**计算机组成原理习题答案**

**第1章 计算机系统概述**

3. 存储程序工作方式的基本思想是什么？

答：程序和数据预先存放在存储器中，机器工作时，自动、逐条地从存储器中取出指令并执行。

4. 计算机的硬件由哪些部件组成？它们各有哪些功能？这些部件如何连接？

答：计算机的硬件由运算器、控制器、存储器、输入设备和输出设备组成，其中，存储器由主存及辅存组成，运算器及控制器合称为CPU，CPU和主存合称为主机，CPU只直接访问主存。

输入设备、输出设备负责程序和数据的输入、输出；存储器负责存储程序和数据；运算器负责处理数据；控制器负责指挥和控制各部件协调工作，以实现程序的自动执行。

计算机硬件的各个部件都通过总线互连。

10. 程序执行的过程是什么？计算机中通常采用什么实现方案？

答：程序执行过程是一个循环的指令执行过程，循环变量为指令地址，指令执行过程又可分为取指令、分析指令、执行指令三个阶段。

计算机通常用寄存器PC存放指令地址，用寄存器IR存放当前指令内容。指令执行过程中，取指令阶段用PC的内容作为地址从主存中读出指令，并存放到IR中；分析指令阶段用ID（指令译码器）分析出IR中内容包含的操作类型、操作数信息；执行指令阶段根据指令分析的结果，实现相应的约定操作。

16. 计算机M的时钟频率为2 GHz；指令集包含A、B两类指令，指令长度都为2个字节，指令执行时的CPI分别为5和8。某程序P的大小为2 MB，其中40%为A类指令，其余为B类指令。程序P执行时，10%的A类指令和20%的B类指令分别执行了20次，其余指令各执行了1次。求程序P执行时的CPU时间及CPI。

答：程序P中共包含2MB/2B＝220条指令，其中，A类指令为0.4×220条，B类指令为0.6×220条。

P执行时，A类指令执行了0.4×220×(10%×20＋90%×1)＝1.16×220条，B类指令执行了0.6×220×(20%×20＋80%×1)＝2.88×220条，共执行了1.16×220＋2.88×220＝4.04×220条指令。

