

考试题型及说明

一、简述题、名词解释

内容参考“期末复习 1.《计算机系统结构》重点知识总结.docx”

二、判断正误题，并改正

内容参考“期末复习 1.《计算机系统结构》重点知识总结.docx”

三、计算题

1、IEEE754

2、根据寻址方式寻找操作数

3、扩展操作码计算

4、1 位原码乘法运算

5、1 位原码不恢复余数除法运算

6、写出并行加法器中，进位链的逻辑表达式：

串行进位链结构： $C_n = G_n + P_n C_{n-1}$

并行进位链结构： $C_n = G_n + P_n G_{n-1} + \dots + P_n \dots P_1 C_0$

7、8259 中控制器计算

(1) ISR、IRR、IMR 寄存器与 INT 信号的关系/8086 输入输出接口状态判断

(2) 向量地址计算：(IBM PC：向量地址=中断类型号 $\times 4$ ； 模型机：向量地址=中断类型号 $\times 2$)，根据向量地址查找中断向量表，获得中断服务程序的入口地址。(IT 周期完成过程)

(3) 中断屏蔽字设计

(4) 8259 中断响应过程

(5) 中断服务程序：单级中断和多重中断流程

四、设计题

1、CPU 指令流程和微命令

2、存储器的主存设计

具体知识点

一、IEEE754

① 32位短浮点数：

31	30 ~ 23	22 ~ 0
S	E	M

在上述的表示格式中：

• S=浮点数的符号位，0表示正数，1表示负数；

• E=阶码，8位，采用移码表示，阶符隐含；

• M=尾数，23位，纯小数表示，且真值=1+M；

• 阶码E采用移码形式，但只偏移 2^7-1 （不是 2^7 ）。

1、采用IEEE754 短浮点数格式，请将十进制数37.25 写成浮点数，并且写出其二进制代码序列。

解：将十进制数37.25 转换为二进制数100101.01，按IEEE754 标准的短实数浮点格式要求，将100101.01 表示为 1.0010101×2^5 ，故浮点数阶码的真值e=5。于是，按IEEE754

标准，得：

数符 $S=0$ ，阶码（移码表示） $E=(e+127)_{10}=(5+127)_{10}=(132)_{10}=(10000100)_2$ ， $M=001010100000\cdots00$ 。

最后得到32 位浮点数的二进制数代码序列为：

01000010000101010000000000000000

2、某一个标准的IEEE754 格式的短浮点数，表示为十六进制形式为2AB03700H，请将其转化成对应的十进制数，要求写出主要的转化步骤。

解：IEEE754 格式：1位数符 S ，8位阶码 E ，23位尾数 M ，而

2AB03700H=0010 1010 1011 0000 0011 0111 0000 0000

符号位 $S=0$ ；

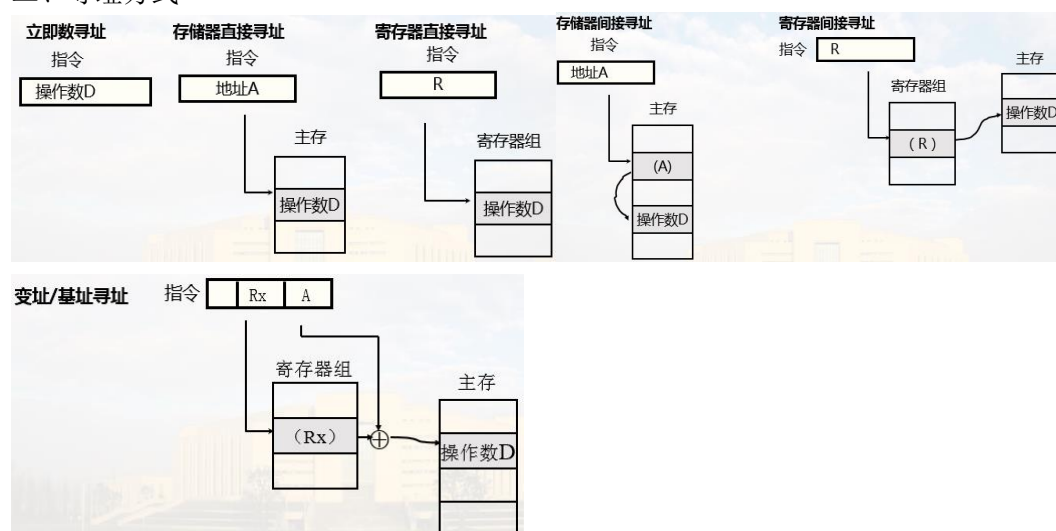
阶码 $E=010\ 1010\ 1=85$ ， $e=85-127=-42$ ；

