

作业答案 0-1

绪论

1. 单片机是把组成微型计算机的各功能部件即（微处理器（CPU））、（存储器（ROM 和 RAM））、（总线）、（定时器/计数器）、（输入/输出接口（I/O 口））及（中断系统）等部件集成在一块芯片上的微型计算机。

2. 什么叫单片机？其主要特点有哪些？

解：

将微处理器（CPU）、存储器（存放程序或数据的 ROM 和 RAM）、总线、定时器/计数器、输入/输出接口（I/O 口）、中断系统和其他多种功能器件集成在一块芯片上的微型计算机，称为单片微型计算机，简称单片机。

单片机的特点：可靠性高、便于扩展、控制功能强、具有丰富的控制指令、低电压、低功耗、片内存储容量较小、集成度高、体积小、性价比高、应用广泛、易于产品化等。

第 1 章 MCS-51 单片机的结构与原理

15. MCS-51 系列单片机的引脚中有多少根 I/O 线？它们与单片机对外的地址总线和数据总线之间有什么关系？其地址总线和数据总线各有多少位？对外可寻址的地址空间有多大？

解：

MCS-51 系列单片机有 4 个 I/O 端口，每个端口都是 8 位双向口，共占 32 根引脚。每个端口都包括一个锁存器（即专用寄存器 P0~P3）、一个输入驱动器和输入缓冲器。通常把 4 个端口称为 P0~P3。在无片外扩展的存储器的系统中，这 4 个端口的每一位都可以作为双向通用 I/O 端口使用。在具有片外扩展存储器的系统中，P2 口作为高 8 位地址线，P0 口分时作为低 8 位地址线和双向数据总线。

MCS-51 系列单片机数据总线为 8 位，地址总线为 16 位，对外可寻址空间为 64KB。

25. 开机复位后，CPU 使用的是哪组工作寄存器（R₀-R_n）？它们的地址是什么？CPU 如何确定和改变当前工作寄存器组（R₀-R_n）？

解：

开机复位后，CPU 使用的是第 0 组工作寄存器。它们的地址是 00H—07H。CPU 通过对程序状态字 PSW 中 RS1 和 RS0 的设置来确定和改变当前工作寄存器组。

27. MCS-51 单片机的时钟周期、机器周期、指令周期是如何定义的？当主频为 12MHz 的时候，一个机器周期是多长时间？执行一条最长的指令需要多长时间？

解：

时钟周期又称为振荡周期，由单片机内部振荡电路 OSC 产生，定义为 OSC 时钟频率的倒数。时钟周期又称为节拍（用 P 表示）。时钟周期是时序中的最小单位。一个状态有两个节拍，

机器周期定义为实现特定功能所需的时间。MCS-51 的机器周期由 12 个时钟周期构成。

执行一条指令所需要的时间称为指令周期，指令周期是时序中的最大单位。由于机器执行不同指令所需的时间不同，因此不同指令所包含的机器周期数也不尽相同。MCS-51 的指令可能包括 1~4 个不等的机器周期。

当 MCS-51 的主频为 12MHz 时，一个机器周期为 1μs。执行一条指令需要的最长时间为 4μs。

作业答案 2

第 2 章 MCS-51 单片机指令系统与汇编语言程序设计

4. 假定累加器 A 中的内容为 30H，执行指令 1000H: MOV A, @A+PC 后，把程序存储器 (1031H) 单元的内容送入累加器 A 中。

7. 指出下列各指令在程序存储器中所占的字节数

- | | |
|----------------------|------|
| (1) MOV DPTR, #1234H | 3 字节 |
| (2) MOVBX A, @DPTR | 1 字节 |
| (3) LJMP LOOP | 3 字节 |
| (4) MOV R0, A | 1 字节 |
| (5) AJMP LOOP | 2 字节 |
| (6) MOV A, 30H | 2 字节 |
| (7) SJMP LOOP | 2 字节 |
| (8) MOV B, #30H | 2 字节 |

