

作业 1

1. 已知 (DS)=2000H, (ES)=2100H, (SS)=1500H, (BX)=0100H, (BP)=0010H, (SI) = 00A0H, 数据段中变量名 VAL 的偏移地址为 0050H, 试指出下列源操作数的寻址方式是什么? 其物理地址为多少? (每题 3 分, 共 36 分)

- (1) MOV AX, 0ABH
- (2) MOV AX, BX
- (3) MOV AX, [100H]
- (4) MOV AX, VAL
- (5) MOV AX, [BX]
- (6) MOV AX, ES:[BX]
- (7) MOV AX, [BP]
- (8) MOV AX, [SI]
- (9) MOV AX, [BX+10]
- (10)MOV AX, VAL[BX]
- (11)MOV AX, [BX][SI]
- (12)MOV AX, VAL[BX][SI]

答案:

- (1) 立即数寻址, 无物理地址
- (2) 寄存器寻址, 无物理地址
- (3) 直接寻址, 20100H
- (4) 直接寻址, 20050H
- (5) 寄存器间接寻址, 20100H
- (6) 寄存器间接寻址, 21100H
- (7) 寄存器间接寻址, 15010H
- (8) 寄存器间接寻址, 200A0H
- (9) 相对寄存器寻址, 2010AH
- (10)相对寄存器寻址, 20150H
- (11)基址变址寻址, 201A0H
- (12)相对基址变址寻址, 201F0H

2. 已知 (DS)=2000H, (BX)=0100H, (SI) = 0002H, 存储单元[20100H]~[20103H]依次存放 12H 34H 56H 78H, [21200H] ~ [21203H]依次存放 2AH 4CH 0B7H 65H, 请说明执行完下列每条指令后 AX 寄存器的内容。(每题 4 分, 共 28 分)

- (1) MOV AX, 1200H
- (2) MOV AX, BX
- (3) MOV AX, [1200H]
- (4) MOV AX, [BX]
- (5) MOV AX, [BX+1100H]
- (6) MOV AX, [BX+SI]
- (7) MOV AX, [BX][SI+1100H]

答案:

(AX)=1200H

(AX)=0100H

(AX)=4C2AH

(AX)=3412H

(AX)=4C2AH

(AX)=7856H

(AX)=65B7H

3. 设 (DS) = 2000H, (BX) = 1256H, (SI) = 528FH, 位移量 TABLE = 20A1H, [232F7H] = 3280H, [264E5H] = 2450H, 则执行下列段内间接寻址的转移指令后, 转移的目的地址分别是多少。(每题 4 分, 共 12 分)

- (1) JMP BX
- (2) JMP TABLE[BX]
- (3) JMP [BX][SI]

答案:

(1) 1256H

(2) 3280H

(3) 2450H

4. 判断下列指令格式是否正确; 如果有错误, 请指出错误及其原因。(每题 3 分, 共 24 分)

- (1) MOV CX, DL
- (2) MOV IP, AX
- (3) MOV ES, 1234H
- (4) MOV ES, DS
- (5) MOV AL, 300
- (6) MOV [SP], AX
- (7) MOV AX, BX+DI
- (8) MOV 20H, AH

答案：

- (1) 错误，数据长度不匹配
- (2) 错误，IP 不能作为目标操作数，不能为其赋值
- (3) 错误，立即数不能直接传送到段寄存器
- (4) 错误，段寄存器之间不能使用 MOV 指令传送数据
- (5) 错误，数据长度不匹配
- (6) 错误，SP 不能用于寄存器间接寻址，只能 BX, BP, SI, DI。
- (7) 错误，寄存器不能相加作为操作数。常数表达式和地址表达式可以。
- (8) 错误，立即数不能作为目标操作数。

作业 2

1. 若想把首地址为 BLOCK 的字数组的第 6 个字送到 DX 中，请使用以下几种寻址方式写出指令。(每题 8 分，共 24 分)

- (1) 寄存器间接寻址
- (2) 寄存器相对寻址
- (3) 基址变址寻址

答案：

- (1) 寄存器间接寻址

```
MOV BX, OFFSET BLOCK  
ADD BX, (6-1)*2  
MOV DX, [BX]
```

(2) 寄存器相对寻址

MOV BX, OFFSET BLOCK

MOV DX, [BX+(6-1)*2]

或者使用以下指令：

MOV BX, (6-1)*2

MOV DX, BLOCK[BX]

(3) 基址变址寻址

MOV BX, OFFSET BLOCK

MOV SI, (6-1)*2

MOV DX, [BX][SI]

2. 判断下列指令格式是否正确；如果有错误，请指出错误及其原因。（每题 4 分，共 20 分）

(1) XCHG [SI], 30H

(2) POP CS

(3) SUB [SI],[DI]

