

**ZÁPADOČESKÁ UNIVERZITA V PLZNI
FAKULTA ELEKTROTECHNICKÁ**

Katedra aplikované elektrotechniky a telekomunikací

DIPLOMOVÁ PRÁCE

Zjednodušená simulace analogové části ECU a vozidla

ZADÁNÍ DIPLOMOVÉ PRÁCE
(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Vojtěch JEŽEK**
Osobní číslo: **E15N0001P**
Studijní program: **N2612 Elektrotechnika a informatika**
Studijní obor: **Elektronika a aplikovaná informatika**
Název tématu: **Zjednodušená simulace analogové části zapojení ECU
a vozidla**
Zadávací katedra: **Katedra aplikované elektroniky a telekomunikací**

Z á s a d y p r o v y p r a c o v á n í :

1. Navrhněte obecné simulační nástroje (funkcí/tříd/modulů).
2. Implementujte simulace ECU s použitím těchto nástrojů.
3. Integrujte tyto simulace do cílového projektu.
4. Porovnejte výsledky simulace a chování cílového ECU.

Rozsah grafických prací: **podle doporučení vedoucího**

Rozsah kvalifikační práce: **40 - 60 stran**

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:

Student si vhodnou literaturu vyhledá v dostupných pramenech podle doporučení vedoucího práce.

Vedoucí diplomové práce:

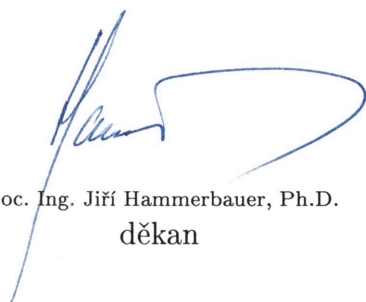
Ing. Kamil Kosturik, Ph.D.

Katedra aplikované elektroniky a telekomunikací

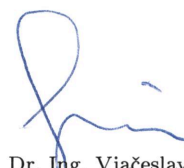
Datum zadání diplomové práce: **10. října 2017**

Termín odevzdání diplomové práce: **24. května 2018**

Doc. Ing. Jiří Hammerbauer, Ph.D.
děkan



Doc. Dr. Ing. Vjačeslav Georgiev
vedoucí katedry



V Plzni dne 10. října 2017

Abstrakt

Cílem této diplomové práce bylo navrhnout a vytvořit simulátor analogových elektronických obvodů, které jsou přítomny v řídicí jednotce automobilu. V první části jsou popsány rozdíly mezi digitální a analogovou simulací. Po této části následuje teoretický rozbor metod pro analýzu elektronických obvodů a metod pro jejich výpočet. V poslední části je rozebrána praktická realizace části simulátoru.

Klíčová slova

Simulace, analogové simulátory, analýza obvodu, metoda razítek, numerické metody, lineární rovnice, nelineární rovnice, diferenciální rovnice

Abstract

The aim of this diploma thesis was to design and create simulator of analog electronic circuits, which are present in the automotive control unit. First part describes the differences between digital and analog simulation. Next part is theoretical analysis of methods for analysis of electronic circuits and methods for their calculation. Last part describes the practical implementation of the part of the simulator.

Key words

Simulation, analog simulators, circuit analysis, modified nodal analysis, numerical methods, linear equations, nonlinear equations, differential equations

Prohlášení

Prohlašuji, že jsem tuto diplomovou/bakalářskou práci vypracoval samostatně, s použitím odborné literatury a pramenů uvedených v seznamu, který je součástí této diplomové práce.

Dále prohlašuji, že veškerý software, použitý při řešení této bakalářské/diplomové práce, je legální.

.....
podpis

V Plzni dne 24.5.2018

Vojtěch Ježek

Obsah

OBSAH	6
SEZNAM SYMBOLŮ A ZKRATEK	8
1 ÚVOD	9
2 SIMULACE ELEKTRICKÝCH OBVODŮ	10
2.1 SIMULACE DIGITÁLNÍCH OBVODŮ.....	11
2.1.1 Jazyk VHDL.....	11
2.1.2 Jazyk Verilog HDL.....	12
2.2 ANALGOVÁ SIMULACE.....	13
3 ANALÝZA ELEKTRICKÉHO OBVODU	15
3.1 TOPOLOGIE OBVODU.....	15
3.2 ZÁKLADNÍ VELIČINY.....	16
3.2.1 Elektrický náboj.....	16
3.2.2 Elektrický proud.....	17
3.2.3 Elektrické napětí.....	17
3.3 PRVKY OBVODU A JEJICH CHOVÁNÍ.....	18
3.3.1 Zdroje napětí.....	19
3.3.2 Zdroj proudu.....	19
3.3.3 Rezistor.....	20
3.3.4 Kondenzátor.....	20
3.3.5 Cívka.....	21
3.3.6 Dioda.....	22
3.3.7 Tranzistor.....	23
3.3.8 Operační zesilovač.....	25
3.3.9 Shrnutí obvodových prvků.....	26
3.4 METODY PRO ANALÝZU ELEKTRICKÉHO OBVODU.....	28
3.4.1 Kirchhoffovy zákony.....	28
3.4.2 Metoda smyčkových proudů.....	30
3.4.3 Metoda uzlových napětí.....	31
3.4.4 Metoda rážitek.....	32
3.5 TYPY SIMULACÍ ELEKTRICKÉHO OBVODU.....	34
3.5.1 DC analýza.....	35
3.5.2 AC analýza.....	35
3.5.3 Tranzientní analýza.....	35
4 MATEMATICKÉ METODY PRO VÝPOČET ELEKTRICKÝCH OBVODŮ	36
4.1 METODY PRO VÝPOČET LINEÁRNÍCH OBVODŮ.....	36
4.1.1 Gaussova eliminační metoda.....	36
4.1.2 Gauss metoda se zpětným chodem.....	38
4.1.3 Inverzní matice.....	39
4.2 METODA PRO VÝPOČET NELINEÁRNÍCH OBVODŮ.....	40
4.2.1 Newtonova metoda pro řešení nelineární rovnice.....	40
4.2.2 Newtonova metoda pro řešení soustav nelineárních rovnic.....	45
4.3 METODY PRO VÝPOČET OBYČEJNÝCH DIFERENCIÁLNÍCH ROVNIC.....	46
4.3.1 Eulerova metoda.....	47
5 IMPLEMENTACE SIMULÁTORU	48
5.1 ZPRACOVÁNÍ NETLISTU.....	48
5.1.1 Struktura netlistu.....	49
5.1.2 Algoritmus zpracování netlistu.....	50
5.1.3 Příkazy netlistu.....	51

5.1.4 Obvodové prvky netlistu.....	51
5.2 IMPLEMENTACE OBVODOVÝCH PRVKŮ.....	52
5.2.1 Vytváření nové součástky.....	53
5.2.2 Postup pro přidání nové součástky do simulátoru.....	54
5.3 VÝPOČTY OBVODŮ.....	56
6 ZÁVĚR.....	57
SEZNAM LITERATURY A INFORMAČNÍCH ZDROJŮ.....	58

Seznam symbolů a zkratek

<i>G</i>	Matice
<i>i</i>	Vektor
<i>k</i>	Boltzmannova konstanta = $1,381E^{-23}$ [eV/K]
<i>q</i>	Náboj jednoho elektronu = $1,602E^{-19}$ [C]
<i>CAD</i>	Computer aided design – počítačem podporované navrhování
<i>ECU</i>	Electronic control unit – elektronicky řízená jednotka.
<i>FFT</i>	Fast Fourier transform – rychlá Fourierova transformace
<i>Qucs</i>	Quite universal circuit simulator – Celkem univerzální simulátor obvodů
<i>SPICE</i>	Simulation program with integrated circuit emphasis - Simulační program hlavně pro integrované obvody
<i>VHDL</i>	VHSIC hardware description language - jazyk pro popis VHSIC
<i>VHSIC</i>	Very high speed integrated circuit – velmi rychlé integrované obvody

1 Úvod

Tato diplomová práce se zabývá návrhem obecného simulačního nástroje pro výpočet elektrických obvodů, resp. ECU. Cílem je vytvoření funkcí a tříd v jazyku C/C++, které by měly být následně integrovány do cílového projektu, a výsledky by měly být porovnány s chováním reálného elektrického obvodu.

Jak již bylo řečeno v předchozím odstavci tak cílem, této práce je vytvoření simulátoru, který dokáže simulovat elektrické obvody, které jsou přítomny v jednotce automobilu, nikoliv simulovat chování mikrokontroléru přítomného v jednotce nebo chování obvodů mimo tuto jednotku.

Mezi součástky, které je nutno nasimulovat, aby bylo možné daný software implementovat do cílového projektu, patří rezistory, kondenzátory, diody, tranzistory a operační zesilovače.

Obvody jednotky mají za úkol přenášet a také zpracovávat analogové popř. digitální signály z výstupu mikrokontroléru na výstupy jednotky a opačně ze vstupů jednotky zasílat a zpracovávat signály do vstupu mikrokontroléru. Těmito daty mohou být například signály pro ovládání jednotlivých ventilů, hodnoty z teplotních čidel, hodnoty z čidel otáček, hodnoty z čidel tlaku atp.

Díky této simulaci by mělo být možné napojit stávající systém simulace mikrokontroléru se simulací zbytku vozidla.

2 Simulace elektrických obvodů

Samotný pojem simulace znamená nahrazení nějaké skutečné věci, stavu nebo procesu pomocí modelu, který je napodobuje. Při počáteční analýze je vhodné uvážit, jak složitý model bude při simulaci potřebný. Složitější modely přinášejí lepší napodobení skutečného objektu, nicméně, sebou přinášejí problém ve složitosti vytvoření takového modelu.

Model může mít podobu například prototypu. Výroba prototypu je však nákladná záležitost a většinou se v prvotní části vývoje nahrazuje matematickými či fyzikálními modely, které plní podobnou funkci. Tyto modely jsou následně zpracovávány pomocí počítačů.

Tento typ simulace tvoří velice důležitou součást prvotního návrhu systému a díky tomu jsou většinou již integrovány do nejrůznějších CAD či jiných simulačních nástrojů.

Výsledkem těchto simulací je buď obraz o tom, jak se bude reálný systém chovat ve skutečnosti, nebo se tímto získávají data o tom, jak se může například změnit nějaká část systému tak, aby se nezměnila funkcionality, ale aby bylo možné snížit finální cenu.

Důvody pro využití simulace, jak již bylo řečeno, jsou hlavně nižší cena simulace oproti nutnosti vytváření drahých prototypů zařízení a také snížení časové náročnosti na vývoj.[1][2]

Pro obor elektrotechniky můžeme simulace rozdělit na simulace silnoproudých zařízení, jako jsou vedení vysokého napětí, konstrukce točivých strojů, transformátorů atp., a na slaboproudé, kdy se simulacemi řeší funkčnost a parametry výsledného obvodu. Další rozdělení slaboproudých obvodů je na digitální a analogové obvody, které budou dále probrány.

2.1 Simulace digitálních obvodů

Při digitální simulaci elektrických obvodů můžeme uvažovat, že vstupy i výstupy jednotlivých komponent mohou nabývat pouze logických hodnot 0 a 1. Toto zjednodušení pomáhá ve značném zjednodušení takovéto simulace. Důležitým výsledkem u takového druhu simulace však není jen to, jak se daný obvod chová s ohledem na vstupní a výstupní logické úrovně, ale lze i zjistit zpoždění signálu na jednotlivých vodičích a logických obvodech a tím předcházet hazardním stavům nebo celkové nestabilitě systému. K tomuto typu analýzy je však nutné znát informaci o tom, na které cílové platformě bude daný program pracovat.

Typickými zástupci programů pro simulaci digitálních obvodů jsou dnes komerční programy od firem Xilinx, Mentor Graphics nebo Synopsys. Vstup pro tyto programy je tvořen popisovacími jazyky VHDL nebo Verilog HDL. [3]

2.1.1 Jazyk VHDL

V evropských zemích se nejčastěji pro popis a simulaci digitálních obvodů používá právě jazyk VHDL. Tento jazyk byl vytvořen v roce 1983 firmami jako IBM, Intermetrics či Texas Instruments a od svého vzniku byl několikrát modifikován. Strukturně vychází z programovacího jazyku ADA, využívaného americkým ministrem obrany. Pro popis a testování jednotlivých bloků číslicového systému v tomto jazyce je jasně daná struktura.

```
library ieee;  
  
entity NameOfEntity is  
    ...  
end;  
  
architecture NameOfArchitecture of NameOfEntity is  
    ...  
end;
```


V první části popisu se definují použité knihovny. Jedná se o knihovny pracující například s datovými typy (například int, float), popřípadě knihovny s matematickými operacemi. V části s názvem entity se definují porty a generické porty. Tyto porty se dále využívají ke spojení s jinými bloky, které nakonec tvoří celý obraz výsledného obvodu. Poslední částí je část s názvem architecture. V této části je definováno chování daného bloku. Takto vytvořené bloky se podrobují syntéze, která kontroluje jak syntaxi vytvořeného programu tak možnost jeho implementace na cílovou platformu.

Pro testování a simulaci těchto bloků se vytváří testovací prostředí nazývané testbench. Tento program je strukturou velice podobný normálnímu prostředí. Jelikož však není potřeba testbench propojovat s ostatními bloky a tudíž nejsou potřeba žádné porty, vynechává se obsah části entity. Další změnou v tomto prostředí je to, že daný kód není třeba syntetizovat pro cílové zařízení a je možné zde používat konstrukce, které by byli na reálný hardware nemožné syntetizovat. [3]

Simulace takových obvodů probíhá tak, že se již vytvořená komponenta připojí přes své porty do testovacího prostředí, kde se simuluje chování testované komponenty na jednotlivé budící signály.

