

# I/O操作的实现

---

- 分以下三个部分介绍

- 第一讲：用户空间I/O软件

- I/O子系统概述
    - 文件的基本概念
    - 用户空间的I/O函数

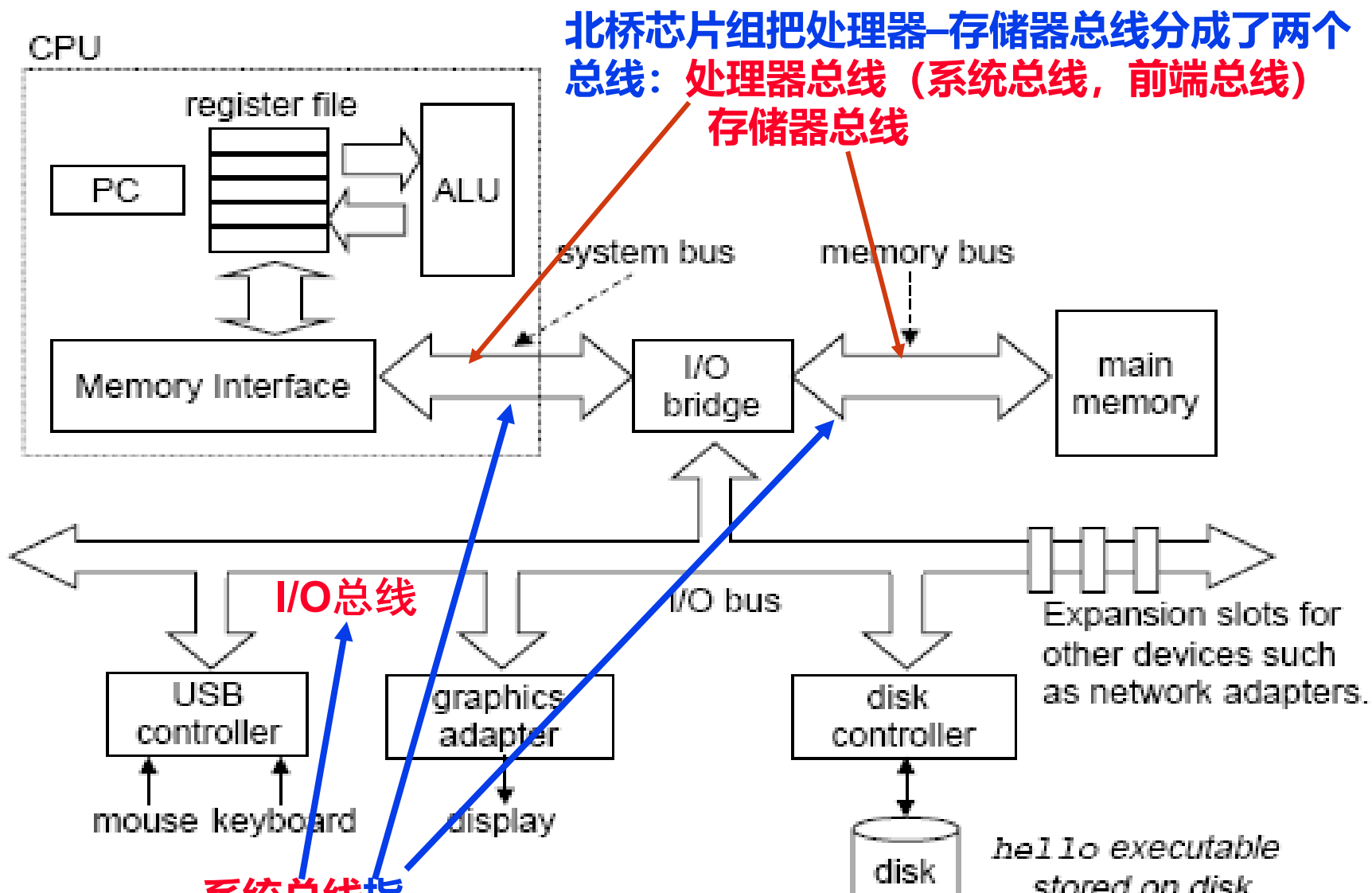
- 第二讲：I/O硬件和软件的接口

- 系统总线及系统互连
    - I/O设备和设备控制器
    - I/O端口及其编址方式
    - I/O控制方式

- 第三讲：内核空间I/O软件

- 与设备无关的I/O软件
    - 设备驱动程序
    - 中断服务程序

# Intel 体系结构中特指的“系统总线”



北桥芯片组把处理器-存储器总线分成了两个总线：处理器总线（系统总线，前端总线）  
存储器总线

I/O总线

系统总线指

系统总线上传传输的信息有哪些？

数据（指令、操作数、中断号）、地址、其他控制/状态/定时等信号！

# 系统总线的组成

- 系统总线通常由一组**控制线**、一组**数据线**和一组**地址线**构成。也有些总线没有单独的地址线，地址信息通过数据线来传送，这种情况称为**数据/地址复用**。
  - **数据线 (Data Bus)**：承载在源和目部件之间传输的信息。数据线的宽度反映一次能传送的数据的位数。
  - **地址线 (Address Bus)**：给出源数据或目的数据所在的主存单元或I/O端口的地址。地址线的宽度反映最大的寻址空间。
  - **控制线 (Control Bus)**：控制对数据线和地址线的访问和使用。用来传输定时信号和命令信息。**典型的控制信号包括：**
    - **时钟 (Clock)**：用于总线同步
    - **复位 (Reset)**：初始化所有设备
    - **总线请求 (Bus Request)**：表明发出该请求信号的设备要使用总线
    - **总线允许 (Bus Grant)**：表明接收到该允许信号的设备可以使用总线
    - **中断请求 (Interrupt Request)**：表明某个中断正在请求
    - **中断回答 (Interrupt Acknowledge)**：表明某个中断请求已被接受
    - **存储器读 (memory read)**：从指定的主存单元中读数据到数据总线上
    - **存储器写 (memory write)**：将数据总线上的数据写到指定主存单元中
    - **I/O读 (I/O read)**：从指定的I/O端口中读数据到数据总线上
    - **I/O写 (I/O Write)**：将数据总线上的数据写到指定的I/O端口中
    - **传输确认 (transmission Acknowledge)**：数据已被接收或已送总线

