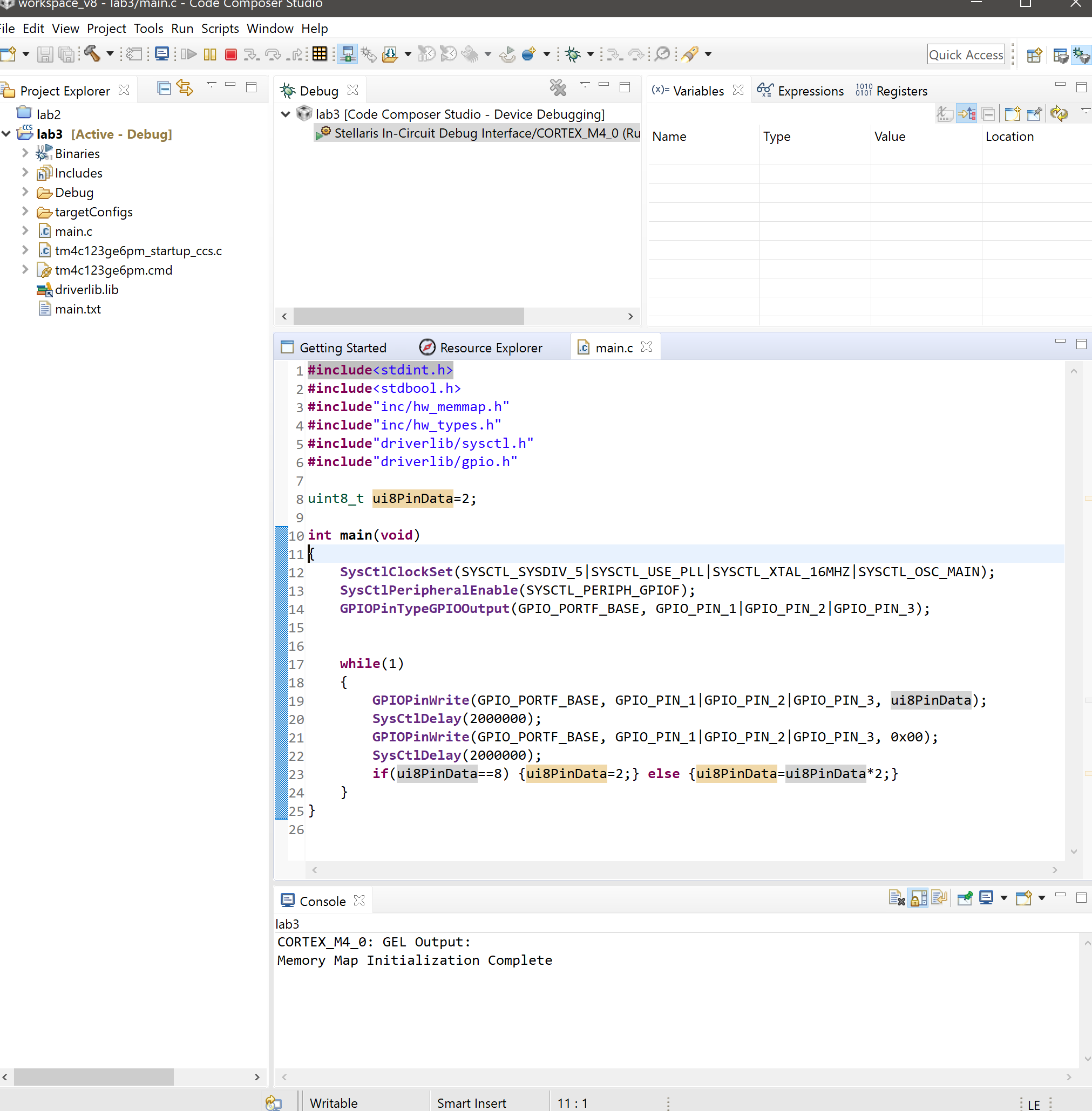
**Date Submitted: 10/1/2018 10:25 PM**

**Task 00: Execute provided code**

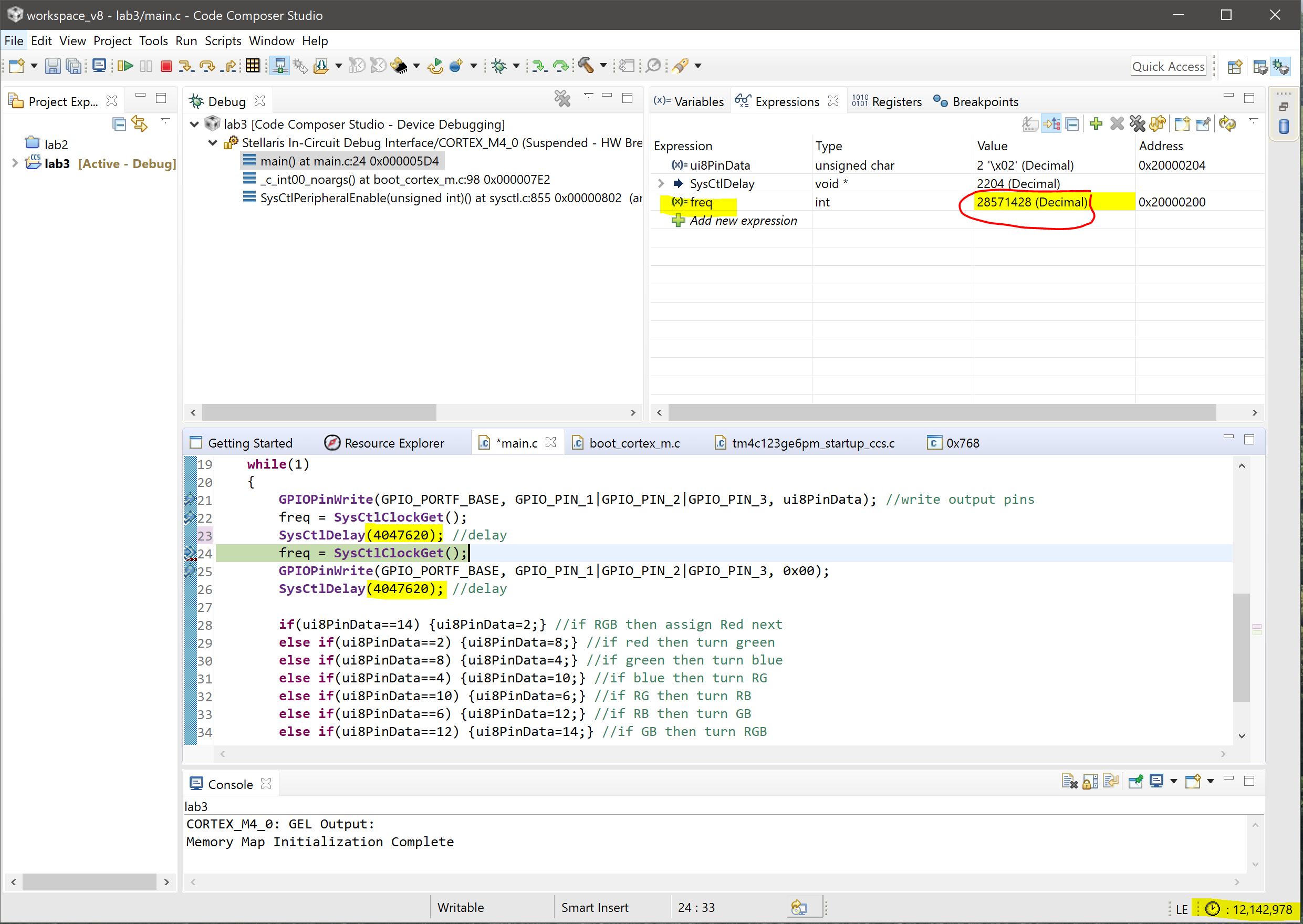


**Youtube Link: No Submission required for Task 00**

**------------------------------------------------------------------------------------**

**Task 01:** Determine the current period and on-time of the LED blinking. Change the delay of the LED blink (approx. 0.425 sec) by changing the delay and clock source and configuration –

Youtube Link: <https://www.youtube.com/watch?v=8mSPPsI1rqw>



Above is a screenshot calculating the delay to measure approximately 0.425 seconds. First I created breakpoints before and after the “SysCtDelay()” function. I preferred to have those two breakpoints also tell me what the clock frequency was. I used SYSCTL\_SYSDIV\_2\_5 instead of the normal SYSCTL\_OSC\_MAIN to change the clock source frequency. By using that the value that I have received using an int variable called “freq” and watching the expression I got 28.57 MHz as my clock source. Then I pressed