

2013 — 2014 学年第 一 学期考试试卷 A

考试科目: 单片机原理

考试班级: 11 电气信息类

课程编码:

考试方式:

姓名

潘晓霞

装

班级

11 电气信息类

学号

201110013001

订

得分	评卷人
11	8

题目	一	二	三	四	五	总分
得分	11	8	6	22.5	31	78.5
				25	53.5	

### 一、 填空题 (每空 1 分, 共 15 分)

1. 已知  $A=0C9H$ ,  $B=8DH$ ,  $CY=1$ , 执行  $SUBB A, B$  指令后  $A=$  3BH。
2. MCS-51 访问内部 RAM 使用 MOV 指令, 访问外部 RAM 使用 MOVX 指令, 访问外部 ROM 使用 MOVC 指令, 访问内部 ROM 使用 STMP vel 指令。
3. 单片机系统复位后, 寄存器  $R_5$  的地址是 05H,  $SP$  寄存器是 07H。
4. MCS-51 单片机地址线的宽度为 64kB, 数据线的宽度为 256B。
5. MCS-51 系列单片机中唯一的一个可供用户使用的 16 位寄存器是 PTR。
6. 单片机 80C51 芯片内部的 ROM 容量是 256B。

线



得分	评卷人
8	李

7. 程序计数器 PC 的内容是 下一条指令地址
8. 单片机 80C51 的定时器在工作方式 1 时的最大计数值是 65536。
9. 共阴极数码管，显示数字“1”的段码是 06H
10. 一个机器周期等于 12 倍振荡周期。

## 二、 选择题 (每小题 1 分, 共 10 分)

1. 在 MCS-51 中, 需外加电路实现中断撤除的是 (B)。-1  
A、定时中断    B、脉冲方式的外部中断    C、串行中断    D、电平方式的外部中断
2. 执行中断返回指令, 要从堆栈弹出断点地址, 以便去执行被中断了的主程序。从堆栈弹出的断点地址送给 (D)。  
A、A    B、CY    C、PC    D、DPTR
3. 单片机 8031 芯片的引脚 (A) 具有外部程序存储器读选通控制功能。  
A、 $\overline{\text{PSEN}}$     B、ALE    C、 $\overline{\text{EA}}$     D、RST
4. 8031 单片机的时钟电路中, 若晶体振荡器频率为 4MHz, 则执行一条双周期指令的时间为 (B)。  
A、3 $\mu\text{s}$     B、6 $\mu\text{s}$     C、12 $\mu\text{s}$     D、2 $\mu\text{s}$
5. MCS-51 系列单片机串行通信工作方式 1 是 (C) 位一帧的异步串行通信方式。  
A、8    B、11    C、10    D、9
6. 在定时器工作方式 0 下, 如果系统晶振频率为 3MHz, 则最大定



时时间为 A。

A、 $2^{13} \times 4\mu s$

B、 $2^{16} \times 2\mu s$

C、 $2^{16} \times 4\mu s$

D、 $2^{14} \times 4\mu s$

7. 要使 MCS-51 系列单片机系统复位, RST 端至少要保持 8 个机器周期为高。

A、4 B、2 C、3 D、1

8. 80C51 单片机芯片的 CPU 字长为 8 位。

A、4 B、8 C、16 D、32

9. 下列叙述中正确的是 D。

A、80C51 中共有 5 个中断源, 因此在芯片上相应的有 5 个中断请求输入引脚。

B、特殊的存取规则使得堆栈已经不是数据存储区的一部分。

C、可以把 PC 看成是数据存储空间的地址指针。

D、CPU 中反映程序运行状态和运算结果特征的寄存器是 PSW。

10. 80C51 单片机中, 设置堆栈指针 SP 为 37H 后就发生子程序调用, 这时 SP 的值变为 38H。

A、37H B、38H C、39H D、3AH

得分	评卷人
6	李

### 三、根据程序写结果 (每空 2 分, 共 10 分)

1. 执行以下程序段后,  $A =$  00H,  $CY =$  1

MOV A, #7AH

MOV R0, #30H

MOV 30H, #0A5H

SETB C

SUBB A, @R0



2. 执行以下程序段后,  $CY = (0)$   
 MOV 24H, #10H  
 MOV C, 24H

- 2

3. 执行以下程序段后,  $30H = (88H)$   $40H = (79H)$

MOV 30H, #99H

MOV 40H, #88H

PUSH 30H

PUSH 40H

POP 30H

POP 40H

得分	评卷人
22.5	2016

#### 四、简答题 (每小题 5 分, 共 25 分)

1. 简述指令 LCALL addr16 与 LJMP addr16 的功能, 并说明这两条指令有什么区别?

