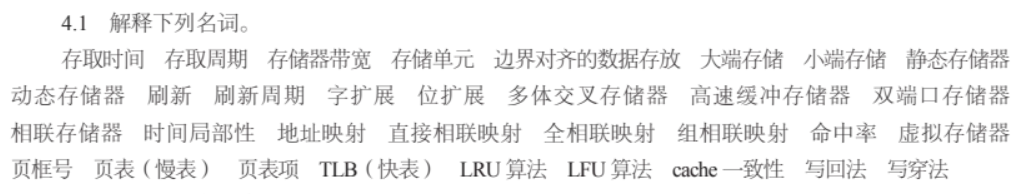
作业讲解四

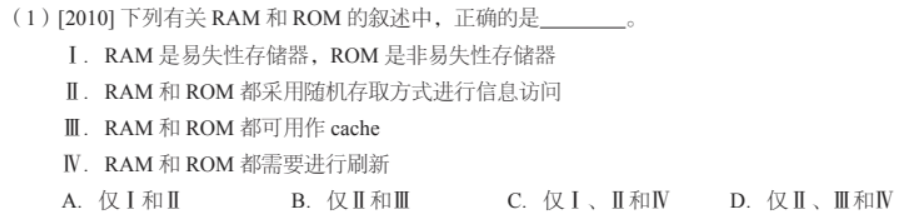
庾晓萍（20420192201952）

4.1

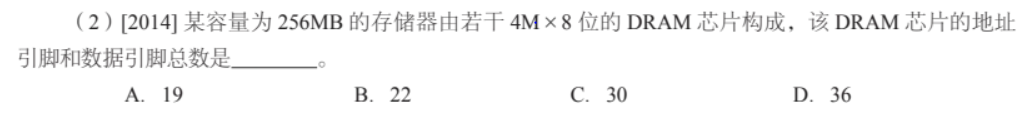


1. 存取时间：又称为存储器的访问时间，是指启动一次存储器操作（读或写分别对应取与存）到该操作完成所经历的时间，注意读写时间可能不同，DRAM（动态随机存取存储器）读慢写快、闪存读快写慢。
2. 存取周期：连续启动两次访问操作之间的最短时间间隔；对主存而言，存储周期除包括存取时间外，还包括存储器状态的稳定恢复时间，所以存储周期略大于存取时间。
3. 存储器带宽：单位时间内存储器所能传输的信息量，常用的单位包括位/秒或字节/秒。带宽是衡量数据传输速率的重要指标，与存取时间的长短和一次传输的数据位的多少有关。一般而言存取时间越短、数据位宽越大、存储带宽越高。
4. 存储单元：主存是机器指令直接操作的存储器，采用主存地址进行随机访问，整个主存从空间逻辑上可以看作一个一维数组mem[]，每个数组元素存储一个m位的数据单元，也就是主存地址对应的存储内容。
5. 边界对齐的数据存放：边界对齐就是按照数据类型的大小进行边界对齐。①双字（4字节）数据起始字节地址的最末3位为000，地址是8的整数倍。②单字数据（2字节）起始字节地址的最低2位为00，地址是4的整数倍。（3）半字数据（short）起始地址的最低一位是0，地址是2的整数倍。（4）单字节数据不存在边界对齐问题（主存按字节编址）（注：定义字是根据处理器的特性决定的。首先ARM是32bit处理器，所以它的字是32bit的。半字就是16bit）
6. 大端存储：存储器的低字节地址单元中存放的是数据的高字节。
7. 小端存储：存储器的低字节地址单元中存放的是数据的低字节。
8. 静态存储器：存储器以静态MOS存储元为基本单元组成的存储器称为静态MOS存储器（SRAM）。
9. 动态存储器：存储器以动态MOS存储元为基本单元组成的存储器称为动态MOS储存器（DRAM）
10. 刷新：DRAM的电容C上的电荷会逐渐泄露，数据只能保存较短时间。为避免数据丢失，必须定期采用类似读操作的方式对存储单元补充电荷，这个过程称为刷新，这也是动态RAM得名的原因。
11. 刷新周期：信息存储到数据丢失之前这段时间称为最大刷新周期，而刷新周期则是存储器实际完成两次完整刷新之间的时间间隔。采用不同材料及不同生产工艺生产的动态存储器的最大刷新周期可能不同。
12. 字扩展：字扩展也称为容量扩展或地址总线扩展。当存储芯片的存储容量不能满足存储器对存储容量的要求时，可采用字扩展方式来扩展存储器。进行字扩展时，将所有存储芯片的数据线、读写控制线各自并联，同时分别与CPU的数据线和读写控制线连接。各存储芯片的片选信号可以由CPU多余的地址线通过译码器译码产生。
13. 位扩展：位扩展又称为字长扩展或数据总线扩展。当存储芯片的数据总线位宽小于CPU数据总线位宽时，采用位扩展方式扩展。进行扩展时，将所有存储芯片的地址线、读写控制线并联后分别与CPU的地址线和读写控制线连接。将存储芯片的数据线依次与CPU的数据线相连，将所有芯片的片选控制线并联后与CPU的访存请求信号MREQ#相连。
14. 多体交叉存储器：多体交叉存储器由多个存储模块构成，这些模块的容量和存取速度相同。根据对多个模块编址方式的不同，其组织方式又可分为高位多体交叉和低位多体交叉两种。
15. 高速缓冲存储器：为了进一步提升CPU访问主存的性能，通常会在CPU与主存之间增加一个隐藏的小容量的快速的SRAM，称为cache。将主存中经常访问或即将访问的数据的副本调度到小容量的SRAM中，使得大部分数据访问都可以在快速的SRAM中进行，从而提升系统性能。
16. 双端口存储器：双端口存储器是指同一个存储器具有两组相互独立的读写控制线路,由于进行并行的独立操作，是一种高速工作的存储器。
17. 相联存储器：与一般存储器按地址访问不同，相联存储器是一种按内容进行访问的存储器，用于存放查找表，其内部存储的基本数据单元是键值对（Key，Value）。CAM的输入不是地址，而是检索关键字key，输出则是该关键字对应的value值。
18. 时间局部性：时间局部性是指当程序访问一个存储位置时，该位置在未来可能会被多次访问，程序的循环结构和调用过程就很好地体现了时间局部性。