因此，TCPU＝(1.16×220×5＋2.88×220×8)/(2×109)＝14.42×220×10-9≈15.12ms；

CPI＝(1.16×220×5＋2.88×220×8)/(4.04×220)＝28.84/4.04≈7.13。

**第2章 数据的表示与运算**

2. 完成下列不同进制数之间的转换。

（1）(347.625)10＝( )2＝( )8＝( )16

（2）(9C.E)16＝( )2＝( )8＝( )10

（3）(11010011)2＝( )10＝( )BCD

答：（1）(347.625)10＝(101011011.101)2＝(533.5)8＝(15B.A)16

（2）(9C.E)16＝(10011100.1110)2＝(234.7)8＝(156.875)10

（3）(11010011)2＝(211)10＝(001000010001)8421BCD

3. 对下列十进制数，分别写出用8位机器数表示时的原码及补码。

（1）＋23/128 （2） －35/64 （3） 43 （4） －72

（5）＋7/32 （6） －9/16 （7）＋91 （8） －33

答：（1）[+23/128]原＝0.0010111， [+23/128]补＝0.0010111；

（2）[-35/64]原＝1.1000110， [-35/64]补＝1.0111010；

（3）[43]原＝00101011， [43]补＝00101011；

（4）[-72]原＝11001000， [-72]补＝10111000；

（5）[+7/32]原＝0.0011100， [+7/32]补＝0.0011100；

（6）[-9/16]原＝1.1001000， [-9/16]补＝1.0111000；

（7）[+91]原＝01011011， [+91]补＝01011011；

（8）[-33]原＝10100001， [-33]补＝11011111。

4. 对下列机器数，为原码时求补码及真值，为补码或反码时求原码及真值。

（1）[*X*]原＝100011 （2）[*X*]补＝0.00011 （3）[*X*]反＝1.01010

（4）[*X*]原＝1.10011  （5）[*X*]补＝101001  （6）[*X*]反＝101011

答：（1）[*X*]补＝111101，*X*＝-00011＝-3；

（2）[*X*]原＝0.00011，*X*＝+0.00011＝+3/32；

（3）[*X*]原＝1.10101，*X*＝-0.10101＝-21/32；

（4）[*X*]补＝1.01101，*X*＝-0.10011＝-19/32；

（5）[*X*]原＝110111，*X*＝-10111＝-23/32；

（6）[*X*]原＝110100，*X*＝-10100＝-20/32。

5. 回答下列问题。

（1）若[*X*]补＝1.01001，求[－*X*]补及*X*；

（2）若[－*X*]补＝101001，求[*X*]补及*X*。

答：（1）[-*X*]补＝0.10111，*X*＝-0.10111＝-23/32。

（2）[*X*]补＝010111，*X*＝+10111＝+23。

6. 回答下列问题。

（1）若*X*＝＋23及－42，分别求8位长度的[*X*]移；

（2）若[*X*]移＝1100101及0011101，分别求*X*。

答：（1）[+23]移＝10010111，[-42]移＝01010110。

（2）[*X*]移＝1100101时的*X*＝+100101＝+37，[*X*]移＝0011101时的*X*＝-100011＝-35。

10. 在奇偶校验码10001101、01101101、10101001中，若只有一个有错误，请问校验码采用的是奇校验还是偶校验？为什么？

答：上述奇偶校验码采用的是偶校验编码方式。

因为3个奇偶校验码中1的个数分别有4个、5个、4个，而只有一个校验码有错误，故第2个奇偶校验码(奇数个1)有错误，因此，校验码采用的是偶校验编码方式。

11. 对于8位数据01101101，求其海明校验码，要求写出过程。

答：（1）先求校验信息位数*k*，根据2*k*－1≥8＋*k*的要求，可得*k*＝4位。

（2）列出*n*＋*k*＝8＋4＝12位校验码中的信息排列：*m*8*m*7*m*6*m*5*p*4*m*4*m*3*m*2*p*3*m*1*p*2*p*1。

（3）设各校验组采用偶校验编码方式，各校验组校验位的值为：

*p*4＝*m*8⊕*m*7⊕*m*6⊕*m*5＝0⊕1⊕1⊕0＝0，

*p*3＝*m*8⊕*m*4⊕*m*3⊕*m*2＝0⊕1⊕1⊕0＝0，

*p*2＝*m*7⊕*m*6⊕*m*4⊕*m*3⊕*m*1＝1⊕1⊕1⊕1⊕1＝1，

*p*1＝*m*7⊕*m*5⊕*m*4⊕*m*2⊕*m*1＝1⊕0⊕1⊕0⊕1＝1；

（4）海明校验码为：011001100111。

14. 对8位长度的定点整数9BH及FFH，分别写出它们采用原码、补码、移码、无符号编码时的真值。

答：机器数表示整数时，9BH＝[-27]原＝[-101]补＝[+27]移＝[155]无；

FFH＝[-127]原＝[-1]补＝[+127]移＝[255]无。

15. 浮点表示格式中，阶码为6位、尾数为10位，可以采用下列编码方式，分别将51/128、－27/1024、7.375、－86.5转换为浮点数。

答：（1）阶码和尾数均为原码时，[51/128]浮＝[0.0110011]浮＝100001 0110011000，

[-27/1024]浮＝[-0.0000011011]浮＝100101 1110110000，

[7.375]浮＝[111.011]浮＝000011 0111011000，

[-86.5]浮＝[-1010110.1]浮＝000111 1101011010。

（2）阶码和尾数均为补码时，[51/128]浮＝111111 0110011000，

[-27/1024]浮＝111011 1001010000，

[7.375]浮＝000011 0111011000，

[-86.5]浮＝000111 1010100110。

17. 若浮点表示格式中，阶码为6位、尾数为10位，可以采用下列编码方式，分别写出浮点数E796H、E696H的规格化数。

（1）阶码和尾数都用原码表示 （2）阶码和尾数都用补码表示

答：E796H＝1110011110010110B，E696H＝1110011010010110B，

（1）阶码和尾数均为原码时，

E796H的尾数[*M*]原＝1110010110，0.5≤|*M*|＜1，故规格化数为E796H；

E696H的尾数[*M*]原＝1010010110，0＜|*M*|＜0.5，需左规1次，

左规后尾数为1100101100，阶码为111000，故规格化数为E32CH。

（2）阶码和尾数均为补码时，

E796H的尾数[*M*]补＝1110010110，0＜|*M*|＜0.5，需左规2次，

左规后尾数为1001011000，阶码为110111，故规格化数为DE58H；

E696H的尾数[*M*]补＝1010010110，0.5≤|*M*|＜1，故规格化数为E696H。

18. 若浮点数采用IEEE 754标准表示，回答下列问题。

（1）写出浮点数99D00000H及59800000H的真值；

（2）分别将－51/128、28.75转换为单精度浮点数。

答：（1）机器码99D00000H＝1 00110011 10100000000000000000000B，

故浮点数的数符*S*＝1、阶码*E*＝00110011、尾数*M*＝10100000000000000000000，

因1＜*E*＜255，故机器码表示的为规格化浮点数，

99D00000H的真值*N*＝(-1)1×251-127×1.10100000000000000000000＝-0.1101×2-75。

机器码59800000H＝0 10110011 00000000000000000000000B，

故浮点数的数符*S*＝0、阶码*E*＝10110011、尾数码*M*＝00000000000000000000000，

因1＜*E*＜255，故机器码表示的为规格化浮点数，

59800000H的真值*N*＝(-1)0×2179-127×1.00000000000000000000000＝+0.1×2+53。

（2）－51/128＝(+110011/10000000)2＝＋1.10011×2-2，阶码的真值为－2，尾数的真值为0.10011；因此，浮点数格式中，数符*S*＝1，阶码*E*＝127＋(－2)＝01111101B，尾数*M*＝100110…0；单精度浮点数为 1 01111101 10011000000000000000000＝3ECC0000H。

28.75＝(+111000.11)2＝＋1.1100011×2+5，阶码的真值为+5，尾数的真值为0.1100011；因此，浮点数格式中，数符*S*＝0，阶码*E*＝127＋(+5)＝10000100B，尾数*M*＝11000110…0；单精度浮点数为 0 10000100 11000110000000000000000＝42630000H。

23. 若下列*A*和*B*用8位补码表示，求[*A*＋*B*]补及[*A*－*B*]补，并判断结果是否溢出。

（1）*A*＝－87，*B*＝13 （2）*A*＝115，*B*＝－24

答：（1）因*A*＝-1010111、*B*＝+0001101，

则[*A*]补＝1 0101001、[*B*]补＝0 0001101、[-*B*]补＝1 1110011，

[*A*＋*B*]补＝1 0101001＋0 0001101＝1 0110110，OF=(1⊕1)(0⊕1)＝0，不溢出；

[*A*－*B*]补＝1 0101001＋1 1110011＝1 0011100，OF=(1⊕1)(1⊕1)＝0，不溢出。

（2）因*A*＝+1110011、*B*＝-0011000，

则[*A*]补＝0 1110011，[*B*]补＝1 1101000，[-*B*]补＝0 0011000，

[*A*＋*B*]补＝0 1110011＋1 1101000＝0 1011011，OF=(0⊕0)(1⊕0)＝0，不溢出；

[*A*－*B*]补＝0 1110011＋0 0011000＝1 0001011，OF=(0⊕1)(0⊕1)＝1，溢出。

26. 若下列*A*和*B*用6位无符号编码表示，求*A*－*B*，并判断结果是否有进位/借位。

（1）*A*＝38，*B*＝22 （2）*A*＝38，*B*＝45

答：（1）因*A*＝100110、*B*＝010110，则[*A*]无＝100110、[*B*]补数＝101010，

[*A*－*B*]无＝100110＋101010＝010000，*A*－*B*＝16，CF=1⊕1=0，无借位。

（2）因*A*＝100110、*B*＝101101，则[*A*]无＝100110，[*B*]补数＝010011，

[*A*－*B*]无＝100110＋010011＝111001，*A*－*B*＝57，CF=0⊕1=1，有借位。

27. 已知机器数*a*＝00101000，假设*a* <<L *n*发生溢出，则*n*的最小值是多少？假设((*a* >>L *n*) <<L *n*)≠*a*，则*n*的最小值又是多少？请分别说明原因。

答：逻辑左移移丢码1时溢出，因此，*n*最小值为3。

逻辑右移移丢码1时影响精度，逻辑左移时低位补0，因此，*n*最小值为4。

30. 对下列*A*和*B*，用原码乘法求*A*×*B*。

（1）*A*＝19，*B*＝35 （2）*A*＝0.110111，*B*＝－0.101110

答：（1）由题意，[*A*]原＝0010011，[*B*]原＝0100011，|*A*|＝010011，|*B*|＝100011，

[*A*×*B*]原的符号位为0⊕0＝0，

|*A*|×|*B*|需进行6次判断-加法-移位操作，其过程如下表所示：

|  |  |  |  |
| --- | --- | --- | --- |
| 循环次数 | 部分积高位 | 乘数  (及部分积低位) | 说明 |
| 6 | 000000 | 1 0 0 0 1 1 | 初始部分积*P*0＝000000 |
| 5 | ＋ 010011  0 010011  001001 | 1 1 0 0 0 1 | 乘数最低位为1，应＋|*A*|  6位加法，0为加法器的进位  部分积及乘数同时右移1位 |
| 4 | ＋ 010011  0 011100  001110 | 0 1 1 0 0 0 | 乘数最低位为1，应＋|*A*|  部分积及乘数同时右移1位 |
| 3 | ＋ 000000  0 001110  000111 | 0 0 1 1 0 0 | 乘数最低位为0，应＋0  部分积及乘数同时右移1位 |
| 2 | ＋ 000000  0 000111  000011 | 1 0 0 1 1 0 | 乘数最低位为0，应＋0  部分积及乘数同时右移1位 |
| 1 | ＋ 000000  0 000011  000001 | 1 1 0 0 1 1 | 乘数最低位为0，应＋0  部分积及乘数同时右移1位 |
| 0 | ＋ 010011  0 010100  001010 | 0 1 1 0 0 1 | 乘数最低位为1，应＋|*A*|  部分积及乘数同时右移1位 |

即*P*＝|*A*|×|*B*|＝001010011001，整数*P*扩展1位的值＝0001010011001，

故[*A*×*B*]原＝0 0001010011001，共14位。

（2）由题意，[*A*]原＝0.110111，[*B*]原＝1.101110，|*A*|＝0.110111，|*B*|＝0.101110，

[*A*×*B*]原的符号位为0⊕1＝1，

|*A*|×|*B*|需进行6次判断-加法-移位操作，其过程如下表所示：

|  |  |  |  |
| --- | --- | --- | --- |
| 循环次数 | 部分积高位 | 乘数  (及部分积低位) | 说明 |
| 6 | 0.000000 | 1 0 1 1 1 0 | 初始部分积*P*0＝0.000000 |
| 5 | ＋ 0.000000  0.000000  0.000000 | 0 1 0 1 1 1 | 乘数最低位为0，应＋0  部分积及乘数同时右移1位 |
| 4 | ＋ 0.110111  0.110111  0.011011 | 1 0 1 0 1 1 | 乘数最低位为1，应＋|*A*|  部分积及乘数同时右移1位 |
| 3 | ＋ 0.110111  1.010010  0.101001 | 0 1 0 1 0 1 | 乘数最低位为1，应＋|*A*|  部分积及乘数同时右移1位 |
| 2 | ＋ 0.110111  1.100000  0.110000 | 0 0 1 0 1 0 | 乘数最低位为1，应＋|*A*|  部分积及乘数同时右移1位 |
| 1 | ＋ 0.000000  0.110000  0.011000 | 0 0 0 1 0 1 | 乘数最低位为0，应＋0  部分积及乘数同时右移1位 |
| 0 | ＋ 0.110111  1.001111  0.100111 | 1 0 0 0 1 0 | 乘数最低位为1，应＋|*A*|  部分积及乘数同时右移1位 |

