ECGR 2254 – Project 1

Due: Friday 10/30/2020 – 6PM

Please note the following:

- **Provide a neatly written report that describes your process. For example, , carefully describe how you chose your time step, mixing words and equations as needed. You don't need to write a lot, but someone who knows what you're doing (i.e. me) should be able to follow your logic.**
- **You do not need to type up a report. A neat, hand-written report is sufficient.**
- **Please identify your collaborators (i.e. two or three people with whom you worked on the project). Each person should submit a unique report, and you do not have to collaborate. That said, I think it's highly likely that you will talk to others, and you should cite that. Working with others is a good way to learn, as long as you re not copying.**

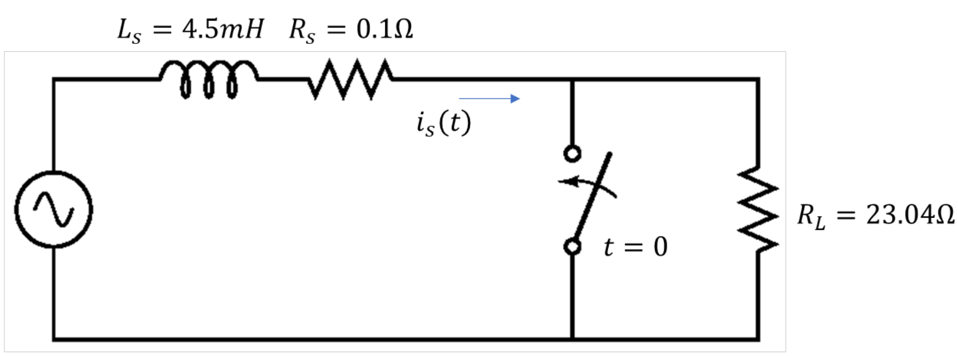
Problem 1 – AC Circuit Simulations

In power-system analysis, it is common to use simulation software to understand the transient conditions that can easily damage equipment. This problem examines two important issues that frequently occur in practice.

Part a

The circuit shown below models the behavior of an AC power system providing power to a single phase of a building absorbing approximately 10kW of power. The building and its loads are modeled as a 23.04Ω resistor. At time $t = 0$, a tree falls on the line coming into the building and causes a short circuit to ground. This situation can be modeled as a switch closing at $t = 0$ as shown below.

The AC system can be modeled as a Thevenin equivalent circuit consisting of voltage source $v_s(t)$ in series with an inductor L_s and a resistor R_s. Protection engineers use simulation programs to select the circuit breaker that must be placed in series with the voltage source to interrupt the extremely dangerous short-circuit current.



$v_s(t) = 480\sqrt{2}\cos(2\pi 60t - \phi)$

Note that ϕ is a variable.

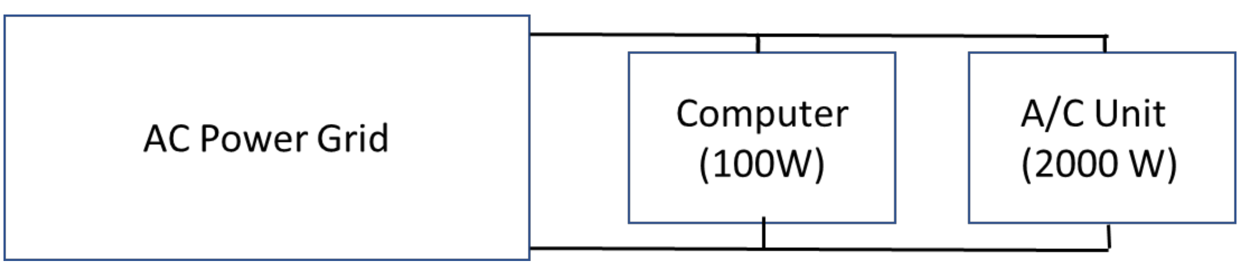
a) Protection engineers often use what we call "short circuit current" to select a circuit breaker. This value is the amplitude of the steady-state current flowing once the switch closes. Use impedance analysis to determine the amplitude of this current. ***I stress that you should use phasor/impedance analysis. Do not waste time solving the complete differential equation.***
b) Write a differential equation that allows you to solve for $i_S(t)$ after $t = 0$.
c) Think carefully about the steady-state and transient solution to your differential equation to determine an appropriate time step that will allow you to see the complete transient and several periods of the current. Explain your choice.
d) Write a MATLAB script that solves for the current $i_S(t)$ after $t = 0$. To approach this problem, assume that the $i_S(0) = 0$. Try several different values of ϕ between 0 and 180 degrees. For what approximate value of ϕ does the value of $i_S(t)$ become the largest after $t = 0$?
e) Using your value of ϕ from part d, use impedance/phasor analysis to determine a function for the current $i_S(t)$ right before $t = 0$. Use this expression to determine the true numeric value for $i_S(0)$. ***I stress that you should use phasor/impedance analysis. Do not waste time solving the complete differential equation.***
f) Rerun your script from part d using the true initial condition for $i_S(0)$. What is the peak current?