Run->Clock->Enable & Reset to only measure ONE clock cycle from the beginning of the delay to the end of the execution of the delay. The clock cycle is shown on the bottom right corner, in this case 12,142,978 cycles were executed during the delay.

Here are the calculations:

Frequency: 28.57 MHz

Delay: 1/f -> 1/28.57 MHz = 35 ns

Delay for LED to blink = 35 ns \* 12,142,978 cycles = 0.425004 s (approx. 0.425s)

**Modified Code:**

**#include**<stdint.h>

**#include**<stdbool.h>

**#include**"inc/hw\_memmap.h"

**#include**"inc/hw\_types.h"

**#include**"driverlib/sysctl.h"

**#include**"driverlib/gpio.h"

uint8\_t ui8PinData=2;

**int** freq;

**int** **main**(**void**)

{

**SysCtlClockSet**(SYSCTL\_SYSDIV\_5|SYSCTL\_USE\_PLL|SYSCTL\_XTAL\_16MHZ|SYSCTL\_SYSDIV\_2\_5);

**SysCtlPeripheralEnable**(SYSCTL\_PERIPH\_GPIOF);

**GPIOPinTypeGPIOOutput**(GPIO\_PORTF\_BASE, GPIO\_PIN\_1|GPIO\_PIN\_2|GPIO\_PIN\_3);

**while**(1)

{

**GPIOPinWrite**(GPIO\_PORTF\_BASE, GPIO\_PIN\_1|GPIO\_PIN\_2|GPIO\_PIN\_3, ui8PinData); //write output pins

freq = **SysCtlClockGet**();

