

**Lab Exercises:**

**3. Write a C program to simulate 4-digit BCD up/down counter on the multiplexed seven-segment display.**

**Program:**

```
#include <LPC17xx.h>
#include <stdio.h>
#define FIRST_SEG 0xF87FFFFF
#define SECOND_SEG 0xF8FFFFFF
#define THIRD_SEG 0xF97FFFFF
#define FOURTH_SEG 0xF9FFFFFF
#define DISABLE_ALL 0xFA7FFFFF

unsigned int dig1=0x00,dig2=0x00,dig3=0x00,dig4=0x00;
unsigned int twenty_count=0x00,dig_count=0x00,temp1=0x00;
unsigned char dec[10]={0x3F,0x06,0x5B,0x4F,0x66,0x6D,0x7D,0x07,0x7F,0x6F};
unsigned char tm0_flag=0x00,one_s_f=0x00;
unsigned long int temp2=0x00000000,i=0;
unsigned int temp3=0x00;
void delay(void);
void display(void);

int main(void)
{
    LPC_PINCON->PINSEL0&=0xFF0000FF;
    LPC_PINCON->PINSEL3&=0xFFC03FFF;
    LPC_GPIO0->FIODIR|=0X00000FF0;
    LPC_GPIO1->FIODIR|=0X07800000;
    LPC_GPIO2->FIODIR = 0x0;
    while(1)
    {
        delay();
        dig_count+=1;
        if(dig_count==0x05)
        {
            dig_count=0x00;
            one_s_f=0xFF;
        }
        if(one_s_f==0xFF)
        {
            one_s_f=0x00;
            if((LPC_GPIO2->FIOPIN & 1))
            {
                dig1+=1;
                if(dig1==0x0A)
                {
```

```

        dig1=0;
        dig2+=1;
        if(dig2==0x0A)
        {
            dig2=0;
            dig3+=1;
            if(dig3==0x0A)
            {
                dig3=0;
                dig4+=1;
                if(dig4==0x0A)
                {
                    dig4=0;
                }
            }
        }
    }
    else
    {
        dig1 -= 1;
        if(dig1==0x0)
        {
            dig1=0x9;
            dig2-=1;
            if(dig2==0x0)
            {
                dig2=0x9;
                dig3-=1;
                if(dig3==0x0)
                {
                    dig3=0x9;
                    dig4-=1;
                    if(dig4==0x0)
                    {
                        dig4=0x9;
                        dig3=0x9;
                        dig2=0x9;
                        dig1=0x9;
                    }
                }
            }
        }
    }
}
display();
}
}

```

```

void display(void)
{
    if(dig_count==0x01)
    {
        temp1=dig1;
        LPC_GPIO1->FIOPIN=FIRST_SEG;
    }

    else if(dig_count==0x02)
    {
        temp1=dig2;
        LPC_GPIO1->FIOPIN=SECOND_SEG;
    }

    else if(dig_count==0x03)
    {
        temp1=dig3;
        LPC_GPIO1->FIOPIN=THIRD_SEG;
    }

    else if(dig_count==0x04)
    {
        temp1=dig4;
        LPC_GPIO1->FIOPIN=FOURTH_SEG;
    }

    temp1 &= 0x0f;
    temp2 = dec[temp1];
    temp2 = temp2<<4;
    LPC_GPIO0->FIOPIN = temp2;

    for(i=0;i<500;i++);

    LPC_GPIO0->FIOCLR = 0xff0;
    LPC_GPIO1->FIOPIN = DISABLE_ALL;
}

```

```

void delay(void)
{
    unsigned int i;

    for(i=0;i<1000;i++);

    if(twenty_count==1000)
    {
        one_s_f=0xff;
        twenty_count=0x00;
    }

    else

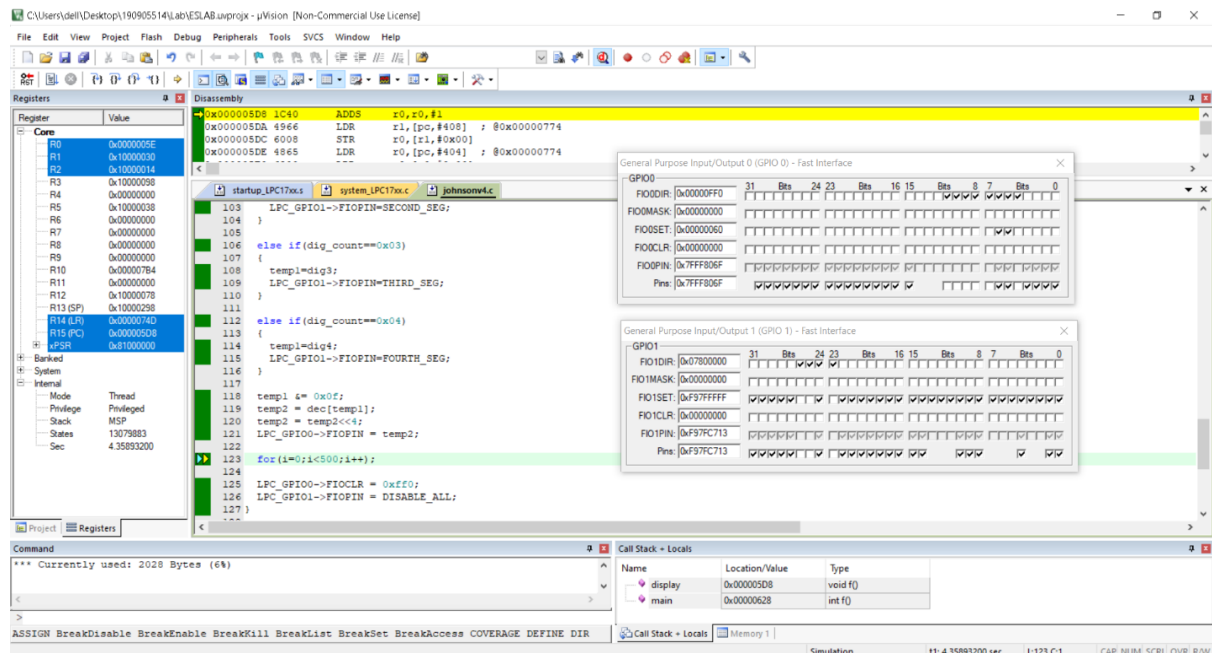
```

```

    {
        twenty_count += 1;
    }
}

```

## Output:



## 4. Write a C program for 4-digit hexadecimal up/down counter on seven segment using a switch and timer with a delay of 1 second between each count.