15. 设堆栈指针 (SP) = 60H，片内 RAM 中的 (30H) = 24H，(31H) = 10H。执行下列程序段后，61H、62H、30H、31H、DPTR 及 SP 中的内容将有何变化？

```

PUSH    30H
PUSH    31H
POP      DPL
POP      DPH
MOV     30H, #00H
MOV     31H, #0FFH

```

解：

结果为： 61H、62H 单元为堆栈单元，其内容已被弹出栈。 (30H)=00H
(31H)=0FFH (DPTR)=2410H (SP)=60H

17. 完成以下的数据传送过程

- (1) R1 的内容传送到 R0。
- (2) 片外 RAM 20H 单元的内容送 R0。
- (3) 片外 RAM 20H 单元的内容送片内 RAM 20H 单元。
- (4) 片外 RAM 1000H 单元的内容送片内 RAM 20H 单元。
- (5) ROM 2000H 单元的内容送 R0。
- (6) ROM 2000H 单元的内容送片内 RAM 20H 单元。
- (7) ROM 2000H 单元的内容送片外 RAM 20H 单元。

解：

- (1) R1 的内容传送到 R0 ；

```

MOV     A,    R1
MOV     R0,   A

```

- (2) 片外 RAM 20H 单元内容送 R0 ；

```

MOV     R1,    #20H
MOVBX   A,    @R1
MOV     R0,    A

```

- (3) 片外 RAM 20H 单元的内容送片内 RAM 20H 单元；

```

MOV     R0,    #20H
MOVBX   A,    @R0
MOV     20H,   A

```

- (4) 片外 RAM 1000H 单元的内容送片内 RAM 20H 单元；

```

MOV     DPTR,   #1000H
MOVBX   A,    @DPTR
MOV     20H,    A

```

(5) ROM 2000H 单元的内容送 R0 单元;

```
CLR    A
MOV     DPTR, #2000H
MOVC    A,@A+DPTR
MOV     R0,    A
```

(6) ROM 2000H 单元的内容送片内 RAM 20H 单元;

```
CLR    A
MOV     DPTR, #2000H
MOVC    A,@A+DPTR
MOV     20H,   A
```

(7) ROM 2000H 单元的内容送片外 RAM 20H 单元。

```
CLR    A
MOV     DPTR, #2000H
MOVC    A,@A+DPTR
MOV     R0,    #20H
MOVX    @R0,   A
```

19. 编程将片内 **RAM** 的 **40H~60H** 单元中内容送到片外 **RAM** 以 **3000H** 开始的单元中。
并将原片内 **RAM** 数据块区域全部清 **0**。

解:

```
ORG 0000H
LJMP MAIN
ORG 0030H
MAIN:  MOV SP, #60H
        MOV R7, #21H
        MOV R1, #40H
        MOV DPTR, #3000H
LOOP:  MOV A, @R1
        MOVX @DPTR, A
        MOV @R1, #00H
        INC R1
        INC DPTR
        DJNZ R7, LOOP
        SJMP $
END
```

23. 设有 **100** 个有符号数, 连续存放在片外 **RAM** 以 **2000H** 为首地址的存储区中, 试编程统计其中正数、负数、零的个数。

解:

```
ORG 0000H
LJMP MAIN
ORG 0030H
MAIN:  MOV R2, #100
        MOV R3, #00H
        MOV R4, #00H
        MOV R5, #00H
        MOV DPTR, #2000H
LOOP:  MOVX A, @DPTR
```

```

        CJNE A, #00H, NET1
        INC R3
        SJMP NET3
NET1:   JB ACC.7  NET2
        INC R4
        SJMP NET3
NET2:   INC R5
NET3:   INC DPTR
        DJNZ R2, LOOP
        SJMP $
END

