

1、

真 值		原 码	反 码	补 码
十进制	二进制			
-13/64	-0.00 1101	1.001 1010	1.110 0101	1.110 0110
29/128	0.001 1101	0.001 1101	0.001 1101	0.001 1101
100	110 0100	0,110 0100	0,110 0100	0,110 0100
-87	-101 0111	1,101 0111	1,010 1000	1,010 1001

2、

1) $[x]_{\text{原}}=1.0100$, $x=-0.0100$

5) $[x]_{\text{原}}=1,1011$, $x=-1011$

3、

十六进制	真值	无符号数	原码	反码	补码	移码
9BH	十进制	155	-27	-100	-101	+27
FFH	十进制	255	-127	0	-1	+127

1) 9BH、FFH 为机器数，本身含符号位。

2) 移码符号位与原、补、反码相反，数值位同补码。

$$4、x_2 = -27/1024 = (-0.000\ 001\ 101\ 1)_2 \\ = 2^{-5} \times (-0.110\ 11)_2$$

则以上各数的浮点规格化数为：

(1) $[x_2]_{\text{浮}}=1, 0101; 1.110\ 110\ 000\ 0$

(2) $[x_2]_{\text{浮}}=1, 1011; 1.001\ 010\ 000\ 0$

(3) $[x_2]_{\text{浮}}=0, 1011; 1.001\ 010\ 000\ 0$

$$5、F = \overline{C}D + \overline{B}\overline{C} + \overline{A}BD + \overline{A}\overline{C}\overline{D} + B\overline{C}\overline{D};$$

$$F(ABCD) = A + \overline{\overline{B}\overline{C}\overline{D}} + B\overline{C}\overline{D}$$

6、原码 1 位： $x^*y^*=0.100111100010$

加符号位， $[x \times y]_{\text{原}} = 1.100 \ 111 \ 100 \ 010$
 补码一位： $[x]_{\text{补}} = x = 0.110111$ ， $[y]_{\text{补}} = 1.010010$ ， $[-x]_{\text{补}} = 1.001001$
 $[x \times y]_{\text{补}} = 1.011 \ 000 \ 011 \ 110 \ 0$ （计算过程略。）

7、解： $[x]_{\text{补}} = 1, 101; 0.101100$ ， $[y]_{\text{补}} = 1, 110; 1.100100$

1) 对阶：

$$[DE]_{\text{补}} = [Ex]_{\text{补}} + [-Ey]_{\text{补}} = 11, 101 + 00, 010 = 11, 111$$

$[DE]_{\text{补}} < 0$ ，应 Ex 向 Ey 对齐，则：

$$[Ex]_{\text{补}} + 1 = 11, 101 + 00, 001 = 11, 110$$

$$[x]_{\text{补}} = 1, 110; 0.010 \ 110$$

2) 尾数运算：

$$[Sx]_{\text{补}} + [Sy]_{\text{补}} = 00.010110 + 11.100100 = 11.111010$$

3) 结果规格化：

$$[x+y]_{\text{补}} = 11, 110; 11.111010 = 11, 011; 11.010000$$

（左规 3 次，阶码减 3，尾数左移 3 位）

4) 舍入：不需舍入。

5) 判阶码是否溢出：无

$$\text{则：} x+y = 2-101 \times (-0.110 \ 000)$$

8、提示：系统性能的提高可用“系统加速比”来表征。

系统加速比 = 未采用 Cache 前的总执行时间 / 采用 Cache 后的总执行时间

系统性能提高 = (系统加速比 - 1) \times 100%

解：1) Cache 的命中率： $4800 / (4800 + 200) = 0.96$;

