**单片机应用技术项目教程（C语言版）习题答案**

**项目一**

1-1 填空题

（1）8031，8751

（2）存储器，定时器/计数器，输入/输出接口电路

（3）上电自动复位，手动复位

（4）12，1μs，1，4

（5）1

（6）4，1，2，5

1-2选择题

（1）A

（2）C

（3）D

（4）C

（5）A

1-3

（1）Keil C51软件的主要功能

Keil C51软件提供了丰富的库函数和功能强大的集成开发调试工具，全Windows界面。μVision2 集成开发环境可以完成工程建立和管理、编译、连接、目标代码的生成、软件仿真和硬件仿真等完整的开发流程。

（2）PROTEUS软件的主要功能

Proteus不仅是模拟电路、数字电路、模/数混合电路的设计与仿真平台，也是目前较先进的单片机和嵌入式系统的设计与仿真平台。在计算机上，能完成从原理图与电路设计、电路分析与仿真、单片机代码级调试与仿真、系统测试与功能验证到形成PCB的完整的电子设计、研发过程。

1-4 简叙单片机的发展及发展趋势。

单片机的发展主要有以下几个方面：

（1）增加字长，提高数据精度和处理的速度；

（2）改进制作工艺，提高单片机的整体性能；

（3）由复杂指令集CISC转向简单指令集RISC的技术；

（4）多功能模块集成技术，使一块“嵌入式”芯片具有多种功能；

（5）微处理器与DSP技术相结合；

（6）融入高级语言的编译程序；

（7）低电压、宽电压、低功耗。

目前，国际市场上8位、16位单片机系列已有很多，32位的单片机也已经进入了实用阶段。随着单片机技术的不断发展，新型单片机还将不断涌现，单片机技术正以惊人的速度向前发展。

1-5 简叙单片机的主要应用领域。

单片机的主要应用领域：工业控制、智能仪表、外设控制、家用电器、机器人、军事装置等。

1-6 简叙单片机的主要特点。

（1）体积小、重量轻，价格低、功能强，电源单一、功耗低，可靠性高、抗干扰能力强。

（2）使用方便灵活、通用性强。

（3）大多数单片机采用哈佛（Harvard）结构体系，单片机的数据存储器空间和程序存储器空间相互独立。

（4）突出控制功能的指令系统。

（5）较低的处理速度和较小的存储容量。

1-7 如果只使用片外ROM，引脚应该如何接？为什么？

（1）引脚应该接地

（2）当为高电平时，访问内部程序存储器，当超过内部程序存储器地址范围后，自动转向外部程序存储器；当为低电平时，则访问外部程序存储器。

1-8 在工作模块2 中，如果把LED闪烁程序的延时函数Delay()写在main()后面，程序应该如何修改？

要在main()函数前面，对Delay()函数的原型进行声明，代码如下：

void Delay();

1-9 单片机最小系统由哪几部分组成？现要求LED的阳极接在P1.0引脚上，请完成LED点亮电路和C语言程序设计。

（1）最小系统就是指由单片机和一些基本的外围电路所组成的一个可以工作的单片机系统。一般来说，它包括单片机、晶振电路和复位电路。

（2）电路如下：



（3）程序如下：

#include <AT89X52.H> //包含AT89X52.H头文件

sbit LED=P1^0; //定义LED为P1.0引脚

void main (void)

{

 LED=1; //P1.0引脚输出高电平，点亮LED

 while(1);

}

1-10 请完成用开关控制LED闪烁快和慢两种效果的电路和C语言程序设计。

（1）电路如图1-20所示

（2）程序如下：

#include <AT89X52.H> //包含AT89X52.H头文件

sbit SW=P3^0; //定义SW是P3.0位对应的引用符号

sbit LED=P1^0; //定义LED是P1.0位对应的引用符号

void Delay() //延时函数

{

 unsigned char i, j;

 for (i=0;i<255;i++)

 for (j=0;j<255;j++);

}

void main (void)

{

 while(1)

 {

 if(SW==0) //开关闭合SW=0：LED闪烁快

 {

LED = 0; // P1.0=0，LED点亮

 Delay(); //延时

 LED = 1; // P1.0=1，LED熄灭

 Delay();

}

 Else //开关断开SW=1：LED闪烁慢

 {

LED = 0; // P1.0=0，LED点亮

 Delay(); Delay(); //延时时间变长

 LED = 1; // P1.0=1，LED熄灭

 Delay(); Delay();