19. 地址映射：主存中的数据块应如何放置到cache中，是任意放置还是按照一定的规则放置，不同的地址映射策略将对cache的性能以及硬件成本带来影响。
20. 直接相联映射：直接相联映射中每一个主存块地址只能映射到cache中固定的行，具体映射规则为：cache行号i=主存块号jmod(cache行数n)（4-4）以上规则等效于将主存按照cache大小进行分区，每个分区中包含的块数与cache的行数相同，因此主存地址可细分为区地址（tag）、区内行索引（index）、块内偏移（offset）三部分，这里index字段就是数据块映射到cache中的行号，和上式中的余数完全相同。
21. 全相联映射：全相联映射方式下，主存中的每一个数据块都可以放置到cache的任意一个数据块中，是一对多的映射关系。新的主存数据块可以载入cache中任何一个空位置，只有cache满时才需要进行数据块置换。全相联映射时cache利用率最高，但查找成本较高，需要CAM提供快速的查找功能。
22. 组相联映射：组相联映射是直接相联映射和全相联映射两种方式的折中，既能提高命中率，又能降低查找硬件的开销。组相联映射将cache分成固定大小的组，每组有k行，称为k-路组相联；主存数据块首先采用直接相联映射的方式定位到cache中固定的组，然后采用全相联映射的方式映射到组内任何一个cache行。
23. 命中率:设Nc为某程序运行期间命中cache的次数，Nm为从主存中访问信息的次数，则命中率h（HitRatio）定义为：h=Nc/Nc+Nm。
24. 虚拟存储器：虚拟存储器充分利用了程序的局部性，采用按需加载的方式加载程序代码和数据。其基本思路是加载程序时并不直接将程序和代码载入主存，而仅仅在相应的虚拟地址转换表（段表、页表）中登记虚拟地址对应的磁盘地址。程序执行并访问该虚拟地址对应的程序或数据时，会产生缺页异常，操作系统会调用异常处理程序并载入实际的程序和代码。根据程序局部性原理，通常程序只需要加载很小一部分空间即可运行，这种方式避免了将程序全部载入主存，大大提高了主存的利用率。
25. 在页式虚拟存储器中，物理地址被划分为物理页号（PPN）和物理页偏移（PPO）两部分，其中物理页号又称为页框号。
26. 页表（慢表）：页式虚拟存储器中虚拟地址与物理地址之间的转换是基于页表进行的。页表是一张保存虚拟页号VPN和物理页号PPN对应关系的查找表，是一个由若干个表项组成的数组；采用VPN作为索引进行访问，每一个表项主要包括有效位和物理页号，另外还包括修改位、使用位、权限位等信息。
27. 页表项：页表常驻内存，并将虚拟地址中的虚拟页号作为索引来访问，每一个虚拟页都对应一个页表项。
28. TLB（快表）：为了进一步降低虚拟存储器地址转换的硬件开销，现代处理器都维护着一个转换旁路缓冲区（TLB），用于缓冲经常访问的页表项PTE。TLB本质上就是一个容量较小的cache，为提高查找速度，大多采用全相联或组相联方式，且采用随机替换算法。TLB离CPU更近，访问速度更快，所以通常将TLB表称为快表。
29. LRU算法：近期最少使用（LeastRecentlyUsed，LRU）算法是将近期内最久未被访问过的行淘汰。为此，每行也需要设置一个计数器，cache每命中一次，对应的命中行计数器清零，其他各行计数器加1，因此它是未访问次数计数器。当需要替换时，比较各特定行的计数值，将计数值最大的行换出。这种算法显然保护了刚载入cache的新数据，符合cache工作原理，因此使cache有较高的命中率。
30. LFU算法：最不经常使用（LeastFrequentlyUsed，LFU）算法将被访问次数最少的cache行淘汰。为此，每行必须设置一个淘汰计数器，其硬件成本较高。新载入的cache行从0开始计数，每命中访问一次，被访问行的计数器加1。当需要替换时，对所有可淘汰行的计数值进行比较，将计数值最小的行淘汰。
31. Cache一致性：需要保证cache与 主存数据的一致性。
32. 写回法：使用写回法，当CPU对cache写命中时，只修改cache的内容而不立即写入主存，只有当此行被替换出cache时才将脏数据写回主存。这种策略使cache在读操作和写操作上都起到高速缓存作用。
33. 写穿法：写穿法也称直写法，其基本思想是当cache写命中时，同时对cache和主存中的同一数据块进行修改，其优点是cache每行无须设置一个修改位以及相应的判别逻辑；而且发生块替换时，被换出的数据块可以直接丢弃，无须写回主存。