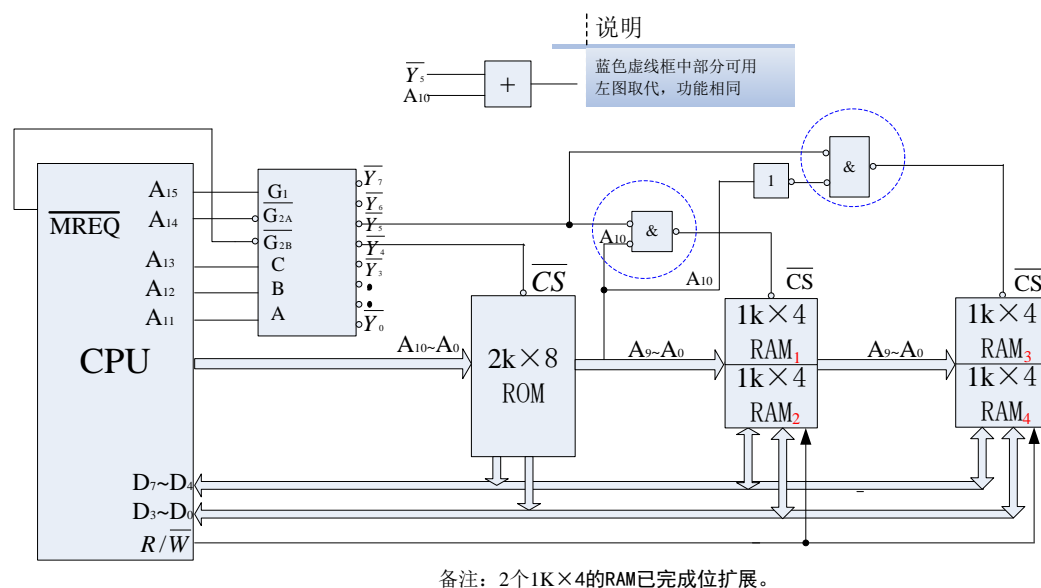
2) 未采用 Cache 前的总执行时间： $5000 \times 150 = 7.5 \times 10^5 \text{ns}$

采用 Cache 后的总执行时间： $4800 \times 30 + 200 \times 150 = 1.74 \times 10^5 \text{ns}$

系统的性能提高 = $(0.75 / 0.174 - 1) \times 100\% = 331\%$ 。

9、

RAM 芯片的地址范围：



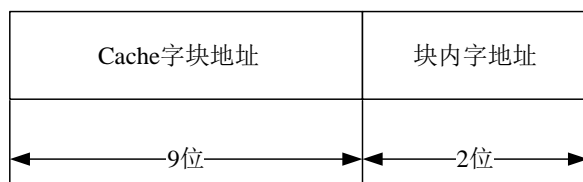
1) RAM1 和 RAM2: A800H~ABFFH 2) RAM3 和 RAM4: AC00H~AFFFH

10、需要的存储周期数: $T+(64-1)T/4=16.75T$

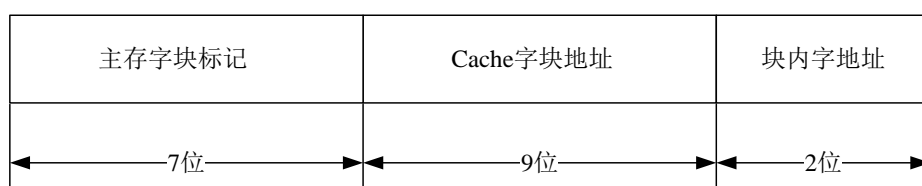
存取周期数取整, 为 17 个。

11、

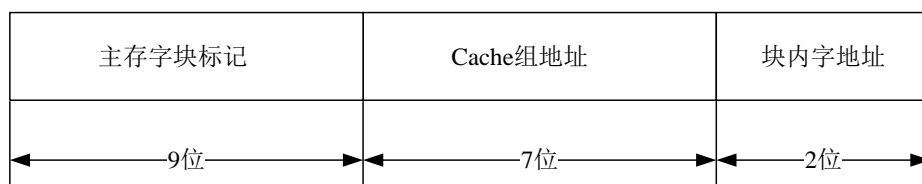
1)



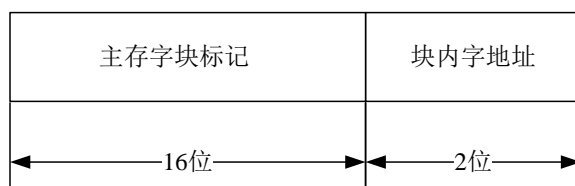
2) 主存地址格式:



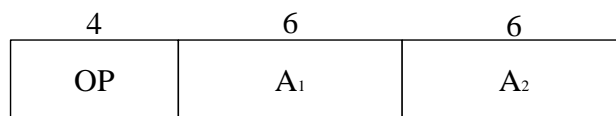
3) 主存地址格式:



4)



12、解: 二地址指令格式如下



设二地址指令格式为该指令系统的基本格式, 4 位操作码共有 16 种编码, 其中 13 种用来定义二地址指令, 还剩 3 种可用作扩展标志。如不考虑零地址指令, 该指令系统最多还能安排:

一地址指令条数 $=3 \times 2^6=192$ 条

13、1) 操作码定长: 二地址指令条数 $=16-M-N$

2) 操作码变长 (采用操作码扩展技术):

设二地址指令有 X 种，则有如下的等式： $((16-X) \times 26-N) \times 26=M$
 则 $X=16 - (M/212+ N/26)$ ，其中 $M/212+ N/26$ 取上整数。

14、解：据题意，相对寻址的转移指令格式如下：

2000H	OP
2001H	A
2002H	XX

弄明白+8 和-9 的含义：转移目标地址和指令地址之间的距离（方便编程人员编程的），而转移指令第二字节内容是相对位移量，为转移目标地址和修正后的PC 值之间的差值。即：

$$A1= (2000+8) -2002=06H$$

$$A2= (2000-9)-2002=F5H \text{（真值：-00001011；写成补码形式：11110101=F5H）}$$

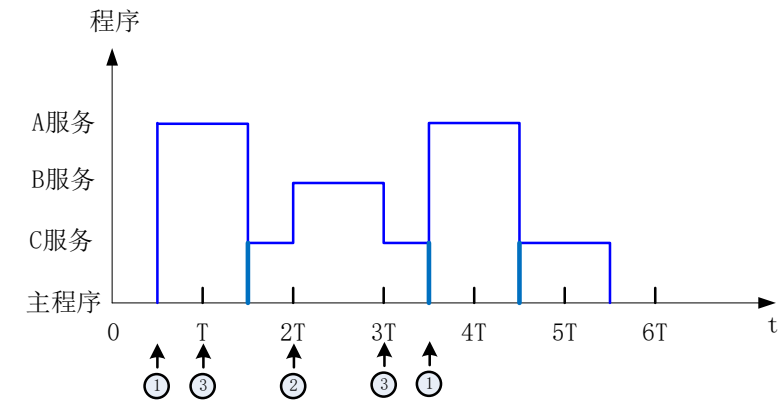
其有效地址各为：

$$EA1= (PC) +A1 = 2002H+0006H= 2008H$$

$$EA2= (PC)+A2 =2002H+FFF5H = 1FF7H \text{（做加法前，要先做符号位扩展：F5H 写成 FFF5H）}$$

15、(1) 8341H =1000 0011 0100 0001
 寻址方式：相对寻址； EA= (PC)+2+形式地址 A=5431+2+0041H=5474H
 (2) 1468H =0001 0100 0100 1000
 寻址方式：直接寻址； EA=形式地址 A=0068H
 (3) 8100H =1000 0001 0000 0000
 寻址方式：寄存器间接寻址； EA= (R1) =3525H
 (4) 6264H =0110 0010 0110 0100
 寻址方式：变址寻址； EA= (R2) +位移量 D=6783H+0064H=67E7H

16.

