

存储单元:多个存储元的集合

一般应具有存储数据和读写数据的功能，以8位二进制作为一个存储单元，也就是一个字节。每个单元有一个地址，是一个整数编码，可以表示为二进制整数。

4K × 8位 的存储器：4K是字数。8是位数

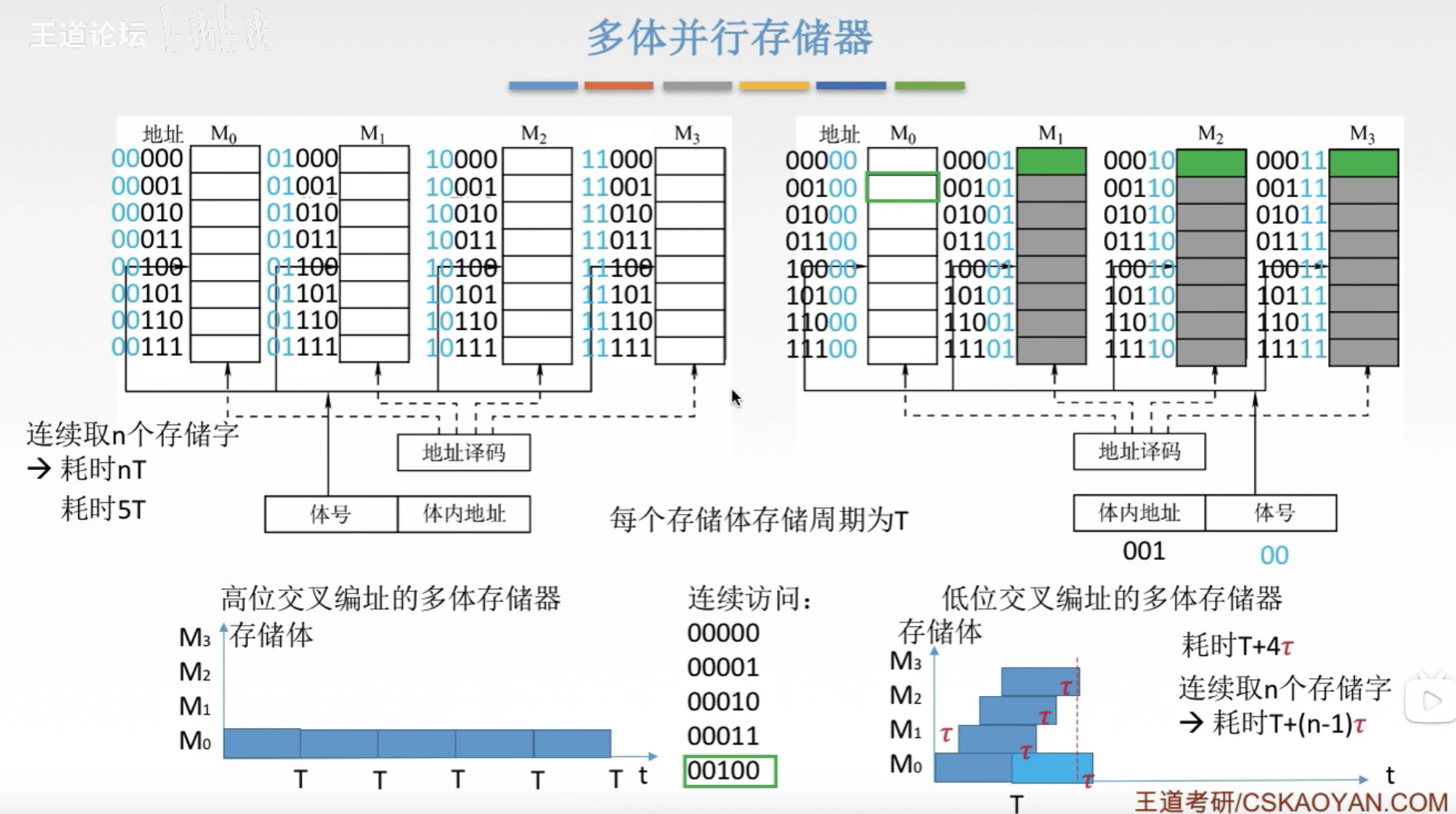
由于CPU和主存储器在速度上不匹配，限制了高速计算。

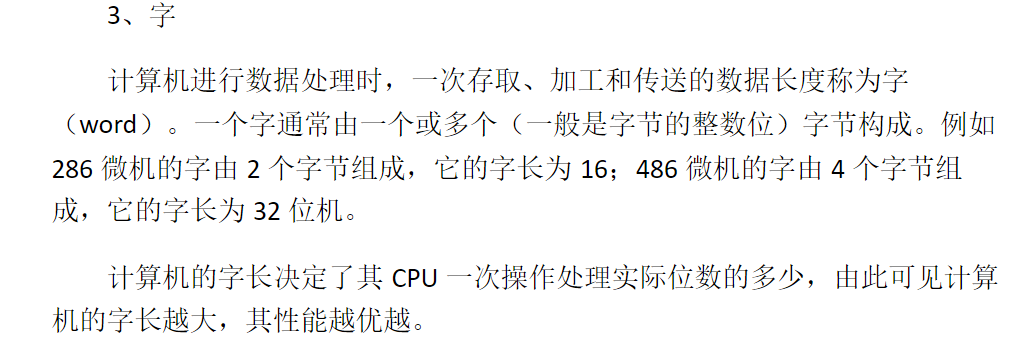
为了使CPU不至因为等待存储器读写操作的完成而无事可做，可以采取一些加速CPU和存储器之间有效传输的特殊措施

