**Lab 2**

**Date Submitted: 11/23/19**

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

**Task 01:**

Memory browser:

A screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generated

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

**Task 02:**

Modified Code:

//---------------------------------------------------------------------------

// Project: Blink TM4C - CCS Lab - STARTER

//

// Author: Eric Wilbur

//

// Date: June 2014

//

//---------------------------------------------------------------------------

//------------------------------------------

// TivaWare Header Files

//------------------------------------------

**#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** <time.h>

//----------------------------------------

// Prototypes

//----------------------------------------

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

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

**void** delay(**void**);

//---------------------------------------

// Globals

//---------------------------------------

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

//---------------------------------------------------------------------------

// main()

//---------------------------------------------------------------------------

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

{

hardware\_init(); // init hardware via Xware

**while**(1) // forever loop

{

ledToggle(); // toggle LED

delay(); // create a delay of ~1/2sec

i16ToggleCount += 1; // keep track of #toggles

}

}

//---------------------------------------------------------------------------

// hardware\_init()

//

// inits GPIO pins for toggling the LED

//---------------------------------------------------------------------------

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

{

//Set CPU Clock to 40MHz. 400MHz PLL/2 = 200 DIV 5 = 40MHz

SysCtlClockSet(SYSCTL\_SYSDIV\_5|SYSCTL\_USE\_PLL|SYSCTL\_XTAL\_16MHZ|SYSCTL\_OSC\_MAIN);

// ADD Tiva-C GPIO setup - enables port, sets pins 1-3 (RGB) pins for output

SysCtlPeripheralEnable(SYSCTL\_PERIPH\_GPIOF);

GPIOPinTypeGPIOOutput(GPIO\_PORTF\_BASE, GPIO\_PIN\_1|GPIO\_PIN\_2|GPIO\_PIN\_3);

// Turn on the LED

GPIOPinWrite(GPIO\_PORTF\_BASE, GPIO\_PIN\_1|GPIO\_PIN\_2|GPIO\_PIN\_3, 4);

}

//---------------------------------------------------------------------------

// ledToggle()

//

// toggles LED on Tiva-C LaunchPad

//---------------------------------------------------------------------------

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

{

// LED values - 2=RED, 4=BLUE, 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, 4);

}

}

//---------------------------------------------------------------------------

// delay()

//

// Creates a 500ms delay via TivaWare fxn

//---------------------------------------------------------------------------

**void** delay(**void**)

{

SysCtlDelay(13400000); // makes delay 2x as slow!

}

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

**Lab 4**

**Date Submitted: 11/17/19**

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

**Task 01:**

Execution:

A screenshot of a cell phone

Description automatically generated

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

**Task 02:**

Log:A screenshot of a cell phone

Description automatically generated

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

**Lab 5**

**Date Submitted: 12/3/19**

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

**Task 01:**

**Modified Code:**

//----------------------------------------

// BIOS header files

//----------------------------------------

**#include** <xdc/std.h> //mandatory - have to include first, for BIOS types

**#include** <ti/sysbios/BIOS.h> //mandatory - if you call APIs like BIOS\_start()

**#include** <xdc/runtime/Log.h> //needed for any Log\_info() call

**#include** <xdc/cfg/global.h> //header file for statically defined objects/handles

//------------------------------------------

// TivaWare Header Files

//------------------------------------------

**#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"

//----------------------------------------

// Prototypes

//----------------------------------------

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

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

**void** **delay**(**void**);

//---------------------------------------

// Globals

//---------------------------------------

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

//---------------------------------------------------------------------------

// main()

//---------------------------------------------------------------------------

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

{

hardware\_init(); // init hardware via Xware

BIOS\_start();

}

//---------------------------------------------------------------------------

// hardware\_init()

//

// inits GPIO pins for toggling the LED

//---------------------------------------------------------------------------

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

{

uint32\_t ui32Period;

//Set CPU Clock to 40MHz. 400MHz PLL/2 = 200 DIV 5 = 40MHz

SysCtlClockSet(SYSCTL\_SYSDIV\_5 | SYSCTL\_USE\_PLL | SYSCTL\_XTAL\_16MHZ | SYSCTL\_OSC\_MAIN);

// ADD Tiva-C GPIO setup - enables port, sets pins 1-3 (RGB) pins for output

SysCtlPeripheralEnable(SYSCTL\_PERIPH\_GPIOF);

GPIOPinTypeGPIOOutput(GPIO\_PORTF\_BASE, GPIO\_PIN\_1 | GPIO\_PIN\_2 | GPIO\_PIN\_3);