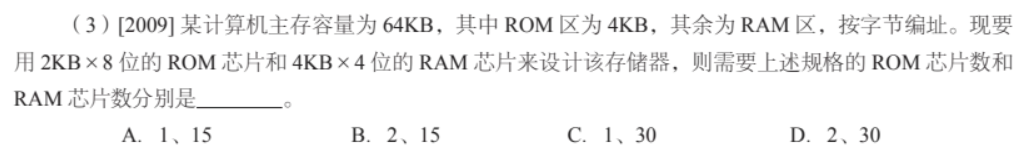
4.2



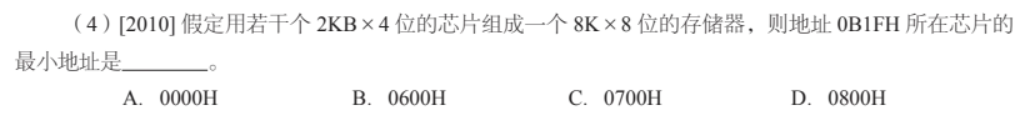
A：RAM和ROM都采用随机存取方式进行信息访问。一般Cache采用高速的SRAM制作，比ROM速度快很多。只有RAM需要刷新。



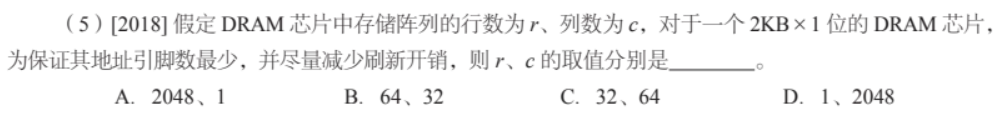
A：4M×8位的芯片数据线应为8根，地址线应为log24M=22根，而DRAM采用地址复用技术，地址线是原来的1/2，且地址信号分行、列两次传送。地址线数为22/2= 11根，所以地址引脚与数据引脚的总数为11+8=19根。如果换为SRAM的话，则地址引脚和数据引脚总数是22+8=30。



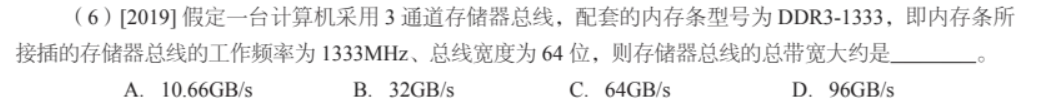
D：首先确定ROM的个数，ROM区为4KB，选用2K×8位的ROM芯片，需要（4K\*8）/（2K\*8）=2片，采用字扩展方式；RAM区为60KB，选用4K×4位的RAM芯片，需要（60K\*8）/（4K\*4）=30片，采用字和位同时扩展方式。



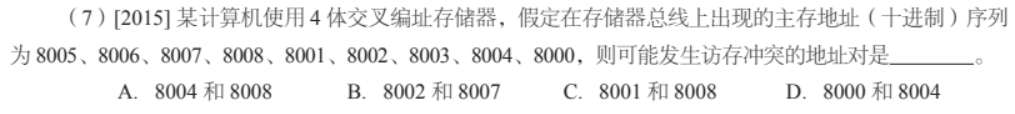
D：芯片的大小为2K×4位，而存储器的大小为8K×8位，需要8片2K×4位的芯片。如果按字节编址，对应一个大小为8K×8位（2^13\*8）的存储器，共13根地址线，一共8片芯片，故应该有三位来表示芯片标号。所以13根地址线中，高3位为片选地址，低10位为片内地址。0B1FH转换为二进制为010 1100011111，片选地址为010（2），那么所在芯片的最小地址为0 1000 00000000即0800H。



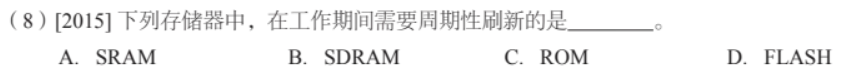
C：根据 DRAM的结构和原理可知，在分时复用的情况下，芯片引脚个数取决于行地址线和列地址线中的较大值。对于一个 2K×1 位的 DRAM 芯片，总共需要 11 条地址线，只有当一个取 5，一个取 6 时可使管脚数最小，而 DRAM 的刷新开销取决于行数，为了减少刷新开销，应该选行数值较少的，因此行地址线应该为5、列地址线为6，即行数 2^5 = 32，列数为 2^6 = 64。



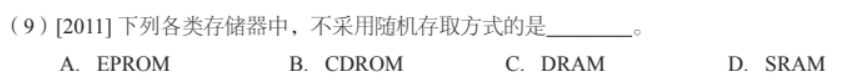
B：总线宽度为64 bits（8字节），则采用三通道的存储器总线的总带宽为：8\*1333\*3=32000MB=32GB。



