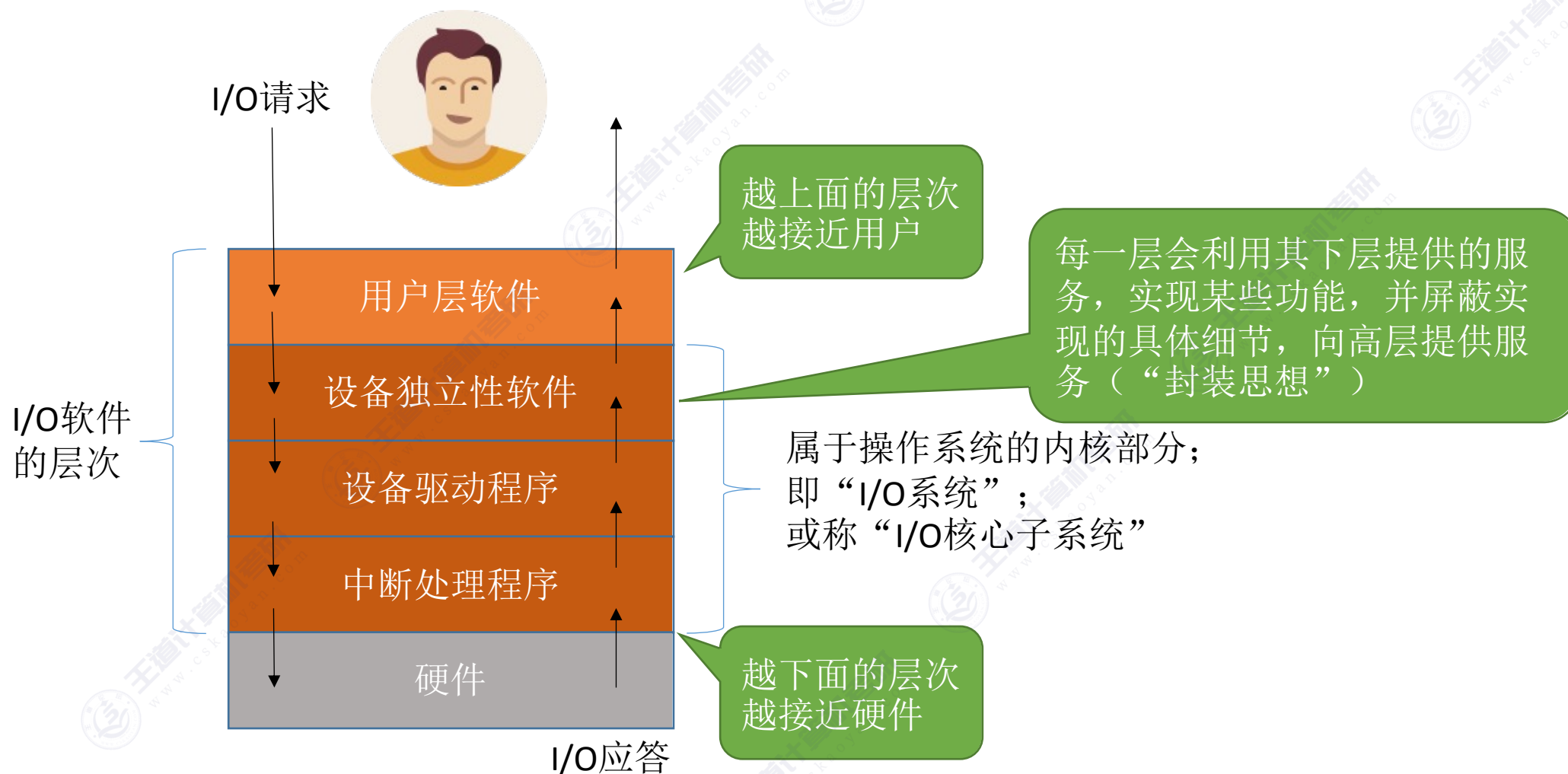


本节内容

# I/O软件层次结构

# 知识总览



# 用户层软件



属于操作系统内核部分



用户层软件实现了与用户交互的接口，用户可直接使用该层提供的、与I/O操作相关的库函数对设备进行操作

Eg: `printf("hello, world!");`

用户层软件将用户请求翻译成格式化的I/O请求，并通过“系统调用”请求操作系统内核的服务

Eg: `printf("hello, world!");` 会被翻译成等价的 `write` 系统调用，当然，用户层软件也会在系统调用时填入相应参数。

Windows 操作系统向外提供的一系列系统调用，但是由于系统调用的格式严格，使用麻烦，因此在用户层上封装了一系列更方便的库函数接口供用户使用（Windows API）

# 设备独立性软件



设备独立性软件，又称设备无关性软件。与设备的硬件特性无关的功能几乎都在这一层实现。

属于操作系统内核部分



主要实现的功能：

①向上层提供统一的调用接口（如 read/write 系统调用）

# 设备独立性软件



设备独立性软件，又称设备无关性软件。与设备的硬件特性无关的功能几乎都在这一层实现。

属于操作系统内核部分



主要实现的功能：  
②设备的保护

原理类似与文件保护。设备被看做是一种特殊的文件，不同用户对各个文件的访问权限是不一样的，同理，对设备的访问权限也不一样。

# 设备独立性软件



设备独立性软件，又称设备无关性软件。与设备的硬件特性无关的功能几乎都在这一层实现。

属于操作系统内核部分



主要实现的功能：  
③ 差错处理

设备独立性软件需要对一些设备的错误进行处理

# 设备独立性软件



设备独立性软件，又称设备无关性软件。与设备的硬件特性无关的功能几乎都在这一层实现。

属于操作系统内核部分



主要实现的功能：  
④设备的分配与回收

# 设备独立性软件



设备独立性软件，又称设备无关性软件。与设备的硬件特性无关的功能几乎都在这一层实现。

