Usb例程使用的是一种面向对象的编程方式，使得程序读起来非常晦涩，感觉程序一层一层在套娃。所以本文尝试梳理usb例程，从大框架下，粗略理解程序运行过程。Usb例程的main函数里，只有两个函数，一个用来初始化usb，一个用于usb任务。我们分两个部分来讲usb例程。第一部分讲它到底是如何初始化usb的，另一个讲它到底是怎么工作的。结合上面思维导图以及程序理解。

APPinit函数里又分为三个函数，如图中所示，第一个用来初始化设备，第二个使能中断，第三个运行usb设备。

重点就在于第一个函数到底是以怎样一个过程来初始化函数的。

从参数上来看，第一个CONTROLLER\_ID。主要有两类控制器id一个是khci，一个是ehci，前者用于usb的full speed， 后者用于high speed。 本实验的CONTROLLER\_ID就是使用khci模块。第二个参数是USB\_DeviceCallback，这个是定义的设备回调函数的地址，第三个则是虚拟串口设备的句柄。

设备回调函数USB\_DeviceCallback，到底是怎样的一个函数呢？它是为了处理usb设备事件而定义的，而它处理的事件，有两种，第一种总线复位事件，第二种设置usb的配置。我们知道，当usb设备插入比如说PC中，在通信前，首先就是主机发起复位信号，复位总线。所以说mcu一旦接收到复位信号后，就需要DeviceCallback处理复位事件。而复位事件中，最重要的就是初始化usb端点0。因为端点0就是用来与主机进行各种传输配置和控制信息的端口。复位以后，主机就会开始枚举操作，发送的获取描述符，获取配置等命令都是发给这个端口，所以必须立刻初始化这个端口0。而这个端口的回调函数是USB\_DeviceControlCallback，这个函数是用来组织，配置数据。当mcu接收到获取描述符的命令后，这个端点的回调函数则会调用USB\_DeviceCh9GetDescriptor，来组织mcu的描述符，组织完的数据再传回去，给mcu从端点0发送。

第二个设置usb的配置则是初始化其他端点，该例程初始化了3个端点。端点0是用来传输控制信息的，当通信建立好后，则该端点就不用了，而使用其他端点传输报文和接受报文。除了端点0是双向的以外，每个端口都只能设置一个传输方向，要么是输入的，要么是输出的。

进入USB\_DeviceInit函数，它主要功能就是将之前回调函数赋予传入的句柄。然后将端点的回调函数，回调参数，以及忙碌状态清除，再将khci的各种控制函数赋予该句柄。

端点的回调函数是用来处理对应端点发生的事件，端点0的回调函数要处理，主机发送过来的获取描述符等等命令，其他端点回调函数，则要处理要么发送数据的事情，要么接受数据，解析数据的任务。Khci的控制函数则包含了各种发送，接受，初始化寄存器等最基本操作。无论mcu是发送数据还是接收数据，最终都需要调用这块函数。

将这些功能全部集中到一个句柄上来，从而通过操作这个句柄，就可以操作所有函数。

该函数最后使用了deviceHandle->controllerInterface->deviceInit. 其实就是调用了khci的USB\_DeviceKhciInit。

跳出USB\_DeviceInit以后，最后调用USB\_DeviceRun，而运行usb其实就是调用了USB\_DeviceKhciControl，也是之前赋予句柄的khci函数之一。

从运行角度来说，首先PC发出复位信号，MCU接受到复位信号产生一次usb复位中断。进入中断函数中，通过寄存器ISTAT值判断为usb复位中断，调用USB\_DeviceNotificationTrigger，而它调用USB\_DeviceNotification来提示这是一个复位事件。

USB\_DeviceNotification拥有两种功能，一种是处理总线复位事件，那么需要调用DeviceCallback来处理，另一种则是初始化完以后，调用端点回调函数来处理对应端点数据。

USB\_DeviceNotification发现是一个复位事件，就调用了USB\_DeviceCallback，USB\_DeviceCallback发现是复位事件，则初始化端点0。这样与PC通信的端点0就准备就绪。

PC开始枚举。PC发来获取描述符请求，mcu发生令牌包中断。USB\_DeviceNotificationTrigger调用USB\_DeviceNotification，然后调用端点0的回调函数，端点0发现是获取描述符请求，最后调用USB\_DeviceCh9GetDescriptor，将mcu描述符组织完成，存放在buffer里，上层函数调用USB\_DeviceKhciSend将buffer从端点0发送出去。

当PC发送设置配置请求时候，mcu发生令牌包中断。调用端点0的回调函数USB\_DeviceCh9SetConfiguration，端点0接收PC的配置，存在buffer里，然后端点0调用USB\_DeviceSetConfigure，而它会调用USB\_DeviceCallback，USB\_DeviceCallback发现是设置配置事件，将1，2，3端口配置完成。

其他请求也是相似过程。

往后收到PC数据就会调用端点2的批量输入回调函数，输出数据给PC则调用端点3的批量输出回调函数。