}

 }

}

**项目二**

2-1填空题

（1）20H，2FH，21H

（2）确定工作寄存器组

（3）00H，1FH，4

（4）P0，P2

（5）07H，0000H，00H，0FFH

（6）3，18H～1FH

（7）bit，20H～2FH

（8）0～255，0～65535

2-2选择题

（1）B，A

（2）B

（3）C

（4）C

（5）B

（6）D

2-3 P0口、P1口、P2口和P3口的负载能力是多少？它们是否具有位寻址功能？

（1）P0口负载能力为8个TTL门电路，P1口、P2口和P3口负载能力为4个TTL门电路。

（2）具有位寻址功能

2-4 在输出时，P0口为什么要外接上拉电阻才能有高电平输出？

由于P0口的输出电路是漏极开路电路，所以必须外接上拉电阻才能有高电平输出。

2-5 MCS-51单片机有哪几个存储空间？是如何分布的？

（1）MCS-51单片机有程序存储器和数据存储器2个存储空间。

（2）MCS-51单片机存储器在物理结构上可分为4个存储空间：内部数据存储器、内部程序存储器、外部数据存储器和外部程序存储器。

从逻辑上分，即从用户使用的角度看，MCS-51单片机存储器分为3个逻辑空间：片内外统一编址的程序存储器地址空间、内部数据存储器地址空间和外部数据存储器地址空间。

2-6 MCS-51单片机内部RAM分成几个不同区域及地址范围？

内部数据存储器在物理上和逻辑上都分为两个地址空间：00H～7FH单元组成的低128字节数据存储器空间和80H～FFH单元组成的高128字节特殊功能寄存器空间。

2-7 PSW的作用是什么？常用的状态标志有哪几位？其作用是什么？能否位寻址?

PSW是一个8位特殊功能寄存器，用于存放程序运行中的各种状态信息，可以位寻址。

（1）CY（PSW.7）：高位进位标志位。

（2）AC（PSW.6）：辅助进位标志位。

（3）F0（PSW.5）：用户标志位。

（4）RS1（PSW.4）、RS0（PSW.3）：寄存器组选择位。

（5）OV（PSW.2）：溢出标志位。

（6）－（PSW.1）：保留位，无定义。

（7）P（PSW.0）：奇偶校验位。

2-8 bit和sbit有什么区别？

bit声明的是位变量，必须定位在MCS-51单片机内部RAM的位寻址空间中。

sbit声明的是可寻址位，用于定义某些特殊位，利用它可以访问芯片内部的RAM中的可寻址位或特殊功能寄存器中的可寻址位。

2-9在C语言里，sbit P1\_0 = 0x90语句作用是什么？能不能直接使用P1.0（说明原因）？

（1）sbit P1\_0 = 0x90语句作用是声明P1\_0为P1中的P1.0引脚。

（2）由于C语言规定标识符是由字母、数字和下划线组成，所以不能直接使用P1.0作为标识符。

2-10 试一试能否将工作模块3的LED循环点亮改为LED双向循环点亮。

LED双向循环点亮代码如下：

#include <AT89X52.H> //包含AT89X52.H头文件

void Delay() //延时函数

{

 unsigned char i, j;

 for (i=0;i<255;i++)

 for (j=0;j<255;j++);

}

void main()

{

 unsigned char i;

 unsigned char temp;

 P1 = 0xff; //十六进制全1,熄灭所有LED

 while(1)

 {

 temp = 0x01; //正方向，初始控制码为0x01

 for (i=0;i<8;i++)

 {

 P1 = ~ temp; //temp值取反送P1口

 Delay();

 temp = temp << 1 ; //temp值左移一位，获得下一个控制码

 }

temp = 0x80; //反方向，初始控制码为0x80

 for (i=0;i<8;i++)

 {

 P1 = ~ temp; //temp值取反送P1口

 Delay();

 temp = temp >> 1 ; //temp值右移一位，获得下一个控制码

 }

 }

 }

2-11 设计用开关控制步进电机转向的AT89S52单片机控制系统，功能要求：开关闭合，正转；开关断开，反转。

（1）电路如下：



（2）程序如下：

#include <AT89X52.H>

sbit SW=P3^0; //符号SW表示 P3.0 引脚

//由delay参数确定延迟时间

void mDelay (unsigned int delay)

{

 unsigned int i;

 for(;delay >0; delay--)

 for(i=0;i<124;i++);

}

void main()

{

 while(1)

 {

 if(SW==0) // SW闭合，正转

{

P1=0x03; //A、B绕组励磁

 mDelay (50);

 P1=0x06; //B、C绕组励磁

 mDelay (50);

 P1=0x0C; //C、D 绕组励磁

 mDelay (50);

 P1=0x09; //D、A绕组励磁

 mDelay (50);

}

else // SW断开，反转

{

P1=0x03; //A、B绕组励磁

mDelay (50);

P1=0x09; //D、A绕组励磁

 mDelay (50);

 P1=0x0C; //C、D 绕组励磁

mDelay (50);

P1=0x06; //B、C绕组励磁

 mDelay (50);

