

Rīgas Tehniskā universitāte
Elektronikas un telekomunikāciju fakultāte
Elektronikas pamatu katedra

Datormācība (spekurss)
10. - 11. nodarbība

Vienkāršu elektrisku shēmu modelēšana

Grupas Nr. REBM01

Emīls Vizulis

Studenta apliecības Nr. 191REB136

Saturs

1	Ievads	2
2	Shēmas zīmēšana, jeb GSchem daļa	2
2.1	Komponenšu izvietošana	2
2.2	Komponenšu savienošana	3
2.3	Paramatru vērtību piešķiršana	4
2.4	Shēmas attēla izvadīšana	4
3	Elementu-mezglu faila izveidošana (*.net)	5
4	Shēmas simulācijas veikšana	5

Darba mērķi

- Iemācīties veidot dokumentus, kas labi izskatās
- Iemācīties kompilēt LaTeX dokumentus lokāli
- Nostiprināt zināšanas par Git.

1 Ievads

Šajā nodarbībā tiks veidots LaTeX fails, kurā tiks aprakstītas un ar bildēm papildinātas 9. nodarbībā veiktās darbības.

9. Nodarbībā mēs iemācījāmies ar gEDA programmām veikt vienkāršu shēmu simulāciju.

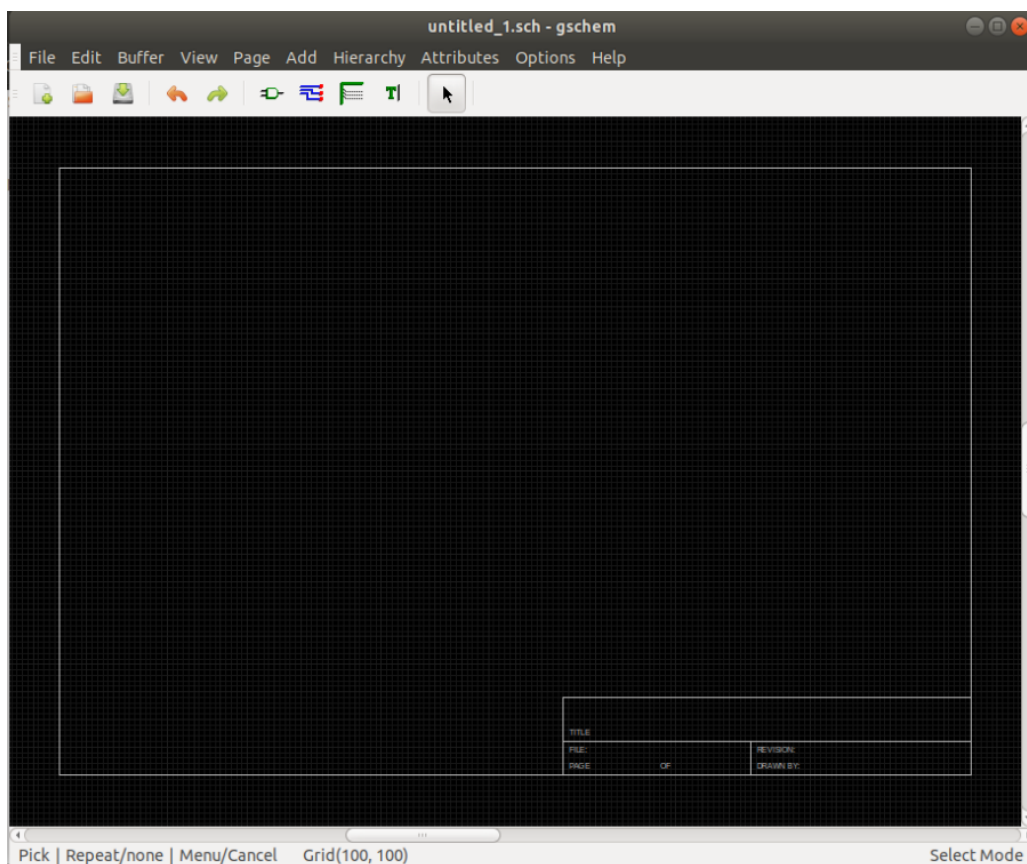
Šīs 9. nodarbībā iegūtās zināšanas mēs izmantosim, lai uzrakstītu aprakstu kā veikt simulāciju, kādas darbības jāveic, lai to izdarītu veiksmīgi.