即*P*＝|*A*|×|*B*|＝0.100111100010，小数*P*扩展1位的值＝0.1001111000100，

故[*A*×*B*]原＝1.1001111000100，共14位。

31. 对下列*A*和*B*，用Booth算法求*A*×*B*。

（1）*A*＝19，*B*＝35 （2）*A*＝0.110111，*B*＝－0.101010

答：（1）由题意[*A*]补＝0010011，[*B*]补＝0100011，[-*A*]补＝1101101，

整数[*A*×*B*]补需进行7次判断-加法-移位操作，其过程如下表所示：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 循环 | 部分积 | 乘数 | 附加位 | 操作说明 |
| 7 | 0000000 | 0 1 0 0 0 1 1 | 0 | 初始[*P*0]补＝0，*b-*1＝0 |
| 6 | ＋1101101  1101101  1110110 | 1 0 1 0 0 0 1 | 1 | *b*0*b-*1＝10，部分积＋[-*A*]补  部分积和乘数同时算术右移1位 |
| 5 | ＋0000000  1110110  1111011 | 0 1 0 1 0 0 0 | 1 | *b*0*b-*1＝11，部分积＋0  部分积和乘数同时算术右移1位 |
| 4 | ＋0010011  0001110  0000111 | 0 0 1 0 1 0 0 | 0 | *b*0*b-*1＝01，部分积＋[*A*]补  部分积和乘数同时算术右移1位 |
| 3 | ＋0000000  0000111  0000011 | 1 0 0 1 0 1 0 | 0 | *b*0*b-*1＝00，部分积＋0  部分积和乘数同时算术右移1位 |
| 2 | ＋0000000  0000011  0000001 | 1 1 0 0 1 0 1 | 0 | *b*0*b-*1＝00，部分积＋0  部分积和乘数同时算术右移1位 |
| 1 | ＋1101101  1101110  1110111 | 0 1 1 0 0 1 0 | 1 | *b*0*b-*1＝10，部分积＋[-*A*]补  部分积和乘数同时算术右移1位 |
| 0 | ＋0010011  0001010  0000101 | 0 0 1 1 0 0 1 |  | *b*0*b-*1＝01，部分积＋[*A*]补  部分积和乘数同时算术右移1位 |

故[*A*×*B*]补＝00001010011001，共14位。

（2）由题意[*A*]补＝0.110111，[*B*]补＝1.010110，[-*A*]补＝1.001001，

小数[*A*×*B*]补需进行7次判断-加法-移位操作(最后1次不移位)，其过程如下表所示：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 循环 | 部分积 | 乘数 | 附加位 | 操作说明 |
| 7 | 0.000000 | 1 0 1 0 1 1 0 | 0 | 初始[*P*0]补＝0，*b-n*＝0 |
| 6 | ＋0.000000  0.000000  0.000000 | 0 1 0 1 0 1 1 | 0 | *b-*(*n*-1)*b-n*＝00，部分积＋0  部分积和乘数同时算术右移1位 |
| 5 | ＋1.001001  1.001001  1.100100 | 1 0 1 0 1 0 1 | 1 | *b-*(*n*-1)*b-n*＝10，部分积＋[-*A*]补  部分积和乘数同时算术右移1位 |
| 4 | ＋0.000000  1.100100  1.110010 | 0 1 0 1 0 1 0 | 1 | *b-*(*n*-1)*b-n*＝11，部分积＋0  部分积和乘数同时算术右移1位 |
| 3 | ＋0.110111  0.101001  0.010100 | 1 0 1 0 1 0 1 | 0 | *b-*(*n*-1)*b-n*＝01，部分积＋[*A*]补  部分积和乘数同时算术右移1位 |
| 2 | ＋1.001001  1.011101  1.101110 | 1 1 0 1 0 1 0 | 1 | *b-*(*n*-1)*b-n*＝10，部分积＋[-*A*]补  部分积和乘数同时算术右移1位 |
| 1 | ＋0.110111  0.100101  0.010010 | 1 1 1 0 1 0 1 | 0 | *b-*(*n*-1)*b-n*＝01，部分积＋[*A*]补  部分积和乘数同时算术右移1位 |
| 0 | ＋1.001001  1.011011  1.011011 | 1 1 1 0 1 0 |  | *b-*(*n*-1)*b-n*＝10，部分积＋[-*A*]补  最后一次不右移(位扩展需末位补0) |

乘积扩展1位时末位补0，故[*A*×*B*]补＝1.0110111110100，共14位。

34. 若浮点表示格式中，尾数用8位补码表示，阶用5位移码表示。浮点运算时，采用双符号位运算，附加位为3位，采用舍入法，请用浮点运算方法计算下列表达式。

（1）[215×11/16]＋[213×(－9/16)]

（2）[2－13×13/16]＋[2－14×(－5/8)]

答：（1）由题意[*A*]浮＝11111;0.1011000，[*B*]浮＝11101;1.0111000，运算过程如下：

①对阶，[*EB*]补＝01101，[-*EB*]补＝10011，

[Δ*E*]移＝[*EA*]移－[*EB*]补＝[*EA*]移＋[-*EB*]补＝11 1111＋11 0011＝11 0010，

Δ*E*＝2，*B*的尾数右移2位、阶码加2，

即[*MA*′]补＝00.1011000 000，[*MB*′]补＝11.1101110 000，[*EF*′]移＝11 1111。

②尾数加减，[*MA*′]补＋[*MB*′]补＝00.1011000 000＋11.1101110 000＝00.1000110 000，

即[*A*′＋*B* ′]浮＝11 1111;00.1000110 000。

③规格化，尾数不溢出、符号位与最高数值位不同，尾数无需规格化，

即[*A*′＋*B* ′]浮＝11 1111;00.1000110 000。

④尾数舍入，由于警戒位为000，选择“舍”，

即[*A*′＋*B* ′]浮＝11 1111;00.1000110

⑤溢出判断，阶码符号位为11，*A*＋*B*结果不溢出。

故[*A*＋*B*]浮＝11111;0.1000110。

（2）由题意[*A*]浮＝00011;0.1101000，[*B*]浮＝00010;1.0110000，运算过程如下：

①对阶，[*EB*]补＝10010，[-*EB*]补＝01110，

[Δ*E*]移＝[*EA*]移－[*EB*]补＝[*EA*]移＋[-*EB*]补＝00 0011＋00 1110＝01 0001，

Δ*E*＝1，*B*的尾数右移1位、阶码加1，

即[*MA*′]补＝00.1101000000，[*MB*′]补＝11.1011000000，[*EF*′]移＝00 0011

②尾数加减，[*MA*′]补＋[*MB*′]补＝00.1101000 000＋11.1011000 000＝00.1000000 000，

即[*A*′＋*B* ′]浮＝00 0011;00.1000000 000

③规格化，尾数不溢出、符号位与最高数值位不同，尾数无需规格化，

即[*A*′＋*B* ′]浮＝00 0011 00.1000000 000

④尾数舍入，由于警戒位为000，选择“舍”，

即[*A*′＋*B* ′]浮＝00 0011;00.1000000

⑤溢出判断，阶码符号位为00，*A*＋*B*结果不溢出。

故[*A*＋*B*]浮＝00011;0.1000000。

36. 图2.38中*F*＝0信号的作用是什么？画出图中Zero方框的内部逻辑。

答：*F*=0信号的功能是表示运算结果是否为零，结果为零时“*F*=0”有效，否则无效。它可以用于实现关系运算A＝B及A≠B。

Zero方框的内部逻辑为：输出＝。

**第3章 存储系统**

3. RAM芯片为什么常采用双译码方式？为什么要设置片选引脚？SRAM芯片为什么要求地址信号先于建立、后于撤销？读操作、写操作的时序有什么不同？

答：将所有存储元组织成二维矩阵形式，可以使存储元间连线长度最短、线路信号延迟最小，从而提高性能、降低成本；相应地，该方式要求RAM芯片采用二维译码方式。

RAM芯片的规格有限，需允许用户利用已有芯片进行二次设计，才能满足用户千变万化的需求；相应地，这种方法要求RAM芯片设置片选引脚，来实现芯片的操作控制。

有效期间时，SRAM将根据信号进行读/写操作；若有效时地址信号尚未稳定，则会对多个存储单元(含非目标存储单元)进行读/写操作，而写操作可能会改变原有内容，因此，SRAM芯片要求地址信号先于建立、后于撤销。

两种操作都是先发送地址信号，后使有效、发送操作命令，读操作在操作结束前接收数据，写操作在操作开始时发送数据**。**

4. 16K×4位SRAM芯片的引脚有哪些？

答：SRAM芯片的地址引脚有log2(16×210)＝14根，数据引脚有4根，还有片选引脚及读写引脚(低电平为写、高电平为读)。

6. DRAM芯片为什么要设置、引脚，为什么没有引脚？

答：为了保持高集成度，DRAM芯片的地址都分两次传送，需用、两个信号来区分当前地址信号是行地址还是列地址。而使信号在整个操作过程中都有效，就可以实现片选引脚功能，无需设置引脚。

13. 现有64K×4位ROM芯片、32K×8位SRAM芯片若干，要求构成256K×8位的存储模块，前64KB为ROM空间，画出存储模块的信号线与内部各个芯片引脚的连接图。

