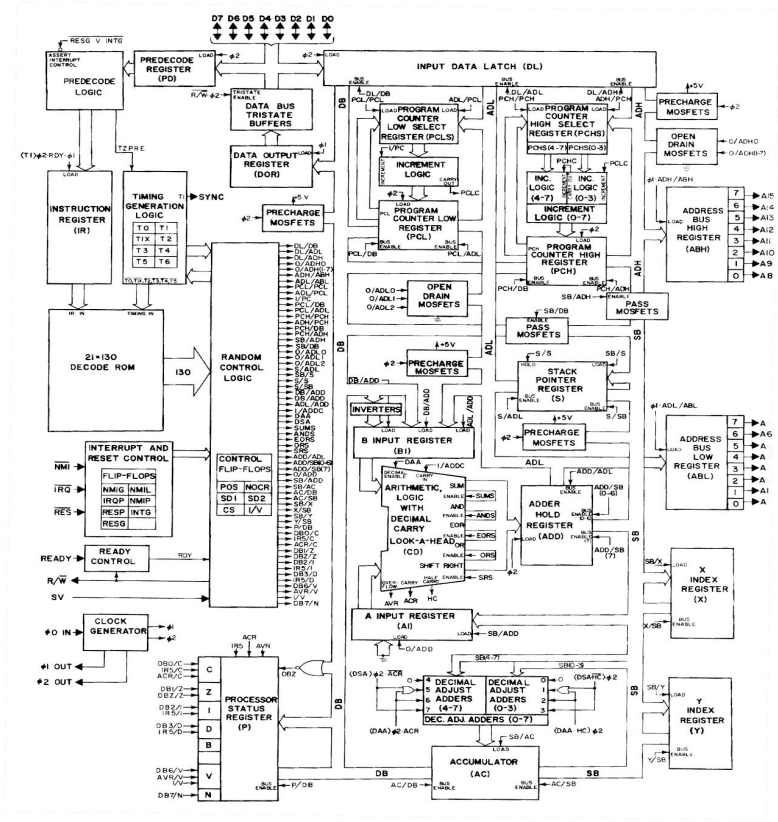
CE 167 - Sensing & Sensor Technologies

Lab 1: Flex, Piezo, and (Simple) Analog Filters

By: Kyle Jeffrey



Introduction

This lab is the introduction to resistive sensors. A flex sensor is a straight potentiometer, changing resistance with deflection. It is a nonlinear sensor and you will need to accommodate that nonlinearity. A piezo sensoris a time based analog signal that must be captured. Each tap generates a voltage spike that must be clipped to prevent damage to systems. Students over the course of this lab will make a musical instrument where the flex sensor sets tone and each tap plays said note.

Part 1: Flex Sensor

What is a Flex Sensor?

The Flex sensor acts as a variable resistor whose resistance changes with the amount of flex (or angle) that the sensor is bent around. One side of the flex sensor is printed with a polymer ink embedded with con-ducting particles. When flat, the particles are closer together, giving a lower resistance. When the flex sensor is bent, the particles are farther apart from each other, giving a higher resistance. The change is non-linear but monotonic (resistance increases with bend, but not on a straight line)