属于操作系统内核部分



主要实现的功能：  
⑤数据缓冲区管理

可以通过缓冲技术屏蔽设备之间数据交换单位大小和传输速度的差异



# 设备独立性软件



设备独立性软件，又称设备无关性软件。与设备的硬件特性无关的功能几乎都在这一层实现。

属于操作系统内核部分



主要实现的功能：

⑥建立逻辑设备名到物理设备名的映射关系；根据设备类型选择调用相应的驱动程序

用户或用户层软件发出I/O操作相关系统调用的系统调用时，需要指明此次要操作的I/O设备的逻辑设备名（eg：去学校打印店打印时，需要选择 打印机1/打印机2/打印机3，其实这些都是逻辑设备名）

设备独立性软件需要通过“逻辑设备表（LUT，Logical Unit Table）”来确定逻辑设备对应的物理设备，并找到该设备对应的设备驱动程序



## 设备独立性软件

属于操作系统内核部分



主要实现的功能：

⑥建立逻辑设备名到物理设备名的映射关系；根据设备类型选择调用相应的驱动程序

逻辑设备名	物理设备名	驱动程序入口地址
/dev/打印机1	3	1024
/dev/打印机2	5	2046
...	...	...

I/O设备被当做一种特殊的文件

不同类型的I/O设备需要有不同的驱动程序处理

操作系统系统可以采用两种方式管理**逻辑设备表（LUT）**：

第一种方式，**整个系统只设置一张LUT**，这就意味着所有用户不能使用相同的逻辑设备名，因此这种方式只适用于单用户操作系统。

第二种方式，**为每个用户设置一张LUT**，各个用户使用的逻辑设备名可以重复，适用于多用户操作系统。系统会在用户登录时为其建立一个用户管理进程，而LUT就存放在用户管理进程的PCB中。



## 设备独立性软件

属于操作系统内核部分



主要实现的功能：

⑥建立逻辑设备名到物理设备名的映射关系；根据设备类型选择调用相应的驱动程序

逻辑设备名	物理设备名	驱动程序入口地址
/dev/打印机1	3	1024
/dev/打印机2	5	2046
...	...	...