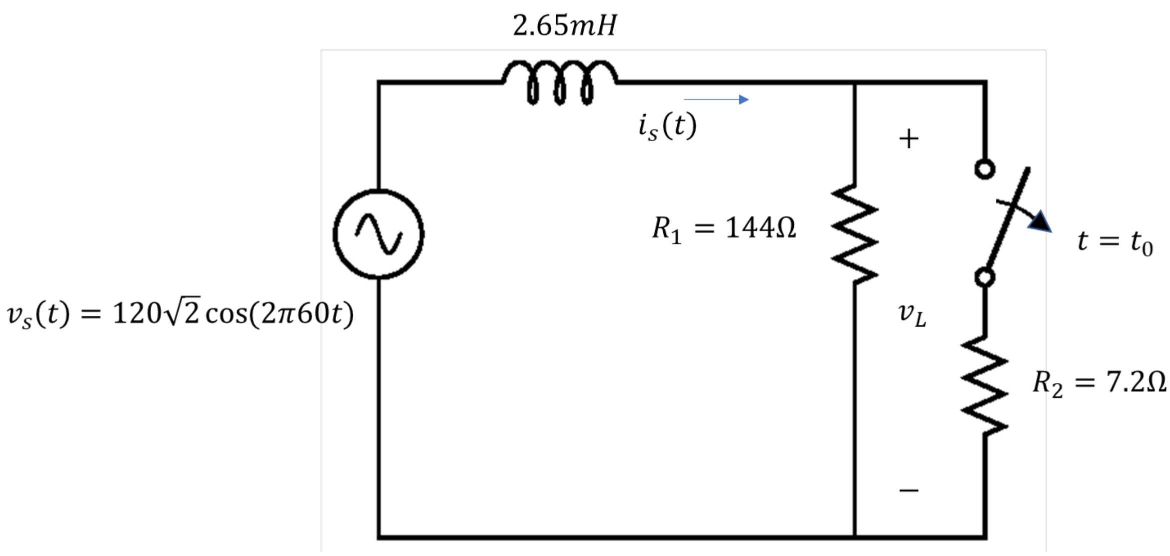
If a protection engineer does not specify a breaker according to the value computed in part f, the breaker could fail and cause a significantly larger power outage.

<u>Part b</u>

When computers and sensitive equipment are connected to a power system, they often have "surge supressors." These devices are designed to absorb excess voltages that can exist in the power system and destroy equipment.

In this problem, you are given the simple circuit shown below, which is a reasonable model for the power-distribution system in your house. The AC grid is represented by a Thevenin equivalent circuit consisting of a voltage source and a 2.65mH inductor. Two loads are connected in parallel. One load is a computer (represented by a 144Ohm resistor) and the other is an air-conditioner (represented by the 7.2Ohm resistor).





The circuit has been operating in steady-state for a long time before $t = 0$. At that time, the switch shown is opened and the current flowing to the air-conditioner stops,

a) Use impedance/phasor analysis to determine a function for the current $i_S(t)$ before $t = 0$. Use this expression to determine and explicit numeric value for $i_S(0)$. ***I stress that you should use phasor/impedance analysis. Do not waste time solving the complete differential equation.***
b) Write a differential equation that allows you to solve for $i_S(t)$ after $t = 0$.
c) Think carefully about the steady-state and transient solution to your differential equation to determine an appropriate time step that will allow you to see the complete transient and one period of the current. Explain your choice.
d) Write a MATLAB script that allows you to plot the voltage $v_L(t)$ after $t = 0$. You can use Ohm's Law to modify your equation from b to solve for the voltage $v_L(t)$.
e) What is the maximum value of the voltage v_L? Include an appropriately labeled plot of $v_L(t)$ in your report. **You may potentially need to include two plots to be able to show the transient and steady-state solutions effectively.**

Normally, engineers simply worry about steady-state operating conditions. This problem should show why it makes sense to use impedance analysis for steady-state operation and why it makes sense to use *numeric simulation to solve for transient conditions*. This sort of transient analysis must be performed when designing a product that must be connected to a grid. Such calculations (and related tests) are needed to have a product approved by UL (Underwriter's Laboratories). Knowing the peak voltage v_L, an engineer would choose a transient voltage suppressor that would clamp the voltage to an upper value much lower than the value calculated here.

Make sure to include your commented code for both parts of this problem in your report.

