

一、填空题

- 1、单片机是将微处理器、一定容量的 RAM 和 ROM 以及 I/O 口、定时器等电路集成在一块芯片上而构成的微型计算机。
- 2、单片机 80C51 片内集成了 4 KB 的 FLASH ROM，共有 5 个中断源。
- 3、两位十六进制数最多可以表示 256 个存储单元。
- 4、在 80C51 中，只有当 EA 引脚接 高 电平时，CPU 才访问片内的 Flash ROM。
- 5、当 CPU 访问片外的存储器时，其低八位地址由 P0 口提供，高八位地址由 P2 口提供，8 位数据由 P0 口提供。
- 6、在 I/O 口中，P0 口在接 LED 时，必须提供上拉电阻，P3 口具有第二功能。
- 7、80C51 具有 64 KB 的字节寻址能力。
- 8、在 80C51 中，片内 RAM 分为地址为 00H~7FH 的真正 RAM 区，和地址为 80H~FFH 的特殊功能寄存器(SFR) 区两个部分。
- 9、在 80C51 中，通用寄存器区共分为 4 组，每组 8 个工作寄存器，当 CPU 复位时，第 0 组寄存器为当前的工作寄存器。
- 10、数据指针 DPTR 是一个 16 位的特殊功能寄存器 寄存器。
- 11、在 80C51 中，一个机器周期包括 12 个振荡周期，而每条指令都由一个或几个机器周期组成，分别有单周期指令、双周期指令和 4 周期指令。
- 12、当系统处于正常工作状态且振荡稳定后，在 RST 引脚上加一个 高 电平并维持 2 个机器周期，可将系统复位。
- 13、单片机 80C51 复位后，其 I/O 口锁存器的值为 0FFH，堆栈指针的值为 07H，SBUF 的值为 不定，内部 RAM 的值不受复位的影响，而其余寄存器的值全部为 0H。
- 14、在 80C51 中，有两种方式可使单片机退出空闲模式，其一是 任何的中断请求被响应，其二是 硬件复位；而只有 硬件复位 方式才能让进入掉电模式的单片机退出掉电模式。
- 15、单片机 80C51 的 5 个中断源分别为 INT0、INT1、T0、T1 以及 TXD/RXD。
- 16、单片机 80C51 的中断要用到 4 个特殊功能寄存器，它们是 TCON、SCON、IE 以及 IP。
- 17、在 80C51 中，外部中断由 IT0(1)位来控制其两种触发方式，分别是 电平 触发方式和 边沿 触发方式。
- 18、中断处理过程分为 4 个阶段，即 中断请求、中断响应、中断服务 以及 中断返回。
- 19、单片机 80C51 片内有两个 16 位的定时/计数器，即 T0 和 T1，它们都有 定时 和 计数 的功能。
- 20、单片机 80C51 的时钟频率为 6MHz，若要求定时 1ms，定时/计数器工作于模式 1，其定时/计数器的初值为 FE0CH。
- 21、单片机 80C51 具有 并行 通信和 串行 通信两种通信方式。
- 22、串行通信有 同步 通信和 异步 通信两种通信方式。
- 23、在异步通信中，数据的帧格式定义一个字符由 4 部分组成，即：起始位、数据位、奇偶校验位 和 停止位。
- 24、串行通信中，为使设备同步工作，需要通信双方有两个共同的要求，一是 通信双方必须采用统一的编码方式，二是 通信双方必须能产生相同的传送速率。
- 25、单片机 80C51 中的串行通信共有 4 种方式，其中方式 0 是用作同步移位寄存器来扩展 I/O 口的。
- 26、设 80C51 的晶振频率为 11.0592MHz，选用定时器 T 工作模式 2 作波特率发生器，波特率为 2400b/s，且 SMOD 置 0，则定时器的初值为 F4H。
- 27、键盘可分为 独立连接 式和 矩阵 式两类。键盘可分为 编码 式和 非编码 式两类。