$M=011\ 0000\ 0011\ 0111\ 0000\ 0000$

M 真值= $(1+2^{-2}+2^{-3}+2^{-10}+2^{-11}+2^{-13}+2^{-14}+2^{-15})$

所以十进制数= $(1+2^{-2}+2^{-3}+2^{-10}+2^{-11}+2^{-13}+2^{-14}+2^{-15})\times 2^{-42}$

二、寻址方式



【例】某主存储器部分单元的地址码与存储器内容对应关系如下：

地址码	存储内容
1000H	A307H
1001H	0B3FH
1002H	1200H
1003H	F03CH
1004H	D024H

(1) 若采用寄存器间址方式读取操作数，指定寄存器R0 的内容为1002H，则操作数是多少？

(2) 若采用自增型寄存器间址方式(R0)+读取操作数，R0 内容为1000H，则操作数是多少？指令执行后R0 的内容是多少？

(3) 若采用自减型寄存器间址方式 (R1)-读取操作数，R1 内容为1003H，则操作数是多少？指令执行后R1 的内容是多少？

(4) 若采用变址寻址方式X(R2)读取操作数，指令中给出形式地址 $d=3H$ ，变址寄存器R2 内容为1000H，则操作数是多少？

解：

- (1) 操作数是1200H。
- (2) 操作数是A307H，指令执行后R0 的内容变为1001H。
- (3) 操作数是1200H，指令执行后R1 的内容为1002H。
- (4) 操作数为F03CH。

三、扩展操作码

1、设某指令系统的指令字为16位, 每个地址码为6位。若二地址指令15条, 单地址指令34条, 则剩下的零地址指令最多有多少条？

解答：操作码是16-12=4bit，零地址指令条数 = $[(2^4-15) \times 2^6-34] \times 2^6$

2、某计算机按字节编址，指令字长固定且只有两种指令格式，其中三地址指令29条，二地址指令107条，每个地址字段6位，则指令字长至少应该是？（至少要多少次访存）

解答：（1）三地址指令有29条，则操作码至少为5位。

（2）如果三地址刚好是5位，则有 $32-29=3$ 条可以用于二地址指令扩展。地址码为6位，则二地址指令最多可以有 $3 \times 2^6=192$ 条 >107 条。所以指令字长至少是 $5+3 \times 6=23$ 。

（3）计算机按字节编址, $23 \bmod 8 \neq 0$ 。

（4）指令字长至少是24位。

3、某种计算机的算术逻辑运算指令，请问最多能表示多少条指令？

0	1	2	3	4	5	6	7	8	15
基本操作	进位	移位	回送	判跳	操作数				

解答： $2^3 \times 2 \times 2^2 \times 2 \times 2 = 256$

四、1位原码计算步骤

【例】 0.1101×1.1011

设置寄存器：

A：存放部分积累加和、乘积高位

B：存放被乘数

C：存放乘数、乘积低位

设置初值：A = 00.0000 B = |X| = 00.1101

C = |Y| = .1011

步数	条件	操作	A	C C _n
			00.0000	.1011
1)	C _n =1	+B	+ 00.1101	
			00.1101	
2)	C _n =1	→	00.0110	1.101
		+B	+ 00.1101	
			01.0011	
3)	C _n =0	→	00.1001	11.10
		+0	+ 00.0000	
			00.1001	
4)	C _n =1	→	00.0100	111.1
		+B	+ 00.1101	
			01.0001	
		→	00.1000	1111
X _原 × Y _原 = 1.10001111				

