

# Exploring Sensor Signal Processing

How to manage unwanted interference in sensor circuits with passive R-C filters.

## Table of Contents

|  |    |
|--|----|
| Chapter 1: Getting to Know Low Pass R-C Filters .....                                  | 2  |
| Introduction .....   | 2  |
| Getting to Know Our Filter Components .....  | 2  |
| R-C Charging over time .....   | 4  |
| RC Time Constant.....  | 5  |
| Introduction to R-C Filters.....   | 6  |
| Circuit Simulation for a Low-Pass Filter in Multisim .....                             | 6  |
| Computing Output Voltage from a Low Pass RC filter at Specific Frequencies .....       | 8  |
| Example 1: Calculate the output voltage of the filter at a frequency of 1000Hz. ....   | 8  |
| Example 2: Calculate the output voltage of the filter at a frequency of 20,000Hz. .... | 9  |
| Example 3: Calculate the output voltage of the filter at a frequency of 30,000Hz. .... | 10 |
| Example 4: Calculate the CUTOFF frequency of our filter. ....                          | 10 |
| Signal Phase Shift .....   | 11 |
| Summarizing the Frequency Response of Our Low Pass Filter .....                        | 14 |
| Second Order Low-Pass Filters.....   | 15 |
| Simulation of Our Second-Order Circuit in Multisim .....                               | 16 |
| Chapter 2: Getting to Know High Pass R-C Filters .....                                 | 18 |
| Capacitive Reactance in a High Pass Filter .....                                       | 18 |
| Simulation of Our Circuit in Multisim .....  | 18 |
| How to Measure Phase Shift with the Oscilloscope .....                                 | 20 |
| Second Order High-Pass Filters.....  | 22 |
| Chapter 3: Working with Automotive Sensors .....                                       | 23 |
| Introduction .....   | 23 |
| Bench Testing a Knock Sensor .....   | 24 |
| <b>Experiment 1:</b> Process Signal Data with Single-Pole Hardware Filters.....        | 25 |
| Designing a Low Pass Filter in National Instruments Multisim.....                      | 26 |
| Applying the Low Pass Filter .....   | 27 |
| <b>Experiment 2:</b> Using a Two-Pole Filter .....                                     | 28 |
| Simulation vs Actual Filter Responses .....  | 29 |
| Summary and Conclusions and Going Further .....  | 30 |

## Chapter 1: Getting to Know Low Pass R-C Filters

### Introduction

Reading analog sensor data in industrial and automotive environments is a real challenge. Engine noise, alternator whine, fuel injector pulses, large 3-phase motors, and ignition spark all contribute to the collection of unwanted electro-magnetic interference (EMI) that plagues our sensor readings. These unwanted frequencies degrade and mask the sensor data we are attempting to collect, interpret, and display to the operator. The data must be robust. Safety systems and collision-avoidance sensors cannot simply work most of the time, for example. They must work all the time.

This paper will explore the possibility of utilizing hardware filtering circuits to achieve the goal of removing EMI and preserving the integrity of our readings. This approach is presented as an alternative to software filtering strategies. It is strongly believed that removing electrical noise at the source is superior to the use of software algorithms alone. However, in some cases, it may make sense to use both. In any case, we will focus on how we might process raw sensor readings so as to make microcontroller interpretation of the readings more consistent.

Before we learn how to create and use filters, it would behoove us to get a basic understanding of how the electronic components for our filters actually work. So, let's jump in!

### Getting to Know Our Filter Components

#### Resistors

First, we will examine resistors. A resistor, as its name implies, resists something. In this case, they resist the flow of electrons in an electronic circuit. Due to the way they are made, electrons have difficulty moving through them. The flow of electrons in a conductor (like a copper wire) is what we call current. Current is signified by the letter "I" and is measured in amperes or just amps. The more current that flows in a circuit, the more work it can do for us. Imagine a small motor attached to a 9v battery. The small battery can only supply so many electrons because of its size. Consequently, I can only power a very small motor like one used on a robot. To run a larger motor, I will need a much larger battery because I need more current.



Figure 1: Typical Carbon Resistors

Sometimes however, we need to cut down the voltage to a particular device. Say I have my 9v battery and I want to run a 5v light bulb. If I don't reduce the power going to the bulb it will overheat and burn out. So, I could use a resistor or two to divide the 9v down to 5v so the bulb will work properly.

Resistors (R) are made of various materials. Many are made of a mixture of carbon and other ingredients. The amount of carbon will determine how many electrons are blocked. Resistors are made in many different values of resistance. Their values are expressed in ohms ( $\Omega$ ). The definition of an ohm goes like this:  $1\Omega$  is the amount of resistance in a circuit with 1 volt and 1 amp of current running through it. We calculate R using Ohm's Law:  $R = \text{Voltage} / \text{Current}$ . Since voltage is referred to as "E" and current is referred to as "I", our formula becomes  $R = E/I$ . In Figure 1, you can see that the resistors are identified by colored bands around them. The color pattern tells us their resistance value.

Finally, to give some perspective on this, when  $R = 1,000,000\Omega$  that is a huge amount of resistance. On the other hand, a resistor of  $10\Omega$  is very small. If you connect an LED to a battery, you would typically use a  $220\text{-}470\Omega$  resistor to limit the current to keep the LED from overheating and burning out. You need to know that resistors convert the electrical energy into heat. So, they come in different sizes ( $\frac{1}{2}$  watt, 1 watt, etc. so they can withstand that heat.

Next, we need to understand a few things about capacitors.

### Capacitors

Figure 2 shows a typical capacitor. This one has a positive (+) and negative (-) lead and must be connected correctly. Many other capacitors are not polarized and can be connected in either direction. When we apply a DC voltage to a capacitor from a battery it will begin to charge up. That means electrons will be stored in the plates of the capacitor. After it is charged it functions like a small battery. Please remember that capacitors store electrical charges. Resistors do not store any energy.

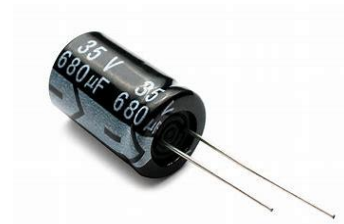


Figure 2: Typical Electrolytic Capacitor

When you connect a capacitor to a battery, it charges up almost instantly. However, there are many times when we want to control the charge and discharge rate of a capacitor. For example, electric vehicles use a pre-charge circuit that slowly lets voltage build up on the motor before the driver accelerates so it doesn't burn the motor and other components. Figure 3A shows such a circuit. There is a resistor that is in series with the capacitor. The resistor is used to slow down the flow of electrons from the battery as they enter the capacitor. This slows down the charging time of the capacitor. This type of circuit is also used in electronic filters as we will see shortly.

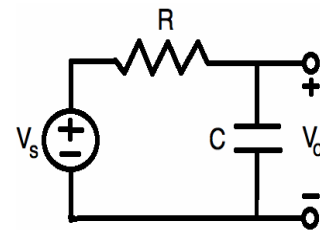


Figure 3A: Simple R-C Circuit

I want to leave you with a visual image of how this charging works. If I connected a capacitor across the battery and an LED or bulb across the capacitor as in Figure 3B, the LED would glow full bright as soon as I pushed my switch (S1) because electrons can flow from the battery, through R1 and through the LED and then return to the battery. In this condition, the LED is being powered by the *battery*. While S1 is closed, some of the battery's energy flows through C1 and it begins to charge. At some point, the capacitor cannot store any more electrons, so it becomes an energy storage device.

If I were to remove the battery power from the circuit by opening the switch, the stored energy in C1 will be discharged or returned into our circuit and current will flow through the LED and back to C1 just as it did when powered by the battery. While C1 has a charge, it will keep the LED glowing until the charge in C1 becomes 0V. So, you would notice that the LED would glow for a short time and then slowly dim out as C1 loses its charge. The size of the

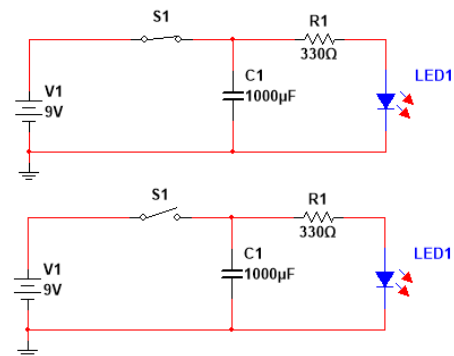


Figure 3B: Simple Capacitor and LED Circuit. When S1 is opened, C1 powers the LED for a second or two until its stored energy is depleted.



capacitor determines how long it takes to become fully charged and discharged. Keep in mind that when electrons/current flows we can do work. Otherwise, we cannot.

This circuit could be used to dim the interior lights of your car, for example. When you open the car door, the battery feeds current to the interior light and charges up a capacitor. When you close the door, the battery is no longer supplying power but the capacitor is. It keeps the lights on for a short time. But, since the capacitor can only store just so much energy, its voltage decreases and the lights slowly dim. Simple. No need for a computer or software. You can just use some discreet parts to do the job. You can choose how long to leave the lights on before they dim out by choosing the resistor-capacitor values. This type of circuit is used in millions of products and that is why I want you to understand how they work. I am sure you will find many uses for these components as you study electronics.

### R-C Charging over time

Let's look at a scope image of this charging process. Without the use of a resistor in our circuit, the capacitor would charge almost instantaneously as current rushes into it. Without a charge on it, a capacitor acts like a piece of wire... a short circuit. For an instant, current rushes in. As the capacitor charges, the voltage increases and the current decreases.

We have seen how resistors are used to control current flow. By adding the resistor in series with my capacitor (C) in Figure 3A, it slows down the flow of the current and creates a delay so it will take a bit of time to charge the capacitor.

Figure 4 shows how a R-C circuit controls the charge time. This image displays the charge curve for a  $2200\Omega$  (2.2k) in series with  $1000\mu\text{F}$  (microfarad) electrolytic (polarized) capacitor.

When I connect the battery, current begins to flow from the resistor into the capacitor. With the values I have chosen, it takes about 10-11 seconds for the capacitor to fully charge. The cool thing is that you can choose how long it takes for this charging to take place by simply changing the value(s) of R, C, or both. There are many online calculators to help you. Let's look at this in more detail.

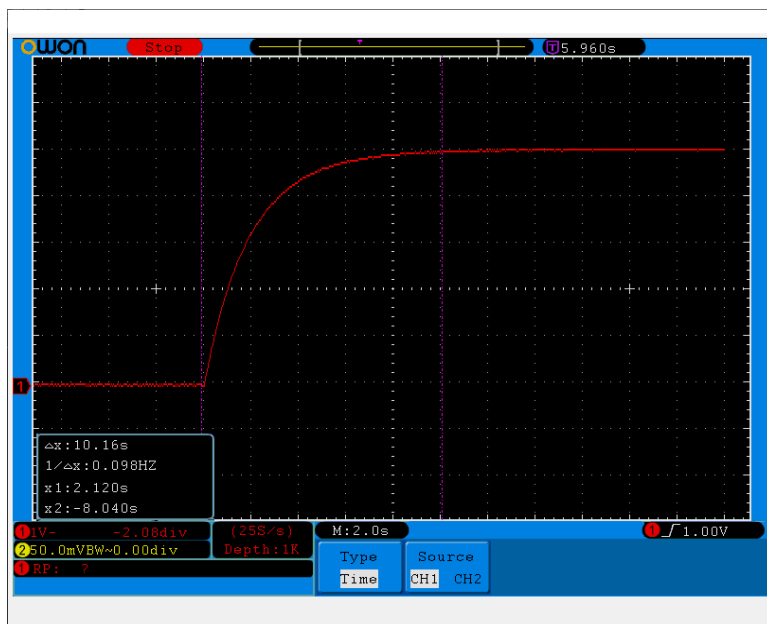


Figure 4: Voltage Increases from zero to near battery voltage in 5-time constants ( $2.2 \times 5 = 10-11$  sec).