答：存储模块需要的ROM芯片为(64K/64K)×(8b/4b)＝2片，需要的RAM芯片为(192K/32K)×(8b/8b)＝6片。各芯片在存储空间中的地址范围如下图左部所示。

可见，存储模块的log2(256KB/1B)＝18根地址引脚中，A17~A16＝00选中ROM0、ROM1芯片，A17~A15＝010~111分别选中RAM2、RAM7芯片。因此，用3-8译码器选择各芯片时，ROM0、ROM1片选引脚的逻辑为，RAM2~RAM7片选引脚的逻辑为~。

因此，存储模块的信号线与内部各芯片引脚的连接如下图右部所示，其中ROM芯片无需连接信号。



17. 某计算机中，CPU有20根地址引脚（A19～A0）、8根数据引脚（D7～D0），控制引脚由、IO/及R/组成。若配置256KB的主存（SRAM），放在CPU可寻址空间的低端。请回答下列问题：

（1）主存的地址线、数据线各有多少位？

（2）主存连接到CPU时，其片选信号的有效逻辑是什么？画出主存各引脚与CPU的连接图。

答：（1）由于CPU的数据引脚宽度为8位，故主存的数据线有8位，地址线有log2(256KB/1B)＝18位。

（2）主存在“CPU有存储器操作、操作地址在主存的地址范围内”时被选中，因此，主存片选信号的有效逻辑是。CPU与主存的连接如下图所示。



20. 某4K×8位SRAM芯片的存储周期为200ns，该芯片的带宽是多少？由4个上述芯片构成的16K×8位多体交叉存储器，若采用交叉访问方式，则访问64个字节最少需要多少个存储周期？启动各存储体轮流工作的时钟频率是多少？

答：SRAM芯片的带宽为：8bit/(200×10-9s)＝4×107bit/s＝40Mbps。

4个交叉存储器采用交叉访问方式工作时，各存储体轮流启动的间隔为200ns/4＝50ns，访问64个字节最少需要200ns＋(64－1)×50ns＝3350ns＝3.35μs。

启动各存储体轮流工作的时钟频率为：1/(200ns/4)＝2×107Hz＝20MHz。

22. 某计算机中，主存按字节编址，CPU有20根地址引脚、8根数据引脚，配置有64KB的Cache，Cache与主存采用直接映射方式，主存块大小为16B。回答下列问题：

（1）为了实现映射，主存地址应该如何划分？各个字段分别为多少位？

（2）每个Cache行的标记为多少位？说明理由。

（3）若访存地址分别为2D058H和2D078H，Cache命中时的标记分别是多少？

答：由题意，主存单元长度为8位，主存地址为20位，Cache行号为log2(64KB/16B)＝12位，主存块内地址为log2(16B/1B)＝4位。

（1）由于采用直接映射方式，故Cache地址由行号(12b)、块内地址(4b)组成，主存地址由区号、区内块号、块内地址(4b)组成，其中区内块号为12b（与Cache行号位数相同），区号为20―12―4＝4位。

（2）每个Cache行的标记为4位。由于直接映射方式时，主存地址中区内块号用于索引，故主存地址中区号用作Cache行的标记，地址变换时用来进行比较。

（3）主存地址2D058H＝0010 1101 0000 0101 1000B中，区号为0010B，区内块号为1101 0000 0101B，故Cache命中时的标记为0010B；

主存地址2D078H＝0010 1101 0000 0111 1000B中，区号为0010B，区内块号为1101 0000 0111B，故Cache命中时的标记为0010B。

24. 将题22中Cache改用4路相联映射方式，回答与题22相同的问题。

答：由题意，主存单元长度为8位，主存地址为20位，Cache行号为log2(64KB/16B)＝12位，主存块内地址为log2(16B/1B)＝4位。

（1）由于采用4路组相联映射方式，故Cache的组内行号为log24＝2位，组号为12―2＝10位，Cache地址由组号(10b)、组内行号(2b)、块内地址(4b)组成，主存地址由群号、群内块号、块内地址(4b)组成，其中群内块号为10b（与Cache组号位数相同），群号为20―10―4＝6位。

（2）每个Cache行的标记为6位。由于4路组相联映射方式时，主存地址中群内块号用于索引，故主存地址中群号用作Cache行的标记，地址变换时用来进行比较。

（3）主存地址2D058H＝0010 1101 0000 0101 1000B中，群号为0010 11B，群内块号为01 0000 0101B，故Cache命中时的标记为001011B；

主存地址2D078H＝0010 1101 0000 0111 1000B中，区号为001011B，群内块号为01 0000 0111B，故Cache命中时的标记为001011B。

25. 某2路组相联Cache有4个行，采用LRU替换算法，主存块大小为8个字。假设Cache初态为空，CPU先从地址0000H起升序连续访问48个字，再从地址002FH起降序连续访问48个字，每次访问1个字，求此时的Cache命中率。

答：由题意，CPU升序访问的48个字放在⎡48＋(0000H%8)/8⎤＝6个主存块中（块号为0~5）；由于002FH＋1－48＝0000H，故CPU降序访问的48个字也放在0#~5#主存块中。

由于是2路组相联映射，故Cache有4/2＝2个组（组号为0和1）；主存块号中后2位为群内块号，其余为群号（假设为2位），故0#、2#、4#主存块映射到Cache的0#组中，群号分别为0、0、1；1#、3#、5#主存块映射到Cache的1#组中，群号分别为0、0、1。

由于CPU按连续地址访问主存，故CPU访存的块地址流为：0、1、2、3、4、5、5、4、3、2、1、0，每个块连续访问8次，每个块第1次访问时缺失、其余次命中。

采用LRU替换算法时，每个行的LRU位为log22＝1位。Cache处理CPU访问的过程如下表所示，其中单元格中“/”左边为块号（为\*时表示有效位为0）、右边为LRU位的值。

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 块地址流 | | | 0 | 1 | 2 | 3 | 4 | 5 | 5 | 4 | 3 | 2 | 1 | 0 |
| 行状态 | 组0 | 行0 | 0/0 | 0/0 | 0/1 | 0/1 | 4/0 | 4/0 | 4/0 | 4/0 | 4/0 | 4/1 | 4/1 | 0/0 |
| 行1 | \*/1 | \*/1 | 2/0 | 2/0 | 2/1 | 2/1 | 2/1 | 2/1 | 2/1 | 2/0 | 2/0 | 2/1 |
| 组1 | 行2 | \*/1 | 1/0 | 1/0 | 1/1 | 1/1 | 5/0 | 5/0 | 5/0 | 5/1 | 5/1 | 1/0 | 1/0 |
| 行3 | \*/1 | \*/1 | \*/1 | 3/0 | 3/0 | 3/1 | 3/1 | 3/1 | 3/0 | 3/0 | 3/1 | 3/1 |
| 操作  状态 | 第1次 | | 调入 | 调入 | 调入 | 调入 | 替换 | 替换 | 命中 | 命中 | 命中 | 命中 | 替换 | 替换 |
| 第2~8次 | | 命中 | 命中 | 命中 | 命中 | 命中 | 命中 | 命中 | 命中 | 命中 | 命中 | 命中 | 命中 |

CPU的48\*2＝96次访存中，有8次不命中，则Cache命中率为(96－8)/96＝0.917。

28. 某计算机的存储器按字节编址，主存地址空间为24位，配置有4MB的主存，主存块大小为32B，Cache有256个行，采用4路组相联映射、LRU替换算法、写回法写策略，Cache行的管理信息至少有多少位？

答：采用4路组相联映射时，Cache组内行号为log24＝2位，组号为log2256―2＝6位；主存地址由群号、群内块号、块内地址组成，其中，块内地址为log2(32B/1B)＝5位，群内块号为6位（与Cache组号相同），群号为24―6―5＝13位。故Cache行的标记为13位。

采用LRU替换算法时，每个Cache行的LRU位为log24＝2位。

采用写回法写策略时，每个Cache行需设置脏位（1位）。

因此，每个Cache行的管理信息至少有：有效位(1位)、标记(13位)、LRU位(2位)及脏位(1位)，共17位。

29. 某Cache-主存层次中，Cache、主存的存取周期分别为30ns、150ns，Cache读/写一个主存块的时间为600ns。CPU执行某程序时，共有700次读操作、300次写操作，其中，读操作缺失50次，写操作缺失15次。若忽略块替换带来的损失时间，请分别计算Cache采用全写法、写回法写策略时的平均访问时间。