Problem 2 – VLSI Layout

Figure 2a shows two cascaded inverters (meaning digital NOT gates). Note that Figure 2a models how two inverters are connected inside of a digital chip. Inverters are extremely common as they form the core of any individual one-bit memory cell.

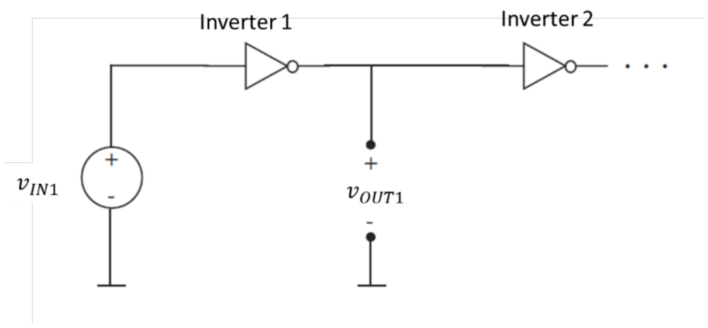
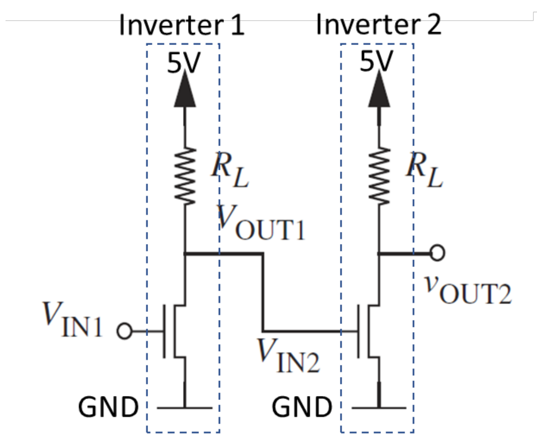


Figure 2a: Model for two cascaded inverters

Physically, the circuit shown in Figure 2a looks something like the following:



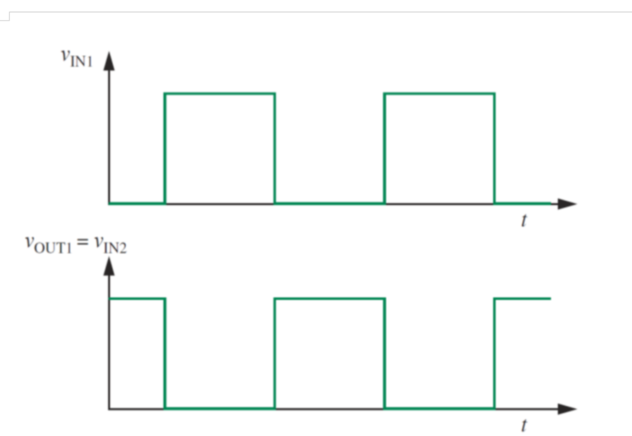


Figure 2b: Physical implementation (left) and ideal operation (right). Note that when V_{IN1} is high, $V_{OUT1} = V_{IN2}$ is low and vice versa.

The dashed boxes in Figure 2b show the circuitry that make an inverter. The bottom device is called a transistor, which is basically a switch. When $V_{IN1} = 5V$, the switch is ON and a resistor R_{ON} appears between V_{OUT1} and ground. When $V_{IN1} = 0V$, the switch is OFF and there is no connection between V_{OUT1} and ground. Figure 2c shows how inverter 1 can be modeled in both cases.

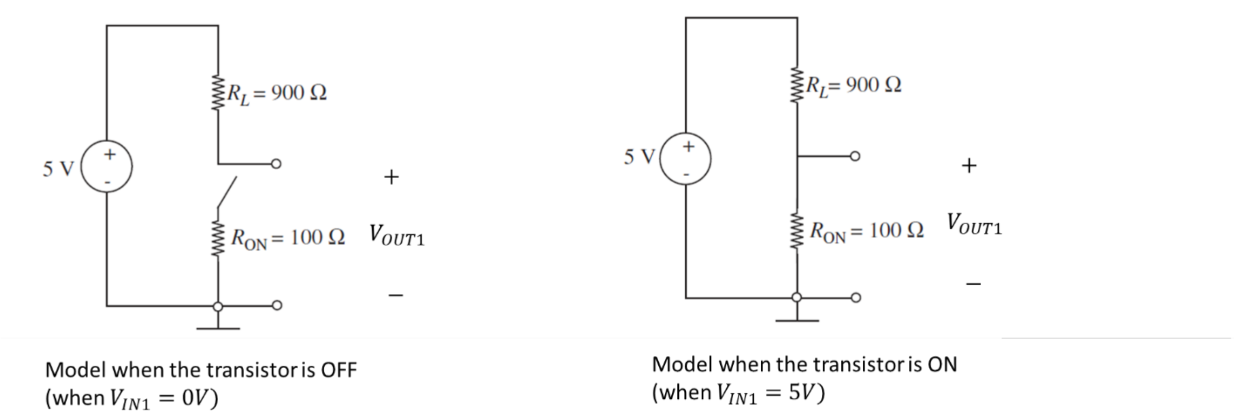


Figure 2c: Models for inverter 1 when the transistor is OFF (left) and when the transistor is ON (right).

a) Show that you have a basic understanding of the Inverter model shown in Figure 2c. Create a Thevenin equivalent circuit for both cases. What is the value of V_{OUT1} when $V_{IN1} = 0V$? What is the value of V_{OUT1} when $V_{IN1} = 5V$? Explain your answers.

