冯诺依曼模型：（以运算器为中心），运算器，控制器，存储器，输入输出系统

Cpu功能：中断处理、数计算和加工、指令控制、外部访问、操作控制、时间控制

基本功能：循环的执行指令、处理异常和中断请求

Cpu中的控制部件仅含CU，control unit，就是控制器，包括ID和μOP控制信号生成电路和时序信号生成电路

所以：进行指令译码并产生微操作信号/控制信号的部件

数据通路：指令执行过程中数据经过的路径和路径上的所有部件

PSR：状态寄存器

GPR：通用寄存器

通用寄存器长度=机器字长，可以存储数据和地址

指令地址作循环变量

累加寄存器（AC）：既存放源操作数又存在目的操作数的寄存器

MAR、MDR的好处是，实现外部操作和内部操作的并行，提高CPU性能

专用寄存器由操作系统控制

基本操作的组成要求：操作具有独立性、不依赖其他操作：源数据/结果存放在时序逻辑部件中，操作功能具有原子性，否则不是基本操作

指令译码的结果需要写两个，指令是什么以及控制信号是什么（eg.读GPR，写MEM

执行指令的操作受操作码、PSW、寻址方式影响

指令执行过程的操作定时：时间基准：主时钟脉冲信号

程序的执行方式分为串行方式和流水方式

！！！！！！！！！产生操作控制信号，循环的执行指令

第三章 存储系统

存储系统概述：

分类 技术指标（存取容量、存取时间、存取周期、传输速度）

层次结构存储系统：程序访问的局部性原理（程序运行时，数据和指令访问所呈现的相对簇聚的现象，包括时间局部性和空间局部性）；cache对程序员透明，辅存不透明（有软件参与）。

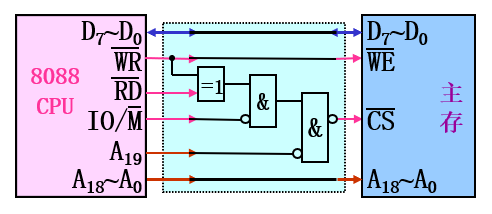
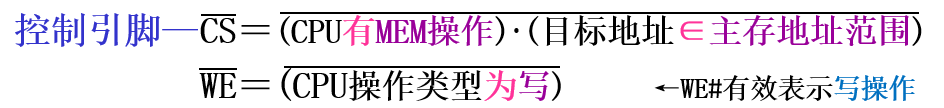
半导体存储技术：

SRAM触发器；DRAM->MOS电容；时序要求【地址先发送；(CS) ̅在地址稳定后有效，(WE) ̅早于/等于(CS) ̅无效(R)/有效(W)；数据(W时)发送时刻无要求、保持时长≥tDW((CS) ̅及(WE) ̅均有效时)】