(1) 运算规则

- (a) 操作数、运算结果用**原码**表示；
- (b) **绝对值**参与运算，符号单独处理；
- (c) 被乘数(B)、累加和(A)取**双符号位**；
- (d) 乘数末位(C_n)为**判断位**，其状态决定下一步操作；
- (e) 作**n次**循环（累加、右移）。

五、原码不恢复余数除法

1、运算规则

- A、B取双符号位，X、Y取绝对值运算， $|X| < |Y|$
- 根据**余数的正负**决定**商值**及**下一步**操作：

$$r_{i+1} = 2r_i + (1 - 2Q_i)Y$$

r_i 为正，则 Q_i 为1，第 $i+1$ 步作 $2r_i - Y$ ；

r_i 为负，则 Q_i 为0，第 $i+1$ 步作 $2r_i + Y$ 。

- 求 n 位商，作 n 步操作；若第 n 步余数为负，则第 $n+1$ 步恢复余数，不移位。

2、实例：

$X=0.10110$, $Y=-0.11111$, 求 X/Y , 给出商 Q 和余数 R

初值: $A=|X|=00.10110$, $B=|Y|=00.11111$, $-B=11.00001$, $C=|Q|=0.00000$

步数	条件	操作	A	C	C_n
	r		00.10110 r_0	0.00000	
1)		←	01.01100 $2r_0$		
		-B	+11.00001		
	为正		00.01101 r_1	0.00001 Q_1	
2)		←	00.11010 $2r_1$		
		-B	+11.00001		
	为负		11.11011 r_2	0.00010 Q_2	
3)		←	11.10110 $2r_2$		
		+B	+00.11111		
	为正		00.10101 r_3	0.00101 Q_3	
4)		←	01.01010 $2r_3$		
		-B	+11.00001		
	为正		00.01011 r_4	0.01011 Q_4	
5)		←	00.10110 $2r_4$		
		-B	+11.00001		
	为负		11.10111 r_5'	0.10110 Q_5	
6)		+B	+00.11111		
	恢复余数		00.10110 r_5		
$Q = -0.10110$					
$R = 0.10110 \times 2^{-5}$					
$X/Y = -0.10110 + \frac{0.10110 \times 2^{-5}}{-0.11111}$					

