ECE 2260 Lab 5: RC Circuits

Hunter Van Horn

February 19, 2025

Purpose

The purpose of this lab is to sumulate and measure the time constant of an RC Circuit

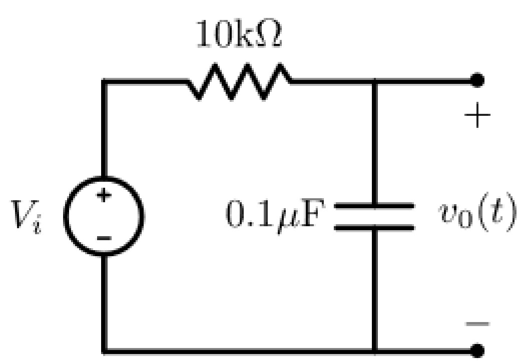
Preliminary

1. In Python, write an analytical solution for $v_o(t)$ (the voltage across the capacitor) assuming V_i is a 0 to 5V step function. Then use Python to graph the function from $t \in [0s, 5\tau]$
2. Using LTspice, create and simulate the RC circuit and use the cursors to identify τ
3. Using the breadboard, build the circuit RC circuit applying 0-5V across using a 100Hz square wave. Measure the time constant using the oscilloscope.

Equipment

- Breadboard
- Multimeter
- Power Supply
- Signal Generator
- Oscilloscope
- Resistor: 10kΩ
- Capacitor: 0.1µF

The Circuit



Python Code

Author: Hunter Van Horn, Dexter Ward
Date: 02/06/2025

Dependencies

[2]:
```python
import numpy as np
import matplotlib.pyplot as plt
```

Voltage Across the Capacitor

[3]:
```python
def capacitorResponse(vf, vo, t, tau):
    return vf+(vo-vf)*np.exp(-t/tau)
```

This function is used to determine the voltage across the capacitor at any given time (t) provided the time constant (τ) and the inital (V_o) and final (V_f) voltage.

Calculating Tau

[4]:
```python
def calculateTau(r, c):
    return r*c
```

This function is used to determine the τ of an RC circuit given the resistor and capacitor values.

Finding Tau

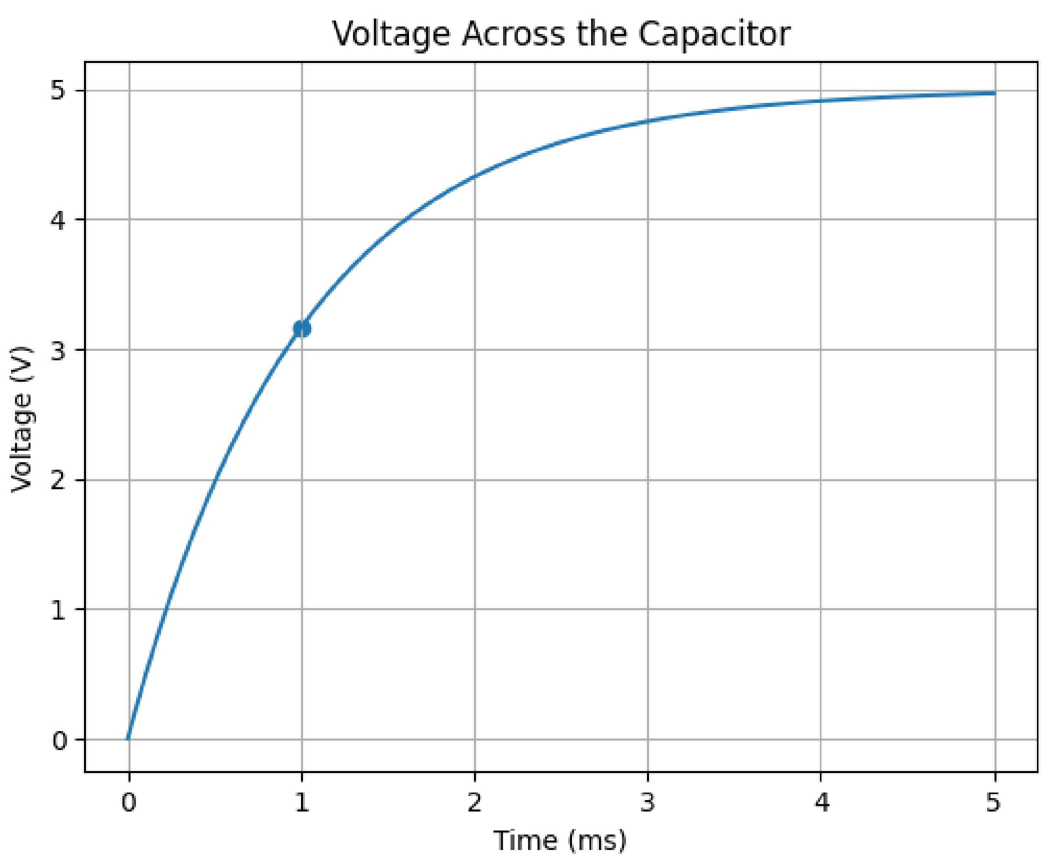
[5]:
```python
def positionAtTau(yVal, vf, vo):
    return np.argmax(yVal > capacitorResponse(vf,vo,1,1))
```