In a real integrated circuit, there will be some inductance in the wire between the two inverters and there will be some capacitance at the input to the second inverter. These "parasitic" elements exist because real wires and real transistors have more complicated physics than that captured by our basic model shown in Figure 2b. In the case of a wire, for instance, the magnetic field around the wire creates a natural inductance. We often ignore this inductance, but sometimes it can matter a lot.

Figure 2d shows the more complete circuit model. If we look at v_{IN2}, we see that it has the same steady-state values we would expect (i.e. V_{IN2} is low when V_{IN1} is high and vice versa), but there is an underdamped response. The graph on the right shows that the parasitic inductance and capacitance cause some delay in the time it takes for V_{IN2} to be a valid LOW signal. For inverter 2 to have a valid LOW voltage at its input, V_{IN2} must be reliably below 1V.

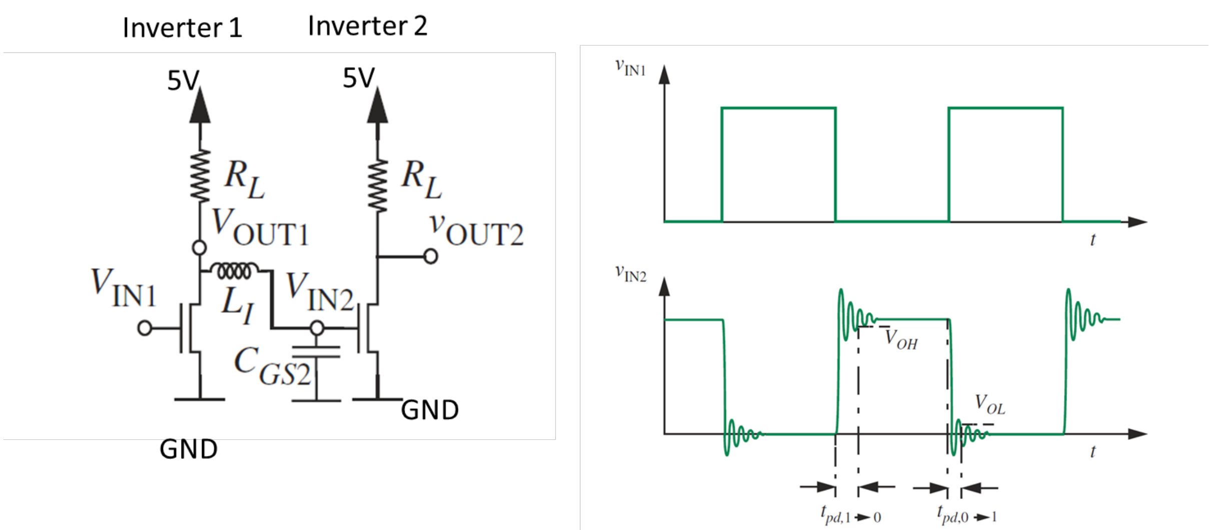


Figure 2d: Real inverter model with parasitic capacitance and inductance (left). Impact of the parasitic elements (right). Compare these waveforms with the ones shown in the ideal case in Figure 2b.

Let's consider the case in which V_{IN1} transitions from low to high. This will cause V_{IN2} to transition from high to low. In this case, the interconnection between inverter 1 and inverter 2 can be modeled as shown in Figure 2e.

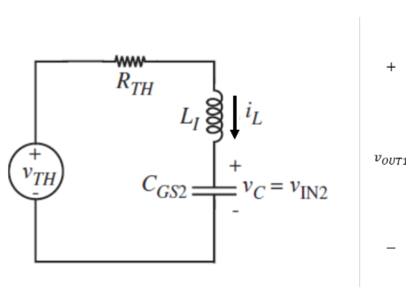


Figure 2e: Model for the circuit

We will now use this model to perform some analysis.

b) The input V_{IN1} has been low for a very long time. If so, what is the value of each of the following:
   - v_{TH}
   - R_{TH}
   - v_{OUT1}
   - v_{IN2}
   - i_L

   Explain your answers.