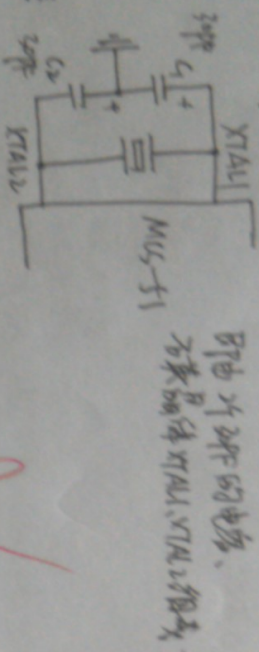
答: LCALL addr16 指令是子程序调用指令, 它的寻址范围是整个程序空间共 64K。  
 LJMP addr16 是长跳转指令, 它的寻址范围也是整个程序空间共 64K。

区别: 虽然 LCALL 和 LJMP 的寻址范围一样, 但 LCALL 是子程序调用, 有栈保护, 有返回; 而 LJMP 是长跳转, 以后才返回。



2. 简述 MCS-51 系列单片机的时钟电路是如何构成的？时钟电路确定后如何确定其时钟周期、机器周期？

答：时钟电路的电路图如下：



石英振荡器周期若为  $T$ ，则时钟周期  $T_c = 2T$   
机器周期  $T_m = 12T$

3. 简述 MCS-51 单片机定时器/计数器 0 如何实现定时功能和计数功能？

答：① 定时功能由信号由单片机内部的电路产生脉冲，通过寄存器所设置，选择其工作方式，设置定时的初值，然后让定时器开始运行。当运行到一定时间后，T0 溢出，此时必须下硬件清零，即可实现定时。

② 计数功能的脉冲信号是由外部的脉冲产生，其他操作基本同定时功能，但设置 M0D 的时候需设置为 C/T 置 1。让计数器开始运行，当运行到一定时间后，T0 溢出，再设置 ACC 让其溢出清零，即可实现计数功能。

4. 80C51 单片机 5 个中断源的中断入口地址分别是多少？在使用中断时为什么要在中断入口地址处加一条无条件跳转指令？

答：

外部中断 $\overline{INT0}$	0003H
定时/计数器中断 T0	000BH
外部中断 $\overline{INT1}$	0013H
定时/计数器中断 T1	001BH
串行中断	0023H

在单片机中加入一条无条件跳转指令是为了在遇到一定的时间作用，以确保中断系统是可靠有效的。

5. 简述 80C51 是如何划分的

答：① 通路器  
② 存储器  
③ 寄存器

得分	评卷
31	机

五、编程

1. (10 分)

要求波特率

接收程序

解：全定时

∴ TH  
TL

中机发送：

RAM

WAIT:

NEXT:



5. 简述 80C51 单片机内部数据存储低 128 单元划分为哪 3 个部分，是如何划分的？

答：① 通用寄存器：00H~1FH

② 位寻址区：20H~2FH

③ 用户 RAM 区：30H~7FH

+5

✓

得分	评卷人
31	甄 (15)

### 五、编程题 (共 40 分)

1. (10 分) 假定甲乙机以方式 1 进行串行数据通信，晶振=11.0592MHz，要求波特率为 9600。甲机发送，乙机接收。请编写甲机发送和乙机接收程序。(采用查询方式)

解：全定时 10 被使用初值  $x = 256 - \frac{110592 \times 10^6}{384 \times 9600} = 256 - 3 = 253 = FDH$

∴ TH0 = #0FDH  
TL0 = #0FDH

甲机发送：

```
MOV TMOD, #02H
MOV TH0, #0FDH
MOV TL0, #0FDH
MOV STH, #00H
SETB TR0
MOV SCON, #40H
MOV A, @R0
MOV SBUF, A
WAIT: JBC TL, NEXT
SJMP WAIT
NEXT: INC R0
LJMP TRAM
```

乙机接收：

```
MOV TMOD, #02H
MOV TH0, #0FDH
MOV TL0, #0FDH
MOV STH, #00H
SETB TR0
MOV SCON, #50H
WAIT: JBC RI, NEXT
SJMP WAIT
NEXT: MOV A, SBUF
MOV @R0, A
INC R0
LJMP WAIT
```



