Ricky Perez
Github root directory: https://github.com/RickyPerez79/da_submissions_2019

Date Submitted: 9/26/2019

Task 00: Execute provided code

Youtube Link: https://www.youtube.com/watch?v=2t_qeYKDPTA

Task 01:

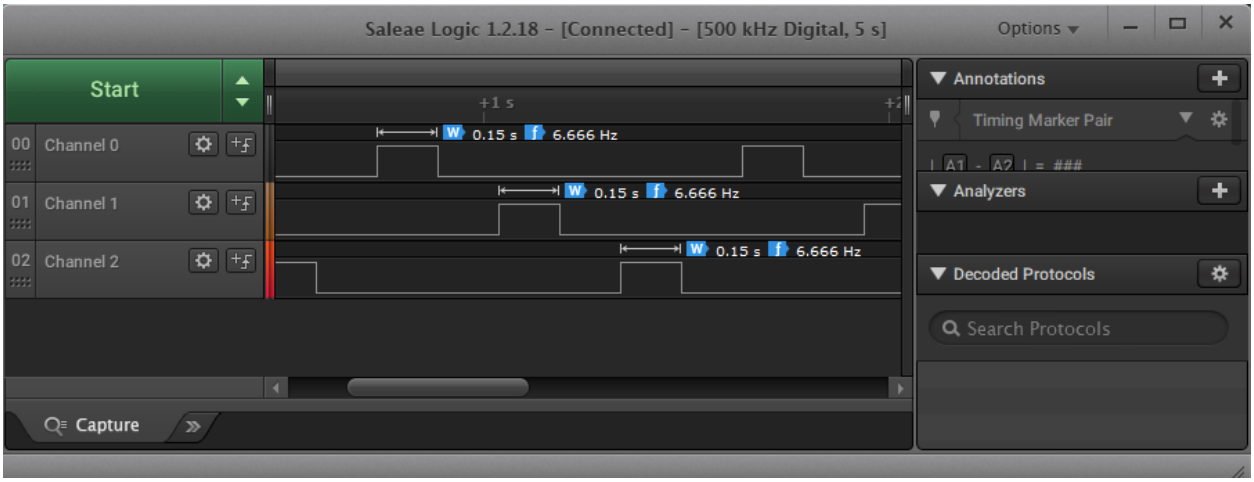
Youtube Link: https://www.youtube.com/watch?v=NvNSERVzbSY

Current period of the LED blinking (Original code)

$$\frac{40MHz}{2MHz} = 20$$

On-time of the LED blinking (Original code)

$$\frac{3}{20} = 0.15\ seconds$$



Modified Code 1:

```
#include <stdint.h>
#include <stdbool.h>
#include "inc/hw_types.h"
#include "inc/hw_memmap.h"
#include "driverlib/sysctl.h"
#include "driverlib/gpio.h"

uint8_t ui8PinData=2;

int main(void)
{
    SysCtlClockSet(SYSCTL_SYSDIV_5|SYSCTL_USE_PLL|SYSCTL_XTAL_16MHZ|SYSCTL_OSC_MAIN);

    SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOF);
    GPIOPinTypeGPIOOutput(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_2|GPIO_PIN_3);
```

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

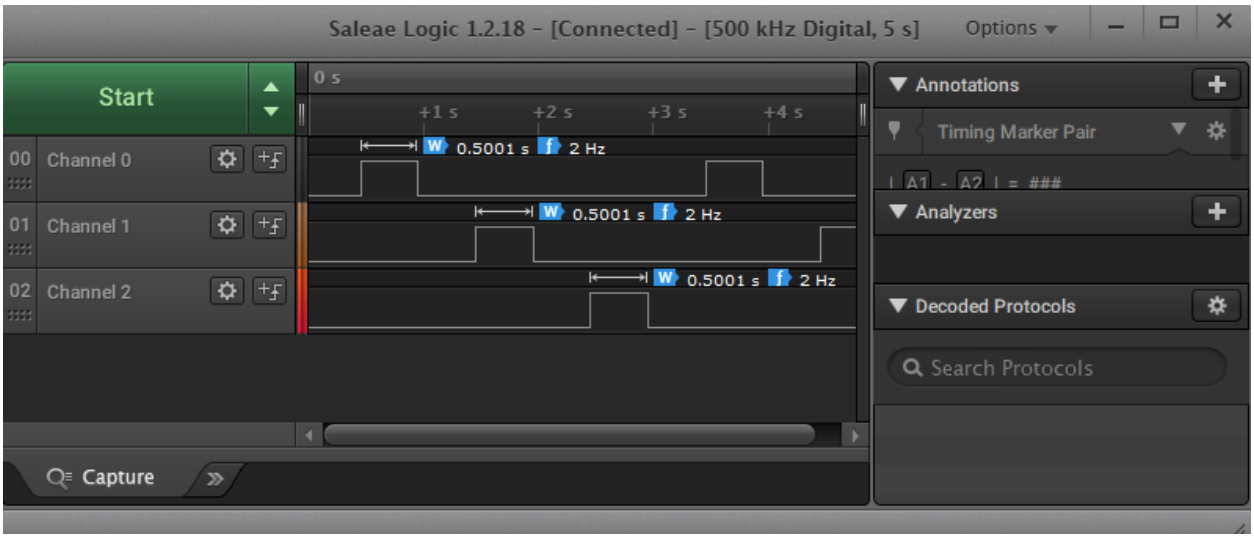
```
    while(1)
    {
        GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_2|GPIO_PIN_3, ui8PinData);
        SysCtlDelay(2000000);
        GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_2|GPIO_PIN_3, 0x00);
        SysCtlDelay(2000000);
        if(ui8PinData==8){ui8PinData=2;} else {ui8PinData*=2;}
    }

}
```