28、LED 数码管有 静态 显示和 动态 显示两种方式。

29、在执行下列指令后，A= 60H，R0= 45H，(60H)= 45H。

MOV A, # 45H

MOV R0, # 60H

MOV @R0, A

XCH A, R0

30、设 RAM 中(2456H)=66H，(2457H)=34H，ROM 中(2456H)=55H，(2457H)=64H。请分析下面程序执行后各寄存器的内容。(A)= 64H，(DPTR)= 2456H。

MOV A, #1

MOV DPTR, #2456H

MOVC A, @A+DPTR

31. 对单片机而言，连接到数据总线上的输出口应具有 (锁存) 功能。

32. 决定程序执行的顺序是 (PC) 寄存器，该寄存器复位时的值为 (0000h)。

33. 单片机内包含组成微机的三个主要功能部件是 CPU、存储器和 (I/O 口)。

34. 80C51 系列单片机字长是 (8) 位，有 (40) 根引脚。

35. ALE 信号的作用是 (锁存允许)。

36. 80C51 复位后，PC= (0000) H。

37. 80C51 的堆栈指针是 (SP)。

38. 80C51 单片机的 P0 口和 P2 口除了可以作为并行口进行数据的输入 / 输出外，通常还用来构建系统的 (地址) 和 (数据)。

39. 80C51 单片机外部中断请求信号有电平方式和 (边沿触发)，在电平方式下，当采集到 INT0、INT1 的有效信号为 (低电平) 时，激活外部中断。

40. 80C51 单片机指令 ANL A, 20H 中源操作数的寻址方式是 (直接)。

41. 串行通讯中有 (同步) 和异步两种基本方式。

42. C51 中存储类型 XDATA 定义的是 (外部) 存储空间，其地址范围是 (0000h~ffffH)。

43. 执行下列程序后，(A)= 35H，(B)= 16H。

MOV A, #9FH

MOV B, #36H

ANL B, A

SETB C

ADDC A, B

二、选择题

1、80C51 是以下哪个公司的产品？(C)

A、INTEL B、AMD C、ATMEL D、PHILIPS

2、80C51 系列单片机是属于 (C) 体系结构。

A、冯诺依曼 B、普林斯顿 C、哈佛 D、图灵

3、以下哪一条指令的写法是错误的 (C)。

A、MOV DPTR,#3F98H B、MOV R0,#0FEH C、MOV 50H,#0FC3DH D、INC R0

4、以下哪一条指令的写法是错误的 (D)。

A、INC DPTR B、MOV R0,#0FEH C、DEC A D、PUSH A

5、以下哪一条指令的写法是错误的 (B)。

A、MOVC A,@A+DPTR B、MOV R0,#FEH C、CPL A D、PUSH ACC

6、以下哪一条是位操作指令 (B)。

A、MOV P0, #0FFH B、CLR P1.0 C、CPL A D、POP PSW

7、以下哪一条是位操作指令 (B)。

A、MOV P1, #0FFH B、MOV C,ACC.1 C、CPL A D、POP PSW

8、以下哪一条是位操作指令 (B)。

A、MOV P0, #0FFH B、SETB TR0 C、CPL R0 D、PUSH PSW

9、以下哪种方式的接口总线最少? (C)

A、SPI B、I2C C、单总线 D、并行通信

10、以下哪个是属于单片机系统前向通道的器件? (A)

