

**Univerza v Mariboru,
Fakulteta za elektrotehniko, računalništvo in informatiko
Inštitut za elektroniko in telekomunikacije**



Tomaž Dogša

**CAE/CAD v elektroniki
SIMULACIJA IN MODELIRANJE
ANALOGNIH VEZIJ**

Maribor 2010

Copyright 2010. Prva izdaja, junij 2010.

NASLOV: CAE/CAD v elektroniki: Simulacija in modeliranje analognih vezij

AVTOR: izr. prof. dr. Tomaž Dogša

VRSTA PUBLIKACIJE: učbenik

IZDAJATELJ: Fakulteta za elektrotehniko, računalništvo in informatiko, Univerza v Mariboru.

RECENZENTI: izr. prof. Tadej Tuma, Fakulteta za elektrotehniko, Univerza v Ljubljani,
doc. dr. Mitja Solar, Fakulteta za elektrotehniko, računalništvo in informatiko,
Univerza v Mariboru.

PRELOM STRANI IN OBLIKOVANJE OVITKA: Tomaž Dogša

LEKTORIRANJE: Tina Škrajnar

Vse pravice so pridržane.

Popravki, rešitve nalog itd.: <http://saturn.uni-mb.si/~dogsa/rnv/rnv.html>
Pet natisnjenih izvodov je v Knjižnici tehniških fakultet, Univerza v Mariboru.

CIP - Kataložni zapis o publikaciji
Univerzitetna knjižnica Maribor

621.38.049.77:004.92(075.8)

DOGŠA, Tomaž

CAE/CAD v elektroniki. Simulacija in modeliranje analognih vezij : [Elektronski vir] / Tomaž Dogša. - 1. izd. -
Maribor : Fakulteta za elektrotehniko, računalništvo in informatiko, Inštitut za elektroniko in
telekomunikacije, 2010

Način dostopa (URL) : <http://saturn.uni-mb.si/~dogsa/rnv/rnv.html>

ISBN 978-961-248-227-5

COBISS.SI-ID 64986881

Avtorjev naslov:

Tomaž Dogša
Fakulteta za elektrotehniko, računalništvo in informatiko,
Smetanova 17, 2000 Maribor
tdogsa@uni-mb.si

Datoteke: G_B_Simulacija_in_modeliranje-analognih_vezij_OVITEK.DOC,
G_B_Simulacija_in_modeliranje-analognih_vezij.doc.

Predgovor

Pri vsakem načrtovanju električnih oziroma elektronskih vezij najpogosteje izvajamo dve aktivnosti: analizo in sintezo oziroma projektiranje. Pri načrtovanju si pomagamo z računalniki, na katerih je nameščena posebna programska oprema. Z akronimom **EDA** (*Electronic Design Automation*) in CAE/CAD (*Computer Aided Engineering/ Computer Aided Design*) označujemo orodja (programsko opremo), ki naj v celoti ali pa vsaj delno avtomatizirajo načrtovanje elektronskih naprav. Najbolj razširjena vrsta EDA orodij so simulatorji in načrtovalniki tiskanine.

V začetnem obdobju je bila ta oprema povprečnemu študentu nedostopna, saj je bila izredno draga. Hiter razvoj računalniške tehnologije je omogočil, da ima danes že vsak študent računalnik, na katerem lahko poganja CAE/CAD programe. Pri delu si lahko pomaga s pestrim izborom literature, ki je naštetja na koncu učbenika.

Simulator SPICE je danes najbolj razširjeno orodje za simulacijo vezja. Od množice različic (PSPICE, OPUS, HSPICE, Intusoft SPICE itd.) sem se odločil za Intusoftovo verzijo, predvsem zaradi prijaznega grafičnega vmesnika in ker je prosto dostopna (demonstracijska verzija). Odločilna lastnost pa je bila možnost uporabe skriptnega jezika.

V učbeniku so za ilustracijo uporabljeni preprosti zgledi, zato da so razumljivi tudi tistim, ki niso elektroni. Ta učbenik pa ne predstavlja vir podrobnih in izčrpnih navodil za uporabo simulatorja, saj so razloženi le osnovni ukazi. Večina poglavij je posvečena simulaciji in modeliranju vezij, nekaj malega pa tudi načrtovanju tiskanine. V prvem poglavju je opisan postopek načrtovanja elektronskega vezja. Le drugo poglavje (Simulacija analognih vezij) je namenjeno opisu sintakse najpomembnejših analiz. Pri modeliranju sem se osredotočil na makromodeliranje, saj so modeli osnovnih elementov zelo podrobno že opisani tudi v domači literaturi [TUMA,1997], [TUMA,2009]. Opise modelov najdemo tudi v okviru računalniških navodil. Zadnje poglavje (Razmeščanje in povezovanje) po vsebini pravzaprav ne spada v področje simulacije. Ker pa je mnogo aktivnosti povezanih tudi s simulacijo (npr. analiza integritete), sem ga vseeno uvrstil v to publikacijo.

Junij, 2010

Tomaž Dogša

VSEBINA

1. SPLOŠNE ZNAČILNOSTI NAČRTOVANJA ELEKTRONSKIH SISTEMOV	7
1.1 Struktura in razvoj sodobnih sistemov	8
1.2 Opisovanje vezij	9
1.3 Življenjski cikel tehničnega sistema	12
1.4 Načrtovanje in preverjanje vezij oziroma elektronskih sistemov	14
1.4.1 Načrtovalski pristopi	16
1.4.2 Načrtovanje za masovno proizvodnjo	17
1.4.3 Načrtovanje kompleksnih sistemov	18
1.4.4 Preverjanje sistemov	21
1.4.5 Razširjene načrtovalske aktivnosti	22
1.5 Simulacija sistemov	26
1.5.1 Sistematični pristop k simulaciji sistemov	29
2. SIMULACIJA ANALOGNIH VEZIJ	31
2.1 Opis vezja	32
2.2 Opisi najpogostejših elementov	35
2.3 Struktura vhodne datoteke za simulator SPICE	37
2.4 Osnovne analize	38
2.4.1 .OP analiza delovne točke	41
2.4.2 .DC analiza enosmerne karakteristike	43
2.4.3 .TRAN analiza prehodnega pojava ali tranzientna analiza	48
2.4.4 .AC frekvenčna analiza	53
2.4.5 .FOUR Fourierjeva analiza	57
2.4.6 .TEMP temperaturna analiza	61
2.4.7 .TF enosmerna analiza majhnih sprememb delovne točke	63
2.5 Vaje	65
3. MODELIRANJE	68
3.1 Notranji oziroma parametrični modeli	70
3.2 Zunanji modeli (makromodeliranje)	72
3.2.1 Strukturni modeli	75
3.2.2 Vedenjski modeli	76
3.2.3 Preverjanje ustreznosti modela	79
3.2.4 Uporaba polinomske regresije pri modeliranju elementov	80
3.2.5 Nelinearni krmiljeni viri	81
3.3 Zgledi modelov	85
3.3.1 Modeliranje tunelske diode 1N3716	85
3.3.2 MOS tranzistor	88
3.3.3 Modeliranje operacijskega ojačevalnika	91
3.4 Vaje	95
4. RAZMEŠČANJE IN POVEZOVANJE	96

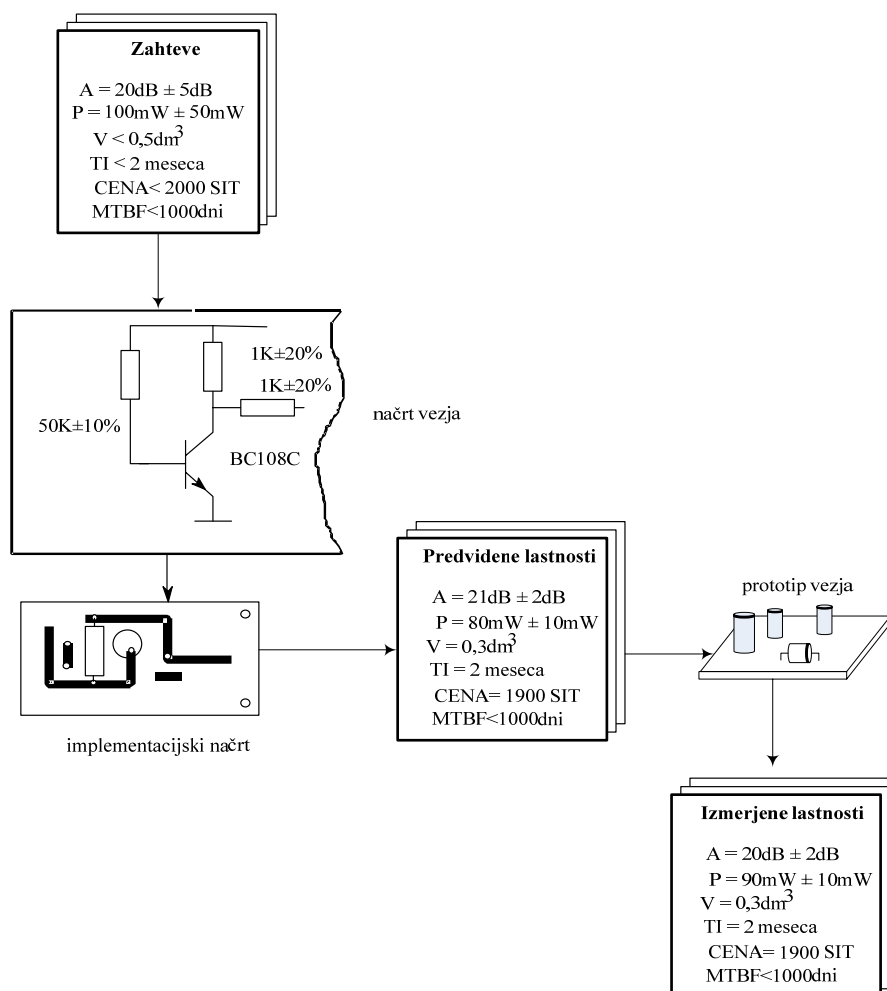
4.1	Razdelitev na module.....	97
4.2	Razmeščanje komponent v modulu	98
4.3	Povezovanje komponent v modulu.....	100
4.4	Preverjanje ustreznosti povezovanja.....	105
4.5	Električne lastnosti povezav in priključkov	105
4.6	Modeliranje povezav.....	108
4.6.1	Koncentriran model linije	108
4.6.2	Model enakomerno distribuirane linije	109
4.6.3	Model brezizgubnega valovoda	110
4.6.4	Model valovoda z izgubami.....	111
4.7	Analiza integritete signala.....	112
4.7.1	Poenostavljeni kriteriji za izbor modela.....	114
4.7.2	Problem hazarda.....	119
4.8	Modeliranje digitalnih vhodov in izhodov.....	119
4.9	Vpliv povezav na lastnosti vezja.....	120
4.10	Izdelava prototipa	121
4.11	Priprava za proizvodnjo	121
5.	SIMULACIJA DIGITALNIH IN ANALOGNO-DIGITALNIH VEZIJ	123
5.1	Značilnosti logičnih (digitalnih) vezij.....	125
5.2	Delovanje logičnega simulatorja.....	128
5.3	Simulacija analogno-digitalnih vezij	130
5.4	Modeliranje logičnih gradnikov v analognem simulatorju	131
5.4.1	Implementacijski model	133
5.4.2	Analogni makromodel	134
5.4.3	Kodni (notranji) model	137
5.4.4	Primerjava modelov	138
5.5	Kombinirani simulator	138
6.	LITERATURA.....	142

1. SPLOŠNE ZNAČILNOSTI NAČRTOVANJA ELEKTRONSKIH SISTEMOV

Cilj vsakega načrtovanja je vezje, ki deluje znotraj dovoljenih odstopanj (glej sliko 1.1). H končnemu rezultatu vodi več različnih poti, ki so odvisne od:

- tehnologije,
- števila vezij (masovna proizvodnja ali pa unikat) in
- zahtevanih karakteristik.

Ne glede na to, ali načrtujemo preprosto ali pa kompleksno vezje, so nekateri načrtovalski koraki v obeh primerih enaki. V tem poglavju bomo opisali tipične načrtovalske korake, s katerimi se srečuje načrtovalec elektronskih sistemov oziroma vezij.

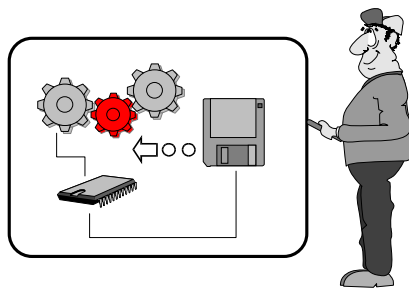


Slika 1.1 Načrtovanje preprostega analognega vezja

1.1. Struktura in razvoj sodobnih sistemov

Elektronika je ena izmed najbolj dinamičnih področij v elektrotehnik, ki od inženirja elektronike zahteva neprestano posodabljanje znanja in hitro prilagajanje potrebam tržišča. Najbolj tipični problemi, s katerimi se ubada današnji načrtovalec, so:

- Sodobni sistem je pogosto sestavljen iz raznovrstnih podsistemov (elektronski del (vezje), elektromehanski del, programska oprema).
- V mnogih primerih razvoj oziroma eksperimentiranje na testnih ploščicah ni mogoče.
- Funkcionalnost sistemov skokovito narašča.
- Čas, ki ga ima načrtovalec na razpolago, je vedno krajši, saj mora produkt čim prej na tržišče.
- Kompleksnost vezij narašča (npr. mikroprocesor).
- Pogosto je treba upoštevati nelinearnosti (npr. v močnostni elektroniki).
- Zahtevana je velika zanesljivost načrtovanja, saj so nekateri prototipi zelo dragi (npr. načrtovanje integriranih vezij).
- Hitrost signalov narašča – upoštevati je treba zakasnitve, parazitne kapacitivnosti in induktivnosti. Elektromagnetna kompatibilnost postaja problematična.
- Zaradi različnih vrst komponent postaja simulacija čedalje zahtevnejša (npr.: analogno-digitalno vezje in elektronsko vezje, ki je kombinirano z mehanskimi komponentami).



Slika 1.2 Struktura sodobnega sistema

Računalniško podprto načrtovanje

Šele ko so se zmogljivosti osebnih računalnikov približale delovnim postajam, se je začel razcvet računalniško podprtega načrtovanja. Pri tovrstnem načrtovanju si pomagamo s programsko opremo, ki v večini primerov zahteva zmogljiv računalnik in dobro grafično opremo. Z akronimom **EDA** (*Electronic Design Automation*) označujemo orodja (programska oprema), ki naj v celoti ali pa vsaj delno avtomatizirajo načrtovanje elektronskih vezij. Najbolj razširjena vrsta EDA orodij so simulatorji in načrtovalniki tiskanine.

Najpogostejše skupine EDA orodij so:

- opisovalniki sistema (schematic entry),
- simulatorji,
- načrtovalniki tiskanine,
- načrtovalniki geometrijske strukture integriranih vezij,
- analizatorji temperature,
- analizatorji integritete signalov.

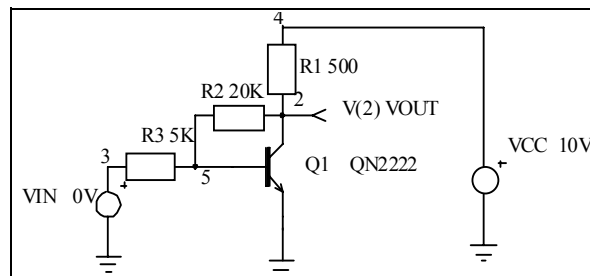
Tabela 1.1 Nekateri akronimi, ki so povezani z računalniško podprtim načrtovanjem

CAE	Computer Aided Engineering
CAD	Computer Aided Design
CAE/CAD	Computer Aided Engineering/Computer Aided Design
CAM	Computer Aided Manufacturing, (Computer Aided Modeling)
ESDA	Electronic System Design Automation
EDA	<i>Electronic Design Automation</i>

1.2. Opisovanje vezij

Skoraj v vsaki fazi načrtovanja, pa tudi pozneje, moramo na določen način opisati izbran sistem oziroma vezje. Opis vezja potrebujemo predvsem pri načrtovanju, implementaciji in pri vzdrževanju. Predmet opisa je **struktura** (npr. blokovna shema, načrt za tiskano vezje) in **obnašanje** oziroma delovanje (npr.: postopek zamenjave usmernika, navodilo za uporabo programa). Opis vezja je lahko namenjen ljudem ali pa strojem (npr.: če želimo simulirati vezje, ga moramo opisati v skladu z določenimi pravili). Poznamo tri vrste formalnih opisov:

- **grafični** (električna shema),
- **besedni** (formalni jeziki, npr.: pascal, VHDL¹, EDIF², SPICE³),
- **kombiniran.**



```
R2 2 5 20K
R3 5 3 5K
VIN 3 0 0V AC 1 PULSE 0 5V
VCC 4 0 10V
Q1 2 5 0 QN2222
R1 2 4 500
```

Slika 1.3 Zgled opisa strukture preprostega ojačevalnika. Zgoraj je grafični, spodaj pa ustrezen besedni opis (SPICE format).

Za opis elektronskih vezij uporabljamo posebna programska orodja, ki jih glede na način opisovanja razvrščamo v tri skupine:

¹ Very high speed hardware description language.

² Electronic Design Interchange Format.

³ SPICE je simulator vezij, katerega bomo pozneje podrobno obravnavali. Če želimo simulirati vezje s simulatorjem SPICE, ga moramo najprej opisati po posebnih pravilih. Temu opisu pravimo lista povezav (angl. netlist).

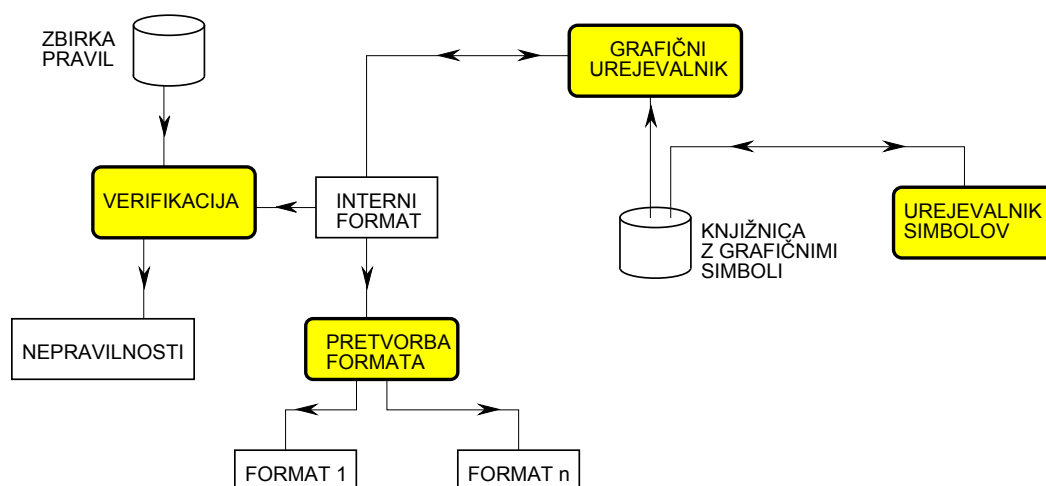
- **grafični urejevalniki** (schematic editor): splošni (npr.: AutoCAD, CorelDraw) in namenski (npr.: Altium, OrCAD, P-CAD, Protel)
- **besedni urejevalniki**: splošni (npr. Notepad) in namenski (npr. VHDL editor)
- **kombinirani** (urejevalnik, ki združuje besedni in grafični način urejanja)

Tabela 1.2 Primerjava med grafičnim in besednim opisom

Grafični opis	Besedni opis
velikost datotek – velika urejanje – potrebujemo poseben urejevalnik lažje razumljiv, bolj pregleden drage vhodno/izhodne naprave enostavna pretvorba v besedni opis	skoraj minimalna velikost datotek navaden besedni urejevalnik manjša preglednost (izjeme) relativno enostavne vhodno/izhodne naprave zelo zahtevna pretvorba v grafični opis

Problemi, s katerimi se srečujemo pri opisovanju sistemov, so:

- Ker ima skoraj vsaka panoga svoj sistem opisovanja, ni enotnega pristopa (npr.: splošna elektrotehnika, elektronske naprave, telekomunikacije, energetski sistemi, inštalacije, informacijski sistemi).
- Zaradi neenotnosti je težko opisovati produkte, ki so plod interdisciplinarnega dela.
- Potrebe po novih simbolih skokovito naraščajo (elektronika, telekomunikacije),
- Ker se kompleksnost produktov večja, postajajo opisi čedalje bolj nepregledni.
- Med različnimi orodji obstaja nekompatibilnost.



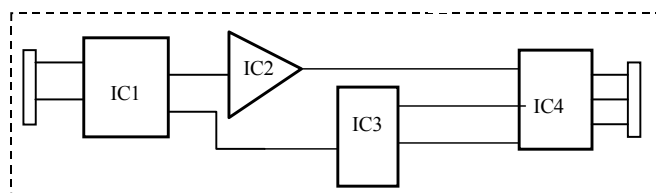
Slika 1.4 Struktura tipičnega programa za grafični opis sistemov

Tipične naloge opisovalnika so:

- Urejanje formalnega opisa sistema
- Pretvorba iz/v druge sisteme opisovanja
- Preverjanje pravilnosti opisa
- Arhiviranje opisa
- Tvorjenje raznih izpisov (npr. kosovnica)

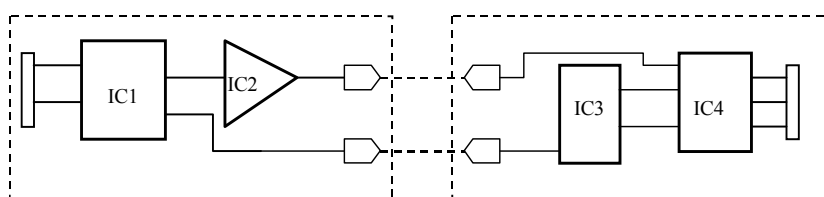
Organizacija načrta

Le zelo preproste načrte lahko narišemo samo na enem listu. Kompleksnejše načrte razdelimo na več listov oziroma datotek. Takemu načrtu pravimo **vodoravno organiziran načrt**. S posebnimi povezovalnimi simboli povemu opisovalniku, kako naj povezuje posamezne dele načrta.

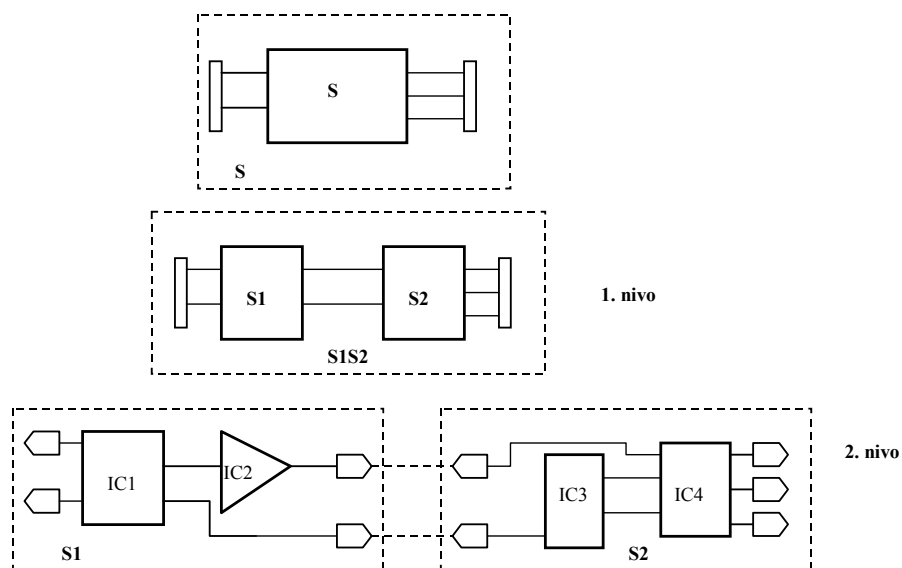


Slika 1.5 Enolistno organiziran načrt

Hierarhično organiziran načrt omogoča veliko preglednost tudi pri zelo kompleksnih sistemih. Temelji na splošno uporabnem principu skrivanja nepomembnih podrobnosti. Nekateri opisovalniki omogočajo tudi **večvariantne** sheme. Ker je struktura vezja narisana na več različnih plasteh, lahko s kombinacijami risalnih plasti tvorimo razne variante načrta.



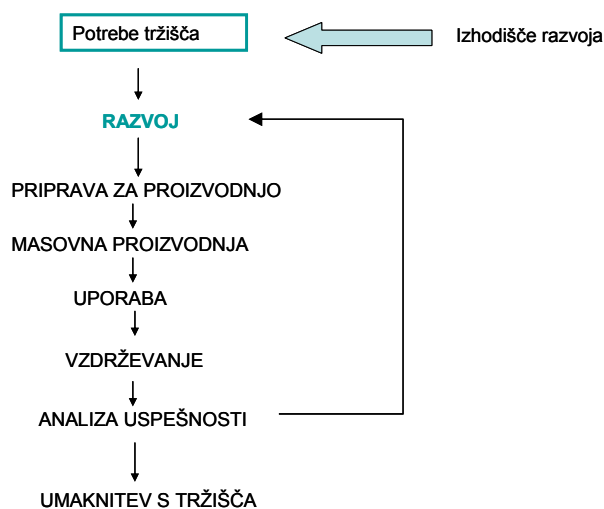
Slika 1.6 Vodoravno organiziran načrt iz slike 1.5



Slika 1.7 Hierarhično organiziran načrt iz slike 1.5

1.3. Življenjski cikel tehničnega sistema

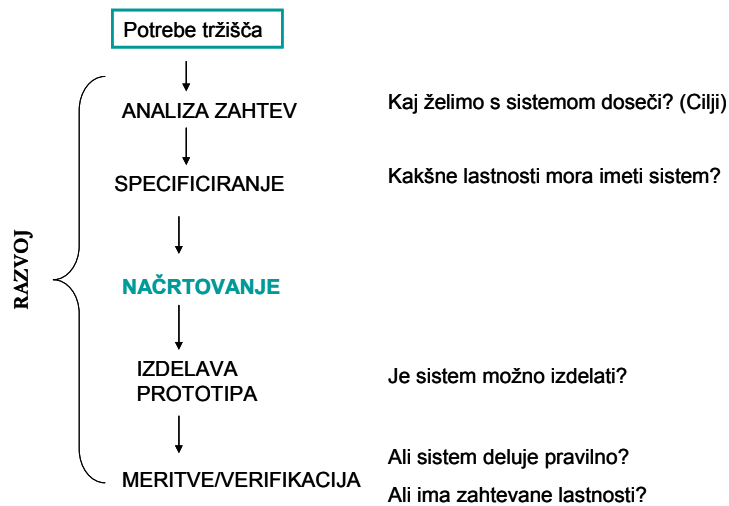
Življenjski cikel nekega produkta oziroma sistema se običajno začne z analizo potreb tržišča (slika 1.8). Temu sledijo razvoj, priprava na proizvodnjo in nato masovna proizvodnja. Na podlagi analize podatkov o okvarah, ki jih opažajo servisi, se lahko odločimo o potrebnih spremembah.



Slika 1.8 Tipični življenjski cikel tehničnega sistema

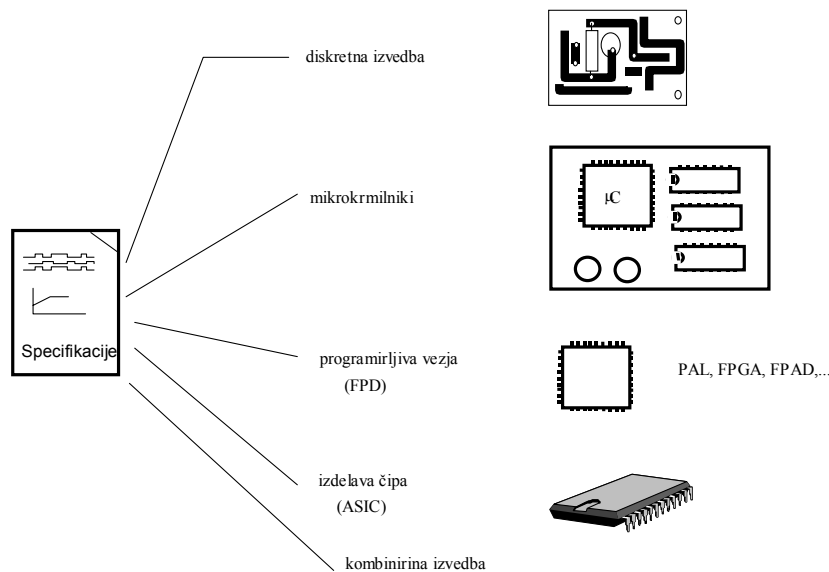
Slika 1.8 prikazuje podrobnejše faze razvoja nekega tehničnega sistema. Potek je odvisen predvsem od:

- načina implementacije (glej sliko 1.10) in števila izdelkov (unikat ali nekaj primerkov ali masovna proizvodnja),
- zahtevane kakovosti (visoka ali nizka) in
- resursov (denar, ljudje, znanje), ki jih imamo na razpolago.



Slika 1.9 Faze v razvoju sistema

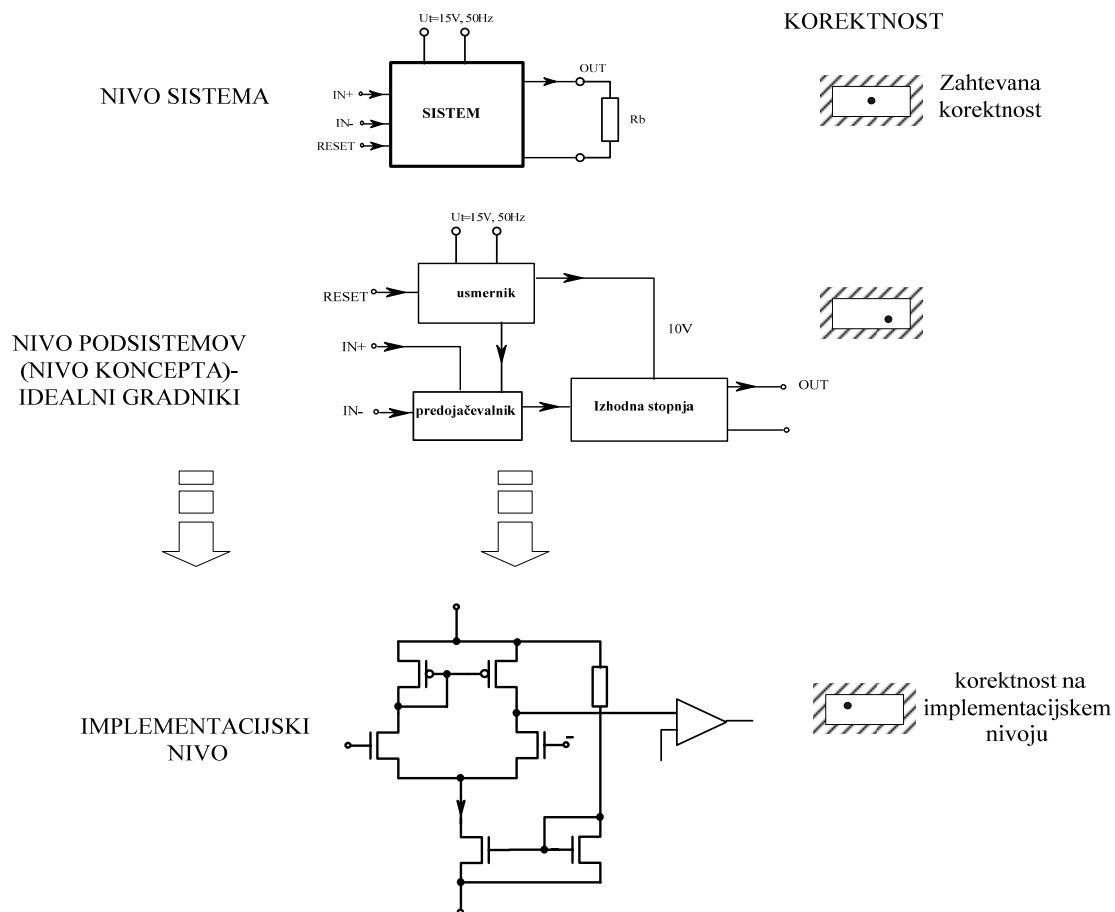
Potrebo po obstoju nekega izdelka v večini primerov narekuje tržišče, ki tudi definira globalne cilje oziroma zahteve. Te se podrobno analizirajo in nato pretvorijo v tehnične specifikacije, na podlagi katerih se začne načrtovanje. V nadaljevanju se bomo osredotočili samo na fazo načrtovanja.



Slika 1.10 Možni načini implementacije elektronskega sistema (FPD: Field Programmable Device, ASIC: Application Specific Integrated Circuits)

treba narediti določen kompromis. Na sam potek načrtovanja ima velik vpliv tudi vrsta implementacije (slika 1.10). **Če je le mogoče, naj bodo vse zahtevane lastnosti definirane kvantitativno. Numerični vrednosti mora biti vedno dodano tudi dopustno odstopanje (slika 1.12). Npr.: $P = (100 \pm 5)W$, $10 < A < 20$.**

Potem ko smo definirali specifikacije, se lahko začne načrtovanje. Najprej je treba opredeliti meje sistema in izbrati implementacijsko tehnologijo (slika 1.13). Vse, kar je zunaj sistema, že obstaja in ne spada v okvir načrtovanja. Npr. breme R_b že obstaja, prav tako velja za izmenično napetost 15V. V večini primerov nadaljujemo načrtovanje na konceptualnem nivoju, kjer uporabljamo **idealne** načrtovalske gradnike. Če na tem nivoju uspemo zadostiti specifikacijam (lastnosti so znotraj dopustnega območja), z načrtovanjem nadaljujemo, sicer moramo spremeniti koncept. Idealne gradnike zamenjujemo z bolj realnimi modeli in pri tem pazimo na **korektnost**⁴ (glej zgled na sliki 1.12). Postopek se konča, ko je vezje sestavljeno le iz implementacijskih gradnikov.



Slika 1.13 Načrtovanje se začne z definicijo mej vezja in konča, ko so lastnosti vezja znotraj dopustnega območja in je vezje sestavljeno samo iz implementacijskih gradnikov

Načrtovanje je torej proces, pri katerem na podlagi specifikacij in drugih zahtev (npr. standardov) tvorimo **implementacijski načrt**. V primeru programirljivih komponent še dodamo **programsko opremo**. Implementacijski načrt vsebuje vse podatke, ki so potrebni za izdelavo

⁴ Korektnost je optimalna oddaljenost nazivne vrednosti neke lastnosti od predpisanih mej.

sistema. V večini primerov je to načrt za tiskanino in vezalni načrt. V primeru integriranih vezij implementacijski načrt predstavljajo dimenzije mask.

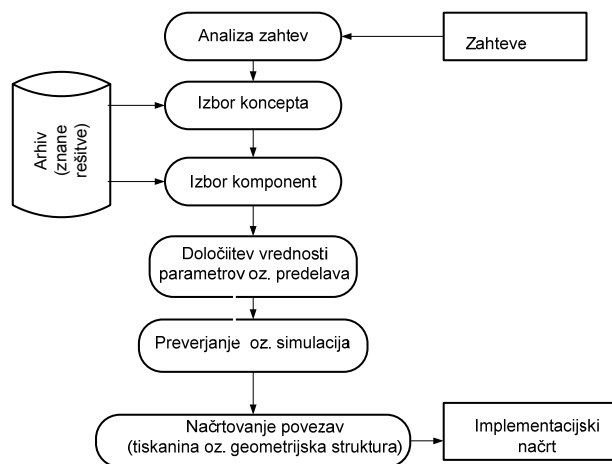
Če načrtujemo za masovno proizvodnjo, mora načrtovalec tvoriti tudi ustrezne **teste**, s katerimi se bo preverjala prisotnost tehnoloških napak⁵.



Slika 1.14 Pri masovni proizvodnji potrebujemo tudi teste za proizvodno linijo

1.4.1. Načrtovalski pristopi

Ne glede na način implementacije (slika 1.10) lahko načrtovalske pristope razdelimo v tri velike skupine: projektiranje, sintezo in kombinacijo obeh. **Predelava že obstoječega sistema** (načrta) oziroma **projektiranje** je najbolj preprosta in zato tudi najpogosteje uporabljena načrtovalska metoda (slika 1.15).



Slika 1.15 Postopek projektiranja

Postopek je zelo preprost: glede na zahteve poiščemo že preizkušen načrt vezja, izberemo ustrezne komponente in če je potrebno, vezje tudi ustrezno predelamo. Če gre za načrtovanje integriranih vezij, ta postopek imenujemo tudi **dimenzioniranje**, saj je naloga načrtovalca določiti ustrezne dimenzije tranzistorjev in drugih komponent. S projektiranjem lahko pridemo zelo hitro do končnega produkta, ki pa v nekaterih primerih (npr. masovna proizvodnja) ni optimalen. Lahko se tudi zgodi, da vezje, ki ga želimo, še ne obstaja, ker so zahteve zelo

⁵ Npr. nepravilno prispajkan tranzistor.

specifične. Ker so novejša vezja zelo pogosto patentirana oziroma avtorsko zaščitena, ta pristop zahteva dodatne stroške.

Kadar se izkaže, da projektiranje ne bo vodilo do zahtevanih rezultatov, se odločimo za **načrtovanje popolnoma novega sistema** oziroma za **sintezo**, ki pa je zelo zahtevna, saj zahteva ogromno znanja in sredstev. Pri načrtovanju integriranih vezij ta pristop imenujemo **načrtovanje po meri** (full custom design).

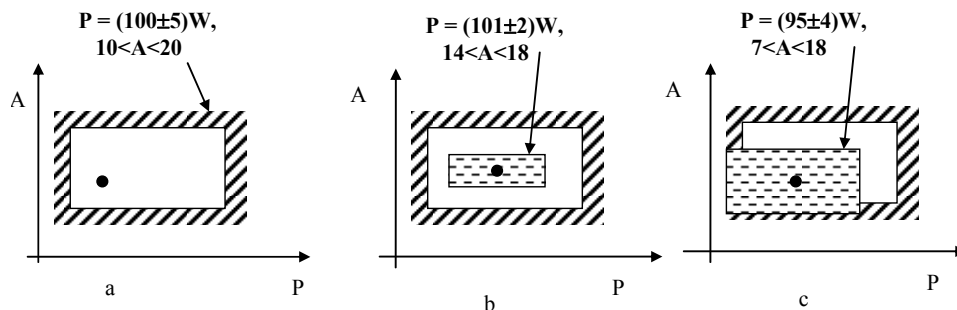
Cilji načrtovanja so odvisni od zahtev tržišča oziroma specifikacij. Tiste, ki se vedno ponavljajo, bomo poimenovali **univerzalni cilji**. Ti so:

- **čim nižja kompleksnost produkta,**
- **čim cenejši produkt,**
- **čim večja korektnost lastnosti,**
- **sistem naj bo sestavljen iz čim več enakih komponent ali elementov,**
- **čim večja robustnost⁶,**
- **čim hitrejša pot do končnega produkta.**

Na postopek načrtovanja vpliva tudi število vezij, ki jih nameravamo izdelati (ali načrtujemo unikatni primerek ali pa bomo vezje masovno proizvajali).

1.4.2. Načrtovanje za masovno proizvodnjo

Načrtovanje unikata je relativno enostavno. Takoj ko so lastnosti vezja znotraj dovoljenih mej, lahko z načrtovanjem končamo (glej zgled na sliki 1.16 a).



Slika 1.16 Izpolnjenost zahtev pri unikatu (a) in pri masovni proizvodnji (b), kjer bodo lastnosti izdelanih vezij ležale znotraj vodoravno črtkanega pravokotnika. Slika (c) prikazuje rešitev, ki pri masovni proizvodnji ne ustreza zahtevam. Nekateri ojačevalniki bodo imeli premajhno ojačenje A in prenizko moč P.

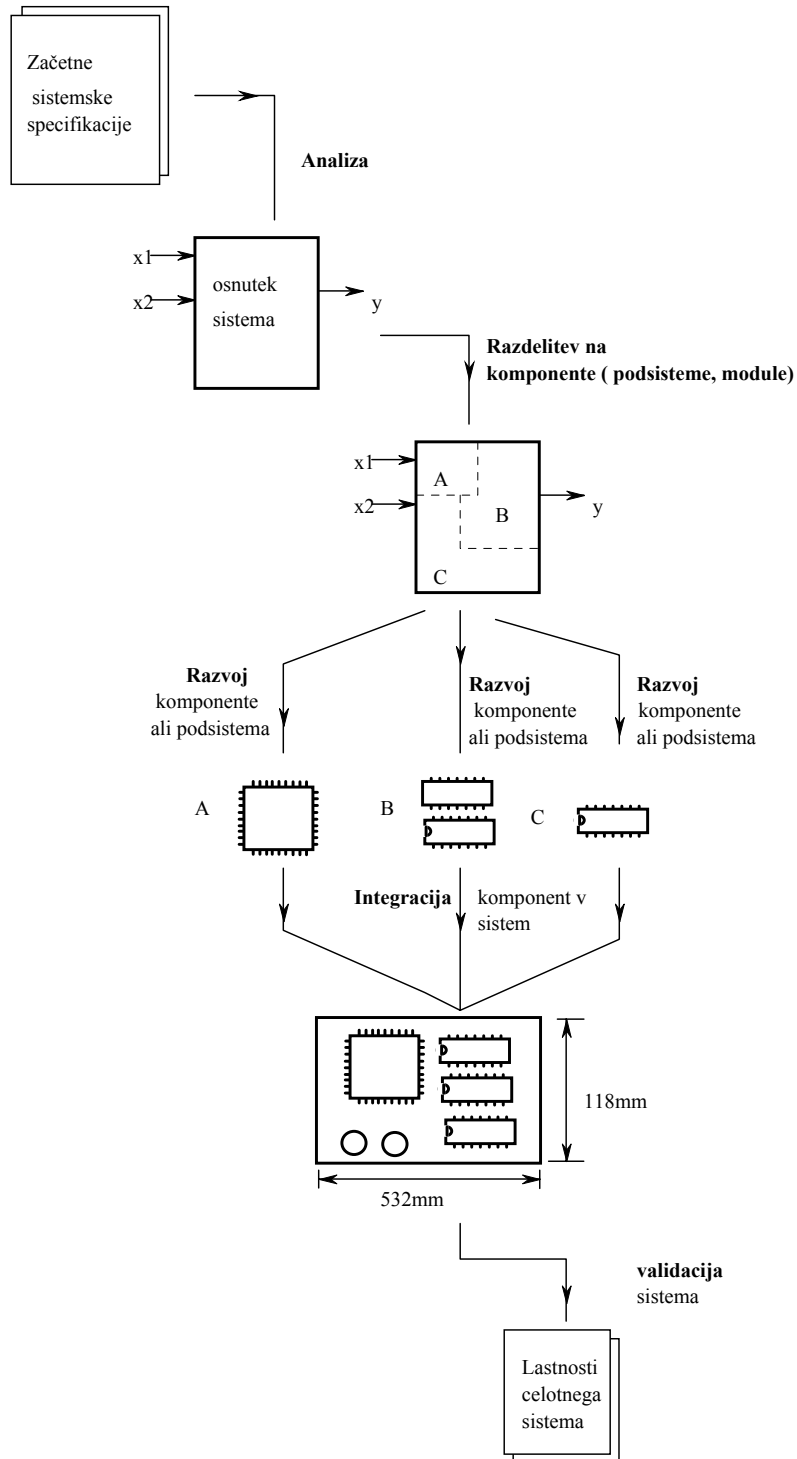
Načrtovanje za masovno proizvodnjo je težavnejše, saj moramo upoštevati tudi variabilnost lastnosti gradnikov. Vsi gradniki, s katerimi bomo sestavljali vezje, ne bodo imeli popolnoma enakih lastnosti, saj bodo te od primerka do primerka variirale. Npr. upor $1k\Omega \pm 10\%$ bo imel pri vsakem vezju za malenkost drugačno vrednost. Ker variacija lastnosti gradnikov vpliva tudi na lastnosti vezja, mora načrtovalec predpisati, kolikšne so lahko dopustne variacije gradnikov, ki še zagotavljajo, da bo vezje imelo lastnosti znotraj dovoljenih mej (glej zgled na sliki 1.16 b). S tem problemom se ukvarja disciplina, ki jo v elektroniki imenujemo analiza in načrtovanje toleranc⁷.