1. Procedure

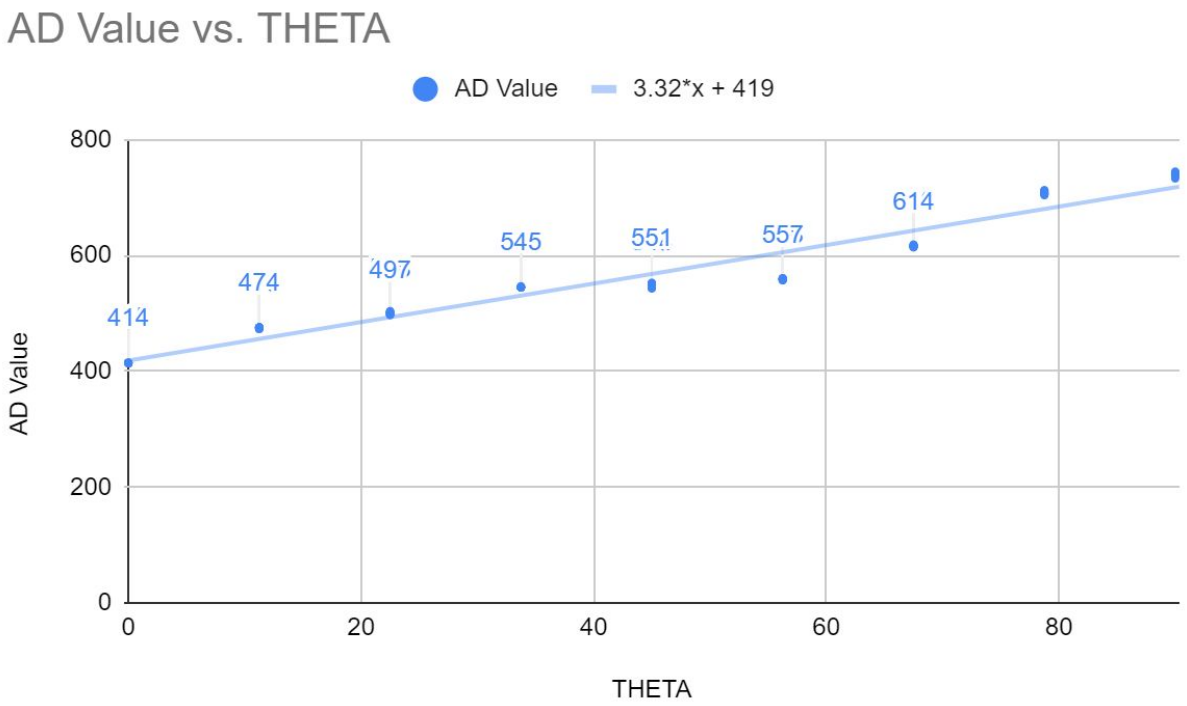
1.1 Assembly

The flex sensor needs to be assembled. There is a blue clincher connector with 2 pins that can be placed on a breadboard. You need to connect this to the flex sensor. If familiar with crimping, then using the clincher should make sense.

1.2 Flex Sensor Linearization

The flex sensor is nonlinear due to its construction and has to be linearized. The basic idea behind doing this is to find a transfer function that describes how the input relates to the output.

This is accomplished by manually taking data points for the AD reading of the flex sensor at different angles of flex from 0 to 90 degrees. Once enough data points are taken, Use regression to create a line of best fit that can be either linear or polynomial. This line of best fit represents a function that maps the AD readings to the angle of flex on the sensor. See below for graph of data points taken.



See reference data sheets attached to back.

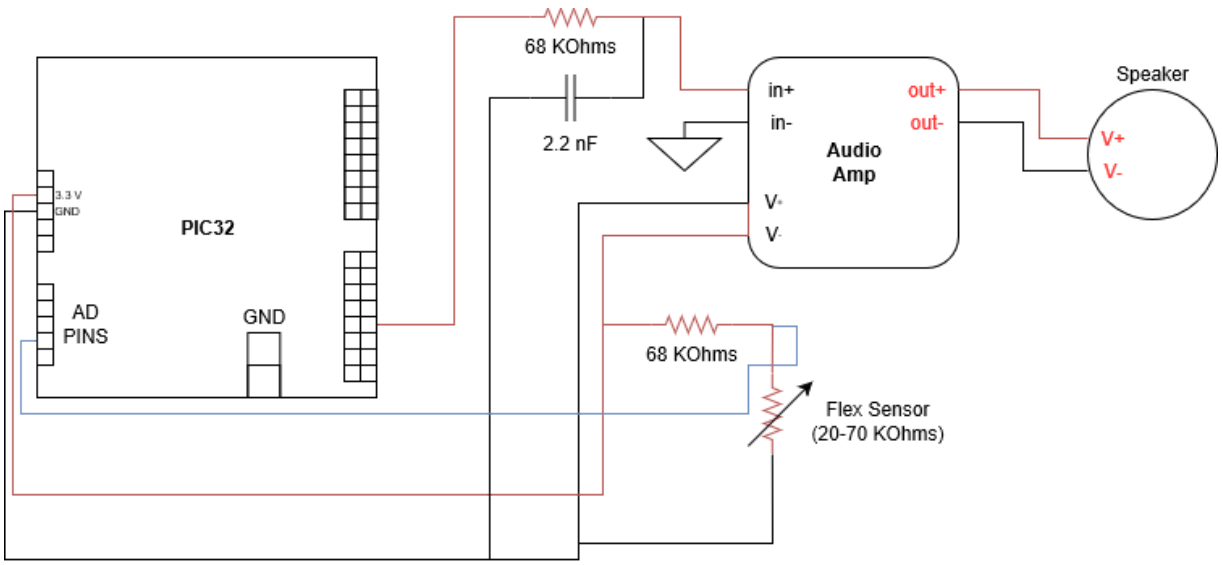
1.3 Testing the Flex Sensor

Now the flex sensor is hooked up to the audio amp and the speaker. A diagram of the circuit can be seen below. Notice that the audio is incredibly scratchy. This is because of the rapidly changing AD readings from the flex sensor. To combat this, an RC low pass filter is used in tandem with some software filtering techniques. In the code, notice that for the AD reading to change, it must read 3 values consecutively that must be within a delta range of the previous reading.

Pseudo-Code

```
main(){
            AD = AD_ReadADPin(AD_A2)
    while (1) {
       newAD = AD_ReadADPin(AD_A2);
        if (abs(AD - newAD) > DELTA) {
           newAD = AD_ReadADPin(AD_A2);
           if (abs(AD - newAD) > DELTA) {
              newAD = AD_ReadADPin(AD_A2);
              if (abs(AD - newAD) > DELTA) {
                 AD = newAD;
                 ToneGeneration_SetFrequency(AD);
}
}
}
```

Schematic


68 KOhms
2.2 nF
in+
in-
out+
out-
Audio
Amp
V+
V-
Speaker
V+
V-
3.3 V
GND
PIC32
AD
PINS
GND
68 KOhms
Flex Sensor
(20-70 KOhms)

Part 2: Piezoelectric Sensor

Piezoelectric sensors generate (large) voltages when deflected or vibrated. “Piezo,” Greek for “pressure,”electricity was discovered by the Curie brothers more than 100 years ago. They found that quartz changedits dimensions when subjected to an electrical field, and conversely, generated electrical charge when me-chanically deformed. One of the first practical applications of the technology was made in the 1920’s by another Frenchman, Langevin, who developed a quartz transmitter and receiver for underwater sound—the first SONAR.