}

 }

}

2-12 设计开关控制电灯点亮的AT89S52单片机控制系统，驱动电路采用 ULN2003A和继电器。功能要求：开关闭合，电灯点亮；开关断开，电灯熄灭。提示：参考【技能训练2-2】单片机驱动继电器电路设计。

**项目三**

3-1填空题

（1）if，switch

（2）无限

（3）1000

（4）\0

（5）该变量在C51硬件系统中所使用的存储区域，并在编译时准确的定位

（6）code

（7）动态，静态

（8）C0H

3-2选择题

（1）A

（2）A

（3）D

（4）A

（5）A

（6）C（因为使用的是汉字状态下的双引号）

（7）C

（8）B

3-3 LED数码管有哪两种结构？是如何实现的？

（1）LED数码管有共阴极和共阳极两种结构。

（2）共阴极结构：是把所有发光二极管的阴极作为公共端（com）连起来。共阳极结构：是把所有发光二极管的阳极作为公共端（com）连起来。

3-4 请简要说明LED数码管静态显示和动态显示的特点，实际设计时应如何选择？

（1）采用静态显示方式，较小的电流就可以获得较高的亮度，且占用CPU时间较少，编程简单，显示便于检测和控制，但其占用的I/O口线较多，硬件电路复杂，成本高，只适合显示位数较少的场合。

（2）当显示位数较多时，采用动态扫描方式可以节省I/O端口资源，硬件电路也较简单；但其稳定度不如静态显示方式；由于CPU要轮番扫描，将占用更多的CPU时间。

3-5 动态显示的过程是什么？

动态扫描过程如下：

（1）首先从段选线上送出字型编码，再控制位选端，字符就显示在指定数码管上，其他位选端无效的数码管都处于熄灭状态，持续1.5ms时间，然后关闭所有显示；

（2）接下来又送出新的字型编码，按照上述过程又显示在另外一位数码管上；

（3）直到每一位数码管都扫描完为止.

动态扫描显示，数码管其实是轮流依次点亮的，但由于人的视觉驻留效应，因此当每个数码管点亮的时间小到一定程度时，人就感觉不出字符的移动或闪烁，觉得每位数码管都一直在显示，达到一种稳定的视觉效果。

3-6 在用共阳极数码管显示的电路中，如果直接将共阳极数码管换成共阴极数码管，能否正常显示？为什么？应采取什么措施？

（1）不能显示。

（2）由于在共阳极数码管显示的电路中，其公共端（com）是接电源（+5V）的，而共阴极数码管的公共端（com）必须接地。

（3）把接电源的公共端，改接为地。

3-7 工作模块八中，如果轻触开关改接P1.7，并且要求实现0～59循环计数，程序应如何修改？

（1）把“P1\_0==0”都修改为“P1\_7==0”

（2）把“if(Count==100)”修改为“if(Count==60)”

3-8 工作模块七，数码管动态显示程序设计时如果把延时时间改为1s，会出现什么情况？

（1）6个数码管轮流依次显示“1、2、3、4、5、6”，每个数码管显示时间是1s；

（2）在6个数码管轮流依次显示“123456”的6个数字之间，6个数码管会熄灭1s。

3-9 如何设计0～999的计数器。

（1）采用动态扫描显示

（2）在工作模块七的基础上，使用高3位数码管。

（3）程序如下：

#include <AT89X52.h> //包含AT89X52.H头文件

/\*定义0～9十个数字的字型码表\*/

unsigned char code Tab[]=

{0x3F,0x06,0x5B,0x4F,0x66,0x6D,0x7D,0x07,0x7F,0x6F};

unsigned char code Col[]={0xfe,0xfd,0xfb}; //位码

unsigned int Count; //定义变量放置计数数值

unsigned char buf[]={0,0,0};

void Delay()

{

 unsigned char i;

 for(i=0;i<250;i++);

}

void main() //同时显示123456

{

 unsigned char j;

 Count=0; //计数变量初始为0

 while(1)

 {

 buf[0]=[Count/100]; //获得百位数

buf[1]=[Count/10%10]; //获得十位数

buf[2]=[Count%10]; //获得个位数

for(j=0;j<3;j++)

 {

 P0=Tab[buf[j]];

 P1=Col[j-1]; //开（共阴）数码管

 Delay();

 P1=0xff; //关数码管

 Delay();

