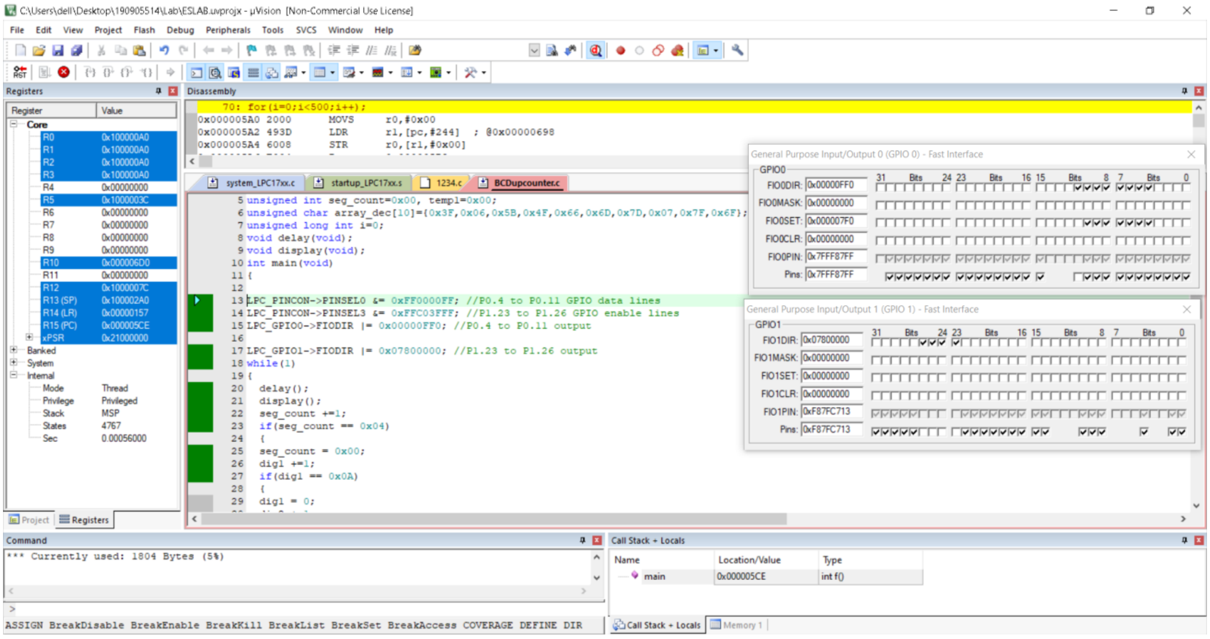
Solved Exercise:

Write a program to simulate 4-digit BCD up counter on the multiplexed seven segment display.

```
#include <LPC17xx.h>
#include <stdio.h>
unsigned int seg_select[4] = {0<<23, 1<<23, 2<<23, 3<<23};
unsigned int dig1=0x00, dig2=0x00, dig3=0x00, dig4=0x00;
unsigned int seg_count=0x00, temp1=0x00;
unsigned char array_dec[10]={0x3F,0x06,0x5B,0x4F,0x66,0x6D,0x7D,0x07,0x7F,0x6F};
unsigned long int i=0;
void delay(void);
void display(void);
int main(void)
{

LPC_PINCON->PINSEL0 &= 0xFF0000FF; //P0.4 to P0.11 GPIO data lines
LPC_PINCON->PINSEL3 &= 0xFFC03FFF; //P1.23 to P1.26 GPIO enable lines
LPC_GPIO0->FIODIR |= 0x00000FF0; //P0.4 to P0.11 output

LPC_GPIO1->FIODIR |= 0x07800000; //P1.23 to P1.26 output
while(1)
{
        delay();
        display();
        seg_count +=1;
        if(seg_count == 0x04)
        {
        seg_count = 0x00;
        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;
        } //end of dig4
```

```
		} //end of dig3
		} //end of dig2
		} //end of dig1
		} //end of seg_count
		}	//end of while(1)

		}//end of main
void display(void) //To Display on 7-segments
{
LPC_GPIO1->FIOPIN = seg_select[seg_count];
if(seg_count == 0x00) // For Segment U8
{
temp1 = dig1;
}
else if(seg_count == 0x01) // For Segment U9
{
temp1 = dig2;
}
else if(seg_count == 0x02) // For Segment U10
{
temp1 = dig3;
}
else if(seg_count == 0x03) // For Segment U11
{
temp1 = dig4;
}
LPC_GPIO0->FIOPIN = array_dec[temp1]<<4; // Taking Data Lines for 7-Seg
for(i=0;i<500;i++);
}
void delay(void)
{ unsigned int i;
for(i=0;i<60000;i++);
}
```

