

ELSOC presents

LTspice Workshop

Joe Li

Introduction

LTspice is a useful simulation software. When circuit designing, it is a good practice to simulate the circuit before buying components to build it. It saves time and money and often prevents you from destroying components. Changing a variable in simulation takes less time than doing so in a circuit. Simulation is also a great tool to help you verify your circuit design or analysis.

In this workshop, we will simulate various circuits and learn how to use basic commands and shortcuts in our way. There are many sections, however it is not expected for everyone to finish all the tasks within the timeframe of the workshop. As the task stays further below, less instructions are given to the task.

Section 1 – Filters

A big part of EE is transferring real world signals (e.g. sound, images, speed) into electrical signals (either digital or analog), manipulating signals with circuits to achieve what we desire, and transferring them back into output (e.g. speaker, monitor, motor).

A basic way of manipulating signals is using filters. Filters are used in various ways to achieve different purposes, which we will discuss in the following sections.

Low Pass Filter

Low pass filters transfer only low frequency signals and filter out high frequency. Applications of low pass filters include but are not limited to bass boost in music and noise reduction. We will be constructing a first order low pass filter.

Procedure

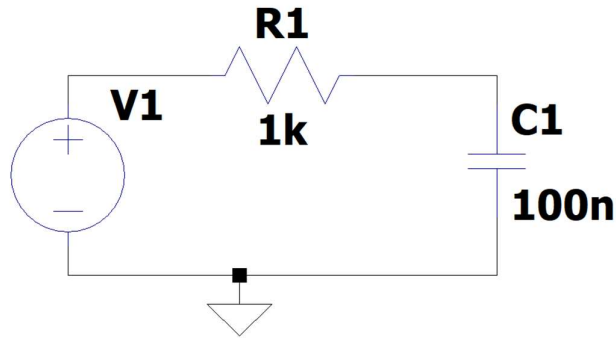


Figure 1

1. Drag the resistor from the drop-down menu bar onto the schematic. Press Ctrl-R to rotate it before placing it down.
2. Drag the capacitor from the drop-down menu and place it onto the schematic.
3. Select the voltage source by clicking on the component button (AND gate) from the drop-down menu, drag it onto the schematic.
4. Click on wire on the drop-down menu and connect the voltage source, resistor and capacitor in series.
5. Drag ground from the drop-down menu and place it at the bottom node.
6. Click on the voltage source and select advanced. In the Function section, select SINE. Set amplitude as 5V and frequency as 100Hz. Leave everything else as default. Then click on OK.
7. By right clicking on the components, change the resistor value to 1k and capacitor value to 100n.
8. Right click at the node between capacitor and resistor and select 'Label Net', name the node "Vout". Do the same for the node between resistor and voltage source but name it "Vin".
9. Click on spice directive on the drop-down menu. Right click on textbox and select 'Help me edit' -> 'Analysis Cmd'. Under transient section, put stop time as 20ms.
10. Run the simulation by clicking on the run button on the drop-down menu.
11. Click on node Vout and node Vin. Compare the two signals in the nodes.

The Signal should be similar. Hence showing at low frequency such as 100Hz, signal is not suppressed.

12. Repeat steps 10 and 11 with the frequency of the voltage source being 1kHz, 10kHz and 100kHz and transient stop time being 2ms, 200us and 20us respectively. Observe the changes to the output magnitude.

Observing the pattern in such a way is very time consuming and inefficient. Hence, we should use **AC analysis** to do such tasks to observe how magnitude and phase changes with respect to frequency. We call this "AC sweep" as we "sweep" across the frequency.

13. Right click on the voltage source, under small signal AC analysis (AC), set AC amplitude to 5.
14. Click on the simulation command and click into 'AC analysis' section.
 - Select Decade as type as sweep
 - Keep points per decade as 10
 - Start frequency as 10
 - Stop frequency as 10Meg, which stands for 10 million hertz
15. Run the simulation by clicking on the run button on the drop-down menu. Click on the output node upon running the simulation. You should see a bode plot.
16. Right click on the y-axis and select Linear, click OK.