c) After being low for a very long time, V_{IN1} suddenly rises to a high level. We will call this time $t = 0$. Determine the following:
   - A differential equation for v_{IN2} for $t \geq 0$ in terms of R_{TH}, v_{TH}, C_{GS2}, and L_l.
   - $v_{IN2}(0)$
   - $\frac{dv_{IN}(0)}{dt}$
   - The value of v_{TH}
   - The value of R_{TH}

   You are given the following component values:

   - $L_l = 100nH$
   - $C_{GS2} = 0.1pF$
   - $R_L = 900\Omega$
   - $R_{ON} = 100\Omega$

   Explain your answers.

d) Think carefully about the steady-state and transient solution to your differential equation to determine an appropriate time step that will allow you to see the complete transient. Explain your choice. The lecture videos and notes form 10/15 will help you.

e) Discretize your equation from Part c and write a MATLAB script that will show $v_{IN2}(t)$. Make your plot over 100 periods of the damped sine wave.

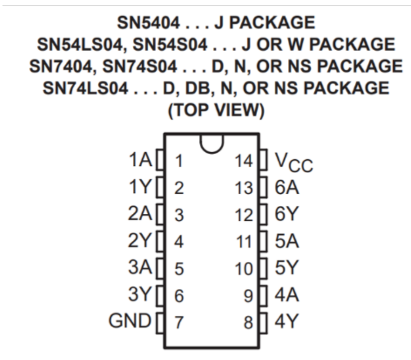
f) If a valid logic low is 1V, how long does it take for V_{IN2} to become reliably lower than 1V? In this case, the term "reliably lower than 1V" refers to the time at which V_{IN1} falls below 1V and never exceeds it again. In your plot, you should see that the voltage initially goes above and below 1V. Include a plot from MATLAB to explain your answer. There is a video on the Canvas page that shows how to zoom on a graph and pull values from it.

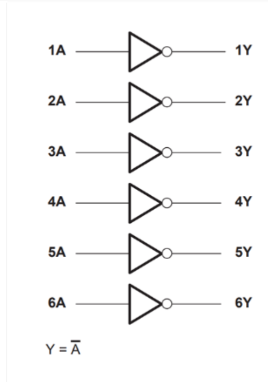
The time that you find in part f is sometimes called a propagation delay. All digital integrated circuits have propagation delays. The figure below shows typical propagation delays for the 74LS04 inverter IC. Note that it takes a maximum value of 22ns for the low-to-high transition at the output of the IC. That delay is somewhat similar to what you should have found here.

Be sure to include your commented code in your report.

switching characteristics, V_{CC} = 5 V, T_A = 25°C (see Figure 1)

PARAMETER	FROM (INPUT)	TO (OUTPUT)	TEST CONDITIONS		SN5404 SN7404 MIN	TYP	MAX	UNIT
t_{PLH}	A	Y	R_L = 400 Ω,	C_L = 15 pF		12	22	ns
t_{PHL}						8	15	





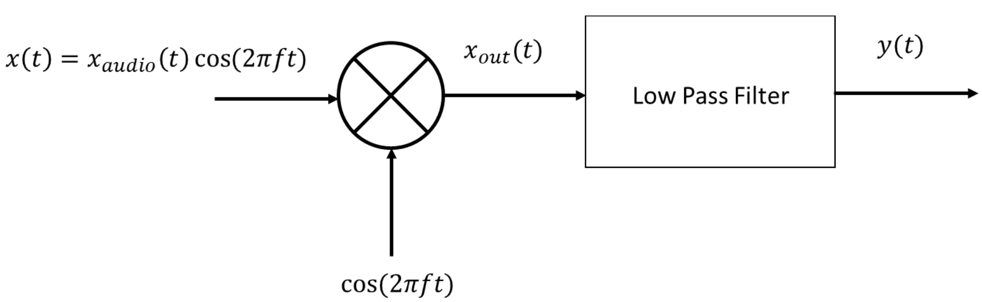
Problem 3 – Traditional Communications

Amplitude modulation (AM) is a very common technique for sending signals over the airwaves. AM is used in AM radio, and in many advanced communications systems such as cellular networks. When sending an audio signal over the airwaves, you multiply the audio signal by a high frequency sinusoid. The transmitted signal might have the form:

$$x(t) = x_{audio}(t)\cos(2\pi f t)$$

In this case, the high frequency cosine wave has a frequency f that is much higher than the audio frequencies. The highest frequencies in voice signals are approximately 2 or 3kHz.

The figure below shows a common demodulator circuit. If the circuit is designed properly, then the output signal $y(t)$ should be $y(t) \approx x_{audio}(t)$. If this signal is inside a computer, it can be sent to a sound card to be played.



