

Univerzita obrany v Brně

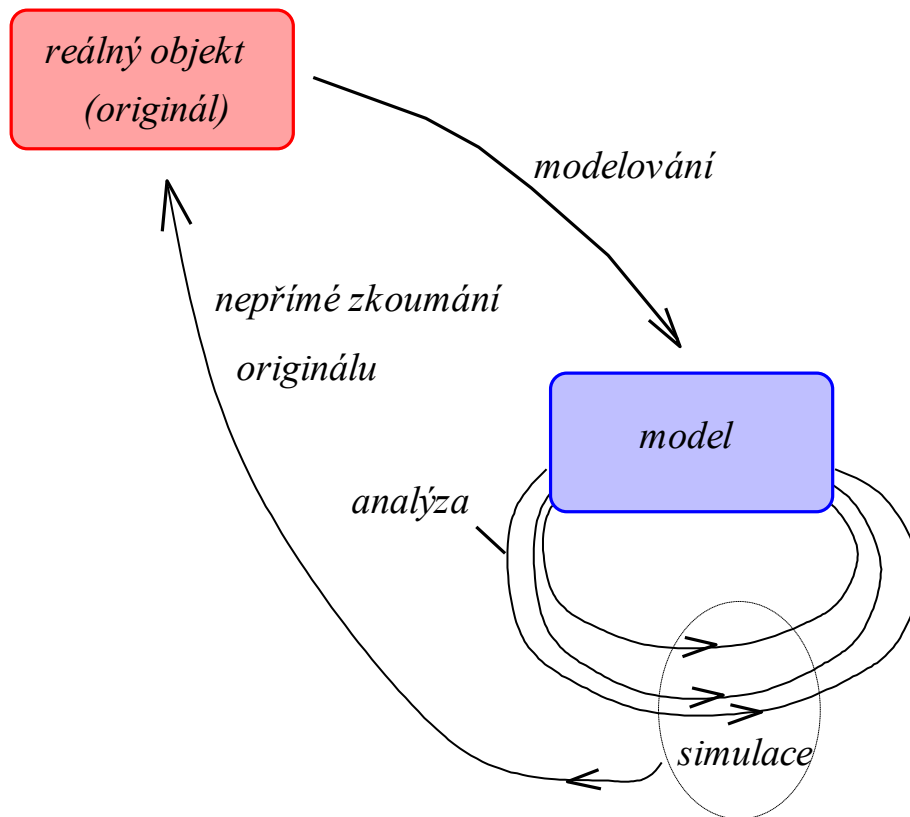
Navrhování obvodů počítačem

verze pro kombinované studium

Autor textu:
prof. Ing. Dalibor Biolek, CSc.

Leden 2020

Modelování, analýza, simulace



Modelování

je proces popisování reality omezenými prostředky, které máme k dispozici. Výsledkem je **model originálního objektu**.

Analýza

je jednorázová činnost, kdy zkoumáním **modelu** se pokoušíme zjistit určitou vlastnost **originálu**.

Simulace

je činnost, kdy analýzou **modelu** se pokoušíme o získání co nejvěrnějšího obrazu chování **originálu** za přesně stanovených podmínek.

Programy třídy SPICE a programy s určitou „SPICE-kompatibilitou“

1972 – SPICE (Simulation Program with Integrated Circuit Emphasis), Berkeley, USA

Dnes:

Programy třídy SPICE-

..HSpice AccuSim Spectre CADENCE-SPICE IG-SPICE I-SPICE Precise SmartSpice SPICE-Plus.

..PSpice..

..DSPICE B2SPICE AllSpice ICAP/4 Z-SPICE SPICE-IT! AimSpice TopSpice WinSpice..

pro PC

pro UNIXové pracovní stanice

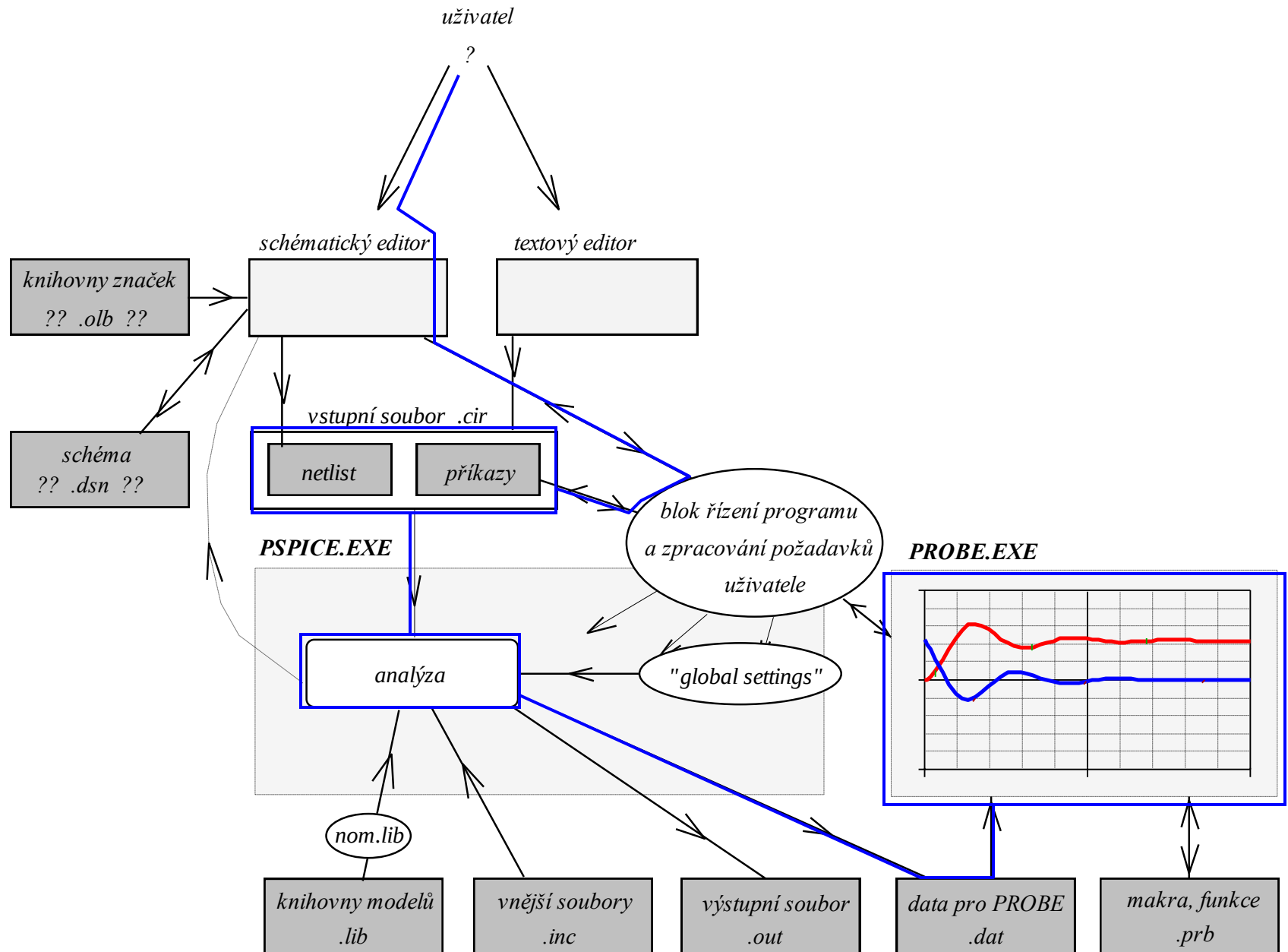
Programy, umožňující práci s modely SPICE nebo přímo využívat jazyka SPICE –

..MicroCap Electronics Workbench(MultiSIM) TINA.

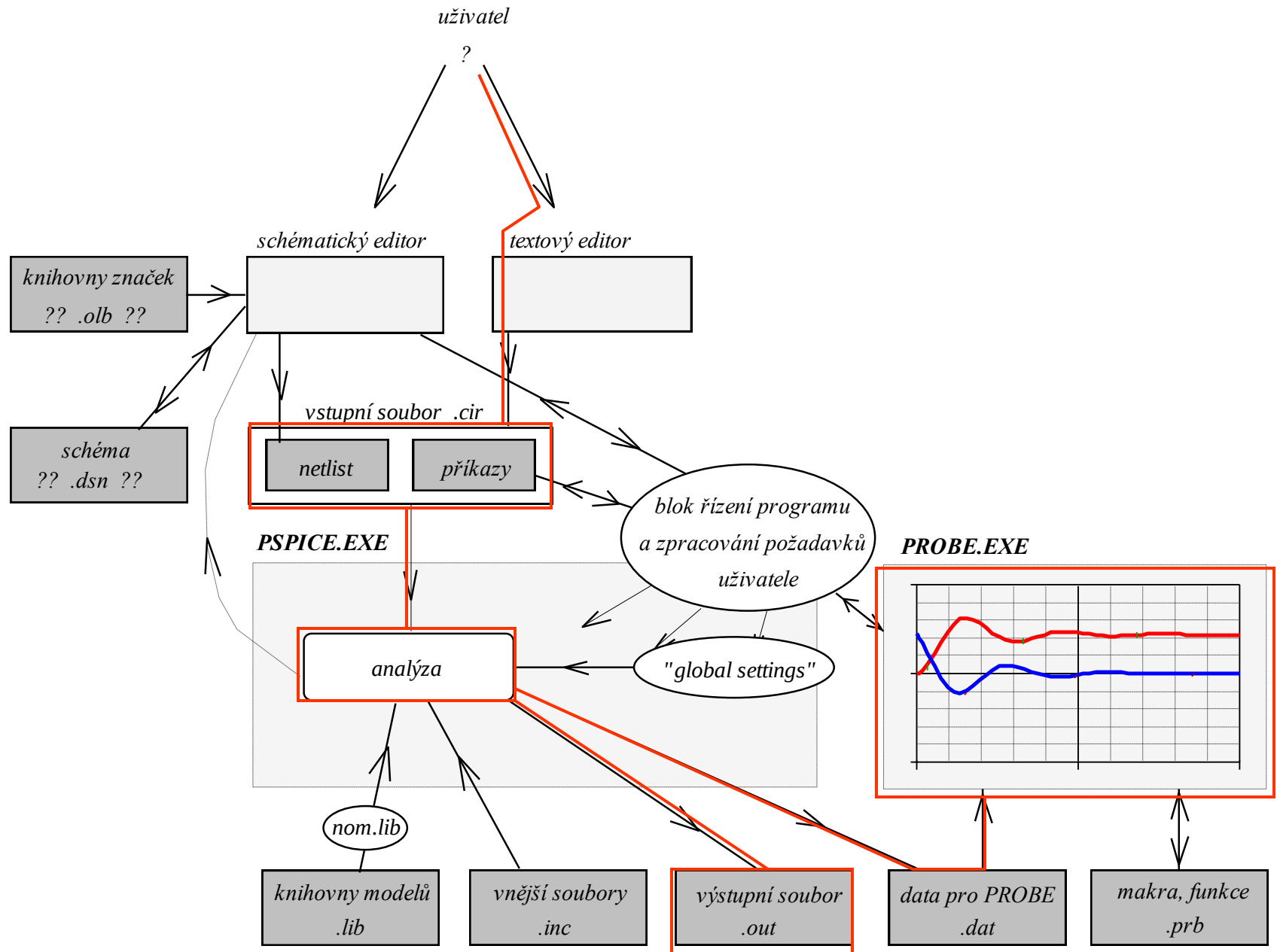
pro PC

Standardy jazyka SPICE - ..SPICE2 (verze G6) SPICE3 .. PSPICE

Typická struktura simulačního programu založeného na PSPICE



Typická struktura simulačního programu založeného na PSPICE



Schéma, vstupní soubor, netlist, simulace

VSTUPNÍ SOUBOR

*.cir

hlavička

popis obvodu = netlist

příkazy pro řízení
simulace

.end

RLC obvod

```
*  
Vzdroj 1 0 4.5V  
Ls 1 2 10mH  
Rs 2 3 100  
Rp 3 0 3.3k  
Cp 3 0 220nF  
*
```

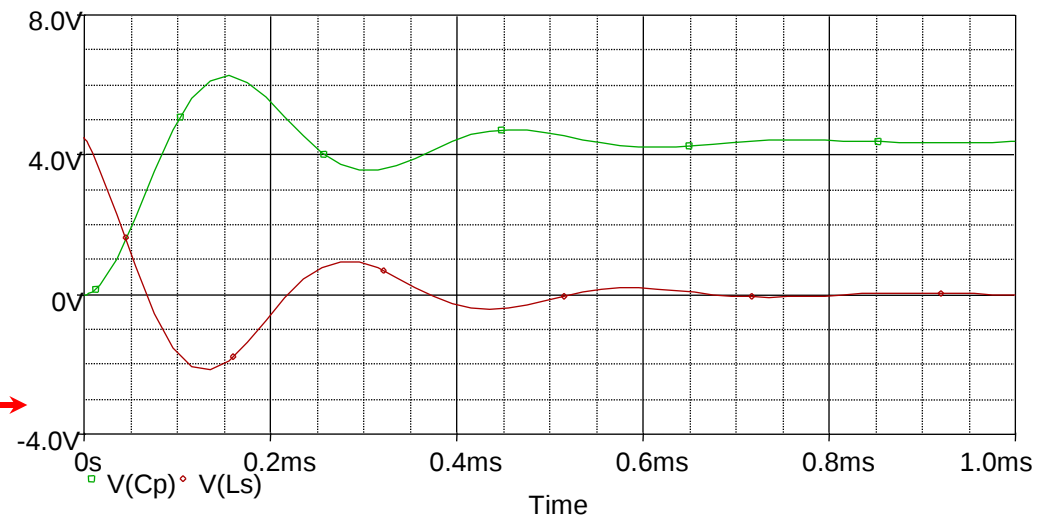
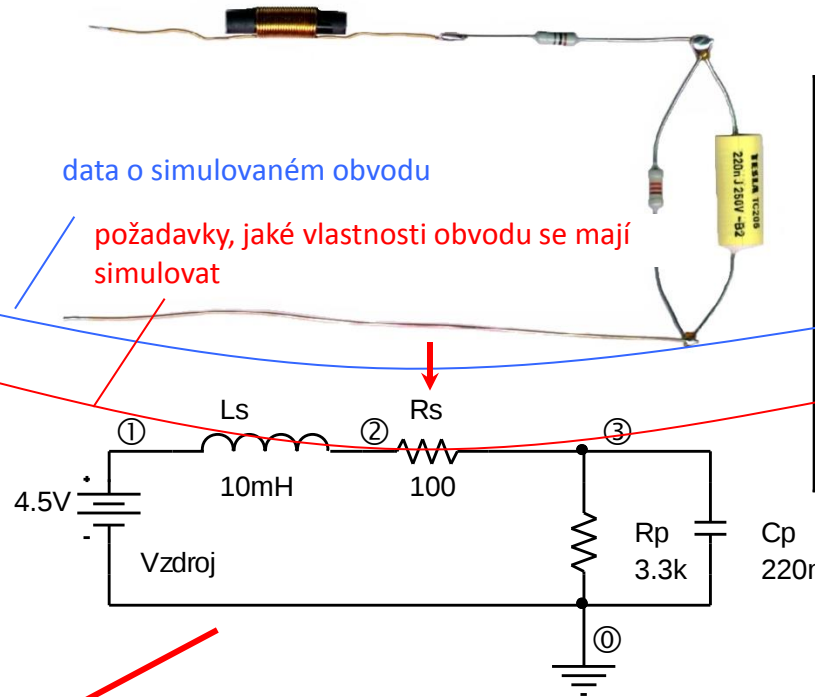
```
.TRAN 1u 1m SKIPBP  
.PROBE V(Cp) V(Ls)  
*  
.end
```

data o simulovaném obvodu

požadavky, jaké vlastnosti obvodu se mají
simulovat

SIMULÁTOR

simulace chování obvodu
podle dat ve vstupním
souboru



Typy analýz, analyzační módy a režimy

analyzační módy

- ~~Klasický~~
- ~~Probe (sonda)~~

základní typy analýz

- ~~Transient (časová)...~~
+ ss. prac. bod + ~~Fourier. analýza a DSP~~
- ~~AC (kmitočtová, střídavá)~~
+ ~~šumová analýza a DSP~~
- ~~DC (stejnosměrná)~~

analyzační režimy

- ~~Klasický~~
- ~~Stepping (krokování jednoduché a vícenásobné)~~
- ~~Teplotní analýza~~
- ~~Performance Analysis (vyhodnocovací analýza)~~
- ~~Monte Carlo a Worst Case (statistická analýza)~~
- ~~Optimization (optimalizace)~~

rozšiřující typy analýz

- ~~Dynamic DC (dynamická stejnosměrná)~~
- ~~Dynamic AC (dynamická střídavá)~~
- ~~Transfer Function (přenosová funkce)~~
- ~~Sensitivity (citlivostní analýza)~~

.....

omezení SPICE oproti MicroCapu

Zvláštnost SPICE: Lze spustit analýzu, i když nejsou zadány požadavky na to, co se má analyzovat. Pak SPICE vypočítá to, co standardně počítá vždy – **stejnosměrný pracovní bod.**

Některá omezení SPICE pro každou analýzu:

Odpor, kapacita a indukčnost nemohou nabývat nulových hodnot.

Ke každému uzlu musí být připojeny aspoň 2 prvky.

V obvodu musí být uzel „0“ – „uzemnění“.

Každý uzel musí mít stejnosměrnou cestu na zem.

Základní pravidla pro tvorbu textových souborů (vstupní soubor, netlist, knihovny..)

Malá a velká písmena se nerozlišují.

Názvy všech obvodových prvků musí být jedinečné. Nelze např. nazvat dvě baterie stejně, i když mají stejná napětí.

První řádek ve vstupním souboru je „titulek“, SPICE jej ignoruje.

Prázdné řádky se ignorují.

Vícenásobné mezery „navíc“ v textu se ignorují (místo jedné mezery jich může být více po sobě).

Každý řádek (kromě prázdného řádku a prvního řádku) musí začínat jedním z těchto symbolů:

Klíčovým znakem určeným pro identifikaci obvodového prvku, např. R pro rezistor

* Obsah řádku je pak chápán jako poznámka.

+ pokračování textu z předchozího řádku (pro lepší čitelnost v případě dlouhých řádek).

. Tečkou začínají dohodnuté příkazy.

Umístíme-li kamkoliv na řádek středník ; , pak vše vpravo od něj je chápáno jako poznámka.

Vstupní soubor musí končit řetězcem .END

Základní pravidla pro tvorbu textových souborů (vstupní soubor, netlist, knihovny..)

Uzly můžeme označovat přirozenými čísly (původní standard SPICE), nebo textovými řetězci, které nesmí obsahovat mezery (umožňuje většina současných verzí SPICE). **Jeden z uzlů musí být označen číslem 0.**

Čísla vyjadřujeme v přirozeném tvaru (0.052) s desetinnou tečkou, nikoliv čárkou, nebo v exponenciálním tvaru (5.2E-2), nebo pomocí inženýrské notace (52m).

V rámci řetězce se nesmí vyskytovat mezery.

V inženýrské notaci jsou tyto zvláštnosti: M nebo m je mili, MEG je mega, u je mikro (velikost písmen je nevýznamná).

Textový řetězec za číslem se ignoruje (1u = 1uF).

Klíčové znaky (první znak v jméně) analogových součástek

R .. rezistor **C** .. kapacitor **L** .. induktor

K .. přiřazení magnetické vazby mezi induktory nebo mag. jádra induktoru (induktorům)

V .. nezávislý zdroj napětí **I** .. nezávislý zdroj proudu

E .. zdroj napětí řízený napětím

F.. zdroj proudu řízený proudem

G.. zdroj proudu řízený napětím

H.. zdroj napětí řízený proudem

D .. dioda

B .. tranzistor GaAsFET (pozor – od SPICE verze 3F5: **B** = univerzální nelineární zdroj!!!)

J .. tranzistor JFET

M .. tranzistor MOSFET

Q .. bipolární tranzistor

Z .. IGBT tranzistor

S .. spínač řízený napětím

W .. spínač řízený proudem

T .. vedení

X .. podobvod (subobvod, funkční blok, makro)

Definice součástky v netlistu, která nevyužívá příkaz .MODEL

například pasivní součástky R, L, C a zdroje

<jméno součástky> <seznam uzlů, na které je připojena> <její vlastnosti> [upřesnění]

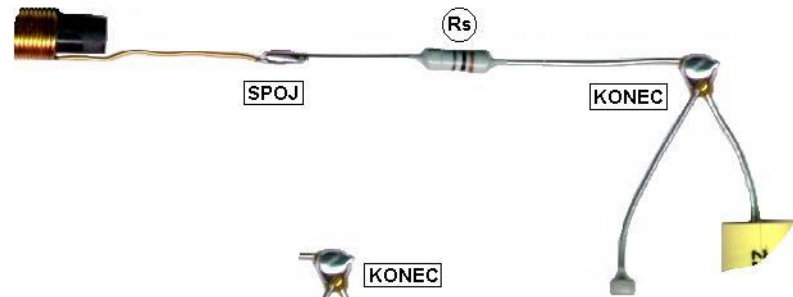
Rezistor

Syntaxe:

Rxx..xx <N⁺> <N⁻> <odpor>

Příklad:

Rs spoj konec 1k



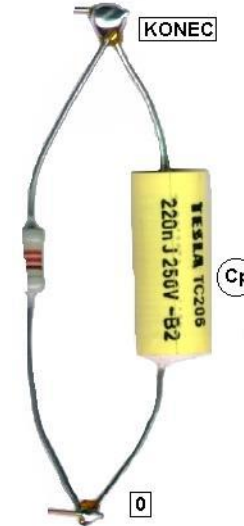
Kapacitor

Syntaxe:

Cxx..xx <N⁺> <N⁻> <kapacita>

Příklad:

Cp konec 0 220nF



Induktor

Syntaxe:

Lxx..xx <N⁺> <N⁻> <indukčnost>

Příklad:

Ls vstup spoj 10mH



Definice součástky v netlistu, která využívá příkaz .MODEL zejména diody a tranzistory

<jméno součástky> <seznam uzlů, na které je připojena> <jméno modelu> [upřesnění]

Příklady seznamu uzlů pro diodu a bipolární tranzistor:

<seznam uzlů> = <anoda katoda> ... dioda

<seznam uzlů> = <kolektor báze emitor> ... bipolární tranzistor

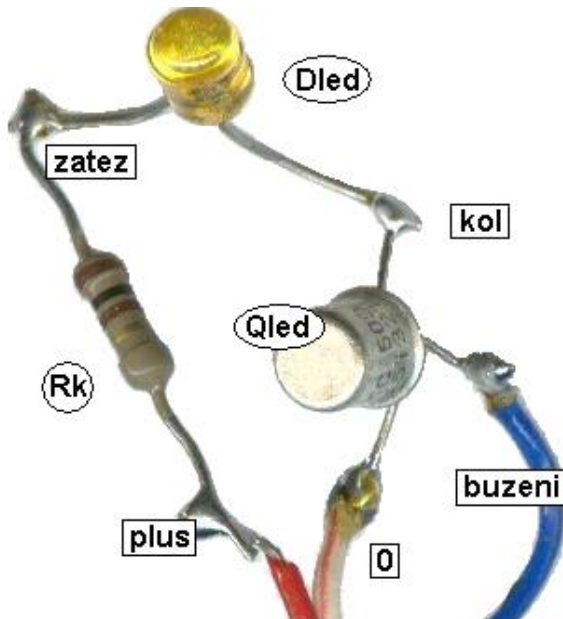
Obecná syntaxe modelu:

.MODEL <jméno modelu> <kód modelu> [() parametry modelu oddělené mezerami []]

Vybrané kódy modelu: **D** ..dioda,

NPN ..bipolární tranzistor NPN,

PNP ..bipolární tranzistor PNP.



```
LED indikace
*
*zapojeni
*
Rk plus zatez 100
Dled zatez kol ledka
Qled kol buzeni 0 KC509
*
*modely soucastek
*
.model ledka D (N=1.7 RS=.7 CJO=23.9P
+ IS=85.3p BV=6 IBV=10U VJ=0.75 TT=4.32U)
.model KC509 NPN (BF=1.50K BR=2 IS=12.0P
+ CJC=2.80P CJE=1.40P RC=1.01U VAF=100
+ TF=718.4P TR=10N MJC=375.7M VJC=690.0M
+ MJE=375.77M VJE=690.00M NF=1.259
+ ISE=33.7P ISC=0.001566F IKF=9.44M
+ IKR=4.62K NE=2.18 RE=32.4M VTF=113.5
+ ITF=533.9K XTF=19.25M)
.end
```

Některé základní příkazy

.TEMP .. definice teploty (temperature)

.TEMP <výčet teplot oddělených mezerami>

Příklady:

.TEMP 58 ; analýza proběhne jednou, pro teplotu 58 stupňů Celsia

.TEMP -28 11 54.5 ; analýza proběhne opakovaně pro teploty -28, 11 a 54,5 stupňů Celsia

.PARAM .. definice globálních parametrů (proměnných)

.PARAM <název parametru>=<hodnota parametru>

Při použití parametru ve výrazu se parametr musí umístit do složených závorek { }.