⁶ Robustnost je odpornost na variacijo gradnikov in postopka izdelave.

⁷ Več o tem je v učbeniku: T. Dogša: "CAE/CAD v elektroniki: Analiza in načrtovanje toleranc", učbenik, FERi, Maribor, 2007.

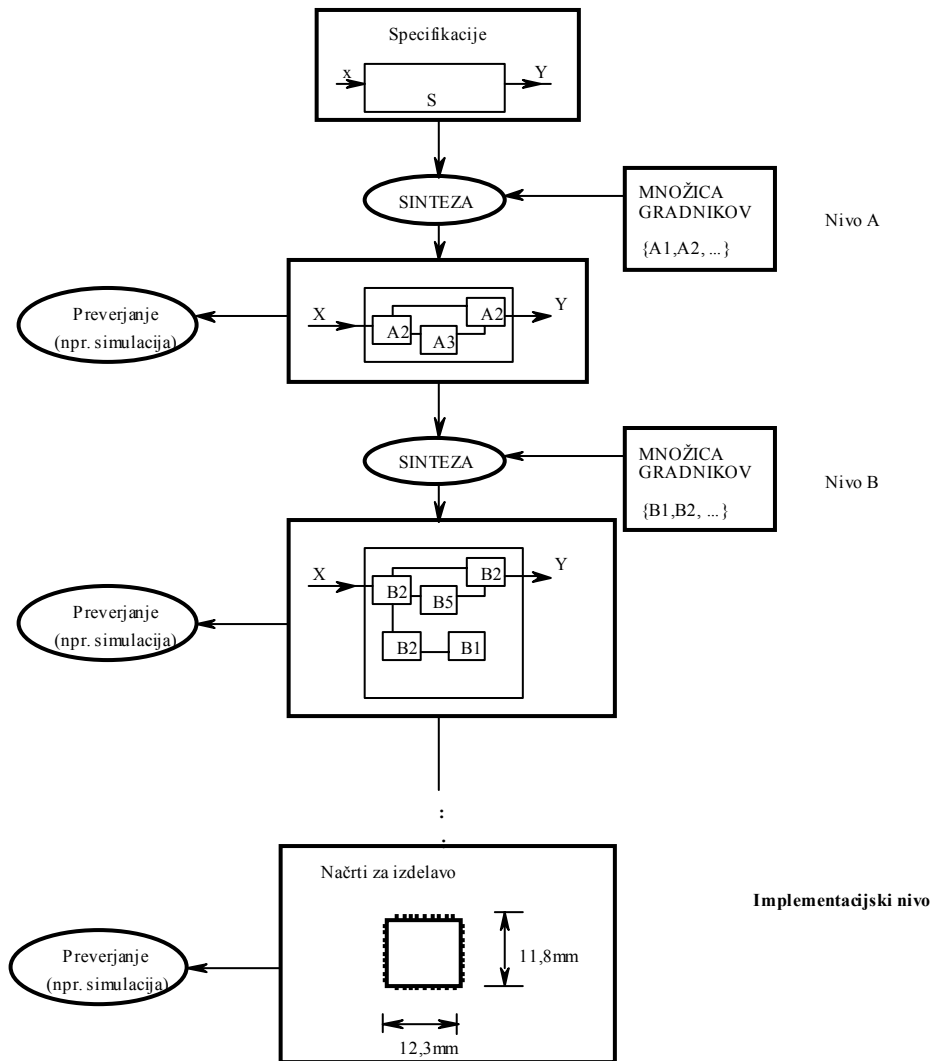
1.4.3. Načrtovanje kompleksnih sistemov

Kompleksni sistem je sestavljen iz velikega števila komponent, razvija ga skupina, zahtevana je kompleksna funkcionalnost, ki jo težko enostavno opišemo. Ker je celoten sistem zelo kompleksen, ga najprej razdelimo na več podsistemov ali modulov (slika 1.17) in nato vsakega razvijamo po modelu, ki je bil prikazan na sliki 1.23 oziroma 1.18.



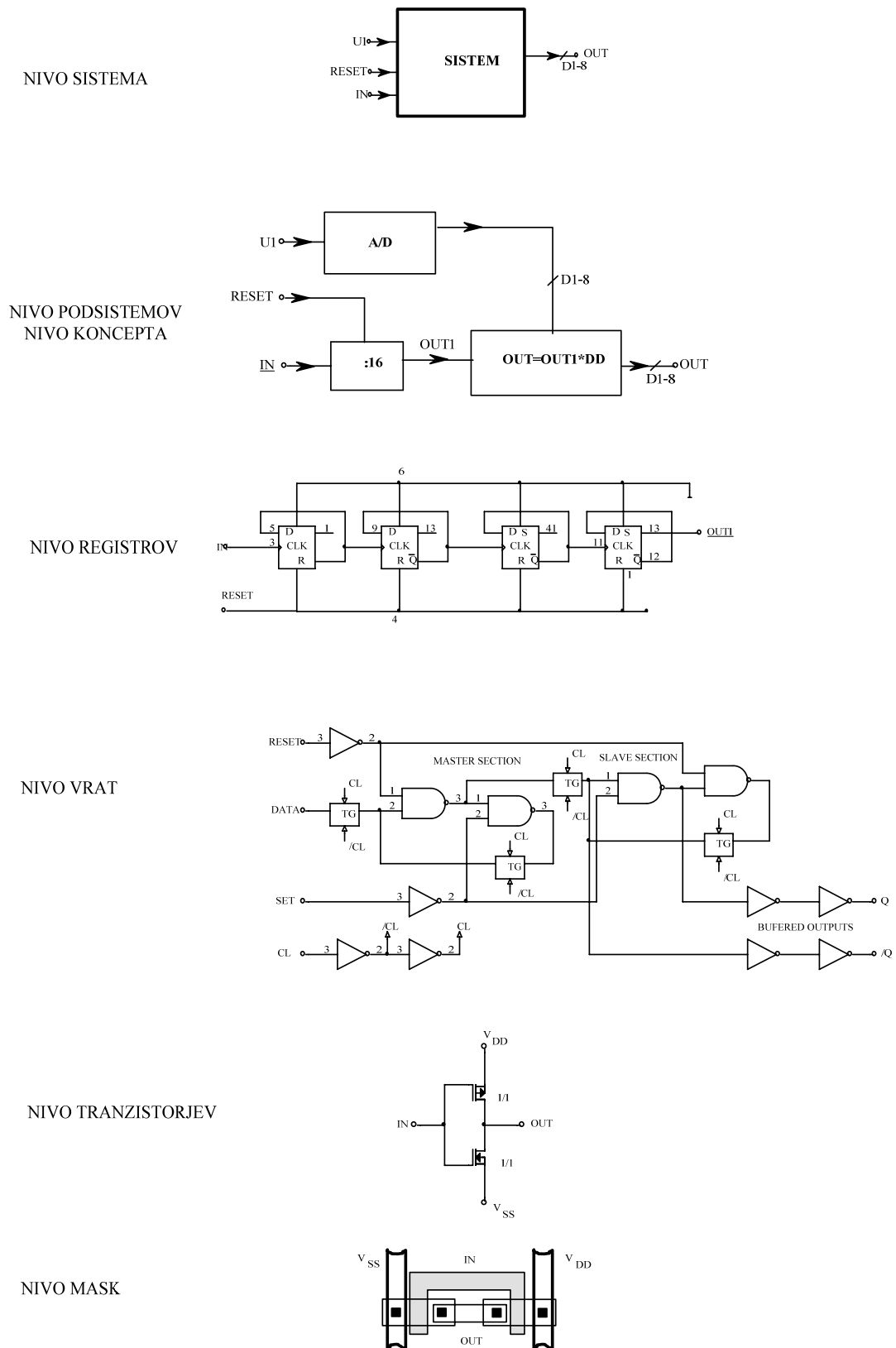
Slika 1.17 Potek načrtovanja kompleksnega elektronskega sistema

Potem ko so postavljene specifikacije za posamezne podsisteme, lahko začnemo njihov razvoj oziroma načrtovanje. Najbolj pogosta načrtovalska metoda, ki se uporablja pri načrtovanju kompleksnih sistemov, je **hierarhično načrtovanje** ali načrtovanje od vrha navzdol⁸ (glej sliko 1.18). Za zgled je prikazano načrtovanje integriranih vezij (slika 1.19). Načrtovanje se začne z opisom sistema, kjer definiramo njegove lastnosti, vhode in izhode. Nato sledijo sinteze na raznih nivojih, kjer se abstraktnost postopoma manjša in hkrati se večja konkretnost. Sinteza se konča, ko pridemo do implementacijskih gradnikov.



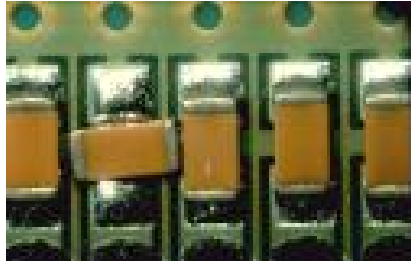
Slika 1.18 Hierarhična dekompozicija

⁸ Angl. top-down design.

Slika 1.19 Zgled načrtovanja čipa⁹ od zgoraj navzdol.⁹ Prikazan je primer kompletnega načrtovanja po naročilu (full custom).

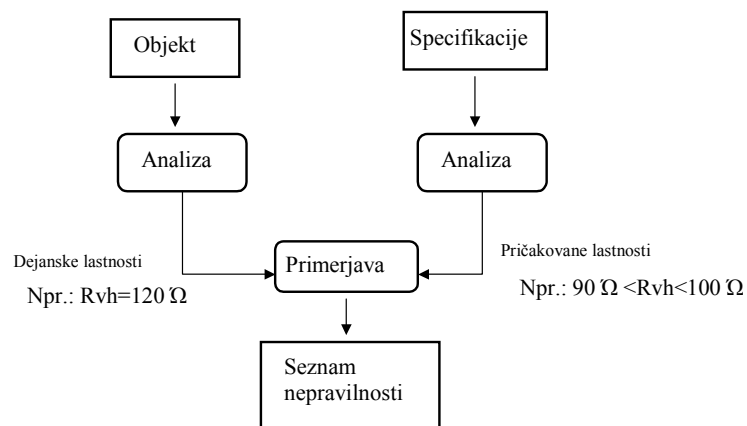
1.4.4. Preverjanje sistemov

Pri vsakem načrtovanju se srečamo z neko vrsto preverjanja, s katero ugotavljamo morebitno prisotnost določenih napak. Napake, ki so vzrok za rezultirajoče nepravilnosti, delimo v dve družini: **načrtovalske napake** (npr. napačno dimenzioniran transformator) in **tehnološke napake** (npr. nepravilno prispajkan tranzistor).



Slika 1.20 Zgled tehnološke napake

Preverjevalne metode oziroma testi so odvisni od tega, ali iščemo tehnološke ali pa načrtovalske napake. Npr. z analizo vezja lahko ugotavljamo samo prisotnost načrtovalskih napak, z meritvijo pa prisotnost tehnoloških in načrtovalskih napak. Za preverjevalne metode oziroma teste za ugotavljanje prisotnosti tehnoloških napak sta odgovorna načrtovalec in deloma tudi tehnolog.



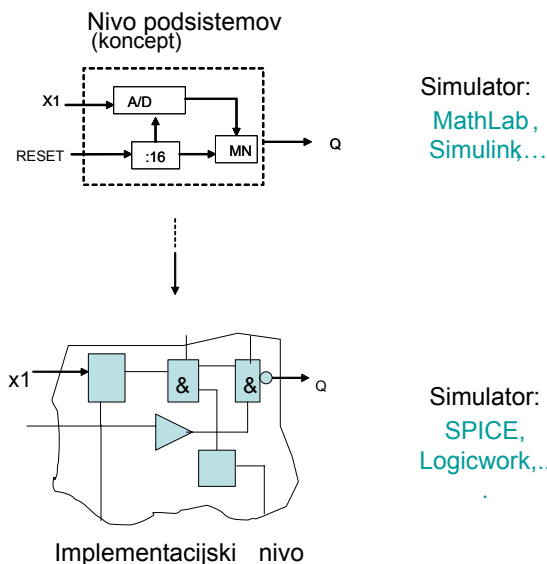
Slika 1.21 Model preverjanja

Tipične preverjevalne metode, ki jih uporabljajo načrtovalci, so:

- analiza oziroma formalno preverjanje pravilnosti,
- testiranje (meritve) in
- simulacija.

Vse vrste preverjanj ustrezajo modelu, ki ga prikazuje slika 1.21. Prisotnost ene ali več napak se kaže v odstopanju resničnih lastnosti od pričakovanih.

Ker stroški odpravljanja napak eksponentialno rastejo s časom, je treba s preverjanjem začeti čim prej (npr. na nivoju podsistemov), in ne šele na koncu razvoja. Preverjanje se običajno začne na konceptualnem nivoju (glej sliko 1.19 in sliko 1.22). Na tem nivoju preverjamo, ali smo izbrali pravi koncept. Tipični simulatorji za ta nivo so: VHDL simulator, Matlab oziroma Simulink. S simulacijo na implementacijskem nivoju preverjamo pravilnost načrtovanja. Dobljeni rezultati se lahko že zelo približajo poznejšim izmerjenim, vendar je simulacija v mnogih primerih zelo dolgotrajna.



Slika 1.22 Dva tipična nivoja, kjer preverjamo pravilnost načrtovanja

1.4.5. Razširjene načrtovalske aktivnosti

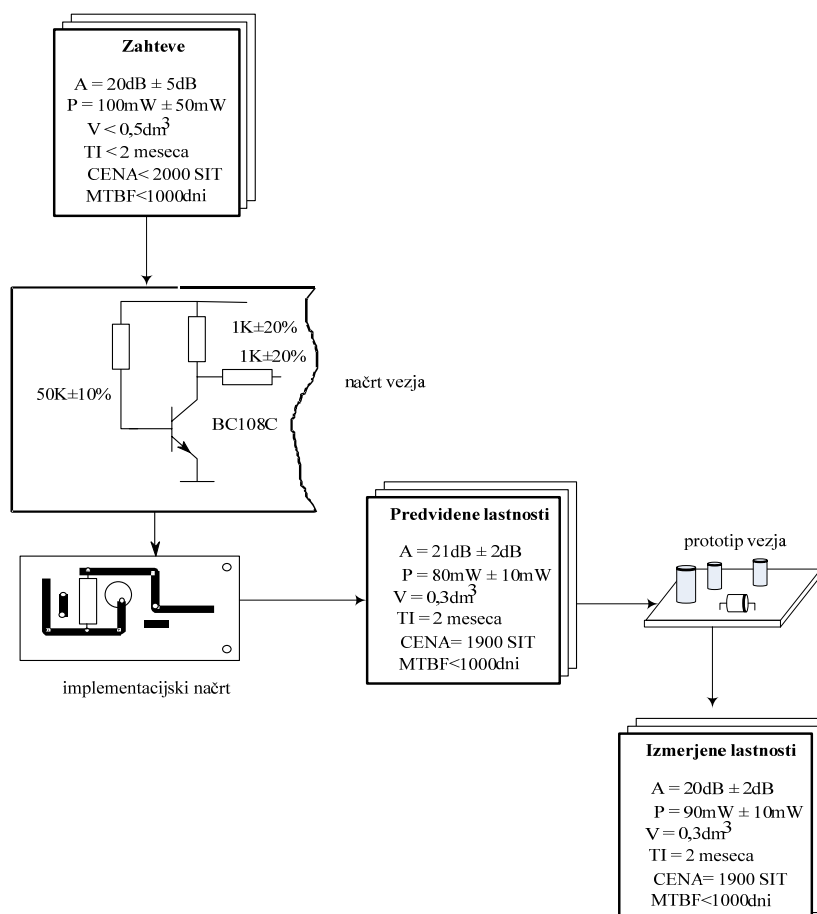
V najbolj enostavnem primeru načrtovanja imamo samo tri aktivnosti: **načrtovanje vezja brez upoštevanja toleranc**¹⁰, **načrtovanje implementacije in meritve**. Pot do načrta za množično proizvodnjo je zelo dolga, saj obsega približno 16 aktivnosti. Kateri koraki bodo uporabljeni, je odvisno od zahtevane kakovosti in predpisov. V nadaljevanju bo na kratko opisan postopek načrtovanja tipičnega elektronskega vezja, ki obsega vseh 16 aktivnosti¹¹.

1. **sinteza oziroma projektiranje in preverjanje na konceptualnem nivoju,**
2. **načrtovanje brez upoštevanja toleranc,**
3. preverjanje funkcionalne pravilnosti,
4. iskanje optimalnih parametrov,
5. analiza občutljivosti na spremembe,
6. načrtovanje toleranc,
7. preverjanje pravilnosti izbora toleranc (analiza toleranc),
8. analiza vpliva zunanjih dejavnikov,
9. **načrtovanje implementacije,**
10. analiza vpliva implementacije na lastnosti,
11. ocena zanesljivosti,

¹⁰ Pravimo mu tudi **načrtovanje z nominalnimi vrednostmi**.

¹¹ Vrstni red korakov je lahko tudi spremenjen.

12. FMEA analiza¹² (analiza možnih okvar in njihovih posledic),
13. načrtovanje testov za proizvodnjo,
14. analiza testabilnosti,
15. izdelava prototipa,
16. meritve.



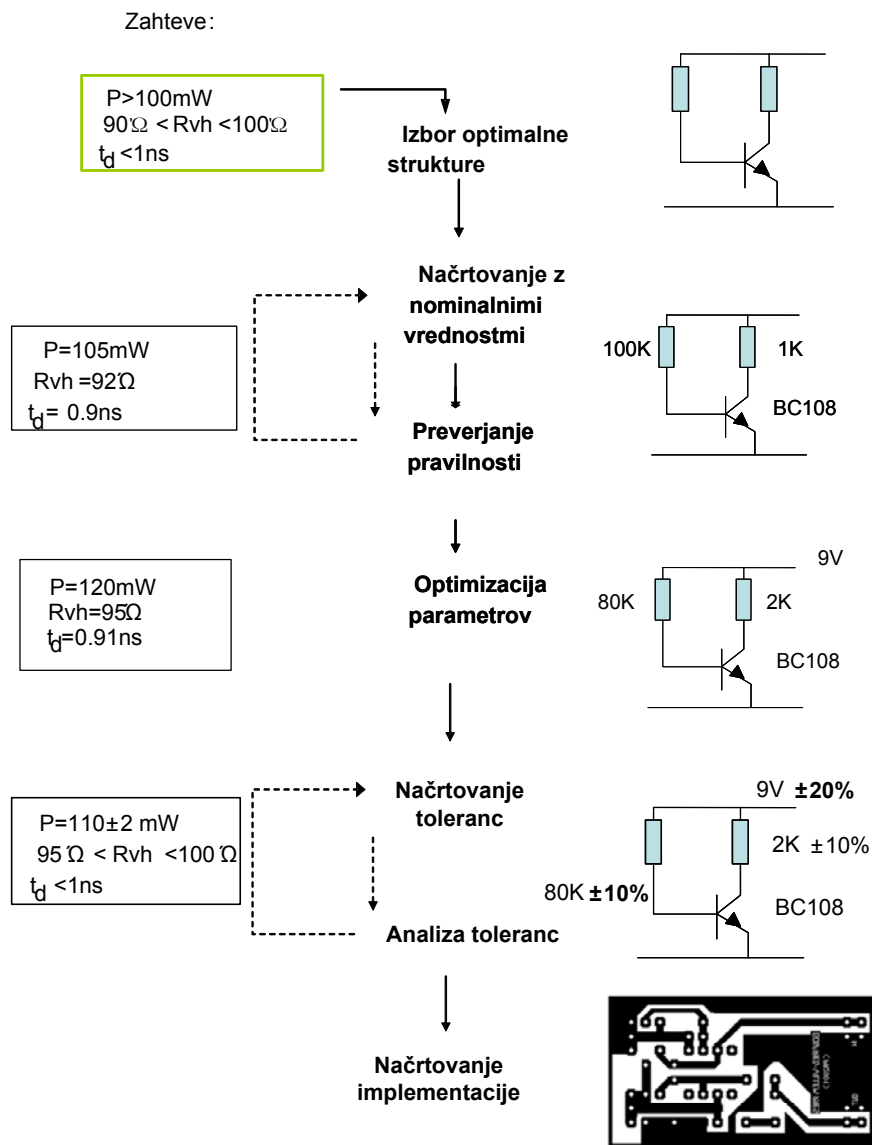
Slika 1.23 Najpomembnejši rezultati načrtovanja.

Načrtovanje se začne na konceptualnem nivoju, kjer gre za sintezo s poenostavljenimi (idealiziranimi) gradniki. Ko načrtovalec preveri¹³ pravilnost svoje ideje, nadaljuje z delom, vendar še ne upošteva nobenih toleranc. Pravilnost načrtovanja nato preveri "peš" ali pa s simulatorjem. Z optimizacijo poskuša ugotoviti, ali lahko s spremembo parametrov izboljša lastnosti vezja. Nato izvede analizo občutljivosti, ki mu pomaga pri zaokroževanju vrednosti parametrov, izboru komponent in poznejšemu načrtovanju toleranc parametrov. Sledi načrtovanje toleranc parametrov in nato preverjanje tega izbora. Ker načrtovalec nima vpliva na vse dejavnike (npr. temperatura, variacija napajalne napetosti itd.), mora preveriti, kako ti vplivajo na lastnosti. Po tem koraku nastane optimalni načrt. Sledita načrtovanje implementacije in analiza njenega vpliva na lastnosti, npr. vpliv parazitnih kapacitivnosti povezav na zakasnitev

¹² Failure Mode and Effect Analysis.

¹³ Tipični simulator, ki se uporablja na konceptualnem nivoju, je Matlab oziroma Simulink.

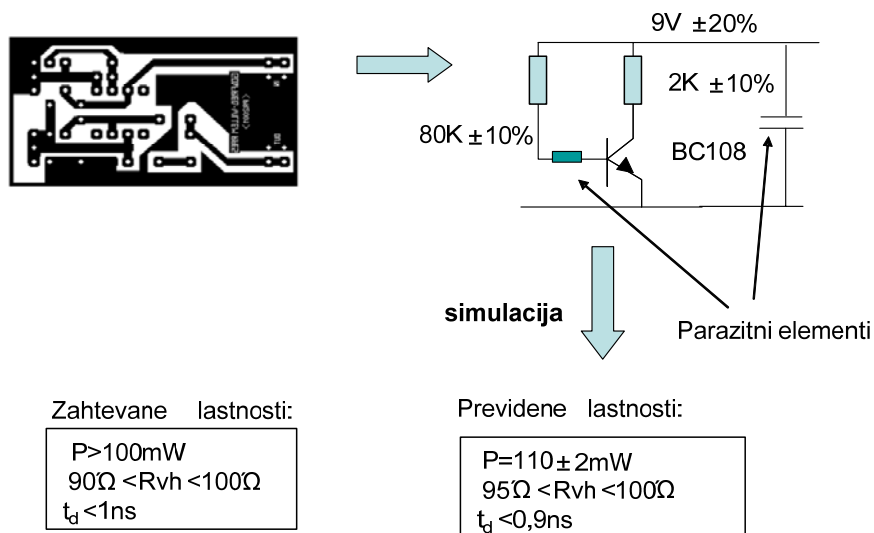
(glej sliko 1.25). Z oceno zanesljivosti ugotovljamo, kolikšna je verjetnost odpovedi. Z analizo možnih okvar določenih komponent preverjamo, kakšne so posledice teh okvar. V bistvu je to analiza varnosti sistema¹⁴. Ker bo med proizvodnjo treba preverjati, ali je vezje pravilno sestavljeno, so potrebni določeni testi. Učinkovitost teh testov preverjamo z analizo testabilnosti¹⁵. Nato sledijo meritve lastnosti, ki jih izvedemo na prototipu vezja. Le v idealnem primeru si koraki sledijo po vrstnem redu.



Slika 1.24 Od zahtev do implementacijskega načrta

¹⁴ Angl. system safety.

¹⁵ Z analizo testabilnosti ugotovljamo, ali je sistem mogoče preveriti brez velikih težav.



Slika 1.25 Analiza vpliva povezav na lastnosti vezja. Vpliv povezav proučujemo z dodajanjem parazitnih upornosti in kapacitivnosti.

1.5. Simulacija sistemov

Simulator je naprava ali program, ki je sposoben imitirati delovanje resničnega sistema. Ker omogoča ocenjevanje lastnosti vezja, ki še ni zgrajeno, je zelo pomembno načrtovalsko orodje. Idealen simulator tako dobro imitira vezje, da se vse napovedane lastnosti popolnoma ujemajo z izmerjenimi. V resnici s simulatorjem napovedujemo obnašanje in druge lastnosti bodočega sistema. Realen simulator je specializiran večinoma samo za opazovanje nekaj lastnosti (npr.: simulira samo tokove, napetosti, ne pa toplotnih razmer). Ker ne upošteva vseh vplivov, se rezultati razlikujejo od izmerjenih.

Tipični vzroki, zaradi katerih nastajajo odstopanja med izmerjenimi in simuliranimi rezultati, so:

Simulacija:

1. neustrezen model (če simuliramo),
2. v simulator smo vnesli napačne podatke,
3. izbrana je bila napačna analiza,
4. nepravilno delovanje simulatorja (malo verjetno),
5. močan vpliv okolja, ki ga nismo upoštevali (npr. temperature);

Meritev:

6. napačno izvedena meritev,
7. odpoved enega ali več elementov,
8. prevelik vpliv toleranc parametrov.

Simulatorje, ki jih uporabljamo v elektrotehniki, delimo na naslednje skupine:

Delitev glede na implementacijo:

- **fizični simulator** (analogni računalnik, pomanjšan model robota)
- **računalniški simulator** (program, ki teče na računalniku)

Delitev glede na obliko rezultatov:

- **numerični simulatorji** (rezultati so numerične vrednosti)
- **simbolični simulatorji** (rezultat je formula)

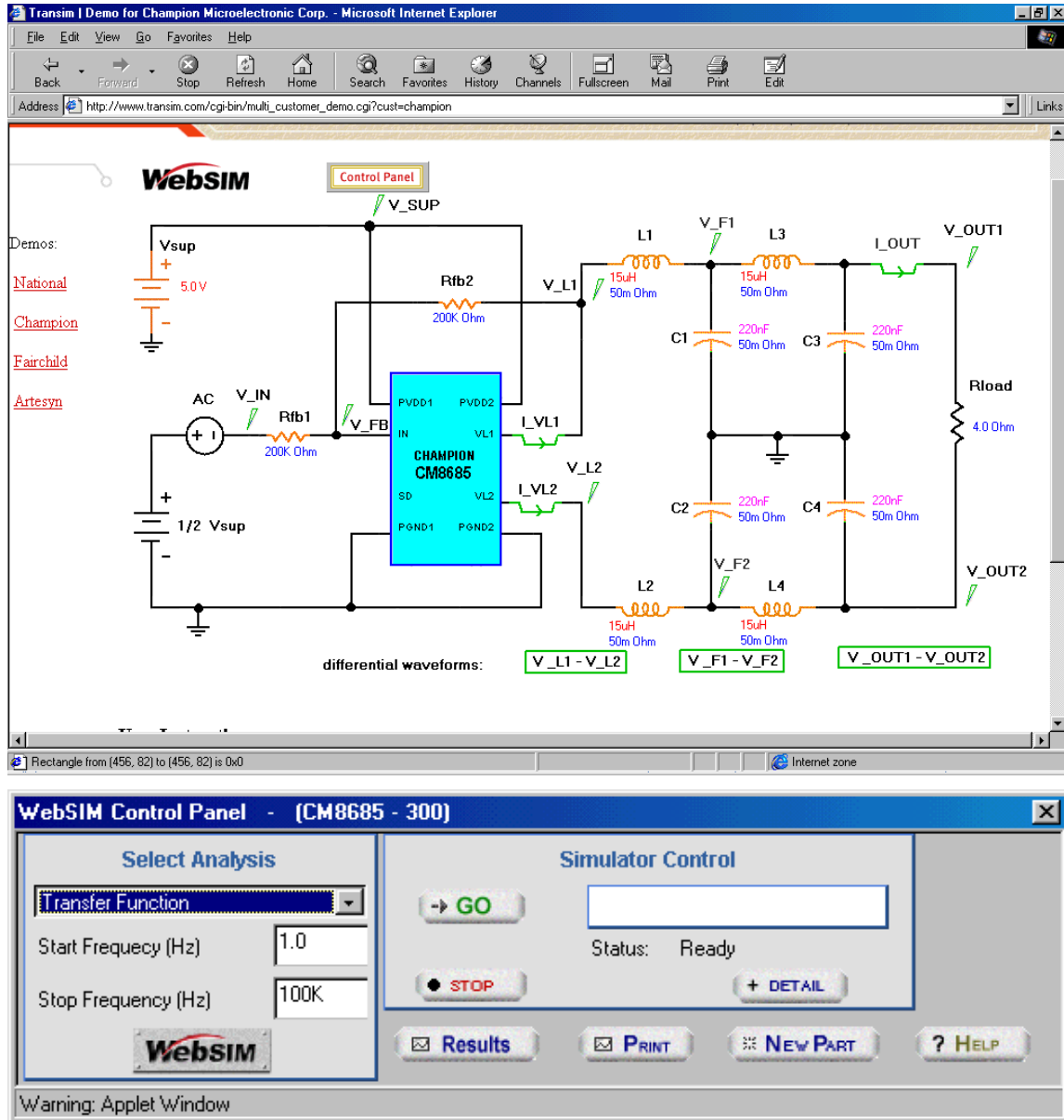
Delitev glede na splošnost:

- **namenski simulatorji** (npr. simulacija izgub v elektromotorju določene vrste, simulacija samo ene vrste ojačevalnika)
- **splošni simulatorji** (niso omejeni samo na eno vrsto vezja)

Delitev glede na vrsto signalov:

- **analogni simulator** (npr. SPICE)
- **digitalni simulator** (glej poznejša poglavja)
- **kombiniran simulator** (glej poznejša poglavja)
- **simulator integritete signalov** (glej poznejša poglavja)

Zanimivo rešitev predstavljajo spletni simulatorji, ki so specializirani za simulacijo aplikacij, ki se pojavljajo kot zgledi v strokovni dokumentaciji, v kateri so opisane lastnosti določenega elementa oziroma komponente¹⁶.



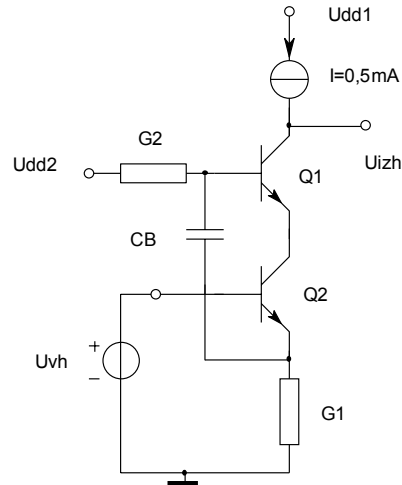
Slika 1.26 Spletni simulator¹⁷ uporablja za prenos vhodnih in izhodnih podatkov kar navaden brskalnik. Uporabnik lahko spreminja samo določene parametre. Simulacija teče na strežniku podjetja, ki je razvilo komponento.

¹⁶ Data sheet.

¹⁷ Vir: <http://www.transim.com>.

Zgled simboličnega analognega simulatorja¹⁸

Simulacija izhodne impedance kaskodnega ojačevalnika.

**Rezultat:**

$$Z_{izh} =$$

$$\begin{aligned} & G2 \text{ GM.Q1 GM.Q2} + G1 \text{ G2 GM.Q1} + G2 \text{ GM.Q2 GP.Q1} \\ & + G2 \text{ GM.Q1 GP.Q2} + \text{GM.Q2 GO.Q1 GP.Q1} + G1 \text{ G2 GP.Q1} \\ & + G2 \text{ GM.Q2 GO.Q1} + G2 \text{ GM.Q1 GO.Q2} + G1 \text{ GO.Q2 GP.Q1} \\ & + G2 \text{ GP.Q1 GP.Q2} + G1 \text{ GO.Q1 GP.Q1} + G1 \text{ G2 GO.Q2} \\ & + G1 \text{ G2 GO.Q1} + \text{GO.Q2 GP.Q1 GP.Q2} + \text{GO.Q1 GP.Q1 GP.Q2} \\ & + G2 \text{ GO.Q2 GP.Q1} + G2 \text{ GO.Q1 GP.Q2} + G2 \text{ GO.Q2 GP.Q2} \\ & + \text{GO.Q1 GO.Q2 GP.Q1} + G2 \text{ GO.Q1 GO.Q2} \end{aligned}$$

$$\begin{aligned} + j\omega CB & (\text{GM.Q1 GM.Q2} + G1 \text{ GM.Q1} + \text{GM.Q1 GP.Q2} + G2 \text{ GM.Q1} \\ & + G1 \text{ GP.Q1} + \text{GM.Q2 GO.Q1} + \text{GP.Q1 GP.Q2} + G1 \text{ GO.Q2} \\ & + G2 \text{ GP.Q1} + G1 \text{ GO.Q1} + \text{GO.Q1 GP.Q2} + \text{GO.Q2 GP.Q2} \\ & + \text{GO.Q1 GP.Q1} + G2 \text{ GO.Q1} + G2 \text{ GO.Q2} + \text{GO.Q1 GO.Q2}) \end{aligned}$$

$$\begin{aligned} \text{GO.Q1} & (G2 \text{ GM.Q2 GP.Q1} + G1 \text{ G2 GP.Q1} + G2 \text{ GP.Q1 GP.Q2} \\ & + G1 \text{ GO.Q2 GP.Q1} + G1 \text{ G2 GO.Q2} + \text{GO.Q2 GP.Q1 GP.Q2} \\ & + G2 \text{ GO.Q2 GP.Q2} + G2 \text{ GO.Q2 GP1}) \end{aligned}$$

$$\begin{aligned} + j\omega CB & (G1 \text{ GP.Q1} + \text{GP.Q1 GP.Q2} + G2 \text{ GP.Q1} + G1 \text{ GO.Q2} \\ & + \text{GO.Q2 GP.Q2} + G2 \text{ GO.Q2}) \end{aligned}$$

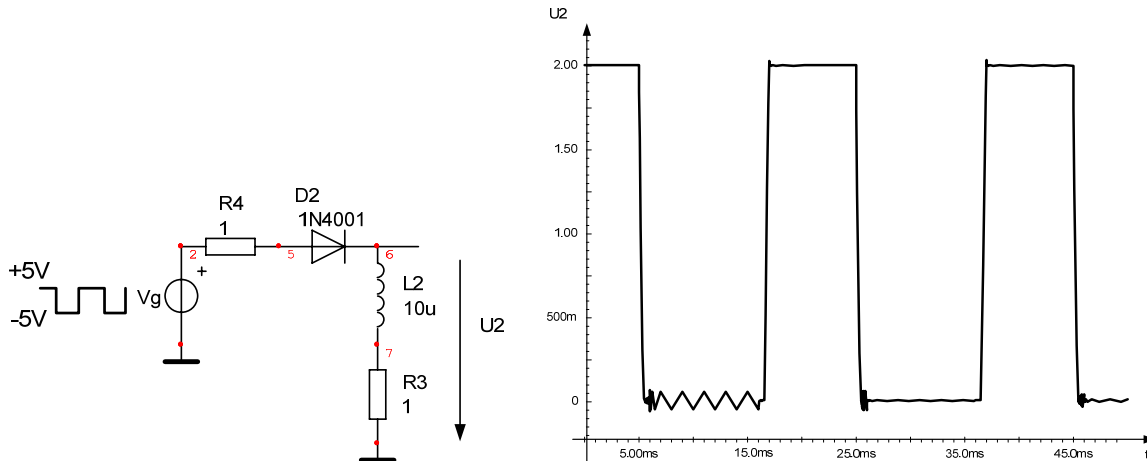
Poenostavitev (25 % napaka):

$$Z_{izh} = \frac{\text{GM.Q1 GM.Q2} (G2 + j\omega CB)}{\text{GO.Q1 GP.Q1} (G2 \text{ GM.Q2} + j\omega CB \text{ G1})}$$

¹⁸ ISAAC (*Interactive Symbolic Analysis of Analog Circuits*). Avtor: MICAS group, Katholieke Universiteit Leuven.

1.5.1. Sistematični pristop k simulaciji sistemov

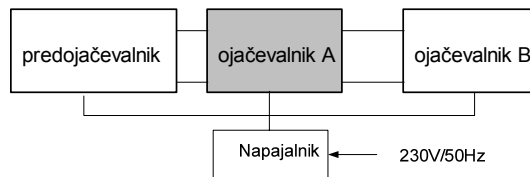
Napačni rezultati simulacije lahko zelo usodno vplivajo na razvoj in uspešnost nekega sistema. Če napak ne odkrijemo pri testiranju prototipov, se bodo pojavile pri končnih uporabnikih. Le sistematični pristop zmanjšuje verjetnost, da bomo pri simulaciji napravili kakšno napako.



Slika 1.27 Ali lahko zaupamo v pravilnost $u_2(t)$?

Tabela 1.3 Sistematičen pristop k simulaciji je sestavljen iz 10 korakov

Ime koraka	Ključno vprašanje v tem koraku
1. opredelitev ciljev simulacije	Kaj želimo s simulacijo izvedeti (cilji)? Katere lastnosti nas zanimajo?
2. izbor ustreznega simulatorja	Kateri simulator je najprimernejši?
3. določitev ustrezne analize	Katera analiza je najprimernejša?
4. izbor in testiranje ustreznosti modelov	Kateri model ustreza vsakemu elementu v sistemu?
5. ocenitev pričakovanih rezultatov (ocenimo interval)	Kako bomo ugotovili pravilnost (ustreznost) rezultatov?
6. priprava vhodnih podatkov	Kaj potrebuje simulator, da bomo lahko simulirali?
7. dodatno procesiranje rezultatov simulacije	Kako naj po simulaciji iz množice podatkov izločimo zahtevane rezultate?
8. simulacija	Kako naj poženem simulator?
9. preverjanje rezultatov simulacije	Ali lahko zaupamo v pravilnost rezultatov?
10. vrednotenje rezultatov	Ali smo dosegli zastavljene cilje?

Zgled:**1. opredelitev ciljev simulacije**

Radi bi ugotovili, ali lahko prekrmilimo ojačevalnik B. Ta bo prekrmljen, če bo na njegovem vходу amplituda večja kot 3 V. Treba je ugotoviti, kolikšna je maksimalna amplituda na izhodu ojačevalnika A.

2. izbor ustreznega simulatorja

Sistem, ki ga bomo simulirali, je analogno nelinearno vezje. Glede na zgornjo frekvenčno mejo 100 MHz in dimenzije ojačevalnika ga lahko obravnavamo kot koncentriran sistem. Najbolj primeren simulator je SPICE.

3. določitev ustrezne analize – Uporabili bomo tranzientno analizo.

4. izbor in testiranje ustreznosti modelov

Element	Model
Povezave in kontakti	vse povezave in kontakti so idealni
R, C	vsi upori in kondenzatorji so idealni
L	upoštevati je treba le enosmerno upornost
Tranzistorji	VF model, ki je v knjižnici SPICE.LIB
Napajalnik	idealni napetostni vir.
Breme	idealni upor R od 1 K do 10 K
Predojačevalnik	idealni napetostni vir za majhne izmenične signale
Vpliv okolice	vpliv zanemarimo

V našem primeru je najbolj kritičen model tranzistorja. S simulacijo meritev bomo preverili njegovo karakteristiko.

5. ocenitev pričakovanih rezultatov

Ocenimo interval, v katerem se bo nahajala amplituda. Glede na napajalno napetost amplituda ne sme preseči 5 V. Spodnja meja je približno 1 V.

6. priprava vhodnih podatkov

Opišemo vezje, dodamo potrebne modele. Potrebujemo VF model tranzistorja BF 242. Če topološka struktura ni v skladu s simulatorjevimi pravili, jo ustrezno spremenimo. Pri tem pazimo, da sprememba bistveno ne vpliva na lastnosti.

7. dodatno procesiranje rezultatov simulacije

V mnogih primerih simulator direktno ne izračuna tega, kar nas zanima. Potrebno je dodatno računanje. V našem primeru bi lahko napisali skript, ki bi iz simulatorjevih rezultatov izločil maksimalno izhodno napetost ojačevalnika A.

8. simulacija

Izvedemo simulacijo in poiščemo zahtevane podatke: $U_{max} = 4,2$ V.

9. preverjanje rezultatov simulacije

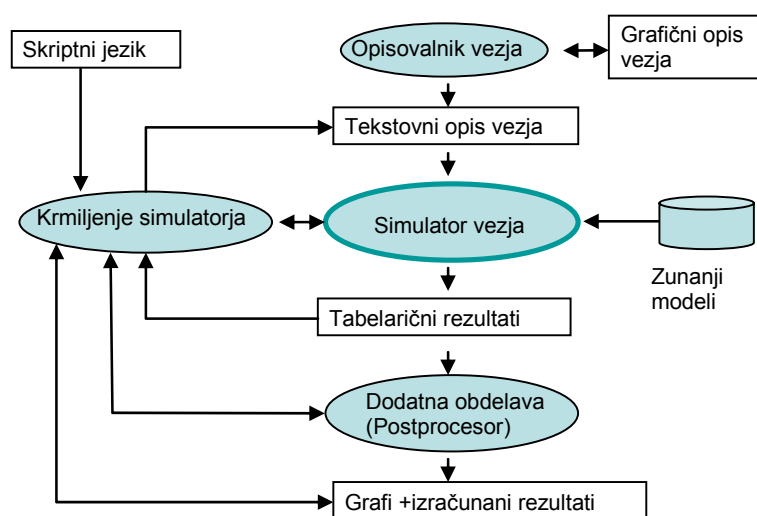
Glede na točko 5. lahko zaupamo v pravilnost rezultata.

10. vrednotenje rezultatov

Ojačevalnik B bo zagotovo prekrmljen.

2. SIMULACIJA ANALOGNIH VEZIJ

Prvi začetki simulatorjev segajo v obdobje med letoma 1960 in 1970, ko so se začeli pojavljati prvi uporabni simulatorji (ECAP, SYSCAP, CANCER, NET-2, SPICE). Iz te skupine se danes najbolj uporablja SPICE. Čeprav je simulator SPICE (*Simulation Program with Integrated Circuit Emphasis*) specializiran predvsem za simulacijo integriranih vezij, je uporaben za simulacijo vseh vrst električnih vezij. Danes obstaja veliko število različic tega simulatorja, katerega jedro so razvili leta 1972 na kalifornijski univerzi v Berkeleyu (University of California). V tem poglavju bodo opisana kratka navodila za uporabo simulatorja SPICE4. Podrobnejša navodila so dostopna v simulatorju s klicem pomoči (Help) ali pa v literaturi¹⁹.