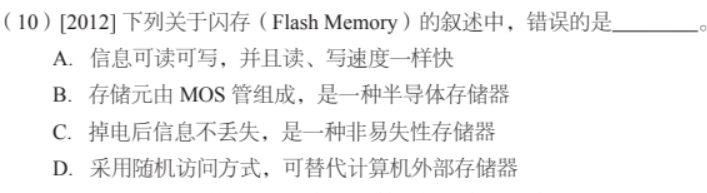
D：主存地址模4后对应的体号分别是：1, 2, 3, 0, 1, 2, 3, 0, 0。模块号=访存地址%存储器交叉模块数 判断可能发生访存冲突的规则是: 给定的访存地址在相邻的4次访问中出现在同一个存储模块内。 因此,根据上表可知8004和8000 对应的模块号都是0,说明这两次访问出现在同一模块内且在相邻访问请求中,会发生冲突。（A选项8004和8008不在一个模块内）



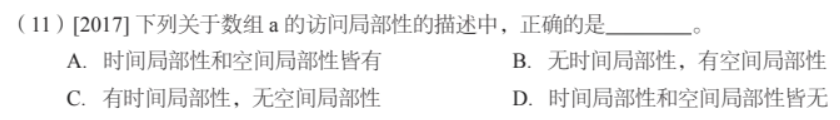
B：SDRAM 表示同步动态随机存储器。DRAM 使用电容存储，所以必须隔一段时间刷新一次，如果存储单元没有被刷新，存储的信息就会丢失。而SRAM则以双稳态为存储单元，不需要周期性的刷新



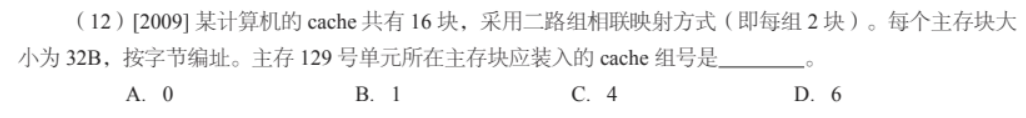
B:EPROM（Erasable Programmable ROM，可擦除可编程ROM）芯片可重复擦除和写入，是随机存取的。CD只读存储器，光盘采用顺序存取方式。DRAM为动态RAM，DRAM保留数据的时间很短，速度也比SRAM慢，不过它还是比任何的ROM都要快，SRAM为静态RAM，SRAM速度非常快，只在要求很苛刻的地方使用，譬如CPU的一级缓冲，二级缓冲。都是随机存取的。



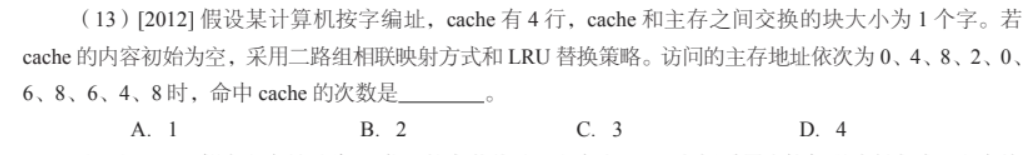
A：闪存的写操作必须在空白区域进行，如果目标区域已经有数据，必须先擦除后写入，而读操作不必如此，所以闪存的读速度比写速度快。其他三项均为闪存的特征。闪存需要将整片或这个分区擦除后才能写入，而且擦除写入时间又长，不能作为一般微机的主存，但可以作为磁盘的补充设备。



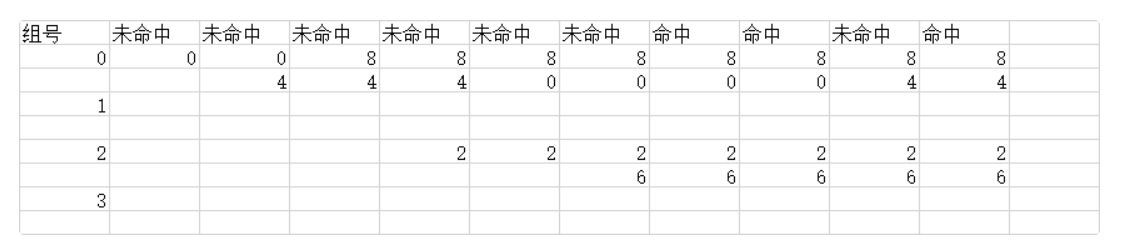
A：时间局部性和空间局部性皆有。

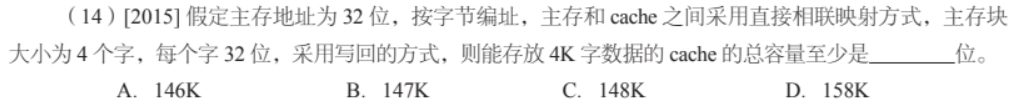


C：每块32B，则129号单元在第五块，组号就是4。



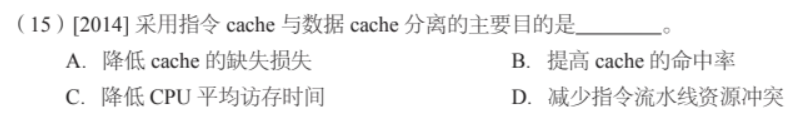
C：4行表示有4个组，每块有一个字，每组有两块。 在组相联映像中，组间是直接映像，组内字块为全相联映像，命中后的字块就是被使用了的字块。

