#include<at89s52.h>

#include<intrins.h>// for using \_nop\_() function

sbit dig\_ctrl\_1=P0^0; // Control pins for the seven segments

sbit dig\_ctrl\_2=P0^1;

sbit dig\_ctrl\_3=P0^2;

sbit dig\_ctrl\_4=P0^3;

sbit sw1=P1^0;

sbit sw2=P1^1;

sbit sw3=P1^2;

sbit led1=P3^7;

unsigned char dig\_disp=0,flag = 1,flag1 = 1,flag2 = 1;

unsigned char dd1=2,dd2=2,td1=3,td2=4;

unsigned char cnt=0,ms=0,ss=0,mm=0,num1=0,num2=0;

char digi\_val[10]={0x40,0xF9,0x24,0x30,0x19,0x12,0x02,0xF8,0x00,0x10};

void msdelay(unsigned int itime)

{

unsigned int i,j ;

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

for(j=0;j<1;j++);

}

void display() interrupt 1 // Function to display the four digit number using multiplexing on seven segment. It uses Timer 0 interrupt to display the four digits one by one after a time delay of 2.5 milli second

{

TL0=0x36;

TH0=0xf6;

P2=0xFF;

dig\_ctrl\_1 = dig\_ctrl\_3 = dig\_ctrl\_2 = dig\_ctrl\_4 = 0;

dig\_disp++;

dig\_disp=dig\_disp%6;

if(flag2 == 1)

{

dd1 = ss/10;

dd2 = ss%10;

td1 = ms/10;

td2 = ms%10;

}

else

{

dd1 = mm/10;

dd2 = mm%10;

td1 = ss/10;

td2 = ss%10;

}

switch(dig\_disp)

{

case 0:

P2=digi\_val[dd1];

dig\_ctrl\_1 = 1;

break;

case 1:

P2= digi\_val[dd2];

dig\_ctrl\_2 = 1;

break;

case 2:

P2= digi\_val[td1];

dig\_ctrl\_3 = 1;

break;

case 3:

P2= digi\_val[td2];

dig\_ctrl\_4 = 1;

break;

}

}

void main()

{

P2=0xFF;

TMOD=0x11; // Intialize Timer 0

TL0=0x36;

TH0=0xF6;

// TH1=0xFC; // initial values loaded to timer

// TL1=0x00;

IE=0x82;

TR0=1; //Start timer0

// TR1=1; //Start timer1

P1 = 0xFF;

while(1) // Forward counting

{

if(flag == 1)

{

\_nop\_();\_nop\_();\_nop\_();\_nop\_();\_nop\_();

\_nop\_();\_nop\_();\_nop\_();\_nop\_();\_nop\_();

\_nop\_();\_nop\_();

// msdelay(1);

cnt++;

if(cnt == 10)

ms++;

if(ms == 100)

{

ms = 0;

ss++;

}

if(ss == 60)

{

ss = 0;

mm ++;

}

if( mm == 60)

mm = 0;

}

if( sw3 == 0 )

{

if(flag1 ==1 )

flag = flag1 = 0;

else

flag = flag1= 1; msdelay(15);

}

if(sw1 == 0)

{

flag2 = 1;

}

if(sw2 == 0)

{

flag2 = 0;

}

}

}