

存储器 · 虚拟存储器

xyfJASON

-
- 1 概述
 - 2 页的存放和查找
 - 3 缺页
 - 4 快速地址转换——TLB
-

1 概述

为了扩大寻址空间、实现存储保护等，使用虚拟存储器，它自动管理由主存和辅存组成的两级存储器层次结构。

虚拟存储器和 cache 的工作原理是一样的，但术语有些不同。虚拟存储器中块被称为页（往往比 cache 的一块要大很多），访问缺失称为缺页。CPU 给出的地址为虚拟地址 (VA)，经过地址转换变成物理地址 (PA) 后进行访存。

和 cache 分块类似，虚拟地址可以分为高位的虚页号 (VPN) 和低位的页内偏移 (VPO)，地址转换时只需将虚页号转换成物理页号，页内偏移不变。

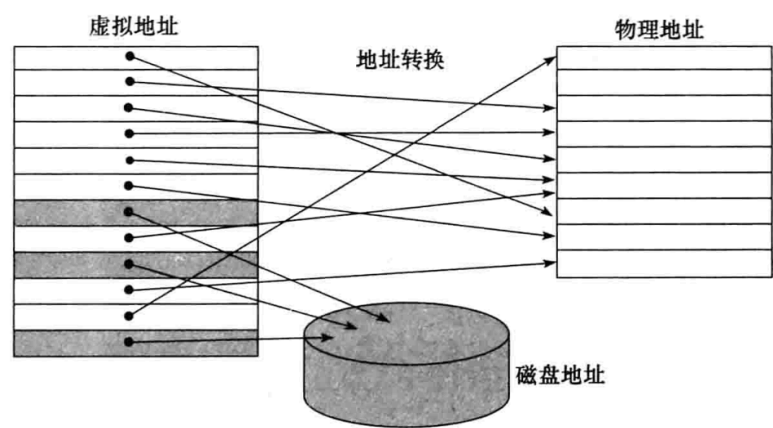


图 5-25 在存储器中，主存中的块（称为页）从一组地址（称为虚拟地址）映射到另一组地址（称为物理地址）。访问主存使用物理地址，而处理器产生虚拟地址。虚拟存储器和物理存储器都被划分成页，因此一个虚页被映射到一个物理页。当然，一个虚页也可能不在主存中，因此无法映射到物理地址；在这种情况下，页就被存在磁盘上。物理页可以被两个指向相同物理地址的虚拟地址共享。这种方法用来使两个不同的程序共享数据或代码

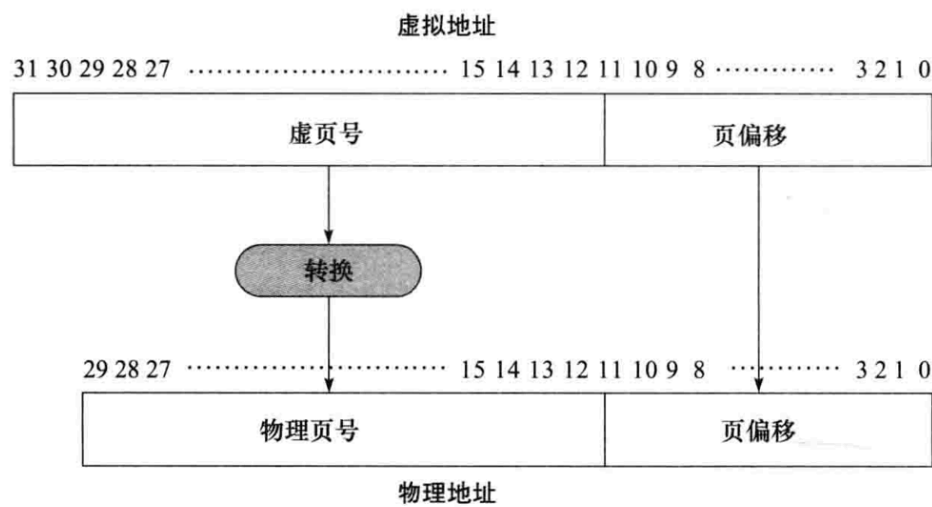


图 5-26 虚拟地址到物理地址的映射。页大小为 $2^{12} = 4\text{KiB}$ 。由于物理页号有 18 位，存储器中物理页数为 2^{18} 。因此，最多可以支持 1GiB 的主存，而虚拟地址空间为 4GiB

除了页式管理，还有一种可变长度块的机制称为段式管理 (segmentation)。段式存储中，地址分为段号和段内偏移。段的独立性使之易于编译、管理、修改、维护和共享，但段的长度不同会给主存的分配带来麻烦，以容易产生碎块。

