/sys/kernel/debug/usb/devices   
是一个内核调试接口文件，记录了当前系统所有 ​USB 控制器、总线、设备、接口和端点​ 的详细信息。

#### ****(1) 拓扑信息（****T****开头行）​****​

T: Bus=04 Lev=00 Prnt=00 Port=00 Cnt=00 Dev#= 1 Spd=5000 MxCh= 4

* ​\*\*Bus\*\*: 总线编号（如 04）。
* ​\*\*Lev\*\*: 拓扑层级（根设备为 0，通过集线器扩展递增）。
* ​\*\*Prnt\*\*: 父设备编号（根设备为 0）。
* ​\*\*Port\*\*: 在父设备中的端口号。
* ​\*\*Dev#\*\*: 设备在总线上的编号（如 1）。
* ​\*\*Spd\*\*: 速度（单位：Mbps），如 5000（USB 3.1 Gen1）、480（USB 2.0）。
* ​\*\*MxCh\*\*: 最大子设备数（根集线器的端口数）。

#### ​****​(2) 总线信息（****B****开头行）​****​

B: Alloc= 0/800 usb, #Ev= 0, #EvUsed= 0

* ​\*\*Alloc\*\*: 已分配/总内存（单位：KB）。
* ​\*\*#Ev\*\*: 总线事件计数（如设备插入、移除）。

#### ​****​(3) 设备信息（****D****开头行）​****​

D: Ver= 3.10 Cls=root\_hub (09) Sub=00 Prot=03 MxPS= 9 #Cfgs= 1

* ​\*\*Ver\*\*: USB 规范版本（如 3.10 表示 USB 3.1 Gen1）。
* ​\*\*Cls\*\*: 设备类代码（如 09 表示根集线器）。
* ​\*\*Sub/Prot\*\*: 子类和协议代码。
* ​\*\*MxPS\*\*: 最大数据包大小。
* ​\*\*#Cfgs\*\*: 支持的配置数量。

#### ​****​(4) 物理端口路径（****P****开头行）​****

P: Vendor=1d6b ProdID=0003 Rev=05.04

* ​\*\*Vendor\*\*: 厂商 ID（如 1d6b 是 Linux Foundation 的 VID）。
* ​\*\*ProdID\*\*: 产品 ID（如 0003 表示 USB 3.0 根集线器）。
* ​\*\*Rev\*\*: 固件版本。

#### ​****​(5) 速度与电源（****S****开头行）​****​

S: Manufacturer=Linux 5.4.0-80-generic xhci-hcd

S: Product=xHCI Host Controller

S: SerialNumber=0000:00:14.0

* ​\*\*Manufacturer\*\*: 控制器驱动/固件信息。
* ​\*\*Product\*\*: 设备名称。
* ​\*\*SerialNumber\*\*: 硬件标识（如 PCI 地址）。

#### ​****​(6) 接口信息（****I****开头行）​****​

I: If#= 0 Alt= 0 #EPs= 1 Cls=Wireless(01) Sub=01 Prot=01 Driver=btusb

* ​\*\*If#\*\*: 接口编号（如 0）。
* ​\*\*Cls/Sub/Prot\*\*: 接口类、子类、协议（如 Wireless）。
* ​\*\*#EPs\*\*: 端点数量。
* ​\*\*Driver\*\*: 绑定的驱动（如 btusb）。

#### ​****​(7) 端点信息（****E****开头行）​****​

E: Ad=81(I) Atr=03(Int.) MxPS= 16 Ivl=1ms

* ​\*\*Ad\*\*: 端点地址（81 表示端点1，输入方向）。（01）表示端点1输出方向
* ​\*\*Atr\*\*: 传输类型（03 表示中断传输）。
* ​\*\*MxPS\*\*: 最大数据包大小。
* ​\*\*Ivl\*\*: 轮询间隔。

# /sys/kernel/debug/usb/xhci/0000\:03\:00.0/event-ring

#### ****1)****cycle****（周期状态）​****​

* ​**含义**​：表示事件环的当前周期（Cycle）状态，用于区分事件的有效性。
* ​**值**​：
  + 0 → 偶数周期（Even Cycle）
  + 1 → 奇数周期（Odd Cycle）
* ​**用途**​：  
  xHCI 规范规定，硬件在写入事件时会将 Cycle 字段设置为当前周期。软件通过比较事件中的 Cycle 值和事件环的当前周期，判断事件是否有效。

#### ​****​(2)****dequeue****（出队指针）​****​

* ​**含义**​：指向事件环中下一个待处理事件的物理地址（即软件已处理到哪个位置）。由软件维护
* ​**值**​：16 进制表示的物理内存地址（如 0x00000000deadbeef）。
* ​**用途**​：  
  驱动通过更新 dequeue 指针通知硬件已处理完某位置之前的事件，硬件可覆盖这些位置写入新事件。

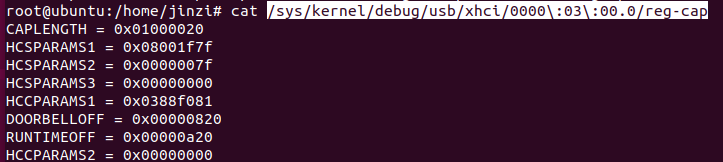
#### ​****​(3)****trbs****（TRB 地址）​****​

* ​**实际字段**​：应为 ​\*\*trb\_address\*\*​（Transfer Request Block Address）。
* ​**含义**​：指向事件环中最后一个已处理事件的 TRB（传输请求块）物理地址。
* ​**用途**​：  
  结合 dequeue 指针，用于同步硬件和软件对事件环的操作。