(4) PUSH AH

(5) ADD AX, DS

答案：

(1) 错误，XCHG 指令的操作数不能为立即数

(2) 错误，POP 指令的操作数不能为 CS，不能人为修改 CS

(3) 错误，SUB 指令的两个操作数不能同时为存储器数据。

(4) 错误，PUSH 指令的操作数不能为 8 位寄存器。

(5) 错误，ADD 指令的操作数不能为段寄存器。

3. 若 TABLE 为数据段中偏移地址为 0032H 的内存单元的符号名，其中存放着 1234H，请问以下两条指令有什么区别？（3 分）执行完指令后 AX 的值分别为什么？（8 分）

(1) MOV AX, TABLE

(2) LEA AX, TABLE

答案：

(1) 区别

- MOV AX, TABLE 指令是从 TABLE 表示的存储器地址中获取一个字数据，传送至 AX 中。

- LEA AX, TABLE 是将 TABLE 的地址装入到 AX 中。

(2) 结果

- (AX)=1234H
- (AX)=0032H

4. 若 (AX)=0E3H, 变量 VALUE 中存放着 79H, 请写出下列指令单独执行后 AX 的值为多少? (每题 3 分, 共 18 分)

- (1) XOR AX, VALUE
- (2) AND AX, VALUE
- (3) OR AX, VALUE
- (4) XOR AX, 0FFH
- (5) AND AX, 0
- (6) TEST AX, 01H

答案:

- (1) 9AH
- (2) 61H
- (3) 0FBH
- (4) 1CH
- (5) 0
- (6) 0E3H

5. 设(DX)=0B9H, (CL)=3, (CF)=1, 请问下列各条指令单独执行后 DX 的值为多少? (每题 3 分, 共 27 分)

- (1) SHR DX, 1
- (2) SAR DX, CL
- (3) SHL DX, CL
- (4) SHL DL, 1
- (5) ROR DX, CL
- (6) ROL DL, CL
- (7) SAL DH, 1
- (8) RCL DX, CL

(9) RCR DL, 1

答案：

- (1) 5CH
- (2) 17H
- (3) 5C8H
- (4) 72H
- (5) 2017H
- (6) 0CDH
- (7) 0B9H
- (8) 5CCH
- (9) 0DCH

作业 3

1. 请分析下列的代码实现什么功能。(20 分)

```
MOV CL, 04  
SHL DX, CL  
MOV BL, AH  
SHL AX, CL  
SHR BL, CL  
OR DL, BL
```

答案：将 (DX,AX) 双字的内容整体向左移 4 位，末位填充 0。

2. 已知 X, Y, Z, V 均为 16 位有符号数，请分析下列中代码的功能，写出其表达式。(20 分)

```
MOV AX, X  
IMUL Y  
MOV CX, AX  
MOV BX, DX  
MOV AX, Z  
CWD  
ADD CX, AX  
ADC BX, DX
```

```
SUB CX, 540  
SBB BX, 0  
MOV AX, V  
CWD  
SUB AX, CX  
SBB DX, BX  
IDIV X
```

答案: $(V - (X * Y + Z - 540)) / X$

3. 已知 X, Y, Z 均为 32 位数据, 分别存放在 X, X+2, Y, Y+2, Z 和 Z+2 的存储单元中, 请编程实现算式 $W = X + Y + 24 - Z$, 并将结果存入 W 和 W+2 单元中。(30 分)

答案:

```
MOV AX, X  
MOV DX, X+2 ; X  
ADD AX, Y  
ADC DX, Y+2 ; X+Y  
ADD AX, 24  
ADC DX, 0 ; X+Y+24  
SUB AX, Z  
SBB DX, Z+2 ; X+Y+24-Z  
MOV W, AX  
MOV W+2, DX ; 存储
```

4. 请使用移位指令实现将 AL 中的无符号数乘以 15 倍, 结果存入 AX 寄存器。(30 分)

答案:

答案不唯一。

方法 1: 8 倍+4 倍+2 倍+1 倍

```
XOR AH, AH  
MOV BX, AX  
MOV CX, AX  
MOV DX, AX  
SHL DX, 3
```

SHL CX, 2
SHL BX, 1
ADD AX, BX
ADD AX, CX
ADD AX, DX

方法 2: 16 倍 - 1 倍

XOR AH, AH
MOV BX, AX
SHL AX, 4
SUB AX, BX

作业 4

1. 请以 16 进制写出以下题目的答案。(每道题 3 分, 共 30 分)

- (1) 十进制 109.6875 和 46.625 对应的十六进制数是_____和_____。
- (2) 二进制 11010110101011.111011 对应的十六进制为_____。
- (3) 设机器字长为 8 位, 则-20 的补码为_____。若机器字长为 32 位, 则-20 的补码为_____。若机器字长为 16 位, 则-240 的补码为_____。
- (4) 设 X 为回车、换行和 ‘2’ 这 3 个字符 ASCII 码之和, 则 X 为_____。哪个字符与 X 有相同的 ASCII 码_____。
- (5) 设某变量在 DS 段中, (DS) =4DE8H, 其物理地址为 50C20H, 则其偏移地址为_____。
- (6) 若 A 和 B 均为有符号数, 若 A≥B 时程序转移到标号 L12, 请写出对应的指令_____。