// Turn on the LED

GPIOPinWrite(GPIO\_PORTF\_BASE, GPIO\_PIN\_1 | GPIO\_PIN\_2 | GPIO\_PIN\_3, 4);

// Timer 2 setup code

SysCtlPeripheralEnable(SYSCTL\_PERIPH\_TIMER2); // enable Timer 2 periph clks

TimerConfigure(TIMER2\_BASE, TIMER\_CFG\_PERIODIC); // cfg Timer 2 mode - periodic

ui32Period = (SysCtlClockGet() / 2); // period = CPU clk div 2 (500ms)

TimerLoadSet(TIMER2\_BASE, TIMER\_A, ui32Period); // set Timer 2 period

TimerIntEnable(TIMER2\_BASE, TIMER\_TIMA\_TIMEOUT); // enables Timer 2 to interrupt CPU

TimerEnable(TIMER2\_BASE, TIMER\_A); // enable Timer 2

}

//---------------------------------------------------------------------------

// ledToggle()

//

// toggles LED on Tiva-C LaunchPad

//---------------------------------------------------------------------------

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

{

TimerIntClear(TIMER2\_BASE, TIMER\_TIMA\_TIMEOUT); // must clear timer flag FROM timer

// LED values - 2=RED, 4=BLUE, 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, 4);

}

i16ToggleCount += 1; // keep track of #toggles

Log\_info1("LED TOGGLED [%u] TIMES", i16ToggleCount); // send toggle count to UIA

}

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

**Task 02:**

**A screenshot of a cell phone

Description automatically generated**

**ROV:**

**A screenshot of a cell phone

Description automatically generated**

**A screenshot of a cell phone

Description automatically generated**

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

**Lab 6**

**Date Submitted: 12/3/19**

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

**Task 01:**

**Modified Code:**

//----------------------------------------

// BIOS header files

//----------------------------------------

**#include** <xdc/std.h> //mandatory - have to include first, for BIOS types

**#include** <ti/sysbios/BIOS.h> //mandatory - if you call APIs like BIOS\_start()

**#include** <xdc/runtime/Log.h> //needed for any Log\_info() call

**#include** <xdc/cfg/global.h> //header file for statically defined objects/handles

//------------------------------------------

// TivaWare Header Files

//------------------------------------------

**#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"

//----------------------------------------

// Prototypes

//----------------------------------------

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

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

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

//---------------------------------------

// Globals

//---------------------------------------

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

//---------------------------------------------------------------------------

// main()

//---------------------------------------------------------------------------

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

{

hardware\_init(); // init hardware via Xware

BIOS\_start();

}

//---------------------------------------------------------------------------

// hardware\_init()

//

// inits GPIO pins for toggling the LED

//---------------------------------------------------------------------------

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

{

uint32\_t ui32Period;

//Set CPU Clock to 40MHz. 400MHz PLL/2 = 200 DIV 5 = 40MHz

SysCtlClockSet(SYSCTL\_SYSDIV\_5 | SYSCTL\_USE\_PLL | SYSCTL\_XTAL\_16MHZ | SYSCTL\_OSC\_MAIN);

// ADD Tiva-C GPIO setup - enables port, sets pins 1-3 (RGB) pins for output

SysCtlPeripheralEnable(SYSCTL\_PERIPH\_GPIOF);

GPIOPinTypeGPIOOutput(GPIO\_PORTF\_BASE, GPIO\_PIN\_1 | GPIO\_PIN\_2 | GPIO\_PIN\_3);

// Turn on the LED

GPIOPinWrite(GPIO\_PORTF\_BASE, GPIO\_PIN\_1 | GPIO\_PIN\_2 | GPIO\_PIN\_3, 4);

// Timer 2 setup code

SysCtlPeripheralEnable(SYSCTL\_PERIPH\_TIMER2);

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

ui32Period = (SysCtlClockGet() / 2); // period = CPU clk div 2 (500ms)

TimerLoadSet(TIMER2\_BASE, TIMER\_A, ui32Period); // set Timer 2 period

TimerIntEnable(TIMER2\_BASE, TIMER\_TIMA\_TIMEOUT); // enables Timer 2

TimerEnable(TIMER2\_BASE, TIMER\_A); // enable Timer 2

}

//---------------------------------------------------------------------------

// ledToggle()

//

// toggles LED on Tiva-C LaunchPad

//---------------------------------------------------------------------------

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

{

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

// LED values - 2=RED, 4=BLUE, 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, 4);