You should now see the behavior of output magnitude when voltage source is at different frequencies. Where the cut-off frequency should be at approximately 2.5kHz. At any frequency higher than this frequency, signal is suppressed. Hence showing the characteristics of a low pass filter.

High Pass Filter

High Pass Filters out low frequencies and preserves high frequency signals. Application includes but is not limited to removing low frequency noise (such as AC mains) and Amplifying Trebles in Audio equalizers. We will construct a high pass filter with keyboard shortcuts.

Procedure

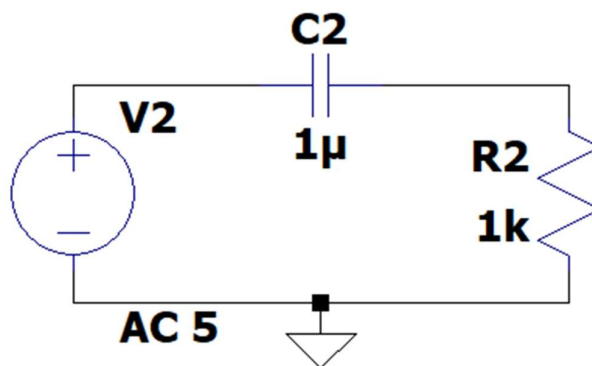


Figure 2

1. Create a new file by pressing Ctrl-N
2. Press C to place down capacitor. Press Ctrl-R to rotate resistor before placing it down.
3. Press R to place down a resistor
4. Press V to place down a voltage supply.
5. Press F3 to select Wire. Then connect voltage supply, capacitor and resistor in series.
6. Press G and place GND symbol on bottom node.
7. Right click on the voltage source, under small signal AC analysis (AC), set AC amplitude to 5.

8. By right clicking on the components, change the resistor value to 1k and capacitor value to 1u.
9. Right click at the node between capacitor and resistor and select 'Label Net', name the node "Vout". Do the same for the node between capacitor and voltage source but name it "Vin".
10. Add a simulation command by pressing S, then right click on textbox and select 'Help me edit' -> 'Analysis Cmd'.
11. Click on the simulation command and click into 'AC analysis' section.
 - Select Decade as type as sweep
 - Keep points per decade as 10
 - Start frequency as 10
 - Stop frequency as 10Meg, which stands for 10 million hertz
12. Run the simulation by clicking on the run button on the drop-down menu. Click on the output node upon running the simulation. You should see a bode plot.
13. Right click on the y-axis and select Linear, click OK.

Observe the magnitude of the output as frequency increases. The cut-off frequency should be approximately 100Hz, any signals with frequency below 100Hz should be suppressed. Hence showing characteristics of a high pass filter.

Band Pass Filter

Band Pass Preserves a band of frequency (e.g. from 100Hz to 2.5kHz) and filters out the rest. Application includes but is not limited to obtaining a radio channel and audio equalizers. A band pass filter can be obtained by cascading a low pass and a high pass transfer function. We will proceed to construct a band pass filter using our previous filter design.

Procedure

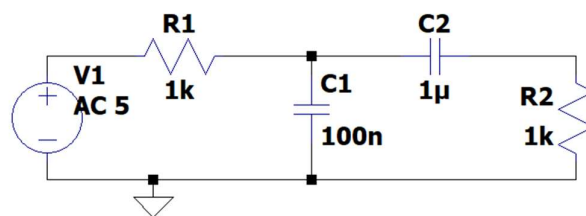


Figure 3

1. Create a new file by pressing Ctrl-N
2. Go back to your low pass filter file. Press F6 and drag across the circuit to copy. Do not place your circuit down.
3. Click the tab of your current file under the drop-down menu and place your circuit on the schematic.
4. Go back to your high pass filter file. Press F6 and drag across the circuit to copy. Do not place your circuit down.

5. Click the tab of your current file under the drop-down menu and place your circuit on the schematic towards the right of your low pass filter.
6. Press F5 and click the second power supply to remove it.
7. Press F3 and cascade the second circuit onto the first one.
8. Press F4 to relabel the nodes as you desire. Vout should be the node towards top right.
9. Add simulation command, similar to steps on previous exercises.
10. Run the simulation by clicking on the run button on the drop-down menu. Click on the output node upon running the simulation. You should see a bode plot.
11. Right click on the y-axis and select Linear, click OK.