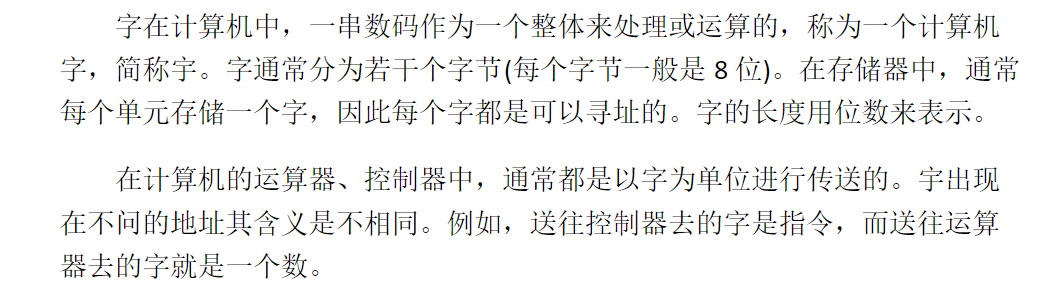
采用空间并行操作方式　　　　　　---双端口存储器

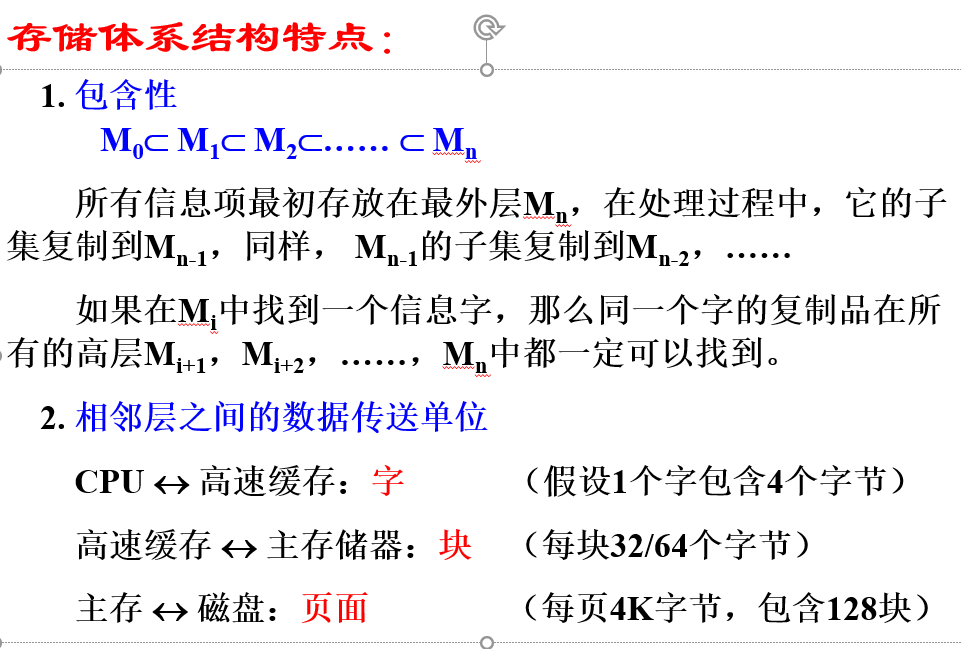
提高时间读出并行性 ---多模块交叉存储器

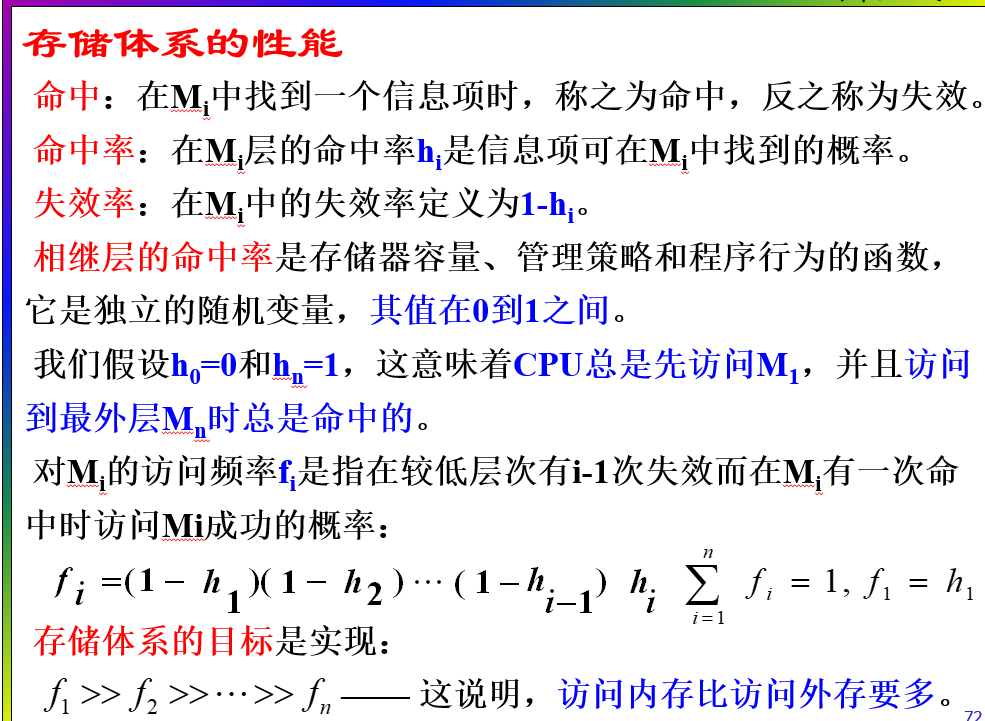
主存储器采用更高速的技术来缩短存储器的读出时间 ---相联存储器

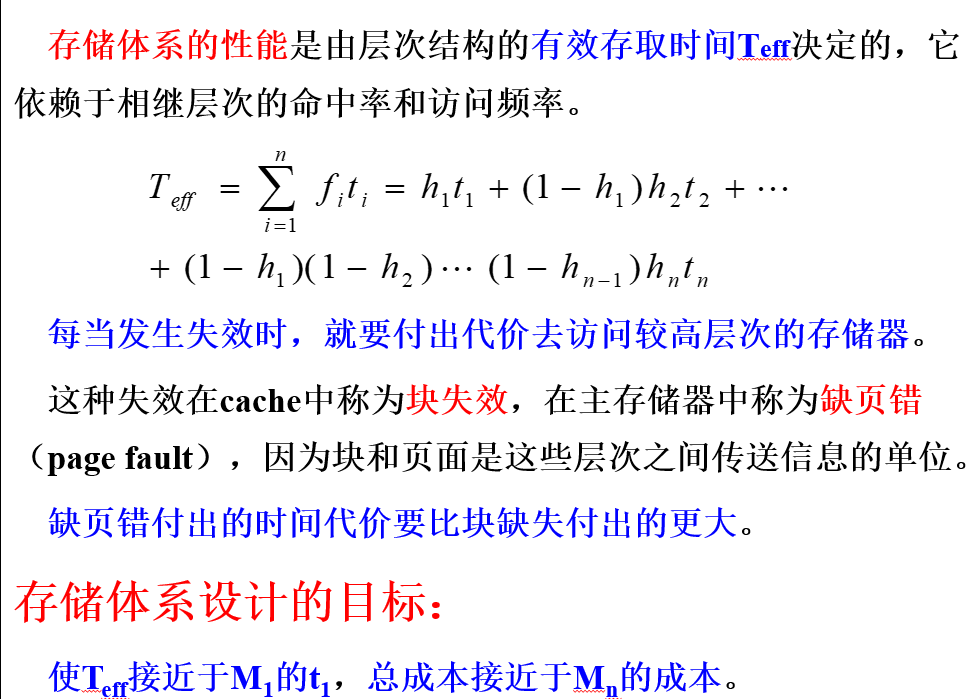


