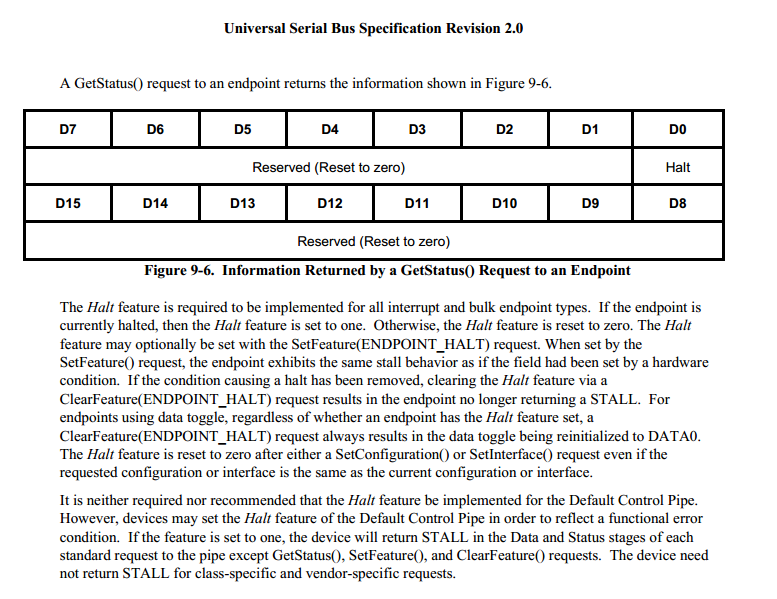
****

**5,Get status**

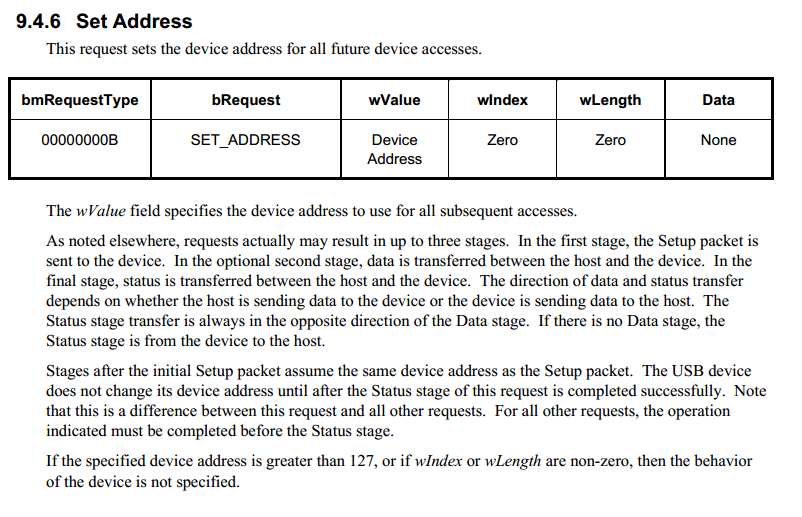
****

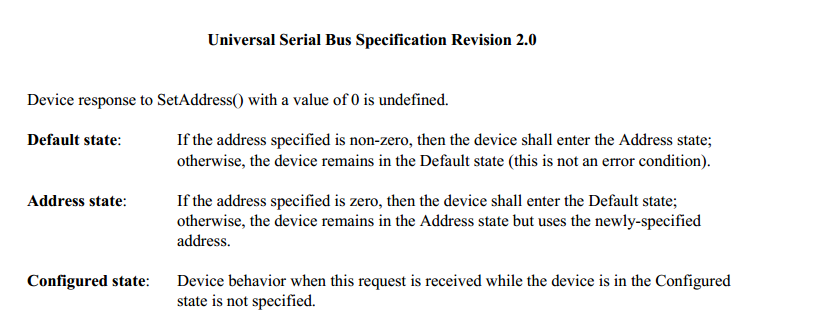
****

****

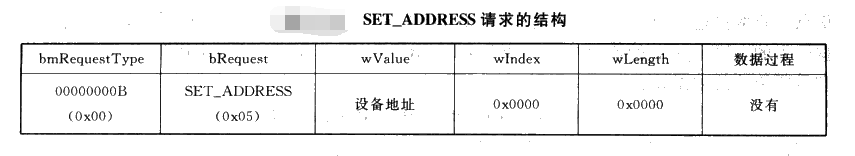
****

**6,Set Address**

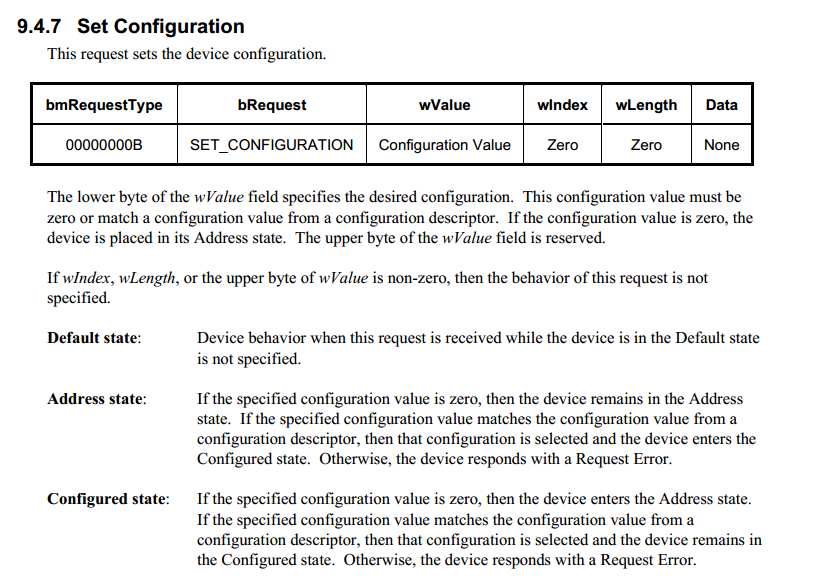
****

****

**SET\_ADDRESS （设置地址）请求是主机请求设备使用指定地址的请求 ， 指定的地址就是包含在8字节数据中的wValue 字段中 。每一个连接在同一个主控制器上的USB设备都需要具有唯一的设备地址，这样主机才能区分每个不同设备。当设备复位后，都是默认使用地址0 。主机从地址为0的设备获取设备描述符，一旦收到第一次设备描述符后，主机就会发设备地址请求，以尽量减伤设备使用公共地址0的时间。设置地址请求是没有数据过程的，因此wlength ,wInedx 均为0 。 当设备收到请求设置地址后，就直接进入状态过程，等待主机读取0长度的状态数据包。 主机成功读取状态数据包（用ACK响应设备）后，设备将启用新的地址。之后主机和设备通信都是使用新地址通信。**

