

得分

## 一、课程教学目标 1

### 简答题（共 50 分）

1. （8 分）举例说明汇编语言指令与机器指令的区别；介绍 8051 单片机指令系统在存储空间、执行时间方面的特点；解释伪指令 ORG、EQU、END、DB 的作用。

1. 计算机所能识别的语言只有机器语言，即由 0 和 1 构成的代码，非常难于记忆和识别。

汇编语言和机器语言都是直接对硬件进行操作，只不过指令采用了英文缩写的标识符，相比机器语言更容易识别和记忆。

举例：要做“10+20”的加法，用机器码指令编程：

0111010000001010 把 10 放到累加器 A 中

0010010000001000 A 加 20，结果仍放在 A 中

用汇编语言编程：以上两条指令可写成：

MOV A, #0AH

ADD A, #14H

8051 单片机指令系统在存储空间在 ROM 中，执行时间有单周期指令，双周期指令和三周期指令。

汇编起始伪指令 ORG，规定程序块或数据块存放的起始地址。

汇编结束伪指令 END，结束汇编

字节数据定义伪指令 DB，从标号指定的地址单元开始，将数据表中的字节数据按顺序依次存入。

字数据定义伪指令 DW，从标号指定的地址单元开始，将数据表中的字数据按从左到右的顺序依次存入。

赋值伪指令 EQU，将表达式的值定义为一个指定的符号名。

2. (8分) 什么是寻址方式? 说明寄存器寻址、直接寻址、寄存器间接寻址各自涉及的存储器空间。分别用三种寻址方式写出汇编语言指令, 实现将片内 RAM 地址为 06H 单元的内容送到累加器 A。

2. 寻址方式: CPU 执行指令时获取操作数的方法。

直接寻址方式适应于片内 RAM 的所有地址空间。

寄存器寻址适应于片内 00H~1FH 的 32 个字节, 用 R0, R1, ..., R7 表示。

寄存器间接寻址只能使用寄存器 R0 或 R1 作为地址指针, 来寻址内部 RAM (00H~FFH) 中的数据, 寄存器间接寻址也适用于访问外部 RAM。

第一种: MOV PSW, #00H

MOV A, R6

第二种: MOV A, 06H

第三种: MOV R0, #06H

MOV A, @R0

3. (8分) 单片机外部中断的触发方式有哪两种? CPU 响应中断的条件有哪些? 当 CPU 响应中断时, PC 的值有什么变化? 中断优先级有几级? 如何设置中断优先级?

3. 有电平触发和脉冲触发。电平方式是低电平有效。只有单片机

在中断要求中引入端  $INT0$  和  $INT1$  上采样到低电平时, 就激活外部中断。

脉冲方式则是脉冲的下降沿有效。这种方式下, 在两个相邻机器周期对中断请求引脚进行采样中, 如前一次为高, 后一次为低, 即为有效中断请求。

因此在这两种中断信号方式下, 中断请求信号的高电平状态和低电平状态都要至少维持一个周期以便电平变化能被单片机采样。

响应中断的条件:

(1) 有中断源发出中断请求

(2) 中断总允许位  $EA=1$ , 即 CPU 开中断

(3) 当前中断的中断源的中断允许位为 1, 即中断没有被屏蔽

(4) 无同级或更高级中断正在服务

(5) 当前指令周期已经结束

(6) 若现行指令为  $RETI$  或访问  $IE$  或  $IP$  指令时, 该指令被嵌套的下一条指令已执行完毕

当 CPU 响应中断时, PC 的值变为中断程序的入口地址。

中断优先级有 2 级, 中断源的级中断优先级寄存器  $IP$  来高低的,  $IP$  中某位设为 1, 相应的中断就是高优先级, 否则就是低优先级。

4. (8分) 如何设置定时器/计数器工作在计数方式? 当定时器/计数器工作在计数方式下时, 计数脉冲从哪个引脚输入? 计数脉冲的最高频率是多少? 当定时器/计数器工作在方式2时, 最大的计数值是多少?

4. 定时器/计数器由TMOD: 定时器/计数器工作方式控制中的C/T控制位来控制。

C/T: counter/timer 控制, 为0, timer 对内部机器周期脉冲计数。  
为1时, counter 对外部输入脉冲计数, 计一次数需要两个机器周期。

处于计数工作模式时, 加法计数器对于芯片端子 T0 (P3.4) 或

~~T0~~ T1 (P3.5) 上的输入脉冲计数;

定时器/计数器模式时, 外部输入的计数脉冲的最高频率为系统时钟频率的 $\frac{1}{2}$ 。

当定时器/计数器工作在方式2时, 最大的计数值为 $2^8 = 256$

5. (8分) 8051 的串口有几种工作方式? 波特率如何确定? 接收到的数据存放在哪个寄存器中? 串口接收完一个字节数据后, 标志位 RI 有什么变化?