**

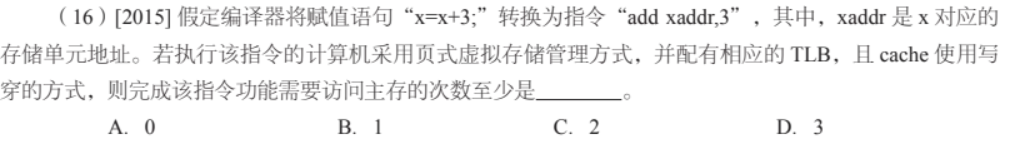


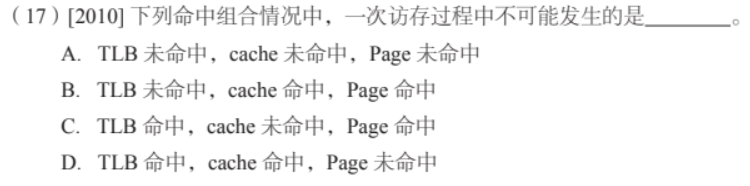
C：每块的Cache和主存区大小相同，所以cache一共需要(4K\*32bit)/(32\*4bit)=1K块，这样所有的容量相加才有4K。Cache存放4K字数据，就是4K\*32bits=4K\*4B=16KB=2^14 B。所以行号+行内索引共14位，则标记位为32（主存地址）-14=18bits。加上2bits的有效位和脏位，共20bits，所以cache总容量=其他（20bits\*1K）+存放的数据（4\*32Kbits）=148K。

(直接相联映射把主存分成若干区，每区与Cache大小相同，地址结构为标记+Cache行号+内块地址。因为是写回（只修改Cache），所以有脏位去标记这一块是否被写了，还有标志位)



D：Cache分为一级L1和二级L2。L1集成在CPU中，称为片内Cache。在L1中还分数据Cache和指令Cache。分别用来存放数据和执行这些数据的指令，两个Cache可以同时被CPU访问，减少了争用Cache所造成的冲突，提高了处理器效能。

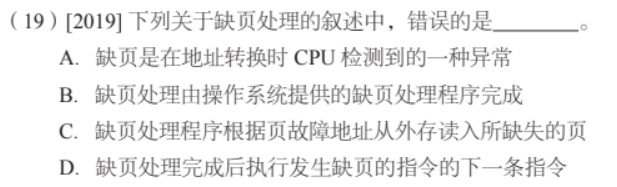
B：上述指令的执行过程可划分为取数、运算和写回过程。取数时如果指令所在的页位于TLB中，则无需访问内存，直接根据TLB中的实页号去找相应的页。因为有cache的存在，所以会先访问cache，此时如果要寻找的页位于cache，则无需访问内存，直接在cache中取出x的值。取出x的值并运算完后，通过写直通法进行写入，因为写直通法是cache和内存一起写入，所以此时必须要访问一次内存，所以答案是1次。



D：TLB即为快表，快表只是慢表(Page)的小小副本，因此TLB命中，必然Page也命中，而当Page命中，TLB则未必命中，故D不可能发生；而Cache的命中与否与TLB、Page的命中与否并无必然联系。

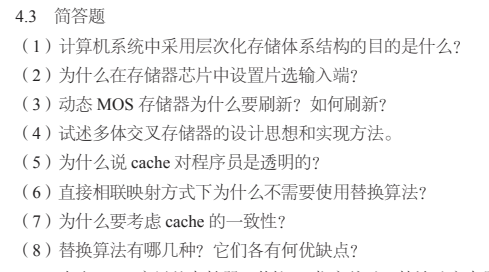


A：虚地址有32位（4GB=2^32B），页面大小为 4KB（2^12B），那么虚页号有20位（2^32/2^12=2^20）。页内地址有12位。实地址有28位（256MB=2^28B），实地址由页框号和页内地址组成，其页内地址与虚地址的页内地址相同都为12位不变，则页框号有16位。虚拟地址为03FF F180H，其中页号为03FFF(20位)，页内地址为180(12位)，根据题目中给出的页表项可知页标记为03FFFH所对应的页框号为0153H，有效位为1，页框号与页内地址之和即为物理地址015 3180 H。



D：缺页处理完成后回到发生缺页的指令。

4.3 简答题



1. 计算机系统中采用层次化存储体系结构的目的是什么?