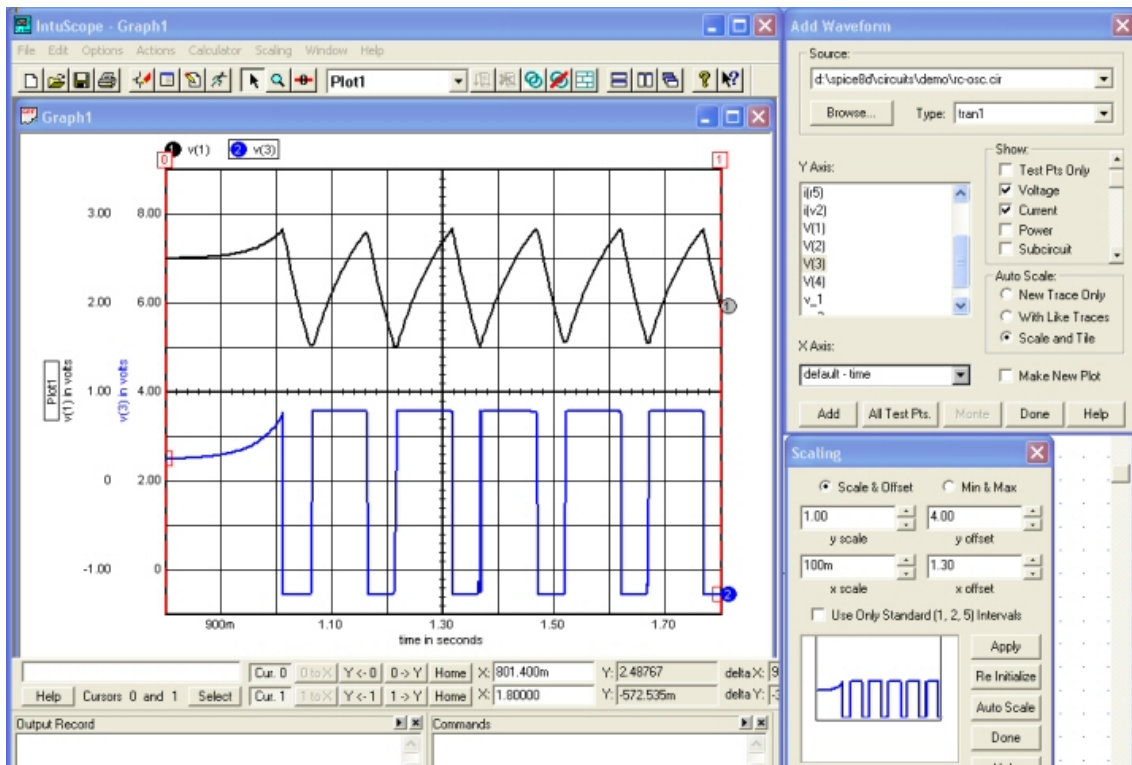


Slika 2.1 Tipična struktura kompletnega simulatorja

Tipičen programski paket je sestavljen iz štirih programov:

- **Opisovalnik vezja** – z njim opisujemo strukturo vezja
- **Simulator** – izvede simulacijo
- **Grafični postprocesor** – omogoča dodatno obdelavo rezultatov
- **Krmilnik analiz** – z njim lahko krmilimo simulator in na ta način izvajamo zahtevnejše analize

¹⁹ Npr. T. Tuma: "Analiza vezij s programom SPICE3", Fakulteta za elektrotehniko v Ljubljani, Ljubljana 1997.



Slika 2.2 Grafični postprocesor – Intuscope od firme IntuSoft

2.1. Opis vezja

Simulator lahko simulira le vezja, ki so sestavljena iz idealnih elementov. Zato je treba vsak realni element v konkretnem vezju zamenjati z ustreznim modelom (več o tej problematiki je v poglavju Modeliranje). Modele, ki jih imamo na razpolago za opis vezja, lahko razdelimo v dve skupini:

- **notranji modeli** – implementirani so v programski kodi kot funkcije oziroma podprogrami. Lastnosti takega modela lahko spreminjamo le s parametri.
- **zunanj modeli** – implementirani so kot vezje. Lastnosti modela lahko spreminjamo s parametri in s spremembo strukture.

Problematika modeliranja bo podrobneje obdelana v posebnem poglavju Modeliranje. Najprej bomo obravnavali le taka vezja, katerih elemente lahko modeliramo samo z notranjimi modeli.

Opis vezja sestavljajo topološki podatki in podatki o modelih elementov. Topološki del opisa vezja mora zadostiti naslednjim pravilom:

1. Vsa vozlišča morajo biti oštevilčena z naravnimi števili²⁰. Vrstni red ni pomemben. Vozlišče 0 je vedno masa.
2. Na eno vozlišče morata biti priključena najmanj dva elementa.

²⁰ Novejše verzije dopuščajo alfanumerična imena vozlišč, npr. Izhod_A.

3. Iz vsakega vozlišča mora obstajati galvanska povezava do vozlišča 0.
4. Zanke, v katerih so sami napetostni viri in/ali tuljave, so prepovedane.
5. Če so v vezju elementi, ki jih simulator ne pozna (niso primitivni elementi), jih moramo zamenjati z ustreznim zunanjim modelom.
6. Vsak element mora imeti svoje ime (oznako), ki se razlikuje od ostalih. Prva črka v imenu je odvisna od vrste primitivnega elementa (glej tabelo 2.1). Npr. oznaka vsakega upora se mora začeti s črko R.

Tabela 2.1 Tipični elementi, ki so modelirani z notranjimi modeli, in njihove značilne prve črke. S črko X je označen zunanji model.

B	nelinearni rezistivni element (napetostno ali tokovno krmiljen)
C	kondenzator
E	napetostno krmiljen napetostni vir
F	tokovno krmiljen tokovni vir
G	napetostno krmiljen tokovni vir
H	tokovno krmiljen napetostni vir
I	neodvisni tokovni vir
K	sklopljena induktivnost
L	tuljava
O	prenosna linija z izgubami
R	upor
S	napetostno krmiljeno stikalo
T	prenosna linija brez izgub
U	porazdeljena URC/URD prenosna linija
V	neodvisni napetostni vir
W	tokovno krmiljeno stikalo
D	dioda
Q	bipolarni tranzistor
J	FET
Z	MESFET
M	MOSFET

X	zunanji model

Opis elementov oziroma notranjih modelov je sestavljen iz topoloških podatkov in vrednosti parametrov, s katerimi določimo njihove lastnosti. Za tiste elemente, ki imajo majhno število parametrov, se uporablja krajši način opisa, ki obsega samo eno vrstico.

Krajši način opisa elementa

<Značilna črka>[dodatni znaki] <n1> <n2> <n3> ... <p1> <p2> ...[začetni pogoji]

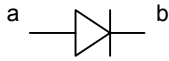
n1, n2, ... vozlišča
p1, p2, ... parametri

Če uporabljamo nelinearni vir z oznako B, so dovoljeni tudi aritmetični izrazi.

Zgled:

2.2. Opisi najpogostejših elementov

Dioda

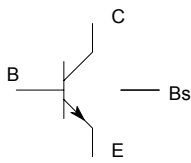


Oznaka notranjega modela : D.

D<ime> <a> <ime modela>²¹

Zgled: D6 1 4 D1N222

Bipolarni tranzistor

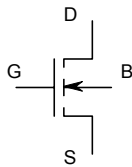


Oznaka notranjega modela : NPN za npn in PNP za pnp tranzistor.

Q<ime> <C> <E> [Bs] <ime modela>²²

Zgled: Q25 34 2 7 BC108

MOS tranzistor

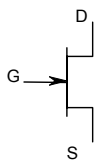


Oznaka notranjega modela : NMOS ali PMOS.

M<ime> <D> <G> <S> <ime modela>²³

Zgled: Mout 1 4 0 5 ICN

FET tranzistor



Oznaka notranjega modela : p-kanalni PJF, n-kanalni NJF.

J<ime> <D> <G> <S> <ime modela>²⁴

Zgled: Jpp 44 13 22 FET

²¹ Dodatni parametri, ki se redkeje uporabljajo: [površinski faktor] [OFF] [IC=<U0>]

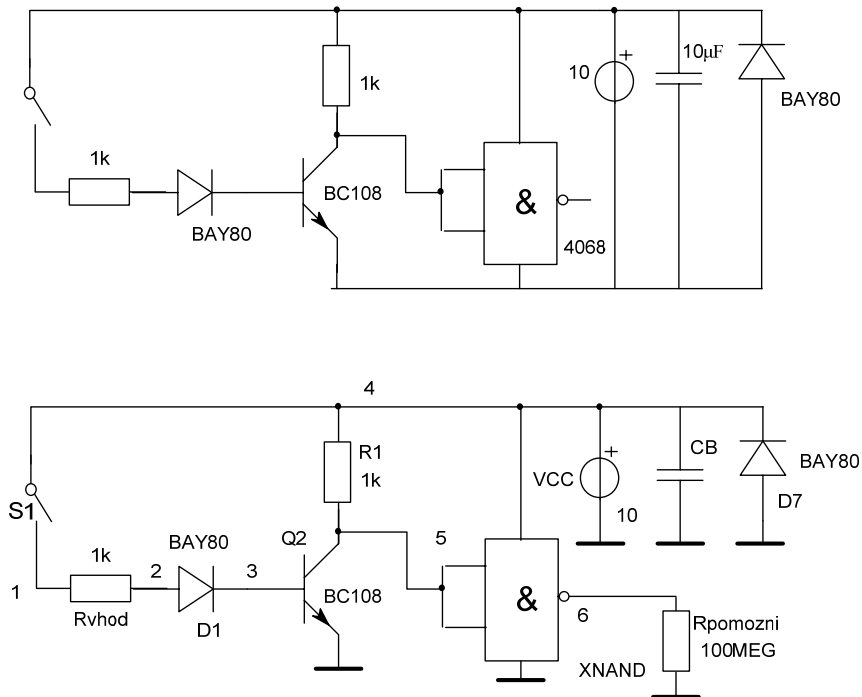
²² Dodatni parametri, ki se redkeje uporabljajo: [površ. faktor] [OFF] [IC=<UBE , UCE>]

²³ Več o tem modelu je v poglavju Modeliranje.

²⁴ Dodatni parametri, ki se redkeje uporabljajo, so: [površinski faktor] [OFF] [IC=<UDS , UGS>]

Opis vezja napravimo po naslednjih korakih:

1. Vezje ustrezno spremenimo, tako da je v skladu s topološkimi pravili²⁵. Dodane elemente ustrezno označimo s komentarjem ali pa z načinom imenovanja.
2. Za vsak element poiščemo ustrezen notranji oziroma zunanji model.
3. Elemente ustrezno označimo in vozlišča oštevilčimo.

Zgled:

Slika 2.3 Zgled originalnega vezja (zgoraj) in za simulacijo pripravljenega vezja (spodaj) z oštevilčenimi vozlišči. Ker model NAND vrat potrebuje breme na izhodu, smo dodali upor 100MEG.

²⁵ Najpogostejši vzrok za topološke nepravilnosti so preveč idealizirani modeli.

2.3. Struktura vhodne datoteke za simulator SPICE

Vhodna datoteka na sliki 2.1 (glej str. 31) ima naslednjo strukturo.

Naslovna vrstica

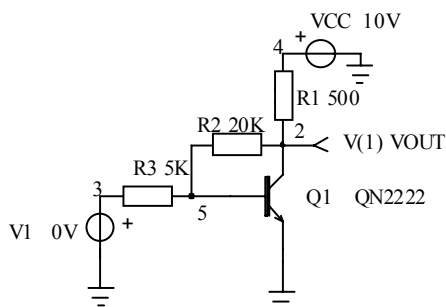
Krmilni blok (segment s posebnimi ukazi)

Ukazi (segment z ukazi za razne analize)

Opis vezja

Opis modelov

Zgled:



```
ZGLED - RNV ZA VERZIJO ISSPICE 4
.CONTROL
  echo ----- delovna točka -----
  op
  show all
  save allcur allpow
.ENDC
```

```
* UKAZI
.DC VIN -3 3 0.1
.TRAN 5NS 1000NS
.TF V(2) VIN
.AC DEC 20 100 100MEG
.PRINT AC VDB(2) VP(2)
.PRINT DC V(2)
.PRINT TRAN V(2) V(3)
*ALIAS V(2)=VOUT
*ALIAS V(3)=VIN
```

```
* OPIS VEZJA
R2 2 5 20K
R3 5 3 5K
VIN 3 0 0V AC 1 PULSE 0 5V 10NS 0 0 200NS AC 1
VCC 4 0 10V
Q1 2 5 0 QN2222
R1 2 4 500
```

```
.MODEL QN2222 NPN(IS=1.9E-14 BF=150 VAF=100 IKF=.175 ISE=5E-11 NE=2.5
+ BR=7.5 VAR=6.38 IKR=.012 ISC=1.9E-13 NC=1.2 RC=.4 XTB=1.5
+ CJE=26PF TF=.5E-9 CJC=11PF TR=30E-9 KF=3.2E-16 AF=1.0)
.END
```

2.4. Osnovne analize

Simulatorje električnih vezij najpogosteje uporabljamo za ugotavljanje funkcionalne pravilnosti, kjer preverjamo, ali ima vezje predpisane lastnosti. Tipični simulator SPICE pozna niz ukazov, s katerimi zelo enostavno izvedemo določeno analizo.

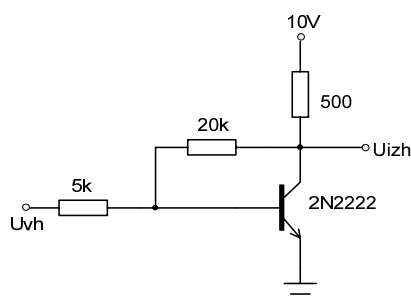
.OP	analiza delovne točke
.TRAN	analiza prehodnega pojava (tranzientna analiza)
.DC	analiza enosmerne karakteristike: prenosna, vhodna, izhodna karakteristika
.AC	frekvenčna analiza (izmenična analiza)
.FOUR	Fourierjeva analiza (harmonska analiza)
.DISTO	analiza popačenj
.NOISE	šumna analiza
.TEMP	temperaturna analiza
.TF	enosmerna analiza majhnih sprememb okrog delovne točke (diferencialna prenosna funkcija)
.SENS	analiza občutljivosti na vejni tok ali vozliščno napetost (diferencialna občutljivost)

Ker izbor prave analize predstavlja za začetnika velik problem, bo pomen vsake analize v nadaljevanju na kratko razložen. Uporabo najpogostejših analiz bomo ilustrirali na naslednjem preprostem zgledu.

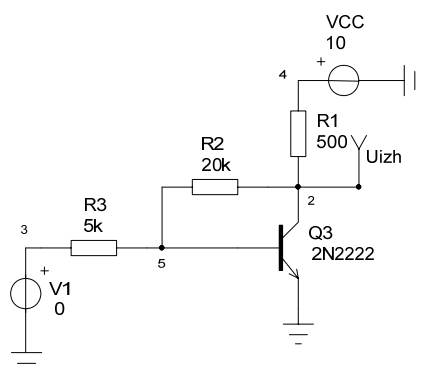
Zgled:**Opis problema**

S pomočjo simulatorja določi:

1. Enosmerne tokove in napetosti.
2. Strmino transkonduktančne karakteristike v delovni točki
3. Napetostno ojačenje v delovni točki za majhne enosmerne signale.
4. Kateri elementi najbolj vplivajo na delovno točko (U_{CE0})?
5. Prenosno enosmerno karakteristiko.
6. Zgornjo frekvenčno mejo.
7. Potek napetostnega ojačenja v odvisnosti od frekvence.
8. Hitrost odziva na pravokotni impulz.
9. Vhodno dinamično upornost.
10. Popačenje ojačevalnika pri vhodni amplitudi 3 V.
11. Prenosno karakteristiko in U_{CE} v odvisnosti od temperature za območje -40 do 70 °C

**Priprava vezja**

Vezje najprej predelamo, kot zahtevajo pravila za simulator: označimo elemente in vstavimo merilne (testne) sonde²⁶ za izpis napetosti in toka.



²⁶ To so grafični objekti, ki jih lahko poljubno poimenujemo, potem ko jih priključimo na vozlišče. Na zgornji sliki je to Uizh.

Napravimo poenostavljeno analizo vezja in ocenimo posamezne vrednosti iskanih veličin:

1. Tokovi delovne točke bodo zagotovo manjši od toka nasičenja tranzistorja:
 $I = V_{CC}/R_1 = 10 \text{ V}/500 \ \Omega = 20 \text{ mA}$.
 Napetosti delovne točke tranzistorju: $U_{BE} \cong 0,7 \text{ V}$ in $0,1 \text{ V} < U_{CE} < 10 \text{ V}$.
2. Maksimalna strmina transkonduktančne karakteristike bo približno
 $g = 30 \cdot I_{cmax} = 30 \cdot 20 = 600 \text{ mA/V}$.
3. Napetostno ojačenje bo približno $A_u = -R_2/R_3 = -4$.
4. Prenosna enosmerna karakteristika: na sredini področja bo naklon približno -4 .
 Maksimalne napetosti bodo $U \cong U_{CE} \cong 5 \text{ V}$, minimalne pa $U \cong U_{CE} \cong U_{CES}$.
5. Zgornja frekvenčna meja bo glede na vrsto tranzistorja približno
 $1 \text{ MHz} < f_{zg} < 200 \text{ MHz}$.
6. Hitrost odziva na stopnico bo v območju nanosekund.
7. Ker ima ojačevalnik napetostno-tokovno povratno vezavo, bo vhodna dinamična upornost približno enaka R_3 .

Opis vezja z ukazi

Spodaj je prikazana vsebina vhodne datoteke, ki jo prebere simulator SPICE (posamezni ukazi bodo v nadaljevanju posebej razloženi).

```
E: \IS SPICE 2004\ojacevalnik.cir Delovna tocka
.control
  save V(3) @V1[i] @V1[p] V(5) @R3[i] @R3[p] V(2) V(4)
+ @R1[i] @R1[p] @R2[i] @R2[p] @VCC[i] @VCC[p] @Q3[icc] @Q3[p]
  alias uizh
  op
  show all : all
  showmod all : all
.endc

V1 3 0 DC=0
R3 3 5 5k
R1 2 4 500
R2 5 2 20k
VCC 4 0 DC=10
Q3 2 5 0 QN2222

.MODEL QN2222 NPN AF=1.0 BF=150 BR=7.5 CJC=11PF CJE=26PF
+ IKF=.175 IKR=.012 IS=1.9E-14 ISC=1.9E-13 ISE=5E-11
+ KF=3.2E-16 NC=1.2 NE=2.5 RC=.4 TF=.5E-9 TR=30E-9 VAF=100
+ VAR=6.38 XTB=1.5
.END
```

V nadaljevanju bodo na tem zgledu opisane najpomembnejše analize.

2.4.1. .OP analiza delovne točke

Definicija:

Delovna točka vezja je niz vrednosti vseh enosmernih tokov in napetosti v vezju.

Ukaz .OP sproži analizo delovne točke. Simulator izračuna: vozliščne potenciale, tokove skozi napetostne vire, napetosti in tokove na polprevodnih elementih ter vrednosti njihovih lineariziranih modelov. Če je k simulatorju dodan postprocesor, lahko z njim izračunamo tudi ostale veličine (npr. vejne napetosti, razne moči itd.).

Zgled (glej enostopenjski ojačevalnik na strani 39):

S pomočjo simulatorja določi:

1. Enosmerne tokove in napetosti.
2. Strmino transkonduktančne karakteristike tranzistorja.

Računalniški izpisi so zaradi velike obsežnosti skrajšani (označeno z dvopičjem!)

```

***** SMALL SIGNAL BIAS SOLUTION - OP

      Node                Voltage
***
V( 4 )                   1.000000e+001
V( 2 )                   5.026658e+000
V( 5 )                   7.006249e-001
V( 3 )                   0.000000e+000
***
          ****          *****
      Node                Voltage

uizh                    5.026658e+000
uizh                    5.026658e+000
q3#collector            5.022766e+000

          Source          Current
          -----          -
          @q3[icc]        9.730420e-003
          @vcc[i]         -9.94668e-003
          @r2[i]          -2.16302e-004
          @r1[i]          -9.94668e-003
          v1#branch       1.401250e-004
          vcc#branch      -9.94668e-003

```

Interpretacija izpisanih rezultatov:

- Napetost v vozlišču 5 je 0,70062 V
- Kolektorski tok (tranzistor q3) je 9,73 mA.

```
Circuit: ZGLED - RNV ZA VERZIJO ISSPICE 4
----- delovna točka -----
BJT: Bipolar Junction Transistor
device      q3
model       qn2222
off         0
icvbe       0
:
vbe       0.701
vbc       -4.32
icc       0.00973
ib        7.62e-005
:
gm        0.354
go          0.000104
:
cs          0
p           0.049
ft          2.57e+008
temp        27

Resistor: Simple linear resistor
device      r1      r3      r2
model       R       R       R
resistance  500     5e+003  2e+004
temp        27     27     27
:

Vsource: Independent voltage source
device      vcc      vin
dc          10      0
acmag       0       1
:
```

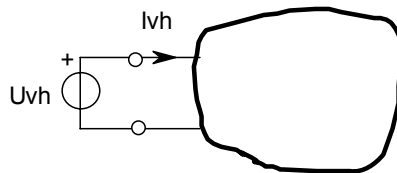
Interpretacija izpisanih rezultatov za tranzistor q3:

- Bazni tok je 76,2 μA .
- Napetost UBE je 0,701 V.
- Kolektorski tok je 9,73 mA.

2.4.2. .DC analiza enosmerne karakteristike

Če v vezju spremenimo vrednost enega ali več virov, se spremenijo tudi napetosti in tokovi. Pri vsaki spremembi preide vezje v novo delovno točko. Odvisnosti med izbranimi tokovi oziroma napetostmi pravimo **enosmerna karakteristika**. Torej, enosmerna karakteristika je niz delovnih točk, ki jo predstavimo v obliki tabele ali pa kot graf. Le redko nam uspe poiskati tudi ustreznemu analitični izraz. Najbolj znane so vhodna, izhodna in prenosna karakteristika.

Enosmerna vhodna karakteristika je relacija med enosmernim vhodnim tokom in vhodno napetostjo. Če spreminjamo vhodno napetost, dobimo prevodnostno obliko $I_{vh} = f(U_{vh})$, če spreminjamo tok, pa uporovno $U_{vh} = f(I_{vh})$. Za dvopole je vhodna karakteristika identična njegovi enosmerni karakteristiki. Podobno velja za izhodno karakteristiko, kjer opazujemo relacijo med izhodnim tokom in napetostjo.



Slika 2.4 Če želimo v laboratoriju izmeriti vhodno karakteristiko, spreminjamo U_{vh} , merimo I_{vh} in nato narišemo $I_{vh} = f(U_{vh})$

Enosmerna prenosna karakteristika opisuje relacijo med izhodnim in vhodnim tokom (ali napetostjo). Obstajajo 4 možne karakteristike:

$$U_{izh} = f(U_{vh})$$

$$U_{izh} = f(I_{vh})$$

$$I_{zh} = f(U_{vh})$$

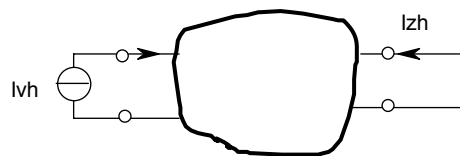
$$I_{zh} = f(I_{vh})$$

napetostna prenosna karakteristika

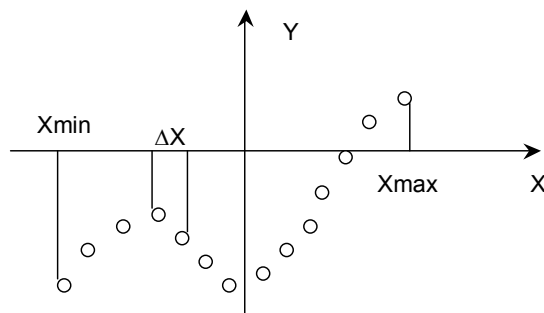
transrezistivna karakteristika

transkonduktančna karakteristika

tokovna prenosna karakteristika



Slika 2.5 Če želimo v laboratoriju izmeriti tokovno prenosno karakteristiko, spreminjamo I_{vh} , merimo I_{zh} in nato narišemo $I_{zh} = f(I_{vh})$

Slika 2.6 Enosmerna karakteristika $y = y(x)$

Sintaksa ukazov, ki so potrebni, da tabeliramo enosmerno karakteristiko (glej sliko 2.6):

.DC <ime vira X> <Xmin> <Xmax> <ΔX>

.PRINT DC < reference na napetosti in tokove>

Če se v karakteristiki pojavlja parameter (npr. $I_c = f(U_{CE}, I_B)$), lahko spreminjanje parametra dosežemo z dodatno zanko, ki ima enako sintakso.

.DC <vir X> <Xmin> <Xmax> <ΔX> <vir X1> <X1min> <X1max> <ΔX1>

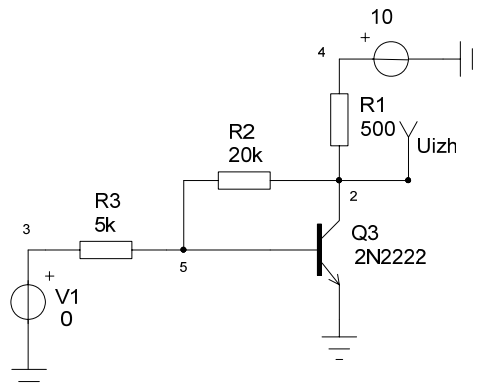
.PRINT DC < reference na napetosti in tokove>

Reference na napetosti in tokove – glej zgled, ki sledi.

Zgled 1:

V(3)	potencial vozlišča številka 3
V(4,5)	napetost med vozliščema 4 in 5
I(VG7)	tok skozi napetostni vir VG7

Zgled 2: (enostopenjski ojačevalnik)



S pomočjo simulatorja določi (glej zgled na strani 39):

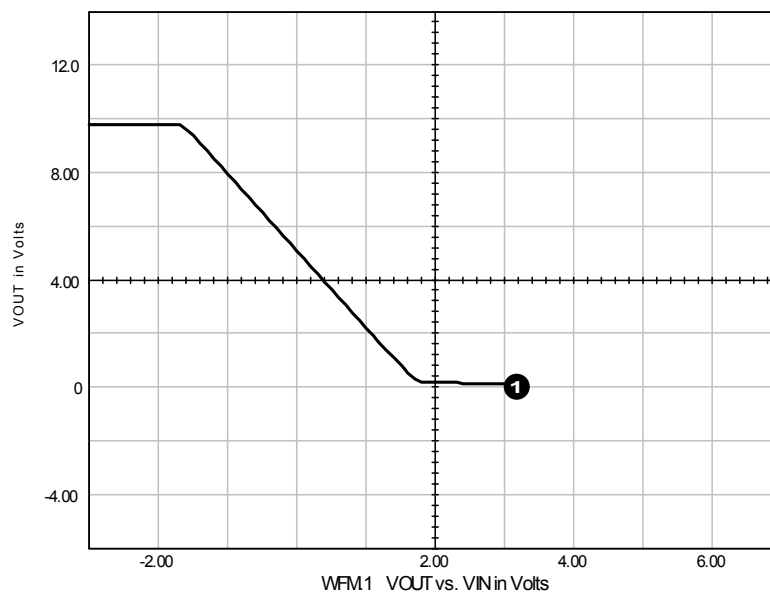
5. Prenosno enosmerno karakteristiko.

Vhod ojačevalnika je vozlišče 3 in izhod vozlišče 2. Ker je na vhod priključen že vir V1, ga bomo v ukazu .DC uporabili.

```
ZGLED - RNV ZA VERZIJO ISSPICE 4
:
.DC V1 -3 3 0.1
.PRINT DC V(2)
:
* OPIS VEZJA
V1 3 0 0V AC 1 PULSE 0 5V 10NS 0 0 200NS
:
```

Rezultati simulacije

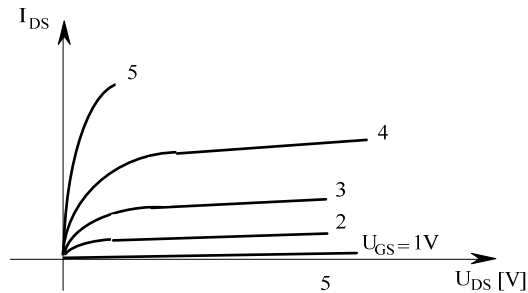
```
ZGLED - RNV ZA VERZIJO ISSPICE 4
*****      DC TRANSFER CURVES      Temperature = 27 Deg C *****
*****
V1          V(2)          INDEX
-3.000000e+000  9.745098e+000      0
-2.900000e+000  9.747059e+000      1
-2.800000e+000  9.749020e+000      2
:
:
3.000000e+000  1.006888e-001      60
```



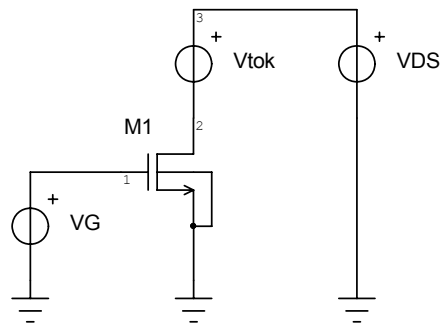
Slika 2.7 Prenosna karakteristika, izrisana s pomočjo grafičnega postprocesorja

Zgled 3: Simulacija izhodne karakteristike MOS

Najprej skiciramo pričakovani rezultat.



Nato pripravimo vezje.

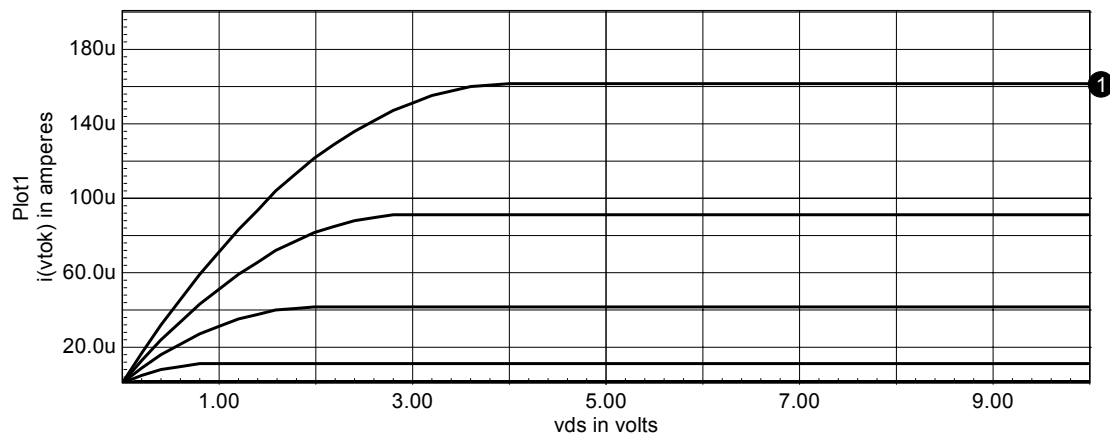


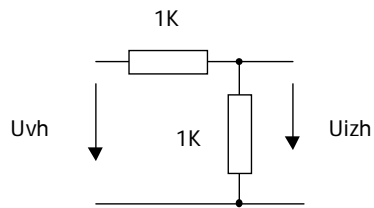
Ker nas zanima tok I_{DS} , vstavimo v ustrezno vejo napetostni vir, z napetostjo nič. Da se bomo spomnili njegove vloge, ga bomo poimenovali Vtok.

```

:
* Simulacija izhodne karakteristike MOS
* tranzistorja, UGS je parameter.
.DC VDS 0 10 0.01 VGS 0 5 1
.PRINT V(1) I(Vtok) V(3)
:

```



Vaja 1

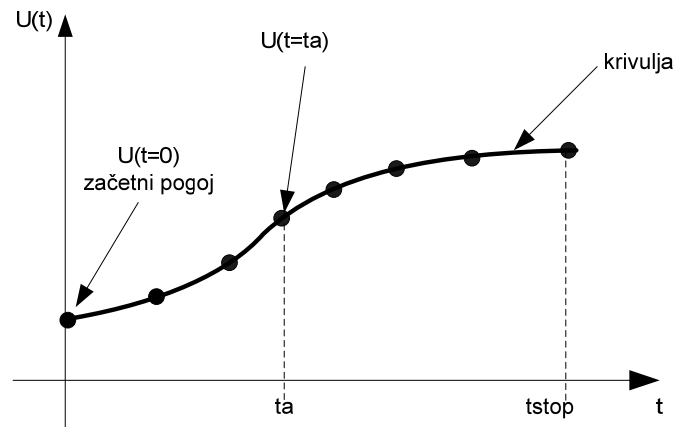
V območju vhodne napetosti -5 do 10 V nas zanimajo:

1. prenosna karakteristika $U_{izh} = f(U_{vh})$,
2. vhodna in
3. izhodna karakteristika.

Tvori ustrezno vhodno datoteko.

2.4.3. .TRAN analiza prehodnega pojava ali tranzientna analiza

S to analizo, ki je ena izmed zelo pomembnih in pogostih, ugotovljamo, kako se spreminjajo posamezne napetosti in tokovi **v odvisnosti od časa**. Če "peš" analiziramo taka vezja, najprej zapišemo diferencialne enačbe in jih nato rešimo. Postopek je zahteven in izvedljiv samo za zelo preprosta vezja. V večini primerov moramo za tovrstno analizo uporabiti simulator.



Slika 2.8 Rezultat tranzientne analize so tokovi in napetosti v odvisnosti od časa. Simulator izračuna vrednost samo v določenih časih, nato pa z interpolacijo nariše krivuljo, ki gre skozi vse točke.

Tranzientna analiza ima smisel, če vezje vsebuje vsaj en reaktiven²⁷ element. Če je vezje rezistivno, so njegovi odzivi neskončno hitri. Razen redkih izjem²⁸ je v vezju, ki ga analiziramo s tranzientno analizo, najmanj en časovno varianten element (npr. generator pravokotnega signala).

Osnovna sintaksa:

.TRAN < $t_{\text{printstep}}$ > < t_{stop} >

.PRINT TRAN < reference na napetosti in tokove >

$t_{\text{printstep}}$ časovni korak izpisovanja in t_{stop} je čas trajanja simulacije.

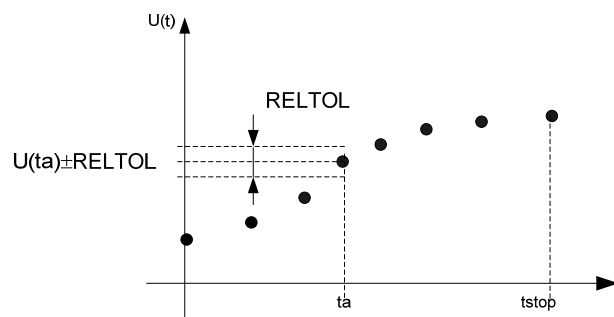
²⁷ Kondenzator ali tuljava.

²⁸ Ta izjema je oscilator.

Neenakomerna razporeditev časovnega koraka bi bila pri izpisu za uporabnika moteča. S parametrom $t_{printstep}$ povemo računalniku, v kakšnem časovnem koraku naj izpiše vrednosti. Z ukazom PRINT se s pomočjo interpolacije izračunajo vmesne vrednosti (slika 2.9 b).

Pri oscilatorjih nastopi začetni prehodni pojav, v katerem se ta razniha. Če nas ta interval ne zanima, potem lahko izpis začnemo šele po določenem času, ki ga nastavimo s parametrom $t_{printstart}$.

Parameter RELTOL definira relativno natančnost izračunov posamezne točke. Privzeta vrednost je 0,001, kar pomeni, da bodo tokovi in napetosti izračunani na 0,1 % natančno. Večanje natančnosti upočasni simulacijo in hkrati tudi zveča število točk.



Slika 2.11 Natančnost izračuna regulira parameter RELTOL. Privzeta vrednost je 0,1 %.

Zgled 2:

```
.tran 100m 1m 50m 1m
.print tran V(4)
```

Simulacija bo trajala 100 ms. Podatki za napetost v vozlišču 4 bodo izpisani po koraku 1ms. Izpis se bo začel po 50 ms. Simulator bo lahko daljšal interni korak integracije največ do 1 ms. Napetost V(4) bo izračunana z natančnostjo 0,1 %.

Za pravilno izvedeno simulacijo moramo poznati tudi začetne pogoje ob času $t = 0$. Začetno stanje je definirano z napetostmi na kondenzatorjih oziroma tokovi skozi tuljave. Obstaja dve možnosti: vezje se nahaja oziroma se ne nahaja v ravnotežnem stanju. Le če se vezje nahaja v ravnotežnem stanju, so začetni pogoji na reaktančnih elementih napetosti in tokovi delovne točke.

Zgled 3:**1. vezje se nahaja v ravnotežnem stanju**

Vezje se nahaja v ravnotežnem stanju oziroma je že tako dolgo vklopljeno, da so izzveneli vsi prehodni pojavi. Ob času $t = 0$ se pojavi stopnica. Izpiši potek napetosti na vozlišču 1 od 0 do 100 μs po koraku 1 μs .

```
:
CB 1 0 10pF
:
.tran 1u 100u
.print tran v(1)
```

2. vezje se ne nahaja v ravnotežnem stanju

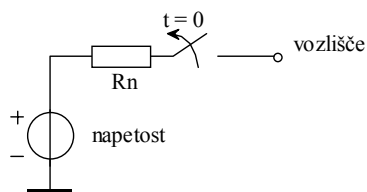
Če poznamo začetne vrednosti na reaktančnih elementih, uporabimo parameter **UIC**.

```
:
CB 1 0 10pF IC=6V
:
.tran 1u 100u UIC
.print tran v(1)
```

b. Če pa poznamo začetne napetosti v določenih vozliščih – uporabimo ukaz:

.IC <vozlišče 1>=<napetost 1> <vozlišče 2>=<napetost 2> ...

Simulator na vsako tako vozlišče priključi napetostni vir z majhno notranjo upornostjo, izračuna delovno točko. Tako dobljena delovna točka predstavlja začetne vrednosti. Ob času $t = 0$ simulator odklopi prej dodane vire in izvaja algoritem za simulacijo prehodnega pojava.

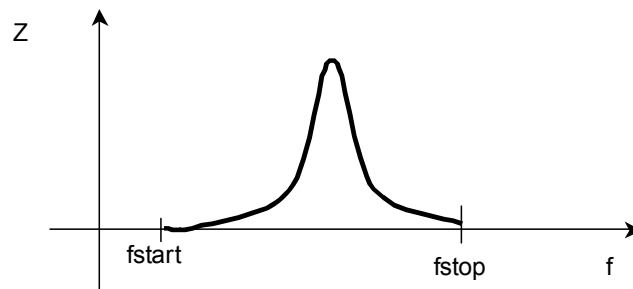


```
.ic v(1)=6
.tran 1u 100u
.print tran v(1)
```

2.4.4. .AC frekvenčna analiza

Frekvenčna karakteristika linearnega vezja prikazuje odvisnost neke lastnosti **od frekvence**, npr. $Z(\omega)$ ali $U_{izh} = U_{izh}(\omega)$. Namesto krožne frekvence ω uporablja SPICE navadno frekvenco. Pri meritvi potrebujemo poseben vir (napetostni ali tokovni), ki mu lahko spreminjamo frekvenco. Z elektronskim voltmetrom ali ampermetrom nato merimo izbrano veličino v določenih točkah. Tudi če je vezje nelinearno, lahko z zadostno majhno amplitudo izmeničnega signala vezje obravnavamo kot linearno.

Ker se ta meritev v praksi zelo pogosto pojavlja, je njena simulacija vgrajena v večino simulatorjev. Pri simulatorju SPICE uporabimo ukaz `.AC`. Ker simulator predpostavlja, da bo izmenični signal dovolj majhen, najprej izračuna delovno točko in vezje v tej točki **linearizira**. Tako dobljeni linearni model vezja nato uporablja pri izračunu kompleksnih tokov in napetosti. Za vsako frekvenco izračuna vrednosti tokov in napetosti. V vezju moramo imeti najmanj en generator s spremenljivo frekvenco. Vsakemu generatorju, ki mu želimo spreminjati frekvenco, dodamo oznako `AC` in velikost amplitude izmeničnega signala. Velikost amplitude izmeničnega vira pri simulaciji ni pomembna, saj simulator uporablja lineariziran model vezja. Če si izberemo za amplitudo 1V oziroma 1A, lahko zelo enostavno določimo razna ojačenja in impedanco oziroma admitanco.



Slika 2.12 Impedanca paralelnega nihajnega kroga

Sintaksa:

```
<generator 1> <vozlišče> <vozlišče> AC <amplituda> [<fazni kot>]
<generator 2> <vozlišče> <vozlišče> AC <amplituda> [<fazni kot>]
:
.AC <vrsta intervala> <število točk v intervalu> <fstart> <fstop>

.PRINT AC <sklicevanje na določeno napetost ali tok>
```

Delitev frekvenc (skala) med `fstart` in `fstop` lahko izbiramo. Na razpolago imamo tvorjenje naslednjih intervalov:

```
LIN   linearna delitev
OCT   delitev na oktave
DEC   logaritemska delitev na dekade
```

Tabela 2.2 Naraščanje frekvence pri različnih načinih delitve za območje 1kHz do 5MHz in 10 točkami na interval

DEC	OCT	LIN	← način delitve
1kHz	1kHz	1kHz	
1.25e+003,	1.07e+003		
1.58e+003,	1.14e+003		
1.99e+003,	1.23e+003		
2.51e+003,	1.31e+003		
3.16e+003,	1.41e+003		
3.98e+003,	1.51e+003		
5.01e+003,	1.62e+003		
6.30e+003,	1.74e+003		
7.94e+003,	1.86e+003		
	2kHz		
	2.14e+003		
	:		
	4kHz		
10kHz	:		
1.25e+004	:		
:	:		
100kHz	:		
:	:	5.56E+05	
:	:	1.11E+06	
:	:	1.66E+06	
1MHz	:	2.22E+06	
:	:	2.77E+06	
:	:	3.33E+06	
:	:	3.88E+06	
:	:	4.44E+06	
5.01MHz	5.01MHz	5.01MHz	
38	124	10	←skupno število točk

Najpogosteje se uporablja delitev na dekade. Sklicevanje na določeno napetost je podobno kot pri DC analizi. Z dodatno črko pri oznaki za napetost (V) oziroma tok (I) dobimo naslednje možne podatke:

V	amplituda
VM	amplituda
VDB	amplituda v dB
VP	kot
VR	realni del
VI	imaginarni del

Če črko V zamenjamo z I, se sklicujemo na tok. Npr. IP(VG) je fazni kot toka, ki teče skozi napetostni vir VG.

Pomni:

- V vezju imamo lahko več generatorjev s spremenljivo frekvenco.
- Število vseh točk je odvisno tudi od načina delitve (glej tabelo 2.2).
- Amplituda izmeničnih generatorjev je lahko le pri simulaciji poljubna velika, saj simulator vezje najprej linearizira. Pri meritvi realnega vezja to ne velja, razen če je linearno.
- Na kot vpliva tudi, ali je vezje krmiljeno s tokovnim ali napetostnim virom ter generatorjeva polariteta.

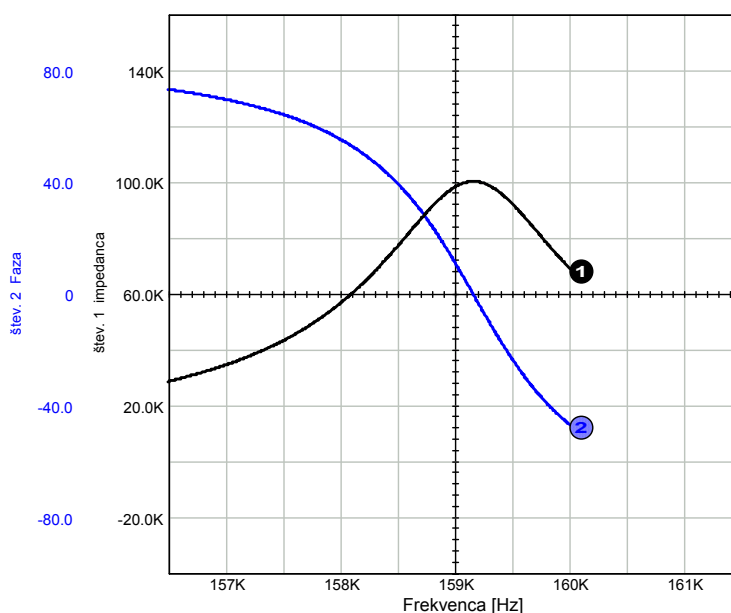
Zgled 1:

Zanima nas impedanca paralelnega nihajnega kroga.

```
Zgled Paralelni nihajni krog - resonančna frek. je 1,592MHz
* Nihajni krog vzbuja s tokovnim generatorjem
L1 1 3 1M
RL 3 0 10
C1 1 0 1N
IG 0 1 AC 1
.AC LIN 100 155K 160K
.PRINT AC V(1) VP(1)
.END
```

Ker je resonančna frekvenca približno 1590 kHz, bomo analizirali vezje v območju od 155 KHz do 160 kHz. Izbrali smo linearno merilo, kar pomeni, da bo v tem intervalu 100 enakomerno naraščajočih točk. Če vzbuja vezje s tokovnim generatorjem³⁰ z amplitudo 1 A, je napetost na njegovih sponkah identična impedanci:

$$\hat{U} = \hat{I} \cdot \hat{Z}$$



³⁰ Pazi, kako je obrnjen tokovni generator.