A、A/D 转换 B、D/A 转换 C、LED 数码管 D、继电器

11、80C51 单片机的 RS1,RS0=01 时, 当前寄存器 R0—R7 占用内部 RAM(B)单元。

A、00H—07H B、08H—0FH C、10H—17H D、18H—1FH

12、80C51 单片机有片内 RAM 容量(A)。

A. 128B B、4KB C、8KB D、256B

13、80C51 单片机的最大时序定时单位是(D)。

A、拍节 B、状态 C、机器周期 D、指令周期

14、80C51 单片机的定时器/计数器工作方式 0 是 (C)。

A、8 位计数器结构 B、16 位计数器结构 C、13 位计数器结构 D、2 个 8 位计数器结构

15、80C51 单片机的外部中断 0 中断入口地址为(C)。

A、000BH B、001BH C、0003H D、0013H

16.在下列 80C51 单片机各条指令中, 错误的是 (A)。

A. MOVC @A+DPTR, A B. MOV A, @R0 C. MOV 20H, A D. MOV B, A

17. 单片机上电复位后, 堆栈区的最大允许范围是 (B) 个单元。

A. 64 B. 120 C. 128 D. 256

18. 80C51 单片机 ALE 引脚是 (C)。

A. 输出高电平 B. 输出低电平 C. 输出矩形脉冲, 频率为 fosc 的 1/6

D. 输出矩形脉冲, 频率为 fosc 的 1/2

19. 80C51 单片机的 XTAL1 和 XTAL2 引脚是 (D) 引脚。

A、外接定时器 B、外接串行口 C、外接中断 D、外接晶振

20.80C51 的串行数据缓冲器 SBUF 用于 (D)。

A. 存放运算中间结果 B. 暂存数据和地址 C. 存放待调试的程序

D. 存放待发送或已接收到的数据

21. 假定设置堆栈指针 SP 的值为 37H, 在进行子程序调用时把断点地址进栈保护后, SP 的值为 (D)。

A. 6H B. 37H C. 38H D. 39H

22. 单片机中的程序计数器 PC 用来 (B)。

A. 存放指令 B. 存放正在执行的指令地址

C. 存放下一条指令地址 D. 存放上一条指令地址

23. 在 80C51 指令中, 下列指令中 (C) 是无条件转移指令。

A. LCALL addr16 B. DJNZ direct,rel C. SJMP rel D. ACALL addr11

24. 80C51 系列单片机的中断系统具有 (A)。

A、5 个中断源 B、6 个中断源 C、2 个中断源 D、3 个中断源

25. 指令和程序是以 (C) 形式存放在程序存储器中。

A、源程序 B、汇编程序 C、二进制编码 D、BCD 码

26. 当需要从 80C51 单片机程序存储器取数据时, 采用的指令为 (B)

