Embedded Systems LAB 6 12.02.2025

Solved exercise

Write a program to turn on/off the LEDs serially.

Code

```
#include <LPC17xx.h>
unsigned int i,j;
unsigned long LED = 0x00000010;
int main(void)
{
 SystemInit(); //Add these two function for its internal operation
 SystemCoreClockUpdate();
 LPC_PINCON->PINSEL0 &= 0xFF0000FF;
 //Configure Port0 PINS P0.4-P0.11 as GPIO function
 LPC_GPIO0->FIODIR |= 0x00000FF0;
//Configure P0.4-P0.11 as output port
 while(1)
 {
LED = 0x00000010;
for(i=1;i<9;i++) //On the LED's serially
{
 LPC_GPIO0->FIOSET = LED;
```

```
 // Turn ON LED at LSB (LED connected to p0.4)
for(j=0;j<100000;j++);
 LED <<= 1;
} //loop for 8 times
LED = 0x00000010;
for(i=1;i<9;i++) //Off the LED's serially
{
LPC_GPIO0->FIOCLR = LED;
 //Turn OFF LED at LSB (LED connected to p0.4)
for(j=0;j<10000;j++);
LED <<= 1;
}
}
}
```

Output

We saw it on LPC1768 on ‘General Purpose LED’.

Q1 Write a C program to display an 8 bit binary up counter on the LEDs.

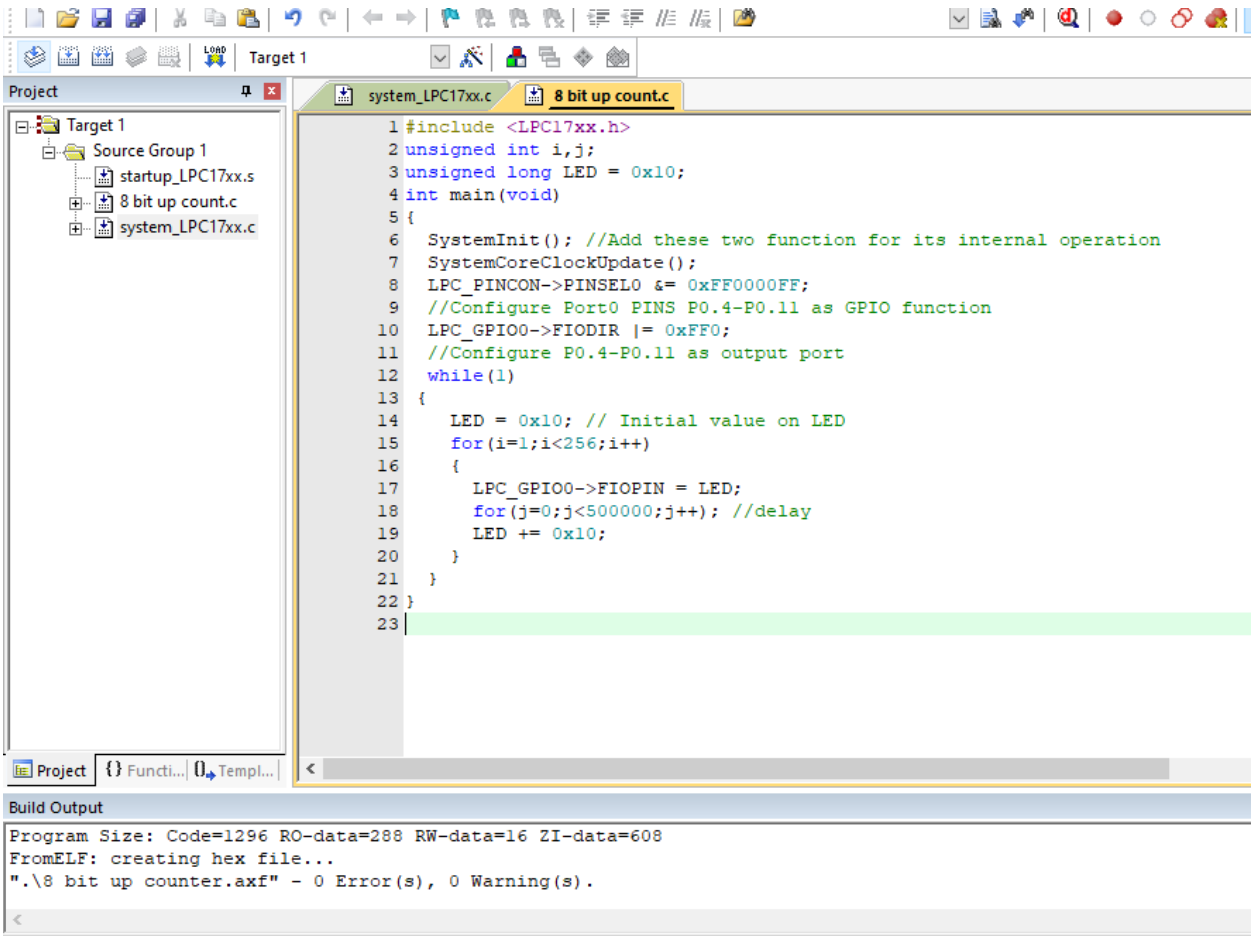
Code

```
#include <LPC17xx.h>
```

```
unsigned int i,j;
unsigned long LED = 0x10;
int main(void)
{
	SystemInit(); //Add these two function for its internal operation
	SystemCoreClockUpdate();
	LPC_PINCON->PINSEL0 &= 0xFF0000FF;
	//Configure Port0 PINS P0.4-P0.11 as GPIO function
	LPC_GPIO0->FIODIR |= 0xFF0;
	//Configure P0.4-P0.11 as output port
	while(1)
 {
		LED = 0x10; // Initial value on LED
		for(i=1;i<256;i++)
		{
			LPC_GPIO0->FIOPIN = LED;
			for(j=0;j<500000;j++); //delay
			LED += 0x10;
		}
	}
}
```

