**Function Test**

1. Test correct values on LCD screen/Oscilloscope
   1. DC values
      1. Test DC voltages in 1 V increments in the -4 to 4 V range, and 0.25 V increments closer to the boundaries (-5 V to -4 V, and 4 V to 5V).

**All of the percent errors between input DC values and output DC values fell below 6%. We will accept these results since they are all within 90% accuracy. We sampled more from closer to the boundaries (-/+5) because these areas are most likely to be attenuated due to our clippers and due to inputting and outputting the signal from the Arduino. See excel spread sheet for calculations.**

* + 1. Qualitatively check LCD screen for accuracy and record value. Ensure that all values between +/- 5V are within 10% of where they should be. Because the Arduino only displays binary numbers to the integer, we will not calculate the error displayed by the LCD screen.
  1. AC values
     1. At 1 Hz intervals between 0 and 15 Hz, the voltage range of the sine waves was tested, and the percent difference between the expected and measured high and low bounds was calculated. Frequencies where the percent difference exceeded +/- 25% of the voltage range were flagged. We chose +/- 25% of the voltage range, because it most closely corresponds with the loss of power at the point -3dB from the maximum power. The -3dB point represents the voltage at the cutoff frequency of the low pass filter, which was designed to accurately portray signals underneath the cutoff frequency.

**This test indicates that our device is accurate for frequencies from 0Hz to 10Hz. See excel sheet in github for calculations.**

* + 1. Test various intervals of the sine wave: -5 to +5, 0 to +5, -5 to 0, -2.5 to 2.5, -1 to 4, -4 to 1 to ensure that the shape of the sine wave is conserved over these various intervals.

**Shape of sine wave was conserved, except for some mild attenuation at the +/-5 extremes. Look at other test results for more information on this attenuation.**

1. Input range voltage protection
   1. This portion of the testing tested the scaled and shifted input signal before it entered the Arduino to ensure that the clipping circuit prevented voltages much higher than 5V or much lower than 0V from entering the Arduino. Signals between -10V and -5 V, and 5V and 10V were tested to make sure that the Arduino would be protected. As the Arduino can tolerate voltages slightly above and below the 0-5 V range, the values were expected to be within 10 percent of their corresponding bounds.

**The percent errors for the voltage protection at +5 volts are higher than at the voltage protection for 0 volts. This is because the clipping at 5V technically occurs at +5.6. This was unavoidable given the values of diodes that were available to us. We preferred to make the cutoff at +5.6 as opposed to +4.9, because the lower value attenuated the input values closer to +5 volts in a way that sacrificed accuracy of the shape and amplitude of our signal. As the Ardunio is able to handle an extra 0.6 V, we decided that this was a sacrifice that we could make in order to preserve our signal. See excel spreadsheet for calculations.**