答：TA＝T命中＋(1－H)×T缺失，T命中＝(T读命中×Num读＋T写命中×Num写)/(Num读＋Num写)，(1－H)×T缺失＝(T读缺失×Num读缺失＋T写缺失×Num写缺失)/(Num读＋Num写)，其中，Num读、Num写、Num读缺失、Num写缺失分别为读操作、写操作、读操作缺失、写操作缺失的次数。

依题意，Num读＝700、Num写＝300、Num读缺失＝50、Num写缺失＝15，T读命中＝30ns，由于忽略了替换开销，故T读缺失＝600ns。

（1）Cache采用全写法写策略时，T写命中＝max(30ns, 150ns)＝150ns，T写缺失＝0ns，故TA＝(30×700＋150×300)/(700＋300)＋(600×50＋0×15)/(700＋300)＝66＋30＝96ns。

（2）Cache采用写回法写策略时，T写命中＝30ns，T写缺失＝600ns（忽略了替换开销），故TA＝(30×700＋30×300)/(700＋300)＋(600×50＋600×15)/(700＋300)＝30＋39＝69ns。

34. 假设主存按字节编址，主存地址为20位，逻辑地址为24位，页式虚拟存储器中，页大小为8KB，页表最多有多少行？只考虑地址变换的实现，页表项至少有多少位？若TLB采用2路组相联映射，共8个条目，TLB条目至少有多少位？

答：页式虚存中，逻辑地址由虚页号、页内地址组成，物理地址由实页号、页内地址组成，页内地址为log2(8KB/1B)＝13位，虚页号为24－13＝11位，实页号为20－13＝7位。

由于页表按虚页号索引，故页表最多有211＝2048行。

只考虑地址变换的实现时，页表项由装入位、实页号等组成，至少需要1＋7＝8位。

地址变换时用虚页号访问TLB，TLB条目由有效位、标记、实页号等组成，实页号类似于Cache行的主存块。采用2路组相联映射时，TLB组号为log2(8/2)＝2位，组内行号为1位；虚页号由群号、群内条目号组成，群内条目号为2位（与TLB组号相同），则群号为11－2＝9位，即TLB条目的标记为9位。因此，TLB条目至少有1＋9＋7＝17位。

35. 基于图3.59所示的存储系统状态，若读请求的地址为0294H，写出存储系统处理访存请求的过程。

答：存储系统中，虚拟地址VA为14位，虚页号VPN为8位，页内地址VPO为6位；物理地址PA为12位，实页号PPN为6位，页内地址PPO为6位；Cache行号CI为4位，块内地址CO为2位，标记CT为6位；TLB组号TLBI为2位，标记TLBT为6位。

依题意，VA＝0294H＝00 0010 1001 0100B，存储系统的处理过程如下：

②MMU从VA中抽出VPN（0000 1010B）、VPO（010100B），用VPN查TLB。

③TLB从VPN中抽出TLBT（000010B）、TLBI（10B），用TLBT与第TLBI组的所有条目进行比较，由于第2组各个条目的有效位都为0，故结果为TLB缺失。

④MMU再用VPN形成PTEA（表基址＋VPN×页表项长度），用PTEA访问页表。

⑤MMU获得的页表项PTE为V＝1、PPN＝33H，故结果为页命中，装入的实页号PPN＝110011B，读请求的物理地址PA＝110011 010100B，其中010100B为PPO。

⑥CPU用PA（110011 010100B）访问L1 Cache。

⑦Cache从PA中抽出CT（110011B）、CI（0101B）及CO（00B），用CT与第CI行进行比较，由于第5行的标记不是110011B，故结果为Cache缺失；Cache形成目标主存块首地址（PA低2位为00），调入主存块到第CI行中，将该行的有效位置为1、标记置为110011B；Cache再读出当前行缓存块的第CO个字节，将它返回给CPU，访存操作完成。

**第4章 指令系统**

3. 指令字长、机器字长、存储字长三者间有何关系？

答：机器字长＝n×存储字长，整数n≥1且为常数。解析：由于机器字长为CPU一次能处理数据的最大位数，故机器字长为常数；由于数据可存放在MEM中，故机器字长为存储字长的倍数；由于数据可表示为多种长度，为了节省其所占存储空间，存储字长通常为最短数据的长度，即n≥1。

指令字长＝m×存储字长，整数m≥1，指令字长与机器字长没有关系。解析：由于指令存放在MEM中，故指令字长为存储字长的倍数；由于指令系统有定长指令字、变长指令字2种结构，故m不一定为常数。

4. 若某指令系统中有10种操作，使用频率分别为0.25、0.20、0.15、0.10、0.08、0.08、0.05、0.04、0.03、0.02，请分别用定长编码、哈夫曼编码、扩展编码（两种长度）进行操作码编码，并计算三种编码方式的平均码长。

答：定长编码为⎡log210⎤＝4位，哈夫曼编码根据哈夫曼树求得，编码如下表中部所示：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 操作Ii | 频率Pi | 定长编码 | 霍夫曼编码 | 霍夫曼扩展编码 |
| I1 | 0.25 | 0000 | 00 | 000 |
| I2 | 0.20 | 0001 | 10 | 001 |
| I3 | 0.15 | 0010 | 010 | 010 |
| I4 | 0.10 | 0011 | 110 | 011 |
| I5 | 0.08 | 0100 | 0110 | 100 |
| I6 | 0.08 | 0101 | 1110 | 101 |
| I7 | 0.05 | 0110 | 01110 | 110 |
| I8 | 0.04 | 0111 | 01111 | 11100 |
| I9 | 0.03 | 1000 | 11110 | 11101 |
| I10 | 0.02 | 1001 | 11111 | 11110 |
| 平均码长(∑IiPi) | | 4 | 2.99 | 3.18 |
| 码长种类 | | 1 | 4 | 2 |

扩展编码根据哈夫曼编码归类后重新编码求得，扩展编码有3位＋5位、2位＋5位两种方案。3位＋5位方案的平均码长为(0.25＋0.20＋0.15＋0.10＋0.08＋0.08＋0.05)×3＋(0.04＋0.03＋0.02)×5＝2.99位，2位＋5位方案的平均码长为(0.25＋0.20＋0.15)×2＋(0.10＋0.08＋0.08＋0.05＋0.04＋0.03＋0.02)×5＝3.2位，故3位＋5位方案较优。3位＋5位方案的编码如上表右部所示。

5. 某指令系统中，指令字长为16位，指令格式有单地址、双地址两种，操作码采用扩展编码方式，单地址指令的地址码为A（6位），双地址指令的地址码为A1（3位）及A2（6位）。回答下列问题：

（1）若双地址指令已经定义了40条，则单地址指令最多有多少条？

（2）若单地址指令已经定义了100条，则双地址指令最多有多少条？

答：依题意，双地址指令的操作码为16b－(3b＋6b)＝7b，单地址指令的操作码为16b－6b＝10b，单地址指令的操作码可划分成7b和3b，7b部分为扩展码。

（1）双地址指令的操作码中，空闲编码数≤27－40＝88种，故单地址指令最多有88×23＝704条。

（2）单地址指令的前7位操作码中，使用编码数≥⎡100/23⎤＝13种，故双地址指令最多有27－13＝115条。

8. 某机器字长为16位，存储单元长度为16位，有符号数用补码表示。指令系统中，有一个基址寄存器B及一个变址寄存器I，指令格式如下图所示，其中，F=00、01、10分别表示立即寻址、基址寻址、相对寻址方式，A用补码表示。

6bit 2bit 8bit

|  |  |  |
| --- | --- | --- |
| 操作码OP | 寻址方式位F | 形式地址A |

若(B)=8000H、(I)=0029H、(PC)=1234H，请回答下列问题：

（1）分别计算指令字8888H、5555H、6699H的操作数值或操作数地址。

（2）若分支指令的转移目标地址为1200H，则该指令采用相对寻址方式时，指令字第二个字节的内容是多少？

解：（1）指令字8888H＝100010 00 10001000B，F＝00表示立即寻址，[OPD]补＝88H；

指令字5555H＝010101 01 01010101B，F＝01表示基址寻址，A＝55H，A扩展为16位的机器数为0055H，OPD的EA＝(B)＋A＝8000H＋0055H＝8055H；

指令字6699H＝011001 10 10011001B，F＝10表示相对寻址，A＝99H，A扩展为16位的机器数为FF99H，指令字存放在16b/16b＝1个存储单元中，转移目标的EA＝(PC)＋A＝(1234H＋1)＋FF99H＝11CEH。

