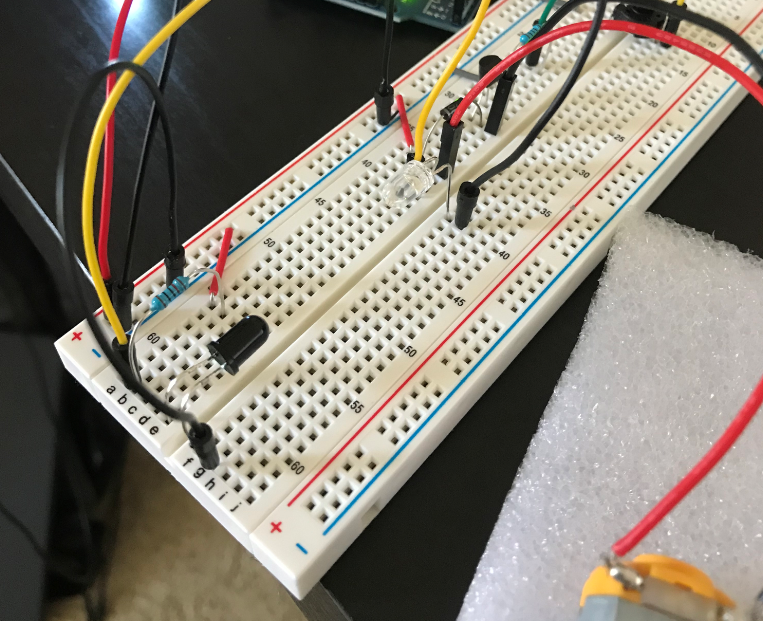
**Lab3 documentation**

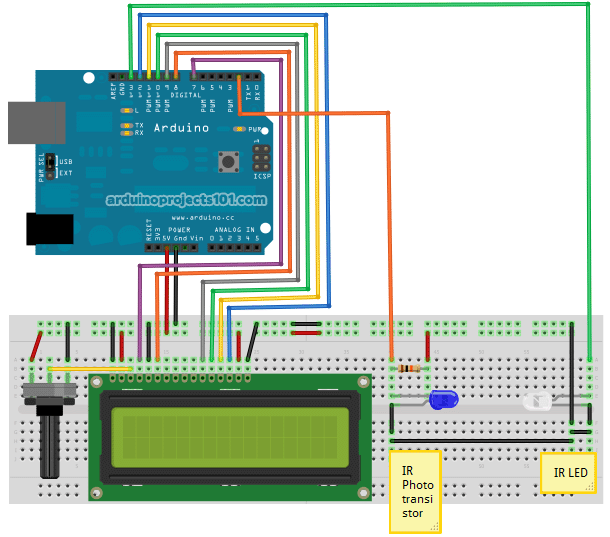
**PWD RPM Relation Table**

|  |  |
| --- | --- |
| **PWD (0~255)** | **RPM** |
| 0 | 0 |
| 70 | 6800~7200 |
| 180 | 9000~11000 |
| 255 | 15000 |

The measure was done by building an Arduino Tachometer. The phototransistor and IR LED components are bought from Amazon for $6. Other than those two, other components are all included in the kit.