## RC Time Constant

To determine how long it takes to charge a capacitor in a DC circuit you simply multiply the resistor value times the capacitor's value. The result is called the *RC Time Constant*. It is symbolized by the Greek letter Tau ( $T$ ). The formula is:

$T = R \times C$  where  $T$  is in seconds,  $R$  is in ohms and  $C$  is in Farads. A farad is a huge quantity so capacitors are created with values in microfarads ( $\mu\text{F}$ ), nanofarads (nf), or picofarads (pf). For reference,  $1000\mu\text{F} = .001\text{Farads}$  or  $1,000,000\text{nF}$  or  $1,000,000,000\text{pF}$ .

**Example:** The RC time constant for my circuit is  $2200\Omega \times .001\text{F} = 2.2$  Seconds. This means that after 2.2 seconds, my capacitor will be charged to about 63% of the applied voltage. In this case, it is based on the voltage of my battery. During the next time constant from 2.2 – 4.4 sec., it will charge another 63% and so on. It takes 5 RC time constants to fully charge the capacitor so in my case it takes about 11 seconds ( $5 \times 2.2$  sec). When the voltage in the capacitor reaches 99% of the applied or battery voltage, we say it is fully charged.

If I removed the battery in Figure 3A and replaced it with a switch, I could discharge the capacitor by closing the switch and allowing current to dissipate through the resistor. It would take the same amount of time to discharge the capacitor through my resistor as it took to charge it in the first place. So, after 11 seconds, the capacitor would be out of energy.

Remember I said the capacitor is like a small battery? Well, the electrons that were stored in the capacitor's plates now flow through the resistor and any component connected to the circuit. The voltage slowly diminishes until it reaches 0 volts. At that point, the capacitor is fully discharged.

On the oscilloscope image in Figure 5, you can see that the voltage goes from 9V (battery voltage) down to zero at the same rate as when the capacitor charged. The voltage exponentially decays until it reaches 0 volts. At this point there is no more energy stored in the capacitor. Again, it takes 5-time constants to discharge. In our case that is  $2.2 \times 5 \sim 11\text{sec}$ .

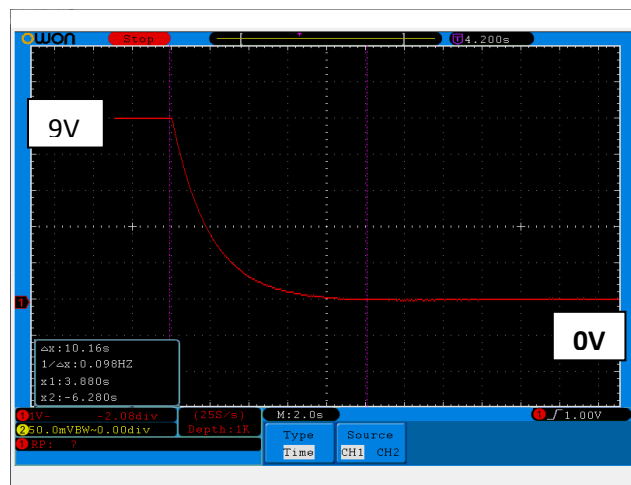


Figure 5: Discharge curve for the same capacitor.

## Introduction to R-C Filters

Now I want to show you how filters work. By combining resistors and capacitors (or resistors and inductors/coils), we can filter out unwanted frequencies or EMI noise. This is very handy when we want to filter out interference that is disrupting the readings of a sensor. Whether we are working with a microcontroller, automotive application, or a musical device, these filters are crucial when we need clean sensor signals or clean sound.

This circuit works on the principal that capacitors react differently to low frequencies than to high ones. In other words, their resistance to current flow is inversely proportional to frequency. At low frequencies, their resistance is high and as frequencies increase, their resistance decreases. Resistors do not react to changes in frequencies. Only capacitors and inductors do. When capacitance/inductance *and* frequency are considered together, the resistance to current flow is called *reactance* and it is designated as  $X_c$ . The reactance of an inductor is designated  $X_L$ . Thus, capacitors and inductors are called *reactive components* because of their unique response to changes in frequency. Now let's look at a simulation in National Instrument's Multisim application to see how we can take advantage of these reactive devices.

First, we will explore low-pass R-C filters. These circuits allow low frequencies to pass through but limit or block high frequencies. This is the circuit that can help us remove high frequency noise in our microcontroller power supply circuits, for example. For this demonstration, I will apply an AC signal to the filter network to simulate analog sensor data. Let's see how our filter reacts to changing frequencies!

## Circuit Simulation for a Low-Pass Filter in Multisim

### Our Circuit Setup:

- For this demo, we will use a 1000 $\Omega$  (1K) resistor and 15nF (.015 $\mu$ F) ceramic capacitor to filter out high frequencies.
- Source is an Input Signal: 10V P-P Sine wave (red Trace on scope)
- Output of RC filter shown in **yellow** trace on scope.
- Figure 6 shows my RC test setup in the simulator.
- Figure 7 is a simple schematic of the circuit.

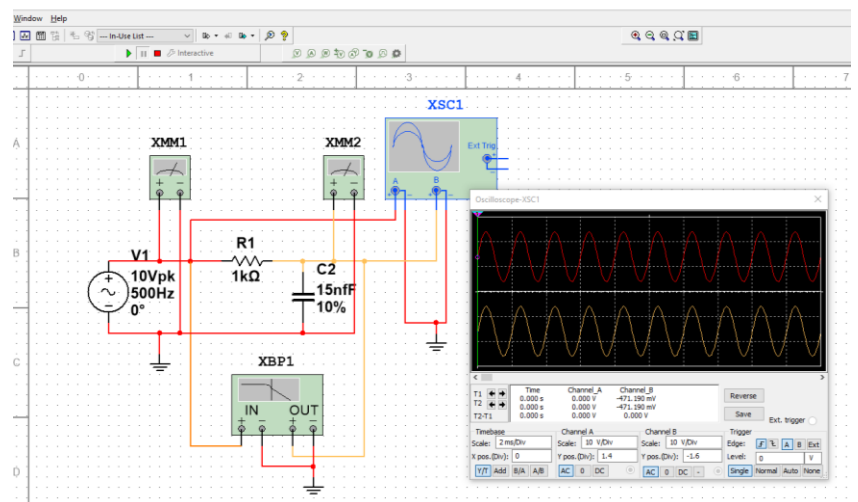


Figure 6: Multisim Layout of My RC Low pass Filter Circuit

Let's figure out how the circuit affects my output signal. At low frequencies, the reactance(resistance) of C is high. Recall the reactance of a capacitor and frequency are inversely related. So a low frequency sound "sees" the resistor as an easier path to the output because the capacitor has very high reactance.

When the frequency is high, C provides an easier path for the electrons to travel and they are routed to ground (B-) and are not passed to the output. That is because the "resistance" in C is less than the resistance in R. This circuit could be used as a Bass control on an amplifier. Low frequencies pass through and higher frequencies are blocked.

To reverse the filter design, just reverse the position of R and C. Then high frequencies would be passed and low frequencies would be blocked. Thus, you would have a *high-pass* filter.

As you will soon see, we can set the threshold by which low frequencies are passed and high frequencies are blocked.

The terms "low" and "high" are arbitrary and determined by the circuit designer. These circuits are used in speaker crossover networks to route high sounds to the tweeter and low sounds to the woofer or sub-woofer.

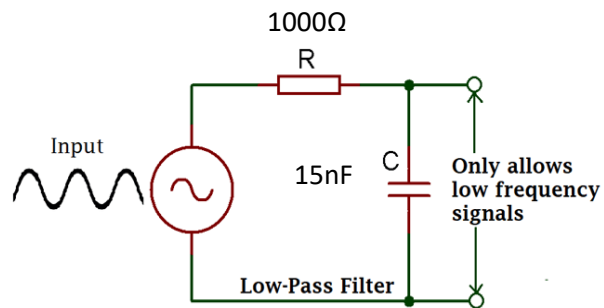


Figure 7: Simple R-C Low-Pass Filter Circuit



## Computing Output Voltage from a Low Pass RC filter at Specific Frequencies

Example 1: Calculate the output voltage of the filter at a frequency of 1000Hz.

1. Compute capacitive reactance at 1000Hz:

$$X_c = \frac{1}{2\pi f c} = 1 \div 2\pi \times 1000 \times (15 \times 10^{-9}) = 1 \div 6280 \times .000000015 = \underline{\underline{10,615\Omega}}$$

2.  $V_{OUT} = (V_{IN} \times X_C) \div \sqrt{R^2 + X_C^2} = 10 \times 10,615 \div \sqrt{1000^2 + 10615^2} = \underline{\underline{9.95V}}$   
The yellow scope trace shows ~9.92V output from the filter (Figure 8).

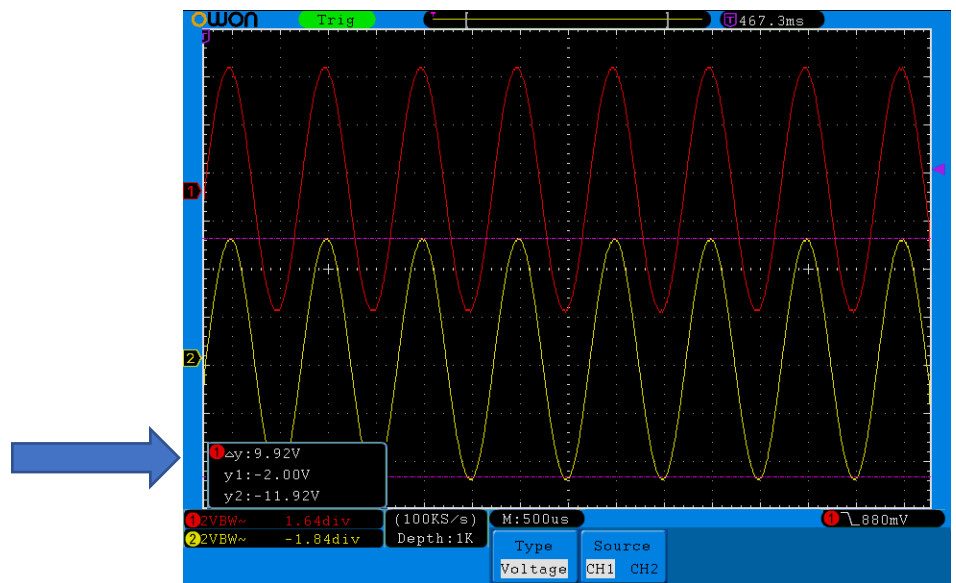


Figure 8: Input & Output Signal at 1KHz

Example 2: Calculate the output voltage of the filter at a frequency of 20,000Hz.

3. Compute capacitive reactance at 20,000Hz:

$$X_c = \frac{1}{2\pi f c} = 1 \div 2\pi \times 20000 * (15 \times 10^{-9}) = 1/ \times .000000015 = \underline{\underline{530.78\Omega}}$$

4.  $V_{OUT} = (V_{IN} \times X_C) \div \sqrt{R^2 + X_C^2} = 10 \times 530.78 \div \sqrt{1000^2 + 530.78^2} = \underline{\underline{4.67V}}$

The yellow scope trace shows ~4.64V output from the filter (Figure 9).