Due to the nature of the response vs frequency, most piezo sensors are used in the flat region away from their resonant peaks, and the DC response is almost zero (that is, they produce a signal to a change in deflection).The output of the piezo element can be increased by laminating strips together in clever ways.The voltages these sensors produce can be very high. Care needs to be taken to ensure that these voltages don't reach the inputs to sensitive electronics without snubbing them to tolerable levels (see Fig. 2).

2. Procedure

2.1 Capture the taps (analog/digital)

Testing the Piezo sensor with the oscilloscope, it was able to generate a 12v signal from peak to peak. This is enough to damage the AD pins on the pic board so a snubbing circuit will be used.

2.2 Test & Implement

After setting up the snubbing circuit with the Piezo sensor, data was used with the AD pin to determine the range of data for the sensor. After many different strategies were used to combat the wobbliness of the input a simple trigger threshold was used. A trigger threshold was determined to be at ~200 that appeared to work from trial and error. If the sensor passes this threshold at any point, the tone frequency is changed to the value of the flex sensor reading.

Pseudo-Code

```
piezoAD = readAD(PIN3)

if(piezoAD > threshold){

        setFrequency(flexAD)

}
```

Part 3: Musical Instrument Redux

Now use both the piezo and flex sensors to create a new musical instrument given the requirements below:

- “Select” a tone based on the flex sensor (that is, once a tone is activated, it is proportional to the amount of flex the sensor has undergone).
- Reliably Activate selected tone with a tap of the piezo sensor. The tone should turn off after a short duration. This duration should be reset if the piezo is tapped again within this period.
- Produce sound that is clear,audible(hopefully not too loud, but definitely not too quiet, because you're using an audio amp), and smooth

3. Procedure

3.1 Implement the Design

The structure of the circuit can be seen in the circuit diagram in the following pages. Both the Piezo and Flex sensor are used in this circuit to create the specified behavior above. Make sure to use the Piezo in parallel with the 1Mohm resistor as specified. Without this “snubbing circuit” the sensor can create a high enough voltage to damage the AD pins. Make sure to use the filtering techniques for the Flex and Piezo sensor from the previous parts.

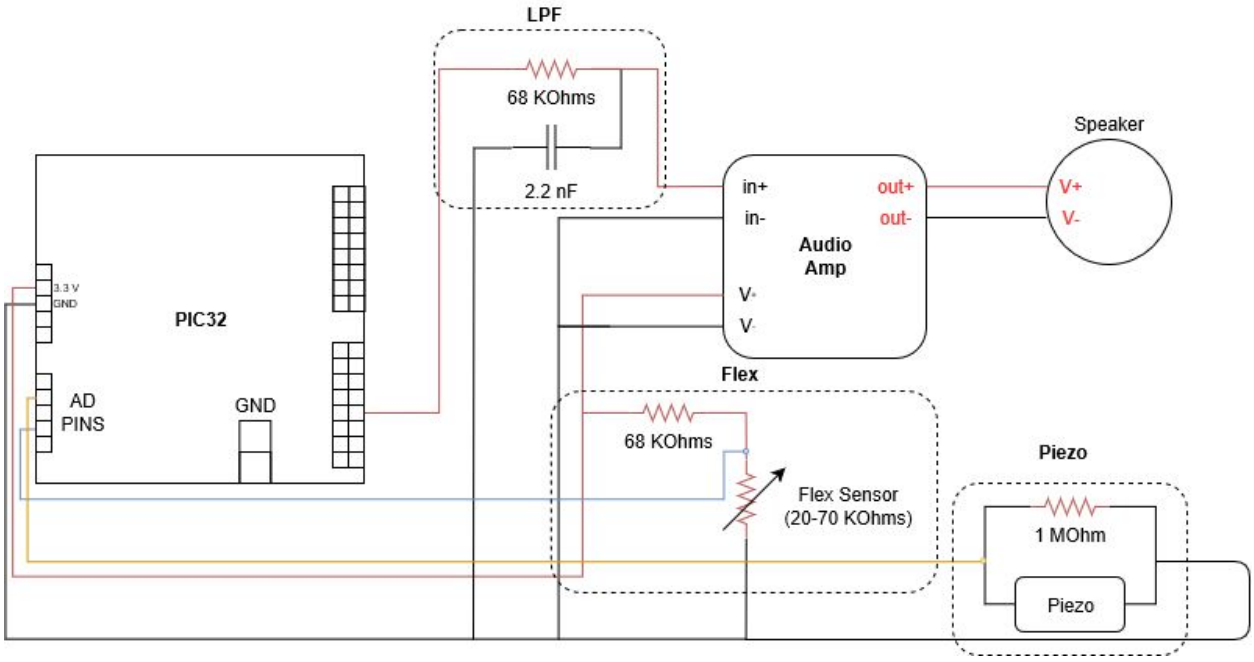
3.2 Testing

Assure that the instrument has a good range of frequencies. This can be done with a linear offset to spread out the AD values sent to the speaker by the Flex Sensor. Also find a reliable way of measuring one second intervals with the PIC32. One way is to use the timer library provided, but a simpler way could be to keep an integer value that increases every loop and use trial and error to measure one second.