Output:


C:\Users\dell\Desktop\190905514\Lab\ESLAB.uvprojx - µVision [Non-Commercial Use License]
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Registers
Disassembly
system_LPC17xx.c
startup_LPC17xx.s
1234.c
BCDupcounter.c
Project
Registers
Command
Call Stack + Locals
Memory 1

Register	Value
Core	
R0	0x100000A0
R1	0x100000A0
R2	0x100000A0
R3	0x100000A0
R4	0x00000000
R5	0x1000003C
R6	0x00000000
R7	0x00000000
R8	0x00000000
R9	0x00000000
R10	0x000006D0
R11	0x00000000
R12	0x1000007C
R13 (SP)	0x100002A0
R14 (LR)	0x00000157
R15 (PC)	0x000005CE
xPSR	0x21000000
Banked	
System	
Internal	
Mode	Thread
Privilege	Privileged
Stack	MSP
States	4767
Sec	0.00056000

```
    70: for(i=0;i<500;i++);
0x000005A0 2000      MOVS     r0,#0x00
0x000005A2 493D      LDR      r1,[pc,#244]  ; @0x00000698
0x000005A4 6008      STR      r0,[r1,#0x00]
```

```
unsigned int seg_count=0x00, templ=0x00;
unsigned char array_dec[10]={0x3F,0x06,0x5B,0x4F,0x66,0x6D,0x7D,0x07,0x7F,0x6F};
unsigned long int i=0;
void delay(void);
void display(void);
int main(void)
{

LPC_PINCON->PINSEL0 &= 0xFF0000FF; //P0.4 to P0.11 GPIO data lines
LPC_PINCON->PINSEL3 &= 0xFFC03FFF; //P1.23 to P1.26 GPIO enable lines
LPC_GPIO0->FIODIR |= 0x00000FF0; //P0.4 to P0.11 output

LPC_GPIO1->FIODIR |= 0x07800000; //P1.23 to P1.26 output
while(1)
{
  delay();
  display();
  seg_count +=1;
  if(seg_count == 0x04)
  {
  seg_count = 0x00;
  digl +=1;
  if(digl == 0x0A)
  {
  digl = 0;
```

General Purpose Input/Output 0 (GPIO 0) - Fast Interface

GPIO0

FIO0DIR: 0x00000FF0
FIO0MASK: 0x00000000
FIO0SET: 0x000007F0
FIO0CLR: 0x00000000
FIO0PIN: 0x7FFF87FF
Pins: 0x7FFF87FF

General Purpose Input/Output 1 (GPIO 1) - Fast Interface

GPIO1

FIO1DIR: 0x07800000
FIO1MASK: 0x00000000
FIO1SET: 0x00000000
FIO1CLR: 0x00000000
FIO1PIN: 0xF87FC713
Pins: 0xF87FC713

```
*** Currently used: 1804 Bytes (5%)
>
ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet BreakAccess COVERAGE DEFINE DIR
```

Name	Location/Value	Type
main	0x000005CE	int f()