# 基本概念

---

- 总线裁决

早期：总线多是共享传输，需确定哪个设备使用总线。

现在：总线多是点对点传输，无需裁决。

- 总线定时

定义总线事务中的每一步何时开始、何时结束。

Synchronous (同步)：用时钟信号来确定每个步骤

Asynchronous(异步)：用握手信号来定时，前一个信号结束就是下一个信号的开始

半同步：结合使用时钟信号和握手信号来定时

- 并行/串行传输

并行传输：一个方向同时传输多位数据信号，故位与位需同步，慢！

串行传输：一个方向只传输一位数据信号，无需在位之间同步，快！

现在总线设计的趋势是：点对点、同步、串行

# 总线的性能指标

---

## ◦ 总线宽度

- 总线中数据线的条数，决定了每次能同时传输的信息位数

## ◦ 总线工作频率

- 每秒传送次数 (MT/s或GT/s)。早期的总线通常一个时钟传送一次数据，此时，工作频率等于总线时钟频率；现在总线一个时钟周期可传送2次或4次数据，因此，工作频率是时钟频率的2倍或4倍

## ◦ 总线带宽

- 总线的最大数据传输率（一秒钟内传输的数据量）
- 总线带宽计算公式： $B=W \times F/N$

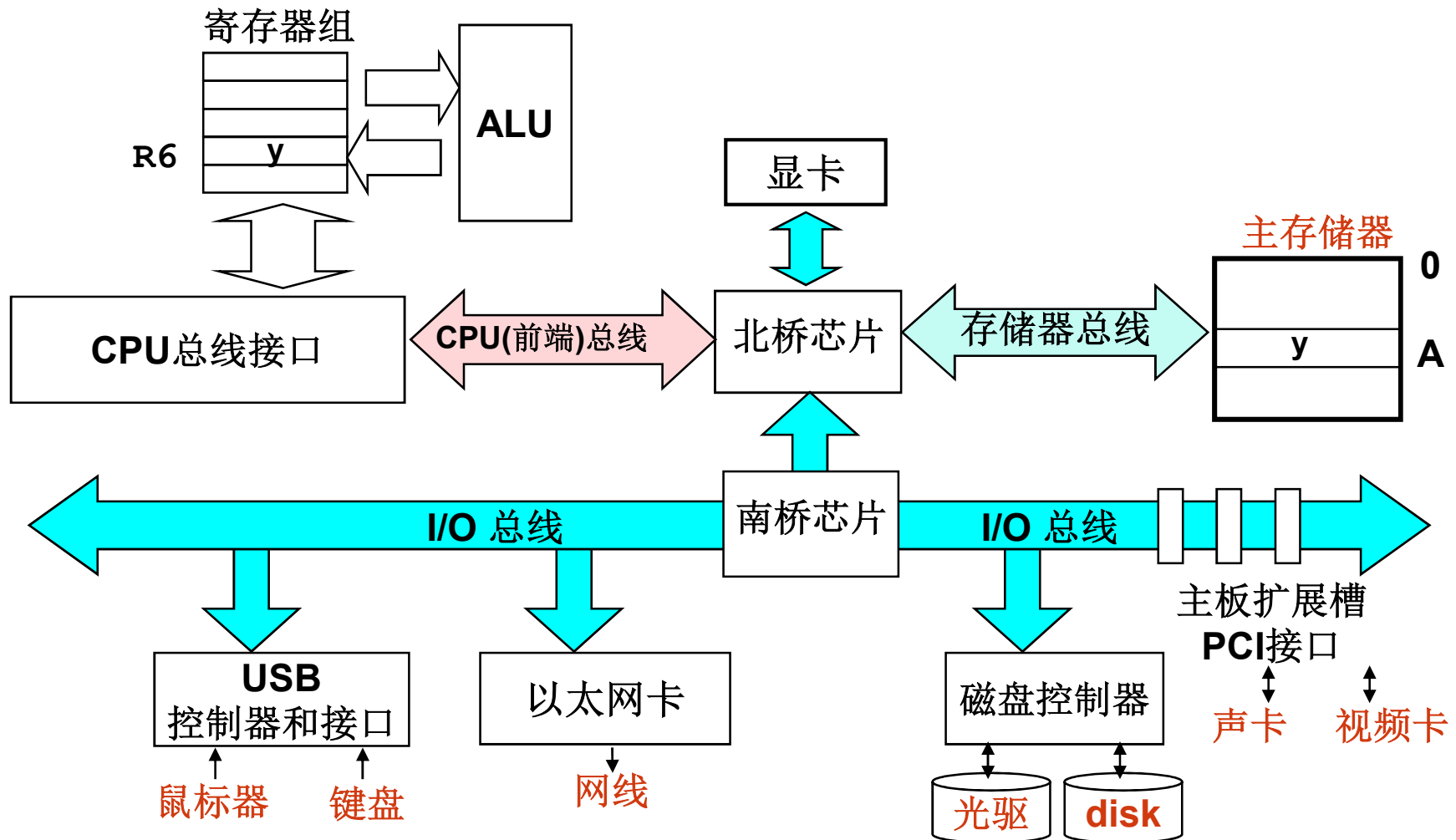
W-总线宽度；F-总线时钟频率；N-完成一次数据传送所用时钟周期数

F/N实际上就是总线工作频率

## ◦ 总线传送方式

- 非突发传送：每个总线事务都传送地址，一个地址对应一次数据传送
- 突发 (Burst) 传送：即成块数据传送。突发传送总线事务中，先传送一个地址，后传送多次数据，后续数据的地址默认为前面地址自动增量

