

第一章

1. 在计算机中为什么使用二进制数存储数据而不使用十进制数存储数据？

答：计算机是由大量的电子器件组成的，在这些电子器件中，电路的通和断、电位的高和低，用两个数字符号“1”和“0”分别表示容易实现。同时二进制的运算法则也很简单，因此，在计算机内部通常用二进制代码来作为内部存储、传输和处理数据。

2. 完成下列数制之间的转换。

(1) $01011100B = 92D$

(2) $0.10110011B = 0.41D$

(3) $135D = 1111101B$

(4) $99.4375D = 1100011.0111B$

3. 组合型 BCD 码和非组合型 BCD 码有什么区别？写出十进制数 254 的组合型 BCD 数和非组合型 BCD 数

答：1) BCD 码是通常的 8421 码，它用 4 个二进制位表示一个十进制位，一个字节可以表示两个十进制位，即 00~99

2) 非组合 BCD 码用 8 个二进制位表示一个十进制位，实际上只是用低 4 个二进制位表示一个十进制位 0~9，高 4 位任意，但通常默认为 0

3) $254(10) = 0010\ 0101\ 0100(BCD)$

$254(10) = 00100101\ 00000100(BCD)$

4. ASCII 码的编码方法是什么？写出十进制数 205 和字符串 A+B=C 的 ASCII 码。

答：1) ASCII 码的编码方法是使用 7 位二进制数来表示所有的大写和 小写字母，数字 0 到 9、标点符号，以及在美式英语中使用的特殊 控制字符

2) 十进制 205 的 ASCII 码是：011 000 101

3) 字符串 A+B=C 的 ASCII 码是：412B423D43

5. 机器数与真值有什么区别？机器数有哪些特点？

答：1) 真值是所表示的数的大小，一般用十进制表征。

机器数原码，补码，反码都是机器数一种表现形式，或说都属于机器数

2) 机器数的特点：

一：数的符号数值化。实用的数据有正数和负数，由于计算机内部的硬件只能表示两种物理状态（用 0 和 1 表示），因此实用数据的正号“+”或负号“-”，在机器里就用一位二进制的 0 或 1 来区别。通常这个符号放在二进制数的最高位，称符号位，以 0 代表符号“+”，以 1 代表符号“-”。因为有符号占据一位，数的形式值就不等于真正的数值，带符号位的机器数对应的数值称为机器数的真值。例如二进制真值数 -011011，它的机器数为 1011011。

二：二进制的位数受机器设备的限制。机器内部设备一次能表示的二进制位数叫机器的字长，一台机器的字长是固定的。字长 8 位叫一个字节（Byte），现在机器字长一般都是字节的整数倍，如字长 8 位、16 位、32 位、64 位。

6. 写出原码、反码、补码的定义，并说明它们有哪些性质。

答：1) 原码，是一种计算机中对数字的二进制表示方法，是最简单的机器数，数码序列中最高位为符号位，符号位为 0 表示正数，符号位为 1 表示负数；其余有效值部分用二进制的绝对值表示

2) 反码，是计算机中表示二进制数的一种方法，若原码的符号位不变，对其余位逐位取反，即得到其反码

3) 补码，正数的补码与原码相同，负数的补码：符号位为 1，其余位为该数绝对值的原码按位取反，然后整个数加 1。

4) 性质

7. 计算机中为什么采用补码形式存储数据？当计算机的字长 $n=16$ 时，补码的数据表示范

围是多少？

答：原因：

1) 因为使用补码可以将符号位和其他位统一处理，同时，减法也可以按加法来处理

2) 两个用补码表示的数相加时，如果最高位（符号位）有进位，则进位被舍弃

当计算机的字长 $n=16$ 时，补码的数据表示范围是：— 32768 ~ +32767

8. 设机器字长 $n=8$ ，求下列各数的原码、反码和补码。

(1) $X=+1101001B$

原码： $X=01101001$ 反码： $X=01101001$ 补码： $X=01101001$

(2) $X=+1111000B$

原码： $X=01111000$ 反码： $X=01111000$ 补码： $X=01111000$

(3) $X=-1000000B$

原码： $X=11000000$ 反码： $X=10111111$ 补码： $X=11000000$

(4) $X=-1101011B$

原码： $X=11101011$ 反码： $X=10010100$ 补码： $X=10010101$

9. 微型计算机系统的硬件由哪些部件组成？简述它们的主要功能。

答：构成计算机的硬件系统通常有“五大件”组成：输入设备、输出设备、存储器、运算器和控制器。

输入设备： 将数据、程序、文字符号、图象、声音等信息输送到计算机中。常用的输入设备有，键盘、鼠标、数字化仪器、光笔、光电阅读器和图象扫描器以及各种传感器等。

输出设备： 将计算机的运算结果或者中间结果打印或显示出来。常用的输出设备有：显示器、打印机、绘图仪等。

存储器： 将输入设备接收到的信息以二进制的数据形式存到存储器中。存储器有两种，分别叫做内存储器和外存储器。

运算器： 是完成各种算术运算和逻辑运算的装置，能作加、减、乘、除等数学运算，也能作比较、判断、查找、逻辑运算等。

控制器： 是计算机指挥和控制其它各部分工作的中心，其工作过程和人的大脑指挥和控制人的各器官一样

10. CPU 执行指令时，通常需要哪些步骤？

答：取指令 ----- 分析指令 ----- 执行指令。

11. 简述计算机的基本工作原理。

答：计算机的基本原理是存储程序和 程序控制。预先要把指挥 计算机 如何进行操作指令序列（称为程序）和原始数据通过 输入设备 输送到计算机内存贮器中。 每一条指令中明确规定了计算机从哪个地址取数，进行什么操作，然后送到什么地址去等步骤

12. 微型计算机的主要性能指标有哪些？

答：常用的微型计算机性能指标主要有：字长，主频，内存容量，指令书，基本指令执行时间，可靠性，兼容性，性能价格比等。

13. 微型系统中采用了哪些先进技术？简述这些先进技术的主要特点。

答：微型系统中采用的先进技术：

第二章

1. 8086/8088 CPU 可直接寻址多少个内存 (字节)单元? 多少个 I/O 端口? 它们的外部数据总线各有多少根?

答: 8086 可寻址 1M 个内存单元, 65536 个 I/O 端口, 外部有 16 根数据线

8088 可寻址 1M 个内存单元, 256 个 I/O 端口, 外部有 8 根数据线。

2. 8086 CPU 由那两部分组成? 它们大致是如何工作的?

答: 8086CPU 由指令执行单位和总线接口单元两大部分构成

工作过程:

1) 读存储器

2) EU 从指令队列中取走指令, 经 EU 控制器译码分析后, 向各部件发控制命令, 已完成执行指令的操作

3) 指令队列满, 则 BIU 处于空闲状态

4) 指令执行过程中, 如果需要进行存取数据, EU 就要求 BIU 完成相应的总线周期

5) 在程序转移时, 先清空队列, 再去新的地址处取之

6) ALU 完成算术运算, 逻辑运算或移位操作

3. CPU、EU 和 BIU 的英文全称和中文含义各是什么?

答: CPU 的英文全称是 Central Processing Unit, 中文名称即中央处理单元, 也称为 微处理器, 是微型计算机的核心部件, 由运算器 ALU, 控制器, 寄存器组以及总线接口等部件组成, 主要完成各种运算, 负责对整机的控制

EU 与 BIU

4. 8086 CPU 有哪些寄存器? 各有什么用途?

答: 寄存器 AX, AL: 在乘法指令中, 存放乘数和乘积;

在除法指令中, 存放被除数和商;

在未组合 BCD 码运算的校正指令中;

在某些串操作指令 (LODS, STOS, SCAS) 中

在输入输出指令中作数据寄存器

寄存器 AH: 在 LAHF 指令中作目的寄存器

寄存器 AL: 在组合型 BCD 码的加减法校正指令中

在 XLAT 指令中作目的寄存器

寄存器 BX: 在 XLST 指令中基址寄存器

寄存器 CX: 在循环指令中作循环次数计数器

寄存器 CL: 在移位指令中作移位次数计数器

寄存器 DX: 在字数据的乘法和除法指令中辅佐累加器

寄存器 SP: 在堆栈操作中作堆栈指针

寄存器 SI: 在串操作指令中作源变址寄存器

寄存器 DI: 在串操作指令中作目的变址寄存器

5. 两个带符号数 1011 0100B 和 1100 0111B 相加运算后各标志位的值是多少? 哪些标志位是有意义的?

答: 两个带符号数 1011 0100B 和 1100 0111B 相加运算后各标志位的值是 101111011

PF=1, ZF=0, SF=0, OF=1 是有意义的

6. 说明下列 8086 引脚信号的功能 AD15~AD0、A19/S6~A16/S3、 \overline{RD} 、 \overline{WR} 、M/ \overline{IO} 、CLK、

RESET、INTR、NMI、ALE、DT/ \overline{R} 、 \overline{DEN} 。

答：AD15~AD0：地址/数据总线；A19/S6~A16/S3：地址/状态总线； \overline{RD} ：读控制

\overline{WR} ：写控制 M/ \overline{IO} ：存取器 I/O 控制 CLK：系统时钟 RESET:系统复位

INTR：可屏蔽中断请求 NMI：不可屏蔽中断请求 ALE：地址锁存允许

DT/ \overline{R} ：数据发送/接受 \overline{DEN} ：数据允许

7. 已知段地址：偏移地址分别为以下数值，它们的物理地址各是什么？

(1)1200:3500H 答：=15500H

(2)FF00:0458H 答：=FF458H

(3)3A60:0100H 答：=3A700H

8. 段基地址装入如下数值，则每段的起始地址和结束地址分别是什么？

(1)1200H 答：1200H 12000H~21FFFH

(2)3F05H 答：3F05H 3F050H~4F04FH

(3)0FFE0H 答：0FFE0H 0FFE0H~1FFDFH

9. 已知：CS:IP=3456:0210H,CPU 要执行的下条指令的物理地址是什么？

答：CS:IP=3456:0210H, 要执行的下一条指令的物理地址是 34770H

10. 什么叫堆栈？它有什么用处？如何设置堆栈？

答：堆栈是 堆栈 是连续的 内存单元，存取方式遵循“先进后出”的原则。主要功能是暂时存放数据和地址，通常用来保护断点和现场。

堆栈的设置

11. 设 SS:SP=2000:0300H 则堆栈在内存中的物理地址范围是什么？执行两条 PUSH 指令后 SS:SP=? 再执行一次 PUSH 后，SS:SP=?

答：SS:SP=2000:0300H，堆栈的物理地址范围是 20000~2FFFFH，执行 2 条 PUSH 后，SS:SP=2000:02FCH，再执行一条 PUSH 后 SS:SP=2000:02FAH

12. 如果从存储单元 2000H 开始存放的字节数据为：3AH,28H,56H，4FH 试画出示意图说明：从 2000H 到 2001H 单元开始取出一个字数据各要进行几次操作，取出的数据分别等于多少。

答：(2000H)=3AH，(2001H)=28H，(2002H)=56H，(2003H)=4FH，从 2000H 取一个字要 1 次操作，数据为 283AH；从 2001H 取一个字要 2 次操作，数据为 5628H。

13. 8086 工作于最小模式时，硬件电路主要由哪些部件组成？为什么要用地地址锁存器、数据缓冲器和时钟产生器？

答：1) 8086 工作于最小模式时，硬件电路主要由 1 片 8086CPU，1 片 8284 时钟发生器，3 片 74LS373 锁存器和 2 片 74LS245 双向数据缓冲器组成。

2) 用地地址锁存器的原因：如果总线上负载超过 10 个，单独靠总线不能驱动，需要加总线驱动器提高带负载能力。

3) 用时钟产生器的原因：产生系统的时钟信号，对 READY 和 RESET 信号进行同步

14. 8086/8088 CPU 各用几片地址锁存器、数据缓冲器构成最小模式系统？为什么？

答：λλλ

8086 CPU

存储器 I/O 接口芯片

时钟发生器 (1 片 ⓧ

地址锁存器 (3 片 ⓧ

8284A)

8282 / 8283 或 74LS373) 8286 / 8287 或 74LS245)

双向数据总线收发器 (2 片)

23:49

15. 时钟产生器 8284A 与 8088CPU 相连时, 输入的晶振频率为 14.31818MHz 从输出端可以产生哪些时钟信号? 他们的频率分别是多少? 8284A 与 8086-2 相连时, 晶振频率为 8MHz 则输出的 OSC 和 CLK86-2 信号的频率分别是多少?