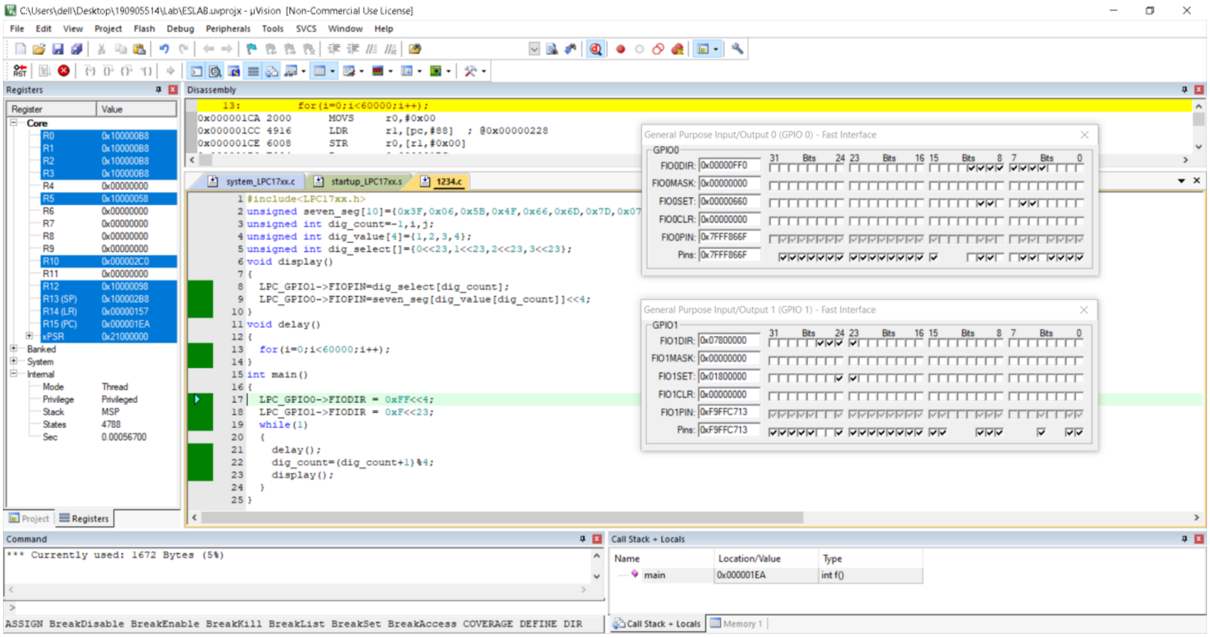
Lab Exercises:

1. Write a C program to display the number “1234” serially in the seven segment display.

Program:

```
#include<LPC17xx.h>
unsigned seven_seg[10]={0x3F,0x06,0x5B,0x4F,0x66,0x6D,0x7D,0x07,0x7F,0x6F};
unsigned int dig_count=-1,i,j;
unsigned int dig_value[4]={1,2,3,4};
unsigned int dig_select[]={0<<23,1<<23,2<<23,3<<23};
void display()
{
        LPC_GPIO1->FIOPIN=dig_select[dig_count];
        LPC_GPIO0->FIOPIN=seven_seg[dig_value[dig_count]]<<4;
}
void delay()
{
        for(i=0;i<60000;i++);
}
int main()
{
        LPC_GPIO0->FIODIR = 0xFF<<4;
        LPC_GPIO1->FIODIR = 0xF<<23;
        while(1)
        {
  delay();
                dig_count=(dig_count+1)%4;
                display();
        }
}
```

Output:


C:\Users\dell\Desktop\190905514\Lab\ESLAB.uvprojx - µVision [Non-Commercial Use License]
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
Registers
Disassembly
system_LPC17xx.c
startup_LPC17xx.s
1234.c
Project
Registers
Command
Call Stack + Locals
Memory 1

Register	Value
Core	
R0	0x100000B8
R1	0x100000B8
R2	0x100000B8
R3	0x100000B8
R4	0x00000000
R5	0x10000058
R6	0x00000000
R7	0x00000000
R8	0x00000000
R9	0x00000000
R10	0x000002C0
R11	0x00000000
R12	0x10000098
R13 (SP)	0x100002B8
R14 (LR)	0x00000157
R15 (PC)	0x000001EA
xPSR	0x21000000
Banked	
System	
Internal	
Mode	Thread
Privilege	Privileged
Stack	MSP
States	4788
Sec	0.00056700

```
    13:          for(i=0;i<60000;i++);
0x000001CA 2000      MOVS     r0,#0x00
0x000001CC 4916      LDR      r1,[pc,#88]  ; @0x00000228
0x000001CE 6008      STR      r0,[r1,#0x00]
```

```
#include<LPC17xx.h>
unsigned seven_seg[10]={0x3F,0x06,0x5B,0x4F,0x66,0x6D,0x7D,0x07
unsigned int dig_count=-1,i,j;
unsigned int dig_value[4]={1,2,3,4};
unsigned int dig_select[]={0<<23,1<<23,2<<23,3<<23};
void display()
{
  LPC_GPIO1->FIOPIN=dig_select[dig_count];
  LPC_GPIO0->FIOPIN=seven_seg[dig_value[dig_count]]<<4;
}
void delay()
{
  for(i=0;i<60000;i++);
}
int main()
{
  LPC_GPIO0->FIODIR = 0xFF<<4;
  LPC_GPIO1->FIODIR = 0xF<<23;
  while(1)
  {
    delay();
    dig_count=(dig_count+1)%4;
    display();
  }
}
```