答：采用层次化存储体系的目的包括两方面:其一是解决快速的CPU和慢速的主存之间的速度差异；其二是解决主存容量不够大的问题。

1. 为什么在存储器芯片中设置片选输入端？

答：由于存储芯片的容量及字长与目标存储器的容量及字长之间可能存在差异，应用存储芯片组织一定容量与字长的存储器时，一般可采用位扩展、字扩展、字位同时扩展等方法来组织。这样就会使用多个存储芯片，从而要设置片选输入端来选择正确的存储芯片来进行操作。

（3）动态MOS存储器为什么要刷新？如何刷新？

答：（1）动态存储单元中，信息以电荷形式存储在栅极电容中。由于电容容量小，所存电荷会在一段时间后逐渐泄漏，为使所存信息能长期保存，需要在电容电荷泄露完之前定时地补充电荷，这一过程称为刷新。（2）①刷新方式：集中刷新、分散刷新和异步刷新。前者存在CPU死时间；分散刷新由于刷新次数过多，降低了存储器的速度；异步刷新是前两者的折中。 ②刷新按行进行,因此设计刷新电路时需要知道动态存储器的内部行、列结构。 ③刷新地址由刷新地址计数提供。

（4）试述多体交叉存储器的设计思想和实现方法。

答：多体交叉存储器由多个存储模块构成，这些模块的容量和存取速度相同，**具有各自独立的地址寄存器、地址译码器、驱动电路和读写控制电路。**根据对多各模块编址方式的不同，又可分为高位多体交叉和低位多体交叉两种方式。(1)高位交叉：按存储器地址的高位地址划分模块，同一存储体内的地址是连续的。当多个目标同时访问存储器时(如CPU和DMA设备同时访问存储器)，如果访问的地方范围处于不同的存储芯片，则提供并行访问。 (2)低位交叉：按存储器地址的低位地址划分模块，同一存储体内的地址不相邻，相邻地址处在不同存储体中。CPU可同时启动多个存储体，并进行并行访问。

1. 为什么说Cache对程序员是透明的？

答：因为在程序员看来，数据是在内存和辅存之间进行交换的，程序员感觉不到中间层cache的存在。

1. 直接映射方式下为什么不需要使用替换算法？

答：因为在直接映射方式中，一个内存块只能固定的映射到cache中的特定行，当有新的主存块调入时, cache特定行中的内容必须调出,因此不需要替换算法去选择替换掉哪一块。

1. 为什么要考虑cache的一致性？

答：正常情况下,cache中的数据是主存数据的副本，当两者不一致时可能导致程序结果不正确，因此，必须考虑并设法保证Cache的一致性。

（8）替换算法有哪几种？他们各有何优缺点？

答：**① 先进先出算法(FIFO)**。（1）基本思想：按照数据块进入Cache的先后决定替换的顺序，即在需要进行替换时，选择最先被调入Cache中的块作为替换块。这种方法要求为每块记录它们进入Cache的先后次序。（2） 优点：FIFO算法系统开销较小。（2） 缺点：是不考虑程序访问的局部性，可能会把一些需要经常使用的块（如循环程序块）也作为最早进入Cache的块而替换掉，因此，可能导致Cache的命中率不高。

**② 近期最少使用(LRU)算法。**（1）基本思想：将近期内长久未被访问过的行换出。为此，每行设置一个计数器，cache每命中一次，命中行计数器清零，其它各行计数器增1，因此它是未访问次数计数器。当需要替换时，比较各特定行的计数值，将计数值最大的行换出。（2） 优点：这种算法显然保护了刚调入Cache的新数据，符合cache工作原理，因而使cache有较高的命中率。LRU算法硬件实现简单。

**③ 最不经常使用(LFU)算法。**（1）基本思想：将一段时间内被访次数最少的那行数据换出。为此，每行设置一个计数器，新调入行的数据从0开始计数，每访问一次被访行的计数器增1。当需要替换时，对这些特定行的计数值进行比较，将计数值最小的行换出。（2） 缺点：一段期间访问情况不能严格反映近期访问情况。例如特定行中的A、B两行，A行在期间的前期多次被访问而后期未被访问，但累积计数值很大，B行是前期不常用而后期却正被频繁访问，但可能由于累积计数小于A行而被LFU算法换出了。

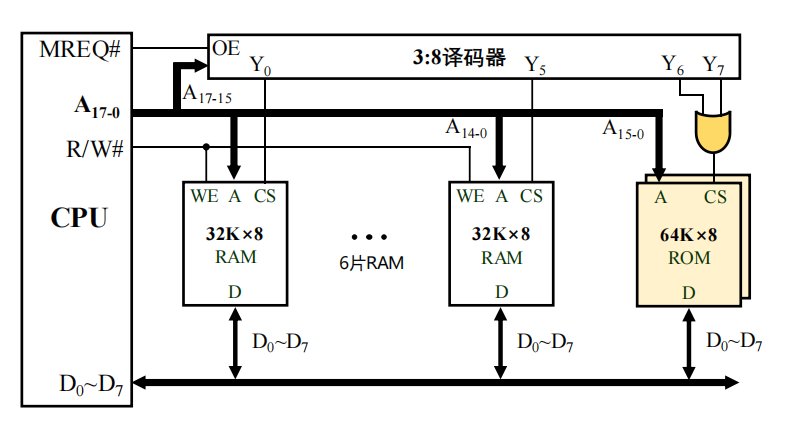
**④ 随机替换算法。**（1）基本思想：需要进行替换时，从特定的行位置中随机地选取一行进行替换。（2）优点：硬件实现最容易，而且速度也比前几种策略快。 （3）缺点：随意换出的数据很可能马上又要用，从而降低命中率和cache工作效率。但这个负面效应随着cache容量增大会减少，模拟研究表明随机替换策略的功效只是稍逊于LFU和LRU。

