Assignment-2 (LT-Spice)

Deenabandhan N
EE23B021

Introduction :

- In this assignment , I have learned how to use file handling and constructing matrices to solve the circuit.
- This assignment involves three parts :
  - File handling
  - Nodes labelling
  - Matrix construction and solving
- I have explained each process and the references made and additional test cases in this pdf file

Assumptions made :

- There is a **GND** in the circuit mentioned in the file.
- We have only **DC** circuits in the circuit file.

File Handling :

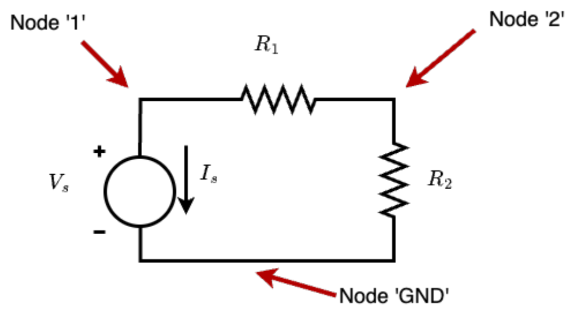
- The function used to extract information from file is **read()**.
- I have used **file.readlines()** to read all the lines from the file and convert into a list .
- There is a **check** variable to check the presence of **.circuit and .end** in the .ckt file.
- It returns the list *elements* whose element is the data stored in each line.

Nodes labelling :

- In this code , the dictionary *nodes* is used to assign each node to a number such that we can perform matrix operations i.e this number will serve as an index to the matrix.
- It is assigned as a **defaultdict()** with initial value to be (-1) to avoid reassigning the same node twice.
- We also have two dictionaries $name\_v$ and $name\_i$ to store the index of the voltage variable and current variable respectively from the matrix which we are going to solve. This is done to back-propagate to the original nodal/voltage names after solving the matrix.
- The code checks if the node is **GND** and assigns it to the *nodes* with the index **c** and assigns $name\_v$[c] as node name and updates c.
- The code then checks if the given element is a voltage source and if it is a voltage source , assigns the $nodes\_i$[c] with the voltage source name and updates c.

Matrix constrcution and solving :

- Consider the following the circuit and the KCL equations corresponding to this circuit is



$$\begin{aligned} I_s + \frac{V_1 - V_2}{R_1} &= 0 \\ \frac{V_2 - 0}{R_2} + \frac{V_2 - V_1}{R_1} &= 0 \end{aligned}$$

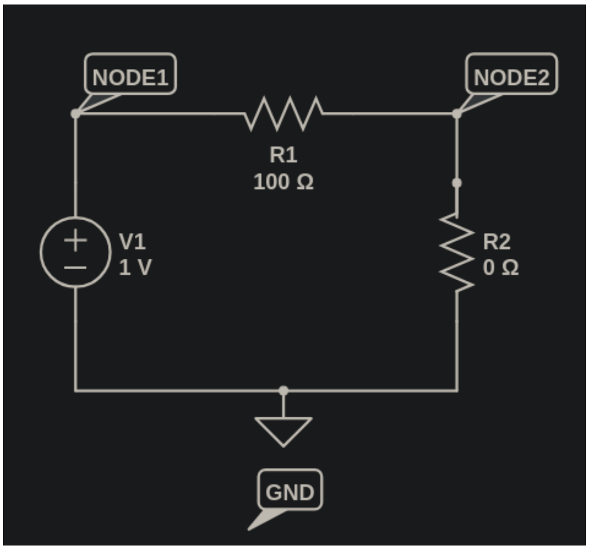
- We have three variables here are : I_s , V_1 , V_2 and the coefficients of V_1 and V_2 in the second equation are $\frac{-1}{R_1}$ and $\frac{1}{R_1} + \frac{1}{R_2}$
- We can infer from the above equations that for 'voltage node' variables , the elements of the matrix can be referred as below.