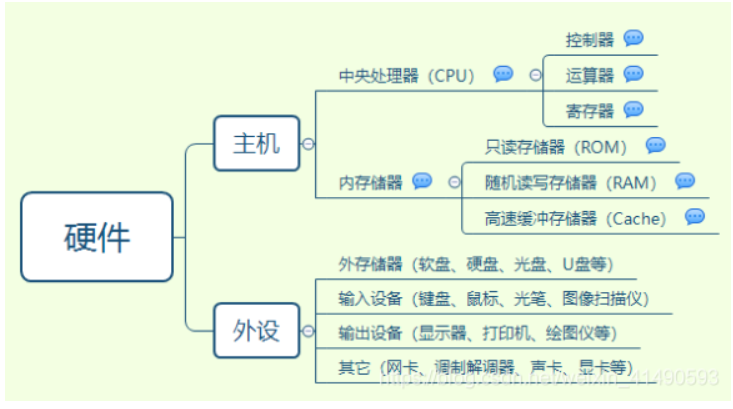




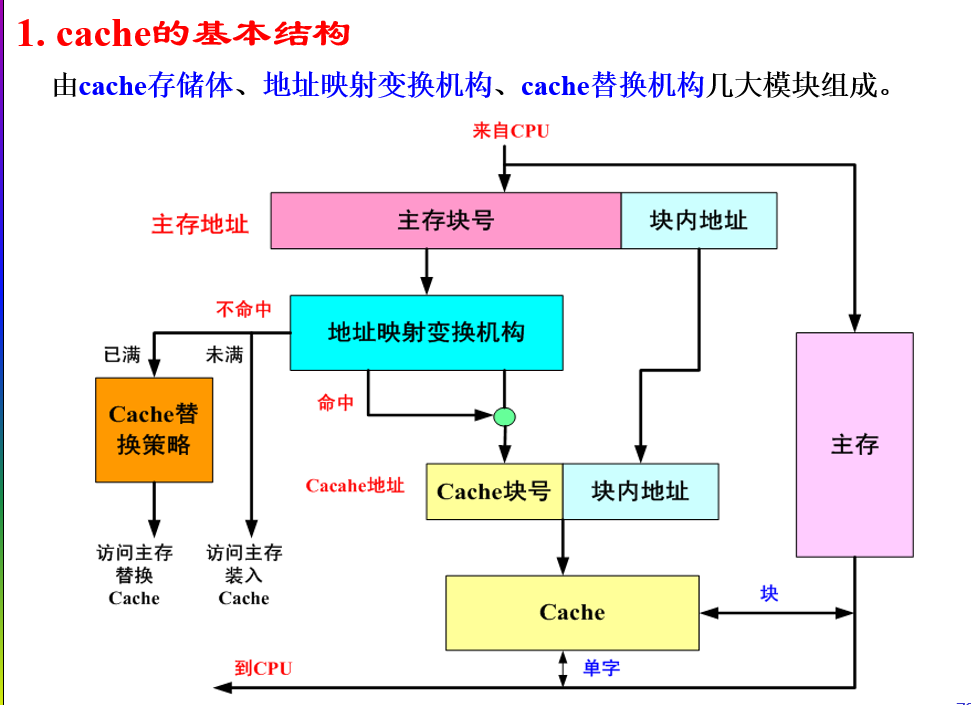








寄存器属于CPU的一个组成部分而缓存只是集成到CPU封装内完全是和CPU独立的器件。另外二者速度相差很大，寄存器存取速度最快

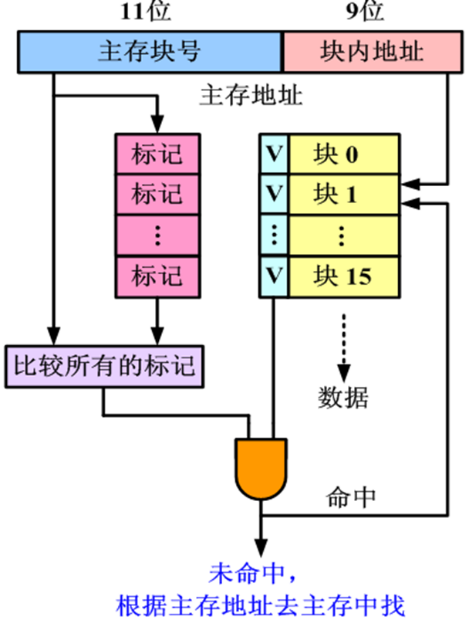


cache和主存等分为相同大小的块，每一块是由若干个字（或字节）组成

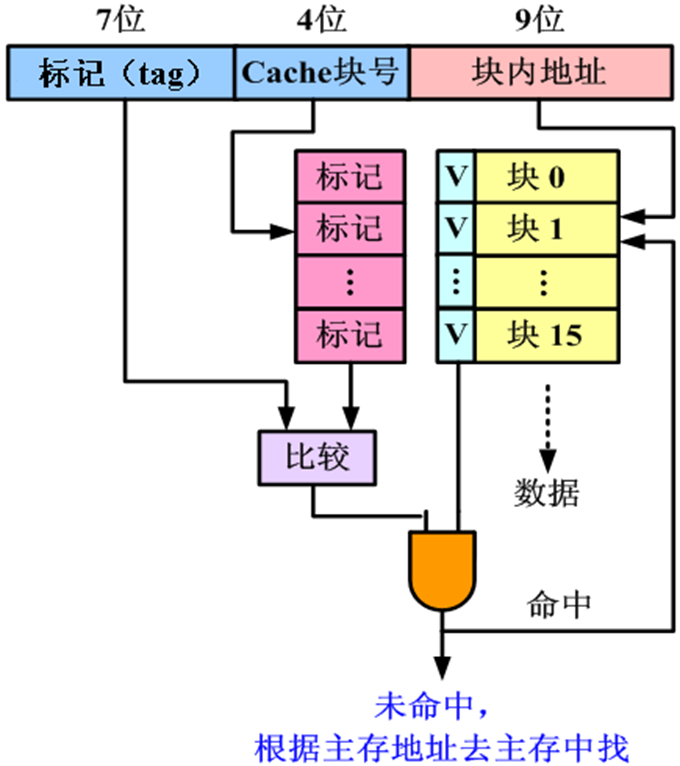