A、MOV A, @R1 B、MOVC A, @A + DPTR

C、MOVB A, @R0 D、MOVX A, @DPTR

三、判断

- 1、在 51 系列单片机的指令系统中，其加法、减法、乘法和除法必须有累加器 A 的参与才能完成。T
- 2、当 80C51 的 EA 引脚接低电平时，CPU 只能访问片外 ROM，而不管片内是否有程序存储器。T
- 3、当 80C51 的 EA 引脚接高电平时，CPU 只能访问片内的 4KB 空间。F
- 4、80C51 系列单片机直接读端口和读端口锁存器的结果永远是相同的。F
- 5、是读端口还是读锁存器是用指令来区别的。T
- 6、在 80C51 的片内 RAM 区中，位地址和部分字节地址是冲突的。F
- 7、中断的矢量地址位于 RAM 区中。F
- 8、在 80C51 中，当 CPU 访问片内、外 ROM 区时用 MOVC 指令，访问片外 RAM 区时用 MOVX 指令，访问片内 RAM 区时用 MOV 指令。T
- 9、工作寄存器区不允许做普通的 RAM 单元来使用。F
- 10、工作寄存器组是通过置位 PSW 中的 RS0 和 RS1 来切换的。T
- 11、特殊功能寄存器可以当作普通的 RAM 单元来使用。F
- 12、访问 128 个位地址用位寻址方式，访问低 128 字节单元用直接或间接寻址方式。T
- 13、堆栈指针 SP 的内容可指向片内 00H~7FH 的任何 RAM 单元，系统复位后，SP 初始化为 00H。F
- 14、DPTR 只能当作一个 16 位的特殊功能寄存器来使用。F
- 15、程序计数器 PC 是一个可以寻址的特殊功能寄存器。F
- 16、单片机 80C51 复位后，其 PC 指针初始化为 0000H，使单片机从该地址单元开始执行程序。T
- 17、单片机系统上电后，其内部 RAM 的值是不确定的。T
- 18、在 80C51 中，当产生中断响应时，所有中断请求标志位都由硬件自动清零。F
- 19、在 51 系列单片机中，中断服务程序从矢量地址开始执行，一直到返回指令 RETI 为止。T
- 20、在执行子程序调用或执行中断服务程序时都将产生压栈的动作。T
- 21、定时/计数器工作于定时方式时，是通过 80C51 片内振荡器输出经 12 分频后的脉冲进行计数，直至溢出为止。T
- 22、定时/计数器工作于计数方式时，是通过 809C51 的 P3.4 和 P3.5 对外部脉冲进行计数，当遇到脉冲下降沿时计数一次。T
- 23、定时/计数器在工作时需要消耗 CPU 的时间。F
- 24、定时/计数器的工作模式寄存器 TMOD 可以进行位寻址。F
- 25、定时/计数器在使用前和溢出后，必须对其赋初值才能正常工作。F
- 26、在 51 系列单片机的指令中，既有带借位的减法指令，又有不带借位的减法指令。F
- 27、单片机 80C51 的定时/计数器是否工作可以通过外部中断进行控制。T
- 28、并行通信的优点是传送速度快，缺点是所需传送线较多，远距离通信不方便。T
- 29、串行通信的优点是只需一对传送线，成本低，适于远距离通信，缺点是传送速度较低。T
- 30、异步通信中，在线路上不传送字符时保持高电平。T
- 31、在异步通信的帧格式中，数据位是低位在前高位在后的排列方式。T
- 32、异步通信中，波特率是指每秒传送二进制代码的位数，单位是 b/s。T
- 33、在 80C51 的串行通信中，串行口的发送和接收都是对特殊功能寄存器 SBUF 进行读/写而实现的。T
- 34、在单片机 809C51 中，串行通信方式 1 和方式 3 的波特率是固定不变的。F
- 35、在单片机 809C51 中，读和写的 SBUF 在物理上是独立的，但地址是相同的。T
- 36、单片机 80C51 一般使用非整数的晶振是为了获得精确的波特率。T
- 37、单片机 809C51 和 PC 机的通信中，使用芯片 MAX232 是为了进行电平转换。T
- 38、在 A/D 转换器中，逐次逼近型在精度上不及双积分型，但双积分型在速度上较低。T
- 39、A/D 转换的精度不仅取决于量化位数，还取决于参考电压。T

四、简答题

1、简述 80C51 单片机中断的概念。

答：当 CPU 正在处理某件事情的时候，外部发生的某一事件请求 CPU 迅速去处理，于是，CPU 暂时中止当前的工作，转去处理所发生的时间。中断服务处理完该事件以后，再回到原来被中止的地方，继续原来的工作，这样的过程称为中断。

2、什么是保护现场，什么是恢复现场？

答：保护现场：当 CPU 响应中断程序时，硬件会自动把断点地址（16 位程序计数器的值）压入堆栈之外，用户还须注意保护有关工作寄存器、累加器、标志位等信息；

恢复现场：在完成中断服务程序后，恢复原工作寄存器、累加器、标志位等的的内容。

3、单片机 80C51 有哪些中断源，对其中断请求如何进行控制？

答：80C51 中断系统有 5 个中断源：

●INT0：外部中断 0 请求，低电平有效。通过 P3.2 引脚输入。

●INT1：外部中断 1 请求，低电平有效。通过 P3.3 引脚输入。

●T0：定时器/计数器 0 溢出中断请求。

●T1：定时器/计数器 1 溢出中断请求。

●TXD/RXD：串行口中断请求。当串行口完成一帧数据的发送或接收时，便请求中断。

4、简述单片机 809C51 中断的自然优先级顺序，如何提高某一中断源的优先级别。

答：中断源（控制位）自然优先级

外部中断 0（PX0）最 高

定时器/计数器 0 溢出中断（PT0）

外部中断 1（PX1）

定时器/计数器 1 溢出中断（PT1）

串行口中断（PS）最 低

若某几个控制位为 1，则相应的中断源就规定为高级中断；反之，若某几个控制位为 0，则相应的中断源就规定为低级中断。当同时接收到几个同一优先级的中断请求时，响应哪个中断源则取决于内部硬件查询顺序（即自然优先级）。