答:

16. 8086 最大模式配置电路中, 8288 总线控制的主要功能是什么? $\overline{\text{MEMR}}$ 、 $\overline{\text{MEMW}}$ 、 $\overline{\text{IOR}}$

和 $\overline{\text{IOW}}$ 信号相当于最小模式中哪些信号的组合?

答:

17. 什么叫总线周期? 一个总线周期一般需要几个时钟周期? 若 8086-2 的时钟频率为 8MHz, 它每秒可执行多少条指令?

答: 总线周期通常指的是 CPU 完成一次访问 MEM 或 I/O 端口操作所需要的时间。一个总线周期由几个时钟周期组成。

8086 的基本总线周期为 4 个时钟周期

每秒可执行的指令

18. 8086 工作于最小模式, 执行存储器读总线周期, T1~T4 周期中主要完成哪些工作?

答: T1 周期: 输出 20 位地址信号, ALE 地址锁存允许信号高电平有效

T2 周期: 高 4 位输出状态信号, 低 16 位高阻态, 准备接收数据

T3 周期: 高 4 位维持状态信号, 低 16 位接收数据

T4 周期: 读总线周期结束

第三章

1. 分别指出下列指令中的源操作数和目的操作数的寻址方式：

- | | | |
|----------------------|---------|---------|
| (1)MOV SI , 200 | 立即寻址 | 寄存器寻址 |
| (2)MOV CX , DATA[SI] | 寄存器相对寻址 | 寄存器寻址 |
| (3)ADD AX , [BX][DI] | 基址变址寻址 | 寄存器寻址 |
| (4)AND AX , BX | 寄存器寻址 | 寄存器寻址 |
| (5)MOV [SI] , AX | 寄存器寻址 | 寄存器间接寻址 |
| (6)PUSHF | 寄存器寻址 | 寄存器间接寻址 |

2. 写出以下指令中内存操作数的所在地址。

- | | |
|------------------------|----|
| (1)MOV AL , [BX+5] | DS |
| (2)MOV [BP+5] , AX | SS |
| (3)INC BYTE PTR[SI+3] | DS |
| (4)MOV DL , ES:[BX+DI] | ES |
| (5)MOV BX , [BX+SI+2] | DS |

3. 设堆栈指针 SP 的初值为 2000H , AX=3000H , BX=5000H , 试问：

(1)执行指令 PUSH AX 后 SP=?

答：执行指令 PUSH AX 后 (SP)=2000H-2=1FFEH;

(2)再执行 PUSH BX 及 POP AX 后 SP=? AX=? BX=?

答：再执行 PUSH BX 及 POP AX 后 (SP)=1FFEH, (AX)=5000H, (BX)=5000H

4. 要想完成把 [2000H] 送 [1000H] 中，用指令：

MOV[1000H] , [2000H] 是否正确？如果不正确，应用什么方法？

答：把[2000H]送[1000H]中,用指令 MOV [1000H] , [2000H] 不正确，应改为：MOV AX,[2000H]
MOV [1000H],AX

5. 假如想从 100 中减去 AL 中的内容，用 SUB 100 , AL 是否正确？如果不正确，应用什么方法？

答：想从 100 中减去 AL 中的内容，用 SUB 100 , AL 不正确，
应改为： MOV BL,100 SUB BL,AL

6. 用两种方法写出从 88H 端口读入信息的指令。再用两种方法写出从 42H 口输出 100H 的指令。

答：读入： (1) IN AL , 88H (2) MOV DX , 0088H IN AL ,

DX 输出： (1) MOV AL , 100H OUT 42H , AL

(2) MOV DX , 0042H MOV AL , 100H OUT DX , AL

7. 假如 AL=20H , BL=10H , 当执行 CMP AL , BL 后，问：

(1)若 AL , BL 中内容是两个无符号数，比较结果如何？影响哪几个标志位？

答：此时 AL>BL , CF=0 , ZF=0

(2)若 AL , BL 中内容是两个有符号数，结果又如何？影响哪几个标志位？

答：此时 AL>BL , OF=0 , SF=0

第四章

1. 编一个程序，统计一个 8 位二进制数中的为“1”的位的个数。

答案：假设 AL 中存放的是需要的 8 位数

```
xor ah,ah
mov cx,8
loop1: sal al,1 ;左移
adc ah,0 ;加 CF
loop loop1
```

最终结果 AH 中为 1 的个数。

2. 编一个程序，使放在 DATA 及 DATA+1 单元的两个 8 位带符号数相乘，乘积放在 DATA+2 及 DATA+3 单元中 (高位在后)。

--先运行如下 sql，建立一个合并函数

```
create function fmerg(@company varchar(200))
returns varchar(8000)
as
begin
declare @str varchar(8000)
set @str=""
select @str=@str+', '+jobname from zhaopin where company=@company set
@str=right(@str,len(@str)-1)
return(@str)
End
go
--调用自定义函数得到结果
```

```
select distinct company, dbo.fmerg(company) from zhaopin
```

3. 若在自 1000H 单元开始有一个 100 个数的数据块，要把它传送到自 2000H 开始的存储区中去，用以下三种方法，分别编制程序：

(1) 不用数据块传送指令；

(2) 用单个传送的数据块传送指令；

(3) 用数据块成组传送指令。

解：

```
(1)      LEA      ST,      1000H
          LEA      DI,      2000H
          MOV      CX,      100
L1:  MOV      AX,      [SI]
          MOV      [DI],    AX
          MOV      [DI],    AX
          LOOP     L1
          HLT
```

```
(2)      LEA      SI,      1000H
          LEA      DI,      2000H
          MOV      CX,      100
          CLD
L1:  MOVSB
```

```

        LOOP    L1
        HLT
(3)      LEA     ST,      1000H
          LEA     DI,      2000H
          MOV     CX,      100
          CLD
          REP     MOVSB
          HLT

```

4．利用变址寄存器，编一个程序，把自 1000H 单元开始的 100 个数传送到自 1070H 开始的储存区中去。

解：

```

        LEA     SI,      1000H
        LEA     DI,      1070H
        MOV     CX,      100
        CLD
        REP     MOVSB
        HLT

```

5．要求同题 4，源地址为 2050H，目的地址为 2000H，数据块长度为 50。

解：

```

        LEA     SI,      2050H
          LEA     DI,      2000H
          MOV     CX,      50
          CLD
          REP     MOVSB

```

HLT

6．编一个程序，把自 1000H 单元开始的 100 个数传送至 1050H 开始的存储区中（注意：数据区有重叠）。

解：

```

        LEA     SI,      1000H
        LEA     DI,      1050H
        ADD     SI,      63H
        ADD     DI,      63H
        MOV     CX,      100
        STD
        REP     MOVSB
        HLT

```

7．在自 0500H 单元开始，存有 100 个数。要求把它们传送到 1000H 开始的存储区中，但在传送过程中要检查数的值，遇到第一个零就停止传送。

解：

```

        LEA     SI,      0500H
          LEA     DI,      1000H
          MOV     CX,      100
        N1: MOV     AL,      [SI]
          CMP     AL,      0

```



```

        JZ      N2
        MOV     [DI], AL
INC     SI
        INC     DI
        LOOP    N1
N2:     HLT

```

8 . 条件同题 7 , 但在传送过程中检查数的值 , 零不传送 , 不是零则传送到目的区。

解 :

```

        MOV     SI, 0500H
        MOV     DI, 1000H
        MOV     CX, 100
N1:     MOV     AL, [SI]
        CMP     AL, 0
        JZ      N2
        MOV     [DI], AL
N2;     INC     SI
        INC     DI
        LOOP    N1
        HLT

```

9 . 把在题 7 中指定的数据块中的正数 , 传送到自 1000H 开始的存储区。

```

MOV     SI, 0500H
        MOV     DI, 1000H
        MOV     CX, 100
N1:     MOV     AL, [SI]
        CMP     AL, 0
        JS      N2
        MOV     [DI], AL
        INC     DI
N2;     INC     SI
        LOOP    N1
        HLT

```

10 . 把在题 7 中指定的数据块中的正数 , 传送到自 1000H 开始的存储区 ; 而把其中的负数 , 传送到自 1100H 开始的存储区。且分别统计正数和负数的个数 , 分别存入 1200H 和 1201H 单元中。

```

MOV     SI, 0500H
        MOV     DI, 1000H
        MOV     BX, 1100H
        MOV     CX, 100H
        MOV     DX, 0
N1:     MOV     AL, [SI]
        CMP     AL, 0
        JS      N2
        MOV     [DI], AL
        INC     DI
        INC     SI

```

```

        INC     DI
        JMP     N3
N2:     MOV     [BX], AL
        INC     DH
        INC     SI
        INC     BX
N3:     LOOP    N1
        MOV     BX, 1200H
        MOV     [BX], DL
        MOV     [BX+1], DH
        HLT

```

11. 自 0500H 单元开始，有 10 个无符号数，编一个程序求这 10 个数的和（用 8 位数运算指令），把和放到 050A 及 050B 单元中（和用两个字节表示），且高位在 050B 单元。

解：

```

        LEA     BX, 0500H
        MOV     AX, 0
        MOV     CL, 10
N1:     MOV     DL, [BX]
        ADD     AL, DL
        ADC     AH, 0
        INC     BX
        DEC     CL
        JNZ     N1
        MOV     050AH, AX

```

12. 自 0200H 单元开始，有 100 个无符号数，编一个程序求这 100 个数的和（用 8 位数运算指令），把和放在 0264H 和 0265H 单元（和用两字节表示），且高位在 0265H 单元。

解：

```

        LEA     BX, 0200H
        MOV     AX, 0
        MOV     CL, 100
N1:     MOV     DL, [BX]
        ADD     AL, DL
        ADC     AH, 0
        INC     BX
        DBC     CL
        JNZ     N1
        MOV     0264H, AL

```

第五章

1.什么是存储器系统？微机中的存储器系统主要分为哪几类？设计目标是什么？

答：(1) 存储器系统：将两个或两个以上速度、容量和价格各不相同的存储器用软件、硬件或软硬件结合的方法连接起来成为一个系统。 (2) 微机中的存储器系统主要分为两种：由Cache和主存储器构成的Cache存储系统，有主存储器和磁盘构成的虚拟存储系统。 (3) 前者的目标是提高存取速度而后者的主要目标是增加存储容量。

2.内部存储器主要分为哪两类？它们的主要区别是什么？

答：(1) 分为ROM和RAM

(2) 主要区别是：ROM在正常工作时只能读出，不能写入，RAM则可读可写。断电后，ROM中的内容不会丢失，RAM中的内容会丢失。

3.为什么动态RAM需要定时刷新？

答：DRAM的存储元以电容来存储信息，由于存在漏电现象，电容中存储的电荷会逐渐泄露，从而使信息丢失或出现错误。因而需要对这些电容进行定时刷新。

4.CPU寻址内存的能力最基本的因素取决于？

答：地址总线的宽度。

5.内存地址从20000H~8BFFFFH共有多少字节？

答：共有 $8BFFFFH - 20000H + 1 = 6C000H$ 个字节

若采用6264芯片构成第7题中的内存空间，需要多少片6264？

答：每个6264芯片的容量位8KB，故需 $432/8=54$ 片。

7.设某微型机内存RAM区的容量为128KB，若有2164芯片构成这样的存储器需多少片2164？至少需要多少根地址线？其中多少根用于片内寻址？

答：需16片2164，至少需17根地址线，需要16根用于片内寻址。

8.什么是字扩展？什么是位扩展？用户自己购买内存条进行内存扩充，是在进行何种存储器扩展？

答：当存储芯片的容量小于所需内存容量时，需要用多个芯片构成满足容量要求的存储器，这就是字扩展。当存储芯片每个单元的字长小于所需内存单元字长时，需要用多个芯片构成满足字长要求的存储模块，这就是位扩展。用户自己购买内存条进行内存扩充，所做的是字扩展的工作。

第六章

1. I/O 接口的作用是什么？具有哪些主要功能？

I/O 接口是一电子电路（以 IC 芯片或接口板形式出现），其内有若干专用寄存器和相应的控制逻辑电路构成。它是 CPU 和 I/O 设备之间交换信息的媒介和桥梁。作用：对于主机，I/O 接口提供了外部设备的工作状态及数据；对于外部设备，I/O 接口记忆了主机送给外设的一切命令和数据，从而使主机与外设之间协调一致地工作。功能：进行端口地址译码设备选择、向 CPU 提供 I/O 设备的状态信息和进行命令译码、进行定时和相应时序控制、对传送数据提供缓冲，以消除计算机与外设的“定时”或数据处理速度上的差异。提供计算机与外设间有关信息格式的相容性变换。提供有关电气的适配、还可以中断方式实现 CPU 与外设之间信息的交换。

