EMBEDDED OPERATING SYSTEMS (CDA 3631C)

LAB ASSIGNMENT 1: Introduction to Board-Support Package (BSP) APIs (LED, BUTTONS)

Date Posted: 7 Sep 2019 Report Due: 17 Sep 2019 by 11:59 PM

Total: 100 points.

Problem Statement 1: **20 pts on successful Demo**

Modify the ***'main.c'*** code used in the tutorial posted on Canvas (provided in *Appendix 1* as well) to perform the following application: ***Blink all 8 LEDs one by one from Left to Right***.
That is, when executed the code shall blink LED0 first, then LED1, LED2…, LED8 and repeat. This should continue *forever* unless a new code is downloaded into the board.

This is straight forward, but you **MUST** include the following functions in your *main.c* :
LED_Initialize(), LED_On(), LED_Off(), LED_GetCount(), and delay().

Problem Statement 2: **20x3 = 60 pts on successful Demo**

Modify the '*main.c*' code in a way such that your code performs following actions:

a) ***Blink all 8 LEDs one by one from Left to Right*** **when Button 1 (USER) is pressed,**
b) ***Blink EVEN numbered LEDs only from Left to Right (i.e., LED0, 2, 4, and 6) when Button 2 (TAMPER) is pressed.***
c) ***Blink two LEDs together in the following pattern: LED#0&1, LED#1&2, LED#2&3…, LED#6&7 when Button 3(WAKEUP) is pressed.***

Also, if you keep a Button pressed, it should continue its specific application until the button is released. For example, if you keep BUTTON 1 pressed: LEDs should go on blinking one by one from Left to Right.

SUGGESTIONS for Problem Statement 2:

1) Include all the Files as shown in tutorial 1 video (posted on canvas). In addition to that you MUST add 'Buttons' from the RUN-TIME Environment (Board support> Buttons).
2) You will find "Buttons_MCBSTM32F200.c" file added on your project. Go through that to find out the predefined functions to use Buttons.
3) Do not forget to include "Board_Buttons.h" in your *main* file.
4) Hint: use *Buttons_GetState()* function to know which button gets pressed (returns 1, 2, and 4 when USER, TAMPER, and WAKEUP buttons are pressed, respectively).

Report: **20 pts**

1) MUST have a cover page with Experiment Number, Experiment Name, Student's Information (Name, ID, etc), and Course Information (Course No, Course Title). COVER page should not contain anything else.
2) Objectives
3) Experimental Setup (board used and summary of the board series, for example: Cortex M3 processor, 128kB RAM, 1 MB ROM and so on.)
4) Codes for problem statement 1 & 2. Codes **MUST** be properly commented so that it is understandable by any reader.
5) Results and Discussion

***** Submit PDF file on CANVAS before DUE Date.**

APPENDIX 1
(The code from the tutorial)

```
#include<stdint.h>
#include "Board_LED.h"              // ::Board Support:LED

void delay(void)                    //:: Performs a specific Delay
{
    uint32_t i;
    for (i=0; i<1000000; i++);
}

int main(void){

    LED_Initialize();               // Initializes Board LEDs to be programmed

    while(1){
    LED_On(0);                      // Turns on LED0
    LED_On(5);                      // Turns on LED5
    delay();                        // waits for a certain delay period
    LED_Off(0);                     // Turns off LED0
    LED_Off(5);                     // Turns off LED5
    delay();
    }
    return 0;
}
```