General Purpose Input/Output 0 (GPIO 0) - Fast Interface

GPIO0

FIO0DIR:	0x00000FF0
FIO0MASK:	0x00000000
FIO0SET:	0x00000660
FIO0CLR:	0x00000000
FIO0PIN:	0x7FFF866F
Pins:	0x7FFF866F

General Purpose Input/Output 1 (GPIO 1) - Fast Interface

GPIO1

FIO1DIR:	0x07800000
FIO1MASK:	0x00000000
FIO1SET:	0x01800000
FIO1CLR:	0x00000000
FIO1PIN:	0xF9FFC713
Pins:	0xF9FFC713

```
*** Currently used: 1672 Bytes (5%)
>
ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet BreakAccess COVERAGE DEFINE DIR
```

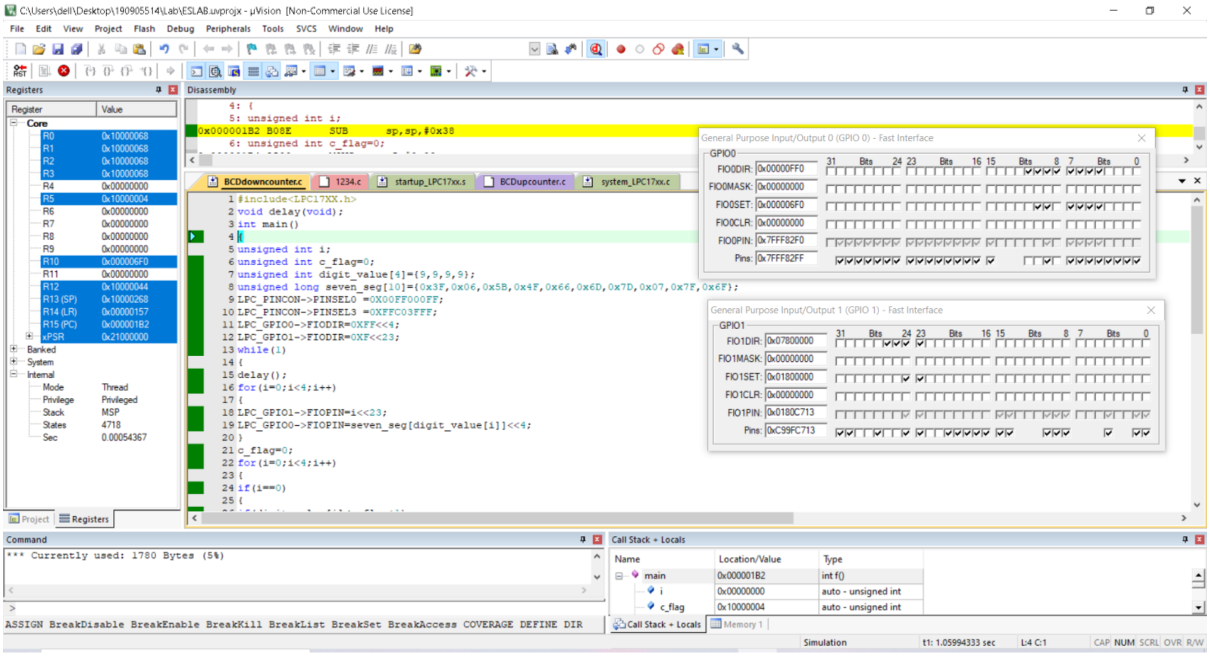
Name	Location/Value	Type
main	0x000001EA	int f()