### Program:

```

#include <LPC17xx.h>
#include <stdio.h>

```

```

#define FIRST_SEG 0XF87FFFFFFF //0<<23
#define SECOND_SEG 0XF8FFFFFFF //1<<23
#define THIRD_SEG 0XF97FFFFFFF //2<<23
#define FOURTH_SEG 0XF9FFFFFFF //3<<23

```

```

unsigned int dig1=0x00,dig2=0x00,dig3=0x00,dig4=0x00;
unsigned int digcount=0x00,temp1=0x00,onesecflag=0x00;
unsigned int
arraydec[16]={0x3F,0x06,0x5B,0x4F,0x66,0x6D,0x7D,0x07,0x7F,0x6F,0x77,0x7C,0x39,0x
5E,0x79,0x71};
unsigned long int temp2=0x00000000,i=0,j=0;

```

```

void delay(void);
void display(void);
void downc(void);
void upc(void);

```

```

int main(void)

```

```

{
    SystemInit();
    SystemCoreClockUpdate();

    LPC_PINCON->PINSEL0 &=0xFF0000FF;
    LPC_PINCON->PINSEL3 &=0xFFC03FFF;
    LPC_PINCON->PINSEL4 &=0XFCFFFFFF;
    LPC_GPIO0->FIODIR |=0x00000FF0;
    LPC_GPIO1->FIODIR |=15<<23;
    LPC_PINCON->PINSEL4 &=0xFFFFFFFFD;
    while(1)
    {
        if(LPC_GPIO2->FIOPIN &0x01)
            upc();
        else
            downc();
    }
}

```

```

void upc(void)
{
    while(LPC_GPIO2->FIOPIN &0x01)
    {
        delay();
        digcount+=1;
        if(digcount==0x05)
        {
            digcount=0x01;
            for( j=0;j<500;j++)
            {
                display();
                if(digcount==0x05)
                    digcount=0x01;
                else
                    digcount++;
            }
            digcount=0x00;
            oneseclflag=0xFF;
        }
        if(onesecflag==0xFF)
        {
            oneseclflag=0x00;
            dig1+=1;
            if(dig1==0x10)
            {
                dig1=0;
                dig2+=1;
                if(dig2==0x10)

```

```

        {
            dig2=0;
            dig3+=1;
            if(dig3==0x10)
            {
                dig3=0;
                dig4+=1;
                if(dig4==0x10)
                {
                    dig4=0;
                }
            }
        }
    }
    display();
}
}

```

```

void downc(void)
{
    while((LPC_GPIO2->FIOPIN &0x01)==0x0)
    {
        delay();
        digcount+=1;
        if(digcount==0x05)
        {
            digcount=0x01;
            for( j=0;j<500;j++)
            {
                display();
                if(digcount==0x05)
                    digcount=0x01;
                else
                    digcount++;
            }
            digcount=0x00;
            oneseclflag=0xFF;
        }
        if(onesecflag==0xFF)
        {
            oneseclflag=0x00;

            if(dig1==0x00)
            {
                dig1=9;
                if(dig2==0x00)
                {

```

```

        dig2=9;

        if(dig3==0x00)
        {
            dig3=9;

            if(dig4==0x00)
            {
                dig4=9;
            }
            else dig4-=1;
        }
        else dig3-=1;
    }
    else dig2-=1;
}
else dig1-=1;
}
display();
}
}

```

```

void display(void)
{
    if(digcount==0x01)
    {
        temp1=dig1;
        LPC_GPIO1->FIOPIN=FIRST_SEG;
    }

    else if(digcount==0x02)
    {
        temp1=dig2;
        LPC_GPIO1->FIOPIN=SECOND_SEG;
    }

    else if(digcount==0x03)
    {
        temp1=dig3;
        LPC_GPIO1->FIOPIN=THIRD_SEG;
    }

    else if(digcount==0x04)
    {
        temp1=dig4;
        LPC_GPIO1->FIOPIN=FOURTH_SEG;
    }

    temp1&=0x0F;
    temp2=arraydec[temp1];
}

```

```
temp2=temp2<<4;
LPC_GPIO0->FIOPIN=temp2;

for(i=0;i<500;i++);

//LPC_GPIO0->FIOCLR = 0x00000FF0;
}
```

```
void delay(void)
{
    unsigned int i;
    for(i=0;i<5000;i++);
}
```

## Output:

The screenshot shows the uVision IDE interface. The main window displays the disassembly of an ARM assembly program. The assembly code includes comments in Chinese and instructions for setting up GPIO pins. Two peripheral configuration windows are open, showing the configuration for GPIO0 and GPIO1. The GPIO0 window shows the FIODIR, FIOSET, FIOCLR, and FIOPIN registers. The GPIO1 window shows the FIODIR, FIOSET, FIOCLR, and FIOPIN registers. The bottom status bar shows the simulation time as 11: 4.35893200 sec.