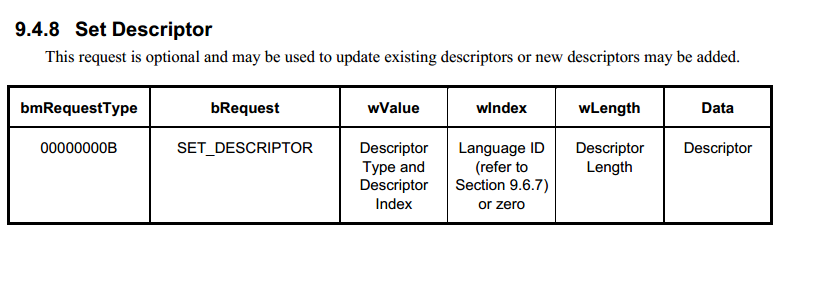
****

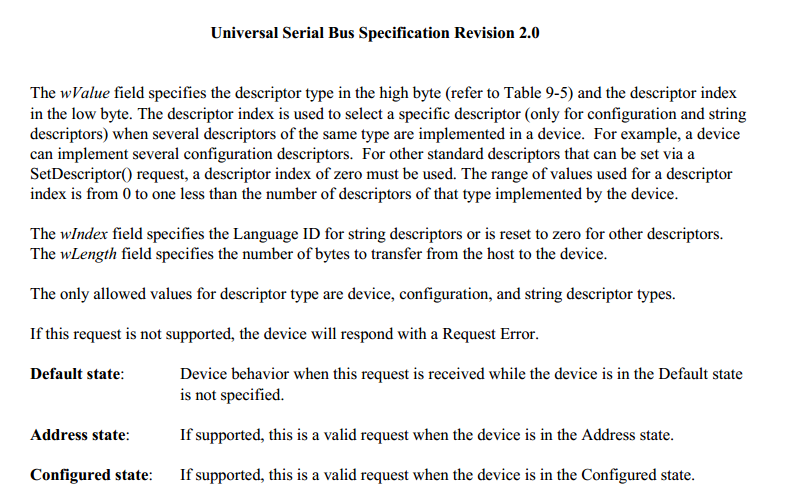
**7,Set Configuration**

****

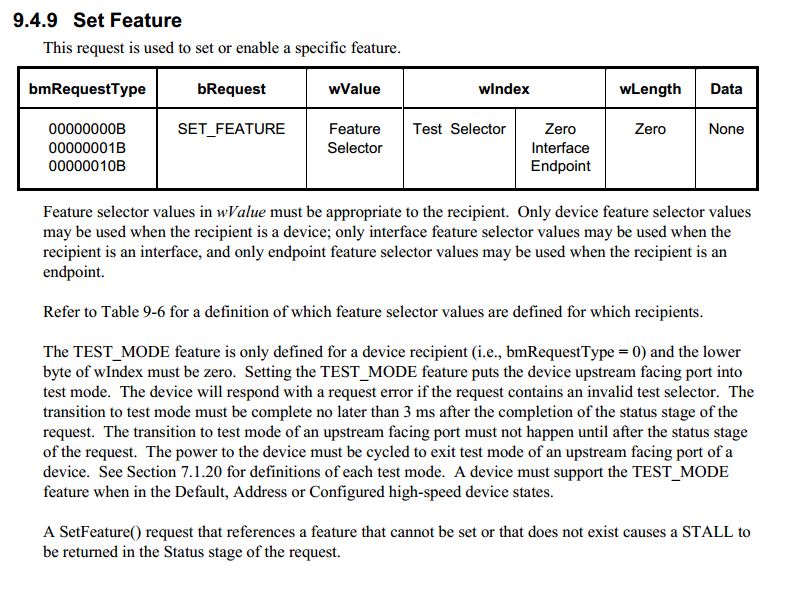
**SET\_CONFIGURATION （设置配置）请求和设置地址请求类似 。在设置地址请求中，wValue 的第一个字节（低字节）为设备的地址；而在设备配置请求中，wValue的第一字节为配置的值。该值与某配置描述符中的配置编号一致时，表述选中该配置。该值通常为1，因为大多数USB设备只有一种配置，配置编号为1 ；如果该值为0， 则会让设备进入设置地址状态。设备只有在收到非0的配置后们才能启用它的非0端点。**