# 回顾：I/O总线,I/O控制器与I/O设备的关系



# 处理器总线

---

## ◦ 前端总线 (Front Side Bus, FSB)

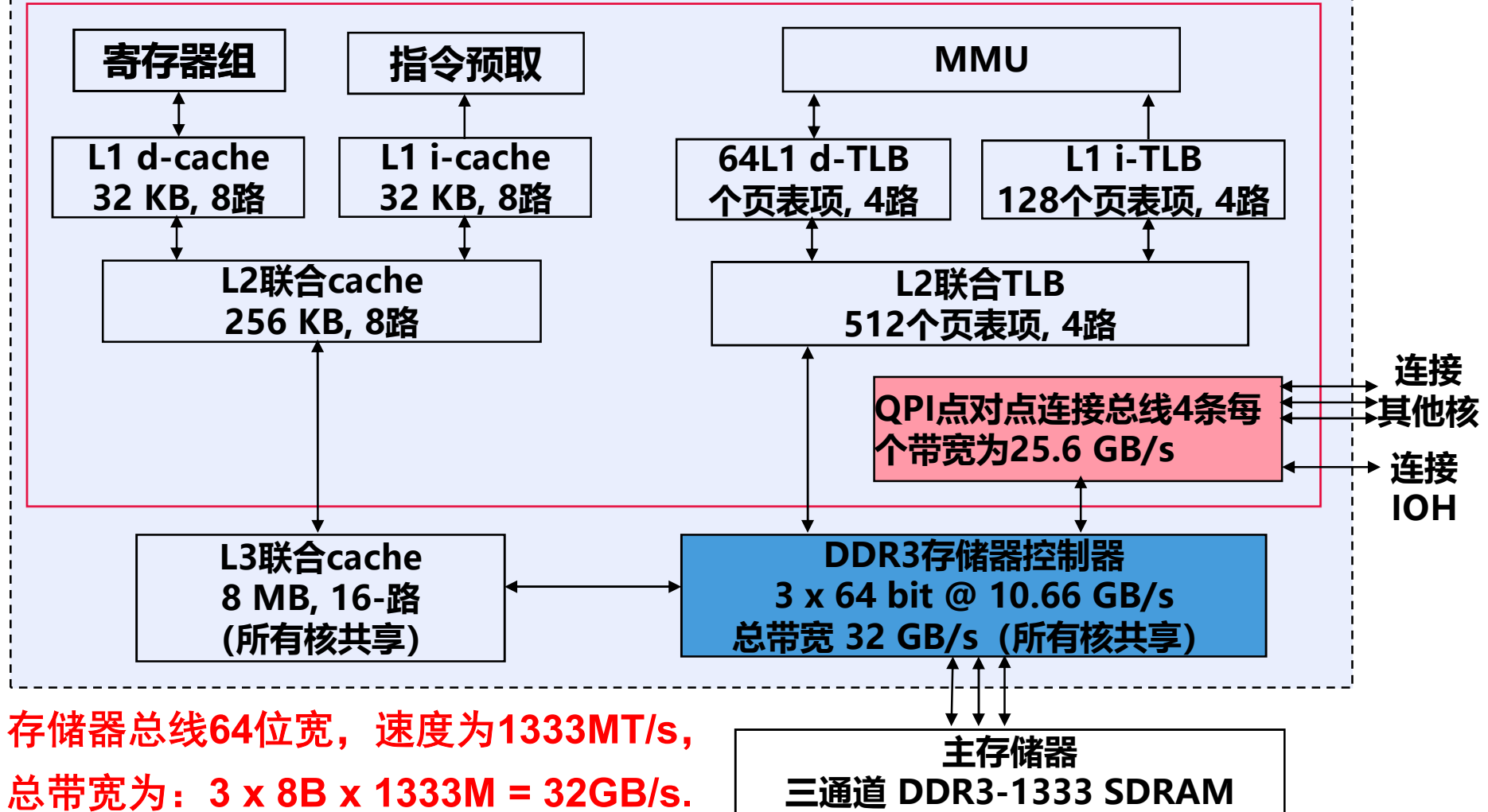
- 早期Intel架构使用, 位于CPU芯片与北桥芯片之间互连
- 从Pentium Pro开始, FSB采用quad pumped技术: **每个总线时钟周期传送4次数据**
- 并行传输、同步定时方式
- 若工作频率为1333MHz (实际单位应是MT/s, 表示每秒传送1333M次, 实际时钟频率为333MHz), 总线宽度为64位, 则总线带宽为 $1333\text{MT/s} \times 8\text{B} = 10.5\text{GB/s}$

## ◦ QPI (Quick Path Interconnect) 总线

- 目前在Intel架构中CPU芯片内部核之间、CPU芯片之间、CPU芯片与IOH (I/O Hub) 芯片之间, 都通过QPI总线互连
- QPI是**基于包交换的串行、高速点对点连接**: 发送方和接收方各有时钟信号, 双方同时传输数据 (各有20条数据线), 每个QPI数据包包含80位, 分两个时钟周期传送, 每个时钟周期传两次, 每次传20位 (16位数据+4位校验位), **QPI总线带宽为: 每秒传送次数 $\times 2\text{B} \times 2$** 。
- QPI总线的速度单位 (工作频率) 为GT/s, 表示每秒传送多少G次。若QPI时钟频率为2.4GHz, 则速度为4.8GT/s, 带宽为 $4.8\text{G} \times 2\text{B} \times 2 = 19.2\text{GB/s}$ 。