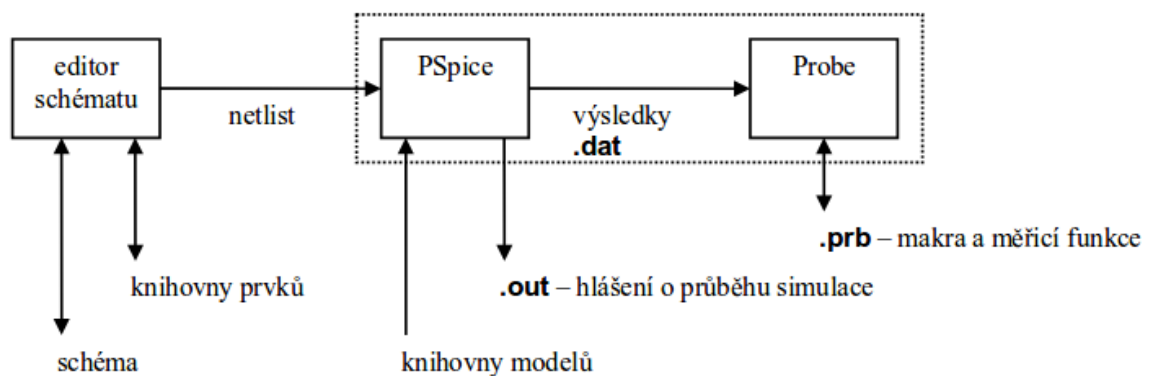
2.1.2 Jazyk Verilog HDL

Jazyk Verilog stejně jako VHDL slouží k popisu digitálních elektronických systémů. Tento jazyk však strukturně spíše připomíná jazyky C/C++ a je spíše používán na americkém kontinentě. Výslednou funkcí je naprosto totožný s jazykem VHDL a je možné ve výsledném programu používat a testovat jak bloky napsané ve VHDL tak v jazyce Verilog.

2.2 Analogová simulace

Dalším typem simulace elektrických obvodů, jak již bylo zmíněno na začátku této kapitoly, je simulace analogová. Na rozdíl od digitální simulace zde nejsou simulovány pouze logické úrovně, ale simulují se zde hodnoty spojité, jako jsou například napětí na jednotlivých rezistorech, napětí na kondenzátorech, proudy cívkami atp. Hlavním posláním této simulace je vytvoření modelů jednotlivých elektronických prvků, jako jsou právě rezistory, cívky, kondenzátory, diody, atp. a simulovat jejich chování v elektrickém obvodu.

Většina dnešních simulátorů až na některé výjimky jsou založeny na stejném principu jako program SPICE, vytvořeném již v 70. letech 20. století na univerzitě v Berkley. Nejčastěji zmiňovaným programem je program PSpice od firmy OrCad. Základní schéma tohoto simulátoru je na obrázku 1.



Obr 1: Základní schéma analogového simulátoru PSpice (převzato z [12])

Moderní analogové simulátory obsahují nástroj, kterým lze kreslit simulovaný obvod. Tímto nástrojem je na obrázku 1 editor schématu. Z takto vytvořeného schématu je vytvořen netlist. Tento soubor obsahuje strukturu daného obvodu se všemi jeho prvky a také nastavení analýz, které budou na daném obvodu simulovány.

* první řádek souboru je ignorován *

.lib [název souboru] ; soubor s připojenými knihovnamí

R_R1 1 2 4k ; rezistor velikosti 4000 ohmů, připojený k uzlu 1 a 2

R_R2 2 0 4k ; rezistor velikosti 4000 ohmů, připojený k uzlu 2 a 0

C_C 2 0 2n ; kondenzátor o velikost 2nF připojený k uzlu 2 a 0

U_U0 0 1 5V ; zdroj konstantního napětí 5 voltů

.op ; výpočet pracovního bodu

.probe V(2) ; na výstup je posláno napětí z uzlu 2

.tran ; nastavení parametrů tranzientní analýzy

.end ; konec souboru

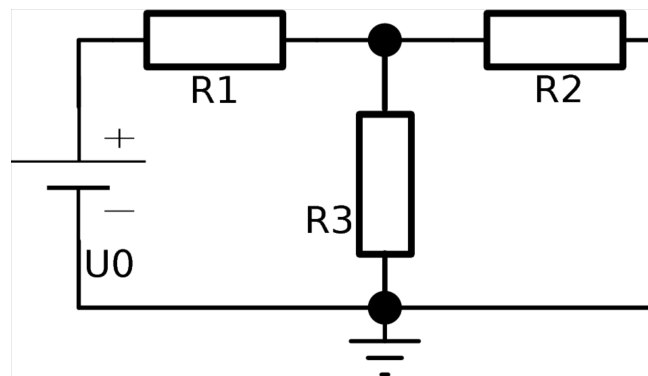
V následujícím textu jsou podrobně vysvětleny funkce jednotlivých řádků. První řádek netlistu je ignorován a je vhodné ho doplnit například o název simulovaného obvodu. Řádky obsahující na svém začátku „lib“ slouží k připojování knihoven z externího souboru. Následuje blok s popisem jednotlivých součástek. Každá součástka má svůj specifický formát a je nutné se podívat do manuálu. Velikosti parametrů mohou být zadávány třemi způsoby. Prvním způsobem je zadání čísel tradičním zápisem čísla (například 0.001, 1000, 1000000). Druhá možnost je zápis čísla pomocí exponentů (například 1E-3, 1E3, 1E6) a třetí možností je zápis čísla pomocí standardních přípon (například 1m, 1k, 1MEG). Příkaz „op“ slouží pro nastavení výpočtu linearizovaného pracovního bodu u všech nelineárních součástek a uložení daných hodnot do souboru. Příkaz „probe“ definuje jaké napětí, popř. proudy, budou zaneseny do výstupního grafu. Předposledním příkazem „tran“ se definuje, že se na daném obvodu má provádět tranzientní analýza. Seznam základních analýz je uveden v kapitole 3.5.

Další částí takového simulátoru je samotné výpočetní jádro pro simulaci (na obrázku 1 pod názvem PSpice). Toto jádro získává údaje o obvodu pomocí netlistu. Díky připojeným knihovnam si načte příslušné prvky obvodu a vytvoří jejich matematické modely, které vloží do výpočetní matice. Výsledky výpočtů pracovních bodů a jednotlivých napětí a proudů jsou ukládány do souborů.

O vykreslení dat se dle obrázku 1 stará blok s názvem Probe. Jeho hlavní funkcí je vykreslovat napětí na jednotlivých uzlech popř. proudy jednotlivými větvemi obvodu podobně jako osciloskop. Dalšími funkcemi tohoto bloku může být například převedení časových dějů do frekvenční oblasti pomocí FFT.[14]

3 Analýza elektrického obvodu

Jako elektrický obvod lze definovat idealizovaný model elektrického či elektronického zařízení, kde jsou znázorněny vazby mezi jednotlivými ideálními obvodovými prvky. K pochopení dějů, jako jsou například přenos, přeměna či akumulace energie, které jsou důležité při simulaci elektrických obvodů, je nutné znát základní pojmy, základní fyzikální veličiny a chování základních elektrických prvků. Příklad takového obvodu je možné vidět na obrázku 2.



Obr 2: Příklad jednoduchého elektrického obvodu

3.1 Topologie obvodu

Pochopení základních pojmů z topologie obvodu je nedílnou součástí k pochopení této kapitoly.

- **Uzel** - Uzel je místo v elektrickém obvodu, kde se střetávají alespoň dva vodiče. Na schématech bude značeno velkými písmeny (např. A, B, C,...)
- **Větev** - Větví obvodu je myšlena ta část obvodu, která spojuje dva uzly.
- **Smyčka** - Smyčka obvodu je libovolný uzavřený okruh v elektrickém obvodu. Na schématech bude značeno jako S a číslo smyčky (např. S1, S2, ...).

3.2 Základní veličiny

V této kapitole jsou probrány definice základních elektrických veličin v elektrickém obvodu. Mezi základní veličiny, které zde budou zmíněny, patří elektrický náboj, elektrický proud a elektrické napětí.

Při zapisování jednotlivých elektrických veličin se v dalším textu bude používat malé písmeno s označením proměnné pro veličiny funkčně závislé na dané proměnné (například $i(t)$... funkce proudu závislá na čase t) a velkými písmeny pro veličiny konstantní (například U ... konstantní napětí U).

3.2.1 Elektrický náboj

Elektrický náboj je fyzikální veličina, která charakterizuje schopnost některých elementárních částic, jako jsou protony a elektrony, působit na své okolí přitažlivou, popřípadě odpudivou silou. Velikost této síly je přímo úměrná násobku velikosti dvou vzájemně na sebe působících nábojů a nepřímo úměrná druhé mocnině vzdálenosti těchto nábojů. Vzorcem lze tento vztah zapsat jako :

$$F = \frac{Q_1 \cdot Q_2}{4 \cdot \pi \cdot \epsilon \cdot r^2} \quad (3.2.1)$$

Kde Q_1 a Q_2 jsou elektrické náboje, které na sebe působí silou, ϵ je permitivita prostředí a r je vzdálenost daných nábojů.

Jednotkou elektrického náboje v soustavě SI je jeden Coulomb [1C]. Jelikož je hodnota jeden Coulomb velká jednotka, tak se mnohdy pracuje s velikostí elektrického náboje jednoho elektronu, která činí $1,6021 \cdot 10^{-19}$ C. [4]

3.2.2 Elektrický proud

Díky definici elektrického náboje lze snadno definovat další elektrickou veličinu a tou je elektrický proud. Elektrický proud je definován jako průtok náboje o velikosti jednoho coulombu vodičem o jednotkovém průřezu za jednotku času. Vzorcem lze tento vztah zapsat pro konstantní náboj neměnný v čase jako:

$$I = \frac{Q}{t} \quad (3.2.2)$$

popř. pro náboj měnící svou velikost v čase jako:

$$i(t) = \frac{dq(t)}{dt} \quad (3.2.3)$$

Kde Q a $q(t)$ jsou velikosti nábojů procházejícím vodičem o jednotkovém průřezu a t je čas.

Jednotkou elektrického proudu je jeden Ampér [1A]. Jelikož je elektrický proud jednotkou skalární, tak se mu přiřazuje směr a znaménko neboli smysl proudu, který je dle dohody dán tokem kladně nabitých elektrických částic. [4]

3.2.3 Elektrické napětí

Elektrické napětí vyjadřuje práci vykonanou elektrickým polem při přenesení náboje Q o velikosti jednoho coulombu z bodu A do bodu B. Zjednodušeně řečeno lze elektrické napětí definovat jako rozdíl elektrických potenciálů ve dvou různých bodech v prostoru. Vzorcem lze toto zapsat jako:

$$U = \frac{W}{Q} = \varphi_2 - \varphi_1 \quad (3.2.4)$$

Kde W je práce vykonaná elektrickým polem, Q je jednotkový náboj a φ_A a φ_B jsou elektrické potenciály v bodech A a B. Jednotkou elektrického napětí je jeden Volt [1V].

3.3 Prvky obvodu a jejich chování

Abychom mohli zkoumat elektrický obvod, je vhodné zavést pojem obvodového prvku. Obvodový prvek je takový prvek, který je charakterizován počtem svých vývodů (dále jen svorek) a svými parametry, díky kterým lze daný prvek matematicky modelovat.

Počet svorek a jeho parametry nejsou však jediné možnosti dělení obvodových prvků. Dále tyto prvky můžeme dělit na aktivní a pasivní. Aktivní prvky neboli generátory přeměňují energii z vnějšího okolí a přeměňují ji na energii elektrickou, kterou dodávají do obvodu. Příkladem takových obvodových prvků může být ideální zdroj napětí a proudu. Naproti tomu prvky pasivní energii spotřebovávají a přeměňují ji na jinou formu energie (například tepelnou, světelnou atp.).

Prvky lze dále dělit podle jejich voltampérových charakteristik na prvky lineární a nelineární. Lineární prvky mají svoje parametry nezávislé na procházejícím proudu nebo napětí na svorkách, ale pouze na svém parametru. Díky tomu je možné popsat danou voltampérovou charakteristiku pomocí lineární funkce. Příkladem lineárních obvodových prvků může být rezistor, kondenzátor a cívka. Naproti tomu prvky nelineární mění své parametry v závislosti na proudu procházejícím skrz prvek nebo napětí na svorkách. Příkladem takového prvku je například dioda, která mění svůj proud dle Shockleyho rovnice.

U všech těchto prvků předpokládáme, že jejich propojení je provedeno pomocí vodičů s nulovým elektrickým odporem. Díky tomu se výpočty velice zjednodušují.

3.3.1 Zdroje napětí

Zdroje napětí jsou velice důležitou částí obvodu. Jedná se aktivní obvodový prvek, který dodává do obvodu elektrické napětí. Parametrem tohoto prvku je napětí na svorkách. Z hlediska časových změn napětí na svorkách můžeme tyto zdroje dále dělit na zdroje s konstantním napětím, které v čase nemění své napětí, zdroje s harmonickým výstupem, kdy na svorkách můžeme pozorovat harmonický signál a na zdroje se speciálními průběhy, jako jsou obdélník, rampa atp. Polarita neboli orientovanost napětí se určuje jako okamžitá hodnota napětí v čase t a většinou se značí pomocí znaku „+“ a „-“ nebo orientovanou šipkou.

Mezi další vlastnosti napěťového zdroje patří nulový vnitřní odpor. Díky této vlastnosti mají tyto prvky schopnost dodávat do obvodu napětí nezávislé na odebíraném proudu.

Další dělení těchto zdrojů je na zdroje nezávislé a na zdroje závislé (neboli řízené), u nichž je napětí na svorkách ovlivněno buď napětím, nebo proudem na prvku v jiné části obvodu. Příkladem nezávislých zdrojů může být baterie či elektrická zásuvka, které mají velmi nízký vnitřní odpor. Příkladem závislého zdroje řízeným napětím může být například nahrazení reálného operačního zesilovače modelem, který je na svém vstupu řízen napětím a jeho výstup je funkcí tohoto napětí.

3.3.2 Zdroj proudu

Podobně jako zdroje napětí jsou i zdroje proudu brány jako aktivní obvodové prvky, které dodávají do obvodu elektrický proud. Jejich parametrem je proud dodávaný do obvodu. Stejně jako u napěťového zdroje lze i proudový zdroj dělit dle časových závislostí. Polarita se u zdroje proudu určuje pomocí orientované šipky.

Na rozdíl od napěťového zdroje má proudový zdroj nekonečně velký vnitřní odpor a tím schopnost dodávat do obvodu proud nezávislý na zátěži.

