

## LABORATOR 2

---

### Introducere în programul LTSpice. Simularea de curent continuu

În acest capitol este făcută o introducere în programul de simulare a circuitelor electrice și electronice SPICE (versiunea LTSpice) și este analizată simularea circuitelor electrice funcționând în curent continuu.

#### 1. Ce este SPICE?

SPICE (Simulation Program with Integrated Circuit Emphasis) este un simulator numeric de circuite electrice și electronice de uz general, open-source. A fost dezvoltat în 1973 de către Laurence Nagel de la Electronics Research Laboratory de la UC Berkeley. Spice este folosit în proiectarea circuitelor integrate, pentru a verifica corectitudinea proiectării și pentru a simula comportarea reală a circuitelor. Pot fi folosite diferite tipuri de analize precum analiza de curent continuu, analiza de regim tranzitoriu, analiza în frecvență, analiza toleranțelor, analiza funcției de transfer, analiza de zgomot etc.

Spice folosește pentru construirea sistemului de ecuații metoda modificată a potențialelor nodurilor adaptată analizei cerute. Plecând de la un fișier text netlist care descrie elementele de circuit și conexiunile dintre ele (fișierul netlist poate rezulta și din realizarea grafică a schemei, în cazul în care implementarea respectivă include și această posibilitate), SPICE generează ecuațiile (sistem de ecuații diferențiale algebrice neliniare) ce sunt rezolvate folosind metode de integrare implicită, metoda Newton și tehnici cu matrici rare. SPICE poate simula cu precizie ridicată circuite care acoperă o gamă variată de aplicații: de la surse în comutație până la celule de memorie, oscilatoare și amplificatoare.

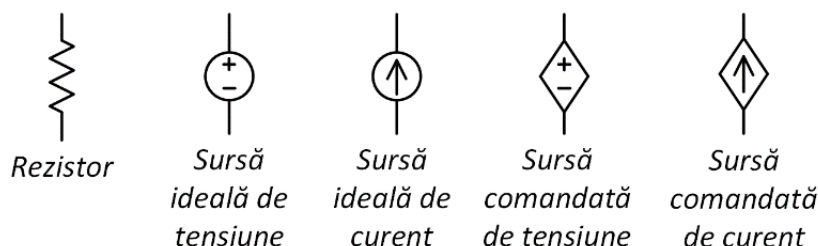
Prima versiune comercială a SPICE a fost ISPICE, iar cele mai importante de-a lungul timpului au fost HSPICE și PSPICE. Producătorii de circuite integrate, în special companiile mari, dețin grupuri pentru dezvoltarea de programe bazate pe SPICE pentru simularea circuitelor. Printre acestea se numără ADICE de la Analog Devices, LTSpice dezvoltată inițial de Linear Technology și disponibilă publicului gratuit, Mica de la Freescale Semiconductor și TINA-TI de la Texas Instruments. Atât LTSpice cât și TINA-TI conțin modele pentru produsele firmelor dezvoltatoare. În acest îndrumar este folosită pentru simularea circuitelor versiunea gratuită LTSpice.

Circuitele analizate cu SPICE pot conține rezistoare, condensatoare, bobine, bobine cuplate, surse de tensiune și de curent independente, surse de tensiune și de curent dependente (comandate), linii de transmisiune și cele mai uzuale dispozitive semiconductoare, diode, tranzistoare bipolare – TB, tranzistoare cu efect de câmp cu grilă-joncțiune – TECJ, tranzistoare cu efect de câmp metal-oxid-semiconductor (MOS) – TECMOS, tranzistoare cu efect de câmp metal-semiconductor – TECMES precum și multe alte componente.

În figura 1.1 sunt prezentate simbolurile conform standardului European IEC și standardele din America și Japonia (care corespund majorității instrumentelor de realizare a schemelor din

diferitele variante Spice). Se pot observa cu ușurință diferențele dintre aceste standarde. De exemplu, mare atenție la diferența dintre simbolul sursei de curent din standardul american și cel al sursei de tensiune din standardul european - spațiul dintre săgeata interioară și contur.

a). simbolizare conform standardelor din America și Japonia:



b). simbolizare conform standardului european IEC

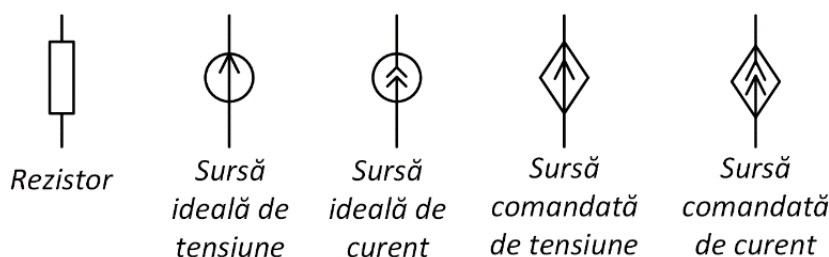


Fig. 1.1. Simboluri folosite în realizarea schemelor electrice (conform standardului European IEC și conform standardelor din America și Japonia).

## 2. Tipuri de analize

Analiza de curent continuu (DC analysis) calculează potențialele și curenții doriți în condițiile în care condensatoarele sunt deconectate și bobinele scurtcircuitate, parcurgând de obicei liniar valoarea unei surse (de curent sau de tensiune), între o valoare inițială și una finală cu un anumit pas. Sunt rezolvate iterativ ecuațiile neliniare care descriu circuitul, neliniaritățile provenind în principal din caracteristicile dispozitivelor semiconductoare.

Analiza de curent alternativ (analiza AC) calculează valoarea complexă a potențialelor nodurilor unui circuit liniar în funcție de frecvența semnalului sinusoidal aplicat la intrare. Pentru circuitele neliniare, cum ar fi de exemplu circuitele cu tranzistoare, acest tip de analiză presupune realizarea condiției de semnal mic. Aceasta înseamnă că amplitudinea tensiunii sursei de excitație se presupune mică față de tensiunea termică ( $V_{th}=kT/q=25.8\text{mV}$  la  $27^\circ\text{C}$ ). Numai în cazul respectării acestei condiții, circuitele neliniare pot fi înlocuite, pentru variații în jurul punctului static de funcționare, prin echivalentul lor liniarizat.

În cazul analizei tranzitorii (transient analysis) se calculează pentru fiecare nod al circuitului formele de undă în funcție de timp. Este o analiză de semnal mare, aceasta însemnând că asupra amplitudinii semnalului de intrare nu se impune nicio restricție.

SPICE construiește și apoi rezolvă ecuațiile corespunzătoare circuitului analizat utilizând metoda potențialelor nodurilor în același fel în care un operator uman scrie mai întâi teoremele Kirchhoff I și II pentru circuitul considerat și relațiile constitutive ale elementelor din laturi și apoi rezolvă sistemul de ecuații rezultat. De aceea cea mai bună cale de a înțelege cum funcționează SPICE este de a rezolva manual un circuit luat ca exemplu.

Se consideră exemplul de mai jos. Fie rețeaua rezistivă din figură:

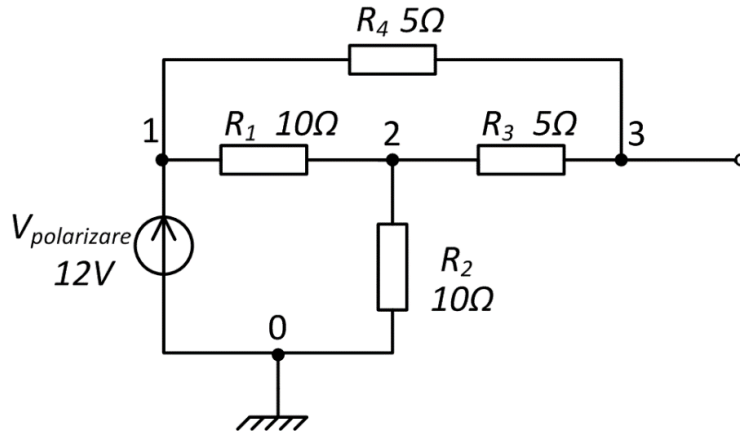


Fig. 1.2. Circuitul considerat (circuit în T podit)

Regimul de curent continuu va fi determinat inițial manual, iar apoi folosind LTSpice.