Příklady:

.PARAM odpor1=1k odpor2=5.2k kapacita=5nF

Rparalelni vstup zem {odpor1*odpor2/(odpor1+odpor2)}

Některé základní příkazy

.STEP .. krokování, vícenásobná analýza pro různé hodnoty krokované proměnné

.STEP <název proměnné> LIST <výčet hodnot oddělených mezerami>

.STEP [LIN] < název proměnné > <počáteční hodnota> <konečná hodnota> <krok>

.STEP OCT < název proměnné > <počáteční hodnota> <konečná hodnota> <počet bodů>

.STEP DEC < název proměnné > <počáteční hodnota> <konečná hodnota> <počet bodů>

LIN (nepovinné): lineární krokování, t.j. další hodnota=současná hodnota+krok

OCT, DEC: logaritmické krokování, t.j. další hodnota=současná hodnota*číslo větší než 1.

Jaké jsou možné názvy krokované proměnné:

- Název nezávislého zdroje napětí nebo proudu. Krokuje se napětí nebo proud tohoto zdroje.
- Typ modelu a jméno modelu, následované krokovaným parametrem v závorkách ().
- Teplota TEMP; Toto krokování nelze současně používat s příkazem .TEMP.
- Globální parametr, formát **PARAM=<název parametru>**. Rovnítko není povinné.

Příklady:

.STEP Vin 0V 3V 1V; analýza proběhne pro napětí zdroje Vin = 0V, 1V, 2V a 3V

.STEP PARAM=odpor LIST 1k 0.1K 2meg; analýza proběhne opakovaně pro 1kΩ, 100Ω, 2MΩ

.STEP DEC NPN Q1(BF) 10 1000 1; krokuje se parametr BF tranzistoru Q1, BF=10, 100, 1000

.STEP TEMP 10 100 20; krokuje se teplota v hodnotách 10, 30, 50, 70 a 90 stupňů Celsia

Současně nelze krokovat více proměnných – podstatné omezení SPICE.

Některé základní příkazy

.LIB .. napojení knihovny modelů na vstupní soubor

.LIB [název souboru]

Příklady:

.LIB pokus.lib ; napojení na soubor s názvem pokus.lib
.LIB "c:\data\ana.lib" ; napojení na soubor v adresáři, na který normálně SPICE „nevidí“
.LIB NOM.LIB ;napojení na všechny knihovny, které jsou vypsány v souboru NOM.LIB
.LIB ;stejný význam jako .LIB NOM.LIB (NOM.LIB je tzv. Master Library)

Co všechno může být obsaženo v souboru *.lib:

Poznámky

Definice modelů (.MODEL)

Definice podobvodů (.SUBCKT)

Definice globálních parametrů (.PARAM)

Definice vlastních funkcí (.FUNC)

Příkazy .LIB

Některé základní příkazy

.INC .. včlenění obsahu vnějšího souboru (INClude)

.INC [název souboru]

Příklady:

.INC zesilovac.cir začlenění celého netlistu do vstupního souboru

.INC "c:\data\pok.txt"

Význam .INC:

Efekt je stejný, jako kdyby se v místě příkazu .INC objevil text z včleněného souboru. Například ve vstupním souboru místo celého netlistu umístíme pouze odkaz na něj.

Co nesmí být obsaženo ve včleňovaném souboru:

Úvodní hlavička, pokud není označena jako poznámka (* nebo ;)

Je povoleno „vnořování“ těchto souborů do 4 úrovní.

Některé základní příkazy

.OP .. způsobí detailní výpis stejnosměrného pracovního bodu do výstupního souboru

.OP

Pozor, nejde o příkaz spouštějící analýzu, pouze jde o výpis výsledků. Pracovní bod se počítá vždy, i při chybějícím příkazu .OP.

Co se vypisuje bez příkazu .OP:

Uzlová napětí, proudy přes zdroje napětí, celkový ztrátový výkon.

Co se navíc vypisuje s příkazem .OP:

Linearizované malosignálové parametry polovodičových prvků a nelineárních řízených zdrojů.

Některé základní příkazy

.OP .. způsobí detailní výpis stejnosměrného pracovního bodu do výstupního

.OP

Pozor, nejde o příkaz spouštějící analýzu, pouze jde o výpis výsledků. Při chybějícím příkazu .OP.

Co se vypisuje bez příkazu .OP:

Uzlová napětí, proudy přes zdroje napětí, celkový ztrátový výkon.

Co se navíc vypisuje s příkazem .OP:

Linearizované malosignálové parametry p

**** BIPOLAR JUNCTION TRANSISTORS

NAME	QT
MODEL	Q2N2222
IB	2.94E-05
IC	5.15E-03
VBE	6.88E-01
VBC	-3.12E+00
VCE	3.81E+00
BETADC	1.75E+02
GM	1.96E-01
RPI	9.53E+02
RX	1.00E+01
RO	1.50E+04
CBE	1.18E-10
CBC	4.17E-12
CJS	0.00E+00
BETAAC	1.87E+02
CBX/CBX2	0.00E+00
FT/FT2	2.55E+08

NODE	VOLTAGE	NODE	VOLTAGE	N
(1)	0.0000	(2)	1.7246	(
(5)	1.0367	(6)	10.0000	

VOLTAGE SOURCE CURRENTS

NAME	CURRENT
------	---------

Vin	0.000E+00
-----	-----------

Vbat	-5.650E-03
------	------------

TOTAL POWER DISSIPATION	5.65E-02	WATTS
-------------------------	----------	-------

Práce s podobvody

Podobvod (subcircuit) je relativně samostatný obvod, který komunikuje s okolím prostřednictvím konkrétních vývodů. Nejprve jej musíme definovat (*deklarovat*), tj. vytvořit jeho model, uložit jej do dané knihovny nebo přímo do vstupního souboru, a pak jej můžeme využívat (*volat*) jako každou jinou součástíku.

Pokud podobvod deklarujeme přímo ve vstupním souboru analyzovaného obvodu: Model podobvodu bude k dispozici jen pro analýzu tohoto obvodu.

Pokud podobvod umístíme do externí knihovny **knihovna.lib**: Model podobvodu bude k dispozici každému, kdo může využívat tuto knihovnu. Na knihovnu se napojíme tak, že do vstupního souboru umístíme příkaz

.lib knihovna.lib

Deklarace podobvodu - syntaxe:

```
.SUBCKT <jméno podobvodu> <seznam vývodů> [params: <parametr=číselná hodnota>]  
<popis vnitřního zapojení>  
.ENDS
```

Použití (volání) podobvodu - syntaxe:

```
Xxx..xx <seznam uzlů podobvodu> <jméno podobvodu> [params: <parametr>=<čís.hodnota>]
```

Práce s podobvod

Příklad – podobvod bez parametrů:

model vedeni

Vin 1 0 1V

Rin 1 2 75ohmu

Xusek1 2 3 0 usek_vedeni

Xusek2 3 4 0 usek_vedeni

Xusek3 4 5 0 usek_vedeni

Xusek4 5 6 0 usek_vedeni

Rout 6 0 75ohmu

*

.lib vedeni.lib

*

.end

Knihovna vedeni.lib:

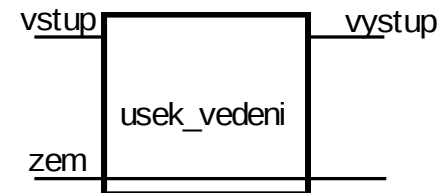
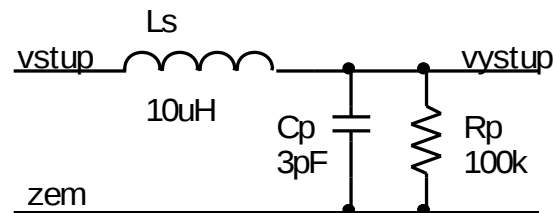
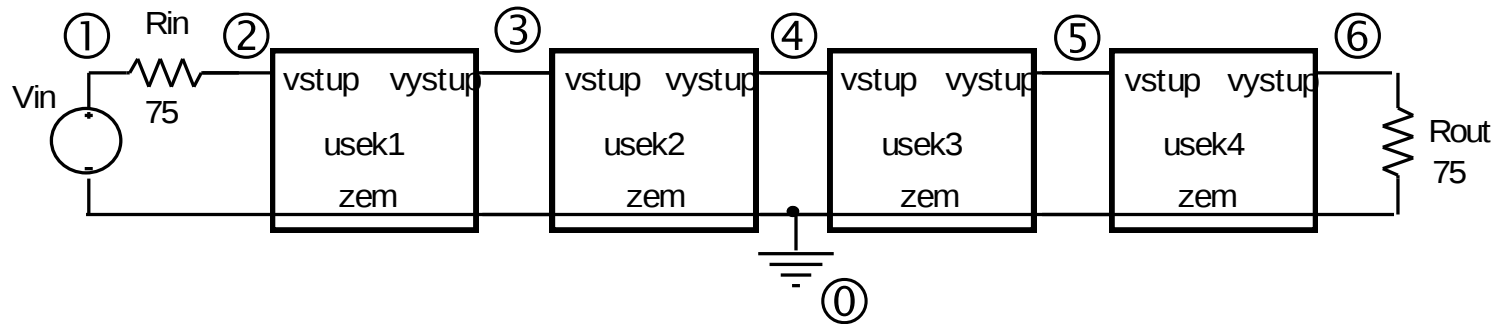
.subckt usek_vedeni vstup vystup zem

Ls vstup vystup 10uH

Cp vystup zem 3pF

Rp vystup zem 100kohmu

.ends usek_vedeni



Práce s podobvod

Příklad – podobvod s parametry:

model vedeni

Vin 1 0 1V

Rin 1 2 75ohmu

Xusek1 2 3 0 usek_vedeni params: Ls=1u Cp=3p Rp=100k

Xusek2 3 4 0 usek_vedeni params: Ls=10u Cp=30p Rp=98k

Xusek3 4 5 0 usek_vedeni params: Ls=34u Cp=4p Rp=1meg

Xusek4 5 6 0 usek_vedeni params: Ls=1u Cp=3p Rp=100k

Rout 6 0 75ohmu

*

.lib vedeni.lib

*

.end

Knihovna vedeni.lib:

.subckt usek_vedeni vstup vystup zem

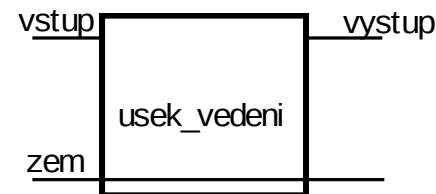
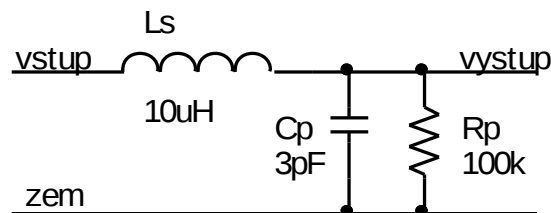
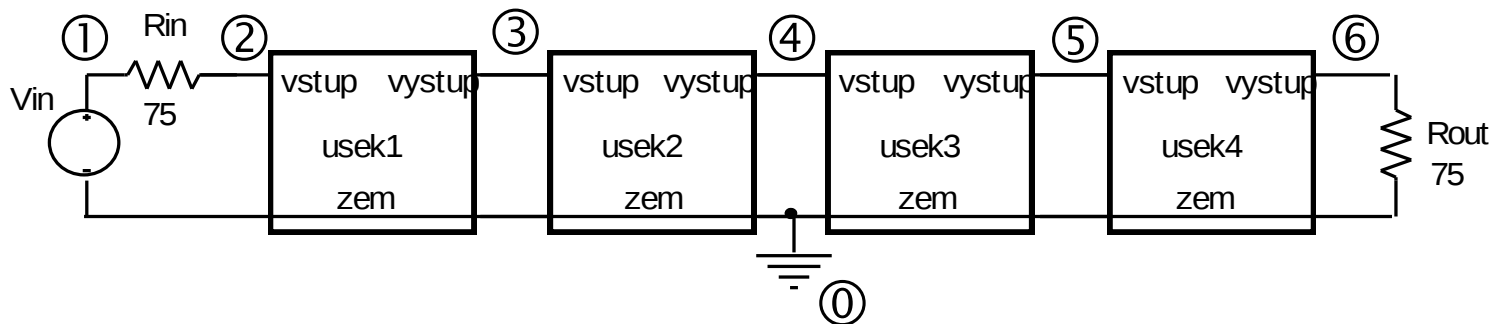
+params: indukcnost=1u kapacita=3p odpor=10

Ls vstup vystup {indukcnost}

Cp vystup zem {kapacita}

Rp vystup zem {odpor}

.ends usek_vedeni



Zabudované funkce PSpice pro modelování prvků - výběr

funkce	význam	poznámka
ABS(x)	$ x $	
COS(x)	$\cos(x)$	x je v radiánech
DDT(x)	dx/dt	jen u analýzy TRANSIENT
EXP(x)	e^x	
IF(a,x,y)	x když a=TRUE, jinak y	a je logický výraz, například IF(I(L1)>0,V(1),-V(1))
IMG(x)	imaginární část x	
LIMIT(x,MIN, MAX)	MIN když $x < \text{MIN}$, MAX když $x > \text{MAX}$, jinak x	oboustranný okrajovač
LOG(x)	$\ln(x)$	základ e
LOG10(x)	$\log(x)$	základ 10
M(x)	$ x $	modul komplexního čísla x
MAX(x,y)	větší z čísel x a y	
MIN(x,y)	menší z čísel x a y	
P(x)	argument komplexního čísla	
PWR(x,y)=(x**y)	$ x ^y$	mocninná funkce
PWRS(x,y)	$ x ^y$ když $x > 0$, $- x ^y$ když $x < 0$	mocninná funkce se znaménkem
R(x)	reálná část x	
SDT(x)	$\int_0^t x(t) dt$	jen u analýzy TRANSIENT
SGN(x)	1 pro $x > 0$, -1 pro $x < 0$, 0 pro $x = 0$	znaménková funkce
SIN(x)	$\sin(x)$	x je v radiánech
STP(x)	1 když $x > 0$, jinak 0	jednotkový skok
SQRT(x)	$\sqrt{x}$	
TABLE(x,x1,y1,x2,y2,...,xn,yn)	Graf funkce vznikne pospojováním bodů o souřadnicích [xk,yk] lomenou čarou. Pro $x < x_1$ je výsledek y1, pro $x > x_n$ je výsledek yn.	

Vytváření vlastních funkcí – příkaz .FUNC

.FUNC <jméno> ([argumenty]) {<vzorec>}

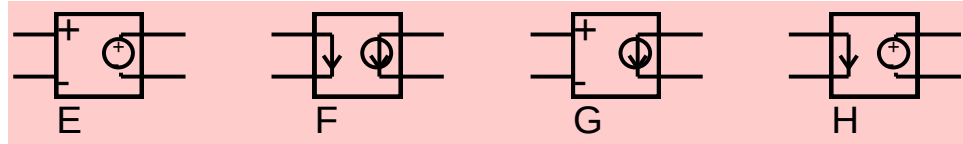
Příklady:

.FUNC LIMITER(x,gain) {LIMIT(gain*x,-15,15)};omezovač se zesílením GAIN, signál x větší
+ než +15V a menší než -15V ořeže
.FUNC MIN3(A,B,C) {MIN(A,MIN(B,C))} ; minimální z čísel A, B, C

Funkce lze psát do zvláštního souboru. Ze vstupního souboru na ně bude „vidět“ po vložení příkazu
.INC soubor

Řízené (závislé) zdroje pro behaviorální modelování (ABM –Analog Behavioral Modeling)

E..zdroj napětí řízený napětím
F..zdroj proudu řízený proudem
G..zdroj proudu řízený napětím
H..zdroj napětí řízený proudem



Zdroje E a G mají stejnou syntaxi. Níže vysvětleno zejména pomocí zdrojů E.

1) Zdroj s řídicími svorkami, lineární zařízení:

E<název> <výstup+> <výstup-> <vstup+> <vstup-> <přenos>

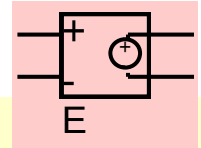
Vzorec: $\text{výstup} = \text{přenos} * \text{vstup}$

Příklady:

Ezesil 2 0 1 0 100 ; lineární zesilovač, vstup 1-0, výstup 2-0, zesílení 100

GOTA out 0 plus minus {gm} ; OTA zesilovač s výstupem out, vstupy plus-minus
+a transkonduktancí gm, která je definována jako globální parametr

Řízené (závislé) zdroje pro behaviorální modelování (ABM –Analog Behavioral Modeling)



2) Zdroj s řídicími svorkami, nelineární zařízení:

E<název> <výstup+> <výstup-> POLY(<dimenze>) {<vstup+> <vstup->} {<koeficient>}

seznam dvojic řídicích svorek

seznam koeficientů polynomu

Vzorec: Nelineární polynomiální závislosti výstupu na řídicích vstupech, viz tabulka

exp. u	p0	p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	...
x1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	...
x1 x2	0	1 0	0 1	2 0	1 1	0 2	3 0	2 1	1 2	0 3	4 0	3 1	2 2	1 3	0 4	...
x1 x2 x3	0	1 0 0	0 1 0	0 0 1	2 0 0	1 1 0	1 0 1	0 2 0	0 1 1	0 0 2	3 0 0	2 1 0	2 0 1	1 2 0	1 1 1	...
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...

a) Dimenze = 1, je jeden vstup x1, výstup y.

$$y = p_0 + p_1 x_1 + p_2 x_1^2 + \dots$$

Příklady:

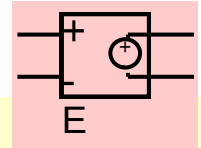
Ekvadrat 2 0 POLY(1) 1 0 0 0 0.1 ; vstupní napětí mezi 1 a 0, výstupní napětí mezi 2 a 0.

+Jde o kvadrátor o vzorci $V(2)=V(1)*V(1)/10$.

Ex out 0 POLY(1) in1 in2 0 5 -0.5 ; vstupní napětí mezi in1 a in2, výstupní napětí mezi

+out a 0. Funkční měnič $V(out)=(1-V(in1,in2))/2$.

Řízené (závislé) zdroje pro behaviorální modelování (ABM –Analog Behavioral Modeling)



2) Zdroj s řídicími svorkami, nelineární zařízení:

E<název> <výstup+> <výstup-> POLY(<dimenze>) {<vstup+> <vstup->} {<koeficient>}

seznam dvojic řídicích svorek

seznam koeficientů polynomu

Vzorec: Nelineární polynomiální závislosti výstupu na řídicích vstupech, viz tabulka

exp. u	p0	p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	...
x1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	...
x1 x2	0	1 0	0 1	2 0	1 1	0 2	3 0	2 1	1 2	0 3	4 0	3 1	2 2	1 3	0 4	...
x1 x2 x3	0	1 0 0	0 1 0	0 0 1	2 0 0	1 1 0	1 0 1	0 2 0	0 1 1	0 0 2	3 0 0	2 1 0	2 0 1	1 2 0	1 1 1	...
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...

b) Dimenze = 2, jsou dva vstupy x1 a x2.

Příklady:

Emulti 3 0 POLY(2) 1 0 2 0 0 0 0 0 0 0.1

V(3)=V(1)*V(2)/10.

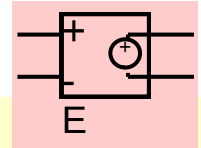
Esum out 0 POLY(2) in1 0 in2 0 0 1 1

$$y = p_0 + p_1x_1 + p_2x_2 + p_3x_1^2 + p_4x_1x_2 + p_5x_2^2 + p_6x_1^3 + p_7x_1^2x_2 + p_8x_1x_2^2 + p_9x_2^3 + \dots$$

; dvojice vstupních napětí mezi 1-0 a 2-0, výstupní +napětí mezi 3-0. Jde o násobičku

; dvojice vstupních V(in1) a V(in2), výstupní napětí +V(out). Sumátor V(out)=V(in1)+V(in2).

Řízené (závislé) zdroje pro behaviorální modelování (ABM –Analog Behavioral Modeling)



3) Zdroj bez řídicích svorek, výstupní napětí je modelováno vzorcem:

E<název> <výstup+> <výstup-> VALUE [=] {<vzorec>}

Příklady:

Ekvadrat out 0 VALUE={V(in)^2/10}

; kvadrátor o vzorci $V(out)=V(1)*V(1)/10$.

Ex 2 0 VALUE {10*(-0.5+STP(V(1)))}

; komparátor: V(2) je buď +5V nebo -5V podle toho, zda V(1) je kladné nebo záporné.

4) Zdroj bez řídicích svorek, výstupní napětí je modelováno po úsecích lineární závislosti:

E<název> <výstup+> <výstup-> TABLE {<vzorec>} [=] (seznam dvojic <xk yk>)

Příklady:

Ekvadrat out 0 VALUE={V(in)^2/10}

; kvadrátor o vzorci $V(out)=V(1)*V(1)/10$.

Ex 2 0 VALUE {10*(-0.5+STP(V(1)))}

; komparátor: V(2) je buď +5V nebo -5V podle toho, zda V(1) je kladné nebo záporné.

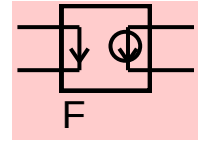
Ex 2 0 TABLE {V(1)} (-4 -5 4 5)

; pro $V(1) < -4V$ je $V(2) = -5V$, pro $V(1) > 4V$ $V(2) = 5V$
+ pro $-4V < V(1) < 4V$ je graf dán spojnicí obou bodů
; pro $10*V(1) < -4V$ je $V(2) = -5V$,
+ pro $10*V(1) > 4V$ $V(2) = 5V$

Ex 2 0 TABLE {10*V(1)} (-4 -5 4 5)

+ pro $-4V < 10*V(1) < 4V$ je graf dán spojnicí bodů
+ $[-0.4, -5]$, $[0.4, 5]$

Řízené (závislé) zdroje pro behaviorální modelování (ABM –Analog Behavioral Modeling)



Zdroje F a H mají stejnou syntaxi. Níže vysvětleno zejména pomocí zdrojů F.