Current period of the LED blinking (Modified code)

$$\mathbf{0.5}\boldsymbol{seconds} = \frac{3}{x} \rightarrow x = 6$$

On-time of the LED blinking (Modified code)

$$\frac{\boldsymbol{40MHz}}{\boldsymbol{y}} = 6 \rightarrow y = 6.66667$$




```
Modified Code 2:
// Task 1
#include <stdint.h>
#include <stdbool.h>
#include "inc/hw_types.h"
#include "inc/hw_memmap.h"
#include "driverlib/sysctl.h"
#include "driverlib/gpio.h"

uint8_t ui8PinData=2;

int main(void)
{
    SysCtlClockSet(SYSCTL_SYSDIV_5|SYSCTL_USE_PLL|SYSCTL_XTAL_16MHZ|SYSCTL_OSC_MAIN);
```

```
    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(6666666);// approx 0.5 seconds
        GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_2|GPIO_PIN_3, 0x00);
        SysCtlDelay(6666666);// approx 0.5 seconds
        if(ui8PinData==8){ui8PinData=2;} else {ui8PinData*=2;}
    }

}
```

Task 02a:

Youtube Link: https://www.youtube.com/watch?v=Q8bh5NhkhU4

Modified Code:

```
// Task 2a
#include <stdint.h>
#include <stdbool.h>
#include "inc/hw_types.h"
#include "inc/hw_memmap.h"
#include "driverlib/sysctl.h"
#include "driverlib/gpio.h"
// pin1 = r
// pin2 = b
// pin3 = g
//uint8_t ui8PinData=2; 001 0 -> r
//uint8_t ui8PinData=4; 010 0 -> b
//uint8_t ui8PinData=8; 100 0 -> g

// B G R
// 4,8,2

int main(void)
{
    uint8_t ui8PinData=4;
    SysCtlClockSet(SYSCTL_SYSDIV_5|SYSCTL_USE_PLL|SYSCTL_XTAL_16MHZ|SYSCTL_OSC_MAIN);

    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(10000000); // made delay slower to see the colors
        GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_2|GPIO_PIN_3, 0x00);
        SysCtlDelay(10000000); // made delay slower to see the colors
        if(ui8PinData==8) {ui8PinData=2;} else {ui8PinData*=2;}
    }

}
```

Task 02b:

Youtube Link: https://www.youtube.com/watch?v=na7ScNuCmeg

```
// Task 2b
#include <stdint.h>
#include <stdbool.h>
#include "inc/hw_types.h"
#include "inc/hw_memmap.h"
#include "driverlib/sysctl.h"
#include "driverlib/gpio.h"
// pin1 = r
// pin2 = b
// pin3 = g
// uint8_t ui8PinData=2; 001 0 -> r
// uint8_t ui8PinData=4; 010 0 -> b
// uint8_t ui8PinData=8; 100 0 -> g

uint8_t ui8PinData=1;

int main(void)
{
    SysCtlClockSet(SYSCTL_SYSDIV_5|SYSCTL_USE_PLL|SYSCTL_XTAL_16MHZ|SYSCTL_OSC_MAIN);

    SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOF);
    GPIOPinTypeGPIOOutput(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_2|GPIO_PIN_3);

    while(1)
    {
        while(ui8PinData<10)
        {
        ui8PinData *= 2; // R,G,B
        GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_2|GPIO_PIN_3, ui8PinData);
        SysCtlDelay(10000000);
        GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_2|GPIO_PIN_3, 0x00);
        SysCtlDelay(10000000);
        if(ui8PinData==4){ui8PinData=10;} // once it reaches to 4 it will exit the
loop
        if(ui8PinData==2){ui8PinData*=2;}
        if(ui8PinData==8){ui8PinData=2;}
        }
        ui8PinData = 8;
        while (ui8PinData < 16)
        {
        /*
          (sequence of blinking with delay – R, G, B, RG, RB, GB, RGB, R, G, …)
```

```
            R = 0010 = 2
            B = 0100 = 4
            G = 1000 = 8
            RG = 1010 = 10
            RB = 0110 = 6
            GB = 1100 = 12
            RBG = 1110 = 14
           */
          ui8PinData +=2; // starts at RG
          GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_3|GPIO_PIN_2|GPIO_PIN_1,ui8PinData);
          SysCtlDelay(10000000);
          GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_3|GPIO_PIN_2|GPIO_PIN_1, 0x00);
          SysCtlDelay(10000000);
          if (ui8PinData == 14) {ui8PinData +=2;}
          if (ui8PinData == 10){ui8PinData = 4;}
          if (ui8PinData == 6) {ui8PinData = 10;}
          }
          ui8PinData = 1; // reinitialize
      }

}
```

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