Se cer potențialele nodurilor circuitului rezistiv în T podit din Fig. 1.2. și să se determine curentul furnizat de sursa de polarizare  $V_{polarizare}$ . Valorile sunt:  $V_{polarizare}=12V$ ,  $R_1=10\Omega$ ,  $R_2=10\Omega$ ,  $R_3=5\Omega$ ,  $R_4=5\Omega$ . Nodurile sunt numerotate 0, 1, 2, 3.

Soluție: Potențialul nodului 1 este egal cu  $V_{polarizare}$ , adică 12V. Potențialele nodurilor 2 și 3 se determină prin rezolvarea sistemului care rezultă prin scrierea ecuațiilor la noduri:

$$\text{Nodul 2: } -G_1V_1 + (G_1 + G_2 + G_3)V_2 - G_3V_3 = 0 \text{ (ecuația 1)}$$

$$\text{Nodul 3: } -G_4V_1 - G_3V_2 + (G_3 + G_4)V_3 = 0 \text{ (ecuația 2)}$$

$G_1, G_2, G_3$  și  $G_4$  fiind conductanțele corespunzătoare rezistoarelor  $R_1, R_2, R_3$  și  $R_4$ :  $G_1 = G_2 = 0.1 [S]$  și  $G_3 = G_4 = 0.2 [S]$ . Rezultă sistemul de două ecuații cu două necunoscute  $V_2, V_3$ :

$$0.4 \cdot V_2 - 0.2 \cdot V_3 = 1.2$$

$$-0.2 \cdot V_2 + 0.4 \cdot V_3 = 2.4$$

ce se rezolvă imediat. Soluția este:

$$V_2 = 8 [V]$$

$$V_3 = 10 [V]$$

Curentul prin sursa de polarizare este egal cu curentul  $I_{R2}$ :

$$I_{polarizare} = I_{R2} = G_2 \cdot V_2 = 0.8 [A]$$

În SPICE ecuațiile 1 și 2 sunt formulate pornind de la graful care descrie topologia circuitului și sunt apoi rezolvate prin utilizarea eliminării Gaussiene. Rezultatele obținute în

exemplul de față prin analiza manuală a circuitului sunt comparate cu acelea obținute folosind SPICE.

### 3. Sintaxa SPICE

A fost folosit programul LTSpice, versiunea gratuită pentru PC. Schema unui circuit trebuie descrisă într-un format care să poată fi înțeles de către SPICE (limbajul de intrare SPICE). În această etapă vom examina tipurile de informații conținute de fișierul de intrare netlist. Schema circuitului poate fi descrisă și direct folosind un editor grafic, caz în care fișierul netlist este generat automat. Fișierul netlist poate fi editat cu orice editor de text.

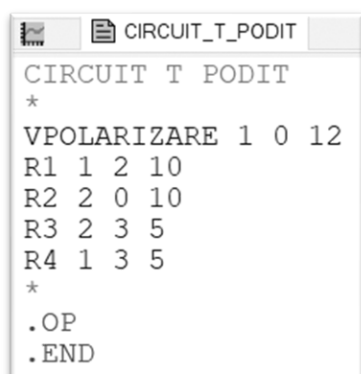
Limbajul de intrare SPICE constă dintr-o succesiune de declarații. O linie care continuă linia precedentă trebuie să înceapă cu un + în prima coloană. O declarație conține un număr de câmpuri separate prin caractere de delimitare care pot fi un spațiu, o virgulă, un semn egal (=) sau o paranteză la dreapta sau la stânga; spațiile suplimentare sunt ignorate. Programul nu face diferența între majuscule și minuscule (este case insensitive).

În cele ce urmează, elementele sintactice ale limbajului SPICE se vor introduce în mod natural prin crearea fișierului de intrare SPICE pentru circuitul în T podit din figura 1.2.

Orice fișier de intrare SPICE trebuie să înceapă cu o declarație de titlu (title statement) prin care se identifică circuitul și trebuie să se termine cu o declarație de încheiere (end statement) care are întotdeauna forma .END. În fișierul de intrare SPICE aceste două linii trebuie să fie întotdeauna prima și respectiv ultima.

Unele linii au în prima coloană un asterisc; acestea sunt declarații comentariu (comment statements) care pot fi folosite pentru comentarii legate de descrierea circuitului și de cererile de analiza.

Specificația SPICE este orientată spre componenta de circuit. Ca urmare, pentru a descrie circuitul din figura 1.2, este necesar ca informațiile conținute în figură (numele, conectivitatea și valoarea elementului) să fie transferate unui fișier text, prezentat mai jos. Descrierea circuitului constă din declarațiile din liniile 3-7; fiecare element poate fi identificat cu ușurință în figură. Celelalte linii din fișierul de intrare SPICE sunt titlul, care specifică circuitul drept CIRCUIT T PODIT, două linii de comentarii care au un asterisc în prima coloană și linia .END care întotdeauna trebuie să încheie fișierul netlist.



```
CIRCUIT T PODIT
*
VPOLARIZARE 1 0 12
R1 1 2 10
R2 2 0 10
R3 2 3 5
R4 1 3 5
*
.OP
.END
```

Alt tip de declarație care trebuie să apară într-un fișier de intrare SPICE este declarația de control (control statement). Declarația de control conține un punct în prima coloană și specifică tipul de analiză care trebuie realizat și variabilele de ieșire care trebuie memorate. Pentru verificarea calculului manual al punctului static de funcționare, programului i se cere, prin introducerea liniei .OP să facă analiza de curent continuu a circuitului (calculul punctului static de funcționare - Operating Point, adică valorile potențialelor nodurilor).

Această descriere a circuitului este salvată în fișierul circuitT.cir care este utilizat ca fișier de intrare pentru SPICE. Descrierea circuitului este citită de către SPICE și compilată într-o reprezentare internă. Ecuațiile la noduri, identice cu ecuațiile 1 și 2 sunt stabilite intern în SPICE și rezolvate ca un sistem de ecuații liniare.

SPICE face distincție între câmpurile alfanumerice (name fields) și câmpurile numerice (number fields). Un câmp alfanumeric, de exemplu Q1, RB sau QMOD trebuie să înceapă cu o literă (A-Z) urmată de alte litere sau numere.

Un câmp numeric poate conține un întreg (de exemplu 5 sau -123), un număr în virgulă mobilă (de exemplu 3.14159), un număr în virgulă mobilă scris în notație inginerescă (de exemplu 1E-16 sau 2.65E5), un întreg sau un număr în virgulă mobilă urmat de unul de factorii de scală recunoscuți de SPICE - a se vedea tabelul următor.

Valoare	Simbol
$10^{-15}$	F
$10^{-12}$	P
$10^{-9}$	N
$10^{-6}$	U
$10^{-3}$	M
$25.4 \cdot 10^{-6}$	MIL
$10^3$	K
$10^6$	MEG
$10^9$	G
$10^{12}$	T

Într-un câmp numeric, o literă care urmează imediat fie după un factor de scală, fie după un număr și nu este factor de scală este ignorată. Deci 10, 10V, 10VOLTS și 10HZ reprezintă același număr, iar M, MA, MSEC reprezintă același factor de scală  $10^{-3}$ . Se pot adăuga unitățile de măsură la câmpul numeric, de exemplu 10V, 1PF sau 10UM. Toate aceste specificații sunt

valabile numai atât timp cât unitatea de măsură nu se confundă cu factorul de scală: de exemplu 1F în SPICE înseamnă 10E-15 și nu 1 Farad. Altă greșeală tipică este dată de confuzia între M, MEG și MIL: 20MHZ nu înseamnă 20 de MegaHertz, ci 20 de miliHertz; iar 0.1MIL nu înseamnă 0.1 milimetri, ci 25.4 micrometri.