I/O设备被当做一种特殊的文件

不同类型的I/O设备需要有不同的驱动程序处理



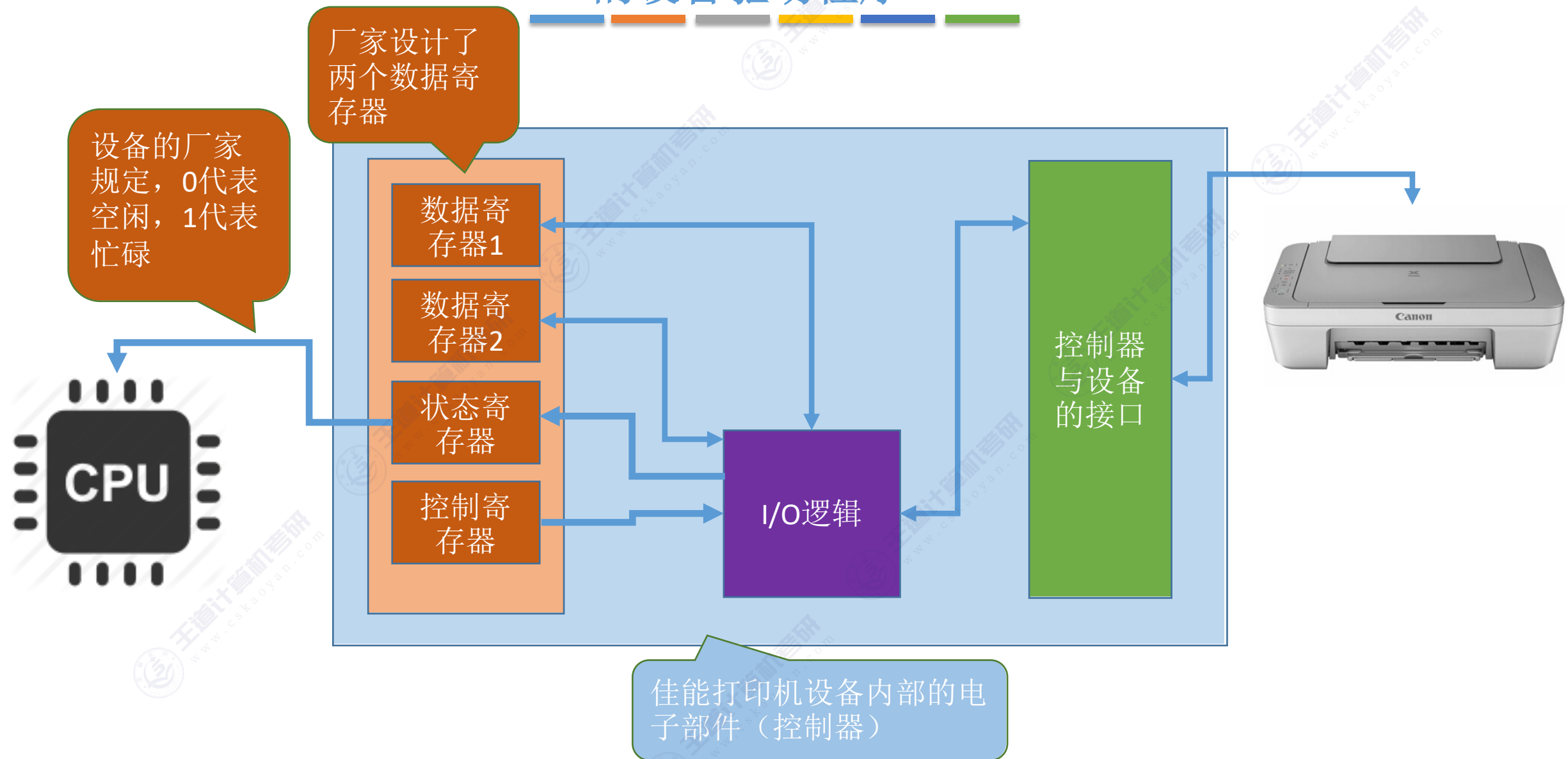
思考：为什么不同类型的I/O设备需要有不同的驱动程序处理？

思考：为何不同的设备需要不同的设备驱动程序？

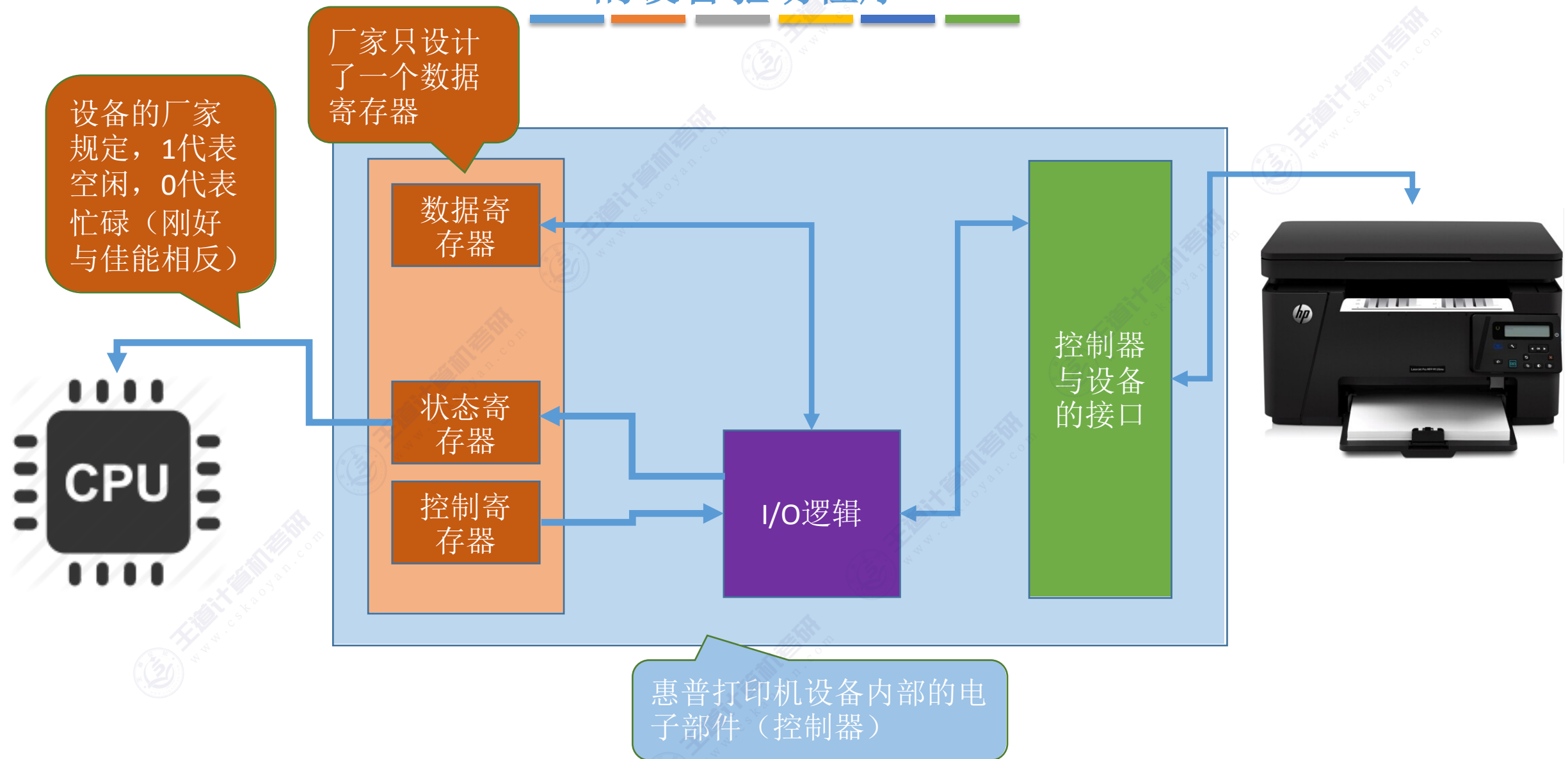


各式各样的设备，外形不同，其内部的电子部件（I/O控制器）也有可能不同