DRAM行刷新，通过Y译码器输出是否有效来确认【集中式、分散式、异步式】；DRAMC管理所有DRAM芯片，共享刷新电路节约成本、使得不同芯片的读写与刷新并行。

主存储器：

DRAM字扩展可以增加RAS信号线个数来起到片选作用。



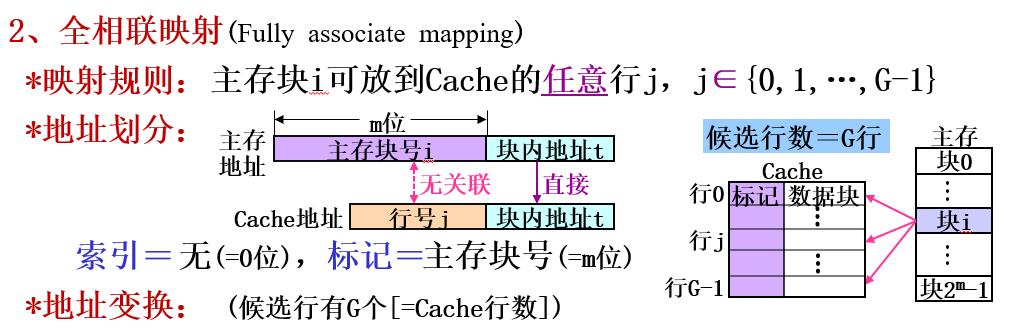
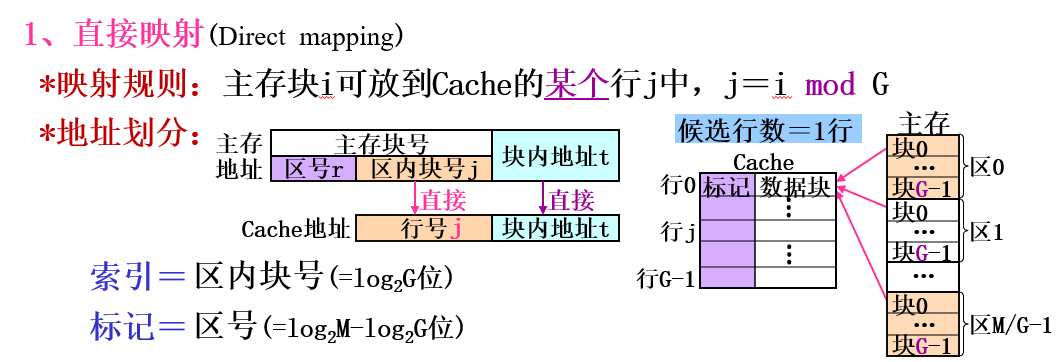
SDRAM：同步DRAM，基于时钟信号，支持突发传送模式；DDRSDRAM：上升沿下降沿分别进行数据传输，单元宽度double，读时分段写入。

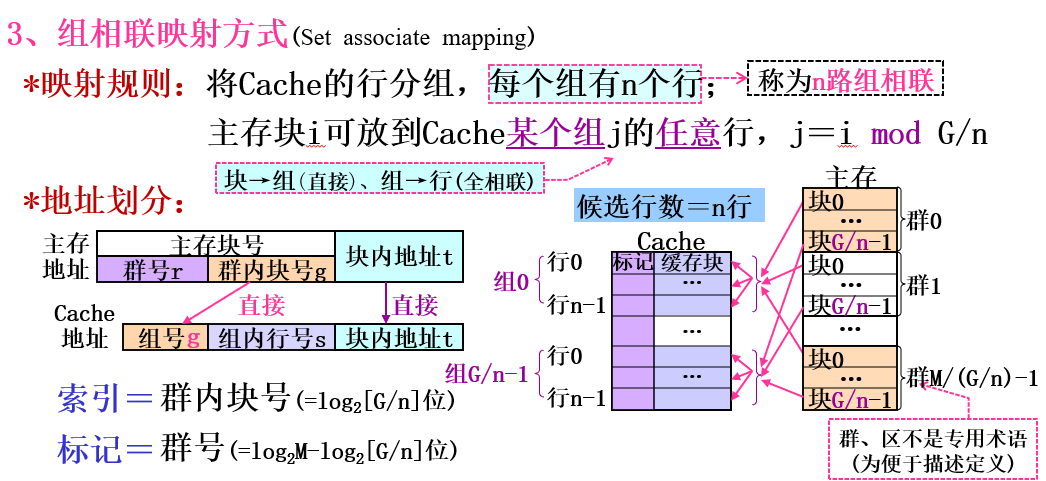
多体交叉存储器：顺序编址【一个存储体中连续存】||交叉编址；交叉访问【轮流启动】、并行访问。