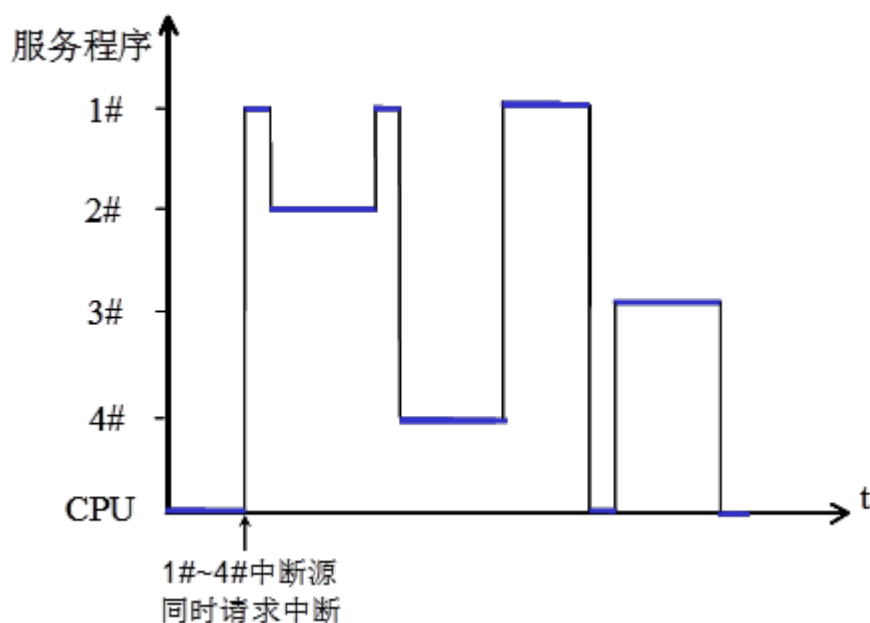


17.

中断源	屏蔽字				
	0	1	2	3	4
L ₀	1	0	0	1	0
L ₁	1	1	1	1	1
L ₂	1	0	1	1	0
L ₃	0	0	0	1	0
L ₄	1	0	1	1	1

18. 1) 响应中断请求后送出的各级中断屏蔽字：1 级的中断屏蔽字为：1010；2 级的中断屏蔽字为：1111；3 级的中断屏蔽字为：0010；4 级的中断屏蔽字为：1011；（屏蔽字的顺序为 IMR₁ IMR₂ IMR₃ IMR₄）

2) CPU 的运行轨迹：



7. 说明组合逻辑控制器与微程序控制器在组成和原理上异同处及优缺点。

解：

(1) 相同之处：1) 基本功能相同，即提供计算机各个部件协同运行所需的控制信号；2) 基本组成也相同，即 PC、IR 及几个步骤完成每一条指令的具体功能。

(2) 不同之处：1) 处理指令执行步骤方法不同：组合逻辑控制器是利用节拍发生器的节拍信号区分指令执行步骤；而微程序控制器是通过微指令地址的衔接区分指令执行步骤；2) 提供控制信号的方案不同：组合逻辑控制器采用组合逻辑电路直接产生控制信号；微程序控制器是通过微地址译码器找到控存单元，取出的微指令才是控制信号。

补充题 1：说明微程序控制器的基本工作原理。

参考答案要点:

将控制器所需要的微操作命令，以微代码的形式编成微指令，存在专门的控制存储器中，CPU执行机器指令时，从控制存储器中取出微指令，对微指令中的操作控制字段进行解释，即产生执行机器指令所需的微操作命令序列。

其具体的工作过程如下：

首先将用户程序的首地址送至PC, 然后进入取指阶段。

- 1) 取机器指令：从控制存储器中读取“取指微指令”，用产生的微命令控制CPU访存，读取机器指令，并送入指令寄存器IR。
- 2) 形成微程序入口地址：根据机器指令的操作码，通过微地址形成电路，产生与该机器指令对应的微程序入口地址，并送入CMAR。
- 3) 逐条取出机器指令对应的微程序并执行之。
- 4) 返回取指微指令，开始又一条机器指令的执行。如此不断重复，直到整个程序执行完为止。

3.微程序控制方式优缺点及应用

1. 优点:

- ①结构规整，设计效率高；
- ②易于修改，扩展指令系统功能；
- ③可靠性高；
- ④性价比高。

2. 缺点:

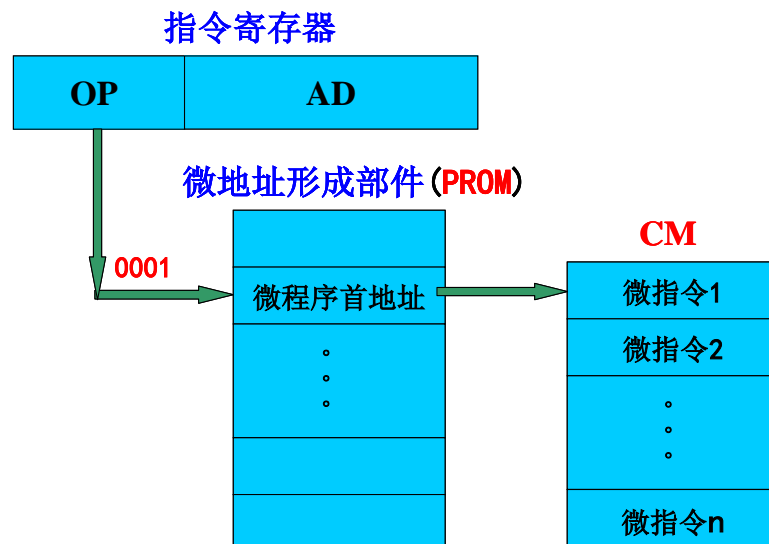
- ①执行速度较慢
- ②执行效率不高

3. 应用:

用于对速度要求不太高的、功能较复杂的机器中，特别适用于系列机。

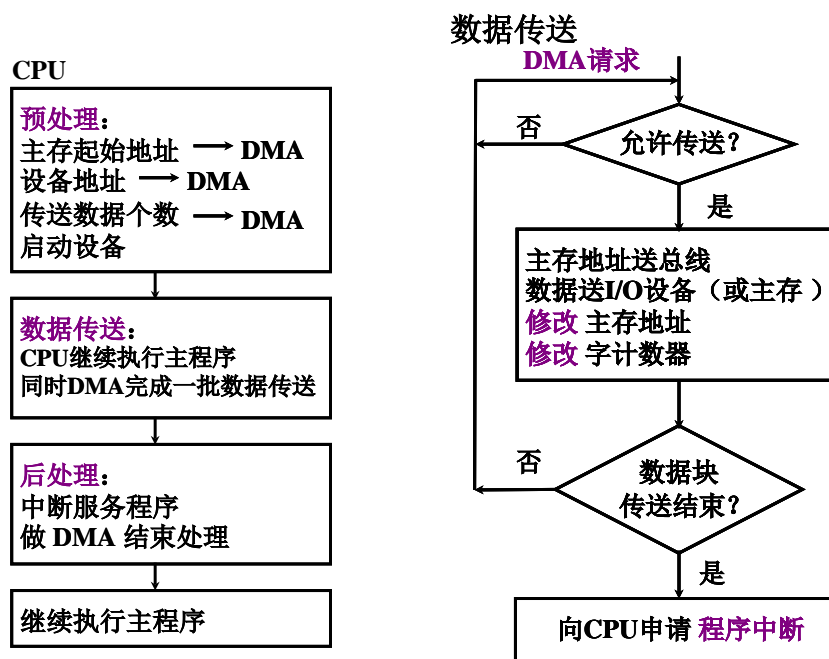
补充题 2: 画图并说明微程序控制器中是如何根据操作码形成相应微程序入口地址的。

参考答案要点:



如上图所示，机器指令取至指令寄存器后，指令的操作码作为微地址形成部件的输入，来形成微指令的地址。因此，可以把微地址形成部件理解为一个编码器。微地址形成部件可采用 PROM 实现，即以指令的操作码作为 PROM 的地址，而相应的存储单元中就存放着对应该指令微程序的首地址。

4.DMA 的数据传送过程可分为预处理、数据传送和后处理 3 个阶段。工作过程如下图所示：



各阶段完成的工作如下:

- 1) 预处理阶段: CPU 执行主程序实现 DMA 传送的初始化设置;
- 2) 数据传送阶段: 由 DMA 控制器实现内存和外设间的数据传送。
- 3) 后处理阶段: 中断处理程序判断传送的正误, 对写入主存的数据进行校验, 完成善后工作。