

计算机系统概述

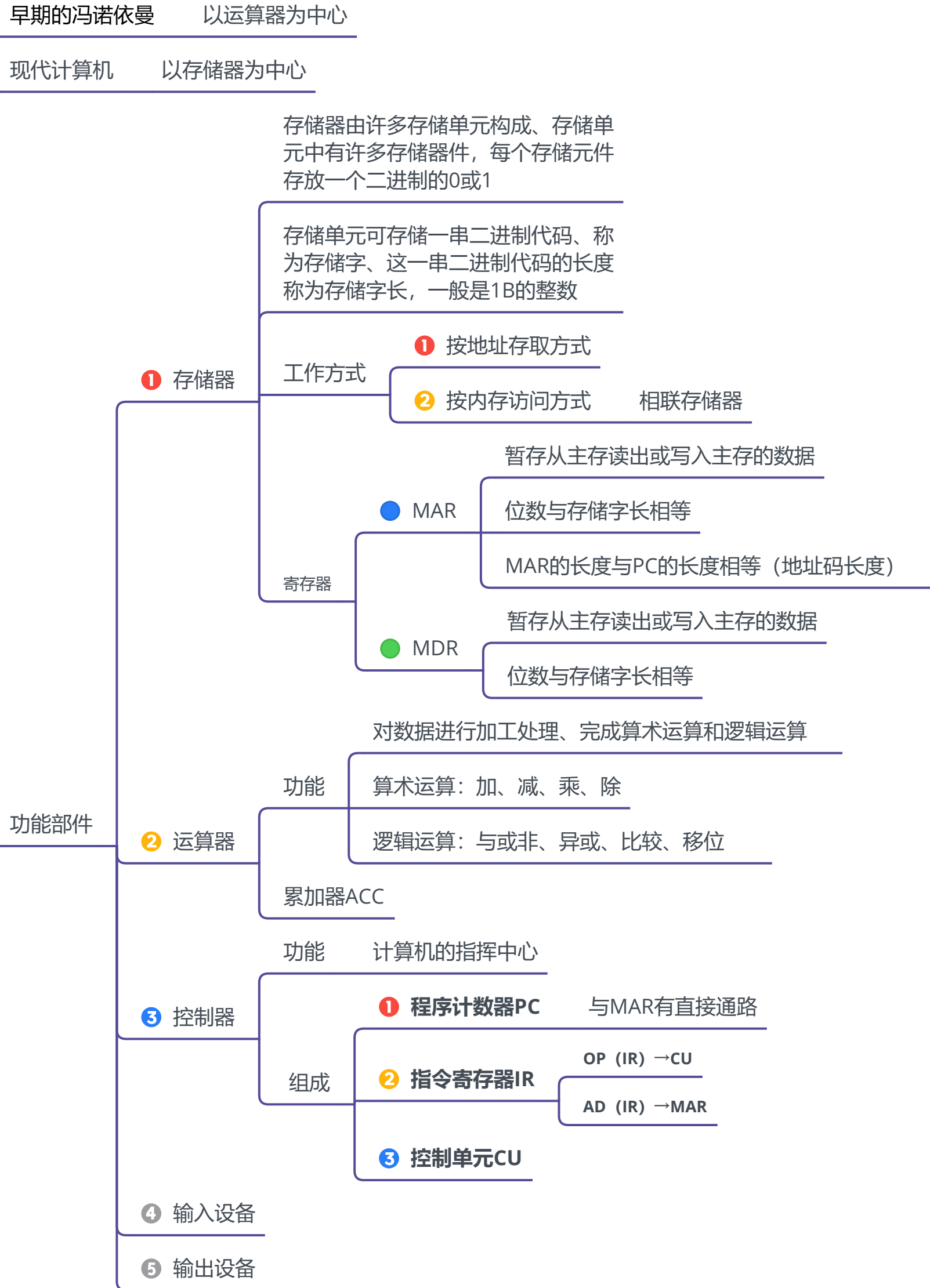


制图：抓码团队 微信公众号：抓码计算机考研

计算机的发展历程



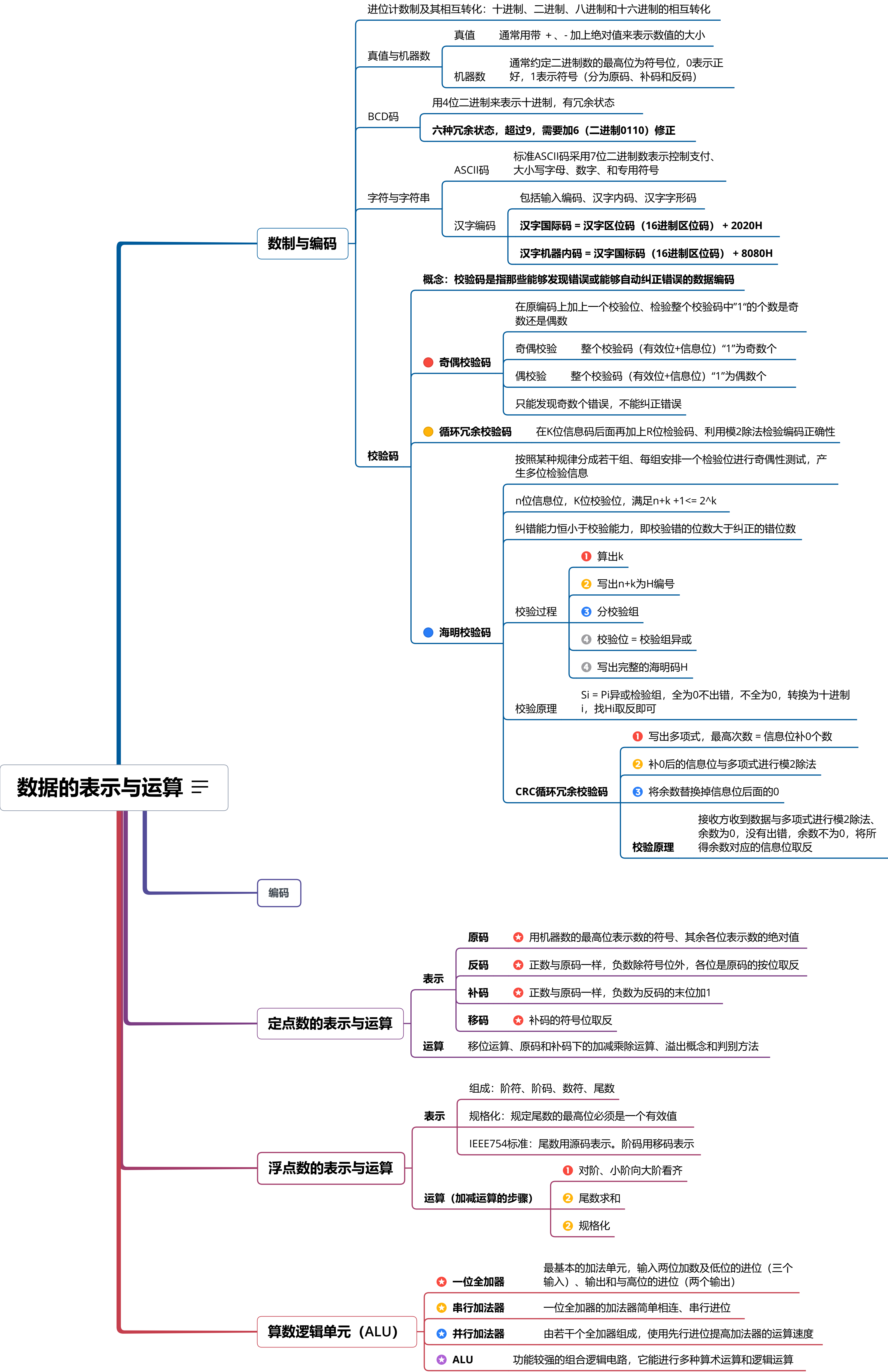
计算机的组成



计算机系统层次结构





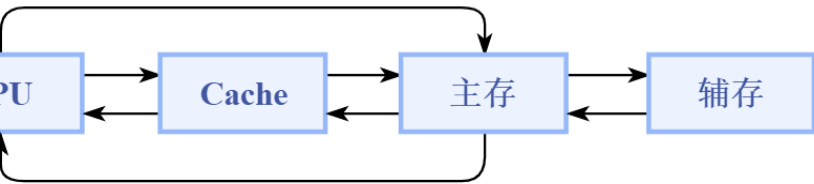


存储系统

存储器的分类

- 按在计算机中的作用
- 按存储介质
- 按存取方式

主存储器（简称主存或内存）、辅助存储器（简称辅存或外存）、高速缓冲存储器（Cache）



存储器的层次化结构

半导体存储器

组成 数据线、地址线、存储矩阵、译码驱动、片选线、读/写电路、读/写控制线

分类

- RAM
 - SRAM 原理：利用双稳态触发器来记忆信息，一般用来做高速缓冲存储器
 - DRAM 原理：利用存储元电路中栅极电容上的电荷来存储信息，需要定期刷新，一般用来做大容量主存系统
 - 集中：用一段固定的时间依次对存储器的所有行逐一刷新