2. (8分) 有三个数分别放在内部 RAM30H、31H 和 32H, 编程求其  
中的最大值, 并将其值存入外部 RAM40H 的单元中。

解:

```

ORG 0000H
LJMP MAIN
ORG 0020H

MAIN: MOV 30H, #07H
      MOV 31H, #2EH
      MOV 32H, #13H
      CLR C
      MOV R2, #02H ; 比较三次
      LOOP: MOV R0, #30H ; 第一个数存入 R
            MOV A, @R0
            JNC NEXT ; 第一个数小于第二个数, 执行 NEXT, 否则向下执行
            SUBB A, @R0
            JC NEXT ; 第一个数大于第二个数, 执行 NEXT, 否则向下执行
            INC R0
            MOV A, @R0
            JNC NEXT ; 第二个数小于第三个数, 执行 NEXT, 否则向下执行
            SUBB A, @R0
            JC NEXT ; 第二个数大于第三个数, 执行 NEXT, 否则向下执行
            INC R0
            MOV A, @R0
            JNC NEXT ; 第三个数小于最大值, 执行 NEXT, 否则向下执行
            SUBB A, @R0
            JC NEXT ; 第三个数大于最大值, 执行 NEXT, 否则向下执行
            INC R0
            MOV A, @R0
            JNC NEXT ; 最大值存入 A
            DJNZ R2, LOOP ; 比较次数减 1
            MOV DPTR, #0040H
            MOVX @DPTR, A ; 把最大值(A)存入外部 RAM 40H 中
            END
NEXT:

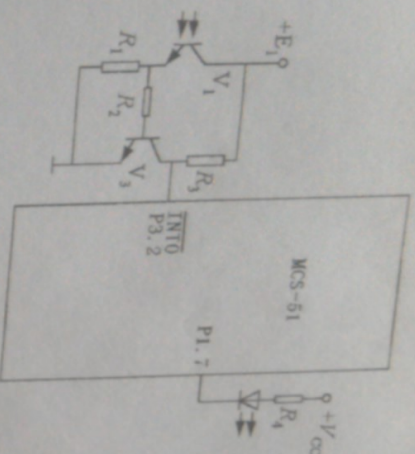
```

3. (22分) 编写警示灯控制程序, 要求警示灯白天熄灭, 晚上断续  
点亮, 时间间隔 2 秒, 即点亮 2 秒, 熄灭 2 秒, 如此周期循环进行。  
要求定时采用定时器 T0、工作方式方式 1, 并采用中断方式定时。  
(单片机晶振为 6MHz)

- (1) 光敏三极管 V1 白天导通, 晚上关断, 试分析单片机的 P3.2  
位白天为高电平还是低电平? 晚上为高电平还是低电平? (4分)
- (2) 编写主程序 (12分)、中断服务子程序 (6分)。



程求其



解: (1) 光敏三极管  $V_1$  白天导通,  $P3.2$  为低电平;  $V_1$  晚上截止, 则  $P3.2$  为高电平。

(2)  $\therefore$  条件  $T0$  工作方式为 1, 设初值为  $x$ ,  $25 = 200000 \mu s = 20 \times 10000 \mu s$

则  $(65536 - x) \times 2 \mu s = 100000 \mu s$

$\therefore x = 15536 = 3CB0H \therefore TH0 = 3CH, TL0 = 08H$

思路:

白天模式 DAY

夜间模式 NIGHT

$T0$ : 中断初值定时;

白天模式

```

MAIN:
    ORG 0000H
    LJMP MAIN
    ORG 0003H
    LJMP NIGHT
    ORG 000BH
    LJMP 000BH
    ORG 0030H
    MOV TMOD, #01H ; 定时器0
    MOV TH0, #3CH
    MOV TL0, #08H
    SETB TR0
    SETB EA
    SETB EX0 ; 开启T0中断

```

SETB EX0 ; 开启T0中断

DAY: SETB P1.7 ; 白天LED灯及P1.7引脚输出高电平

NIGHT: SJMP DAY ; 夜晚模式的循环, 直到天亮

push ACC

push PSW

SETB RS0

CLR R4

MOV TMOD, #01H

MOV TH0, #3CH

MOV TL0, #08H

SETB TR0

SETB EA

SETB EX0

SETB EX0

SETB EX0

MOV TH0, #3CH

MOV TL0, #08H

SETB TR0

SETB EA

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

MOV TH0, #3CH

MOV TL0, #08H

SETB TR0

SETB EA

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

MOV TH0, #3CH

MOV TL0, #08H

SETB TR0

SETB EA

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

MOV TH0, #3CH

MOV TL0, #08H

SETB TR0

SETB EA

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

MOV TH0, #3CH

MOV TL0, #08H

SETB TR0

SETB EA

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

MOV TH0, #3CH

MOV TL0, #08H

SETB TR0

SETB EA

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

MOV TH0, #3CH

MOV TL0, #08H

SETB TR0

SETB EA

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

MOV TH0, #3CH

MOV TL0, #08H

SETB TR0

SETB EA

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

MOV TH0, #3CH

MOV TL0, #08H

SETB TR0

SETB EA

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

MOV TH0, #3CH

MOV TL0, #08H

SETB TR0

SETB EA

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

SETB EX0

17

P3.2

定时。

进行。

中断