Shēmas simulācijai veicamās darbības Lai veiktu shēmas simulāciju, tā vispirms ir jauzīmē un jāapstrādā, lai programmās varētu apstrādāt tām piešķirto informāciju. Veicot datu sagatavošanu apstrādei ir jāzin noteiktas lietas, lai darbs notiktu veiksmīgi un nebūtu jāsaskaras ar liekām problēmām.

2 Shēmas zīmēšana, jeb GSchem daļa

Sāksim ar shēmas zīmēšanu, šim darbam mēs izmantosim 'gschem' programmu, to var palaist ierakstot konsolē 'gschem'.

Atvērsies šāds logs:

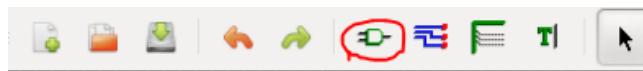


1. attēls. Shēmas zīmēšanas vides logs

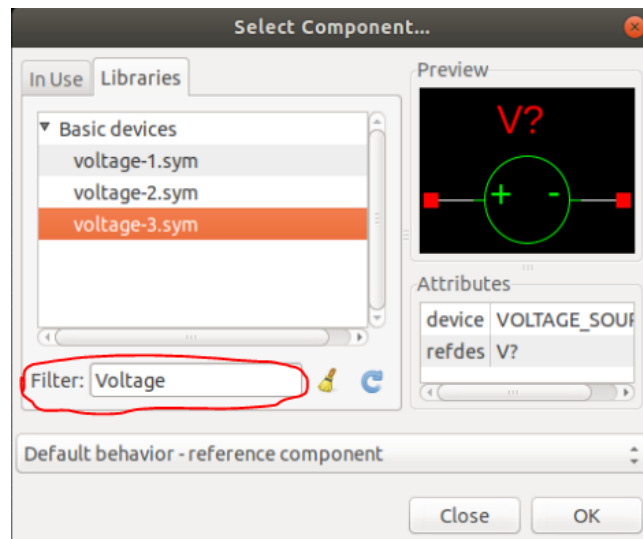
2.1 Komponentu izvietošana

Vispirms veiksīm shēmas zīmēšanu un tikai pēc tam veiksīm vērtību ievadišanu. Lai sāktu zīmēt shēmu, mums tajā ir jāievieto komponentes, tās var atrast zem pogas 'Add component', kas atvērs sarakstu ar daudzām

komponentēm. Lai vieglāk būtu atrast komponenti kura mums ir nepieciešama, var izmantot meklētāju jeb filtru 'Filter:' tajā ierakstot komponentes nosaukumu angļu valodā.



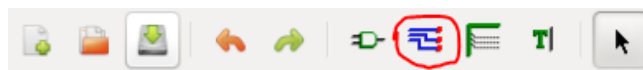
2. attēls. Ikona, lai sāktu izvietot komponents



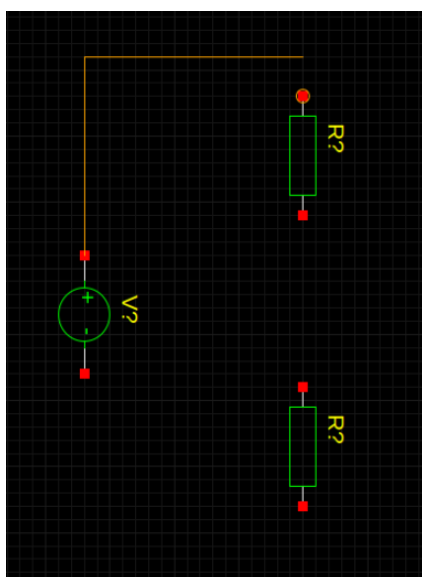
3. attēls. Komponenšu izvēles logs ar meklētāju

