Lab form – Final Project

Name:Yifan zhu
Section:001
Value of N: 3

Guidelines

1. Every figure should have a label, a caption, and annotations as needed.
2. Accompany every figure by a brief explanation of what is being shown.
3. Pay attention to English grammar.

Submit this form just ONCE during lab Weeks 10-12, but before Monday, November 13, 2023.

- It is not necessary to cover ALL controller requirements a-e on the form.
    - A Week 10 submission might cover two requirements.
    - A Week 11 submission might cover three requirements.
    - A Week 12 submission might cover four requirements.
- Any subset of requirements may be discussed. For example, a submission may cover requirements c-e, yet omit requirement b if the keypad controller is not fully functional.

Below are guidelines for discussing each of the controller requirements b-e, and extra credit xc.

Controller requirement b

Use the oscilloscope to measure the pwm signal for two duty cycles greater than 0% but less than 100%. Construct a table to display pwm characteristics for all button presses 0-A. The table should have four columns: button pressed, desired duty cycle, actual duty cycle, percent difference between desired and actual duty cycle.

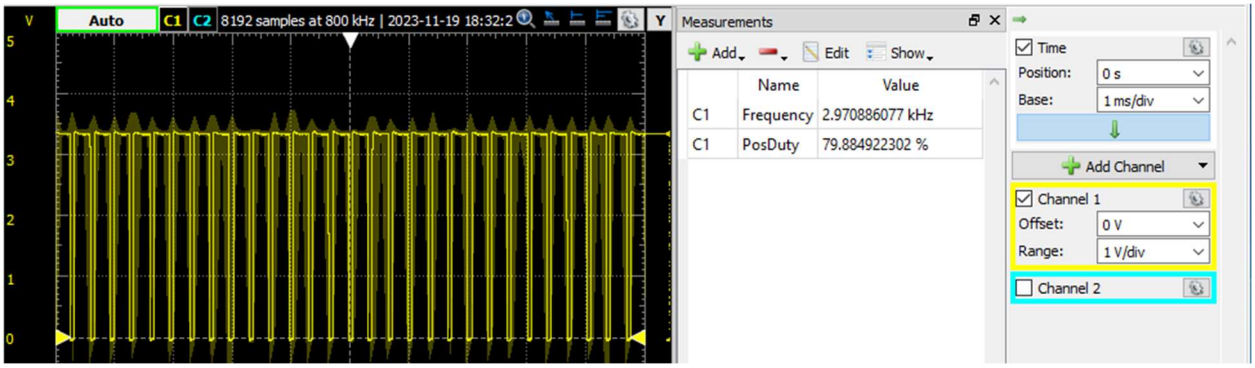

Figure1 - Output and frequency at 80% PWM duty cycle

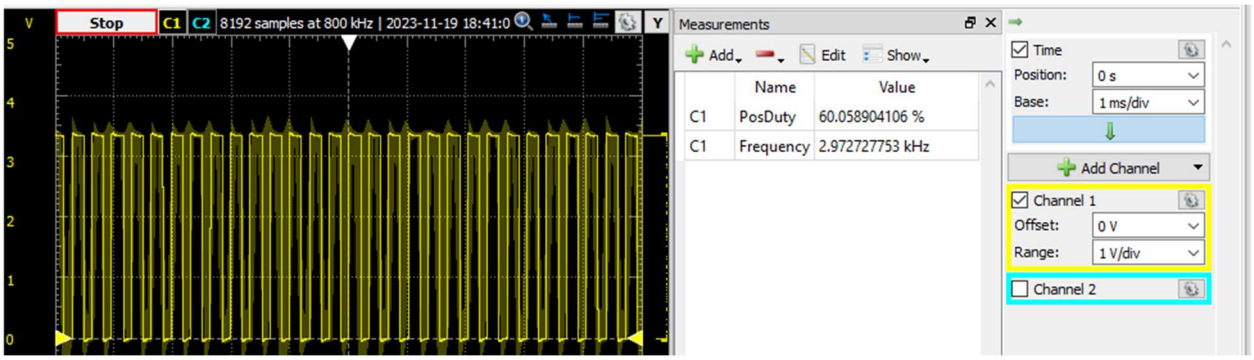

Figure2 - Output and frequency at 60% PWM duty cycle

The above two pictures are the output diagrams of different duty cycles obtained by connecting CH1 to the PWM pin and ground.

button pressed	desired duty cycle (%)	actual duty cycle (%)	percent difference (%)
0	0	0	0
1	10	9.8	-0.02
2	20	20.05	0.0025
3	30	39.8	0.326666667
4	40	39.9	-0.0025
5	50	50.06	0.0012
6	60	60.06	0.001
7	70	70.1	0.001428571
8	80	79.9	-0.00125
9	90	90.08	0.000888889
A	100	99.8	-0.002

Table1 - After the 0-A button is pressed, the required duty cycle, actual duty cycle, and their percentage difference are obtained

```
static unsigned int pwm_map [4][4]={
{0x64,0xC8,0x12C,0x3E8},
{0x190,0x1F4,0x258,0x0},
{0x2BC,0x320,0x384,0x0},
{0x0,0x0,0x0,0x0}
};
```

Pwm implements array mapping, and then implements Frequency 3khz through pwm = pwm_map[rowNum][colNum] / 3;.

