

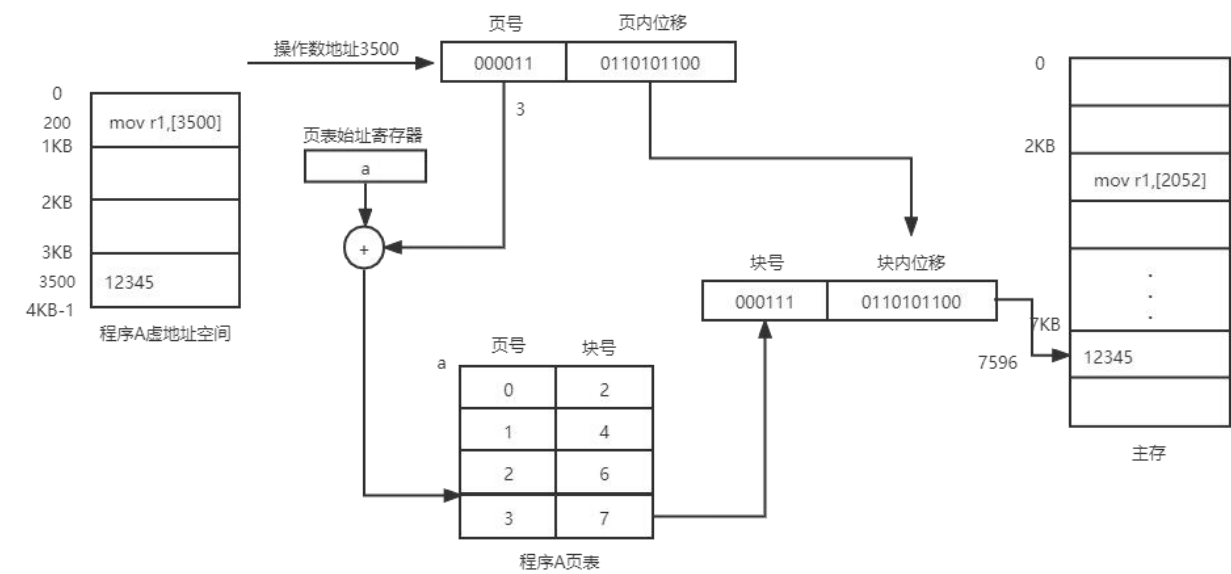
第 7 章作业

7-14、

(1) 程序 A 的页表：

页号	块号
0	2
1	4
2	6
3	7

(2)



根据计算可知，最终物理地址为 7596.

7-16、

不可以，会出现数据写入或读出错误，应该等到数据交换完毕后再进行调度。为了实现正确的页面调度，页表中除了页号和主存块号外，还包括辅存地址：用来标识该页面在辅存中的位置；中断位：用来标识该页面是否在主存中，若为 1 则表明不在主存中；访问位：用来表示该页在内存中时是否被访问过；修改位：用来表示该页在内存中时是否被修改过。调度和置换则根据不同算法来制定，根据上述中断位、访问位和修改位来判断。