答案: (1) 6D.BH, 2E.AH; (2) 35AB.ECH; (3) 0ECH, 0FFFFFECH, 0FF10H;
(4) 49H, ‘I’; (5) 2DA0H; (6) CMP A,B JGE L12

2. 已知 (DS) = 1A8CH, (BX) = 0B4H, 变量 Var 的偏移地址为 12H, (SI) = 34H。请分析以下每条指令的寻址方式和第 2 个操作数的物理地址。(每题 5 分, 共 25 分)

- (1) MOV CX, [2DH]
- (2) MOV CX, [BX]
- (3) MOV CX, Var[BX]

- (4) MOV CX, [BX][SI]
(5) MOV CX, Var[BX][SI]

答案：

- (1) 直接寻址, 1A8EDH
(2) 寄存器间接寻址, 1A974H
(3) 相对寄存器寻址, 1A986H
(4) 基址变址寻址, 1A9A8H
(5) 相对基址变址寻址, 1A9BAH

3. 已知 (AX) = 012CH, (BX) = 0DC1EH, (CL) = 4, 请写出以下指令执行后 AX 或 BX 的值（请以十六进制给出结果。注意：每个题目单独计算，而不是基于前一个题目的结果去考虑）。(每题 3 分，共 45 分)

- (1) AND AX, 0F0FH (AX) = _____
(2) XOR BX, 0F0FH (BX) = _____
(3) OR BX, 0F0FH (BX) = _____
(4) XOR BX, BX (BX) = _____
(5) MOV CH, 2
 MUL CH (AX) = _____
(6) DIV CL (AX) = _____
(7) DIV BL (AX) = _____
(8) SAR BX, CL (BX) = _____
(9) ROR BX, CL (BX) = _____
(10) TEST AX, BX (AX) = _____
(11) NOT AX (AX) = _____
(12) NEG AX (AX) = _____
(13) MOV CL, 7
 ROL BX, CL (BX) = _____
(14) MOV CL, 5
 SAL AL, CL (AX) = _____
(15) MOV CX, 5
 MUL CX (AX) = _____

答案：

- (1) 10CH; (2) 0D311H; (3) 0DF1FH; (4) 0; (5) 58H; (6) 4BH;
(7) 0AH; (8) 0FDC1H; (9) 0EDC1H; (10) 12CH; (11) 0FED3H; (12) 0FED4H;
(13) (BX)=0F6EH, (AX)=12CH; (14) 110H; (5) 5DCH.

作业 5

1. 已知某数据段的定义如下，其段地址为 149BH，请以单元格形式画出其内每个变量在内存中的存储结构，写出其每个存储单元的值（以 16 进制补码形式），用变量名+箭头标注每个变量起始偏移地址所在的位置，并给出 Count1 和 Count2 的值（以十进制给出）。（20 分）

MyData segment

Var1	DB	14, ?, 'bc', 45, \$-2,\$*30
Count1	EQU	\$ - Var1
Var2	DW	'Ba', 2 DUP (26, 'A')
Var3	DW	Var2
Var4	DD	-50
Var5	DD	Var4
Count2	EQU	Var5 - Var2

MyData ends

Count1 = (In decimal)

Count2 = (In decimal)

答案：

评分规则：表格中红字部分每个格 0.5 分，共 16 分；Count1 和 Count2 各 2 分。整个题目共 20 分。

Count1 = 7, Count2 = 16

	14H	Segment Addr of
	9BH	Var4
	0	Offset Addr of
Var5 (23)	13H	Var4
	OFFH	
	OFFH	-50
	OFFH	
	0CEH	
Var4 (19)	0	Offset Addr of
	7	Var2
	0	'A'
Var3 (17)	41H	
	0	26
	1AH	
	0	'A'
	41H	
	0	26
Var2 (7)	1AH	
	42H	'B'
	61H	'a'
	0B4H	6*30
	03H	5-3
	2DH	45
	63H	'c'

2. 设有一个串 Password 为‘It is A80m-94Fb-311D-7491-C25F!’，以‘\$’结尾。程序员 Bob 想设计一个完整格式的汇编程序来统计其数字的个数，存入变量 Count；同时，将每个大写字符转化为小写字符。基于以上需求，请设计程序进行实现。(共 80 分)

答案：框中每条指令错，扣 4 分，其他指令错扣 2 分。

DATA SEGMENT

Password DB ‘It is A80m-94Fb-311D-7491-C25F!', '\$'

; Define the string Password

Count DW ?