Zgled 2: (enostopenjski ojačevalnik)

S pomočjo simulatorja določi:

6. Zgornjo frekvenčno mejo.

```
ZGLED - RNV ZA VERZIJO ISSPICE 4
:
:
.AC DEC 20 100 100MEG
.PRINT AC VDB(2) VP(2)
:
* OPIS VEZJA
V1 3 0 0V AC 1 PULSE 0 5V 10NS 0 0 200NS
:
```

Rezultati simulacije

```
*** Wed Oct 14 16:51:49 1998 ***** IsSpice4 ver. 4f3.6*** 8/4/95

ZGLED - RNV ZA VERZIJO ISSPICE 4

***** AC ANALYSIS Temperature = 27 Deg C *****

***** SMALL SIGNAL BIAS SOLUTION - AC ANALYSIS *****

Node Voltage

***
V( 2 ) 5.026658e+000
V( 5 ) 7.006249e-001
V( 3 ) 0.000000e+000
V( 4 ) 1.000000e+001
q1#collector 5.022766e+000
vcc#branch -9.94668e-003
vin#branch 1.401250e-004

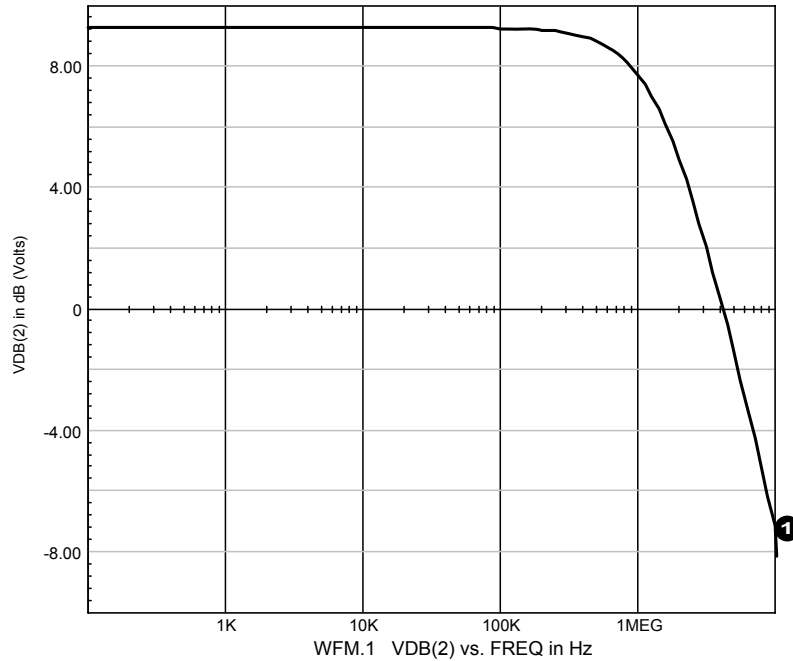
***
*****

FREQUENCY IMAGINARY V(2) VP(2) INDEX

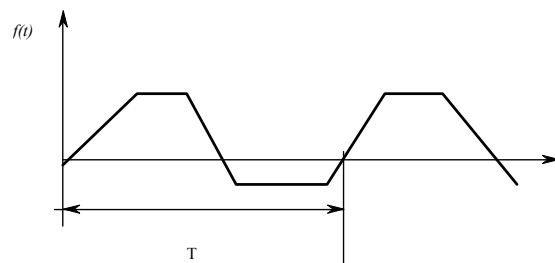
1.000000e+003, 0.000000e+000 2.895805e+000 1.799623e+002 0
1.122018e+003, 0.000000e+000 2.895805e+000 1.799577e+002 1
1.258925e+003, 0.000000e+000 2.895805e+000 1.799526e+002 2
1.412538e+003, 0.000000e+000 2.895804e+000 1.799468e+002 3
1.584893e+003, 0.000000e+000 2.895804e+000 1.799403e+002 4
1.778279e+003, 0.000000e+000 2.895804e+000 1.799330e+002 5
1.995262e+003, 0.000000e+000 2.895803e+000 1.799248e+002 6

8.912509e+008, 0.000000e+000 1.665167e-003 1.431668e+001 119
1.000000e+009, 0.000000e+000 1.339585e-003 1.168133e+001 120

*** Wed Oct 14 16:51:49 1998 ***** IsSpice4 ver. 4f3.6* 8/4/95 *
```

2.4.5. .FOUR Fourierjeva analiza



Slika 2.13 Zgled za periodični signal

Vsako periodično funkcijo lahko izrazimo kot vsoto preprostih funkcij:

$$f(t) = f_1(t) + f_2(t) + \dots \quad (2.1)$$

V elektrotehniki so te funkcije v večini primerov kosinusi:

$$f(t) = A_0 + A_1 \cos(\omega_1 t + \varphi_1) + A_2 \cos(2\omega_1 t + \varphi_2) + A_3 \cos(3\omega_1 t + \varphi_3) + \dots \quad (2.2)$$

A_0 enosmerna komponenta ali povprečna vrednost

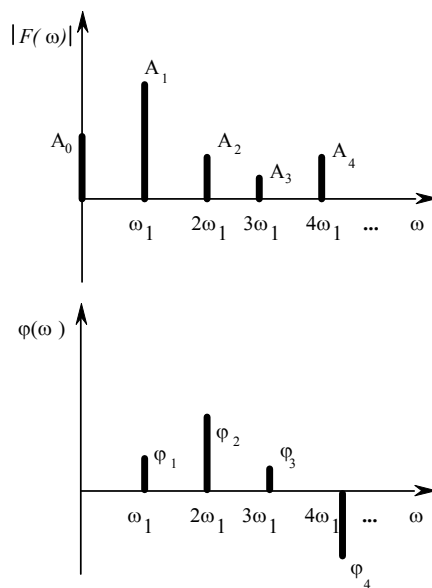
ω_1 osnovna harmonska frekvenca (komponenta) z amplitudo A_1 in kotom φ_1

$$\omega_1 = 2\pi f_1 = \frac{2\pi}{T} \quad \text{ali} \quad f_1 = \frac{1}{T} \quad (2.3)$$

Višje harmonske komponente:

- $2\omega_1$ druga harmonska frekvenca (komponenta) z amplitudo A_2 in kotom φ_2
 $3\omega_1$ tretja harmonska frekvenca (komponenta) z amplitudo A_3 in kotom φ_3
 :

Postopek izračuna se imenuje *Fourierjeva transformacija*. Grafični prikaz harmonskih komponent imenujemo *spekter* (slika 2.14).



Slika 2.14 Zgled nekega spektra

Pri izračunu numerične Fourierjeve transformacije simulator uporabi rezultate, ki jih je dobil pri tranzientni analizi.

```
.TRAN <t_step> <t_stop> [t_start [t_max]]
```

```
.FOUR <osnovna frekv.> <referenca na nap. ali tok> <ref. na  
nap. ali tok>...
```

Pazi! Pri izračunu transformacije simulator uporablja podatke iz intervala $(t_{\text{stop}} - T, t_{\text{stop}})$.

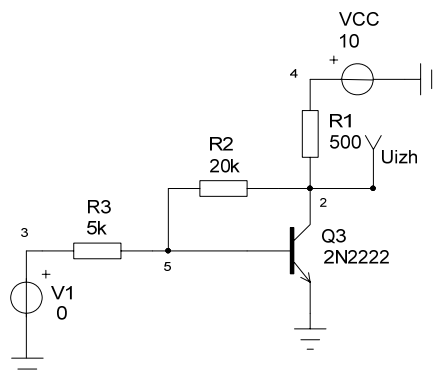
Problemi:

- Nepravilna določitev osnovne frekvence lahko povzroči velike napake pri izračunu.
- V večini primerov nas zanima popačenje signala, katerega amplituda se več ne spreminja. V takih primerih je treba preveriti, ali je prehodni pojav izzvenel.

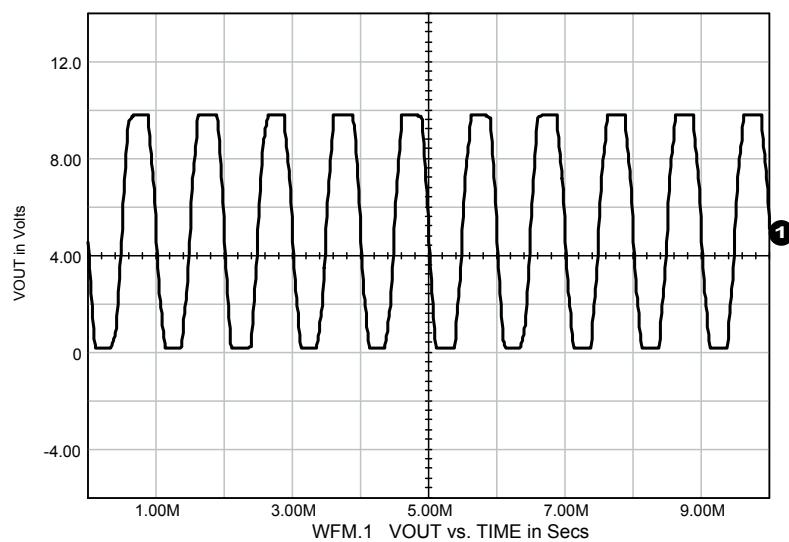
Zgled (glej enostopenjski ojačevalnik na strani 39)

S pomočjo simulatorja določi:

10. Popačenje ojačevalnika pri vhodni sinusni amplitudi 3 V.



```
ZGLED - RNV ZA VERZIJO ISSPICE 4
:
.PRINT TRAN V(2) V(3)
.TRAN 0.01MS 10MS
.FOUR 1K V(2)
* OPIS VEZJA
:
V1 3 0 0V SIN 0 3 1KHZ
:
.END
```



Slika 2.15 Zanima nas popačenje izhodnega signala

```

Fourier analysis for v(2):
No. Harmonics: 10, THD: 15.4088 %, Gridsize:200, Interpolation Degree: 1

```

Harmonic	Frequency	Magnitude	Phase	Norm. Mag	Norm. Phase
0	0	4.97155	0	0	0
1	1000	5.63409	179.847	1	0
2	2000	0.0615327	85.8485	0.0109215	-93.998
3	3000	0.860067	-178.8	0.152654	-358.65
4	4000	0.0164965	70.2784	0.00292798	-109.57
5	5000	0.0255587	-105.34	0.00453643	-285.18
6	6000	0.00308444	-18.831	0.00054746	-198.68
7	7000	0.0124478	80.7982	0.00220938	-99.049
8	8000	0.00462278	-89.488	0.0008205	-269.33
9	9000	0.0952222	156.143	0.0169011	-23.704

Slika 2.16 Rezultati harmonske analize. THD (Total Harmonic Distortion) je merilo za stopnjo popačenosti sinusnega signala.

Natančnost Fourierjeve analize

Natančnost je odvisna od števila točk, numeričnih algoritmov in od kakovosti modela.

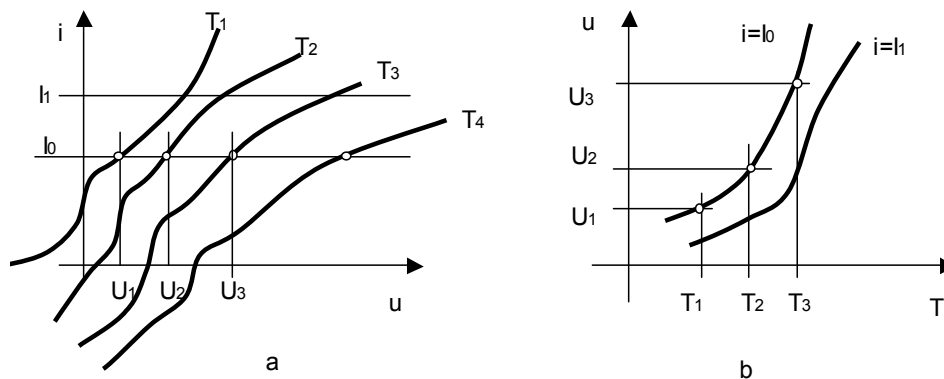
Pri prejšnjem zgledu smo v Fourierjevo analizo vključili tudi vhodni sinusni signal. Njegova popačenost bi morala biti nič. Zaradi numeričnih pogreškov znaša pri starejši verziji Pspice celo 5,5 %.

Tabela 2.3 Popačenje idealnega vhodnega signala, ki so ga izračunale različne verzije simulatorja. Pravilni rezultat je THD = 0.

Verzija simulatorja	THD za vhodni signal	THD za izhodni signal
IS SPICE 1.41	2,459883 %	12,897433 %
PSpice 3.01	5,585568	26,28660
SPICE3F4	2,60272	16.8914
IsSpice4 ver. 4f3.6	0.0086053	15.4088
IsSpice4 ver. 8.8S	0.00860531	15.4088

2.4.6. .TEMP temperaturna analiza

Natančna temperaturna analiza je zelo zahtevna, saj na karakteristiko elementa ne vpliva samo temperatura celotne okolice ampak tudi disipacija, segrevanje sosednjih elementov in način hlajenja. Tipični simulator SPICE je sposoben izvajati le preprosto temperaturno analizo, kjer se predpostavlja, da imajo vsi elementi enako in konstantno temperaturo³¹.



Slika 2.17 Zgled temperaturne odvisnosti nekega nelinearnega upora

Temperaturno odvisnost neke lastnosti lahko prikazujemo na dva načina:

- temperatura je parameter ali
- temperatura je na abscisi.

V večini primerov je temperaturna odvisnost lastnosti približno linearna monotona funkcija. V teh primerih simuliramo samo pri treh temperaturah: sobni, minimalni in maksimalni. Temperaturo nastavljamo z ukazom:

```
.OPTIONS TEMP <temperatura>
```

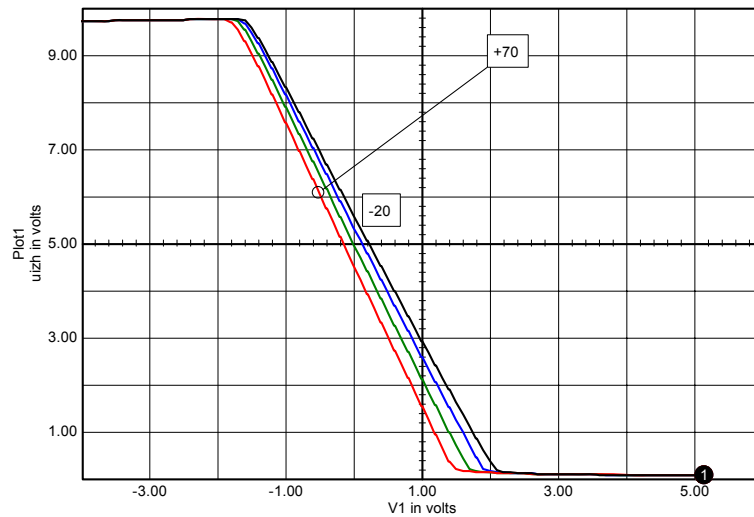
Za več temperatur ponovimo analizo in podatke v grafičnem postprocesorju združimo.

³¹ Pri nekaterih simulatorjih (npr. Intusof SPICE) lahko temperaturo med simulacijo spreminjamo.

Zgled:

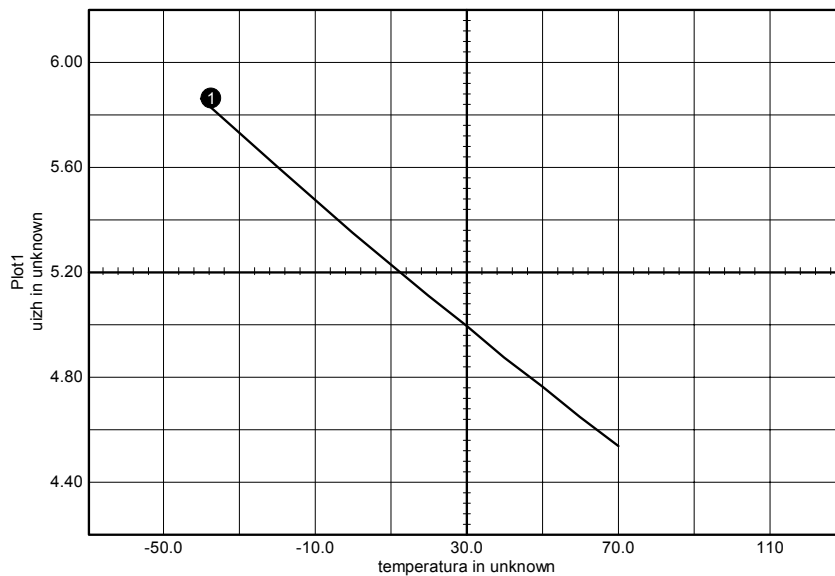
Določi prenosno karakteristiko in U_{izh} v odvisnosti od temperature za območje -40 do 70 °C

1 uizh_-20 2 uizh-0 3 uizh_+30 4 uizh_70



Slika 2.18 Prenosna karakteristika v odvisnosti od temperature

1 uizh



Slika 2.19 U_{izh} v odvisnosti od temp. (-40 do $+70$ °C)

2.4.7. .TF enosmerna analiza majhnih sprememb delovne točke

Pogosto nas zanima, kakšne so lastnosti vezja, če se neka enosmerna napetost (ali tok) na generatorju **počasi** spreminja okrog določene vrednosti. Če so spremembe zelo počasne, nimajo reaktančni elementi nobenega vpliva. Največkrat nas zanimajo predvsem ojačenje vhodnega signala, vhodna ter izhodna upornost takega vezja. (Če želimo proučevati vpliv reaktivnih elementov, potem uporabimo .AC analizo ali pa tranzientno analizo!) Analiza majhnih počasnih sprememb okrog delovne točke je še posebej zanimiva za integrirana vezja, saj so večinoma brez kondenzatorjev oziroma tuljav. Vezjem brez reaktivnih elementov pravimo rezistivna vezja. Če so izpolnjeni prej opisani pogoji, lahko uporabimo posebno analizo, ki je označena s .tf (transfer function)

.TF <izhodna napetost ali tok> <vhodni vir>

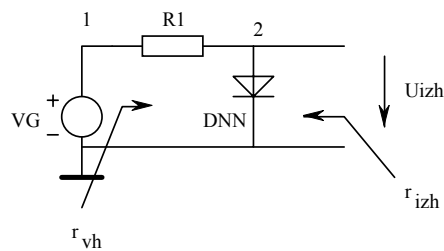
Vhodni vir je lahko napetostni ali tokovni, izhodna spremenljivka je lahko tok ali napetost. Ta analiza najprej izračuna delovno točko in nato:

a. vrednost prenosne funkcije v delovni točki: $A = \frac{\partial \text{izhodna_spremenljivka}}{\partial \text{vhodna_spremenljivka}} \Big|_Q$

b. vhodno upornost $r_{vh} = \frac{\partial u_{vh}}{\partial i_{vh}} \Big|_Q$

c. izhodno upornost $r_{izh} = \frac{\partial u_{izh}}{\partial i_{izh}} \Big|_Q$

Zgled 1:



```
R1 1 2 100
D1 2 0 DNN
VG 1 0 2V
.MODEL DNN D
.TF V(2) VG
.END
```

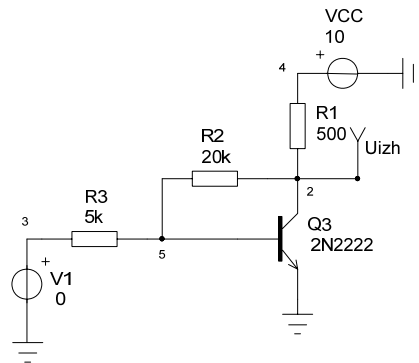
***** SMALL SIGNAL DC TRANSFER FUNCTION

```
output_impedance_at_V(2)          1.982162e+000
vg#Input_impedance                1.020222e+002
Transfer_function1.982162e-002
```

Zgled 2: (glej enostopenjski ojačevalnik na strani 39)

S pomočjo simulatorja določi:

9. Vhodno dinamično upornost.



Vhodna datoteka

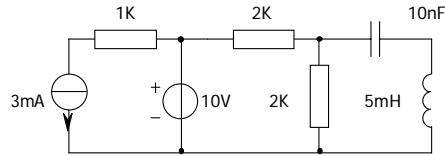
```
:
.TF V(2) V1
:
```

Izhodna datoteka

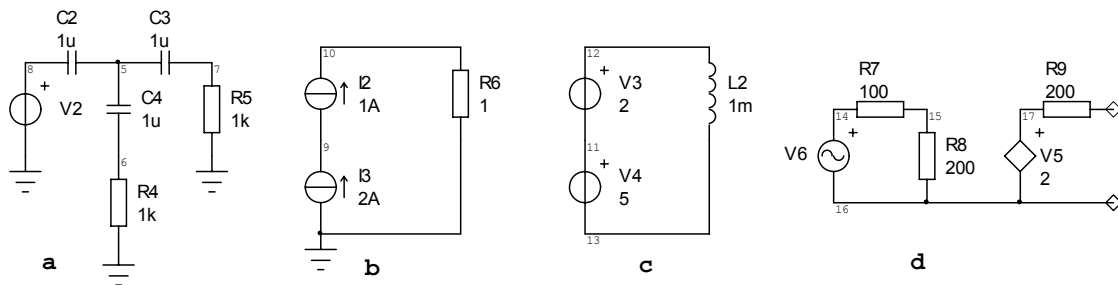
```
:
***** SMALL SIGNAL DC TRANSFER FUNCTION
          output_impedance_at_V(2)          1.281419e+002
          v1#Input_impedance               5.089593e+003
          Transfer_function                 -2.89581e+000
:
:
```


2.5. Vaje

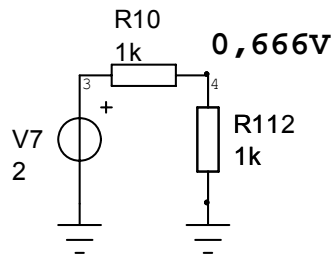
Vaja 1 Opiši vezje v skladu s pravili simulatorja SPICE.



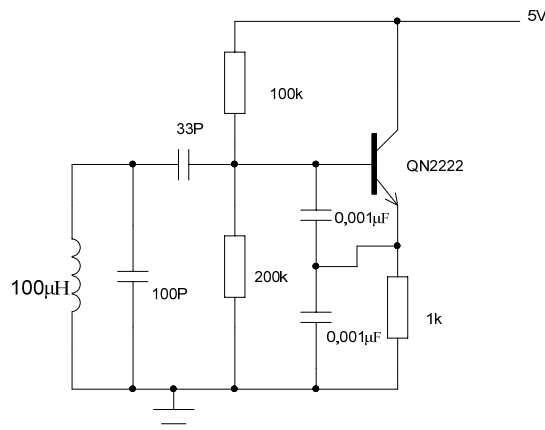
Vaja 2 Ugotovi, ali so vezja narisana v skladu s topološkimi pravili, ki jih zahteva simulator SPICE. Če je potrebno, vezja ustrezno spremeni.



Vaja 3 Kako si razlagaš rezultat te simulacije?



Vaja 4 Opiši vezje v skladu s pravili simulatorja SPICE.



Vaja 5 Nariši vezje, ki ustreza temu opisu:

```
E:\Simulacije\CMOS_invertor_1\tok_zrcalo.cir
```

```
M1 2 2 1 1 MN3 L=2u W=10u
M2 3 2 1 1 MN3 L=2u W=50u
I1 3 2 DC=10u
V1 3 0 DC=3
V2 1 0 DC=-3
.MODEL MN3 NMOS VT=1
.END
```

Vaja 6 Nariši vezje, ki ustreza temu opisu:

```
* bipolarno vezje: datoteka amp.cir (imenik DEMO)
R1 1 2 1K
R2 5 4 1K
Q1 6 2 5 QN2222
R3 5 0 10
V0 3 0 DC=10V
V1 1 0 DC=.58V
Q2 4 6 3 QN2907
.MODEL QN2222 NPN BF=105 BR=4 CJC=12.2P CJE=35.5P IKF=.5
.MODEL QN2907 PNP BF=200 BR=6 CJC=19PF CJE=23PF IKF=100E-3
.end
```

Vaja 7 Opiši analize, ki ustrezajo posameznim stavkom. Če je mogoče, skiciraj rezultat z grafom. Potek grafa si izmisli.

- a.

```
.DC Vgen -2 10 100m
.print DC V(2), V(4,5)
```
- b.

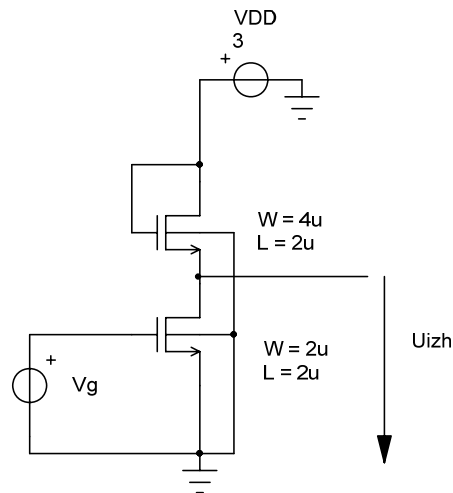
```
.tran 10u 100m
.print tran V(8), I(vcc)
```
- b.

```
.AC DEC 10 1k 100k
.print ac Vm(8), Vdb(7)
```

Vaja 8 Glede na podatke, ki nas zanimajo, izberi ustrezno analizo.

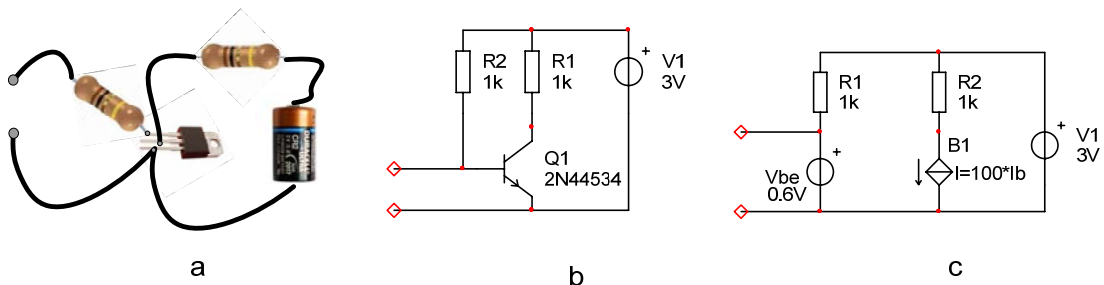
- a. Zanima nas enosmerna napetost UCE in tok skozi upor.
- b. Zanima nas frekvenčna odvisnost ojačenja.
- c. Zanima nas spreminjanje kolektorskega toka v odvisnosti od časa.
- d. Zanima nas enosmerna karakteristika diode $I_d = f(U_d)$.
- e. Zanima nas, kolikšno je popačenje sinusnega signala.

Vaja 9 Na sliki je prikazan preprost NMOS ojačevalnik. Zanimata nas prenosna karakteristika $U_{izh} = f(V_g)$ in karakteristika $I_{DD} = f(V_g)$ v območju napajalne napetosti. Opiši vezje v skladu s pravili simulatorja SPICE. Za model MOS tranzistorjev uporabi naslednji stavek: `.MODEL NMOS3 NMOS VT = 0,8`.



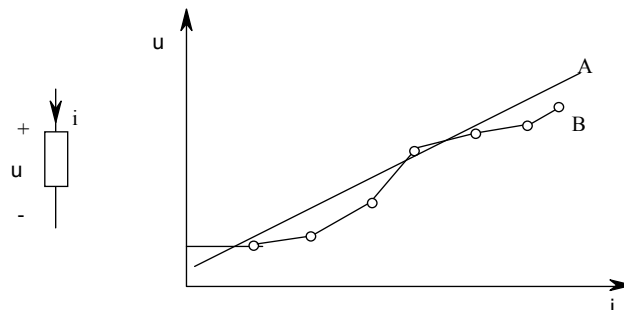
3. MODELIRANJE

Konkretno³² električno vezje je sestavljeno iz množice med seboj povezanih komponent (slika 3.1a). Če želimo vezje analizirati, moramo vsako konkretno komponento zamenjati z abstraktnim električnim elementom, ki ima natančno opredeljen odnos med tokom, napetostjo, elektrino in fluksom (slika 3.1b). Ta odnos lahko opišemo z enačbami ali pa z ekvivalentnim vezjem, ki je sestavljeno samo iz elementov, ki imajo natančno definirane lastnosti (slika 3.1c). Ta proces imenujemo modeliranje, ekvivalentno vezje pa model. **Model je enačba ali vezje, ki se v predvidenem področju in v območju dovoljenih odstopanj obnaša podobno kot določen element oziroma resnična komponenta** (glej sliko 3.2). Ker bomo obravnavali samo električne modele, bomo atribut "električni" v nadaljevanju izpustili.



Slika 3.1 Konkretno električno vezje je sestavljeno iz komponent (a), isto vezje, sestavljeno iz električnih elementov (b) in model celotnega vezja, sestavljenega iz primitivnih idealnih elementov (c).

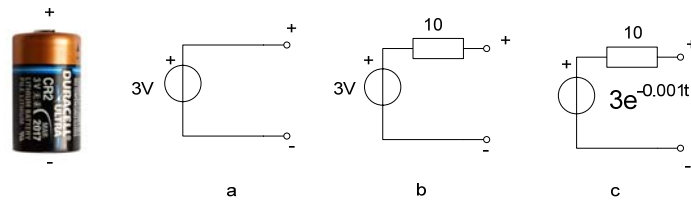
Zelo pogosto so modeli preprostih komponent sestavljeni iz primitivnih električnih elementov (slika 3.1 c). Primitivni ali elementarni električni elementi so tisti elementi, ki jih ne moremo sestaviti iz še bolj preprostih. Njihove lastnosti so idealizirane in zelo natančno definirane. Tipični primitivni elementi so: idealen upor, idealen kondenzator, idealna tuljava, idealen napetostni in tokovni vir (navaden ali krmiljen). Najpreprostejši model je sestavljen samo iz enega električnega elementa (npr. idealnega napetostnega vira, glej sliko 3.3 a).



Slika 3.2 Model ima vedno samo podobno karakteristiko kot komponenta. Izmerjena karakteristika komponente (B) in karakteristika modela (A).

³² Mišljeno je vezje na fizičnem nivoju.

Ker je natančnost simuliranja odvisna predvsem od natančnosti modelov, je izbor modela izredno pomemben. Namesto atributa natančnost se pogosto uporablja termin **red modela**. Pravil, ki bi enoumno določala, kako določimo red, ne obstajajo. Velja le to, da je model prvega reda najenostavnejši in opisuje samo najvažnejše lastnosti elementov. Takšen model ima tudi relativno majhno področje, znotraj katerega je uporaben. Mnogo vplivov (npr. temperatura, parazitne kapacitivnosti) je zanemarjenih (glej sliko 3.3). Z višanjem reda se vključujejo drugi vplivi in fenomeni in zaradi tega se manjšajo razlike med rezultati simulacije in meritev. **Večjo natančnost modela je mogoče doseči le s povečanjem njegove kompleksnosti.**



Slika 3.3 Trije modeli 3 V baterijskega vložka: model prvega reda (a), model drugega (b) in tretjega reda (c).

Zgled:

Najbolj preprost model linearnega upora je kar idealen upor, ki je opisan z izrazom $i = u/R$. Imenovali bi ga lahko tudi model prvega reda. Ta model ne upošteva vpliva temperature, parazitnih kapacitivnosti in induktivnosti. Če je treba upoštevati tudi temperaturno odvisnost, lahko model izboljšamo in dobimo model drugega reda:

$$i = \frac{u}{R(1 + T_{C1}(T - T_0) + T_{C2}(T - T_0)^2)}$$

Čeprav neko vezje vsebuje enake komponente, njihove karakteristike niso nikoli popolnoma enake. Če bi npr. izmerili karakteristiki dveh diod BAY80, bi opazili majhne razlike. Te razlike proizvajalci opišejo z ustrežno vrednostjo toleranc. Vključevanje toleranc v model bi modeliranje zelo zakompliciralo, saj bi takšen model poleg električnih karakteristik moral upoštevati še verjetnostne parametre. Če je v model vključen tudi verjetnostni vidik, ga imenujemo **stohastični model**. Model, ki ne vsebuje verjetnostnih elementov, spada v skupino **determinističnih modelov**. Modeliranje električnih elementov večinoma temelji na determinističnih modelih, vpliv toleranc pa se obravnava posebej (glej poglavje Analiza toleranc).

Najpogostejši modeli elementov (npr. dioda, bipolarni tranzistor itd.) so vgrajeni kot funkcije v programski kodi simulatorja. Zato jim pravimo *notranji* ali *parametrični modeli*³³. Če za določen element (npr. tiristor) notranji model ne obstaja, ga je treba ustvariti. Tako ustvarjene modele imenujemo *zunanji modeli*.

³³ Nekateri jih imenujejo tudi kodni modeli.

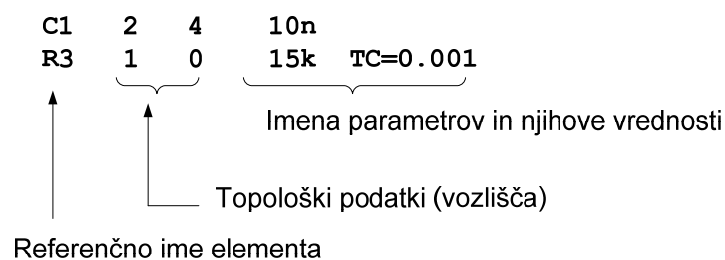
Modele, pri katerih se predpostavlja, da imajo vsi elementi enako notranjo temperaturo, ki je neodvisna od zunanje in od disipacije, imenujemo *izotermični modeli*. Večina simulatorjev uporablja le izotermične modele. Kadar gre za temperaturno občutljive elemente oziroma za večje disipacije, je treba izotermične modele z ustreznim dodatnim termalnim modelom spremeniti v *elektrotermične*.

Pomni!

- Natančnost simulacije je odvisna od natančnosti modelov.
- Večja natančnost modela pomeni večjo kompleksnost njegove strukture.
- Vsak model ima omejeno območje veljavnosti. Zunaj tega območja je neuporaben.

3.1. Notranji oziroma parametrični modeli

V simulator so običajno vgrajeni modeli tistih elementov, ki se najpogosteje uporabljajo (npr. upor, dioda, bipolarni tranzistor itd.). Glede na način opisovanja obnašanja modela ločimo dve vrsti notranjih modelov: preprosti in zahtevnejši model. Najosnovnejši notranji modeli so **modeli primitivnih elementov**³⁴.

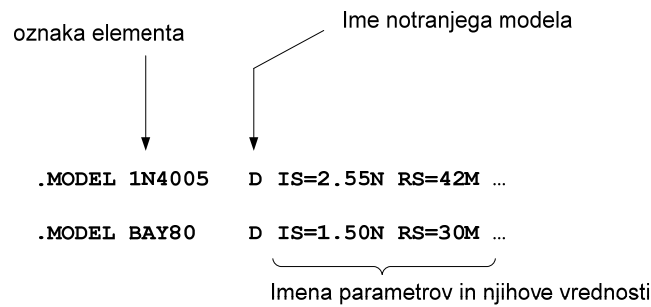


Slika 3.4 Najbolj preprosti notranji modeli so modeli primitivnih elementov

Obnašanje notranjega modela je v simulatorju definirano (programirano) kot funkcija, ki ima več parametrov³⁵. Modeliranje osnovnih polprevodnih elementov zahteva zahtevnejše modele. Npr. model bipolarnega tranzistorja, ki temelji na fizikalnem dogajanju v polprevodniku, zahteva nastavitve približno 40 parametrov. Za MOS tranzistor je vgrajenih celo več variant modelov. Parametre najpogosteje objavi kar tovarna, ki izdeluje konkreten element. Za zahtevnejše notranje modele uporabljamo identifikator `.MODEL`. Vrednosti parametrov so lahko vezane na podatke iz meritev karakteristik (npr. upor) ali pa na tehnološke podatke (npr. debelino oksida, koncentracijo primesi itd.).

³⁴ Angl. primitive device.

³⁵ Model tranzistorja, ki se uporablja za dimenzije, manjše od 1 mikrona BSIM3 (*Berkeley Short-channel IGFET Model*), je opisan s približno 20KLOC (21 datotek) v programskem jeziku C. (Vir: Lemaitre, C. McAndrew: "An open source software tool for compact modeling applications", IEEE Circuits and Devices Magazine, marec/april 2004, str.6–9).



Slika 3.5 Modela diod 1N4005 in BAY80 (prikazanih je samo nekaj prvih parametrov)

Na tržišču obstaja kar nekaj programov, ki pomagajo pri določevanju parametrov za notranje in nekatere zunanje modele. Tak program, ki ga imenujemo *modelirnik*, na podlagi podatkov iz običajnih katalogov oziroma meritev izračuna parametre modela (glej sliko 3.6).

```

===== SPICEMOD 2.1 === Diode (D) ===== 02-03-1997

.MODEL Name (D1N4005) = D1N4005           Affects Parameter:
      Type (Ge, Si, GaAs) = Si
      Rated Forward Current, IF =          1.000 A   All Parameters
      Medium Current (IF/10), IM =         0.100 A   IS
      Voltage at IM, VM =                  0.800 V   IS
      Low Current (IF/100), IL =           0.010 A   N
      Voltage at IL, VL =                  0.695 V   N
      High Current (10 * IF), IH =         10.000 A   RS
      Voltage at IH, VH =                  1.419 V   RS
      DC Blocking Voltage, VR =            600.000 V  BV
      Maximum Reverse Current, IR =        10.000 uA  IBV
      Junction Capacitance, CJ =           32.000 pF  CJO
      at Voltage (near 1 volt), VJ =       1.000 V   CJO
      Reverse Recovery Time, trr =         3.000 us   TT

===== DIODE SPICE PARAMETERS:

.MODEL D1N4005 D (IS=2.55N RS=42M N=1.75 BV=600 IBV=10U
+ CJO=42.4P VJ=.75 M=.333 TT=4.32U)
* 600 Volt 1 Amp 3 us Si Diode 02-03-1998

SELECT WITH ARROWS,TYPE NEW DATA F1=HELP F3=RESET ALL Esc WHEN DONE

```

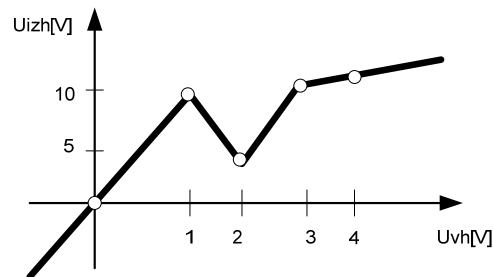
Slika 3.6 Modeliranje diode 1N4005 s programom SpiceMod (Intusoft). Rezultat (model) je prikazan z nagnjeno pisavo.

Z nakupom posebnega razvojnega orodja lahko ustvarjamo tudi lastne notranje modele, ki jih opišemo z ustreznim programskim jezikom (npr. C++), prevedemo in nato vgradimo v simulator. Tem modelom pravimo tudi *kodni modeli*³⁶.

Pri *tabelaričnih modelih*³⁷ je karakteristika elementa opisana v obliki tabele. Za napetosti, ki ležijo med podanimi točkami, simulator z interpolacijo izračuna ustrezno vrednost toka. Najpogosteje se ti modeli uporabljajo za modeliranje nelinearnih analognih gradnikov in logičnih gradnikov.

³⁶ Angl. code based models.³⁷ Angl. look-up models.

Zgled: Ojačevalnik z odsekoma linearno prenosno karakteristiko



Slika 3.7 Prenosna karakteristika

Prenosno karakteristiko bomo modelirali s posebnim napetostno krmiljenim napetostnim virom, katerega lastnosti določimo s tabelo. To je notranji model z imenom Pwl. Ker ostri prehodi med segmenti pogosto povzročajo konvergenčne težave, jih lahko omehčamo z opcijo `input_domain`.

```
A_ods_lin in out pwl_1
.model pwl_1 Pwl(xy_array=[
+0 0
+1 10
+2 5
+3 10
+4 12
+ ] input_domain=100m fraction=true)
```

Slika 3.8 Tabelarni model ojačevalnika z odsekoma linearno prenosno karakteristiko³⁸ (glej sliko 3.7). Prenosna karakteristika je podana s tabelo ($X = U_{vh}$ in $Y = U_{izh}$). Znak + pomeni, da gre za nadaljevanje prejšnje vrstice.

3.2. Zunanji modeli (makromodeliranje)

V simulator so vgrajeni (opisani s programsko kodo) le modeli najpogostejših elementov, s katerimi se srečujejo uporabniki. Za druge elemente (npr. integrator, stikalo, tiristor, tunelska dioda itd.) moramo sami tvoriti ustrezen model. Ker ti modeli niso vgrajeni v simulator, jim pravimo **zunanji modeli**. Tvorjenje zunanjega modela imenujemo tudi **makromodeliranje**. Kakovostno makromodeliranje je ena izmed najzahtevnejših aktivnosti, ki jih izvajamo v okviru simulacij. Ločimo tri vrste zunanjih modelov: strukturni modeli, vedenjski modeli (kodni in tabelarni) in kombinirani modeli.

Potem ko smo tvorili določen model, nas v večini primerov pri simulaciji več ne zanima, kaj se dogaja v njegovi notranjosti. Zaradi večje preglednosti lahko strukturo modela skrijemo v določen objekt, ki ga pri simulatorju SPICE imenujemo *podvezje*. Podvezje (subcircuit) je del vezja, katerega notranjost lahko skrijemo. Definiramo ga z identifikatorjem `.SUBCKT`, uporabimo pa tako, da pred ime podvezja dodamo črko X.

³⁸ Simulator Intusoft SPICE.

Sintaksa opisa:

```
.SUBCKT <ime_podvezja> <n1 n2 ...>
* n1, n2 ... so izhodna oziroma vhodna vozlišča.
```

```
  Opis vezja
```

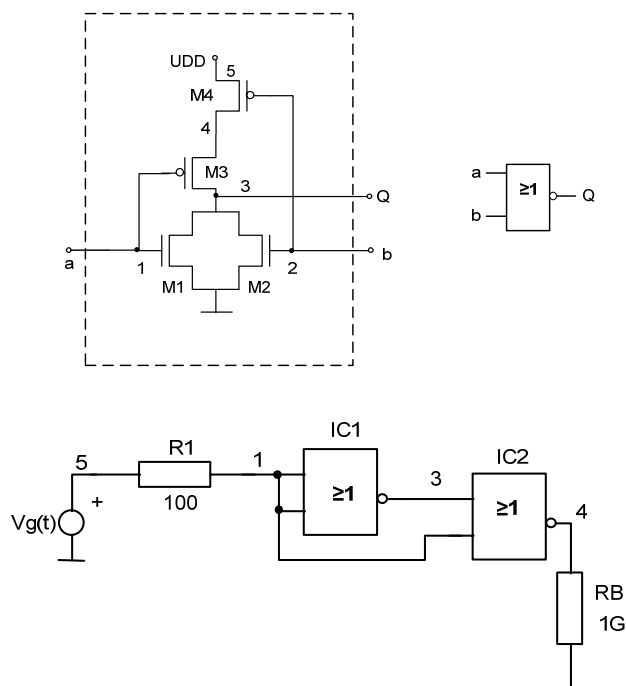
```
.ENDS [ime]
```

Sintaksa klica:

```
X<referenčno ime> <m1 m2 ..> <ime_podvezja>
```

Zgled:

NOR vrata smo definirali kot podvezje.



