ECE 2260 - Fundamentals of Electrical Circuits: Lab 5

Hunter Van Horn

February 19, 2025

Purpose

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

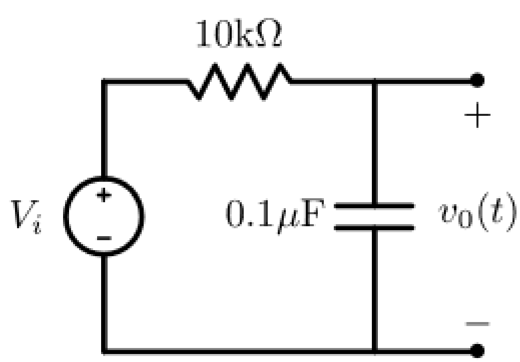
Preliminary

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

[1]:
```
import numpy as np
import matplotlib.pyplot as plt
```

Voltage Across the Capacitor

[2]:
```
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

[3]:
```
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

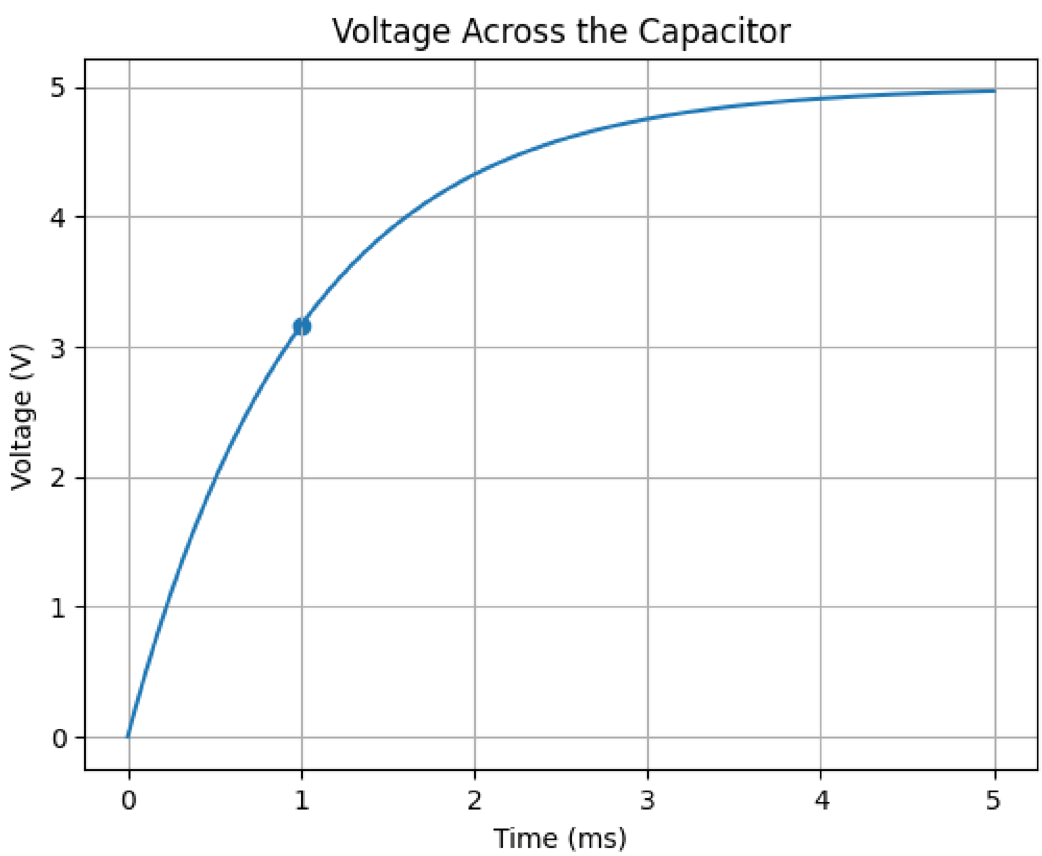
[4]:
```
def positionAtTau(yVal, vf, vo):
    return np.argmax(yVal > capacitorResponse(vf,vo,1,1))
```