Jsou to zdroje napětí (H) a proudu (F), řízené **proudem**. Proud je snímán vždy na nezávislém zdroji napětí, který musí být specifikován v definici řízeného zdroje namísto řídicích svorek. Chceme-li, aby se tento zdroj choval jako ideální „ampérmetr“, musíme jeho napětí položit rovné nule.

1) Lineární řízený zdroj:

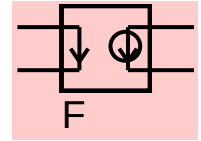
F<název> <výstup+> <výstup -> V<název řídicího zdroje> <přenos>

Vzorec: $\text{výstup} = \text{přenos} * \text{proud řídicím zdrojem}$

Příklad:

Fzesil 2 0 Vin 100 ; lineární proudový zesilovač, vstup 1-0, výstup 2-0, zesílení 100
Vin 1 0 0V ; pomocný zdroj napětí mezi svorkami 1-0 o nulovém napětí

Řízené (závislé) zdroje pro behaviorální modelování (ABM –Analog Behavioral Modeling)



2) Nelineární řízený zdroj:

F<název> <výstup+> <výstup-> POLY(<dimenze>) {V<název říd. zdroje>} {<koefficient>}

seznam řídících zdrojů

seznam koeficientů polynomu

exp. u	p0	p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	...
x1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	...
x1 x2	0	1 0	0 1	2 0	1 1	0 2	3 0	2 1	1 2	0 3	4 0	3 1	2 2	1 3	0 4	...
x1 x2 x3	0	1 0 0	0 1 0	0 0 1	2 0 0	1 1 0	1 0 1	0 2 0	0 1 1	0 0 2	3 0 0	2 1 0	2 0 1	1 2 0	1 1 1	...
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...

Příklad:

Fdif out 0 POLY(2) V1 V2 0 1 -1 ; výstupní proud je rozdíl proudů tekoucích přes zdroje V1 a V2

Řízené (závislé) spínače pro behaviorální modelování

S .. spínač řízený napětím
W .. spínač řízený proudem



Spínač „S“:

S<název> <spínač+> <spínač-> <řízení+> <řízení-> <jméno modelu>

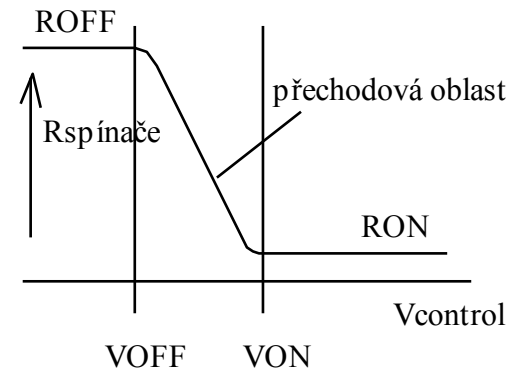
Definice modelu spínače:

.MODEL <jméno modelu> VSWITCH [parametry modelu]

V PSpice existují dva různé modely spínače:

„**Variable-Resistance Switch**“ (VRS) a „**Short-Transition Switch**“ (STS).

VRS			
parametr	popis	jednotka	default
ROFF	odpor v rozpojeném stavu	Ohm	1MEG
RON	odpor v sepnutém stavu	Ohm	1
VOFF	vypínací napětí	V	0
VON	spínací napětí	V	1



Řízené (závislé) spínače pro behaviorální modelování

S .. spínač řízený napětím
W .. spínač řízený proudem



Spínač „S“:

S<název> <spínač+> <spínač-> <řízení+> <řízení-> <jméno modelu>

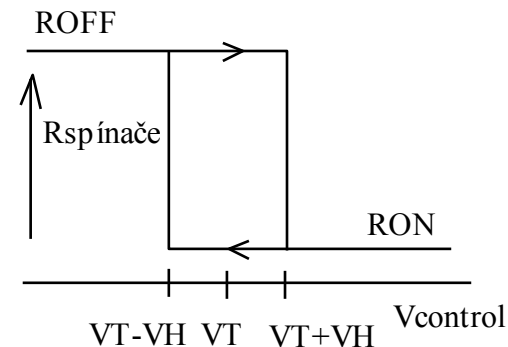
Definice modelu spínače:

.MODEL <jméno modelu> VSWITCH [parametry modelu]

V PSpice existují dva různé modely spínače:

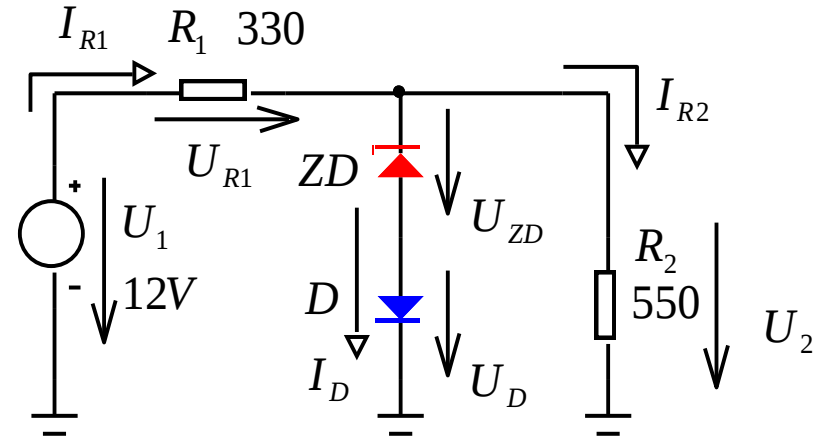
„Variable-Resistance Switch“ (VRS) a „Short-Transition Switch“ (STS).

STS			
parametr	popis	jednotka	default
ROFF	odpor v rozpojeném stavu	Ohm	1T
RON	odpor v sepnutém stavu	Ohm	1
VT	prahové napětí	V	0
VH	hysterezi napětí	V	0
TD	časové zpoždění	s	0

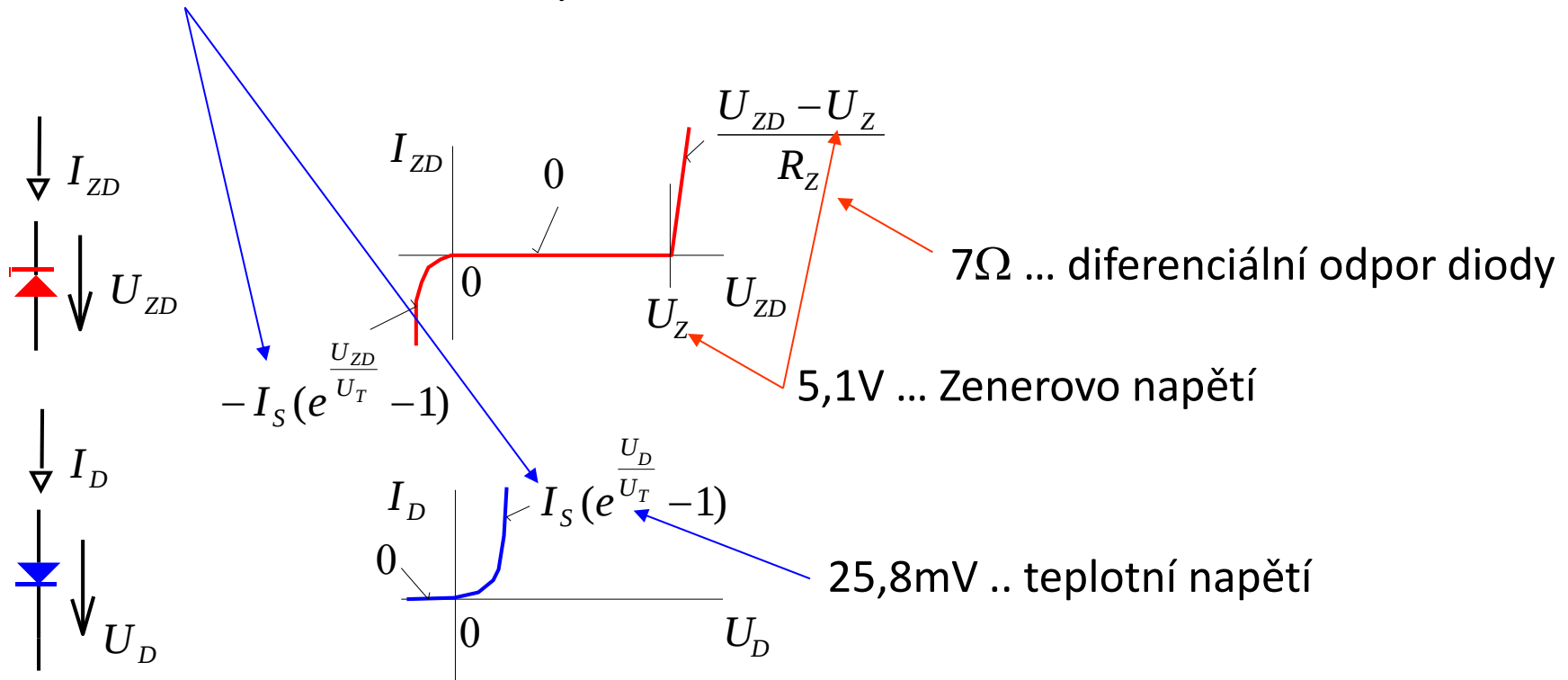


Výpočet ss pracovního bodu jako základ všech analýz

Analýza stabilizátoru napětí. ZD je Zenerova dioda 1N3826 o napětí $U_Z = 5.1 \text{ V}$, D je „klasická“ křemíková dioda 1N459.



$3,5 \cdot 10^{-13} \text{ A}$.. saturační proud



Výpočet ss pracovního bodu jako základ všech analýz

“Ruční” řešení:

$$\left(R_Z + \frac{R_1 R_2}{R_1 + R_2}\right) I_D + U_T \ln\left(1 + \frac{I_D}{I_S}\right) + U_Z - \frac{R_2}{R_1 + R_2} U_1 = 0$$

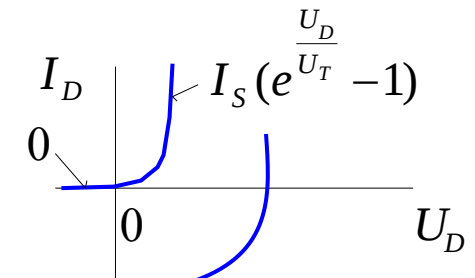
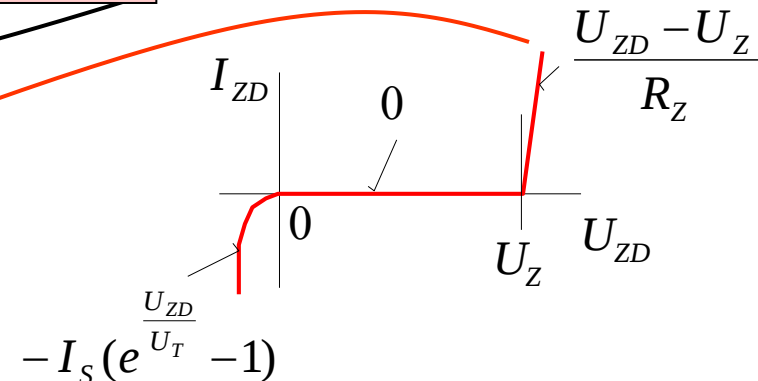
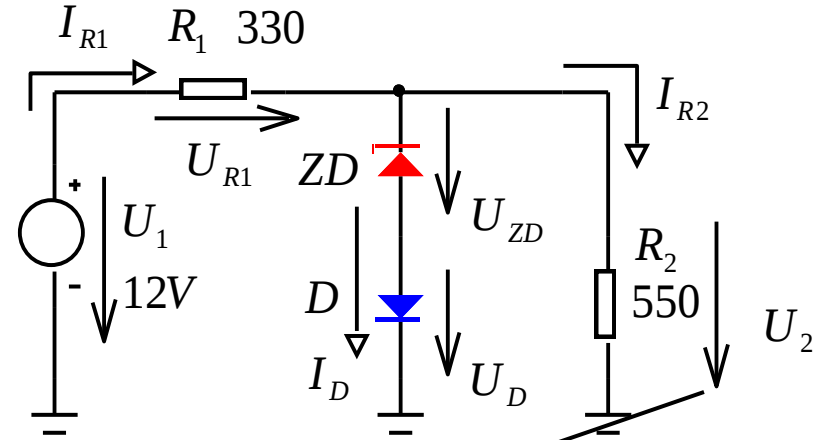
$$f(I_D) \doteq I_D + 1,2118 \cdot 10^{-4} \ln(1 + 2,8571 \cdot 10^{12} I_D) - 1,1125 \cdot 10^{-2} = 0$$

$$U_2 = \frac{R_2}{R_1 + R_2} U_1 - \frac{R_1 R_2}{R_1 + R_2} I_D$$

$$U_{ZD} = R_Z I_D + U_Z$$

$$U_2 = R_Z I_D + U_Z + U_T \ln\left(1 + \frac{I_D}{I_S}\right)$$

$$U_D = U_T \ln\left(1 + \frac{I_D}{I_S}\right)$$

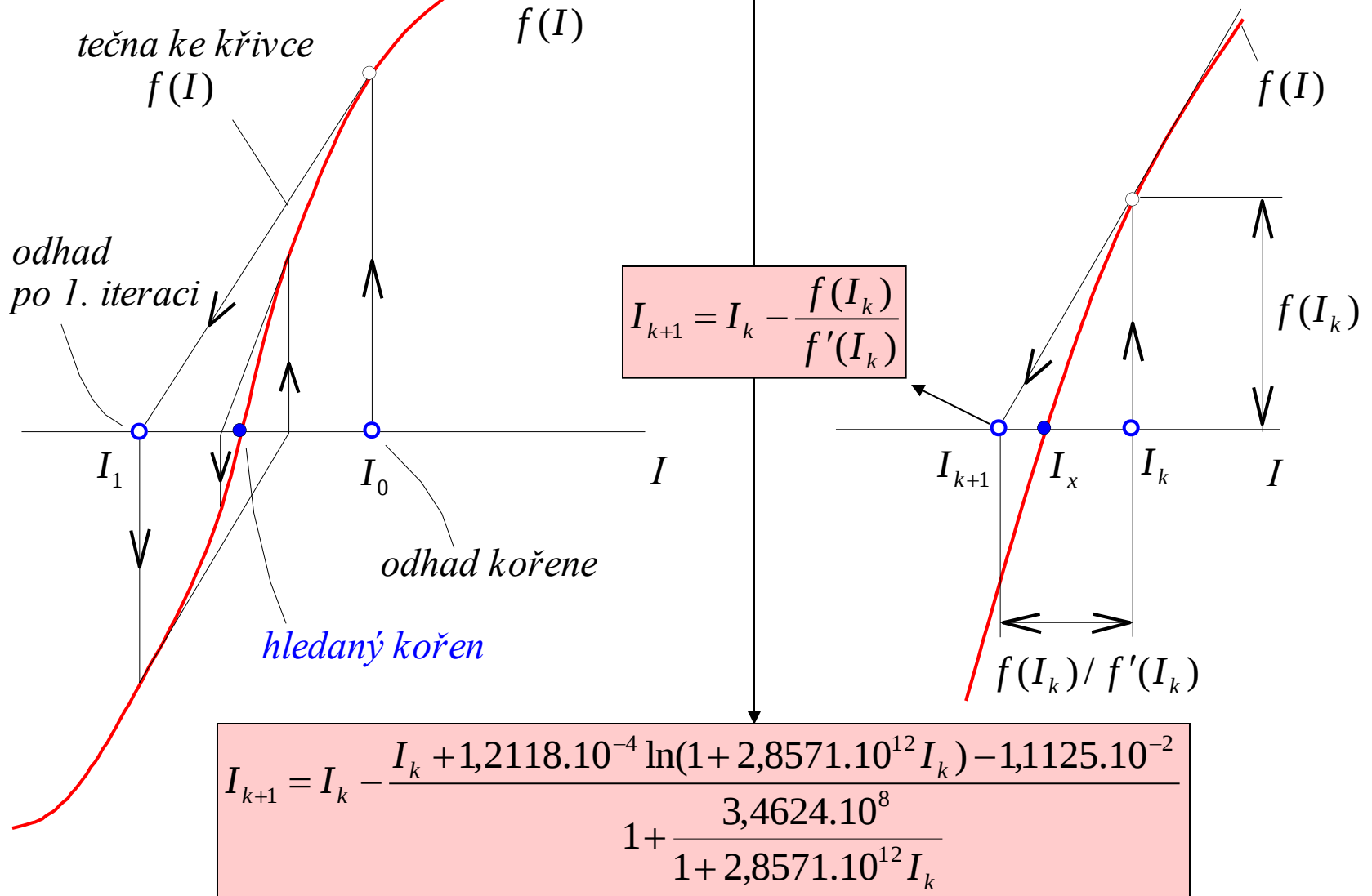


Výpočet ss pracovního bodu jako základ všech analýz

“Ruční” řešení:

$$f(I_D) \doteq I_D + 1,2118 \cdot 10^{-4} \ln(1 + 2,8571 \cdot 10^{12} I_D) - 1,1125 \cdot 10^{-2} = 0$$

Newtonova iterační metoda



Výpočet ss pracovního bodu jako základ všech analýz

“Ruční” řešení:

například MATLAB:

```
function y=newt(x)
```

```
y=x-(x+1.2118e-4*log(1+2.8571e12*x)-1.1125e-2)/(1+3.4624e8/(1+2.8571e12*x));
```

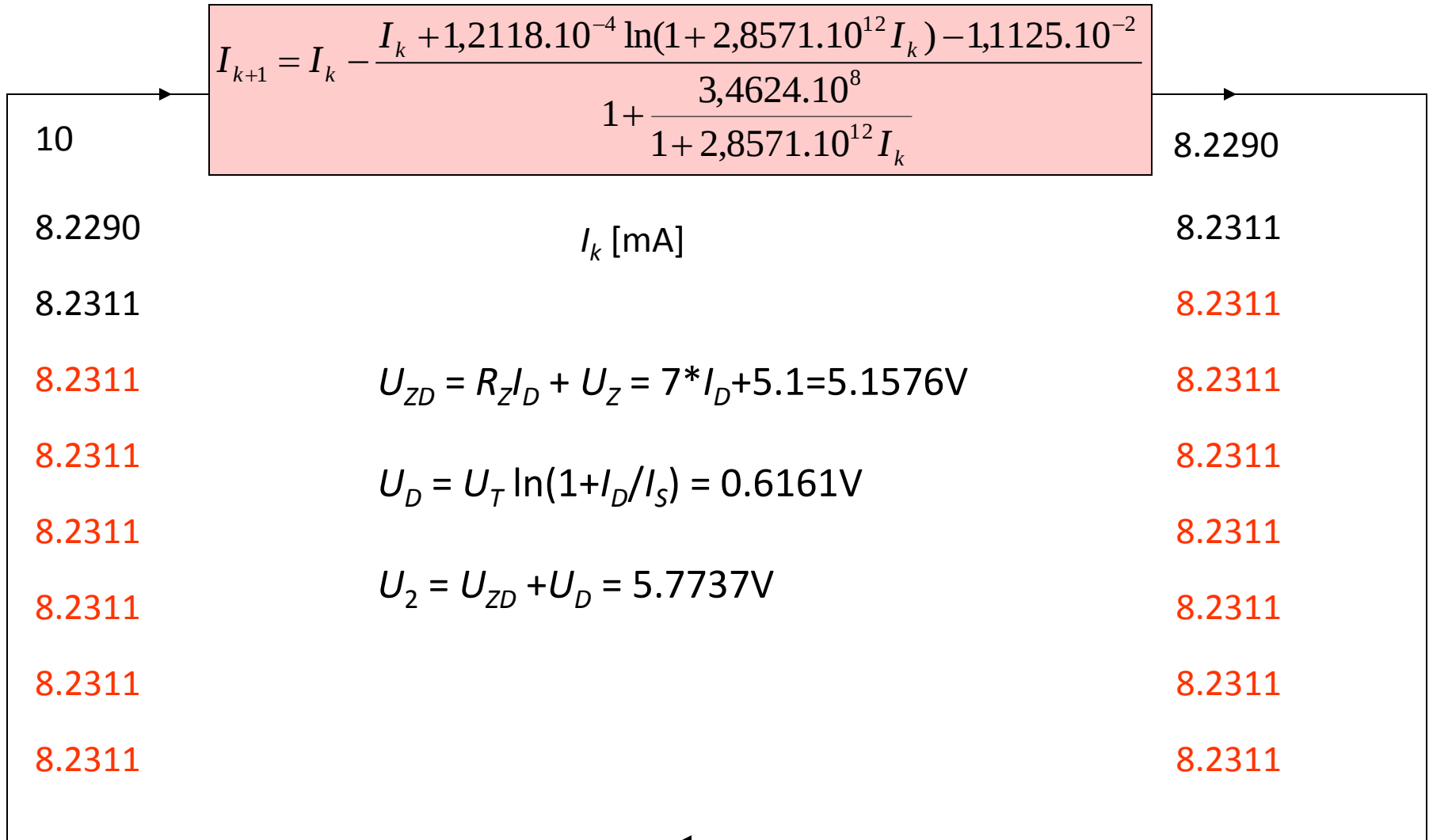
$$I_{k+1} = I_k - \frac{I_k + 1,2118 \cdot 10^{-4} \ln(1 + 2,8571 \cdot 10^{12} I_k) - 1,1125 \cdot 10^{-2}}{1 + \frac{3,4624 \cdot 10^8}{1 + 2,8571 \cdot 10^{12} I_k}}$$

0		3.2131e-8
3.2131e-8	I_k [mA]	2.8668e-6
2.8668e-6		2.4023e-4
2.4023e-4		1.9028e-2
1.9028e-5		1.2333e-3
1.2333		7.8144
7.8144		8.2309
8.2309		8.2311
8.2311		8.2311

Výpočet ss pracovního bodu jako základ všech analýz

“Ruční” řešení:

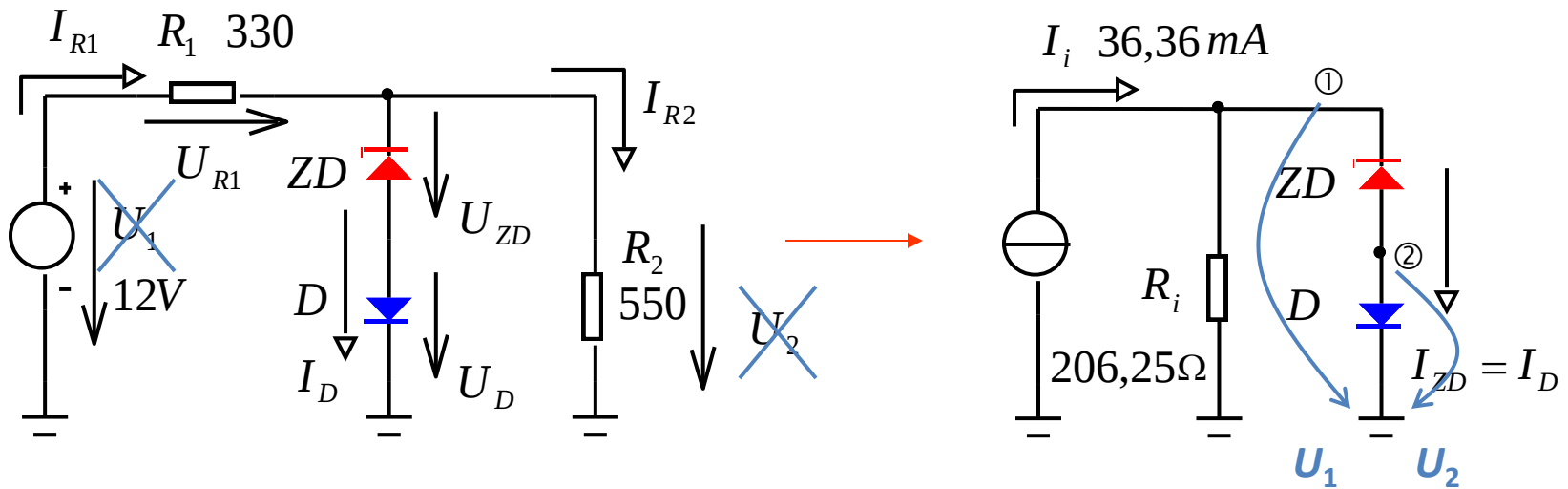
Newtonova iterační metoda



Výpočet ss pracovního bodu jako základ všech analýz

Algoritmické řešení:

Newtonova-Raphsonova iterační metoda



$$\begin{array}{l}
 \textcircled{1} \\
 \textcircled{2}
 \end{array}
 \begin{array}{|c|}
 \hline
 f_1(U_1, U_2) \\
 \hline
 f_2(U_1, U_2) \\
 \hline
 \end{array}
 =
 \begin{array}{|c|}
 \hline
 G_i U_1 + I_{ZD} - I_i \\
 \hline
 I_D - I_{ZD} \\
 \hline
 \end{array}
 =
 \begin{array}{|c|}
 \hline
 0 \\
 \hline
 0 \\
 \hline
 \end{array}$$

vektor $\mathbf{f}$

$$I_D = I_S \left(e^{\frac{U_2}{U_T}} - 1 \right)$$

$$I_{ZD} = G_Z (U_1 - U_2 - U_Z)$$

Výpočet ss pracovního bodu jako základ všech analýz

Algoritmické řešení:

Newtonova-Raphsonova iterační metoda

$$\begin{array}{lcl}
 \textcircled{1} & \begin{bmatrix} f_1(U_1, U_2) \end{bmatrix} & = \begin{bmatrix} G_i U_1 + I_{ZD} - I_i \end{bmatrix} = \begin{bmatrix} 0 \end{bmatrix} \\
 \textcircled{2} & \begin{bmatrix} f_2(U_1, U_2) \end{bmatrix} & = \begin{bmatrix} I_D - I_{ZD} \end{bmatrix} = \begin{bmatrix} 0 \end{bmatrix}
 \end{array}$$

vektor $\mathbf{f}$

$I_D = I_s \left(e^{\frac{U_2}{U_T}} - 1 \right)$
 $I_{ZD} = G_Z (U_1 - U_2 - U_Z)$

$$\mathbf{f} = \begin{bmatrix} G_i U_1 + G_Z (U_1 - U_2 - U_Z) - I_i \\ I_s \left(e^{\frac{U_2}{U_T}} - 1 \right) - G_Z (U_1 - U_2 - U_Z) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Výpočet ss pracovního bodu jako základ všech analýz

Algoritmické řešení:

Newtonova-Raphsonova iterační metoda

$$\begin{bmatrix} U_{1,k+1} \\ U_{2,k+1} \end{bmatrix} = \begin{bmatrix} U_{1,k} \\ U_{2,k} \end{bmatrix} - \begin{bmatrix} \frac{\partial f_{1,k}}{\partial U_1} & \frac{\partial f_{1,k}}{\partial U_2} \\ \frac{\partial f_{2,k}}{\partial U_1} & \frac{\partial f_{2,k}}{\partial U_2} \end{bmatrix}^{-1} \begin{bmatrix} f_{1,k} \\ f_{2,k} \end{bmatrix}$$

$$U_{k+1} = U_k - J^{-1} \cdot f_k$$

$$J = \begin{bmatrix} G_i + G_z & -G_z \\ -G_z & \frac{I_s}{U_T} e^{\frac{U_2}{U_T}} + G_z \end{bmatrix} = \begin{bmatrix} 0,1477 & -0,1429 \\ -0,1429 & 1,3566 \cdot 10^{-11} e^{38,76 U_2} + 0,1429 \end{bmatrix}$$

Výpočet ss pracovního bodu jako základ všech analýz

Algorithmické řešení:

Newtonova-Raphsonova iterační metoda

$$\mathbf{U}_{k+1} = \mathbf{U}_k - \mathbf{J}^{-1} \mathbf{f}_k$$

$U_1 \ U_2 \text{ [V]}$			
0	0	0.1251	-2.4820
0.1251	-2.4820	0.2482	-3.6927
0.2482	-3.6927	0.3692	-4.2529
.....			
5.7754	0.6169	5.7750	0.6165
5.7750	0.6165	5.7751	0.6166
5.7751	0.6166	5.7751	0.6165
5.7751	0.6165	5.7751	0.6165

92. iterace



Výpočet ss pracovního bodu jako základ všech analýz

Algoritmické řešení:

Newtonova-Raphsonova iterační metoda

$$\mathbf{U}_{k+1} = \mathbf{U}_k - \mathbf{J}^{-1} \mathbf{f}_k$$

$U_1 \ U_2 \text{ [V]}$			
6	0.65	5.8290	0.6494
5.8290	0.6494	5.7967	0.6363
5.7967	0.6363	5.7883	0.6299
.....			
5.7754	0.6169	5.7753	0.6165
5.7753	0.6165	5.7752	0.6166
5.7752	0.6166	5.7751	0.6165
5.7751	0.6165	5.7751	0.6165

22. iterace

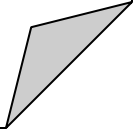


Výpočet ss pracovního bodu jako základ všech analýz

Algoritmické řešení:

Newtonova-Raphsonova iterační metoda

Algoritmus stejnosměrné analýzy nelineárních obvodů:

- 1.** Nastaví se počáteční hodnota vektoru neznámých veličin, tj. uzlových napětí.
 - 2.** Na základě Kirchhoffových rovnic pro nezávislé uzly, modelů součástek a aktuálních hodnot uzlových napětí se vyplní prvky vektoru $\mathbf{f}$ a prvky Jakobiho matice $\mathbf{J}$.
 - 3.** Pomocí rekurentní Newtonovy-Raphsonovy formule se vypočte nový odhad uzlových napětí.
 - 4.** Body 2 a 3 se opakují v cyklu tak dlouho, dokud není splněno zadané chybové kritérium.
- 

Výpočet ss pracovního bodu jako základ všech analýz

Problémy s konvergencí - úvod

Obvod může mít několik pracovních bodů. Jen některé z nich jsou stabilní (mohou se udržet).

Výsledek hledání pracovního bodu programem:

Program najde bod, který nás zajímá.

Program najde bod, o který nestojíme.

Program nenajde v daném počtu iterací žádné řešení.

Výsledek hledání pracovního bodu **můžeme ovlivnit** volbou počátečního odhadu – nulté iterace.

Výpočet ss pracovního bodu jako základ všech analýz

Problémy s konvergencí - úvod

.NODESET ... stanovení počátečního odhadu řešení některých uzlových napětí a proudů induktory

Formát:

.NODESET <V(<node> [,<node>])=<value> >*

.NODESET <I(<inductor>)=<value>>

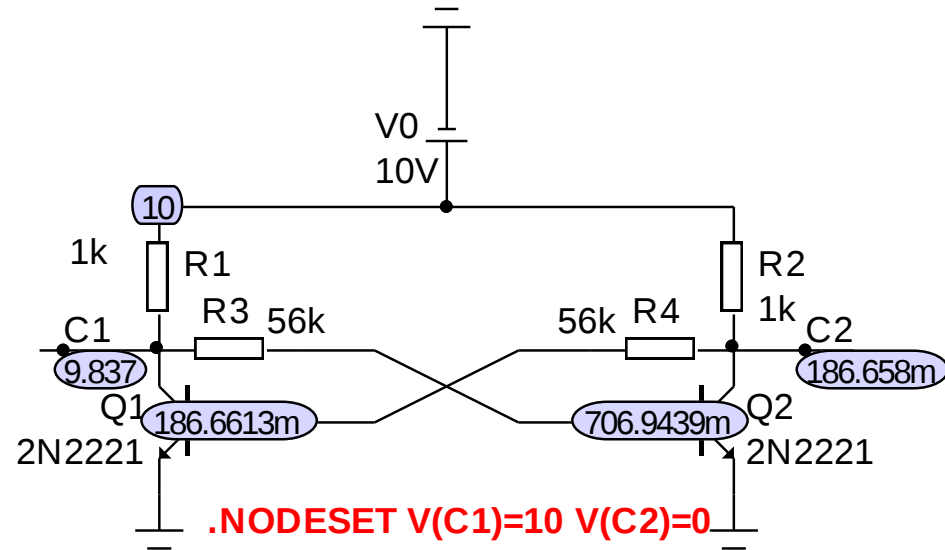
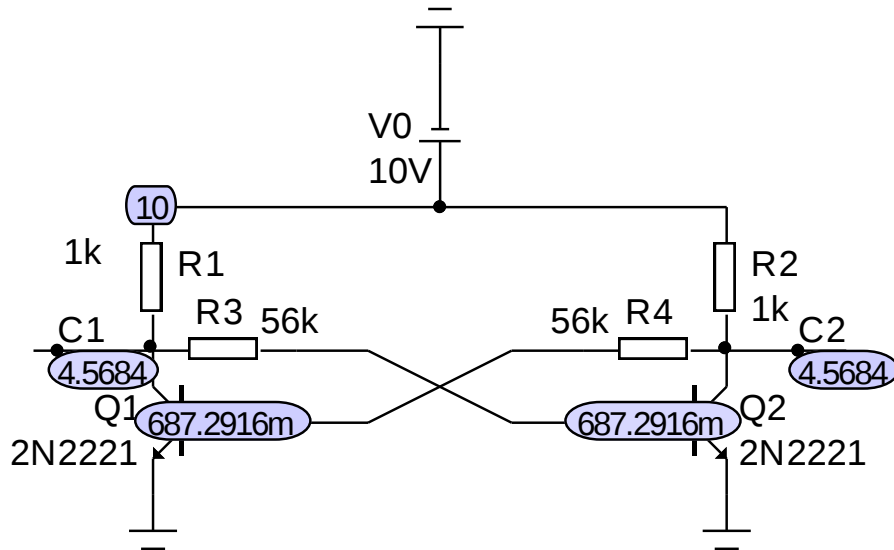
Příklad:

.NODESET V(12)=5V V(in,out)=-5 I(Lx)=12mA

Výpočet ss pracovního bodu jako základ všech analýz

Problémy s konvergencí - úvod

Příklad analýzy bistabilního klopného obvodu:



Výpočet ss pracovního bodu jako základ všech analýz

Problémy s konvergencí - úvod

Další pravidla pro používání příkazu .NODESET:

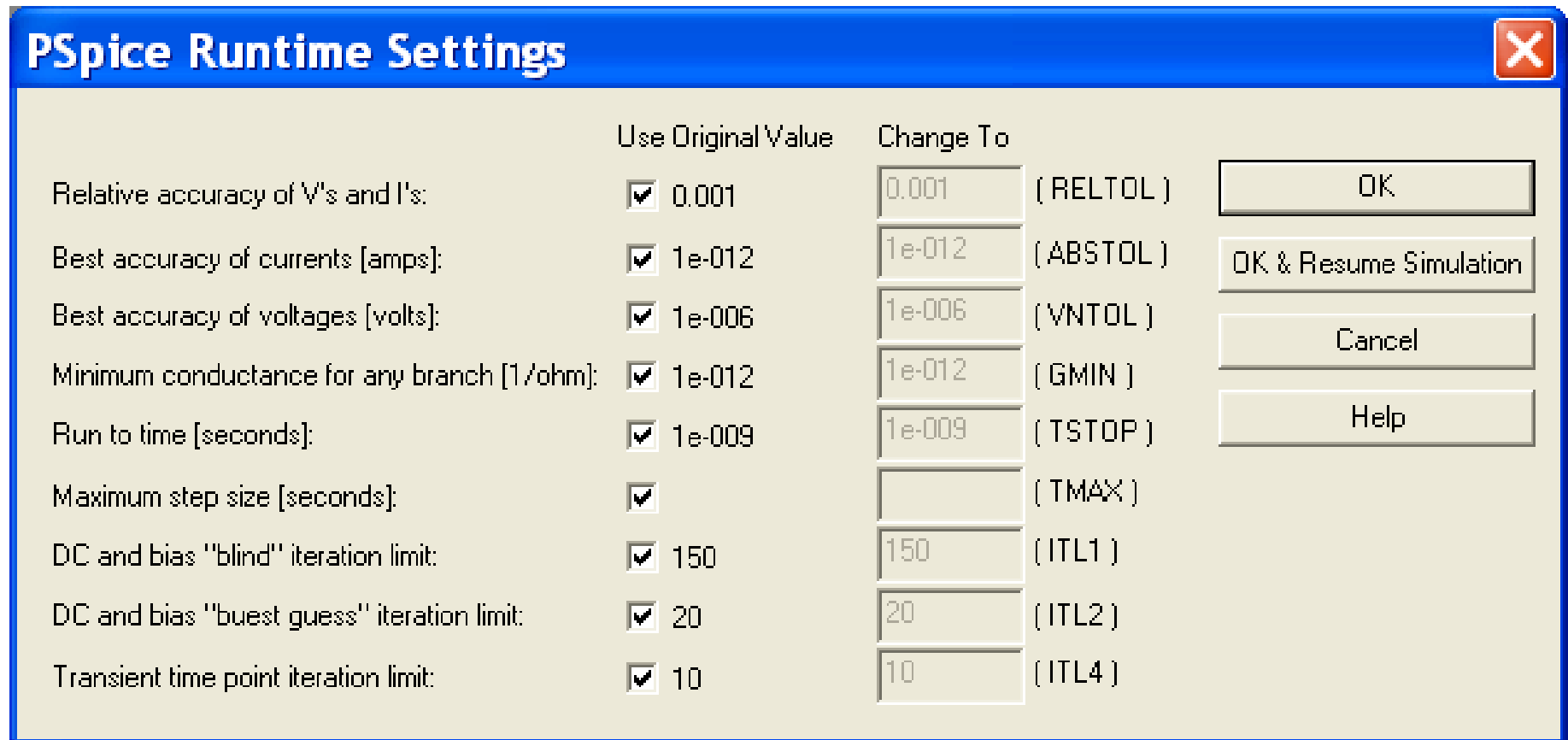
Je-li současně přítomen příkaz .IC, pak je .NODESET ignorován.

.NODESET se uplatňuje při výpočtu klidového pracovního bodu:

- pokud není uveden žádný příkaz pro analýzu, tedy pokud se počítá pouze ss pracovní bod,
- při analýze .TRANSIENT (pak se uplatní při výpočtu počátečního bodu časového průběhu),
- při analýze .AC
- při analýze .DC (pak se uplatní při výpočtu prvního bodu křivky).

Výpočet ss pracovního bodu jako základ všech analýz

Globální podmínky simulace - úvod



The image shows a screenshot of the "PSpice Runtime Settings" dialog box. The dialog has a blue title bar with the text "PSpice Runtime Settings" and a red close button in the top right corner. The main area is light beige and contains a table of settings. Each row has a description, a checkbox, a "Use Original Value" column, a "Change To" column with a text input field, and a parameter name in parentheses. To the right of the table are four buttons: "OK", "OK & Resume Simulation", "Cancel", and "Help".

		Use Original Value	Change To	
Relative accuracy of V's and I's:	☒	0.001	0.001	(RELTOL)
Best accuracy of currents [amps]:	☒	1e-012	1e-012	(ABSTOL)
Best accuracy of voltages [volts]:	☒	1e-006	1e-006	(VNTOL)
Minimum conductance for any branch [1/ohm]:	☒	1e-012	1e-012	(GMIN)
Run to time [seconds]:	☒	1e-009	1e-009	(TSTOP)
Maximum step size [seconds]:	☒			(TMAX)
DC and bias "blind" iteration limit:	☒	150	150	(ITL1)
DC and bias "buest guess" iteration limit:	☒	20	20	(ITL2)
Transient time point iteration limit:	☒	10	10	(ITL4)

Buttons: OK, OK & Resume Simulation, Cancel, Help

“Ladicí konstanty“ simulátoru, kterými ovlivňujeme výpočet pracovního bodu.

Výpočet ss pracovního bodu jako základ všech analýz

Globální podmínky simulace - úvod

Přesnost výpočtů je řízena třemi kritérii:

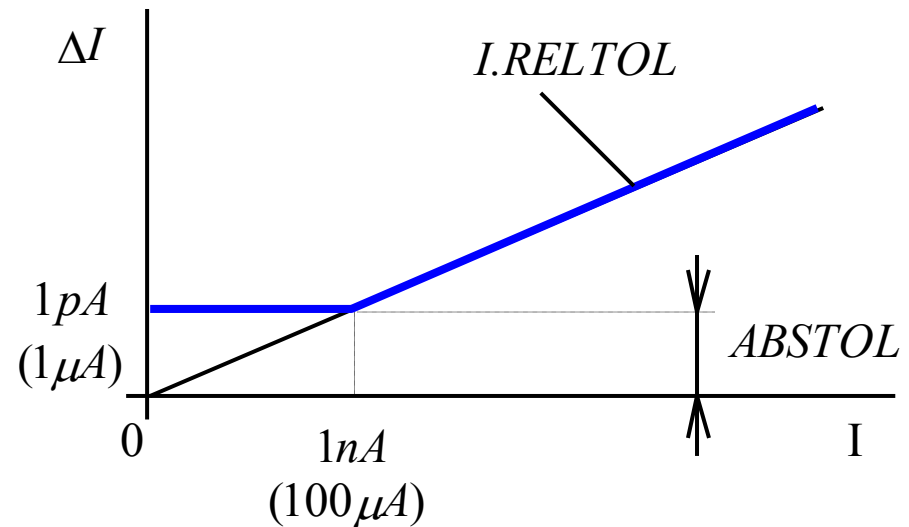
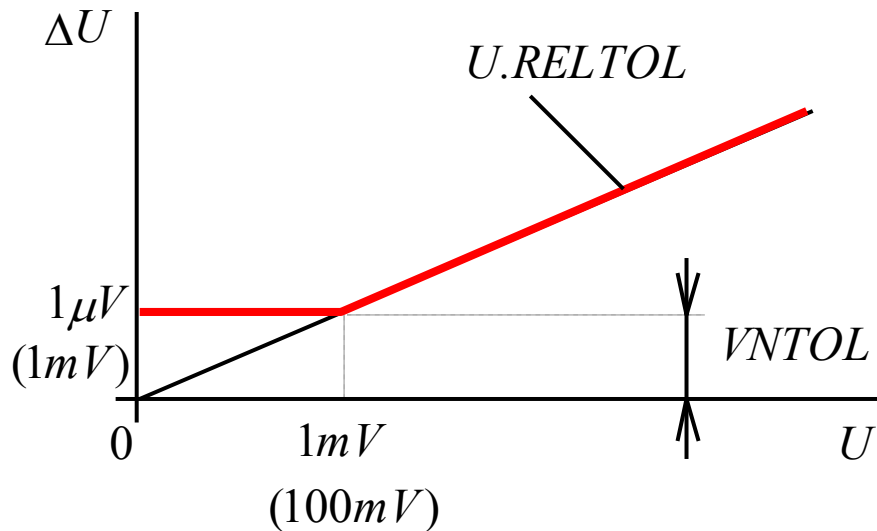
ABSTOL – Absolutní chyba ve výpočtu proudů.

VNTOL – Absolutní chyba ve výpočtu napětí.

RELTOL – Relativní chyba ve výpočtu napětí a proudů.

$$\Delta U = \text{MAX}(VNTOL, U \cdot \text{RELTOL})$$

$$\Delta I = \text{MAX}(ABSTOL, I \cdot \text{RELTOL})$$



Výpočet ss pracovního bodu jako základ všech analýz

Globální podmínky simulace - úvod

Maximální počet iterací:

ITL1 (Standard Default = 100, Power Default = 150):

Určuje maximální počet iterací při počítání stejnosměrného pracovního bodu (.OP, inicializace časové analýzy, předstupeň kmitočtové analýzy).

ITL2 (Default = 20):

Určuje maximální počet iterací při počítání stejnosměrného pracovního bodu v každém kroku analýzy „DC“.

Pokud nedojde k nalezení řešení v daném počtu iterací, je možné vyzkoušet buď zvětšení parametrů RELTOL, ABSTOL a VNTOL (zlepšení konvergence na úkor přesnosti výsledků), nebo zvětšit povolený počet iterací (pokusit se o konvergenci zvětšením počtu iterací, tj. na úkor rychlosti výpočtů).

Výpočet ss pracovního bodu jako základ všech analýz

Globální podmínky simulace - úvod

Parametr Gmin:

Jsou-li v modelech polovodičových diod použity nulové hodnoty paralelních vodivostí, pak parametr *GMIN* tyto nulové hodnoty automaticky nahradí. Zvětšování *GMIN* tedy může rovněž napomoci při problémech s konvergencí obvodů s polovodičovými strukturami.

Označování stejnosměrných napětí a proudů v PSpice

V(číslo uzlu)

Příklad: V(18)

napětí uzlu oproti referenčnímu uzlu

napětí mezi uzlem 18 a uzlem 0

V([název uzlu])

Příklad: V([vstup])

napětí uzlu, nazvaného jiným řetězcem než číslem, oproti referenčnímu uzlu

napětí mezi uzlem „vstup“ a referenčním uzlem

V(číslo uzlu1, číslo uzlu2)

Příklad: V(3,11)

napětí mezi uzlem1 a uzlem2, stejné jako V(číslo uzlu1)-V(číslo uzlu2)

napětí mezi uzlem č. 3 a uzlem č. 11, tj. V(3)-V(11)

V(jméno dvojpólu)

Příklad: V(Rx)

napětí mezi svorkama 1 a 2 dvojpólu; svorka 1 je první svorka, uvedená v deklaraci dvojpólu

napětí na rezistoru Rx

V<jméno svorky>(jméno vícepólu)

Příklad: Vb(Q1)

napětí mezi svorkou vícepólu a ref. uzlem

napětí mezi bází tranzistoru Q1 a ref. uzlem

V<jméno svorky1><jméno svorky2> (jméno vícepólu)

Příklad: Veb(Q1)

napětí mezi svorkami 1 a 2 vícepólu

napětí mezi emitorem a bází tranzistoru Q1

Označování stejnosměrných napětí a proudů v PSpice

I(jméno dvojpolu)

proud tekoucí vnitřkem dvojpolu od svorky 1 do svorky 2; svorka 1 je první svorka, uvedená v deklaraci dvojpolu
proud diodou D2

Příklad: I(D2)

I<jméno svorky>(jméno vícepolu)

proud vtékající do svorky vícepolu

Příklad: Ib(Qx)

proud tekoucí do báze tranzistoru Qx

Označování stejnosměrných napětí a proudů v PROBE

Při volání .PROBE můžeme uvést napětí a proudy v konvencích PSpice. Automaticky bude provedena konverze do formátu PROBE. Při volání „prázdného“ příkazu .PROBE je třeba zadávat napětí a proudy podle konvencí PROBE.