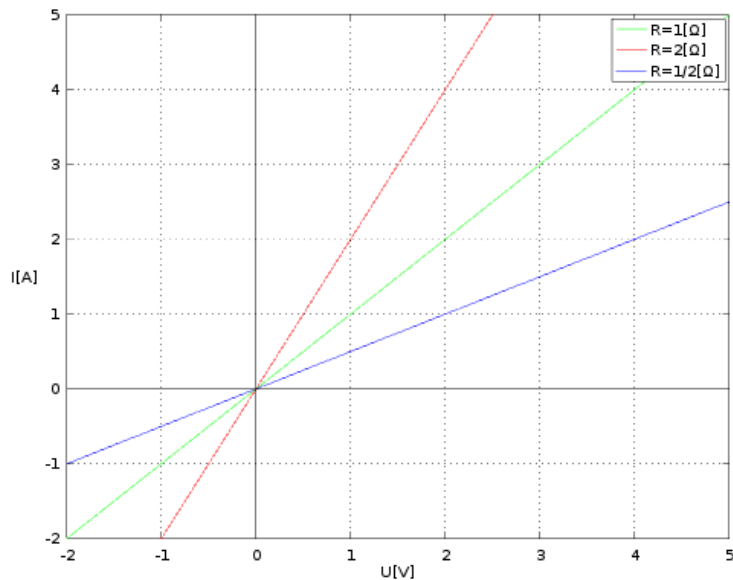
Dělení na nezávislé a závislé je stejné jako u napěťového zdroje.

3.3.3 Rezistor

Nejznámějším prvkem, který je využíván v elektrických obvodech je bezesporu rezistor. Jedná se o pasivní prvek s lineární voltampérovou charakteristikou. Jeho jediným parametrem je elektrický odpor R (jednotky Ohm [Ω]) a platí pro něj Ohmův zákon:

$$u(t) = R \cdot i(t) \quad (3.3.1)$$

Kde $u(t)$ je napětí na svorkách rezistoru, $i(t)$ je proud procházející skrz rezistor a R je jeho elektrický odpor. Voltampérová charakteristika rezistoru pro různé hodnoty parametru R vypadá následovně:



Obr 3: Voltampérová charakteristika rezistoru

3.3.4 Kondenzátor

Dalším prvkem hojně využívaným v elektrických obvodech je kondenzátor. Znovu se jedná o pasivní a lineární prvek. Jeho parametr je kapacita C (jednotky Farad [F]). Vztah mezi napětím a proudem je dán vztahem 3.3.2 nebo 3.3.3.

$$i(t) = C \frac{du(t)}{dt} \quad (3.3.2)$$

popř.

$$u(t) = \frac{1}{C} \int_0^T i(t) dt + u(0) \quad (3.3.3)$$

Kde $u(t)$ je napětí na svorkách kondenzátoru, $i(t)$ je proud procházející skrz kondenzátor, C je jeho kapacita a $u(0)$ je počáteční napětí na kondenzátoru v čase $t=0$.

I když se jedná o lineární prvek, tak je složité díky derivaci zakreslit jeho voltampérovou charakteristiku. Proto se častěji používá jiná charakteristika a to voltcoulombová charakteristika, která vypadá stejně jako voltampérová charakteristika u rezistoru, nicméně v této charakteristice je závislá veličina náboj na kondenzátoru a nikoliv proud. Parametrem je výše uvedená kapacita C .

3.3.5 Cívka

Posledním pasivním a lineárním prvkem, který zde bude zmíněný je cívka. Tento prvek je charakterizován jeho indukčností L (jednotky Henry [H]) a platí pro něj vzorec:

$$u(t) = L \frac{di(t)}{dt} \quad (3.3.4)$$

popř.

$$i(t) = \frac{1}{L} \int_0^T u(t) dt + i(0) \quad (3.3.5)$$

Kde $u(t)$ je napětí na svorkách cívky, $i(t)$ je proud procházející skrz cívku, L je její indukčnost a $i(0)$ je počáteční proud procházející skrz cívku v čase 0.

Stejně jako u kondenzátoru je velice složité nakreslit voltampérovou charakteristiku cívky. U této součástky se kreslí graf závislosti magnetického toku cívky v závislosti na proudu, který je zase prakticky stejný jako voltampérová charakteristika rezistoru.

3.3.6 Dioda

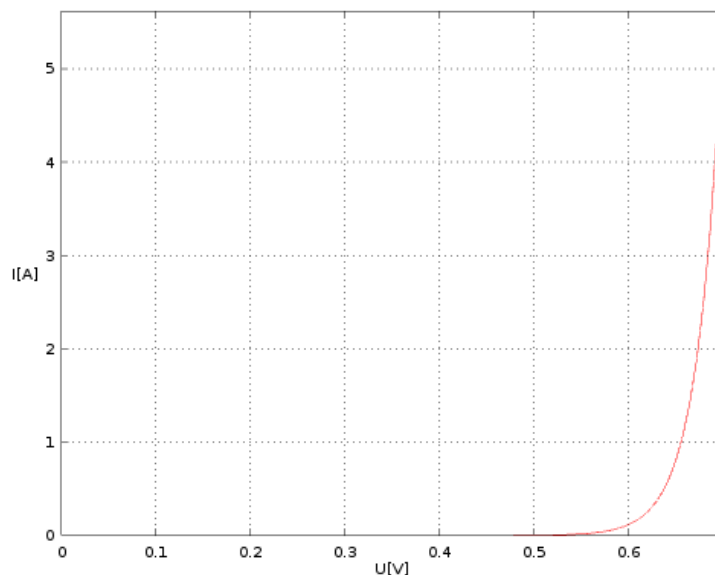
Nyní se přesouváme z lineárních prvků na prvky nelineární. Prvním takovým nelineárním prvkem je dioda. Tato elektronická součástka obsahuje zpravidla jeden přechod PN. Dioda je charakterizována v propustném směru tzv. Shockleyho rovnicí:

$$I_D = I_S \cdot \left(e^{\frac{U_D}{U_T}} - 1 \right) \quad (3.3.6)$$

Kde I_D je proud diodou, I_S je reverzní saturační proud diody U_T je teplotní napětí přechodu PN a U_D je napětí na diodě. Teplotní napětí U_T můžeme vyjádřit jako:

$$U_T = \frac{n \cdot k \cdot T}{q} \quad (3.3.7)$$

Kde n je emisní koeficient, který se pohybuje od čísla 1 do 2, k je Boltzmannova konstanta, q je náboj jednoho elektronu a T je teplota diody v stupních Kelvina. Jak je vidět z rovnice 3.3.6 a 3.3.7 tak proud diodou je závislý nejenom na přiloženém napětí, ale i teplotě dané diody. Grafem této funkce pro hodnoty proudu $I_S = 10E-12$ [A], $t = 26,85$ [°C] a $n = 1$ je následující funkce:



Obr 4: Voltampérová charakteristika diody v propustném směru

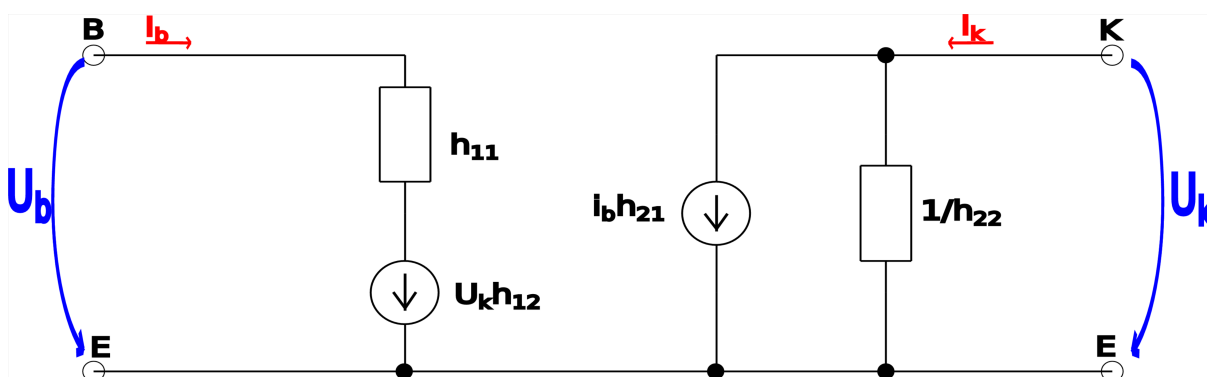
V oblasti závěrného napětí teče přes diodu jen velmi malý závěrný proud až do hodnoty tzv. průrazného napětí, které se u klasických diod pohybuje v řádu jednotek až stovek voltů. V tomto okamžiku začne proud značně narůstat a dochází k poškození diody. Díky těmto vyjmenovaným vlastnostem se klasická dioda nejčastěji používá v propustném směru pro

usměrnění střídavého napětí. Naopak některé typy diod, např. Zenerova dioda, se používají v oblasti závěrného napětí, kde se chovají jako relativně přesný zdroj referenčního napětí.

3.3.7 Tranzistor

Dalším prvkem v elektronických obvodech je tranzistor. Jedná se o součástku nelineární, se třemi svorkami obecně značenými jako báze, kolektor a emitor. Existují dva základní typy. Takzvané tranzistory bipolární a tranzistory unipolární.

Dle složení vrstev jednotlivých polovodičů můžeme tranzistory bipolární dále dělit na tranzistory NPN a PNP. V praxi se velmi často používá náhradní schéma tranzistoru, které je na obrázku 5. Díky tomuto modelu lze snadněji určit jednotlivé převodní charakteristiky. Vstupní charakteristika bipolárního tranzistoru je velice podobná voltampérové charakteristice diody. Jediný rozdíl nastává, že při změně proudu bázi se charakteristika posouvá v y-ové ose. Vliv U_k na U_b je zanedbatelný. Z tohoto důvodu lze psát $U_b = f(I_b)$.



Obr 5: Náhradní schéma tranzistoru

U výstupní charakteristiky je však situace jiná. Proud I_k závisí nejenom na napětí U_k , ale i na proudu I_b . Z tohoto důvodu se většinou volí proud I_b jako parametr (viz obrázek 6). Rovnicemi lze tyto dvě charakteristiky zapsat jako:

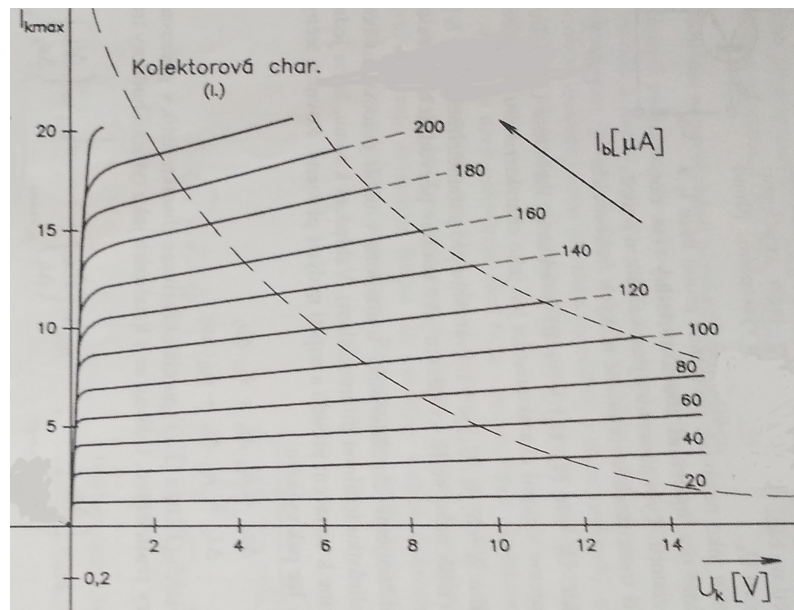
$$\Delta U_b = h_{11} \cdot \Delta I_b + h_{12} \cdot \Delta U_k \quad (3.3.8)$$

$$\Delta I_k = h_{21} \cdot \Delta I_b + h_{22} \cdot \Delta U_k \quad (3.3.9)$$

h-parametry těchto rovnic lze získat následovně:

$$h_{11} = \left(\frac{\Delta U_b}{\Delta I_b} \right)_{U_k = \text{konst.}}, \quad h_{12} = \left(\frac{\Delta U_b}{\Delta U_k} \right)_{I_b = \text{konst.}}, \quad h_{21} = \left(\frac{\Delta I_k}{\Delta I_b} \right)_{U_k = \text{konst.}}, \quad h_{22} = \left(\frac{\Delta I_k}{\Delta U_k} \right)_{I_b = \text{konst.}} \quad (3.3.10)$$

Kde h_{11} je vstupní odpor tranzistoru (v řádu 10^2 až $10^4 \Omega$), h_{12} je zpětný napěťový přenos (v řádu 10^{-4}), h_{21} je proudové zesílení tranzistoru (v řádu 10^1 až 10^3) a h_{22} je výstupní vodivost (v řádu 10^{-3} až 10^{-5} S).