Slika 3.9 NOR vrata smo definirali kot podvezje in nato podvezje uporabili pri opisu glavnega vezja

Zgled:

```
.SUBCKT NOR      1          2          3
*               a          b          Q
* CMOS NOR vrata
* Shema je na datoteki NOR_2.DWG
* Verzija 5.6.2010

VDD 5 0 5V
M1 3 1 0 0 MN_08 W=2U L=2U
M2 3 2 0 0 MN_08 W=2U L=2U
M3 3 1 4 5 MP_08 W=4U L=2U
M4 4 2 5 5 MP_08 W=4U L=2U
.ENDS NOR

* Opis glavnega vezja
R1 5 1 100
VG 5 1 PULSE (0 5)
XIC1 1 1 3 NOR
XIC2 3 1 4 NOR
RB 4 0 1G

*
.MODEL MN_08 NMOS VTO=0.7 GAMMA=0.4 KP=110U LAMBDA=0.04
.MODEL MP_08 PMOS VTO=-0.7 GAMMA=0.57 KP=50U LAMBDA=0.05

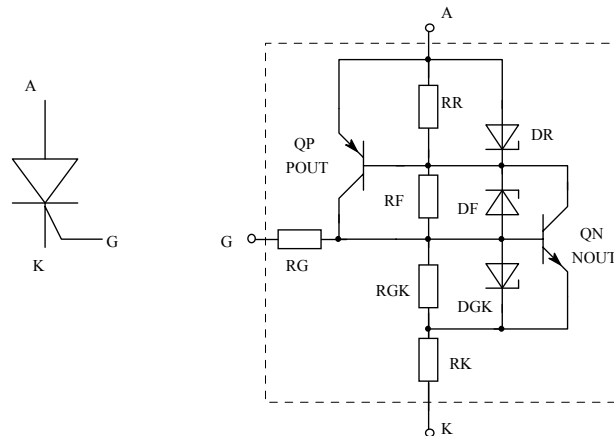
* Ukazi za analizo
:
.END NOR
```

3.2.1. Strukturni modeli

Če je model zgrajen na podlagi poznavanja strukture elementa, ga imenujemo **strukturni model**. Pri teh zunanjih modelih moramo najprej poiskati ustrezno strukturo in nato določiti vrednosti parametrov. Strukturo določimo na podlagi poznavanja fizikalnega dogajanja ali pa s pomočjo analogije s podobnimi elementi oziroma komponentami oziroma s sintezo³⁹.

V večini primerov imajo elementi, ki spadajo v isto skupino (npr. različne vrste tiristorjev), tudi enako strukturo modela in se razlikujejo le po vrednosti parametrov. Za nekatere najbolj pogoste zunanje modele obstajajo računalniška orodja, ki nam pomagajo določiti konkretne vrednosti parametrov. V program vnesemo podatke iz kataloga in nato dobimo konkreten zunanji model, ki je običajno definiran v obliki podvezja. Na slikah 3.10 in 3.12 je za zgled prikazan konkreten model tiristorja S2N4171.

Zgled: modeliranje tiristorja S2N4171:



Slika 3.10 Struktura splošnega tiristorskega modela

```

SPICEMOD 2.1 ==== SCR (X) ===== 02-05-1998

.SUBCKT Name (S2N4171) = S2N4171           Affects:
Peak Repetitive Forward Blocking V., VDRM = 300.000 V  BV(ZF), RF
Peak Repetitive Reverse Blocking V., VRRM = 300.000 V  BV(ZR), RR
Gate to Cathode Reverse Voltage, VRGM = 5.000 V      BV(ZGK)
RMS Forward Current, IT(rms) = 8.000 A              All param.
Peak Gate Current, IGM = 2.000 A                    RG
Peak Gate Voltage, VGM = 10.000 V                   RG
Peak Forward Blocking Current, IDRM = 1.000 mA      IBV(ZF), RF
Peak Reverse Blocking Current, IRRM = 1.000 mA      IBV(ZR), RR
Gate Trigger Current, IGT = 10.000 mA              RGK
Gate Trigger Voltage, VGT = 0.750 V                 RGK
Forward (on) Voltage, VTM = 1.400 V                 RK, RC
at Current (near 2*IT), ITM = 16.000 A             RK, RC
Holding Current, IH = 10.000 mA                     RGK
Turn-On Time (td + tr), t(on) = 1.000 us           TF
Turn-Off Time, t(off) = 15.002 us                  TR
Forward Voltage Application Rate, dv/dt = 50.000 V/us CJC, CJE

```

Slika 3.11 V katalogu poiščemo podatke o tiristorju S2N4171 in jih vnesemo v modelirnik SpiceMod (Intusoft)

³⁹ T. Dogša: *Uvod v analizo in sintezo nelinearnih vezij*, zapiski predavanj (v pripravi).

```

*****
*SYM=SCR
.SUBCKT S2N4171 1 2 3
*   TERMINALS:  A G K
*   300 Volt  8 Amp  SCR  06-30-1992
QP  6 4 1  POUT OFF
QN  4 6 5  NOUT OFF
RF  6 4    600K
RR  1 4    400K
RGK 6 5   112
RG  2 6    4.61
RK  3 5    8.13M
DF  6 4    ZF
DR  1 4    ZR
DGK 6 5   ZGK
.MODEL ZF  D (IS=3.2F IBV=100U BV=300 RS=90K)
.MODEL ZR  D (IS=3.2F IBV=100U BV=400)
.MODEL ZGK D (IS=3.2F IBV=100U BV=5)
.MODEL POUT PNP (IS=3.2P BF=1 CJE=670P)
.MODEL NOUT NPN (IS=3.2P BF=100 RC=32.5M
+ CJE=670P CJC=134P TF=179N TR=25.5U)
.ENDS
*****

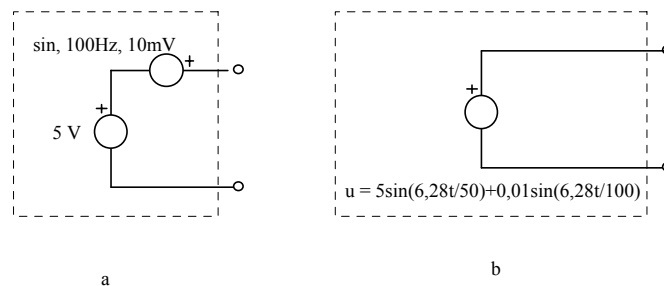
```

Slika 3.12 Zunanji model tiristorja S2N4171, ki ga napravi program SpiceMod (Intusoft)

3.2.2. Vedenjski modeli

Kadar opišemo obnašanje (vedenje) elementa z matematičnimi izrazi oziroma vezjem, ki nima nobene povezave z resnično strukturo elementa, ki ga modeliramo, potem takšen model imenujemo **vedenjski model** (glej sliko 3.13).

Zgled: Model 5 V usmernika, ki ima valovitost 10 mV.



Slika 3.13 Strukturni model (a) in vedenjski model, opisan s kodo (b)

Najbolj preproste vedenjske modele lahko tvorimo s pomočjo polinomskih generatorjev ali pa s pomočjo ustreznega programskega jezika.

Tvorjenje vedenjskih modelov podpirajo novejšje verzije simulatorjev, ki omogočajo opis skoraj poljubne odvisnosti med električnimi veličinami, temperaturo, frekvenco in časom. Na ta način lahko opišemo skoraj poljuben vedenjski model. Na sliki 3.14 je prikazan opis navadne diode⁴⁰.

```
// Ideal Diode
module diode (a, c);
  inout a, c;
  electrical a, c;
  parameter real IS = 1.0e-14; // saturation current
  real idio;
  /**
   * Calculate nonlinear current through diode depending on
   * - thermal voltage $vt (at ambient temperature of
   *   simulated circuit) and
   * - voltage between terminals
   */
  analog begin
    idio = IS * (limexp(V(a,c)/$vt) - 1);
    I(a,c) <+ idio;
  end
endmodule
```

Slika 3.14 Opis obnašanja diode z Verilog-AMS jezikom⁴¹

Naslednji zgledi so izbrani simulatorja Intusoft Spice. Vedenjski model opišemo z naslednjim stavkom:

```
<oznaka elementa> <vozlišča> <lastnost> = <opis karakteristike>
```

Oznaka elementa je lahko R ali C ali L ali B. Opis funkcije ima podobno sintakso kot programski jezik BASIC. V opisu karakteristike lahko uporabimo naslednje spremenljivke: TEMP, i, u, t, FREQ.

Karakteristiko navadnega in krmiljenega nelinearnega rezistivnega dvopola lahko opišemo z naslednjim stavkom:

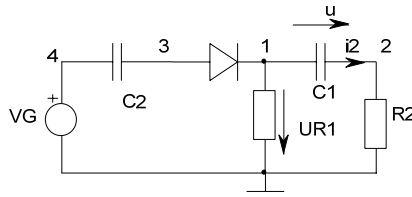
```
B[oznaka] <vozl.1> <vozl.2> <I=<aritm. izraz>>|<U=<aritm. izraz>>
```

⁴⁰ <http://en.wikipedia.org/wiki/Verilog-AMS>, prebrano 16. 3. 2010.

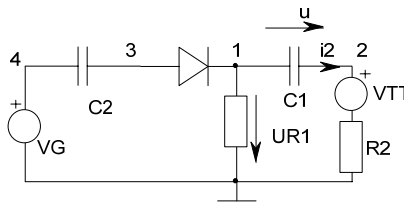
⁴¹ <http://www.vhdl.org/verilog-ams/htmlpages/overview.html>, prebrano 16. 3. 2010.

Zgled 1: Nelinearni kondenzator

V narisnem vezju ima kondenzator C1 naslednjo karakteristiko: $C = e^{u_{i2}} + u_{R1}$



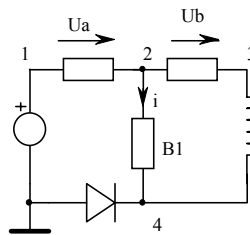
Rešitev: C1 1 2 C=exp (V(1,2)*I(VTT))+V(1)

**Zgled 2: Komparator**

B1 3 0 v=v(1,2) < 0 ? 5 : .1

* Če je razlika $v(1)-v(2)$ manjša od 0 potem $v(3)=5V$ sicer $v(3)=.1V$

Zgled 3: Nelinearni napetostno krmiljen dvopol B1: $i = \cos(u_a) + u_b$



Rešitev: B1 2 4 I=cos (V1,2))+V(2,3)

3.2.3. Preverjanje ustreznosti modela

Ker je pravilnost rezultata simulacije najbolj odvisna od kakovosti modelov, je treba preveriti njihovo veljavnost. Najbolje je, da preverimo vsak model, ki ga bomo uporabili pri simulaciji. Model je treba **zagotovo preveriti**, kadar:

- gre za velike tokove ali napetosti,
- gre za zelo majhne tokove ali napetosti,
- gre za šum, ki ga je treba upoštevati,
- predvidevamo vpliv temperature,
- je pomembna stopnja popačenja,
- gre za visoke frekvence oziroma zelo hitre signale.

Preverjanje veljavnosti modelov zahteva veliko znanja in izkušenj. Model lahko preverjamo na naslednje načine:

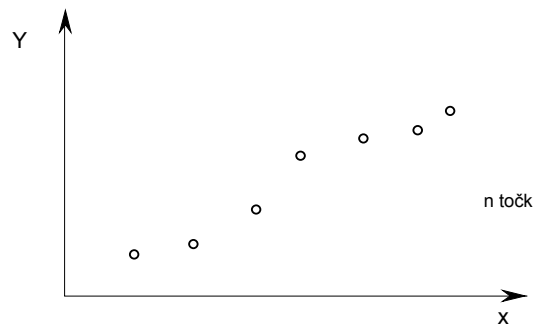
- pregledamo tekstovni opis modela (preverimo npr., ali je vpliv temperature vključen med parametri) ali pa
- simuliramo meritev.

3.2.4. Uporaba polinomske regresije pri modeliranju elementov

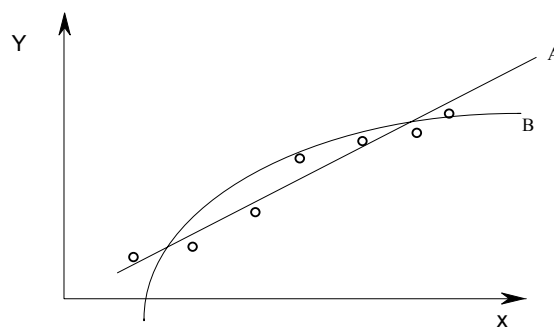
V mnogih primerih ne poznamo analitičnega izraza za karakteristiko elementa (npr. tunnelska dioda, tlivka) in je karakteristika določena z meritvijo ter podana v obliki grafa ali tabele. Najenostavnejši pristop je uporaba tabelaričnega modela. Če simulator ne pozna tabelaričnih modelov, moramo poiskati takšno funkcijo, ki se točkam najbolj prilega, in jo nato uporabiti v vedenjskem modelu. Modeliramo lahko celotno karakteristiko ali pa samo del, ki ga pri simulaciji potrebujemo.

Neznano funkcijsko odvisnost lahko iščemo v obliki:

- linearne funkcije $y = p_0 + p_1x$
- polinoma stopnje k : $y = p_0 + p_1x + p_2x^2 + \dots + p_kx^k$
- druge funkcije (npr. $y = e^x$)



Slika 3.15 Karakteristika je podana v obliki grafa oziroma merilnih točk



Slika 3.16 A: linearna regresija, B: polinomska regresija

Ker najpogosteje iščemo neznano odvisnost v obliki polinoma, ki vključuje tudi linearno funkcijo, bomo v nadaljevanju obravnavali samo ta primer. Postopek, s katerim določimo koeficiente polinoma, imenujemo *polinomska regresija*. Simulator SPICE ima vgrajene 4 notranje modele, katerih karakteristika je polinom poljubne dimenzije in stopnje. Ti modeli so krmiljeni tokovni ali pa napetostni generatorji.

Postopek

Iz grafa oziroma tabele odčitamo n točk, ki naj bodo glede na absciso enakomerno razporejene.

x_1	y_1
x_2	y_2
\vdots	\vdots
x_n	y_n

Glede na število točk in izbrano stopnjo polinoma obstaja naslednja povezava:

$n = k$	polinom gre natanko skozi vse točke
$n > k$	polinom se optimalno prilega, glede na določen kriterij
$n < k$	rešitev ne obstaja – previsoka stopnja polinoma

Za izračun koeficientov polinoma potrebujemo računalniški program⁴², saj je postopek numerično zelo zahteven. Ko smo izračunali koeficiente, lahko pregledamo, kako dobro se polinom prilega resnični karakteristiki elementa. Za vsako točko x lahko izračunam ustrezni $y = f(x)$, ki pa se v splošnem razlikuje od resnične vrednosti y .

x_1	y_1	$y(x_1)$
x_2	y_2	$y(x_2)$
\vdots	\vdots	
x_n	y_n	$y(x_n)$

Obstaja več možnih kriterijev oziroma pokazateljev kakovosti prileganja. Najbolj pogosti so:

- **standardna napaka SE,**
- **korelacijski koeficient r^2 in**
- **vsota kvadratov odstopanj od posameznih točk.** $\sum_{i=1}^n (y(x_i) - y_i)^2$

Stopnjo polinoma izberemo glede na najboljše prileganje. Pri tem pazimo, da se polinom najboljše prilega tistemu delu karakteristike, ki se bo v simulaciji uporabljal. Če smo trdno prepričani, da vemo, kateri del karakteristike bo simulator uporabljal, lahko modeliramo samo ta del. Na ta način dosežemo boljše prileganje polinoma.

Težavnost modeliranja bomo na koncu tega poglavja ilustrirali z daljšim zgledom, v katerem nastopa tunnelska dioda. Izbrali smo jo zato, ker ima zelo nelinearno karakteristiko.

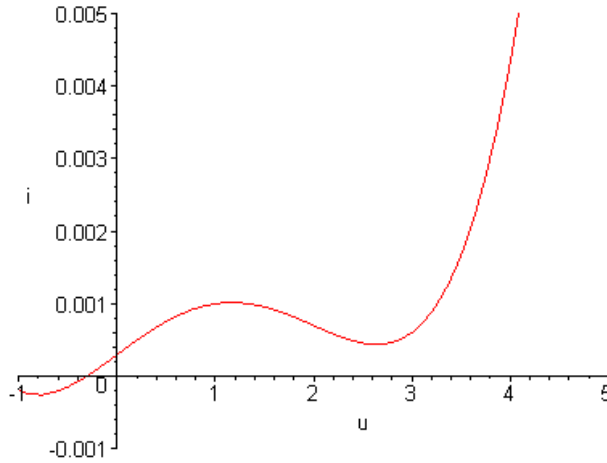
3.2.5. Nelinearni krmiljeni viri

Karakteristiko napetostno krmiljenega elementa $i = f(u)$ lahko izrazim v obliki polinoma:

⁴² P. Lutus: *Polynomial/Trig Regression Data Fit*, <http://www.arachnoid.com/polysolve/index.html>, 26.2.2008.

$$y = f(x_a) = p_0 + p_1 x_a + p_2 x_a^2 + \dots + p_k x_a^k$$

Pogosto nimamo samo ene, ampak več krmilnih veličin, npr.: $i = f(u_a, u_b, u_c)$



Slika 3.17 Zgled za karakteristiko nelinearnega rezistivnega dvopola

Posplošen zapis polinoma:

dimenzija polinoma $nd \equiv$ število krmilnih veličin

$$nd=1 \quad y = f(x_a) = p_0 + p_1 x_a + p_2 x_a^2 + \dots + p_k x_a^k$$

$$nd=2 \quad y = f(x_a, x_b) = p_0 + p_1 x_a + p_2 x_b + p_3 x_a^2 + p_4 x_a x_b + p_5 x_b^2 + p_6 x_a^3 + p_7 x_a^2 x_b + \dots$$

$$nd=3 \quad y = f(x_a, x_b, x_c) = p_0 + p_1 x_a + p_2 x_b + p_3 x_c + \dots$$

Zgled: Določi dimenzijo, stopnjo in ustrezne koeficiente

$$i = 3u - 4u^3$$

$$i = f(u) = 0 + 3u + 0u^2 - 4u^3 \quad p_0 = 0, p_1 = 3, p_2 = 0 \text{ in } p_3 = -4$$

To je enodimenzionalni polinom tretje stopnje.

Vaja Določi dimenzijo, stopnjo in ustrezne koeficiente

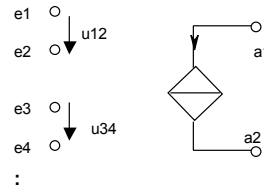
$$u = 4 + u_2^2 + 10u_1 u_2$$

$$i = 10 - i_1 i_2$$

Simulator SPICE ima vgrajene 4 notranje modele, katerih karakteristika je polinom poljubne dimenzije in stopnje. Ti modeli so krmiljeni tokovni oziroma napetostni generatorji.

Nelinearni napetostno krmiljen tokovni generator

$$i = f(u_{12}, u_{34}, u_{56}, \dots)$$

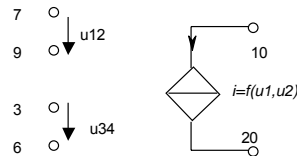


Sintaksa:

G[ime]<a₁><a₂>[POLY(n_d)] <e₁> <e₂> [e₃ e₄...] [p₀ p₁...] IC=U₁₂, U₃₄...

Zgled:

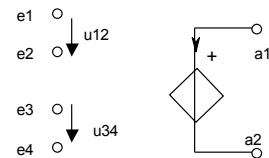
$$i = 4 + u_2^2 + 10u_1u_2$$



G 10 20 POLY(2) (7 9 3 6) (4 0 0 1 10)

Nelinearni napetostno krmiljen napetostni generator

$$u = f(u_{12}, u_{34}, u_{56}, \dots)$$

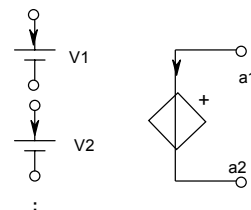


Sintaksa:

E[ime]<a₁><a₂>[POLY(n_d)] <e₁> <e₂> [e₃ e₄...] [p₀ p₁...] [IC=U₁₂, U₃₄...]

Nelinearni tokovno krmiljen napetostni vir

$$u = f(i_1, i_2, i_3, \dots)$$

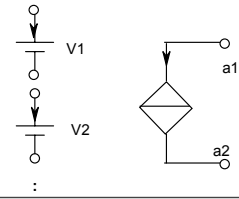


Sintaksa:

H[ime]<a₁><a₂>[POLY(n_d)] V₁ [V₂...] [p₀ p₁...] [IC=I₁, I₂...]

Nelinearni tokovno krmiljen tokovni vir

$$i = f(i_1, i_2, i_3, \dots)$$



Sintaksa:

F[ime]<a₁> <a₂>[POLY(n_d)] V₁[V₂...] [p₀ p₁...] [IC=I₁, I₂...]

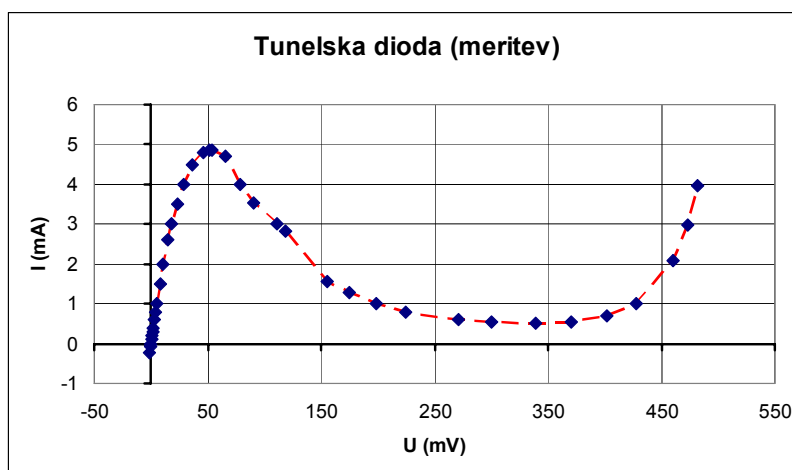
Zgledi modelov

Modele komponent v večini primerov ponujajo kar proizvajalci na svojih spletnih straneh. Če modela ne najdemo, ga moramo ustvariti sami. Za zgled bo prikazano modeliranje tunelske diode. Dodan bo še kratek opis modela MOS tranzistorja in makromodel operacijskega ojačevalnika.

3.2.6. Modeliranje tunelske diode 1N3716

Za ilustracijo modeliranja smo izbrali tunelsko diodo, ki ima izrazito nelinearno karakteristiko. Omejili se bomo samo na modeliranje rezistivne karakteristike. Dinamične lastnosti modeliramo z dodajanjem ustreznih reaktivnih elementov.

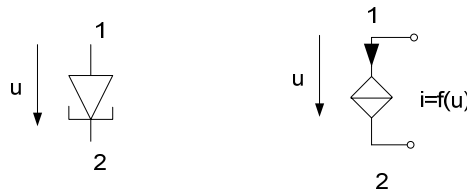
u[mV]	I[mA]	u[mV]	I[mA]
-1,1	-0,24	53,6	4,87
-0,4	-0,09	66	4,7
0	0	79	4,01
0,5	0,1	91	3,52
1	0,2	111	3,02
1,5	0,3	119	2,83
2	0,4	155	1,55
2,9	0,6	175	1,29
4	0,8	199	0,99
5,1	1	225	0,8
7,9	1,5	271	0,6
10,1	2	300	0,54
15,3	2,6	339	0,51
18,6	3	370	0,55
23,2	3,5	402	0,71
29	4	428	1,01
37	4,5	460	2,08
45,7	4,8	473	2,98
52	4,86	482	3,97



Slika 3.18 Izmerjena karakteristika tunelske diode 1N3716 v obliki tabele (levo) in grafa (desno)

V nadaljevanju bomo primerjali tri modele tunelske diode, ki so napravljeni z ustreznim napetostno krmiljenim tokovnim virom (glej sliko 3.19) :

- **vedenski model, ki temelji na polinomu,**
- **vedenski model, ki temelji na fizikalnem modelu,**
- **tabelarični model.**



Slika 3.19 Tunelska dioda in njen model, ki je napetostno krmiljen tokovnim virom

Polinomski vedenjski model:

$$i = p_0 + p_1 u + p_2 u^2 + \dots + p_k u^k$$

Na karakteristiki (grafu) smo odčitali 16 enakomerno razmaknjenih točk in izračunali koeficiente polinomov različnih stopenj. Za celotno področje izmerjenih vrednosti rešitev ni konvergirala. Zato smo se odločili samo za področje z negativno upornostjo (od 0 do 340 mV). Najbolje se je prilegal ($r^2 = 0,9975$ SE = $1,311E-4$) polinom 5. stopnje:

$$i = 3,523E-5 + 0,222u - 3,43888u^2 + 20,46115u^3 - 54,13565u^4 + 53,22137u^5$$

- Za modeliranje diode lahko uporabimo nelinearni napetostno krmiljen tokovni generator $i = f(u)$, ki ga ustreza primitivnemu elementu z začetno črko G. Če simulator pozna splošni vedenjski model, potem namesto tunelske diode⁴³ uporabimo nelinearni element z oznako B:

$$\begin{aligned} \text{B2 } 1 \ 2 \ \text{I} = & 3.523\text{E-}5 + 0.222*\text{V}(1,2) - 3.43888*\text{V}(1,2)^2 \\ & + 20.46115*\text{V}(1,2)^3 - 54.13565*\text{V}(1,2)^4 + 53.22137*\text{V}(1,2)^5 \end{aligned}$$

Fizikalni vedenjski model

$$i = I_s (e^{u/U_T} - 1) + I_p \frac{u}{U_p} e^{(1-u/U_p)} + G_e u$$

Glede na obliko grafa smo izbrali naslednje vrednosti parametrov:

$$\begin{aligned} I_s &= 1,5 \cdot 10^{-11} \text{ A} \\ U^T &= 25 \text{ mV} \\ I_p &= 4,8 \text{ mA} \\ U_p &= 0,05 \text{ V} \\ G_e &= 0,5 \text{ mS} \end{aligned}$$

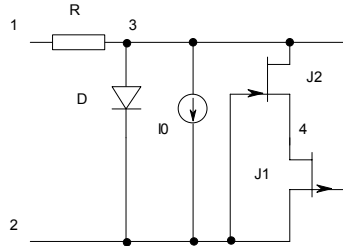
Tabelarični model

Celotno tabelo (glej sliko 3.18) smo vnesli v napetostno krmiljen nelinearni upor.

⁴³ Prikazan je primer, ko je dioda med vozliščema 1 in 2.

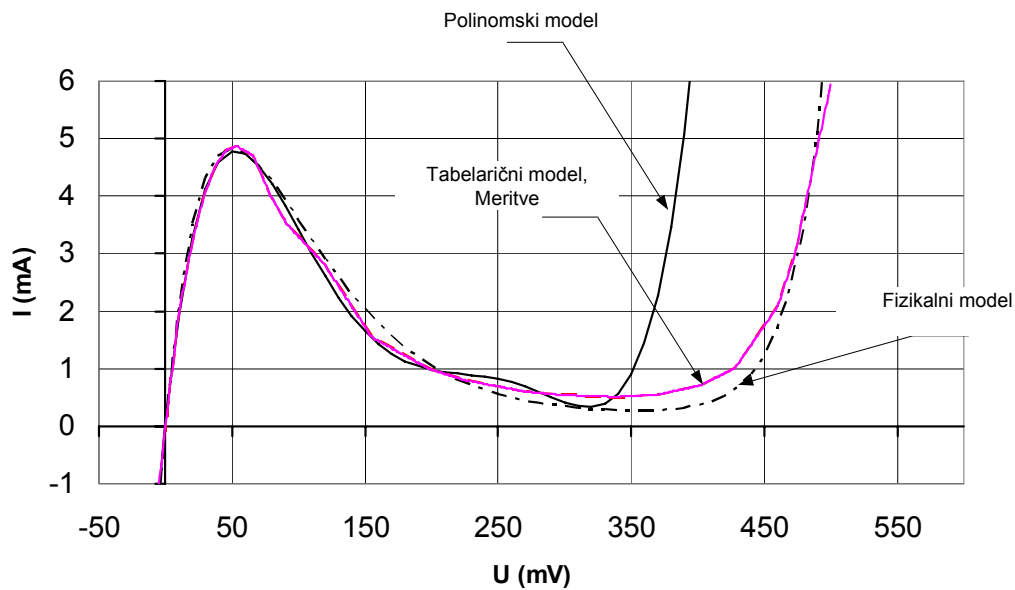
Strukturni model

Obstaja tudi strukturni model⁴⁴ tunelske diode, do katerega pridemo s sintezo⁴⁵. Značilna oblika karakteristike je dosežena s pozitivno povratno vezavo dveh spojnih FET tranzistorjev.



Slika 3.20 Strukturni model tunelske diode

Primerjava modelov



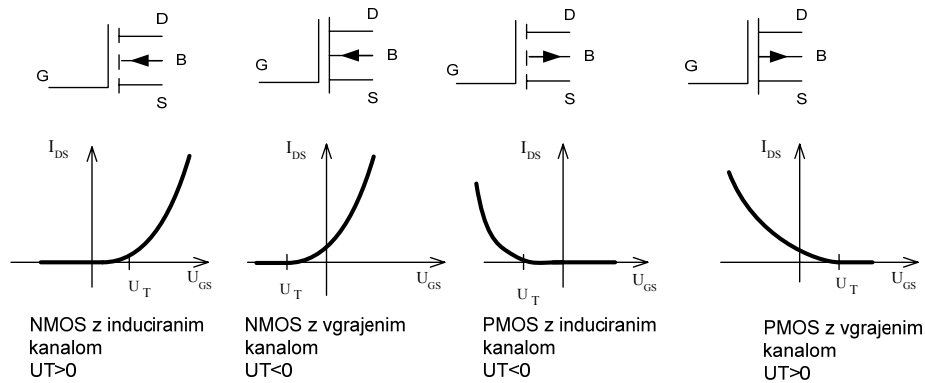
Slika 3.21 Primerjava modelov z izmerjenimi vrednostmi

Primerjava (rezistivnih karakteristik) kaže na izredno dobro prileganje tabelaričnega modela, ki je tako majhno, da se na grafu niti ne opazi.

⁴⁴ Karl Heinz Muller: "RF/Microwave Analysis", Intusoft Newsletter Issue #51, november 1997, (<http://www.intusoft.com/nlhtm/nl51.htm>).

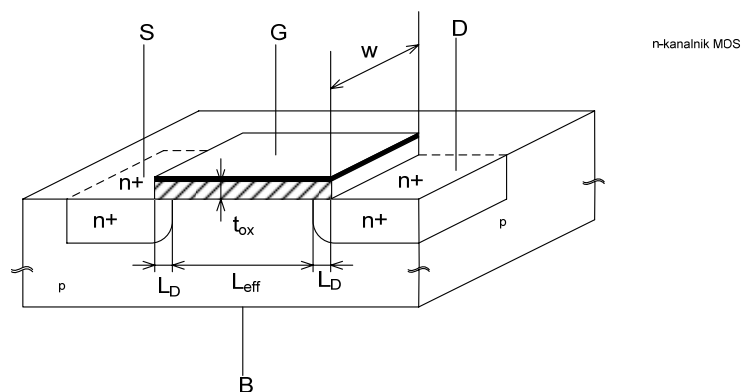
⁴⁵ Glej npr. T. Dogša: "Uvod v analizo in sintezo analognih vezij", zapiski predavanj (v pripravi).

3.2.7. MOS tranzistor



Slika 3.22 Karakteristike raznih vrst MOS tranzistorjev

MOS tranzistorje ločimo glede na vrsto kanala (p ali n) in glede na način ustvarjanja (MOS z induciranim kanalom ali MOS z vgrajenim kanalom). Ker ima MOS tranzistor z vgrajenim kanalom enak analitični izraz za karakteristiko kot MOS z induciranim kanalom, ne obstaja dva različna modela, saj lahko to definiramo z ustreznim predznakom pragovne napetosti.



Slika 3.23 Struktura in transkonduktančna karakteristika NMOS tranzistorja

Obstaja več modelov, ki jih izbiramo s parametrom LEVEL. Najbolj enostaven in tudi najmanj natančen je Shichman-Hodgesov model, ki se uporablja predvsem pri približnih izračunih enosmernih razmer in pri načrtovanju.

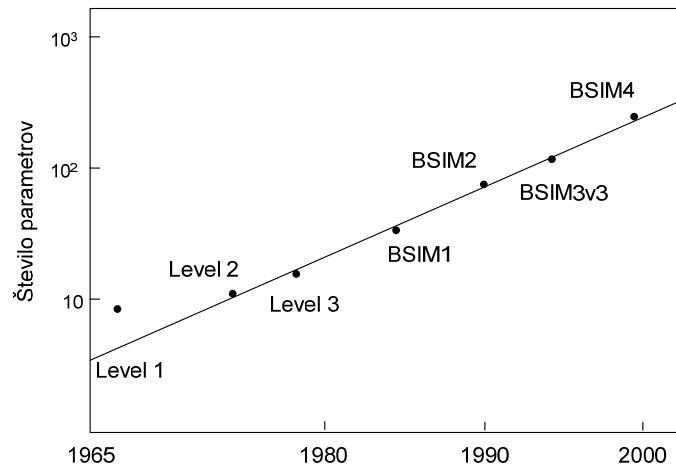
LEVEL = 1 → Shichman-Hodgesov model (prvi in tudi najpreprostejši model)

LEVEL = 2 → Analitični model

:

:
 LEVEL = 50 BSIM1
 BSIM2
 BSIM3
 BSIM4

S krajšanjem minimalnih dimenzij število parametrov in kompleksnost novih modelov strmo naraščata (glej sliko 3.24), saj je treba upoštevati določene pojave, ki jih lahko pri večjih dimenzijah zanemarimo.



Slika 3.24 Kronološki razvoj MOS modelov⁴⁶

Topološke in geometrijske lastnosti MOS tranzistorja opišemo z naslednjim stavkom:

```
M<referenčna oznaka> <D> <G> <S> <B> <ime modela> [<L=vrednost>
+ W=vrednost> <AD=vrednost> <PD=vrednost> <AS=vrednost>
+ <PS=vrednost>]
```

D, G, S, B : vozlišča, na katera so priključeni ponor, vrata, izvor in okolica (bulk)
 ime modela: ime modela, ki definira karakteristiko tipičnega MOS tranzistorja

Parametri, ki se uporabljajo samo za modeliranje integriranega MOS tranzistorja:

L : dolžina vrat
 W : širina vrat
 AD: površina ponora
 PD: obseg ponora
 AS: površina izvora
 PS: obseg izvora

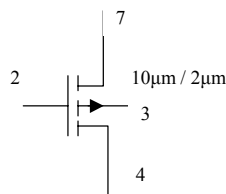
⁴⁶ M. Chan and C. Hu, "The Engineering of BSIM for the Nano-Technology Era and Beyond", Technical Proceedings of the 5th International Conference on Modeling and Simulation of Microsystems, pp. 662–665, San Juan, Puerto Rico, april 21–25, 2002.

MOS model je opisan z identifikatorjem .MODEL

```
.MODEL <ime modela> <tip kanala> LEVEL=<varianta modela>
+ <niz parametrov>
```

<ime modela>	: vsak model je treba poimenovati
<tip kanala>	: NMOS ali PMOS
<varianta modela >	: naravno število, s katerim izbiramo varianto modela
<niz parametrov>	: niz parametrov, ki so vezani na varianto modela

Zgled: Opis in model integriranega MOS tranzistorja



Opis topoloških in geometrijskih lastnosti:

```
M2 7 2 4 3 MN W=10U L=2U
```

Preprost model (Shichman-Hodgesov model, LEVEL = 1):

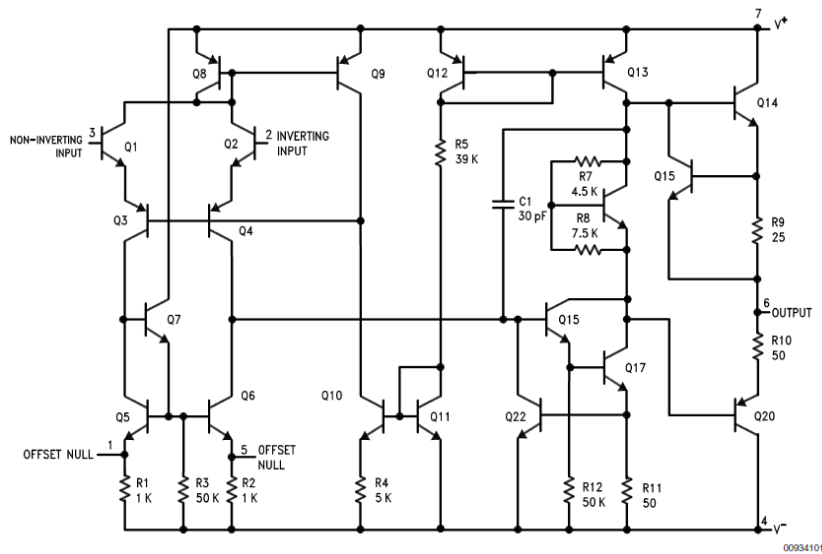
```
.MODEL MN NMOS LEVEL=1 VTO=0.5 KP=120U GAMMA=0.5 LAMBDA=0.04 TOX=7.7n
```

Bolj natančne rezultate dobimo z modelom MOS BSIM3v3 (LEVEL=8):

```
.MODEL MN NMOS LEVEL=8 MOBMOD=1 CAPMOD=2 NOIMOD=3
+ K1=5.0296E-1 K2=3.3985E-02 K3=-1.136E+00 K3B=-4.399E-01 NCH=2.611E+17 VTH0=4.979E-01
+ VOFF=-8.925E-02 DVT0=5.000E+01 DVT1=1.039E+00 DVT2=-8.375E-03 KETA=2.032E-02
+ PSCBE1=3.518E+08 PSCBE2=7.491E-05 DVT0W=1.089E-01 DVT1W=6.671E+04 DVT2W=-1.352E-02
+ UA=4.705E-12 UB=2.137E-18 UC=1.000E-20 U0=4.758E+02 DSUB=5.000E-01 ETA0=1.415E-02
+ ETAB=-1.221E-01 NFACTOR=4.136E-01 EM=4.100E+07 PCLM=6.948E-01 PDIBLC1=3.571E-01
+ PDIBLC2=2.065E-03 DROUT=5.000E-01 A0=2.541E+00 A1=0.000E+00 A2=1.000E+00 PVAG=0.000E+00
+ VSAT=1.338E+05 AGS=2.408E-01 B0=4.301E-09 B1=0.000E+00 DELTA=1.442E-02 PDIBLCB=3.222E-01
+ W0=2.673E-07 DLC=3.0000E-08 DWC=9.403E-08 DWB=0.000E+00 DWG=0.000E+00
+ LL=0.000E+00 LW=0.000E+00 LWL=0.000E+00 LLN=1.000E+00 LWN=1.000E+00 WL=0.000E+00
+ WW=-1.297E-14 WWL=-9.411E-21 WLN=1.000E+00 WWN=1.000E+00
+ TNOM=27.0 AT=3.300E+04 UTE=-1.800E+00 KT1=-3.302E-01 KT2=2.200E-02 KT1L=0.000E+00
+ UA1=0.000E+00 UB1=0.000E+00 UC1=0.000E+00 PRT=0.000E+00 CGDO=1.300E-10 CGSO=1.200E-10
+ CGBO=1.100E-10 CGDL=1.310E-10 CGSL=1.310E-10 CKAPPA=6.000E-01
+ CF=0.000E+00 ELM=5.000E+00 XPART=1.000E+00 CLC=1.000E-15 CLE=6.000E-01
+ RDSW=3.449E+02 CDSC=0.000E+00 CDSCB=1.500E-03 CDSCD=1.000E-03 PRWB=-2.416E-01
+ PRWG=0.000E+00 CIT=4.441E-04 TOX=7.575E-09 NGATE=0.000E+00 NLX=1.888E-07
+ ALPHA0=0.000E+00 BETA0=3.000E+01 AF=1.3600E+00 KF=5.1E-27 EF=1.000E+00 NOIA=1.73E+19
+ NOIB=7.000E+04 NOIC=-5.64E-13 RSH=7.000E+01 LINT=-5.005E-08 WINT=9.403E-08 XJ=3.000E-07
+ JS=1.000E-05 JSW=0.000E+00 CJ=9.400E-04 CJSW=2.500E-10 MJ=3.400E-01 MJSW=2.300E-01
+ PB=6.900E-01
```

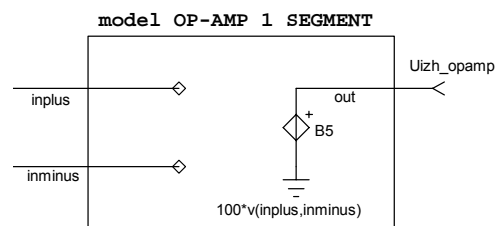
3.2.8. Modeliranje operacijskega ojačevalnika

Tipični ojačevalnik vsebuje nekaj deset tranzistorjev in uporov ter nekaj diod. Le načrtovalec, ki so mu znane struktura in vrednosti elementov, lahko za model ojačevalnika uporabi kar shemo vezja. Takšen model vezja je sicer najbolj natančen, vendar ni uporaben za druge ojačevalnike. Zato se za modeliranje poljubnih operacijskih ojačevalnikov uporablja niz strukturnih makro modelov, ki se razlikujejo po svoji kakovosti in kompleksnosti. Z atributom "red modela" označujemo njegovo natančnost. Ker ne obstaja noben dogovor o tem, kaj pomeni npr. model drugega reda, velja samo to, da višji ko je red, bolj je model natančen. V nadaljevanju bomo za ilustracijo prikazali nekaj modelov operacijskih ojačevalnikov.

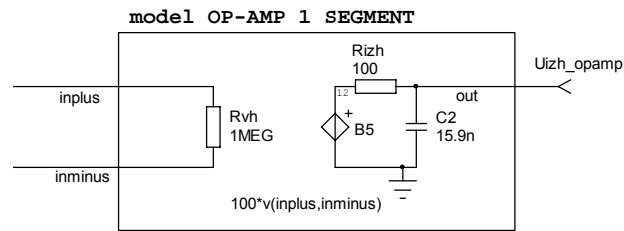


Slika 3.25 Vezje operacijskega ojačevalnika UA741 (varianta, ki jo izdeluje National Semiconductor)

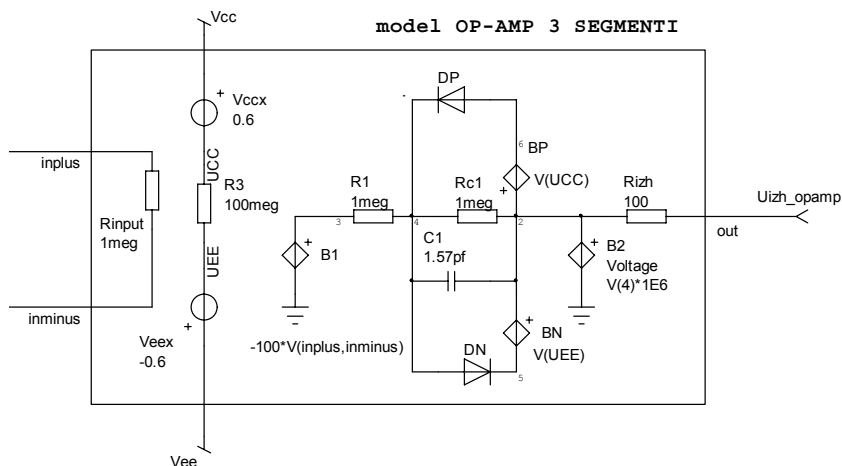
Model prvega reda:



- $A_u = 1000$ (krmiljen vir B5)
- $R_{izh} = 0$
- $R_{vh} = \infty$
- $f_{zg} = \infty$
- ni nobenih popačenj, temperaturne odvisnosti ter šuma, preostale napetosti, ni sofaznega ojačenja
- ni omejitev amplitud in izhodnega toka, deluje pravilno samo za simetrične napajalne napetosti

Model drugega reda:

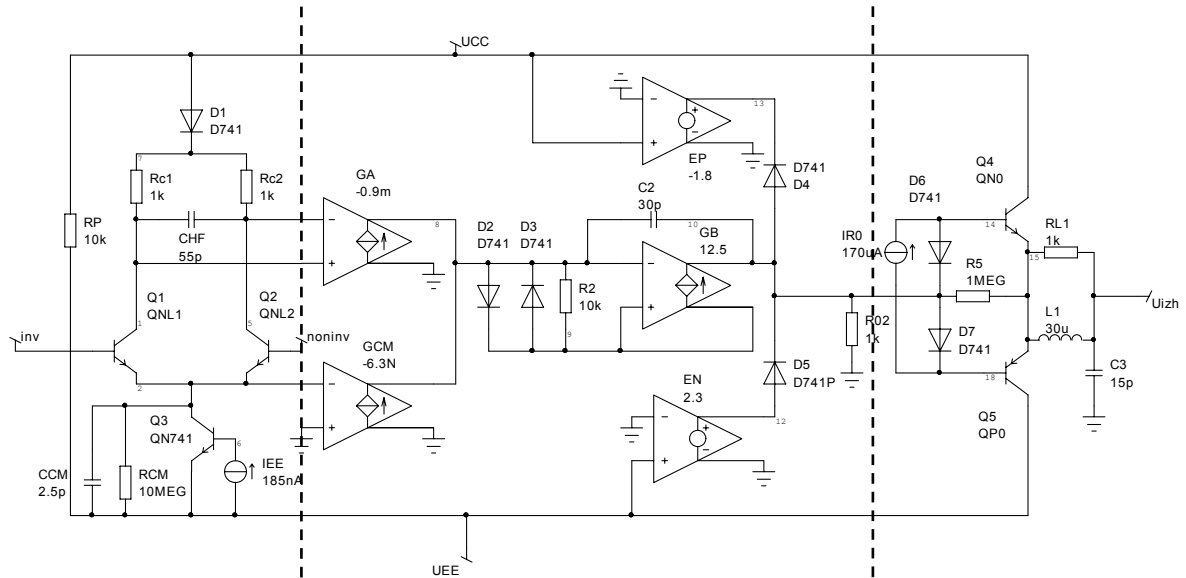
- $A_u = 1000$ (krmiljen vir B5)
- $R_{izh} = 100$
- $R_{vh} = 1\text{MEG}$
- $f_{zg} = 1/(2\pi R_{izh}C_2) = 100\text{kHz}$
- ni nobenih popačenj, temperaturne odvisnosti ter šuma, preostale napetosti, ni sofaznega ojačenja
- ni omejitvev amplitud in izhodnega toka, deluje pravilno samo za simetrične napajalne napetosti

Model tretjega reda:

- $A_u = 1000$ (krmiljen vir B1)
- $R_{izh} = 100$
- $R_{vh} = 1\text{MEG}$ (R_{input})
- $f_{zg} = 1/(2\pi R_{c1}C_1) = 100\text{kHz}$
- $U_{izhmax} = V_{cc} - V_{ccx}$
- $U_{izhmin} = V_{ee} - V_{vee}$
- ni nobenih popačenj, temperaturne odvisnosti ter šuma, preostale napetosti, ni sofaznega ojačenja, deluje pravilno samo za simetrične napajalne napetosti

Model četrtega reda⁴⁷:

Model četrtega reda [MEARES,1988] vsebuje kar 30 elementov. Struktura je primerna za večino bipolarnih operacijskih ojačevalnikov. Za zgled na sliki 3.26 je izbran model ojačevalnika uA741.