Zadávání v PROBE = zadávání v PSpice s těmito **výjimkami**:

V(název uzlu) napětí uzlu, nazvaného jiným řetězcem než číslem, oproti ref. uzlu; **oproti PSpice se nepoužívají závorky []**

Příklad: V(vstup) napětí mezi uzlem „vstup“ a referenčním uzlem

Nejsou podporovány zápisy

V(jméno dvojpólu)

V<jméno svorky>(jméno vícepólu)

V<jméno svorky1><jméno svorky2> (jméno vícepólu)

(s výjimkou, že jsou volány s příkazem .PROBE, tj. že veličiny jsou vypočteny PSpicem a uloženy v souboru .dat)

Stejnoseměrná analýza=analýza stejnosměrných poměrů

- jednorázový výpočet stejnosměrného pracovního bodu (žádný příkaz pro analýzu, příp. .OP)
- **analýza DC** (příkaz .DC)
- analýza „Dynamic DC“ – dynamická, interaktivní DC analýza (umožňuje MicroCap, nikoliv SPICE)

Analýza DC

= napodobování funkce přístroje pro snímání stejnosměrných charakteristik nelineárních obvodů (AV charakteristiky diod...).

Analýza DC

Analýza DC

= napodobování funkce přístroje pro snímání stejnosměrných charakteristik nelineárních obvodů (AV charakteristiky diod...).



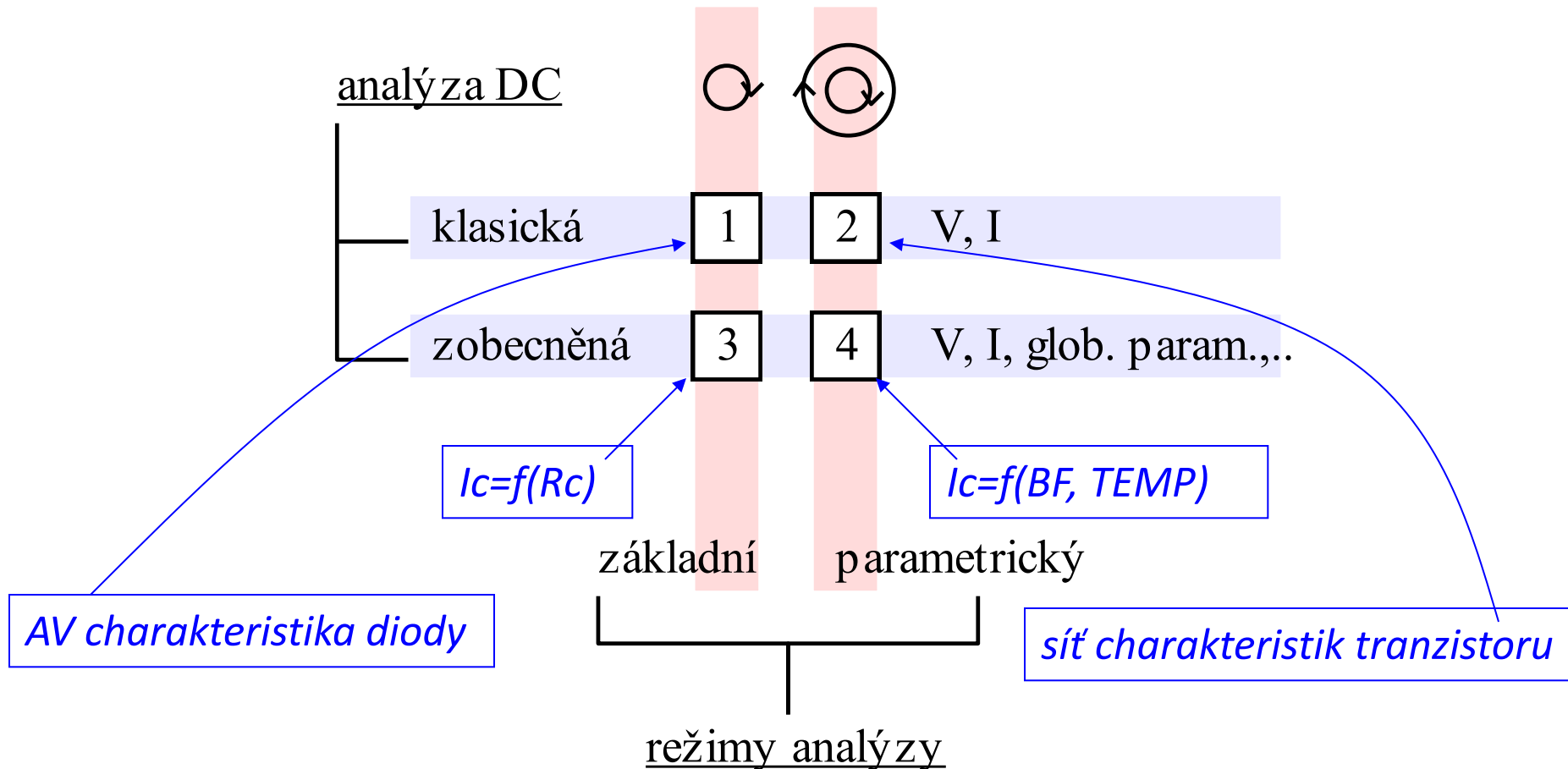
Analýza DC



Analýza DC

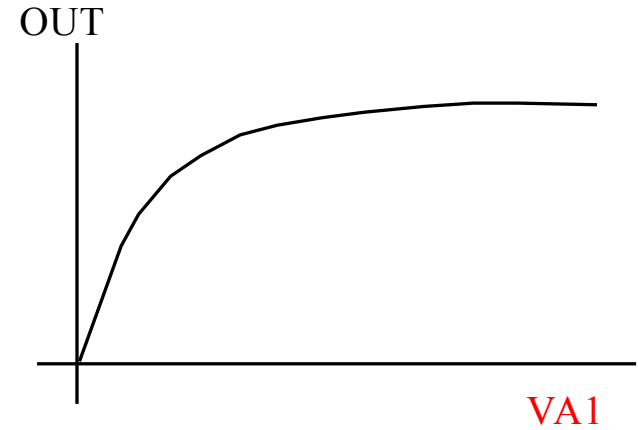
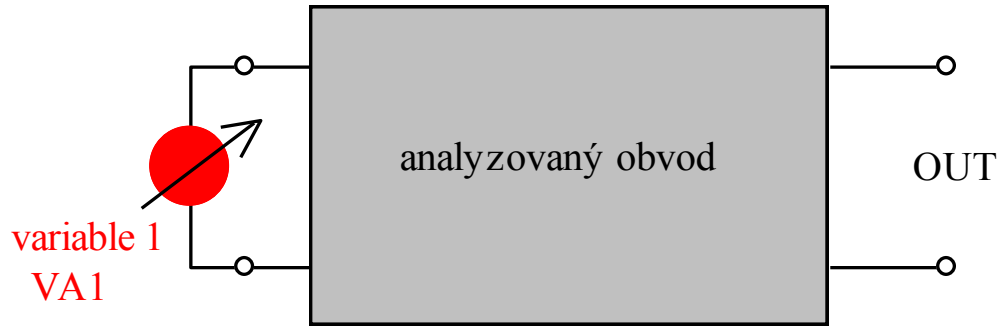
Analýza DC

= napodobování funkce přístroje pro snímání stejnosměrných charakteristik nelineárních obvodů (AV charakteristiky diod...).

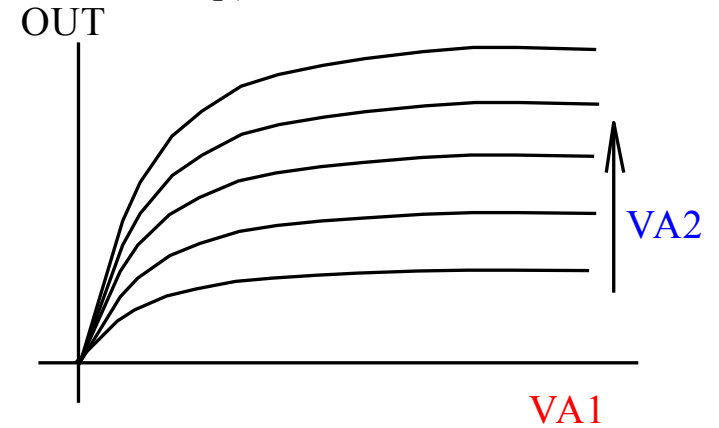
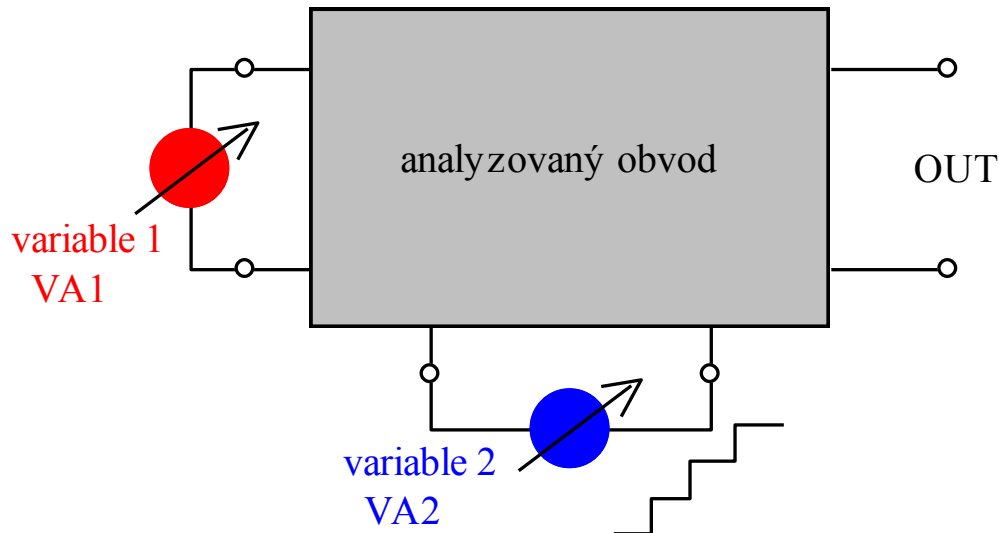


Analýza DC

základní režim (jednoduché krokování / Single sweep):

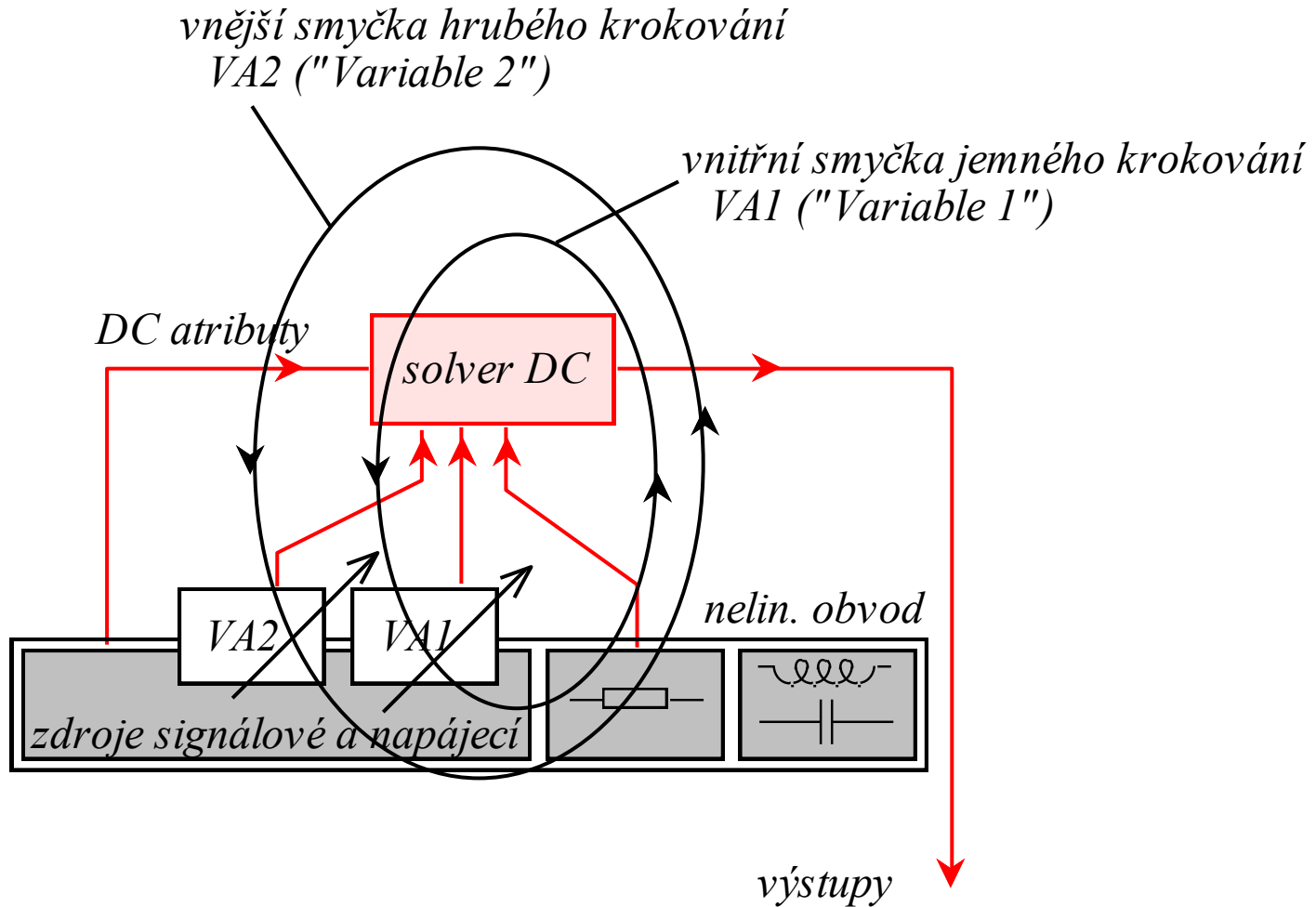


parametrický režim (vnořené krokování/ Nested sweep):



Analýza DC

Jak postupuje simulátor při DC analýze



Analýza DC

Co všechno můžeme rozmítat (povolené typy proměnných VA1 a VA2)

Při **klasické** stejnosměrné analýze:

- napětí a proudy nezávislých zdrojů v obvodu.

Při **zobecněné** stejnosměrné analýze:

- napětí a proudy nezávislých zdrojů v obvodu,
- teplota,
- libovolný parametr z modelů součástek v obvodu,
- libovolná globální proměnná.

Jak se chovají ostatní zdroje a nekrované parametry:

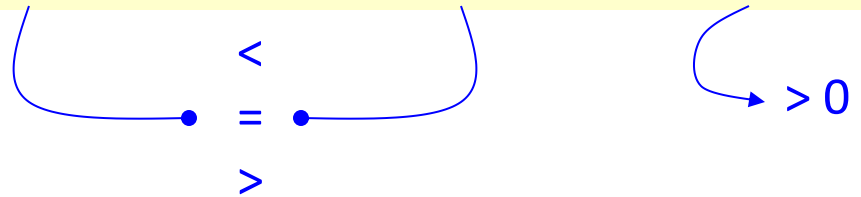
- nezávislé zdroje jsou nahrazeny zdroji o parametrech DC
- řízené zdroje pracují se stejnosměrnými vzorci
- v modelech prvků se uplatňují pouze stejnosměrné parametry.

Analýza DC

Syntaxe příkazu .DC: základní režim (jednoduché krokování)

Lineární krokování:

.DC [LIN] <jméno VA1> <počáteční hodnota> <konečná hodnota> <krok>



Příklady:

- | | |
|---------------------------|--|
| .DC LIN Vbat 0V 15V 1V | ; rozmítání napětí zdroje Vbat od 0V do 15V po 1V |
| .DC Vbat 0V 15V 1V | ; stejné jako v předchozím případě |
| .DC Idc 100mA -100mA 1mA | ; zmenšování proudu zdroje Idc od 100mA do -100mA s krokem 1mA |
| .DC NPN Qt(BF) 100 500 10 | ; rozmítání parametru BF všech tranzistorů, které mají model Qt, od 100 do 500 s krokem 10 |
| .DC PARAM Rx 1k 10k 1k | ; rozmítání globálního parametru Rx od 1k do 10k po 1k |

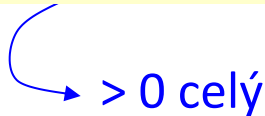
Analýza DC

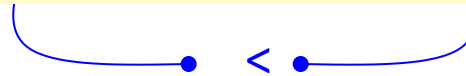
Syntaxe příkazu .DC: základní režim (jednoduché krokování)

Logaritmické krokování:

**.DC DEC <jméno VA1> <počáteční hodnota> <konečná hodnota>
+<počet kroků na dekádu>**

**.DC OCT <jméno VA1> <počáteční hodnota> <konečná hodnota>
+<počet kroků na oktávu>**

 > 0 celý

 <

Příklady:

.DC DEC Vx 1mV 1V 1

; rozmítání napětí zdroje Vx v hodnotách 1mV, 10mV, 100mV, 1V

.DC OCT PARAM Cx 1n 16n 1

; rozmítání globálního parametru Cx v hodnotách 1nF, 2nF, 4nF, 8nF a 16nF

Analýza DC

Syntaxe příkazu .DC: základní režim (jednoduché krokování)

Krokování „LIST“:

.DC <jméno VA1> LIST <výčet hodnot>

Příklady:

.DC Vbat LIST 5V 15V

; provede se výpočet pro dvě napětí 5V a 15V

.DC PARAM beta LIST 100

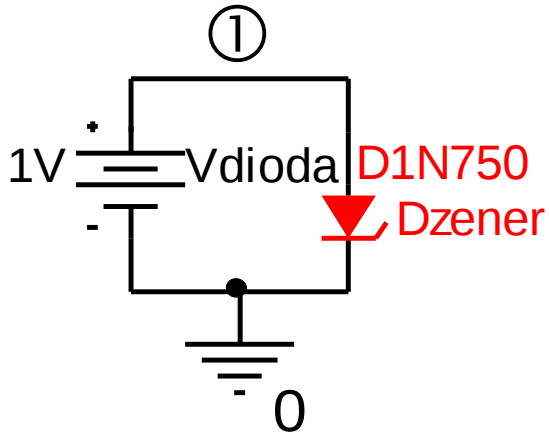
; provede se výpočet pro jedinou hodnotu
globálního parametru beta = 100

.DC PARAM beta 100 100 1

; stejný efekt jako v předchozím případě (jde o
lineární krokování, kde konečná hodnota je
rovna počáteční, na velikosti kroku nezáleží)

Analýza DC

Ukázky simulací:



Zenerka

*

Dzener 1 0 D1N750

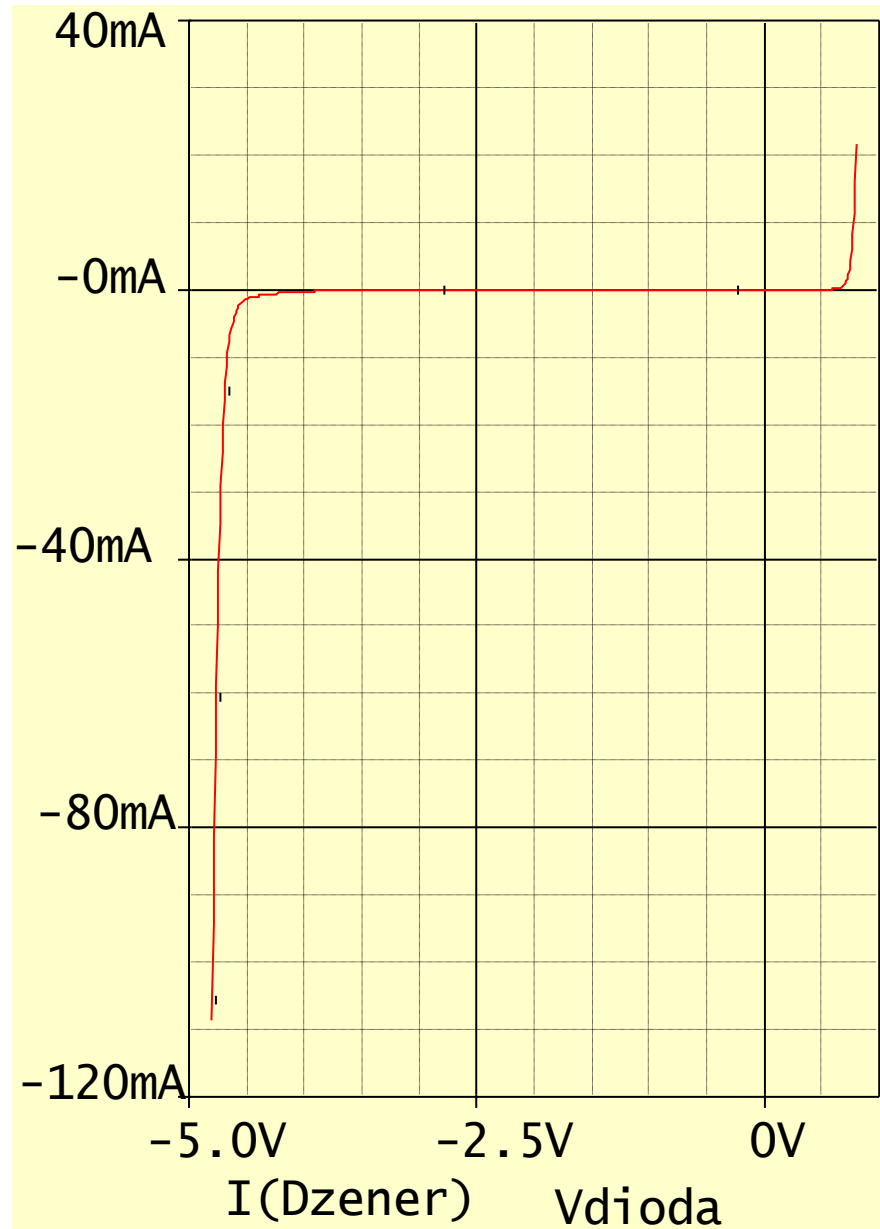
Vdioda 1 0 1V

.DC Vdioda -4.8 0.8 0.01

.probe

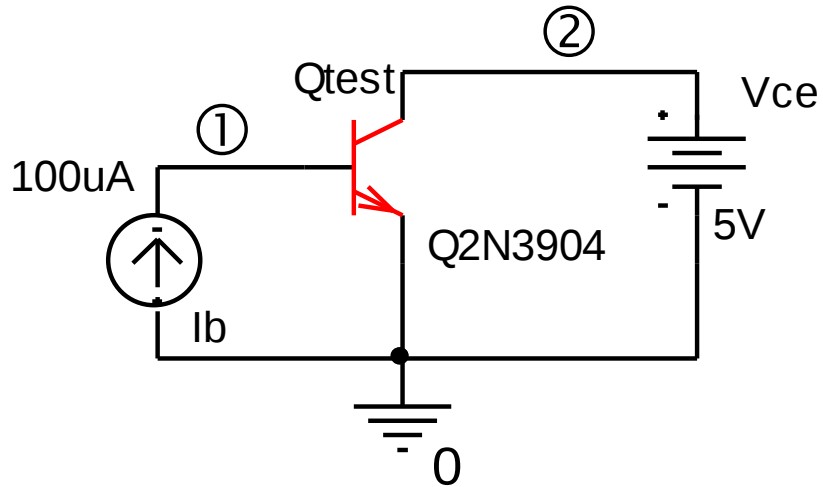
.lib

.END



Analýza DC

Ukázky simulací:



AV char

Qtest 2 1 0 Q2N3904

Vce 2 0 5V

Ib 0 1 100uA

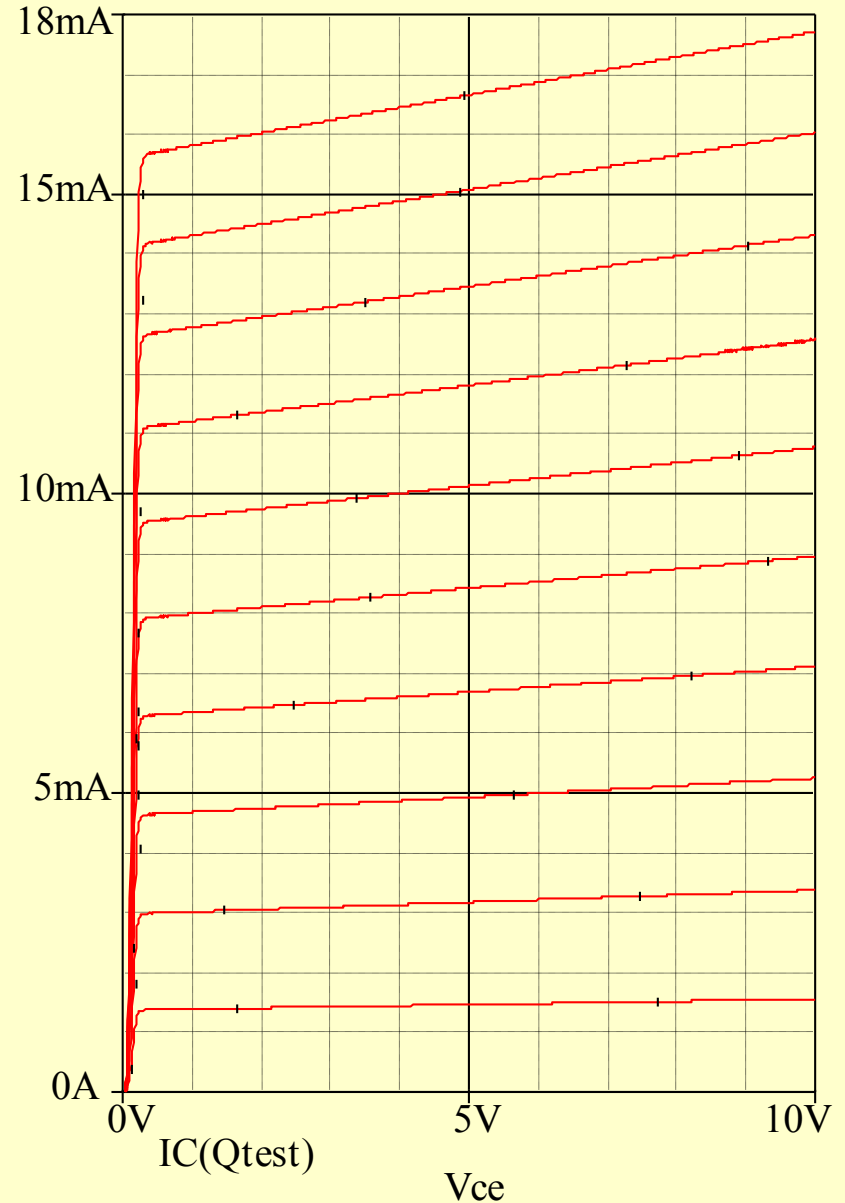
.DC Vce 0V 10V 0.01V

+Ib 0A 100uA 10uA

.probe

.lib

.END



Analýza DC

Příkaz .PRINT – vypisování výsledků analýzy do výstupního souboru

Tento příkaz funguje pouze ve spolupráci s příkazy pro základní analýzy, např. .DC.

.PRINT <označení analýzy> <výčet napětí a proudů>

Příklady:

.DC Vbat 15 15 1 ;

DC analýza pro jedinou hodnotu
rozmítaného zdroje Vbat

.PRINT V(R1) I(Rx) V([uzel])

; do vstupního souboru se zapíše 1 řádek s
hodnotami napětí na R1, proudu přes Rx, a
napětí mezi uzlem o názvu uzel a ref. uzlem

.DC Vbat 5 15 1 ;

DC analýza pro 11 hodnot Vbat od 5V do
15V po 1V

.PRINT V(R1) I(Rx) V([uzel])

; do vstupního souboru se zapíše 11 řádků s
hodnotami napětí na R1, proudu Rx, a napětí
mezi uzlem o názvu uzel a ref. uzlem;
každému řádku budou odpovídat hodnoty,
vypočtené pro jednu z krokovaných hodnot
Vbat

Analýza DC

Příkaz .PRINT – vypisování výsledků analýzy do výstupního souboru

Počet míst vypisovaných číselných údajů je určován položkou **NUMDGT** v globálních podmínkách simulace. NUMDGT = počet míst mantisy za desetinnou tečkou plus 1 (zjednodušeně). Standardně je NUMDGT=4. Chceme-li např. výpis na 5 desetinných míst, umístíme do vstupního souboru příkaz

.OPTIONS NUMDGT=6

Analýza Transient

Analýza Transient = časová analýza, analýza časových průběhů

= napodobování funkce „intelligentního osciloskopu“

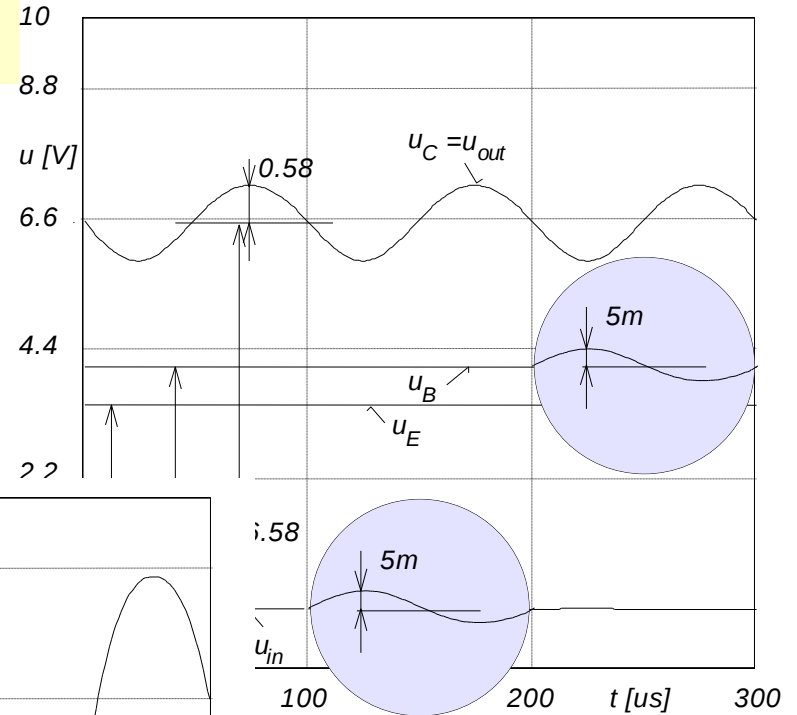
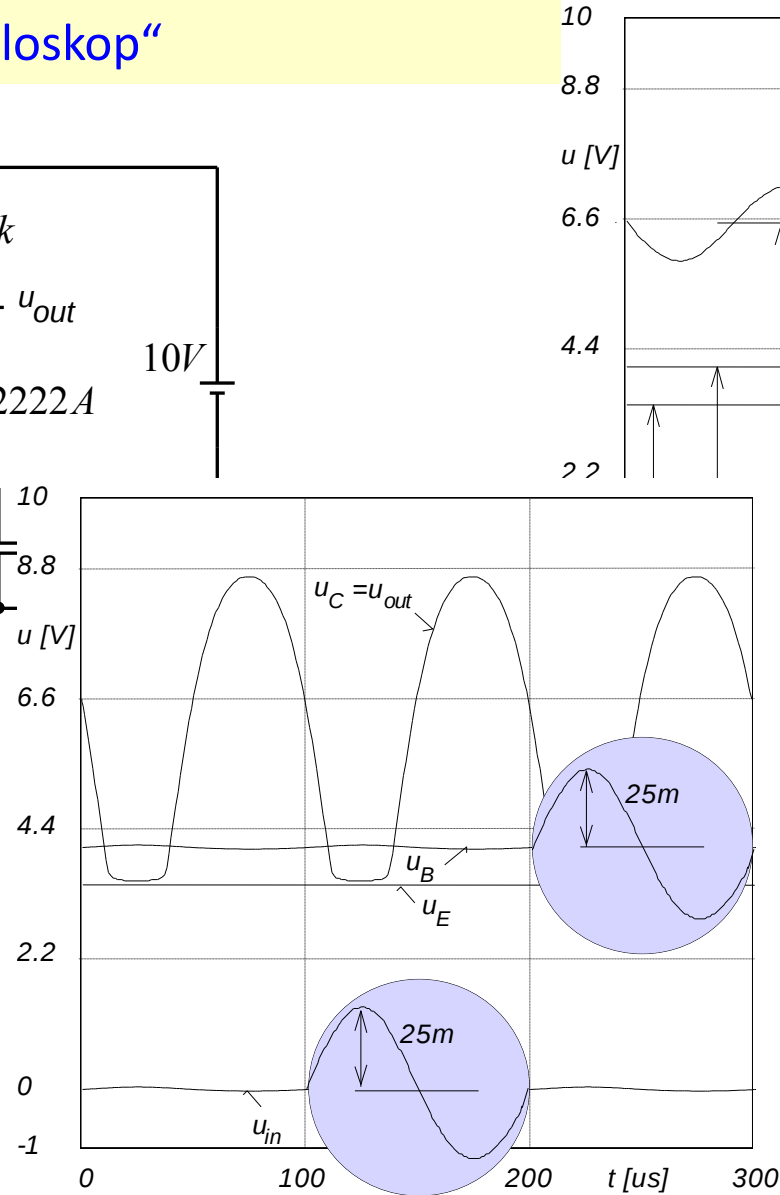
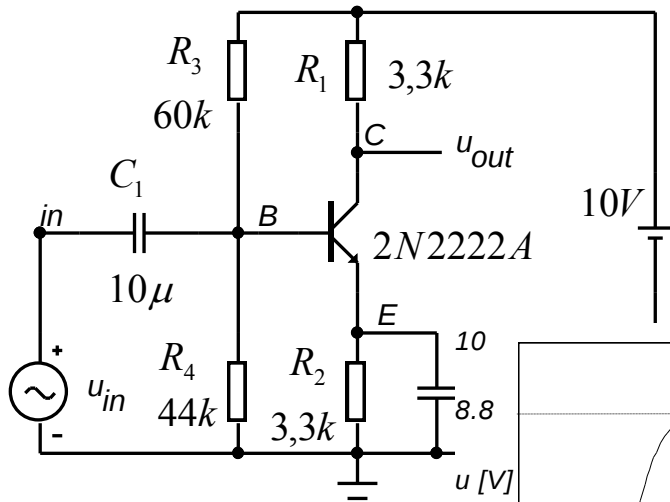
= různé způsoby dalšího zpracování analyzovaných signálů (zejména FFT).

„Intelligentní“ osciloskop:

- časové průběhy napětí, proudu a všeho, co lze popsat vzorci, ve všech možných stavech obvodu (přechodové, ustálené, jednorázové děje..),
- sledování vlivu teploty a dalších parametrů na časové průběhy

Analýza Transient

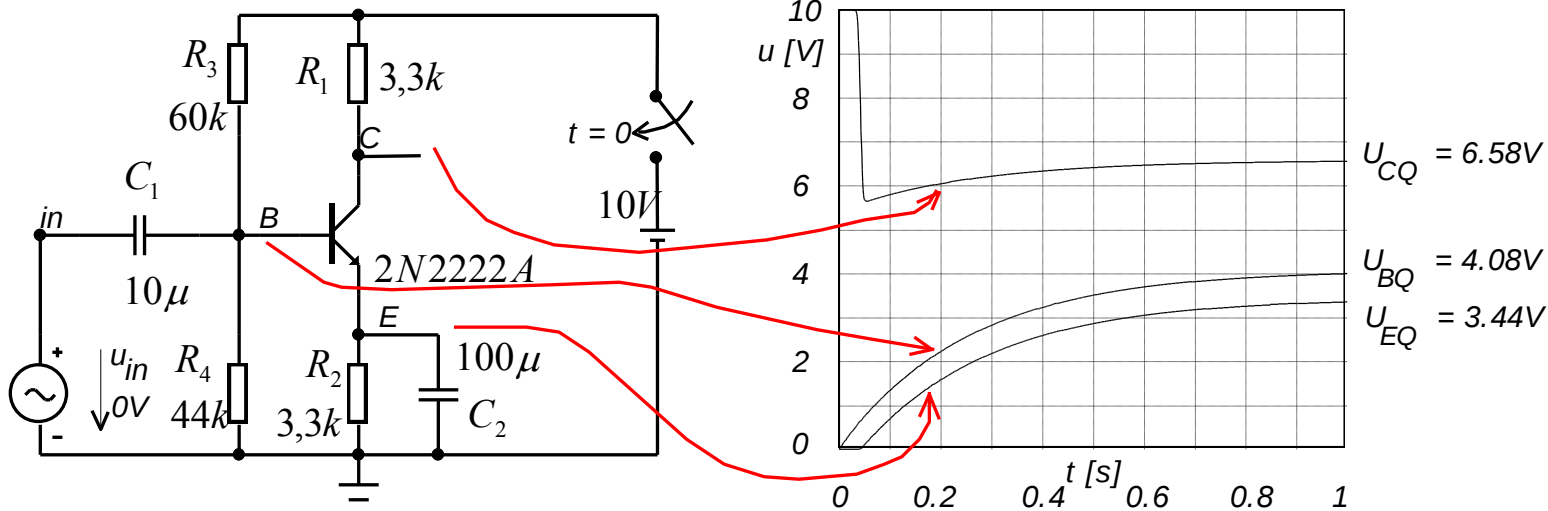
„Intelligentní osciloskop“



1. Vypočte se ss prac.bod.
2. Spustí se časová analýza při působení střídavého vstupního napětí.

Analýza Transient

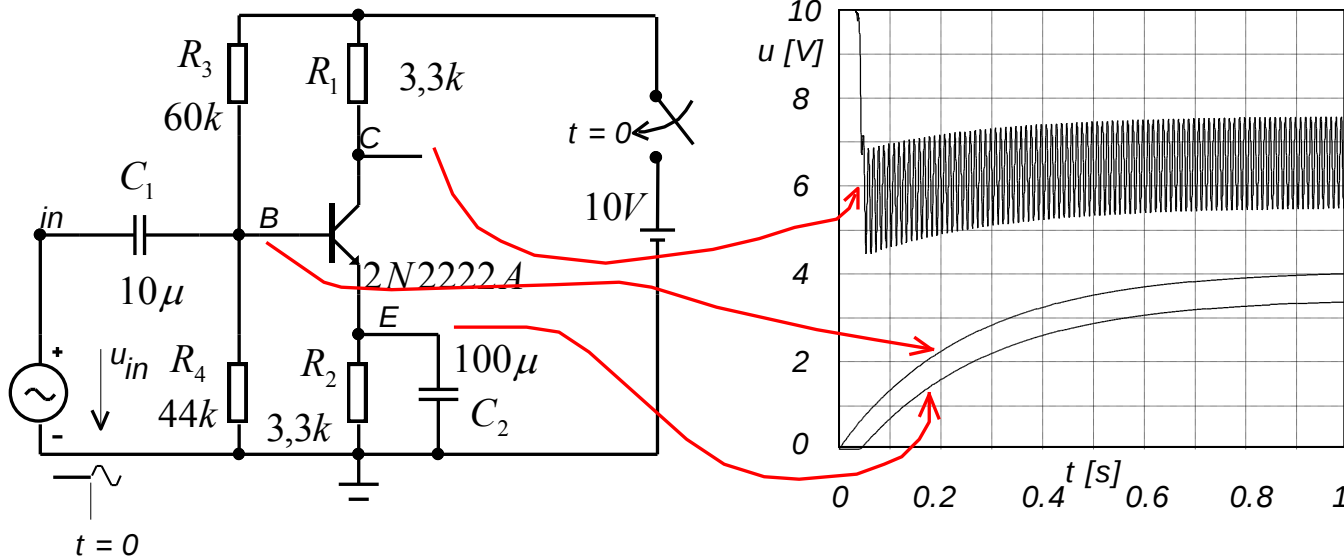
„Intelligentní osciloskop“



1. Potlačí se výpočet ss pracovního bodu,
2. Spustí se časová analýza při neuvažování střídavého vstupního napětí.

Analýza Transient

„Intelligentní osciloskop“



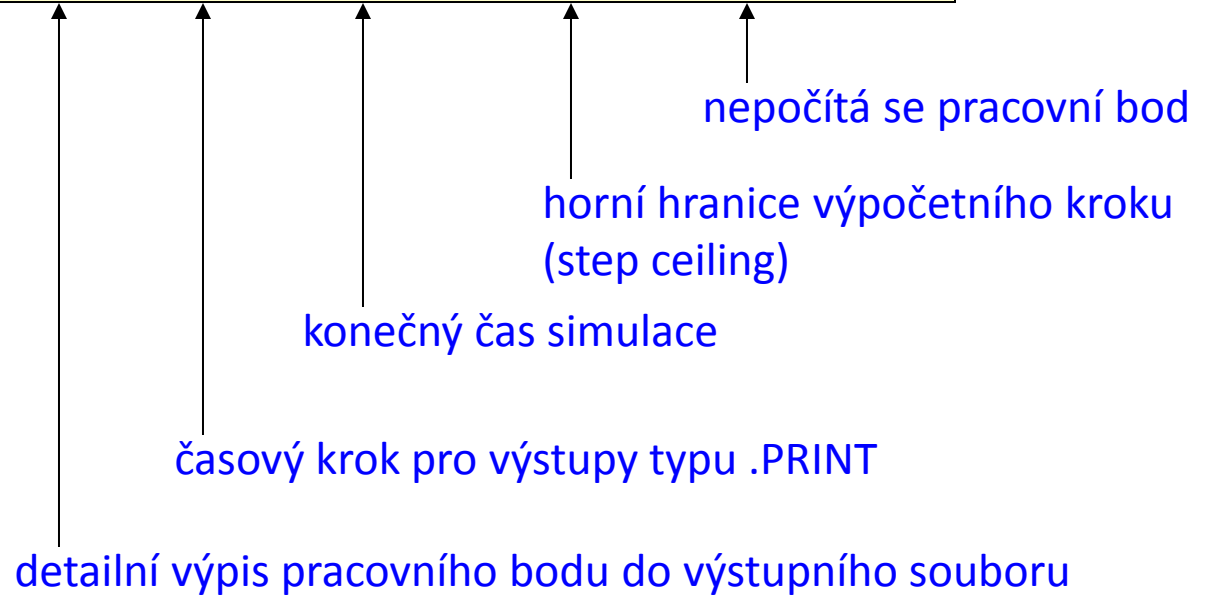
1. Potlačí se výpočet ss pracovního bodu,
2. Spustí se časová analýza při uvažování střídavého vstupního napětí.

Analýza Transient

Syntaxe časové analýzy (zjednodušená)

alternativa: UIC

.TRAN [/OP] <Tstep> <Tstop> [max. krok][SKIPBP]



Příklady:

.TRAN 10u 1m 25u ; analýza od času 0 do 1ms, max. krok 25us; napřed se vypočte prac. bod

.TRAN 10u 1m SKIPBP; analýza do 1ms, max. krok $1\text{m}/50=20\text{us}$, prac.bod se nepočítá

Analýza Transient

Důležitá poznámka:

Pracovní bod počítaný v analýze Transient se obecně liší od pracovního bodu počítaného při stejnosměrné analýze (příkazy .OP, .DC, nebo bez příkazu):

- Klasický pracovní bod se počítá **z atributů DC** zdrojů
- Pracovní bod se při analýze Transient počítá **z počátečních hodnot zdrojů v čase 0**.

Analýza Transient

Na čem závisí signály generované obvodem:

- na počátečních podmínkách v obvodu
(na počátečních napětích v uzlech a proudech induktory)
- na tom, zda je nastaven pracovní bod nebo ne
- na vstupních signálech
- na samotném obvodu

Analýza Transient

Zadávání počátečních podmínek – IC (Initial Conditions)

- v rámci definice prvků typu C a L
- příkazem .IC
- čtením počátečních podmínek ze souboru

Analýza Transient

Zadávání počátečních podmínek – IC (Initial Conditions)

Rozšířené zadávání prvků C a L:

C<jméno> <+ svorka> <- svorka> [IC=<počáteční hodnota napětí>]
L<jméno> <+ svorka> <- svorka> [IC=<počáteční hodnota proudu>]

Počáteční napětí na C je orientováno od svorky + ke svorce -.

Počáteční proud induktorem teče od svorky + do svorky – vnitřkem induktoru.

Chybí-li specifikace IC, platí implicitní nulové hodnoty.

Příklady:

Celektrolyt 6 2 IC=15V

;počáteční napětí na kapacitou je 15V od uzlu 6 do uzlu 2

Ltlumivka 44 21 IC=2.4A

;počáteční proud 2.4A teče induktorem od uzlu 44 do uzlu 21

Cx vstup baze 10uF

;kapacitor je na počátku časové analýzy vybitý

Analýza Transient

Zadávání počátečních podmínek – IC (Initial Conditions)

Zadávání počátečních podmínek příkazem .IC:

```
.IC <V(uzel1 [,uzel2])=<počáteční napětí>  
.IC <I(inductor)= <počáteční proud>
```

Příkazy se dají sdružovat do jednoho řádku.

Příklad:

```
.IC V(5)=4V V(12,3)=3V I(Lx)=-25mA
```

Analýza Transient

Zadávání počátečních podmínek – IC (Initial Conditions)

Zadávání počátečních podmínek ze souboru:

```
.LOADBIAS <jméno souboru>
```

Soubor lze získat na základě předchozí analýzy obvodu s použitím příkazu .SAVEBIAS.

Soubor je textový a obsahuje příkazy typu .NODESET, tj. výčet hodnot všech uzlových napětí a proudů induktorů v obvodu (tzv. stavové proměnné).

V případě nutnosti je lze editací souboru přepsat na příkazy typu .IC (viz dále).