 - 刷新
 - 分散：把对每一行刷新的时间分散到各个工作周期中去
 - 异步：把每行刷新分散到一整个刷新周期中去
- ROM
 - MROM 在芯片制造商生产过程中直接写入，以后任何人都无法改变其内容
 - PROM 允许用户用专门的设备写入程序，写入后的内容无法改变
 - EPROM 允许用户写入程序，用户可以对其内容进行多次改写；需要修改时，要将其全部擦除（不可局部擦除）
 - EEPROM 和EPROM运行原理一样，但是既可以局部擦除，又可以全部擦除
 - Flash（闪速存储器） 在不加电时仍可长期保存信息且能进行快速的擦除重写

主存与CPU的连接

扩展

- 1 位扩展：将芯片地址、片选和读写控制端相应并联，数据端分别引出。增加存储字长
- 2 字扩展：将芯片的地址、数据、读写控制线相应并联、片选译码选择相应的片/片组。增加存储单元个数
- 3 字扩展位：既增加了存储单元个数，又增加存储字长

双口RAM和多模块存储器

双端口RAM：有两个独立的端口，允许两个独立的控制器同时异步地访问存储单元

多模块存储器 单体多字：一次并行读出多个字，地址必须顺序排列并且处于同一单元

多体并行

- 1 高位交叉编址（顺序方式）：高位地址为体号、低位地址为体内地址
- 2 低位交叉（交叉方式）：低位地址为体号、高位地址为体内地址

高速缓冲存储器Cache

引入目的：解决CPU与主存速度不匹配的矛盾

映射方式

- 1 直接映射：主存数据块只能装入到Cache中唯一的位置
- 1 全相联映射：可以把主存数据块装入Cache中的任何位置
- 3 组相联映射：将Cache分为若干组、组间直接映射、组内全相联映射

替换算法

- 1 先进先出（FIFO）算法：选择最早调入的块进行替换
- 2 近期最少使用（LRU）算法：选择近期内长久未访问的块进行替换
- 3 最不经常用：将一段时间内被访问次数最少的存储行换出
- 3 随机算法：随机确定被替换Cache块

写策略

- 1 直写法（写直达法）：写操作时只把数据同时写入主存和Cache
- 2 写回法：写操作时只把数据写入Cache，而不写回主存，只有当Cache数据被换出时才写回主存