This function can be used to find the voltage at τ in a given array of voltage values across a capacitor provided V_o and V_f. This function is able to determine the voltage at $t = \tau$ using the capacitor response function. From there the function utilizes the numpy argmax function to determine the first array position that exceeds the determined voltage.

Graphing

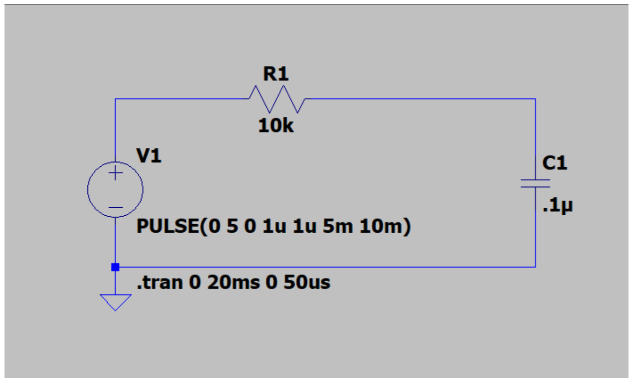
[6]:
```python
def graphResponse():
    r = 10000
    c = .0000001
    tau = calculateTau(r,c)
    vf = 5
    vo = 0
    stepSize = .000001
    xVal = np.array([0])
    yVal = np.array([capacitorResponse(vf, vo, xVal[-1], tau)])
```

```
    while(xVal[-1] < 5*tau):
        xVal = np.append(xVal, [xVal[-1]+stepSize])
        yVal = np.append(yVal, [capacitorResponse(vf, vo, xVal[-1], tau)])

    xVal = xVal*1000

    plt.plot(xVal,yVal)
    plt.scatter(xVal[positionAtTau(yVal, vf, vo)],yVal[positionAtTau(yVal, vf,␣
 ↪vo)])
    plt.xlabel('Time (ms)')
    plt.ylabel('Voltage (V)')
    plt.title('Voltage Across the Capacitor')
    plt.grid()
    plt.show()

if __name__ == '__main__':
    graphResponse()
```

