**Date Submitted: 12/13/19**

**Task 01:**

Youtube Link: https://youtu.be/GelY8UNbkVY

**Modified Code:**

**#include** <xdc/std.h>

**#include** <ti/sysbios/BIOS.h>

**#include** <xdc/runtime/Log.h>

**#include** <xdc/cfg/global.h>

**#include** <stdint.h>

**#include** <stdbool.h>

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

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

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

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

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

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

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

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

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

**#include** "driverlib/pin\_map.h"

**#include** "utils/uartstdio.h"

**#include** "utils/uartstdio.c"

**void** **hardware\_init**(**void**);

**void** **ledToggle**(**void**);

**void** **Timer\_ISR**(**void**);

**void** **initADC**();

**void** **getADC3**(**void**);

**void** **InitConsole**(**void**);

**void** **UARTdisplayADC**(**void**);

**volatile** int16\_t i16ToggleCount = 0;

**volatile** int16\_t i16InstanceCount = 0;

uint32\_t ADCValues[1];

// variable used to store the output of the ADC C3

uint32\_t adc3 ;

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

{

hardware\_init();

initADC();

InitConsole();

BIOS\_start();

}

**void** **hardware\_init**(**void**)

{

uint32\_t ui32Period;

**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);

// Turn on LED

**GPIOPinWrite**(GPIO\_PORTF\_BASE, GPIO\_PIN\_1|GPIO\_PIN\_2|GPIO\_PIN\_3, 8);

// Timer 2 setup

**SysCtlPeripheralEnable**(SYSCTL\_PERIPH\_TIMER2);

**TimerConfigure**(TIMER2\_BASE, TIMER\_CFG\_PERIODIC);

ui32Period = (**SysCtlClockGet**() / 20);

**TimerLoadSet**(TIMER2\_BASE, TIMER\_A, ui32Period);

**TimerIntEnable**(TIMER2\_BASE, TIMER\_TIMA\_TIMEOUT);

**TimerEnable**(TIMER2\_BASE, TIMER\_A);

}

**void** **UARTdisplayADC**(**void**) {

**while**(1) {

Semaphore\_pend(UARTSem, BIOS\_WAIT\_FOREVER);

**UARTprintf**("ADC CH3 Value: %d\n\n", adc3);

}

}

**void** **ledToggle**(**void**)

{

**while**(1)

{

Semaphore\_pend(LEDSem, BIOS\_WAIT\_FOREVER);

// LED values: 8=GREEN

**if**(**GPIOPinRead**(GPIO\_PORTF\_BASE, GPIO\_PIN\_2))

{

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

}

**else**

{

**GPIOPinWrite**(GPIO\_PORTF\_BASE, GPIO\_PIN\_2, 8);

}

i16ToggleCount += 1;

Log\_info1("LED TOGGLED [%u] TIMES",i16ToggleCount);

}

}

**void** **Timer\_ISR**(**void**)

{

**TimerIntClear**(TIMER2\_BASE, TIMER\_TIMA\_TIMEOUT);

**if**(i16InstanceCount == 10)

{

Semaphore\_post(ADC3Sem);

}

**else** **if** (i16InstanceCount == 20)

{

Semaphore\_post(UARTSem);

}

**else** **if**(i16InstanceCount == 30)

{

Semaphore\_post(LEDSem);

i16InstanceCount = 0;

}

i16InstanceCount++;

}

**void** **InitConsole**(**void**)

{

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

**GPIOPinConfigure**(GPIO\_PA0\_U0RX);

**GPIOPinConfigure**(GPIO\_PA1\_U0TX);

**SysCtlPeripheralEnable**(SYSCTL\_PERIPH\_UART0);

**UARTClockSourceSet**(UART0\_BASE, UART\_CLOCK\_PIOSC);

**GPIOPinTypeUART**(GPIO\_PORTA\_BASE, GPIO\_PIN\_0 | GPIO\_PIN\_1);

**UARTStdioConfig**(0, 115200, 16000000);

}

// Initialize ADC0

**void** **initADC**() {

**SysCtlPeripheralEnable**(SYSCTL\_PERIPH\_ADC0);

**SysCtlDelay**(3);

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

**SysCtlDelay**(3);

**GPIOPinTypeADC**(GPIO\_PORTE\_BASE, GPIO\_PIN\_0);

**ADCSequenceConfigure**(ADC0\_BASE, 3, ADC\_TRIGGER\_PROCESSOR, 0);

**ADCSequenceStepConfigure**(ADC0\_BASE, 3, 0, ADC\_CTL\_CH3 | ADC\_CTL\_IE | ADC\_CTL\_END);

**ADCSequenceEnable**(ADC0\_BASE, 3);

**ADCIntClear**(ADC0\_BASE, 3);

}

// Receives value from ADC0 CH 3

**void** **getADC3**(**void**) {

**while**(1) {

Semaphore\_pend(ADC3Sem, BIOS\_WAIT\_FOREVER);

**ADCProcessorTrigger**(ADC0\_BASE, 3);

**while**(!**ADCIntStatus**(ADC0\_BASE, 3, false))

{

}

**ADCIntClear**(ADC0\_BASE, 3);

**ADCSequenceDataGet**(ADC0\_BASE, 3, ADCValues);

adc3 = ADCValues[0];

}

}

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