; Define the variable Count

DATA ENDS

CODE SEGMENT

ASSUME CS:CODE, DS:DATA

; Assume segment registers

Start:

MOV AX, DATA

; Set DS

MOV DS, AX

MOV BX, offset Password 或
LEA BX, Password

; Set BX to be the offset of string Password

MOV CX,0

; CX is used to count the number of digits

AGAIN:

MOV AL, [BX]

CMP AL,'\$'

; Judge the ending of Password

JE DONE

CMP AL,'0'

; Judge whether AL is a digit or not

JB NEXT

CMP AL,'9'

JA UPPER_CASE

; If outside of digit 0~9, judge the upper-case letter

INC CX

JMP NEXT (Optional)

; It is a digit, and update the number of digits

; Judge whether AL is an upper-case letter or not

UPPER_CASE:

CMP AL,'A'

JB NEXT

CMP AL,'Z'

JA NEXT

OR AL, 20H 或
ADD AL, 20H 或
XOR AL, 20H

; Convert the upper-case letter to lower-case letter

MOV [BX], AL

; Write lower-case letter back to Password

NEXT:

INC BX

; Ready to deal with the next character in Password

JMP AGAIN

; Program jump

DONE:

MOV Count, CX

; Save the number of digits to variable Count

; Program exit

MOV AX,4C00H

INT 21H

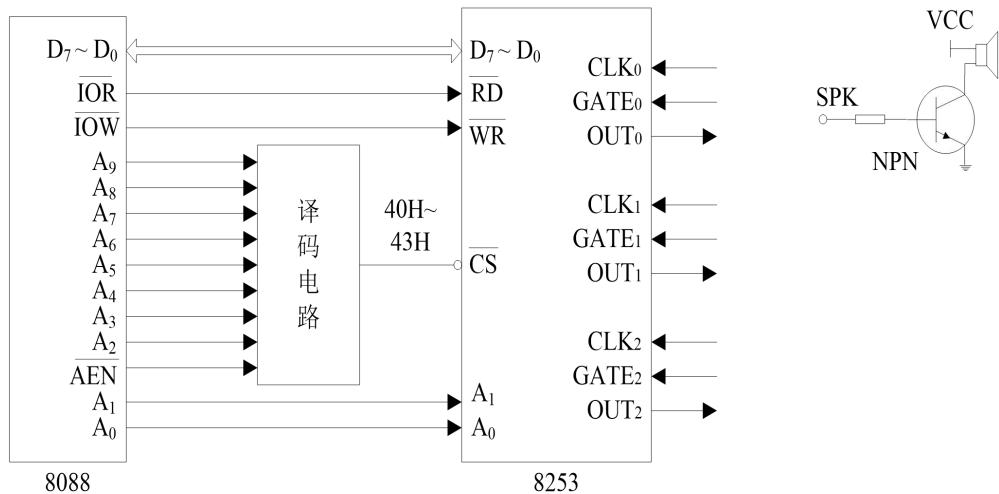
CODE ENDS

END START

作业 6

1. 已知有图 1 所示的电路。现有一个 6MHz 的时钟源，1 个扬声器，1 个 LED 灯，若干保护电阻，+5V 电源和 GND。请完成以下设计。(50 分)

- (1) 设有一个逻辑电平开关和上拉电阻。若要实现每按开关 8 0000 下，LED 灯亮一下然后熄灭，如此反复。请问：一个计数器是否够用？所选用计数器的工作方式和初值分别是什么？(10 分)
- (2) 请根据(1)的要求，在图 1 上进行硬件设计(直接在图 1 上画出所选用计数器的 CLK、GATE 和 OUT、逻辑开关、上拉电阻、LED 灯、保护电阻、+5V 电源和 GND 等之间的连线)。(10 分)
- (3) 请根据 (1) ~ (2) 编写程序段进行实现。(10 分)
- (4) 若要使图 1 电路不断发出“叮铃铃”的电话声(3 个音节的频率分别为 750Hz、600Hz 和 300Hz)，请在图 1 上设计相应的硬件电路，并编程实现。(20 分)



答案：

第 1 题：

- (1) $N = 80000 > 65535$ ，故一个计数器不能满足要求，需要 2 个计数器级联。可以选择其中任意 2 个计数器，这里选择 T_0 和 T_1 ，它们分别工作在方式 3 和方式 2。初值可拆分为 $N_0 = 200$, $N_1 = 400$ (可有多种拆分形式，只要保证其乘积为 80000 即可)。
- (2) 硬件设计见图。
- (3) 初始化编程