全相联映射

访问Cache时，需要和Cache的全部标记进行“比较”才能判断出所访问主存地址的内容是否已经在Cache中。

允许主存中的每一个字块映射到cache的任何一个字块位置上，同时比较所有标记（使用相联存储器），最灵活但成本最高的一种方式。



直接映射

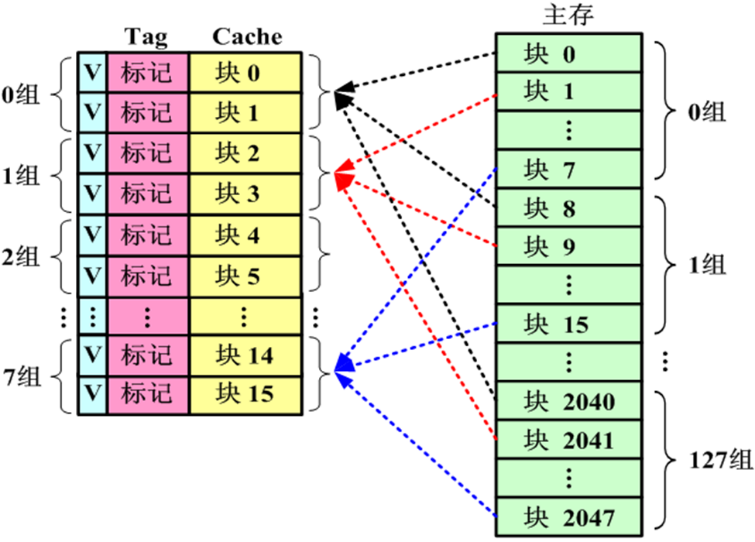


组相联映射

Cache与主存均分组，  
主存中一个组内的块数  
与Cache的分组数相同，  
主存中的各块与Cache的  