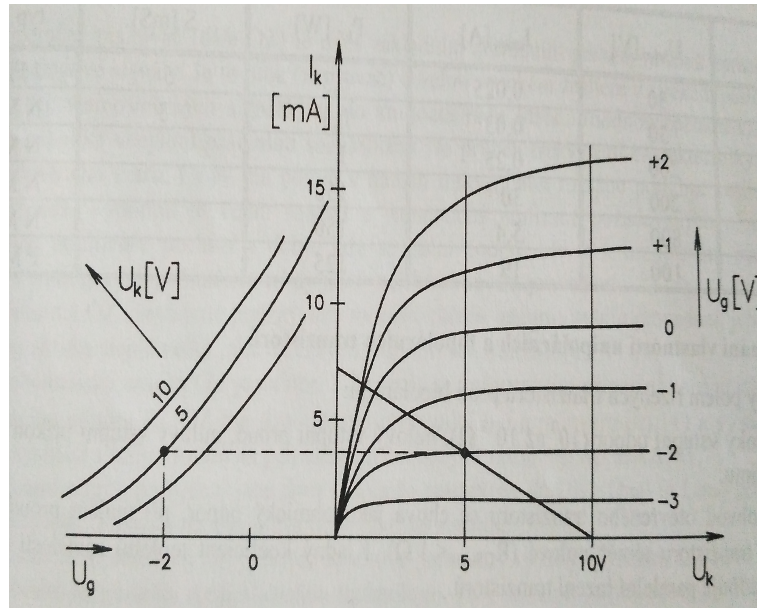


Obr 6: Kolektorová charakteristika bipolárního tranzistoru (převzato z [17])

Nyní si popíšeme základní rovnice pro unipolární tranzistory. Podle zvyklostí se pro unipolární tranzistory v dalším textu bude používat výraz gate pro bázi, výraz source pro kolektor a výraz drain pro emitor. Pro unipolární tranzistory jsou výpočty rovnic o něco jednodušší z důvodu toho, že tento tranzistor má velice velký vstupní odpor a není tedy nutné počítat s proudem do svorky s názvem gate. Tímto se redukuje počet charakteristik pouze na jednu a to výstupní. Vzorcem lze tuto linearizaci v okolí pracovního bodu zapsat jako:

$$\Delta I_k = y_{21} \cdot \Delta U_G + y_{22} \cdot \Delta U_S \quad (3.3.11)$$

Kde parametr y_{21} je řídicí strmost (v řádu desetin až desítek mS) a y_{22} je výstupní vodivost (v řádu 10^{-5} až 10^{-3} mS).



Obr 7: Charakteristiky unipolárního tranzistoru (převzato z [17])

3.3.8 Operační zesilovač

Podobně jako tranzistor má i ideální operační zesilovač pouze tři svorky. Invertující vstup (označený „-“), neinvertující vstup (označený „+“) a výstup. Mezi základní vlastnosti ideálního operačního zesilovače patří nekonečné zesílení v otevřené smyčce, nekonečná šířka pásma, dokonalé potlačení souhlasného signálu, nekonečná vstupní impedance a nulová výstupní impedance. Díky těmto vlastnostem lze psát pouze rovnice pro napěťové zesílení.

$$U_{out} = A \cdot U_{diff} \quad (3.3.12)$$

Kde U_{out} je výstupní napětí operačního zesilovače, A je jeho zesílení a U_{diff} je napěťová diference mezi invertujícím a neinvertujícím vstupem.

3.3.9 Shrnutí obvodových prvků

Pro doplnění jsou v tabulce 1 uvedeny schématické značky jednotlivých elektrických prvků, které jsou standardně používány v analogových simulátorech typu PSpice.

Tabulka 1: Seznam základních elektrotechnických prvků a jejich schématické značky

Název elektrotechnického prvku	Značka elektrotechnického prvku
Zdroj napětí	
Zdroj proudu	
Rezistor	
Kondenzátor	
Cívka	
Dioda	
Tranzistor bipolární NPN	
Tranzistor bipolární PNP	
Tranzistor unipolární N-kanál	
Tranzistor unipolární P-kanál	
Operační zesilovač	

3.4 Metody pro analýzu elektrického obvodu

V této části jsou vysvětleny některé metody, které mohou být použity pro analýzu elektrického obvodu. Jedná se o metody algoritnické, které mohou být implementovány v počítači jako program. Mezi tyto metody patří Kirchhoffovy zákony, metoda smyčkových proudů, metoda uzlových napětí a v neposlední řadě modifikace metody uzlových napětí v podobě metody razítek.

3.4.1 Kirchhoffovy zákony

Mezi nejznámější metody patří Kirchhoffovy zákony. Tyto zákony jsou pojmenovány podle německého fyzika Gustava Kirchhoffa a byly poprvé prezentovány již v roce 1845. Většina zde představených metod jsou pouze deriváty této metody. Výhodou této je její univerzálnost a tím možnost analyzovat jakýkoliv obvod bez ohledu na linearitu či nelinearitu nebo na to, jestli se jedná o obvod, který je časově proměnný či nikoliv.

Prvním Kirchhoffův zákon nebo též uzlový zákon říká, že suma proudů vtékající/vytékající z uzlu je rovna nule a lze ho pomocí rovnice vyjádřit takto:

$$\sum_{k=1}^n I_k = 0 \quad (3.4.1)$$

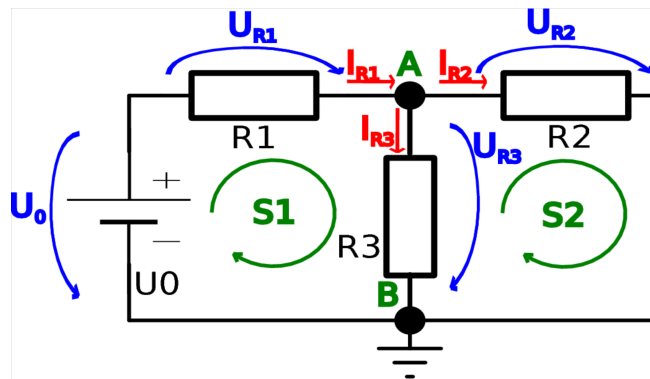
Kde I_k jsou jednotlivé proudy uzlu a n je počet uzlů. Znaménko u jednotlivých proudů je vhodné volit tak, že proudy vstupující do uzlu jsou brány s kladným znaménkem a proudy vystupující z uzlu se záporným.

Druhý Kirchhoffův zákon vychází ze zákona o zachování energie a říká, že suma napětí v uzavřené smyčce je rovna nule a lze ji podobně jako první zákon zapsat pomocí rovnice takto:

$$\sum_{k=1}^n U_k = 0 \quad (3.4.2)$$

Kde U_k jsou jednotlivá napětí na větvích a n je počet větví ve zkoumané smyčce. Znaménko pro jednotlivá napětí je voleno tak, že napětí orientována proti směru smyčky jsou se znaménkem „-“ a napětí orientována po směru smyčky jsou se znaménkem „+“.

Pro názornost této metody byl vybrán jednoduchý obvod dle obrázku 8. V dalším textu bude popsán algoritmus výpočtu.



Obr 8: Schéma obvodu pro demonstraci Kirchhoffových zákonů

Pro sestavení jednotlivých rovnic pro uzly a smyčky je nutné nejdříve označit v elektrickém obvodu nezávislé uzly a smyčky. U uzlů je možné zvolit z uzlu A, nebo B. Oba tyto uzly poskytují stejnou informaci a není nutné je zapisovat oba. U smyček jsou zvoleny smyčky S1 a S2. V dalším kroku je již možné sestavit rovnice. Rovnice pro uzel A lze zapsat následovně:

$$A : I_{R1} - I_{R2} - I_{R3} = 0 \quad (3.4.3)$$

V této rovnici je znázorněno, že proud I_{R1} vtéká do uzlu A a proudy I_{R2} a I_{R3} z něj vytékají. Nyní můžeme sestavit rovnice pro smyčky S1 a S2:

$$S1 : -U_0 + U_{R1} + U_{R3} = -U_0 + I_{R1} \cdot R1 + I_{R3} \cdot R3 = 0 \quad (3.4.4)$$

$$S2 : -U_{R3} + U_{R2} = -I_{R3} \cdot R3 + I_{R2} \cdot R2 = 0 \quad (3.4.5)$$

Tyto rovnice, kde neznámé jsou větrové proudy (I_{R1} , I_{R2} a I_{R3}) lze následně zapsat v maticovém tvaru jako:

$$\begin{pmatrix} 1 & -1 & -1 \\ R1 & 0 & R3 \\ 0 & R2 & R3 \end{pmatrix} \cdot \begin{pmatrix} I_{R1} \\ I_{R2} \\ I_{R3} \end{pmatrix} = \begin{pmatrix} U_0 \\ 0 \\ 0 \end{pmatrix} \quad (3.4.6)$$

Tato soustava rovnic obsahuje tři neznámé a lze řešit velice snadno například jednou z metod popsaných v kapitole 4.1. Napětí na jednotlivých rezistorech lze získat vynásobením proudů velikostí daného rezistoru.

3.4.2 Metoda smyčkových proudů

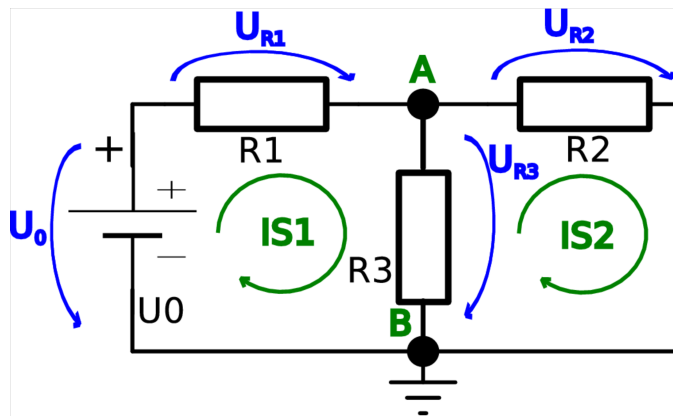
Tato metoda přímo vychází z Kirchhoffových zákonů. Úprava spočívá v tom, že reálné proudy jsou nahrazeny proudy fiktivními (viz obrázek 9), které protékají skrze celou smyčku. U zvoleného obvodu by náhrada vypadala takto:

$$I_{R1} = I_{S1} \quad (3.4.7)$$

$$I_{R2} = I_{S2} \quad (3.4.8)$$

$$I_{R3} = I_{S1} - I_{S2} \quad (3.4.9)$$

Algoritmus je ukázán na stejném obvodu (viz obrázek 9) jako metoda Kirchhoffových zákonů.



Obr 9: Schéma obvodu pro demonstraci metody smyčkových proudů

Na rozdíl od Kirchhoffových zákonů není nutné počítat s uzly. Po této substituci je možné sestavit rovnice jen pro nezávislé smyčky jako:

$$S1: -U_0 + I_{S1} \cdot R1 + (I_{S1} - I_{S2}) \cdot R3 = -U_0 + I_{S1} \cdot (R1 + R3) - I_{S2} \cdot R3 = 0 \quad (3.4.10)$$

$$S2: I_{S2} \cdot R2 - (I_{S1} - I_{S2}) \cdot R3 = I_{S2} \cdot (R2 + R3) - I_{S1} \cdot R3 = 0 \quad (3.4.11)$$

a po zapsání do maticového tvaru:

$$\begin{pmatrix} R1 + R3 & R3 \\ -R3 & R2 + R3 \end{pmatrix} \cdot \begin{pmatrix} I_{S1} \\ I_{S2} \end{pmatrix} = \begin{pmatrix} U_0 \\ 0 \end{pmatrix} \quad (3.4.12)$$

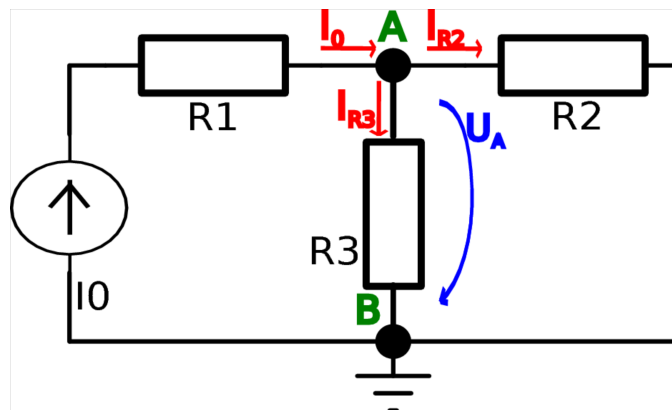
Posledním krokem této analýzy je zpětná substituce fiktivních proudů na reálné a dopočtení napětí.

Jak je z maticového zápisu vidět, počet proměnných se snížil z počátečního počtu tři na dva. Při takto malém obvodu není takové snížení moc patrné, nicméně při počítání rozsáhlých obvodů může být tato úspora značná.

3.4.3 Metoda uzlových napětí

Další metodou založenou na aplikaci Kirchhoffových zákonů je metoda uzlových napětí. Algoritmus této metody je relativně jednoduchý.

V prvním kroku se zvolí tzv. referenční uzel. Jedná se o uzel obvodu, ke kterému jsou následně vedena všechna napětí z jednotlivých uzlů. Nejčastěji se jako referenční uzel volí uzel s nulovým potenciálem, tedy zem. Následně očíslováme či jinak označíme všechny uzly v obvodu. Poté vytvoříme orientovaná napětí z každého uzlu do uzlu referenčního. Tím získáme neznámá uzlová napětí v obvodu, který je znázorněn na obrázku 10. Jedná se o podobný obvod jako při metodách popsaných výše, pouze konstantní zdroj napětí byl zaměněn za konstantní zdroj proudu.



Obr 10: Schéma obvodu pro demonstraci metody uzlových napětí

Nyní můžeme přejít k sestavování rovnic obvodu podle prvního Kirchhoffova zákona pro proudy v jednotlivých uzlech. Jelikož máme v tomto obvodu pouze jediný uzel mimo referenční, tak vyjde pouze jedna rovnice.

$$I_0 = I_{R2} + I_{R3} \quad (3.4.13)$$

Po dosazení neznámých uzlových napětí se tato rovnice může přepsat jako:

$$I_0 = \frac{U_A}{R_2} + \frac{U_A}{R_3} \quad (3.4.14)$$

Nyní již máme pouze jednu rovnici o jedné neznámé, která lze velmi snadno vyčíslit bez použití matic. Při složitějším obvodu z více než jedním uzlem by však zapsání do maticového tvaru bylo vhodné a tvar takovéto matice by byl následovný:

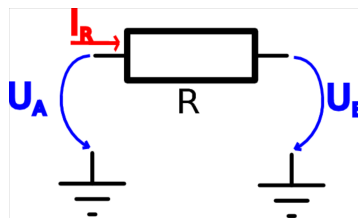
$$\mathbf{G} \cdot \mathbf{u} = \mathbf{i} \quad (3.4.15)$$

Kde \mathbf{G} je vodivostní matice obvodu, \mathbf{u} je vektor neznámých uzlových napětí a \mathbf{i} je vektor budících proudů v jednotlivých uzlech obvodu. Velkou výhodou tohoto algoritmu je snadná implementovatelnost do počítačového programu, kdy je možné přímo ze schématu vytvoření matice \mathbf{G} i vektoru \mathbf{i} . Nevýhodou tohoto algoritmu je nemožnost přímého vytvoření prvků jako jsou zdroje napětí nebo prvků, které nemají svoji admitanční matici.

