#include<reg52.h> //包含头文件，一般情况不需要改动，头文件包含特殊功能寄存器的定义

//sbit IR=P3^2; //红外接口标志

/\*------------------------------------------------

全局变量声明

------------------------------------------------\*/

unsigned char code table[10]={0x3f,0x06,0x5b,0x4f,0x66,0x6d,0x7d,0x07,0x7f,0x6f};// 共阴数码管 显示段码值0~9

unsigned char irtime;//红外用全局变量

bit irpro\_ok,irok; //处理OK，接收OK

unsigned char IRcord[4];

unsigned char irdata[33];

/\*------------------------------------------------

定时器0中断处理

------------------------------------------------\*/

void tim0\_isr (void) interrupt 1 using 1

{

irtime++; //用于计数2个下降沿之间的时间

}

/\*------------------------------------------------

外部中断0中断处理

------------------------------------------------\*/

void EX0\_ISR(void) interrupt 0 //外部中断0服务函数

{

static unsigned char i; //接收红外信号处理

static bit startflag; //是否开始处理标志位

if(startflag)

{

if(irtime<63&&irtime>=33)//引导码 TC9012的头码，9ms+4.5ms 头码的持续时间的范围，33(8.448ms)<63(16.128ms)，

i=0;

irdata[i]=irtime;//存储每个电平的持续时间，用于以后判断是0还是1

irtime=0;

i++;

if(i==33) //33是33位的意思，包括32位的数据和一位的头码。

{

irok=1;

i=0;

}

}

else

{

irtime=0;

startflag=1;

}

}

/\*------------------------------------------------

定时器0初始化

------------------------------------------------\*/

void TIM0init(void)//定时器0初始化

{

TMOD=0x02;//定时器0工作方式2，TH0是重装值，TL0是初值

TH0=0x00; //装初值

TL0=0x00;

ET0=1; //开中断

TR0=1;

}

/\*------------------------------------------------

外部中断0初始化

------------------------------------------------\*/

void EX0init(void)

{

IT0 = 1; //指定外部中断0下降沿触发，INT0 (P3.2)

EX0 = 1; //打开外部中断

EA = 1; //开总中断

}

/\*------------------------------------------------

键值处理

------------------------------------------------\*/

void Ir\_work(void)//红外键值散转程序

{

switch(IRcord[2])//判断第三个数码前两个是用户码，最后一个是反码，第三个才是真正的数据码。

{

case 0x16:P0=table[0];break;//0 按下遥控器上面0-9的按键，数码管显示相应的按键值

case 0x0c:P0=table[1];break;//1

case 0x18:P0=table[2];break;//2

case 0x5e:P0=table[3];break;//3

case 0x08:P0=table[4];break;//4

case 0x1c:P0=table[5];break;//5

case 0x5a:P0=table[6];break;//6

case 0x42:P0=table[7];break;//7

case 0x52:P0=table[8];break;//8

case 0x4a:P0=table[9];break;//9

default:break;

}

irpro\_ok=0;//处理完成标志

}

/\*------------------------------------------------

红外码值处理

------------------------------------------------\*/

void Irpro(void)//红外码值处理函数 分析出那些是1那些是0

{

unsigned char i, j, k;

unsigned char cord,value;

k=1; //前导码没数据，从第二个开始，就是用户码开始。

for(i=0;i<4;i++) //处理4个字节

{

for(j=1;j<=8;j++) //处理1个字节8位

{

cord=irdata[k]; //

if(cord>7) //低电平下降沿到下一个下降沿的宽度是0.56+0.565=1.125ms，高电平则是0.56+1.69=2.25ms，同样我们也给出一个范围用于区分它们，可以这样识别 （1.125ms + 2.25ms ）/ 2=1.68ms,

//大于1.68为高，小于1.68为低。

//假设使用12M晶振 定时器的单位数值时1us，使用8位定时器自动重装，将得到每个定时周期0.256ms的时长， 1.68/0.256=6.59，约等于7.也就是 定时器0计数次数。

value|=0x80;//最高位就给1，

if(j<8)

{

value>>=1;

}

k++;

}

IRcord[i]=value;

value=0;

}

irpro\_ok=1;//处理完毕标志位置1

}

/\*------------------------------------------------

主函数

------------------------------------------------\*/

void main(void)

{

EX0init(); //初始化外部中断

TIM0init();//初始化定时器

P1=0x00;; //取位码 第一位数码管选通，即二进制1111 1110

while(1)//主循环

{

if(irok) //如果接收好了进行红外处理

{

Irpro();

irok=0;

}

if(irpro\_ok) //如果处理好后进行工作处理，如按对应的按键后显示对应的数字等

{

Ir\_work();

}

}

}