组号有固定的映射关系，  
但可以自由映射到对应的  
Cache组中的任一块。

当cache只有一个组，并且  
每组16块时，为全相联映射；

当有cache16组，并且每组一  
个块时，为直接映射。



每个程序的虚地址空间可以远大于实地址空间，也可以远小于实地址空间。前一种情况以提高存储容量为目的，后一种情况则以地址变换为目的。后者通常出现在多用户或多任务系统中：实存空间较大，而单个任务并不需要很大的地址空间，较小的虚存空间则可以缩短指令中地址字段的长度。

虚拟存储器的用户程序以虚拟地址编址并存放在辅存中。

程序运行时，由地址变换机构依据当时分配给该程序的实地址空间把程序的一部分调入实存。

每次访存时，首先判断该虚地址所对应的部分是否在实存中：如果是，则进行地址转换并用实地址访问主存；否则，按照某种算法将辅存中的部分程序调度进内存，再按同样的方法访问主存

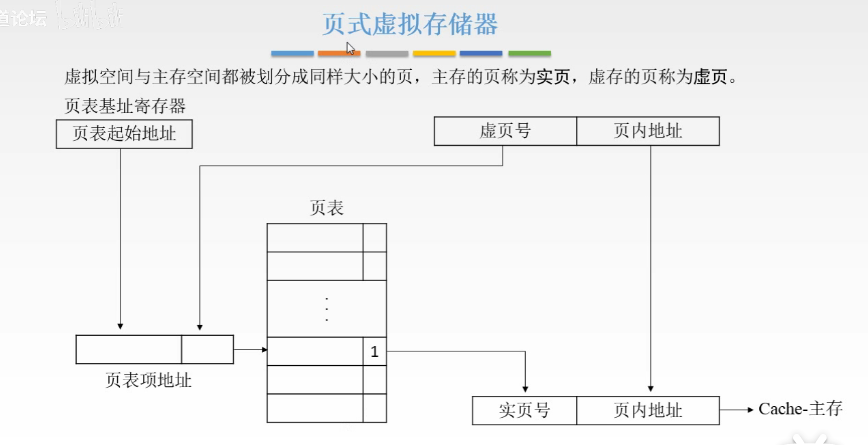
每个程序可以拥有一个虚拟的存储器，它具有辅存的容量和接近主存的访问速度。但这个虚拟存储器是由主存和辅存管理部件构成的概念模型，不是实际的物理存储器。