 }

Count++; //计数加1

if(Count==1000) //判断计数是否到1000

 {

 Count=0; //到1000，计数器清0

}

 }

}

**项目四**

4-1填空题

（1）10ms

（2）5，2，外部中断0（），中断允许寄存器IE，中断优先级寄存器IP

（3）0003H，001BH

（4）电平触发方式，边沿触发方式

4-2选择题

（1）C

（2）A

（3）B

（4）C

（5）B

4-3什么叫中断？中断有什么优点？

（1）当CPU正在执行某个程序时，由计算机内部或外部的原因引起的紧急事件向CPU发出请求处理的信号，CPU在允许的情况下响应请求处理信号，暂时停止正在执行的程序，保护好断点处的现场，转向执行一个用于处理该紧急事件的程序，处理完后又返回被中止的程序断点处，继续执行原程序，这一过程就称为中断。

（2）能大大地提高计算机处理事件的能力，提高效率，增强实时性。

4-4什么叫中断源？ MCS－51系列单片机有几个中断源，各中断标志是如何产生的，又如何清0的？CPU响应中断时，它们的中断矢量地址分别是多少？

（1）向CPU发出中断请求的来源，或引起中断的原因称为中断源。

（2）MCS－51系列单片机有5个中断源，它们的中断标志是由硬件自动置“1”，其中只有串口的发送中断标志位TI（串口的接收中断标志位RI）是由软件清0的，其它中断标志是由硬件自动清0。

（3）CPU响应中断时，它们的中断矢量地址如下表所示：

|  |  |
| --- | --- |
| 中断源 | 中断入口地址 |
| 外部中断0 | 0003H |
| 定时器T0中断 | 000BH |
| 外部中断1 | 0013H |
| 定时器T1中断 | 001BH |
| 串行口中断 | 0023H |

4-5外部中断有哪两种触发方式？对触发脉冲或电平有什么要求？如何选择和设定？

（1）外部中断有电平触发方式和边沿触发方式两种触发方式。

（2）在电平触发方式下，CPU响应中断后硬件不能自动使IE1清0，也不能由软件使IE1清0，所以在中断返回前必须撤消引脚上的低电平，使IE1置0，否则将产生另一次中断请求，造成出错；在边沿触发方式下，为保证CPU在两个机器周期内检测到先高后低的负跳变，输入高低电平的持续时间至少要保持12个时钟周期。

（3）设置IT0（IT1）为0时，表示外部中断为电平触发方式；设置IT0（IT1）为1时，表示外部中断为边沿触发方式。

4-6 MCS－51系列单片机的中断系统中有几个优先级？如何设定？

（1）MCS－51系列单片机的中断系统中有2个优先级，分别为高优先级和低优先级。

（2）中断优先级寄存器IP的作用是设定各中断源的优先级别，IP的每一位均可以由软件来置1或清0，置1是高优先级，清0是低优先级。

4-7 CPU响应中断有哪些条件？在什么情况下中断响应会受阻？

（1）CPU并非在任何时刻都响应中断请求，而是在条件满足之后才会响应。响应中断的条件是：

1）有中断源发出中断请求。

2）总中断开启，即中断总允许位EA=1。

3）申请中断的中断源的中断允许位为1。

在满足以上条件的基础上，CPU一般会响应中断。

（2）有下列任何一种情况存在，中断响应都会受到阻断：

1）CPU正在响应一个同级或更高优先级的中断。

2）当前指令尚未执行完。

3）正在执行中断返回或访问专用寄存器IE、IP的指令。CPU在执行完上述指令之后，要再执行一条指令才能响应中断请求。

4-8 MCS－51单片机中断处理的过程如何描述？

中断处理过程可以分为三个阶段：中断响应、中断处理、中断返回。

（1）中断响应是CPU对中断源发出的中断请求进行的响应，包括保护断点和把程序转向中断服务函数的入口地址。

（2）中断处理就是执行中断服务函数。中断服务函数从中断入口地址开始执行，直到函数结束为止。

（3）中断返回是指中断服务完成后，CPU返回到原来程序的断点（即原来断开的位置），继续执行原来的程序。

4-9 机械式按键组成的键盘，如何消除按键抖动？独立式按键和矩阵式按键分别有什么特点？实用于什么场合？

（1）消除抖动常用的方法有下列两种：硬件去抖和软件去抖。在键数较少时，可采用硬件去抖；当键数较多时，采用软件去抖。

（2）独立式按键的特点是每个按键单独占用一根I/O口线，每个按键的工作不会影响其他I/O线的状态；矩阵式按键的特点是由行线和列线组成，按键位于行、列的交叉点上，节省了很多I/O口。