```

24. 试编一查找程序，从外部RAM首地址为2000H、长度为9FH的数据块中找出第一个ASCII码A，将其地址送到20A0H和20A1H单元中。

解：

```

        ORG 0000H
        LJMP MAIN
        ORG 0030H
MAIN:   MOV SP, #60H
        MOV DPTR, #2000H
        MOV R7, #9FH
LP0:    MOVX A, @DPTR
        CJNE A, #41H, LP1
        MOV R2, DPL
        MOV R3, DPH
        MOV DPTR, #20A0H
        MOV A, R2
        MOVX @DPTR, A
        INC DPTR
        MOV A, R3
        MOVX @DPTR, A
        SJMP LP2
LP1:    INC DPTR
LP2:    DJNZ R7, LP0
        END

```

另一种方法：

```

        ORG 0000H
        LJMP START
        ORG 0030H
START:  MOV R0, #9FH
        MOV DPTR, #2000H
        MOV R1, #00H      ;R1 为 DPTR 地址的低八位
LOOP:   MOVX A, @DPTR
        CJNZ A, #65, LOOP1
        MOV DPTR, #20A0H
        MOV A, R0
        MOVX @DPTR, A      ; 把字符 'A' 的低八位地址放到 20A0 地址中
        INC DPTR
        MOV A, #20H
        MOVX @DPTR, A      ; 把字符 'A' 的高八位地址放到 20A1 地址中
        SJMP $
LOOP1:  INC R1
        INC DPTR

```

```

    DJNZ R0,LOOP
    SJMP $
    END

```

27. 编程实现将片外 **RAM** 中 **2400H~2450H** 单元中的数传送到 **2500H~2550H** 单元中。
解：

```

        ORG    0000H
        LJMP   MAIN
        ORG    0030H
MAIN:    MOV    R7,#51H
        MOV    DPTR,#2400H
LOOP:    MOVX   A,@DPTR
        MOV    DPH,#25H
        MOVX   @DPTR,A
        MOV    DPH,#24H
        INC    DPTR
        DJNZ   R7,LOOP
        SJMP   $
    END

```

另一种方法：

```

        ORG    0000H
        LJMP   MAIN
        ORG    0030H
MAIN:    MOV    30H,#24H
        MOV    31H,#00H
        MOV    32H,#25H
        MOV    33H,#00H
        MOV    R2,#51H
LOOP:    MOV    DPH,30H
        MOV    DPL,31H
        MOVX   A,@DPTR
        MOV    DPH,32H
        MOV    DPL,33H

        MOVX   @DPTR,A
        INC    31H
        INC    33H
        DJNZ   R2,LOOP
        SJMP   $
    END

```

第三章作业答案

6. 设 (TMOD) = 0A5H, 则定时器 T0 的状态是(方式 1 计数), 定时器 T1 的状态是(方式 2 定时)。或

设 (TMOD) = 0A5H, 则定时器 T0 的状态是(软件控制的 16 位计数器), 定时器 T1 的状态是(软硬件控制的可自动重装初值的 8 位定时器)。

27. 请写出 $\overline{\text{INT1}}$ 为低电平触发的中断系统初始化程序。

解:

$\overline{\text{INT1}}$ 为低电平触发的中断系统初始化程序如下:

```

ORG 0000H
LJMP MAIN
ORG 0013H
LJMP INTN1
ORG 0100H
MAIN: MOV SP, #60H

      SETB EA
      SETB EX1      ; 开  $\overline{\text{INT1}}$  中断
      CLR PX1       ; 令  $\overline{\text{INT1}}$  为低优先级
      CLR IT1       ; 令  $\overline{\text{INT1}}$  为电平触发
      SJMP $

INTN1: .....
      RETI
      END

```

用 MOV 指令实现:

```

MOV IE, #84H

ANL IP, #0FBH (或 ORL IP, #04H)
ANL TCON, #0FBH

```

28. 说明 MCS-51 单片机响应中断后, 中断服务的入口地址。

解:

中断服务程序的入口地址如下表:

中断源	中断矢量
外部中断 0 ($\overline{\text{INT0}}$)	0003H
定时器 T0 中断	000BH
外部中断 1 ($\overline{\text{INT1}}$)	0013H
定时器 T1 中断	001BH
串行口中断	0023H

36. 使用一个定时器, 如何通过软硬结合方法实现较长时间的定时?