# 存储器总线

CPU芯片内含的核，共4个



从Core i7开始，北桥在CPU芯片内，CPU通过存储器总线（即内存条插槽，图中为三通道插槽）直接和内存条相连。3个存控包含在CPU芯片内。



# I/O总线

---

I/O总线用于为系统中的各种I/O设备提供输入输出通路

I/O总线在物理上可以是主板上的I/O扩展槽，如：

第一代：ISA/EISA总线、VESA总线，早被淘汰

第二代：PCI、AGP、PCI-X，被逐渐淘汰

第三代：PCI-Express（串行总线，主流总线）

## PCI-Express总线

两个PCI-Express设备之间以一个链路（link）相连

每个链路包含多条通路（lane），可以是1,2,4,8,16或32条

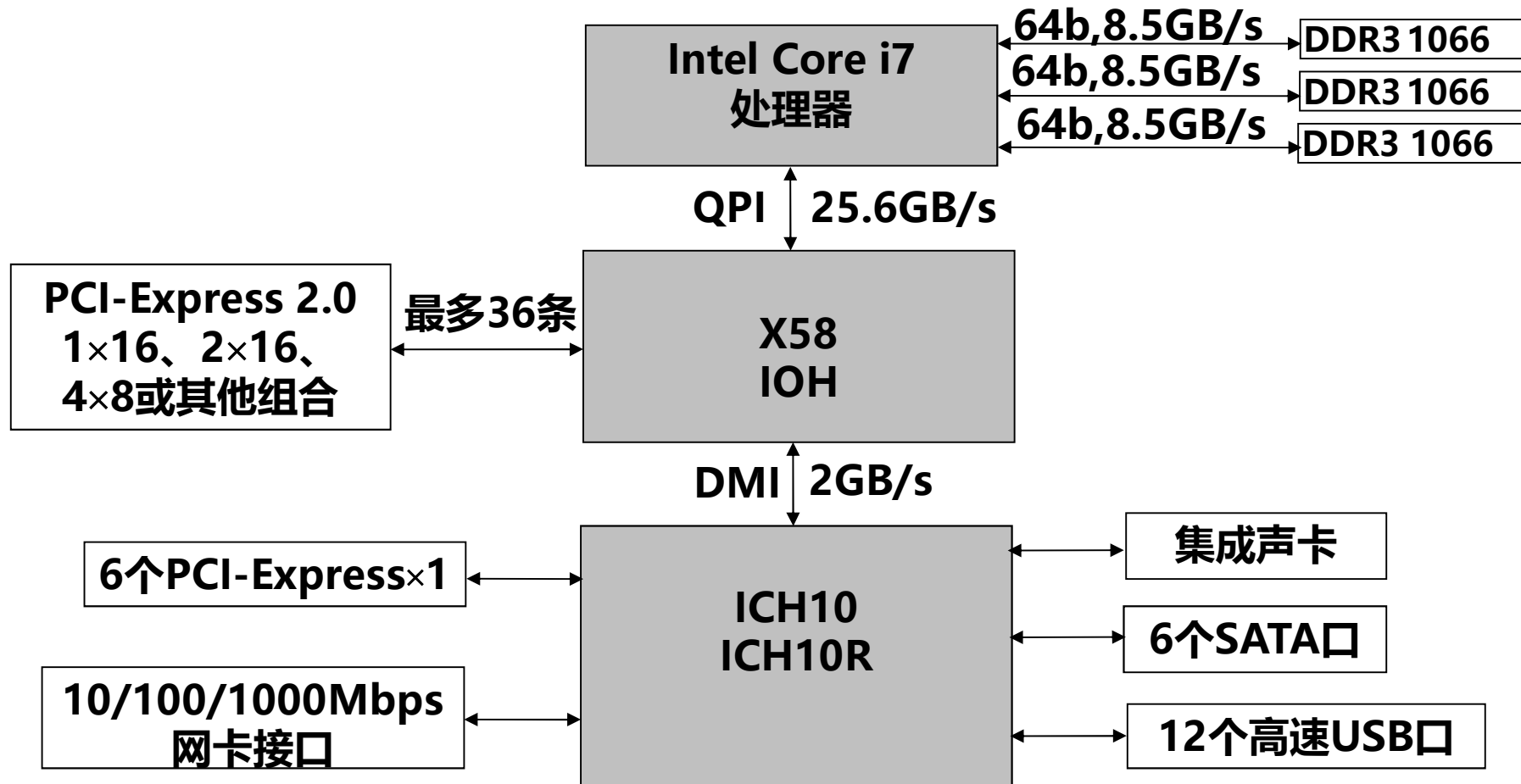
PCI-Express×n表示一个具有n条通路的PCI-Express链路

每条通路可同时发送和接受，每个数据字节被转换为10位信息被传输

PCI-Express1.0下，每条通路的发送和接受速率都是2.5Gb/s，故PCI-Express×n的带宽为： $2.5\text{Gb/s} \times 2 \times n / 10 = 0.5\text{GB/s} \times n$ 。

PCI-Express1.0下，PCI-Express×2的带宽为1GB/s，PCI-Express×4的带宽为2GB/s，PCI-Express×16的带宽为8GB/s。

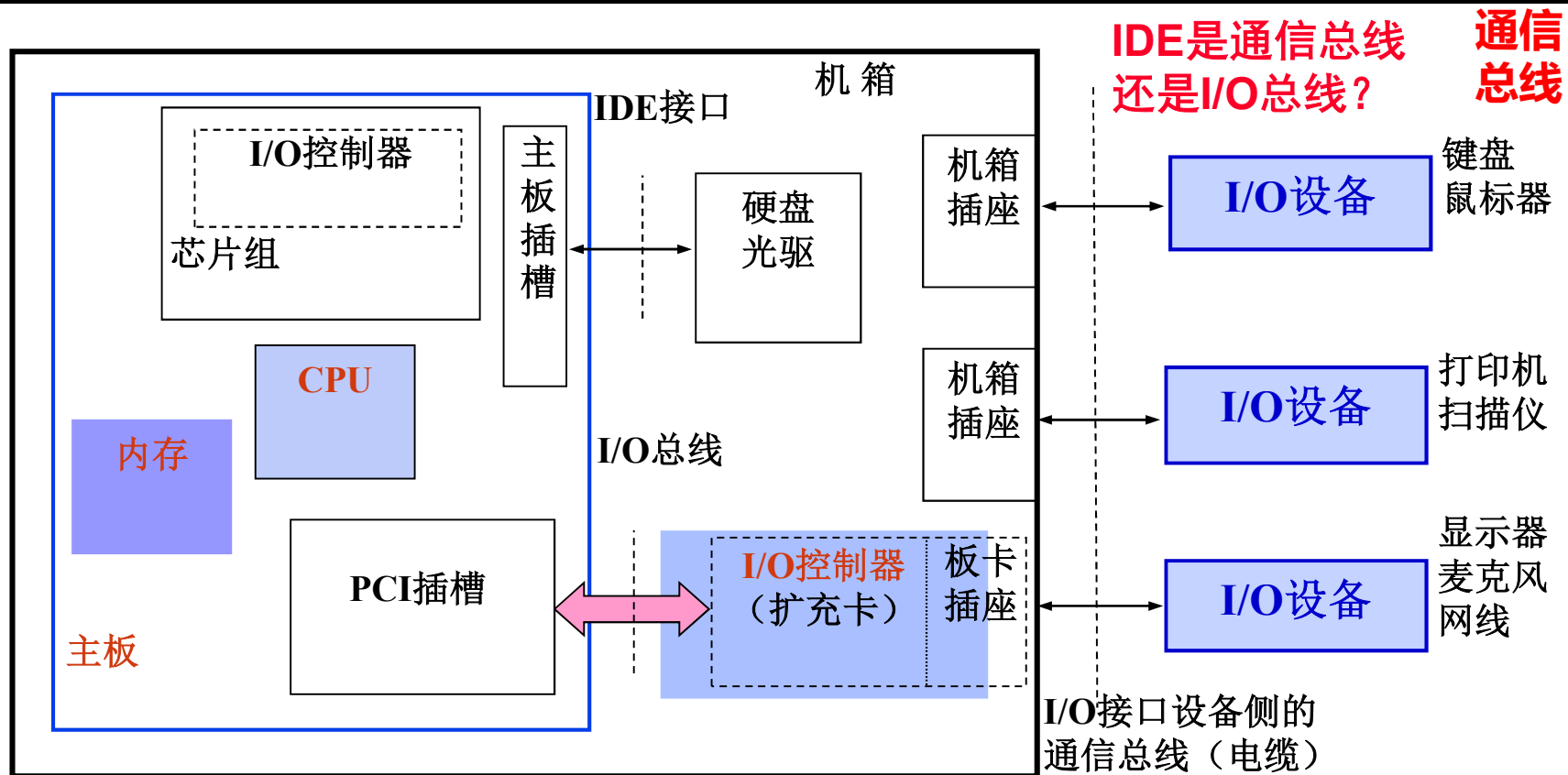
# 基于Core i7系列处理器的互连结构举例



**QPI总线的带宽为： $6.4\text{GT/s} \times 2\text{B} \times 2 = 25.6\text{GB/s}$**

**每个存储器总线的带宽为： $64\text{b}/8 \times 1066\text{ MT/s} = 8.5\text{ GB/s}$  .**

# I/O总线,I/O控制器与I/O设备的关系



- ° **I/O设备**通常是物理上相互独立的设备，它们一般通过**通信总线**与**I/O控制器**连接
- ° **I/O控制器 (I/O接口)**在扩展卡或者南桥芯片内，通过**I/O总线**连接**CPU**和**MM**
- ° **I/O总线**经过北桥芯片与**内存**、**CPU**连接