虚存通过增设地址映射表机构来实现程序在主存中的定位。

这种定位技术是把程序分割成若干个较小的段或页，用相应的映射表机构，来指明该程序的某段或某页是否已装入主存：

若已装入主存，则应同时指明其在主存中所处的开始位置；

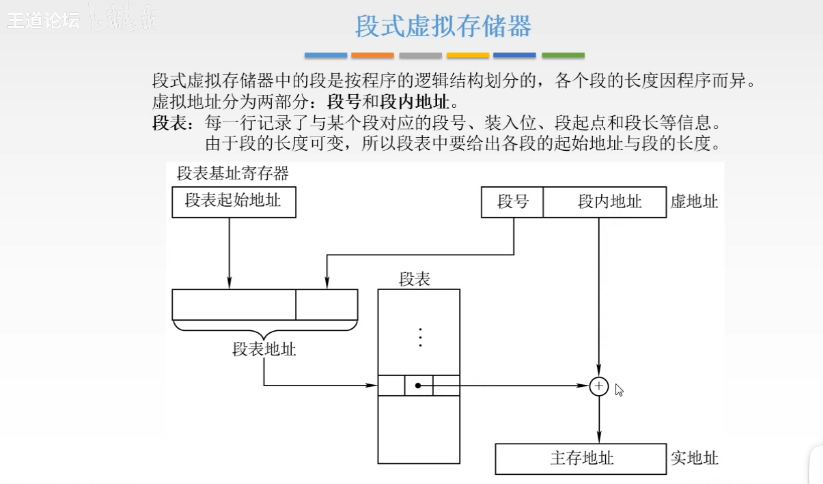
若未装入主存，则应到辅存中去调段或页，并建立起程序空间和实存空间的地址映射关系。这样，程序执行时通过查映射表，将程序（虚）地址变成主存地址再访问主存。



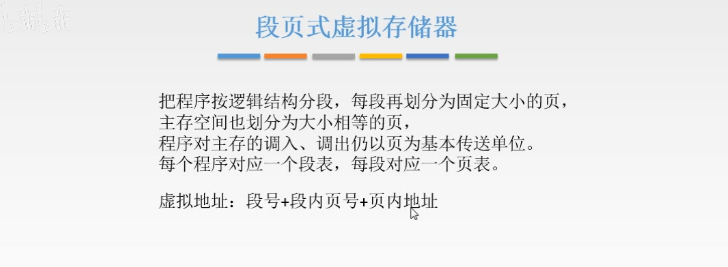
**地址变换时，用逻辑页号作为页表内的偏移地址索引页表（将虚页号看作页表数组下标）并找到相应物理页号，用物理页号作为实存地址的高字段，再与虚地址的页内偏移量拼接，就构成完整的物理地址。 当数据或指令不在主存时，产生页面失效（缺页），此时调用外页表把程序的虚地址变换成辅存的实际地址，从辅存调出该虚页，而后根据页表指出实页号再把虚页内容调入主存。**

**通常，页表的基地址保存在寄存器中，而页表保存在主存中。**

**为了节省页表本身占用的主存空间，一些系统把页表存储在虚存中，因而页表本身也要进行分页。当一个进程运行时，其页表中一部分在主存中，另一部分则在辅存中保存。**



段内地址长度是可变的



段表中每个表项对应一个段，其中“有效位”表示该段的页表是否已装入主存。若未装入，则应请求从辅存中调入页表；若已装入，则地址字段表示该段的页表在主存中的起始位置。 “段长”字段表示该段页表的行数。

每个段有一个页表，页表中用“有效位”指明此段该页是否已装入主存。若未装入，就需从辅存调页；若已装入，则用页号字段指明该页在主存中的页号。

如果有多个用户在机器上运行，多道程序的每一道需要一个基号，由它指明该道程序的段表起始地址。

段页式虚拟存储器在程序地址向实际主存地址变换时，首先要查段表，然后查页表。

段页式虚拟存储器缺点：映射过程中需要多次查表，实现复杂度高。