Analýza Transient

Zadávání počátečních podmínek – IC (Initial Conditions)

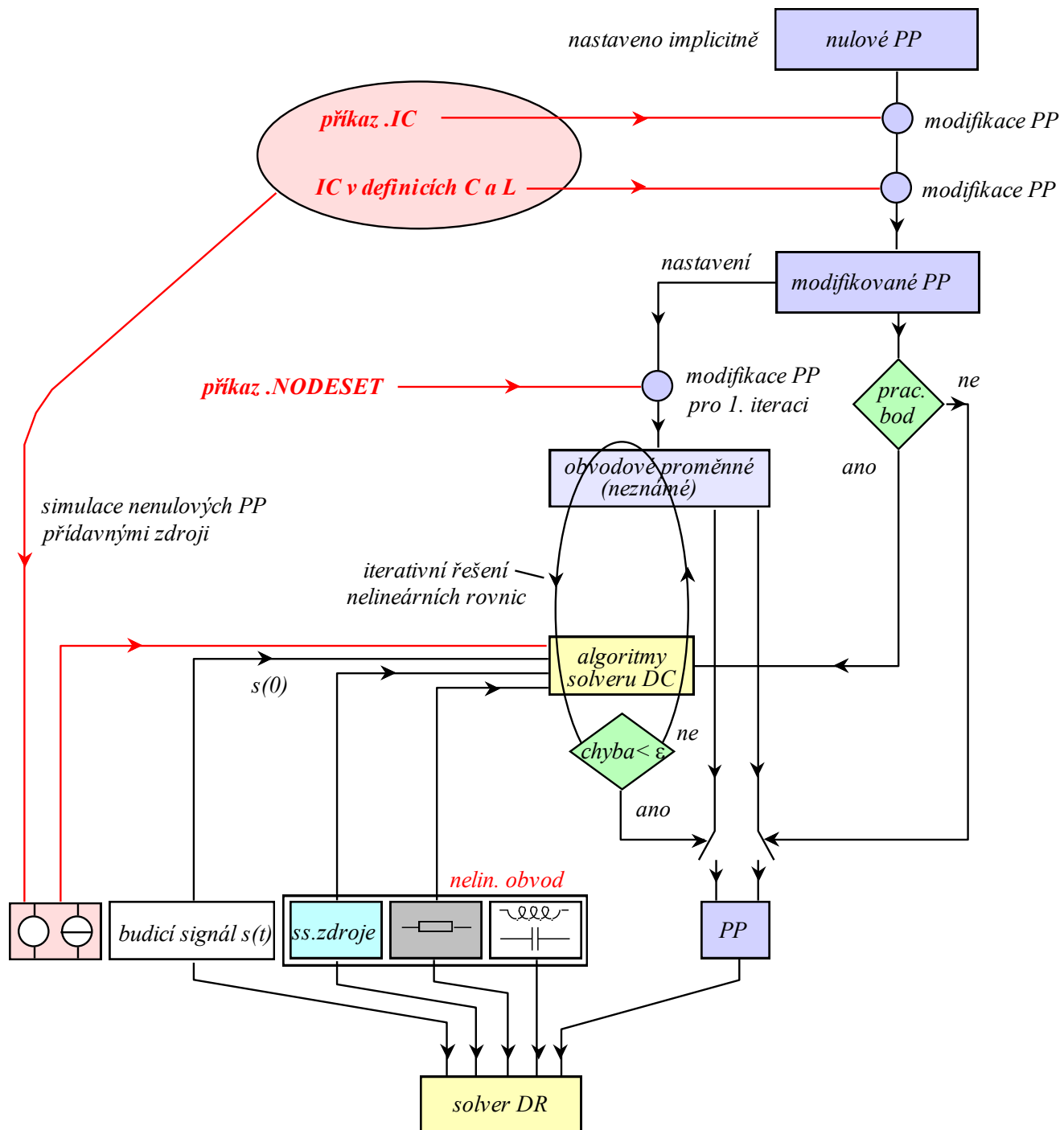
Zadávání počátečních podmínek ze souboru:

.LOADBIAS <jméno souboru>

Tvorba textového souboru:

syntaxe příkazu .SAVEBIAS je bohatá, v příkladech uvedeny jsou nejzákladnější varianty:

- | | |
|---|---|
| .SAVEBIAS prac_bod.txt OP | ; stavové proměnné odpovídající ss prac. bodu |
| .SAVEBIAS PP.txt TRAN | ; stavové proměnné v prvním bodě analýzy
Transient |
| .SAVEBIAS xyz.tra TRAN TIME=10ms | ; stavové proměnné v čase bezprostředně
následující za 10ms |
| .SAVEBIAS MKO.obv TRAN NOSBCKT TIME=10ms | ; stavové proměnné v čase bezprostředně
následující za 10ms; v souboru nebudou údaje o
vnitřních proměnných podobvodů |



Analýza Transient

Mechanismus působení příkazů .IC a .NODESET v analýze TRANSIENT

Rozdíly mezi .IC a .NODESET:

.IC způsobí, že při případném výpočtu pracovního bodu budou nastavené počáteční hodnoty napětí a proudů „drženy“ až do konce výpočtu. Ostatní „volné“ veličiny se dopočtou. Tedy – časová analýza začne s počátečními hodnotami, uvedenými v příkazu .IC.

.NODESET je jen iterační násada pro výpočet pracovního bodu. Na konci výpočtu mohou být úplně jiné hodnoty.

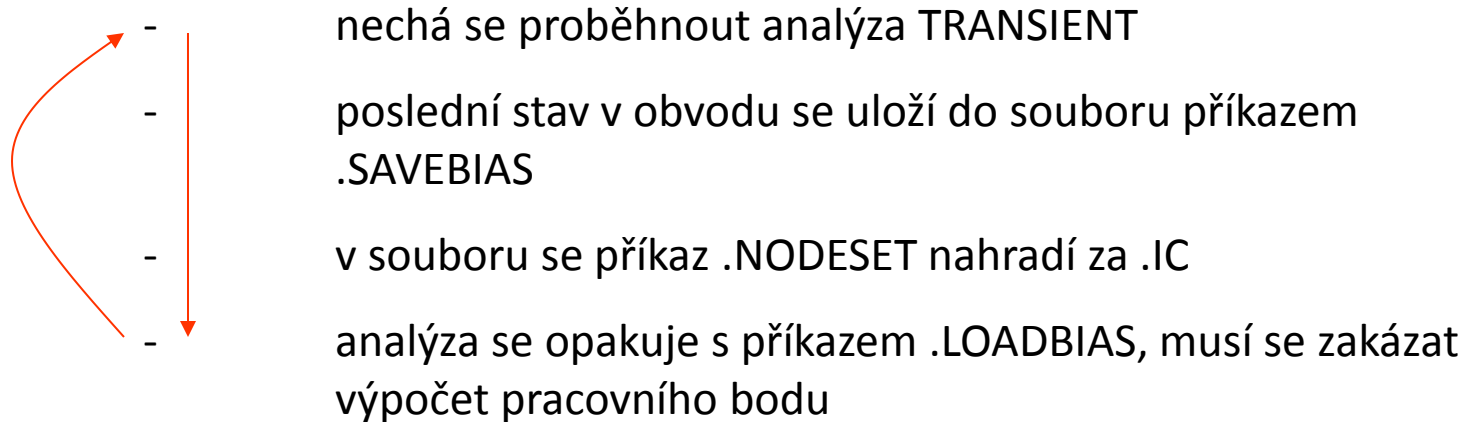
Příkaz .IC má přednost před .NODESET, pokud jsou použity současně.

IC v definici C a L má přednost před příkazem .IC.

Analýza Transient

Mechanismus působení příkazů .IC a .NODESET v analýze TRANSIENT

Metoda hledání ustálených stavů v PSPICE:



Pozn.: V Micro-Capu se ustálené stavy hledají elegantně v režimu LEAVE.

Analýza Transient

Kombinace (ne)povolení prac. bodu a volby počátečních podmínek

PP	výpočet pracovního bodu	
	povolit	zakázat
nulové	<i>I.</i> - Obvod je již připojen k napájecím zdrojům, řešíme jeho reakci na vstupní signály (nejčastější typ úlohy).	<i>IV.</i> - Rozběh oscilátorů a generátorů kmitů. - Analýza pasivních obvodů s nulovými počátečními stavy. - Hledání stejnosměrných pracovních bodů u obvodů, kde klasický výpočet selhává.
modifikované pomocí .IC, IC	<i>II.</i> - Řešení přechodných dějů se známými PP v obvodech, které obsahují např. IO s nastaveným ss pracovním bodem.	<i>V.</i> - Ustálené kmity autonomních obvodů. - Start analýzy z přesně definovaných stavů obvodu. - Řešení přechodných dějů v obvodech s nenulovými počátečními podmínkami. - Postupný přechod obvodu do ustáleného stavu sledem opakovaných analyzačních běhů.
modifikované pomocí .NODESET	<i>III.</i> - Hledání pracovních bodů, které leží blízko zadaných počátečních podmínek, a časová analýza vycházející z těchto bodů.	<i>VI.</i> - Nemá smysl.

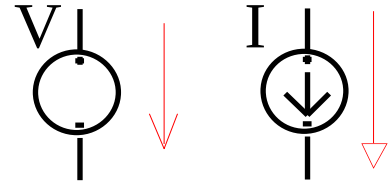
Analýza Transient

Vytváření signálů = zdrojů časových průběhů

- použitím zdrojů signálů, tj. nezávislých a závislých (řízených) zdrojů napětí a proudů
- použitím STIMULUS EDITORU

Nezávislé zdroje napětí a proudu – obecná syntaxe:

**V <jméno zdroje> <+ uzel> <- uzel> [specifikace vlastností]
I <jméno zdroje> <+ uzel> <- uzel> [specifikace vlastností]**



**[[DC] <hodnota>][AC <amplituda> [<fáze>]] [stimulus=<jméno stimulu>]
+ [<definice čas. průběhu>]**

amplituda a fáze signálu pro analýzu AC

podrobnosti viz manuál

specifikace časového průběhu jen pro analýzu Transient

V nebo I konstantního signálu nebo hodnota pro výpočet DC pracovního bodu

Analýza Transient

Vytváření signálů = zdrojů časových průběhů

Nezávislé zdroje napětí a proudu – obecná syntaxe:

**[[DC] <hodnota>][AC <amplituda> [<fáze>]] [stimulus=<jméno stimulu>]
+ [<definice čas. průběhu>]**

PULSE(<seznam parametrů>) impulzní průběhy

SIN(<seznam parametrů>) harmonický signál (zobecněný)

EXP(<seznam parametrů>) exponenciální průběhy

PWL(<seznam parametrů>) po částech lineární signál
(graf ve tvaru lomené čáry)

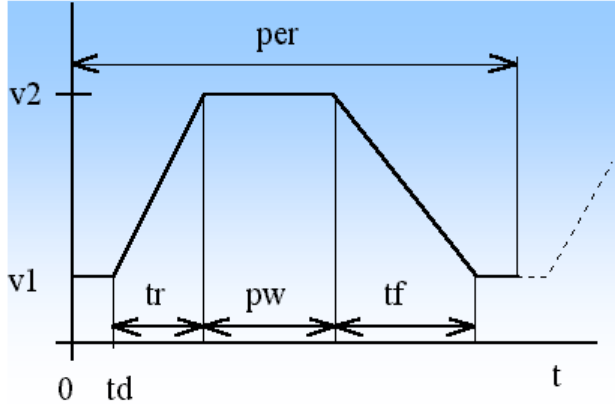
SFFM (<seznam parametrů>) kmitočtově modulovaný signál

Analýza Transient

Vytváření signálů = zdrojů časových průběhů

Zdroj impulzů

PULSE(<v1> <v2> <td> <tr> <tf> <pw> <per>)

formát	atributy				časový průběh
PULSE	název	popis	jednotka	přednast.	
	v1 (i1)	poč. hodnota	V (A)	-	
	v2 (i2)	výška impulzu	V (A)	-	
	td	zpoždění	s	0	
	tr	doba náběhu	s	Tstep	
	tf	doba doběhu	s	Tstep	
	pw	šířka impulzu	s	Tstop	
	per	perioda	s	Tstop	

Příklady:

VTTL 1 0 DC 1 **PULSE(0 5 0 1n 1n 10u 20u)**
ISAW in out **PULSE(0 1 0 1m 0.1m 0 1.1m)**

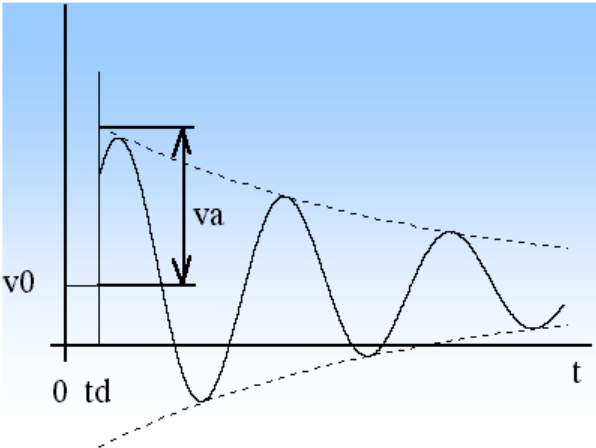
; obdélníkový průběh se střídou prakticky 1:1
; pilovitý průběh

Analýza Transient

Vytváření signálů = zdrojů časových průběhů

Zdroj zobecněného harmonického signálu

SIN(<v0> <va> <freq> <td> <df> <phase>)

formát	atributy				časový průběh
SIN	<u>název</u>	<u>popis</u>	<u>jednotka</u>	<u>přednast.</u>	
	v0 (i0)	posunutí	V (A)	-	
	va (ia)	amplituda	V (A)	-	
	freq	kmitočet	Hz	1/Tstop	
	td	zpoždění	s	0	
	df	tlumicí faktor	1/s	0	
	phase	poč. fáze	stupně	0	
	df = 1/časová konstanta tlumení, může být i záporná nulová poč. fáze znamená sinusový signál.				

Příklady:

VSINUS 1 0 **SIN(0 1)**

; sinusoida o ampl. 1V; v simulačním čase se vykreslí 1 perioda

lx 3 4 **SIN 1 1 1k 0 1k 90**

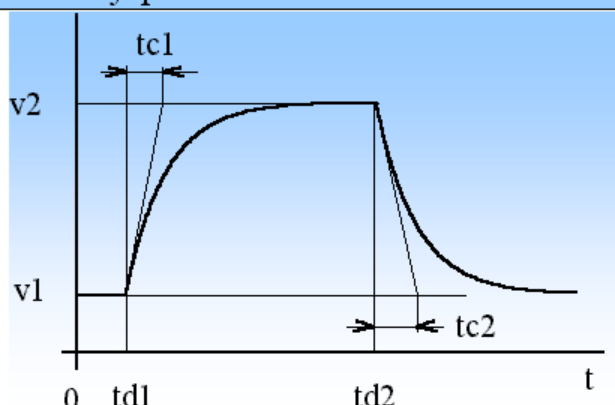
; kosinusovka se ss složkou 1V, f=1kHz a čas. konstanta 1ms

Analýza Transient

Vytváření signálů = zdrojů časových průběhů

Zdroj exponenciálních impulzů

EXP(<v1> <v2> <td1> <tc1> <td2> <tc2>)

formát	atributy				časový průběh
EXP	<u>název</u>	<u>popis</u>	<u>jednotka</u>	<u>přednast.</u>	
	v1 (i1)	poč. hodnota	V (A)	-	
	v2 (i2)	špičk. hodnota	V (A)	-	
	td1	začátek náběhu	Hz	0	
	tc1	τ náběhu	s	Tstep	
	td2	konec náběhu	s	td1+Tstep	
	tc2	τ doběhu	s	Tstep	

Příklad:

Vrc out 0 **EXP 0 1 0 1m 10m** ; simulace nabíjení kapacitou přes rezistor na napětí 1V z 0V, +s časovou konstantou 1ms; v čase 10ms se kapacitor vybije +s časovou konstantou Tstep

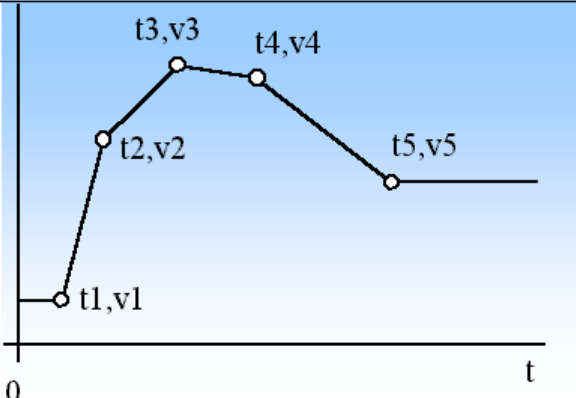
Analýza Transient

Vytváření signálů = zdrojů časových průběhů

Zdroj po úsecích lineárního signálu

PWL(t1,v1) (t2,v2)

← další možnosti viz manuál..

formát	atributy	časový průběh
PWL	Časový průběh signálu je tvořen lomenou čarou, která spojuje body o zadaných souřadnicích. Signál „vlevo“ od prvního bodu a „vpravo“ od posledního bodu na časové ose je prodloužením vertikálních souřadnic krajních bodů.	

Příklad:

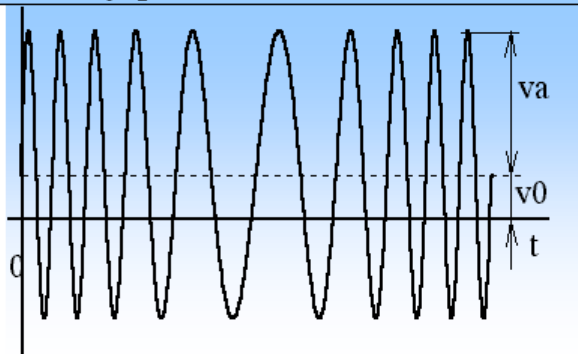
Vtroj out 0 **PWL (0,0) (1m,1) (2m,0)** ; trojúhelníkový impulz trvající 2ms

Analýza Transient

Vytváření signálů = zdrojů časových průběhů

Zdroj kmitočtově modulovaného signálu

SFFM(<v0> <va> <freq> <mi> <fm>)

formát	atributy				časový průběh
SFFM	<u>název</u>	<u>popis</u>	<u>jednotka</u>	<u>přednast.</u>	
	v0 (i0)	posunutí	V (A)	-	
	va (ia)	amplituda	V (A)	-	
	freq	kmitočet nosné	Hz	1/Tstop	
	mi	index FM= $\Delta F/f_m$	-	0	
	<u>fm</u>	<u>modulační kmit.</u>	Hz	1/Tstop	
	ΔF je kmitočtový zdvih.				

Příklad:

Vsirena 1 0 **SFFM 0 1 1k 200 1**

; napětí představující zvuk sirény o základním +kmitočtu 1kHz, který je rozmítán „nahoru“ a „dolů“ +o 200Hz periodicky jedenkrát za sekundu.

Analýza Transient

Vytváření signálů = zdrojů časových průběhů

Závislé zdroje napětí a proudu

Ke generaci signálů jsou vhodné zejména řízené zdroje, jejichž napětí, resp. proud lze **definovat vzorcem**, v němž figuruje proměnná **TIME** (simulační čas).

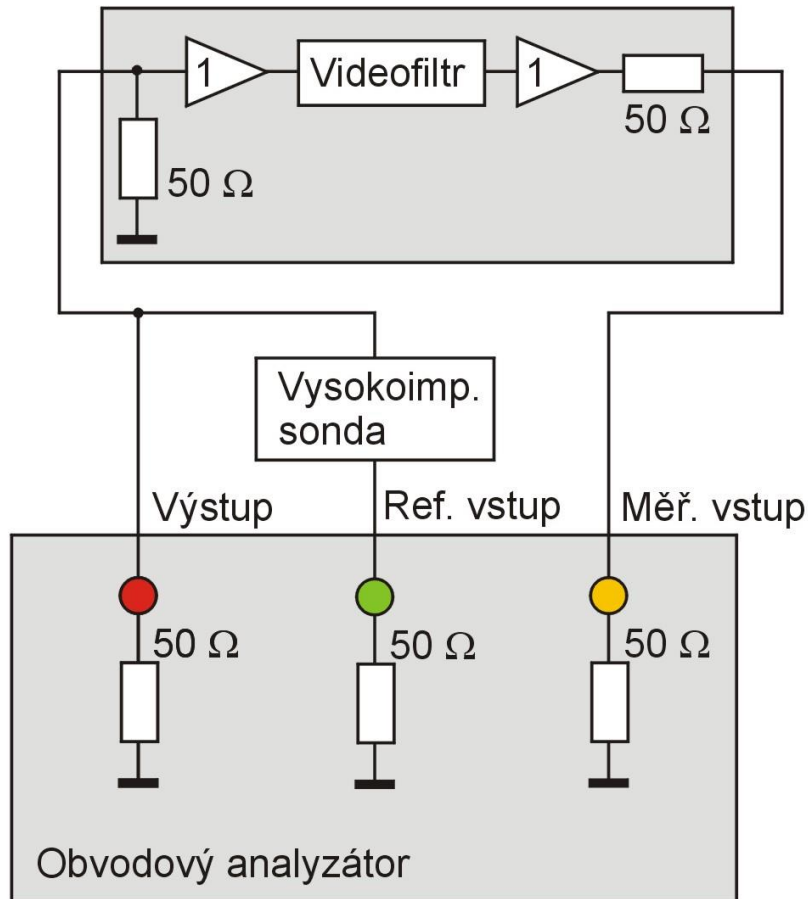
Příklad:

EGPSK 1 0 value={ $\sin(2 \cdot 3.14159 \cdot 1k \cdot \text{time} + V(2))$ } ; napětí z modulátoru PSK, kde
+ počáteční fáze je modulována
+ napětím uzlu 2

Analýza AC

Analýza AC = kmitočtová analýza, analýza obvodu v harmonickém ustáleném stavu

= napodobování funkce „intelligentního“ obvodového analyzátoru



Měřicí vodiče 50 Ω



Analýza AC

Analýza AC = kmitočtová analýza, analýza obvodu v harmonickém ustáleném stavu

= napodobování funkce „inteligentního“ obvodového analyzátoru

Další příbuzné analýzy:

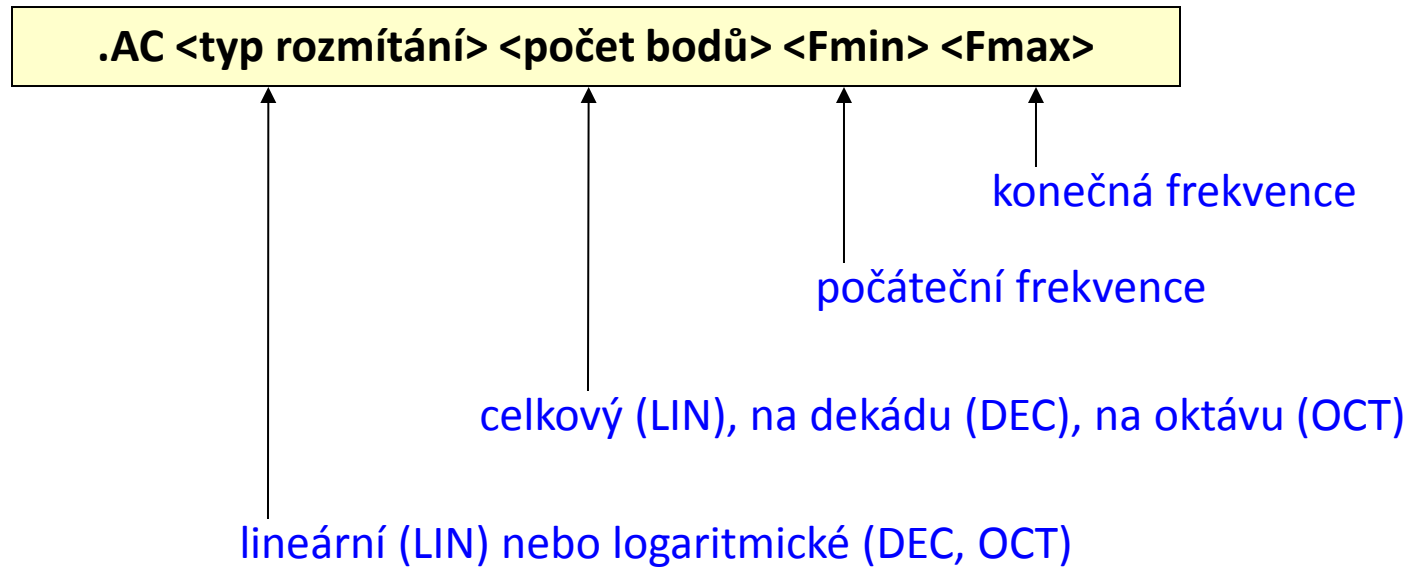
- šumová analýza (.NOISE)
- analýza nelineárního zkreslení (.DISTO, ne v Pspice)
- analýza nulových bodů a pólů (.PZ, jen v Spice 3)
- různé způsoby dalšího zpracování analyzovaných funkcí (IFFT..)