Pseudo-Code

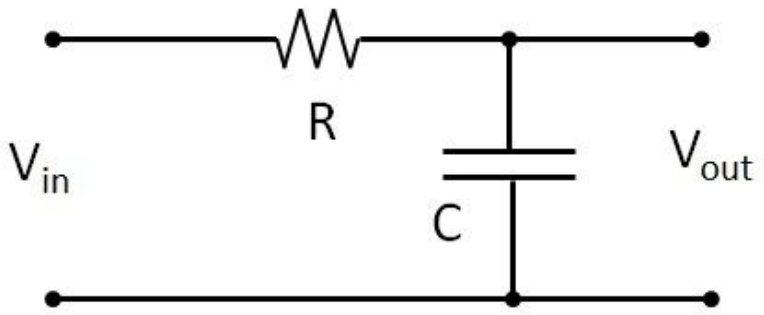
```
PiezoAd = AD_ReadADPin(PiezoPIN)

if(PiezoAD > threshold){
        Time = 1 sec
        ToneGeneration_ToneOn()
        ToneGeneration_SetFrequency(FlexAD)
}
if(Time > 0){
        Time--
}else{
        ToneGeneration_ToneOff()
}
```

Schematic


LPF
68 KOhms
2.2 nF
Speaker
V+
V-
in+
in-
out+
out-
Audio
Amp
V+
V-
3.3 V
GND
PIC32
AD
PINS
GND
Flex
68 KOhms
Flex Sensor
(20-70 KOhms)
Piezo
1 MOhm
Piezo

Part 4: Analog Filtering

Low Pass Filter



Using KCL:

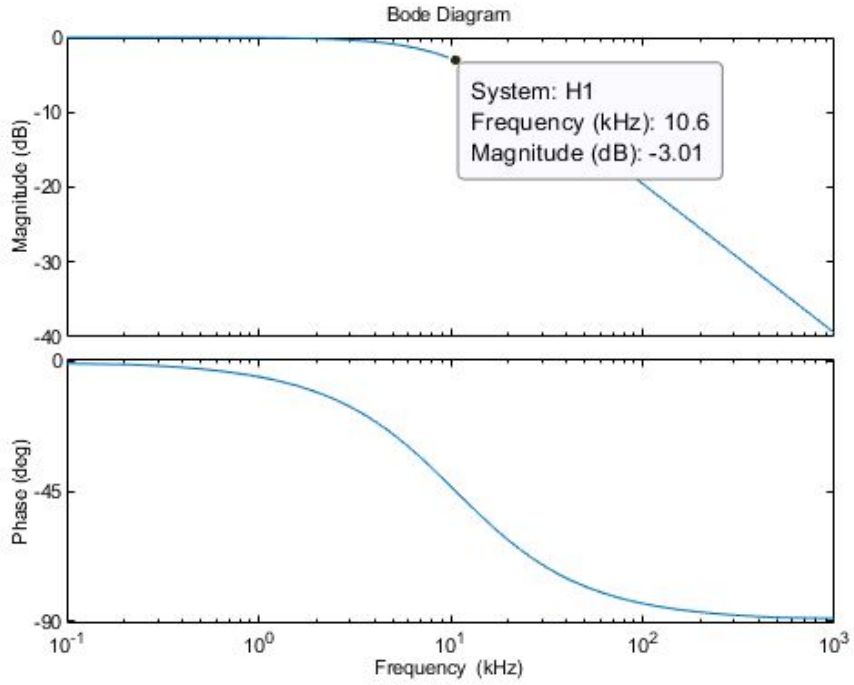
$$\frac{Vin - Vout}{R} = \frac{Vout}{s} \rightarrow \frac{Vin}{R} = Vout(\frac{1}{R} + \frac{1}{s}) \rightarrow H(t) = \frac{Vout}{Vin} = \frac{1}{1 + sRC}$$

4.1 Mathematical Analysis:

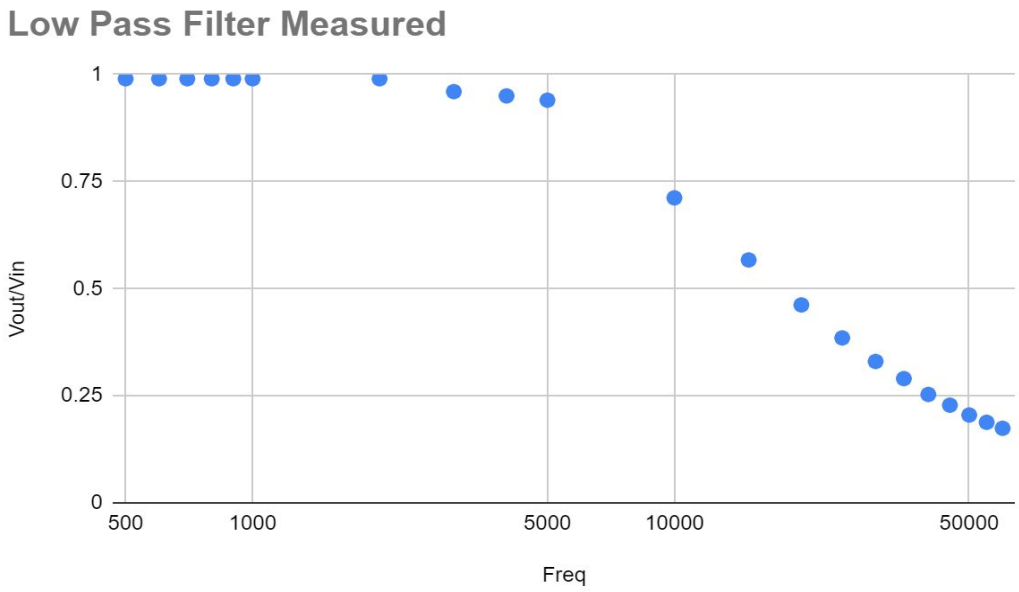
With mathematical analysis on an RC circuit, the transfer function and the cutoff frequency can be determined. The cutoff frequency of a filter is -3dB or ~.707 Vin. See the KCL voltage analysis below that yields the transfer function:

$$H(t) = \frac{1}{1 + sRC}$$

Graph transfer function of circuit with R = 10K, C= 1.5 nF. Cutoff Frequency ~10.06Khz.