2. Write a C program to simulate a 4-digit BCD down counter.

Program:

```
#include<LPC17XX.h>
void delay(void);
int main()
{
unsigned int i;
unsigned int c_flag=0;
unsigned int digit_value[4]={9,9,9,9};
unsigned long seven_seg[10]={0x3F,0x06,0x5B,0x4F,0x66,0x6D,0x7D,0x07,0x7F,0x6F};
LPC_PINCON->PINSEL0 =0X00FF000FF;
LPC_PINCON->PINSEL3 =0XFFC03FFF;
LPC_GPIO0->FIODIR=0XFF<<4;
LPC_GPIO1->FIODIR=0XF<<23;
while(1)
{
delay();
for(i=0;i<4;i++)
{
LPC_GPIO1->FIOPIN=i<<23;
LPC_GPIO0->FIOPIN=seven_seg[digit_value[i]]<<4;
}
c_flag=0;
for(i=0;i<4;i++)
{
if(i==0)
{
if(digit_value[i]<c_flag+1)
{
digit_value[i]=digit_value[i]-1-c_flag+10;
c_flag=1;
}
else
{
digit_value[i]=digit_value[i]-1-c_flag;
c_flag=0;
}
}
else
{
if(digit_value[i]<c_flag)
{
digit_value[i]=digit_value[i]-c_flag+10;
c_flag=1;
}
else
{
digit_value[i]=digit_value[i]-c_flag;
c_flag=0;
}
}
}
}
return 2;
}
void delay(void)
{ unsigned int i;
for(i=0;i<1000;i++);
}
```

Output:



Register	Value
Core	
R0	0x10000068
R1	0x10000068
R2	0x10000068
R3	0x10000068
R4	0x00000000
R5	0x10000004
R6	0x00000000
R7	0x00000000
R8	0x00000000
R9	0x00000000
R10	0x000006F0
R11	0x00000000
R12	0x10000044
R13 (SP)	0x10000268
R14 (LR)	0x00000157
R15 (PC)	0x000001B2
xPSR	0x21000000
Banked	
System	
Internal	
Mode	Thread
Privilege	Privileged
Stack	MSP
States	4718
Sec	0.00054367

```
     4: {
     5: unsigned int i;
0x000001B2 B08E      SUB      sp,sp,#0x38
     6: unsigned int c_flag=0;
```

```
#include<LPC17XX.h>
void delay(void);
int main()
{
unsigned int i;
unsigned int c_flag=0;
unsigned int digit_value[4]={9,9,9,9};
unsigned long seven_seg[10]={0x3F,0x06,0x5B,0x4F,0x66,0x6D,0x7D,0x07,0x7F,0x6F};
LPC_PINCON->PINSEL0 =0X00FF000FF;
LPC_PINCON->PINSEL3 =0XFFC03FFF;
LPC_GPIO0->FIODIR=0XFF<<4;
LPC_GPIO1->FIODIR=0XF<<23;
while(1)
{
delay();
for(i=0;i<4;i++)
{
LPC_GPIO1->FIOPIN=i<<23;
LPC_GPIO0->FIOPIN=seven_seg[digit_value[i]]<<4;
}
c_flag=0;
for(i=0;i<4;i++)
{
if(i==0)
{
```

GPIO0
FIO0DIR: 0x00000FF0
FIO0MASK: 0x00000000
FIO0SET: 0x000006F0
FIO0CLR: 0x00000000
FIO0PIN: 0x7FFF82F0
Pins: 0x7FFF82FF

GPIO1
FIO1DIR: 0x07800000
FIO1MASK: 0x00000000
FIO1SET: 0x01800000
FIO1CLR: 0x00000000
FIO1PIN: 0x0180C713
Pins: 0xC99FC713

```
*** Currently used: 1780 Bytes (5%)
>
ASSIGN BreakDisable BreakEnable BreakKill BreakList BreakSet BreakAccess COVERAGE DEFINE DIR
```

Name	Location/Value	Type
main	0x000001B2	int f()
i	0x00000000	auto - unsigned int
c_flag	0x10000004	auto - unsigned int