高速缓冲存储器：

Cache存储空间组织：有效位V、标记Tag、状态、缓存块（块号+块内地址）

映射规则：直接映射【】、全相联映射【】、组相联映射【】





替换算法：在查找范围内找牺牲行：随机RAND【】、先进先出FIFO【使用计数器，适用于组相联cache】、最近最少LRU【要增设LRU位，位数等于log行数】

写策略：全写法【写命中，同时写，写缺失，只主存】、写回法【写命中，只cache，写缺失，稍后写，需要一个脏位】

虚拟存储器：

存储器管理单元MMU，

第四章 指令系统

指令系统组成：指令格式：操作码（定长、变长->哈夫曼编码拓展编码）+地址码

操作数存放方式：

寄存器中操作码有一个长度位W，表示长/短数据，指令只有一种数据长度时可以缺省

存储器中LSB和MSB标志，涉及端序（大端：MSB在最低位->数据地址，小端相反）和对齐。

寻址方式：八种【立即（有符号数）、寄存器（速度快、地址码短）、直接（简单但位数多）、间址（多级指针）、寄存器间接、基址（地址码含偏移量，寄存器中存基址，程序重定位）、变址（地址码存放基准地址，变址寄存器存放变址值，如数组）、相对（地址码中有形式地址A，作偏移量，用于PC的指令寻址，记得在PC+1的基础上进行偏移，可以画一个图）】+隐含寻址

**第一单元 计算机系统概述**

存储程序方式：

⑴程序和数据预先存放在存储器中

⑵机器工作时，自动、逐条地取出指令并执行

主机：CPU（运算器+控制器）+MM（主存储器）

外设：输入设备，输出设备，辅助存储器

CPU：运算器，控制器

ALU：算术逻辑单元（只进行整型运算）

FPU：浮点运算部件

CU：Controller Unit，控制单元

PC：Program Counter，程序计数器

IR：Instruction Register，指令寄存器

ID：Instruction Decode，指令译码器

存储元：可存储一个二进制位的元件，存储单元—可同时存储一串二进制位的元件(与一个地址相对应)，存储阵列—所有存储单元的集合

存储单元：一般以8位二进制作为一个存储单元，也就是一个字节

存储阵列：多个存储单元排列形成

存储单元长度：存储字长,一个字节由多少位二进制组成

存储单元地址：每个存储单元被赋予的唯一编号，存储单元长度—一个存储单元能存储的二进制信息位数

存储字：存储单元中存储的二进制信息

存储字长：存储单元中容纳的二进制信息个数

总线：一组用于信息传输的公共信号线

主设备：发送信息（CPU）

从设备：接收信息（存储器等）

I/O接口：实现数据缓冲，格式转换，通信控制，协调总线

系统总线：连接计算机各大部分的总线

DBus：Data Bus，数据总线

ABus：Address Bus，地址总线

Cbus：Control Bus，控制总线

实际机器：所编写程序可悲硬件识别和执行的机器

虚拟机器：除了实际机器的机器

解释：先将程序转换为低级机器上的等效程序，再运行

翻译：对程序中每一条语句，都转去执行低级机器上的一段等效程序

指令：告诉计算机从事某一特殊运算的代码

语句：

计算机结构：硬件的概念性结构和功能特性

计算机组成：计算机结构的逻辑实现

计算机实现：物理结构（硬件）

ISA：Instruction Set Architecture

指令地址: 编程时：指令在程序MEM中的首地址(首个单元)

                 执行时：指令在主存中的首地址(由系统随机分配)