2. CPU 与 I/O 接口之间传送的信息有哪些？各表示什么含义？

CPU 与 I/O 接口电路之间传送的信息有数据信息，包括三种形式：数字量、模拟量、开关量；状态信息是外设通过接口往 CPU 传送的，如：“准备好”（READY）信号、“忙”（BUSY）信号；控制信息是 CPU 通过接口传送给外设的，如：外设的启动信号、停止信号就是常见的控制信息

3. CPU 与 I/O 设备数据传送的控制方式有哪几种？它们各有什么特点？

CPU 与 I/O 设备传输数据的控制方式通常有三种：程序方式、中断方式和 DMA 方式。程序方式：指用输入/输出指令，来控制信息传输的方式，是一种软件控制方式，根据程序控制的方法不同，又可以分为无条件传送方式和条件传送方式。

中断控制方式是利用向 CPU 发送中断的方式控制外围设备和 CPU 之间的数据传送。它的优点是大大提高了 CPU 的利用率且能支持多道程序和设备的并行操作。它的缺点是由于数据缓冲寄存器比较小，如果中断次数较多，仍然占用了大量 CPU 时间；在外围设备较多时，由于中断次数的急剧增加，可能造成 CPU 无法响应中断而出现中断丢失的现象；如果外围设备速度比较快，可能会出现 CPU 来不及从数据缓冲寄存器中取走数据而丢失数据的情况。

存储器直接存取方式（DMA）：外设利用专用的接口（DMA 控制器）直接与存储器进行高速数据传送，并不经过 CPU（CPU 不参与数据传送工作），总线控制权不在 CPU 处，而由 DMA 控制器控制。其特点是接口电路复杂，硬件开销大。大批量数据传送速度极快。适用于存储器与存储器之间、存储器与外设之间的大批量数据传送的场合。

4. 什么叫端口？I/O 端口的编址方式有哪两种？它们各有什么特点？

解：端口指输入/输出接口中的寄存器

I/O 端口有两种编址方式；

统一编址方式是将 I/O 端口与内存单元统一起来进行编号。该方式优点是不需要专门的 I/O 指令，对 I/O 端口操作的指令类型多；缺点是端口要占用部分存储器的地址空间，不容易区分是访问存储器还是外部设备。

独立编址的端口单独构成 I/O 地址空间，不占用存储器地址。优点是地址空间独立，控制电路和地址译码电路简单，采用专用的 I/O 指令，使得端口操作的指令在形式上与存储器操作指令有明显区别，程序容易阅读；缺点是指令类别少，一般只进行传送操作。

5. 8255A 的三种工作方式的特点是什么？各适合使用在什么场合？

方式 0：没有固定的用于应答式传送的联络信号线，CPU 可以采用无条件传送方式与 8255A 交换数据。

方式 1：有专用的中断请求和联络信号线，因此，方式 1 经常用于查询传送或中断传送方式。

方式 2：PA 口为双向选通输入/输出或双向应答式输入/输出。

6. 设 8255A 的 A 口工作于方式 1 输出，B 口工作于方式 0 输入，试编写初始化程序（设端口地址为 40H ~ 43H）。

解：

```
MOV DX, 43H
```

```
MOV AL, 10100010B(A2H)
```

```
OUT DX,AL
```

7. 使用 8255A 作为开关和 LED 指示灯的接口。要求 8255A 的 A 口连接 8 个开关，B 口连接 8 个 LED 指示灯，将 A 口的开关状态读入，然后送至 B 口控制指示灯亮、灭。试画出接口电路设计图，并编写程序实现之。

解：电路图参见教材 P193 图 7-14。A 口接入 8 个开关，B 口用 LED 替代数码管，共阴接法。设 8255 四个端口的地址为 FFE0H ~ FFE3H。

```
DATA SEGMENT
```

```
PORTA EQU 0FFE0H
```

```
PORTB EQU 0FFE1H
```

```
CONTR EQU 0FFE3H
```

```
DATA ENDS
```

```
CODE SEGMENT
```

```
ASSUME CS:CODE, DS:DATA
```

```
START: MOV AX, DATA
```

```
MOV DS, AX
```

```
MOV AL, 90H ; 初始化 8255,PA 为输入，PB 为输出
```

```
MOV DX, CONTR
```

```
OUT DX, AL
```

```
L0: MOV DX, PORTA ; 读 PA 口
```

```
IN AL, DX
```

```
NOT AL ; 按下开关灯亮
```

```
MOV DX, PORTB
```

```
OUT DX,AL ; 送 PB 口显示
```

```
MOV CX,1000 ; 延时
```

```
L1: DEC CX
```

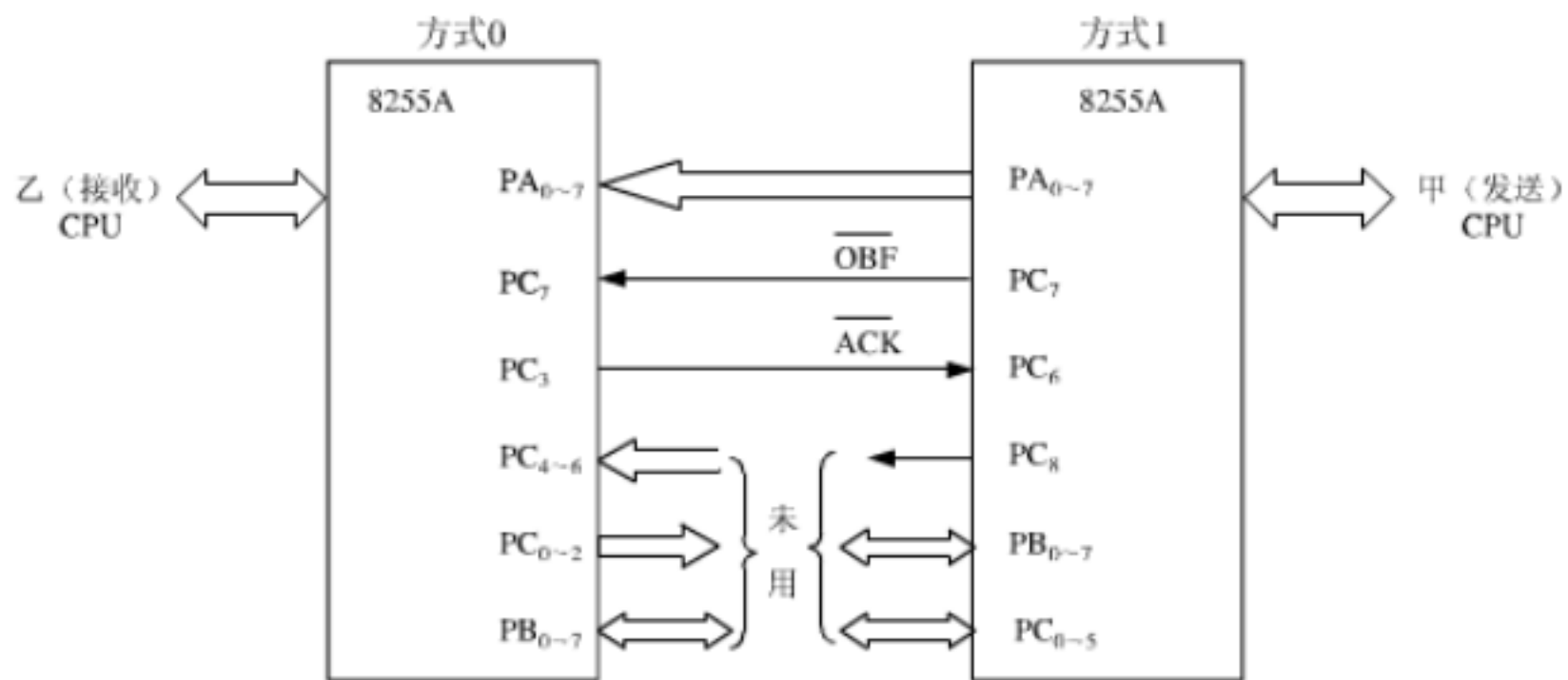
```
JNZ L1
```

```
JMP L0
```

```
CODE ENDS
```

```
END START
```

8. 在甲乙两台微机之间并行传送 1 KB 的数据，甲机发送，乙机接收。要求甲机一侧的 8255A 工作在方式 1，乙机一侧的 8255A 工作在方式 0，双机都采用查询方式传送数据。试画出通信接口电路图，并编写甲机的发送程序和乙机的接收程序。



1 甲机发送程序段

```

MOV    DX,303H                ;8255A 命令口
MOV    AL,10100000B           ;初始化工作方式字
OUT    DX,AL
MOV    AL,0DH                 ;置发送中断允许 INTEA=1
OUT    DX,AL                  ;PC6=1
MOV    SI,OFFSET BUFS         ;设置发送区的指针
MOV    CX,3FFH                ;发送字节数
MOV    DX,300H                ;向 A 口写第一个数，产生
                                ;第一个 OBF 信号
MOV    AL,[SI]                ;送给乙方，以便获取乙方的
                                ;ACK 信号
OUT    DX,AL
INC    SI                     ;内存地址加 1
DEC    CX                     ;传送字节数减 1

L:MOV   DX,302H                ;8255A 状态口
IN      AL,DX                  ;查发送中断请求 INTRA=1 ?
AND     AL,08H                 ;PC3=1
JZ      L                     ;若无中断请求，则等待
                                ;若有中断请求，则向 A 口写数
MOV     DX,300H                ;8255APA 口地址
MOV     AL,[SI]                ;从内存取数
OUT     DX,AL                  ;通过 A 口向乙机发送第二个数据
INC     SI                     ;内存地址加 1
DEC     CX                     ;字节数减 1
JNZ     L                     ;字节未完，继续
MOV     AH,4C00H               ;已完，退出
INT     21H                    ;返回 DOS
BUFS    DB 1024 个数据

```

2、乙机接受数据


```

MOV    DX,303H                ;8255A 命令口
MOV    AL,10100000B           ;初始化工作方式字
OUT    DX,AL

MOV    AL,00000111B           ;置  $\overline{ACK}$  =1(PC3=1)
OUT    DX,AL
MOV    SI,OFFSET BUFR         ;设置接收数据的指针
MOV    CX,3FFH                ;接收字节数
L1:MOV    DX,302H              ;8255APC 口
    IN    AL,DX                ;查甲机的 OBF=0?
                                ;(乙机的 PC7=0)
AND    AL,80H                  ;即查甲机是否有数据发来
JNZ     L1                     ;若无数据请求,则等待
                                ;若有数据,则从口读取
MOV    DX,300H                ;8255A PA 口地址
IN     AL,DX                   ;从 A 口读入数据
MOV    [DI],AL                 ;存入内存
MOV    DX,303H                ;产生 ACK 信号,并发给甲机
MOV    AL,00000110B           ;PC3 置 " 0 "
OUT    DX,AL
NOP
NOP
MOV    AL,00000111B           ;PC3 置 " 1 "
OUT    DX,AL
INC    DI                      ;内存地址加 1
DEC    CX                      ;字节数减 1
JNZ    L1                      ;字节未完,则继续
MOV    AX,4C00H                ;已完,退出
INT    21H                    ;返回 DOS
BUFR    DB 1024DU(?)