（3）在键数较少时，可采用独立式按键；在在键数较少时，可采用矩阵式按键。

4-10 如何使用C语言书写定时器1中断的中断服务程序的首部？要求使用寄存器组2

代码如下：

void t1 (void) interrupt 3 using 2

**项目五**

5-1 填空题

（1）机器周期，外部的事件

（2）定时器

（3）工作方式控制寄存器TMOD，控制寄存器TCON

（4）10H，15H

5-2选择题

（1）C

（2）C

（3）B

（4）D

5-3 定时器/计数器有哪几种工作方式？各有什么特点？

（1）定时器/计数器有方式0、方式1、方式2和方式3四种工作方式。

（2）方式0是13位计数器；方式1是16位计数器；方式2是自动重装初值8位计数器；方式3的定时器0分为两个独立的8位计数器，方式3的定时器1无中断的计数器。

5-4 控制寄存器TMOD和TCON各位的定义是什么？怎样确定各定时器/计数器的工作方式？

（1）TMOD（地址为89H）的作用是设置T0、T1的工作方式。低4位用于控制T0，高4位用于控制T1；TCON（地址为88H）的作用是控制定时器的启动与停止，并保存T0、T1的溢出和中断标志。

（2）是通过工作方式控制位M0和M1，来确定各定时器/计数器工作方式的，如下表所示：

|  |  |
| --- | --- |
| M1 M0 | 工作方式 |
| 0 0 | 方式0 |
| 0 1 | 方式1 |
| 1 0 | 方式2 |
| 1 1 | 方式3 |

5-5 在工作方式3中，定时器/计数器T0和T1的应用有什么不同？

定时器0分为两个独立的8位加1计数器TH0和TL0。TL0既可用于定时，也能用于计数；TH0只能用于定时。

定时器1不能工作在方式3下。

5-6 定时器/计数器作定时器用时，其定时时间与哪些因素有关？作计数器时，对外界计数频率有何限制？

（1）定时器的定时时间与系统的时钟频率有关。因一个机器周期等于12个时钟周期，所以计数频率应为系统时钟频率的1/12（即机器周期）。如晶振频率为12MHz，则机器周期为1μs。通过改变定时器的定时初值，并适当选择定时器的长度（8位、13位或16位），以调整定时时间长短。

（2）外部计数脉冲的频率不能高于晶振频率的1/24。

5-7当定时器T0工作于方式3时，如何使运行中的定时器T1停止下来？

当定时器T0工作于方式3时，T1的控制位TR1、TF1和中断源都被T0占用了。这时若要使运行中的定时器T1停止下来，可将T1设为方式3。

5-8 晶振fosc=6MHZ，T0工作在方式1，最大定时等于多少？

最大定时时间等于131.072ms。

5-9 已知单片机时钟频率fosc=12MHZ，当要求定时时间为50ms和25ms时，试编写定时器/计数器的初始化程序。

（1）定时时间为50ms时，定时器/计数器的初始化程序：

TMOD =0x01; //T0方式1计时50ms

 TH0 = (65536-50000)/256; //定时器T0的高8位

 TL0 = (65536-50000)%256;

EA = 1; //允许所有中断

ET0 = 1; //允许T0中断

 TR0 = 1;

（2）定时时间为25ms时，定时器/计数器的初始化程序：

TMOD =0x01; //T0方式1计时25ms

 TH0 = (65536-25000)/256; //定时器T0的高8位

 TL0 = (65536-25000)%256;

EA = 1; //允许所有中断

ET0 = 1; //允许T0中断

 TR0 = 1;

5-10已知AT89C51时钟频率fosc=6MHZ,试利用定时器编写程序，使P1.0输出一个占空比为1/4的脉冲波。

假设，脉冲周期是100ms。占空比为1/4时：高电平是25ms，低电平是75ms。使P1.0输出一个占空比为1/4的脉冲波，代码如下：

#include <AT89x52.H>

int t;

void main()

{

 EA = 1; //允许所有中断

 ET0 = 1; //允许T0中断

 TMOD =0x01; //T0方式1计时25ms

 TH0 = (65536-25000)/256; //定时器T0的高8位

 TL0 = (65536-25000)%256;

 TR0 = 1;

 while(1);

}

/\* 定时器0中断服务子程序 \*/

void intserv1 (void) interrupt 1 using 1

{

 TH0= -25000/256;

 TL0= -25000%256;

 P1\_0=0; //P1.0输出低电平，保持75ms

t++;

 if(t==3) //t为3时，75ms定时时间到，低电平变为高电平

 {

 t=0;

 P1\_0=1; //75ms定时时间到，P1.0输出高电平，保持25ms

 }

}

5-11 试用中断方式设计秒发生器，即在AT89C51的P1.0口每秒产生一个机器周期的正脉冲，由P1.1口每分钟产生一个机器周期的正脉冲。

假设，时钟频率为12MHZ，机器周期既为1微妙。代码如下：

#include <AT89x52.H>

#include <intrins.h>

unsigned char Second,t;

void main()

{

 EA = 1; //允许所有中断

 ET0 = 1; //允许T0中断

 TMOD =0x01; //T0方式1计时50ms

 TH0 = (65536-50000)/256; //定时器T0的高8位

 TL0 = (65536-50000)%256;