逻辑地址：指令地址

物理地址：运行时在存储器中的地址

机器字长：CPU一次能处理的二进制（整型）位数

时钟周期：CPU主时钟脉冲宽度

主频：主时钟脉冲频率

主存编址单位：

主存单元长度：

主存地址空间：主存为最大容量时的地址空间

CPU可寻址空间：主存地址空间

响应时间：一个任务从提交到完成的总时间（T响应＝TCPU＋T等待）

吞吐率：单位时间能处理的工作量

CPI：Cycles Per Instruction（一条指令所需的平均时钟周期）

MIPS：每秒百万次指令（主频/CPI/10^6）

MFLOPS：每秒百万次浮点运算

**第二单元 数据的表示与运算**

数制：进位计数制，二进制，八进制，十进制，十六进制

码制：

机器数：计算机内部用编码表示的数

真值：数学上的数

原码：最高位为符号位，其余各位为真值的绝对值

补码：正数的补码为本身，负数的补码为取反加一

移码：正数为前加一位1，复数取反加1首位 补0（000的移码为1000）

模：表示机器数的位数

补数：  补码

同余：余数相同

BCD码：二进制数表示十进制的编码

交换码：用于字符的索引、传送 （例：ASCII码）

内码：用于字符的存储和处理

码距：任意两个码字中位值不相同的个数的最小值

校验码：由数据信息和校验信息组成的编码

奇偶校验：检测校验码中1的个数是奇数还是偶数（所有位取异或的结果补在最后）

海明校验：多重奇偶校验，将数据位分为多个有重叠的组，每个组进行奇偶校验

循环冗余校验(CRC)：

数据类型：一个值的集合，及定义在该值集上的一组操作（例：8位二进制及二进制逻辑）

数据表示：

截断运算：保留数据的低位部分

位扩展运算：零扩展，符号扩展

零扩展：无符号数补零

符号扩展：有符号数补符号位

上溢：浮点数的阶大于最大阶码，是真正的溢出，应进行溢出处理（正上溢，负上溢）

下溢：浮点数的阶小于最小阶码，下溢可以认为是机器零不是溢出

机器零：机器零在数轴上表示为0点及其附近的一段区域，即在计算机中小到机器数的精度达不到的数均视为“机器零”，而真值零则表示0这一个点。

最大正数：

最大负数：

规格化数：尾数的最大数为1

左规：尾数左移一位，阶码减一

右规：尾数右移一位，阶码加一

溢出标志：Overflow Flag OF

进位/借位标志：CF

零标志：ZF

符号标志：SF

逻辑左移，右移：Logic 整体移位，移出丢弃，空位补0

算术左移，右移：Arithmetic，有符号数，符号位不变，空位补符号位，其余与逻辑相同。

对阶：将两个进行运算的浮点数的阶码对齐的操作

规格化：进行左规或者右规，使之成为规格化数，提高精度

组合逻辑电路：这还用解释？

时序逻辑电路：这还用解释？

触发器：这还用解释？

锁存器：这还用解释？

全加器：这还用解释？

加法器：这还用解释？

总线互连：

点点互连：

**第三单元 存储系统**

RAM：Random Access Memory 随机存取存储器

ROM：Read Only Memory 只读存储器

SRAM： Static Ram 静态随机存储器

DRAM： Dynamic Ram 动态随机存储器

存取时间：存储器从启动一次存储器操作（读或者写）到完成该操作所需的时间

存取周期：连续进行两次存储器操作的最短间隔

存储器带宽：单位时间里存储器所存取的信息量，也称为存储器在单位时间内读出/写入的位数或字节数

时间局部性：最近访问的信息，不久后将会被再次访问

空间局部性：与最近访问信息相邻的信息不久后会被再次访问

刷新：数据是靠电容特性存储的。由于电容会放电，数据会随时间消失，要维持数据，就要不断的给它充电，这个充电的过程称之为刷新。

行刷新：逐行进行刷新

刷新周期：每个存储元相邻两次刷新的最大间隔时间

集中式刷新：刷新操作集中在一段时间内连续进行，此时读写操作等待（死时间或死区）

分散式刷新：各行的刷新操作分散在各个存取周期中进行，存取周期前一半用于读写，后一半用于刷新，不存在死区但存取周期加倍

异步式刷新：各行的刷新操作均匀分散在刷新周期中进行，每过一段时间就进行一次刷新操作，当刷新周期结束时正好刷新一遍，死区总和为一个刷新周期。

DRAMC：DRAM控制器，管理DRAM芯片，共享刷新电路，使不同芯片刷新和读写并行，消除冲突

位扩展：多个存储器并联增加存储字长（16位分为4个4位），适用于数据长度不够

字扩展：多个存储器串联增加存储字数，适用于数据数量不够

字位扩展：同时进行子扩展位扩展

有效逻辑：片选引脚的信号

SDRAM：原有的DRAM都属于异步器件，在访问过程中，地址和数据信号需要一直保持，CPU和DRAMC在访问完成前都需要等待，而SDRAM使同步器件，在确定的几个时钟周期后会给出响应，CPU和DRAMC无需等待。

DDR SDRAM：Double Data Rate SDRAM，双倍速率SDRAM，在上升沿和下降沿分别进行数据传输，DDRX表示存在2X倍I/O的存储单元宽度，一个时钟周期存入更多的数据。