}

i16ToggleCount += 1; // keep track of #toggles

Log\_info1("LED TOGGLED [%u] TIMES", i16ToggleCount); // send toggle count to UIA

}

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

{

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

Swi\_post(LEDSwi);

}

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

**Task 02:**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a cell phone

Description automatically generated**

**A screenshot of a cell phone

Description automatically generated A screenshot of a cell phone

Description automatically generated**

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

**Lab 7**

**Date Submitted: 12/3/19**

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

**Task 01:**

**Modified Code:**

//----------------------------------------

// BIOS header files

//----------------------------------------

**#include** <xdc/std.h> //mandatory - have to include first, for BIOS types

**#include** <ti/sysbios/BIOS.h> //mandatory - if you call APIs like BIOS\_start()

**#include** <xdc/runtime/Log.h> //needed for any Log\_info() call

**#include** <xdc/cfg/global.h> //header file for statically defined objects/handles

**#include** <xdc/runtime/Timestamp.h> // used for Timestamp() calls

//------------------------------------------

// TivaWare Header Files

//------------------------------------------

**#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"

//----------------------------------------

// Prototypes

//----------------------------------------

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

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

//void Timer\_ISR(void);

//---------------------------------------

// Globals

//---------------------------------------

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

//---------------------------------------------------------------------------

// main()

//---------------------------------------------------------------------------

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

{

hardware\_init(); // init hardware via Xware

BIOS\_start();

}

//---------------------------------------------------------------------------

// hardware\_init()

//

// inits GPIO pins for toggling the LED

//---------------------------------------------------------------------------

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

{

// uint32\_t ui32Period;

//Set CPU Clock to 40MHz. 400MHz PLL/2 = 200 DIV 5 = 40MHz

SysCtlClockSet(SYSCTL\_SYSDIV\_5|SYSCTL\_USE\_PLL|SYSCTL\_XTAL\_16MHZ|SYSCTL\_OSC\_MAIN);

// ADD Tiva-C GPIO setup - enables port, sets pins 1-3 (RGB) pins for output

SysCtlPeripheralEnable(SYSCTL\_PERIPH\_GPIOF);

GPIOPinTypeGPIOOutput(GPIO\_PORTF\_BASE, GPIO\_PIN\_1|GPIO\_PIN\_2|GPIO\_PIN\_3);

// Turn on the LED

GPIOPinWrite(GPIO\_PORTF\_BASE, GPIO\_PIN\_1|GPIO\_PIN\_2|GPIO\_PIN\_3, 4);

}

//---------------------------------------------------------------------------

// ledToggle()

//

// toggles LED on Tiva-C LaunchPad

//---------------------------------------------------------------------------

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

{

**static** uint32\_t ui32\_t0, ui32\_t1, ui32\_t2, ui32start, ui32stop, ui32delta; // used for Timestamp calculations

ui32\_t0 = Timestamp\_get32(); // calculate Timestamp() overhead (ui32\_t2)

ui32\_t1 = Timestamp\_get32();

ui32\_t2 = ui32\_t1 - ui32\_t0;

// LED values - 2=RED, 4=BLUE, 8=GREEN

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

{

ui32start = Timestamp\_get32(); // get starting Timer snapshot for LED benchmark

GPIOPinWrite(GPIO\_PORTF\_BASE, GPIO\_PIN\_1|GPIO\_PIN\_2|GPIO\_PIN\_3, 0); // toggle GPIO/LED

ui32stop = Timestamp\_get32(); // get ending Timer snapshot for LED benchmark

ui32delta = ui32stop - ui32start - ui32\_t2; // calculate LED toggle benchmark

Log\_info1("LED BENCHMARK = [%u] TM4C CYCLES", ui32delta); // send LED benchmark to Log display

}

**else**

{

GPIOPinWrite(GPIO\_PORTF\_BASE, GPIO\_PIN\_2, 4);

}

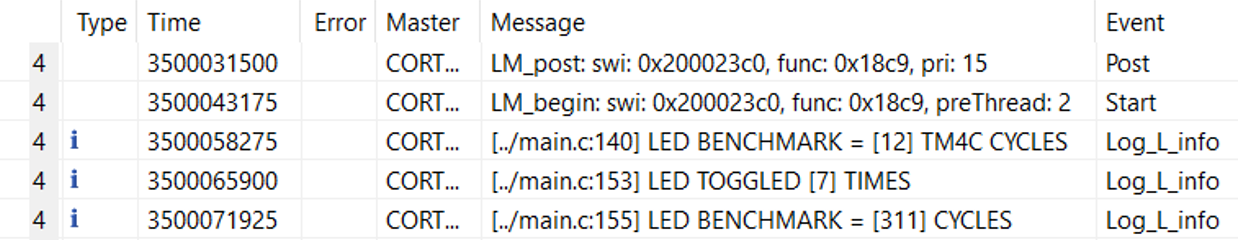
i16ToggleCount += 1; // keep track of #toggles