```

9. 8254 定时/计数器有哪几种工作方式？它们的主要特点是什么？
共有六种工作方式，分别对应与六种不同的用途。

解：

方式 0：计数结束

方式 1：可重复触发的单稳态触发器

方式 2：分频器

方式 3：方波发生器

方式 4：软件触发的选通信号发生器

方式 5：硬件触发的选通信号发生器

10. 利用 8254 的通道 1，产生 500 Hz 的方波信号。设输入时钟频率 $CKL1=2.5\text{ MHz}$ ，端口地址为 FFA0H ~ FFA3H，试编制初始化程序。

解：

```
MOV AL, 01110110B
```

```
MOV DX, 0FFA3H
```

```

OUT DX, AL
MOV AX, 5000 ;2.5MHz/500Hz=5000
MOV DX, 0FFA1H
OUT DX, AL
MOV AL, AH
OUT DX, AL

```

11. 某系统使用 8254 的通道 0 作为计数器，计满 1 000，向 CPU 发中断请求，试编写初始化程序 (端口地址自设)。

解:

设端口地址为 80H~83H

```

MOV AL, 00110000B
OUT 83H, AL
MOV AX, 1000
OUT 80H, AL
MOV AL, AH
OUT 80H, AL

```

12. 采用 8254 的通道 0 产生周期为 10 ms 的方波信号，设输入时钟的频率为 100kHz，8254 的端口地址为 38H ~ 3BH，试编写初始化程序。

解: MOV AL, 00110110B

```

OUT 3BH, AL
MOV AX, 1000 ; 10ms*100kHz=1000
OUT 38H, AL
MOV AL, AH
OUT 38H, AL

```

注意：计数值的高低字节的装入；

13. 串行通信有哪几种通信方式？它们的特点是什么？

有同步通信和异步通信两种

(1) 同步通信

同步通信是一种连续串行传送数据的通信方式，一次通信只传送一帧信息。这里的信息帧与异步通信中的字符帧不同，通常含有若干个数据字符。它们均由同步字符、数据字符和校验字符 (CRC) 组成。其中同步字符位于帧开头，用于确认数据字符的开始。数据字符在同步字符之后，个数没有限制，由所需传输的数据块长度来决定；校验字符有 1 到 2 个，用于接收端对接收到的字符序列进行正确性的校验。同步通信的缺点是要求发送时钟和接收时钟保持严格的同步。

(2) 异步通信

异步通信中，在异步通信中有两个比较重要的指标：字符帧格式和波特率。数据通常以字符或者字节为单位组成字符帧传送。字符帧由发送端逐帧发送，通过传输线被接收设备逐帧接收。发送端和接收端可以由各自的时钟来控制数据的发送和接收，这两个时钟源彼此独立，互不同步。接收端检测到传输线上发送过来的低电平逻辑 "0" (即字符帧起始位) 时，确定发送端已开始发送数据，每当接收端收到字符帧中的停止位时，就知道一帧字符已经发送完毕。

14. 异步串行通信的数据格式是什么？

异步通信数据帧的第一位是开始位，在通信线上没有数据传送时处于逻辑 "1" 状态。当发

送设备要发送一个字符数据时，首先发出一个逻辑“0”信号，这个逻辑低电平就是起始位。起始位通过通信线传向接收设备，当接收设备检测到这个逻辑低电平后，就开始准备接收数据位信号。因此，起始位所起的作用就是表示字符传送开始。当接收设备收到起始位后，紧接着就会收到数据位。数据位的个数可以是5,6,7或8位的数据。在字符数据传送过程中，数据位从最低位开始传输。数据发送完之后，可以发送奇偶校验位。奇偶校验位用于有限差错检测，通信双方在通信时需约定一致的奇偶校验方式。就数据传送而言，奇偶校验位是冗余位，但它表示数据的一种性质，这种性质用于检错，虽有限但很容易实现。在奇偶位或数据位之后发送的是停止位，可以是1位、1.5位或2位。停止位是一个字符数据的结束标志。在异步通信中，字符数据以图所示的格式一个一个地传送。在发送间隙，即空闲时，通信线路总是处于逻辑“1”状态，每个字符数据的传送均以逻辑“0”开始。

15. 什么是波特率？假设异步传输的一帧信息由1位起始位、7位数据位、1位校验位和1位停止位构成，传送的波特率为9600，则每秒钟能传输的字符个数是多少？

解：波特率表示每秒钟传送二进制的位数。每秒钟能传输的字符个数为 $9600/10=960$

16. 一个异步串行发送器，发送的字符格式为：1位起始位，7位数据位，1位奇偶校验位和2位停止位，若每秒发送100个字符，则其波特率为多少？

解：波特率为 $100 \times 11=1100$ 。

17. RS-232C的逻辑电平是如何定义的？它与计算机连接时，为什么要进行电平转换？

EIA-RS-232C对电器特性、逻辑电平和各种信号线功能都作了规定。

在TxD和RxD上：

逻辑1(MARK)=-3V ~ -15V

逻辑0(SPACE)=+3 ~ +15V

在RTS、CTS、DSR、DTR和DCD等控制线上：

信号有效（接通，ON状态，正电压）=+3V ~ +15V

信号无效（断开，OFF状态，负电压）=-3V ~ -15V

以上规定说明了RS-232C标准对逻辑电平的定义。对于数据（信息码）：逻辑“1”（传号）的电平低于-3V，逻辑“0”（空号）的电平高于+3V；对于控制信号；接通状态（ON）即信号有效的电平高于+3V，断开状态（OFF）即信号无效的电平低于-3V，也就是当传输电平的绝对值大于3V时，电路可以有效地检查出来，介于-3~+3V之间的电压无意义，低于-15V或高于+15V的电压也认为无意义，因此，实际工作时，应保证电平在±（3~15）V之间。

EIA RS-232C与TTL转换：EIA RS-232C是用正负电压来表示逻辑状态，与TTL以高低电平表示逻辑状态的规定不同。因此，为了能够同计算机接口或终端的TTL器件连接，必须在EIA RS-232C与TTL电路之间进行电平和逻辑关系的变换。实现这种变换的方法可用分立元件，也可用集成电路芯片。目前较为广泛地使用集成电路转换器件，如MC1488、SN75150芯片可完成TTL电平到EIA电平的转换，而MC1489、SN75154可实现EIA电平到TTL电平的转换。MAX232芯片可完成TTL EIA双向电平转换。

第七章

1. 在中断响应过程中，8086 往 8259A 发的两个 INTA# 信号分别起什么作用？

解：59A 接收到第一个 /INTA 有效的负脉冲后，将被响应的中断源置入 ISR（正在服务寄存器）的对应位，即 ISR 对应位置 1，同时把 IRR（中断请求寄存器）的对应位清 0；当 8259A 接收到第二个 /INTA 有效的负脉冲后，就将其中断类型号送入数据总线。

2. 8086 最多可有多少级中断？按照产生中断的方法分为哪两大类？

解：8086 最多可有 8 级中断。按照产生中断的方法分为硬件中断和软件中断两大类。

3. 非屏蔽中断有什么特点？可屏蔽中断有什么特点？分别用在什么场合？

解：非屏蔽中断不受 CPU 中断标志的影响，无论中断标志是允许中断或禁止中断，在非屏蔽中断引线上的中断请求，CPU 在当前指令执行完以后，都要影响此中断。可屏蔽中断则受中断标志影响，只是在开中断（允许中断）情况下，CPU 才响应可屏蔽中断。非屏蔽中断，使用在 CPU 必须要响应的场合，例如电源故障等

4. 什么叫中断向量？它放在哪里？对应于 ICH 的中断向量存放在哪里？如果 1CH 的中断处理子程序从 5110H:2030H 开始，则中断向量应怎样存放？

答：中断向量是用来提供中断入口地址的一个地址指针；

对应于 ICH 的中断向量存放在 $1CH \times 4 = 70H$ 开始的 4 个单元

如果 1CH 的中断处理子程序从 5110H:2030H 开始，则中断向量应如下存放：

0070H:30H

0071H:20H

0072H:10H

0073H:51H

5. 从 8086/8088 的中断向量表中可以看到，如果一个用户想定义某个中断，应该选择在什么范围？

解：从 8086/8088 的中断向量表中可以看出，由于系统占用了部分中断类型码，主要包括：

（1）专用中断：0~4，占中断向量表 000~013H。

（2）系统备用中断：5~31H

（3）用户使用中断：32H~0FFH

如果一个用户想定义一个中断，可以选择中断类型码 32H-FFH，其中断向量在中断向量表的 0C8H-01BFH。

6. 非屏蔽中断处理程序的入口地址怎样寻找？

解：非屏蔽对应类型 2，它位于中断向量表 0000:0008H-0000:000BH 处，4 个单元的值即为非屏蔽中断处理程序的入口地址：08H、09H 放偏移量，0AH、0BH 放段地址。

7. 叙述可屏蔽中断的响应过程，一个可屏蔽中断或者非屏蔽中断响应后，堆栈顶部四个单元中是什么内容？

解：当 CPU 在 INTR 引脚上接受一个高电平的中断请求信号并且当前的中断允许标志为 1，CPU 就会在当前指令执行完后开始响应外部的中断请求，具体如下：

- 1、从数据总线上读取外设送来的中断类型码，将其存入内部暂存器中；
- 2、将标志寄存器的值推入堆栈；
- 3、将标志寄存器中 IF 和 TF 清零；
- 4、将断点保护到堆栈中；
- 5、根据中断类型获取中断向量转入中断处理子程序；6、处理完后恢复现场。

响应后堆栈的顶部 4 个单元是 IP,CS。

8. 一个可屏蔽中断请求来到时，通常只要中断允许标志为 1，便可在执行完当前指令后响

应，在哪些情况下有例外？

解：如果发出中断请求信号时，正好碰到 CPU 执行封锁指令，由于 CPU 封锁指令和下一条指令合在一起看成一个整体，所以必须等到下一条指令执行完后才响应中断。如果是执行往寄存器传送数据指令，那一定要等下一条指令执行完后，才允许中断。无总线请求；CPU 执行完当前指令。

9. 在编写中断处理子程序时，为什么要在子程序中保护许多寄存器？

解：因为在用户程序运行时，会在寄存器中有中间结果，当在中断服务程序中要使用这些寄存器前要把这些寄存器的内容推至堆栈保存（称为保护现场）。在从中断服务程序返回至用户程序时把这些内容从堆栈恢复至寄存器中（称为恢复现场）。

10. 中断指令执行时，堆栈的内容有什么变化？中断处理子程序的入口地址是怎样得到的？

解：中断指令执行时，堆栈内容变化如下：标志寄存器被推入堆栈，且 SP 减 2，然后 CPU 将主程序的下一条指令地址即断点地址推入堆栈，即段值 CS 推入堆栈且 SP 减 2，偏移量推入堆栈，且 SP 减 2。中断处理子程序的入口地址即中断向量，对应的中断类型号的 4 倍即为中断向量地址，在该地址处的 4 个字节内容，高地址单元两个字节为中断入口地址的段地址，低地址单元两个字节为中断入口地址的偏移地址。

11. 中断返回指令 IRET 和普通子程序返回指令 RET 在执行时，具体操作内容有什么不同？

解：因为在响应中断时，CPU 保护标志寄存器和保护断点。所以，IRET 指令要恢复断点和标志寄存器。

12. 若在一个系统中有五个中断源，它们的优先权排列为：1、2、3、4、5，它们的中断服务程序的入口地址分别为：3000H、3020H、3050H、3080H、30A0H。编一个程序，使得当有中断请求 CPU 响应时，能用查询方式转至申请中断的优先权最高的源的中断服务程序。

13. 设置中断优先级的目的是什么？

解：若有多个外设同时请求中断，它们之间是有轻重缓急的，设置中断优先级就能使 CPU 按照轻重缓急的要求来响应中断。而且，设置了中断优先级可以使优先级高的中断比优先级低的中断优先处理，实现中断嵌套。

14. 可编程中断控制器 8259A 在中断处理时，协助 CPU 完成哪些功能？

解：在中断响应周期，8259A 可为 CPU 提供响应的中断类型码

15. 8259A 具有哪些中断操作功能？指出与这些功能相对应的控制字（ICW/OCW）的内容？

解：ICW1 的 D4 必须为 1。D0 确定是否送 ICW4，若根据选择 ICW4 的各位应为 0，则可令 D0 位（即 IC4）为 0，则不送 ICW4。D1 位 SNGL，规定系统中是单片 8259A 工作还是级连工作。D2 位 ADI，规定 CALL 地址的间隔，D2=1，则间隔为 4，这适用于建立一个转移指令表；D2=0，则间隔为 8。D3 位 LTIM，规定中断请求输入线的触发方式，D3=1 为电平触发方式，此时边沿检测逻辑断开；D3=0 则为边沿触发方式。解] ICW1 的 D4 必须为 1。D0 确定是否送 ICW4，若根据选择 ICW4 的各位应为 0，则可令 D0 位（即 IC4）为 0，则不送 ICW4。D1 位 SNGL，规定系统中是单片 8259A 工作还是级连工作。D2 位 ADI，规定 CALL 地址的间隔，D2=1，则间隔为 4，这适用于建立一个转移指令表；D2=0，则间隔为 8。D3 位 LTIM，规定中断请求输入线的触发方式，D3=1 为电平触发方式，此时边沿检测逻辑断开；D3=0 则为边沿触发方式。