Descrierea circuitului se compune dintr-un număr de declarații de element ce conțin numele elementului de circuit, nodurile din circuit la care este conectat elementul de circuit și valorile parametrilor care determină caracteristicile electrice ale elementului de circuit. Formatul general al unei declarații de element este:

```
Un_ nume nod1 nod2 <nod3...> <MODEL_ nume> <valoare1...>
```

Primul câmp conține întotdeauna numele elementului. Numele elementului trebuie să înceapă cu o literă care definește tipul elementului, de exemplu R pentru rezistoare, Q pentru tranzistoare bipolare și V pentru surse de tensiune. Cu excepția primei litere, restul numelui elementului poate conține atât caractere cât și numere.

Câmpurile următoare nod1, nod2, <nod3...> reprezintă numerele nodurilor la care este conectat elementul. Pentru specificarea nodurilor este permisă și folosirea numelor. De remarcat că numerotarea nodurilor începe cu 0, nodul 0 fiind nodul de referință (potențialul  $V_0=0$ ). De asemenea începutul și sfârșitul unui element reprezintă un nod, deci în SPICE vor fi mai multe noduri decât la rezolvarea analitică. Un nod este intersecția a două sau a mai multor laturi.

În funcție de tipul lor, elementele de circuit sunt caracterizate printr-o valoare *valoare1* sau un model MODEL\_ nume, urmate de una sau de mai multe valori opționale. Pentru fiecare MODEL\_ nume la care se face referire, specificația de circuit trebuie să conțină o declarație .MODEL. Declarația .MODEL aparține unei alte categorii de declarații, declarațiilor globale (global statements).

În continuare sunt date câteva exemple de declarații de elemente:

- Rezistorul liniar: **R\_ nume N1 N2 valoare**
- Sursa independentă de tensiune (Obs: mai întâi se pune nodul  $N_2$ ):  
**V\_ nume N2 N1 [DC] valoare**
- Sursa independentă de curent: **I\_ nume N1 N2 [DC] valoare**

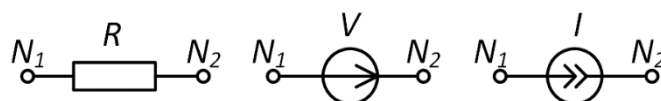


Fig. 1.3 Simbolurile rezistorului, al sursei de tensiune și de curent și notarea nodurilor.

Un alt tip de declarație folosită în fișierul de intrare SPICE este declarația (declarațiile) de control. Declarațiile de control specifică tipurile de analize pe care trebuie să le realizeze SPICE și totodată definește stările inițiale. Toate declarațiile de control încep cu un punct în prima coloană.

În exemplul anterior, calcularea punctului static de funcționare este cerută prin linia .OP și are ca rezultat calcularea potențialelor nodurilor.

În rezumat, orice fișier de intrare SPICE are următoarea structură generală:

```
Declarația de titlu
```

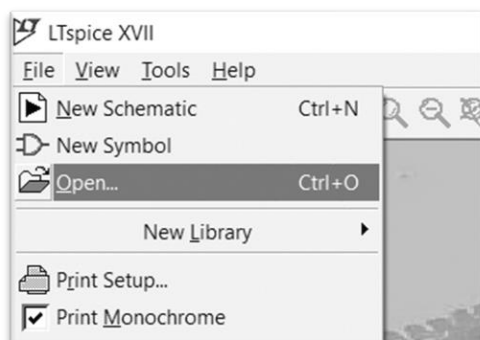
```
*Declarații comentariu  
Declarații de element  
Declarații globale  
Declarații de control  
.END
```

#### 4. Simulări de curent continuu

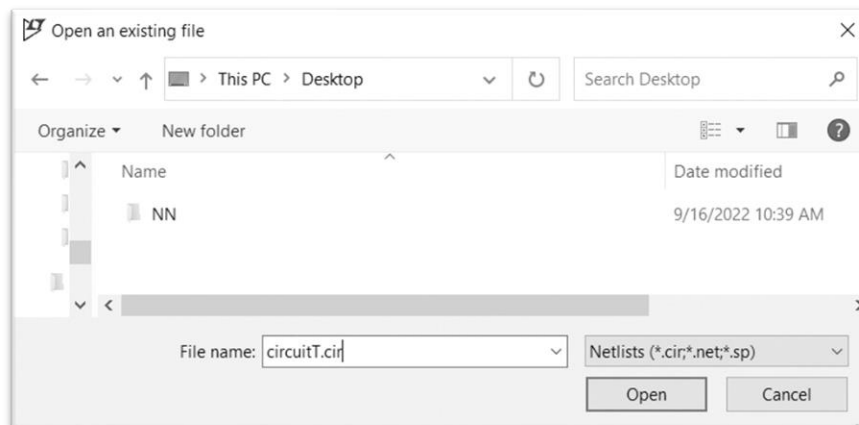
Se vor scrie datele corespunzătoare circuitului în T podit de mai sus.

Se parcurg pașii de mai jos pentru introducerea componentelor schemei:

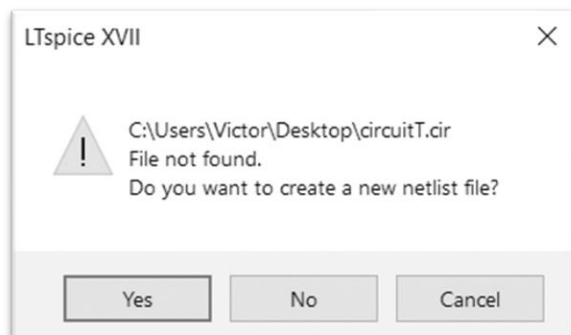
- Se apasă Open:



- Se introduce numele fișierului circuitT.cir:



- Se apasă 'Yes':



- Se scrie fișierul netlist ca în imagine:

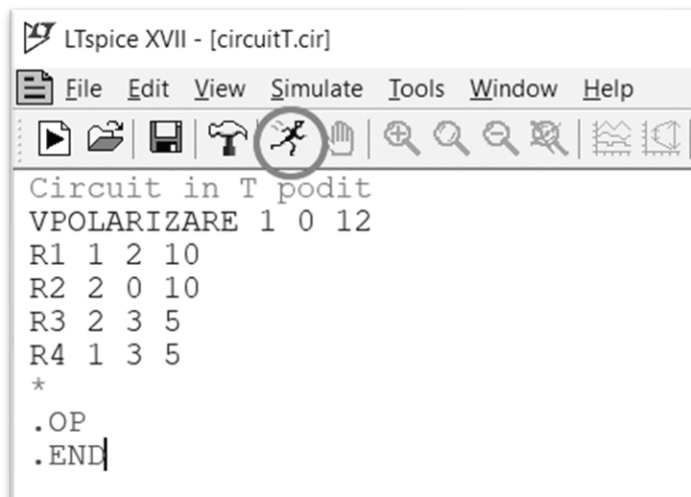


```

LTspice XVII - [circuitT.cir]
File Edit View Simulate Tools Window Help
[Icons]
Circuit in T podit
VPOLARIZARE 1 0 12
R1 1 2 10
R2 2 0 10
R3 2 3 5
R4 1 3 5
*
.OP
.END

```

- Se rulează simularea apăsând pe butonul încercuit:

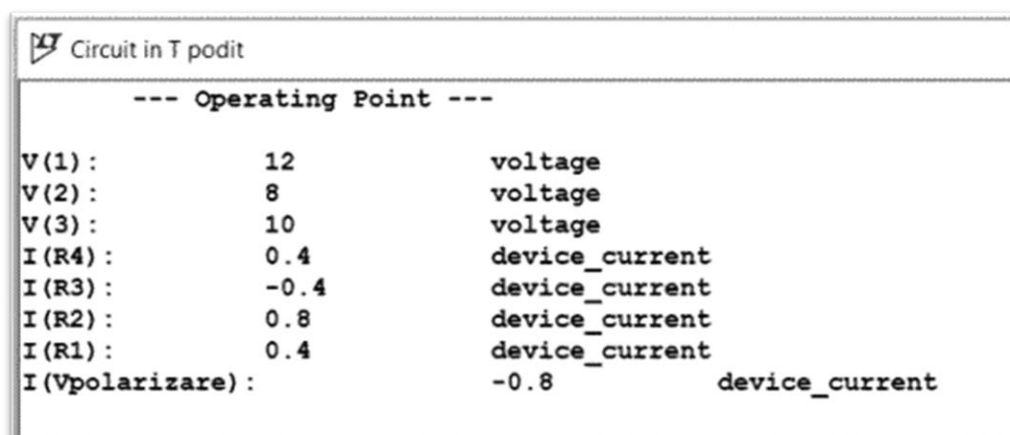


```

LTspice XVII - [circuitT.cir]
File Edit View Simulate Tools Window Help
[Icons]
Circuit in T podit
VPOLARIZARE 1 0 12
R1 1 2 10
R2 2 0 10
R3 2 3 5
R4 1 3 5
*
.OP
.END

```

- Se obține rezultatul



```

Circuit in T podit
--- Operating Point ---
V(1):      12      voltage
V(2):       8      voltage
V(3):      10      voltage
I(R4):      0.4     device_current
I(R3):     -0.4     device_current
I(R2):      0.8     device_current
I(R1):      0.4     device_current
I(Vpolarizare): -0.8 device_current

```



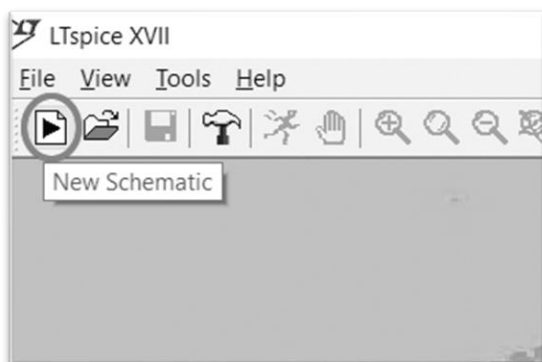
Fișierul este dat mai jos. Se poate verifica identitatea dintre rezultatele obținute în urma simulării și rezultatele obținute prin analiza manuală (potențialele nodurilor).

```
--- Operating Point ---
V(1):  12      voltage
V(2):   8      voltage
V(3):  10      voltage
I(R4):  0.4    device_current
I(R3): -0.4    device_current
I(R2):  0.8    device_current
I(R1):  0.4    device_current
I(Vpolarizare): -0.8 device_current
```

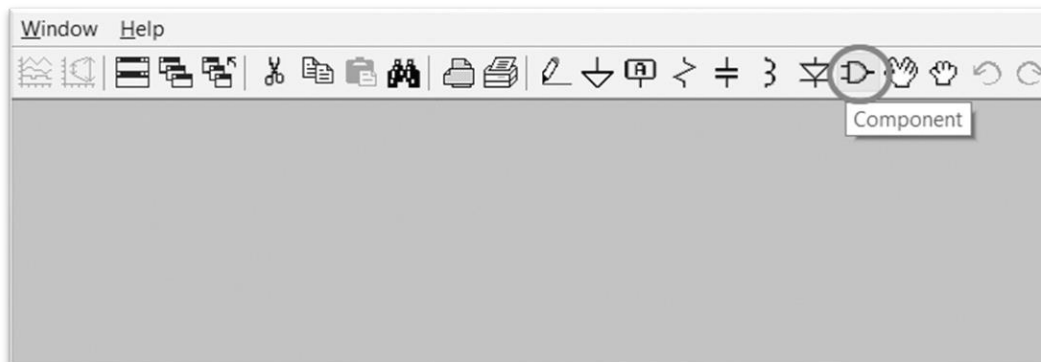
## 5. Introducerea schemei în LTSpice

În continuare vom introduce schema circuitului în T podit (se folosesc simbolurile din standardul american) urmând să analizăm apoi fișierul netlist rezultat.

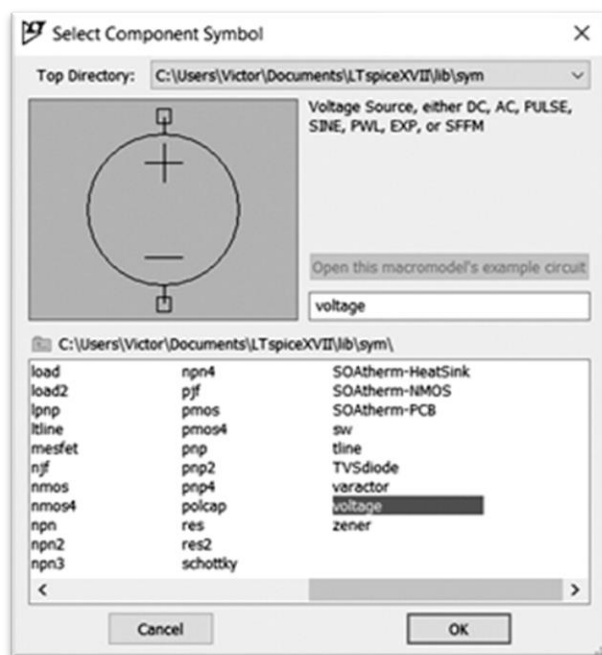
- Se apasă New Schematic:



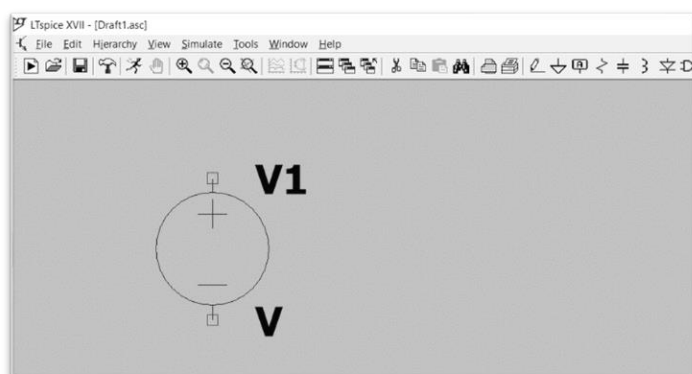
- Se apasă butonul pentru a introduce o componentă:



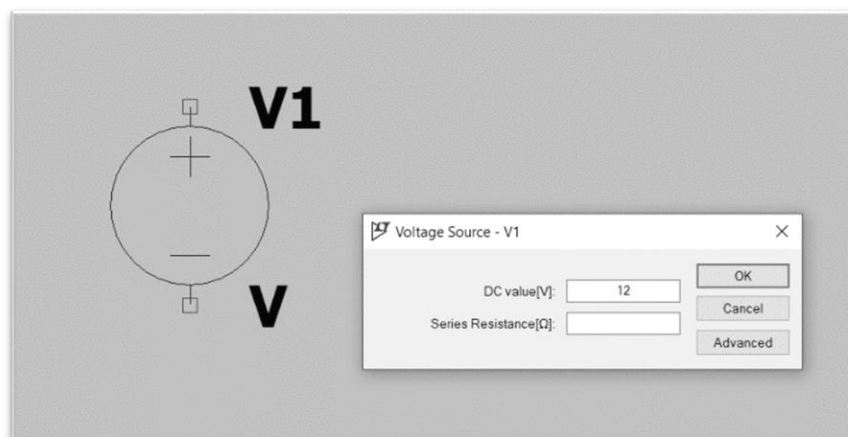
- Se introduce sursa de tensiune:



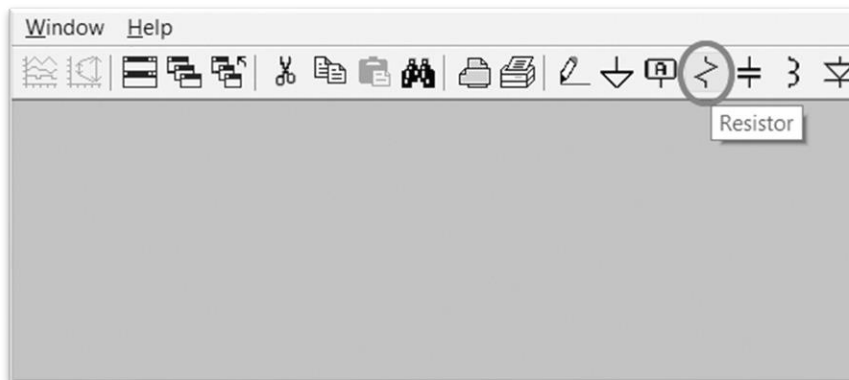
- Se plasează sursa de tensiune și se apasă pe butonul drept de la mouse pentru a încheia:



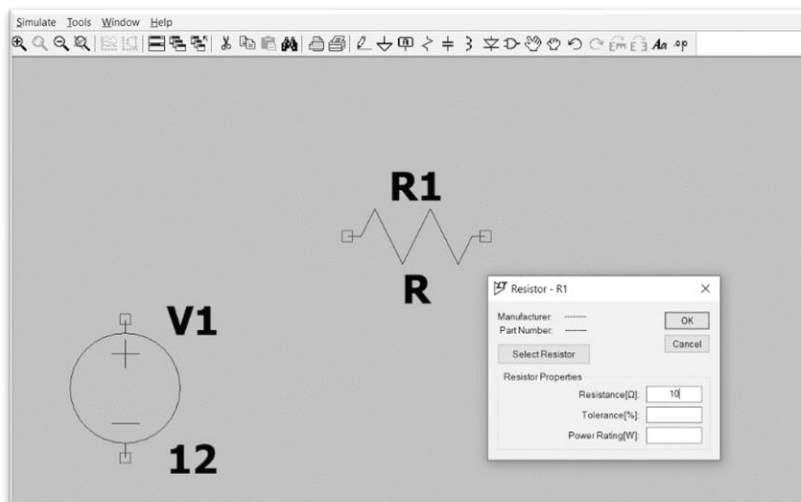
- Se apasă pe butonul drept de la mouse și se introduce valoarea de 12V:



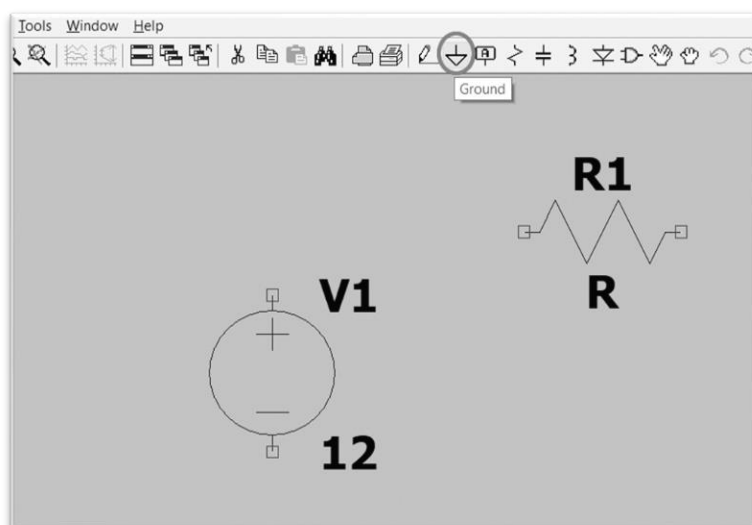
- Se apasă pe butonul corespunzător rezistorului, se apasă CTRL+R pentru a roti componenta, se apasă pe butonul stâng de la mouse pentru a plasa rezistorul și apoi pe butonul drept de la mouse:



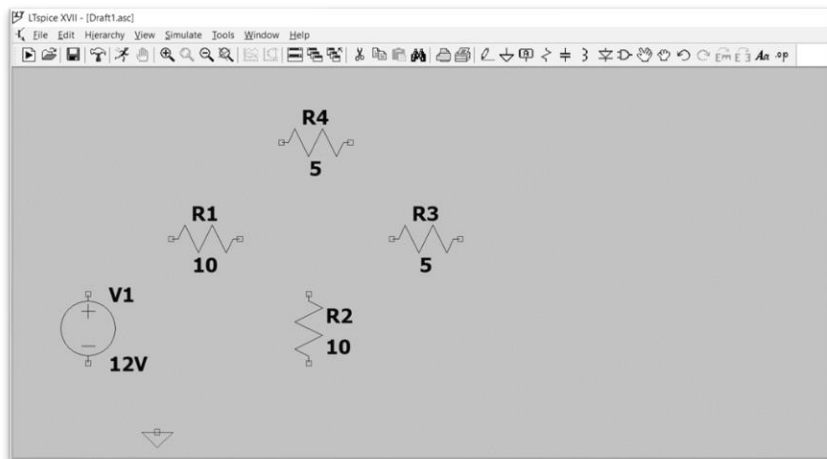
- Se apasă pe butonul drept de la mouse pentru introducerea valorii lui  $R_1$  ( $10\Omega$ ):



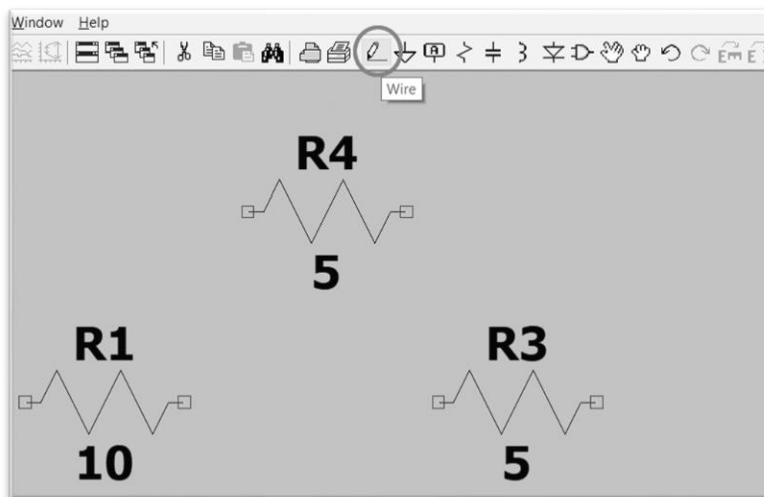
- Se apasă pe butonul corespunzător pentru introducerea nodului de potențial nul (masă):