Observe the magnitude of the output signal as frequency changes, note the signal is preserved when the frequency is between 100Hz and 2.5kHz and suppressed outside of the band. Hence circuit shows characteristics of a band pass filter.

Section 2 – Digital and Analog

Digital and Analog Signal both have their own advantages and disadvantages. It is important to know how to convert digital signal into analog signal and vice versa. In this section, we will build an analog to digital converter and a digital to analog converter.

Comparators

A comparator compares the signal between positive and negative terminal. If the positive terminal has a larger voltage, output will be HIGH (~5V), otherwise, it will be LOW (~0V). We will simulate how the LT1721 Comparator behaves as we change the input voltage.

Procedure

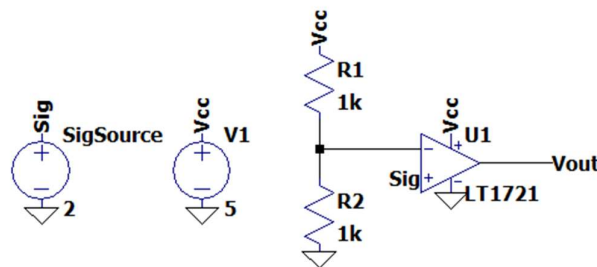


Figure 4

1. Place down a comparator by pressing F2 and searching for LT1721.
2. Place two resistors on top of each other towards the left of the comparator. Give them a value of 1k.
3. Press V and place two voltage supply anywhere on the schematic.
4. Press G and place GND on both voltage supply and the bottom resistor.
5. Press F4 and label the voltage source positive terminal node. Name one of them "Vcc" and the other one "Sig" (standing for signal).

6. For the voltage source with node labelled Vcc, set it's DC value to 5.
7. Label the top node of the top resistor and the positive terminal of the comparator "Vcc". This directly connects the Vcc voltage supply into the node with the "Vcc" label. Note all node with the same label are considered as the same node. This is great for keeping the schematic clean.
8. Label the positive input of the comparator to be "Sig".
9. Connect the two resistor and join them to the negative input of the comparator.
10. Label the output of the comparator "Vout".
11. Add simulation command, set it as transient with stop time as 10ms
12. Set the voltage supply with output node "Sig" to 2V and run the simulation

Note the output is now close to 0V, as 2V is less than the compared voltage of 2.5V from the voltage divider. Due to comparator limitations, it cannot achieve 0V.

13. Set the voltage supply with output node "Sig" to 4V and run the simulation

Now "Sig" is higher than the compared voltage 2.5V, hence output is close to 5V. Again, due to comparator limitations, it cannot achieve 5V.

Evaluating output from every voltage, again, is very annoying. Therefore, we should simulate this using a DC sweep, which measures output for every voltage.

14. By right clicking on the name of the voltage source, Name the voltage source with the node "Sig" as "SigSource".
15. Click on the simulation command. Navigate to DC sweep.
 - Set the first source to SigSource.
 - Type of sweep to linear
 - Start value as 0. Stop value as 5.
16. Run the simulation command, click on the nodes Vout and Sig

Observe the output voltage steps up once Sig reaches 2.5V

Analog to Digital Convertor (ADC)

Analog to Digital Convertor is used to convert an analog voltage into a value in digital in binary. This is important to analyze data in the digital domain, such as the use of microcontrollers and FPGAs. Digital data holds an advantage in terms of accuracy and less affected by other factors such as noise. We will use LTspice to simulate a 2-bit ADC.