解:

设定好定时器的定时时间, 采用中断方式用软件进行溢出次数累计, 从而得到较长的定时时间, 定时时间=定时器的定时时间×软件累计的溢出次数。

37. 利用定时器输出周期为 2 ms 的方波, 设单片机晶振频率为 6 MHz。试编程实现之。

解:

选用定时器/计数器 T0 作定时器, 工作在方式 1, 输出为 P1.0 引脚, 2 ms 的方波可

由 1 ms 的高低电平相间隔而成，因而只要每隔 1 ms 对 P1.0 取反一次即可得到这个方波。
初值的计算如下： $T0=12/(6\times10^6)=2\times10^{-6}S$

$$TC=M-T/T0=2^{16}-1\times10^{-3}/2\times10^{-6}=65536-500=65036=FE0CH$$

当定时器/计数器采用方式0时，初值为：

$$TC=M-T/T0=2^{13}-1\times10^{-3}/2\times10^{-6}=8192-500=7692=1E0CH,$$

则真正的16位计数初值为：F00CH（高8位，低5位）

程序如下：

```

ORG 0000H
LJMP START
ORG 000BH
LJMP T0INT          ; T0 中断入口
ORG 0030H
START: MOV SP, #60H    ; 初始化程序
      MOV TL0, #0CH    ; T0 赋初值
      MOV TH0, #0FEH
      MOV TMOD, #01H   ; 定时器/计数器 0 工作于方式 1
      SETB TR0         ; 启动 T0
      SETB ET0         ; 开 T0 中断
      SETB EA         ; 开总允许中断
      SJMP $

T0INT:                                PUSH ACC

      PUSH PSW
      CPL P1.0
      MOV TL0, #0CH
      MOV TH0, #0FEH
      SETB TR0         ; 启动 T0
      POP PSW
      POP ACC
      RETI
      END

```

当采用查询方式时：

兰色字部分应该为：

```

LP1: JBC TF0, LP2
      SJMP LP1
LP2:  CPL P1.0
      MOV TL0, #0CH
      MOV TH0, #0FEH
      SETB TR0
      SJMP LP1
      END

```

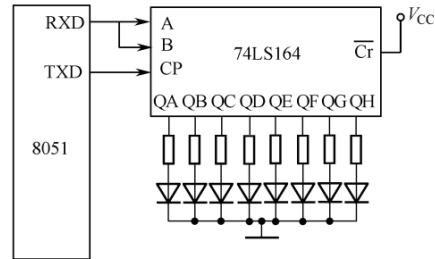
48. 某异步通信接口按方式 3 传送，已知其每分钟传送 3600 个字符，计算其传送波特率。

解：11（位）×3600（字符/分钟）=39600b/分钟=660b/s（方式 3 为每个字符 11 位）。

49. 利用 8051 串行口控制 8 位发光二极管工作，要求发光二极管每隔 1s 交替地亮、灭，画出电路图并编写程序。

解：

电路图如下：



程序如下：

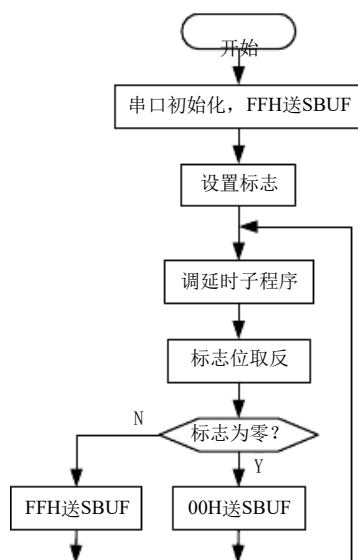
```

ORG    0000H
LJMP   START
ORG    0030H
START: MOV    SCON, #00H
      MOV    A, #0FFH
      CLR    TI
LP1:   MOV    SBUF, A
      JNB    TI, $
      CLR    TI
      ACALL  DELAY
      CPL    A
      SJMP   LP1
DELAY: MOV    R7, #8           ; 1s 延时程序
D1:    MOV    R6, #250
D2:    MOV    R5, #248
D3:    DJNZ   R5, D3
      NOP
      DJNZ   R6, D2
      DJNZ   R7, D1
      RET
      END