Output

We saw binary up counter working on LPC1768 ‘General LED display’.



Q2 Write a C program to read a key and display an 8 bit up/down counter on the LEDs.

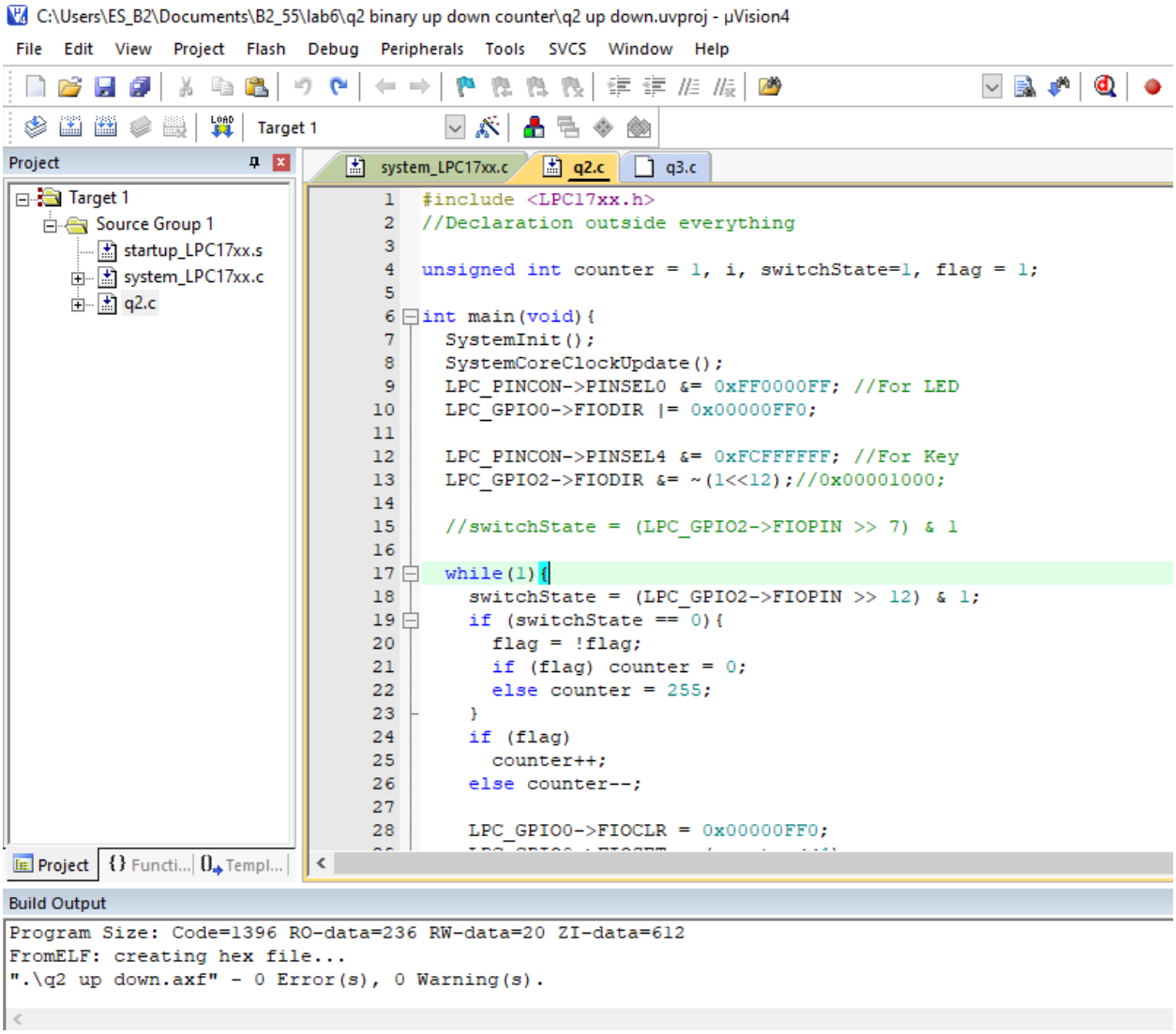
Code

```
#include <LPC17xx.h>

//Declaration outside everything
```

```
unsigned int counter = 1, i, switchState=1, flag = 1;

int main(void){
	SystemInit();
	SystemCoreClockUpdate();
	LPC_PINCON->PINSEL0 &= 0xFF0000FF; //For LED
	LPC_GPIO0->FIODIR |= 0x00000FF0;

	LPC_PINCON->PINSEL4 &= 0xFCFFFFFF; //For Key
	LPC_GPIO2->FIODIR &= ~(1<<12);//0x00001000;

	//switchState = (LPC_GPIO2->FIOPIN >> 7) & 1

	while(1){
		switchState = (LPC_GPIO2->FIOPIN >> 12) & 1;
		if (switchState == 0){
			flag = !flag;
			if (flag) counter = 0;
			else counter = 255;
		}
		if (flag)
			counter++;
		else counter--;
```

```
            LPC_GPIO0->FIOCLR = 0x00000FF0;
            LPC_GPIO0->FIOSET = (counter<<4);
            for (i=0; i<100000; i++); //delay

            if (counter>0xFF || counter==0)
                  break;
      }
}
```

Output



Here we observe that we press ‘SW2’ key two times then it starts decrementing, if we press it again then it starts upcount.