虚拟存储器

引入目的：解决主存不足的问题

基本分类

- 1 页式存储器 以页为基本单位。主存空间和虚拟空间都划为若干大小相等的页。
- 2 段式存储器 以段为基本单位。讲主存按段分配，各段的长度因程序而异。
- 2 段页式存储器 讲程序按照其逻辑结构划分段，每段再划分为若干页；主存空间也划分为若干同样大小的页。段式和页式存储的结合。但是要经过两级查表才能完成地址转换，比较费时
- 3 快表（TLB） 讲当前最常用的信息放在一个小容器的高速存储器中，构成快表。快表扮演的角色是作为页表的Cache，对快表的查找和管理全部用硬件来实现。

组成部分

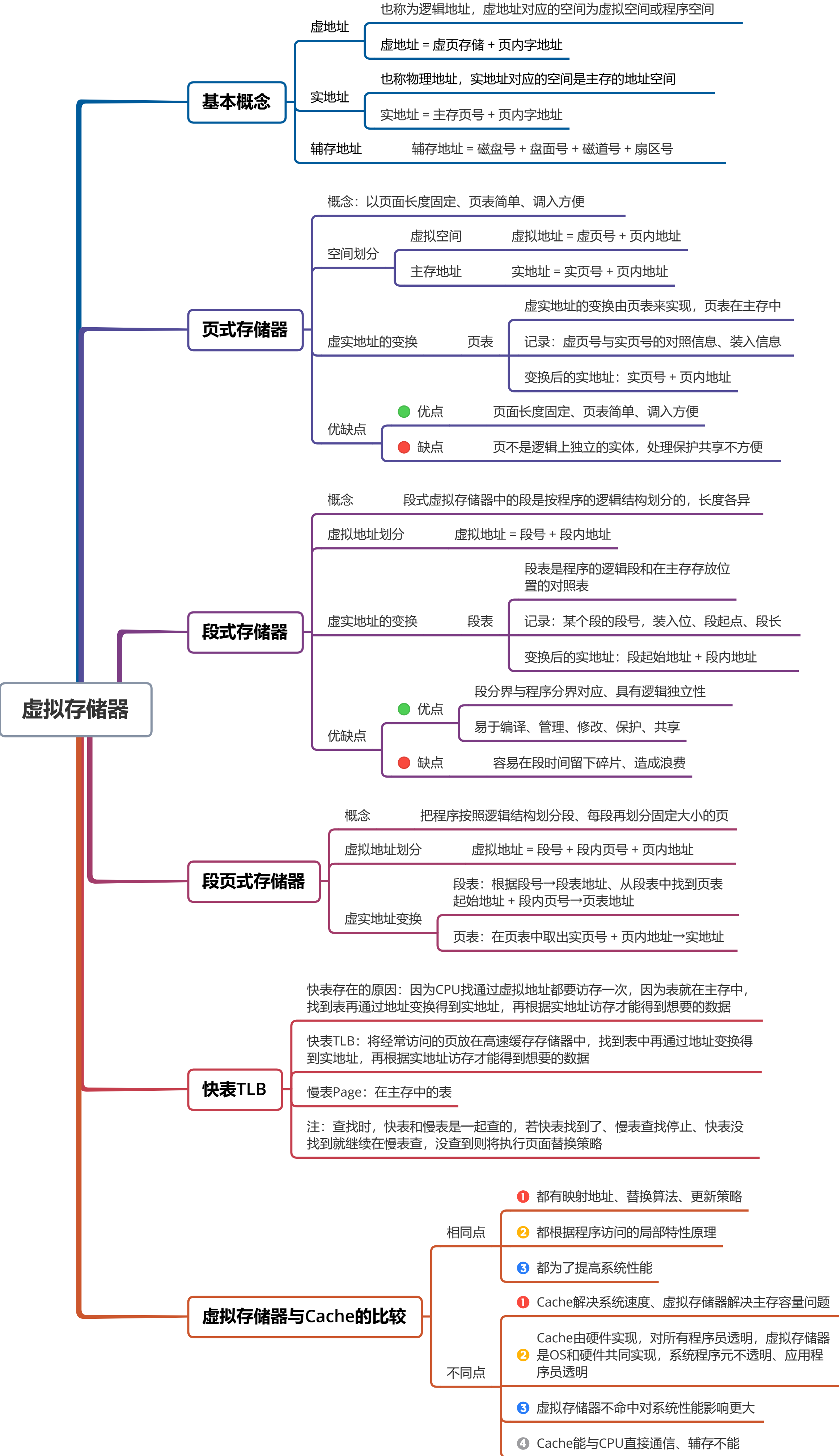
- 1 页表机制：通过查表获取相关信息
- 1 中断机制：要访问页不在内存时产生缺页中断
- 3 地址变换机构：把逻辑地址变换成物理地址
- 4 内存和外存：需要一定容量的内存和外存的支持

