

Windows鼠标键盘(PS2)驱动框架

原创 叶之香 于 2018-05-24 13:57:54 发布 6861 收藏 11 版权

分类专栏: Windows驱动学习 文章标签: Windows鼠标键盘驱动



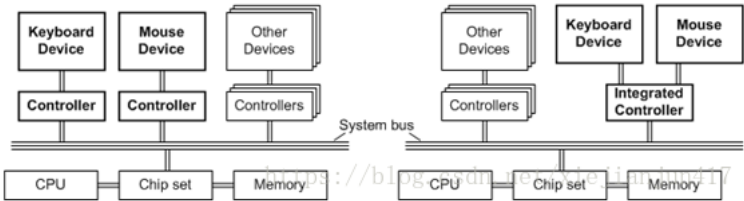
Windows驱动学习 专栏收录该内容

0 订阅 5 篇文章 订阅专栏

- 常用的鼠标类型：串口鼠标、PS/2鼠标、USB鼠标(HID)
- 常用的键盘类型：串口键盘、PS/2鼠标、USB键盘(HID)

非HID鼠标键盘(PS/2鼠标键盘)

- 物理配置方式



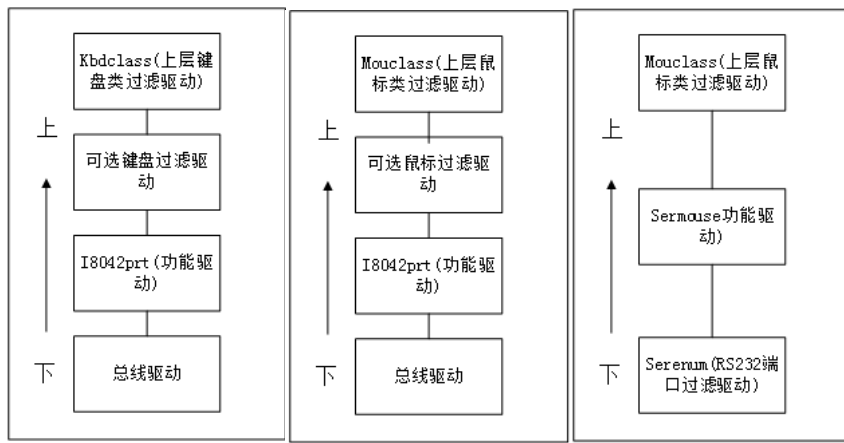
上图左边展示了键盘鼠标设备通过独立的控制器连接到系统总线。一种典型的配置由通过i8042控制器操作的PS/2类型键盘，和通过串口操作的串口鼠标组成。