**SysCtlDelay**(4047620); //delay for 0.425s

freq = **SysCtlClockGet**();

**GPIOPinWrite**(GPIO\_PORTF\_BASE, GPIO\_PIN\_1|GPIO\_PIN\_2|GPIO\_PIN\_3, 0x00);

**SysCtlDelay**(4047620); //delay

**if**(ui8PinData==8) {ui8PinData=2;}

**else** {ui8PinData=ui8PinData\*2;}

}

**------------------------------------------------------------------------------------**

**Task 02:** Task 02: Change the a) sequence of LED blinking (from RGB sequence to BGR), and b) blink one LED, two LED, and three LED at an instance and with a sequence

Youtube Link A: <https://www.youtube.com/watch?v=ltwX5JI3Sxc>

Youtube Link B: <https://www.youtube.com/watch?v=oL3WZa6GhrQ>

Here this code has been implemented for both A and B using if statements to execute the next order of LED lights that need to be toggled on or off. This way there is a specific order for each LED to turn on. The codes for both a and b are written below.

**Modified Code A:**

**#include**<stdint.h>

**#include**<stdbool.h>

**#include**"inc/hw\_memmap.h"

**#include**"inc/hw\_types.h"

**#include**"driverlib/sysctl.h"

**#include**"driverlib/gpio.h"

uint8\_t ui8PinData=4; //set to blue

**int** freq;

**int** **main**(**void**)

{

**SysCtlClockSet**(SYSCTL\_SYSDIV\_5|SYSCTL\_USE\_PLL|SYSCTL\_XTAL\_16MHZ|SYSCTL\_SYSDIV\_2\_5);

**SysCtlPeripheralEnable**(SYSCTL\_PERIPH\_GPIOF);

**GPIOPinTypeGPIOOutput**(GPIO\_PORTF\_BASE, GPIO\_PIN\_1|GPIO\_PIN\_2|GPIO\_PIN\_3);

**while**(1)

{

**GPIOPinWrite**(GPIO\_PORTF\_BASE, GPIO\_PIN\_1|GPIO\_PIN\_2|GPIO\_PIN\_3, ui8PinData); //write output pins

freq = **SysCtlClockGet**();

**SysCtlDelay**(4047620); //delay

freq = **SysCtlClockGet**();

**GPIOPinWrite**(GPIO\_PORTF\_BASE, GPIO\_PIN\_1|GPIO\_PIN\_2|GPIO\_PIN\_3, 0x00);

**SysCtlDelay**(4047620); //delay

**if**(ui8PinData==4) {ui8PinData=8;} //if blue then assign green next

**else** **if**(ui8PinData==8) {ui8PinData=2;} //if green then turn red

**else** **if**(ui8PinData==2) {ui8PinData=4;} //if red then turn blue

}

}

**Modified Code B:**

**#include**<stdint.h>

**#include**<stdbool.h>

**#include**"inc/hw\_memmap.h"

**#include**"inc/hw\_types.h"

**#include**"driverlib/sysctl.h"

**#include**"driverlib/gpio.h"

uint8\_t ui8PinData=2;

**int** freq;

**int** **main**(**void**)

{

**SysCtlClockSet**(SYSCTL\_SYSDIV\_5|SYSCTL\_USE\_PLL|SYSCTL\_XTAL\_16MHZ|SYSCTL\_SYSDIV\_2\_5);

**SysCtlPeripheralEnable**(SYSCTL\_PERIPH\_GPIOF);

**GPIOPinTypeGPIOOutput**(GPIO\_PORTF\_BASE, GPIO\_PIN\_1|GPIO\_PIN\_2|GPIO\_PIN\_3);

**while**(1)

{

**GPIOPinWrite**(GPIO\_PORTF\_BASE, GPIO\_PIN\_1|GPIO\_PIN\_2|GPIO\_PIN\_3, ui8PinData); //write output pins

freq = **SysCtlClockGet**();

**SysCtlDelay**(4047620); //delay

freq = **SysCtlClockGet**();

**GPIOPinWrite**(GPIO\_PORTF\_BASE, GPIO\_PIN\_1|GPIO\_PIN\_2|GPIO\_PIN\_3, 0x00);

**SysCtlDelay**(4047620); //delay

**if**(ui8PinData==14) {ui8PinData=2;} //if RGB then assign Red next

**else** **if**(ui8PinData==2) {ui8PinData=8;} //if red then turn green

**else** **if**(ui8PinData==8) {ui8PinData=4;} //if green then turn blue

**else** **if**(ui8PinData==4) {ui8PinData=10;} //if blue then turn RG

**else** **if**(ui8PinData==10) {ui8PinData=6;} //if RG then turn RB

**else** **if**(ui8PinData==6) {ui8PinData=12;} //if RB then turn GB

**else** **if**(ui8PinData==12) {ui8PinData=14;} //if GB then turn RGB

}

}

**------------------------------------------------------------------------------------**