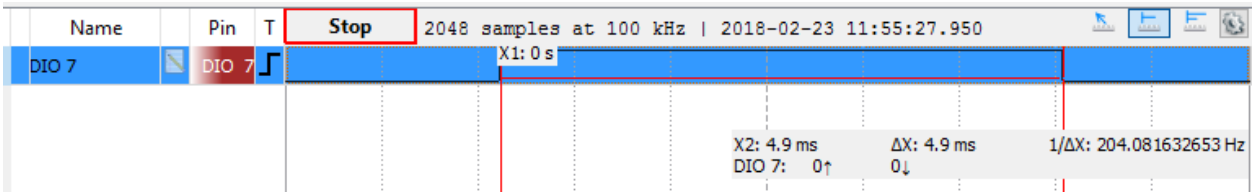
ECE 241 Lab 5 Report Spring 2018
Name: Graham Wood Lab Section A

Objective: Reading a simple switch

Prelab: The code was written and uploaded to the website. The code written and seen below prevents against wave bounce when a switch is pressed. The program does this by managing the state of the timer and implementing a wait period to prevent the bounce in the signal from trigger the switch multiple times.

Part 1) In part one, the code was uploaded to the Arduino and verified that the program only writes the message when the button is initially pressed. The program write's the message once per button click and not multiple times.

Part 2) In part 2 the analog discovery device is used in conjunction with Wave to capture the timing of the bounce when the button is pressed on the Arduino. The screenshot of the pulse can be seen below.



Questions:

What was the time you observed on pin 13, and how close to 5 milliseconds was it?

As seen in the screenshot above, the measured time was 4.9 milliseconds, only .1 milliseconds off from the predicted time. This may have been due

Appendix A) Code for Part 1

```
 int ButtonState; //integer that tracks the state that the system is in

unsigned long ButtonTime; //Time to maintain how many milliseconds since the
button was pushed

int ButtonNextState ( int input ) { //function whose purpose is to manage the
button press procedure on the arduino
  switch(ButtonState) { //switch case that's executed depending on the
current state the button is in
    case 1 : //Idle state
      if (input == LOW) {
         ButtonTime = millis();
         ButtonState = 2; //if input is LOW the state transitions to Wait
         digitalWrite(13, HIGH);//turn on LED turns on the LED
         }
    case 2:
      if (input == HIGH) {
        ButtonState = 1; //sets the state back to idle if input is HIGH
      }
        else if (millis() - ButtonTime >= 5) {
          ButtonState = 3;
          digitalWrite(13, LOW); //sets the LED back off
```

```
          return 1; //returns a one indicating that the button has been
pressed
        }
    case 3:
        if (input == HIGH) {
          ButtonState = 1; //sets the state to Idle
        }
           return 0; //returns a zero
        }
  }


void setup() {
  pinMode(13,OUTPUT); //sets pin 13 (LED) to an output
  ButtonState = 1; //initial state is 1
  Serial.begin(9600); //sets the baud rate to 9600
  }
```