16. 什么是中断响应周期？在中断响应中 8086CPU 和 8259A 一般完成哪些工作？

解：中断响应周期是从用户程序至中断服务程序的过渡周期。在第一个中断响应周期，8259A 确定中断的优先级；在第二个中断响应周期，8259A 送出中断向量，CPU 获取此向量转至中断服务程序的入口。

17. 8086 有哪几种中断？哪些是硬件中断？哪些是软件中断？

解：8086CPU 有两种中断：软件中断——由指令的执行所引起的；硬件中断——由外部（主

要是外设)的请求所引起的。(1)外部中断 8086 有两条外部中断请求：NMI —— non maskable interrupt (非屏蔽中断)和 INTR (可屏蔽中断)。(2)内部中断 8086 可以有几种产生内部中断的情况：DIV 或 IDIV 指令、INT 指令、INTO 指令和单步执行。

18. 什么是 8086 的中断向量？中断向量表是什么？ 8086 的中断向量表放在何处？

解：中断向量表是存放中断服务程序入口地址的表格。 8086 系统的中断向量表存放在存储器的最低端，占用 1024 个字节单元。（或中断向量表在存储器中占用的地址范围为 0000H~003FFH）

19. 8259A 的初始化命令字和操作命令字有什么差别？它们分别对应于编程结构中哪些内部寄存器？

解：8259A 的初始化命令字是使 8259A 在开始前处在正确的起始状态。初始化命令主要规定 8259A 的工作模式和中断向量。初始化命令字写至 ICW1 —— ICW4。

8259A 操作命令字主要用于规定 8259A 的工作方式，例如中断屏蔽、结束中断、优先权旋转和中断状态。工作命令字可在 8259A 已经初始化以后的任何时间写入。

20. 8259A 的中断屏蔽寄存器 IMR 和 8086/8088 的中断允许标志 IF 有什么差别？在中断响应过程中，它们怎样配合起来工作？

答：1.若 IMR 的某位为 0 则该位对应的引脚上的中断请求未加屏蔽，让它通过而进入中断优先级裁决器作裁决。若 IMR 的某位为 1 则屏蔽该位对应的引脚上的中断请求，不让他进入中断优先级裁决器。而 8086/8088CPU 的中断允许标志 IF 为 1 则允许 INTR 引脚进入的中断，IF 为 0 则屏蔽 INTR 引脚进入的中断。与 8259A 的 IMR 位为 0 为 1 正好相反。2.在中断响应过程中，IMR 用于对外设向 8259A 发中断申请的允许/屏蔽，而 CPU 的 IF 用于对 8259A 由 INT 向 CPU 的 INTR 引脚发中断申请的允许/屏蔽。

21. 8259A 有几种结束中断处理的方式？各自应用在什么场合？除了中断自动结束方式以外，其他情况下如果没有在中断处理程序中发中断结束命令，会出现什么问题？

解：8259A 结束中断处理的方式有：

一般 EOI 方式：只有在当前结束的中断总是尚未处理完的级别最高的中断时，才能使用这种结束方式。制定 EOI 方式：适合于在任何情况下使用。自动 EOI 方式：只有在一些以预定速率发生中断，且不会发生同级中断互相打断或低级中断打断高级中断的情况下，才使用自动 EOI 方式。

22. 8259A 引入中断请求的方式有哪几种？如果对 8259A 用查询方式引入中断请求，那会有什么特点？中断查询方式用在什么场合？

答：1.引入中断请求的方式有：边沿触发方式、电平触发方式、中断查询方式三种。

2.中断查询方式的特点：

设备仍然通过向 8259A 发中断请求信号要求 CPU 服务，但 8259A 不使用 INT 信号向 CPU 发中断请求信号。

CPU 内部的中断允许触发器复位，所以禁止了外部对 CPU 的中断请求。 7.2 7.3 7.4 7.5 7.6 7.7

CPU 要使用软件查询来确认中断源，从而实现对设备的中断服务。

23. 8259A 的初始化命令字有哪些？它们各自有什么含义？哪几个应写入奇地址？哪几个应写入偶地址？

答：1、8259A 的初始化命令字有 ICW1、ICW2、ICW3、ICW4 共四个。

2、ICW1 —— 芯片控制初始化命令字。 ICW2 —— 设置中断类型码的初始化命令字。 ICW3 —— 标志主片/从片的初始化命令字。 ICW4 —— 方式控制初始化命令字。

3、ICW2、ICW3、ICW4 必须写入奇地址端口中。

4、ICW1 必须写入偶地址端口中。

第八章

1. 8253 芯片有哪几个计数通道？每个计数通道可工作于哪几种工作方式？这些操作方式的主要特点是什么？

答：8253 内部包含 3 个完全相同的计数器 / 定时器通道，即 0~2 计数通道，对 3 个通道的操作完全是独立的。8253 的每个通道都有 6 种不同的工作方式。方式 0——计数结束中断方式：当对 8253 的任一通道写入控制字，并选定工作于方式 0 时，该通道的输出端 OUT 立即变为低电平。要使 8253 能够进行计数，门控信号 GATE 必须为高电平。经过 $n + 1$ 个脉冲后，计数器减为 0，这时 OUT 引脚由低电平变成高电平。OUT 引脚上的高电平信号，一直保持到对该计数器装入新的计数值，或设置新的工作方式为止。在计数的过程中，如果 GATE 变为低电平，则暂停减 1 计数，计数器保持 GATE 有效时的值不变，OUT 仍为低电平。待 GATE 回到高电平后，又继续往下计数。方式 1——可编程单稳态输出方式：当 CPU 用控制字设定某计数器工作于方式 1 时，该计数器的输出 OUT 立即变为高电平。GATE 出现一个上升沿后，在下一个时钟脉冲的下降沿，将 n 装入计数器的执行部件，同时，输出端 OUT 由高电平向低电平跳变。当计数器的值减为零时，输出端 OUT 产生由低到高的正跳变，在 OUT 引脚上得到一个 n 个时钟宽度的负单脉冲。在计数过程中，若 GATE 产生负跳变，不会影响计数过程的进行。但若在计数器回零前，GATE 又产生从低到高的正跳变，则 8253 又将初值 n 装入计数器执行部件，重新开始计数，其结果会使输出的单脉冲宽度加宽。方式 2——比率发生器：当对某一计数通道写入控制字，选定工作方式 2 时，OUT 端输出高电平。如果 GATE 为高电平，则在写入计数值后的下一个时钟脉冲时，将计数值装入执行部件，此后，计数器随着时钟脉冲的输入而递减计数。当计数值减为 1 时，OUT 端由高电平变为低电平，待计数器的值减为 0 时，OUT 引脚又回到高电平，即低电平的持续时间等于一个输入时钟周期。与此同时，还将计数初值重新装入计数器，开始一个新的计数过程，并由此循环计数。如果装入计数器的初值为 n ，那么在 OUT 引脚上，每隔 n 个时钟脉冲就产生一个负脉冲，其宽度与时钟脉冲的周期相同，频率为输入时钟脉冲频率的 n 分之一。在操作过程中，任何时候都可由 CPU 重新写入新的计数值，不影响当前计数过程的进行。当计数值减为 0 时，一个计数周期结束，8253 将按新写入的计数值进行计数。在计数过程中，当 GATE 变为低电平时，使 OUT 变为高电平，禁止计数；当 GATE 从低电平变为高电平，GATE 端产生上升沿，则在下一个时钟脉冲时，把预置的计数初值装入计数器，从初值开始递减计数，并循环进行。方式 3——方波发生器：方式 3 和方式 2 的工作相类似，但从输出端得到的是对称的方波或基本对称的矩形波。如果写入计数器的初值为偶数，则当 8253 进行计数时，每输入一个时钟脉冲，均使计数值减 2。计数值减为 0 时，OUT 输出引脚由高电平变成低电平，同时自动重新装入计数初值，继续进行计数。当计数值减为 0 时，OUT 引脚又回到高电平，同时再一次将计数初值装入计数器，开始新一轮循环计数；如果写入计数器的初值为奇数，则当输出端 OUT 为高电平时，第一个时钟脉冲使计数器减 1，以后每来一个时钟脉冲，都使计数器减 2，当计数值减为 0 时，输出端 OUT 由高电平变为低电平，同时自动重新装入计数初值继续进行计数。这时第一个时钟脉冲使计数器减 3，以后每个时钟脉冲都使计数器减 2，计数值减为 0 时，OUT 端又回到高电平，并重新装入计数初值后，开始新一轮循环计数。方式 4——软件触发选通：当对 8253 写入控制字，进入工作方式 4 后，OUT 端输出变为高电平，如果 GATE 为高电平，那么，写入计数初值后，在下一个时钟脉冲后沿将自动把计数初值装入执行部件，并开始计数。当计数值成为 0 时，OUT 端输出变低，经过一个时钟周期后，又回到高电平，形成一个负脉冲。若在计数过程中写入一个新的计数值，则在现行计数周期内不受影响，但当计数值回 0 后，将按新的计数初值进行计数，同样也只计一次。如果在计数的过程中 GATE 变为低电平，则停止计数，当 GATE 变为高电平

后，又重新将初值装入计数器，从初值开始计数，直至计数器的值减为 0 时，从 OUT 端输出一个负脉冲。方式 5——硬件触发选通：编程进入工作方式 5 后，OUT 端输出高电平。当装入计数值 n 后，GATE 引脚上输入一个从低到高的正跳变信号时，才能在下一个时钟脉冲后沿把计数初值装入执行部件，并开始减 1 计数。当计数器的值减为 0 时，输出端 OUT 产生一个宽度为一个时钟周期的负脉冲，然后 OUT 又回到高电平。计数器回 0 后，8253 又自动将计数值 n 装入执行部件，但并不开始计数，要等到 GATE 端输入正跳变后，才又开始减 1 计数。计数器在计数过程中，不受门控信号 GATE 电平的影响，但只要计数器未回 0，GATE 的上升沿却能多次触发计数器，使它重新从计数初值 n 开始计数，直到计数值减为 0 时，才输出一个负脉冲。如果在计数过程中写入新的计数值，但没有触发脉冲，则计数过程不受影响。当计数器的值减为 0 后，GATE 端又输入正跳变触发脉冲时，将按新写入的初值进行计数。

2. 8253 的最高工作频率是多少？8254 与 8253 的主要区别是什么？

答：8253 的最高计数频率能达到 2MHz。Intel 8254 是 8253 的增强型产品，它与 8253 的引脚兼容，功能几乎完全相同，不同之处在于以下两点：（1）8253 的最大输入时钟频率为 2MHz，而 8254 的最大输入时钟频率可高达 5MHz。（2）8254 有读回（Read-back）功能，可以同时锁存 1~3 个计数器的计数值及状态值，供 CPU 读取，而 8253 每次只能锁存和读取一个通道的计数器，且不能读取状态值。

对 8253 进行初始化编程分哪几步进行？

答：（1）写入控制字用输出指令向控制字寄存器写入一个控制字，以选定计数器通道，规定该计数器的工作方式和计数格式。写入控制字还起到复位作用，使输出端 OUT 变为规定的初始状态，并使计数器清 0。（2）写入计数初值用输出指令向选中的计数器端口地址中写入一个计数初值，初值设置时要符合控制字中有关格式规定。