常规传输：异于突发传输

突发传输：在一个数据被访问后，连续传输多个数据

RAM的工作频率：I/O频率/X（DDRX）

I/O频率：工作频率×X（DDRX）

顺序编址：连续编址，同一存储体的存储单元地址是连续的

交叉编址：模m编址，同一存储体的存储单元地址差m（m为存储体数）

交叉访问：轮流访问各个存储体，多个数据分时I/O

并行访问：同时访问多个存储体，多个数据同时I/O

**命中与缺失：访存信息已经在Cache种，称为命中；不在为缺失**

**命中率：命中次数与访存总数的比值，用H表示**

**命中时间：在命中时，Cache完成访存操作所用的全部时间**

**缺失开销：在缺失时，Cache先与主存交换信息，再完成访存操作的所用时间，由和主存访问时间构成。**

**平均访问时间：**

块：Cache与主存交换信息的单位，由多个存储字组成

主存块：主存中的区域

缓存块：Cache中的区域

行或槽：存放缓存块和相应管理信息的存储空间（和块相对应）

标记：表示Cache行中的数据来自哪个存储块，标记记录主存块号的一部分

有效位：表示行内数据是否有效，V=0表示空闲（是否有数据）

目录表：所有行的管理信息

数据区：所有行的块数据信息

Cache容量：Cache数据区的容量

Cache总容量：Cache数据区、管理区容量之和

**候选行：可以被覆盖的行**

**牺牲行：无空闲行需要被覆盖的行**

**目标行：**

**空闲行：V=0的行**

**地址映射规则：地址空间映射的方法**

**替换算法：从候选行找出牺牲块的方法**

**写策略：Cache将CPU所写数据写回主存的时机和方法**

直接映射：相应的块只映射到相应的行，,主存块i可放到第j行的映射函数，G表示Cache行数

全相联映射：一个主存块可以放到任意的行里

组相联映射：直接映射和全相联映射的结合，将行分为多个组，对应着相应的块，组内全相联映射

相联度：候选行数，一个主存块能映射到的Cache行数

**RAND：随机，随机选择一个牺牲块**

**FIFO：先进先出，最早进入的块为牺牲快**

**LRU算法：最近最少使用，近期最少使用的块为牺牲块，不会降低命中率**

**LRU位：计数值，用于表示访问的顺序，在Cache行管理信息中**

全写法：写命中时，将数据写入Cache，同时写入主存；写缺失时，直接将数据写入主存，不将目标块调入Cache

写回法：写命中时，数据只写入Cache，不写入主存；写缺失时，目标块调入Cache，数据只写入Cache；

按写分配法：写回法

不按写分配法：全写法

脏位：在Cache管理信息中，表示改行是否被改写过

**哈佛结构：将Cache组织成指令Cache与数据Cache，解决指令和数据的并行访问冲突**

**冯·诺依曼结构：单个存储器的结构**

**分离Cache：采用哈佛结构的Cache**

**联合Cache：指令和数据混存的Cache**

程序地址：程序中的MEM地址，MEM编址单位同主存、从0开始编址

程序地址空间：空间很大，位数固定？

地址变换：将程序逻辑地址转换为物理地址

MMU：存储器管理单元

分区管理方式：主存划分成若干分区，每个进程分配有1个分区(空间连续)

分页管理方式：进程划分成若干大小相等的页()，主存划分成若干同等大小的页框()，每个进程一次性分配到所需的页框(可不连续)