（2）转移目标地址＝(PC)＋A，即1200H＝(1234H＋1)＋A，A＝-53，[A]补＝11001011B，即指令字第二个字节的内容是11001011B＝CBH。

9. 某机器字长为16位，存储器按字节编址，指令格式如下，请回答下列问题：

5bit 1bit 2bit 8bit

|  |  |  |  |
| --- | --- | --- | --- |
| 操作码OP | 寻址方式位F | 通用寄存器号R | 地址参数A |

（1）该指令格式最多可定义多少条指令？指令系统中有几个通用寄存器？

（2）若该指令格式用作单地址指令，指令中没有空闲位，也无其他隐含约定，则操作数可以有哪几种寻址方式？存储器地址空间是多大？

（3）若该指令格式用作双地址指令，指令中没有空闲位，也无其他隐含约定，则两个操作数可以有哪几种寻址方式组合？

答：（1）该指令格式最多可定义25＝32条指令，因为指令条数指指令系统支持的操作类型个数，指令条数≤操作码个数；指令系统中有22＝4个通用寄存器。

（2）操作数可有基址寻址、变址寻址2种寻址方式；由于基址/变址寄存器的长度＝机器字长＝16位，故操作数的EA为16位，MEM地址空间＝EA的寻址空间＝16位。

（3）两个操作数的寻址方式组合可有寄存器＋立即、寄存器＋直接、寄存器＋间接、寄存器间接＋立即、寄存器间接＋直接、寄存器间接＋间接供6种。其中，直接寻址、间接寻址的寻址范围为28。

11. 基于图4.16的指令系统，写出实现下列C代码的机器指令序列。其中，i、s、A[0]的地址自行给定。

int i, A[100], s=0;

for ( i=0; i< 100; i++ ) s += A[i];

答：该指令系统中，机器字长为8位，MEM按字节编址，故int型数据位8位。

假设i、A[0]、s地址分别为0、1、101；语句“for ( i=0; i< 100; i++ ) s += A[i];”等价于“int \*p=&A[0]，j=100; s=0; do {s += \*p; p++; j--;} while(j≠0);”，对应的机器指令、指令功能的序列如下（从102单元开始）：

66H： 00000000 00000001 ;R0←1 即p=&A[0]

68H： 00000100 01100100 ;R1←100 即j=100

6AH： 00001000 00000000 ;R2←0 即s=0

6CH： 01011000 ;R2←(R2)＋M[(R0)] 即s←s＋\*p

//或R3←M[(R0)]，R2←(R2)＋(R3)

6DH： 10000000 ;R0←(R0)＋1 即p++

6EH： 10010100 ;R1←(R1)－1 即j--

6FH： 11010000 01101100 ;JNZ 6CH 即j≠0时转s←s＋\*p

//或JNZ -4

71H： 00111000 ;M[(R0)]←(R2) 即M[p]=s

**第5章 中央处理器**

3. 指令执行过程包含哪些步骤？CPU中有哪些基本操作？

答：指令执行过程包括取指令、指令译码、取操作数、数据操作、保存结果、计算指令地址6个步骤。其中，取指令、指令译码步骤对所有指令是通用的；计算指令地址步骤通常与其他5个步骤重叠，以缩短指令周期，因为计算指令地址与数据操作无关。

CPU中的基本操作有寄存器间数据传送、存储器读、存储器写、算逻运算4种，它们的功能分别是：RD←(RS)、MDR←M[(MAR)]、M[(MAR)]←(MDR)、RD←(RS1) op (RS2)，其中RS、RD、MAR、MDR都是CPU中的寄存器。

5. 续例5.1，写出下列指令执行阶段的μOP序列及μOPCmd序列。

（1）ADD（寄存器间接寻址） （2）JNZ（双字长格式）

答：（1）指令R1←(R1)＋M[(R0)]执行阶段的μOP序列及μOPCmd序列为：

t4：MAR←(R0) t4：GRout、Rsel、MARin

t5：MDR←M[(MAR)] t5：Read、WMFC

t6：Y←(R1) t6：GRout、Yin

t7：Z←(Y)＋(MDR) t7：MDRout、op＝00、Zin

t8：R1←(Z)，End←1 t8：Zout、GRin，End

（2）ZF＝0时，指令JNZ Addr执行阶段的μOP序列及μOPCmd序列为：

t4：MAR←(PC)  t4：PCout、MARin

t5：MDR←M[(MAR)]，PC←(PC)＋1 t5：Read、WMFC，PC+1

t6：Y←(PC)  t6：PCout、Yin

t7：Z←(Y)＋(MDR) t7：MDRout、op＝00、Zin

t8：PC←(Z)，End←1 t8：Zout、PCin，End

ZF＝1时，指令JNZ Addr执行阶段的μOP序列及μOPCmd序列为：

t4：PC←(PC)＋1，End←1 t4：PC+1，End

6. 续例5.2，写出下列指令执行阶段的μOP序列及μOPCmd序列。

（1）LD （2）ADD（寄存器间接寻址）

（3）INC （4）JNZ（单字长格式）

答：（1）指令R1←M[(R0)]执行阶段的μOP序列及μOPCmd序列为：

t4：MAR←(R0) t4：MARsel＝0、MARin

t5：MDR←M[(MAR)] t5：Read、WMFC

t6：R1←(MDR)，End←1 t6：GRsel＝1、GRin，End

（2）指令R1←(R1)＋M[(R0)]执行阶段的μOP序列及μOPCmd序列为：

t4：MAR←(R0) t4：MARsel＝0、MARin

t5：MDR←M[(MAR)] t5：Read、WMFC

t6：R1←(R1)＋(MDR)，End←1 t6：ALUAsel＝1、ALUBsel＝0、op＝00、

GRsel＝0、GRin，End

（3）指令R1←(R1)＋1执行阶段的μOP序列及μOPCmd序列为：

t4：R1←(R1)＋1，End←1 t4：ALUAsel＝1、ALUBsel＝3、op＝00、

GRsel＝0、GRin，End

（4）ZF＝0时，指令JNZ Disp执行阶段的μOP序列及μOPCmd序列为：

t4：PC←(PC)＋(ExtU)，End←1 t4：ALUAsel＝0、ALUBsel＝2、op＝00、

GRsel＝0、PCin，End

ZF＝1时，指令JNZ Addr执行阶段的μOP序列及μOPCmd序列为：

t4：End←1 t4：End

10. 基于图5.12的ALU，画出支持slt/sltu指令功能的ALU电路，可以在ALUctr中增加一根信号线，用新的编码表示slt/sltu指令的操作类型。

答：slt/sltu指令功能为rd←((rs)<(rt))? 1 : 0，比较操作可用ALU中的加减法器实现，1/0选择需增设MUX来实现。另外，ALUctr还需要增加一根信号线，相应的ALU电路如下：



假设ALU中加减法器的最高位进位信号、溢出信号、结果最高位信号分别为cout、overflow、F[31]，则有符号数A＜B的条件为overflow⊕F[31]＝1；无符号数A＜B的条件为cout⊕Subctr＝1。假设ALUctr[2..0]＝100、101表示有符号比较、无符号比较，则

Sigctr＝(ALUctr[2]·ALUctr[1])(ALUctr[0]·(overflow⊕F[31])＋ALUctr[0]·(cout⊕Subctr))

11. 图5.20的数据通路中，若DMEM的dout引脚外部设置有MDR寄存器，请写出MIPS的lw指令执行阶段的μOPCmd序列。

答：设置MDR后，读DMEM时MemRd就无需多保持一个节拍，lw指令执行阶段的μOPCmd序列如下：

t3：Extctr、ALUAsrc＝0、ALUBsrc＝1、ALUctr＝0、ALUOWr

t4：MemRd、WMFC

t5：RegAsrc＝0、RegDsrc＝0、RegWr，End ；读操作无需保持

12. 若改进图5.20的数据通路，使之能支持bne指令，如何改进？

答：数据通路支持beq、bne指令时，需增设控制信号Isbeq，以区分当前指令类型，假设当前指令为beq时Isbeq＝1，为其他指令时Isbeq＝0。分支指令在执行阶段重写PC的逻辑应该为PCWrB·(Isbeq·ZF＋Isbeq·ZF)＝PCWrB·Isbeq⊕ZF。