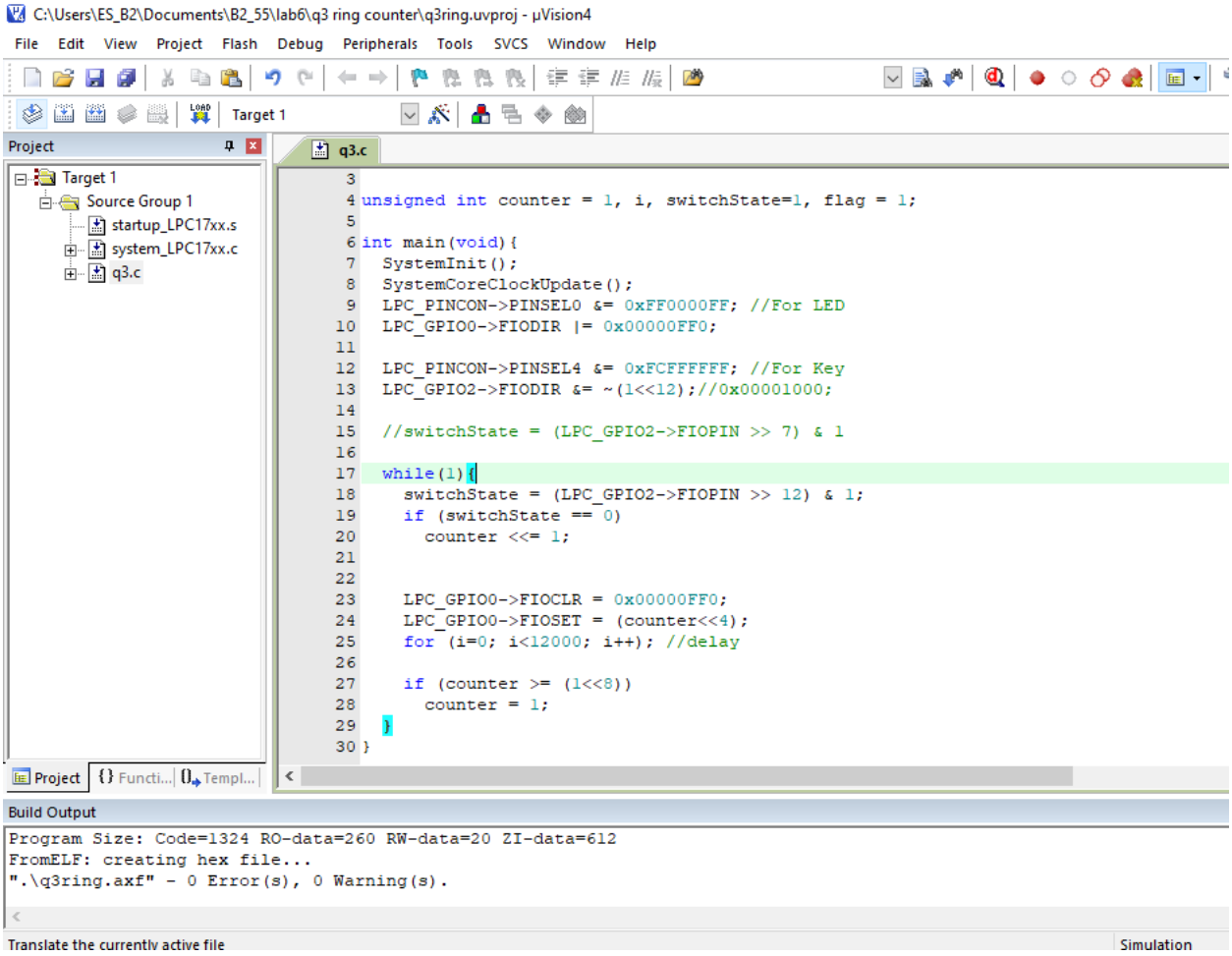
Q3 8 bit ring counter with key press SW2.

Code

```
#include <LPC17xx.h>
//Declaration outside everything

unsigned int counter = 1, i, switchState=1, flag = 1;

int main(void){
	SystemInit();
	SystemCoreClockUpdate();
	LPC_PINCON->PINSEL0 &= 0xFF0000FF; //For LED
	LPC_GPIO0->FIODIR |= 0x00000FF0;

	LPC_PINCON->PINSEL4 &= 0xFCFFFFFF; //For Key
	LPC_GPIO2->FIODIR &= ~(1<<12);//0x00001000;

	//switchState = (LPC_GPIO2->FIOPIN >> 7) & 1

	while(1){
		switchState = (LPC_GPIO2->FIOPIN >> 12) & 1;
		if (switchState == 0)
			counter <<= 1;
```

```
        LPC_GPIO0->FIOCLR = 0x00000FF0;
        LPC_GPIO0->FIOSET = (counter<<4);
        for (i=0; i<12000; i++); //delay

        if (counter >= (1<<8))
            counter = 1;
    }
}
```

Output



Here we observed that when ‘SW2’ key is pressed, the LED light moves to the next position, just like ring counter, hence simulating ring counter.