****

**Tachometer setup**

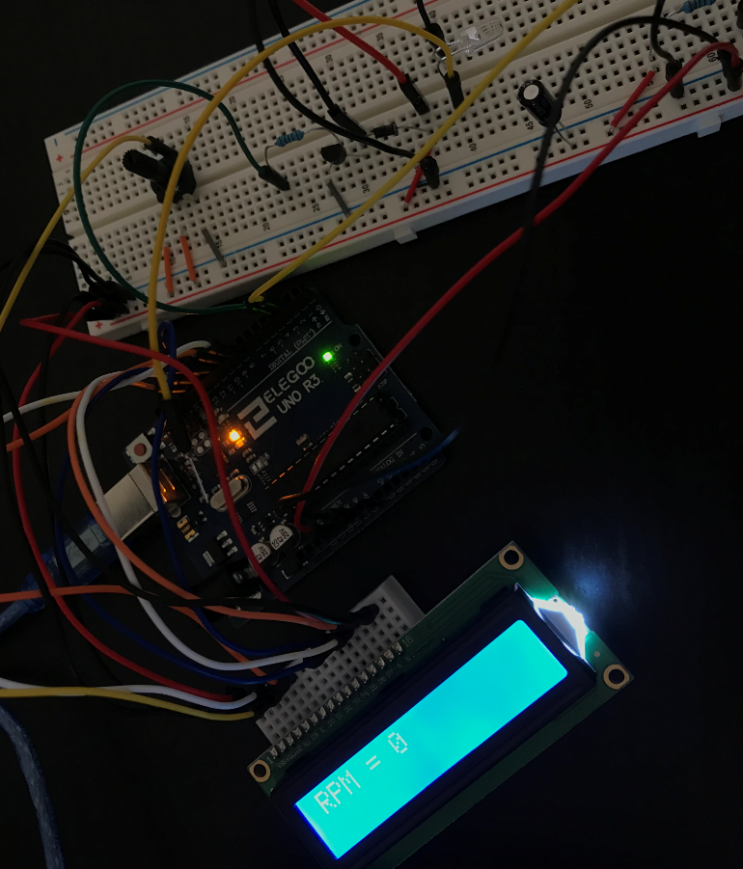


**Tachometer schematic**

Once the relationship between PWD and RPM is found. I started the c step of the Task. To accomplish this, I wrote the code as below.



**LCD Display RPMs for task 2**

****

**Whole circuit setup (including tachometer)**

**Things I’ve learned from doing this step:**

1. How to use IR phototransistor and IR emitter as a tachometer to measure motor RPM.
2. How to use LCD on an Arduino setup.
3. A decoupling capacitor is needed whenever there is a motor connected. Motor is an inductive load. And this characteristic of motor might cause transients on power and data lines to LCD and other components. (Without decoupling capacitor, LCD will start displaying random characters once the motor starts running.)
4. Adjust display contrast by using a potentiometer.

/\*\* lab3 Zhibo Wang

 \*  Motor Control with tachometer

 \*/

#include <LiquidCrystal.h>

LiquidCrystal lcd(8, 9, 10, 11, 12, 13);

const int motorpin = 3;

int speed; // range 0 ~ 255

// int passFlag = 0;

const int irLED = 7;

volatile byte breakNum;

unsigned int rpm;

void setup()

{

    Serial.begin(9600);

    lcd.begin(16, 2);

    pinMode(motorpin, OUTPUT);

    pinMode(irLED, OUTPUT);

    digitalWrite(irLED, HIGH);

    attachInterrupt(0, breakCount, FALLING);

    // analogWrite(motorpin, speed);

    // delay(3000);

    // digitalWrite(motorpin, LOW);

    // rpm = 0;

    // breakNum = 0;

    tableRotate();

}

void loop()

{

    // if (passFlag == 0)

    // {

    //     tableRotate();

    //     passFlag++;

    // }

    // delay(1000);

    // detachInterrupt(0);

    // rpm = 60 \* breakNum;

    // breakNum = 0;

    // motorswitch();

    // attachInterrupt(0, breakCount, FALLING);

}

// void motorswitch()

// {

//     analogWrite(motorpin, speed); // run fan for seven seconds and then turn it off as required

// }

void breakCount()

{

    breakNum++;

}

void tableRotate()

{

    lcddp("Max Speed");

    speed = 255;

    analogWrite(motorpin, speed);

    delay(4000);

    lcddp("0");

    speed = 0;

    analogWrite(motorpin, speed);

    delay(2000);

    lcddp("Half Speed");

    speed = 60;

    analogWrite(motorpin, speed);

    delay(5000);

    lcddp("0");

    speed = 0;

    analogWrite(motorpin, speed);

    delay(2000);

    lcddp("0.75 Speed");

    speed = 180;

    analogWrite(motorpin, speed);

    delay(3000);

    lcddp("0");

    speed = 0;

    analogWrite(motorpin, speed);

}

void lcddp(String spd)

{

    lcd.clear();

    lcd.print("RPM = ");

    lcd.print(spd);

    Serial.print("rpm = ");

    Serial.println(spd);

}

**Technical Reference:**

Measure RPM w/ DIY Arduino Optical Tachometer using Infrared LED & Phototransistor

<https://www.youtube.com/watch?v=3nOFtz6wg20>

<https://www.makertrove.com/post/2017/11/11/arduino-optical-tachometer>

<http://arduinoprojects101.com/arduino-rpm-counter-tachometer/>

HOW TO MAKE AN ARDUINO OHM METER

<http://www.circuitbasics.com/arduino-ohm-meter/>