页：每个进程所占空间被分成的若干大小相等的区域

页框：主存空间被划分成的若干大小相等的区域

页表：管理页和页框的映射关系

页表项：装入位、修改位、访问位、修改位、保护位等

虚拟存储器(VM)：以透明方式为程序提供的、比主存空间大得多的存储空间

虚拟地址：虚拟存储器存储单元地址

虚拟地址空间：虚拟存储器地址空间

**段式VM：虚存空间按程序结构划分成若干个段，主存空间以段为单位进行分配**

**页式VM：虚存、主存空间按页大小划分成若干个页，主存空间以页为单位进行分配**

**段页式VM：虚存空间先分段、再分页，主存空间只分页，主存空间以页为单位进行分配**

**虚页号：**

**实页号：**

**页表基：**

**TLB：快表，后备转换缓冲区**

**第四章 指令系统**

**机器指令：硬件可直接识别和执行的命令**

**指令系统：所有机器指令的集合，即指令集**

**指令格式：指令中所有信息(操作及操作数)的编码格式**

源操作数：用存放部件的内容表示

目的操作数：存放部件本身表示

跳转：无条件转移指令，在任何条件下都能进行转移操作

分支：条件转移指令，在条件满足时才进行转移操作

**操作码：用编码表示指令的格式和操作类型**

**地址码：用编码表示指令的各个操作数地址及下条指令地址**

**规整性：**

**平均码长：**

指令条数：

定长编码：长度不可变的操作码

变长编码：长度变化的操作码

扩展编码：随时加长的操作码

指令字长：指令的长度

单地址指令：指令中只有一个地址

双地址指令：指令中有两个地址

单字长指令：指令长度为一个机器字长

双字长指令：指令长度为两个机器字长

定长指令字结构：所有指令的字长都相等

变长指令字结构：与上相反

**大端：数据的最高有效字节MSB存放在最低地址单元**

**小端方式：数据的最低有效字节LSB存放在最低地址单元**

**不对齐：数据可从任意位置开始存放**

**边界对齐：数据地址为数据长度的倍数**

**4字节对齐方式：**

寻址方式：根据地址码形成操作数地址或指令地址的方法

有效地址(EA)：指令中的存储单元地址

顺序寻址：下一条指令由PC+1形成（Program Counter）

跳跃寻址：下条指令由当前指令的地址码形成

立即寻址：OPD放在指令REG中，地址参数为OPD本身

寄存器寻址：OPD放在数据REG中，地址参数为数据REG号

直接寻址：OPD放在MEM中，地址参数为OPD的EA

间接寻址：OPD放在MEM中，地址参数为存放OPD的EA的EA（把OPD的EA存入MEM）

寄存器间接寻址：OPD放在MEM中，地址参数为存放OPD的EA的地址REG号（把OPD的EA存入REG）

变址寻址：OPD放在MEM中，地址参数为变址REG号及形式地址(基准地址)

基址寻址：OPD放在MEM中，地址参数为基址REG号及形式地址(偏移地址)

相对寻址：指令放在MEM中，地址参数为形式地址(偏移量)