```

另一种方法：

主程序框图如下：



程序如下：

```

                ORG    0000H
                LJMP   START
                ORG    0030H
START:  MOV    SCON, #00H
        MOV    SBUF, #0FFH
        SETB   C
        MOV    00H, C
        AA:    ACALL  DELAY
        CPL    00H
        MOV    C, 00H
        JC     BB
        MOV    SBUF, #00H
        SJMP   CC
        BB:    MOV    SBUF, #0FFH
        CC:    AJMP   AA
DELAY:  MOV    R7, #8           ; 1s 延时程序
        D1:    MOV    R6, #250
        D2:    MOV    R5, #248
        D3:    DJNZ   R5, D3
             NOP
             DJNZ   R6, D2
             DJNZ   R7, D1
             RET
        END

```

第四章作业答案

16. MCS-51 单片机系统中，片外程序存储器和片外数据存储器共用**16**位地址线和**8**位数据线，为何不会产生冲突？

解：

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和写由 $\overline{RD}$ 和 $\overline{WR}$ 信号控制，而程序存储器由读选通信号 $\overline{PSEN}$ 控制，这些信号在逻辑上时序上不会产生冲突，并且，访问二者的指令也不同。因此，两者虽然共处于同一地址空间，

但由于控制信号和访问指令不同，所以不会发生总线冲突。

18. 某单片机应用系统，需扩展**2片 8KB 的 EPROM**和**2片 8KB 的 RAM**，采用地址译码法，画出硬件连接图，并指出各芯片的地址范围。

解：

硬件连接电路图如图 4.18 所示。各芯片的地址范围为：

2764 (1#)：0000H~1FFFFH

2764 (2#)：2000H~3FFFFH

6264 (1#)：4000H~5FFFFH

6264 (2#)：6000H~7FFFFH

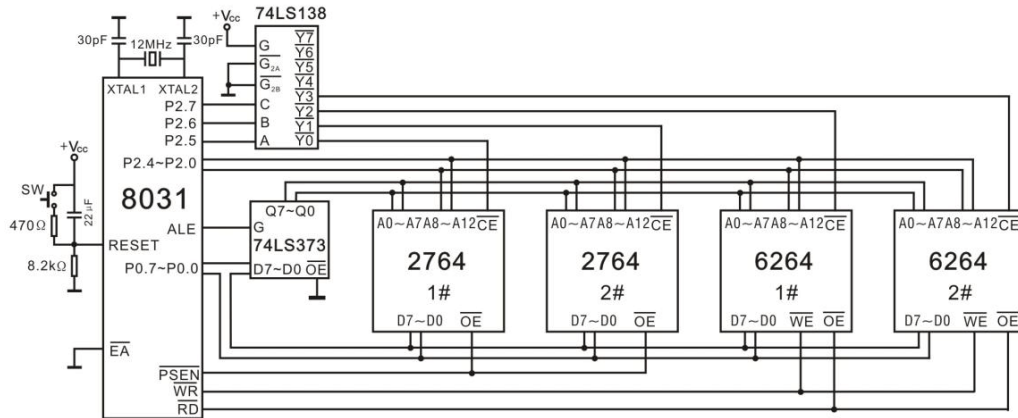


图 4.18 4.18 题硬件连接电路图

21. **8255A** 的端口地址为 **7F00H~7F03H**，试编程对 **8255A** 初始化，使 **A** 口按方式 **0** 输入，**B** 口按方式 **1** 输出。