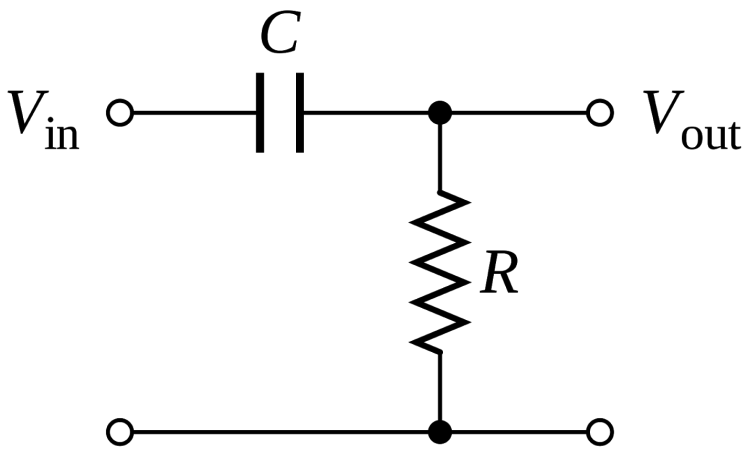


4.2 Measured Analysis:



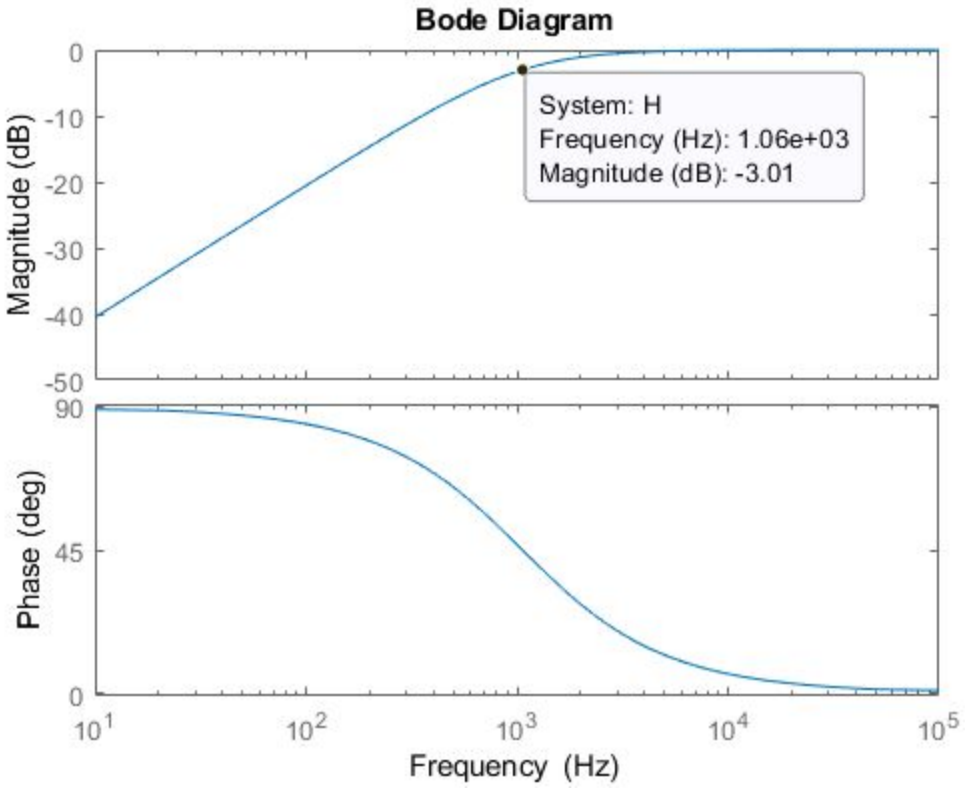
See back for data tables

High Pass Filter

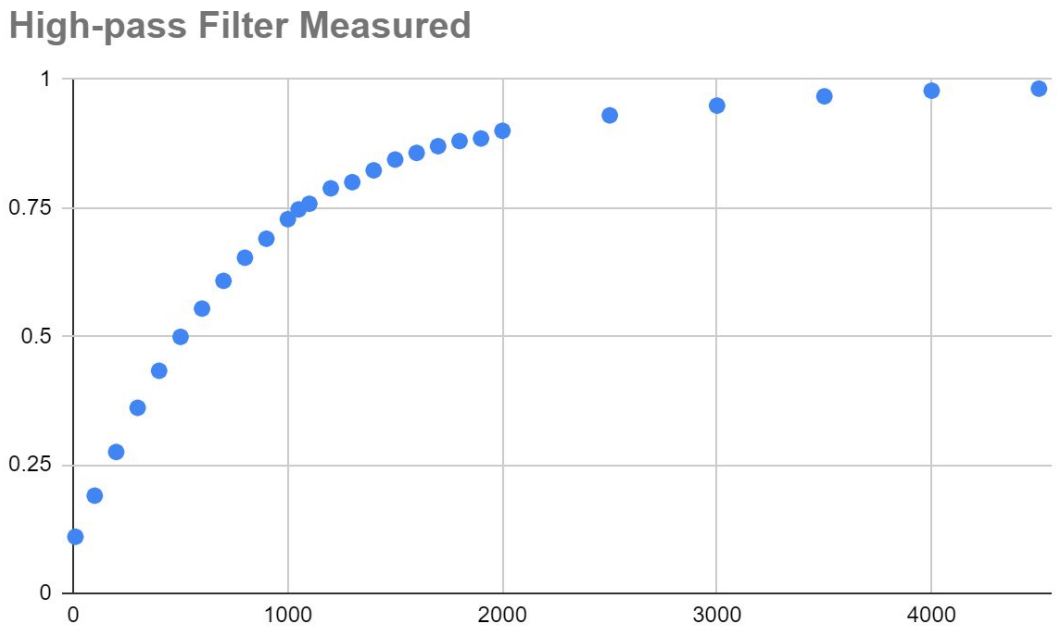


4.3 Mathematical Analysis:

The Transfer function was determined from mathematical KCL analysis seen below. Using the same resistor and capacitor values, the cutoff frequency is the same for the circuit



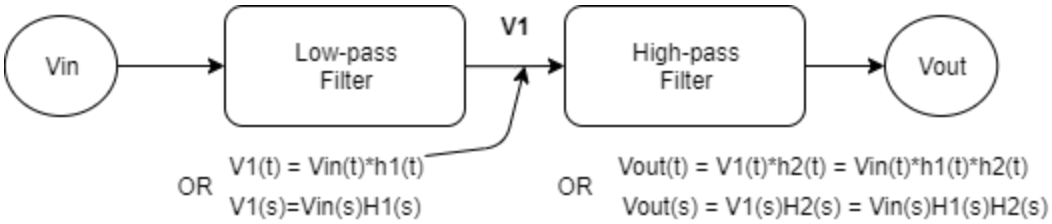
4.4 Measured Analysis:



See back for data tables

Band-Pass Filter:

A band-pass filter has a range of frequencies that pass and therefore two cutoff frequencies on the higher and the lower end. In theory, we can simply cascade the high pass and low pass filter to achieve this filter. In reality, the downstream filter carries some impedance that affects the ideal values generated by the first filter. Below is an idealized band pass filter.



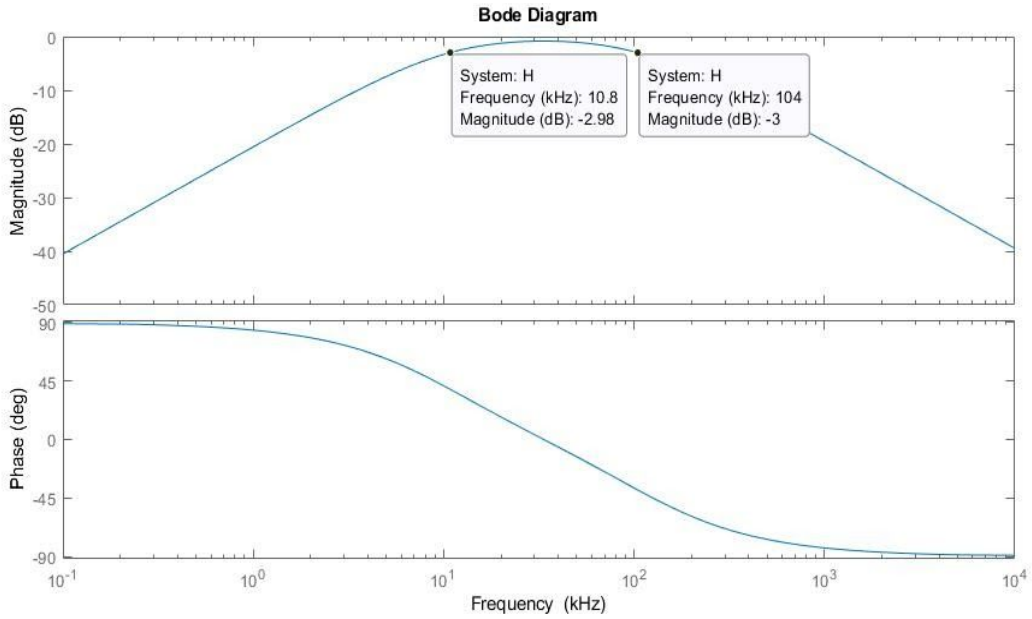
Now cascading the filters in the format featured above, the transfer functions can be multiplied together as shown:

$$H(s) = H_1(s)H_2(s) = (\frac{sR_1C_1}{1+sR_1C_1})(\frac{1}{1+sR_2C_2})$$

$$H(s) = \frac{sR_1C_1}{(1+sR_1C_1)(1+sR_2C_2)}$$

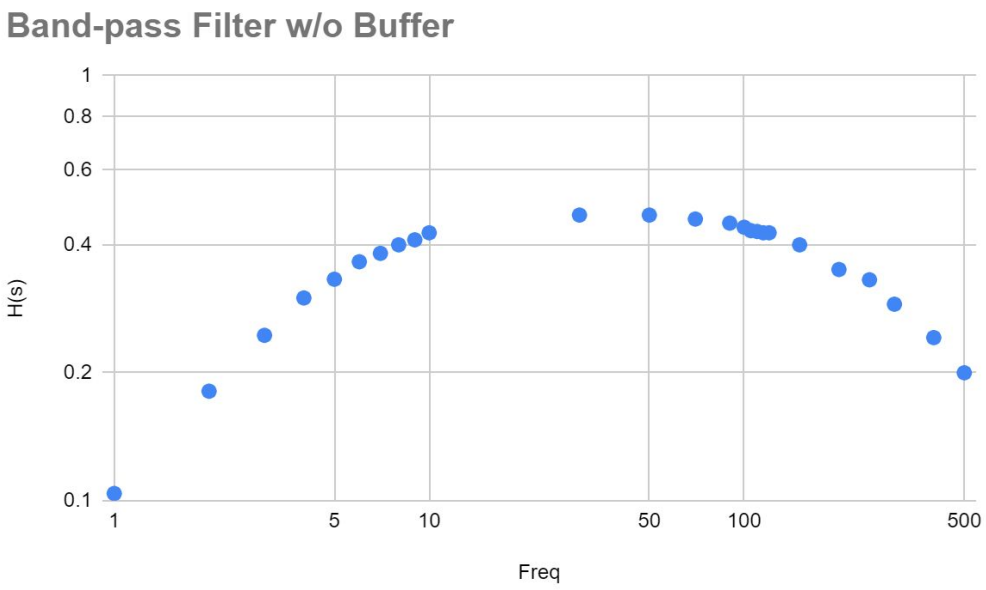
4.5 Mathematical Analysis

Using values R1=1K, R2=10K, C1=1.5 nF, C2=1.5nF, bandpass should appear like bode plot below.

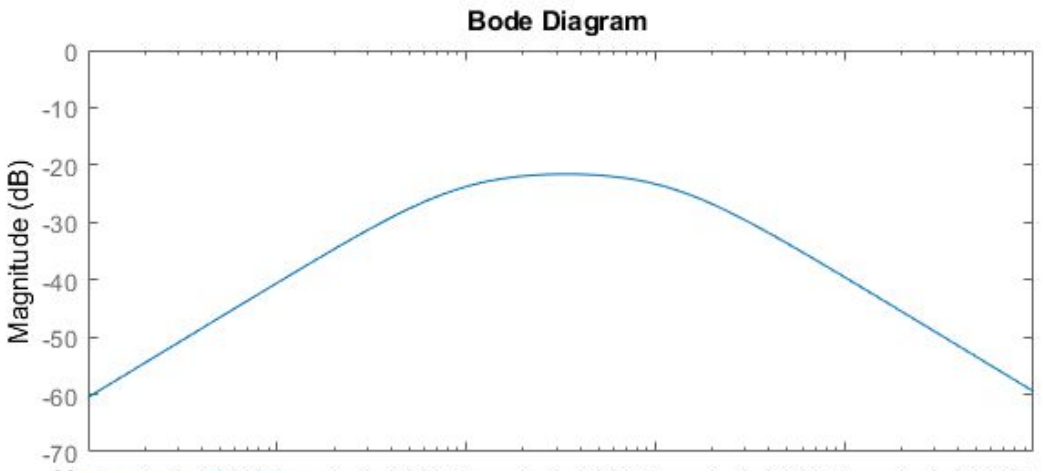


4.4 Measured Analysis w/o Buffer:

But after doing analysis using the real components, graphed between these frequencies, the bode plot below was generated:



Notice that the output never reaches 0dB, and loses about half the input voltage through the filtering. This is because of impedance coming from the second filter. If KCL analysis is done on the circuit diagram then a different function is generated. Which matches our measured graph.