Controller requirement c

Use the oscilloscope to measure voltage at the transistor collector (MOSFET drain) for one duty cycle greater than 0% but less than 100%. Using the Measurements tool, show average voltage,

frequency, and duty cycle. Discuss how collector (drain) voltage is related to the voltage across the motor winding.

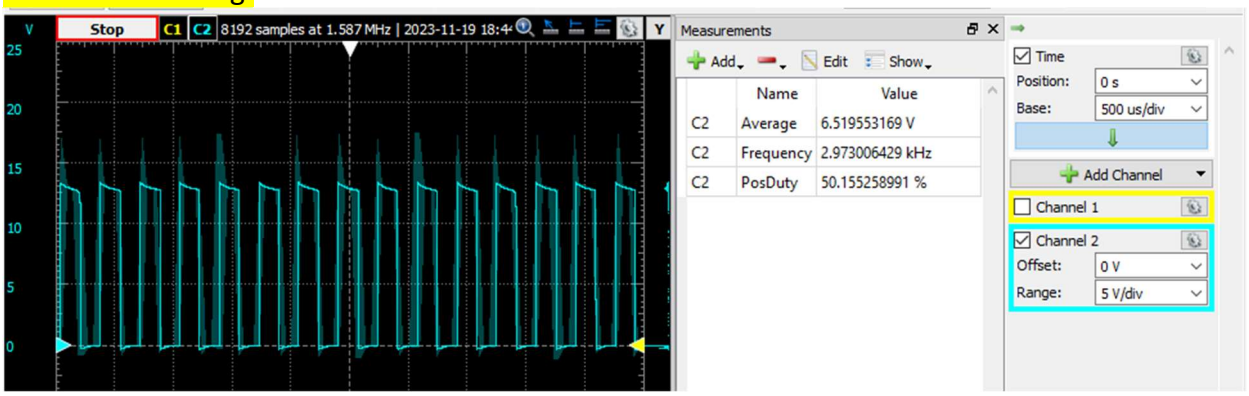

Figure3 - DC motor driver output diagram controlled by PWM signal

The picture above shows the average voltage at the MOSFET drain measured by connecting the motor and ground wires to chanel2.
As well as the frequency and duty cycle of the motor.

Controller requirement d

Use the oscilloscope to show that the counter period is N/5 seconds. Using logic analyzer data, show that the counter works as per specifications d.2, d.3, and d.4.

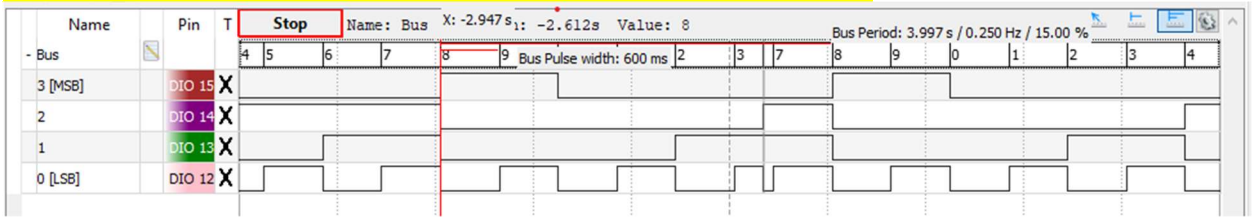

Figure4 - Counter period

This figure shows the counter period is 600ms.

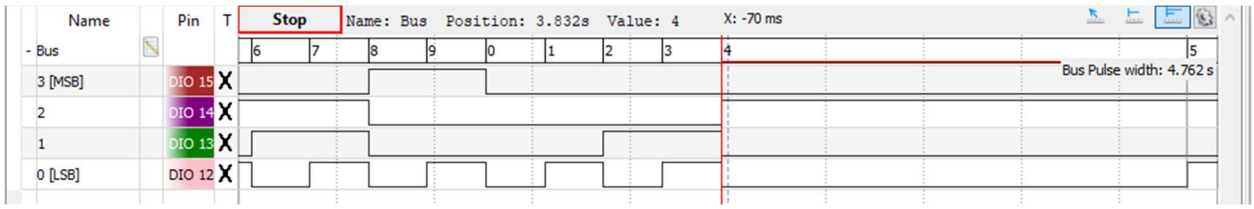

Figure5 - Counter is stopped

The button (*) is pressed and the counter is stopped, this image verifies that the button (*) is used to start/stop the counter.