解：

程序如下：

```
ORG    0000H

        LMP    START
START:  MOV    SP, #60H
        MOV    DPTR, #7F03H
        MOV    A, #10010100B
        MOVX   @DPTR, A
        SJMP   $
        END
```

25. 使用 **8255A** 或者 **8155** 的 **B** 端口驱动红色和绿色发光二极管各**4**只，且红、绿发光二极管轮流发光各**1s**，不断循环，试画出包括地址译码器、**8255A** 或 **8155** 与发光管部分的接口电路图，并编写控制程序。

解：

使用 8255A，电路连接图如图 4.25 所示。

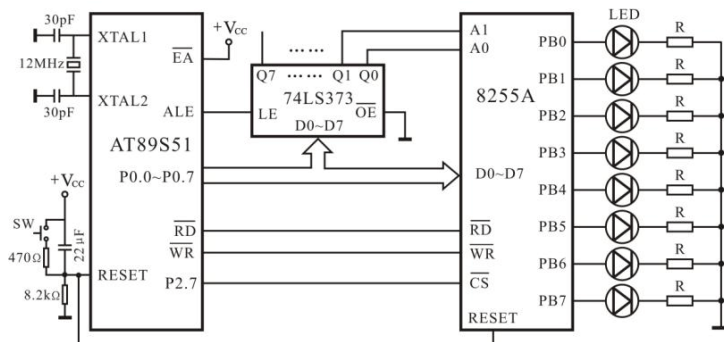


图 4.25 4.25 题硬件连接电路图

其中，PB0~PB3 接红色发光二极管，PB4~PB7 接绿色发光二极管。设 MCS-51 单片机主频为 12MHz。

程序如下：

```

ORG 0000H
LJMP START
ORG 0030H
START: MOV SP, #60H
      MOV DPTR, #7FFFH      ; 数据指针指向 8255A 控制口
      MOV A, #80H
      MOVX @DPTR, A         ; 工作方式字送 8255A 控制口
      MOV DPTR, #7FFDH      ; 数据指针指向 8255A 的 B 口
      MOV A, #0FH           ; 置红色发光二极管亮
LP1:  MOVX @DPTR, A         ; 置红色发光二极管亮
      LCALL DELAY           ; 调用 1S 延时子程序
      CPL A                 ; 置发光二极管亮反色
      SJMP LP1              ; 循环执行
DELAY: MOV R7, #8           ; 1s 延时子程序
      D1: MOV R6, #250
      D2: MOV R5, #248
      D3: NOP
          DJNZ R5, D3
          DJNZ R6, D2
          DJNZ R7, D1
      RET
      END

```

采用定时器 T0 方式 1 中断实现 1s 定时。

1s=50ms×20 次。

T0 方式 1 实现 50ms 定时，初值= $2^{16}-50\text{ms}/1\mu\text{s}=15536=3\text{CB}0\text{H}$

(1s=20ms×50 次，初值= $2^{16}-20\text{ms}/1\mu\text{s}=45536=\text{B1E}0\text{H}$)

```

ORG 0000H
LJMP START
ORG 000BH
LJMP TT0
ORG 0030H
START: MOV SP, #60H
      MOV DPTR, #7FFFH      ; 数据指针指向 8255A 控制口
      MOV A, #80H
      MOVX @DPTR, A         ; 工作方式字送 8255A 控制口
      MOV DPTR, #7FFDH      ; 数据指针指向 8255A 的 B 口
      MOV A, #0FH           ; 置红色发光二极管亮
      MOV R2, A
      MOVX @DPTR, A         ; 置红色发光二极管亮
      MOV 30H, #00H         ; 次数计数单元初值
      MOV TL0, #0B0H        ; T0 赋初值
      MOV TH0, #3CH

```

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/638115054055006045>