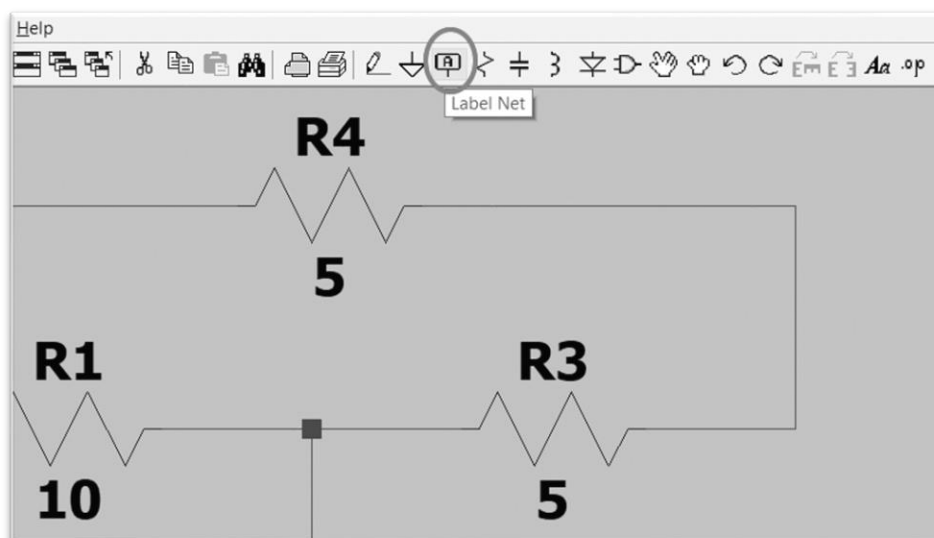
- Se introduc toate componentele schemei (repetând corespunzător pașii de mai sus):



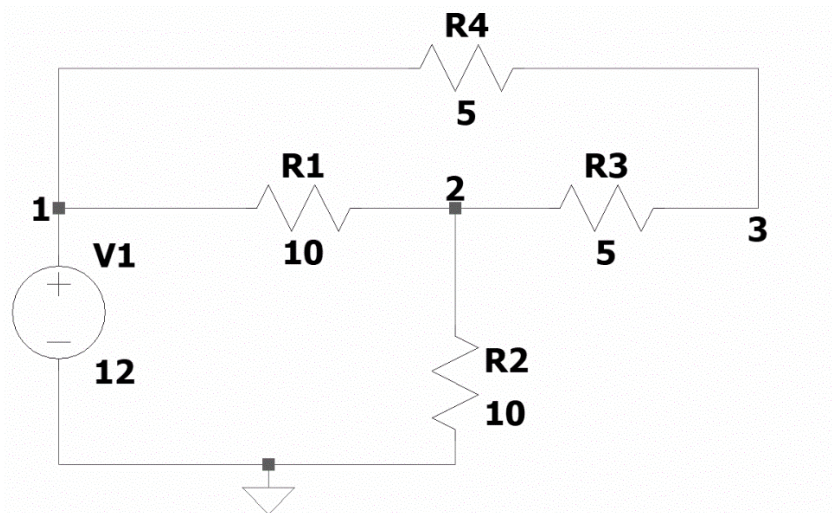
- Se introduc conexiunile folosind butonul încercuit:



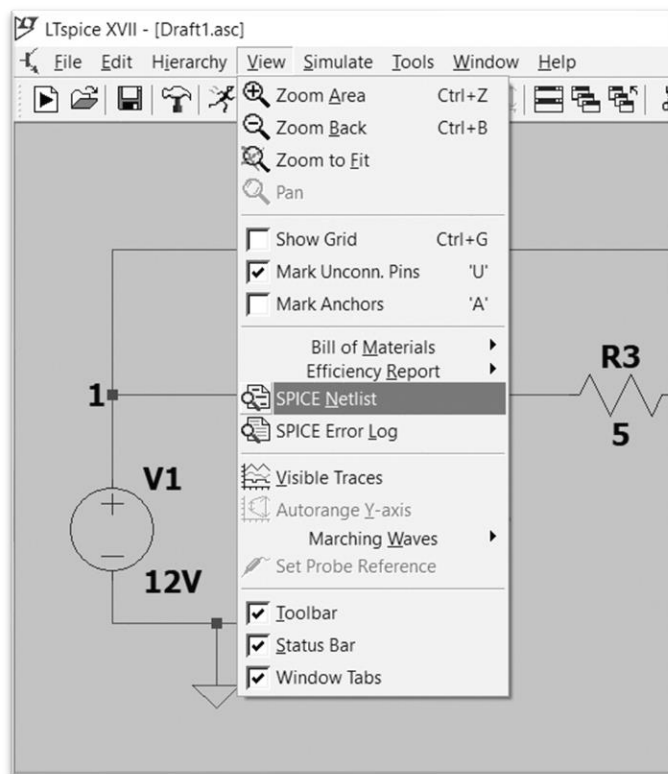
- Se numerotează nodurile folosind butonul Label Net :



- Se obține schema finală:



- Se selectează ca în figură pentru a vedea fișierul netlist:

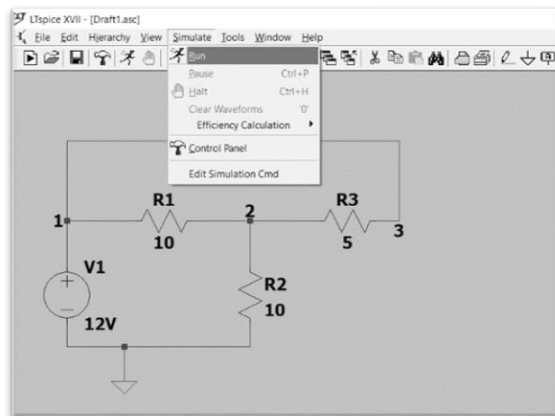


Fișierul netlist va arăta astfel:

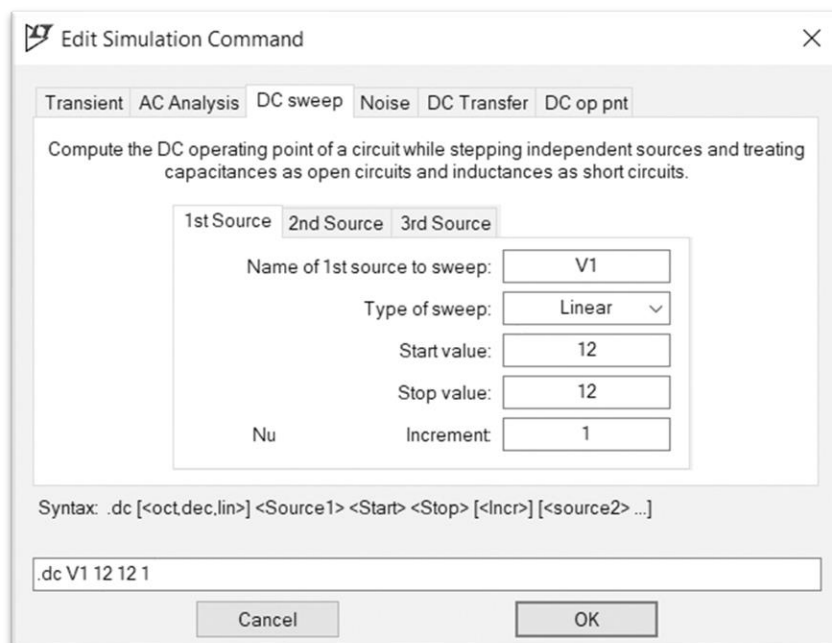
```
* C:\Users\**** \Documents\LTspiceXVII\Draft1.asc
V1 1 0 12V
R1 2 1 10
R2 2 0 10
R3 3 2 5
R4 3 1 5
.backanno
.end
```

Se urmează pașii de mai jos pentru a realiza simularea circuitului:

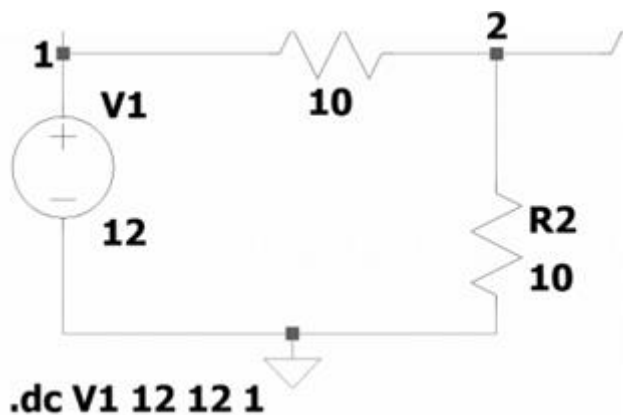
- Se selectează ca în figură pentru începerea simulării de curent continuu:



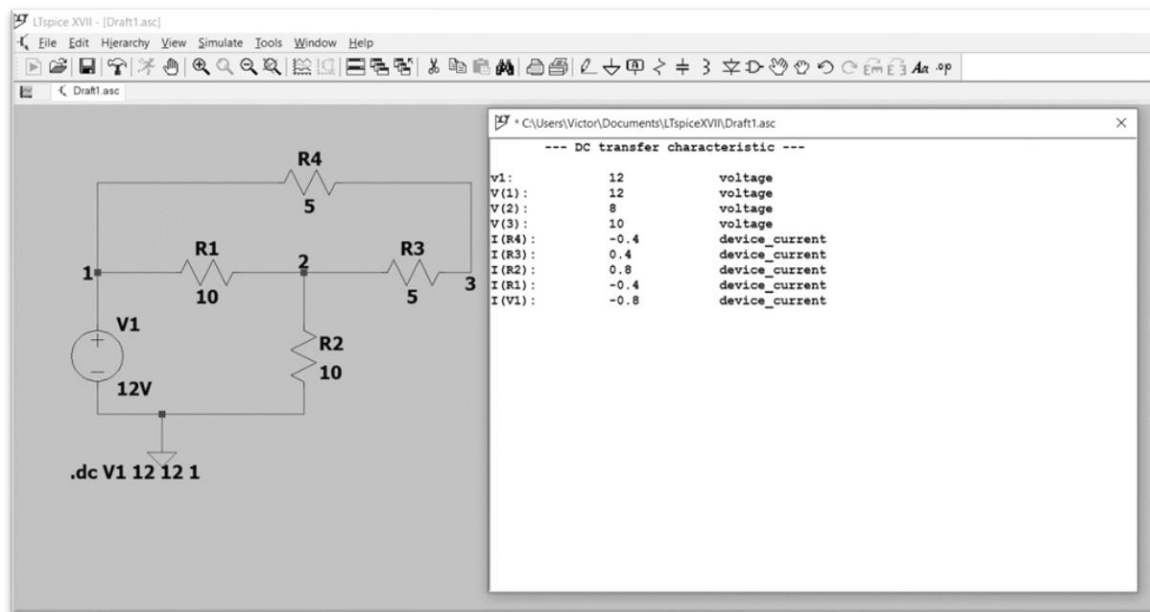
- Se introduc valorile ca în figură:



- Se observă directiva .DC în partea de jos:



- Se observă rezultatele (potențiale și curenți):



În ultimul pas din caseta de mai sus se observă rezultatele simulării.

```

--- DC transfer characteristic ---
v1:      12      voltage
V(1):    12      voltage
V(2):     8      voltage
V(3):    10      voltage
I(R4):   -0.4    device_current
I(R3):    0.4    device_current
I(R2):    0.8    device_current
I(R1):   -0.4    device_current
I(V1):  -0.8    device_current
    
```

Se observă confirmarea rezultatelor obținute analitic.

În continuare se va folosi un alt circuit de curent continuu. Schema lui împreună cu notarea nodurilor este dată în Figura 1.4.

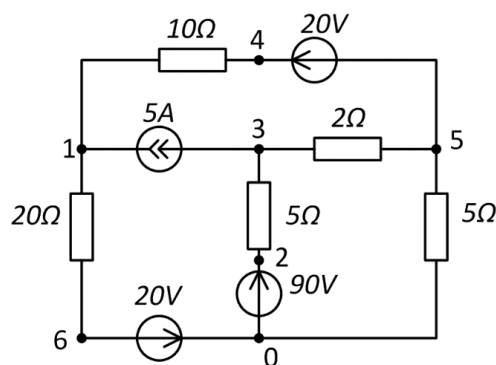


Fig. 1.4. Circuitul de simulat

Pentru a putea afișa alte valori în afară de potențialele nodurilor, trebuie folosită comanda .PRINT. Ea are multe utilizări dar accentul se va pune aici asupra tipăririi tensiunilor și a curenților DC. Comanda .PRINT poate fi repetată în analiză de atâtea ori cât este necesar. Pot fi listate oricâte elemente se doresc.

Totuși trebuie avut în vedere că instrucțiunea .PRINT a fost proiectată pentru a lucra cu o analiza DC sau AC. Aceasta este o metodă de a varia un parametru pe un domeniu de valori astfel încât să avem toate cazurile rezolvate imediat. Adesea nu dorim să facem o analiză pe mai multe valori ale unui parametru. Putem constrânge analiza stabilind domeniul astfel încât să ruleze pe o singură valoare. De obicei o analiza DC este realizată modificând valorile unei surse. Iată care este sintaxa unei analize DC folosind tipul liniar implicit al domeniului:

```
.DC variabila_de_analiza valoare_de_start valoare_finala pas
```

Pentru exemplul nostru alegem sursa de tensiune și stabilim domeniul variabilei la o singură valoare.

```
.DC V1 90 90 1
```

Deoarece valoarea inițială este egală cu valoarea finală, analiza va fi realizată pentru un singur caz, adică pentru  $V_1=90V$ . De remarcat că motivul pentru care includem instrucțiunea .DC este de a activa comanda .PRINT. Comanda .PRINT nu va funcționa decât dacă este activă o comandă .DC sau .AC.

Atunci când se folosește instrucțiunea .DC, aceasta suprapune orice valoare a sursei corespunzătoare din lista de descriere a circuitului.

În plus față de afișarea tensiunilor nodurilor la care se tastează "V" urmat de numărul nodului între paranteze, poate fi afișată și tensiunea între orice pereche de noduri. V(m,n) afișează tensiunea de la nodul m la nodul n. De exemplu:

```
.PRINT DC V(0,3)
```

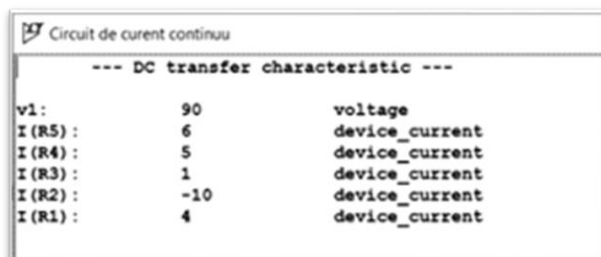
Pentru a afișa curenți, se scrie litera "I" cu numele elementului între paranteze. Curentul afișat este cel care parcurge elementul de la nodul din stânga la cel din dreapta din fișierul .CIR. Dacă se dorește schimbarea semnului curentului afișat, de exemplu pentru un rezistor, trebuie schimbate între ele cele 2 noduri ale rezistorului.