5、简述 51 系列单片机中断响应的条件。

答：●有中断源发出中断请求；

●中断总允许位 EA=1，即 CPU 开中断；

●申请中断的中断源的中断允许位为 1，即中断没有屏蔽；

●无同级或更高级中断正在被服务；

●当前的指令周期已经结束；

6、简述定时/计数器 4 种工作模式中方式 0、1、2 的特点。

答：方式 0：是 13 位的定时器/计数器，寄存器 TLx 存低 5 位，THx 存高 8 位。

方式 1：是 16 位的定时器/计数器；

方式 2：把 TL0（或 TL1）配置成一个可以自动重载的 8 位定时器/计数器；

7、简述 80C51 串口通信的四种方式及其特点。

方式 0：同步移位寄存器输入/输出方式，常用于扩展 I/O 口。波特率固定为振荡频率的 1/12，并不受 PCON 寄存器中 SMOD 位的影响。

方式 1：用于串行发送或接收，为 10 位通用异步接口。TXD 与 RXD 分别用于发送与接收数据。收发一帧数据的格式为 1 位起始位、8 位数据位（低位在前）、1 位停止位，共 10 位。波特率由定时器 T1 的溢出率与 SMOD 值同时决定。

方式 2：用于串行发送或接收，为 11 位通用异步接口。TXD 与 RXD 分别用于发送与接收数据。收发一帧数据的格式为 1 位起始位、8 位数据位（低位在前）、1 位可编程的第 9 数据位和 1 位停止

位，共 11 位。波特率取决于 PCON 中 SMOD 位的值：当 SMOD=0 时，波特率为 的 1/64；当 SMOD=1 时，波特率为 的 1/32。

方式 3：用于串行发送或接收，为 11 位通用异步接口。TXD 与 RXD 分别用于发送与接收数据。帧格式与方式 2 相同，波特率与方式 1 相同。

- 8、简述在使用普通按键的时候，为什么要进行去抖动处理，如何处理。

键抖动会引起一次按键被误读多次。为了确保 CPU 对键的一次闭合仅做一次处理，必须去除键抖动。在键闭合稳定时，读取键的状态，并且必须判别；在键释放稳定后，再作处理。按键的抖动，可用硬件或软件两种方法消除。

- 9、简述 LED 数码管动态扫描的原理及其实现方式。

动态扫描的原理是利用人的视觉暂留，让人觉得各位 LED 同时点亮一样。逐位轮流点亮各个 LED，每一位保持 1ms，在 10~20ms 之内再一次点亮，重复不止，就可以实现动态扫描。

五、编程题

1、用 P1 口控制 LED 单色灯实现流水灯显示，单色灯 0 亮 1 灭，已知晶振 12MHz，

2、用 P0 口控制 LED 单色灯实现走马灯显示，单色灯 0 亮 1 灭，已知晶振 12MHz，

3、数码管 0~9 的显示。

4、用一个定时/计数器加软件计数器的方式，实现一秒的时钟基准信号，试写出程序并加以说明。

（设晶振频率为 12MHz，由 P1.0 口输出秒信号。）

```
#include<reg51.h>
sbit P10 = P1^0;
unsigned char k;
void main()
{
    TMOD = 0x01;
    TL0 = 0xb0;
    TH0 = 0x3c;
    TR0 = 1;
    ET0 = 1;
    EA = 1;
    P10 = 1;
    while(1);
}
void t0_isr() interrupt 1
{
    TL0 = 0xb0;
    TH0 = 0x3c;
    k++;
    if(k==20)
    {
        k=0;
        P10 = ~P10;
    }
}
```