改进图5.20的数据通路时，只需将逻辑PCWrB·ZF换成上述逻辑即可。

17. 续例5.1，组织支持这5条指令执行的时序系统，画出时序系统的各种时序信号序列（格式同图5.25）。假设μOP采用同步控制方式，设计时序信号形成电路。

答：为了便于实现，时序信号用来表示时间次序，共需要6个节拍信号，时序信号序列共有如下2种：



由图可见，6个节拍的下一状态产生函数如下：

T1＝T0，T2＝T1，T3＝T2，T4＝⋅T3，T5＝T4，

T0＝⋅T5＋(JNZ⋅ZF)⋅T3＋⋅⋅⋅⋅。

时序信号形成电路中，同步定时方式的定时逻辑为CP＝CLK，环形信号发生器保存当前状态、实现下一状态产生函数，总体电路如下：



19. 续例5.1，基于题17所设计的时序系统，设计支持这5条指令执行的μOP控制信号形成电路。

答：整合5条指令执行的μOPCmd序列，可形成状态转换图（≤19种状态），将每个状态打上时间戳，将每个状态中的每个μOPCmd（共19种）填入下列μOPCmd使用时间表。表中All表示对所有指令通用，JNZ0、JNZ1分别表示ZF＝0时、ZF＝1时的JNZ指令。

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | T0 | T1 | T2 | T3 | T4 | T5 |
| PCout | All |  |  | JNZ0|MOV |  |  |
| PCin |  |  |  |  |  | JNZ0 |
| PC+1 |  | All |  |  | MOV |  |
| IRin |  |  | All |  |  |  |
| GRout |  |  |  | LD|ST|SUB | ST|SUB |  |
| Rsel |  |  |  | LD|ST | SUB |  |
| GRin |  |  |  |  |  | LD|SUB|MOV |
| ExtUout |  |  |  |  | JNZ0 |  |
| Yin |  |  |  | SUB|JNZ0 |  |  |
| op |  |  |  |  | SUB(01)|JNZ0(00) |  |
| Zout |  |  |  |  |  | SUB|JNZ0 |
| Zin |  |  |  |  | SUB|JNZ0 |  |
| MARin | All |  |  | LD|ST|MOV |  |  |
| MDRout |  |  | All |  |  | LD|MOV |
| MDRin |  |  |  |  | ST |  |
| Read |  | All |  |  | LD|MOV |  |
| Write |  |  |  |  |  | ST |
| WMFC |  | All |  |  | LD|MOV | ST |
| End |  |  |  | JNZ1 |  | LD|ST|SUB|JNZ0|MOV |

将上表按行汇总、逻辑化简，可得到各个μOPCmd的逻辑表达式。如：

PCout＝T0＋T3·(JNZ·＋MOV)，PCin＝T5·(JNZ·)，…，

End＝T3·(JNZ·ZF )＋T5·JNZ·ZF。

使用组合逻辑电路实现上述逻辑表达式，即可实现μOP控制信号形成电路（略）。

23. 某微程序控制器中，CS容量为512×40位，微指令采用水平型格式，顺序控制字段采用断定法（下址法＋测试网络法）形成下条微指令地址，所有指令执行过程的状态转换图中有4种分支点。请回答下列问题：

（1）请设计该微程序控制器的微指令格式。

（2）该微程序控制器中至少有多少个微命令？说明理由。

答：CS的地址为⎡log2512⎤＝9位。

（1）微指令格式的顺序控制字段中，方式位为⎡log2(1＋4)⎤＝3位，下址为9位，通常测试网络中测试条件的位数＜下址位数，故顺序控制字段为3＋9共12位。

微指令格式的操作控制字段由*m*个子字段组成，共40－12＝28位。微指令格式及顺序控制字段的格式如下图所示。



（2）操作控制字段有直接编码、字段直接编码、字段间接编码3种组织方式，直接编码方式支持的微命令最少（≤28个），故该微程序控制器中至少有28个微命令。

25. 假设MIPS CPU的指令执行过程如图5.21所示，异常或中断事件有溢出、非法操作码、缺页、可屏蔽中断四种类型，请回答下列问题：

（1）每种事件分别在哪些指令执行过程的哪些状态能够被检测到？

（2）每种事件被检测到后，CPU分别在什么时候响应事件？响应过程各需要完成哪些工作？

（3）哪一种事件的事件类型识别需要多个时钟周期才能完成？说明理由。

答：（1）溢出事件可在ex\_r状态被检测到，非法操作码事件可在id状态被检测到，缺页事件可在if或mem\_r或mem\_w状态被检测到，可屏蔽中断可在任一状态被检测到。

（2）异常事件（前3个）在被检测到的下个时钟周期响应，中断事件（最后1个）在指令执行结束时（最后一个状态的下个时钟周期）响应，为了简化实现，常在指令执行过程的最后一个状态（用μOPCmd End表示）检测中断事件。

事件响应过程中，需要完成的任务有①保存断点及程序状态，保存异常类型；②关中断；③识别事件类型并转入处理程序。对识别事件类型而言，异常事件只需读取专用寄存器即可，而中断事件需要通过专门的中断响应操作访问外设来实现；对转入处理程序而言，有向量方式和非向量方式两种，异常和中断事件的操作是相同的。

（3）可屏蔽中断事件。因为异常事件同时只会发生一个，故异常事件类型存放在CPU的专用寄存器中，识别时用寄存器间传送μOP即可实现，故需1个时钟周期；而中断事件同时会有多个发生，故中断事件类型存放在各个外设（I/O接口）中，识别时需进行CPU外部访问来实现，故需多个时钟周期。

27. 若指令执行过程分为取指、译码、取数、执行、写结果5个阶段，各个阶段的操作时延分别为10ns、5ns、10ns、8ns、7ns。将数据通路组织成流水线时，段间寄存器的时延为2ns。请回答下列问题：

（1）若采用串行方式执行10000条指令，共需要多少时间？

（2）流水线的拍长应为多少？

（3）若采用流水方式执行10000指令，流水线的吞吐率、加速比、效率各是多少？

答：（1）每条指令的指令周期为10ns＋5ns＋10ns＋8ns＋7ns＝40ns，执行10000条指令所需时间*T*串行＝10000×40ns＝0.4ms。

（2）流水线的拍长Δ*t*＝max{10ns, 5ns, 10ns, 8ns, 7ns}＋2ns＝12ns。

（3）*T*流水＝5Δ*t+*(10000－1)Δ*t*＝10004×12ns，

流水线的吞吐率*Tp*＝10000/*T*流水＝10000/(10004×12ns)＝83.3 MIPS，

流水线的加速比*S*＝*T*串行/*T*流水＝0.4ms/(10004×12ns)＝3.33，

流水线的效率*E*＝[(10000×5×12ns)/5]/(10004×12ns)≈1。

29. 某MIPS五段指令流水线中，GPRs写操作安排在前半拍进行，有如下指令序列：

addu $3, $4, $5

addu $6, $3, $3

lw $7, 0($6)

add $8, $7, $6

请回答下列问题：

（1）哪些指令之间存在数据冒险？

（2）若数据冒险仅采用阻塞法处理，指令序列执行时间为多少拍？

（3）若流水线存在EX段→EX段的转发线路，指令序列执行时间又为多少拍？

答：（1）I1-I2、I2-I3、I2-I4、I3-I4存在数据冒险，其中I3-I4还属于load-use冒险。

（2）由于GPRs写操作安排在前半拍进行，故冒险处理时，I2因I1-I2冒险需停顿2拍；I3因I2-I3冒险需停顿2拍；I4因I3-I4冒险需停顿2拍，导致I2-I4冒险自动消除。指令序列执行时间为5Δ*t*＋(4－1)Δ*t*＋(2Δ*t*＋2Δ*t*＋2Δ*t*)＝14Δ*t*。

（3）由于存在EX段→EX段的转发线路，相邻指令间的RAW冒险（不含load-use冒险）处理无需停顿，其余冒险采用阻塞法处理，故冒险处理时，I1-I2、I2-I3冒险无需停顿，I4因I3-I4冒险需停顿2拍，导致I2-I4冒险自动消除。指令序列执行时间为5Δ*t*＋(4－1)Δ*t*＋2Δ*t*＝10Δ*t*。

**第6章 总线**

3. 某同步总线的时钟频率为33.3MHz，每个时钟周期可以传送一次数据，若总线带宽为532.8Mbps，寻址空间为4G，该总线的地址总线和数据总线宽度各为多少？