- Se creează fișierul netlist:





- Se observă rezultatele (potențiale și curenți):



Se obține fișierul de mai jos. Se poate verifica identitatea dintre rezultatele obținute în urma simulării și rezultatele obținute prin analiza manuală (potențialele nodurilor).

```

--- DC transfer characteristic ---
v1:      90      voltage
I(R5):   6       device_current
I(R4):   5       device_current
I(R3):   1       device_current
I(R2):  -10      device_current
I(R1):   4       device_current
    
```

## 6. Exerciții

Să se afle potențialele nodurilor și curenții prin laturi pentru următoarele circuite analitic pe hârtie și apoi folosind LTSpice:

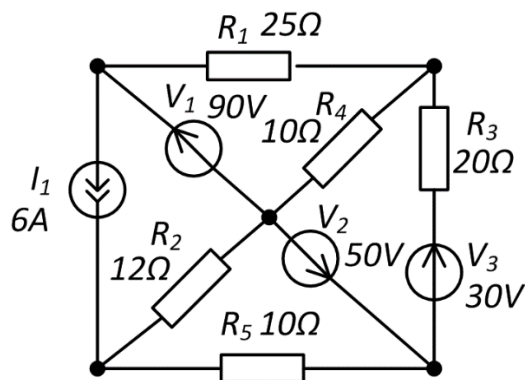


Fig. 1.5. Exercițiul 1

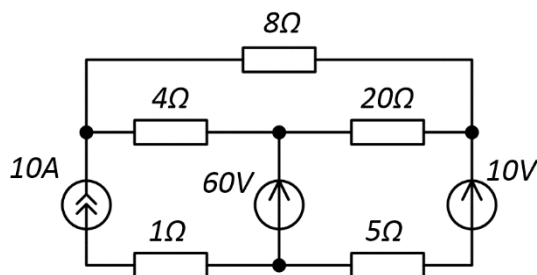


Fig. 1.6. Exercițiul 2

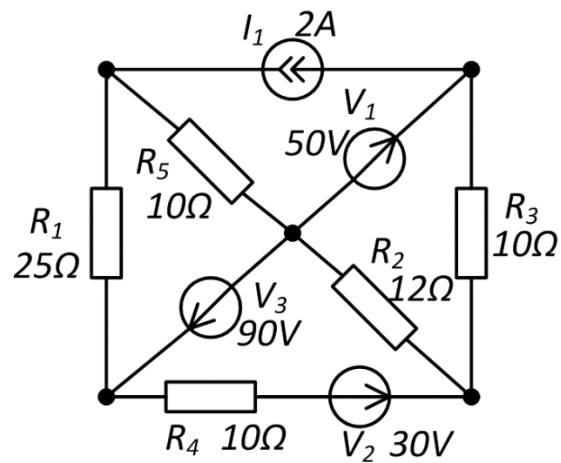


Fig. 1.7. Exercițiul 3

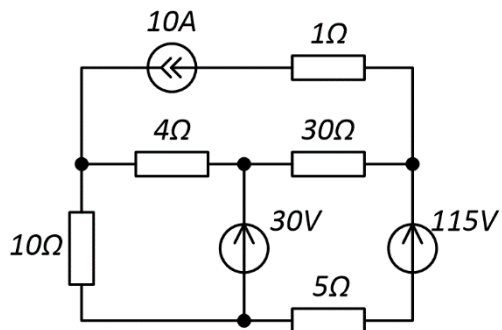


Fig. 1.8. Exercițiul 4

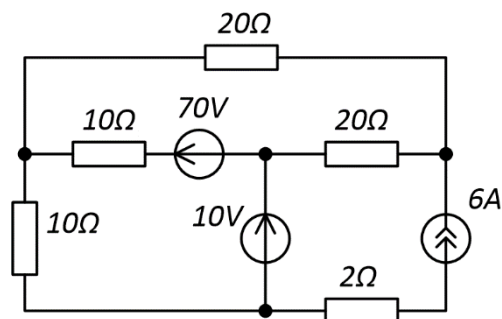


Fig. 1.9. Exercițiul 5

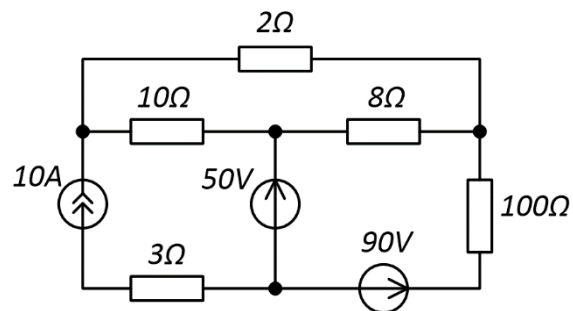


Fig. 1.10. Exercițiul 6

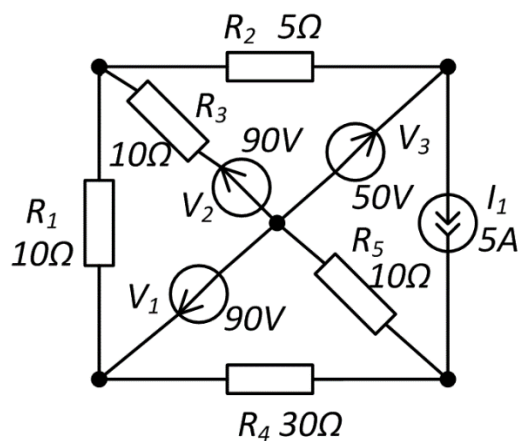


Fig. 1.11. Exercițiul 7

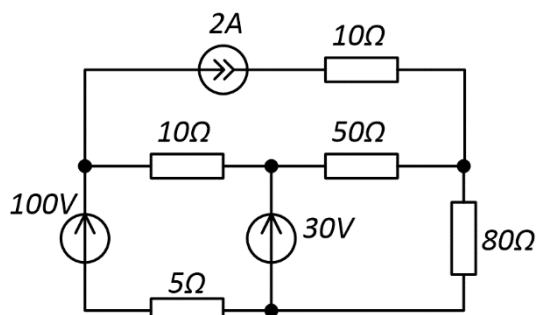


Fig. 1.12. Exercițiul 8

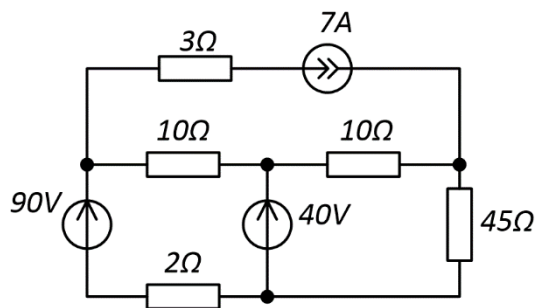


Fig. 1.13. Exercițiul 9

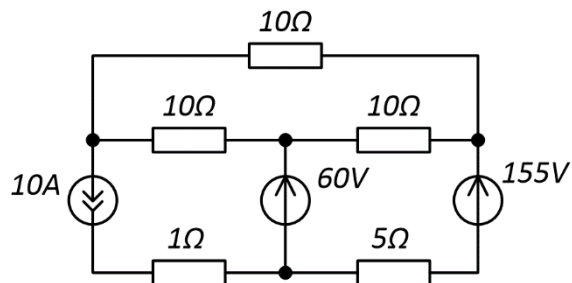


Fig. 1.14. Exercițiul 10

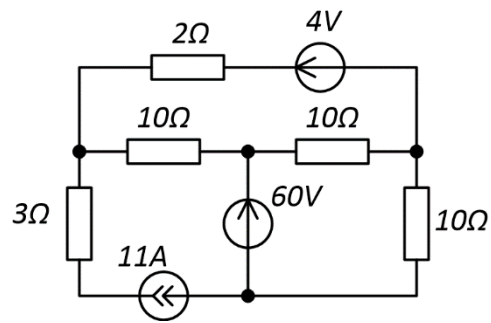


Fig. 1.15. Exercițiul 11

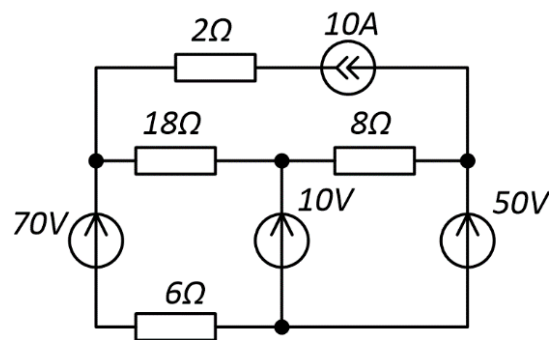


Fig. 1.16. Exercițiul 12