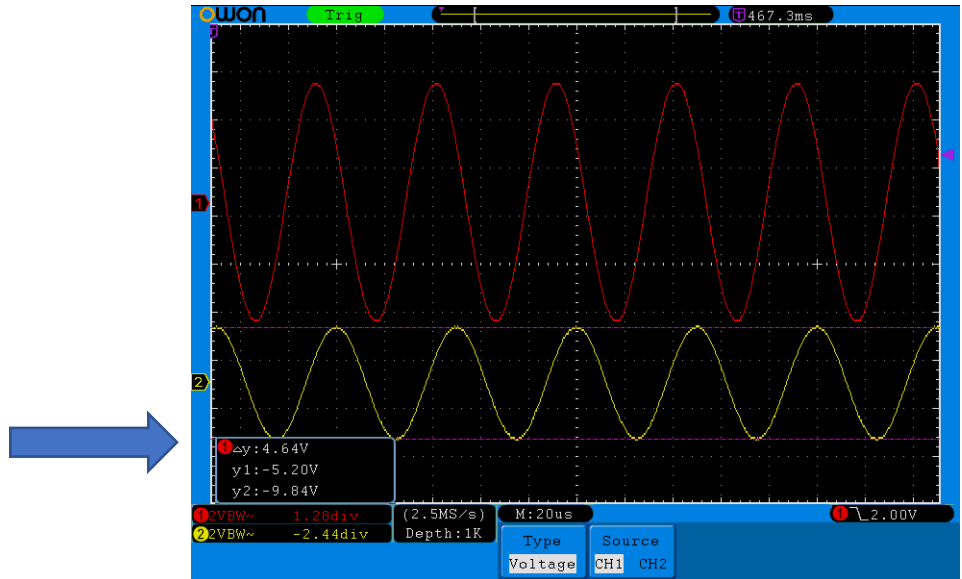


Figure 9: Input & Output Signal at 20KHz

Example 3: Calculate the output voltage of the filter at a frequency of 30,000Hz.

5. Compute capacitive reactance at 30,000Hz:

$$X_c = \frac{1}{2\pi f c} = 1 \div 2\pi \times 30000 \times (15 \times 10^{-9}) = 1/ x .000000015 = \underline{\underline{353.85\Omega}}$$

6.  $V_{OUT} = (V_{IN} \times X_C) \div \sqrt{R^2 + X_C^2} = 10 \times 353.85 \div \sqrt{1000^2 + 353.85^2} = \underline{\underline{3.33V}}$

The yellow scope trace shows ~3.36V output from the filter (Figure 10).

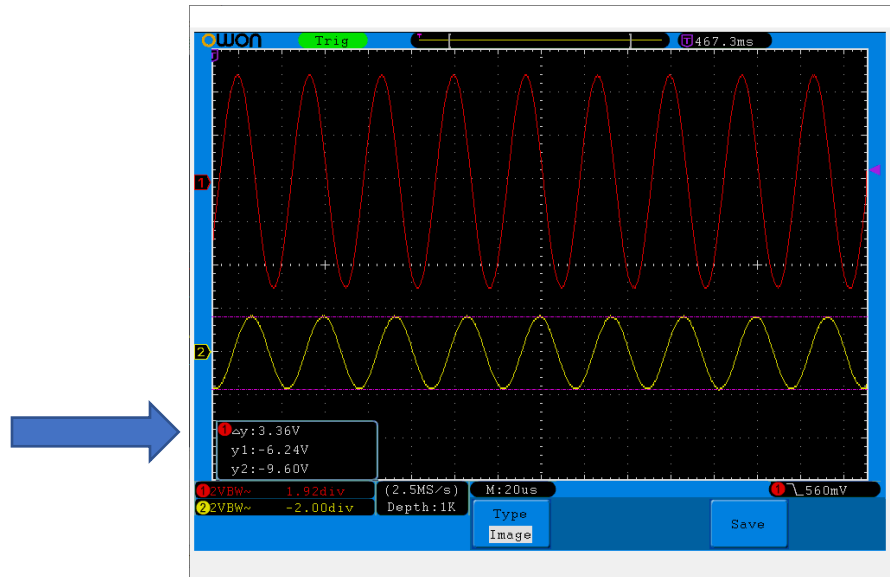


Figure 10: Input & Output Signal at 30KHz.

Example 4: Calculate the CUTOFF frequency of our filter.

We have learned how the filter responds to changes in frequency. As the frequency of our input signal increases, the reactance of the capacitor declines. This allows more current to flow through the capacitor to ground which reduces our output voltage.

The corner/cut-off frequency ( $f_c$ ) formula is  $f_c = \frac{1}{2\pi RC}$

7. Using our 1000 $\Omega$  resistor and 15nF capacitor filter, compute  $f_c$ :

8.  $f_c = 1 \div 6.28 \times 1000 \times .000000015 = \underline{\underline{10,615\text{Hz}}}$

You can see in Figure 11 that the cut-off frequency at -3dB is 10.518KHz which is close to our calculations.

The actual tolerances of our components and the calibration of the instruments account for the discrepancy in the calculated and observed values.

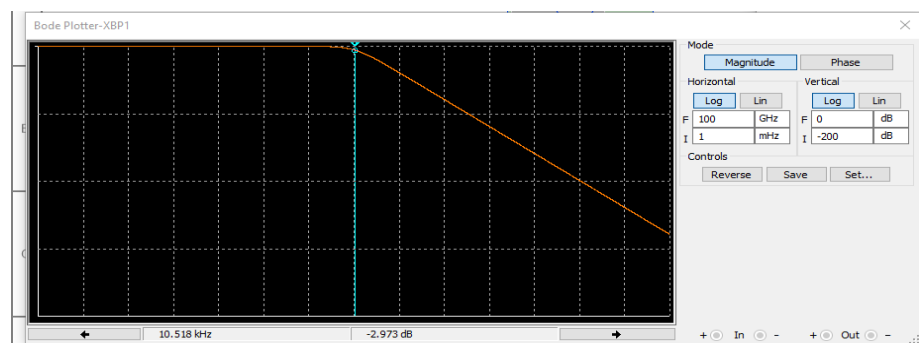


Figure 11: Bode Plot showing the corner frequency of 10,518Hz at ~-3db

When the resistance of the resistor and the resistance(reactance) of the capacitor are equal, the output of the filter voltage is 70.7% of the input voltage. Or, the output is reduced by about 30%. This is equivalent to a -3dB decline in output voltage.

The -3dB level was chosen by electrical engineers because it is the point where the output power level is 50% less than the input. This is used by convention in many applications and is often referred to as the **half-power level**. If you are familiar with voltage dividers, you know that when two resistors of equal value are in series with a voltage, the output across  $R_2$  will be  $\frac{1}{2}$  the input voltage. When we compare two power levels where  $P_{out} / P_{in} = .5$ , this corresponds to a -3dB loss or attenuation in the power level ( $dB_{out} = 10\text{Log} (.5) = -3\text{dB}$ ). But in our case, we are *not* measuring power. We are measuring voltage across a resistance and a reactance. Therefore, the associated power ratio is equal to the square of the voltage ( $P = V^2 / R$ ) so we use a different formula. **The formula for comparing two voltages or two currents in dB is:**

$$dB = 20\text{Log}_{10} (V_{out} / V_{in})$$

The decibel scale is logarithmic and when the output voltage is compared to the input voltage **at the cutoff frequency**, output voltage is always  $.707 \times V_{in}$ .

When the dB is negative, it means the output is less than the input signal. This cut-off frequency defines the boundary between frequencies passed and frequencies blocked. The frequencies beyond the cut-off are attenuated significantly (Figure 6). In a single pole/stage RC filter, the output voltage decreases at about 20dB/decade, where a decade represents a 10x increase in frequency. For example, Table 1 shows that at 10KHz, our output voltage is 7.28V and at 100KHz it is 1.06V which is -19.5dB. Notice the roll off is linear past the cutoff frequency.

## Signal Phase Shift

Because of the nature of how capacitors respond to AC signals at specific frequencies, it takes some time for the capacitor to charge and discharge as the sine wave voltage is applied. Thus, there is a “lag” or phase shift between the input and output voltages of the filter.

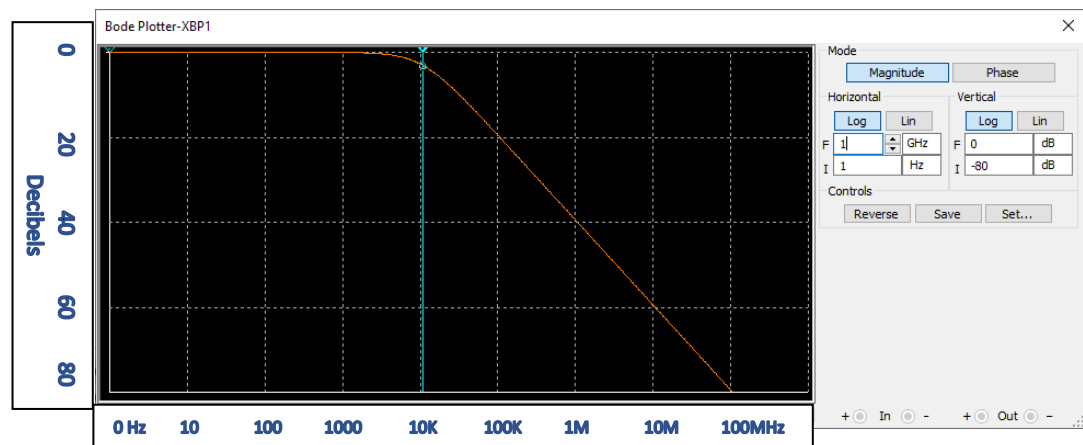


Figure 12: Roll off of the filter is -20dB/decade.

The output signal is delayed with respect to the input signal. On the scope, the yellow line ramps up *after* the input signal is applied.

Look at Figures 9, 10, & 14. Notice the output signal (yellow) is shifted to the right of the red input signal. As the frequency

increases from 1000Hz to 30,000Hz, so does the phase shift. Compare Figure 9 to Figure 14 to see the most noticeable phase shift. At the cutoff frequency, this lag is always  $-45^\circ$ .

To better demonstrate phase shift at different frequencies, I changed the input signal of our filter to be a square wave.

At 5000Hz, the yellow output voltage is nearly equal to the input voltage. In Figure 14, you can see how the capacitor charges by ramping up in response to the input signal. When the input voltage switches to 0V, the capacitor discharges. Then the cycle repeats.

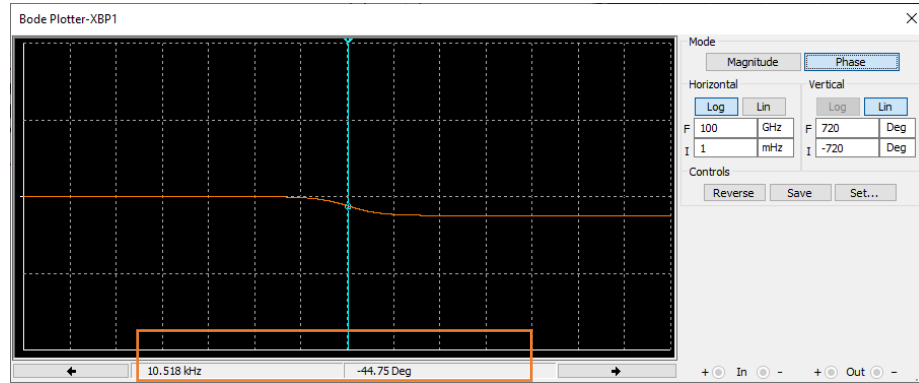


Figure 13: At the cut-off frequency, the output voltage lags the input voltage by 45 degrees.