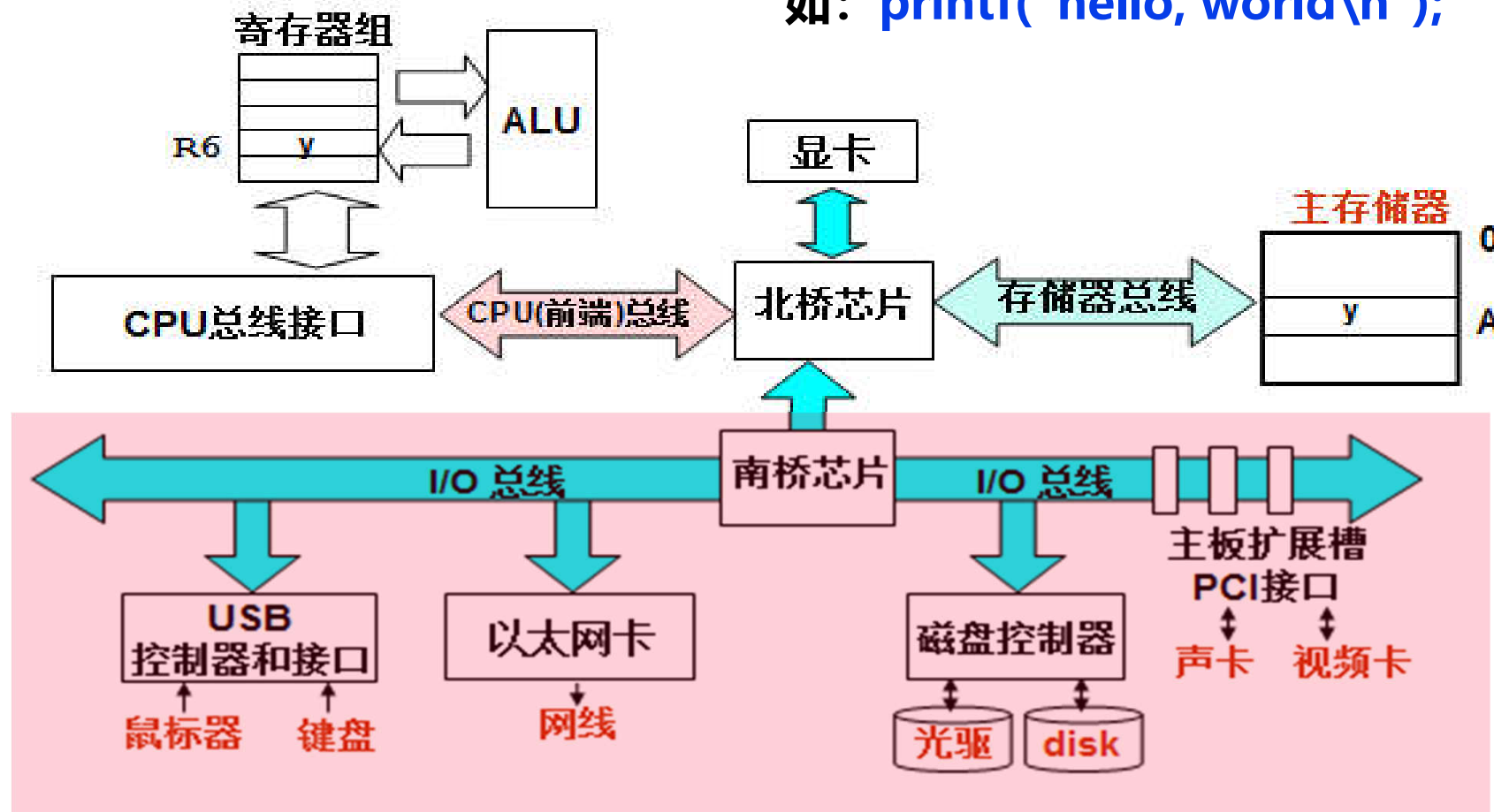
# I/O硬件的组成

I/O硬件建立了外设与主机之间的“通路”：

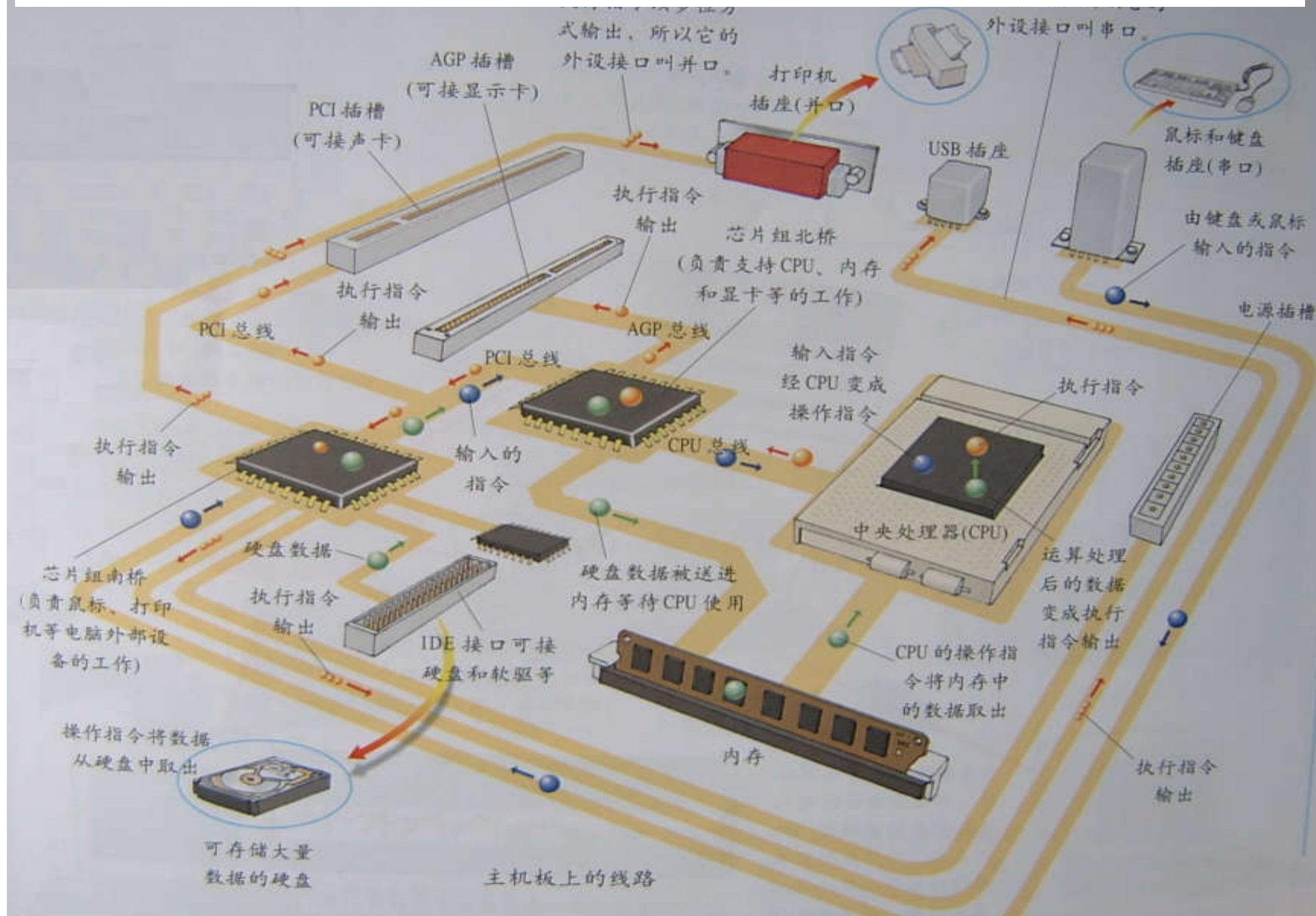
主机----北桥---I/O总线----南桥（设备控制器）----电缆----外设