2.2 Komponenšu savienošana

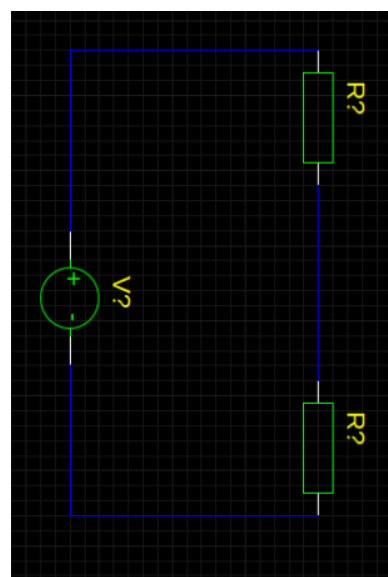
Kad komponentes ir izvietotas vēlamajās vietās, tās jāsavieno ar vadiem, jeb šajā gadījumā tie ir iedomāti savienojumi, kuriem nav pretestības. Vadus sāk zīmēt nospiežot uz pogas 'Add netsmode', kas pārslēgs režīmu lai sāktu zīmēt vadus. Tagad nospiežot jebkurā vietā uz shēmas (sākumpunktā) un pēctam citā (beigu punktā) tiks uzzīmēts vads starp šiem diviem punktiem. Tāpēc, lai savienotu komponentes, kā sākumpunktu izmantosim vienas komponentes kādu no izvadiem un kā beigu punktu kādas citas komponentes kādu no izvadiem.



4. attēls. Ikona, lai sāktu zīmēt vadus



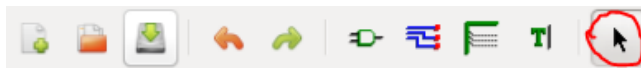
5. attēls. Zīmēšanas sākums



6. attēls. Pabeigta komponenšu savienošana

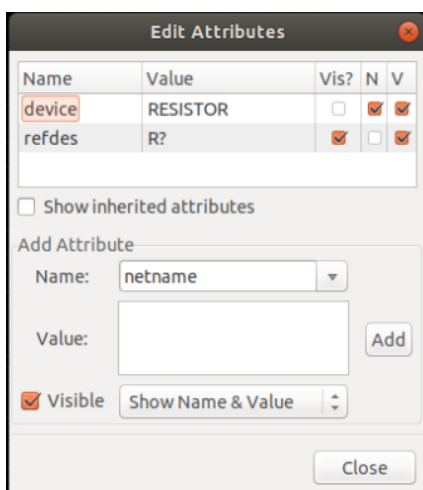
2.3 Parametru vērtību piešķiršana

Kad shēma ir uzzīmēta, ir nepieciešams norādīt elementu vērtības. Lai to varētu sākt darīt vispirms jāizslēdz vadu zīmēšanas režīms, ja to neizdarīsim, mēģinot atlasīt elementu kuram piešķirsim vērtības, mēs turpinātu zīmēt vadus, ko šobrīd mums īsti vairs nevajag. To var izdarīt ar pogu kuras ikona ir tāda pati, vai līdzīga peles kursoram (uzbraucot uz šīs pogas parādīsies **'Select mode'**). Tagad kad esam izslēguši vadu zīmēšanas režīmu, varam sākt vērtību piešķiršanu elementiem.