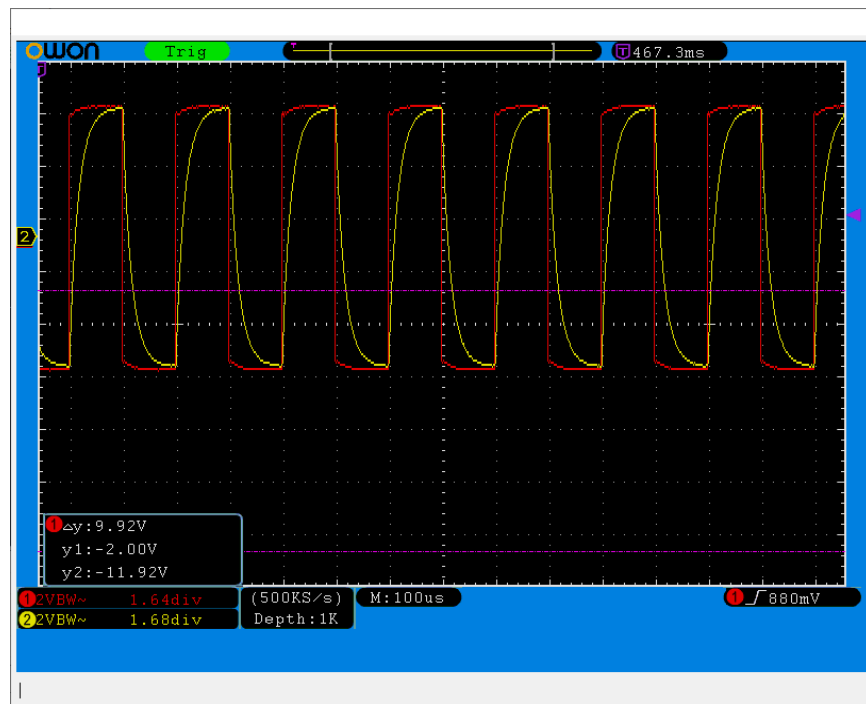


Figure 14: Square wave input signal shows how capacitors charge & discharge. The yellow output voltage is nearly equal to the input voltage at lower frequencies.

Now, notice what happens at higher (25,000Hz) frequencies (Figure 15). This sawtooth pattern emerges. The ramping is steeper and drops off fast. But the amplitude of the yellow output signal is also much lower. In fact, it is 5.75 Volts at the higher frequency.

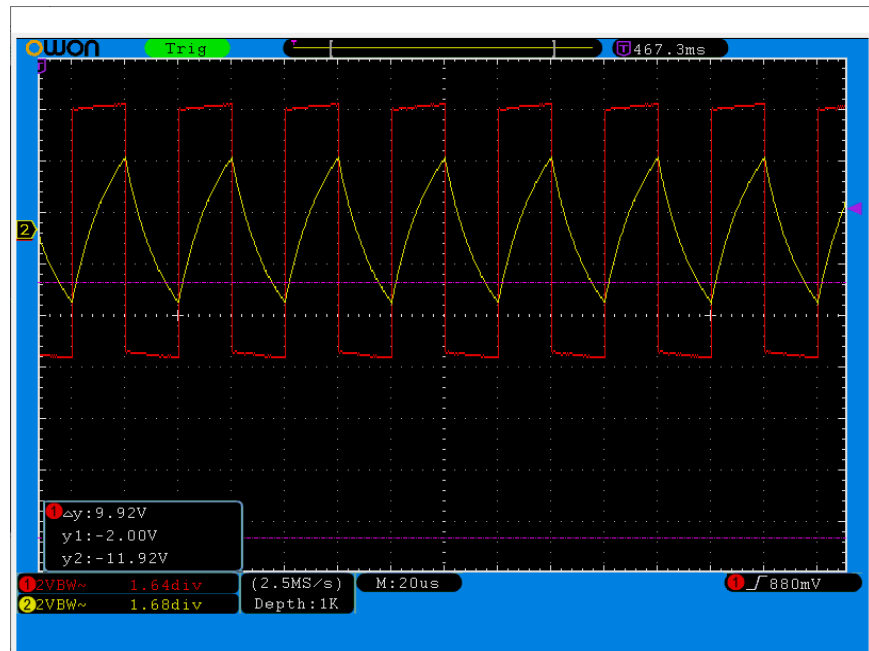


Figure 15: 25,000Hz Square wave shows yellow capacitor voltage is lower than at low frequencies.



## Summarizing the Frequency Response of Our Low Pass Filter

Now that we understand the major properties and characteristics of our filter, it might be helpful to visualize the frequency response in the table below. Notice how the capacitive reactance ( $X_c$ ) and the total impedance ( $Z$ ) of the resistor and  $X_c$  are affected by the input frequency. Also notice that the output of our signal is unchanged until the frequency is increased to  $\sim 500\text{Hz}$ . At that point, the input signal is beginning to be attenuated by the filter. At the cutoff frequency (10,610Hz), the output signal is reduced to about 7.07V or 70% from the input voltage of 10V. Looking at it another way, the signal has been decreased by  $\sim 30\%$  which is a -3dB drop in signal strength. Also notice that the phase angle or phase shift is  $-45^\circ$ . This means the output signal at cut-off is  $45^\circ$  behind the input signal. As the frequency increases, the output signal is delayed even further and will eventually approach  $90^\circ$  behind the input signal.

Table 1: Summary of Our Filter's Frequency Response

| Input(V) | R1( $\Omega$ ) | C1 (nf) |              |                |                |                  |                     |                     |                |              |
|----------|----------------|---------|--------------|----------------|----------------|------------------|---------------------|---------------------|----------------|--------------|
| 10.0     | 1000.0         | 15.0    | Freq (Hz)    | $X_c$          | Total Z        | Actual Output(V) | %Volt Out / Volt In | % Output Attenuated | Magnitude (dB) | Phase (deg)  |
|          |                |         | 1            | 10610329.54    | 10610329.59    | 10.00            | 100.0               | 0.0                 | 0.0            | 0.0          |
|          |                |         | 2            | 5305164.77     | 5305164.86     | 10.00            | 100.0               | 0.0                 | 0.0            | 0.0          |
|          |                |         | 5            | 2122065.91     | 2122066.14     | 10.00            | 100.0               | 0.0                 | 0.0            | 0.0          |
|          |                |         | 10           | 1061032.95     | 1061033.43     | 10.00            | 100.0               | 0.0                 | 0.0            | -0.1         |
|          |                |         | 20           | 530516.48      | 530517.42      | 10.00            | 100.0               | 0.0                 | 0.0            | -0.1         |
|          |                |         | 50           | 212206.59      | 212208.95      | 10.00            | 100.0               | 0.0                 | 0.0            | -0.3         |
|          |                |         | 100          | 106103.30      | 106108.01      | 10.00            | 100.0               | 0.0                 | 0.0            | -0.5         |
|          |                |         | 200          | 53051.65       | 53061.07       | 10.00            | 100.0               | 0.0                 | 0.0            | -1.1         |
|          |                |         | 500          | 21220.66       | 21244.21       | 9.99             | 99.9                | 0.1                 | 0.0            | -2.7         |
|          |                |         | 1000         | 10610.33       | 10657.35       | 9.96             | 99.6                | 0.4                 | 0.0            | -5.4         |
|          |                |         | 2000         | 5305.16        | 5398.59        | 9.83             | 98.3                | 1.7                 | -0.2           | -10.7        |
|          |                |         | 5000         | 2122.07        | 2345.88        | 9.05             | 90.5                | 9.5                 | -0.9           | -25.2        |
|          |                |         | 10000        | 1061.03        | 1458.01        | 7.28             | 72.8                | 27.2                | -2.8           | -43.3        |
|          |                |         | <b>10610</b> | <b>1000.03</b> | <b>1414.24</b> | <b>7.07</b>      | <b>70.7</b>         | <b>29.3</b>         | <b>-3.0</b>    | <b>-45.0</b> |
|          |                |         | 20000        | 530.52         | 1132.01        | 4.69             | 46.9                | 53.1                | -6.6           | -62.1        |
|          |                |         | 50000        | 212.21         | 1022.27        | 2.08             | 20.8                | 79.2                | -13.7          | -78.0        |
|          |                |         | 100000       | 106.10         | 1005.61        | 1.06             | 10.6                | 89.4                | -19.5          | -83.9        |
|          |                |         | 200000       | 53.05          | 1001.41        | 0.53             | 5.3                 | 94.7                | -25.5          | -87.0        |
|          |                |         | 500000       | 21.22          | 1000.23        | 0.21             | 2.1                 | 97.9                | -33.5          | -88.8        |

**Cutoff**

**Freq.=**

**10,610**

**-45.00**

Remember, in our RC filter, we only have one reactive component. Namely the capacitor. The resistor does not react to frequency changes. When the frequencies are low, the reactance of the capacitor is very high. The capacitor acts like a super high value resistor. No current goes through it. Our signal goes

through the resistor. But, when frequencies increase, the reactance (resistance) of the capacitor greatly diminishes.

This now provides an easy path to ground. So, the electrons go to ground and very little voltage is measured at the output of the filter. Also, the RC combination consumes some power. That is why the output voltage of a passive RC filter is always lower than the input. This type of filter does not amplify a signal.

If you want to amplify a signal you would use an active filter. Active filters are built from operational amplifier integrated circuits and would be used in audio and Ham radio applications where the signal will drive speakers or headphones. In our next section we will look at a filter that has two reactive components. Namely, two capacitors and two resistors. You will see how this affects our output curve.

## Second Order Low-Pass Filters

I want to call your attention to the Bode plot on the right. It shows the frequency response of our filter. Even after the corner/cut-off frequency, we can see that there will still be some low frequencies passed. In other words, the roll-off or slope of the line is pretty gradual.

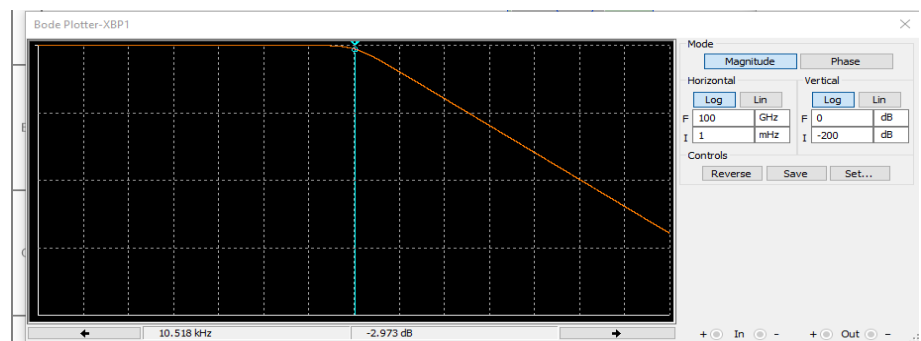


Figure 16: Bode Plot of our filter and the cut-off frequency.

The curve on the left side of the blue line shows the “pass-band” of the filter. These are the frequencies we want. The slope marks the area under the curve of the “stop band” or frequencies we don’t want. A gradual slope means a range of sounds we may not want will make it through our filter. This may or may not be a problem in your circuit but let’s look at a scenario where this slope and the response of our filter can be improved upon.

Instead of making a filter for a sensor, for example, let’s suppose you are attempting to filter high and low frequencies for your new loud speakers you purchased for your home surround sound system. A typical 2-way speaker consists of a woofer and tweeter. The woofer is engineered to reproduce low frequencies like bass and voice while tweeters are engineered to reproduce high frequencies such as that from a flute or guitar. If you connect your new speaker to your amplifier, you will get sound but it won’t be what you really want. What speaker manufacturers do is to build a filter network similar to what we have done, so that the high frequencies are split off from the lows and directed to the correct device. This is called a “cross-over” network. The high sounds and low sounds cross over at the cut-off frequency and are sent to the correct speaker. That way, the tweeter can handle the high notes and the woofer can handle the low notes. This makes the sound more appealing and realistic to our ears.

To help the speaker crossover filter work better, it would be nice if our low-pass filter would more accurately differentiate low and high tones. It would be nice if the slope you notice would discriminate between low and high sounds and really block higher frequencies more accurately. To change the characteristics of a filter, you can simply add another low-pass filter! This new circuit will be called a 2-pole, or second order filter. Let’s see how it works.

To make a 2-poll filter, we simply connect two single poll filters together (see Figure 17). This way, the slope of our line will become steeper and more of the sounds we are trying to block will be filtered out. You may recall that our single-poll filter roll-off results in a -20dB attenuation per decade of frequency. Well, with a 2-stage (second order) filter the roll-off will be -40dB/decade. This should allow our woofer to more accurately reproduce the low frequencies and higher frequencies will be blocked.