3.4.4 Metoda razítek

Modifikací metody uzlových napětí lze docílit větší univerzálnosti a odstranit bariéru ve formě nemožnosti přímého použití napěťových zdrojů a prvků bez admitanční matice. Tato metoda přiřazuje každému prvku tzv. „razítko“, které je pak dosazeno do finální matice a vektorů. Důsledkem této metody je ten, že se nebudeme omezovat u vektoru \mathbf{u} pouze na uzlová napětí, ale doplníme je i o neznáme proudy. Další důsledek je, že matice \mathbf{G} v sobě nebude obsahovat pouze admitanční prvky. Z těchto dvou důsledků logicky vyplývá i poslední změna, kdy vektor \mathbf{i} nyní bude obsahovat i napěťové zdroje.

Algoritmus této metody lze znovu velice jednoduše demonstrovat na obvodu z obrázku 2.



Obr 11: Schéma rezistoru se zakreslenými uzlovými napětími

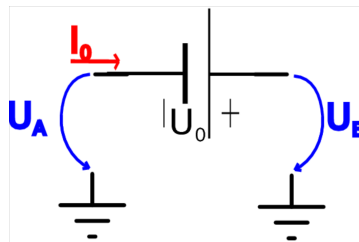
Nejdříve si vytvoříme razítko rezistoru. Začneme tak, že si daný rezistor nakreslíme a zakreslíme u něj proud a napětí na jeho svorkách tak, jak je znázorněno na obrázku 11. Poté si z druhého Kirchhoffova zákona si můžeme vyjádřit:

$$U_A - U_B = I \cdot R \Rightarrow I = \frac{U_A - U_B}{R} \quad (3.4.16)$$

Kde U_A a U_B jsou uzlová napětí na rezistoru, R je jeho odpor a I je proud přes něj tekoucí. Nyní si tuto rovnici vyjádříme v maticovém tvaru.

$$\begin{pmatrix} \frac{1}{R} & -\frac{1}{R} \\ -\frac{1}{R} & \frac{1}{R} \end{pmatrix} \cdot \begin{pmatrix} U_A \\ U_B \end{pmatrix} = \begin{pmatrix} I \\ -I \end{pmatrix} \quad (3.4.17)$$

Tato matice říká, že proud s kladným znaménkem do uzlu A vtéká a z uzlu B z něj vytéká.



Obr 12: Schéma napětového zdroje se zakreslenými uzlovými napětími

Podobně jako u rezistoru budeme postupovat u dalšího prvku a tím je napětový zdroj. Z druhého Kirchhoffova zákona si můžeme vyjádřit:

$$U_A - U_B = U_0 \quad (3.4.18)$$

V této rovnici však nevystupuje proud, který zdroj dodává do obvodu. Proto si ho do matice zavedeme jako neznámou. Získáme tím následující razítko.

$$\begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & -1 \\ 1 & -1 & 0 \end{pmatrix} \cdot \begin{pmatrix} U_A \\ U_B \\ I \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ U_0 \end{pmatrix} \quad (3.4.19)$$

Jak je vidět z matice tak poslední řádka vystihuje rovnici 3.4.19. První a druhý řádek pouze přidává do obvodu proud ze zdroje jako neznámou. Razítka pro další prvky obvodu bychom sestavovali analogicky.

Pro další část tohoto algoritmu se vraťme k obvodu 2. Vytvoříme si matici, která bude obsahovat tolik řádků a sloupců jako je uzlů, včetně těch, které spojují dva prvky obvodu a jsou tak v rozporu s definicí uzlů v kapitole 3.1) a napět'ových zdrojů tzn. počet uzlů = 3, počet zdrojů = 1 a vložíme do nich razítka prvků.

$$\begin{pmatrix} \frac{1}{R_2} + \frac{1}{R_3} & 0 & -\left(\frac{1}{R_2} + \frac{1}{R_3}\right) & 1 \\ 0 & \frac{1}{R_1} & -\frac{1}{R_1} & -1 \\ -\left(\frac{1}{R_2} + \frac{1}{R_3}\right) & -\frac{1}{R_1} & \frac{1}{R_2} + \frac{1}{R_3} + \frac{1}{R_1} & 0 \\ 1 & -1 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} U_{REF} \\ U_A \\ U_B \\ I_0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ U_0 \end{pmatrix} \quad (3.4.20)$$

Nyní vyškrtáme řádky a sloupce týkající se referenčního uzlu a získáme tím výslednou matici pro metodu razítek:

$$\begin{pmatrix} \frac{1}{R_1} & -\frac{1}{R_1} & -1 \\ -\frac{1}{R_1} & \frac{1}{R_2} + \frac{1}{R_3} + \frac{1}{R_1} & 0 \\ -1 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} U_A \\ U_B \\ I_0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ U_0 \end{pmatrix} \quad (3.4.21)$$

Jak je vidět ze vzorce 3.4.21 počet rovnic se zvýšil na tři jako u Kirchhoffových zákonů. Tato metoda obecně zvyšuje počet neznámých nutných k výpočtu, nicméně je dnes implementována do většiny simulátorů hlavně kvůli své jednoduchosti a možnosti vytvářet obvodové matice přímo ze schématu.

3.5 Typy simulací elektrického obvodu

Všechny dnešní simulátory dovolují alespoň tři druhy základních analýz. Jedná se o analýzu AC, DC a tranzientní.

3.5.1 DC analýza

Základním druhem analýzy v elektrickém obvodu je DC analýza. Tato analýza umožňuje zkoumání statických charakteristik obvodových prvků jako jsou voltampérové charakteristiky diod, nelineární charakteristiky tranzistorů atp. Jinými slovy se při tomto druhu analýzy vypočítávají pracovní body jednotlivých prvků při změně budících napětí.

3.5.2 AC analýza

Dalším základním druhem analýzy je AC. Tato metoda zkoumá obvod z pohledu buzení střídavými signály. Nutnou podmínkou pro zahájení takovéto analýzy je výpočet pracovních bodů všech součástek.

3.5.3 Tranzientní analýza

Poslední základní analýzou je analýza tranzientní. I u této metody je nutné vypočítat pracovní body všech součástek. Výsledkem tranzientní analýzy je chování obvodu v závislosti na čase.

4 Matematické metody pro výpočet elektrických obvodů

Jak již bylo řečeno v kapitole 3.4, tak každá algoritmická metoda lze zapsat jako soustava rovnic v maticovém tvaru. V této kapitole jsou rozebrány matematické metody, které lze použít pro řešení takových soustav rovnic.

4.1 Metody pro výpočet lineárních obvodů

V této kapitole jsou uvedeny některé metody, které mohou být použity při řešení lineárních elektrických obvodů. Mezi tyto metody patří Gaussova eliminační metoda a z ní odvozená metoda pro výpočet inverzní matice. Lineární soustava rovnic znamená, že se každá neznámá x vyskytuje v rovnici pouze v první mocnině.

Potom lze vyjádřit jako soustavu n lineárních rovnic o n neznámých jako:

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n &= b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n &= b_2 \\ &\vdots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n &= b_n \end{aligned} \quad (4.1.1)$$

Tato soustava rovnic lze také přepsat do maticového tvaru jako:

$$\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix} \quad (4.1.2)$$

4.1.1 Gaussova eliminační metoda

Z hlediska pochopení se jako nejjednodušší metoda pro výpočet lineárních soustav rovnic jeví Gaussova eliminační metoda. Princip této metody spočívá v postupném zjednodušování matice za pomoci násobení řádků či sloupců matice pomocí konstanty nebo přičítání obsahu jednoho řádku k řádku jinému tak, aby se pod hlavní diagonálou vyskytovaly pouze nuly. Nejlépe se daná metoda vysvětluje na příkladech.

Mějme matici \mathbf{A} , vektor neznámých \mathbf{x} a vektor pravých stran \mathbf{b} v takovémto tvaru:

$$\mathbf{A} = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 4 \\ 3 & 2 & 1 \end{pmatrix}, \mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}, \mathbf{b} = \begin{pmatrix} 14 \\ 26 \\ 10 \end{pmatrix} \quad (4.1.3)$$

Nyní je potřeba vynulovat prvky pod hlavní diagonálou. K tomuto účelu vezmeme dva řádky matice \mathbf{A} a k nim související prvky z vektoru \mathbf{b} a uložíme je do řádkových vektorů. Označme je jako R_1 (například $R_1 = (1 \ 2 \ 3 \ 14)$) a R_2 (například $R_2 = (4 \ 5 \ 4 \ 26)$). Nyní vypočteme koeficient tak, že u vektoru R_2 vezmeme první nenulový prvek (tedy 4) a vydělíme ho s prvkem na stejné pozici ve vektoru R_1 (tedy číslem 1). Tímto číslem vynásobíme celý vektor R_1 . Poté R_1 bude vypadat následovně : $R_1 = (4 \ 8 \ 12 \ 56)$. Nyní od sebe odečteme vektor R_2 a R_1 . Tím získáme vektor, který bude vypadat následovně $(0 \ -3 \ -8 \ -30)$. Tento vektor obsahuje nový řádek matice \mathbf{A} a poslední prvek je nový prvek vektoru \mathbf{b} .

Pokud daný algoritmus implementujeme na naší matici získáme v prvním kroku novou matici \mathbf{A} a nový vektor \mathbf{b} .

$$\begin{pmatrix} 1 & 2 & 3 \\ 0 & -3 & -8 \\ 0 & -4 & -8 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 14 \\ -30 \\ -32 \end{pmatrix} \quad (4.1.4)$$

Tento postup opakujeme, dokud nezískáme pod hlavní diagonálou samé nuly.

$$\begin{pmatrix} 1 & 2 & 3 \\ 0 & -3 & -8 \\ 0 & 0 & 2,6667 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 14 \\ -30 \\ 8 \end{pmatrix} \quad (4.1.5)$$

Tímto jsme získali soustavu rovnic v následujícím tvaru:

$$\begin{aligned} 1 \cdot x_1 + 2 \cdot x_2 + 3 \cdot x_3 &= 14 \\ -3 \cdot x_2 - 8 \cdot x_3 &= -30 \\ 2,667 \cdot x_3 &= 8 \end{aligned} \quad (4.1.6)$$

Jak je vidět z poslední rovnice, tak hodnota x_3 je rovna 3. Postupným dosazováním lze dopočítat zbylé proměnné.

4.1.2 Gauss metoda se zpětným chodem

Tato metoda přímo vychází z Gaussovy eliminační metody. Postup celé metody je stejný až do doby vyčíslení jednotlivých proměnných, kdy se místo pracného dosazování zopakuje celý proces eliminace znovu. Tentokrát se však nuluje ta část matice, která je nad hlavní diagonálou. Pro ilustraci použijeme výsledek z předchozího příklad.

$$\begin{pmatrix} 1 & 2 & 3 \\ 0 & -3 & -8 \\ 0 & 0 & 2,6667 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 14 \\ -30 \\ 8 \end{pmatrix} \quad (4.1.7)$$

V prvním kroku této eliminace tedy získáme nuly ve třetím sloupci.

$$A = \begin{pmatrix} 1 & 2 & 0 \\ 0 & -3 & 0 \\ 0 & 0 & 2,667 \end{pmatrix}, x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}, b = \begin{pmatrix} 5 \\ -6 \\ 8 \end{pmatrix} \quad (4.1.8)$$

V dalším kroku již získáme prvky je na hlavní diagonále.

$$A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -3 & 0 \\ 0 & 0 & 2,6667 \end{pmatrix}, x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}, b = \begin{pmatrix} 1 \\ -6 \\ 8 \end{pmatrix} \quad (4.1.9)$$

Nyní již máme soustavu rovnic, která lze velice snadno vyčíslit.

$$\begin{aligned} 1 \cdot x_1 &= 1 \Rightarrow x_1 = 1 \\ -3 \cdot x_2 &= -6 \Rightarrow x_2 = 2 \\ 2,6667 \cdot x_3 &= 8 \Rightarrow x_3 = 3 \end{aligned} \quad (4.1.10)$$

4.1.3 Inverzní matice

Pokud se matice často nemění a jedná-li se o čtvercovou regulární matici je vhodné k této matici vypočítat matici inverzní. Inverzní matice je taková matice, která při násobení s maticí původní vytvoří matici jednotkovou. Jednotková matice je taková matice, která má na své hlavní diagonále číslo 1 a ve zbytku samé nuly.