六、写出并行加法器中，进位链的逻辑表达式：

串行进位链结构： $C_n = G_n + P_n C_{n-1}$

并行进位链结构： $C_n = G_n + P_n G_{n-1} + \dots + P_n \dots P_1 C_0$

【例】已知操作数 A_i 、 B_i ，初始进位 C_0 。试写出 C_6 的逻辑式。

串行进位： $C_6 = G_6 + P_6 C_5$

并行进位： $C_6 = G_6 + P_6 G_5 + P_6 P_5 G_4 + \dots + P_6 P_5 \dots P_1 C_0$

分级同时进位，4 位一组：

$C_6 = G_6 + P_6 G_5 + P_6 P_5 C_1$

$C_1 = G_1 + P_1 C_0$

$G_1 = G_4 + P_4 G_3 + P_4 P_3 G_2 + P_4 P_3 P_2 G_1$

$P_1 = P_4 P_3 P_2 P_1$

$G_i = A_i B_i$

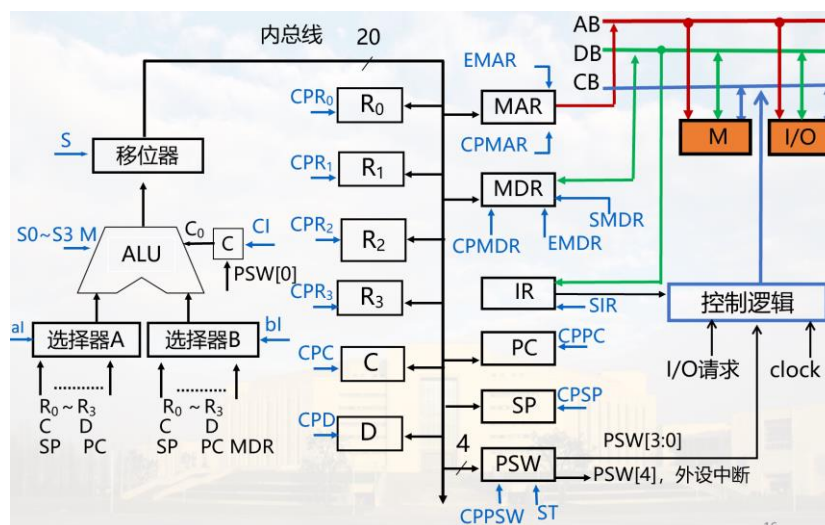
$P_i = A_i \oplus B_i$

七、指令流程和操作时间表（MOV指令和双操作数）

1、模型机寻址方式

寄存器寻址	000	R	(R) 为操作数
寄存器间址	001	(R)	(R) 为操作数地址
自减型寄存器间址	010	-(R) -(SP)	(R)-1 为操作数地址 (SP)-1 为栈顶地址
立即/自增型寄存器间址	011	(R)+ (SP)+ (PC)+	(R) 为操作数地址，访问后(R)+1 (SP) 为栈顶地址，出栈后(SP)+1 (PC) 为立即数地址，取数后(PC)+1
直接/自增型双间址	100	@(R)+ @(PC)+	(R) 为间接地址，访问后(R)+1 (PC) 为间接地址，取数后(PC)+1
变址	101	X(R) X(PC)	(R)+d 为有效地址 (PC)+d 为有效地址
跳步	110	SKP	跳过下条指令执行

2、模型机的结构（数据单向流动、寄存器级）



【例1】MOV (R0), (SP)+;

FTO: $M \rightarrow IR$ EMAR、R、SIR

$PC+1 \rightarrow PC$ $PC \rightarrow A, A+1, DM, CPPC, 1 \rightarrow ST, CPT(\bar{P}), CPFT(\bar{P}),$
 $CPST(\bar{P}), CPDT(\bar{P}), CPET(\bar{P})$

ST0: $SP \rightarrow MAR$ $SP \rightarrow A, \text{输出}A, DM, CPMAR, T+1, CPT(\bar{P})$

ST1: $M \rightarrow MDR \rightarrow C$ $EMAR, R, SMDR, MDR \rightarrow B, \text{输出}B, DM, CPC, T+1, CPT(\bar{P})$

ST2: $SP+1 \rightarrow SP$ $SP \rightarrow A, A+1, DM, CPSP, 1 \rightarrow DT, CPT(\bar{P}), CPFT(\bar{P}),$
 $CPST(\bar{P}), CPDT(\bar{P}), CPET(\bar{P});$

DT0: $RO \rightarrow MAR$ $RO \rightarrow A, \text{输出}A, DM, CPMAR, 1 \rightarrow ET, CPT(\bar{P}),$
 $CPST(\bar{P}), CPDT(\bar{P}), CPET(\bar{P});$

ET0: $C \rightarrow MDR$ $C \rightarrow A, \text{输出}A, DM, CPMDR, T+1, CPT(\bar{P})$

$MDR \rightarrow M$ $EMAR, W, T+1, CPT(\bar{P})$

$PC \rightarrow MAR$ $PC \rightarrow A, \text{输出}A, DM, CPMAR, 1 \rightarrow FT, CPT(\bar{P}), CPFT(\bar{P}),$
 $CPST(\bar{P}), CPDT(\bar{P}), CPET(\bar{P})$

【例2】ADD (R1)+, X(RO);

FT0: ...

ST0: $PC \rightarrow MAR$ $PC \rightarrow A, \text{输出}A, DM, CPMAR, T+1, CPT(\bar{P})$

ST1: $M \rightarrow MDR \rightarrow C$ $EMAR, R, SMDR, MDR \rightarrow B, \text{输出}B, DM, CPC, T+1,$
 $CPT(\bar{P})$

ST2: $C+ RO \rightarrow MAR$ $C \rightarrow A, RO \rightarrow B, A+B, DM, CPMAR, T+1, CPT(\bar{P}),$

ST3: $M \rightarrow MDR \rightarrow C$ $EMAR, R, SMDR, MDR \rightarrow B, \text{输出}B, DM, CPC, T+1,$
 $CPT(\bar{P})$

ST4: $PC+1 \rightarrow PC$ $PC \rightarrow A, A+1, DM, CPPC, 1 \rightarrow DT, CPT(\bar{P}), CPFT(\bar{P}),$
 $CPST(\bar{P}), CPDT(\bar{P}), CPET(\bar{P})$