 TR0 = 1;

 P1\_0=0;

 P1\_1=0;

 while(1);

}

/\* 定时器0中断服务子程序 \*/

void intserv1 (void) interrupt 1 using 1

{

 TH0= -25000/256;

 TL0= -25000%256;

 t++;

 if(t==20) //t为20时，1秒计时时间到

 {

 t=0;

 P1\_0=1; //1秒计时时间到，P1.0输出高电平

\_nop\_;

P1\_0=0;

 Second++;

 if(Second>=60)

 {

Second=0;

 P1\_1=1; //1分钟计时时间到，P1.1输出高电平

\_nop\_;

P1\_1=0;

 }

 }

}

5-12 试用定时器中断技术设计一个秒闪电路，要求使发光二极管LED每秒闪亮400ms，设时钟频率为6MHZ。

假设，LED阴极接P1.0引脚，电路在此省略。时钟频率为6MHZ，机器周期为2微妙。代码如下：

#include <AT89x52.H>

unsigned char t10,t4;

void main()

{

 EA = 1; //允许所有中断

 ET0 = 1; //允许T0中断

 TMOD =0x01; //T0方式1计时100ms

 TH0 = (65536-50000)/256; //定时器T0的高8位

 TL0 = (65536-50000)%256;

 TR0 = 1;

 P1\_0=0;

 while(1);

}

/\* 定时器0中断服务子程序 \*/

void intserv1 (void) interrupt 1 using 1

{

 TH0= -25000/256;

 TL0= -25000%256;

 t10++;

t4++;

 if(t10==10) //t10为10时，1秒计时时间到

 {

 t10=0;

t4=0；

 P1\_0=1; //1秒计时时间到，P1.0输出高电平

}

If(t4==4)

{

t4=0;

P1\_0=0; //400ms计时时间到，P1.0输出低电平

}

}

**项目六**

6-1填空题

（1）8，A/D转换器，8

（2）数字，模拟

（3）0.0196

（4）两级数据输入锁存

（5）500k，定时器中断，定时器中断服务程序

6-2 A/D和D/A转换器的作用分别是什么？各在什么场合下使用？

（1）A/D转换器就是把模拟量转变成数字量，以适应单片机对其进行处理；D/A转换器就是把单片机输出的数字量转变成模拟量，以驱动外围部件。

（2）A/D转换器使用在单片机的输入部分，因为外界物理量通常都是模拟信号，而单片机内均是数字信号，所以在单片机的输入端需要模/数转换；D/A转换器在单片机的输出部分，因为执行机构的控制量通常都是模拟信号，而单片机内均是数字信号，因此在单片机的输出端需要数/模转换。

6-3 D/A转换器的主要性能指标有哪些？设某DAC有二进制14位，满量程模拟输出电压10V，试问它的分辨率和转换精度各为多少？

（1）D/A转换器的主要性能指标有：分辨率、线性度、转换精度、建立时间、工作温度范围、电源抑制比、输入形式、输出形式等。

（2）分辨率是D/A转换器能分辨的[最小值](http://www.so.com/s?q=%E6%9C%80%E5%B0%8F%E5%80%BC&ie=utf-8&src=internal_wenda_recommend_textn" \t "https://wenda.so.com/q/_blank)，14位DAC的分辨率为10V/16384=0.61mV。

（3）[转换精度](http://www.so.com/s?q=%E8%BD%AC%E6%8D%A2%E7%B2%BE%E5%BA%A6&ie=utf-8&src=internal_wenda_recommend_textn" \t "https://wenda.so.com/q/_blank)是D/A转换器将[数字量](http://www.so.com/s?q=%E6%95%B0%E5%AD%97%E9%87%8F&ie=utf-8&src=internal_wenda_recommend_textn" \t "https://wenda.so.com/q/_blank)转换为[模拟量](http://www.so.com/s?q=%E6%A8%A1%E6%8B%9F%E9%87%8F&ie=utf-8&src=internal_wenda_recommend_textn" \t "https://wenda.so.com/q/_blank)的理论值与实际值相比。