# 思考：为何不同的设备需要不同的设备驱动程序？



# 思考：为何不同的设备需要不同的设备驱动程序？





## 思考：为何不同的设备需要不同的设备驱动程序？



佳能打印机的厂家规定状态寄存器为 0 代表空闲，1 代表忙碌。有两个数据寄存器



惠普打印机的厂家规定状态寄存器为 1 代表空闲，0 代表忙碌。有一个数据寄存器



不同设备的内部硬件特性也不同，这些特性只有厂家才知道，因此厂家须提供与设备相对应的驱动程序，CPU 执行驱动程序的指令序列，来完成设置设备寄存器，检查设备状态等工作



## 设备驱动程序

逻辑设备  
表 (LUT)

逻辑设备名	物理设备名	驱动程序入口地址
/dev/打印机1	3	1024
/dev/打印机2	5	2046
...	...	...

属于操作系统  
内核部分



主要负责对硬件设备的具体控制，将上层发出的一系列命令（如 read/write）转化成特定设备“能听得懂”的一系列操作。包括设置设备寄存器；检查设备状态等

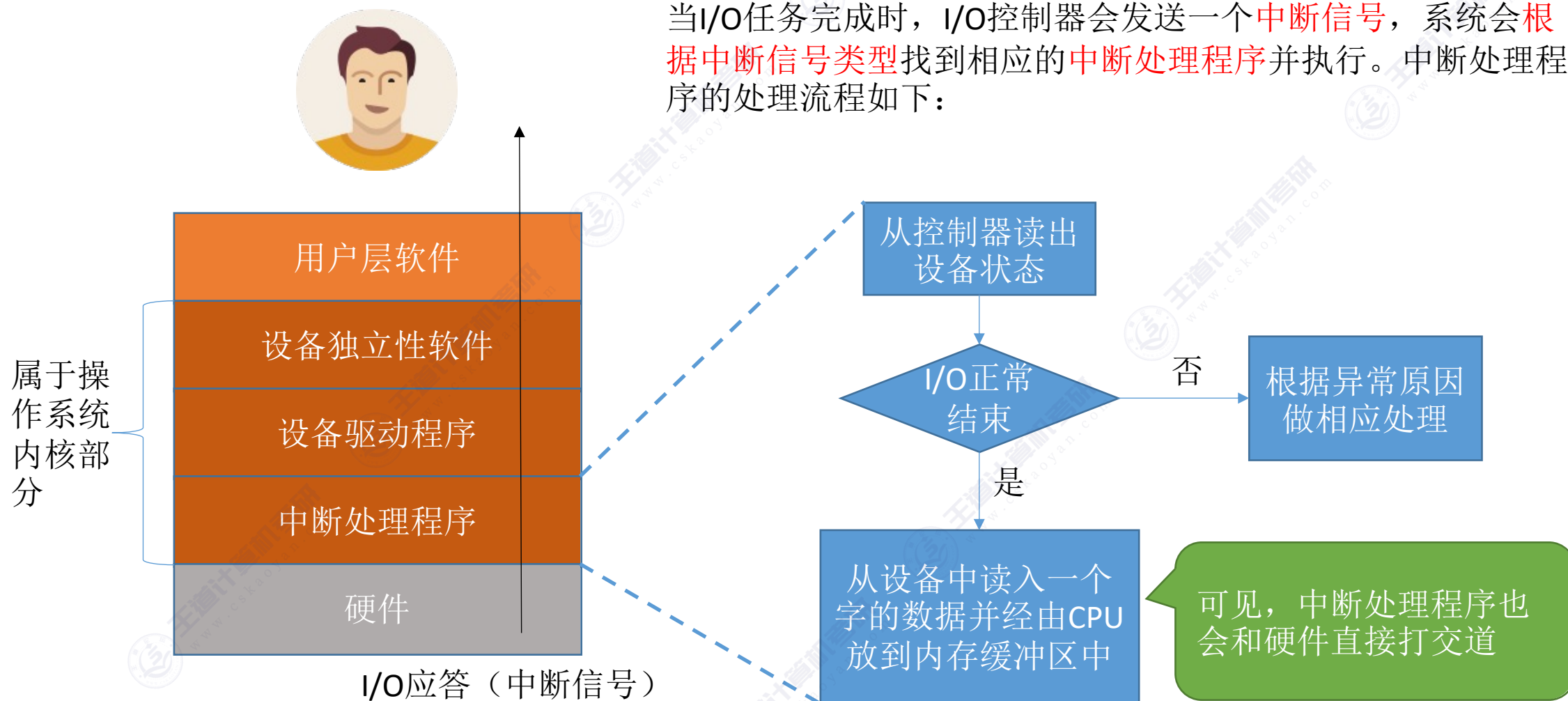
不同的I/O设备有不同的硬件特性，具体细节只有设备的厂家才知道。因此厂家需要根据设备的硬件特性设计并提供相应的驱动程序。