MOV AL, 00100111B ; 27H

MOV 43H, AL

```
MOV AL, 2 ; initial value is 200
```

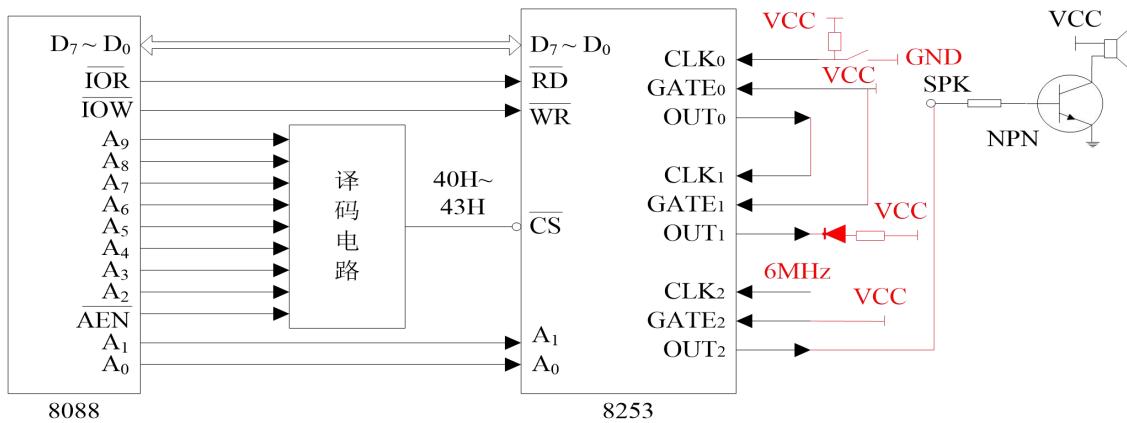
```
MOV 40H, AL
```

```
MOV AL, 01100101B ; 65H
```

```
MOV 43H, AL
```

```
MOV AL, 4 ; initial value is 400
```

```
MOV 41H, AL
```



(4) “Ding-Ling-Ling” 3 个音节对应的初值为： $6M/750H=8000$, $6M/600H = 10000$, $6M/300H=20000$ 。OUT₂ 用于驱动扬声器。设 delay500ms 为已定义好的延迟子程序，若考虑音节之间的间歇，则在每次初始化 counter2 后 call delay500ms；若不考虑该间歇，则可以删除 call delay500ms。

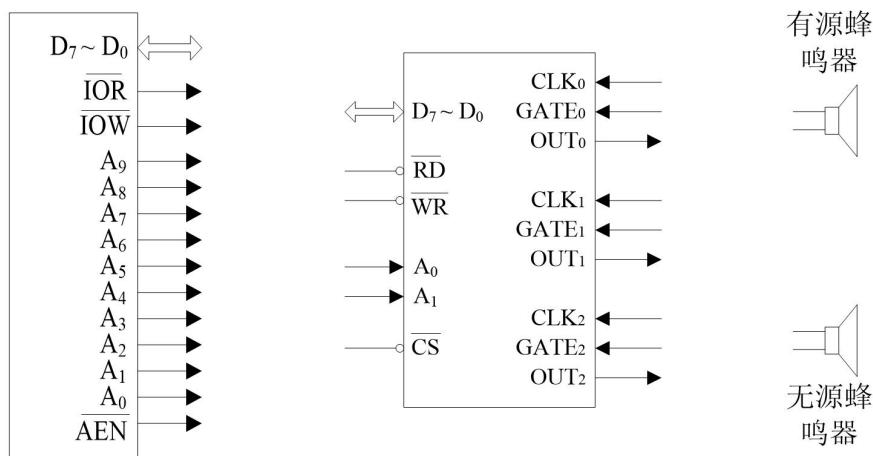
```
L1: MOV AL, 10100111B ; 0A7H
      OUT 43H, AL
      MOV AL, 80H
      OUT 42H, AL
      CALL delay500ms ; 音节之间的间歇（可选）
      MOV AL, 10110110B ; 0B6H
      OUT 43H, AL
      MOV AX, 10000
      OUT 42H, AL
      MOV AL, AH
      OUT 42H, AL
```

```

CALL delay500ms
MOV AL, 10110110B ; 0B6H
OUT 43H, AL
MOV AX, 20000
OUT 42H, AL
MOV AL, AH
OUT 42H, AL
CALL delay500ms
JMP L1

```

2. 图 2 中有 8088 处理器、8254 定时/计数器、有源蜂鸣器、无源蜂鸣器。现有一个 5MHz 的时钟源以及 +5V 和 GND。已知无源蜂鸣器输入不同频率的方波信号后会发出不同音调的声音，而有源蜂鸣器只要两端分别接 +5V 和 GND 即可发出固定频率的声音。请根据以下要求完成设计。(50 分)



- (1) 假设 8254 的地址范围为 1D0H~1D3H，“汇编语言”四个字的发音频率分别为 1250Hz、2000Hz、625Hz、1000Hz，请在图 2 上完善电路设计，用 8254 驱动相应的蜂鸣器使其重复播放“汇编语言”，并编写主要程序段进行实现（假设不考虑每个音节的发音时长和两个音节之间的间隙）。(20 分)
- (2) 已知有独立按键、上拉电阻等元器件，请在图 2 的基础上设计一个按键计次电路，使得每按键 18 万次，相应的蜂鸣器产生一次报警然后迅速停止，之后重复。(10 分)
- (3) 请根据 (3) 的要求编写主要程序段进行实现。(20 分)