6-4决定ADC0809模拟电压输入路数的引脚有哪几条？

有3位地址输入线ADDA、ADDB和ADDC，其中ADDA为低位地址，ADDC为高位地址。

6-5 试述ADC0809的特性。

ADC0809主要特性有以下几个方面：

（1）8路8位A／D转换器，即分辨率8位；

（2）具有转换起停控制端；

（3）转换时间为100μs；

（4）单个＋5V电源供电；

（5）模拟输入电压范围0～＋5V，不需零点和满刻度校准；

（6）工作温度范围为-40～＋85摄氏度；

（7）低功耗，约15mW。

6-6 ADC0809的时钟如何提供，通过采用的频率是多少？

ADC0809的内部没有时钟电路，有时钟信号引脚，所需时钟信号由外界提供，通常使用频率为500kHz的时钟信号。

6-7 简述DAC0832的用途和特性。

DAC0832是8位的D/A转换集成芯片，在单片机应用系统中得到广泛的应用。其主要特性如下：

（1）分辨率为2-8位，即DAC0832分辨率是1/256。数字量的位数越多，分辨率就越高，转换器对输入量变化的敏感程度就越高；

（2）电流稳定时间1us；

（3）可单缓冲、双缓冲或直接数字输入；

（4）只需在满量程下调整其线性度；

（5）单一电源供电（+5V～+15V）；

（6）低功耗，20mW。

6-8 DAC0832和MCS-51接口时有哪三种工作方式？各有什么特点？适合在什么场合下使用?

DAC0832有直通方式、单缓冲方式和双缓冲方式三种工作方式。

（1）直通方式：单片机输出的数字量可以被数模转换器直接转换输出。

（2）单缓冲方式：DAC0832的两个输入寄存器中有一个处于直通方式，另一个处于受控的锁存方式。在实际应用中，如果只有一路模拟量输出，或虽有几路模拟量但并不要求同步输出的情况，就可采用单缓冲方式。

（3）双缓冲方式。把DAC0832的两个锁存器都设置成受控锁存方式。双缓冲方式用于多路数/模转换系统，以实现多路模拟信号同步输出的目的。

6-9 编程输出10KHZ的方波、三角波。

（1）电路图如图6-13所示。

（2）输出10KHZ的方波

10KHZ方波的周期=1/10KHZ=100微妙，既高电平保持50微妙、低电平保持50微妙。在T0方式1、晶振12MHz的条件下，定时50μs的定时初值为：X=65536-50。代码如下：

#include<absacc.h> //绝对地址头文件

#include<at89x52.h> //AT89S52的头文件

#define dac0832 XBYTE[0x7fff] //绝对地址的处理

delay() //50μs的延时函数，定时器T0中断采用查询方式

{

 TH0=(65536-50)/256;

 TL0=(65536-50)%256;

 TR0=1;

 while(!TF0);

 TF0=0;

}

main()

{

 TMOD= 0x01; //设置T0工作方式1

 while(1)

 {

 dac0832=0; //DA转换输出低电平

 delay(); //低电平保持50μs

dac0832=0; //DA转换输出高电平

 delay(); //高电平保持50μs

 }

}

（3）输出三角波

输出三角波的程序，工作模块17实现的是正锯齿波输出。在工作模块17基础上在增加一个反锯齿波输出，既先输出一个正锯齿波，紧接着输出一个反锯齿波。这样就可以输出三角波了，代码如下：

#include<absacc.h> //绝对地址头文件

#include<at89x52.h> //AT89S52的头文件

#define dac0832 XBYTE[0x7fff] //绝对地址的处理

delay() //1ms的延时函数

{

 TH0=0xfc;

 TL0=0x18;

 TR0=1;

 while(!TF0);

 TF0=0;

}

main()

{

 int i;

 TMOD = 0x01; //设置T0工作方式1

 while(1)

 {

 for(i=0;i<=255;i++) //形成正锯齿波的输出

 {

 dac0832=i; //DA转换输出

 delay();

 }

for(i=255;i>=0;i--) //形成反锯齿波的输出

 {

 dac0832=i; //DA转换输出

 delay();

 }

 }

}

**项目七**

7-1选择题

（1）D

（2）C

（3）C

（4）D

（5）D

（6）B

（7）A

（8）A

7-2 串行数据传送与并行数据传送相比的主要优点和用途是什么？

并行通信是指将待发送数据的各位同时传送，串行通信则将数据一位一位的按顺序传送。

串行数据传送与并行数据传送相比的主要优点是传输线少，长距离传送时成本低，且可以利用电话网等现成的设备。串行数据传送在单片机双机、多机以及单片机与PC机之间通信等方面被广泛应用。