如何把用户I/O请求转换为对设备的控制命令并完成设备I/O任务，需要I/O软件与I/O硬件之间的协调工作

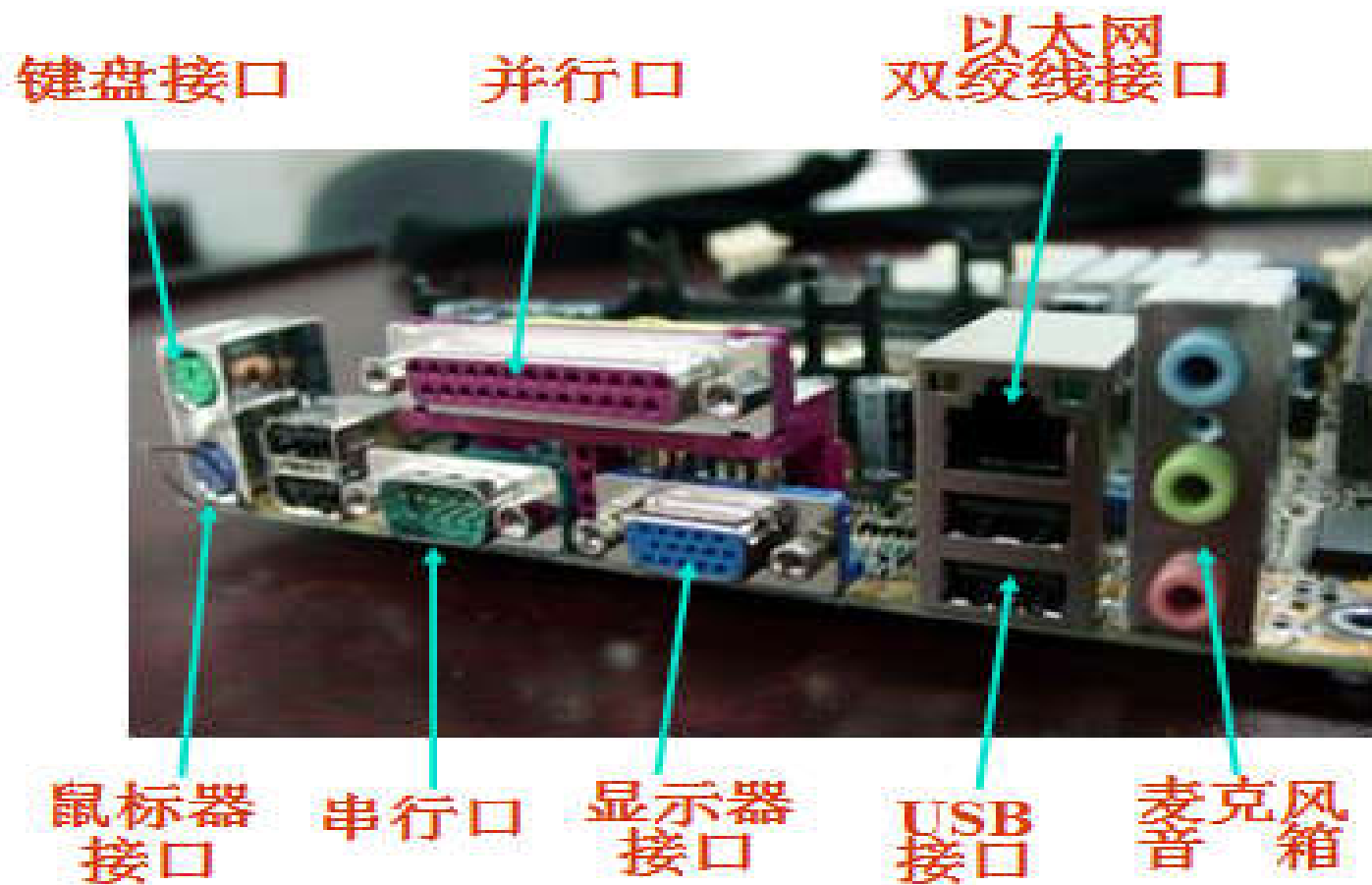
如： `printf("hello, world\n");`



# 主机—北桥--I/O总线--南桥（带连接器设备控制器）--电缆--外设



# 连接外部设备的连接器



主机—北桥--I/O总线--南桥（带连接器设备控制器）--电缆--外设

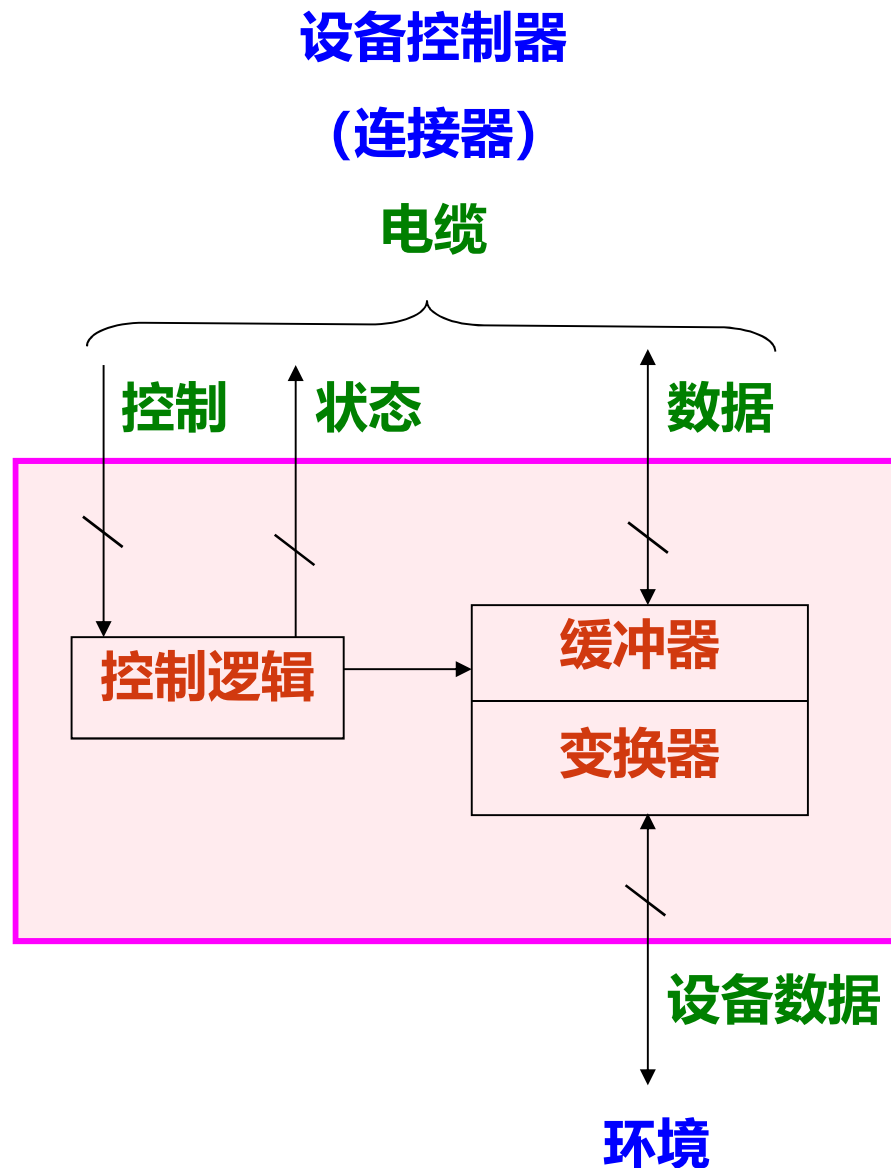
# 外部设备的通用模型

- 通过**电缆**与设备控制器（I/O接口）进行数据、状态和控制信息的传送
- 控制逻辑**根据控制信息控制设备的操作，并检测设备状态
- 缓冲器**用于保存交换的数据信息
- 变换器**用于在电信号形式（内部数据）和其他形式的设备数据之间进行转换