注：驱动程序一般会以一个独立进程的方式存在。

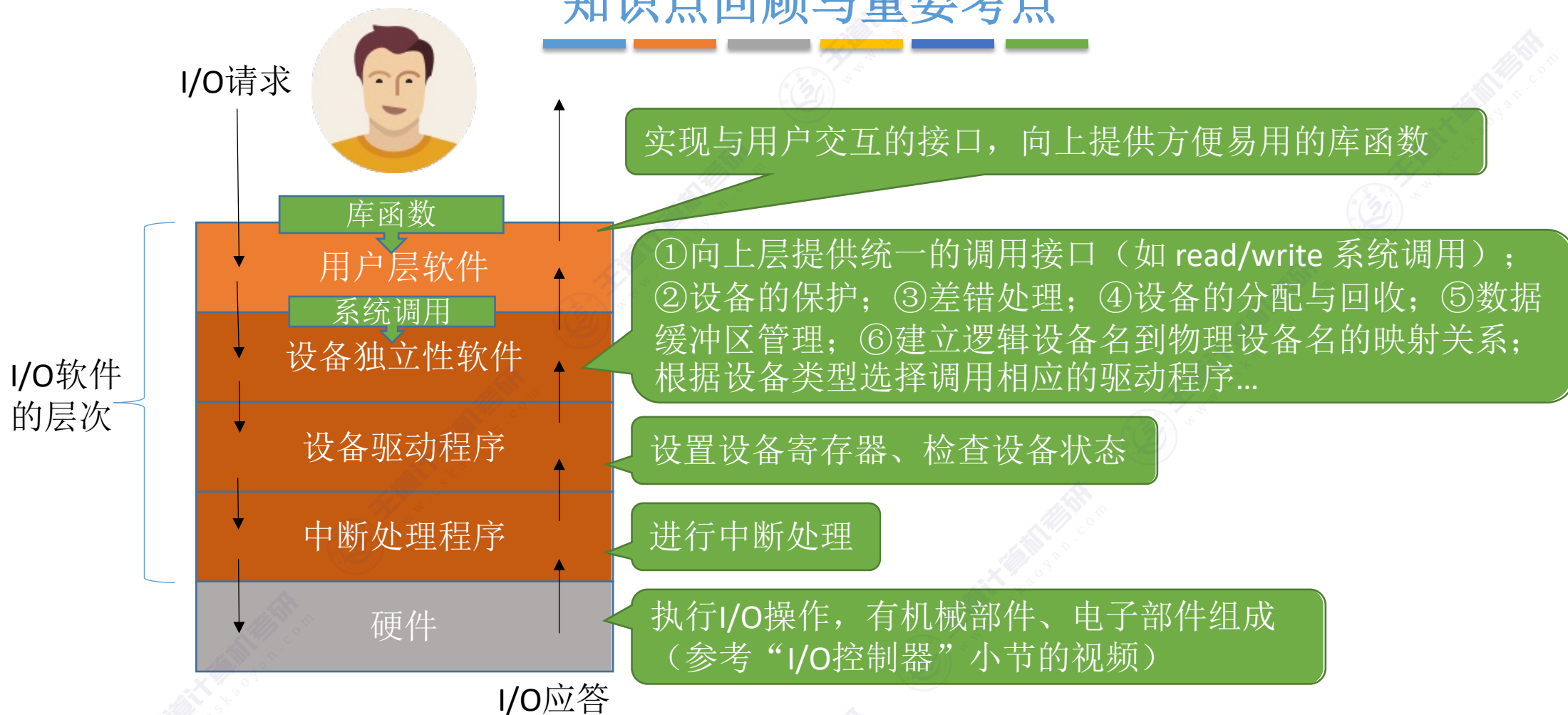


# 中断处理程序

当I/O任务完成时，I/O控制器会发送一个**中断信号**，系统会根据**中断信号类型**找到相应的**中断处理程序**并执行。中断处理程序的处理流程如下：

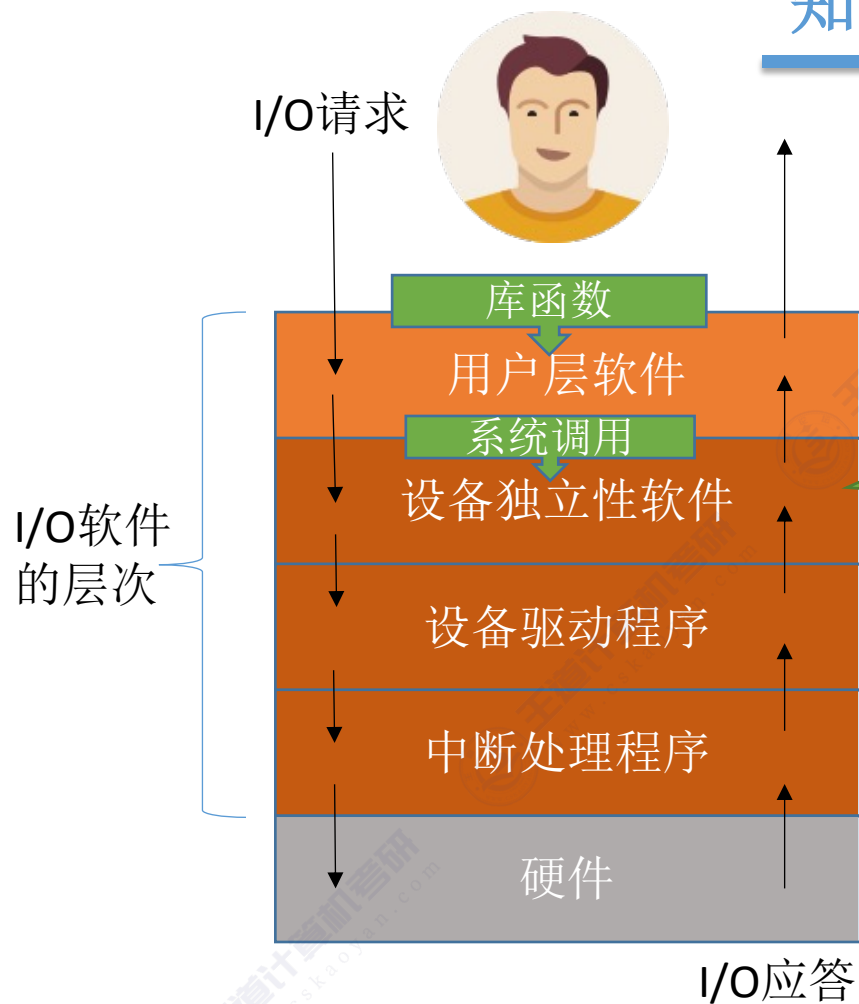


## 知识点回顾与重要考点



理解并记住I/O软件各个层次之间的顺序，要能够推理判断某个处理应该是在哪个层次完成的（最常考的是设备独立性软件、设备驱动程序这两层。只需理解一个特点即可：直接涉及到硬件具体细节、且与中断无关的操作肯定是在设备驱动程序层完成的；没有涉及硬件的、对各种设备都需要进行的管理工作都是在设备独立性软件层完成的）