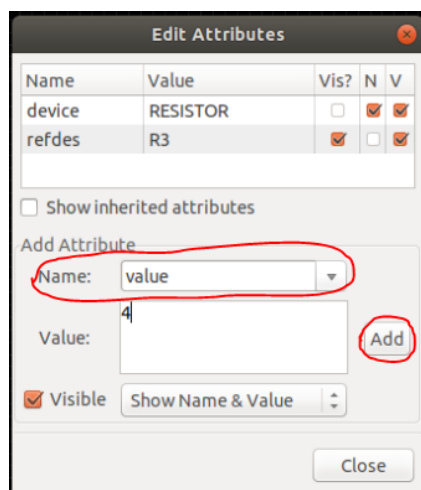


7. attēls. Ikona, lai sāktu izvietot komponentas

Divreiz noklikšķinot uz elementa atvēršies logs kurā būs redzam dažādi tā raksturojošie parametri. Atribūts **'refdes'** norāda šī elementa norādīto nosaukumu, nomainot šī atribūta vērtību mēs elementam piešķirsim nausaukumu, kuru mēs pēctam spēsim atpazīt (Piemēram **'V1'** - avotam, **'R3'** - pretestībai utt.). Diemžēl, mums nav vēl atribūta kurš norāda uz elementa nomināla vērtību, tādēļ to pievienosim. Lai pievienotu nomināla vērtību, atribūtu sarakstā atradīsim atribūtu kurš norāda elementa vērtību **'Value'** un kad esam atlasījuši šo atribūtu iedosim viņam skaitlisko vērtību, kad tas ir izdarīts nospidīsim **'Add'** un būsime elementam piešķīruši nepieciešamās vērtības (Sarežģītākiem elementiem varētu būt arī nepieciešams piešķirt vairāk vērtību). To pašu atkārtosim arī citiem elementiem.



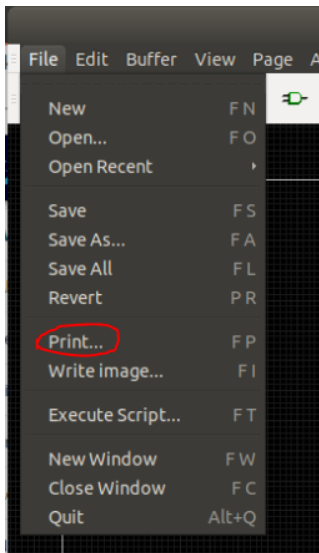
8. attēls. Komponentes atribūti



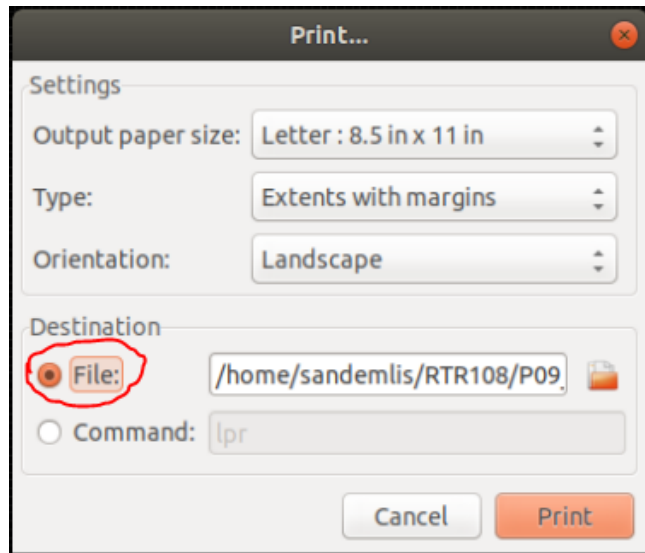
9. attēls. Atribūtu pievienošana

2.4 Shēmas attēla izvadīšana

GSchem piedāvā arī izvadīt shēmas attēlu, kuru varēs apskatīties bez paša GSchem klātbūtnes. Lai to izdarītu programmas loga augšpusē uzspiedīsim uz file, kas atvērs izvēlni. Šajā izvēlnē nospiedīsim **'print...'**. Tik atvērts lodziņš kurā tik piedāvādas dažādas attēla izvadīšanas opcijas. Mums tikai būs nepieciešams atlasīt, lai tiek izvadīts uz failu **'File:'** un nospiežot print, tiks izveidots fails ar shēmas attēlu.



10. attēls. Attēla izvadīšana (Print)



11. attēls. Opcija, lai izvadītu uz failu

3 Elementu-mezglu faila izveidošana (*.net)

Šis fails (saraksts) ir nepieciešams, lai mēs no izveidotās shēmas varētu veikt simulācijas, tas saturēs elementu vērtības un meglus pie kuriem šie elementi ir savienoti. Šīs darbības veikšanai izmantosim komandu 'gnetlist' Tālāk tiks norādīts komandas pielietojšanas piemērs, lai šo darbību varētu veikt:

```
1 gnetlist -g spice -o [fails kur] [fails no kura]
```

"[fails kur]" norāda komandai failu, kurā tiks veikts elementu-mezglu saraksts

"[fails no kura]" norāda shēmas failu, no kura tiks veidots elementu-mezglu saraksts (parasti beidzas ar .net)

