本节内容

具有快表的 地址变换机 构 知识总览

局部性原理

具有快表的地址变换机构

是基本地址变换

什么是快表 (TLB)

引入快表后, 地址的变换过程

王道考研/CSKAOYAN.COM

王道考研/CSKAOYAN.COM

局部性原理

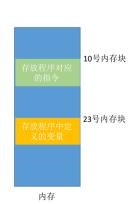
int i = 0;
int a[100];
while (i < 100) {
 a[i] = i;
 i++;
}</pre>

这个程序执行时, 会很频繁地访问 10 号、23号内存块

时间局部性: 如果执行了程序中的某条指令,那么不久后这条指令很有可能再次执行;如果某个数据被访问过,不久之后该数据很可能再次被访问。(因为程序中存在大量的循环)

空间局部性: 一旦程序访问了某个存储单元,在不久之后,其附近的存储单元也很有可能被访问。(因为很多数据在内存中都是连续存放的)

上小节介绍的基本地址变换机构中,每次要访问一个逻辑地址,都需要<mark>查询内存中的页表</mark>。由于局部性原理,可能连续很多次查到的都是同一个页表项。既然如此,能否利用这个特性减少访问页表的次数呢?

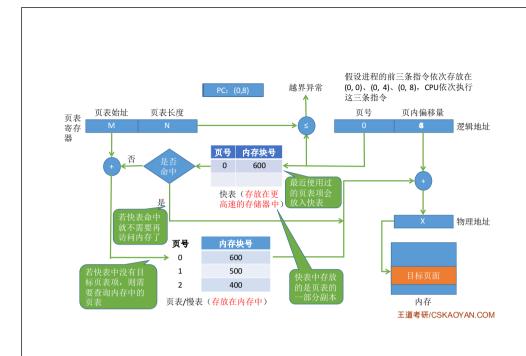


王道考研/CSKAOYAN.COM

什么是快表 (TLB)

快表,又称联想寄存器(TLB),是一种访问速度比内存快很多的高速缓冲存储器,用来存放当前访问的若干页表项,以加速地址变换的过程。与此对应,内存中的页表常称为慢表。

王道考研/CSKAOYAN.COM



知识回顾与重要考点

	地址变换过程	访问一个逻辑地 址的访存次数
基本地址变换机构	①算页号、页内偏移量 ②检查页号合法性 ③查页表,找到页面存放的内存块号 ④根据内存块号与页内偏移量得到物理地址 ⑤访问目标内存单元	两次访存
具有快表的 地址变换机 构	①算页号、页内偏移量 ②检查页号合法性 ③查快表。若命中,即可知道页面存放的内存块号,可直接进行⑤; 若未命中则进行④ ④查页表,找到页面存放的内存块号,并且将页表项复制到快表中 ⑤根据内存块号与页内偏移量得到物理地址 ⑥访问目标内存单元	快表命中,只需 一次访存 快表未命中,需 要两次访存

王道考研/CSKAOYAN.COM

引入快表后,地址的变换过程

- ① CPU给出逻辑地址,由某个硬件算得页号、页内偏移量,将页号与快表中的所有页号进行比较。
- ② 如果找到匹配的页号,说明要访问的页表项在快表中有副本,则直接从中取出该页对应的内存块号,再将内存块号与页内偏移量拼接形成物理地址,最后,访问该物理地址对应的内存单元。因此,若快表命中,则访问某个逻辑地址仅需一次访存即可。
- ③ 如果没有找到匹配的页号,则需要访问内存中的页表,找到对应页表项,得到页面存放的内存块号,再将内存块号与页内偏移量拼接形成物理地址,最后,访问该物理地址对应的内存单元。因此,若快表未命中,则访问某个逻辑地址需要两次访存(注意:在找到页表项后,应同时将其存入快表,以便后面可能的再次访问。但若快表已满,则必须按照一定的算法对旧的页表项进行替换)

由于查询快表的速度比查询页表的速度快很多,因此只要快表命中,就可以节省很多时间。 因为局部性原理,一般来说快表的命中率可以达到 90% 以上。

例:某系统使用基本分页存储管理,并采用了具有快表的地址变换机构。访问一次快表耗时 1us,访问一次内存耗时 100us。若快表的命中率为 90%,那么访问一个逻辑地址的平均耗时是多少? (1+100)*0.9+(1+100+100)*0.1=111 us

有的系统支持快表和慢表同时查找,如果是这样,平均耗时应该是 (1+100)*0.9+(100+100)*0.1=110.9 us

若未采用快表机制,则访问一个逻辑地址需要 100+100 = 200us 显然,引入快表机制后,访问一个逻辑地址的速度快多了。

王道考研/CSKAOYAN.COM