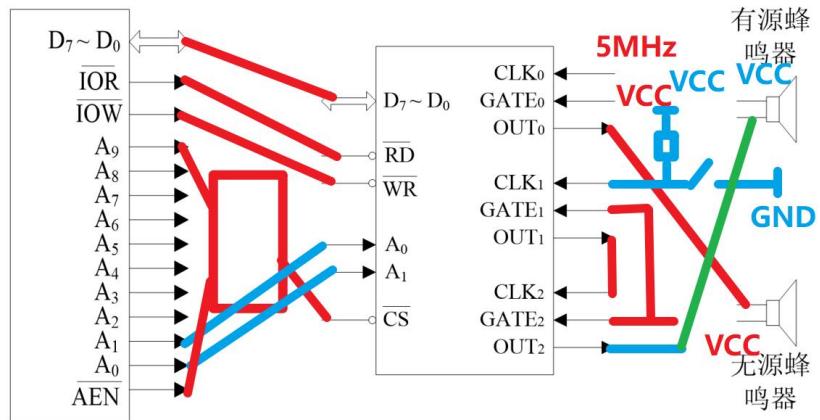
答案：

- (1) “汇编语言”四个字发音对应的初值为： $5M/1250=4000$ ， $5M/2000=2500$ ，

$5M/625=8000$, $5M/1000=5000$ 。使用 Counter 0 驱动无源蜂鸣器，其初始化编程如下：

```
MOV DX, 1D3H      ; “汇”
MOV AL, 00100111B
OUT DX, AL
MOV DX, 1D0H
MOV AL, 40H
OUT DX, AL
MOV DX, 1D3H      ; “编”
MOV AL, 00100111B
OUT DX, AL
MOV DX, 1D0H
MOV AL, 25H
OUT DX, AL
MOV DX, 1D3H      ; “语”
MOV AL, 00100111B
OUT DX, AL
MOV DX, 1D0H
MOV AL, 80H
OUT DX, AL
MOV DX, 1D3H      ; “言”
MOV AL, 00100111B
OUT DX, AL
MOV DX, 1D0H
MOV AL, 50H
OUT DX, AL
```

(2) 见图中蓝色和绿色连线部分。



注意：上图中 A1 和 A0 需要交换一下位置。

- (3) $180000 > 65535$, 需要 Counter1 和 Counter2 级联, 工作方式分别为方式 3 和方式 2, 拆分的初值分为为 300 和 600 (可以任意拆分, 只要保证其乘积为 180000 且均小于 65535 即可)。初始化编程如下：

```

MOV DX, 1D3H
MOV AL, 01100111B
OUT DX, AL
MOV DX, 1D1H
MOV AL, 3
OUT DX, AL

```

```

MOV DX, 1D3H
MOV AL, 10100101B
OUT DX, AL
MOV DX, 1D2H
MOV AL, 6
OUT DX, AL

```

作业 7

- 什么是中断类型号？什么是中断向量表？请画出中断向量表的结构。已知一个中断的类型号为 25H，中断向量为 5566H: 7788H，试画图说明它在中断向量表中的位置和内容，并编程实现：分别采用直接写入法和 DOS 功能调用方法 2 种方法来设置该中断的向量。(20 分)

答案：

中断类型号：微机系统中有多个中断源。为便于区分，为每个中断源分配一个编号，即中断类型号。(2分)

中断向量表：将系统中所有的中断向量集中起来，按中断类型号从小到大的顺序存放在内存某区域。该存储区域称为中断向量表。(2分)

中断向量表的结构如下。(4分)



中断向量的绘制：(4分，每个值1分)



直接写入法设置中断向量：(4分)

```
CLI          ; 关中断  
MOV AX, 0  
MOV DS, AX      ; 中断向量表段基址为 0  
MOV AX, OFFSET P1 ; 中断类型号为 n  
MOV BX, 25H*4
```

```
MOV DS:[BX], AX      ; 偏移地址放入 4n,4n+1 单元  
MOV AX, SEG P1  
MOV DS:[BX+2], AX    ; 段地址写入 4n+2,4n+3 单元  
STI                  ; 开中断
```

DOS 功能调用法设置中断向量：(4 分)

```
PUSH DS             ; 原 DS 入栈  
MOV AX, SEG P1      ; 段基址存入 DS 中  
MOV DS, AX  
MOV AX, OFFSET P1  
MOV DX, AX          ; 偏移地址存入 DX 中  
MOV AL, 25H          ; 中断类型号在 AL 中  
MOV AH, 25H          ; DOS 调用功能号  
INT 21H  
POP DS              ; DS 出栈
```

2. 已知中断向量表中，001C4H 单元中存放 2200H，001C6H 单元中存放 3040H，则对应中断的类型号是多少？中断服务程序的入口地址是什么？请分别采用直接写入法和 DOS 功能调用两种方法进行编程实现。(12 分)