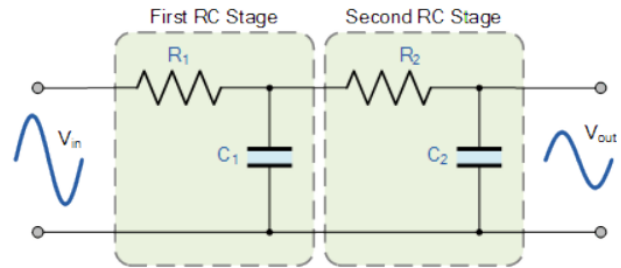


Figure 17: Second-Order Low Pass Filter

### Simulation of Our Second-Order Circuit in Multisim

Our original circuit has been revised by adding another resistor-capacitor combination (Figure 18). However, adding more capacitors affects our corner frequency. From the following formula,  $f_c$  = the new corner frequency:

$$f_c = \frac{1}{2\pi\sqrt{R1*C1*R2*C2}} \text{ Hz.}$$

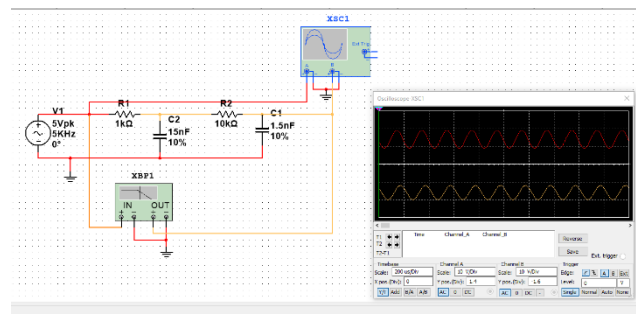


Figure 18: Second Order Low Pass Filter (Input =

Also, for our new circuit, I am going to choose  $R_2$  and  $C_2$  to be different values. The second stage component values will be a factor of 10 different in size. Thus,  $R_2 = 10 \times R_1$  and  $C_2 = C_1 \div 10$ .

This helps prevent the first filter stage from “loading down” the second stage. So, I will use  $R_2 = 10\text{K}$  and  $C_2 = 1.5\text{nf}$  as you can see in Figure 18. By choosing this new ratio of  $R_2$  and  $C_2$ , our RC time constant remains the same ( $R*C = .000015$ ). Which means the charge/discharge time of the capacitors is the same. However, the cutoff frequency changes.

The new calculated cut-off frequency at -3dB is:  $f_{(-3\text{dB})} = f_c \sqrt{2^{\frac{1}{n}} - 1} = 6,796.91\text{Hz}$  ( $n=2$ ).

Figure 19 compares the frequency response of our filters. The roll off is steeper in the two-stage version. Notice that the actual cutoff for the two-stage version is  $\sim 6,438\text{Hz}$ . This is different from the predicted value because the parts I had were 10% tolerance. If I had chosen 1% tolerance resistors and capacitors my calculated vs actual values would be closer to each other. If you work with high frequencies (radio, cellular, etc.) the values you choose become more critical than low frequency circuits.

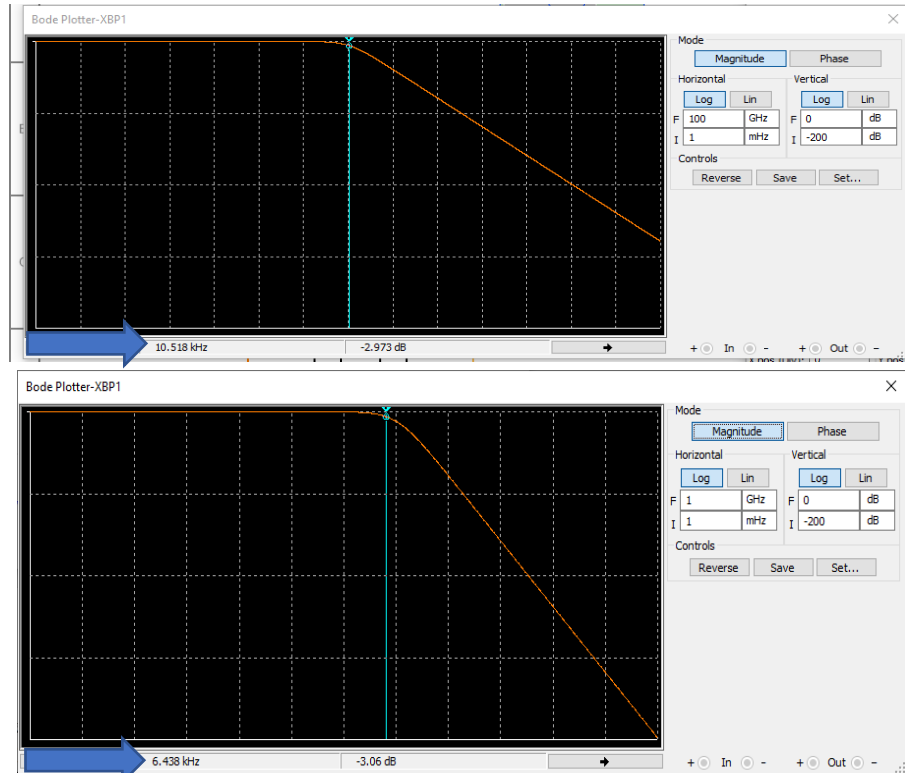


Figure 19: Comparison of Slope and Corner Frequency at -3db for First & Second Order Filters

## Chapter 2: Getting to Know High Pass R-C Filters

### Capacitive Reactance in a High Pass Filter

The circuit in Figure 2-1 is called a high-pass filter because it allows high frequencies to pass and blocks low frequency signals. This is just the opposite to the low-pass circuit we have been working with. As we have discovered, both the resistor and capacitor resist current flow. But only the capacitor reacts differently to changes in frequency. That is why we call its “resistance” *capacitive reactance* ( $X_c$ ).

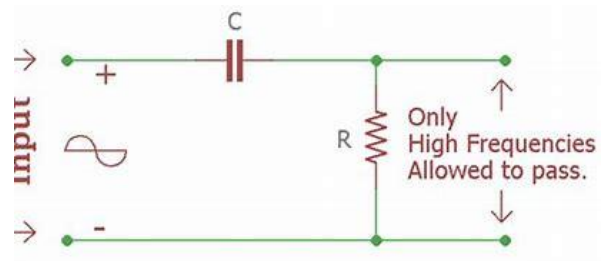


Figure 2-1: Simple R-C High-Pass Filter Circuit

If you have worked with voltage dividers, you can see how the resistor and cap form a voltage divider network. The difference is, *this* divider responds to changes in frequency. Thus, we have a *frequency-dependent voltage divider*. In our circuit, only the capacitor reacts to changes in frequency and so we have what is called a *single-pole* high-pass filter.

### Simulation of Our Circuit in Multisim

#### Our Setup:

- We will use our same components we have been using: the 1K resistor and 15nF (.015 $\mu$ F) ceramic capacitor. But this time the signal is connected in series with the capacitor, not the resistor.
- AC Input Signal: 10V P-P Sine wave (Red Trace on scope)
- Output of RC filter shown in yellow trace (Figure 2-2).

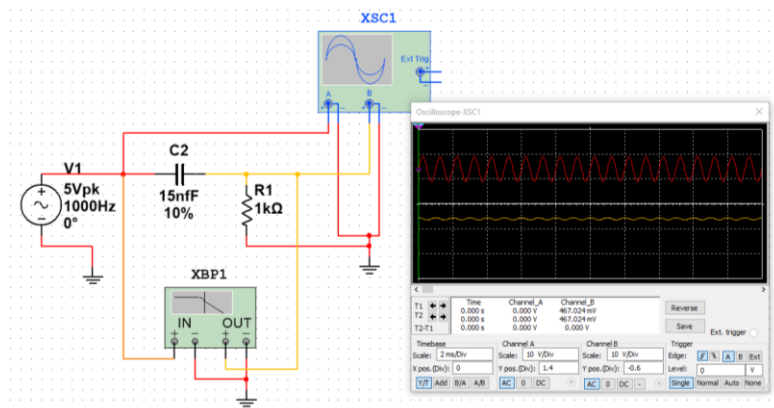


Figure 2-2: High Pass Filter in Multisim (5V Pk = 10V P-P)

Notice I have redesigned the circuit by swapping the positions of the resistor and capacitor. Now, the input signal is connected to the capacitor. Since the reactance of the capacitor at low frequencies is very high, it filters out those low signals. In Figure 2-2, the input signal is 1000Hz. Here the output is opposite to that of the low pass filter. Notice how the output signal (yellow) in Figure 2-2 is extremely small. This is because it is resisting the low frequencies in the signal. Again, just the reverse of the low pass filter design. Swapping the components is all that is needed to make a high or low pass filter.

Let's see what happens as we increase the input frequency.

Now the input signal is 30,000Hz. Figure 2-3 shows the yellow output is nearly the same voltage as the input. Again, just opposite of the low pass circuit.

The Bode plot in Figure 2-4 shows the characteristics of our filter. When the input signal approaches the cut-off frequency, only then are those higher tones allowed to pass.

However, notice we still have a gradual ramp. To change that, we can build a 2-stage high pass filter. We will see that soon.

At higher frequencies, capacitors cannot charge and discharge fast enough to keep up with the sine wave. Their reactance is low and they appear as a short circuit or a wire to the rest of the circuit.

Remember, the resistor-capacitor values not only determine the cutoff frequency, but they also determine how fast the capacitor charges and discharges. The cut-off or corner frequency occurs when the resistance of the resistor and the reactance of the capacitor are equal.

Lastly, the phase of our signal differs from the low-pass filter also. When an AC signal is applied to a capacitor, the current LEADS the voltage and is 45° out of phase at the cutoff frequency (Figure 2-5).

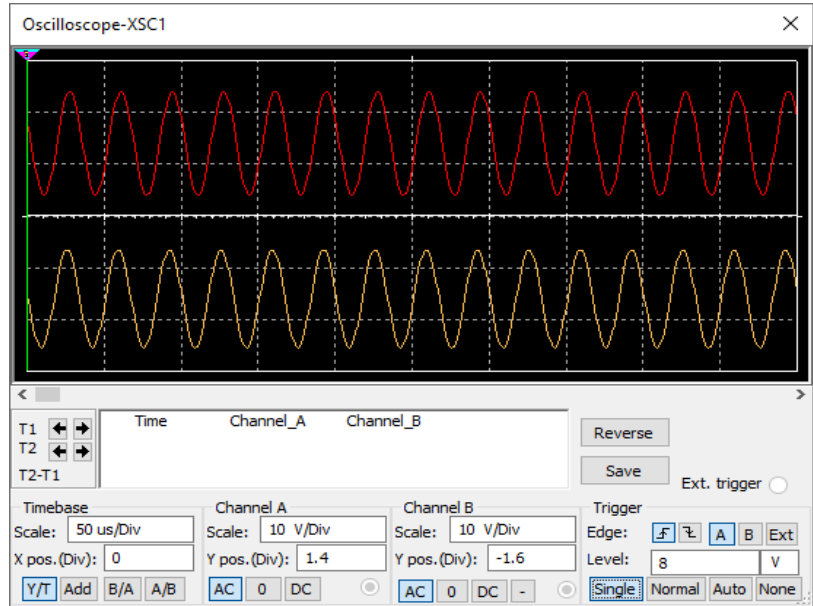


Figure 2-3: High Pass Filter with 30kHz Input Signal

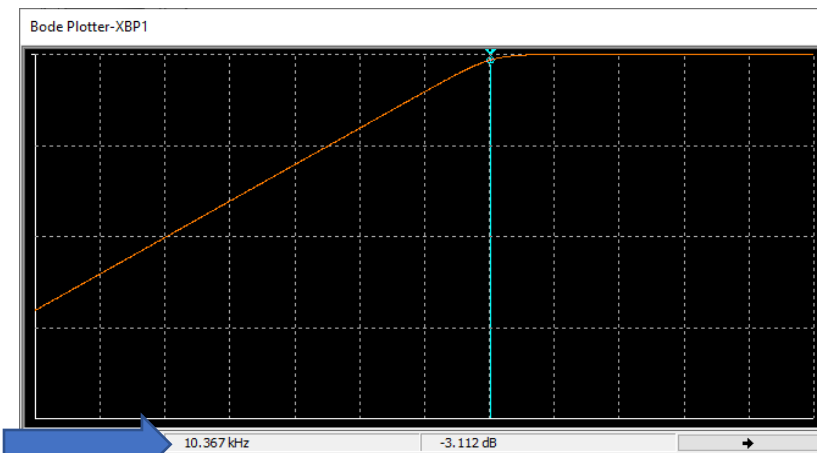


Figure 2-4: Bode Plot for High-Pass Filter. Cut-off Frequency at -3db is ~10,367Hz