隐含寻址：通过操作码形成地址

偏移地址：

偏移量：

地址寄存器：

数据寄存器：

通用寄存器(GPR)：

CISC：复杂指令系统计算机

RISC：精简指令系统计算机

**第五章 中央处理器**

**BIU：总线接口单元**

**中断机构：**

**MAR：存储器地址寄存器**

**MDR：存储器数据寄存器**

**通用寄存器组(GPRs)：即可存放数据又可存放地址**

**状态寄存器(PSR)：存放程序运行的状态**

**PSW：状态程序字，运算结果标志和执行方式标志**

指令周期：CPU取出一条指令并执行所需的时间

中断周期：中断响应所需要的时间

主时钟脉冲：CPU的脉冲

单周期CPU：指令周期为一个时钟周期的CPU

多周期CPU：指令周期多个时钟周期

CPU内部的4种基本操作：取数指令LD，存数指令ST，减法指令SUB，分支指令JNZ

**数据通路：指令执行过程中数据所经过的部件及其路径**

**总线结构：多个部件输出端通过同一信号线连接其他部件输入端**

**点点结构：每个输入端通过不同信号线连接其他部件输出端**

**微操作(μOP)：CPU内的原子操作，**

**μOPCmd：微操作命令**

**节拍：完成一个μOP的时间**

**μOPCmd序列：多个μOP通过不同的节拍信号进行时序控制形成的序列**

**状态转换图：**

硬布线CU：采用有限状态机方法来描述CPU工作流程中所需要的μOP信号

微程序CU：采用微程序方法来描述CPU工作流程中所需要的μOP信号

机器周期：CPU周期

工作脉冲：每个节拍内需要同步脉冲配合工作

同步控制：各μOP的时序只受统一的基准时钟信号(主时钟脉冲信号)控制

异步控制：各μOP的时序只受专门的联络信号(应答信号)控制

联合控制：各μOP的时序受基准时钟信号及联络信号的共同控制

有限状态机：由一组状态组成，每个状态有一组动作（输出），不同状态之间由事件（输入）触发转换

**微命令：每一条微命令对应一组μOP**

**微指令：按一定格式编排、用二进制编码表示、可同时执行的一组微命令**

**微程序：微程序由若干微指令组成，来执行每一条指令**

**CS：控制存储器**

**微指令周期：从CS种取出一条微指令并执行的全部时间**

事件：改变程序正常执行顺序的特殊情况

异常：由CPU内部执行指令所引发的意外

中断：CPU外部的设备产生的请求事件

响应：CPU从当前程序转到处理程序的过程

处理：CPU执行处理程序处理事件的过程

返回：CPU从处理程序返回当前程序的过程

故障：可能修复的异常

陷阱：预先安排的异常

终止：不可修复的异常

可屏蔽中断：可以暂不处理（稍后处理）的中断

不可屏蔽中断：必须立即处理的中断

中断允许位(IF)：标记CPU是处于屏蔽中断（0）还是允许中断（1）状态

**断点：事件的返回地址**

**PSW：程序状态字**

**处理程序入口地址：**

**非向量方式：所有事件共用一个处理程序，处理程序的入口是固定的**

**向量方式:每个事件都有一个处理程序，所有处理程序地址存放在一个管理表中（中断向量表）**

**IVT：中断向量表**

流水线的段：每个操作阶段称为段

拍：一个段所需要的时间，通常为一个时钟周期

填入：填入段

流水：段在处理

排空：段处理结束，输出

段间寄存器：

冒险：流水线在某些状态下无法正确执行后续指令

**流水线的吞吐率：单位时间流水线所完成任务或输出结果的数量**

**加速比：采用流水方式执行和采用串行方式执行的速度之比**

**效率：部件使用时间与整个运行时间的比值**

**多功能流水线：能完成多种功能的流水线，例如指令流水线**

**动态流水线：各个段可在流水时连接成不同功能**

**线性流水线：串行连接的**

**顺序流水线：流入和写出顺序完全相同**

**标量流水线：处理标量数据**

结构冒险：争用硬件资源，引起流水线停顿

数据冒险：指令所需数据不可用，引起流水线停顿

控制冒险：指令执行顺序发生改变，引起流水线停顿

RAW冒险：Read After Write尚未写入引起的数据冒险

load-use冒险：加载使用数据冒险

阻塞：使冲突指令及其后续指令停顿

转发：产生数据的段可把数据送到下一个段或较早的段

乱序执行：让后续无冒险的指令先执行

分支预测：预测分支指令的转移方向，并执行该方向的指令，预测错误则执行另一方向的指令

延迟分支：

气泡：nop指令不会改变任何信息

延迟槽：

**超级流水线：增加流水线的段数，增加并行，提高指令输出速度**

**超标量流水线：CPU中有多条流水线，多个指令并行执行**

**VLIW流水线：VLIW指令包含多个操作字段，相当于多个标量指令**

**第六章 总线**

**总线事务:总线上一堆设备之间进行一次信息交换**

同步总线：

异步总线：

并行总线：

串行总线：

片内总线：芯片内部的总线