## 知识点回顾与重要考点

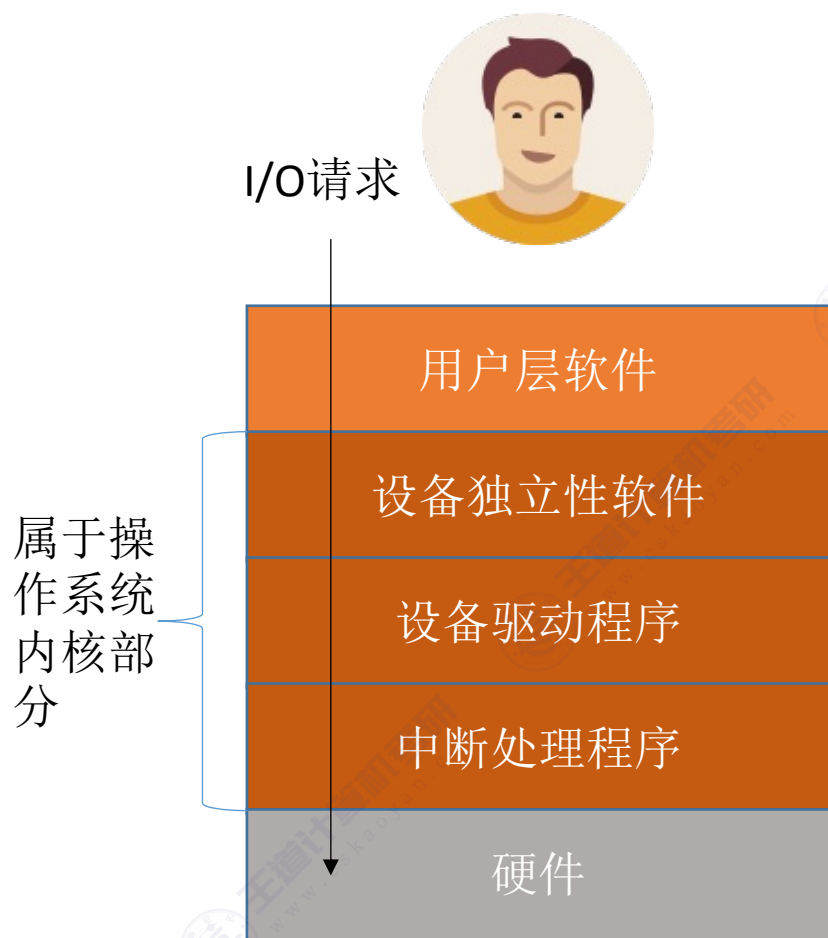


⑥建立逻辑设备名到物理设备名的映射关系；根据设备类型选择调用相应的驱动程序...（*逻辑设备表的作用*）

逻辑设备名	物理设备名	驱动程序入口地址
/dev/打印机1	3	1024
/dev/打印机2	5	2046
...	...	...

逻辑设备表（LUT，Logical Unit Table）

# 中断处理程序



用户通过调用用户层软件提供的库函数发出的I/O请求

- 用户层软件通过“系统调用”请求设备独立性软件层的服务
- 设备独立性软件层根据LUT调用设备对应的驱动程序
- 驱动程序向I/O控制器发出具体命令
- 等待I/O完成的进程应该被阻塞，因此需要进程切换，而进程切换必然需要中断处理



公众号：王道在线



b站：王道计算机教育



抖音：王道计算机考研