答案：

- (1) $001C4H/4 = 113$ (2 分)
- (2) 入口地址：3040H: 2200H (2 分)
- (3) 直接写入法 (4 分)

```
CLI                  ; 关中断  
MOV AX, 0  
MOV DS, AX          ; 中断向量表段基址为 0  
MOV AX, OFFSET P1  ; 中断类型号为 n  
MOV BX, 1C4H  
MOV DS:[BX], AX      ; 偏移地址放入 4n,4n+1 单元  
MOV AX, SEG P1
```

```
MOV DS:[BX+2], AX ; 段地址写入 4n+2,4n+3 单元  
STI ; 开中断
```

(4) DOS 功能调用法 (4 分)

```
PUSH DS ; 原 DS 入栈  
MOV AX, SEG P1 ; 段基址存入 DS 中  
MOV DS, AX  
MOV AX, OFFSET P1  
MOV DX, AX ; 偏移地址存入 DX 中  
MOV AL, 113 ; 中断类型号在 AL 中  
MOV AH, 25H ; DOS 调用功能号  
INT 21H  
POP DS ; DS 出栈
```

3. 按照要求对 8259A 进行初始化编程：单片 8259A 应用于 8086 系统，中断请求信号为边沿触发，中断类型号为 80H~87H，采用自动中断结束方式、特殊全嵌套、非缓冲方式。假设 CPU 的 A1 地址线连接 8259A 的 A0 信号线，8259A 的端口地址为 4A0H 和 4A2H。(10 分)

- (1) 请分析 ICW1~ICW4 的访问地址；(4 分)
- (2) 请编写 8259A 初始化程序段。(6 分)

答案：

(1) 因为 CPU 的 A1 地址线连到 8259A 的 A0，需要根据 8259A 的 A0 来确定奇偶地址。当 A0=0 时为偶地址（对应 4A0H），A0=1 时为奇地址（4A2H）。故 ICW1~ICW4 的访问地址依次为 4A0H, 4A2H, 4A2H, 4A2H。

(2) 初始化程序段 (6 分，每个 ICW 值 2 分)

```
MOV AL, 00010011B ; ICW1: 边沿触发、单片、需 ICW4  
MOV DX, 4A0H  
OUT DX, AL  
MOV AL, 80H ; ICW2: 80H~87H 的高 5 位  
MOV DX, 4A2H  
OUT DX, AL  
MOV AL, 00010011B ; ICW4: 特殊全嵌套、非缓冲方式、AEOI  
MOV DX, 4A2H
```

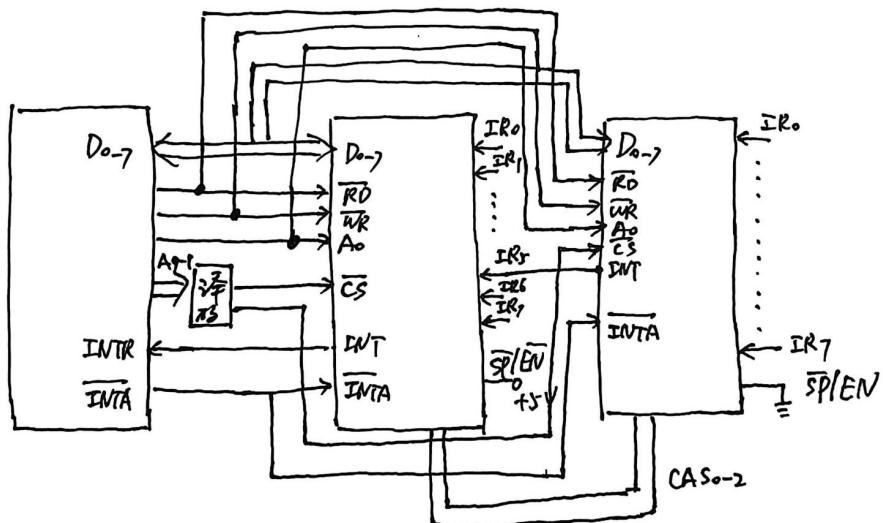
OUT DX, AL

4. 设 8086 系统中有两片 8259A，从片 8259A 连主片 8259A 的 IR5 端。设 CPU 地址线的 A1 分别连两片 8259A 的 A0 端，主片的端口地址为 2B0H 和 2B2H，从片的端口地址为 2C0H 和 2C2H，主片 IR0 的中断类型号为 50H，从片 IR0 的中断类型号为 60H；所有的中断请求都是上升沿触发，采用非自动中断结束方式，SP/EN 用作输入。

- (1) 请画出 CPU 与主从片 8259A 之间的硬件连线图;
 - (2) 请分析主、从片的奇偶端口地址; (4 分)
 - (3) 请编程分别实现对主、从片 8259A 的初始化。 (8 分)

答案：

- (1) 示意图如下: (8分)



- (2) 主从片的奇地址分别为 2B2H 和 2C2H, 偶地址分别为 2B0H 和 2C0H。(4 分)
(3) 初始化编程

主片：(4分)