所有设备都可抽象成该通用模型！

设备所用电缆中有三种信号线：

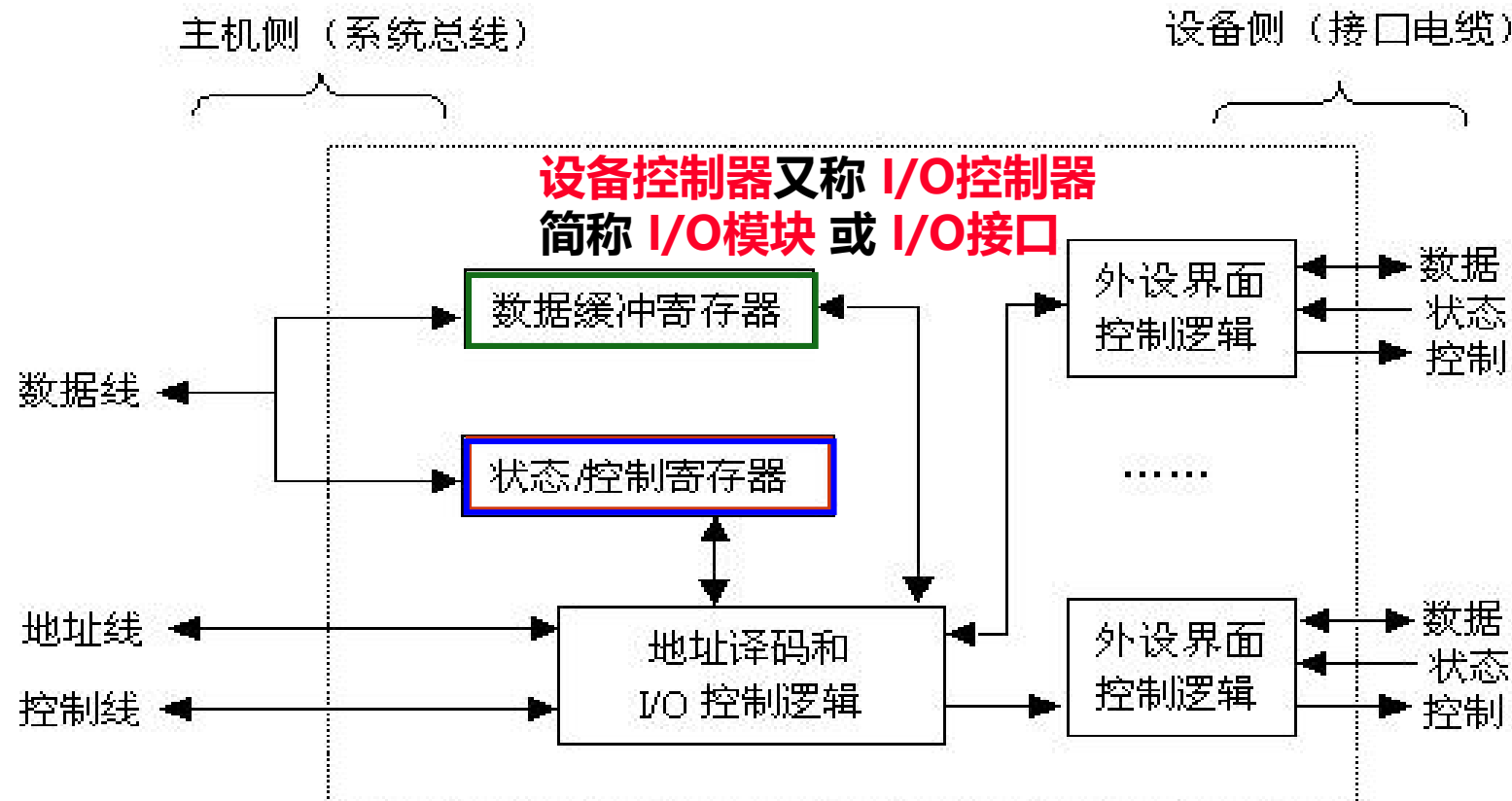
控制信号、状态信号、数据信号





# 设备控制器的结构

- 设备控制器的一般结构：不同I/O模块在复杂性和控制外设的数量上相差很大



通过发送命令字到I/O控制寄存器来向设备发送命令

通过从状态寄存器读取状态字来获取外设或I/O控制器的状态信息

通过向I/O控制器发送或读取数据来和外设进行数据交换

将I/O控制器中CPU能够访问的各类寄存器称为I/O端口

对外设的访问通过向I/O端口发命令、读状态、读/写数据来进行

上传  
下达



# 显卡的外部连接特征

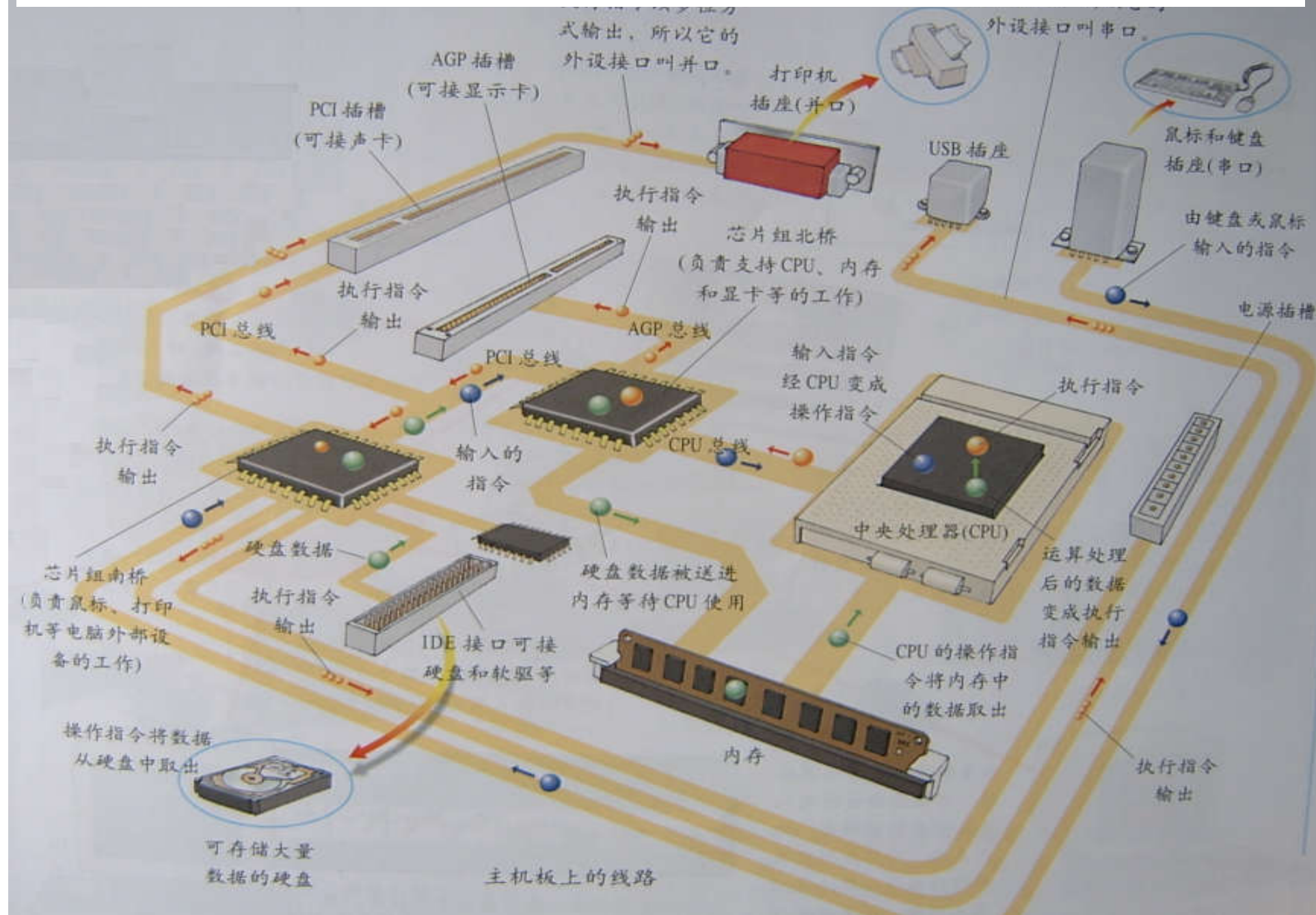


**VGA连接器  
连接到显示器**

**连接到 I/O总线  
(主机侧)**

**将I/O控制器中CPU能够访问的各类寄存器称为I/O端口  
对外设的访问通过向I/O端口发命令、读状态、读/写数据来进行**

# 主机—北桥--I/O总线--南桥（带连接器设备控制器）--电缆--外设



# I/O端口的寻址方式

---

- 对I/O端口读写就是向I/O设备送出命令或从设备读状态或读/写数据
  - 一个I/O控制器可能会占有多个端口地址
  - I/O端口必须编号后，CPU才能访问它
  - I/O设备的寻址方式就是I/O端口的编号方式
- 教室和办公室可以连号  
(统一编址)，也可单  
独编号 (独立编址)