Log\_info1("LED TOGGLED [%u] TIMES",i16ToggleCount); // send toggle count to UIA

}

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

**Task 02:**

****

**A screenshot of a cell phone

Description automatically generated**

**A picture containing screenshot

Description automatically generated**

**A screenshot of a cell phone

Description automatically generated**

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

**Lab 8**

**Date Submitted: 12/3/19**

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

**Task 01:**

**Modified Code:**

//----------------------------------------

// BIOS header files

//----------------------------------------

**#include** <xdc/std.h> //mandatory - have to include first, for BIOS types

**#include** <ti/sysbios/BIOS.h> //mandatory - if you call APIs like BIOS\_start()

**#include** <xdc/runtime/Log.h> //needed for any Log\_info() call

**#include** <xdc/cfg/global.h> //header file for statically defined objects/handles

//------------------------------------------

// TivaWare Header Files

//------------------------------------------

**#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"

//----------------------------------------

// Prototypes

//----------------------------------------

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

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

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

//---------------------------------------

// Globals

//---------------------------------------

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

//---------------------------------------------------------------------------

// main()

//---------------------------------------------------------------------------

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

{

hardware\_init(); // init hardware via Xware

BIOS\_start();

}

//---------------------------------------------------------------------------

// hardware\_init()

//

// inits GPIO pins for toggling the LED

//---------------------------------------------------------------------------

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

{

uint32\_t ui32Period;

//Set CPU Clock to 40MHz. 400MHz PLL/2 = 200 DIV 5 = 40MHz

SysCtlClockSet(SYSCTL\_SYSDIV\_5|SYSCTL\_USE\_PLL|SYSCTL\_XTAL\_16MHZ|SYSCTL\_OSC\_MAIN);

// ADD Tiva-C GPIO setup - enables port, sets pins 1-3 (RGB) pins for output

SysCtlPeripheralEnable(SYSCTL\_PERIPH\_GPIOF);

GPIOPinTypeGPIOOutput(GPIO\_PORTF\_BASE, GPIO\_PIN\_1|GPIO\_PIN\_2|GPIO\_PIN\_3);

// Turn on the LED

GPIOPinWrite(GPIO\_PORTF\_BASE, GPIO\_PIN\_1|GPIO\_PIN\_2|GPIO\_PIN\_3, 4);

// Timer 2 setup code

SysCtlPeripheralEnable(SYSCTL\_PERIPH\_TIMER2); // enable Timer 2 periph clks

TimerConfigure(TIMER2\_BASE, TIMER\_CFG\_PERIODIC); // cfg Timer 2 mode - periodic

ui32Period = (SysCtlClockGet() /2); // period = CPU clk div 2 (500ms)

TimerLoadSet(TIMER2\_BASE, TIMER\_A, ui32Period); // set Timer 2 period

TimerIntEnable(TIMER2\_BASE, TIMER\_TIMA\_TIMEOUT); // enables Timer 2 to interrupt CPU

TimerEnable(TIMER2\_BASE, TIMER\_A); // enable Timer 2

}

//---------------------------------------------------------------------------

// ledToggle()

//

// toggles LED on Tiva-C LaunchPad

//---------------------------------------------------------------------------

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

{

**while**(1)

{

Semaphore\_pend(LEDSem, BIOS\_WAIT\_FOREVER); // wait for Sem from ISR

// LED values - 2=RED, 4=BLUE, 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, 4);

}

i16ToggleCount += 1; // keep track of #toggles

Log\_info1("LED TOGGLED [%u] TIMES",i16ToggleCount); // send toggle count to UIA

}

}

//---------------------------------------------------------------------------

// Timer ISR - called by BIOS Hwi (see app.cfg)

//

// Posts Swi (or later a Semaphore) to toggle the LED

//---------------------------------------------------------------------------

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

{

TimerIntClear(TIMER2\_BASE, TIMER\_TIMA\_TIMEOUT); // must clear timer flag FROM timer

Semaphore\_post(LEDSem);

}

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

**Task 02:**

**A screenshot of a cell phone

Description automatically generated**

**A screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generated**

**A screenshot of a cell phone

Description automatically generated**