共有四种工作方式

工作方式	SM0	SM1	波特率
0	0	0	$\frac{f_{osc}}{12}$
1	0	1	$\frac{2}{32} \times \frac{f_{osc}}{12} \times \frac{1}{2^K - 初值}$
2	1	0	$\frac{f_{osc}}{32}$ 或 $\frac{f_{osc}}{64}$
3	1	1	$\frac{2^{SMOD}}{32} \times \frac{f_{osc}}{12} \times \frac{1}{2^K - 初值}$

数据缓冲寄存器

接收完数据 RI=1



6. (10分) 说明独立按键和矩阵键盘的工作原理, 以及行扫描和线反转进行矩阵键盘识别的过程。

6.

(1) 独立式按键的每个按键都单独占用一根I/O口线与89C51单片机的引脚相连, 所有按键都会连接到公共地或同一个电源上, 每个按键能够独立工作而不互相影响, 多用于所需按键较少的场合。  
对独立式按键进行编程可以采用JNB或JB来查询哪一个按键被按下, 然后转向相应的处理程序。

(2) 矩阵式键盘:

矩阵式键盘的按键的触点接于由行、列母线构成的矩阵电路的交叉处, 每当一个按键按下时将接通相应的行列母线。

矩阵式键盘工作的时候首先判断是否有按键被按下, 当检测到有按键按下时将判断是哪个按键被按下并获取键码值, 然后根据不同的键码值转向不同的功能程序。

行扫描法和线反转法:

行扫描法:

(1) 规定行输出, 列输入。

(2) 将全“0”数据从行线上输出, 读取列线, 若全“1”则认为没有按键, 否则认为有按键。

(3) 在有按键情况下, 逐次使其中的行线输出“0”, 其它行线为“1”, 读取列线, 若读取的数据为全“1”, 则表明该行没有按键, 若读取的非全“1”, 根据行列线上的数据, 可以唯一确定哪一个键按下。

行反转法:

(1) 首先规定行线为输出, 列线为输入。

(2) 从行线上输出全“0”的数据, 从列线读取数据, 如果读取数据为全“1”, 则说明没有键按下, 否则认为有按键。

(3) 改变输入、输出端口, 将(2)中获得的数据从列线输出, 从行线读取数据, 根据两次读取的数据, 就可以唯一确定哪一个键按下。

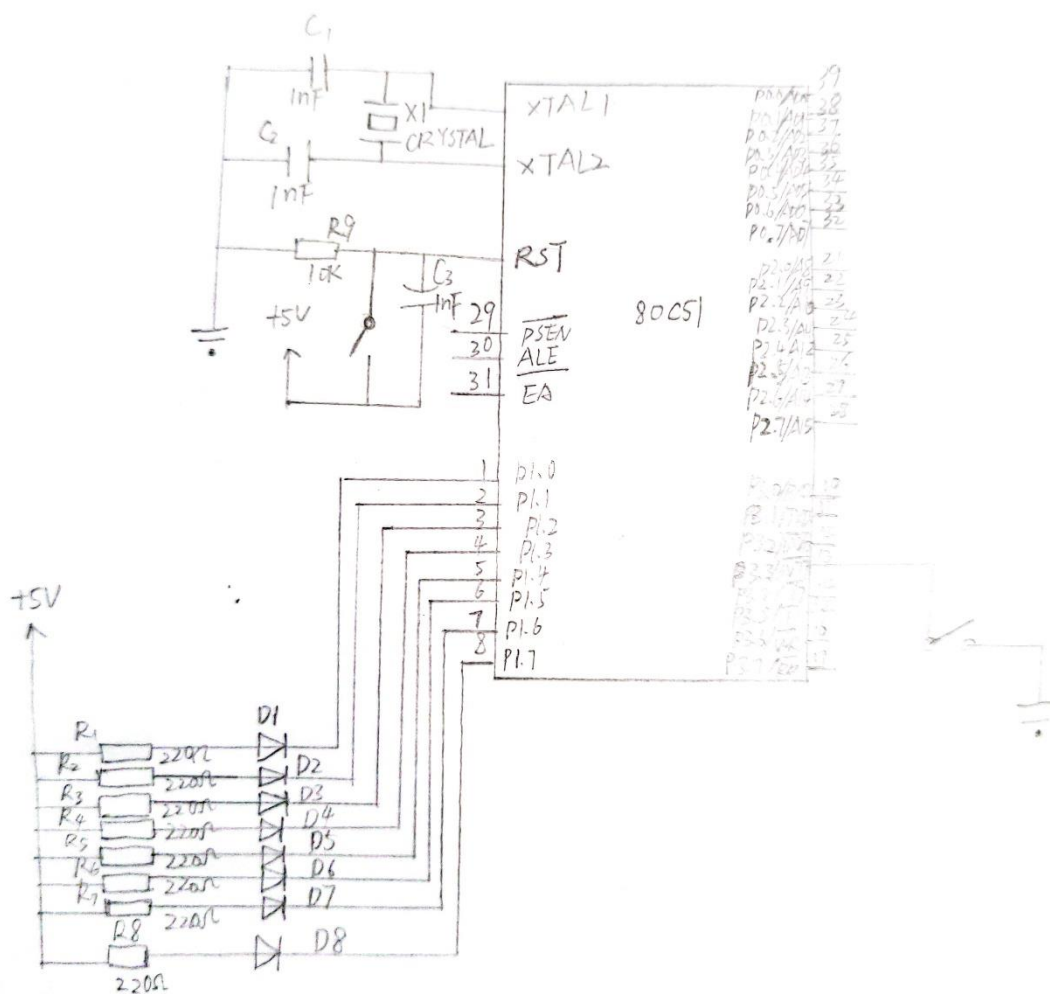
得分

## 二、课程教学目标 2 设计题（共 50 分）

7. (15 分) 回答问题:

7-1. 画出 8051 单片机的 P1 口连接 8 个发光二极管, P3.3 外接一个按键开关 K1 的硬件连接图。

7-2. 设 8051 的晶振频率为 12MHz, 对定时器/计数器 1 编程, 利用定时器中断, 控制 8 个 LED 中的高两位 LED 一直保持熄灭状态, 低 6 位 LED 闪烁, 亮灭间隔时间为 800ms。按键 K1 闭合可以切换定时器 1 的启动与停止。



```

7-2.  ORG 0000H
      LJMP MAIN.
      ORG 0013H
      LJMP EXINT1; 中断1子程序
      ORG 001BH
      LJMP INTT1.
      ORG 0100H
      MAIN: MOV SP, #60H
            MOV TMOD, #10H
            MOV TH1, #63H; 定时40ms
            MOV TL1, #0C0H
            MOV R2, #00H.
            SETB EA
            SETB ET1
            SETB TR1
            SETB IT1
            SETB EX
            MOV A, #0C0H
            MOV A, #0C0H
            SJMP $
      EXINT1:
            CPL TR1
            RETI

```

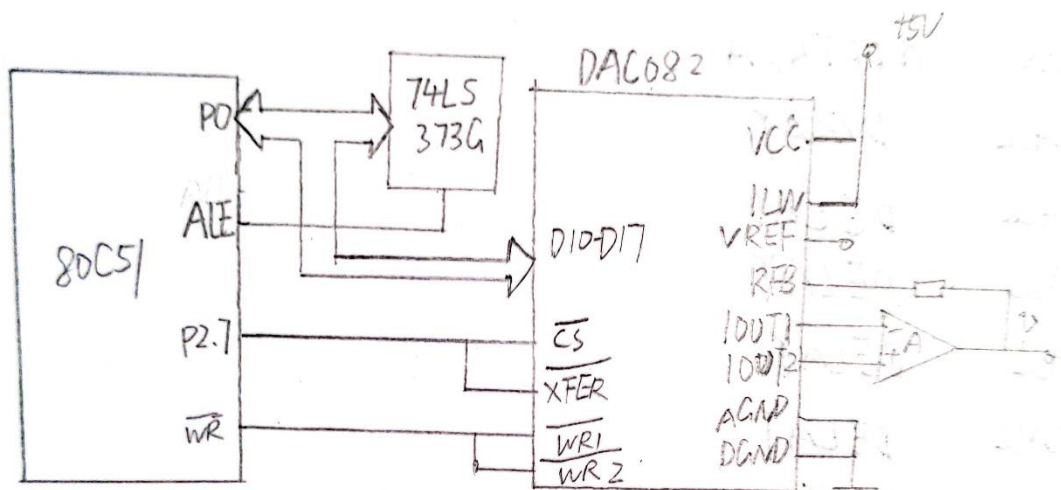
```

      INTT1:
            MOV TH1, #63H
            MOV TL1, #0C0H
            INC R2
            CJNE R2, #20, NEXT;
            中断20次, 为80ms
            MOV P1, A
            XRL A, #3FH
            MOV R2, #00H
      NEXT: RETI
      END