DT0: $R1 \rightarrow MAR$ $R1 \rightarrow A, \text{输出}A, DM, CPMAR, T+1, CPT(\bar{P})$

DT1: $M \rightarrow MDR \rightarrow D$ $EMAR, R, SMDR, MDR \rightarrow B, \text{输出}B, DM, CPD, T+1,$

$\overline{CPT(P)}$

DT2: $R1+1 \rightarrow R1$ $R1 \rightarrow A, A+1, DM, CPR1, 1 \rightarrow ET, \overline{CPT(P)}, \overline{CPFT(P)},$

$\overline{CPST(P)}, \overline{CPDT(P)}, \overline{CPET(P)}$

ET0: $C+D \rightarrow MDR$ $C \rightarrow A, D \rightarrow B, A+B, DM, CPMDR, T+1, \overline{CPT(P)}$

ET1: $MDR \rightarrow M$ $EMAR, W, T+1, \overline{CPT(P)}$

ET2: $PC \rightarrow MAR$ $PC \rightarrow A, \text{输出} A, DM, CPMAR, 1 \rightarrow FT, \overline{CPT(P)}, \overline{CPFT(P)},$

$\overline{CPST(P)}, \overline{CPDT(P)}, \overline{CPET(P)}$

【例3】根据模型机结构，回答下列问题

(1) 补充下面源操作ST的指令流程对应的微命令

ST0: $PC \rightarrow MAR$ $PC \rightarrow A, \text{直通} A, DM, CPMAR, T+1, \overline{CPT(P)}$

ST1: $M \rightarrow MDR \rightarrow C$ $EMAR, R, SMDR, MDR \rightarrow B, \text{直通} B, DM, CPC,$
 $T+1, \overline{CPT(P)}$

ST2: $PC+1 \rightarrow PC$ $PC \rightarrow A, A+1, DM, CPPC, T+1, \overline{CPT(P)}$

ST3: $C+R0 \rightarrow MAR$ $R0 \rightarrow A, C \rightarrow B, A \text{ 加 } B, DM, CPMAR, T+1, \overline{CPT(P)}$

ST4: $M \rightarrow MDR \rightarrow C$ $EMAR, R, SMDR, MDR \rightarrow B, \text{直通} B, DM, CPC$
 $\overline{CPT(P)}, \overline{CPFT(P)}, \overline{CPST(P)}, \overline{CPDT(P)}, \overline{CPET(P)}$

(2) 补充目的操作DT和执行操作ET的微命令对应的指令流程

DT0: $R0 \rightarrow MAR$ $R0 \rightarrow B, \text{直通} B, DM, CPMAR$

DT1: $M \rightarrow MDR \rightarrow D$ $EMAR, R (SMDR), MDR \rightarrow B, \text{直通} B, DM, CPD$

ET0: $C-D \rightarrow MDR$ $C \rightarrow A, D \rightarrow B, A-B, DM, CPMDR$

ET1: $MDR \rightarrow M$ $W (EMDR)$

ET2: $PC \rightarrow MAR$ $PC \rightarrow A, \text{直通} A, DM, CPMAR$

(3) 写出上述指令流程对应的机器指令

$SUB (R0), X(R0)$

八、主存设计

1、存储芯片型号

(1) 存储单元数（芯片地址引脚数）

1K ($2^{10}=400H, 0-3FFH$) 芯片地址线10条, A0-A9

2K ($2^{11}=800H, 0-7FFH$) 芯片地址线11条, A0-A10

4K ($2^{12}=1000H, 0-FFFH$) 芯片地址线12条, A0-A11

8K ($2^{13}=2000H, 0-1FFFH$) 芯片地址线13条, A0-A12

16K ($2^{14}=4000\text{H}$, 0-3FFFH) 芯片地址线14条, A0-A13

32K ($2^{15}=8000\text{H}$, 0-7FFFH) 芯片地址线15条, A0-A14

(2) 存储单元位 (bit) 数

通常有4bit和8bit。若用4bit的芯片, 则两块芯片为1组 (8bit的存储单元), 例如: 某种芯片为1Kx4, 则选两块芯片构成容量为1Kx8。

2、注意事项

(1) 计算芯片数量

(2) 画图时, 必须画R/W, 芯片地址线、数据线 (若由两块4bit芯片构成1组的话, 一片数据线接总线的D0-D3, 另一片数据线连总线的D4-D7)

