CMU Mechatronics Team B

Sensor Lab 2/2/16

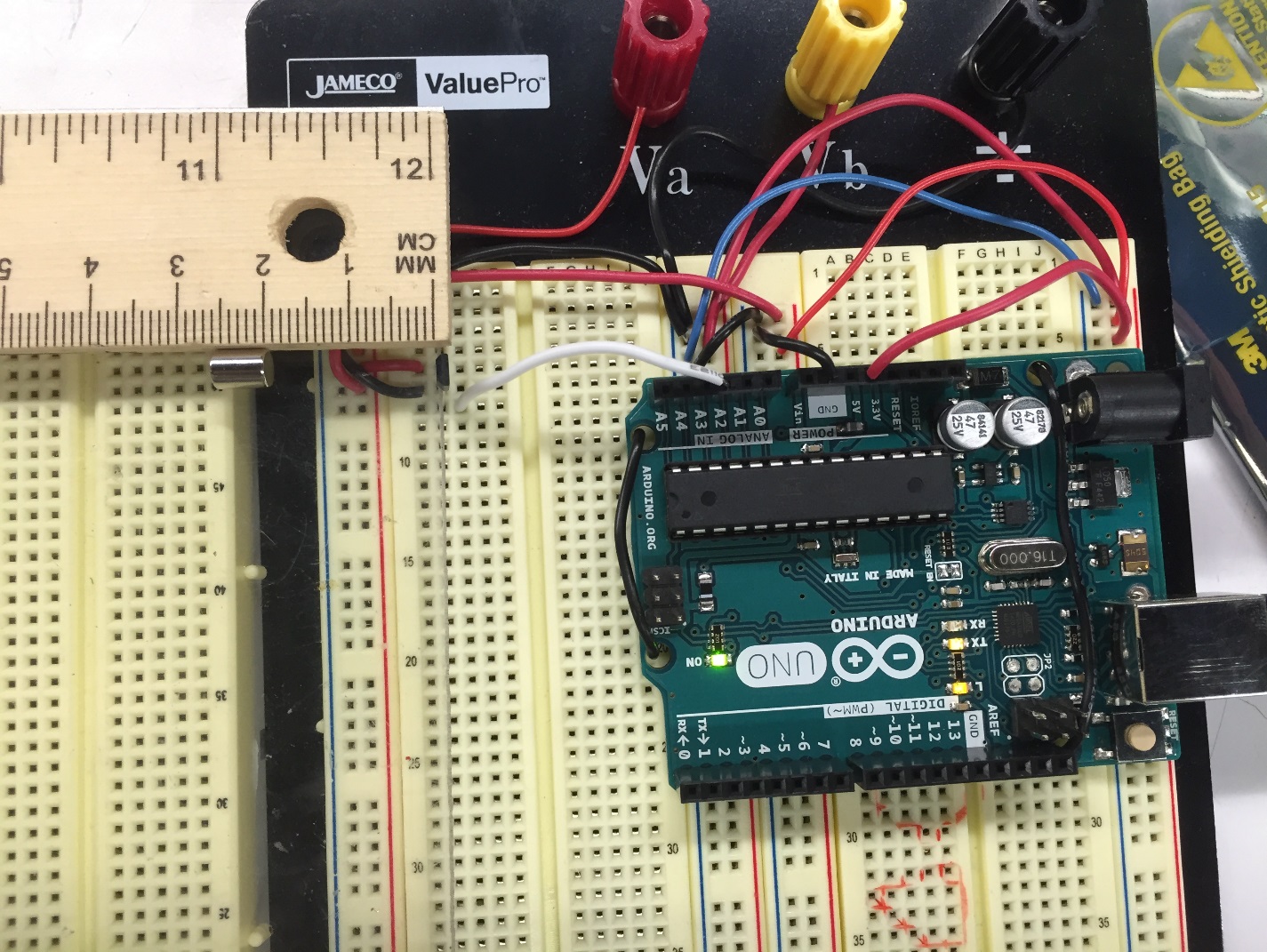
Nishant Pol

Hall Effect Sensor Characterization

Sensor: TI DRV5053CAQLPG (Digikey 296-38526-1-ND)

Magnet: McMaster 5862K963 Neodymium Disk Magnet, Nickel Plated, 0.187” Diameter ¼” Thick, 3lbs Maximum Pull

Test Setup:



Magnet

Sensor

Plastic plate

Test Procedure:

Magnet was oriented so on approaching the sensor, the output voltage increased. During the test, the magnet was brought closer to the sensor, then pushed farther from the sensor, with the same side of the magnet facing the sensor at all times. Since the sensor is bipolar, the response for the opposite magnet side would be similar, but have a decrease in voltage as the magnet is pushed closer to the sensor.

An Arduino was used to read the voltage as a raw ADC value between 0 to 1024. Using an oscilloscope, it was determined that the steady state voltage output with no magnet was 1V, and the output would increase to 2V with the magnet in one orientation, and 0V in the opposite orientation.

Test results:

Sensor working range is 0.5cm to 2cm. Response is non-linear, however for our application, we would use a lookup table. Plot shows magnet approaching the sensor, and retracting from sensor.

Focusing on the working range of the sensor, it is evident that the sensor has some hysteresis. However, in our application, the sensor would be used as the magnet approaches the sensor, so hysteresis is not a problem.

Code:

/\*

Analog input, analog output, serial output

Reads an analog input pin, maps the result to a range from 0 to 255

and uses the result to set the pulsewidth modulation (PWM) of an output pin.

Also prints the results to the serial monitor.

The circuit:

\* potentiometer connected to analog pin 0.

Center pin of the potentiometer goes to the analog pin.

side pins of the potentiometer go to +5V and ground

\* LED connected from digital pin 9 to ground

created 29 Dec. 2008

modified 9 Apr 2012

by Tom Igoe

This example code is in the public domain.

\*/

// These constants won't change. They're used to give names

// to the pins used:

const int analogInPin = A2; // Analog input pin that the potentiometer is attached to

const int analogOutPin = 9; // Analog output pin that the LED is attached to

int sensorValue = 0; // value read from the pot

int outputValue = 0; // value output to the PWM (analog out)

void setup() {

// initialize serial communications at 9600 bps:

Serial.begin(9600);

}

void loop() {

// read the analog in value:

sensorValue = analogRead(analogInPin);

// map it to the range of the analog out:

outputValue = map(sensorValue, 0, 1023, 0, 255);

// change the analog out value:

analogWrite(analogOutPin, outputValue);

// print the results to the serial monitor:

//Serial.print("sensor = " );

Serial.print(sensorValue);

//Serial.print("\t output = ");

//Serial.println(outputValue);

Serial.print('\n');

// wait 2 milliseconds before the next loop

// for the analog-to-digital converter to settle

// after the last reading:

delay(2);

}