$$\text{mat[i , j]} = \begin{cases} \Sigma\ (-1 \times \text{Admittances) connected between the nodes } \mathbf{i} \text{ and } \mathbf{j} & \text{if } \mathbf{i} \neq \mathbf{j} \\ \Sigma\ (\text{Admittances) connected to the node } \mathbf{i} & \text{if } \mathbf{i} = \mathbf{j} \end{cases}$$

- So in this code while iterating through the elements we follow the above condition in the matrix if it is a 'R' element.
- For the current source case , if the current comes into the node , we add it to the constant matrix , else we subtract it from the respective element of the constant matrix.
- If it is a voltage source , we will have two equations : One for doing 'KCL' and other for matching the voltage difference between the nodes with the given source voltage
- So for the first case we will make the current variable i.e current passing through the node to be +1 if it is going to the node and -1 if it is going out of the node.
- For the second case we will make the node voltage variable at the first node to be +1 and second node to be (-1) and the constant matrix will get the value of the source voltage.

Additional test cases :

Handling the case of Resistance=0 :

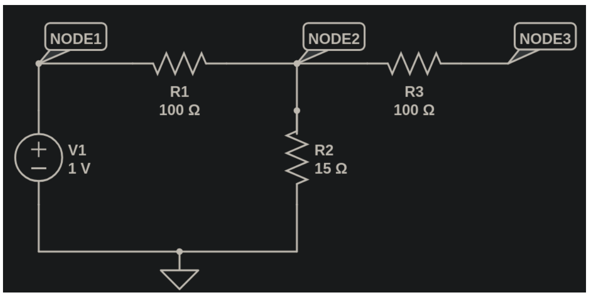


- When resistance between two node is zero , we cannot perform **KCL** as it involves division of voltage difference by resistance.
- To avoid this , I have replaced every such case by changing it into a voltage source of voltage 0.
- To make count of this change , I have used **tc** to be my counter variable and used it to assign voltage name as a string

```
f"v#{tc}"
```

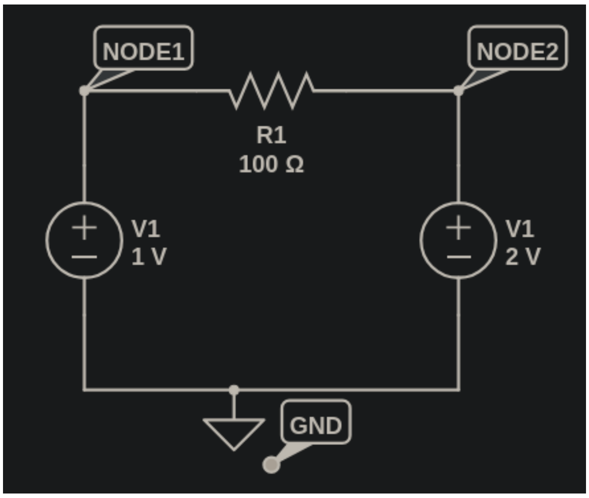
- I have intentionally used **v#** , as # is used as a comment in .ckt files and no other voltage source will have such name.
- I am also storing the elements that represent resistors with 0 resistance in a list **rm_elem** which would be removed from elements as we have replaced it with the voltage source of 0 voltage.
- At the end of the process , I am removing the elements that are common in **elements** and **rm_elem**.

Handling when the circuit contains a floating wire :



- In this case $V_{node2} = V_{node3}$ for which my initial code was raising error So I rectified it by storing the nodes of the voltage also.
- The floating element can even be a Voltage source but it cannot be a current source as we current cannot flow through the open circuit.
- The test case for this is given in the */extra* folder.

Having two voltage sources with the same name :



- In this case , there is two voltage sources with same name which if happens , we cannot distinguish between the current passing through the voltage sources.
- Even though it is a circuit file issue , I have raised an error.

Having nun-numerical values :

- If the file has non-numerical values as resistance or current or potential difference , this code will raise an error.

```
.circuit
v1 1 GND dc 2
r1 1 GND kgu
.end
```

References :

- File handling
- Numerical python documentation
- Nodal analysis matrix.
- Discussions of edge cases with
  - Deepak Charan EE23B022
  - Nishanth EE23B047