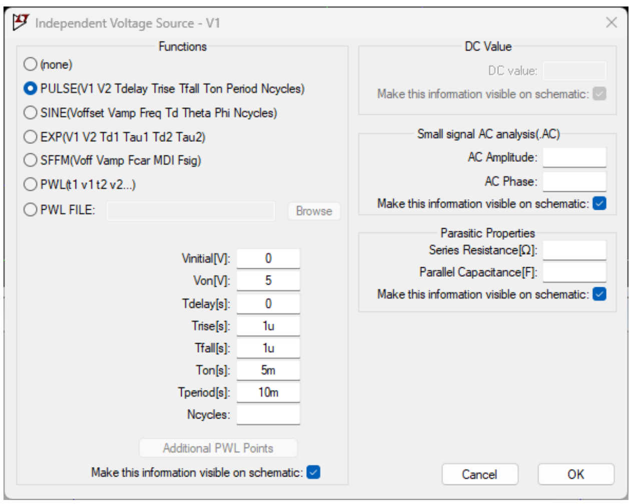


LTspice Simulation

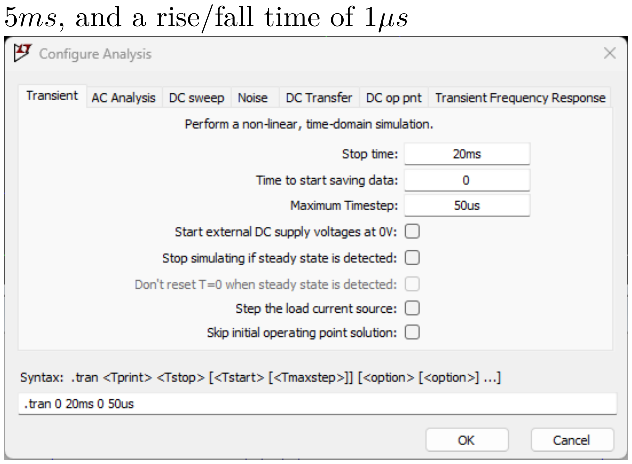
Circuit Diagram



Settings

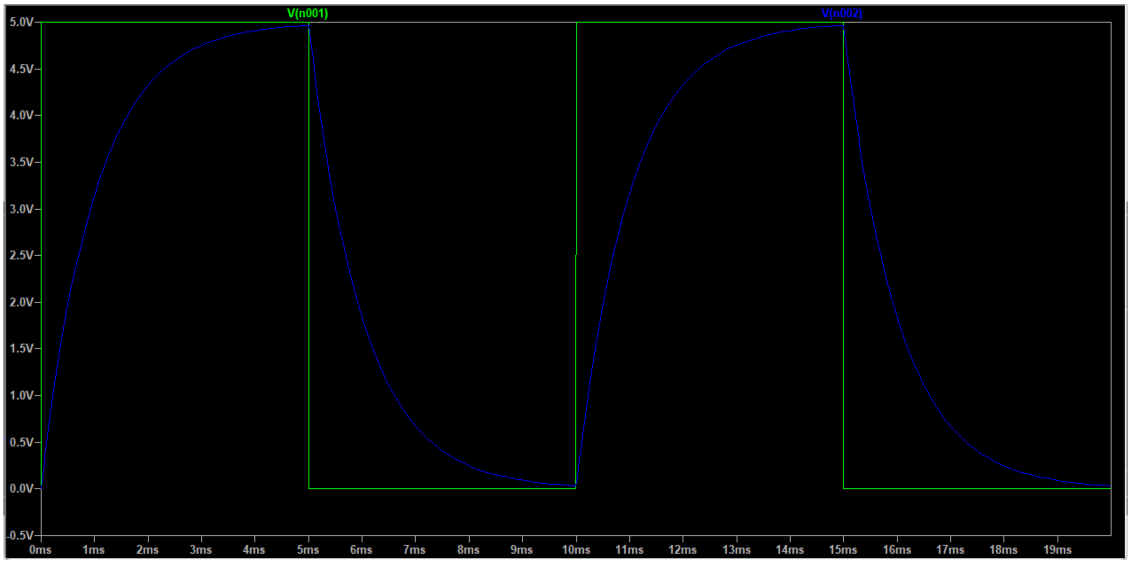


This sets the voltage source to a square wave from 0 to $5V$ with a period of $10ms$, an on time of $5ms$, and a rise/fall time of $1\mu s$

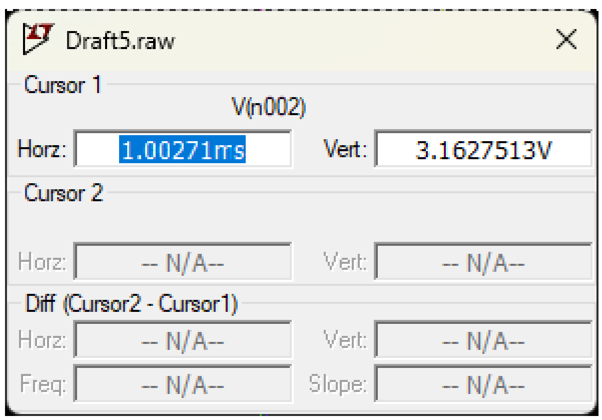


This sets the oscilloscoope output to show two periods with a maximum step size of $50\mu s$