Výpočet inverzní matice probíhá podobně jako Gaussova eliminace. Nejdříve si vytvoříme matici, která bude mít dvojnásobný počet sloupců než matice původní. Do levé části zapíšeme matici původní a do pravé jednotkovou.

$$\begin{pmatrix} 1 & 2 & 3 & 1 & 0 & 0 \\ 4 & 5 & 4 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 & 0 & 0 \end{pmatrix} \quad (4.1.11)$$

Nyní začneme provádět úpravy stejné jako při Gaussově eliminaci, ale aplikujeme je na celý řádek.

$$\begin{pmatrix} 1 & 2 & 3 & 1 & 0 & 0 \\ 0 & -3 & -8 & -4 & 1 & 0 \\ 0 & -4 & -8 & -3 & 0 & 1 \end{pmatrix} \quad (4.1.12)$$

Tímto získáme v levé části matici nuly pod hlavní diagonálou.

$$\begin{pmatrix} 1 & 2 & 3 & 1 & 0 & 0 \\ 0 & -3 & -8 & -4 & 1 & 0 \\ 0 & 0 & 2,6667 & 2,3333 & -1,3333 & 1 \end{pmatrix} \quad (4.1.13)$$

Nyní je třeba udělat totéž nad hlavní diagonálou.

$$\begin{pmatrix} 1 & 0 & 0 & 0,375 & -0,5 & 0,875 \\ 0 & -3 & 0 & 3 & -3 & 3 \\ 0 & 0 & 2,6667 & 2,3333 & -1,3333 & 1 \end{pmatrix} \quad (4.1.14)$$

Abychom dostali na pravé straně inverzní matici tak musíme na levé straně dostat jednotkovou matici. Toho docílíme tak, že všechny řádky matice vydělíme hodnotou prvku na hlavní diagonále.

$$\begin{pmatrix} 1 & 0 & 0 & 0,375 & -0,5 & 0,875 \\ 0 & 1 & 0 & -1 & 1 & -1 \\ 0 & 0 & 1 & 0,875 & -0,5 & 0,375 \end{pmatrix} \quad (4.1.15)$$

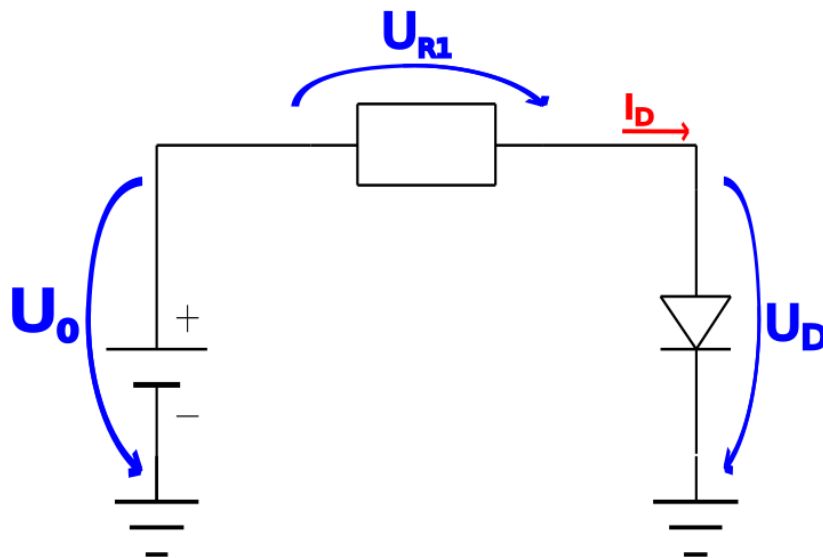
Tím jsme získali inverzní matici k matici původní a nyní je možné vynásobit tuto matici s vektorem pravých stran čímž získáme konečný výsledek.

$$\begin{pmatrix} 0,375 & -0,5 & 0,875 \\ -1 & 1 & -1 \\ 0,875 & -0,5 & 0,375 \end{pmatrix} \cdot \begin{pmatrix} 14 \\ 26 \\ 10 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} \quad (4.1.16)$$

4.2 Metoda pro výpočet nelineárních obvodů

Nelineární obvod je takový obvod ve kterém se objevuje alespoň jeden nelineární prvek. Výsledkem toho je, že se v dané rovnici nevyskytují proměnné x pouze v první mocnině, ale je možné, že se vyskytují například i uvnitř funkce jiné. Cílem tohoto výpočtu je zjistit kořeny dané funkce tzn. nelézt takové x_0 , pro které platí $f(x_0) = 0$.

Příkladem takového obvodu může být jednoduchý elektrický obvod znázorněný na následujícím obrázku.



Obr 13: Schéma obvodu pro řešení nelineárních rovnic

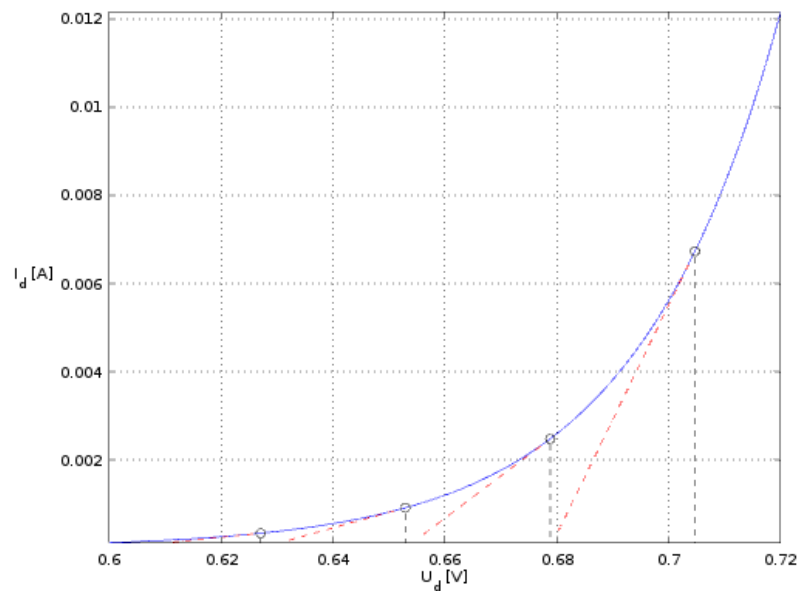
4.2.1 Newtonova metoda pro řešení nelineární rovnice

Na začátku této metody je nutné si určit výchozí předpoklad, že daná funkce, která bude řešená má v celém svém řešeném rozsahu derivaci. Poté iterační vzorec této metody můžeme napsat jako:

$$x[k+1] = x[k] - \frac{f(x[k])}{f'(x[k])} \quad (4.2.1)$$

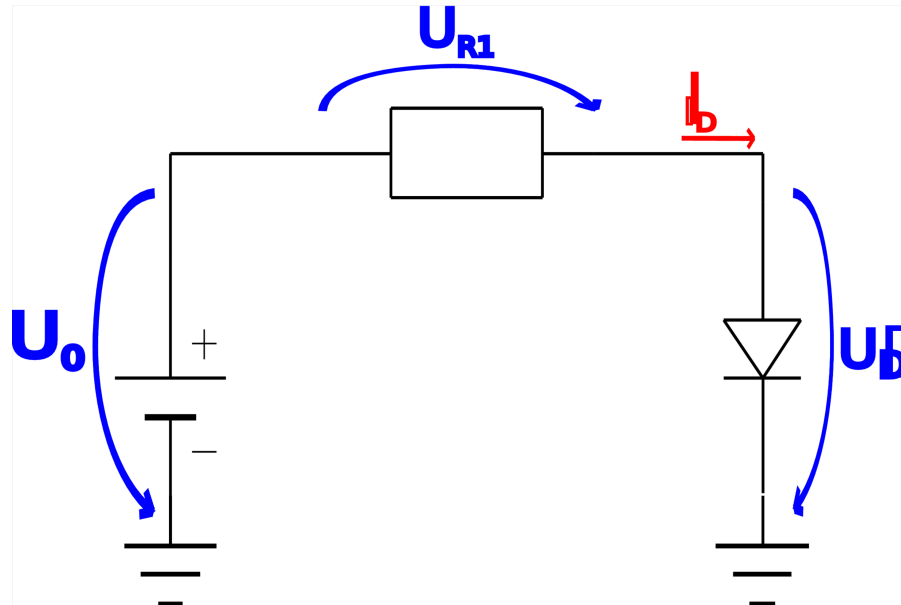
Kde $x[k+1]$ je výsledná hodnota při dané iteraci, $x[k]$ je hodnota výsledku předchozí iterace, $f(x[k])$ je funkční hodnota v bodě $x[k]$ a $f'(x[k])$ je funkční hodnota v bodě $x[k]$ pro derivovanou funkci.

Graficky lze tuto metodu znázornit pomocí grafu 14. V tomto grafu je vidět postupná konvergence metody ke kořenu funkce.



Obr 14: Grafické znázornění Newtonovy metody

Nyní vezmeme obvod dle obrázku 15. Pro obvod platí: $R_1=100$, $I_s=10E-15$, $U_T=26E-3V/^\circ C$ a $U_0 = 3V$.



Obr 15: Schéma nelineárního obvodu

Dle druhého Kirchhoffova zákona můžeme odvodit rovnici funkce $f(U_D)$ jako:

$$U_0 = R1 \cdot I_D + U_D = R1 \cdot I_s \cdot \left(e^{\frac{U_D}{U_T}} - 1 \right) + U_D \Rightarrow f(U_D) = R1 \cdot I_s \cdot \left(e^{\frac{U_D}{U_T}} - 1 \right) + U_D - U_0 = 0 \quad (4.2.2)$$

Derivace této funkce podle U_D vypadá následovně:

$$\frac{df(U_D)}{dU_D} = \frac{R1 \cdot I_s}{U_T} \cdot \left(e^{\frac{U_D}{U_T}} \right) + 1 \quad (4.2.3)$$

Nyní tuto rovnici přepíšeme do iteračního vzorce Newtonovy metody a dosadíme zadané hodnoty.

$$U_D[k+1] = U_D[k] - \frac{R1 \cdot I_s \cdot \left(e^{\frac{U_D[k]}{U_T}} - 1 \right) + U_D[k] - U_0}{\frac{R1 \cdot I_s}{U_T} \cdot \left(e^{\frac{U_D[k]}{U_T}} \right) + 1} \quad (4.2.4)$$

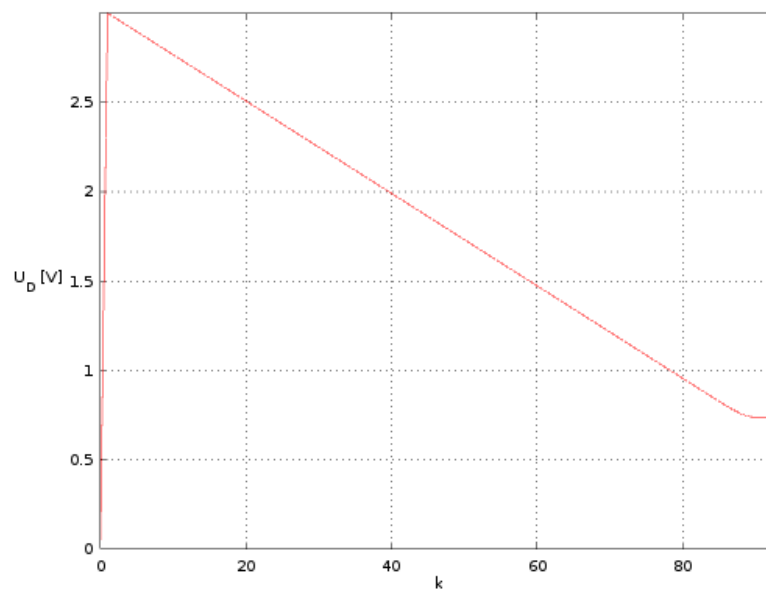
$$U_D[k+1] = U_D[k] - \frac{100 \cdot 10E-15 \cdot \left(e^{\frac{U_D[k]}{26E-3}} - 1 \right) + U_D[k] - 3}{\frac{100 \cdot 10E-15}{26E-3} \cdot \left(e^{\frac{U_D[k]}{26E-3}} \right) + 1}$$

V první fázi zvolme jako počáteční hodnotu $U_D[0] = 0$.

Tabulka 2: Konvergence Newtonovy metody při počáteční podmínce 0V

k	$U_D[k]$	$U_D[k+1]$
0	0	3
1	3	2,974125
2	2,974125	2,948250
92	0,736106	0,736095
93	0,736095	0,736095

Jak je vidět z tabulky 2 a grafu 16 metoda získává přesnější výsledek až při 93 iteraci.

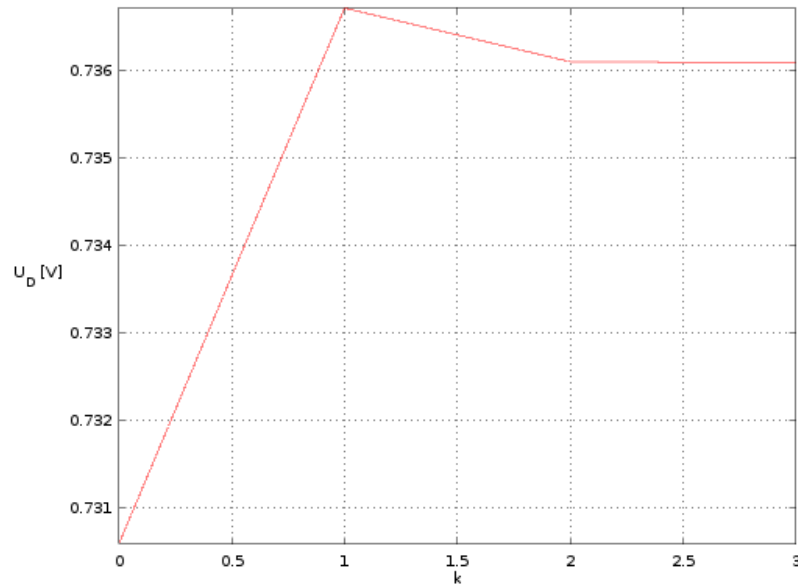


Obr 16: Graf konvergence Newtonovy metody při počáteční podmínce 0V

Pokud zvolíme počáteční hodnotu $U_D[0] = 0,730580$ metoda generuje následující výsledky.

Tabulka 3: Konvergence Newtonovy metody při počáteční podmínce 0,730580V

k	$U_D[k]$	$U_D[k+1]$
0	0,730580	0,736720
1	0,736720	0,736100
2	0,736100	0,736095
3	0,736095	0,736095



Obr 17: Graf konvergence Newtonovy metody při počáteční podmínce 0,73058V

Jak je vidět z předchozích grafů a tabulek tak volba počáteční podmínky zrychluje konvergenci daného algoritmu. Hodnota napětí rovna 0,730580 není zvolena náhodně. Jedná se o hodnotu tzv. kritického napětí, kdy převodní charakteristika diody má nejmenší poloměr zakřivení křivky. [9]

Všechny iterační algoritmy mají tu nevýhodu, že konvergují k přesnému výsledku až v nekonečnu. Z tohoto důvodu se většinou volí zastavovací podmínka taková, že dva po sobě jdoucí prvky jsou menší než určitá mez ε , kterou si zvolíme.

$$\|x[k+1] - x[k]\| < \varepsilon \quad (4.2.5)$$

4.2.2 Newtonova metoda pro řešení soustav nelineárních rovnic

Podobný princip jako při výpočtu jedné nelineární rovnice, lze aplikovat pro soustavu nelineárních rovnic. Soustavou nelineárních rovnic rozumíme takovou soustavu rovnic, kde alespoň jedna rovnice je nelineární. Newtonův zobecněný iterační vzorec vypadá velmi podobně jako vzorec pro jednu rovnici

$$x[k+1] = x[k] + F'(x[k])^{-1} \cdot F(x[k]) \quad (4.2.6)$$

$F(\mathbf{x}[k])$ je vektorová funkce n reálných proměnných, $\mathbf{x}[k]$ je vektor n proměnných a $F'(\mathbf{x}[k])$ je derivovaná vektorová funkce n reálných proměnných.

