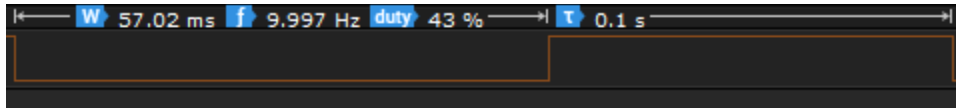


Date Submitted: 09/28/2019

Task 00: Execute provided code

Youtube Link: <https://www.youtube.com/watch?v=X2JonuHu03w>

Task 01: Change the toggle of the GPIO @ 10 Hz using Timer0 with a 43% duty cycle and verify wavefo



Youtube Link: http://www.youtube.com/watch?v=1TJP7hi7_mI

Modified Code:

```

#include <stdint.h>
#include <stdbool.h>
#include "inc/tm4c123gh6pm.h"
#include "inc/hw_memmap.h"
#include "inc/hw_types.h"
#include "driverlib/sysctl.h"
#include "driverlib/interrupt.h"
#include "driverlib/gpio.h"
#include "driverlib/timer.h"

uint32_t ui32PeriodHigh;
uint32_t ui32PeriodLow;

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);

    SysCtlPeripheralEnable(SYSCTL_PERIPH_TIMER0);
    TimerConfigure(TIMER0_BASE, TIMER_CFG_PERIODIC);

    ui32PeriodHigh = 43*(SysCtlClockGet() / 10)/100; //43% duty cycle
    ui32PeriodLow  = 57*(SysCtlClockGet() / 10)/100;
    TimerLoadSet(TIMER0_BASE, TIMER_A, ui32PeriodHigh - 1);

    IntEnable(INT_TIMER0A);
    TimerIntEnable(TIMER0_BASE, TIMER_TIMA_TIMEOUT);
    IntMasterEnable();

    TimerEnable(TIMER0_BASE, TIMER_A);

    while(1)
    {
    }
}

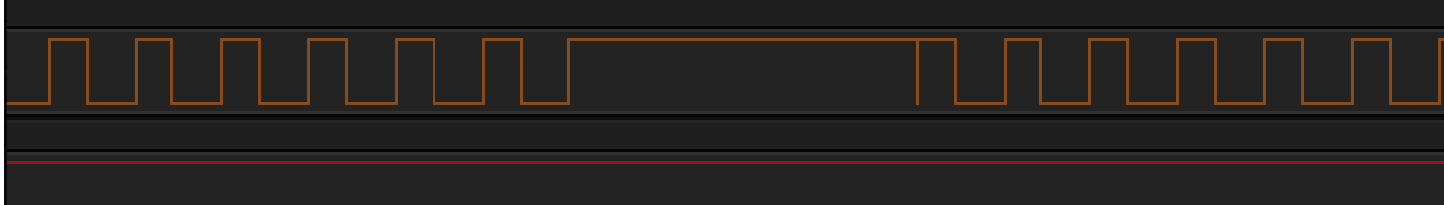
void Timer0IntHandler(void)
{
    // Clear the timer interrupt
    TimerIntClear(TIMER0_BASE, TIMER_TIMA_TIMEOUT);

    // Read the current state of the GPIO pin and
    // write back the opposite state
    if(GPIOPinRead(GPIO_PORTF_BASE, GPIO_PIN_2))
    {
        TimerLoadSet(TIMER0_BASE, TIMER_A, ui32PeriodLow - 1);
        GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_2|GPIO_PIN_3, 0);
    }
    else
    {
        TimerLoadSet(TIMER0_BASE, TIMER_A, ui32PeriodHigh - 1);
        GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_2, 4);
    }
}

```

Task 02: Include a GPIO Interrupt to Task 01 from switch SW2 to turn ON and the LED for 1 sec. Use a Timer1 to calculate the 1 sec delay. The toggle of the GPIO is suspended when executing the interrupt.

Youtube Link: <http://www.youtube.com/watch?v=1wjSk-SMx2A>



Modified Code:

```
SysCtlClockSet(SYSCTL_SYSDIV_5|SYSCTL_USE_PLL|SYSCTL_XTAL_16MHZ|SYSCTL_OSC_MAIN);

SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOF);
GPIOPinTypeGPIOInput(GPIO_PORTF_BASE, GPIO_PIN_4);
GPIOPadConfigSet(GPIO_PORTF_BASE,GPIO_PIN_4,GPIO_STRENGTH_2MA,GPIO_PIN_TYPE_STD_WPU);
GPIOPinTypeGPIOOutput(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_2|GPIO_PIN_3);

GPIOIntEnable(GPIO_PORTF_BASE,GPIO_INT_PIN_4);
GPIOIntTypeSet(GPIO_PORTF_BASE,GPIO_INT_PIN_4,GPIO_RISING_EDGE);


SysCtlPeripheralEnable(SYSCTL_PERIPH_TIMER0);
TimerConfigure(TIMER0_BASE, TIMER_CFG_PERIODIC);

ui32Period = (SysCtlClockGet() / 100); //in order for switch to create the 1sec delay
ui32PeriodHigh = 43*(SysCtlClockGet() / 10)/100; //43% duty cycle
ui32PeriodLow = 57*(SysCtlClockGet() / 10)/100;
TimerLoadSet(TIMER0_BASE, TIMER_A, ui32PeriodHigh -1);
IntEnable(INT_GPIOF);
IntEnable(INT_TIMER0A);
TimerIntEnable(TIMER0_BASE, TIMER_TIMA_TIMEOUT);
IntMasterEnable();

TimerEnable(TIMER0_BASE, TIMER_A);

while(1)
{
}

void timer1A_delayMs(int ttime)
{
    int i;
    SYSCTL_RCGCTIMER_R |= 2;
    TIMER1_CTL_R = 0;
    TIMER1_CFG_R = 0X04;
    TIMER1_TAMR_R = 0X02;
    TIMER1_TAILR_R = 64000 - 1;
    TIMER1_TAPR_R = 250 - 1;
    TIMER1_ICR_R = 0X1;
    TIMER1_CTL_R |= 0X01;

    for(i=0;i<ttime;i++){
        while((TIMER1_RIS_R & 0X01) == 0);
        TIMER1_ICR_R = 0X01;
    }
}

void PortFPin4IntHandler(void)
{
    GPIOIntClear(GPIO_PORTF_BASE,GPIO_INT_PIN_4);
    GPIOPinWrite(GPIO_PORTF_BASE,GPIO_PIN_2,GPIO_PIN_2);
    //Call Timer 1 Delay
    timer1A_delayMs(1);
    GPIOPinWrite(GPIO_PORTF_BASE,GPIO_PIN_2,0);
}
```