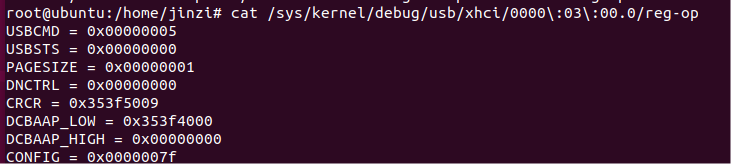
#### ​****​(4)****enqueue****（出队指针）​****​

* ​**含义**​：硬件（控制器）​\*\*​ 维护，指向事件环中下一个可写入事件的位置。硬件生成新事件时，会写入 enqueue 指向的位置，然后递增指针
* ​**值**​：16 进制表示的物理内存地址（如 0x00000000deadbeef）。
* ​**用途**​：  
  驱动通过更新 dequeue 指针通知硬件已处理完某位置之前的事件，硬件可覆盖这些位置写入新事件。

# /sys/kernel/debug/usb/xhci/0000\:03\:00.0/reg-cap

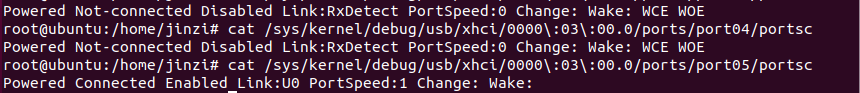


# /sys/kernel/debug/usb/xhci/0000\:03\:00.0/reg-op



# /sys/kernel/debug/usb/xhci/0000\:03\:00.0/ports/port02/portsc

 port02这里对应的是设备dev的号而非端口号。



### \*\*1. **Powered: Yes**\*\*​

* ​**含义**​：端口已通电（Power Enabled）。
* ​**寄存器位**​：对应 PORTSC 寄存器的 ​**PP (Port Power)​**​ 位。
* ​**状态**​：
  + Yes：端口供电正常，可为连接的设备提供电源。
  + No：端口未通电（可能因过流保护或手动禁用）。

### ​\*\*2. **Connected: No**\*\*​

* ​**含义**​：当前无设备连接（No Device Connected）。
* ​**寄存器位**​：对应 ​**CCS (Current Connect Status)​**​ 位。
* ​**状态**​：
  + Yes：检测到设备插入。
  + No：端口空闲，未连接设备。

### ​**3. Enabled（隐含字段）​**​

* ​**含义**​：端口是否启用（Port Enabled）。
* ​**寄存器位**​：对应 ​**PED (Port Enabled/Disabled)​**​ 位。
* ​**状态**​（需结合寄存器值）：
  + Enabled：端口已启用，可正常通信。
  + Disabled：端口被禁用（可能因错误或手动关闭）。

### ​\*\*4. **Link: U0**\*\*​

* ​**含义**​：USB 3.0+ 链路状态（Link State）。
* ​**寄存器位**​：对应 ​**PLS (Port Link State)​**​ 字段（4 位）。
* ​**常见状态**​：
  + ​\*\*U0\*\*​：活动状态（正常工作，全速传输）。
  + ​\*\*U1/U2/U3\*\*​：低功耗休眠状态（深度省电）。
  + ​\*\*Rx.Detect\*\*​：检测设备插入时的初始状态。

### ​\*\*5. **PortSpeed: 1**\*\*​

* ​**含义**​：端口的协商速度（Speed ID）。
* ​**寄存器位**​：对应 ​**PS (Port Speed)​**​ 字段（4 位）。
* ​**速度映射**​：
  + ​\*\*1**​：USB 2.0 ​**High-Speed\*\*​（480 Mbps）。
  + ​\*\*4**​：USB 3.0 ​**SuperSpeed\*\*​（5 Gbps）。
  + ​\*\*5**​：USB 3.1 ​**SuperSpeed+​\*\*​（10 Gbps）。

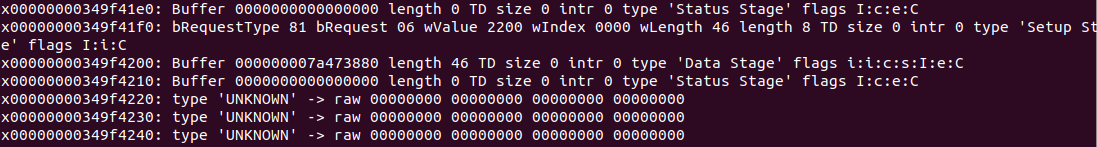
### ​**6. Change:（空值）​**​

* ​**含义**​：端口状态变化标志（Pending Events）。
* ​**寄存器位**​：对应 ​**CSC (Connect Status Change)​**、**PEC (Port Enabled Change)​**​ 等事件位。
* ​**状态**​：
  + 若显示具体标志（如 CSC=1），表示有未处理事件（如设备插入/拔出）。
  + 空值表示当前无待处理事件。

### ​\*\*7. **Wake: WCE WOE**\*\*​

* ​**含义**​：唤醒功能配置（Wake Control）。
* ​**寄存器位**​：
  + ​**WCE (Wake on Connect Enable)​**​：连接设备时触发唤醒。
  + ​**WOE (Wake Over Current Enable)​**​：过流事件触发唤醒。
* ​**状态**​：
  + WCE 或 WOE 存在时，表示唤醒功能已启用。

# /sys/kernel/debug/usb/xhci/0000\:03\:00.0/devices/01/ep00/trbs

  
显示设备端点的trbs