段页式管理结合段式和页式的优点，先分段，段内再分页。

2 页的存放和查找

由于缺页的代价十分高，所以我们允许一个虚拟页映射到任何一个物理页，类比 cache 与主存映射的全相联方式。

但全相联时，项的定位比较困难，我们不可能全部检索，因此采用页表 (page table) 进行查找。页表的每一项称为页表项 (PTE, page table entry)，以虚拟页号 (VPN, virtual page number) 作为索引，能找到相应的物理页号 (PPN, physical page number)。每个程序都有其自己的页表。页表存放在主存中（也因而可能被调入 cache 中），其位置由一个指向页表首地址的页表基址寄存器 (PTBR) 给出。

下图解释了页表的查找方式：

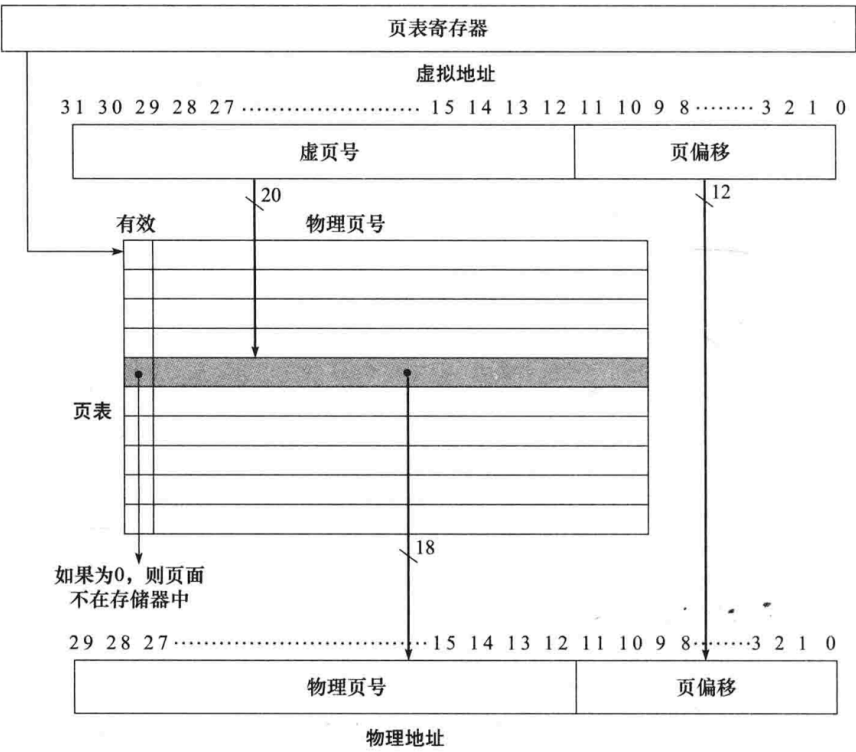
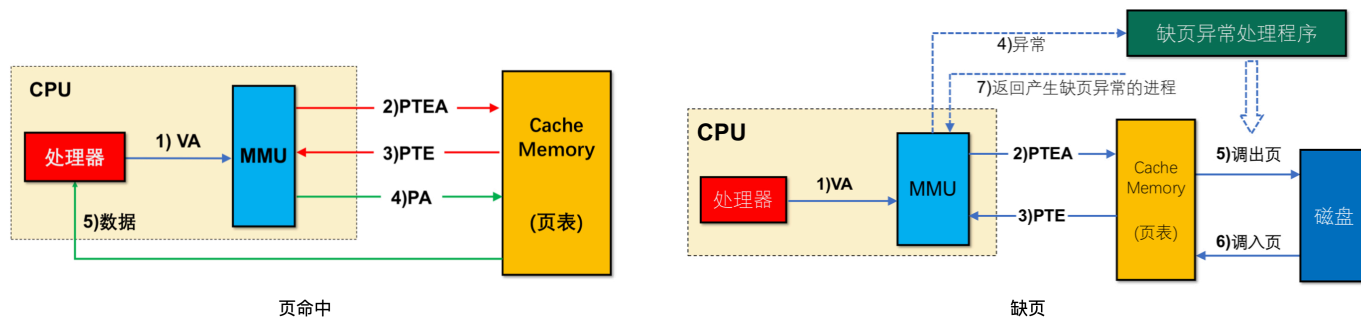


图 5-27 用虚页号来索引页表以获得对应的物理地址部分。假定地址为 32 位。页表的首地址由页表指针给出。在本图中，页大小为 2^{12} 字节，即 4KiB，虚拟地址空间为 2^{32} 字节，即 4GiB，物理地址空间为 2^{30} 字节，可以支持高达 1GiB 的主存。页表项数为 2^{20} ，即 100 万项。每一项的有效位指出了映射是否合法。如果该位为 0，那么该页就不在主存中。尽管图中所示的页表项宽度只需 19 位，但为了寻址方便，通常让它有 32 位。其他位则用来存放每页都要保留的基本的附加信息，如保护信息

3 缺页

当页表指示 CPU 需要访问的页不在主存中时，称为缺页 (page faults)（类比 cache 的未命中）。此时若主存中所有页都在使用，则操作系统将选择一页进行替换，由于缺页的代价十分高，所以采用近似的最近最少使用 (LRU) 替换策略。



