

(Highlighting: Task 01(no highlighting), Task 02, Task 03)

**Task01: Submit a comprehensive commented file of the original code.**

```
/*Shabrya Lott
 * Tiva_c Lab03
 * Usage: This is a simple program that enables toggling on the TIVA_C launchpad
 * Input: NONE
 * Output: Red, Blue and Green LED's lit in specified cycle
 */
#include <stdint.h> //variable definitions for the C99 standard
#include <stdbool.h> //Boolean definitions for the C99 standard
#include "inc/hw_memmap.h" //macros defining the memory map of Tiva C Series
#include "inc/hw_types.h" //defines common types and macros
#include "driverlib/sysctl.h" //defines macros for System Control API of Driverlib
#include "driverlib/gpio.h" //defines macros for GPIO API of Driverlib

uint8_t ui8PinData=2; //unsigned 8-bit int that is used to cycle through LEDs

int main(void)
{
    //sets clock: xtal = 16Mhz, 400MHz PLL divided by 10
    SysCtlClockSet(SYSCTL_SYSDIV_5|SYSCTL_USE_PLL|SYSCTL_XTAL_16MHZ|SYSCTL_OSC_M
AIN);
    //enables PORT F
    SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOF);
    //set 3 GPIO pins conneced to the LEDs as ouputs
    GPIOPinTypeGPIOOutput(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_2|GPIO_PIN_3);

    while(1)
    {
        //turn on the LED specified in ui8PinData
        GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_2|GPIO_PIN_3, ui8PinData);
        SysCtlDelay(2000000); //delay = 2000000 * 3 = 6000000 CPU cycles
        // turn all LEDs off
        GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_2|GPIO_PIN_3, 0x00);

        SysCtlDelay(2000000); //delay = 2000000 * 3 = 6000000 CPU cycles
        //set ui8PinData to the next LED color in the sequence
        //0010 (red), 0100 (blue), 1000 (green), ...
        if(ui8PinData==8) {ui8PinData=2;} else {ui8PinData=ui8PinData*2;}
    }
}
```

**Task 02: Change the delay of the LED blink (approx. 0.333 sec)  
by changing the clock source and configuration**

```
int main(void)
{
    //2000000 loop * 3 CPU CYCLES = 6000000 CPU CYCLE
    //6000000/ freq = 0.333sec => freq = 6000000/0.333 sec = 18.18MHZ
    //sets clock: xtal = 16Mhz, 400MHz PLL divided by 22
    SysCtlClockSet(SYSCTL_SYSDIV_11|SYSCTL_USE_PLL|SYSCTL_XTAL_16MHZ|SYSCTL_OSC_
MAIN);
```

### Task 03: a) Change the sequence of LED blinking

// Shabrya Lott CPE403 Lab3

//Task03 a: Change the sequence of the LED blinking

```
#include <stdint.h> //variable definitions for the C99 standard
#include <stdbool.h> //Boolean definitions for the C99 standard
#include "inc/hw_memmap.h" //macros defining the memory map of Tiva C Series
#include "inc/hw_types.h" //defines common types and macros
#include "driverlib/sysctl.h" //defines macros for System Control API of Driverlib
#include "driverlib/gpio.h" //defines macros for GPIO API of Driverlib
```

uint8\_t ui8PinData=8; //unsigned 8-bit int that is used to cycle through LEDs in reverse

int main(void)

```
{
    .
    .
    .

    while(1)
    {
        .
        .
        .
        //1000 (green), 0100 (blue), 0010 (red), ...
        if(ui8PinData==2) {ui8PinData=8;} else {ui8PinData=ui8PinData/2;}
    }
}
```

### Task 03: b) Blink two LEDs at an instance and with a sequence

```
#include <stdint.h> //variable definitions for the C99 standard
#include <stdbool.h> //Boolean definitions for the C99 standard
#include "inc/hw_memmap.h" //macros defining the memory map of Tiva C Series
#include "inc/hw_types.h" //defines common types and macros
#include "driverlib/sysctl.h" //defines macros for System Control API of Driverlib
#include "driverlib/gpio.h" //defines macros for GPIO API of Driverlib
```

uint8\_t ui8PinData=6; //unsigned 8-bit int that is used to cycle through LEDs

int main(void)

```
{
    .
    .
    .

    while(1)
    {
        .
        .
        .
        //blink two LEDs simultaneously in a sequence
        if(ui8PinData==6) {ui8PinData=10;} // 0110 (BR-purple)
        else if (ui8PinData==10) {ui8PinData=12;} // 1010(RG-yellow)
        else {ui8PinData=6;} // 1100(GB-aqua)
    }
}
```