原理。根据这一原理，很容易设想，只要将 CPU 近期要用到的程序和数据提前从主存送到 Cache，那么就可以做到 CPU 在一定时间内只访问 Cache。一般 Cache 采用高速的 SRAM 制作，其价格比主存贵，但因其容量远小于主存，因此能很好地解决速度和成本的矛盾。

2. Cache 的工作原理

图 4.49 是 Cache-主存存储空间的基本结构示意图。

主存由 2^n 个可编址的字组成，每个字有唯一的 n 位地址。为了与 Cache 映射，将主存与缓存都分成若干块，每块内又包含若干个字，并使它们的块大小相同（即块内的字数相同）。这就将主存的地址分成两段：高 m 位表示主存的块地址，低 b 位表示块内地址，则 $2^m = M$ 表示主存的块数。同样，缓存的地址也分为两段：高 c 位表示缓存的块号，低 b 位表示块内地址，则 $2^c = C$ 表示缓存块数，且 C 远小于 M。主存与缓存地址中都用 b 位表示其块内字数，即 $B = 2^b$ 反映了块的大小，称 B 为块长。

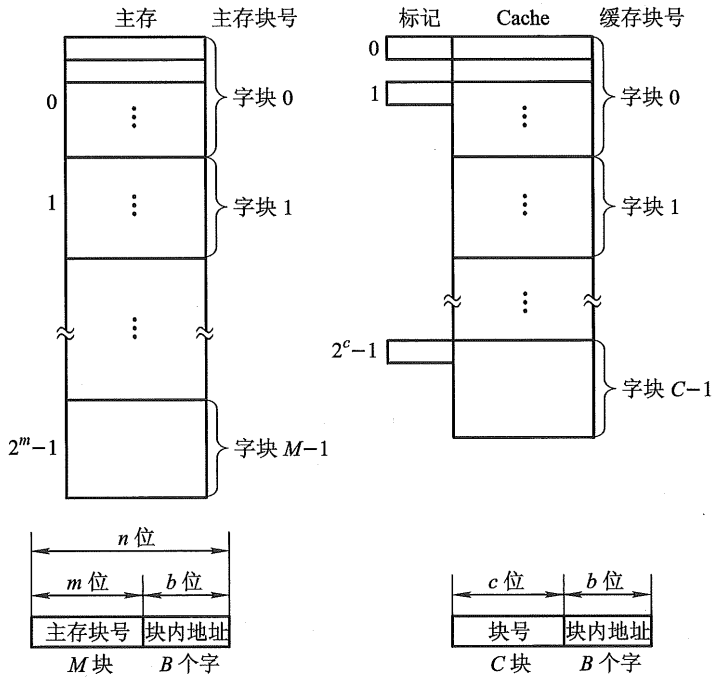

图 4.49　Cache-主存存储空间的基本结构

任何时刻都有一些主存块处在缓存块中。CPU 欲读取主存某字时，有两种可能：一种是所需要的字已在缓存中，即可直接访问 Cache（CPU 与 Cache 之间通常一次传送一个字）；另一种是所需的字不在 Cache 内，此时需将该字所在的主存整个字块一次调入 Cache 中（Cache 与主存之间是字块传送）。如果主存块已调入缓存块，则称该主存块与缓存块建立了对应关系。

上述第一种情况为CPU访问Cache命中,第二种情况为CPU访问Cache不命中。由于缓存的块数 C 远小于主存的块数 M,因此,一个缓存块不能唯一地、永久地只对应一个主存块,故每个缓存块需设一个标记(参见图4.49),用来表示当前存放的是哪一个主存块,该标记的内容相当于主存块的编号。CPU读信息时,要将主存地址的高 m 位(或 m 位中的一部分)与缓存块的标记进行比较,以判断所读的信息是否已在缓存中(参见图4.54)。

Cache的容量与块长是影响Cache效率的重要因素,通常用"命中率"来衡量Cache的效率。命中率是指CPU要访问的信息已在Cache内的比率。

在一个程序执行期间,设 N_c 为访问Cache的总命中次数,N_m 为访问主存的总次数,则命中率 h 为

$$h=\frac{N_c}{N_c+N_m}$$

设 t_c 为命中时的Cache访问时间,t_m 为未命中时的主存访问时间,$1-h$ 表示未命中率,则Cache-主存系统的平均访问时间 t_a 为

$$t_a=ht_c+(1-h)t_m$$

当然,以较小的硬件代价使Cache-主存系统的平均访问时间 t_a 越接近于 t_c 越好。用 e 表示访问效率,则有

$$e=\frac{t_c}{t_a}\times 100\%=\frac{t_c}{ht_c+(1-h)t_m}\times 100\%$$

可见,为提高访问效率,命中率 h 越接近1越好。

例4.7 假设CPU执行某段程序时,共访问Cache命中2000次,访问主存50次。已知Cache的存取周期为50 ns,主存的存取周期为200 ns。求Cache-主存系统的命中率、效率和平均访问时间。