Slika 3.26 Model četrtega reda [MEARES, 1988]. Konkretni vrednosti parametrov se nanašajo na UA741.

Model je sestavljen iz treh delov: vhodne, vmesne in izhodne stopnje. Modelirane so skoraj vse lastnosti razen šuma in temperature. Vprašljivo je tudi delovanje modela v primeru nesimetričnih napajalnih napetosti, saj so nekateri notranji elementi vezani na maso.

⁴⁷ M. Alexander, D. Bowers: "SPICE-Compatible Op Amp Macro-Models", Analog Devices Application Note AN-138, 1990.

```

* CONNECTIONS:   NON-INVERTING INPUT
*                | INVERTING INPUT
*                | | POSITIVE POWER SUPPLY
*                | | | NEGATIVE POWER SUPPLY
*                | | | | OUTPUT
*                | | | | |
.SUBCKT UA741A  1 2 3 4 5
*
QNI1 10 2 13 QNI1
QNI2 12 1 13 QNI2
Q3 13 14 4 QN741
IEE 4 14 185NA
CCM 13 4 2.5PF
RCM 13 4 10MEG
RC1 11 10 1K
RC2 11 12 1K
CHF 10 12 55PF
D1 3 11 D741
RP 3 4 10K
GA 0 15 12 10 0.9M
GCM 0 15 13 0 6.3N
R2 15 0 100K
D2 15 0 D741 ;OFF
D3 0 15 D741 ;OFF
C2 15 16 30PF
GB 16 0 15 0 12.5
RO2 16 0 1000
D4 16 17 D741P ;OFF
EP 17 0 3 0 -1.8 1
D5 18 16 D741P ;OFF
EN 0 18 0 4 -2.3 1
D6 19 16 D741
D7 16 20 D741
IRO 20 19 170UA
RR0 16 21 1MEG
Q4 3 19 21 QNO
Q5 4 20 21 QPO
L1 21 5 30U
RL1 21 5 1K
.MODEL D741 D CJO=3PF
.MODEL QN741 NPN
.MODEL QNI1 NPN NF=1.5 BF=111 IS=8E-16 CJE=3PF
.MODEL QNI2 NPN NF=1.5 BF=144 IS=8.3E-16 CJE=3PF
.MODEL QNO NPN BF=150 CJC=3P IS=1E-14
.MODEL QPO PNP BF=150 CJC=3P IS=1E-14
.MODEL D741P D RS=1M
.ENDS UA741A

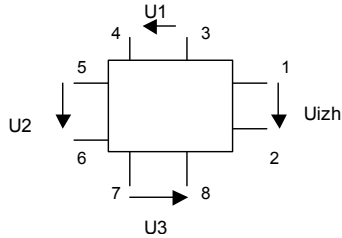
```

Slika 3.27 Lista povezav za model iz slike 3.26

3.3. Vaje

Vaja 1 – Modeliranje seštevalnika

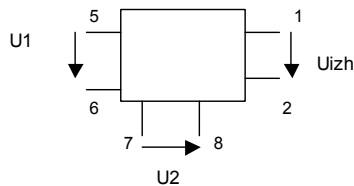
Tvori zelo preprost model seštevalnika: $U_{izh} = U_1 + U_2 + U_3$. Mogoči sta dve rešitvi.



Model izboljšaj tako, da ne bo nobenih omejitev glede priključenih vhodnih napetosti.

Vaja 2 – Modeliranje množilnika

Tvori model množilnika: $U_{izh} = 4U_1U_2$. Kako lahko modeliramo zakasnitev, vhodno in izhodno upornost? Tvori podvezje.



Vaja 3 – Modeliranje funkcijskega generatorja

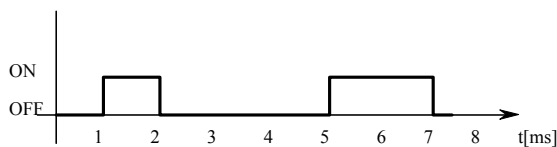
Potrebujemo model generatorja, ki ima notranjo upornost 2 K in napetost: v intervalu $\pi/2 \leq t \leq 2\pi$ je U sinusna napetost 50 Hz, 2 V, v intervalu $0 \leq t < \pi/2$ je $U = 0$.

Vaja 4 – Modeliranje nelinearnega upora

Modeliraj dvopol, ki ima karakteristiko: $i = 1,5[mA/V]u + 1[mA/V^2]u^2$

Vaja 5 – Modeliranje stikala

Modeliraj stikalo ($ROFF = \infty$, $RON = 1m\Omega$), ki se bo vklopljalo po narisanim programu.



Vaja 6 – Modeliranje nelinearnega kondenzatorja

Tvori model kondenzatorja, ki ima karakteristiko: $C(u) = 5 + 4u^2$ [pF]

Vaja 7 – Modeliranje časovno variantnega in temperaturno občutljivega upora

Tvori model upora, katerega karakteristika se spreminja v odvisnosti od časa t in temperature T : $R(t, T) = 1000 + 1000\sqrt{t} + 5T$

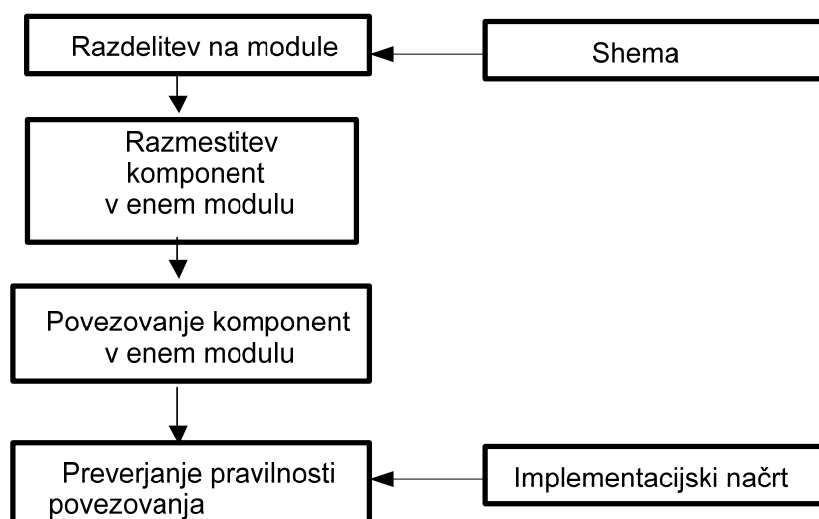
4. RAZMEŠČANJE IN POVEZOVANJE

Potem ko smo napravili načrt vezja in ga s simulacijo preverili, sledi načrtovanje implementacije, kjer je treba povezati komponente na čim manjši površini s čim krajšimi povezavami.

Tipična področja, ki se srečujejo s tem problemom, so:

- načrtovanje tiskanin (PCB – Printed Circuit Board),
- načrtovanje integriranih vezij (VLSI – Very-Large-Scale Integration),
- logistika.

Postopek načrtovanja tiskanine je sestavljen iz štirih osnovnih korakov, ki jih prikazuje slika 4.28.

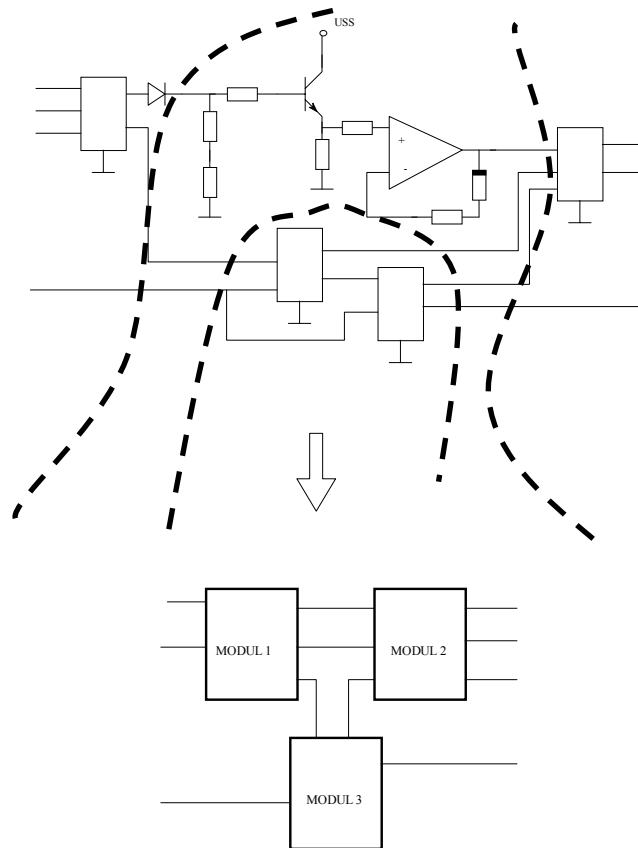


Slika 4.28 Osnovni koraki pri načrtovanju implementacije

V nadaljevanju bodo podrobneje opisani vsi štirje koraki.

4.1. Razdelitev na module

Kadar je vezje preobsežno, ga ni mogoče implementirati na eni tiskanini (kot en modul). V tem primeru je potrebna razdelitev vezja na več delov oziroma modulov.



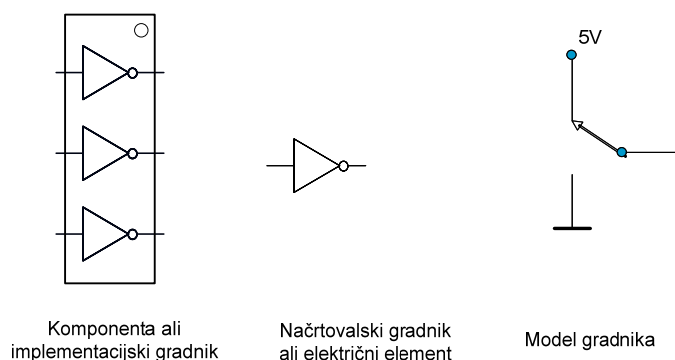
Slika 4.29 Zgled vezja, ki smo ga razdelili na tri module

Tipični kriteriji, ki vplivajo na odločitev o razdelitvi:

- Delitev naj bo taka, da bo potrebnih čim manj povezav med moduli.
- Tehnološki kriterij – elementi, ki bodo izdelani z isto tehnologijo, naj bodo skupaj.
- Funkcionalni kriterij – elementi, ki tvorijo določeno zaključeno funkcionalno enoto, naj bodo skupaj (npr. ojačevalnik, usmernik).
- Motnje – elementi oziroma deli vezja, ki utegnejo motiti (moteče vplivati), naj ne bodo skupaj z občutljivimi elementi.
- Hitrost – za veliko hitrost so potrebne kratke razdalje.
- Prostorske omejitve in zahteve – vsi elementi, ki imajo približno enako višino, naj bodo skupaj.

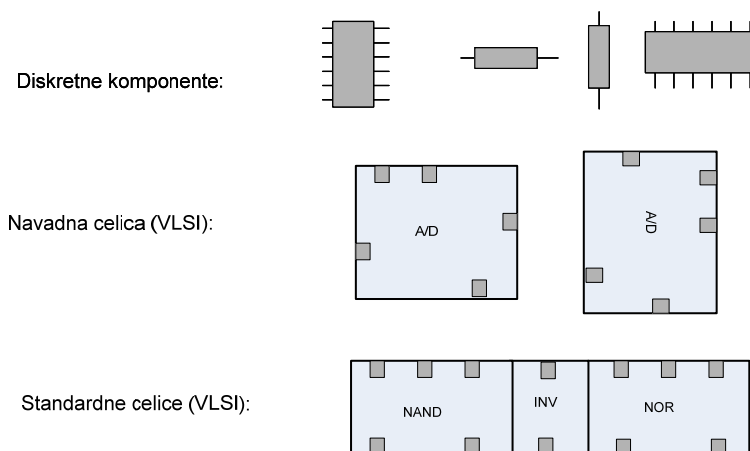
V večini primerov je treba med posameznimi kriteriji delati kompromise.

4.2. Razmeščanje komponent v modulu



Slika 4.30 Načrtovalec mora ločiti med komponento, načrtovalskim gradnikom in modelom

Komponenta: fizični element, ki ima najmanj dva priključka in vsebuje enega ali več načrtovalskih gradnikov. Pri razmestitvi na tiskanini lahko komponente v večini primerov poljubno obrnemo. Izjema so standardne celice⁴⁸, ki jih uporabljamo pri načrtovanju VLSI integriranih vezij.



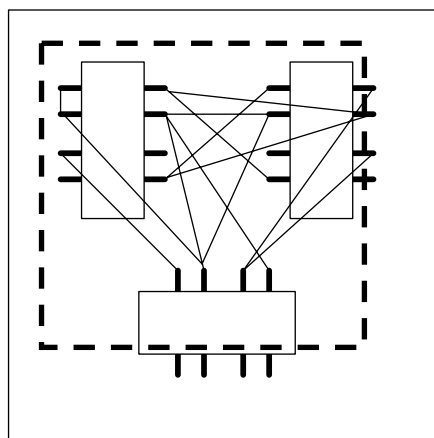
Slika 4.31 Tipične vrste komponent. Standardne celice morajo ležati v vrsti.

Ko komponente razmestimo, hkrati tudi definiramo površino modula.

⁴⁸ V mikroelektroniki so standardne celice gradniki, ki jih moramo polagati v vrsto.

Lastnosti kakovostne razmestitve:

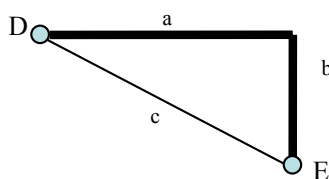
- je na čim manjši površini,
- ima zadosten prostor za povezave,
- ima kratke povezave,
- upoštevane so določene omejitve – predpisan je položaj glede na sosednje komponente (npr. VLSI standardne celice).



Slika 4.32 Za test ustreznosti razmestitve uporabljamo osnutek povezav

Razmestitev je lahko samodejna ali pa ročna. Ko so komponente razmeščene, izrišemo osnutek povezav⁴⁹. To so direktne povezave med priključki komponent, ki jih program zelo hitro nariše in oceni njihovo poznejšo dolžino (glej sliko 4.33) . Sedaj ima program že dovolj podatkov, da lahko:

- izmed več variant izberemo optimalno razmestitev,
- ocenimo potreben čas samodejnega povezovanja,
- identificiramo povezave, s katerimi bodo morebitne težave pri poznejši analizi integritete signalov (ta analiza bo obravnavana v poznejši poglavjih).



Slika 4.33 Ocena dolžine povezave med točkama D in E. Direktna povezava je hipotenuza c , Manhattan razdalja pa vsota $a + b$. Slednja je v povprečju zelo dober približek poznejše resnične povezave.

Optimalno razmestitev izmed več variant ocenimo vizualno ali kvantitativno z izračunom preproste stroškovne funkcije:

⁴⁹ Angl. rats nest diagram.

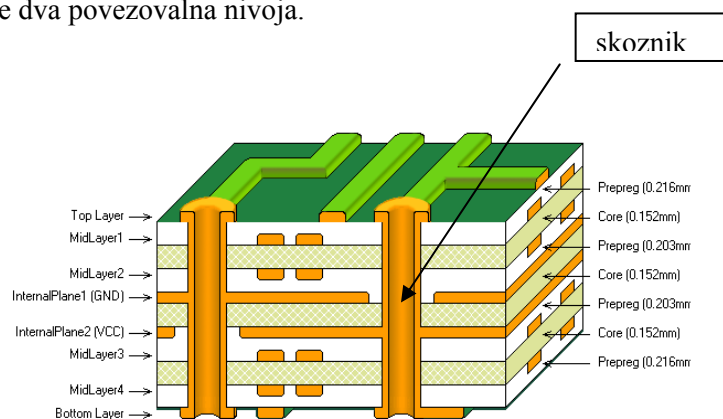
$$Q = a \sum_{i=1}^m d_i + b \cdot A \quad (4.1)$$

- Q: število kazenskih točk
 m: število povezav
 d_i: dolžina i-te povezave (direktna ali pa Manhattan)
 a, b: utežnostni faktor
 A: površina, ki jo zavzemajo komponente (črtkan pravokotnik na sliki 4.32)

Optimalna razmestitev je ovrednotena z najnižjim Q.

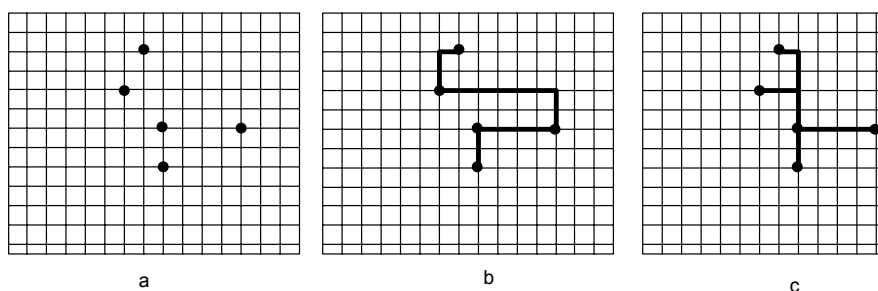
4.3. Povezovanje komponent v modulu

Problem križanja povezav – Problem križanja lahko rešimo z vertikalno povezavo (via, skozenik), ki poveže dva povezovalna nivoja.



Slika 4.34 Presek osemslojne tiskanine

Problem najkrajše povezave



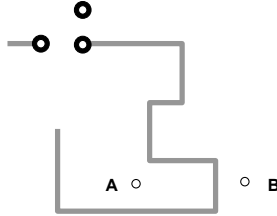
Slika 4.35 Točke (a) lahko povežemo na več načinov (b in c). Varianta c je boljša od b.

Samodejno povezovanje uvrščamo v problem **labirinta**⁵⁰. Iskanje poti iz labirinta je težak matematični problem, ki ga računalnik rešuje s pomočjo zahtevnih algoritmov. V začetnem

⁵⁰ Angl. maze.

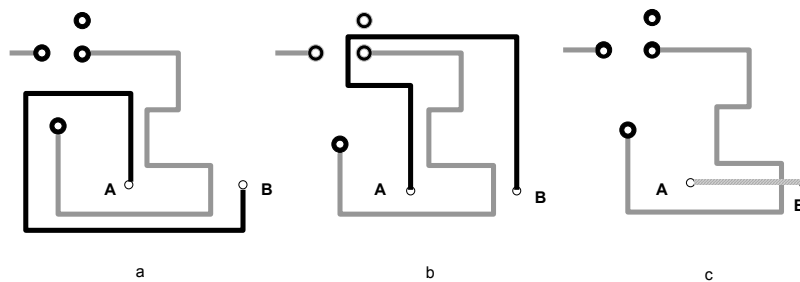
povezovanju obstaja več rešitev, katerih število pozneje pada. Pri iskanju rešitve je pogosto potrebna tudi pomoč načrtovalca.

Zgled: Želimo povezati točki A in B (glej sliko 4.36). V večini primerov obstaja več možnosti, včasih tudi nobena.



Slika 4.36 Točki A in B je treba povezati

Pri izbiri poti se program opira na določeno strategijo, katere parametre lahko nastavljam. S to strategijo lahko povezovalnik izbira alternative, ki nam najbolj ustrezajo. Strategija temelji na stroškovni funkciji, s katero izračunamo stroške določene alternative (glej tabelo 4.1).



Slika 4.37 Možne rešitve (glej sliko 4.36)

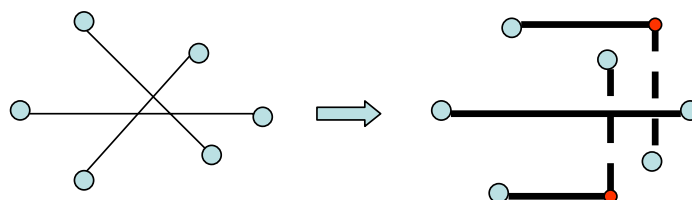
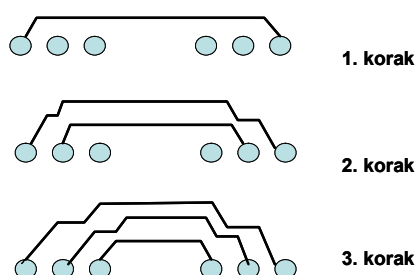
Obstajajo tri alternative: uporabimo dolgo povezavo na eni plasti (slika 4.37 a in b) ali uporabimo vertikalno povezavo in povezavo izvedemo na drugi plasti (slika 4.37 c). Povezovalnik bo izračunal vrednosti stroškovnih točk za vse alternative in izbral najcenejšo.

Tabela 4.1 Parametri, s katerimi vplivamo na strategijo

Parameter	Stroškovne točke (privzeta vrednost)	Območje
Stroški skoznika	10	2–20
Stroški širine kanala	3	0–10
Stroški dolžine povezave	1	1–5
:	:	:

Najpogostejši algoritmi:

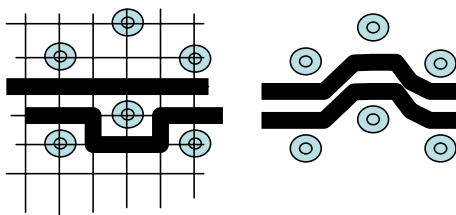
- Leejev algoritem⁵¹,
- Rip-up and retry,
- Rip-up and retry and push aside.

Slika 4.38 Osnutek povezav in povezave, ki jih ustvari koordinatni povezovalnik⁵²Slika 4.39 *Rip-up and retry and push aside* algoritem**Ostali parametri, ki vplivajo na potek povezovanja:**

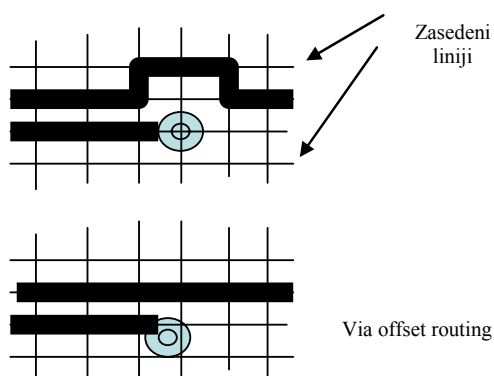
1. velikost ploščice,
2. razmestitev komponent,
3. število komponent,
4. število povezav,
5. širina povezave (trace size),
6. razdalja med povezavami (trace to trace clearance),
7. velikost kontaktne blazinice (pad size),
8. velikost mreže,
9. smer napredovanja.

⁵¹ Lee C. Y. : "An Algorithm for Path Connection and its Applications", IRE Transactions on Electronic Computers: 346–365, September, 1961.

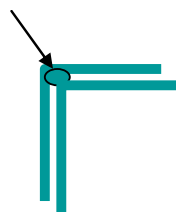
⁵² Sinonim: X-Y router.



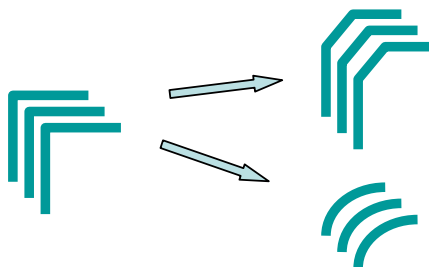
Slika 4.40 Povezovanje s pomočjo mreže (levo) in brez mreže (desno)



Slika 4.41 Odmik vertikalne povezave sprostí eno linijo



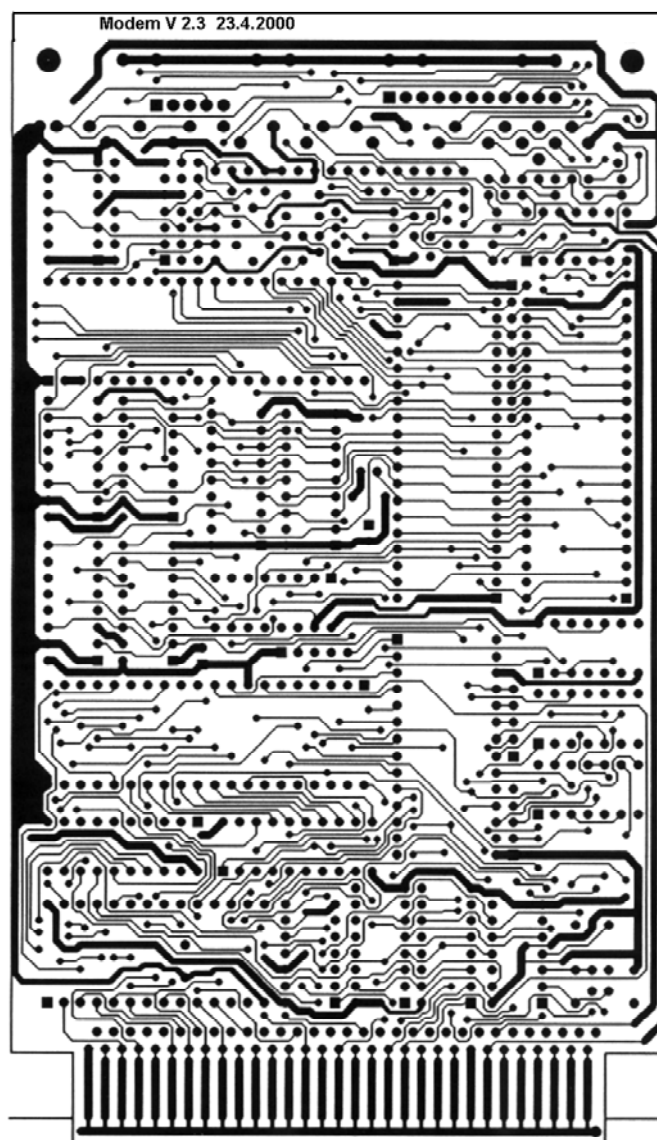
Slika 4.42 Če so robovi ostri, lahko mehurček prepreči kislini dostop do prevodne plasti



Slika 4.43 Oblanje robov preprečuje ugnezditev mehurčkov in neželeno sevanje na konicah

Kakovostno povezovanje:

1. Čim krajše povezave.
2. Ni ostrih robov.
3. Ni težav z integriteto signalov.
4. Modul ne ustvarja motenj (ustreza EMI zahtevam).

Zgled tiskanine:

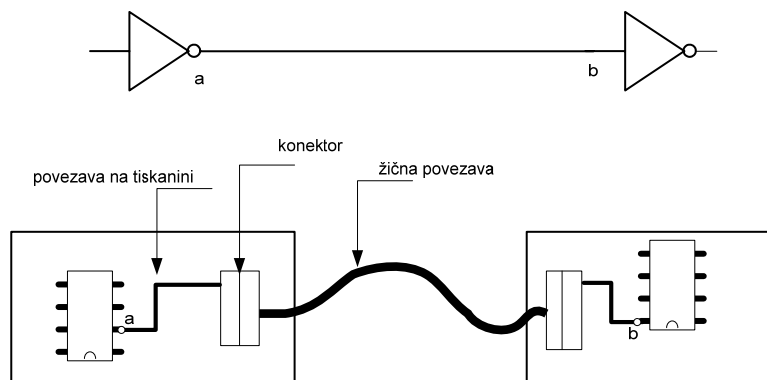
4.4. Preverjanje ustreznosti povezovanja

Potem ko smo napravili načrt za implementacijo (tiskanina ali pa maske za VLSI integrirana vezja), sledi preverjanje ustreznosti povezovanja. Tipični testi, s katerimi preverjamo ustreznost, so:

1. Preverjanje skladnosti s shemo (**LVS: Layout Versus Schematic**) – **Je topologija sheme identična topologiji implementacije?**
2. Preverjanje integritete signala (**SI: Signal Integrity analysis**) – **Kako vpliva posamezna povezava na obliko signala?**
3. Preverjanje vpliva povezav na lastnosti vezja – **Kako vplivajo vse povezave na lastnosti vezja?**
4. Preverjanje skladnosti z načrtovalskimi pravili (**DRC: Design Rule Check**) – **Ali smo pri načrtovanju implementacije kršili načrtovalska pravila?**
5. Preverjanje skladnosti z električnimi pravili (**ERC: Electrical Rule Check**) – **Ali so v vezju določene načrtovalske napake?**
6. Preverjanje EMI/EMC kompatibilnosti (**EMI: electromagnetic interference, EMC: electromagnetic compatibility**) – **Ali vezje ustreza EMI/EMC predpisom?**

Najbolj osnovni test je preverjanje skladnosti s shemo. Ostali se izvajajo po potrebi. V nadaljevanju bodo podrobneje razloženi nekateri testi.

4.5. Električne lastnosti povezav in priključkov



Slika 4.44 Realizacija povezave med točkama a in b je sestavljena iz petih segmentov: dveh povezav na tiskanini, dveh konektorjev in žične povezave.

Pri simulaciji oziroma analizi vezja se v večini primerov predpostavlja, da so vse povezave idealne, to je, da so brez izgub in frekvenčnih omejitev. Kot bo pokazano pozneje, ta predpostavka ne drži vedno. V nadaljevanju bomo obravnavali problematiko analize povezav. Podobni problemi se pojavljajo tudi znotraj komponente oziroma na nivoju integriranega

vezja⁵³. Povezavo med dvema točkama je lahko sestavljena iz enega ali pa več segmentov. Na sliki 4.44 je povezava med točkama a in b sestavljena iz povezav na tiskanini, konektorjev in žične povezave. Ker lahko vse te tri elemente obravnavamo na enak način, bomo v nadaljevanju uporabljali samo termin *povezava* oziroma *linija*.

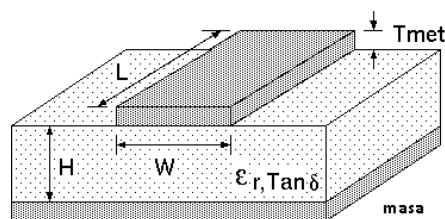
Tabela 4.2 Tipične zakasnitve⁵⁴

Vrsta povezave	Tipična dolžina	Zakasnitev
Od naprave do naprave (koaksialni kabel)	10m–1m	10ns–100ns
Od tiskanine do tiskanine (koaksialni kabel, ploščati kabel)	1m–10cm	1ns–10ns
Od komponente do komponente (stripline, microstrip)	10cm–1cm	100ps–1ns
Od tabletko do kontaktne blazinice(stripline, microstrip)	3cm–1cm	100ps–300ps
Znotraj čipa, (microstrip, poly)	1cm–1mm	10ps–100ps

Električne lastnosti povezave so odvisne od dimenzij in snovnih lastnosti materiala. Najpogostejše električne lastnosti, ki jih potrebujejo modeli povezav, so:

- karakteristična impedanca ($Z_0 = \sqrt{\frac{L}{C}}$)
- kapacitivnost na enoto dolžine (C)
- upornost na enoto dolžine (R)
- induktivnost na enoto dolžine (L)
- zakasnitev (td)

Te lastnosti moramo izračunati sami ali pa jih izračuna program, s katerim načrtujemo tiskanino.



Slika 4.45 Presek tipične povezave (microstrip) na tiskanini in pripadajoči parametri

Zgled:

Na tiskanini je linija v obliki microstripa, ki je široka 2,54 mm in dolga 300 mm. Na začetku povezave je vir z notranjo upornostjo 10 Ω. Povezava je zaključena z visoko upornostjo (100 k). Tiskanina, ki ima enak profil, kot ga prikazuje slika 4.45, ima naslednje lastnosti: specifična

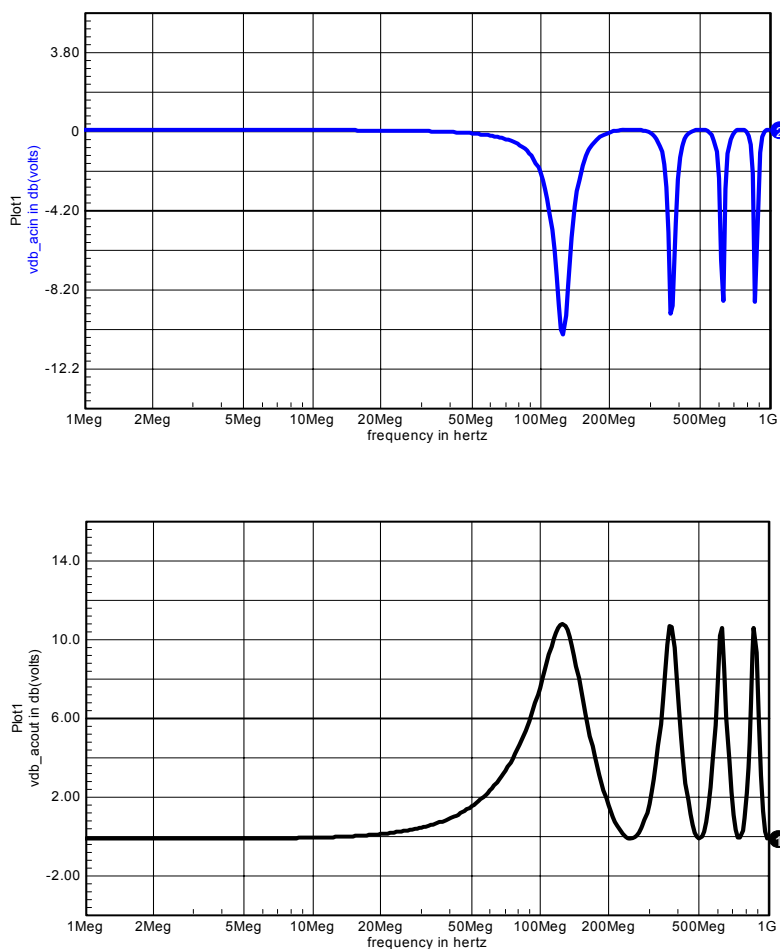
⁵³ ASIC layout.

⁵⁴ "New IBIS Models From Intel", Intusoft Newsletter, maj, 1993, <http://www.intusoft.com/nlpdf/nl30.pdf>, (prebrano 23. 2. 2010).

upornost bakra = $1,7 \cdot 10^{-8} \Omega\text{m}$, $T_{\text{met}} = 0,035 \text{ mm}$, $H = 1,55 \text{ mm}$, $\text{Tan} = 0,01$, $\epsilon_R = 5,5$, najvišja frekvenca = 100 MHz.

S pomočjo kalkulatorja⁵⁵ izračunamo lastnosti linije:

- karakteristična impedanca $Z_0 = 50 \Omega$,
- zakasnitev $t_d = 2 \text{ ns}$,
- induktivnost na enoto dolžine $L = 326 \text{ nH/m}$,
- kapacitivnost na enoto dolžine $C = 136 \text{ pF/m}$,
- upornost na enoto dolžine $R = 2,6 \Omega/\text{m}$,
- prevodnost substrata $G = 3913 \mu\text{S}$.



Slika 4.46 Frekvenčna karakteristika povezave iz zгледа. Z vidika vira (zgoraj) in z vidika bremena (spodaj). Do približno 20 MHz linija predstavlja enostaven kratek stik.

⁵⁵ Pomagamo si lahko z različnimi namenski kalkulatorji, kot je Microstrip Analysis/Synthesis Calculator, <http://mcalc.sourceforge.net>, (prebrano 23. 2. 2010).

4.6. Modeliranje povezav

Idealna povezava je imuna na sosednje povezave in nima nobene upornosti, kapacitivnosti in induktivnosti. Signal se prenese v neskončno kratkem času. Zaradi teh lastnosti ne spremeni oblike signala in ne povzroči nobene zakasnitve. Ker realna povezava nima takih lastnosti, spremeni obliko signala.

Pri nizkih frekvencah lahko opazimo le padec napetosti, ki nastane zaradi upornosti povezave. Ko se frekvenca večja, opazimo zakasnitev signalov in spremembo njihove oblike. Ko se valovna dolžina najvišje frekvence v spektru začne približevati dimenzijam vezja, smo na meji, kjer še veljajo Kirchhoffovi zakoni oziroma vezja več ne moremo obravnavati kot koncentrirano vezje. Npr. 1 GHz signal ima valovno dolžino 30 cm. Pojavljati se začnejo fenomeni, ki so značilni za valovode (npr. odboji⁵⁶, prenehaji).

Spremenjena oblika se kaže kot: zakasnitev, padec amplitude, popačenje in pojav dušenih oscilacij. Tipični vzroki za te pojave so:

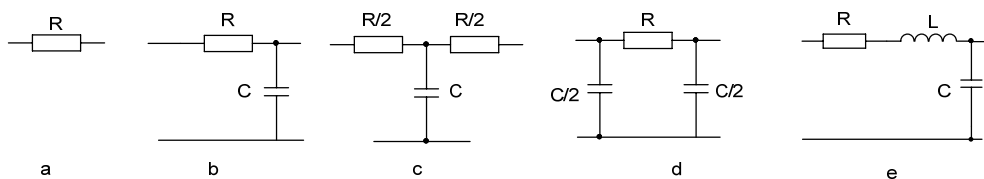
1. Ker linija ni pravilno zaključena, se pojavijo odboji. Linijo je treba obravnavati kot valovod. Tipična meja je $t_r = 2,5..10t_d$. Če v tej formuli izberemo sredino, se linija obnaša kot valovod, če velja $t_r < 5t_d$.
2. Upornost, kapacitivnost in induktivnost povezave.
3. Vpliv sosednjih povezav (presluh).

Lastnosti linij (povezav) so odvisne od geometrijskih in snovnih parametrov ter oblike oziroma frekvence signala. Modele povezav delimo v tri skupine:

- koncentrirani modeli (lumped element model),
- porazdeljeni modeli (distributed transmission line model) in
- modeli valovoda (transmission line model).

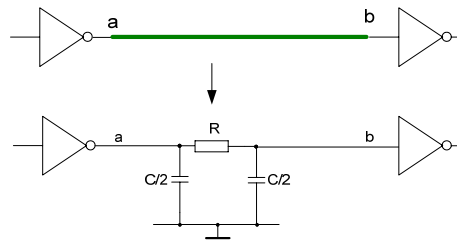
4.6.1. Koncentriran model linije

Koncentriran model povezave je vezje, ki je sestavljeno iz nekaj linearnih reaktanc in upornosti (glej sliko 4.48). Uporabljamo ga za modeliranje osnovnih lastnosti povezav, ki se pojavljajo pri nizkih hitrostih (padec napetosti in zakasnitev). Ker je sestavljen iz primitivnih gradnikov, je uporaben v vseh verzijah simulatorja SPICE.



Slika 4.47 Tipični segmenti

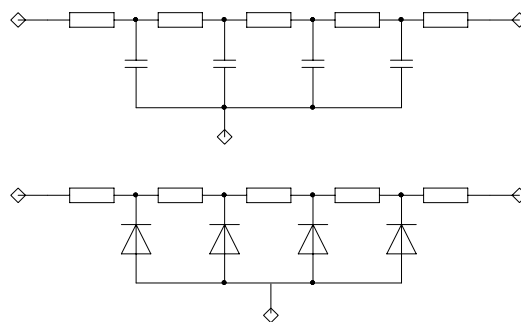
⁵⁶ Če je povezava zaključena z neustrezno impedanco, lahko pride v nekaterih primerih do odbojev oziroma prenehajev.

Slika 4.48 Preprost koncentriran model linije, ki ima upornost R in kapacitivnost C

4.6.2. Model enakomerno distribuirane linije

Pri dolgih linijah se lahko zgodi, da so koncentrirani modeli premalo natančni. V takih primerih se odločimo za porazdeljen model, ki je sestavljen iz N zaporedno vezanih enakih segmentov (glej sliko 4.47). Vse fizične lastnosti se tako razdelijo na N delov. Npr., če je upornost linije 10Ω in imamo 5 segmentov, ima vsak $R = 2\Omega$. Ko gre število N proti neskončnosti, postane takšen model zelo dober približek resnične povezave. Večinoma zadostujejo 3 do 4 segmenti [WESTE,2005]. Za modeliranje lahko uporabimo zunanji ali pa enega izmed dveh notranjih modelov⁵⁷: model enakomerno distribuirane linije ali model izgubne prenosne linije (lossy transmission line).

Notranji model enakomerno distribuirane linije (Distributed URC/URD Transmission Line) je tropol, ki ga sestavlja niz zaporedno vezanih RC členov (glej sliko 4.49). Pri integriranih vezjih lahko v modelu namesto kondenzatorjev uporabimo zaporno polarizirane diode. Najpogosteje se ta model uporablja za povezave znotraj čipa.



Slika 4.49 Zgled URC modela enakomerno distribuirane linije, ki je sestavljen iz štirih RC ali pa RD členov. Členi imajo strukturo vrste T (glej sliko 4.47).

Model linije je sestavljen iz dveh stavkov: v prvem je definiran element, katerega ime se mora začeti s prvo črko U . Sledita dolžina linije (L) v metrih in število segmentov (N). Z večanjem števila segmentov se večja natančnost, vendar se hkrati zelo povečuje čas simulacije. Če N izpustimo, potem ga SPICE sam izračuna po notranji formuli. Vsak U element se sklicuje na stavek `.model`, v katerem moramo navesti ime notranjega modela (URC) in vnesti še ostale parametre.

⁵⁷ Med HSPICE in PSPICE ter Intusoftovim SPICE simulatorjem obstajajo velike razlike v notranjih modelih prenosnih linij.

Zgled:

```

U1 2 3 0 POVEZAVA L=50U N=3
U4 7 9 0 POVEZAVA L=100U N=5
.MODEL POVEZAVA URC FMAX=100MEG RPERL=2 CPREL=10PF

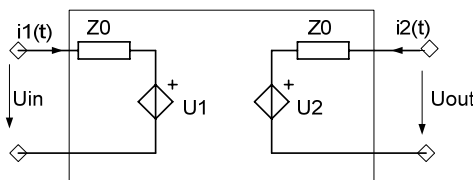
```

V vezju sta dve liniji: U1 in U5. Prva je dolga 50 μm , druga pa 100 μm . Obe se sklicujeta na model z imenom POVEZAVA. Linija ima upornost 2 Ω/m in kapacitivnost 10 pF/m. Najvišja frekvenca, ki nas še zanima, je 100 MHz. V liniji U1 bodo v modelu uporabljeni trije segmenti, v liniji U4 pa pet.