(3) 片选信号 (全译码)

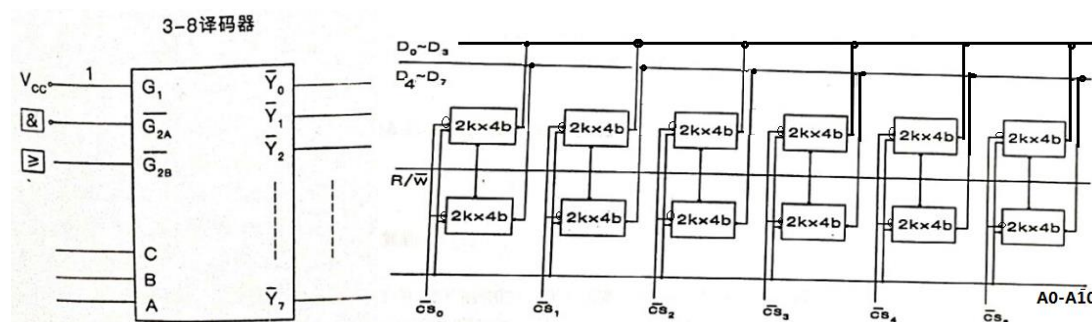
74LS138 译码器										输入								输出							
G1	G2	G3	A2	A1	A0	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0												
\bar{Y}_0	\bar{Y}_1	\bar{Y}_2	\bar{Y}_3	\bar{Y}_4	\bar{Y}_5	\bar{Y}_6	\bar{Y}_7																		
000	001	010	011	100	101	110	111																		
1	0	0	0	0	0	0	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1
1	0	0	0	0	1	0	0	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1
1	0	0	0	0	1	1	0	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	0	0	0	1	0	0	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	0	0	1	0	0	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	0	0	1	1	0	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

【例1】用2Kx4b的若干芯片构成一个12KB的存储器, 其地址范围在C0000H~C2FFFH和C4000H~C6FFFH之间, 数据总线D0~D7, 地址总线是A0~A19, 芯片读写控制信号R/W, 且片选信号为3-8译码器输出。

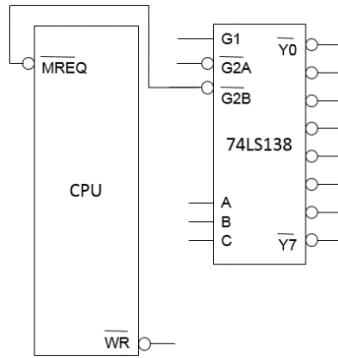
(1) 需要2Kx4b的芯片多少片? 每组芯片地址线如何分配? (12片, A0~A10)

(2) 哪些地址线作3-8译码器的使能端, 哪些做3-8译码器输入端? (使能端: A19~A15, 片选: A11~A13)

(3) 画出存错逻辑电路图 (3-8译码器使能端、输入端、输出连线, 以及组成12KB存储芯片电路图)



【例2】设CPU有16根地址线, 8根数据线, 用MREQ作为访存控制信号 (低电平有效), 用WR作为读写控制信号 (高电平读, 低电平写)。现有下列芯片: 1Kx4的RAM, 2Kx4的RAM, 4Kx4的RAM, 2Kx8的ROM, 4Kx8的ROM, 以及74LS138译码器。其中存储芯片引脚 RW (高电平读, 低电平写), 片选引脚CS。仅选用上述芯片 (不增加其他门电路和芯片), 要求地址空间分配6000H-67FFH为系统程序区 (ROM芯片), 6800H-77FFH为用户程序区 (RAM芯片)。