This function can be used to find 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

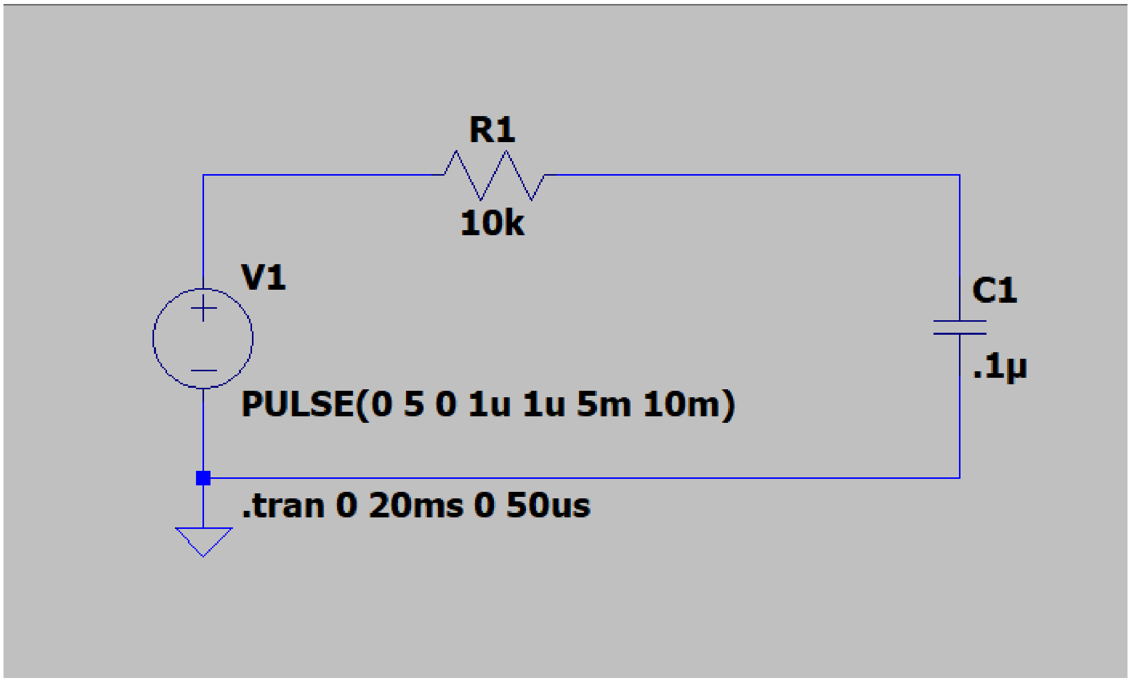
[ ]:
```
def graphResponse():
    r = 10000
    c = .0000001
    tau = calculateTau(r,c)
    vf = 5
    vo = 0
    stepSize = .000001
    xVal = np.array([0])
    yVal = np.array([capacitorResponse(vf, vo, xVal[-1], tau)])
```

```
    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()
```