„Inteligentní“ obvodový analyzátor:

- amplitudy a počáteční fáze střídavých napětí a proudů kdekoli v obvodu při buzení obvodů z harmonických zdrojů o jednotné frekvenci
- možnost centrálního řízení této frekvence
- kmitočtové charakteristiky všeho typu
- sledování vlivu teploty a dalších parametrů na charakteristiky

Analýza AC

Syntaxe analýzy AC



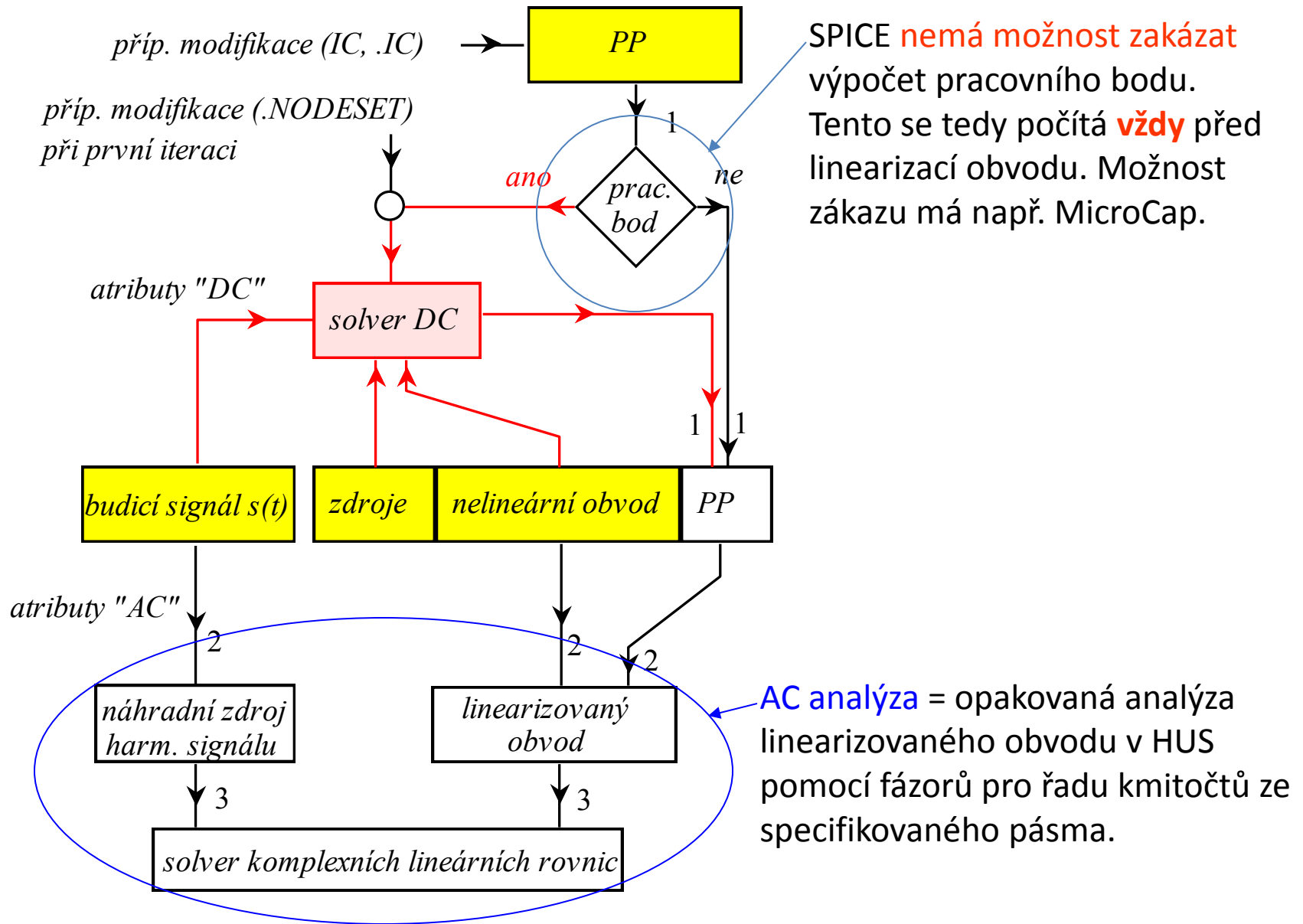
Příklady:

.AC LIN 200 10 1k; analýza od kmitočtu 10Hz do 1kHz v 200 bodech, lineární dělení

.AC DEC 5 1 100k; analýza od 1Hz do 100kHz, každá dekáda obsahuje 5 bodů

Analýza AC

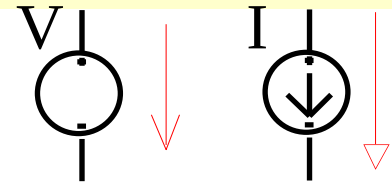
Jak postupuje simulátor při analýze AC



Analýza AC

Jaké jsou zdroje střídavého signálu v obvodu:

Nezávislé zdroje napětí a proudu s atributy „AC“ - příklady:



Vin 1 2 AC 10mV 90

;harmonický signál o amplitudě 10mV a počáteční fázi 90 stupňů

IAC 3 5 DC 10mA AC 5mA 0

;harmonický signál o amplitudě 5mA a nulové počáteční fázi; atribut DC se uplatní při výpočtu stejnosměrného pracovního bodu.

IAC 3 5 DC 10mA AC 5mA 0

;stejně jako u předchozího příkladu

Vbat 1 2 5V

;harmonický signál o nulové amplitudě; DC atribut 5V se uplatní při výpočtu stejnosměrného pracovního bodu

Vsinus x y AC 10 SIN 0 1 5k

;harmonický signál o amplitudě 10V a poč. fázi 0; nulová stejnosměrná složka vstupuje do výpočtu pracovního bodu; časový průběh (sinusový o amplitudě 1V a kmitočtu 5kHz) se uplatní jen při analýze Transient a nemá s analýzou AC nic společného

Kmitočet signálů z těchto zdrojů je stanoven „centrálně“ příkazem .AC.

Analýza AC

Reprezentace střídavého signálu v paměti

V paměti je každý signál obvodu při analýze .AC reprezentován jako **komplexní číslo = fázor**.

Modul komplexního čísla znamená **amplitudu** signálu.

Argument komplexního čísla znamená **počáteční fázi** ve stupních.

Analýza AC

SPICE konvence – reprezentace proměnných při analýze AC:

V nebo VM nebo I nebo IM

VP nebo IP

VR nebo IR

VI nebo II

VDB nebo IDB

modul napětí V, resp. proudu I

počáteční fáze napětí V, resp. proudu I

reálná část napětí V, resp. proudu I

imaginární část napětí V, resp. proudu I

modul napětí, resp. proudu v decibelech

Příklady:

V(2)

modul napětí mezi uzlem 2 a referenčním uzlem

VM(x,z)

modul napětí mezi uzly x a z

VI(R12)

imaginární část napětí na rezistoru R12

VDB(D1)

modul napětí mezi anodou a katodou diody D1 v decibelech

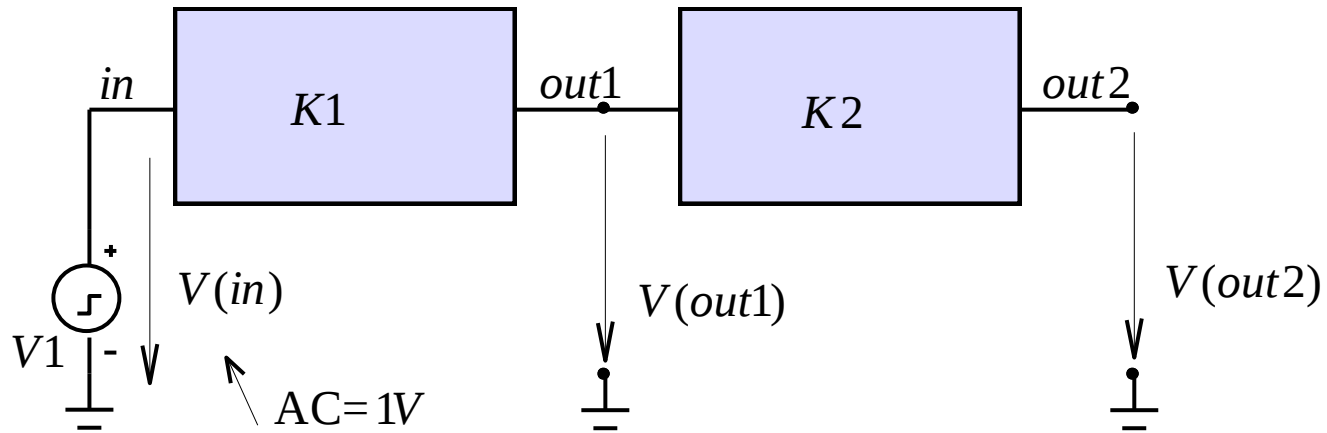
Analýza AC

Funkce SPICE a PROBE, aplikovatelné ve vzorcích na proměnné V a I při analýze AC:

M()	modul
P()	argument (počáteční faze)
R()	reálná část
IMG()	imaginární část
DB()	decibely

Analýza AC

Způsob získávání kmitočtových charakteristik při analýze AC:



$$V(out1) = K1$$

$$V(out2) = K1.K2$$

$$V(out2)/V(out1) = K2$$

Analýza AC

Závislé (řízené) zdroje pro transformaci signálů v analýze AC

vysvětleno na zdrojích typu E

Bloky s kmitočtově závislými vlastnostmi, modelovanými **vzorci**

E<název> <výstup+> <výstup-> LAPLACE {<výraz>}[=] {<výraz_s>}

musí obsahovat napětí
nebo proud v obvodu

představuje vzorec obvodové funkce,
obsahující Laplaceův operátor s

výstup= {<výraz>}*{<výraz_s>}

Příklad:

Vin 1 0 AC 1

.param f0 1k pi 3.14159

Elap 2 0 LAPLACE {V(1)} {2*pi*f0/(s+2*pi*f0)}

;modelování filtru 1. řádu
typu DP, vstup 1-0, výstup 2-0

Analýza AC

Závislé (řízené) zdroje pro transformaci signálů v analýze AC

vysvětleno na zdrojích typu E

Bloky s kmitočtově závislými vlastnostmi, modelovanými **tabulkami**

E<název> <výstup+> <výstup->
+FREQ {<výraz>}[=] [KEYWORD] <<f>,<m>, <p>>* [DELAY=<zpoždění>]

trojice čísel, jejichž význam je dán klíčovým slovem KEYWORD

zpoždění bloku, které ovlivní výslednou fázovou kmitočt. charakteristiku

výstupní napětí= {<výraz>}*{modelovaná kmitočtová závislost}

Analýza AC

Závislé (řízené) zdroje pro transformaci signálů v analýze AC

vysvětleno na zdrojích typu E

Bloky s kmitočtově závislými vlastnostmi, modelovanými **tabulkami**

E<název> <výstup+> <výstup->

+FREQ {<výraz>}[=] [KEYWORD] <<f>,<m>, <p>>* [DELAY=<zpoždění>]

KEYWORD	f	m	p
není uvedeno	kmitočet	modul v db	fáze ve stupních
DB	kmitočet	modul v db	fáze ve stupních
MAG	kmitočet	modul	fáze ve stupních
DEG	kmitočet	modul v db	fáze ve stupních
RAD	kmitočet	modul v db	fáze v radiánech
R_I	kmitočet	reálná část	imaginární část

Analýza AC

Bloky s kmitočtově závislými vlastnostmi, modelovanými **tabulkami**

```
E<název> <výstup+> <výstup->  
+FREQ {<výraz>}[=] [KEYWORD] <<f>,<m>, <p>>* [DELAY=<zpoždění>]
```

Příklad:

Vin 1 0 AC 1

```
Elap 2 0 FREQ {V(1)}= (10,0,0)  
+(100,0,0) (1k,-2,-25) (2k,-30,-33)  
+(10k,-50,-45) (100k,-70,-88)  
.AC DEC 10 10 100k  
.probe  
.end
```

Příklad:

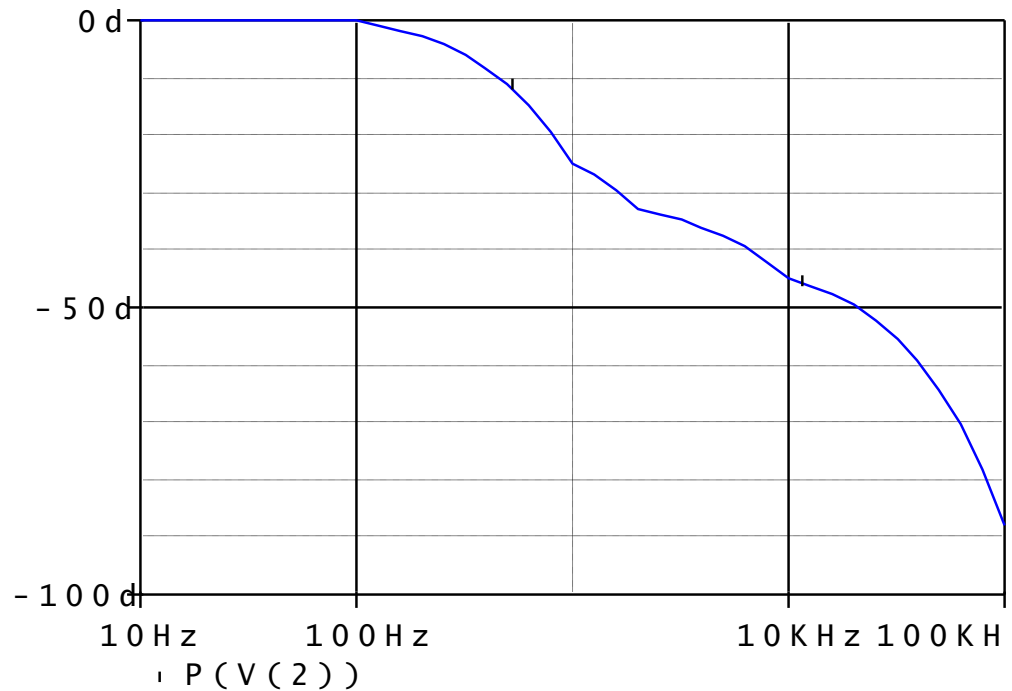
Vin 1 0 AC 1

Elap 2 0 FREQ {V(1)}= (10,0,0)
 +(100,0,0) (1k,-2,-25) (2k,-30,-33)
 +(10k,-50,-45) (100k,-70,-88)

.AC DEC 10 10 100k

.probe

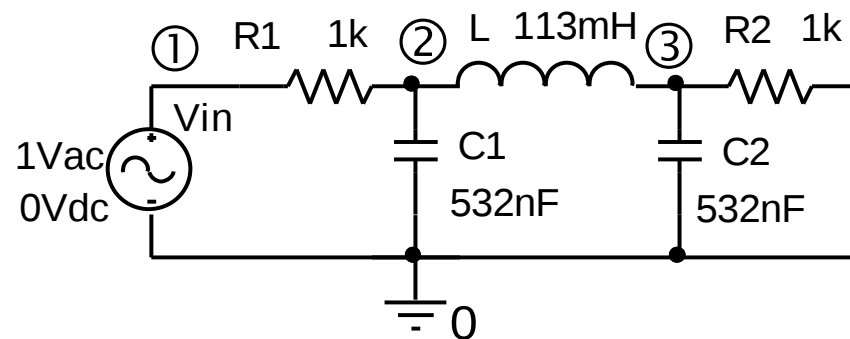
.end



Analýza AC

Použití příkazu .PRINT v analýze AC

```
.....  
.AC .....  
.PRINT AC <veličina1,veličina2..>  
.....
```



Filtr DP 3. radu

Vin 1 0 AC 1

R1 1 2 1k

C1 2 0 532n

L 2 3 113m

C2 3 0 532n

R2 3 0 1k

.AC LIN 1 1k 2k

.print AC V(3) VP(3)

.print AC VR(3) VI(3)

.end

FREQ	V(3)	VP(3)
1.000E+03	3.573E-01	1.689E+02

FREQ	VR(3)	VI(3)
1.000E+03	-3.506E-01	6.865E-02

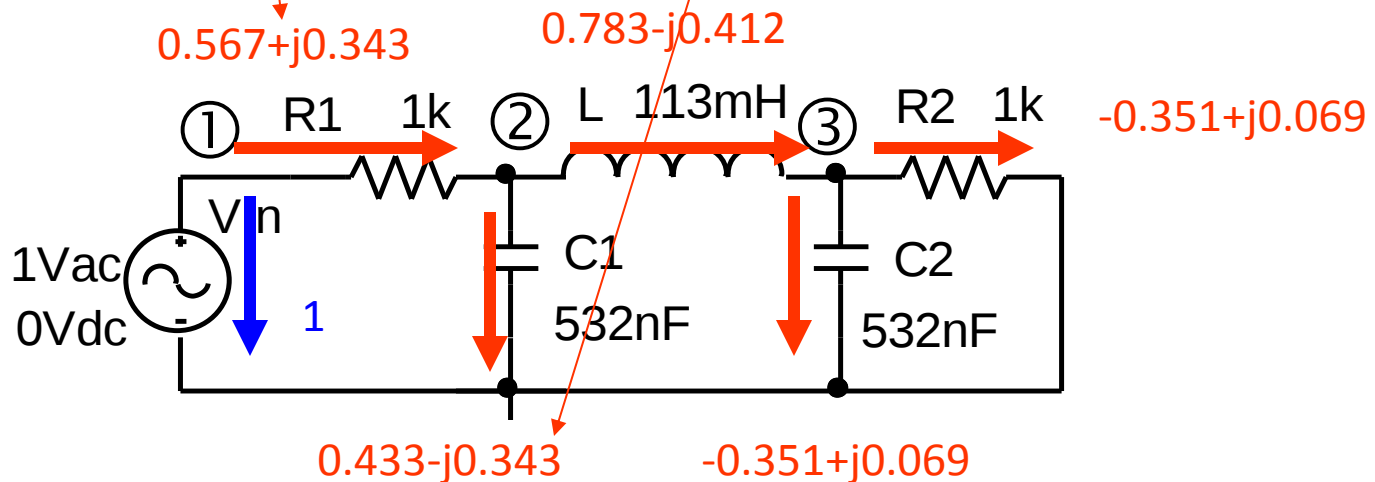
Analýza AC

Použití příkazu .PRINT v analýze AC

```
.print AC VR(R1) VI(R1) VR(C1) VI(C1) VR(L) VI(L) VR(C2) VI(C2) +VR(R2) VI(R2)
```

FREQ	VR(R1)	VI(R1)	VR(C1)	VI(C1)	VR(L)
1.000E+03	5.672E-01	3.432E-01	4.328E-01	-3.432E-01	7.834E-01
FREQ	VI(L)	VR(C2)	VI(C2)	VR(R2)	VI(R2)
1.000E+03	-4.119E-01	-3.506E-01	6.865E-02	-3.506E-01	6.865E-02

f=1kHz:



Analýza AC

Kmitočtové charakteristiky v analýze AC

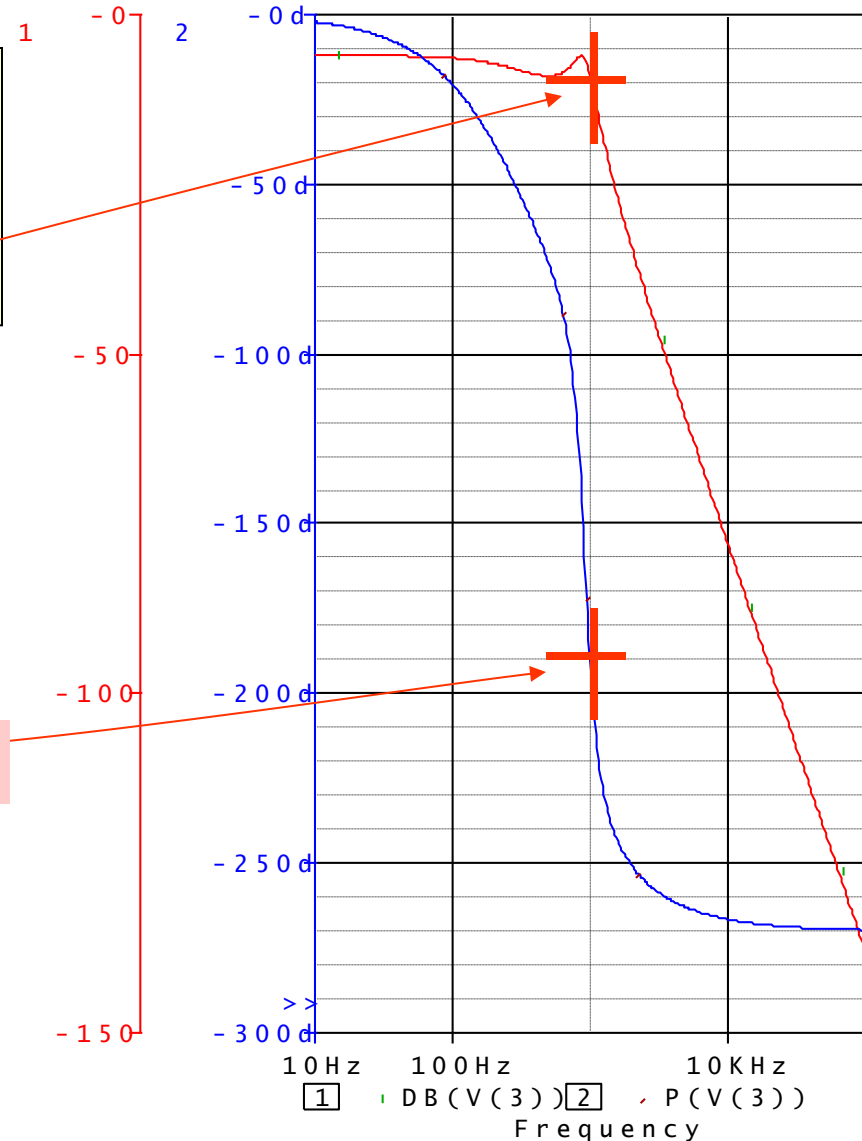
```
.....  
.AC DEC 100 10 100k  
.probe  
.end
```

0.3573 = -8.94 dB

FREQ	V(3)	VP(3)
1.000E+03	3.573E-01	1.689E+02

169 stupňů = -191 stupňů

FREQ	VR(3)	VI(3)
1.000E+03	-3.506E-01	6.865E-02



Analýza AC

Kmitočtové charakteristiky v analýze AC

```
.....  
.AC DEC 100 10 100k  
.probe  
.end
```

FREQ	V(3)	VP(3)
1.000E+03	3.573E-01	1.689E+02

FREQ	VR(3)	VI(3)
1.000E+03	-3.506E-01	6.865E-02