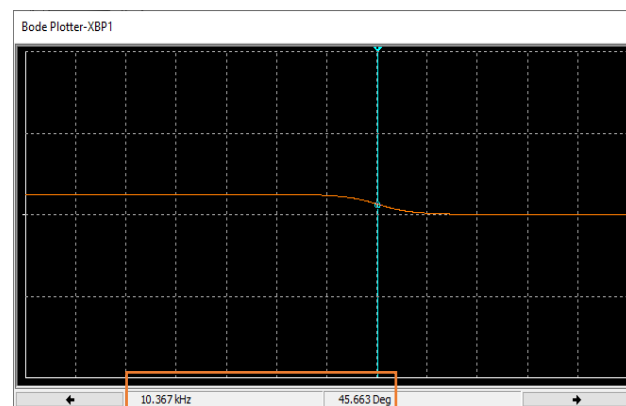


Figure 2-5: At the cut-off frequency, the output voltage LEADS the input voltage by +45 degrees.



## How to Measure Phase Shift with the Oscilloscope

To compute the difference in phase I used the scope cursors. Refer to Figure 2-6 below. The yellow line is our output at 10kHz which is very close to the cut-off frequency. The red line is the input signal.

1. Use the time/vertical cursor for channel 1 and move it over the spot where the yellow trace is over the bottom its sine wave. This is the blue vertical line in Figure 2-6.
2. Now move the cursor for channel 2 (Yellow vertical line) until it is located at the bottom of the sine wave for our input signal.
3. On the data table of the scope, it shows T2-T1 which is the time difference in the two points being measured. Notice it is 12.500 $\mu$ s.

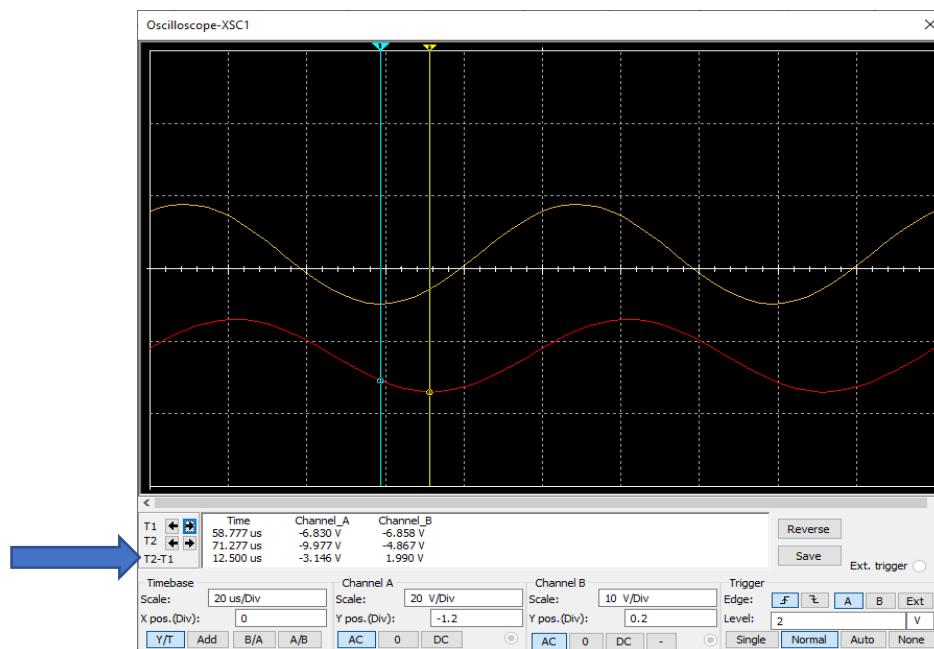


Figure 2-6: Measuring the Phase Shift of the Filter

4. Next, we need the *Period* of the wave. That just means how long does it take to go 1 complete cycle? In Figure 2-7, I moved the blue cursor to the start of a cycle where the yellow trace crosses the X axis and moved the yellow cursor to the point where it crosses again (360°). The T2-T1 value is 100µs this time. Now we can compute the phase shift angle in degrees.

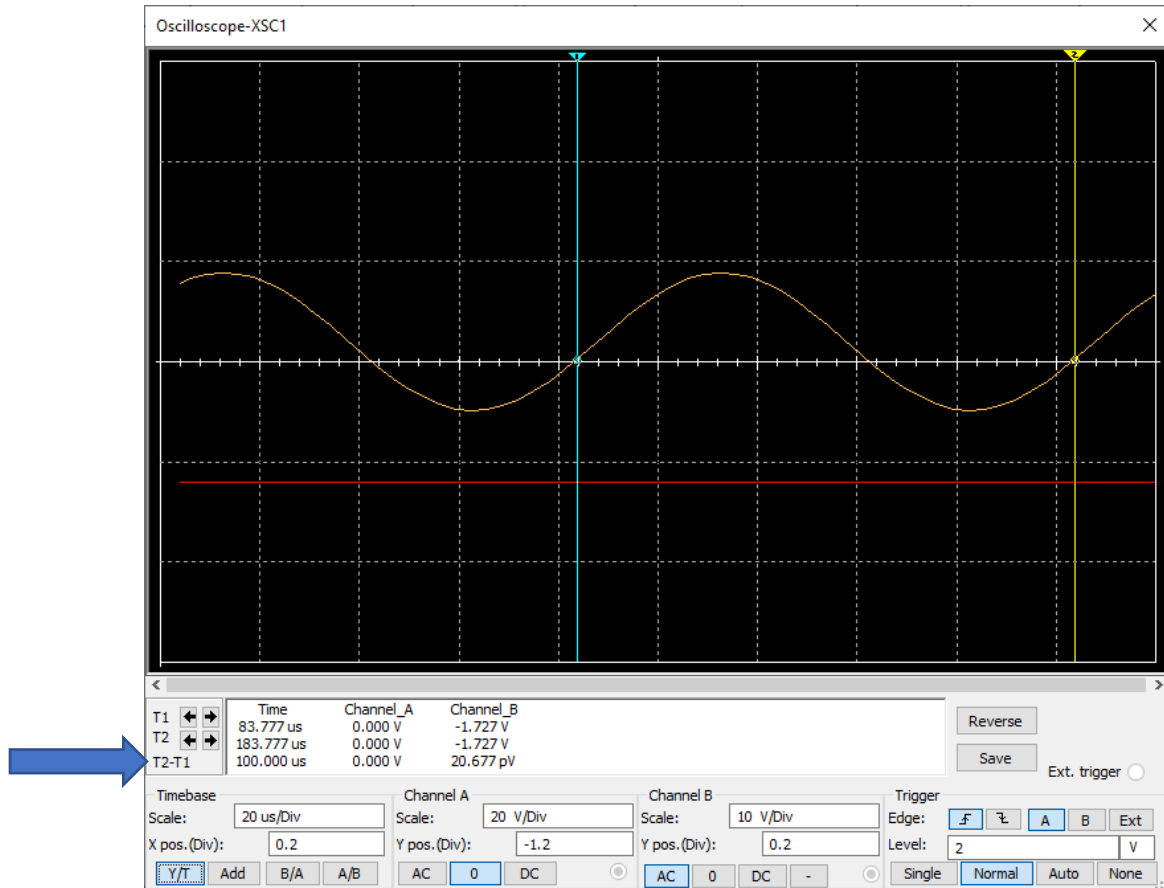


Figure 2-7: Measuring the period of the wave.

5. The formula is  $\Theta = \frac{\Delta T}{T} * 360 = \frac{12.5}{100} * 360 = \underline{45^\circ}$

## Second Order High-Pass Filters

To be consistent with our discussion of second order low-pass filters, let's end our discussion with a second order *high-pass filter* (Figure 2-8). You may recall that adding another stage or pole to a filter will dramatically steepen the curve for the band stop frequencies. Just as I added a second stage for the low-pass design, here is the second order high-pass design. I used exactly the same components but the corner frequency will change just as it did in the low-pass version.

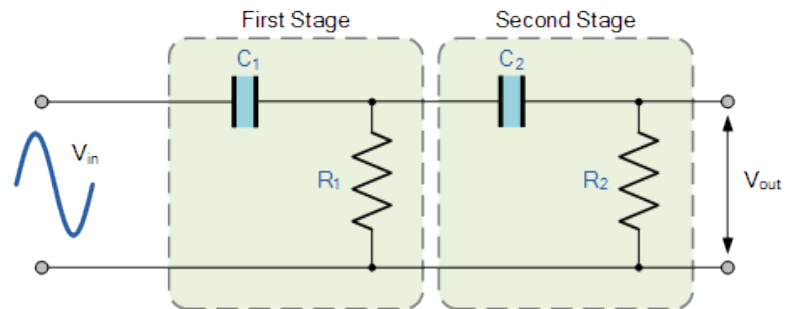


Figure 2-8: Two Stage High Pass Filter

Figure 2-9 shows the circuit design in Multisim and the accompanying wave forms. The red trace is the input and the output is yellow. I am using a 10v pk-to-pk signal at 10kHz.

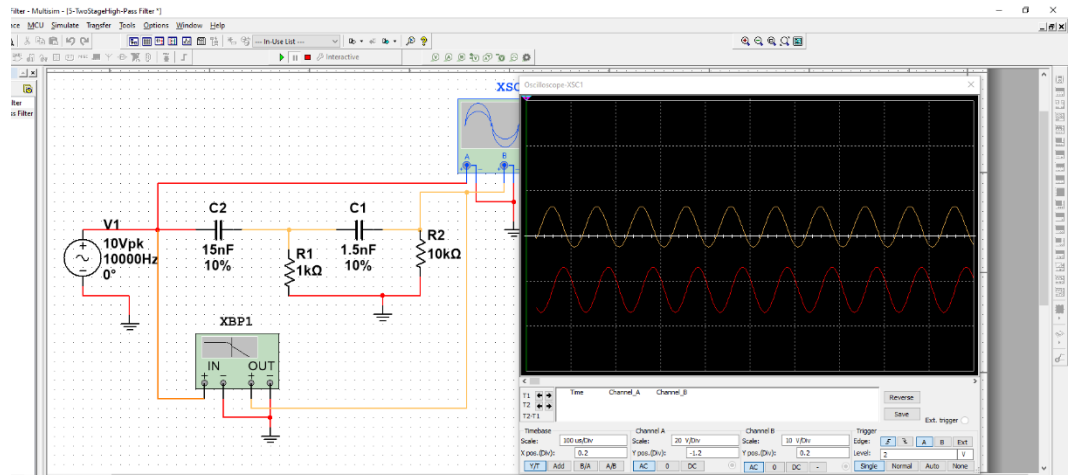


Figure 2-9: My Two-Stage High-Pass Circuit in Multisim

Notice how the Bode plot in Figure 2-9 changes for the second order filter. The slope is steeper and the corner frequency changed also. At -3dB, the cutoff frequency is now 17,433Hz instead of 10,367Hz.