解:(1) Cache的命中率为

$$2000/(2000+50)=0.97$$

(2) 由题可知,访问主存的时间是访问Cache时间的4倍(200/50=4)。

设访问Cache的时间为 t,访问主存的时间为 $4t$,Cache-主存系统的访问效率为 e,则

$$e=\frac{\text{访问 Cache 的时间}}{\text{平均访问时间}}\times 100\%$$

$$=\frac{t}{0.97\times t+(1-0.97)\times 4t}\times 100\%=91.7\%$$

(3) 平均访问时间为

$$50\ \text{ns}\times 0.97+200\ \text{ns}\times(1-0.97)=54.5\ \text{ns}$$

一般而言,Cache容量越大,其CPU的命中率就越高。当然容量也没必要太大,太大会增加成本,而且当Cache容量达到一定值时,命中率已不因容量的增大而有明显的提高。因此,Cache容量是总成本价与命中率的折中值。例如,80386的主存最大容量为4 GB,与其配套的Cache容

量为 16 KB 或 32 KB,其命中率可达 95%以上。

块长与命中率之间的关系更为复杂,它取决于各程序的局部特性。当块由小到大增长时,起初会因局部性原理使命中率有所提高。由局部性原理指出,在已被访问字的附近,近期也可能被访问,因此,增大块长,可将更多有用字存入缓存,提高其命中率。可是,倘若继续增大块长,命中率很可能下降,这是因为所装入缓存的有用数据反而少于被替换掉的有用数据。由于块长的增大,导致缓存中块数的减少,而新装入的块要覆盖旧块,很可能出现少数块刚刚装入就被覆盖,因此命中率反而下降。再者,块增大后,追加上的字距离已被访问的字更远,故近期被访问的可能性会更小。块长的最优值是很难确定的,一般每块取 4 至 8 个可编址单位(字或字节)较好,也可取一个主存周期所能调出主存的信息长度。例如,CRAY-1 的主存是 16 体交叉,每个体为单字宽,其存放指令的 Cache 块长为 16 个字。又如,IBM 370/168 机主存是 4 体交叉,每个体宽为 64 位(8 个字节),其 Cache 块长为 32 个字节。

3. Cache 的基本结构

Cache 的基本结构原理框图如图 4.50 所示。

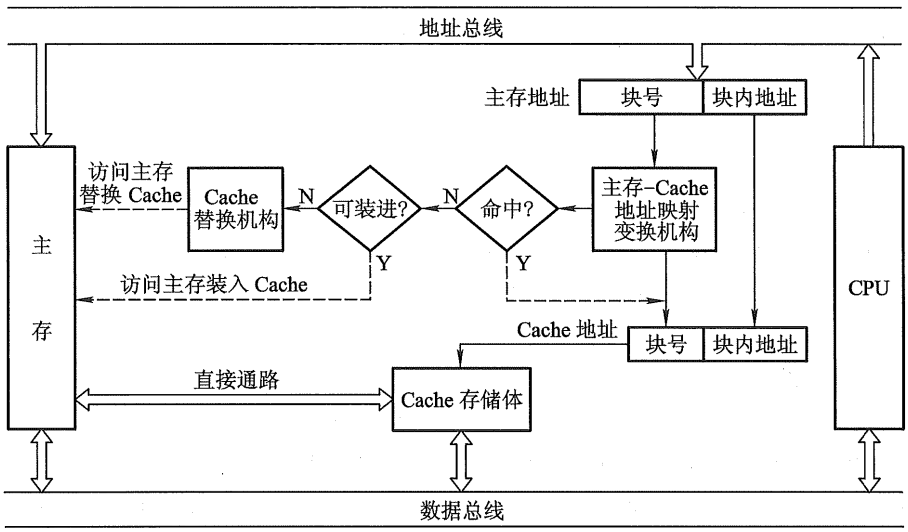

图 4.50　Cache 的基本结构原理框图

它主要由 Cache 存储体、地址映射变换机构、Cache 替换机构几大模块组成。

(1) Cache 存储体

Cache 存储体以块为单位与主存交换信息,为加速 Cache 与主存之间的调动,主存大多采用多体结构,且 Cache 访存的优先级最高。

(2) 地址映射变换机构

地址映射变换机构是将 CPU 送来的主存地址转换为 Cache 地址。由于主存和 Cache 的块大小相同,块内地址都是相对于块的起始地址的偏移量(即低位地址相同),因此地址变换主要