4. 设 8253 的通道 0~2 和控制口的地址分别为 300H，302H，304H 和 306H，系统的时钟脉冲频率为 2MHz，要求：

- （1）通道 0 输出 1kHz 方波；
- （2）通道 1 输出 500Hz 序列负脉冲；
- （3）通道 2 输出单脉冲，宽度为 400μs。

试图画出硬件连线图，并编写各通道的初始化程序。

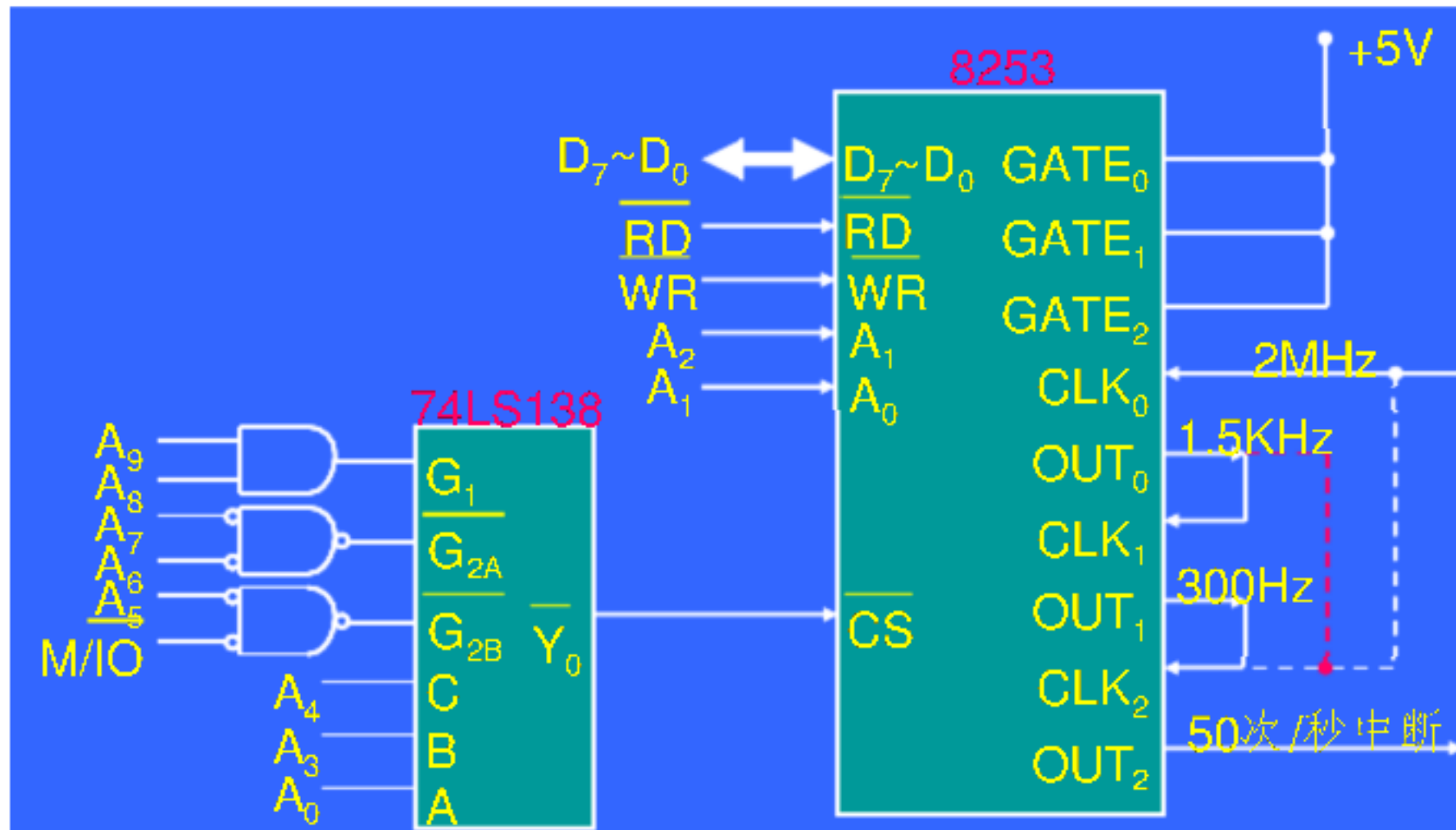
答：通道 0 工作在方式 3， $n_0 = 2\text{MHz}/1.5\text{KHz} = 1334$

通道 1 工作在方式 2， $n_1 = 1.5\text{KHz}/300\text{Hz} = 5$

通道 2 工作在方式 0，当 CLK2 = 2MHz 时， $n_2 = 2\text{MHz}/50\text{Hz} - 1 = 39999$ ；

当 CLK2 = OUT0 = 1.5KHz 时， $n_2 = 1.5\text{KHz}/50\text{Hz} - 1 = 29$

当 CLK2 = OUT1 = 300Hz 时， $n_2 = 300\text{Hz}/50\text{Hz} - 1 = 5$



初始化程序如下：

通道 0 初始化：

```
MOV DX, 306H
```

```
MOV AL, 00110111B(37H)      ; 方式 3, 先读 /写低 8 位 ,  
; 后读 /写低 8 位 , BCD 计数
```

```
OUT DX, AL
```

```
MOV DX, 300H
```

```
MOV AL, 34H                  ; 初值低 8 位
```

```
OUT DX, AL
```

```
MOV AL, 13H                  ; 初值高 8 位
```

```
OUT DX, AL
```

通道 1 初始化：

```
MOV DX, 306H
```

```
MOV AL, 01010101B(55H)      ; 方式 2, 只读 /写低 8 位 , BCD 计数
```

```
OUT DX, AL
```

```
MOV DX, 302H
```

```
MOV AL, 05H                  ; 初值
```

```
OUT DX, AL
```

通道 2 初始化：

```
MOV DX, 306H
```

```
MOV AL, 10010001B(91H)      ; 方式 0, 只读 /写低 8 位 , BCD 计数
```

```
OUT DX, AL
```

```
MOV DX, 304H
```

```
MOV AL, 29H( 或 05H) ; 初值
```

```
OUT DX, AL
```

5. 设 8253 的口地址为 40H~43H, 时钟频率 $f=5\text{MHz}$, 通道 2 接一个 LED 显示器。要求：LED 显示器点亮 4 秒钟后, 再熄灭 4 秒钟, 并不断重复该过程, 试编写 8253 的初始化程序。

第九章

1. DMA 控制器 8237A 有哪两种工作状态，其工作特点如何？

一种是系统总线的主控者，这是它工作的主方式。在取代 CPU 控制 DMA 传送时，它应提供存储器的地址和必要的读写控制信号，数据是在 I/O 设备与存储器之间通过数据总线直接传递；另一种是在成为主控者之前，必须由 CPU 对它编程以确定通道的选择、数据传送的模式、存储器区域首地址、传送总字节数等。在 DMA 传送之后，也有可能由 CPU 读取 DMA 控制器的状态。这时 8237A 如同一般 I/O 端口设备一样，是系统总线的从设备，这是 8237A 工作的从方式

2. 8237A 的当前地址寄存器、当前字节计数寄存器和基字节数寄存器各保存什么值？

当前地址寄存器：每个通道都有一个 16 位的当前地址寄存器，用于存放 DMA 传送的存储器地址。

当前字节计数寄存器：每个通道都有一个 16 位长的当前字节计数寄存器。它保存当前 DMA 传送的字节数。

基字节数寄存器：每个通道都有一个 16 位的基字节数寄存器，它用来存放对应通道当前字节计数器的初值。

3. 8237A 进行 DMA 数据传送时有几种传送方式？其特点是什么？

单字节传送方式、数据块传送方式、请求传送方式、级联方式

4. 8237A 有几种对其 DMA 通道屏蔽位操作的方法？

通道屏蔽字：可用之灵对屏蔽寄存器写入通道屏蔽字来对单个屏蔽位进行操作，使之置位或复位。

主屏蔽字：8237A 还允许使用主屏蔽命令设置通道的屏蔽触发器。

5. 设置 PC 的 8237A 通道 2 传送 1 KB 数据，请给其字节数寄存器编程。

传送 1KB (0400H 字节) 的字节数寄存器的初始值为 03 FFH。通道 2 字节数寄存器的端口地址：8237A 的基地址 +09H；清除字节指示器的端口地址：8237A 的基地址 +0CH。当 8237A 通道 2 传送 1KB 数据，对字节数寄存器初始化编程如下：

DMA EQU 00H；设 8237A 的基地址为 00H

输出清除字节指示器命令

OUT DMA+0CH,AL；发清除字节指示器命令（指向 16 位字节数寄存器的低 8 位）

字节数寄存器的初始化操作

MOV AL,0FFH；设置字节数寄存器低 8 位初始值

OUT DMA+09H,AL；写字节数寄存器低 8 位初始值

MOV AL,03H；设置字节数寄存器高 8 位初始值

OUT DMA+09H,AL；写字节数寄存器高 8 位初始值

6. 若 8237A 的端口基地址为 000H，要求通道 0 和通道 1 工作在单字节读传输，地址减 1 变化，无自动预置功能。通道 2 和通道 3 工作在数据块传输方式，地址加 1 变化，有自动预置功能。8237A 的 DACK 为高电平有效，DREQ 为低电平有效，用固定优先级方式启动 8237A 工作，试编写 8237A 的初始化程序。

DMA EQU 000H；8237A 的基地址为 00H

输出主清除命令

OUT DMA+0DH,AL；发总清命令

写入方式字：单字节读传输，地址减 1 变化，无自动预置功能，选择通道 0

MOV AL,01101000B；方式字

OUT DMA+0BH,AL；写入方式字

写入方式字：单字节读传输，地址减 1 变化，无自动预置功能，选择通道 1
 MOV AL , 01101001B; 方式字
 OUT DMA+0BH , AL; 写入方式字
 写入方式字：数据块传输方式，地址加 1 变化，有自动预置功能，选择通道 2
 MOV AL , 10010010; 方式字
 OUT DMA+0BH , AL; 写入方式字
 写入方式字：数据块传输方式，地址加 1 变化，有自动预置功能，选择通道 3
 MOV AL , 10010010B; 方式字
 OUT DMA+0BH , AL; 写入方式字
 写入命令字： DACK 为高电平有效， DREQ 为低电平有效，用固定优先级方式
 MOV AL , 11000000 B; 命令字
 OUT DMA+08H , AL; 写入 8237A

第十章

1. 什么是 A/D , D/A 转换器？

把模拟信号转换成数字信号量的器件为模 / 数转换器，简称 A/D 转换器。

2. A/D 和 D/A 转换器在微型计算机应用中起什么作用？

在微机应用中 A/D 转换器完成输入模拟量到数字量的转换，供微机采集数据。D/A 转换器完成微机输出数字量到模拟量的转换，实现微机控制。

3. D/A 转换器的主要参数有哪几种？参数反映了 D/A 转换器什么性能？

1.分辨率：该参数是描述 D/A 转换对输入变量变化的敏感程度。具体指 D/A 转换器能分辨的最小电压值。

2.转换时间：指数字量输入到模拟量输出达到稳定所需的时间。一般电流型 D/A 转换器在几秒到几百微秒之内；而电压型 D/A 转换器转换较慢，取决于运算放大器的响应时间。

3.转换精度：指 D/A 转换器实际输出与理论值之间的误差，一般采用数字量的最低有效位作为衡量单位。

4.线性度：当数字量变化时，D/A 转换器输出的模拟量按比例变化的程度。

4. A/D 转换器的主要参数有哪几种？参数反映了 A/D 转换器什么性能？

1.分辨率

分辨率是指输出数字量变化一个最小单位（最低位的变化），对应输入模拟量需要变化的量。输出位数越多，分辨率越高。通常以输出二进制码的位数表示分辨率。

2.相对精度

相对精度是指实际转换值偏离理想特性的误差。通常以数字量最低位所代表模拟输入值来衡量，如相对精度不超过 $\pm 1/2\text{LSB}$

3.转换时间

转换时间是指 A/D 转换器从接到转换命令起到输出稳定的数字量为止所需要的时间。它反映 A/D 转换器的转换速度。

此外，还有输入电压范围、功率损耗等。

5. 分辨率和精度有什么区别？

“精度”是用来描述物理量的准确程度，其反应的是测量值与真实值之间的误差，而“分辨率”是用来描述刻度划分的，其反应的是数值读取过程中所能读取的最小变化值。

6. DAC 0832 有哪几种工作方式？每种工作方式适用于什么场合？

DAC0832 有以下主要特点，1 满足 TTL 电平规范的逻辑输入，2 分辨率为 8 位，3 建立时间为 1 微秒，4 功耗 20mW，5 电流输出型 D/A 转换器；有三种工作方式，1 双缓冲方式，这种方式特别适用与要求同时输出多个模拟量的场合。2 单缓冲方式，这种方式下，只需执行一次写操作，即完成 D/A 转换，可以提高 DAC 的数据吞吐量。3 直通方式，这种方式可用于不采用微机的控制系统。

7. ADC 把模拟量信号转换为数字量信号，转换步骤是什么？转换过程用到什么电路？

模拟信号转换为数字信号，一般分为四个步骤进行，即取样、保持、量化和编码。前两个步骤在取样-保持电路中完成，后两步骤则在 ADC 中完成。

8. ADC 与微处理器接口的基本任务是什么？

向 ADC 转发启动转换信号；向 CPU 提供转换结束信号，把转换好的数据送入微处理器。

9. ADC 中的转换结束信号（EOC）起什么作用？

是转化结束信号，表示当前的 A/D 转换已经完成，可作为查询信息，也可接中断申请信号

10. 如果 0809 与微机接口采用中断方式，EOC 应如何与微处理器连接？程序又有什么改进？

采用直接与 CPU 的 INTR 脚连接或通过 8259A 接 CPU。

设 ADC0809 的端口号为 PORTAD，则当主程序中的指令 OUT PORTAD，AL 执行后，A/D 转换器开始转换，当转换结束时 EOC 端发一个高电平作为转换结束信号，此信号产生中断请求，CPU 响应中断后，调用中断处理程序，在中断处理程序中用 IN AL,PORTAD 取转换结果。