置换算法

- 1 OPT：选择以后不再用的页
- 2 FIFO：选择最先装入的页面
- 3 LRU：选择最近最少使用的页
- 4 CLOCK：选择最近未使用的页
- 5 改进型CLOCK：考虑页面修改问题

地址翻译：TBL→页表（TBL不命中）→Cache→主存→CPU



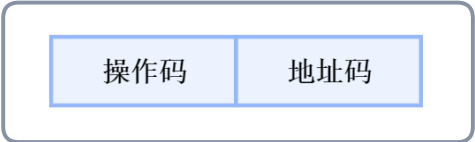


指令系统

指令格式

概念：一条指令就是机器语言的一个语句，它是一组有意义的二进制代码。

基本格式	<div>操作码</div> <div>指出指令中该指令应该执行什么性质的操作和具体有何种功能。如加法、减法、移位等</div>
	<div>地址码</div> <div>也称为操作数字。用来指明改指令的源操作数的地址、结果的地址以及下一条指令的地址。</div>
定长操作码	在指令的最高位部分分配固定的若干位（定长）表示操作码
★ 扩展操作码	全部指令的操作码字段的位数是不固定的



寻址方式

根据指令中给出的地址码字段寻找有效地址的方式。指令中地址码给出的地址称为形式地址（用字母A表示）。这个地址有可能不能直接用来访问主存

有效地址

指能够直接访问主存的地址（用字母EA表示）。形式地址经过某种寻址方式的转换才能变为有效地址。

形式地址

寻址方式

有效地址

指令寻址方式

★ 顺序寻址

通过程序计数器PC加1，自动形成下一条指令的地址

★ 跳跃寻址

通过转移指令直接或间接给出下一条指令的地址

★ 数据寻址方式

立即寻址

操作数本身设在指令内。给出的不是操作数地址，而是操作数本身

直接寻址

指令中地址字段给出的地址A的操作数就是有效地址，即形式地址等于操作数的真实地址：EA = A

隐含寻址

操作数地址不明显给出，而是隐含在操作码中或某个寄存器中

间接寻址

直接给出操作数有效地址所在的存储单元的地址

寄存器寻址

直接给出操作数所在的寄存器的编号

寄存器间接寻址

给出的操作数放在主存单元地址的某个寄存器编号，该寄存器编号中的地址才是有效地址。

基址寻址

将基址寄存器中的内容加上指令格式中的形式地址才是有效地址

偏移寻址

变址寻址

将变址寄存器中的内容加上指令格式中的形式地址才是有效地址

相对寻址

把程序计数器PC的内容加上指令格式中的形式地址才是有效地址

堆栈寻址

从规定的栈中取出操作数

CISC和RISC的基本概念

CISC（复杂指令系统计算机）	指令数目多、字长不固定；寻址方式多、寄存器数量少、一般为微程序控制
RISC（精简指令系统计算机）	指令书目少，字长固定；寻址方式少，寄存器数量多，一般为组合逻辑控制

