Geovanni Portillo
Github root directory: https://github.com/portig1/submissions_E2

Date Submitted: **September 24, 2019**

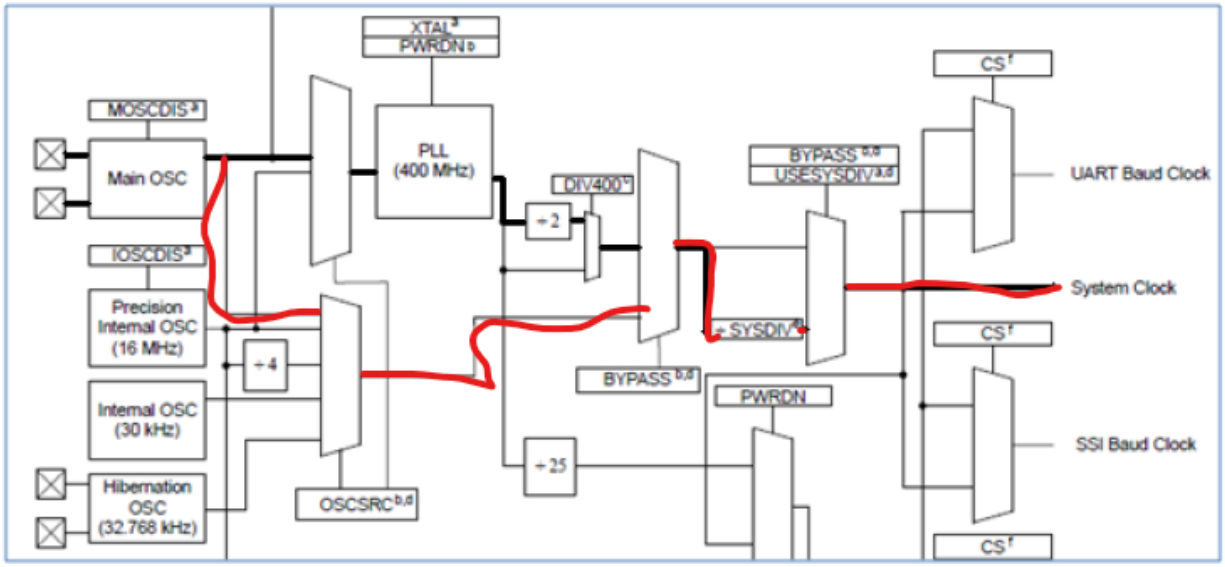
Task 00: Execute provided code

Youtube Link: https://youtu.be/kP4MioswWSA

Task 01:

Youtube Link: https://youtu.be/j5rTLOqpxf4

Modified Schematic (if applicable):



Modified Code:

```
uint8_t ui8PinData=2;

int main(void)
{
    SysCtlClockSet(SYSCTL_USE_OSC|SYSCTL_OSC_MAIN);
    //Running the main oscillator which has a 16MHz crystal

    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);
        SysCtlDelay(8000000/3);
        //F=16MHz, to get delay of 0.5s, 16M * 0.5 = 8M, however the SYsCtlDelay
takes 3 cycles so the number needs to be divided by 3
        GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_2|GPIO_PIN_3, 0x00);
```