7-3 简述MCS-51系列单片机串行口四种工作方式的接收和发送数据的过程。

（1）方式0。串行接口工作方式0为同步移位寄存器方式，其波特率是固定的，为fosc（振荡频率）的1/12。

1）方式0发送：数据从RxD引脚串行输出，TxD引脚输出同步脉冲。

2）方式0接收：在满足REN=1和RI=0的条件下，串行口处于方式0输入。

（2）方式1。在方式1时，串行口被设置为波特率可变的8位异步通信接口。

1）方式l发送。串行口以方式1发送时，数据位由TxD端输出，发送1帧信息为10位，其中一位起始位、八位数据位（先低位后高位）和一个停止位“1”。

2）方式1接收。当串行口设置为方式1，且REN=1时，串行口处于方式1的输入状态。当检测到起始位有效时，开始接收一帧的其余信息。

（3）方式2。被定义为9位异步通信接口。

1）方式2发送。发送数据由TXD端输出，发送一帧信息为11位，其中一位起始位（0）、八位数据位（先低位后高位）、一位可控位1或0的第9位数据和一位停止位“1”。

2）方式2接收。当串行口置为方式2时，且REN=1，串行口以方式2接收数据。方式2的接收与方式1基本相似。

（4）方式3。方式3为波特率可变的9位异步通信方式，除了波特率有所区别之外，其余都与方式2相同。

7-4 串行口有几种工作方式？各工作方式的波特率如何确定？

（1）串行口有4种工作方式，分别为方式0、方式1、方式2和方式3。

（2）各工作方式波特率的确定如下：

1）方式0的波特率确定

在方式0下，串行口通信的波特率是固定的，其值为fosc/12（fosc为主机频率）。

2）方式2的波特率确定

在方式2下，通信波特率为fosc/32或fosc/64.用户可以根据PCON中SMOD位状态来驱使串行口在那个波特率下工作。选定公式为：



这就是说，若SMOD=0，则所选波特率为fosc/64；若SMOD=1，则波特率为fosc/32。

3）方式1或方式3的波特率确定

在这两种方式下，串行口波特率是由定时器的溢出率决定的，因而波特率也是可变的。相应公式为：



定时器T1溢出率的计算公式为：



式中，K为定时器T1的位数，它和定时器T1的设定方式有关。即：

若定时器T1为方式0，则K=13

若定时器T1为方式1，则K=16

若定时器T1为方式2或3，则K=8

由上两式可知，方式1或方式3下所选波特率常常需要通过计算来确定初值，因为该初值是要在定时器T1初值化时使用的。

7-5 定时器1做串行口波特率发生器时，为什么常采用方式2 ？

因为定时器T1在方式2下工作时，TH1和TLl分别设定为两个8位重装计数器（当TL1从全“1”变为全“0”时，TH1重装TLl）。这种方式，不仅可使操作方便，也可避免因重装初值（时间常数初值）而带来的定时误差。

7-6 使用AT89S52的串行口按工作方式1进行串行数据通讯，假定波特率为2400b/s，以中断方式传送数据，请编写全双工通讯程序。

（1）串口波特率设置如下：

采用时钟频率fosc=11.0592MHz，SMOD=1，定时器T1方式2，且串行传输波特率为2400b/s；根据公式变换可得T1定时初值计算公式为：



根据以上条件可得初值为232。

（2）串行口按工作方式1进行串行数据通讯，其中发送采用查询方式，接收采用中断方式。代码如下：

#include <AT89X52.h>

unsigned char tmp; //存放接收数据

unsigned char send; //存放发送数据

void main(void)

{

 TMOD=0x20; //T1为方式2

 TH1=232; //波特率为2400b/s

 TL1=232;

 PCON = 0x80; //SMOD = 1

 IE=0x90; //开启串行口中断

 TR1=1;

 IP=0x10; //设置串行口中断为高优先级

 SCON=0x50; //设串口为方式1、允许串行口接收

 while(1)

{

SBUF=send; //发送数据

 while(!TI); //等待发送数据结束（数据发送完，TI由硬件置位）

TI=0; //TI复位

}

}

/\* 串行口的中断服务子程序 \*/

void serial(void) interrupt 4 using 1

{

 if(RI==1)

 {

 RI=0; //接收中断标志位复位

 tmp=SBUF; //接收发送端传送来的数据

 }

}

7-7 简述串口通信初始化步骤。

（1）确定定时器T1的工作方式——写TMOD寄存器。

（2）计算定时器T1的初值——装载初值。

（3）启动定时器T1——TR1。

（4）确定串口的工作方式——写SCON寄存器。

（5）使用串口中断方式时——开启中断源、确定中断优先级。

7-8 简述多机通信的过程。

（1）主、从机工作于方式2或方式3，主机置SM2=0，REN=1；从机置SM2=1，REN=1。

（2）主机置位TB8=1，向从机发送寻址地址帧，各从机因满足接收条件(SM2=1,RB8=1)，从而接收到主机发来的地址，并与本机地址进行比较。

（3）地址一致的从机（被寻址机）将SM2清0，并向主机返回地址，供主机核对。地址不一致的从机（未被寻址机）保持SM2=1。

（4）主机核对返回的地址，若与此前发出的地址一致则准备发送数据；若不一致则返回第2步重新发送地址帧。

（5）主机向从机发送数据，此时主机的TB8=0，只有被选中的那台从机能接收到该数据。其他从机则舍弃该数据。

（6）本次通信结束后，从机重新置SM2=1，等待下次通信。