I hope our discussion of passive filters has helped you understand how they work. In the next chapter, we will discover ways to filter unwanted signals from sensors before their signals are processed in software.

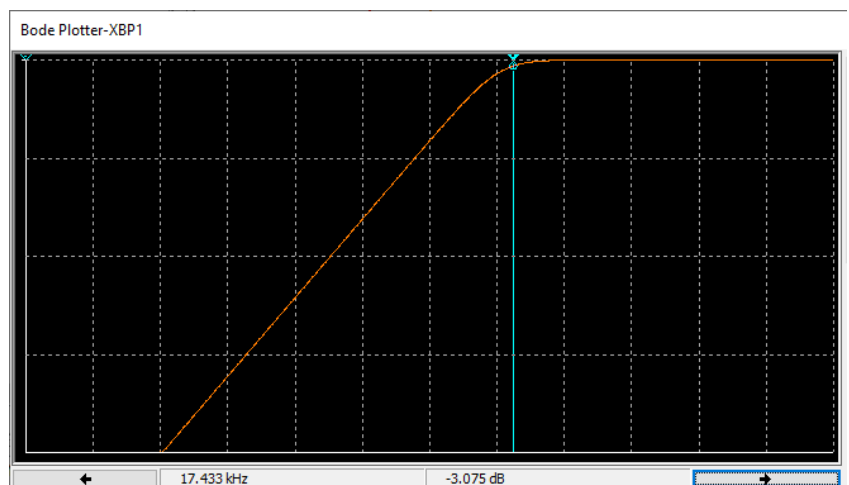


Figure 2-9: Bode plot for a second-order high-pass filter.

## Chapter 3: Working with Automotive Sensors

### Introduction

The sensor we will look at here is the knock sensor (Figure 3-1). This sensor is used to notify the Engine Control Module (ECM) that the engine is “pinging”. When an engine runs too lean, the fuel is burned prematurely (detonation) and the driver hears a ping or rattling/knocking noise while going up a hill, for instance. Left unchecked, the engine can overheat and eventually burn a hole in the piston.



Figure 3-1: Typical Knock Sensor

When a lean condition is triggered by the sensor, the ECU will add fuel to the engine or retard the ignition timing (or both) in order to mitigate this situation. Automotive engineers typically use a “knock sensor” made from a piezoelectric crystal device. These devices are used in industrial manufacturing, medical imaging, and are even used to detect flaws in aircraft engine components. They work in a rather simple manner. Let’s see how.

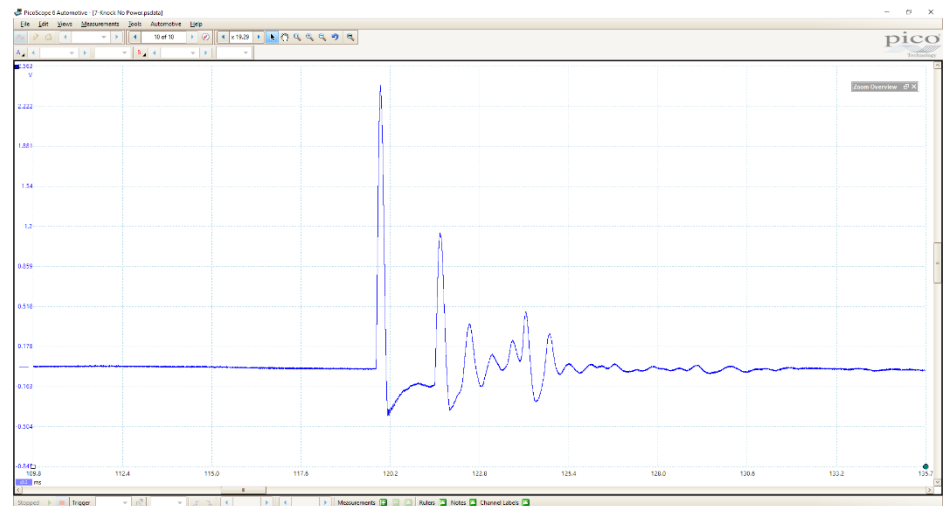


Figure 3-2: Knock Sensor Signal After Being Tapped on The Workbench

These sensors detect irregular engine vibrations caused by this pinging. As such, they are similar to accelerometers. Figure 3-2 shows the signal from my testing. Notice that there is one large pulse as the sensor is tapped against the workbench. You can see a lot of other signals as the vibration decays.

Special ceramic discs are used that respond to pressure or vibration. When detonation occurs, the vibrations are telescoped through the engine block and picked up by this sensor. The vibrations place pressure on the disc. The disc creates an AC voltage which it sends on to the ECU for interpretation. The output signal is proportional to the pressure or vibration the sensor detects.

They are called signal generators because they create their own signal without the need for a reference voltage. Manufacturers tune these sensors to be sensitive to the frequency that corresponds to a detonation/pinging condition in each of their engines. Thus, they are application specific. It turns out they are very accurate and reliable devices.

## Bench Testing a Knock Sensor

To test one of these sensors, I connected it to my Picoscope as shown in Figure 3-3. There are two connector pins. I connected my Picoscope probe to one pin and the probe's ground to the other. I then lightly tapped the sensor repeatedly on the bench to simulate pinging.



Figure 3-3: Bench Testing the Knock Sensor

Figure 3-4 shows the results of that test in the scope image. You can see the large spikes and the “ringing” between those spikes as it was being tapped. Remember, it is not necessary to have any power connections for this test.

These devices create their own current flow without the need for any external power.

Each spike measures the gentle force of the sensor striking the bench. From these images, it is easy to see why a microcontroller might have difficulty interpreting the results. Wouldn't it be nice if we could “clean” up the signal so the extraneous noise from those small ringing frequencies would be eliminated or at least minimized?

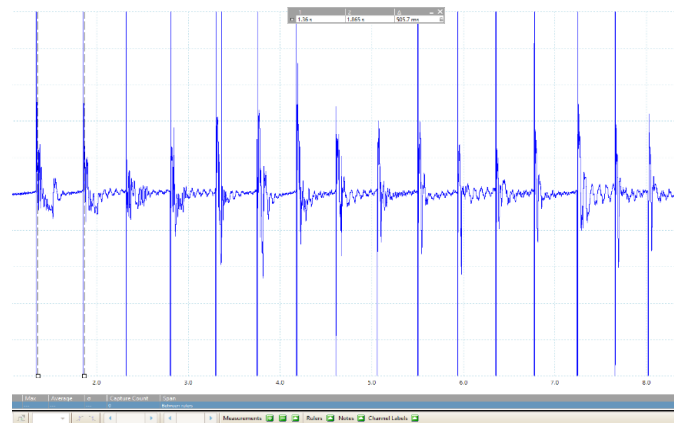


Figure 3-4: Knock Sensor Signal Created by Tapping on It Against the Workbench About 2 Times/Second

## Experiment 1: Process Signal Data with Single-Pole Hardware Filters

To demonstrate what I would like to do, I want to show you another zoomed-in image that shows those unwanted frequencies. Figure 3-5 shows the expanded scope image from my original image (Figure 3-4). I measured some of the noise pulses using the vertical cursors (dotted vertical lines) and discovered they ranged in frequency from, 35-2000Hz.

Let's see if we can build a filter to clean up this signal. Since the frequency of my signal was around 2Hz (caused by the speed of my tapping), I decided to filter out all frequencies above 15Hz to see what would happen.

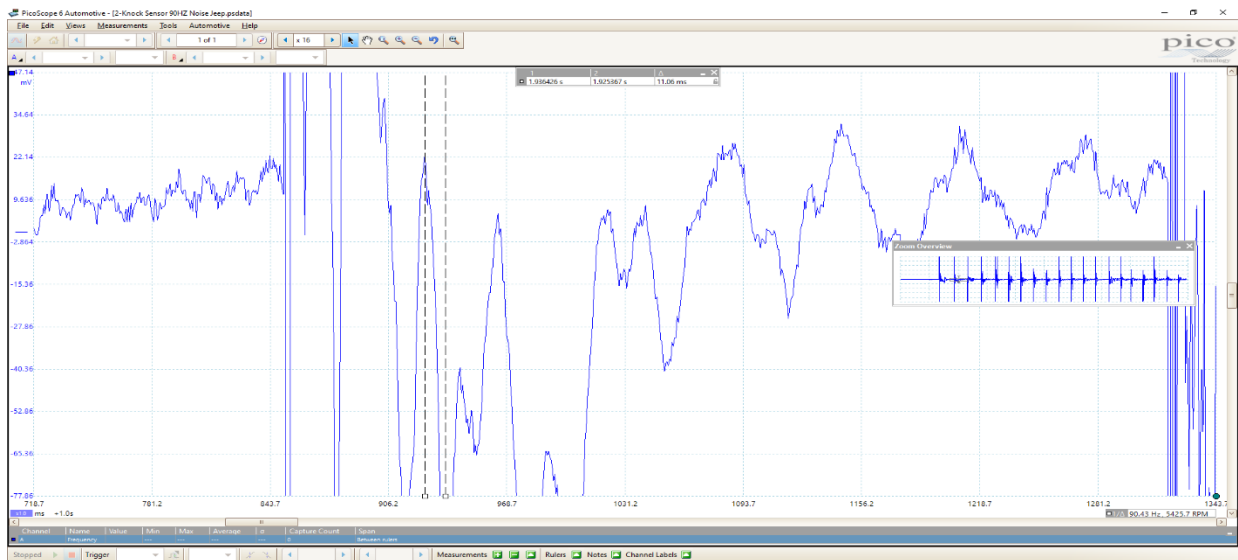


Figure 3-5: Expanded View of the Signal in Figure 4 That Shows 35-2000Hz Noise Embedded in the Desired Signal.

However, if you were designing a filter for a real application, you would have to determine the maximum frequency your sensor would produce and filter out all frequencies above that. For example, a wheel or gear rotation sensor needs to pick up the maximum RPM the object will rotate. Once that is determined, you can filter out all frequencies above that.

Now it is time to design our filter!



## Designing a Low Pass Filter in National Instruments Multisim

In order to make this signal easier for the MCU or controller to decipher, I decided to apply a single-pole, passive, low-pass RC filter using one resistor and one ceramic (non-polarized) capacitor. The single-pole refers to the fact we are using only one reactive component, the capacitor, that is sensitive to frequency changes. The resistor is not. Recall that filters with two capacitors are called two-pole because they use two reactive components, and so on. You can use several stages or poles in your design.

In my test scenario, the pulses I want to preserve occur at around 2Hz or so in this case. Recall, that just happened to be the speed at which I tapped the sensor. So, I decided that any frequency above 15Hz was just noise or unwanted pulses. Keep in mind, filters are not perfect. You cannot build one that passes 2Hz signals but absolutely blocks 3Hz signals and above. That is why I chose a frequency slightly larger than the signal I want to preserve.

My filter setup from the Multisim simulator is shown in Figure 3-6. The schematic shows how the input signal is connected to the resistor and we measure the output across the capacitor. This is what makes up a low pass filter.

I used a 10V Peak-to-Peak sine wave at a frequency of 2Hz to represent my sensor signal I observed while tapping it. To create a filter with the 15Hz cutoff, I chose a 1000Ω (1K) resistor and 10μF (microfarad) ceramic capacitor. Calculators are online for us to determine this cutoff frequency ( $f_c$ ).