(1) 需要选用哪几种存储芯片？各需要多少片？并写出各个存储芯片的地址范围。

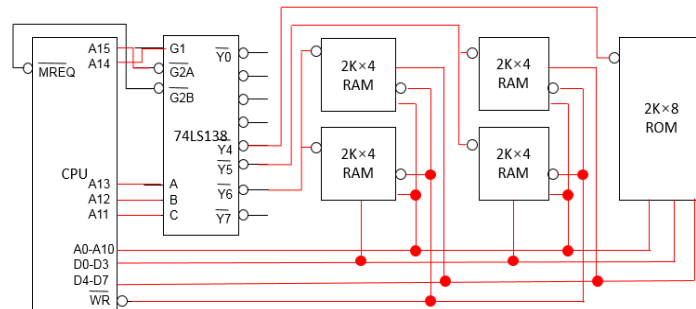
需要一片 $2K \times 8$ 的ROM；四片 $2K \times 4$ 的RAM。

$2K \times 8$ 的ROM的地址范围：6000H-67FFH；

第1组（两片 $2K \times 4$ ）RAM的地址范围：6800H-6FFFH；

第2组（两片 $2K \times 4$ ）RAM的地址范围：7000H-77FFH

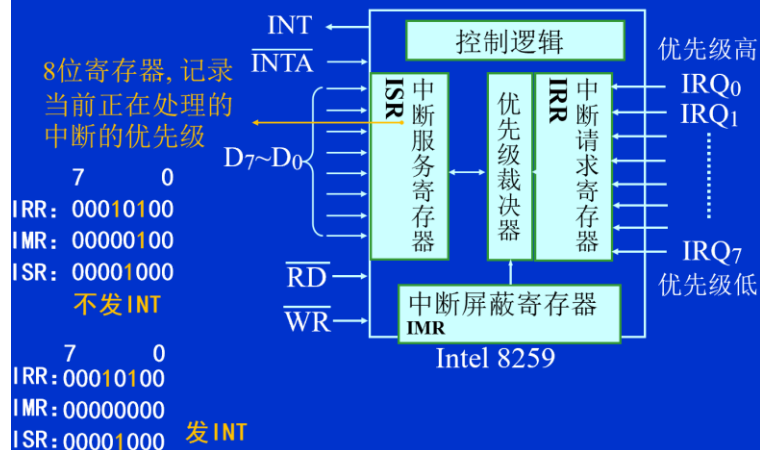
(2) 画出CPU、74LS138和存储芯片之间的连接图。(7分)



九、中断计算

② 采用公共请求线并行判优中断控制器

比如集成的中断控制器(如Intel 8259)集中解决请求信号的接收、屏蔽、判优、编码等问题。



1、计算向量地址

IBM PC: 向量地址=中断类型号 \times 4

模型机: 向量地址=中断类型号+2

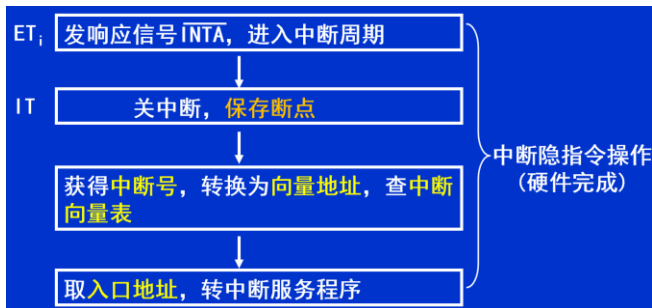
2、根据中断优先级, 判断中断控制器8259是否发出INT信号

	7	0
IRR:	00010100	
IMR:	00000100	
ISR:	00001000	

7 0
IRR: 00010100
IMR: 00000000
ISR: 00001000

中断请求 → Intel 8259 (未被屏蔽的请求判优, 生成相应中断号)

- 公共请求INT → CPU →
- CPU响应中断请求(如果CPU允许响应中断) →
- CPU发出响应信号 $\overline{\text{INTA}}$ 到Intel 8259 →
- Intel 8259将中断号送往数据总线 →
- CPU从数据总线读取中断号, 由中断号形成中断服务程序入口地址, 并转入中断服务程序)