缺页时的替换流程为：

- 操作系统负责将数据从磁盘迁移到内存中
- 当前进程挂起
- 操作系统负责所有的替换策略
- 唤醒挂起进程

4 快速地址转换——TLB

由于页表存放在主存中，因此程序每次访存至少需要两次：第一次访存先获得物理地址（上图第 2、3 步），第二次访存才获得数据（上图第 4、5 步）。为提高访问性能，利用页表的访问局部性，我们用一个特殊的 cache，即快表 (TLB, Translation-Lookaside Buffer) 辅助地址转换。

TLB 内记录最近使用地址的映射信息，即 PTE。CPU 寻址时，优先在 TLB 中寻址，从而可以避免每次都要访问页表。

TLB 与页表之间的映射关系类似于 cache 与主存之间的映射关系——对来自 CPU 的虚页号进行逻辑划分，得到相应的标识和数据：

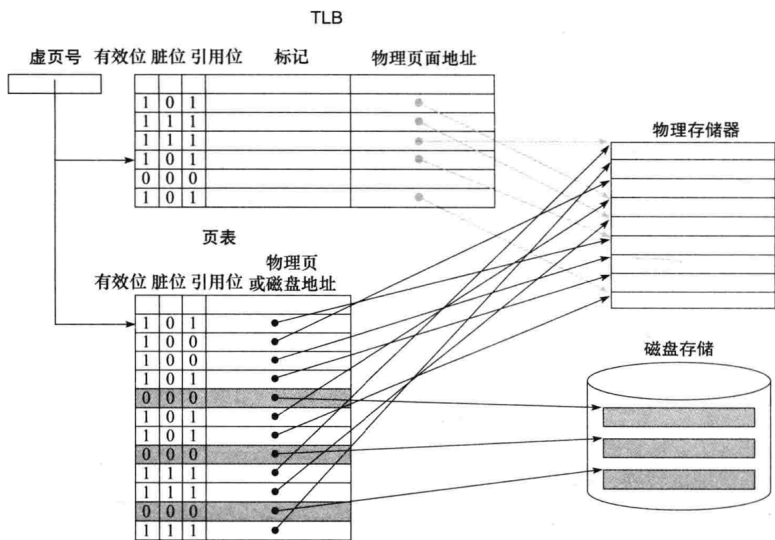
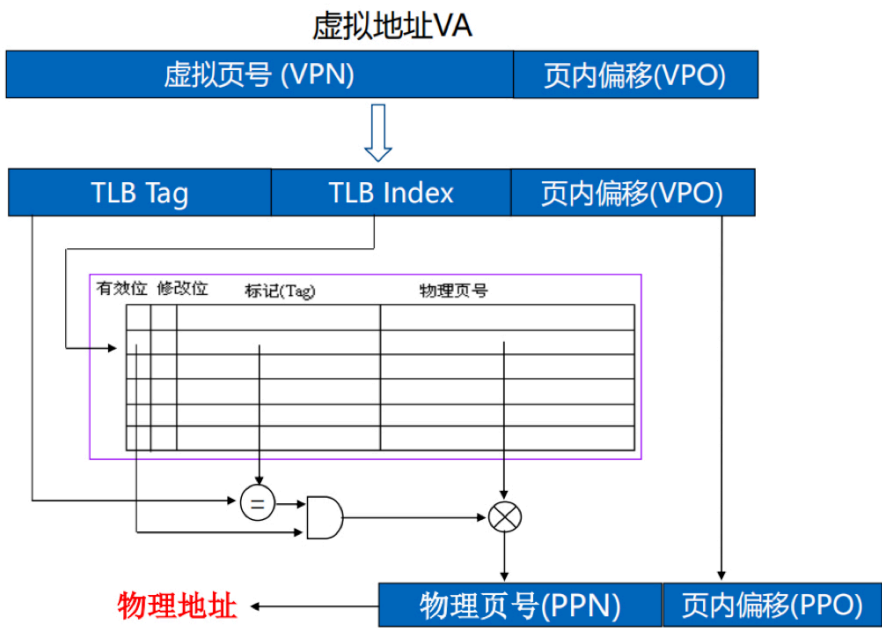


图 5-29 TLB 作为页表的 cache，用于存放映射到物理页中的那些项。TLB 包含了页表中虚页到物理页映射的一个子集。TLB 映射以灰线显示。因为 TLB 是一个 cache，它必须有标记域。如果一个页在 TLB 中没有匹配的项，就必须检查页表。页表或者提供该页的物理页号（可用来创建一个 TLB 项），或者指出该页在磁盘上，这时就会发生缺页。由于页表对于每个虚页都有一个相应的项，并不需要标记；换句话说，不同于 TLB，页表并不是 cache

于是乎，现在从虚拟地址得到物理地址的总流程如下：