Derivovaná vektorová funkce $F'(\mathbf{x}[k])$ se též nazývá Jacobiho matice. Značí se J a vypadá následovně:

$$J = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} x[k] & \frac{\partial f_1}{\partial x_2} x[k] & \cdots & \frac{\partial f_1}{\partial x_n} x[k] \\ \frac{\partial f_2}{\partial x_1} x[k] & \frac{\partial f_2}{\partial x_2} x[k] & \cdots & \frac{\partial f_2}{\partial x_n} x[k] \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} x[k] & \frac{\partial f_n}{\partial x_2} x[k] & \cdots & \frac{\partial f_n}{\partial x_n} x[k] \end{pmatrix} \quad (4.2.7)$$

Jak je vidět, tato metoda se velmi podobá Newtonově metodě pro výpočet jedné nelineární rovnice, jen je zobecněna pro soustavy rovnic. Velký rozdíl ovšem přichází, pokud se začneme zabývat konvergencí této metody. Obecně lze nalézt iterační metody pro výpočet jedné nelineární rovnice, které vždy konvergují. U soustavy nelineárních rovnic taková metoda neexistuje a tím ani neexistuje uspokojivý jednoznačný postup pro jejich řešení. [16]

Tato metoda není výjimkou. Při nevhodně zvoleném počátečním vektoru $\mathbf{x}[0]$ metoda diverguje a je nutné se vrátit na počátek algoritmu a zvolit jiné počáteční podmínky.

Stejně jako u Newtonovy metody pro výpočet jedné rovnice i zde se volí zastavovací podmínka, kdy volíme požadovaný rozdíl mezi dvěma po sobě jdoucími iteracemi s tím rozdílem, že se aplikuje na celý vektor.

4.3 Metody pro výpočet obyčejných diferenciálních rovnic

V elektrických obvodech se velmi často vyskytují prvky se setrvačným charakterem jako jsou kondenzátory a cívky. Tyto prvky způsobují, že se lineární rovnice mění na obyčejné diferenciální rovnice. Metod pro řešení takových rovnic je spousta, ale my se v této práci omezíme na tu nejjednodušší, a tou je Eulerova metoda.

4.3.1 Eulerova metoda

Eulerova metoda je jednokroková metoda určená k řešení obyčejných diferenciálních rovnic. K jeho výpočtu je nutné znát počáteční podmínku a derivovanou funkci.

Obecný rekurentní vzorec Eulerovy metody vypadá následovně:

$$y_{k+1} = y_k + h \cdot f(x_k, y_k) \quad (4.3.1)$$

y_k je přibližné řešení z předchozího kroku, h je krok iterace a $f(x_k, y_k)$ je derivovaná funkce.

Principem této metody je nahrazení derivace pomocí diference.

$$\frac{df(t)}{dt} = \lim_{h \rightarrow 0} \frac{f(t) - f(t-h)}{h} \approx \frac{f(t) - f(t-h)}{h} \quad (4.3.2)$$

Použijeme-li tuto metodu na rovnici kondenzátoru, kde nahradíme derivací pomocí diference pak získáme vztah:

$$i(t) = C \cdot \frac{du(t)}{dt} \approx i(t) = C \frac{u(t) - u(t-h)}{h} \quad (4.3.3)$$

Po přepsání tohoto vzorce do obecné formy vychází vzorec pro kondenzátor:

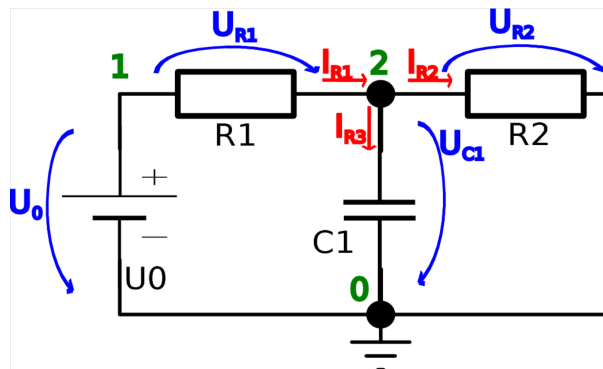
$$u(t) = u(t-h) + h \frac{i(t)}{C} \quad (4.3.4)$$

5 Implementace simulátoru

V této kapitole je popsána implementace simulátoru elektrických obvodů. Celý program byl napsán v programu Visual Studio 2017. Dle požadavku byl použit jazyk C++. Pro alespoň základní funkcionalitu programu bylo nutné vytvořit třídy pro čtení popř. zápis do souboru, parsování netlistu, třídy pro matice a vektory, třídy pro jednotlivé obvodové prvky pro simulaci obvodu a třídu pro vytvoření obvodové matice a následné analýzy elektronického obvodu.

5.1 Zpracování netlistu

Pro simulaci obvodu je nejprve nutné znát strukturu tohoto obvodu a informace o tom, jaký typ analýzy a s jakými parametry se má provádět. Jak již bylo řečeno v kapitole 2.2, tyto informace jsou uloženy v textovém souboru obecně známého jako netlist. Pro náš program byl zvolen podobný formát netlistu, jako je tomu u klasických analogových simulátorů.



Obr 18: Schéma analyzovaného obvodu

5.1.1 Struktura netlistu

Příkladem netlistu, který popisuje obvod 18 je následující kód:

; první řádek souboru je ignorován

.begin

R R1 1 2 4k

R R2 0 2 4k

C C1 2 0 2n

U U0 0 1 5

.probe 2

.tran 0 1E-3

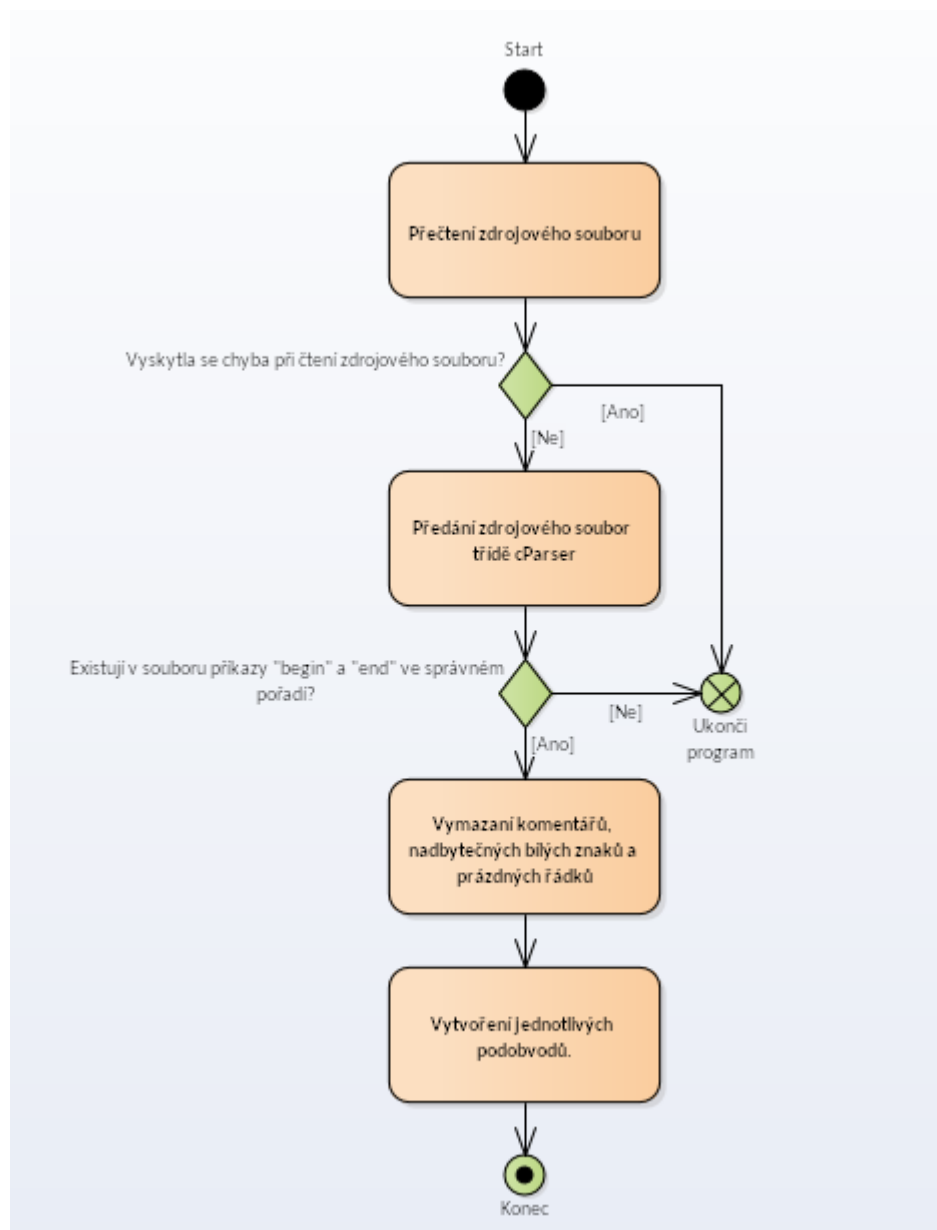
.dc 10E-10 1000

.end

První řádek kódu je vždy ignorován a je vhodné ho doplnit například o název simulovaného obvodu. Na rozdíl od běžných simulátoru, kde je možné připojovat externí knihovny pomocí příkazu „lib“, u našeho toto není možné. Další příkaz, který tedy musí následovat, je příkaz „begin“. Tento příkaz značí začátek obvodu. Pokračovat bychom měli seznamem obvodových prvků. Seznam možných obvodových prvků, které jsou zatím implementovány, je v kapitole 5.1.4. Následují příkazy pro obvodovou analýzu. Těmito příkazy mohou být „tran“, „probe“ a „dc“. Všechny tyto příkazy jsou povinné a pokud nejsou uvedeny, daná simulace vytiskne chybu hlášku a program skončí s chybovým kódem -1. Soubor končí příkazem „end“. Dále je možné pokračovat definicí dalšího obvodu, který musí mít stejnou strukturu, jak již bylo popsáno tzn. „begin“, seznam prvků, příkazy analýzy a „end“. Komentáře jsou uvozeny znakem „;“. Vše za tímto znakem je následně ignorováno.

5.1.2 Algoritmus zpracování netlistu

Algoritmus zpracování netlistu je následující. Třída pro zpracování textových souborů cFiles přečte zdrojový netlist. Ten je pak poslán skrze vnitřní buffer do další třídy cParser. Tato třída má za úkol odstranění komentářů, prázdných řádků a nadbytečných bílých znaků. Tato třída také kontroluje jestli jsou přítomny všechny požadované příkazy ve zdrojovém netlistu. Pokud ne je vypsána chybová hláška na konzoly a program vrací návratovou hodnotu -1.



5.1.3 Příkazy netlistu

Základními příkazy netlistu jsou „begin“ a „end“. Tyto příkazy určují začátek a konec obvodu a nemají žádné parametry. Dalším příkazem je příkaz „probe“. Tento příkaz určuje jaké napětí jsou vypisovány do výstupního souboru. Parametry tohoto příkazu jsou čísla sledovaných uzlů oddělených mezerami. Příkaz „tran“ slouží k nastavení tranzientní analýzy. Jeho parametry jsou krok simulace a konečný čas simulace. Pokud je první parametr, tedy krok simulace, rovný nule, tak se krok simulace počítá z konečného času simulace jako jeho tisícina. Příkaz „dc“ slouží k nastavení DC analýzy. Jeho parametry jsou maximální požadovaná chyba mezi dvěma po sobě jdoucími vzorky a maximální počet kroků iterace. Při přesažení tohoto čísla je jasné, že daná metoda nekonverguje a simulace je ukončena.

5.1.4 Obvodové prvky netlistu

V současné době jsou úspěšně implementovány celkem tři komponenty. Jedná se o rezistor, kondenzátor a zdroj napětí. Syntaxe zápisů těchto prvků je následující¹.

- R [název rezistoru] [uzel_1] [uzel_2] [hodnota odporu]
- C [název kondenzátoru] [uzel_1] [uzel_2] [hodnota kondenzátoru] {inicializační hodnota kondenzátoru}
- U [název napěťového zdroje] [uzel_(-)] [uzel_(+)] [hodnota napětí]

¹ Parametr v hranatých závorkách je povinný. Parametr ve složených závorkách je nepovinný.

5.2 Implementace obvodových prvků

Metoda pro výpočet razítek obvodových prvků byla nastíněna v kapitole 3.4.4. Tato kapitola obsahuje souhrn jednotlivých obvodových prvků a jejich implementaci do simulátoru.

Tabulka 4. Razítka pro obvodové prvky							
Součástka	Parametr	Proměnná	Matice G			Vektor I	
Zdroj napětí	U	-	uzel(+)	uzel(+)	uzel(-)	I	i
			uzel(+)	0	0	1	0
			uzel(-)	0	0	-1	0
			I	1	-1	0	U
Rezistor	R	-	uzel(+)	uzel(+)	uzel(-)	i	
			uzel(+)	$\frac{1}{R}$	$-\frac{1}{R}$	0	
			uzel(-)	$-\frac{1}{R}$	$\frac{1}{R}$	0	
Kondenzátor	C, h	U_c	uzel(+)	uzel(+)	uzel(-)	i	
			uzel(+)	$\frac{C}{h}$	$-\frac{C}{h}$	$-\frac{C}{h} \cdot u_c(k)$	
			uzel(-)	$-\frac{C}{h}$	$\frac{C}{h}$	$\frac{C}{h} \cdot u_c(k)$	
Cívka	L, h	I_c	uzel(+)	uzel(+)	uzel(-)	I	i
			uzel(+)	0	0	1	0
			uzel(-)	0	0	-1	0
			I	1	-1	$\frac{L}{h}$	$-\frac{L}{h} i(k)$
Dioda	-	U_d	uzel(+)	uzel(+)	uzel(-)	i	
			uzel(+)	$\frac{I_s}{U_t} e^{\frac{U_d}{U_t}}$	$-\frac{I_s}{U_t} e^{\frac{U_d}{U_t}}$	$-I_{eq}$	
			uzel(-)	$-\frac{I_s}{U_t} e^{\frac{U_d}{U_t}}$	$\frac{I_s}{U_t} e^{\frac{U_d}{U_t}}$	I_{eq}	

Proudový vektor u diody se doplňuje proudem I_{eq} , jehož vorec vypadá následovně:

$$I_{eq} = \frac{I_s}{U_t} e^{\frac{U_d(k)}{U_t}} \cdot U_d(k) - I_s \cdot \left(e^{\frac{U_d(k)}{U_t}} - 1 \right) \quad (5.2.1)$$

5.2.1 Vytváření nové součástky

Pro vytvoření nové komponenty je nutné vytvořit samostatnou třídu. Na příkladu třídy, která reprezentuje implementaci obvodového prvku rezistoru, budou vysvětleny základní metody, které musí mít každá nově vytvořená třída.