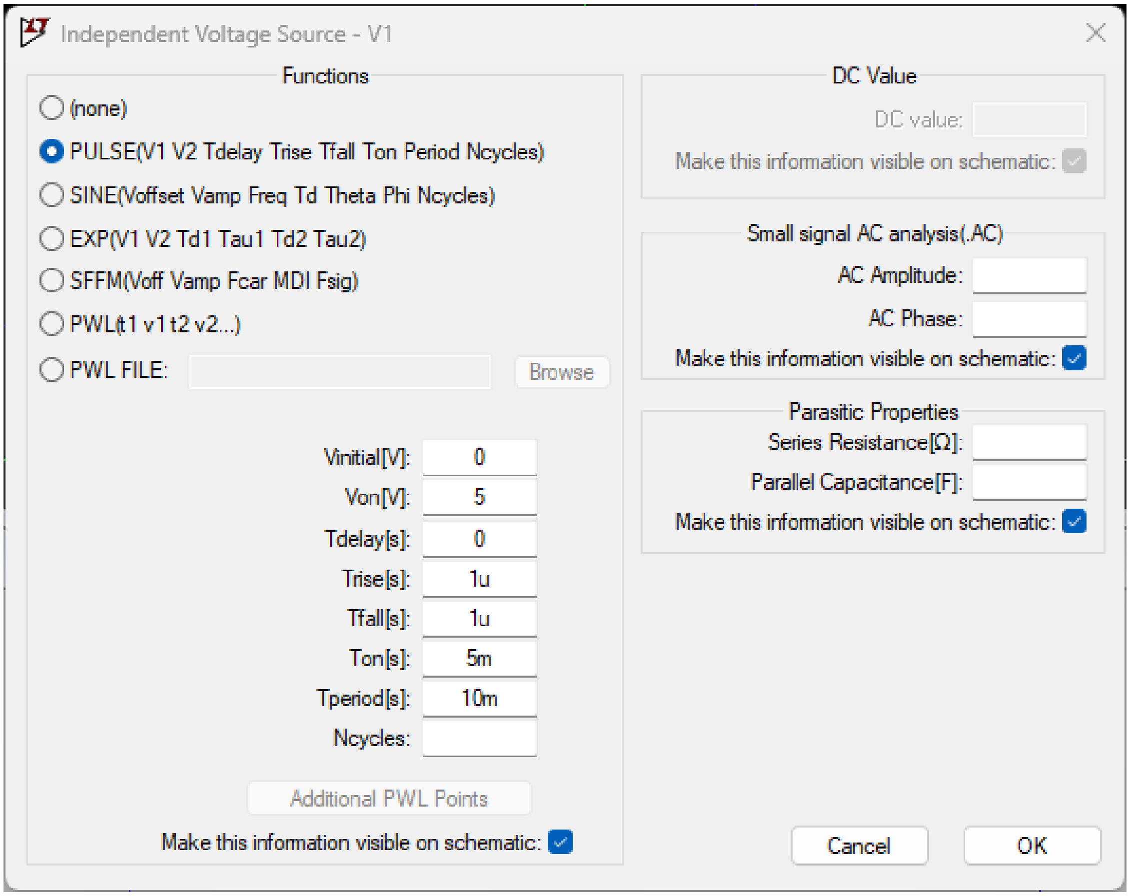


LTspice Simulation

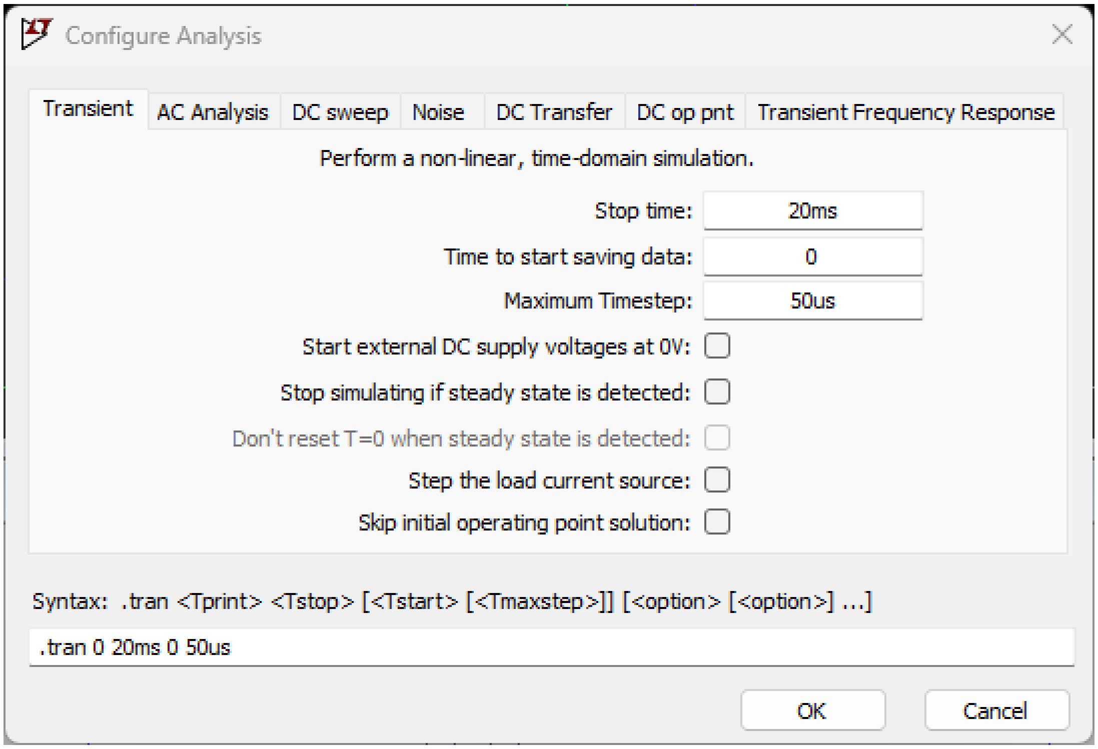
Circuit Diagram



Settings

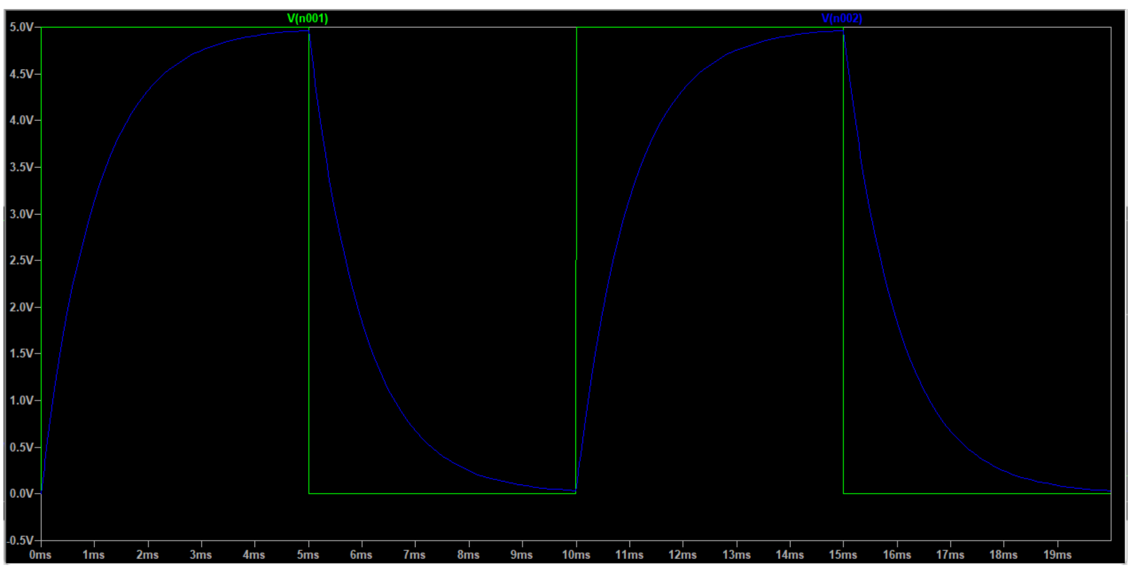


This sets the voltage source to a square wave from 0 to $5V$ with a period of $10ms$, an on time of $5ms$, and a rise/fall time of $1\mu s$



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

Oscilloscope Trace



Building the Circuit