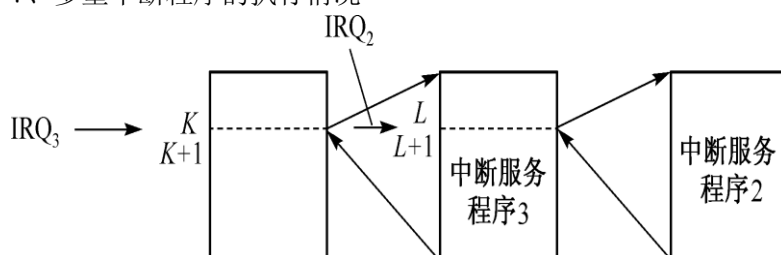
6、多重中断服务程序

```

TIMER PROC FAR
    //保护现场
    PUSH  AX
    PUSH  BX
    PUSH  CX
    PUSH  DX
    PUSH  SI
    PUSH  DI
    //开中断
    STI
    ...      ;处理中断事务
    //关中断
    CLI
    //恢复现场（先进后出原则）
    POP  DI
    POP  SI
    POP  DX
    POP  CX
    POP  BX
    POP  AX
    //开中断
    STI
    //返回主程序
    IRET
TIMER ENDP

```

7、多重中断程序的执行情况



十、判断题

1. 与定点数相比，在位数相同的情况下，浮点数表示范围大、精度高。 (v)
2. 若8位补码表示的机器数01001010，则该机器数的真值为74。 (v)
3. 原码加减运算比补码加减运算简单。 (x)
4. 原码乘法运算比补码乘法运算简单。 (v)
5. 进位制中的基数是：一个数位中允许使用的最大数码值。 (x)
6. 压栈操作是指:将内容写入堆栈指针SP内。 (x)
7. 为减少指令中地址的数目，可以采用隐地址方式。 (v)

8. CPU是计算机的核心部件，包括寄存器组、控存、缓存和主存等部件。 (x)
9. 模型机中的程序计数器PC对汇编程序员可见。 (v)
10. PSW寄存器的特征位是程序员设置的 (x)
11. 暂存器C用于存放源操作数或者源操作数地址，暂存器D存放目的操作数或者目的操作数地址 (v)
12. IR寄存器适用于存放当前指令地址 (x)
13. 现代微处理器缓存采用三级缓存L1、L2和L3，三级缓存都部署与CPU内部。 (v)
14. 晶体振荡器产生的时钟频率低于CPU的主频 (v)
15. 在计算机系统中，可能同时采用同步控制方式与异步控制方式。 (v)
16. 在同步控制方式中，各指令的执行时间并不一定都相同。 (v)
17. 扩展同步总线是指允许时钟周期数可变。 (v)
18. 同步控制的同一时序信号都由CPU产生 (x)
19. 同步控制的时钟周期长度固定不变 (v)
20. 扩展同步控制根据实际时间分配时钟周期数，时钟周期长度不变 (v)
21. 并行加法器中的进位链，必定是并行进位链。 (x)
22. 串行加法器的进位信号是通过串行链逐位形成的，并行加法器的进位信号是同时形成的。 (x)
23. 并行加法器的运算速度取决于全加器单元的速度。 (x)
24. 串行进位链是串行加法器中的进位链。 (x)
25. 运算器实现移位操作，必须使用移位线路。 (x)
26. 并行加法器中的进位链，必定是并行进位链。 (x)
27. 若采用并行进位链，则 $C_3 = G_3 + P_3 C_2$ 。 (x)
28. 中断周期结束后，CPU应进入主程序的下一条指令取指周期。 (x)
29. 在中断方式中，数据传送由中断控制器控制。 (x)
30. DMA控制方式下，外设传送信息到内存，其传送途径为外设→CPU→内存。 (x)
31. 采用DMA方式传送信息，一旦开始传送后，CPU必须停止执行程序一段时间，直到批量传送结束，才能恢复工作。 (x)
32. CM 属于 CPU 的一部分 (v)
33. 微程序控制方式下，微程序存放在主存中。 (x)
34. CPU访问主存储器的时间是由存储体的容量决定的，存储容量越大，访问存储器所需要的时间就越长。 (x)
35. 存储器包括主存和辅存，全部都在CPU外部 (x)
36. 主设备发送数据，从设备接收数据 (x)
37. 串行接口与系统总线串行传送，接口与外设串行传送 (x)