答：由于总线寻址空间为4G，故地址总线宽度为log2(4G)＝32位。

设数据总线宽度为W位，则有532.8Mbps＝W×33.3MHZ/1，即W＝16位。

4. 一个总线周期的操作包含哪些阶段？每个阶段的任务是什么？哪些阶段只有主设备参与？

答：包括申请及分配阶段、寻址阶段、传送数据阶段、结束阶段。

申请及分配阶段的任务是确定哪个请求使用总线的主设备，获得下个总线传输周期的总线使用权；寻址阶段的任务是由获得总线使用权的主设备发出本次操作的从设备地址及操作命令，所有从设备主动判别自身是否为目标设备，被选中时响应总线操作；传送数据阶段的任务是根据操作命令，双方进行数据交换，数据由源设备发出、目的设备接收；结束阶段的任务是参与传送的主设备、从设备都从总线上撤销自己所发出的信号，让出总线使用权。

申请与分配阶段只有主设备参与。

6. 链式查询方式仲裁的原理是什么？有什么特点？

答：仲裁通过自动轮询各个主设备来实现，轮询时有总线请求、被询问的主设备获得总线使用权，即BSi＝BRi·BGiIN，BGiOUT＝·BGiIN。

特点是所需仲裁线最少（2根），但不能保证仲裁的公平性（固定优先级总裁策略所致），容易产生断链现象。

9. 某16位地址/数据复用的同步总线中，总线时钟频率为8MHz，每个总线事务只传输1个数据、需要4个时钟周期。该总线的可寻址空间、数据传输率各为多少？

答：该总线的可寻址空间为216＝64K，数据传输率为16b×8MHz/4＝32Mbps。

10. 某32位同步总线中，总线时钟信号CLK的频率为50MHz，总线事务支持突发传输方式，每个时钟周期可以传送一个地址或数据。存储器读总线事务的时序为地址期（1 CLK）、等待期（3 CLK）、8个数据期（8 CLK），存储器写总线事务的时序为地址期（1 CLK）、等待期（2 CLK）、8个数据期（8 CLK）、恢复期（2 CLK）。通过总线读存储器、写存储器的数据传输率分别是多少？

答：存储器读总线事务需要1＋3＋8＝12个CLK、可读出8个数据，通过总线读存储器的数据传输率为(32b×8)/(12×1/50MHz)＝133.33MBps。

存储器写总线事务需要1＋2＋8＋2＝13个CLK、可写入8个数据，通过总线写存储器的数据传输率为(32b×8)/(13×1/50MHz)＝123.08MBps。

15. 某32位半同步总线中，总线时钟信号CLK的频率为50MHz，总线事务支持突发传输模式，每个时钟周期可以传送一个地址或数据，总线传输协议的时序为地址期（1 CLK）、等待期（*k* CLK）、数据期（*n* CLK）、恢复期（*k* CLK），读事务没有恢复期，写事务没有等待期，*k*及*n*都可变，*k*取决于从设备延迟，*n*由主设备确定。连接到该总线上的存储器为4体交叉存储器（采用交叉访问工作方式），每个存储体的存取周期为80ns。请回答下列问题：

（1）通过总线从存储器读4B数据、向存储器写4B数据各需要多少时间？

（2）通过总线从存储器读16B数据、向存储器写16B数据各需要多少时间？

答：依题意，总线时钟周期*T*C为1/50MHz＝20ns；存储体的数据宽度为32b。

（1）读/写4B数据都需访问4B/32b＝1个存储体，访存延迟为80ns，总线事务的*k*＝⎡80ns/*T*C⎤＝4，*n*＝1，读4B数据所需时间为(1＋4＋1)×*T*C＝120ns，写4B数据所需时间为(1＋1＋4)×*T*C＝120ns。

（2）读/写16B数据都需访问16B/32b＝4个存储体，对4体交叉存储器共进行4/4＝1次交叉访问，每个存储体启动间隔为80ns/4＝20ns，总线事务的*k*＝⎡80ns/*T*C⎤＝4，*n*＝4，数据传输间隔为⎡20ns/*T*C⎤＝1个时钟周期，读16B数据所需时间为[(1＋4＋1×4)×*T*C]×1＝180ns，写16B数据所需时间为[(1＋1×4＋4)×*T*C]×1＝180ns。

**第7章 输入/输出系统**

4. 简述外设识别自身是否为总线事务的目标从设备的方法。

答：每个外设独有唯一的设备号，保存在所连接的I/O接口中。I/O接口一直监视总线状态，当有I/O总线事务时，将所存设备号与总线上地址进行比较，从而识别出自身是否为总线事务的目标从设备。

9. 某磁盘组有6个双面盘片，最外两侧盘面为保护面（不记录信息）。盘片存储区域内径为22cm，外径为33cm，磁道间距最小为0.25cm，磁道位密度为1600bit/cm。假设磁盘转速为5400r/min，平均寻道时间为8ms。

（1）此磁盘组的存储容量、数据传输率、平均寻址时间分别是多少？

（2）若采用定长记录格式，每个扇区的数据容量为512B、地址等辅助信息为90B，写出磁盘地址的组成格式及其参数。

答：（1）盘片磁道数为(33－22)÷2÷0.25＝22个，每个磁道的存储容量＝最内圈磁道的存储容量＝22×3.14×1600＝13816B，故磁盘组的存储容量＝(6×2－2)×22×13816B＝3039520B≈2.9MB，数据传输率＝13816B×(5400÷60)＝1243440B/s≈1.24MB/s，平均寻址时间＝8ms＋(1÷(5400÷60))÷2＝13.56ms。

（2）磁盘地址由磁道号、盘面号、扇区号组成，其中，磁道号为⎡log222⎤＝5位，盘面号为⎡log210⎤＝4位，扇区号为⎡log2[13816B÷(512B＋90B)]⎤＝5位。

10. 假设连接主存及外设的半同步总线宽度为32位、时钟频率为100MHz，支持16B长度的突发传送。总线事务中，发送地址及命令需要一个时钟周期，等待从设备完成操作后，每个时钟周期都可以传送一个数据。若主存支持成组传送，首个32位数据的存取周期为210ns，随后每存取一个32位数据的时间为20ns，磁盘的数据传输率为10MB/s，则最多可有多少个磁盘同时进行传输？

答：总线时钟CLK的周期为1÷100MHz＝10ns，主存存取首个32位数据需要210ns÷10ns＝21个CLK，随后每存取一个32位数据需要20ns÷10ns＝2个CLK，即每2个CLK才能传送一个32位数据，故通过总线读/写主存中16B数据需要[1＋21＋2×(16B÷4B)]×10ns＝300ns，主存的数据传输率为16B÷300ns＝53.33MB/s。

磁盘的数据传输率为10MB/s时，最多可支持⎣53.33÷10⎦＝5个磁盘同时进行传输。

15. 简述可屏蔽中断请求得到响应的条件。

答：可屏蔽中断请求得到响应的条件为：处于开中断状态（IF＝1）、当前指令结束时（End＝1）、无更紧急的中断请求（如NMI＝0）。

16. 简述向量中断方式下，中断响应需要完成的任务。

答：中断响应需要完成的任务有①保存断点及程序状态；②关中断；③识别事件类型并转入处理程序。向量中断方式下，识别事件类型并转入处理程序包含的子任务有：识别中断源、获得处理程序入口地址、将处理程序入口地址写入PC。识别中断源需通过CPU外部的中断响应操作来实现，获得处理程序入口地址需通过查询放在主存中的IVT实现。

17. 某计算机的主频为200MHz、CPI为5，设备A的数据传输率为32Kbps、每次I/O可传送32位数据。程序查询方式中，一次I/O的查询平均需要49个指令周期；程序中断方式中，中断响应需要15个时钟周期，中断服务程序有10条指令。

（1）程序查询方式中，CPU用于I/O的时间占CPU总时间的百分比是多少？

（2）程序中断方式中，CPU用于I/O的时间占CPU总时间的百分比又是多少？

答：依题意，CPU主时钟周期为1÷200MHz＝5ns，设备A每次传送32位数据，每秒可传送32Kbps÷32b＝1000次，每次传送需要1÷1000＝1ms。

（1）轮询方式中，一次I/O所需的CPU时间为(49＋1)×5×5ns＝1.25μs，CPU用于I/O的时间占CPU总时间的百分比1.25μs÷1ms＝0.125%。

（2）中断方式中，一次I/O所需的CPU时间为(15＋10×5)×5ns＝0.325μs，CPU用于I/O的时间占CPU总时间的百分比0.325μs÷1ms＝0.0325%。