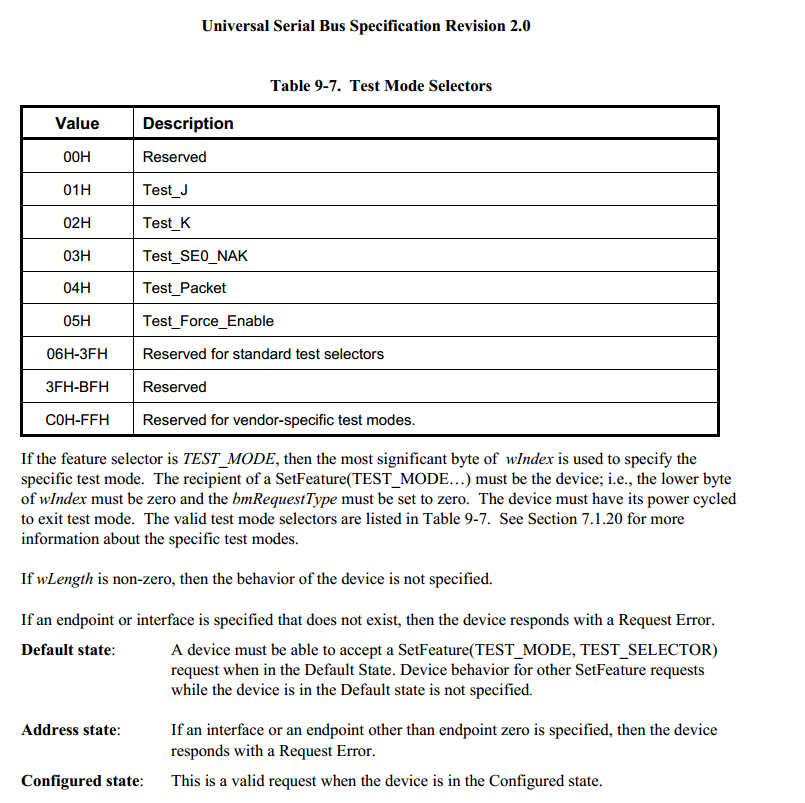
**8, Set Descriptor**

****

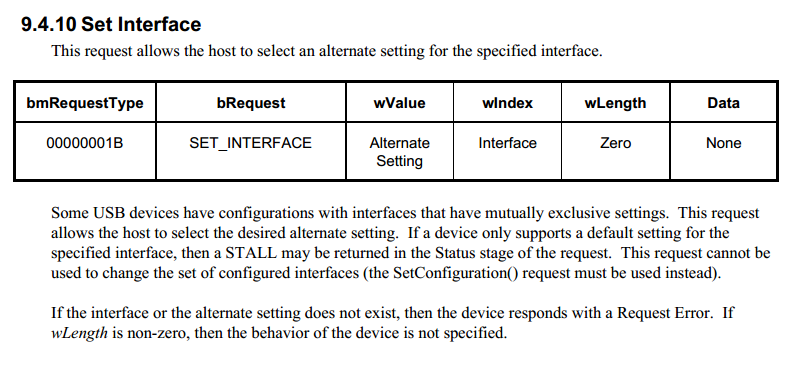
****

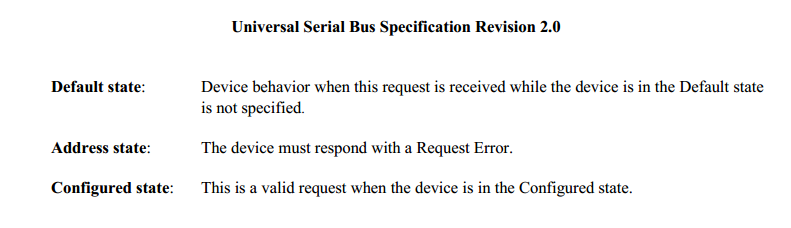
**9 , Set Feature**

****

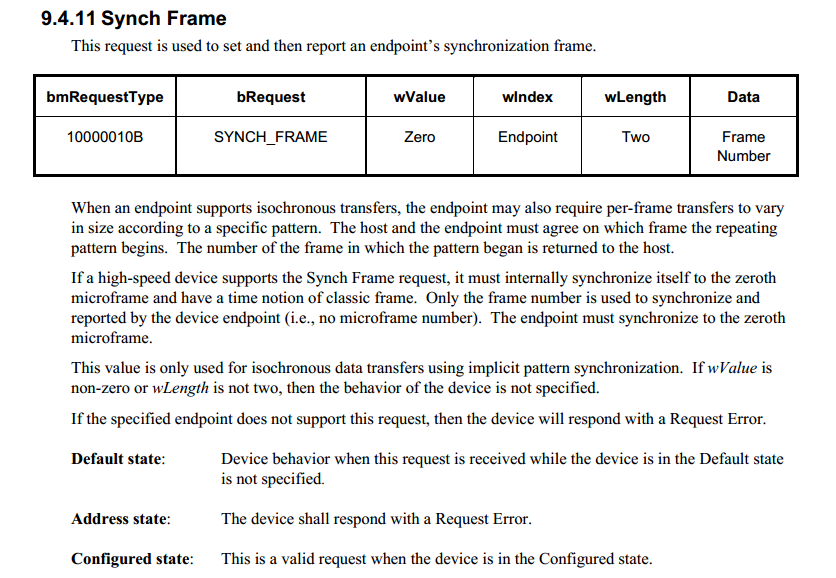
****

**10 , Set Interface**

****

****

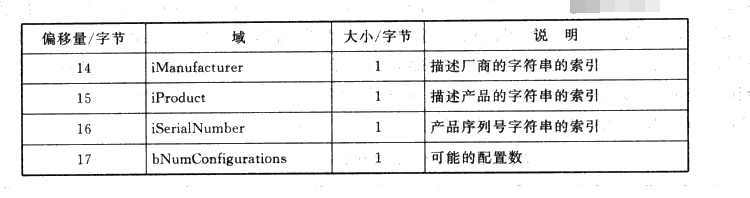
**11,Synch Frame**

****

**设备描述符的实现**

**首先看一下设备描述符的结构 ，**

****

****

**设备描述符长度为18个字节 ， 这个可以看请求设备描述符的命令 ：**

****

**bLength 长度为1个字节，表示该描述符的长度。设备描述符的长度为18个字节 【0x12】 ;**

**bDescriptorType 描述符类型，长度为一个字节，值为【0x01】。**

**bcdUSB 该设备使用的USB协议的版本 ，长度为2字节 。可取2.0或1.1等。 注意它是用BCD码表示的，例如USB2.0 协议就是0X200，USB1.1就是0X0110。**

**bDeviceClass 长度为1字节 ，是设备使用的类代码 。设备的类代码由USB协会规定，具体的类代码可查阅USB相关文档。对于大多数标准的USB设备类，该字段通常设置为0，而在接口描述符中的bInterfaceClass 中指定接口所实现的功能 。当bDeviceClass 为0时，**

**下面的bDeviceSubClass也必须为0.如果bDeviceClass 为0xff ,表示厂商自定义的设备类。**

**bDeviceSubClass 长度为1字节，是设备所使用的子类代码。