4.6.3. Model brezizgubnega valovoda

Kadar velja $t_r < 5t_d$, je treba povezavo obravnavati kot valovod. Če lahko zanemarimo upornost povezave, potem lahko uporabimo notranji model brezizgubne linije (lossless transmission line)⁵⁸, s katerim lahko modeliramo idealni valovod in pojave, ki so z njim povezani (npr. odboji). Ker izgube dušijo tudi odboje, model prikazuje najbolj neugoden primer odbojev. Element ima značilno prvo črko T. Čeprav je vgrajen v vse verzije simulatorja SPICE, njegov simbol ni standardiziran. Ekvivalentni model je preprosto dvovhodno vezje (glej sliko 4.50), ki ustrezno zakasni tok in napetost. Zakasnitev je opravljena z notranjim algoritmom.

$$\begin{aligned}
 u_{in}(t) &= u_{out}(t - t_d) + i_2(t - t_d) \cdot Z_0 \\
 u_{out}(t) &= u_{in}(t - t_d) + i_1(t - t_d) \cdot Z_0
 \end{aligned}
 \quad (4.2)$$



Slika 4.50 Ekvivalentni model brezizgubne linije. S krmiljenima viroma U1 in U2 je izvedena relacija 4.2.

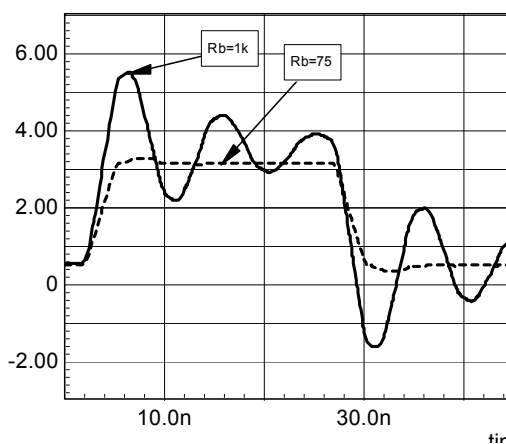
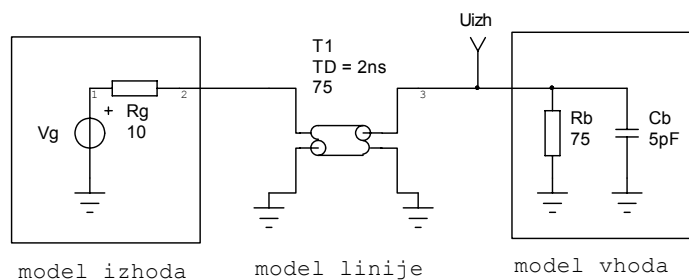
Lastnosti modela nastavljam s štirimi parametri. Obvezna sta dva parametra: prvi je karakteristična impedanca Z_0 , ki jo izračunamo glede na dimenzije in snovne lastnosti. Naslednji obvezni parameter⁵⁹ je zakasnitev (TD) signala med vhomom in izhodom.

Zgled:

Zakasnitev povezave med izhodom in vhomom je 2 ns. Karakteristična impedanca te povezave je 75 Ω . in čas vzpona je 3 ns. Glede na te podatke lahko uporabimo enačbo (4.6). Ker je pogoj v tej enačbi izpolnjen, ne moremo linije zamenjati s preprostim kratkim stikom. Izgubno upornost linije bomo zanemarili. Za modeliranje povezave bomo izbrali zelo preprost model brezizgubne linije: T1 2 0 3 0 $Z_0 = 75$ TD = 2 ns in opravili analizo integritete signala.

⁵⁸ Sinonim: idealna linija.

⁵⁹ Alternativa zakasnitvi je najvišja frekvenca.



Iz analize je razvidno, da se v primeru $R_b = 1\text{ k}\Omega$, zelo poruši oblika (integriteta) signala. Vzrok je v nepravilni impedanci na izhodu linije. Zadnji prenehaj lahko že povzroči preklon logičnega gradnika, saj je njegova napetost približno 2 V. Če je $R_b = Z_0 = 75\ \Omega$, je integriteta signala zelo dobro ohranjena.

4.6.4. Model valovoda z izgubami

Tudi model izgubne linije (lossy transmission line model) je vgrajen v simulator. Pravilnih napetosti in tokov simulator ne določi s pomočjo ustreznega električnega modela, ampak s konvolucijo, ki je hitrejša. Ime elementa se mora začeti s črko O. Ta element se sklicuje na stavek `.model`, kjer moramo navesti ime notranjega modela (LTRA) in vnesti še najmanj pet parametrov. Lastnosti linije so definirane s petimi parametri: R, L, C, G (vse enote se nanašajo na dolžinski meter) in dolžino linije (LEN). Z ostalimi 11 parametri lahko vplivamo na algoritem. Z njim lahko modeliramo tudi brezizgubno linijo ($R = 0$, $G = 0$) ali pa enakomerno distribuirano RC linijo ($L = 0$, $G = 0$). Ker je povrh tega še hitrejši, se ga zelo pogosto uporablja namesto obeh prejšnjih notranjih modelov.

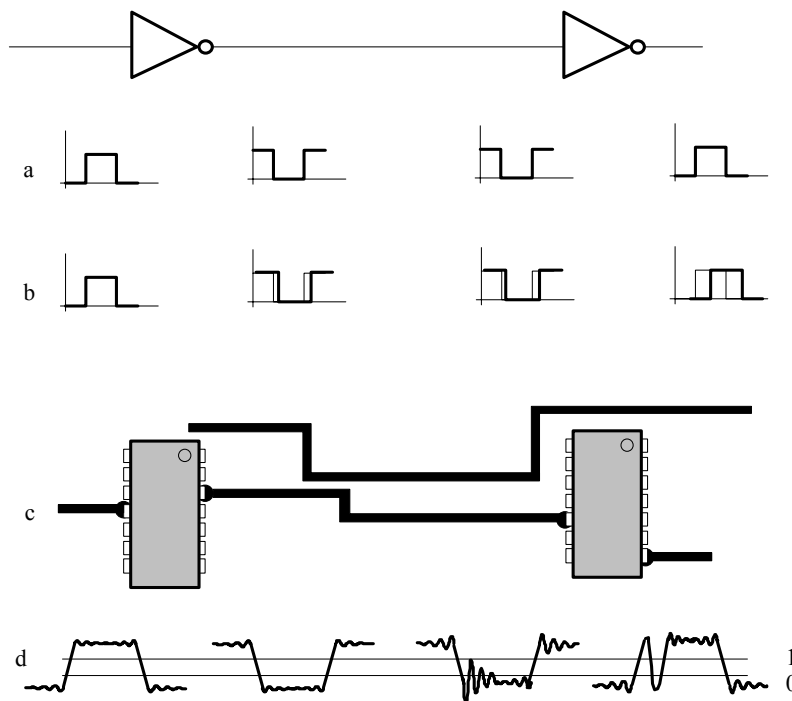
Zgled:

```
OBX 7 0 9 0 POVEZAVA
.MODEL POVEZAVA LTRA R=0.2 G=0 L=2n C=4p LEN=0.24
```

Povezava z oznako OBX ima naslednje lastnosti: 0,2 Ω /m, 2 nH/m, 4 pF/m in je dolga 24 cm.

4.7. Analiza integritete signala

Integriteta signala je sposobnost digitalnega signala, da korektno opravi svojo nalogo (pravočasnost in pravilni nivoji)⁶⁰. Izguba integritete se kaže kot spremenjena oblika signala in dodatna zakasnitev, zaradi katere vezje ne deluje pravilno.



Slika 4.51 a) idealne razmere, b) upoštevane so samo zakasnitve elementov – fizična implementacija povezav še ni upoštevana, c) fizična implementacija, d) meritve – integriteta signala je porušena

Dokler lahko zakasnitev in medsebojni vpliv povezav zanemarimo⁶¹, običajno nimamo problemov z integriteto. Zaradi vedno hitrejših vezij teh pojavov pogosto ne moremo več zanemariti. Odločilni faktorji, ki vplivajo na integriteto, so:

- zakasnitev povezave,
- impedanca vira in bremena,

⁶⁰ Več o tej problematiki: L. Green: *Understanding the Importance of Signal Integrity* IEEE Circuit and Devices, nov. 1999, str. 7–10.

⁶¹ Pogosto lahko zakasnitev samo enega segmenta zanemarimo. Kadar pa imamo veliko število elementov vezanih v serijo, moramo biti previdni, saj se zakasnitve posameznih segmentov seštevajo.

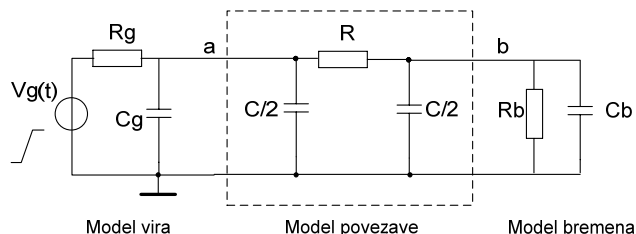
- presluh in
- čas vzpona in spusta logičnega signala oziroma najvišja frekvenca.

Analiza integritete je zelo pogosto vgrajena v načrtovalnik tiskanine. Iz podatkov o tiskanini program za vsako linijo izračuna njene parametre, nato pa izvede analizo. Če nimamo posebnega programa za analizo integritete, jo lahko opravimo tudi sami za vsako linijo posebej.

Vsaka analiza integritete (tista, ki jo izvajamo "peš", ali pa tista, ki je vgrajena v načrtovalnik) temelji na enem izmed naslednjih pristopov:

1. Izberemo preprost model vira in bremena ter **uporabimo analitično rešitev** oziroma posebni algoritem (slika 4.52).
2. Izberemo preprost model vira in bremena ter uporabimo **preprost program za analizo linearnih vezij** (slika 4.52).
3. Izberemo kompleksen model vira in bremena ter uporabimo program za analizo vezij (npr. **SPICE**). Glej sliko 4.53.
4. Vse povezave zamenjamo z ustreznimi modeli in ponovimo simulacijo vezja.

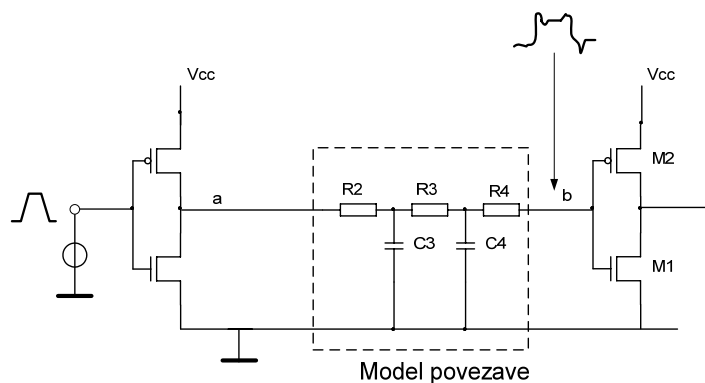
Najhitrejša, vendar tudi najmanj natančna je prva metoda, ki jo uporablja velika večina načrtovalnikov tiskanine oziroma topologije čipa. Ker imamo preprost model vira in bremena (slika 4.52) in ker je tudi model linije preprosto linearno vezje, je mogoče izpeljati splošen analitični izraz za odziv na impulz (zakasnitev, oblika, prenehaj itd.). Ker gre samo za izračun formule, lahko simulator za vsako linijo zelo hitro izvede analizo integritete. Ta pristop se uporablja v mnogih programih za načrtovanje in analizo tiskanin (npr. Altium Designer), kjer simulator analizira veliko število povezav. Slabost tega pristopa je, da smo pri modeliranju omejeni na niz izbranih modelov.



Slika 4.52 Preprost linearen model vira, povezave in bremena. Za tako preprost model lahko rešitev izrazimo na analitični način.

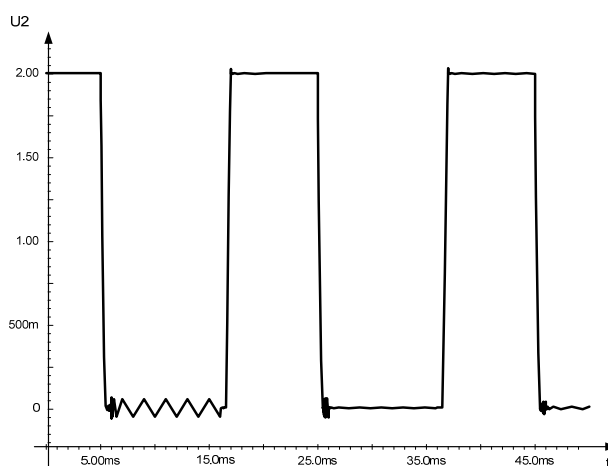
Če namesto analitične rešitve uporabimo preprost simulator, potem lahko uporabimo tudi svoje modele. Pri obeh metodah je bilo predpostavljeno, da nelinearnost vira oziroma bremena nima bistvenega vpliva na rezultat. Če to ne drži, potem vir in breme zamenjamo z nelinearnim modelom in uporabimo simulator SPICE (glej sliko 4.53). Če izračun parametrov povezave in analiza nista avtomatizirana, potem sta druga in tretja metoda uporabni le za manjše število povezav.

Najbolj natančne rezultate dobimo, če vse povezave zamenjamo z ustreznimi modeli in ponovno izvedemo simulacijo (več o tem je v poglavju Vpliv povezav na lastnosti vezja, ki je na strani 120).



Slika 4.53 Analiza integritete povezave med dvema invertorjema

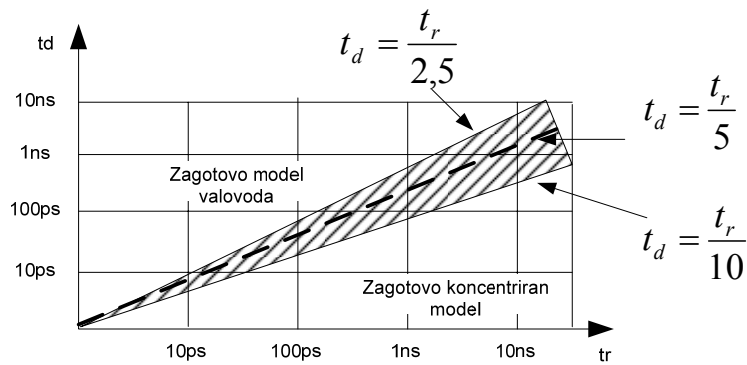
Pri analizi integritete moramo preveriti, ali gre v resnici za spremembo oblike signala, ali pa je to le posledica numeričnih algoritmov (glej sliko 4.54).



Slika 4.54 Oscilacije v območju med 5 ms in 15 ms niso posledica odbojev, ampak konvergenčnih problemov.

4.7.1. Poenostavljeni kriteriji za izbor modela

Ker lahko neustrezen model povzroči popolnoma napačne rezultate, je izbor modela odgovoren in tudi zahteven postopek, ki na koncu zahteva tudi ustrezno preverjanje pravilnosti izbora. Pri izboru je pomembno upoštevati tudi čas simulacije, saj nekateri modeli potrebujejo znatno več časa. Npr. model izgubnega valovoda je dokaj univerzalno uporaben, vendar se čas simulacije podaljša za 50-krat v primerjavi s preprostimi modeli. To pa je zelo pomembno, kadar analiziramo veliko število povezav.



Slika 4.55 Kriterij za izbor modela valovoda oziroma koncentriranega modela

Zelo pomembna je odločitev, ali je treba linijo modelirati kot valovod. Ker prehod ni oster, ampak zvezen, je meja približno v intervalu (slika 4.55) :

$$t_d = \frac{t_r}{k} ; 2,5 \leq k \leq 10 \quad (4.3)$$

Zgornji izraz lahko izrazimo na več načinov. Za model valovoda se je treba odločiti, če velja eden izmed naslednjih izrazov:

$$d > \frac{\lambda}{k} \quad \text{ali} \quad f > \frac{c}{kd} \quad (4.4)$$

$$f > \frac{1}{kt_d} \quad (4.5)$$

$$\begin{aligned} t_r &< \frac{kd}{c} \\ t_r &< kt_d \end{aligned} \quad (4.6)$$

Če upoštevamo, da je hitrost signala $v = \sqrt{LC}$, lahko zgornji pogoj izrazimo z dolžino d :

$$t_r < kd\sqrt{LC} \quad (4.7)$$

$$d > \frac{t_r}{k\sqrt{LC}} \quad (4.8)$$

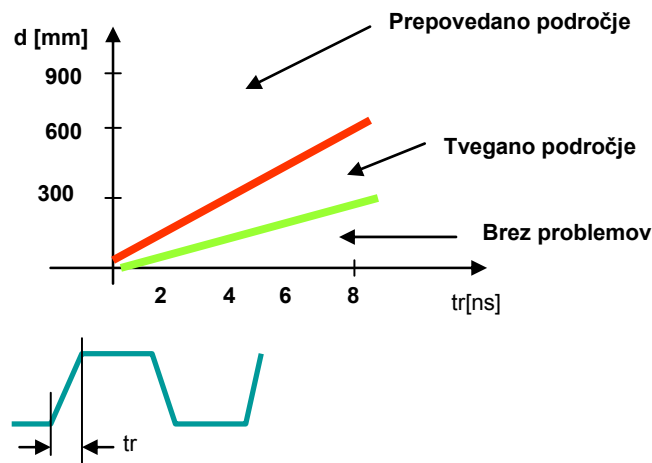
Pomen posameznih spremenljivk:

- f: najvišja frekvenca v vezju
- λ : valovna dolžina signala z najvišjo frekvenco
- t_d : zakasnitev povezave – izmerimo ali pa izračunamo. Vrednost je odvisna od geometrije in snovnih lastnosti.
- d: dolžina povezave
- c: hitrost elektromagnetnega valovanja (300 000 km/s)
- t_r : čas vzpona ali spusta digitalnega signala

- L: induktivnost povezave na enoto dolžine
 C: kapacitivnost povezave na enoto dolžine
 k: mejni faktor (glej slika 4.55 in enačbo 4.3)

Pogosto poznamo čas vzpona in bi potrebovali najvišjo frekvenco v spektru. Vrednost lahko ocenimo z enačbo:

$$f \approx \frac{0,35}{t_r} \quad (4.9)$$



Slika 4.56 Maksimalna varna dolžina povezave d v odvisnosti od časa vzpona⁶² t_r za tipično tiskanino

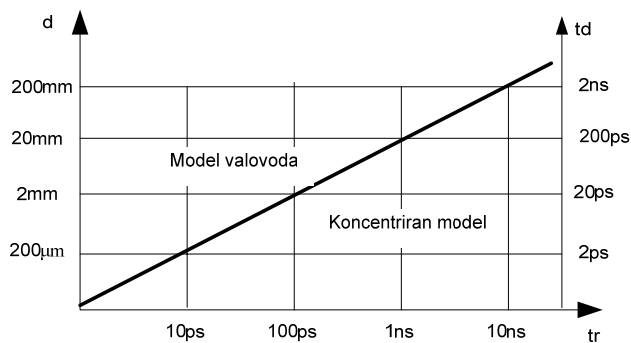
Zgled:

Nariši diagram, ki bo prikazal mejo med modelom navadne in prenosne linije v odvisnosti od časa vzpona in razdalje. Mejni faktor k naj bo 5.

- $R = 0,2 \Omega/\mu\text{m}$
 $L = 0,5 \text{ pH}/\mu\text{m}$
 $C = 0,2 \text{ fF}/\mu\text{m}$

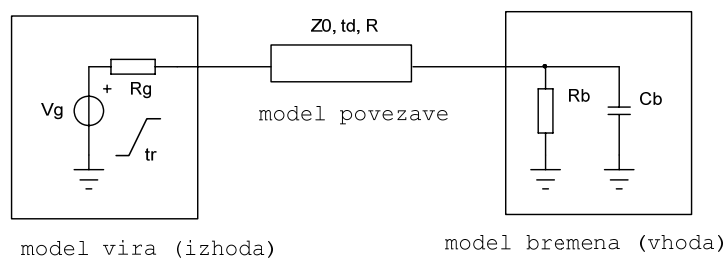
Mejno področje (formula 4.8):
$$d = \frac{t_r}{5 \sqrt{\frac{0,5 \cdot 10^{-12}}{10^{-6}} \cdot \frac{0,2 \cdot 10^{-15}}{10^{-6}}}}} = 0,2 \cdot 10^8 \cdot t_r$$

⁶² John Berrie: *The defensive design of printed-circuit boards*, IEEE Spectrum, Volume 36, Issue 9 (September 1999), str. 76–81, 1999, ISSN:0018-9235.



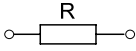
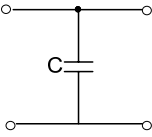
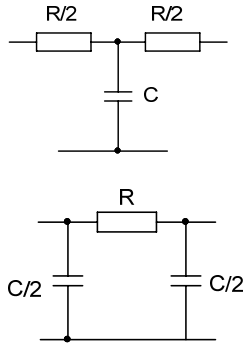
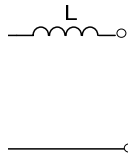
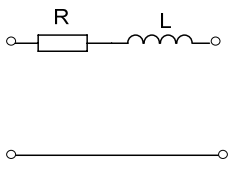
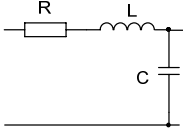
Če je npr. čas vzpona 10 ns, lahko linijo, ki je krajša od 200 mm, modeliramo s koncentriranim modelom.

Pri izboru modela povezave si pomagamo s tabelo 4.3, ki temelji na preprostem modelu vira, povezave ter bremena (slika 4.57).



Slika 4.57 Preprost linearen model, s katerim analiziramo lastnosti povezave oziroma integriteto signala posamezne povezave

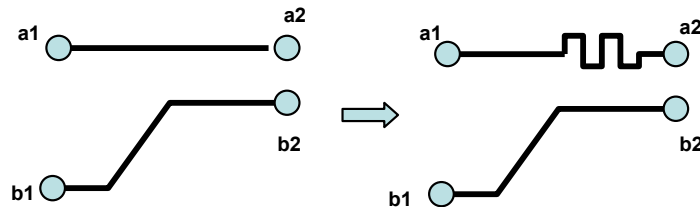
Tabela 4.3 Poenostavljeni kriteriji za izbor modela⁶³ povezave. Mejni faktor $k = 5$.

Model	Pogoji	Opomba
	$t_r > 5t_d$ $R > 0,1Rg$	Glede na razdaljo imamo nizke frekvence. Največ izgub je na liniji. Zakasnitve nas ne zanimajo.
	$t_r > 5t_d$ $(R + Rg)C > 0,1t_r$	Glede na razdaljo imamo nizke frekvence. Kapacitivnost povezave C najbolj vpliva na zakasnitev.
	$t_r > 5t_d$ $(R + Rg)C > 0,1t_r$ $R > 0,1Rg$	Glede na razdaljo imamo nizke frekvence. Kapacitivnost povezave C najbolj vpliva na zakasnitev. Največ izgub je na liniji.
	$t_r > 5t_d$ $\frac{L}{(R + Rg)} > 0,1t_r$	Glede na razdaljo imamo nizke frekvence. Induktivnost povezave L najbolj vpliva na zakasnitev.
	$t_r > 5t_d$ $R > 0,1Rg$ $\frac{L}{(R + Rg)} > 0,1t_r$	Glede na razdaljo imamo nizke frekvence. Induktivnost povezave L najbolj vpliva na zakasnitev. Največ izgub je na liniji.
	$t_r > 5t_d$ $R > 0,1Rg$	Glede na razdaljo imamo nizke frekvence. Induktivnost povezave L tudi vpliva na zakasnitev. Največ izgub je na liniji.
T (model brezizgubnega valovoda)	$t_r < 5t_d$ $R \approx 0$	Glede na razdaljo imamo visoke frekvence. Vpliv izgub je zanemarljiv.
O (model izgubnega valovoda)	$t_r < 5t_d$ $R > 0$	Glede na razdaljo imamo visoke frekvence. Vpliv izgub ni zanemarljiv.

⁶³ Priporočila so iz Star-Hspice Manual - Release 2001.2 - June 2001, http://www.ece.uci.edu/docs/hspice/hspice_2001_2-269.html (prebrano 17. 2. 2010).

4.7.2. Problem hazarda

Eden izmed vzrokov za pojav hazarda pri logičnih vezjih je različna zakasnitev signalov na vhodih logičnega gradnika. Ta problem lahko s simulatorjem identificiramo in nato rešimo z uglaševanjem oziroma ustreznim podaljševanjem linij⁶⁴. Uglaševanje je vgrajeno v večino sodobnih načrtovalnikov tiskanin.



Slika 4.58 Problem hazarda lahko rešujemo tudi z uglaševanjem linij

4.8. Modeliranje digitalnih vhodov in izhodov

Najbolj znan zunanji model digitalnih vhodov in izhodov je **IBIS model** (I/O Buffer Information Specification). To je standardiziran format opisovanja obnašanja izhodov in vhodov (glej zgled na sliki 4.59 in sliki 4.60). Mnogo sodobnih EDA orodij⁶⁵ prepozna IBIS format. IBIS parametre običajno dobimo z meritvijo ali pa z analogno simulacijo (SPICE). Ker IBIS parametri temeljijo na električnem modelu, jih je enostavno pretvoriti v model, ki ustreza sintaksi SPICE simulatorja. Za modeliranje se uporabljata tudi jezika VHDL-AMS in Verilog-AMS⁶⁶, ki sta namenjena opisu strukture in obnašanja analognih in digitalno-analognih vezij.

```

| *****
|
| [Component]      Am188EMLV_TQFP_100
| [Manufacturer]   AMD
| [Package]
| | variable      typ          min          max
| R_pkg           0.12         NA           NA
| L_pkg           10.33nH      NA           NA
| C_pkg           0.82pF       NA           NA
|
| | Am188EMLV TQFP Pin Numbers
| [Pin]  signal_name  model_name  R_pin  L_pin  C_pin
| 1      AD0         iobpad     0.12  10.33nH  0.82pF
|        AO8         iobpad     0.12  10.33nH  0.82pF
|        :           :

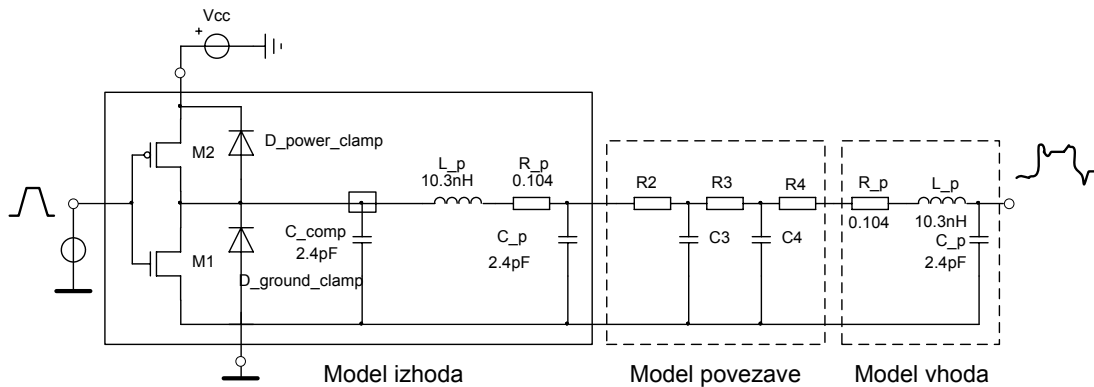
```

Slika 4.59 Zgled opisa v IBIS formatu (prikazan je samo del)

⁶⁴ Ang. line tuning.

⁶⁵ Npr. Altium Designer.

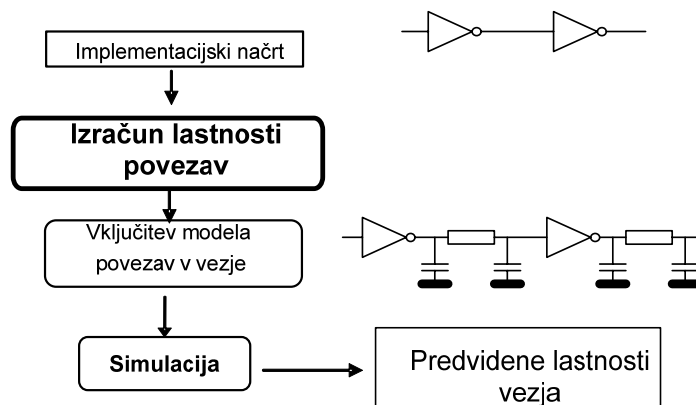
⁶⁶ <http://www.eda-stds.org/verilog-ams/htmlpages/lit.html>, prebrano 16. 3. 2010.



Slika 4.60 Kompletan model, s katerim lahko analiziramo integriteto signala med točkama a in b (glej sliko 4.48). Model vhoda in izhoda temelji na IBIS modelu.

4.9. Vpliv povezav na lastnosti vezja

Pri analizi integritete smo analizirali samo **posamezne** povezave, ne pa celotnega vezja. Le če vezje ni preveč kompleksno in če uporabljamo enostavne modele povezav, lahko model **vseh** povezav vključimo v celotno vezje in **ponovno izvedemo vse analize**. Na ta način lahko ugotovimo, kako bodo vplivale vse povezave na lastnosti implementiranega vezja. Ta analiza je v mnogih primerih izvedljiva le, če je popolnoma avtomatizirana. Zelo pogosto se uporablja pri načrtovanju integriranih vezij. Pri velikem številu povezav se izredno poveča kompleksnost vezja, kar tudi zelo poveča čas simulacije.



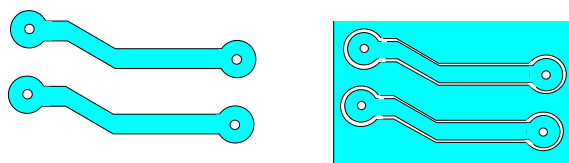
Slika 4.61 Ali so povezave bistveno degradirale lastnosti?

4.10. Izdelava prototipa



Slika 4.62 Koordinatni rezkalnik tiskanin

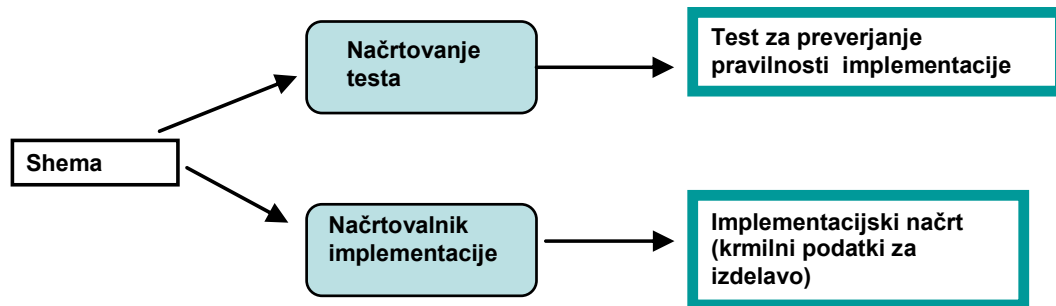
Preden se odločimo za masovno proizvodnjo, običajno izdelamo prototip, kjer še zadnjič preverimo pravilnost načrtovanja. Prototip lahko tudi izdelamo s pomočjo posebnih koordinatnih rezkalnikov tiskanin, ki z odstranitvijo bakra na robovih dosežejo povezljivost.



Slika 4.63 Običajna tiskanina (levo) in tiskanina, ki jo izdelata koordinatni rezkalnik tiskanin (desno)

4.11. Priprava za proizvodnjo

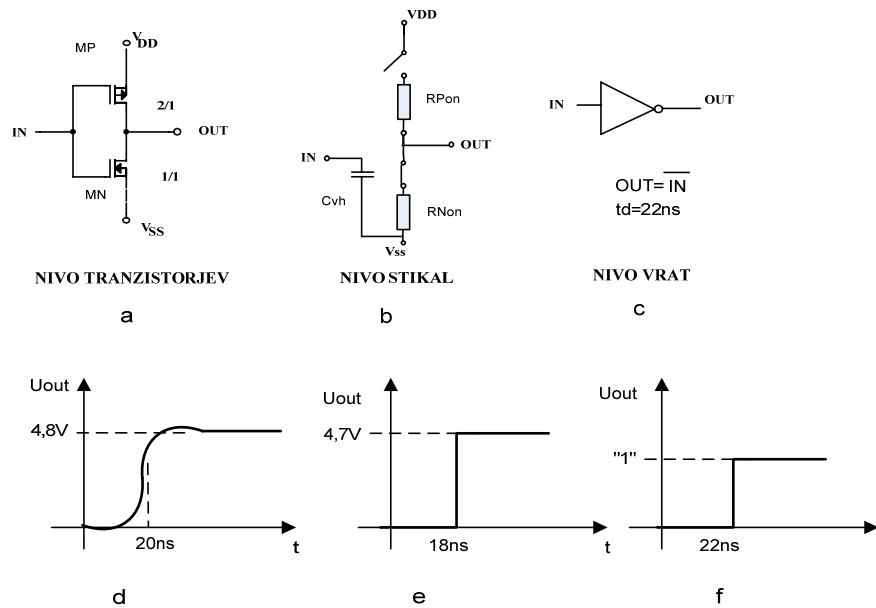
Večino strojev (npr. fotorisalnik), s katerimi izdelujemo elektronska vezja oziroma tiskanine, krmilimo s podatki v Gerber formatu. Za urejanje tega zapisa obstaja tudi poseben urejevalnik Gerber zapisa. Za proizvodnjo potrebujemo predvsem implementacijski načrt in teste, s katerimi bomo preverili pravilnost sestavljanja.



Slika 4.64 Za proizvodnjo potrebujemo implementacijski načrt in teste za preverjanje pravilnosti implementacije

5. SIMULACIJA DIGITALNIH IN ANALOGNO-DIGITALNIH VEZIJ

Elektronska vezja delimo na analogna, digitalna (logična) ali kombinirana (analogno/digitalna). Najprej bomo obravnavali simulacijo logičnih, pozneje pa kombiniranih vezij.



Slika 5.1 Rezultati simulacije invertorja: model s tranzistorji (a), stikalni model (b) in simulacija na nivoju vrat (c)

S simulatorji logičnih vezij preverjamo funkcionalno (logično) in časovno pravilnost logičnega vezja. Če se vhodni logični signali pravilno preslikujejo na izhode, potem je vezje funkcionalno pravilno. Vezje je časovno pravilno, če se te preslikave opravijo tudi ob pravem času. Za preverjanje pravilnosti načrtovanja logičnih vezij ima načrtovalec na razpolago naslednje skupine simulatorjev, ki so specializirani za posamezne nivoje, ki so prikazani na zgladu (slika 5.6):

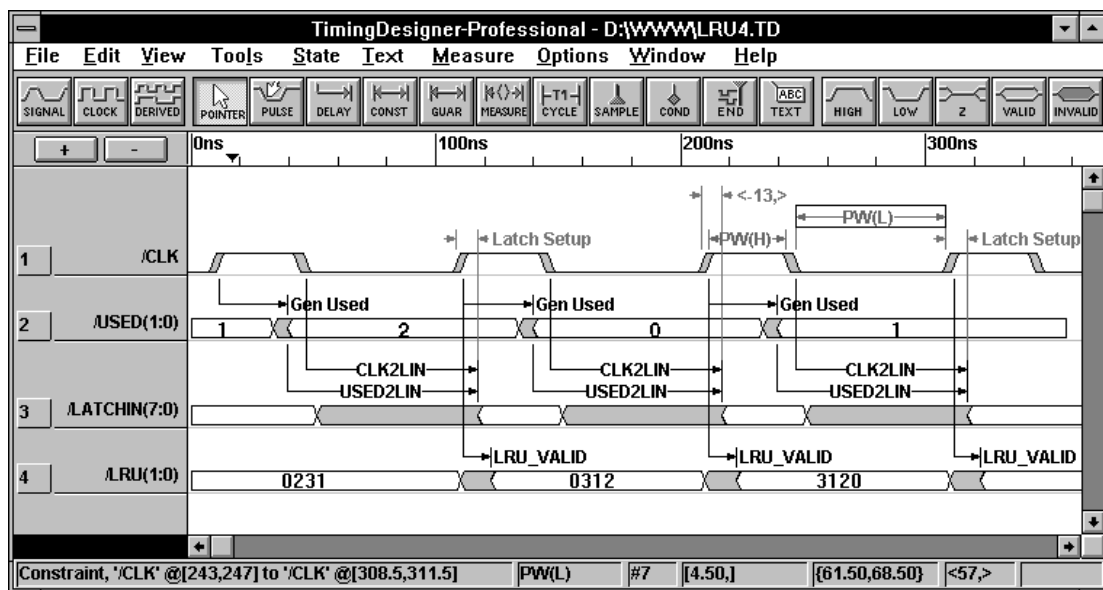
- Simulator vezij (**Transistor-level simulator, circuit-level simulator, device-level simulators**) – SPICE, ISPICE, PSPICE, HSPICE, Microcap.
- Simulator na nivoju stikal (**Switch-level simulator**) – Silos, Tegas, IRSIM, ESIM, RSIM, RNL, Cosmos, and Mossim.
- Simulator na nivoju vrat (**Gate-level simulator**) – LogicWork.
- Statični analizator zakasnitev (**Static timing analyser**) – TimingDesigner.
- Vedenjski simulator (**Behavioral simulator**), Funkcijski simulator⁶⁷ (**Functional simulator**) – Verilog, Matlab.

⁶⁷ Večinoma gre za sinonima: vedenjski = funkcijski.

Najbolj natančne rezultate daje **simulacija na nivoju tranzistorjev** (npr. simulator SPICE), saj uporablja zelo natančne modele tranzistorjev, ki tvorijo logično vezje. Zato potrebuje tudi največjo računalniško moč, oziroma praktična uporaba je omejena na manjše število elementov. Slika 5.1 a prikazuje strukturo preprostega invertorja, 5.1 d pa rezultat simulacije.

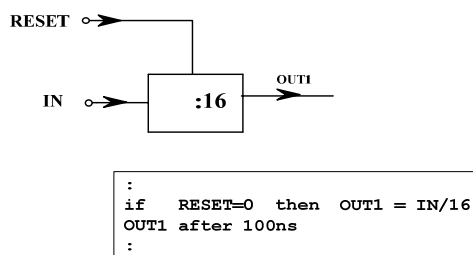
Če se odrečemo tako veliki natančnosti in se zadovoljimo samo s približno zakasnitvijo in približno vrednostjo logičnih nivojev, lahko uporabimo **simulator na nivoju stikal**. Vsak tranzistor modeliramo kot stikalo, katerega zakasnitev je odvisna od izhodnih upornosti in vhodnih kapacitivnosti oziroma obremenitev na izhodu (glej sliko 5.1 b). Preverjamo lahko funkcionalno pravilnost, zakasnitve in nivo signala (slika 5.1 e). Na zakasnitev signala vplivajo čas vzpona vhodnih signalov, zakasnitev posameznih tranzistorjev in število bremen. Zaradi te poenostavitve se je hitrost simulacije zelo povečala.

Za zelo obsežna vezja lahko uporabimo **simulator na nivoju vrat**. Funkcionalnost osnovnih gradnikov je vgrajena v simulator v obliki notranjih modelov. Vpliv bremen (vhodnih kapacitivnosti) na zakasnitev ni upoštevan, tudi odvisnost zakasnitve od vhodnega signala ne. Vsak izhod ima konstantno zakasnitev (slika 5.1 c) in predpisano logično funkcijo, ki preslika vhodne logične vrednosti na izhod. Vsi signali so označeni z 0 ali 1 (slika 5.1 f). Natančnost zakasnitev se je zaradi tega poslabšala, vendar se je zelo povečala hitrost simulacije.



Slika 5.2 Prikaz statične časovne analize

Za ocenitev zakasnitev lahko uporabimo tudi **statični analizator zakasnitev**. Analizator v bistvu ni simulator, saj vezja ne simulira, ampak samo izračunava zakasnitve raznih poti (glej sliko 5.2). V zelo kompleksnem vezju lahko poišče kritično pot. Če signal potuje po tej poti, ima največjo zakasnitev. Uporabljamo ga tudi za specificiranje vhodnih in izhodnih signalov.

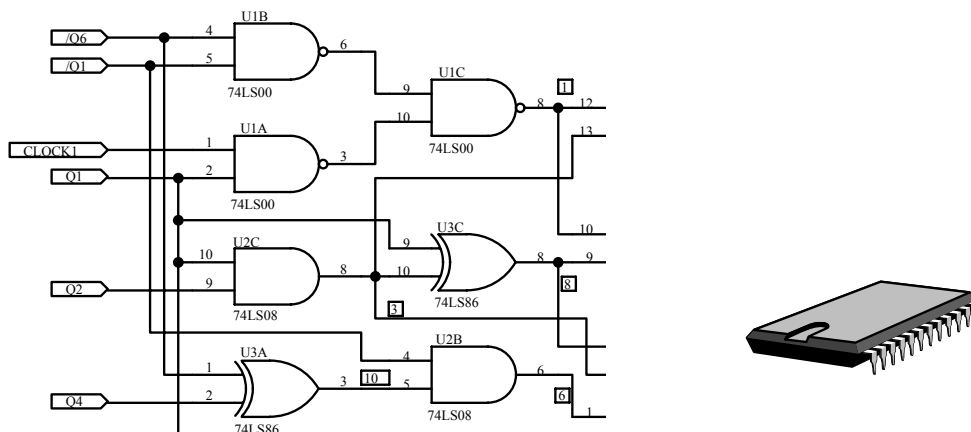


Slika 5.3 Na nivoju podsistemov ali sistemov uporabljamo vedenjski simulator

Z **vedenjskim simulatorjem** preverjamo pravilnost obnašanja vezja na konceptualnem nivoju oziroma na nivoju podsistemov (slika 5.3). Gradniki, s katerimi opisujemo obnašanje vezja, imajo lahko zelo veliko funkcionalnost, ki je opisana s programskim jezikom. Vsakemu podsistemu lahko dodamo tudi konstantno zakasnitev. S tem simulatorjem preverjamo predvsem funkcionalno pravilnost delovanja. Če so dodane tudi zakasnitve podsistemov, lahko dobimo zelo grobe ocene zakasnitev.

Z nižanjem nivoja raste natančnost simulacije in hkrati se daljša čas simuliranja.