- **Resistor()** - Tento konstruktor je velice důležitý. Bez něho by nebylo možné vkládat instance této třídy do objektu vector standardní knihovny `<vector>`, která je použita pro ukládání jednotlivých prvků obvodu. Tento konstruktor by měl inicializovat hodnoty všech vnitřních proměnných třídy na základní hodnoty.
- **Resistor(std::string)** – Přetížený konstruktor s parametrem `std::string` má za úkol vytvořit instanci nové třídy s již nastavenými správnými parametry. Tzn. jako parametr je předávána již předzpracovaný řetězec z třídy `cParser`. Tím by mělo být zajištěno, že se v řetězci neobjevují například přebytečné bílé znaky. Dále je tento řetězec v této metodě zpracováván. Ke správnému zpracování jednotlivých textových polí je použita třída `cTranslateUnits`, která dokáže překládat jednotlivé textové řetězce na číselné hodnoty.
- **Resistor(const Resistor &)** - Tento kopírovací konstruktor slouží při kopírování instance třídy. Při absenci, nebo špatné implementaci dochází k nečekaným pádům programu.
- **cMatrix Init(cMatrix)** – Tato metoda slouží k prvotní inicializaci matice **G**. Jejím parametrem i návratovou hodnotou je třída `cMatrix`. Daná metoda tedy vezme matici z parametru, zvětší ji na požadovanou velikost a doplní hodnoty prvků.
- **cMatrix GetG(cMatrix, cVector)** – Tato metoda aktualizuje matici **G** s přihlédnutím na předchozí hodnoty napětí, které jsou předávány funkci pomocí parametru.
- **cVector GetI(cVector i, cVector u)** – Tato metoda vytváří a aktualizuje hodnoty ve vektoru **i**.

- **~Resistor()** - Každá třída by měla mít virtuální destruktorku, který slouží k uvolnění zdrojů, které třída využívala.

5.2.2 Postup pro přidání nové součástky do simulátoru

Po vytvoření nové třídy pro součástku je nutné přidat informaci o nové součástce do třídy `cCircuit`.

```
/*Obsah souboru cCircuit.h*/  
  
...  
  
std::vector<Resistor> Rs;  
  
...  
  
/*Obsah souboru cCircuit.cpp*/  
  
...  
  
int cCircuit::SET_GK(){  
...  
  
    for (int i = 0; i < Rs.size(); i++) {  
        Gk=Rs[i].Init(Gk);  
    }  
  
...  
  
    for (int i = 0; i < Rs.size(); i++) {  
        Gk=Rs[i].GetG(Gk, vk);  
    }  
  
...  
}  
  
...
```

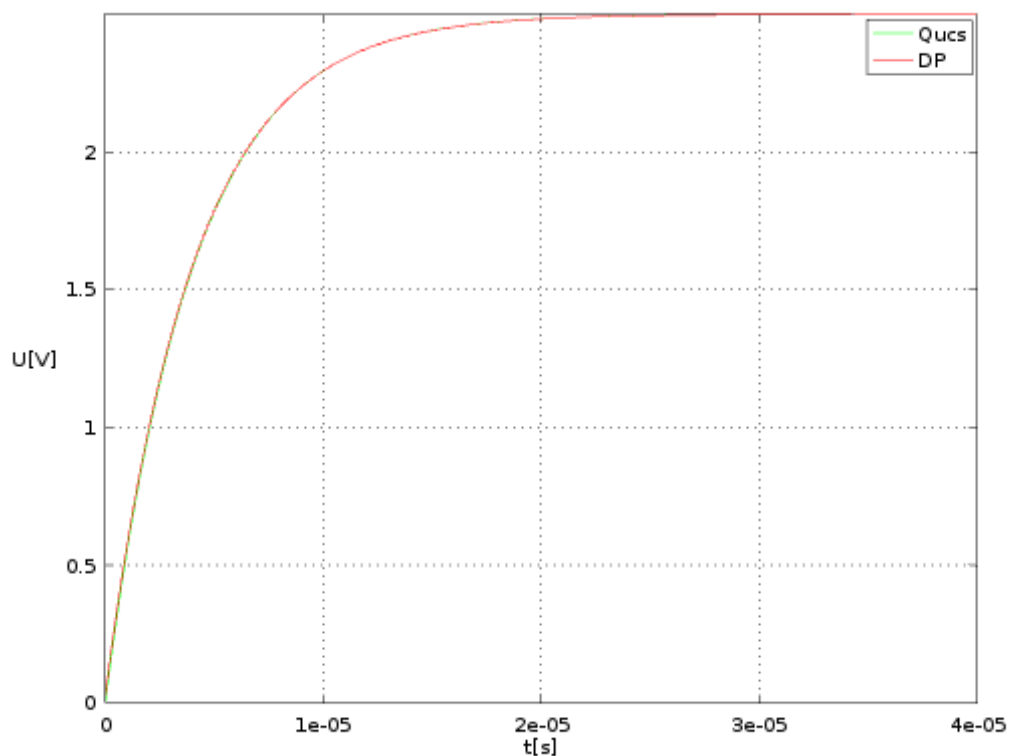
```
void cCircuit::SET_IK()
{
...
    for (int i = 0; i < Rs.size(); i++) {
        lk=Rs[i].Getl(lk, vk);
    }
...
}
...
int cCircuit::SET_CIRCUIT_ELEMENTS(std::vector<std::string> cd){
...
if (cd[line].substr(0, 1) == "R") {
    Resistor tmp(cd[line]);
    Rs.push_back(tmp);
    flag |= dCOMMPONENT;
}
...
}
```

Prvním krokem je vytvoření nové proměnné typu `std::vector<jméno_komponenty>` ve třídě `cCircuit` tak, jak je vidět na ukázce. Dále je nutné vytvořit ve funkcích `SET_GK` a `SET_IK` cyklus, který vyplní data z komponenty do matic a vektorů. Poslední úprava spočívá v naplnění, již zmíněné, proměnné ve funkci `SET_CIRCUIT_ELEMENTS`. Poslední řádek v této funkci slouží jako informace pro třídu `cParser`, že v obvodu existuje nějaká komponenta.

5.3 Výpočty obvodů

Pro kontrolní simulaci byl vybrán obvod z obrázku 18. Simulace byla kontrolována s programem Qucs, který je volně šiřitelnou alternativou programu PSpice.

Pro výpočet rázů setrvačných prvků byla použita Eulerova metoda prvního řádu. Tato metoda dává vcelku uspokojivé výsledky pokud je nastavený malý časový krok u tranzientní analýzy (viz obrázek 19).



Obr 19: Srovnání tranzientní analýzy s programem Qucs

Pro řešení prvků s nelineární voltampérovou charakteristikou byla použita Newtonova metoda. I přes nesčetné množství pokusů, se u tohoto druhu analýzy nepodařilo dosáhnout univerzální metody, která by zajistila konvergenci. Lineární obvody jsou řešeny pomocí inverzní matice.

6 Závěr

Tato práce se zabývala návrhem simulátoru analogových elektronických systémů. Do tohoto simulátoru měly být implementovány jednotlivé obvody ECU a odsimulovány. Následně se tento simulátor měl integrovat do cílového projektu a porovnat jeho výsledky s chováním reálného ECU. Jak je vidět z požadavků práce, každý další bod zadání je závislý na dokončení předchozího bodu a nesplnění jednoho bodu vede k nesplnění bodů dalších.

V teoretické části této práce jsou rozebrány teoretické předpoklady pro vytvoření takového simulátoru. Jsou zde rozebrány jednotlivé prvky obvodu, včetně jejich matematických modelů. Dále zde jsou popsány metody pro analýzu obvodu, včetně jejich algoritmů. Tato část je pak ukončena přehledem metod pro výpočty lineárních, nelineárních a diferenciálních rovnic, které se při tomto druhu simulací často vyskytují.

V praktické části je rozebrána implementace daného simulátoru.

V úvodu práce byly definovány minimální požadavky pro běh simulátoru. Těmito požadavky bylo vytvoření prvků, jako je rezistory, kondenzátory, diody, tranzistory a operační zesilovače. Samozřejmostí je vytvoření zdrojů napětí a zdrojů proudu. Z těchto komponent byly vytvořeny pouze tři. Těmito prvky jsou rezistor, kondenzátor a zdroj napětí.

Simulace nelineárních prvků, jako jsou diody a tranzistory se ukázala být poněkud problematická. Je to z důvodu to, že nejsou dostupné numerické metody, které by konvergovaly za každé situace. Při této simulaci vždy docházelo k divergenci metody a nebylo možné ji odstranit ani metodou, které je popsána ve zdrojové literatuře ([9]), kdy autor problém divergence řeší pomocí různých limit.

Prvky, jako je cívka, zdroj proudu a operační zesilovač nebyly implementován z časových důvodů. Nicméně modely těchto prvků jsou velmi jednoduché a velmi podobné již vytvořeným prvkům.

Jelikož nebyly implementovány všechny potřebné prvky pro simulaci ECU, tak nebylo možné splnit další body zadání.

Seznam literatury a informačních zdrojů

- [1] Příspěvatelé Wikipedie, Simulace [online], Wikipedie: Otevřená encyklopedie, c2017, Datum poslední revize 5. 10. 2017, 06:39 UTC, [citováno 15. 05. 2018] <<https://cs.wikipedia.org/w/index.php?title=Simulace&oldid=15410419>>
- [2] BIOLEK, Dalibor. Řešíme elektronické obvody, aneb, Kniha o jejich analýze. Praha: BEN - technická literatura, 2004. ISBN 80-7300-125-x.
- [3] PINKER, Jiří a Martin POUPA. Číslicové systémy a jazyk VHDL. Praha: BEN - technická literatura, 2006. ISBN 80-7300-198-5.
- [4] HAVLÍČEK, Václav, Martin POKORNÝ a Ivan ZEMÁNEK. Elektrické obvody 1. Praha: Česká technika - nakladatelství ČVUT, 2005. ISBN 80-010-3299-x.
- [5] Wikipedie: Otevřená encyklopedie: Elektrické napětí [online]. c2017 [citováno 15. 05. 2018]. Dostupný z WWW: <https://cs.wikipedia.org/w/index.php?title=Elektrick%C3%A9_nap%C4%9Bt%C3%AD&oldid=15684775>
- [6] KOTLAN, Jiří. Základy teoretické elektrotechniky. Plzeň: Západočeská univerzita, 1995. ISBN 80-708-2172-8.
- [7] MYSLÍK, Jiří. Elektrické obvody: (příručka pro praxi a učebnice pro střední a vysoké školy). Praha: BEN - technická literatura, 1997. ISBN 80-860-5619-8.
- [8] MAYER, Daniel. Úvod do teorie elektrických obvodů: učebnice pro elektrotechnické fakulty. Praha: Státní nakladatelství technické literatury, 1978. ISBN 04-536-78.
- [9] JAHN, Stefan, Michael MARGRAF, Vincent HABCHI a Raimund JACOB. Qucs Technical Papers [online]. 2007, , 264 [cit. 2018-05-19]. Dostupné z: <http://qucs.sourceforge.net/docs/technical/technical.pdf>
- [10] SHI, Guoyong. PRINCIPLES OF CIRCUIT SIMULATION: Nonlinear Device Stamping [online]. 2013 [cit. 2018-05-19]. Dostupné z: http://ic.sjtu.edu.cn/ic/msda/wp-content/uploads/sites/7/2013/02/lect08_nonlinear_stamping.pdf
- [11] SHI, Guoyong. PRINCIPLES OF CIRCUIT SIMULATION PRINCIPLES OF CIRCUIT SIMULATION: Element Stamping [online]. 2013 [cit. 2018-05-19]. Dostupné z: http://ic.sjtu.edu.cn/ic/msda/wp-content/uploads/sites/7/2013/02/lect07_element_stamping.pdf
- [12] KOLKA, Zdeněk. Analýza elektronických obvodů programem OrCAD PSpice [online]. Brno, 2004 [cit. 2018-05-19]. Dostupné z: <https://user.unob.cz/biolek/vyukaVUT/skripta/PSPICE.pdf>
- [13] SUCHÁNEK, Vladimír, ed. a BENDA, Hansjochen. *Dioda, tranzistor a tyristor názorně: programovaný kurs*. 2. nezm. vyd. Praha: SNTL, 1983. 303s
- [14] PINKER, Jiří a KOUCKÝ, Václav. *Analogové elektronické systémy*. 1. vyd. Plzeň: ZČU, 1997. 305 s. ISBN 80-7082-389-5.
- [15] DIETMEIER, Ulrich. *Vzorce pro elektroniku*. 1. čes. vyd. Praha: BEN, 1999. 255 s. ISBN 80-86056-53-8.
- [16] VITÁSEK, Emil. *Numerické metody*. Vyd. 1. Praha: SNTL, 1987. 512 s. Technický

průkaz; Sv. 67. Česká matice technická; Roč. 1987, čís. 428.

- [17] UHLÍŘ, Ivan et al. *Elektrické obvody a elektronika*. Vyd. 1, dotisk. Praha: Vydavatelství ČVUT, 2004. 142 s. ISBN 80-01-02466-0.