当类代码不为0和0xff时，子类代码由USB协议规定。**

**bDeviceProtocol 长度为1字节，是设备所使用的的协议，协议代码由USB协会规定。该字段为0时，表示设备不是使用类所定义的协议。该字段为0xff时，表示设备使用厂商自定义的协议。bDeviceProtocol 必须要结合设备类和设备子类联合使用才有意义，因此类代码为0时，bDeviceProtocol也要为0 。**

**bMaxPackSize0 长度为1字节，是端点0的包长的最大包长。它的取值可以为8,16,32,64 。**

**idVender 长度为2字节 ，它是厂商的ID号 。 该ID 由USB协会分配，不能随意使用。 申请ID 号，需要交钱 。主机通常靠厂商的厂商ID 号，产品ID号以及产品序列号来安装和加载驱动的。**

**idProduct 长度为2个字节 ，是产品ID号。 与产生ID不同，产品ID是产商自己分配 。**

**bcdDevice 长度为2字节 ，是设备版本号 。**

**iManufacturer 长度为1字节 ，是描述厂商的字符串的索引值。当该值为0时，表示没有厂商字符串。主机获取设备描述符时，会将索引值放在wValue的第一个字节中，用来选择不同的字符串。**

**iProduct 长度为1字节 ，是描述产品的字符串的索引值。当该值为0时，表示没有产品字符串。当第一次插上某个USB设备时，会在Windows的右下角弹出一个对话框，显示发现新硬件，并且会显示该设备的名称。其实这两显示的信息就是从产品字符串里获取的。如果想让它显示出所需要的信息，应该修改产品字符串 。**

**iSerialNumber 长度为1字节，是设备的序列号字符串索引值。最好给你的每一个产品质地一个唯一的序列号。**

**bNumConfigurations 长度为1字节，表示设备有多少种配置 。每种配置都会有一个配置描述符，主机通过发送设置配置来选择某一种配置。大部分设备只有一个配置，即该字段值为1 。**

**下面为一个设备描述符的实现：**

****