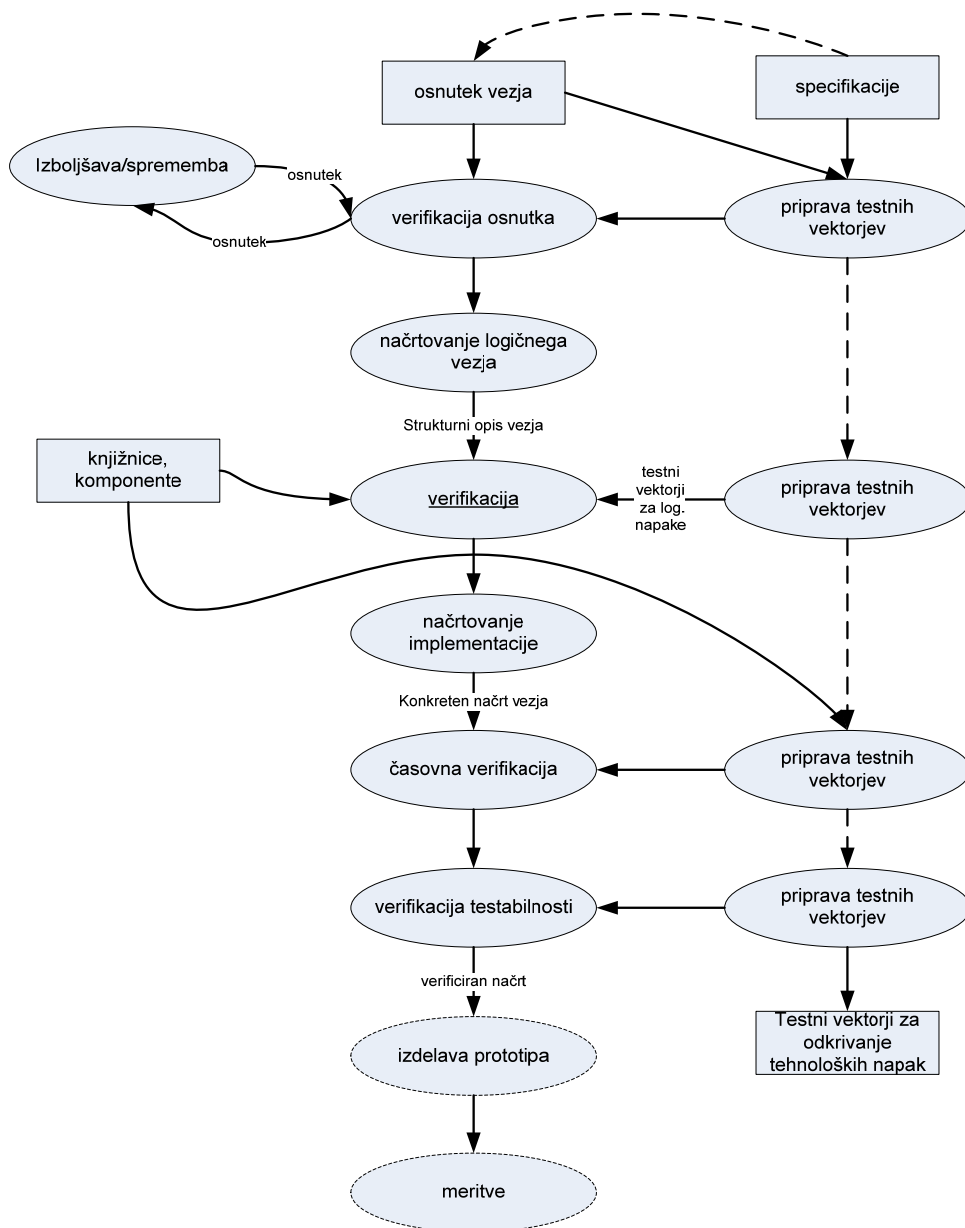
5.1. Značilnosti logičnih (digitalnih) vezij



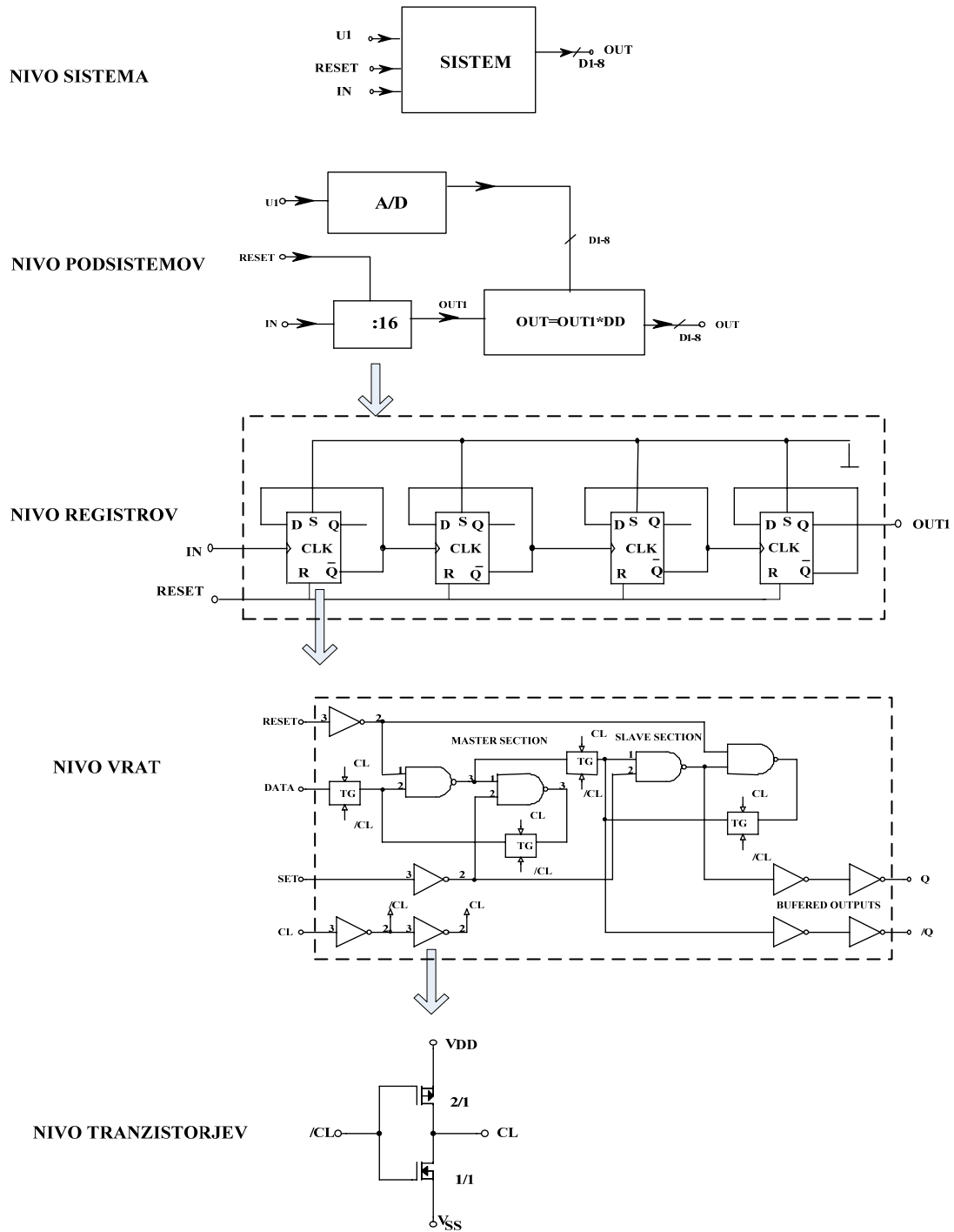
Slika 5.4 Zgled digitalnega vezja

- Gradniki so zelo dobro definirani in standardizirani.
- Sinteza in analiza sta dobro razviti.
- Učinkovite testirne metode.
- Velika kompleksnost.
- Veliko število vhodno/izhodnih priključkov.
- Velike hitrosti.

Postopek načrtovanja kompleksnih (VLSI) logičnih vezij



Slika 5.5 Postopek načrtovanja kompleksnih (VLSI) logičnih vezij



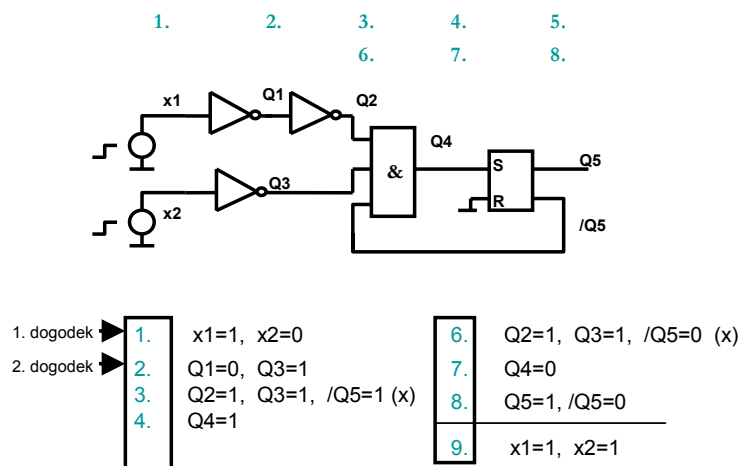
Slika 5.6 Razvoj VLSI sistema poteka od zgoraj navzdol

5.2. Delovanje logičnega simulatorja

Vsi logični simulatorji so **dogodkovno krmiljeni simulatorji** (event-driven simulators). Ker je njihovo delovanje preprostejše kot delovanje analognih simulatorjev, so sposobni simulirati zelo kompleksna vezja logična vezja.

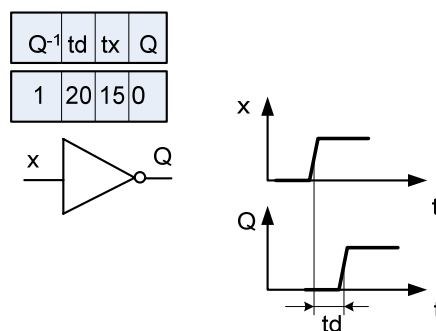
Poenostavljen algoritem delovanja: Simulator najprej določi začetna stanja. Dogodek je vsaka sprememba stanja vozlišča, ki nastane zaradi sprememb na vseh in zaradi sprememb stanja nekega elementa. Če ni nobenega novega dogodka, vsa stanja ohranijo svoje vrednosti. Ob pojavu dogodka simulator ovrednoti vrednosti izhodov. Prvi dogodki se ustvarijo, ko se na vseh spremenijo vrednosti signalov. Ker se zaradi tega spremenijo izhodi nekaterih gradnikov, se ustvarijo novi dogodki. Simulacija se ustavi, ko na vseh ni več sprememb. Zaradi povratnih vezav lahko nastanejo oscilacije oziroma se dogodki nočejo umiriti.

Zgled:



Slika 5.7 Zgled za prikaz delovanja logičnega simulatorja

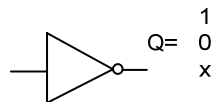
Simulacija zakasnitev



Slika 5.8 Simulacija zakasnitev (t_d je zakasnitev, t_x števec, ki šteje čas, Q^{-1} predhodno stanje). Na izhodu Q je vrednost logične 0 tako dolgo, dokler števec t_x ne doseže vrednosti 20.

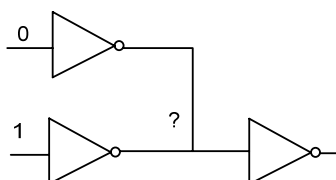
Modeliranje signala oziroma izhoda logičnega gradnika

a. vrednost signala



Slika 5.9 Na izhodu je lahko: logična 1, 0 ali pa nedefinirana vrednost, ki jo običajno označujemo z x

b. moč signala (signal strenght)



Slika 5.10 Ker lahko izhode nekaterih logičnih gradnikov povežemo, mora simulator upoštevati tudi tak primer

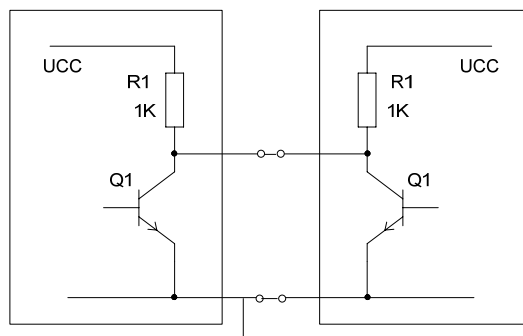
Izhode nekaterih logičnih gradnikov lahko povežemo in s tem ustvarimo dodatno logično operacijo (glej sliko 5.10). Da bi lahko simulator pravilno rešil take primere, je treba vpeljati dodatni atribut – **moč signala**. Vsak izhod gradnika tvori signal ustrezne moči, ki je predstavljena s celim številom. Moč signala je povezana z izhodno upornostjo izhodne stopnje logičnega gradnika: nižja je izhodna upornost, večja je moč signala. Najpogosteje se v simulatorjih uporablja od 4 do 9 različnih moči signalov. Moč signala se definira za vse (tri) logične nivoje, kar pomeni, da ima lahko na primer nek izhod šibko logično enico in močno logično ničlo.

Pri povezavi dveh izhodov zato vedno prevlada močnejši signal. Če sta povezana dva izhoda z enako močjo, dobimo nedefinirano vrednost (npr. 4 in 4) ali pa vrednost, ki ustreza AND operaciji (npr. 2 in 2). Vsak simulator ima vgrajeno pravilnostno tabelo, kjer so definirane vse mogoče kombinacije povezanih izhodov in pripadajoč rezultat.

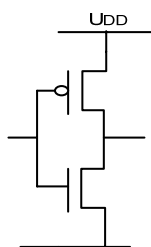
Tabela 5.1 Tipične moči logičnih signalov

	Vrsta	Zgled
4	Forsiran	Napajalna napetost
3	Močan rezistivni	CMOS izhod
2	Šibek rezistivni	Odprt kolektor
1	Visoka impedanca	Izhodna tranzistorja ne prevajata

Tri vrednosti signalov in štiri različne moči tvorijo 12 kombinacij.



Slika 5.11 Izhod narisane logičnega gradnika ustvarja močno logično ničlo, ker prevaja Q1 in ima zato nizko upornost. Ko tranzistor ne prevaja, ima zelo visoko izhodno upornost. To se kaže kot šibka logična enica, saj je izhoda povezan z uporom 1 k na UCC. Če izhoda povežemo, rezultat ustreza AND operaciji.

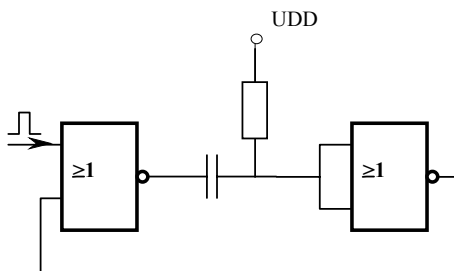


Slika 5.12 Izhod CMOS invertorja ima močno logično enico in ničlo

5.3. Simulacija analogno-digitalnih vezij

Elektronska vezja lahko razdelimo v štiri skupine:

1. popolnoma analogno vezje,
2. analogno-digitalno vezje – prevladuje analogni del. Digitalni del vsebuje samo nekaj elementov,
3. analogno-digitalno vezje – prevladuje digitalni del oziroma oba dela sta približno enako kompleksna,
4. popolnoma digitalno vezje.



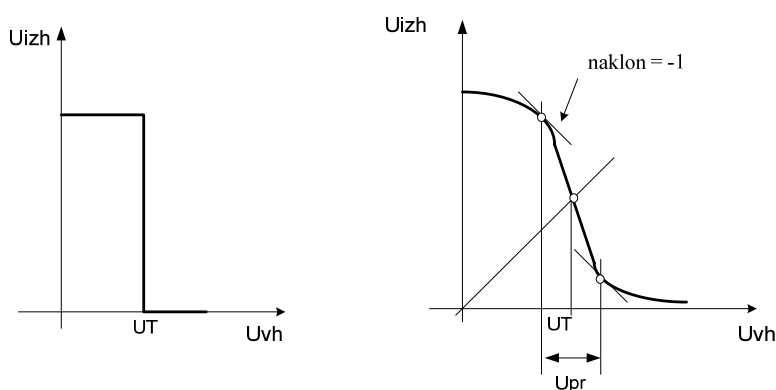
Slika 5.13 Zgled preprostega analogno-digitalnega vezja (monostabilni multivibrator)

Kadar prevladuje analogni del in je v vezju le nekaj logičnih gradnikov, lahko uporabimo analogni simulator, logične gradnike pa ustrezno modeliramo. Ker je pri večjem številu logičnih gradnikov simulacija počasna, je v takem primeru primernejši kombiniran simulator.

Pravilno modeliranje logičnih gradnikov je za simulacijo analogno-digitalnih vezij ključnega pomena. Vedno je treba izbrati kompromis med hitrostjo in natančnostjo simulacije. Model logičnega gradnika je lahko notranji ali zunanji.

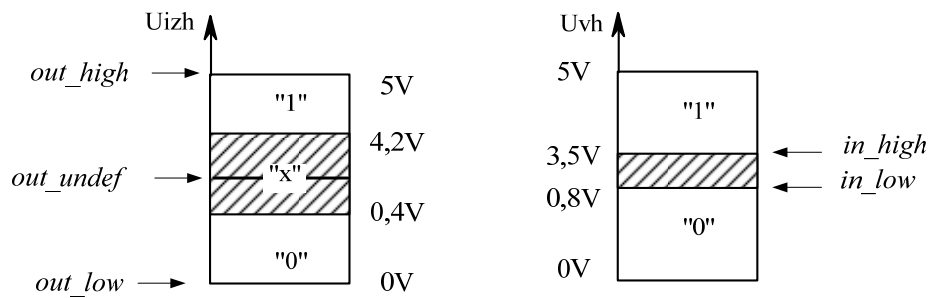
5.4. Modeliranje logičnih gradnikov v analognem simulatorju

Lastnosti logičnega gradnika so opisane s prenosno, vhodno in izhodno karakteristiko ter zakasnitvami. Preklopna karakteristika je enosmerna prenosna karakteristika logičnega vezja. Preklopno področje U_{pr} predstavlja interval med točkama z naklonom -45° in se približno ujema s prepovedanim področjem. Preklopna napetost U_T je točka, v kateri je vhodna napetost enaka izhodni. Preklopna napetost U_T leži vedno znotraj prepovedanega področja.



Slika 5.14 Idealna in tipična preklopna (prenosna) karakteristika vrat

Ker imajo digitalna vezja veliko število gradnikov, je zelo pomembno, da so modeli enostavni oziroma sestavljeni iz malega števila elementov. Zato pogosto ne modeliramo prenosnih karakteristik, ampak samo **preklopno napetost, logične nivoje in zakasnitve**. Zakasnitve v večini primerov modeliramo kar z ustreznimi RC členi.

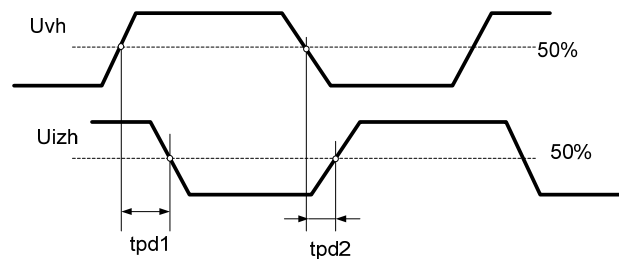


Slika 5.15 Logični nivoji (konkretne vrednosti se nanašajo na tipično CMOS družino)

Logični nivoji

Za simulacijo so najpomembnejši naslednje vrednosti (glej sliko 5.15):

- in_low : maksimalna napetost logične ničle na vhodu
- in_high : minimalna napetost logične enice
- out_low : napetost logične ničle na izhodu
- out_high : napetost logične enice na izhodu
- out_undef : napetost, ki se naj pojavi na izhodu, kadar je vhodna napetost znotraj prepovedanega območja ($in_low < U_{vh} < in_high$)

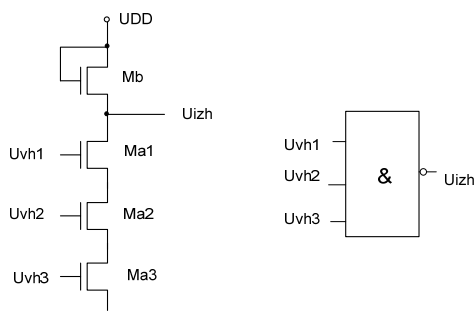
Slika 5.16 Zakasnitev invertorja. Namesto oznak $tpd1$ in $tpd2$ se uporabljata $rise_delay$ in $fall_delay$.

V nadaljevanju bomo na zgledu trovhodnih NAND (logična enica bo 3 V in logična ničla 0 V) vrat opisali pet variant modelov, ki jih uvrščamo v eno izmed treh skupin:

- **implementacijski model,**
- **analogni makromodel in**
- **notranji (kodni) model.**

5.4.1. Implementacijski model

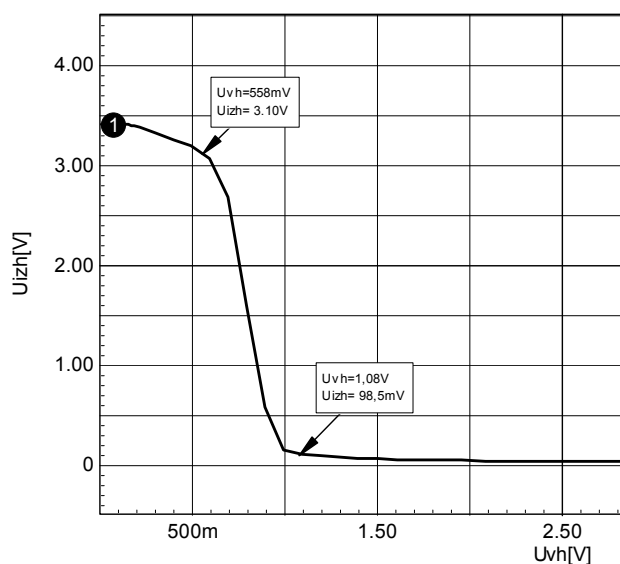
Če poznamo notranjo strukturo gradnika (dimenzije tranzistorjev in njihovi parametri), lahko tvorimo implementacijski model, ki najbolje predvidi lastnosti (preklopna karakteristika, disipacija, zakasnitve itd.) logičnega gradnika. Pri kompleksnih vezjih se simulacija zaradi tega zelo upočasni.



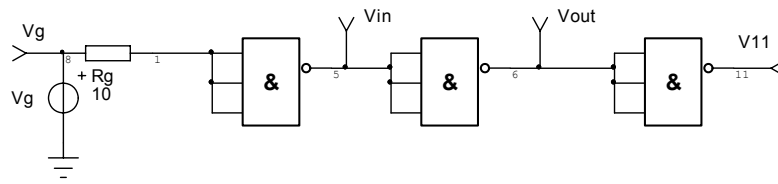
Slika 5.17 Trovhodna NAND vrata napravljena z NMOS tranzistorji

Iz prenosne karakteristike je razvidno, da je napetost logične enice med 3,5 V in 3,1 V in logične ničle med 98 mV in 0 V. Kot tipično vrednost za enico bomo izbrali 3V in za ničlo 0 V. Preklopna napetost (točka, kjer je $U_{vh} = U_{zh}$), je približno 1 V.

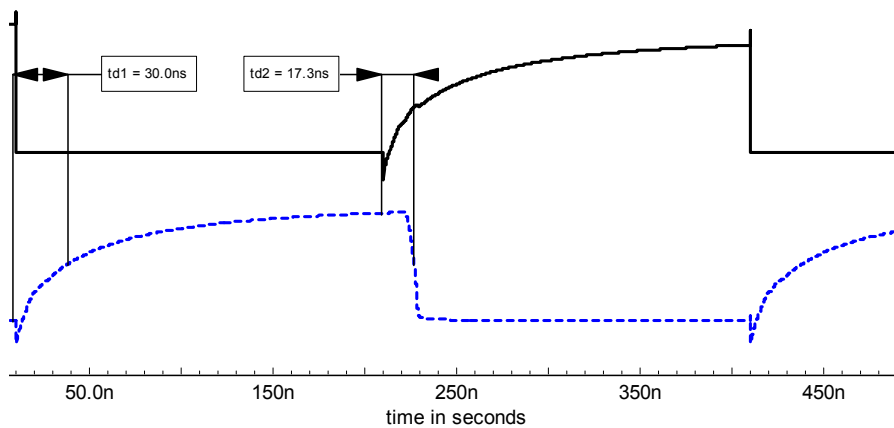
Meritev zakasnitev se izvede tako, da vhod krmilimo z enako komponento. To velja tudi za breme (glej sliko 5.19). Zakasnitev pri dvigovanju signala (rise_delay) je približno 30 ns in pri spustu (fall_delay) je 17 ns. Če se uporablja samo kot enoten parameter, je to povprečje teh dveh zakasnitev (24 ns).



Slika 5.18 Prenosna karakteristika (vsi vhodi so povezani). Označeni sta vhodni napetosti, pri katerih je naklon enak -1.



Slika 5.19 Meritev zakasnitve se izvede tako, da vhod krmilimo z enako komponento. To velja tudi za breme.



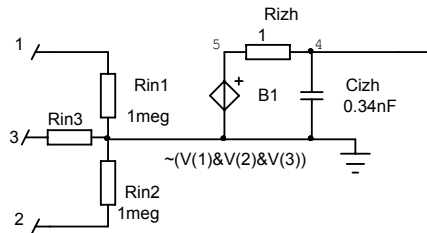
Slika 5.20 Zakasnitve za komponento, ki je na sredini. Zakasnitve so merjene med točkama, ki znašata 50 % napetosti logične enice.

5.4.2. Analogni makromodel

Pri teh modelih poskušamo z najbolj preprostimi analognimi gradniki modelirati prenosno funkcijo logičnega gradnika in ustrezno zakasnitev. Osrednji del modela je pogosto polinomski vir, ki ima dodane upore za modeliranje vhodne in izhodne upornosti. S kondenzatorji modeliramo ustrezne zakasnitve. Zaradi preprostosti ti modeli ne predstavljajo velikega bremena za računalnik. Njihova prednost je tudi ta, da za njihovo uporabo ne potrebujemo posebnih simulatorjev (mixed mode simulator).

Zgled 1:

Nekateri simulatorji imajo definirane posebne logične operatorje (npr. AND, OR in negacija), s katerimi lahko tvorimo logično funkcijo (glej sliko 5.21). Napetosti logičnih nivojev običajno nastavljamo s parametri (npr. v stavku OPTIONS). Zakasnitve in vhodne upornosti logičnega gradnika modeliramo z ustreznimi RC členi. Trovhodna NAND vrata iz prejšnjega zglada bomo modelirali z analognim makromodelom.



```
.OPTIONS Lone=3 Lthresh=1 Lzero=0
:

.SUBCKT NAND3 1 2 3 4
*           A B C Out
* model: makro 1
B1 5 0 V= ~(V(1)&V(2)&V(3))
Rin1 1 0 1meg
Rin2 2 0 1meg
Rin3 3 0 1meg
Rizh 5 4 1
Cizh 4 0.34NF
.ENDS
```

Slika 5.21 Model trovhodnih NAND vrat, ki je sestavljen iz vedenjskega in strukturnega modela⁶⁸. NAND operacija je opisana z logičnim izrazom $V = \sim(V(1)\&V(2)\&V(3))$. Z uporabo Rizh in kondenzatorjem Cizh je modelirana zakasnitev. Njegova karakteristika je zelo podobna idealni. Vrednost logične enice je 3 in ničle 0. Preklopna napetost je pri 1 V, zakasnitev⁶⁹ je 24 ns.

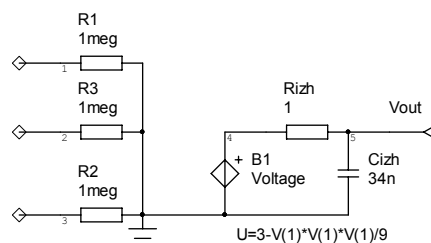
Zgled 2: Modeliranje trovhodnih NAND vrat. Logična enica naj bo 3 V.

Ker imamo 3 vhode, bomo izbrali napetostno krmiljen vir⁷⁰ vrste B, ki je definiran z enačbo:

$$U_{izh} = f(U_{vh_1}, U_{vh_2}, U_{vh_3}) = p_0 - p_4 U_{vh_1} \cdot U_{vh_2} \cdot U_{vh_3}$$

Če izberemo $p_0 = 3$, $p_4 = 1/9$, dobimo prenosno karakteristiko:

$U_{izh} = 3 - 1/9 \cdot U_{vh_1} \cdot U_{vh_2} \cdot U_{vh_3}$. Če ima logična enica natanko 3 V in logična ničla ustreza natanko 0 V, potem prejšnji izraz predstavlja NAND funkcijo. To je eden izmed najstarejših modelov logičnih gradnikov, ki se danes redko uporablja.



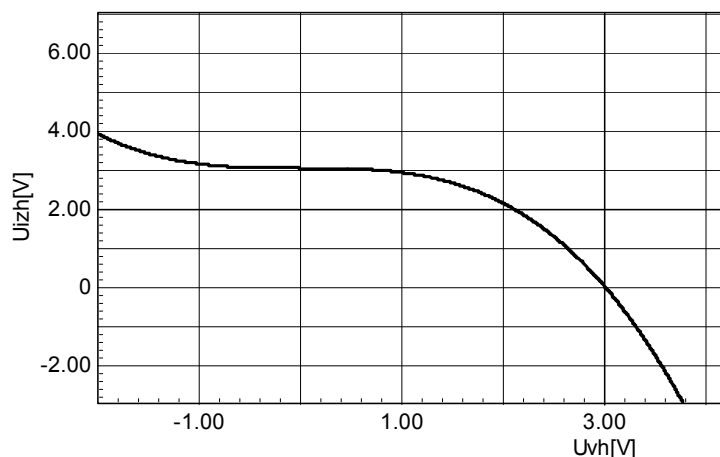
Slika 5.22 Model, ki ga bomo poimenovali makro 2

⁶⁸ Npr. Intusoft SPICE.

⁶⁹ $t_d = \ln(2) \cdot C_{izh} \cdot R_{izh}$

⁷⁰ Namesto B vira bi lahko izbrali tridimenzionalni polinomski vir.

Preklopno območje pri tem modelu ni ostro določeno, saj zavzema celotno območje vhodnih napetosti, to je od 0 V do 5 V (glej sliko 5.23). Naslednja slabost je velika občutljivost na višino vhodne napetosti. Če je vhodna napetost različna od 5 V, se pri zaporedni vezavi več vrat vrednost logične enice manjša.



Slika 5.23 Prenosna karakteristika vezja iz slike 5.22 je $U_{izh} = 3 - U_{vh}^3$. Vhodi so povezani ($U_{vh} = U_{vh1} = U_{vh3}$).

Zgled 3: Polinomski model NAND vrat s strmo preklopno karakteristiko

Slabost prejšnjega modela lahko odpravimo, če s polinomsko regresijo določimo koeficiente polinoma, ki se čim boljše prilaga idealni ali pa resnični preklopni karakteristiki.

```
.SUBCKT NAND3 in1 in2 in3 out
* vhod1 vhod2 vhod3 Izhod
* NAND3 3-VHODNA NAND vrata logična 1 je 3V

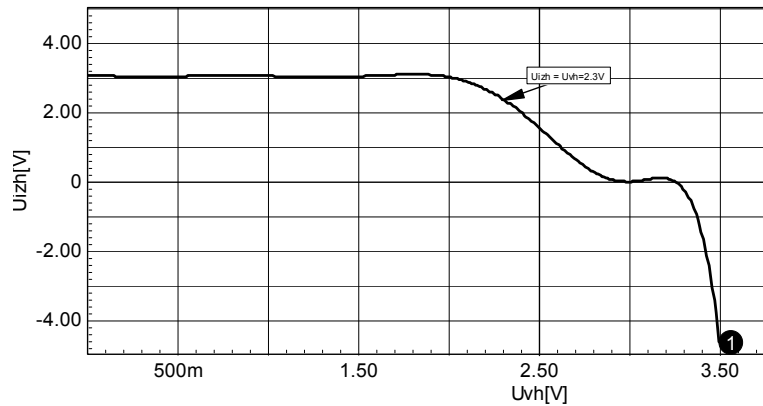
* model: makro 3

Rvh1 in1 0 1E6
Rvh2 in2 0 1E6
Rvh3 in3 0 1E6
E1 5 0 POLY(3) (in1 0) (in2 0) (in3 0) (0 1 1 1)
R5 5 0 1E6
E10 50 0 (5 0) (0 0.333333)
R10 50 0 1E6

G2 0 4 50 0
+ 1.002290E0,2.624314E-3,-1.491959E-1,8.325288E-2,1.321210E0,
+ -3.086306E0,2.880211E0,-1.322610E0,2.952994E-1,-2.558351E-2,
R4 4 0 1
C1 4 0 .34NF
EOUT out 0 (4 0) (0 3)
ROUT out 0 1MEG
.ENDS
*****
```

S polinomskim virom E1 modeliramo seštevalnik $U = U_1 + U_2 + U_3$, z virom G2 pa polinom 9. stopnje, ki ima podobno obliko kot idealna preklopna karakteristika. Ker je polinom izračunan

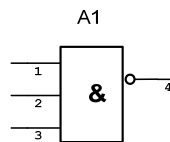
za napetost logične enice 1 V, vsoto napetosti delimo s 3. Iz preklopne karakteristike vrat je razvidno, da je preklopna napetost pri 2,3 V (glej sliko 5.24). Preklopno območje je od 2,0 V do 3,2V. Veljavnost modela je v območju od -0,1 V do 3,1 V.



Slika 5.24 Prenosna karakteristika polinomskega modela NAND vrat s strogo preklopno karakteristiko in preklopno napetostjo 2,3 V.

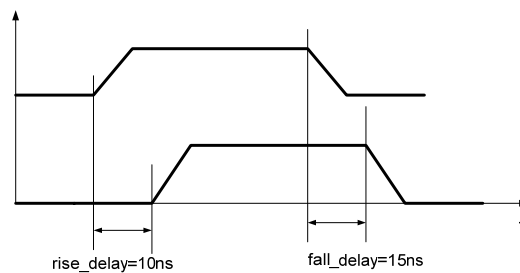
5.4.3. Kodni (notranji) model

Sodobni simulatorji vsebujejo tudi notranje modele osnovnih logičnih gradnikov, s pomočjo katerih lahko tvorimo lastne zunanje modele (glej sliko 5.25).



```
A1 [1 2 3] 4 NAND3
.MODEL NAND3 D_nand( rise_delay=30n fall_delay=17n
+ input_load=1p)
```

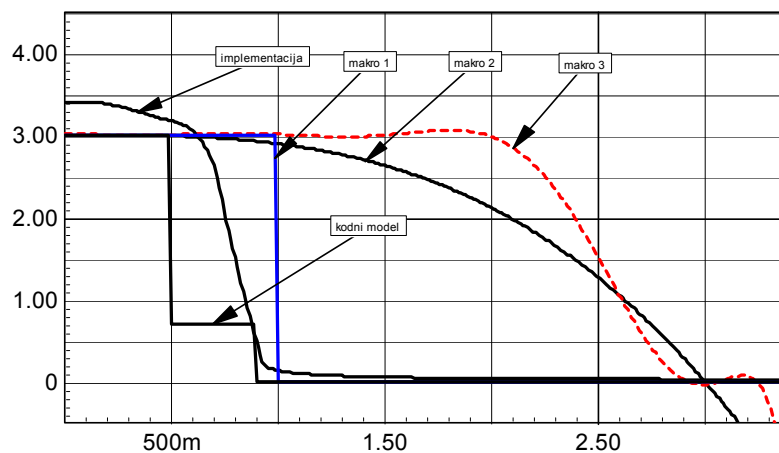
Slika 5.25 Kodni model⁷¹ trovhodnih NAND vrat. Ime notranjega modela je D_nand.



Slika 5.26 Definicija zakasnitev (Intusoft SPICE)

⁷¹ Intusoft SPICE.

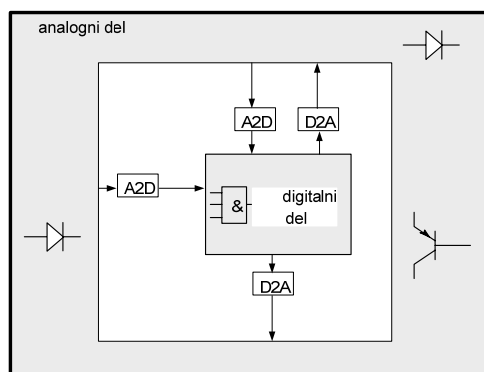
5.4.4. Primerjava modelov



Slika 5.27 Primerjava prenosnih karakteristik vseh 5 modelov

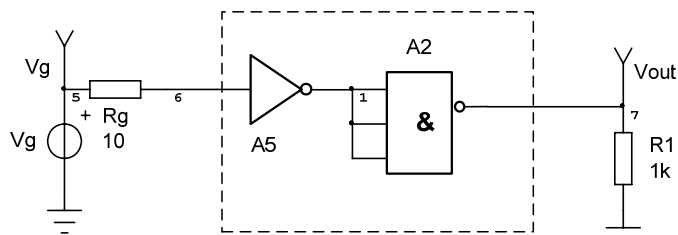
5.5. Kombinirani simulator

Kombinirani simulator (ang. mixed-mode simulator) je napravljen z združitvijo analognega in dogodkovno krmiljenega simulatorja. Večina sodobnih simulatorjev analognih elektronskih vezij je kombiniranih. Takšen simulator je sposoben simulirati obsežna analognodigitalna vezja. Osnovni logični gradniki so definirani kot notranji modeli, druge gradnike pa dobimo z ustrežno vezavo notranjih modelov ali pa tvorimo lasten notranji (kodni) model. Njegovo obnašanje opišemo s programskim jezikom (tipično C jezik ali pa VHDL) in jo vključimo v simulator⁷².



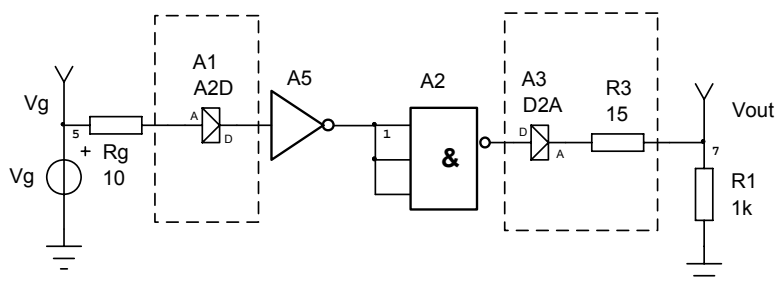
Slika 5.28 Analogno-digitalno vezje je treba ločiti na analogne in digitalne segmente

⁷² Za tvorjenje lastnih notranjih modelov potrebujemo posebna programska orodja.



Slika 5.29 Zgled analogno-digitalnega vezja, ki ga simuliramo s kombiniranim simulatorjem. Meja med digitalnim in analognim delom je označena s črtkanim okvirjem.

Za povezavo med analognimi in digitalnimi deli vezja skrbijo posebni vmesniki (mostiči), s katerimi definiramo vrednosti logičnih nivojev. Ime notranjega modela je `adc_bridge` oziroma `dac_bridge` (slika 5.28). Na zgledu, ki ga prikazujeta sliki 5.29 in 5.30, so označeni z A2D oziroma D2A. Če so priključki logičnih gradnikov v knjižnici ustrezno definirani⁷³, simulator med digitalni in analogni del sam vstavi ustrezne vmesnike. Samodejno vstavljene vmesnike vidimo samo v tekstovnem opisu (glej sliko 5.30 in 5.31).



Slika 5.30 Vstavljene vmesnike so označeni s črtkanim okvirjem (za ilustracijo je vezje posebej narisano na podlagi tekstovnega opisa iz slike 5.31)

⁷³ Npr. za CMOS invertor: *FAMILY CMOSin CMOSout.

```
*C:\UC_PROG\RVN\skripta\simulacije\Logicni_modeli_NAND3\Invertor_code2.cir
Vg 5 0 DC=10 PULSE 0.5 5 10n 0 0 100n
Rg 5 6 10

A5_Din_6 [ 6] [ 6_Din] A2D
A5_6_Din 1 INV_CDA5
.MODEL INV_CDA5 d_inverter( rise_delay=40n fall_delay=30n
+ input_load=1p)

A2 [ 1 1 1 ] 7 Dout NAND3_CDA2
.MODEL NAND3_CDA2 D_Nand( rise_delay=30n fall_delay=17n
+ input_load=1p)

X A2 Dout 7 7_Dout 7 D2Ag
.SUBCKT D2Ag in out
A1 [in] [1] D2A
R1 1 out 15
.ENDS

R1 7 0 1k

.MODEL D2A dac_bridge( out_low=0 out_high=3.50 out_undef=2.00
+ t_rise=1N t_fall=1N input_load=1P)
.MODEL A2D adc_bridge( in_low=1.00 in_high=3.00 rise_delay=.5N
+ fall_delay=.5N)
```

Slika 5.31 Tekstovni opis vezja iz slike 5.29

```
*****
*SRC=NAND3_CD;NAND3_CD ;Simple;Digital CMOS(code);NAND3
*SYM=NAND3_ISO
*HELP issyntax.hlp And
*NAME=D_Nand;A;A;[3]#1
*FAMILY_CMOSin CMOSout
*PARAM=rise_delay;rise delay;real;1n;1p -;no;-;yes
*PARAM=fall_delay;fall delay;real;1n;1p -;no;-;yes
*PARAM=input_load;input load value (F);real;1p;-;no;-;yes
.MODEL NAnd3_CD D_nand(Rise_Delay=1N Fall_Delay=1N
+ Input_Load=1P)
*****
```

Slika 5.32 Kompletan opis NAND3 modela, ki se uporablja v knjižnicah simulatorja Intusoft SPICE

Razlaga najpomembnejših parametrov⁷⁴ za oba vmesnika (dac_bridge in adc_bridge):

A2D vmesnik (za CMOS vhode):

in_low: maksimalna napetost logične ničle na vhodu je 1 V
in_high: minimalna napetost logične enice na vhodu je 3 V

D2A vmesnik (CMOS izhode):

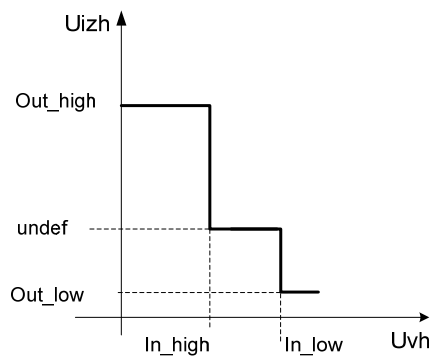
⁷⁴ Glej tudi sliko 5.15.

out_low: napetost logične ničle na izhodu je 0 V
out_high: napetost logične enice na izhodu je 3,5 V
out_undef: napetost, ki naj se pojavi na izhodu, kadar je vhodna napetost znotraj prepovedanega območja, je 2 V

rise_delay: zakasnitev pri vzponu je 0,5 ns
fall_delay: zakasnitev pri spustu je 0,5 ns

Če parametra *out_undef* ne nastavimo, ga simulator sam izračuna po formuli:

$$\text{out_undef} = (\text{in_high} + \text{in_low}) / 2$$



Slika 5.33 Prenosna karakteristika kodnega modela je sestavljena iz treh segmentov. Vrednosti posameznih točk so odvisne samo od vmesnikov.

6. LITERATURA

- [BALARAM,2000] A. Balaram: "*Modeling for printed-circuit board simulation*", IEEE Spectrum, oktober 2000, str.70–74.
- [BOLLINGER,1989] H. Bollinger, D. Larson, P. Arana: "*Autorouter Technology Update*", Circuit Design, oktober 1989, Vol. 6, šte. 10, str. 25–39.
- [BRATKOVIČ,1992] F. Bratkovič: "*Računalniško načrtovanje vezij – občutljivost in optimizacija*", Fakulteta za elektrotehniko v Ljubljani, Ljubljana 1992.
- [CALAHAN,1972] Donald A. Calahan: "*Computer Aided Network Design*", McGraw Hill, 1972.
- [DAI,1992] Wayne-Ming Dai: "*Multichip routing and placement*", IEEE Spectrum, november 1992, str. 61–64.
- [DOGŠA,2007] T. Dogša: "*CAE/CAD v elektroniki: Analiza in načrtovanje toleranc*", učbenik, FERi, Maribor 2007.
- [DOGŠA,2009] T. Dogša: "*CAE/CAD v elektroniki: Zbirka nalog z rešitvami*", zbirka vaj, FERi, Maribor 2009.
- [MARSH,1989] M. Marsh: "*Why doesn't my autorouter work*", Circuit Design, oktober 1989, Vol. 6, šte. 10, str. 8–14.
- [MEARES,1988] L. G. Meares and C. E. Hymowitz: "*Simulating with Spice*", Intusoft, San Pedro, CA, 1988.
- [RITCHEY,2000] Lee W. Ritchey: "*PCB routers and routing methods*", Printed Circuit Design, februar 2000, str. 26–29.
- [TUMA, 1997] T. Tuma: "*Analiza vezij s programom SPICE3*", Fakulteta za elektrotehniko v Ljubljani, Ljubljana 1997.
- [TUMA,2009] T. Tuma, Á. Bürmen: "*Circuit simulation with SPICE OPUS*", Birkhäuser Boston 2009.
- [ISMAIL,1999] Yehea I. Ismail, Eby G. Friedman, Jos'e L. Neves: "*Figures of merit to characterize the importance of on-chip inductance*", IEEE transactions on very large scale integration (vlsi) systems, vol. 7, no. 4, december 1999, str. 442–449.
- [WESTE,2005] N. Weste, K. Eshraghian: "*Principles of CMOS VLSI Design*", Addison-Wesley publishing company, 2. izdaja, 2005.