第十一章

1.什么是总线？总线是如何分类的？总线的主要参数有哪些？

总线是指计算机中多个部件之间公用的一组连线，由它构成系统插件间、插件的芯片间或系统间的标准信息通路。

微处理器芯片总线，内总线，外部总线

主要参数：总线的带宽，总线的位宽，总线的工作频率

2.ISA 总线的主要性能指标是什么？

ISA 总线的主要性能指标如下：

1).I/O 地址空间 0100H - - - 03FFH

2).24 位地址线可直接寻址的内容为 16 MB

3).8/16 位数据线

4).62+36 引脚

5).最大传输率 8 MB/S

6).DMA 通道功能

7).开放式总线结构，允许多个 CPU 共享系统资源。

3. PCI 总线的特点是什么？

(1) 线性突发传输，支持总线主控方式和同步操作 (2) 独立于处理器 (3) 即插即用 (4) 适用于各种机型 (5) 多总线共存 (6) 预留发展空间 (7) 数据线和地址线复用结构 (8) 节约线路空间，降低设计成本

4. PCI 总线信号可分哪几类？各类信号是多少条？

PCI 信号可分为必备和可选两大类。如果是主设备，必备信号为 49 条；如果是从设备，则必备信号是 47 条。可选的信号为 51 条，主要用于 64 位扩展、中断请求和高速缓存支持等。

5. 什么是 PCI/ISA 桥的负向译码？

PCI 总线上的总线事务在三个时钟周期后，没有得到任何 PCI 设备响应时（即总线请求的 PCI 总线地址不在这些设备的 BAR 空间中），PCI-to-ISA 桥将被动地接收这个数据请求。这个过程被称为 PCI 总线的负向译码。

第十二章

1.非编码键盘一般需要解决几个问题？识别被按键有哪几种方法，各有什么优缺点？

(1) 决定是否有键按下 (2) 如果有键按下，决定是哪一个键被按下 (3) 确定被按键的读数 (4) 反弹跳 (或称去抖动) (5) 不管一次按键持续时间多长，仅采样一个数据 (6) 处理同时按键

识别按键有两种方法：

(1) 行扫描法：先进行全扫描判断是否有键被按下，将所有行线置低电平，然后扫描全部列线，如果扫描的列值全是高电平，说明没有任何一个键被按下；如果读入的列值不是全“1”，说明有按键按下；如果所有列线全是高电平，说明第一行没有键被按下，接着扫描第二行，以此类推，直到找到被按下的键。

(2) 行反转法：行反转法：又称线反转法，利用可编程并行接口（如 8255A）来实现。将行线接到并行口先工作在输出方式，将列线接到另一个并行口先工作再输入方式。编程使 CPU 通过输出端口向各行线全部送低电平，然后读入列线的值。如果有某一个键被按下，则必有一条列线为低电平。然后通过编程对两个并行端口进行方式设置，并将刚才读到的列线值通过所连接的并行口再输出到列线，然后读取行线的值，那么闭合键所对应的行线比为低电平，这样当一个键被按下时，就可以读到一对唯一的列值和行值

(3) 行列扫描法：是通过计数译码使各行依次输出低电平。在扫描每一行时读列线，若读得的结果全为“1”，说明没有键按下，或者未扫描到闭合键；若某一列为低电平，说明有键按下，而且行号和列号仍相同，则键码确定无疑，即得到了闭合键的行列扫描码。

2. 与 PC 键盘发生关联的是哪两类键盘程序，他们各自的特点是什么？

计算机系统与键盘发生联系通过硬件中断 09H 或软件中断 16H。

特点：硬件中断 09H 是由按键动作引发的中断。在此中断中对所有键盘进行了扫描码定义。

软件中断 16H 是 BIOS 中断调用的一个功能。

3. 试简单说明 CRT 显示器的工作原理。

CRT 显示器主要由阴极射线管、视频放大驱动电路和同步描述电路等三部分组成。

工作原理：主要是由灯丝加热阴极，阴极发射电子，然后在加速极电场的作用下，经聚焦极聚成很细的光束，在阳极高压作用下，获得巨大的能量，以极高的速度去轰击荧光粉层。这些电子束轰击的目标是荧光屏上的三原色。为此，电子枪发射的电子束不是一束，而是三束，他们分别受电脑显卡 RGB 三个基色视频信号电压的控制，去轰击各自的荧光粉单元，从而在显示屏上显示出完整的图像。

4. 什么叫光栅扫描？在光栅扫描中，电子束受哪些信号的控制？

光栅扫描显示器显示图形时，电子束依照固定的扫描线和规定的扫描顺序进行扫描。

显示器上图像的显示实际上是在光栅扫描的过程中将图像信号分解成按时间分布的视频信号去控制电子束在各条光栅位置上点的亮度和色彩。

5. 在字符型显示器上，如果可以显示 40×80 个字符，显示缓存容量至少为多少？

$40 \times 80 = 3200 = 3.125\text{KB}$

6. 一个分辨率为 1024×768 的显示器，每个像素可以有 16 个灰度等级，那么相应的缓存容量应为多少？

显示器在图形模式下，显示缓存的最少容量与分辨率和颜色相关。若每个像素为 16 个灰度级，则每个像素应由 4 位表示，所以显示缓存的容量为 $1024 \times 768 \times 4/8 = 384\text{KB}$ 。

7. 常见的打印机接口有哪几种工作方式？说明并行打印机有哪些接口信号，怎样与主机进行连接，信号如何传递？

计算机主机和打印机之间的数据传输即可用并行方式，也可用串行方式。并行打印机通常采

用 Centronics 并行接口标准，该标准定义了 36 脚插头座。打印机与主机之间通过一根电缆线链接，点连线的一头插座为 36 芯，与打印机相联，另一头为 25 芯，与主机并行接口相联。36 条信号线按功能可分为：8 条数据线、9 条控制和状态线，15 条地线、1 条 +5V 电源线，其余 3 条不用。其中的 8 条数据线 DATA~DATA7、打印机接收数据的选通信号 $\overline{\text{STROBE}}$ 打印机回送给主机的忙信号 BUSY、打印机应答信号 $\overline{\text{ACK}}$ 以及地线是打印机和主机通信的基本信号线，它们是必不可少的，其它课时实际情况加以取舍。

第十三章

1、请说明 51 系列单片机 8051 与 8031 之间的区别？

解答：

它们的结构基本相同，其主要差别反映在存储器的配置上。8051 内部设有 4K 字节的掩模 ROM 程序存储器，8031 片内没有程序存储器。

8031 的特点：

8031 片内不带程序存储器 ROM，使用时用户需外接程序存储器和一片逻辑电路 373，外接的程序存储器多为 EPROM 的 2764 系列。用户若想对写入到 EPROM 中的程序进行修改，必须先用一种特殊的紫外线灯将其照射擦除，之后再可写入。写入到外接程序存储器的程序代码没有什么保密性可言。

8051 的特点

8051 片内有 4K ROM，无须外接外存储器和 373，更能体现“单片”的简练。但是你编的程序你无法烧写到其 ROM 中，只有将程序交芯片厂代你烧写，并是一次性的，今后你和芯片厂都不能改写其内容。

2、说明 MCS—51 系统单片机扩展 I/O 采用的编址方法。

解答：

MCS-51 使用统一编址的方式，每一接口芯片中的一个功能寄存器（端口）的地址就相当于一个 RAM 单元。

在 51 单片机中扩展的 I/O 口采用与片外数据存储器相同的寻址方法，所有扩展的 I/O 口，以及通过扩展 I/O 口连接的外设都与片外 RAM 统一编址，因此，对片外 I/O 口得输入/输出指令就是访问片外 RAM 的指令。I/O 寄存器与数据存储器单元同等对待，统一编址。不需要专门的 I/O 指令，直接使用访问数据存储器的指令进行 I/O 操作，简单、方便且功能强。

3、试叙述将 P1 口作为 I/O 口使用的注意事项。

解答：

P1 口的结构最简单，用途也单一，仅作为数据输入/输出端口使用。输出的信息有锁存，输入有读引脚和读锁存器之分。

注意：在 P1 口作为通用的 I/O 口使用时，在从 I/O 端口读入数据时，应该首先向相应的 I/O 口内部锁存器写：“1”。

举例：从 P1 口的低四位输入数据

MOV P1, #00001111b ;; 先给 P1 口低四位写 1

MOV A, P1 ;; 再读 P1 口的低四位

4、芯片 74LS244 能用作 8051 输出 I/O 口扩展吗？为什么？芯片 74LS377 能用作 8051 输入 I/O 口扩展吗？为什么？

解答：

输出 I/O 口扩展芯片必须具有数据跟随和数据锁存功能，而输入 I/O 口扩展芯片要具有高阻隔断功能和数据跟随功能，即输出时刻保持与输入相同，所以 244 和 377 不能混用。

5、在 MCS—51 系列单片机中，外部程序存储器和数据存储器共用 16 位地址，为什么不会发生数据冲突？

解答：

因为控制信号线的不同：

外扩的 RAM 芯片既能读出又能写入，所以通常都有读写控制引脚，记为 OE 和 WE。外扩 RAM 的读、写控制引脚分别与 MCS-51 的 RD 和 WR 引脚相连。

外扩的 EPROM 在正常使用中只能读出，不能写入，故 EPROM 芯片没有写入控制引脚，只

PSW 各位的定义如下：

Cy—高位进位标志。

AC—辅助进位标志。

F0—用户标志。

RS1、RS0—选择工作寄存器组位。

OV—溢出标志。

P—奇偶校验位。

10、为什么说单 MCS-51 系列单片机的存储器结构独立？这种独特结构有什么突出特点？

解答：

因为 MCS-51 单片机采用的是哈佛结构，这种结构的特点是程序存储器和数据存储器截然分开，各有自己的寻址系统、控制信号和功能。

这种结构的优点是能有效地使用较大而固定的程序和频繁地处理大量的数据或变量。

11、什么叫开中断？关中断？

解答：

中断是指单片机在运行别的代码是，系统可以通过中断方式打断连续的运行，先运行中断服务函数。

开中断就是指系统可以在连续运行是中断，去运行中断服务函数，关中断就是指关闭系统中断，不允许系统打断连续的运行。

12、MCS-51 系列单片机中有哪些无条件跳转指令？

解答：

无条件跳转指令包括长跳转指令、绝对跳转指令、短跳转指令、变址寻址跳转指令等 4 条。

13、89C51 单片机低功耗方式有几种？各有什么特点？

解答：

89C51 提供两种节电工作方式，即空闲（等待、待机）方式和掉电（停机）工作方式。

空闲方式和掉电方式是通过 SFR 中的 PCON（地址 87H）相应位置 1 而启动的。

当 CPU 执行完置 IDL=1（PCON.1）的指令后，系统进入空闲工作方式。这时，内部时钟不向 CPU 提供，而只供给中断、串行口、定时器部分。CPU 的内部状态维持，即包括堆栈指针 SP、程序计数器 PC、程序状态字 PSW、累加器 ACC 所有的内容保持不变，端口状态也保持不变。ALE 和 PSEN 保持逻辑高电平。

当 CPU 执行一条 PCON.1 位（PD）为 1 的指令后，系统进入掉电工作方式。在这种工作方式下，内部振荡器停止工作。由于没有振荡时钟，因此，所有的功能部件都停止工作。但内部 RAM 区和特殊功能寄存器的内容被保留，而端口的输出状态值都保存在对应的 SFR 中，ALE 和 PSEN 都为低电平。

14、请根据学过的指令知识编写 10s 延时程序。

解答：

设晶振频率为 12MHz，用软件编写程序为：

10ms=10000us=4T250 次 10 次

MOV R6, #10

LOOP:MOV R7, #250

LOOP1: NOP

NOP

DJNZ R7, LOOP1

DJNZ R6, LOOP

RET

15.请编写一个 16 位有符号数除法程序。

解答：

用 idiv 编写

```
mov al,0f7h      ; -9 送 al
cbw              ; 字节符号扩展位字
mov bl,2         ; 注意除数不可为立即数
idiv bl          ; 结果：商为 al=fch (-4)
                ; 余数： ah=ffh (-1)
```