The formula is:  $f_c = \frac{1}{2\pi RC}$

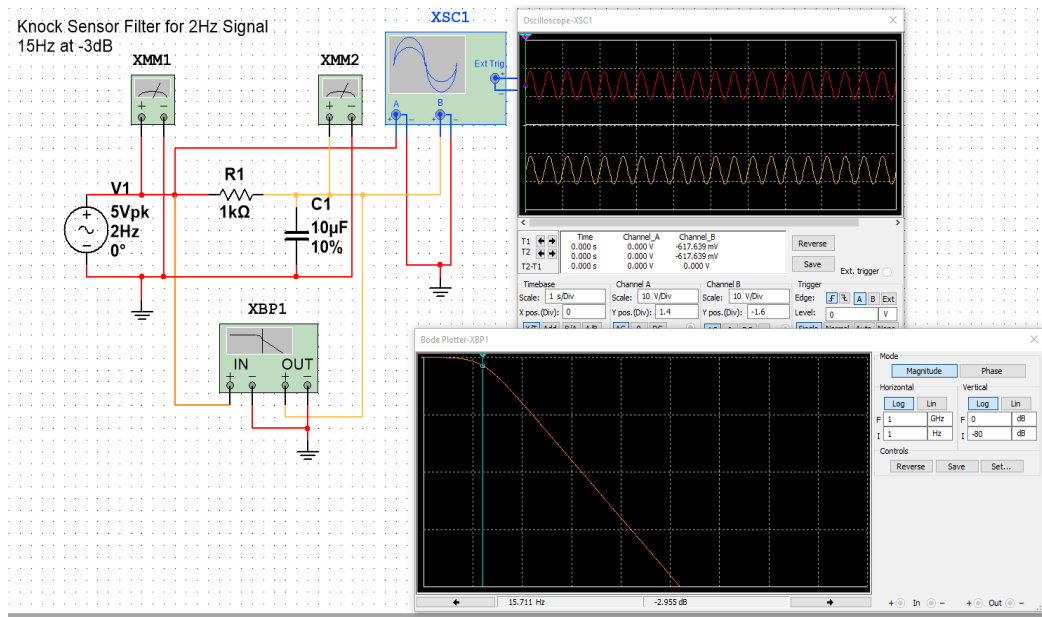


Figure 3-6: RC Low-Pass Filter Simulation. The Red Trace Shows Input Signal and the Yellow Shows the Output. At 2Hz, the Output Signal is Passed Through the Filter Easily. The Bode Plot Shows the Frequency Response and the Output Voltage of The Filter. The Chart Illustrates How Frequencies Above 15Hz Are Dramatically Reduced/Attenuated.

The components I chose should pass frequencies from 1-15Hz but will start blocking or attenuating signals above that.

If you look at the scope image, the red trace shows my 2Hz input signal. The yellow trace shows the output of the filter. Notice the amplitude of the two traces are nearly equal. This demonstrates how easily my low frequency signal (2Hz) passes through. In other words, the amplitude (voltage) of the output is almost identical to the amplitude of the input signal. The Bode plot in the second scope image shows how signals above my cutoff frequency are blocked. The graph shows the output voltage dropping quickly after cutoff (the green vertical line). This means those high frequency signals I identified in the 30-2000Hz range will be reduced in strength and mostly eliminated. This should make our sensor signal much easier to process with an analog to digital converter (ADC), for example.

### Applying the Low Pass Filter

Now it is time to compare the original signal with the filtered or unwanted noise image shown in Figure 3-7. The signals are significantly cleaner. With the filter in place, the sensor pulses should be more easily detected by my microcontroller! Remember, this filter allows frequencies *below* 15Hz to pass through freely while frequencies above that are attenuated.



Figure 3-7: Knock Sensor Trace With 15Hz Low Pass Filter Installed

## Experiment 2: Using a Two-Pole Filter

Before we finish up our discussion of filters, I wanted to use two filter stages so as to improve filter functioning. The design shown in Figure 3-8 is a two-stage or two-pole low-pass filter. It is made up of two single stage filters. In this circuit, I now use two resistors and two capacitors. This design causes the steeper roll off in the graph and removes more unwanted frequencies just after cutoff. This steeper roll-off attenuates signals at 40dB/decade compared to 20dB/Decade.

With more components, our formulas change slightly. For example, to compute the cutoff frequency for 2-pole filter:

$$f_c = \frac{1}{2\pi\sqrt{R_1C_1R_2C_2}} \text{ Hz}$$

When two poles are used, the decrease in signal strength at cutoff is twice that of the single-pole design which is -3dB. So, at the calculated cutoff of 15.91Hz, the two-pole design will reduce the signal by -6dB. Hence, the graph is twice as steep. The term *decade* refers to the fact that the x-axis on my graphs are in logarithmic scale. The frequency is plotted as 0, 10, 100, 1000, 10,000, 100,000 Hz.

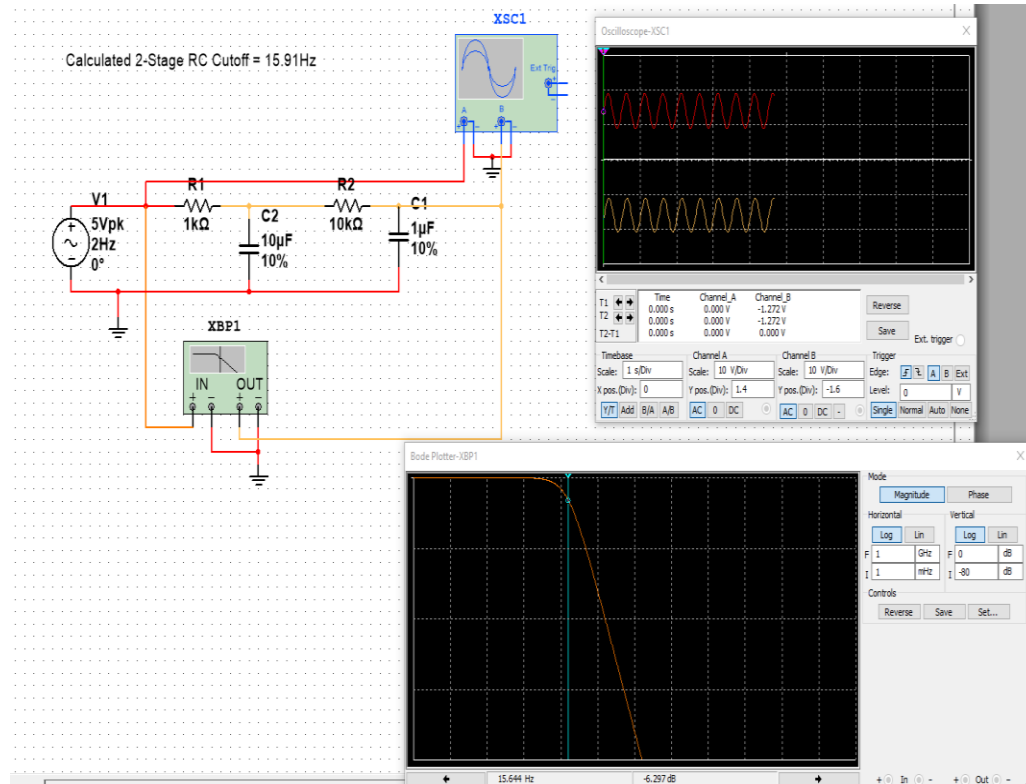


Figure 3-8: Two-Pole (two-stage) RC filter design. Notice the roll off of the filter is twice as steep as my original design.

## Simulation vs Actual Filter Responses

### Single-Pole Design

For clarity, let's compare the frequency response for both filters side by side. You will discover that the simulator results and your actual readings will be different because the quality of our physical components are not perfect.

We calculated the cutoff formula for the 1-Pole filter as 15.9Hz. However, Figure 3-9 shows the actual reading is slightly lower.

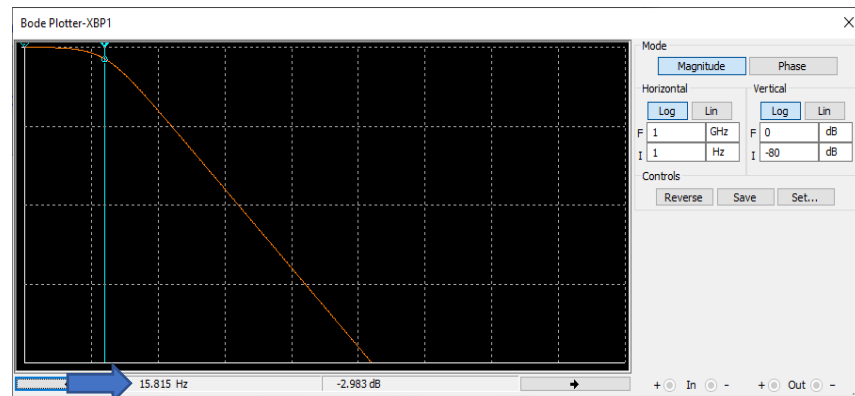


Figure 3-9: Bode Plot for Single-Pole Design

### Two-Pole Design

With the use of a 2-pole filter, the discrepancy is more pronounced. This is due to the effects of the added resistor and capacitor. It actually lowers the cutoff frequency.

We calculate the 2-pole cutoff value with a slightly modified formula. At -3dB, the cutoff ( $f_{-3dB}$ ) is calculated as:

$$\begin{aligned} f_{-3dB} &= f_c \sqrt{2\left(\frac{1}{n}\right) - 1} \\ &= 15.91(\sqrt{1.41 - 1}) \\ &= 15.91(.64) \\ &= \underline{\underline{10.18\text{Hz}}} \end{aligned}$$

where  $n$  = the number of filter stages.

Figure 3-10 shows the *actual* cutoff frequency is now 9.7Hz.

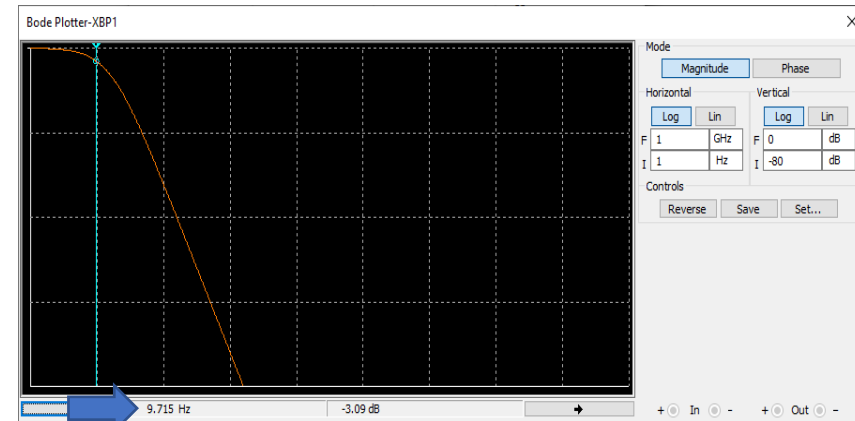


Figure 3-10: Bode Plot for Two-Pole Filter. The Roll Off is Twice as Steep as the Single-Pole Design (40dB vs 20dB/Decade)

Finally, it is important to remember that the frequency response of these filters will vary with the resistance of the load or the circuit you are connecting them to. Most textbooks use a 50Ω load when they show simulations. So be sure and test your circuit with that in mind.

---

## Summary and Conclusions and Going Further

**A**nalog signal processing is a difficult but necessary element of telemetry systems in many industries. We have explored the use of hardware filtering techniques that can help improve the accuracy and reliability in automotive sensor signals. While this paper only focused on knock sensors, the results can be generalized to many sensor circuits including Hall Effect Crank/Camshaft sensors, IR, accelerometer, and RPM monitoring systems.

The use of passive RC filters appears to be a viable solution to the problem of how to eliminate EMI in low voltage sensor circuits. It is a good idea to remove unwanted noise at the source instead of using software algorithms such as moving average and exponential moving average. While the software solutions are quite effective in many scenarios, hardware filters should be strongly considered when gathering sensor readings in noisy industrial and automotive environments. We have seen how both single-pole and two-pole (two-stage) filters can be quite effective at rapidly attenuating unwanted frequencies in the “stop band” of RC filters.

Keep in mind, the more stages/poles that are added to the filter, the more the output signal is reduced. When the output needs to be amplified so as to accommodate analog-to-digital converters (ADC), active filters using op-amps should be used so the gain can be calibrated to match the needs of the microcontroller.

Cheers!