Ja shēmā viss tika izdarīts pareizi, tad jauniegūtajā faila saturam jāizskatās līdzīgi šim.

```
* Spice netlister for gnetlist
R2 2 0 7
R1 1 2 3
V1 1 0 13.6
.END
```

12. attēls. Elementu-Mezglu saraksts

4 Shēmas simulācijas veikšana

Lai veiktu shēmas simulāciju ir jāpalaiž NGSpice ierakstot 'ngspice' konsolē Kad NGSpice ir palaists būs jāieraksta dažas komandas kuras tiks izmantotas, lai iegūtu simulācijas rezultātus.

Šīs komandas jāraksta ir šādā secībā, lai veiktu simulāciju

```
1 source [fails]
2 tran [solis] [beigas] [saakums]
```

Komanda 'source' norāda failu, kuram tiks veikta simulācija, šis fails jānorāda "[fails]" vietā. Šajā failā, kuru izmantos, lai veiktu simulāciju, ir jābūt mezlu-elementu sarakstam.

Komanda 'tran' veiks simulāciju laika posmā,

"[solis]" norāda ik pēc cik ilga laika tiks veikta mērījumu fiksēšana,

"[beigas]" norāda beigu laiku, kad beigs veikt mērījumu piefiksēšanu,

un "[sākums]" norāda laiku, kurā sākt veikt mērījumu piefiksēšanu

Norādot laiku, ja netiek norādītas mērvienības, ievadītie cipari tiek uzskatīti kā sekundes, bet ja vēlas norādīt mērīšanas intervālu mazākā diapazonā, piemēram ja ir straujas izmaiņas, būs laiks jānorāda ar mērvienību, piemēram ms, us, utt.

```

ngspice 1 -> tran 1 4
Doing analysis at TEMP = 27.000000 and TNOM = 27.000000

Initial Transient Solution
-----

Node                Voltage
-----
2                   9.52
1                  13.6
v1#branch          -1.36

No. of Data Rows : 59

```

13. attēls. Simulācija

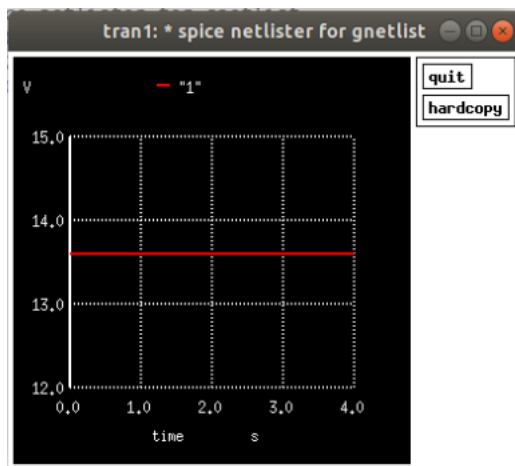
Tālāk norādītās komandas tiek izmantotas, lai izvadītu veikto simulāciju grafikus.

```

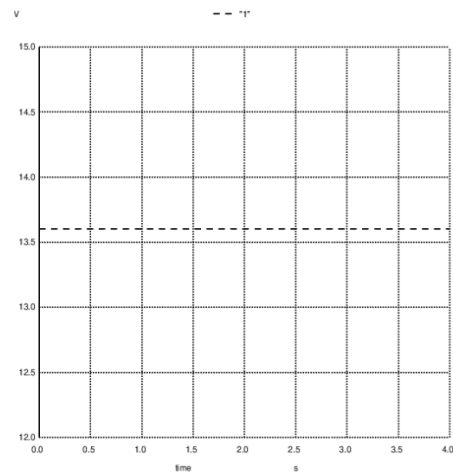
1 plot "[mezgls]"          vai arī      plot "[mezgls1]" "[mezgls2]"
2 hardcopy [fails kur] "[mezgls]"      vai arī      hardcopy [fails kur] "[mezgls1]" "[mezgls2]"

```

Komanda plot parādīs uz ekrāna grafiku, bet komanda hardcopy grafiku izvadīs failā.



14. attēls. Plot izvadītais grafiks



15. attēls. Hardcopy izvadītais grafiks