第十四章

1、说明下列指令中各个操作数的寻址方式。

- (1)ADD A , 30H
- (2)MOV 30H , 20H
- (3)MOV A , @R0
- (4)MOVBX A , @R1
- (5)SJMP \$
- (6)MOV R0 , #20H
- (7)ORL C , 00H
- (8)MOV DPTR , #2000H
- (9)MOVC A , @A+PC
- (10)ANL 20H , #30H
- (11)ANL C , / 30H
- (12)CPL C

解答：

- (1) 源操作数为：直接寻址方式；目的操作数为：寄存器寻址方式。
- (2) 源操作数为：直接寻址方式；目的操作数为：直接寻址方式。
- (3) 源操作数为：寄存器间接寻址方式；目的操作数为：寄存器寻址方式。
- (4) 源操作数为：寄存器间接寻址方式；目的操作数为：寄存器寻址方式。
- (5) 操作数为：相对寻址方式。
- (6) 源操作数为：立即寻址方式；目的操作数为：寄存器寻址方式。
- (7) 源操作数为：位寻址方式；目的操作数为：位寻址方式。
- (8) 源操作数为：立即寻址方式；目的操作数为：寄存器寻址方式。
- (9) 源操作数为：变址寻址方式；目的操作数为：寄存器寻址方式。
- (10) 源操作数为：立即寻址方式；目的操作数为：直接寻址方式。
- (11) 源操作数为：位寻址方式；目的操作数为：位寻址方式。
- (12) 操作数为：位寻址方式。

2、写出下列程序的运行结果。

```
MOV SP,# 40H
MOV A , #20H
MOV B , #30H
PUSH A
PUSH B
POP A
POP B
```

(SP)= (A)= (B)=

解答：(SP) =40H, (A)=30H, (B)=20H

3、在设 A=0FH , R0=30H , 内部 RAM 的 (30H) =0AH 、 (31H) =0BH 、 (32H) =0CH , 请指出在执行下列程序段后上述各单元内容的变化。

```
MOV A , @R0
MOV @R0 , 32H
MOV 32H , A
MOV R0 , #31H
```

MOV A, @R0

解答：R0=31H, A=0BH, (30H)=0CH、(31H)=0BH、(32H)=0AH

4、请用数据传送指令来实现下列要求的数据传送。

- (1) R0 的内容输出到 R1
- (2) 内部 RAM20H 单元的内容传送到 A 中。
- (3) 外部 RAM30H 单元的内容送到 R0。
- (4) 外部 RAM30H 单元的内容送内部 RAM20H 单元。
- (5) 外部 RAM1000H 单元的内容送内部 RAM20H 单元。
- (6) 程序存储器 ROM2000H 单元的内容送 R1。
- (7) ROM2000H 单元的内容送到内部 RAM20H 单元。

解答：

- (1) MOV A,R0
MOV R1, A
- (2) MOV A,20H
- (3) MOV R0,#30H
- (4) MOV R1,#30H
MOVX A,@R1
MOV 20H,A
- (5) MOV DPTR,#1000H
MOVX A,@DPTR
MOV 20H,A
- (6) MOV DPTR,#2000H
CLR A
MOVC A,@A+DPTR
MOV R1,A
- (7) MOV DPTR,#2000H
CLR A
MOVC A,@A+DPTR
MOV 20H,A

5、设 A=5AH, R1=30H, (30H)=E0H, Cy=1。分析下列各指令执行后 A 的内容以及对标志位的影响（每条指令都以题中规定的原始数据参加操作）。

解答：

- (1) XCH A, R1; A=30H
- (2) XCH A, 30H; A=E0H
- (3) XCH A, @R1; A=E0H
- (4) XCHD A, @R1; A=50H
- (5) SWAP A; A=A5H
- (6) ADD A, R1; A=8AH
- (7) ADD A, 30H; A=30H
- (8) ADD A, #30H; A=8AH
- (9) ADDC A, 30H; A=31H
- (10) INC A; A=5BH
- (11) SUBB A, 30H; A=79H
- (12) SUBB A, #30H; A=29H

(13)DEC A ; A=59H

(14)RL A ; A=B2H

(15)RLC A ; A=B5H

(16)CPL A ; A=A5H

(17)CLR A ; A=00H

(18)ANL A, 30H ; A=40H

(19)ORL A, @R1 ; A=FAH

(20)CRL A, #30H ; A=BAH

6、在 MCS-51 单片机中有几种寻址方式 ?写出每一种寻址方式。

解答：有 7 种寻址方式，分别是立即寻址方式、直接寻址方式、寄存器寻址方式、寄存器间接寻址方式、变址寻址方式、相对寻址方式和位寻址方式。

7、什么是堆栈 ?

解答：堆栈是一个特殊的存储空间，有一端是固定的，称为栈底，另一端是活动的，称为栈顶，数据的存取是在栈顶进行的。数据的存取遵循先进后出，后进先出的原则。

8、在 MCS-51 单片机的指令系统中，有关堆栈操作的指令有哪些 ?

解答：PUSH 和 POP 指令

9、已知 CJNE 指令的一般格式为：CJNE 操作数 1，操作数 2 . Rel，简述怎样使用 CJNE 指令判断两个操作数的大小。

解答：若操作数 1 操作数 2，则进位标志 C=0，否则 C=1。

10、写出 AJMP 和 LJMP 指令的共同点和区别。

解答：共同点是都是无条件转移，转到指定的标号执行程序。

区别：

(1)转移范围不同，LJMP 指令的转移范围为 64KB，AJMP 指令的转移范围为 2KB。

(2)指令字节数不同，LJMP 指令是 3 字节指令，AJMP 指令是 2 字节指令。

(3)AJMP 指令的机器码特殊。

11、写出 ACALL 和 LCALL 指令的共同点和区别。

解答：共同点是都是调用子程序指令，转到指定的标号执行子程序。

区别：

(1)转移范围不同，LCALL 指令的调用范围为 64KB，ACALL 指令的调用范围为 2KB。

(2)指令字节数不同，LCALL 指令是 3 字节指令，ACALL 指令是 2 字节指令。

(3)ACALL 指令的机器码特殊。

12、RET 和 RETI 指令主要有哪些区别 ?

解答：区别：

(1)使用场合不同，RET 指令在子程序中使用，RETI 指令在中断服务程序中使用。

(2)从功能上看，RET 指令从堆栈中自动取出断点地址给 PC，使之返回到调用指令的下一个指令，继续执行主程序。而 RETI 指令除了具有 RET 指令的功能之外，还有清除中断响应时被设置的优先级状态、开放低级中断以及恢复中断逻辑等功能。

13、NOP 指令的用途是什么 ?

解答：常常用于程序的等待或时间的延迟。

14、设计一段程序实现功能：把片外 RAM2000H 单元开始的 40 个单元的数据传送到片外 RAM0000H 开始的 40 个单元中。

解答：

```
ORG      0000H
MOV      DPTR,    #2000H
```

```

MOV      R2,      #40H
MOV      R1 ,     #30H
LOOP1 :  MOVX     A,      @DPTR
MOV      @R1 ,    A
INC      R1
INC      DPTR
DJNZ     R2       LOOP1
MOV      DPTR,    #0000H
MOV      R2 ,     #40H
MOV      R1 ,     #30H
LOOP2 :  MOV      A,      @R1
MOVBX    @DPTR,   A
INC      R1
INC      DPTR
DJNZ     R2       LOOP2
SJMP     $
END

```

15、设计一段程序实现功能：把片内 RAM30H~3FH 单元中的 16 个数的存放顺序颠倒过来。
解答：

```

ORG      0000H
MOV      SP,      30H
MOV      R0 ,     #30H
MOV      R1 ,     #10H
LOOP1 :  MOV      A,      @R0
PUSH     ACC
INC      R0
DJNZ     R1 ,     LOOP1
MOV      R0,      #30H
MOV      R1 ,     #10H
LOOP2 :  POP      ACC
MOV      @R0 ,    A
INC      R0
DJNZ     R1 ,     LOOP2
AJMP     $
END

```

16、设计一段程序实现功能：找出从片内 RAM30H 开始的 16 个单元中最小值所在的单元，并将该单元的内容改成 0FFH。

解答：

```

ORG 000H
LJMP MAIN
ORG 100H
MAIN:

```

;;给片内 RAM30H 单元开始的 16 个单元赋值。

```

MOV R0,#30H

```

```

MOV R1,#(table1-table)
MOV DPTR,#TABLE
LOP:
CLR A
MOVC A,@A+DPTR
MOV @R0,A
INC R0
INC DPTR
DJNZ R1,LOP
;;查找片内 RAM30H 单元开始的 16 个单元的最小值，并存于 B 寄存器。
MOV R0,#30H
MOV R1,#(table1-table)
MOV B,#0FFH
LOP0:
MOV A,@R0
CJNE A,B,$+3
JNC LOP1
MOV B,@R0
LOP1:
INC R0
DJNZ R1,LOP0
;;读出片内 RAM30H 单元开始的 16 个单元的内容，并和最小值比较，相等则把该单元的值替换成 0ffh.
MOV R0,#30H
MOV R1,#(table1-table)
LOP2:
MOV A,@R0
CJNE A,B,LOP3
MOV @R0,#0FFH
LOP3:
INC R0
DJNZ R1,LOP2
SJMP $
TABLE:
DB 10H,25H,2H,00H,12H,33H,45H,00H,78H
DB 80H,0AAH,0BBH,12H,23H,34H,00H
TABLE1:
END

```

第十五章

1、51 单片机存储器为什么分为程序存储器空间和数据存储器空间？ I/O 接口扩展属于哪个空间？

解答：

单片机由于地址线的引脚数目少，分配的地址空间有限，故 ROM 和 RAM 的地址是分开编址的。为两个不同的逻辑空间，如一个程序存储器空间地址为 0000H~FFFFH。一个片外数据存储器空间地址为 0000H~FFFFH。访问时为了加以区分，采用不同的执行访问不同的存储器。

2、在单片机扩展系统中，程序存储器和数据存储器共用 16 位地址线和 8 位数据线，为什么两个存储空间不会发生冲突？

解答：

程序存储器和数据存储器虽然共用 16 位地址线和 8 位数据线，但由于数据存储器的读写信号由 RD (P3.7) 和 WR (P3.6) 信号控制，而程序存储器由读选通信号 PSEN 控制，因此虽然共处同一地址空间，但控制信号不同，所以不会发生总线冲突。

3、在简单并行 I/O 口的扩展中，用 74LS273 锁存器扩展输出口接 8 个 LED 发光二极管，用 74LS244 三态锁存器扩展输入口，接 8 个开关，P2.7 与 \overline{WR} 一起作为输出锁存器的选通控制

信号，用 P2.6 与 \overline{RD} 一起作为输入缓冲器的选通控制信号。请确定该扩展系统输入口、输出口的端口地址，并编写程序实现一个开关控制一个 LED 发光二极管的控制功能。

4、某单片机系统扩展 8255A 作 I/O 接口，硬件连接采用单片机的 A0、A1 分别接 8255A 的 A0、A1，单片机的 P2.6 接 8255A 的 \overline{CS} ，试画出单片机系统的硬件电路图，并确定 8255A 四个端口的地址。

5、用 8255A 扩展单片机的 I/O 口，要求 8255A 的 PA 口为输入口，PA 口的每一位接一个开关，PB 口为输出口，输出的每一位接一个发光二极管。当某个开关接“1”时，相应位上的发光二极管点亮（输出“0”亮），试编写相应的程序。设 8255A 的 PA 口地址为 7FFCH，PB 口地址为 7FFDH，PC 口地址为 7FFE H，控制口地址为 7FFFH。

解答：

DATA SEGMENT

PORTA EQU 7FFCH

PORTB EQU 7FFDH

CONTR EQU 7FFFH

DATA ENDS

CODE SEGMENT

ASSUME CS:CODE, DS:DATA

START: MOV AX, DATA

MOV DS, AX

MOV AL, 90H ;初始化 8255，PA 为输入，PB 为输出

MOV DX, CONTR

OUT DX, AL

L0: MOV DX, PORTA ;读 PA 口

IN AL, DX


```

        NOT AL                ; 按下开关灯亮
        MOV DX, PORTB
        OUT DX, AL            ;送 PB 口显示
        MOV CX, 1000          ;延时
L1 :    DEC CX
        JNZ L1
        JMP L0
CODE    ENDS
        END      START

```