# PCI总线驱动程序设计

## 总线概念

总线是一种传输信号的信道；总线是连接一个或多个导体的电气连线。总线由电气接口和编程接口组成。

PCI 是Peripheral Component Interconnect（外围设备互联）的简称，是在桌面及更大型的计算机上普遍使用的外设总线。

PCI总线具有三个非常显著的优点：

1. 在计算机和外设间传输数据时具有更好的性能
2. 能够尽量独立于具体的平台
3. 方便地实现即插即用

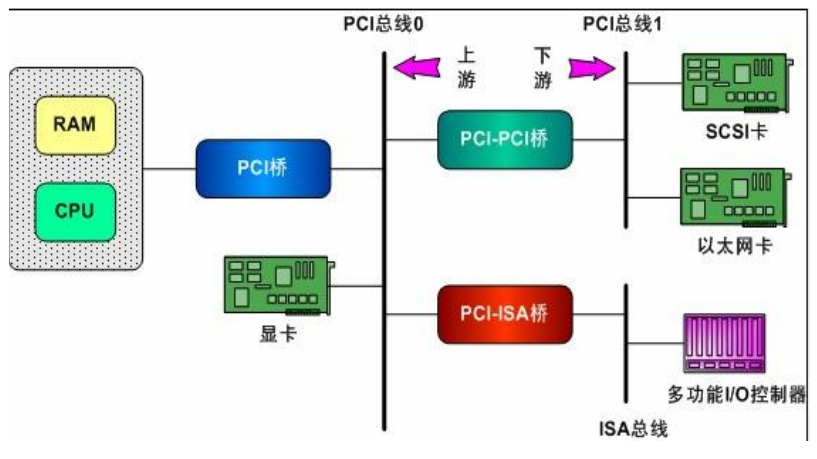
PCI设备寻址

每个PCI设备有一个总线号、一个设备号和一个功能号确定。PCI规范允许一个系统最多拥有256条总线，每条总线最多带32个设备，但每个设备可以是最多8个功能的多功能块。

40000000 –400003FF：0000:00:1f.1

40000000 –400003FF是它所映射的内存地址，而0000:00:1f:1则是这个PCI外设的地址。第一个16位表示域，第二个8位表示一个总线号，第三个5位表示一个设备号，最后3位表示功能号。

结构图如下



## 驱动设计

驱动描述

struct pci\_driver {

……..

const struct pci\_device\_id \*id\_table;

int (\*probe) (struct pci\_dev \*dev, const struct pci\_device\_id \*id);

void (\*remove) (struct pci\_dev \*dev);

/\* Device removed (NULL if not a hot-plug capable driver) \*/

……..

}

驱动注册

pci\_register\_driver(struct pci\_driver \*drv)

使能设备

在PCI驱动使用PCI设备的任何资源（I/0区或者中断）之前，必须调用如下函数来使能设备：

int pci\_enable\_device(struct pci\_dev \*dev)

获取基地址

一个PCI设备最多可以实现6个地址区域，大多数PCI设备在这些区域实现I/O寄存器。

pci\_resource\_start(struct pci\_dev \*dev, int bar)

返回指定区域的起始地址，这个区域通过参数bar指定，规范从0-5，表示6个PCI区域中的一个。

pci\_resource\_end(struct pci\_dev \*dev, int bar)

返回指定区域的末地址