系统总线：内部连接CPU、主存、外设等部件的总线

通信总线：连接 主机与外设之间的总线USB

信号电平：

单端方式：

差分方式：

信号线复用：

**总线宽度：总线位数**

**总线带宽：最大传输率**

**总线工作频率：时钟频率**

总线操作：总线上完成一次数据传输所需的操作，比总线事务多了总线仲裁

总线仲裁：多个主设备发出总线请求信号，确定总线归属权

总线周期：传输周期+空闲（仲裁）

总线传输周期：传输数据的周期

地址期：读取地址

数据期：传输数据

隐藏式仲裁：申请及分配阶段安排在总线传输周期结束阶段重叠

**集中式仲裁：有统一的总线仲裁器**

**分布式仲裁：总线仲裁逻辑分散在各个主设备中**

**链式查询方式：自动轮询各个主设备，被询问时可获得使用权**

**计数器定时查询方式：轮询，避免链式查询的断链现象**

**独立请求方式：根据请求线的连接次序仲裁**

全互锁：主/从设备发出信号后，都必须等对方撤销信号后才撤销自己的

半互锁：主设备发出信号，从设备发出信号后，主设备撤销信号

不互锁：主设备和从设备各自约定时间撤销信号

**总线标准：总线连接和传输信息时，需要遵守的协议和规定**

**ISA：**

**PCI：**

**USB：**

**QPI：**

**前端总线(FSB)：**

**主机总线HOST：**

**I/O总线：**

**存储器总线：**

**总线桥：**

**南桥：**

**北桥**：

**第七章 输入/输出系统**

**I/O指令：操作码、设备码、命令码**

**I/O端口地址：I/O指令中的设备码**

**统一编址：主存单元、I/O端口共用一个地址空间**

**独立编址：主存单元、I/O端口都从零开始编址**

**设备选择电路：监视总线状态，有I/O事务时，比较总线上地址与所存设备号**

**条件传送方式：先启动设备，等到设备就绪时，才能进行传送，又称异步传送方式**

**无条件传送方式：可随时进行数据传送，又称同步传送方式**

**字符设备：每次传输都为一个字符的设备**

**块设备：传输数据块的设备**

**I/O方式：指主机对数据传送的管理方式**

**程序查询方式：CPU启动设备后，不断地查询设备状态，当设备准备就绪时，才进行数据传送**

**程序中断方式：CPU启动设备后，继续执行现行程序；设备准备就绪时，向CPU提出请求； CPU响应请求，传送数据，返回现行程序**

**DMA方式：Direct Memory Access，CPU发送传输需求、启动设备后，继续执行现行程序；DMA接口实现数据传送(1批，几千个，适用于块设备)，结束时向CPU提出请求；CPU响应请求，处理结束事宜，返回现行程序**

**通道方式：CPU执行访管指令后，继续执行现行程序；通道执行通道程序(主存中)，结束/异常时向CPU提出请求；CPU响应请求，处理相应状况，返回现行程序**

存储密度：Ds＝道密度×位密度

数据传输率：Dr＝磁道容量（位数/道）×磁盘转速

寻址时间：磁头从起始位置移动到目的位置的全部时间

磁道：

扇区：

RAID：磁盘阵列

条带：映射的数据单位

CD：

DVD：

**PS/2：**

**IDE：**

**SATA：**

**VGA：**

**HDMI：**

I/O接口：主机-外设间的连接电路，负责中转各种信息

I/O端口：可与数据总线交换信息的寄存器 (数据口/控制口/状态口)

数据口：

状态口：

控制口：

并行接口：接口-外设间同时传送n位数据

串行接口：接口-外设间同时传送1位数据

可编程接口：可通过软件选择接口的当前功能

查询接口：CPU控制数据传送、状态查询（字符设备）

中断接口：CPU控制数据传送，接口实现状态报告（字符设备）

**独占查询：启动设备后，立即开始查询状态、直到就绪**

**定时查询：启动设备后，稍后开始查询状态、直到就绪**

向量中断：识别中断类型，获得事件处理程序入口地址，有各自的中断处理程序

非向量中断：同上，共用中断处理程序

单重中断：中断处理过程中不再处理新的中断请求

多重中断：中断处理过程中若遇到更紧急的中断，转去执行新的中断程序

中断类型号：（可不提供）

IVT：中断向量表

中断向量地址：

中断向量：存储中断程序入口地址

串行判优：硬件自动轮询，查询次序→优先级(静态)[共用请求，向量型]

并行判优：硬件算法控制，算法→优先级(动态)[独立请求，向量型]

软件判优：软件查询，查询次序→优先级(静态)[[共用请求，非向量]

**CPU停止访问方式：DMA接口在首个数据准备传送时，申请总线使用权；在所有数据传送结束时，放弃总线使用权**

**周期挪用方式：DMA接口在外设准备就绪时，请求总线使用权；在等待外设准备时，放弃总线使用权**

**预处理：CPU使用I/O指令设置传送参数、启动外设**

**后处理：CPU响应中断请求，完成结束处理工作(如数据校验、开始下次)**