KCL Analysis

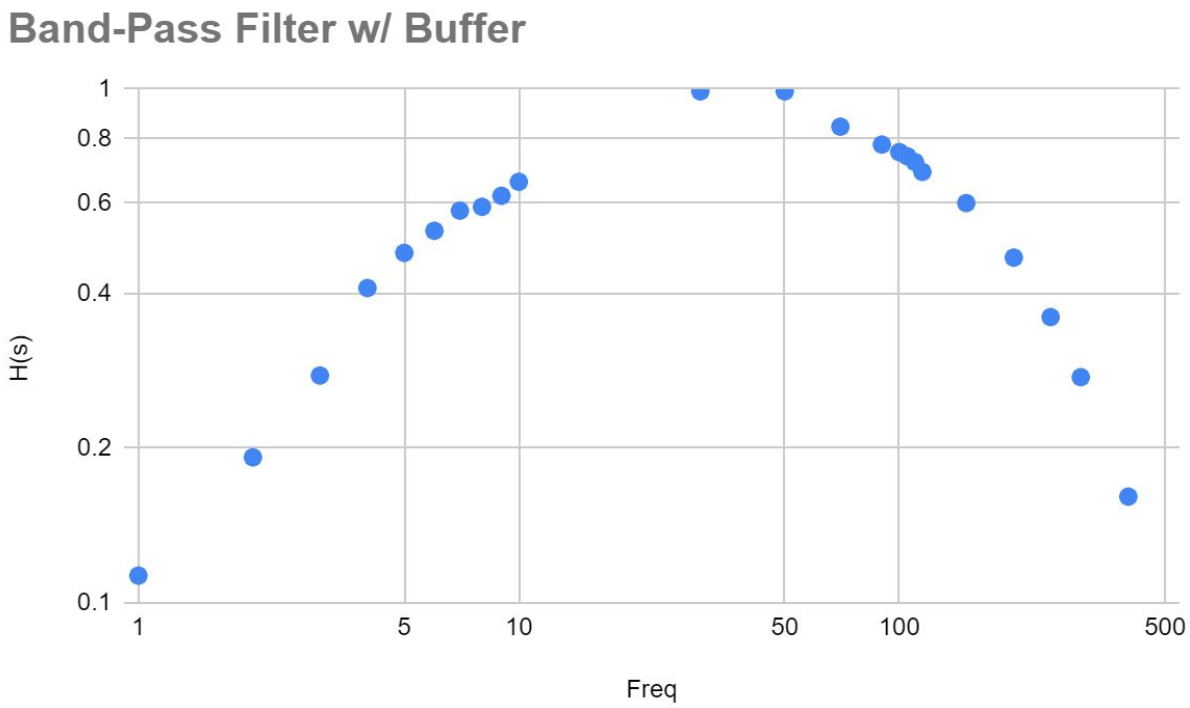
The KCL work is shown in the references in the back. The transfer function calculated is:

$$H(s) = \frac{sR_1C_1}{s^2R_1R_2C_1C_2 + s(R_1C_1 + R_2C_2 + R_1C_2) + 1}$$

The equation is nearly the same as the Ideal Transfer functions except for the s*R1*C2 term.

4.4 Measured Analysis with Buffer:

Since the downstream filter creates an impedance upstream in the bandpass filter, a buffer will be used in between the output of the first filter and the input of the second. This blocks any impedance. Below is the measured BandPass with the buffer:



Notice the band is between 10Khz-100Khz like the ideal Band-Pass.

Conclusion

In conclusion, the Piezo, Flex sensors and more analog filters have been introduced into the class lexicon. Sensors used are often cheap with extreme variation in values which requires good filtering techniques using both software and hardware. A better understanding of analog filters is useful in creating expected values.

REFERENCES

Linearization

THETA	AD Value
0	414
0	414
0	414
0	414
0	414
0	415
0	414
0	414
0	415
0	414
11.25	476
11.25	476
11.25	476
11.25	476
11.25	476
11.25	476
11.25	475
11.25	475
11.25	474
11.25	473
11.25	473
11.25	474
22.5	504
22.5	503
22.5	502
22.5	501
22.5	500
22.5	501
22.5	500

22.5	498
22.5	498
22.5	497
33.75	547
33.75	545
33.75	545
33.75	545
33.75	545
33.75	545
33.75	545
33.75	545
33.75	545
33.75	545
33.75	545
45	551
45	553
45	551
45	547
45	552
45	551
45	546
45	543
45	544
45	551
56.25	561
56.25	560
56.25	560
56.25	560
56.25	559
56.25	560
56.25	558
56.25	558
56.25	558
56.25	557
67.5	619

67.5	618
67.5	617
67.5	617
67.5	616
67.5	617
67.5	615
67.5	615
67.5	616
67.5	614
78.75	711
78.75	713
78.75	707
78.75	706
78.75	704
78.75	707
78.75	712
78.75	706
78.75	708
78.75	709
90	737
90	737
90	738
90	733
90	738
90	739
90	740
90	741
90	743
90	743
90	745

Low Pass Filter

Freq (Hz)	Vout/Vin
500	0.99
600	0.99
700	0.99
800	0.99
900	0.99
1000	0.99
2000	0.99
3000	0.96
4000	0.95
5000	0.94
10000	0.712
15000	0.567
20000	0.462
25000	0.385
30000	0.33
35000	0.29
40000	0.253
45000	0.228
50000	0.205
55000	0.188
60000	0.174

High Pass FIlter

Freq(Khz)	H(s)
1	0.01
10	0.09
20	0.175
30	0.261
40	0.333
50	0.399
60	0.454
70	0.508
80	0.553
90	0.59
100	0.628
105	0.647
110	0.658
120	0.688
130	0.7
140	0.723
150	0.744
160	0.757
170	0.77
180	0.78
190	0.785
200	0.8
250	0.83
300	0.849
350	0.867
400	0.878
450	0.882

Freq(Khz)	H(s)
1	0.104
2	0.181
3	0.245
4	0.3
5	0.332
6	0.365
7	0.382
8	0.4
9	0.411
10	0.427
30	0.47
50	0.47
70	0.46
90	0.45
100	0.44
105	0.432
110	0.43
115	0.427
120	0.427
150	0.4
200	0.35
250	0.331
300	0.29
400	0.242
500	0.2

BandPass w/o Buffer

Freq(KHZ)	H(s)
1	0.113
2	0.192
3	0.277
4	0.41
5	0.48
6	0.53
7	0.58
8	0.59
9	0.62
10	0.66
30	0.99
50	0.99
70	0.845
90	0.78
100	0.754
105	0.74
110	0.721
115	0.69
150	0.6
200	0.47
250	0.36
300	0.275
400	0.161
500	0.075