上图右边展示了键盘鼠标设备通过集成的键盘鼠标控制器连接到系统总线。一种典型的配置由通过i8042控制器操作的PS/2类型键盘和鼠标组成。

Windows支持同时连接多个键盘和鼠标设备。然而Windows不支持通过单独的客户端访问每个设备。Win32子系统以独占方式打开所有的键盘和鼠标设备。

- 非HID鼠标键盘标准配置

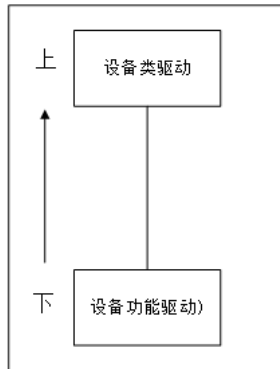
1. 标准的键盘鼠标驱动栈



即插即用PS/2键盘

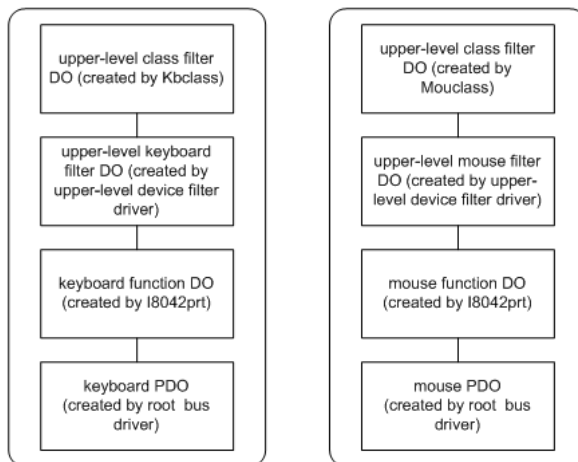
即插即用PS/2鼠标

即插即用串口鼠标



传统鼠标键盘

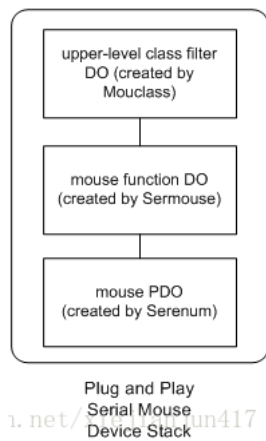
2. 键盘鼠标的设备对象结构



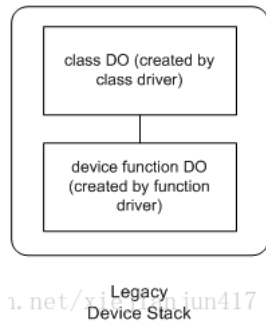
PS/2
Keyboard
Device Stack

PS/2 Mouse
Device Stack

上图描述了即插即用PS/2鼠标键盘的设备对象结构。类驱动创建上层类过滤设备对象，并通过可选的设备过滤对象附加到下层功能设备对象。上层设备过滤驱动创建上层设备过滤设备对象。I8042prt创建功能设备对象，并附加到总线驱动创建的物理设备对象。



上图描述了即插即用串口鼠标键盘的设备对象结构



上图描述了传统鼠标键盘的设备对象结构

3. 多个键盘鼠标的设备对象结构

Kbdclass和Mouclass支持以两种不同方式管理多个鼠标键盘设备。

① one-to-one模式

在一对一(one-to-one)模式中，每个设备有自己独立的设备栈。类驱动创建和附加独立的类设备对象到每个设备栈中。每个设备栈有自身的控制状态和输入缓冲。微软Win32子系统通过与设备关联的唯一文件对象访问输入。

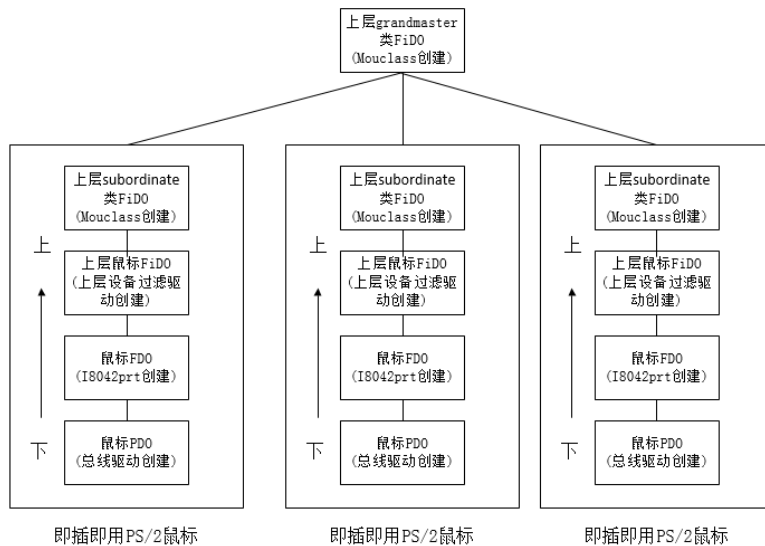
② grandmaster模式

在grandmaster模式中，类驱动以如下方式操作所有设备：

- 类驱动创建代表所有设备的grandmaster类设备对象，并为每个设备创建subordinate类设备对象。类驱动附加subordinate类设备对象到每个设备栈。在subordinate类设备对象下，设备栈和one-to-one模式创建的设备栈相似。
- grandmaster类设备对象控制所有subordinate类设备对象的操作
- Win32子系统通过代表grandmaster类设备的文件对象来访问所有设备的输入
- 所有设备的输入缓存在grandmaster的数据队列中

Kbdclass和Mouclass通过查询注册表路径

(HKLM\CurrentControlSet\SYSTEM\Services\<classservice>\Parameters)下的ConnectMultiplePorts值来决定采用哪种模式。ConnectMultiplePorts为0时，采用grandmaster模式；否则采用one-to-one模式。



上图描述了Grandmaster模式下鼠标的设备栈

- 非HID鼠标键盘的操作

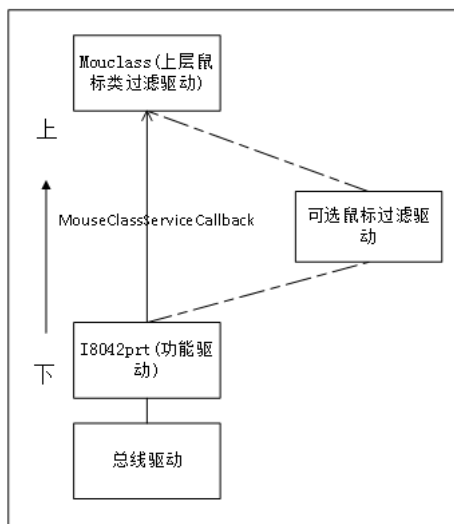
1. 打开和关闭键盘鼠标设备

Win32子系统以独占方式打开所有的键盘和鼠标设备。对每一个设备类，Win32子系统处理所有设备的输入，就像输入来自一个输入设备。应用程序不能请求只接收来自某一个特定设备的输入。