4.6

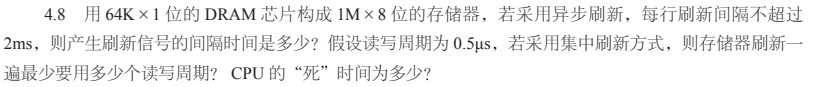


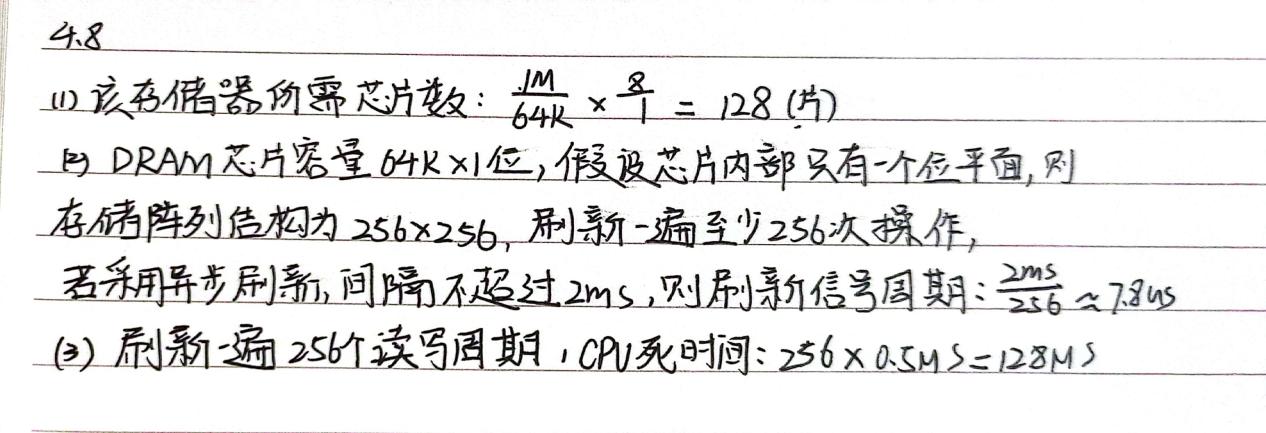
只读区域（ROM）的地址空间为:30000 H-3FFFF H。设计256K×8位储存器，需要64K×4位ROM芯片2片，需要32K×8位RAM芯片的片数为：（256K-64K）/32K=6片。

设计如下：存储器的00000 H-2FFFF H存储空间为RAM芯片，也就是32K×8位RAM芯片6 片，采用字扩展连接。存储空间30000H-3FFFFH使用64K×4位ROM芯片2片，采用位扩展方式连接。数据线条数为8条：D0-D7。地址线的条数为18条：A0-A17，其中A17-A15为片选信号的输入端，A15-0分配给ROM，A14-0分配给RAM。 设计图如下:

**

4.8





4.11



0000 0000 0000 0000 0000 0000

0000 0000 0000 0000 0000 1000

**0000 0001 00**00 0000 0000 0100

**0000 0001** 1111 1111 1111 1100

**1111 1111 1111 1111 1111 10**00

（地址转换为2进制）

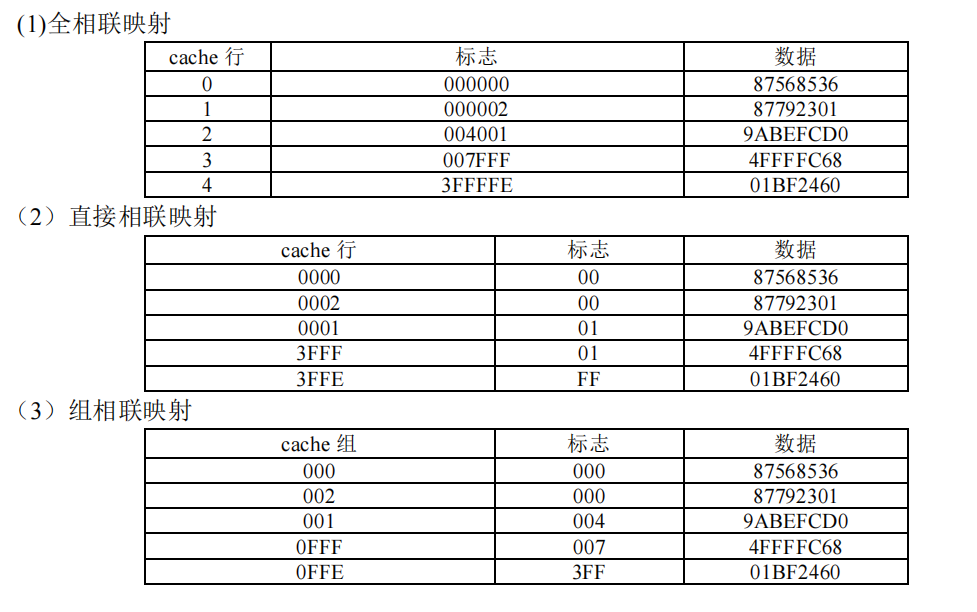
主存容量为2^(14+5-3+8)，即物理地址24位。

采用全相联映射，块内地址为2位(每一块是32位字（4字节，2^2）)，标记位为22位。

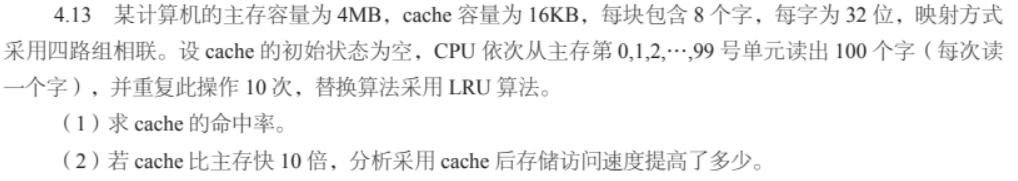
采用直接相连映射，一共14行（块地址14位），块内地址2位，标志位占据8位。

采用四路组相联映射，分为4组，组地址12位（14-2），块内地址为2，标记位10位。

全相联映射：



4.13

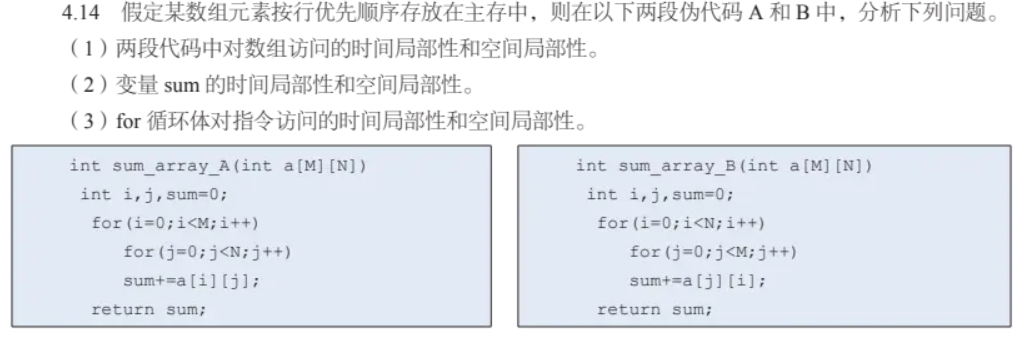


答：分析：每字块有8个字，每字32位（4字节），故主存地址字段中字块内地址为3+2=5位。Cache容量为16KB=2^14B，字块大小为8\*32/8=2^5B，得Cache地址共14位。Cache共有2^9块，根据四路组相联，Cache共分为2^7组（128组）。

1. 每个字块中有8个字，且初态为空，因此CPU读 0 号字单元时，未命中，访问主存，将该字所在的主存块调入Cache第0组的任一一块，其他同理。CPU在第一次读0，8，16……88均未命中，即不命中次数为 100/8 = 13 （上取整）。总访问次数为 100\*10=1000，Cache 命中率 = （ 1000-13 ） /1000=98.7%

（2）设Cache的周期为T，则主存周期为10T。无Cache时访主存需时10T\*100\*10=10000T，加入 Cache 后需时：（1000-13）\*T+13\*10T=1117T。速度提高10000/1117=8.95倍。

4.14



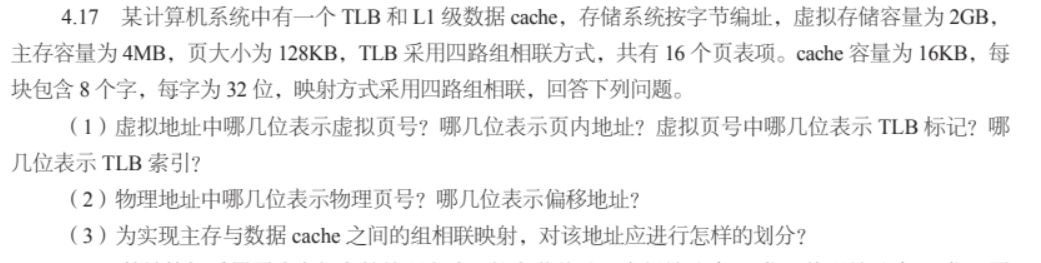
答：(1)对数组访问，程序 A 中具有很好的空间局部性，不存在时间局部性；而程序 B 空间局部性不佳，同样也不存在时间局部性；

(2)变量 sum 在循环中被多次使用，故具有良好的时间局部性。

(3)for 循环中的指令会被反复循环执行，因此具有较好的时间局部性，另外循环体中

的机器指令序列通常会顺序执行，因此也具有一定的空间局部性。

4.17



答：(1)虚拟地址31位（2GB=2^31 B）。页大小128KB（2^17），故0-16（17位）是页内地址。虚拟地址的高 14 位（31-17）表示的是虚页号。TLB为4路组相连，故低 2 位为 TLB 组索引，虚页号的高 12 位（14-2）为 TLB 标记。

虚拟地址=虚页号（TLB标记+TLB索引）+页内地址

1. 物理地址22位（4MB=2^22 B）。低 17 位表示偏移地址，高 5 位为物理页号。[通过虚拟地址的虚页号与物理地址的物理页号（页框号）的对应映射，即将虚拟地址转换为物理地址。]
2. cache的映射方式为直接映射，（块）行大小为8\*32/8(2^5)B，故需要低5位定位块内地址，剩余22-5=17位；共有2^9（2^14/2^5）行，采用四路组相联方式，即共有2^7组，组索引7位，剩下的10位作为标记位（17-7）。