7-21、

(1) 先进先出算法：

页面序列	0	1	3	0	5	2	0
栈顶	0	1*	3*	3	5*	2*	0*
		0	1	1	3	5	2
			0	0	1	3	5

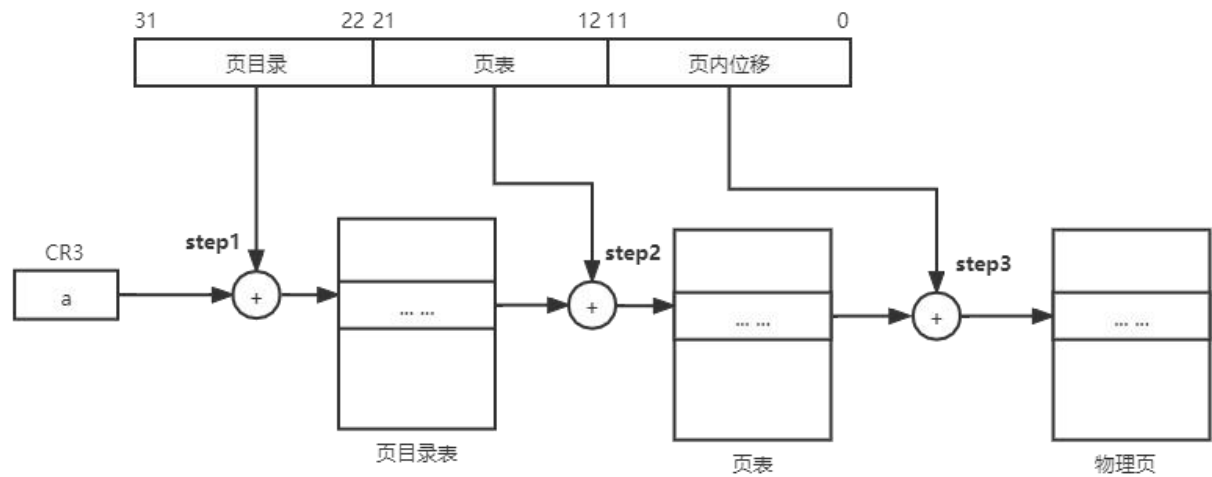
缺页中断次数为 5 次。

(2) LRU 算法：

页面序列	0	1	3	0	5	2	0
栈顶	0	1*	3*	0	5*	2*	0
		0	1	3	0	5	2
			0	1	3	0	5

缺页中断次数为 4 次。

7-27、



**Step1:** 由存放正在使用的页目录表的起始物理地址的寄存器 **cr3** 与分页结构中的页目录字段的内容相加，结果为指向页目录表项的地址。

**Step2:** 由指向页目录表项的值作为当前使用页表的起始地址，与页表字段的内容相加，结果为指向页表表项的地址。

**Step3:** 由指向的页表表项指示的该页的物理页的主存地址与分页结构中的页内位移相加，得到最终的物理地址。

7-31、

(1) 32 位线性地址中，高 10 位用来指示第一级页表表项，次高 10 位用来指示第二级页表表项，低 12 位则用来表示页内位移，即块内位移，具体位数见下图。故物理页帧的大小为  $2^{12}\text{bits} = 4\text{KB}$ 。



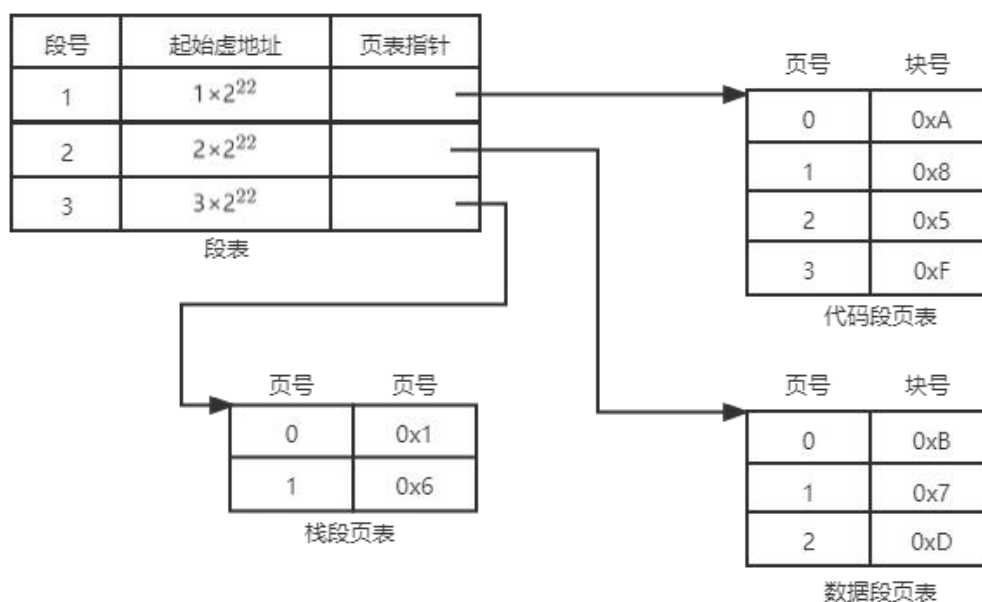
(2)

一次完整的数据访问需要访存 3 次，第一次查第一级页表，第二次查指示的第二级页表，最后查找指示的物理页帧，故一次完整的数据访问所耗费时间为  $500\text{ns} \times 3 = 1500\text{ns}$ 。

7-32、

(1) 总的虚拟地址 24 位，段类型占 2 位，故代码段内偏移占 22 位，故进程的代码段空间最大为  $2^{22}\text{bits} = 4\text{MB}$ 。而整个虚拟地址空间最大为  $2^{24}\text{bits} = 16\text{MB}$ 。

(2)



(3)

$8006\text{ADH} = (\underline{1000\ 0000\ 0000\ 0110\ 1010\ 1101})_2$

横线部分为段号，可知其为数据段，低 11 位双横线为页内偏移。中间部分为页号，可知页号为 0。

查找数据段页表，可知块号为 0xB。

故物理地址为  $(101\ 1)_2 \times 2\text{K} + (110\ 1010\ 1101)_2 = 5\text{EAD H}$