在从PNP manager接收到GUID_CLASS_KEYBOARD或GUID_CLASS_MOUSE设备接口启用通知后，Win32子系统动态地打开PNP输入设备。相反地，在接收到相应设备接口禁用后，Win32子系统关闭PNP输入设备。Win32子系统也通过名字打开传统设备(比如：\Device\KeyboardLegacyClass0)。需要注意的是一旦Win32子系统成功打开传统设备后，不能确定是否设备已经移除。

2. 连接类服务回调和过滤服务回调到设备

在设备打开前，类驱动必须连接相应的类服务到设备。在类驱动附加类设备对象到设备栈后，类驱动连接相应的类服务。功能驱动用类服务回调来传输设备的输入数据到设备相应类设备的数据队列。功能驱动的ISR(Interrupt Service routines)调用类服务回调。Kbdclass提供服务回调：KeyboardClassServiceCallback；Mouclass提供服务回调：MouseClassServiceCallback。



即插即用PS/2鼠标

可通过安装上层过滤驱动来修改类服务回调函数。WDK中的Kbfiltr和Moufiltr例子分别定义了KbFilter_ServiceCallback和MouFilter_ServiceCallback过滤回调函数。在自定义的过滤回调函数中，可修改传输到类设备数据队列的数据，比如删除、变换、或者插入数据。

类和过滤服务回调函数以如下方式进行连接：

- 类驱动发送内部设备连接请求到设备栈 (IOCTL_INTERNAL_KEYBOARD_CONNECT或者IOCTL_INTERNAL_MOUSE_CONNECT)。连接参数CONNECT_DATA中包含指定类设备对象的指针和类服务回调函数指针

- 过滤驱动接收到连接请求后，保存类连接参数，并用过滤驱动连接参数替换类驱动连接参数。过滤连接参数包含指向过滤设备对象和过滤驱动服务回调函数的指针。过滤驱动发送连接请求到功能驱动。

类和过滤服务回调函数以如下方式被调用：

- 功能驱动使用过滤驱动连接参数来调用过滤服务回调函数
- 在过滤输入数据后，过滤服务回调函数使用保存的类连接参数来调用类服务回调函数

3.查询和设置键盘设备

I8042prt驱动支持如下内部设备控制请求，查询键盘信息，设置键盘参数。

①IOCTL_KEYBOARD_QUERY_ATTRIBUTES

输出：KEYBOARD_ATTRIBUTES

②IOCTL_KEYBOARD_QUERY_INDICATOR

输出：KEYBOARD_INDICATOR_PARAMETERS

③IOCTL_KEYBOARD_QUERY_TYPEMATIC

输出：KEYBOARD_TYPEMATIC_PARAMETERS

④IOCTL_KEYBOARD_SET_INDICATOR

输入：KEYBOARD_INDICATOR_PARAMETERS

⑤IOCTL_KEYBOARD_SET_TYPEMATIC

输入：KEYBOARD_TYPEMATIC_PARAMETERS

4.查询鼠标设备

I8042prt驱动支持如下内部设备控制请求，查询鼠标信息。

①IOCTL_MOUSE_QUERY_ATTRIBUTES

输出：MOUSE_ATTRIBUTES

5.绝对指向设备的特殊要求

对于设备类型为GUID_CLASS_MOUSE的设备，设备功能驱动需：

- 处理设备特定的输入
- 创建MouseClassServiceCallback需要的MOUSE_INPUT_DATA结构
- 调用MouseClassServiceCallback将MOUSE_INPUT_DATA数据结构发送Mouclass数据队列

对于绝对指向设备，设备功能驱动必须如下设置MOUSE_INPUT_DATA中的LastX、LastY和Flags成员：

- 变化设备输入数据

$$\text{LastX} = ((\text{device input x value}) * 0xFFFF) / (\text{Maximum x capability of the device})$$

$$\text{LastY} = ((\text{device input y value}) * 0xFFFF) / (\text{Maximum y capability of the device})$$

- 设置Flags标志为MOUSE_MOVE_ABSOLUTE
- 如果输入数据需要被映射到虚拟桌面，则需要设置Flags标志为MOUSE_VIRTUAL_DESKTOP。如果未设置为MOUSE_VIRTUAL_DESKTOP，输入数据将被映射到主监视器