## (1) 统一编址方式 (内存映射方式)

与主存空间统一编址，主存单元和I/O端口在同一个地址空间中。

(将I/O端口映射到某个主存地址区域，故也称“存储器映射方式”)

例如，RISC机器、Motorola公司的处理器等采用该方案

VRAM (显示存储器) 通常也和主存统一编址

## (2) 独立编址方式 (特殊I/O指令方式)

单独编号，不和主存单元一起编，使成为一个独立的I/O地址空间

(因为需专门I/O指令，故也称为“特殊I/O指令方式”)

例如，Intel处理器就是独立编址方式

# 驱动程序与I/O指令

---

- 控制外设进行输入/输出的底层I/O软件是**驱动程序**
- 驱动程序设计者应了解设备控制器及设备的工作原理，包括：**设备控制器中有哪些用户可访问的寄存器、控制/状态寄存器中每一位的含义、设备控制器与外设之间的通信协议**等，而关于外设的机械特性，程序员则无需了解。驱动程序通过访问**I/O端口**控制外设进行I/O：
  - 将控制命令送到**控制寄存器**来启动外设工作；
  - 读取**状态寄存器**了解外设和设备控制器的状态；
  - 访问**数据缓冲寄存器**进行数据的输入和输出。
- 对I/O端口的访问操作由**I/O指令**完成，它们是一种特权指令
- IA-32中的I/O指令：in、ins、out和outs
  - in和ins用于将**I/O端口**的内容取到CPU内的**通用寄存器**中；
  - out和outs用于将**通用寄存器**内容输出到**I/O端口**。

如 **IN AL, DX**: DX中存放I/O端口地址，将I/O端口中的内容取到AL中