```



8. (10 分) 画出单片机与 DAC0832 单缓冲方式输入、单极性输出的连接电路图，并编写程序实现 DAC0832 输出占空比 30% 的方波，方波的电压范围是 2V~4V。



8. MOV DPTR, #7FFFH

LOOP: MOV A, #102.

MOVX @DPTR, A

ACALL DELAY

ACALL DELAY

ACALL DELAY

ACALL DELAY

ACALL DELAY

ACALL DELAY

ACALL DELAY

MOV A, #204

MOVX @DPTR, A

ACALL DELAY

ACALL DELAY

ACALL DELAY

SJMP LOOP

DELAY: MOV R7, #0FFFH

DJNZ R7, \$

RET

9. (15 分) 单片机外接两个共阳极数码管的电路如图 1 所示。回答问题:

9-1. 动态数码显示的原理是什么?

9-2. 编写汇编语言程序, 将片内 RAM 40H 单元中的两位**压缩 BCD** 码转换成**非压缩 BCD 码**, 显示在数码管上, 数码管动态扫描每位显示时间约为 2ms。(共阳极数码管 0-9 的段码: 0c0H, 0f9H, 0a4H, 0b0H, 99H, 92H, 82H, 0f8H, 80H, 90H)

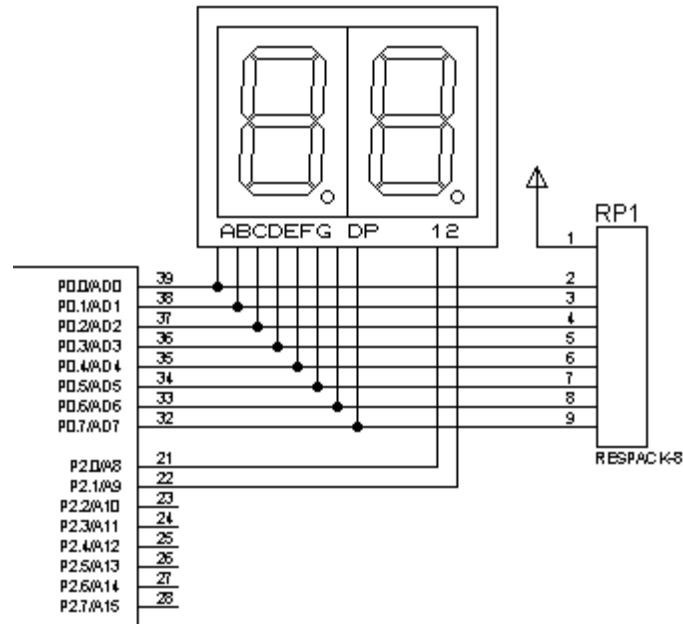


图 1 8051 与数码管接口电路

9-1. 段选码、位选码每送入一次后延时1ms, 因人眼的视觉暂留时间为100ms, 所以每段显示的间隔不要超过20ms, 并保持延时一段时间, 以造成视觉暂留效果, 给人看上去每个数码管都在亮

动态显示接口——所有数码管的段码线对应并联接在一个并行口上而每位数码管的公共端分别由一位I/O口线控制。

工作原理: 快速切换, 每时刻只有一组数码管导通工作。利用视觉暂留特性, 获得连续显示效果。

9-2.

```

ORG 0000H
LJMP MAIN
ORG 0003H
LJMP INT0SUB
ORG 0040H
MAIN: MOV SP, #30H
      MOV A, #81H
      MOV IE, A
      MOV A, #01H
      MOV TCON, A
      MOV R0, #00H
      MOV DPTR, #TABLE
      LCALL DISPLAY
      JMP MAIN
TABLE: DB 3FH, 06H, 5BH, 4FH, 66H, 6DH, 7DH, 07H, 7FH, 6FH
      MOV A, R0
      MOV B, #0AH
      DIV AB
      MOV A, B
      MOV C, A, @A+DPTR
      MOV P2, A
      JMP DISPLAY
INT0SUB:
      INC R0

```

```

      RET
DELAY30: MOV R7, #0FFH
DO: MOV R6, #0FFH
D00: MOV R5, #0FFH
D1: DJNZ R5, D1
D10: DJNZ R6, D10
DJNZ R7, D0
      RET
END.