Procedure

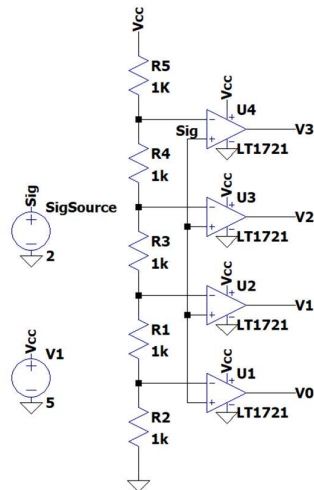


Figure 5

1. Building on to the comparator schematic, remove the resistor, "Vcc" label on top of it and "Sig" in the positive terminal of the comparator by pressing F5 and clicking onto it.
2. Press F6 and copy the entire comparator circuit, excluding the GND below the resistor and both voltage supply. You can drag the GND downwards by pressing F8.
3. Repeat this such that you have 4 comparators in the circuit.
4. Add an extra resistor on top and label the top node "Vcc" by pressing F4.
5. Join the negative input of all comparators and label the node Sig.
6. Label the four outputs V0, V1, V2, V3.

Now if we wish to plot multiple graphs to represent V0, V1, V2 and V3 individually.

7. Run the simulation
8. Probe V0 to see output of V0 on the graph.
9. Right click on the plot and select 'Add Plot Plane'.
10. Probe V1 to see output V1 on a different graph.
11. Repeat for V2 and V3.

This signal needs to be encoded for it to be 2-bit digital signal. However, to have a 2-bit encoder, we need to download it online and import it into LTspice library folders. For the time being, we will create a 4-2 priority encoder from scratch.

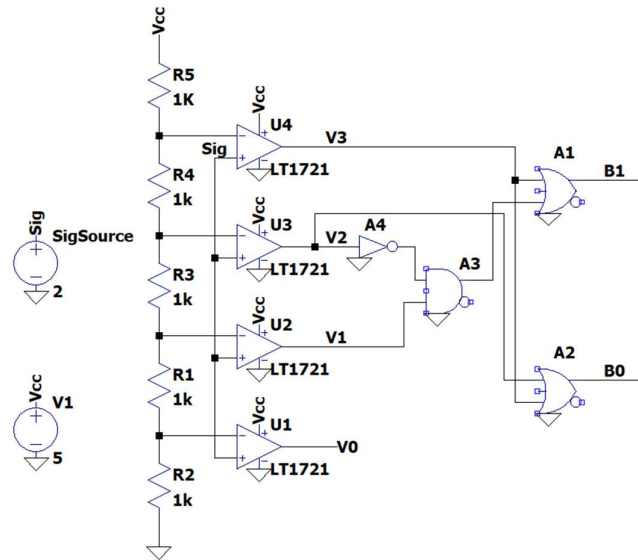


Figure 6

12. Press F2 and search for 'or', place down two OR gates.
13. Press F2 and search for 'and', place down one AND gate.
14. Press F2 and search for 'inv', place down one NOT gate.
15. GND all the logic gates and wire them as shown in *Figure 6*.
16. Run the simulation and plot B0, B1 on separate graph

You should be able to observe binary values as voltage goes higher.

Digital to Analog Convertor (DAC)

Digital to analog converter is important when we want to obtain an analog signal from a digital one. It is important as some systems operate in analog or some output has to be an analog output, such as speakers. We will proceed to convert the digital signal from the decoded ADC from previous exercise into a digital signal using a DAC ladder.

Procedure

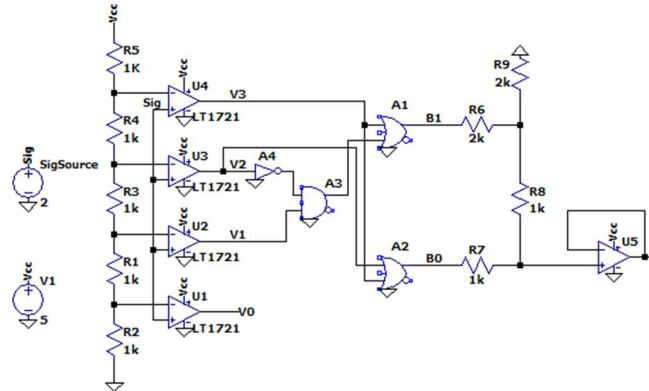


Figure 7

1. Building on to the ADC, we place a resistor horizontally in front of B0 and B1, give them a value of 2k.
2. Place a resistor vertically towards the right between the two 2k resistor, give it a value of 1k.
3. Place a resistor vertically towards the right above all the previous resistors placed. Give it a value of 2k. Place GND at the above node of the resistor.
4. Place an OP amp (you can choose Universal OP amp) by pressing F2 and searching for opamp2.
5. Label the positive supply Vcc and negative supply GND.
6. Wire the schematic as shown in diagram, name the output "Vout"
7. Run the simulation and probe the output

We should observe a poorly quantized analog signal attempting to resemble the input. Note as the number of bits increase, the closer the ADC resembles the original signal. 2-bits are very minimal.