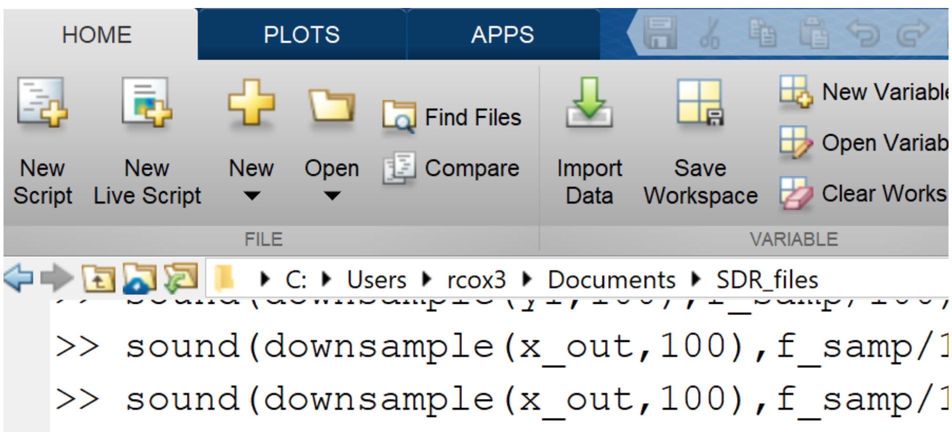
Please answer the following questions:

a) The first step in the demodulator is to multiply $x(t) = x_{audio}(t)\cos(2\pi ft)$ with $\cos(2\pi ft)$. (In other words, $x_{out}(t) = x(t)\cos(2\pi ft) = x_{audio}(t)\cos^2(2\pi ft)$). Simply this expression using trig identities. The final expression should be in terms of $x_{audio}(t)$ and should have one term with $\cos(4\pi ft)$.

b) Download the file named "problem3.bin" from Canvas. This file contains a signal $x(t)$ that was received over a radio. In this case, please note the following:

- $f = 50\text{kHz}$
- The signal $x(t)$ has been sampled by an analog-to-digital converter. The sampling frequency is $f_{samp} = 2.205\text{MHz}$, so $\Delta t = 1/f_{samp}$.
- The analog-to-digital converter saved the values as 32-bit floating point numbers.

Place the file "problem3.bin" in your active MATLAB directory. You can find your active MATLAB directory in the command window. In my version of MATLAB, I see the following:



This tells me that I'm currently in the folder named, "C:\Users\rcox3\Documents\SDR_files". Your command window might look a little different, but you should see something similar that tells you your active directory.

Create a MATLAB script in this same folder. It is important to have the script saved in your active directory. Inside this script, you will need to write the following code to load "problem3.bin":

```
fileID = fopen('problem3.bin','r');

x=fread(fileID,'precision');

fclose(fileID);
```

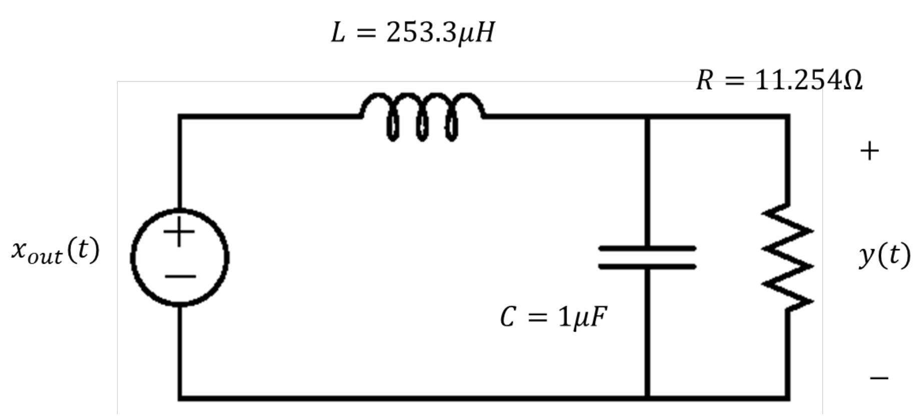
Note that the word `precision` should be replaced by the appropriate precision. Use the information at the following link to determine whether `precision` should be `single` or `double`:

https://www.mathworks.com/help/matlab/matlab_prog/floating-point-numbers.html

Remember, I said the analog-to-digital converter provided 32-bit floating point values. Once this file is loaded properly, please do the following:

- Create a time row vector having the same length as x. It should start at $t = 0$ and increase in steps of $\Delta t = 1/f_{samp}$.
- Check the size of x and determine if it is a row vector or column vector.
- Create a vector containing $\cos(2\pi f t)$
- Compute $x_{out}(t) = x(t)\cos(2\pi f t)$. Since this must be an elementwise operation, make sure that $x(t)$ and $\cos(2\pi f t)$ are both row vectors. If you need to convert between column and row vectors, use the transpose command described here: https://www.mathworks.com/help/matlab/ref/transpose.html
- Make a plot of $x_{out}(t)$ and $x(t)$ on the same axes. Zoom into this graph so you can see the underlying periods of $x_{out}(t)$ and $x(t)$. How do the frequencies of these two signals relate? Explain this using your result from part a. There is a video in Canvas that shows how to zoom a MATLAB graph. Include appropriate graphs and explanations in your report.

c) Your result in part a contained two different signals, one that is simply a scaled version of $x_{audio}(t)$ and one that is multiplied by a cosine. The term multiplied by the cosine should have a very high frequency. This term needs to be removed by a low-pass filter before the audio signal can be played. If not, the signal will sound noisy. One way to create a digital low-pass filter is to discretize the differential equation for an analog filter. The circuit shown below is an analog low-pass filter.



The signal $x_{out}(t)$ is the multiplier output, and the signal $y(t)$ is the voltage across the capacitor. For this problem, please do the following:

- Write out the differential equation for $y(t)$ in terms of $x_{OUT}(t)$, R,C, and L.
- Write a MATLAB script that will compute $y(t)$ assuming that $y(0) = 0$ and and$y'(0) = 0$. In this case, you don't have to compute Δt since it must be set by f_{sample}.
- Make a graph of $x_{out}(t)$ and $y(t)$ on the same axes. Zoom in on the graph. Explain what the filter has done.

d) Now, it's time to listen to the signals. $x_{audio}(t)$ was originally a wav file sampled at $22.05kHz$. To listen to $x_{audio}(t)$, you must downsample it by a factor of 100. To do this, use the downsample command in MATLAB:

```
x_audio = downsample(y,100)
```

This downsample operation keeps every 100th sample of y and throws the other ones away. Thus, $x_{audio}(1) = y_1(1)$, $x_{audio}(2) = y_1(101)$, $x_{audio}(3) = y_1(201)$, etc.

Use the `sound` command to play the sound. Try the following two operations:

- `sound(x_audio,22.05e3)`
- `sound(downsample(x_out,100),22.05e3)`

The downsampled version of $x_{out}(t)$ should sound noisier than $x_{audio}(t)$. Why?

I recommend playing around with the sound command. How do the waveforms sound when the sampling frequency is incorrectly given?

Note that the sampling frequency is the rate at which the digital samples in MATLAB are passed to the sound card.

In case you ever play the wrong sound, enter `clear sound`

Be sure to include your commented code in your report.

Problem 4 – Complex Communications

Modern communications systems tend to use complex signals instead of the real signals used in Problem 3. In this case, I used a software-defined radio with a digital signal processor (DSP) to create a received signal of the form:

$$x(t) = x_{audio}(t)e^{j2\pi ft}$$

In this case, I can send two audio signals at the same time on the same frequency channel. $x_{audio}(t)$ has the form:

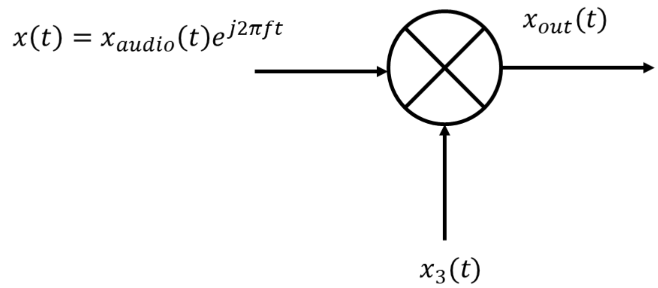
$$x_{audio}(t) = x_1(t) + jx_2(t)$$

where $x_1(t)$ and $x_2(t)$ are two separate wav files. This approach is much simpler than the one used in Problem 3, and it is commonly implemented in many digital signal processors (DSPs). I have stored the received signal $x(t)$ in a file named "problem4.bin". This file is setup in the way that DSPs store complex data. Namely, the real and imaginary parts are each 32-bit floating point values that alternate as shown below:

$Real(x(1))$
$Imag(x(1))$
$Real(x(2))$
$Imag(x(2))$
…

Please do the following:

a) A demodulator for the complex data is shown below:



Choose a signal $x_3(t)$ that will make $x_{out}(t) = x_{audio}(t) = x_1(t) + jx_2(t)$. Explain your choice.

b) Download the file named "problem4.bin" from Canvas. Place this file in your active MATLAB directory as before. Write a script that loads this file into a vector named `x_in` using the same commands used in Problem 3. The only difference here is that the real and imaginary parts of x alternate as noted above. Create the complex vector `x` by alternating the entries from `x_in`. So your vector x should have entries like so: $x(1) = x_{in}(1) + jx_{in}(2)$, $x(2) = x_{in}(3) + jx_{in}(4)$, etc. Please do the following:

- Write code that will load the data file "problem4.bin" into the vector `x_in`
- Load the values from `x_in` into the complex vector `x` as described above.
- Create $x_{out}(t) = x_{audio}(t)$ by multiplying $x(t)$ by $x_3(t)$.

