AE6705 Lab 3
Code Composer Studio

Yunqing (JJ) Jia

September 23, 2024

1. How should the code be modified if the requirement was to switch off LED1 after a small arbitrary delay?
   In main loop where we are checking the state of S1, add in an empty for loop followed by code to turn off the LED:

```
if S1 is pressed {
    turn on LED1;
    \\ empty for loop that runs some # of iterations
    for (int i = 0; i < count; i++) {};
    turn off LED1;
}
```

2. Suppose you were to write a program that toggles LED1 whenever S1 is pressed. With proper operation of the program you would expect to see the light turn on when you press and release S1, and then turn off when you press and release S1 again. Suppose you write this program as follows:

```
while(1){
    // Read button
    usiButton1 = DL_GPIO_readPins( GPIOA, DL_GPIO_PIN_18 );

    //If button is pressed...
    if ( usiButton1 > 0) {
        DL_GPIO_togglePins( GPIOA, DL_GPIO_PIN_0 );
    }
}
```

   What is one (or more) potential problem(s) with this code? (Hint: See Lecture 2, slide 9). What would be a potential software modification that would fix this problem?
   The mechanical switch/contact might bounce between open and closed states when the button is pressed, and if the microcontroller is checking the state of the button during every clock cycle (every $8^{-7)}$ second), the LED pin will get toggled a bunch of times during that time period. Depending on when the button is pressed and released, we don't know what state the LED will end up in. Adding in a slight delay in the if block and then take action only if the button is still pressed (i.e. the switch is in a stable state) can help mitigate this.