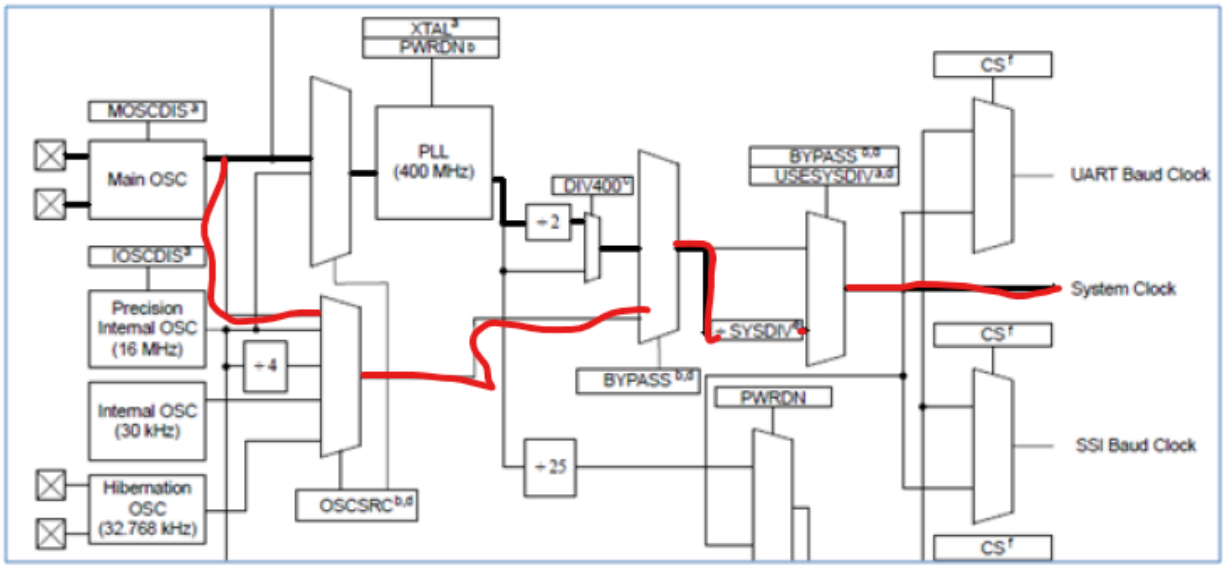
Grading scheme: 30% Coding, 30% Documentation, 40% Execution/Video.

```
        SysCtlDelay(8000000/3);
       // if(ui8PinData==8) {ui8PinData=2;} else {ui8PinData=ui8PinData*2;}
       //Commented out to demonstrate only the red light blinking
    }
}
```

Task 02:

Youtube Link: https://youtu.be/QAuaSuvOjYM

Modified Schematic (if applicable):



Modified Code:

```
uint8_t ui8PinData=2;

void blink();
int main(void)
{
    SysCtlClockSet(SYSCTL_USE_OSC|SYSCTL_OSC_MAIN);
        //Running the main oscillator which has a 16MHz crystal

    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);
        SysCtlDelay(8000000/3);
        GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_2|GPIO_PIN_3, 0x00);
        SysCtlDelay(8000000/3);
        if(ui8PinData==8) {ui8PinData=2;} else {ui8PinData=ui8PinData*2;}   //The
sequence of colors is BGR when ui8PinData is initialized at 4
        //if(ui8PinData==2) {ui8PinData=8;} else {ui8PinData=ui8PinData/2;} //The
sequence of colors is RGB when ui8PinData is initialized at 2
```

Grading scheme: 30% Coding, 30% Documentation, 40% Execution/Video.

```
    }
    */


    while(1)
    {
        ui8PinData = 2; //R
        blink();
        ui8PinData = 8; //G
        blink();
        ui8PinData = 4; //B
        blink();
        ui8PinData = 10; //RG
        blink();
        ui8PinData = 6; //RB
        blink();
        ui8PinData = 12; //GB
        blink();
        ui8PinData = 14; //RGB
        blink();
    }

}

void blink() {  //Since ui8PinData is a global variable, I don't to include it in the arguments
    //blink will turn the pins on and off with the current ui8PinData value
    GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_2|GPIO_PIN_3, ui8PinData);
    SysCtlDelay(8000000/3);
    //F=16MHz, to get delay of 0.5s, 16M * 0.5 = 8M, however the SYsCtlDelay takes 3 cycles so the number needs to be divided by 3
    GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_2|GPIO_PIN_3, 0x00);
    SysCtlDelay(8000000/3);
}
```

Grading scheme: 30% Coding, 30% Documentation, 40% Execution/Video.