As in Problem 3, you are told the following:

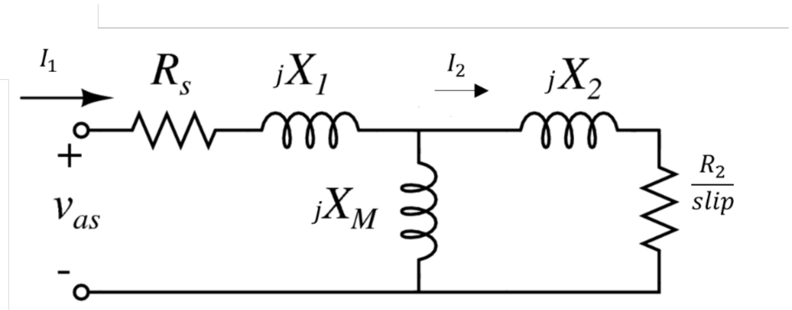
- $f = 50kHz$
- $f_{samp} = 2.205MHz$, so $\Delta t = 1/f_{samp}$
- Before computing $x_{out}(t) = x_{audio}(t) = x(t)x_3(t)$, make sure that both $x(t)$ and $x_3(t)$ are row vectors. If you need to convert between column and row vectors, use the transpose command described here: https://www.mathworks.com/help/matlab/ref/transpose.html

c) Your output $x_{out}(t)$ should be $x_{out}(t) = x_{audio}(t) = x_1(t) + jx_2(t)$. $x_1(t)$ and $x_2(t)$ are both wav files sampled at $22.05kHz$. Use the downsample and sound commands just like you did in Problem 3 to listen to the two signals. They should be very different, one light and one dark.

Be sure to include your commented code in your report.

Problem 5 – Induction Motor

You are given an induction motor, which is an AC motor. It has the following equivalent circuit:



The motor is connected to a 2300V AC voltage source at 60Hz. You are told the following about the induction motor circuit:

- Since the machine is connected to a 60Hz voltage source, we model the inductances as reactances (i.e.$j\omega L = jX_L$). In this case, you are told directly what the X values are in Ohms assuming $\omega = 2\pi 60$ rad/sec. (see below).
- One of the resistances depends upon what is known as slip. Slip is defined as follows:

$$slip = \frac{2\pi 30 - \Omega(t)}{2\pi 30},$$

  where $\Omega(t)$ is the motor speed. The full speed of the motor is $\Omega = 2\pi 30$ rad/sec. If the motor reaches this full speed, the slip becomes 0 and thus the resistance becomes infinite and the current i_2 becomes zero. Don't worry about the physics. What matters for you is that one of the resistances depends upon the speed of the motor.

You are given the following parameters:

- $R_s = 0.262\ Ohms$
- $X_1 = 1.206\ Ohms$
- $X_M = 54.02\ Ohms$
- $X_2 = 1.206\ Ohms$
- $R_2 = 0.187\ Ohms$

The mechanical portion of the induction motor is also governed by Newton's Second Law. In this case, we have:

$$J\frac{d\Omega}{dt} = T_m - \beta\Omega(t)^2$$

In this case, the torque T_m produced by the motor is

$$T_m = \frac{3|I_2|^2\left(\frac{R_2}{slip}\right)}{2\pi 30}$$

Note that $|I_2|$ is the magnitude of the static phasor for I_2.

You are also given the following mechanical parameters:

- $J = 63.87$
- $\beta = 0.2565$

a) To solve this problem, first use mesh analysis to write two equations that can be solved for the phasors I_1 and I_2 at any given speed. Note that you can solve this simply using impedance/phasor analysis. (I'm only asking you to write the equations here – you cannot solve them until you have computed the speed).
b) Discretize the differential equation, letting $\Delta t = 0.001s$. Write a MATLAB script that allows you to plot the speed $\Omega(t)$ and the current phasor I_1. To approach this problem, I suggest you do the following at each time step:
   - Step 1: Compute slip
   - Step 2: Solve the system of equations you found in part a.
   - Step 3: Use the result from solving the system of equations to determine T_m.
   - Step 4: Find $\Omega(t + \Delta t)$.
c) Generate plots showing $\Omega(t)$ and the current phasor I_1 for 10 seconds.

This type of analysis is often used to analyze what happens during motor starting. You should notice that the current is quite large when the speed is increasing, and then it decreases as the motor reaches steady-state. I will upload a file showing what the output should be.