Section 3 – NE555 timer

The NE555 timer is a very inexpensive chip, but also very useful. They can be used in Astable mode, Monostable mode, and Bistable mode. We will attempt to simulate Monostable and Astable mode in an NE555 timer chip.

Monostable mode

When a negative pulse is sent through, the NE555 timer outputs another pulse for a set period of time.

Procedure

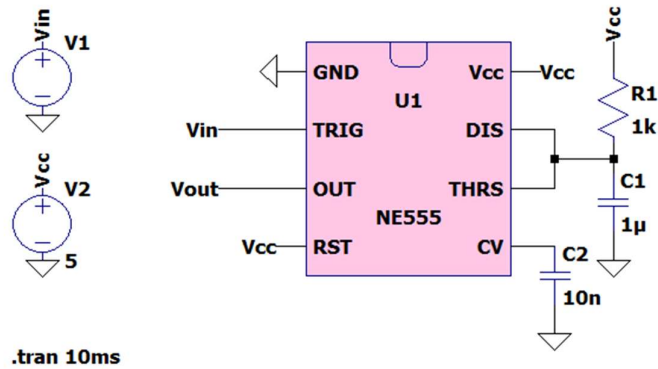


Figure 8

1. Press F2 and search for NE555, place it on the schematic.
2. Set up the circuit as shown in Figure 8.
3. Right click on voltage source that produces Vin and click advanced.
4. Select Pulse, and type in the following
 - Vinitial : 5V
 - Von : 0V
 - Tdelay : 3ms
 - Ton : 200us
 - Tcycles : 1
5. Run the simulation and observe Vout and Vin.

Astable mode

Astable mode produces a cheap square wave

Procedure

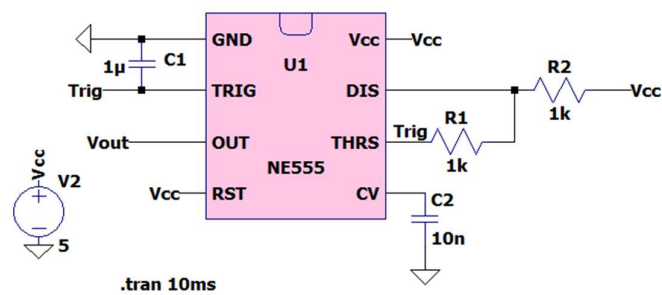

























Figure 9

1. Set up the circuit as shown in Figure 9.
2. Run the simulation and observe Vout.

End

List Of Keyboard Shortcuts

Symbol	Name	Windows / Linux	MacOS
	New Schematic	Ctrl N	Cmd N
	Open file	Ctrl O	Cmd O
	Save	Ctrl S	Cmd S
	Rotate	Ctrl R	Cmd R / Ctrl R
	Mirror	Ctrl E	Cmd E / Ctrl E
	Zoom to Rectangle	Ctrl Z	Z
	Zoom Back	Ctrl B	track pad / scroll wheel
	Zoom to Fit	Space bar	Space bar
	Components	F2	fn-F2
	Wire	F3	fn-F3
	Label Net	F4	fn-F4
	Cut / Delete	F5 / Delete	fn-F5
	Copy	F6	fn-F6
	Move	F7	fn-F7
	Drag	F8	fn-F8
	Undo	F9	Ctrl Z / Cmd Z / fn-F9
	Redo	Shift-F9	Shift + Ctrl Z / Cmd Z / fn-F9
	GND	G	G
	Resistor	R	R
	Capacitor	C	C
	Inductor	L	L
	Text	T	T
	Spice Directive	S	S