```

10. (10 分) ADC0809 与单片机的接口如图 2 所示。回答问题:

10-1. START、EOC 信号的作用是什么?

10-2. 编写程序采集通道 0 的 A/D 转换结果, 以查询方式连续采集 32 个数据, 存放到片内 RAM 20H 开始的单元中。

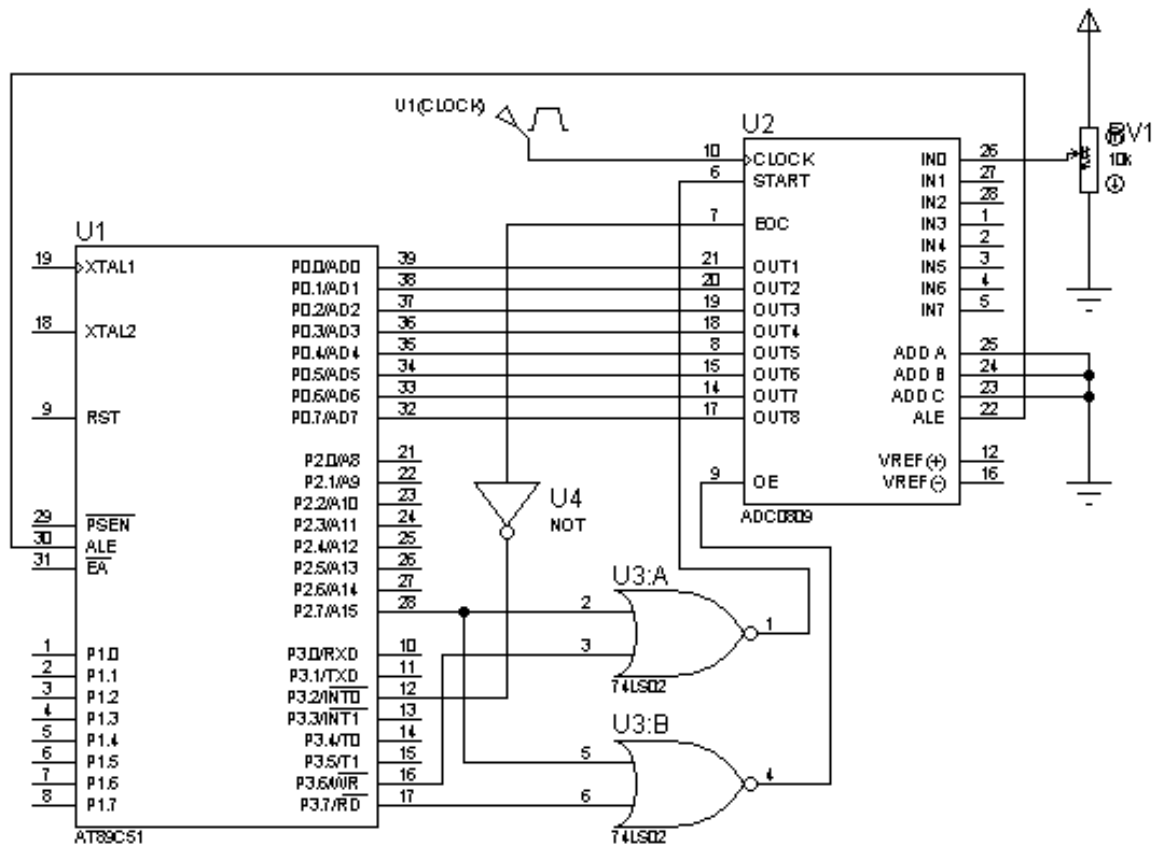


图 2 8051 与 ADC0809 接口电路

10-1. START 启动信号, 加上正脉冲后, A/D 转换开始进行。

EOC. 转换结束信号, 是芯片的输出信号。

转换开始后, EOC 信号变低; 转换结束时, EOC 返回高电平。

这个信号可以作为 A/D 转换器的状态信号来查询, 也可以直接用作中断请求信号。

10-2:

ADC: ORG 0100H.

MOV R0, #20H

MOV R2, #32

MOV DPTR, #0000H

LOOP: MOVX @DPTR, A

JNB P3.2, \$

MOVX A, @DPTR

MOV @R0, A

INC R0

DJNZ R2, LOOP

RET.