MOV AL, 00010001B ;ICW1: 边沿触发、级联、需 ICW4

MOV DX, 2B0H

OUT DX, AL

MOV AL, 50H ; ICW2: 50H~57H 的高 5 位

MOV DX, 2B2H

OUT DX, AL

MOV AL, 00100000B ;ICW3: 主片 IRO5 接有从片

```
MOV DX, 2B2H  
OUT DX, AL  
MOV AL, 00010001B ; ICW4: 特殊全嵌套、非缓冲方式、EOI  
MOV DX, 2B2H  
OUT DX, AL
```

从片: (4 分)

```
MOV AL, 00010001B ; ICW1: 边沿触发、级联、需 ICW4  
MOV DX, 2C0H  
OUT DX, AL  
MOV AL, 60H ; ICW2: 60H~67H 的高 5 位  
MOV DX, 2C2H  
OUT DX, AL  
MOV AL, 00000101B ; ICW3: 从片连到主片 IRQ5 上  
MOV DX, 2C2H  
OUT DX, AL  
MOV AL, 00000001B ; ICW4: 普通全嵌套、非缓冲方式、EOI  
MOV DX, 2C2H  
OUT DX, AL
```

作业 8

在 PC 系统中, 主片的 IR7 和从片的 IR1(设类型号 30H~37H)分别接脉冲开关 K1 和 K2。(100 分)

要求:

(1) 按不同的开关, 进入不同的 ISR, 有不同的相应。(80 分)

- 按 K1 时, 在屏幕上显示一些内容(自定义), 例如: “M7” or “This is a Master IR7 int”。
- 按 K2 时, 在屏幕上显示一些内容(自定义), 例如: “S1” or “This is a Slave IR1 int”。

(2) 尝试同时按 K1 和 K2, 设计一种方案来通过输出辨别这两个中断的优先级。(20 分)

答案：

(1) 编程。(80 分)

```
SSTACK SEGMENT STACK
```

```
    DW 32 DUP(?)
```

```
SSTACK ENDS
```

```
CODE SEGMENT
```

```
    ASSUME CS:CODE
```

```
START: PUSH DS
```

```
    MOV AX, 0000H
```

```
    MOV DS, AX
```

```
    MOV AX, OFFSET MIR7 ;设置主片的中断向量
```

```
    MOV SI, 003CH ; IR7
```

```
    MOV [SI], AX
```

```
    MOV AX, CS
```

```
    MOV SI, 003EH
```

```
    MOV [SI], AX
```

```
    MOV AX, OFFSET SIR1 ;设置从片的中断向量
```

```
    MOV SI, 00C4H ; IR1
```

```
    MOV [SI], AX
```

```
    MOV AX, CS
```

```
    MOV SI, 00C6H
```

```
    MOV [SI], AX
```

```
    CLI
```

```
    POP DS
```

```
;初始化主片 8259
```

```
    MOV AL, 11H
```

```
    OUT 20H, AL ;ICW1
```

```
    MOV AL, 08H
```

```
OUT 21H, AL           ;ICW2
MOV AL, 04H
OUT 21H, AL           ;ICW3
MOV AL, 01H
OUT 21H, AL           ;ICW4
;初始化从片 8259
MOV AL, 11H
OUT 0A0H, AL          ;ICW1
MOV AL, 30H
OUT 0A1H, AL          ;ICW2
MOV AL, 02H
OUT 0A1H, AL          ;ICW3
MOV AL, 01H
OUT 0A1H, AL          ;ICW4

MOV AL, 0FDH
OUT 0A1H,AL           ;OCW1 = 1111 1101
MOV AL, 6BH
OUT 21H, AL           ;主 8259 OCW1
STI

AA1: NOP
JMP AA1

MIR7:   STI
        CALL DELAY
        MOV AX, 014DH
        INT 10H           ;显示字符 M
        MOV AX, 0137H
        INT 10H           ;显示字符 7
        MOV AX, 0120H
        INT 10H
```

```
MOV AL, 20H
OUT 20H, AL      ;中断结束命令
IRET

SIR1: STI

MOV AX, 0153H
INT 10H          ;显示字符 S
MOV AX, 0131H
INT 10H          ;显示字符 1
MOV AX, 0120H
INT 10H
MOV AL, 20H
OUT 0A0H, AL
OUT 20H, AL
IRET

CODE    ENDS

END    START
```

(2) 嵌套方案。(20 分)

修改 MIR7 的中断服务程序，让其单个显示字母'A'~'Z'（非字符串），确保该中断服务程序比较长；修改 SIR1 的中断服务程序，让其输出其他字符（例如空格或数字等）。同时按 KK1 和 KK2，捕捉在'A'~'Z'中间嵌入空格或数字的情况，说明 SIR1 中断嵌入进 MIR7，SIR1 的优先级比 MIR7 更高。