Oscilloscope Trace

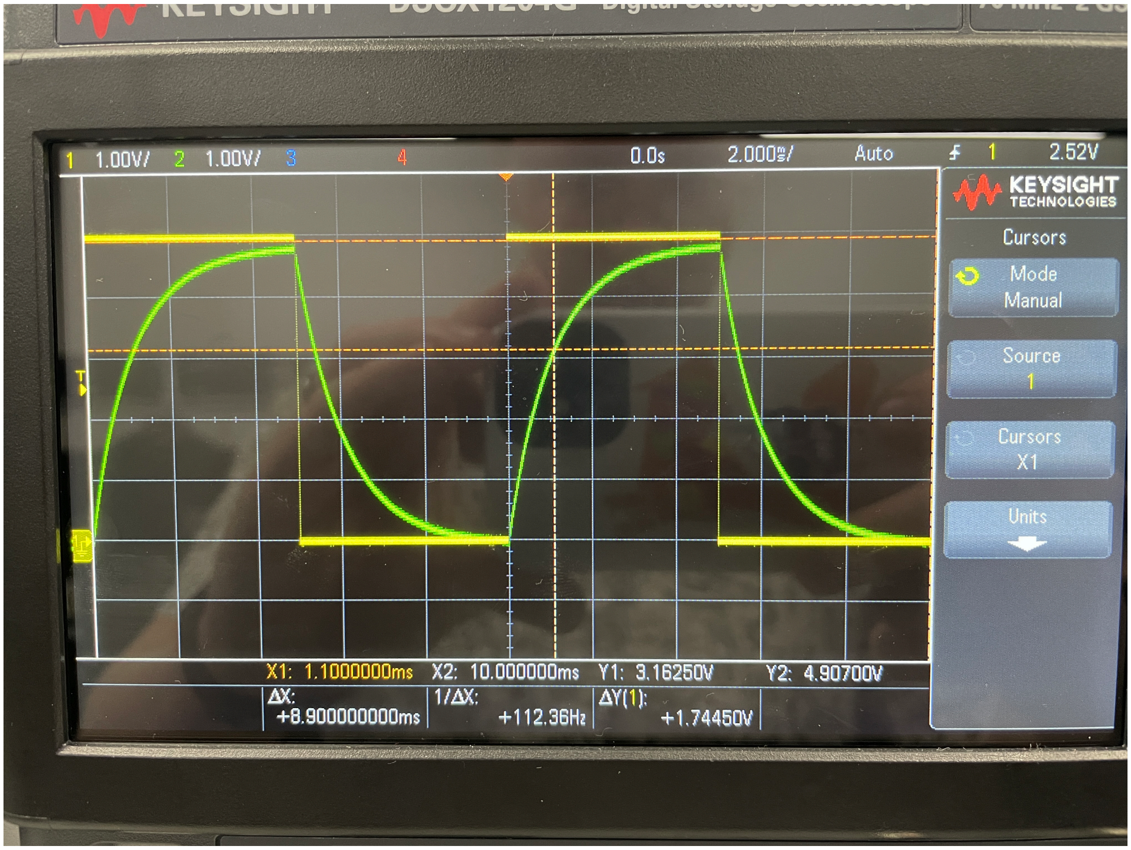


Osciloscope Measurement



Building the Circuit

Oscilloscope Trace



Conclusion

	Python	LTspice	Breadboard	Analytical
Voltage at τ	3.1606027941428163	3.1627513	3.16250	3.160602794
—	—	—	—	—
% Error	0.000%	.068%	.060%	0.000%

In this lab my lab partner, Dexter Ward, and I created a python program that was able to take an input voltage, a resistance value, and a capacitor value and graph the transient response of the RC circuit. Once the program had calculated all of the y-values for the curve we were able to pass that into a different function that used the equation $V = V_f e^{-1}$ to calculate the voltage at τ and returned the array position of the value. We then created the circuit in LTspice and ran the spice simulation. Using the simulated oscilloscope we were able to identify the voltage at τ. We then built the circuit on a breadboard and used a wave generator to power the circuit. Using an oscilloscope and the cursor function we were able to measure the voltage at τ. These measurements, along with the analytical answer are shown above, as is the percentage error for each voltage value. All voltage measurements were were within .1% of the analytical solution for the given circuit.