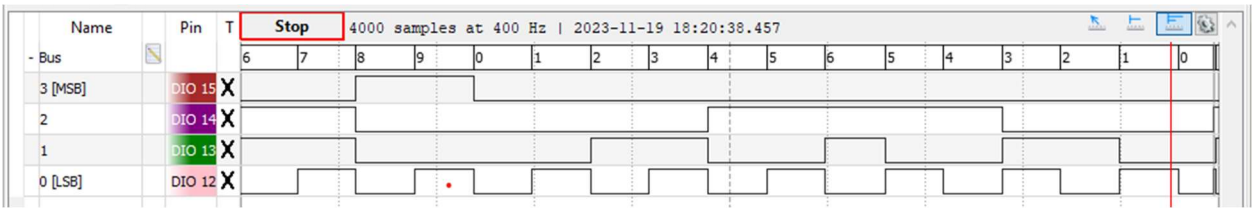

Figure6 - Counter value rolls over and repeats

This figure verifies that the counter can be flipped and counted repeatedly.

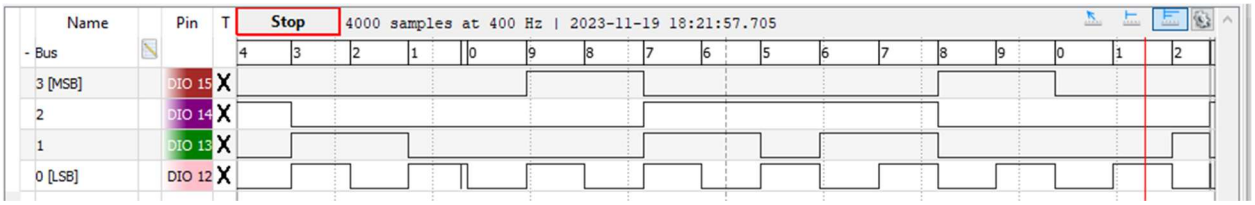

Figure7 - Counting direction changes

When the button (#) is pressed, the counting direction changes from down counting to up counting. This figure verifies that the button (#) can change the counting direction (up/down).

```
if(button == 0x0E) {//press button 0x0E counter run or not run
        //toggle TIM6 counter
        TIM6->CR1 ^= 0x0001;
}

if(button == 0x0F){
        // press button * counter up output,press again counter down out put
   if (up == 0){//when up=0,run=1 count work,run=1,run=0,count not work
     up = 1;
     }
  else{
     up = 0;
     }
}
```

The functions implemented by the above two buttons.

Controller requirement e

Present evidence that the controller meets requirements e.3 and e.4.

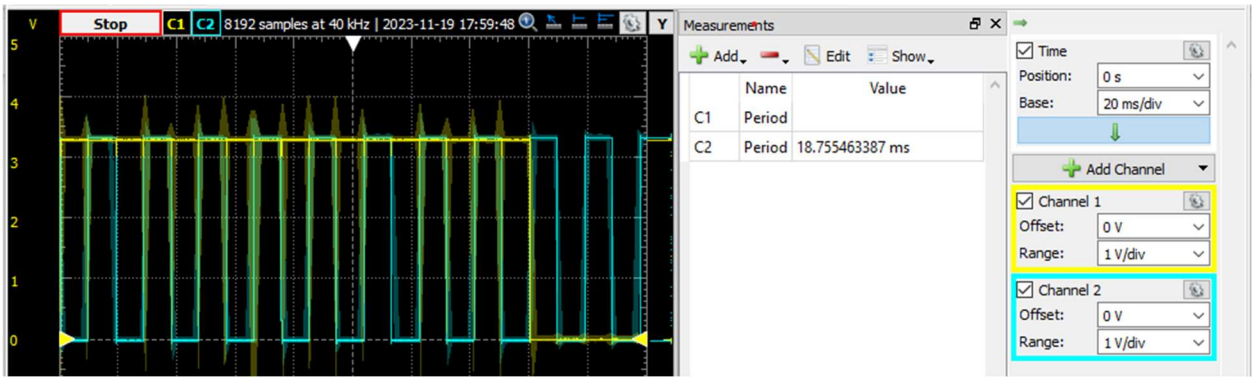

Figure8 - Feedback control of motor speed (CH1 is the counter period, CH2 is the loop period for ADC sampling)

This picture did not successfully show the period of the counter, but TA has confirmed that the data ADC sampling period is equal to the counter period divided by 64.

I wrote an additional loop to realize that the ADC sampling period is equal to the counter period divided by 64

```
if (counterc==64) {
        counterc =0;
        adc_in=ADC1->DR;
 if (e==0){
        e=1;
        }
 else {
        e =0;
        }
}

GPIOB->ODR |= e <<1;
counterc ++;
```

Controller extra credit xc

Present evidence that the controller meets all requirements (a-e) and extra credit specifics.

Fall 2023