

Počítačová simulace analogových obvodů

1. Program PSpice
 - 1.1. Vstupní soubor *.cir
 - 1.2. Součástky v PSpice
 - 1.3. Příkazy PSpice
 - 1.4. Standardní modely součástek v PSpice
2. Program Probe
3. Program Schematics
 - 3.1. Základní operace se Schematics
 - 3.2. Tvorba schémat pomocí Schematics

Počítačová simulace analogových obvodů

Simulační programy jsou užitečným nástrojem při návrhu obvodů a systémů. Role simulačního programu je v rychlém oddělení nemožných řešení od řešení pravděpodobně dobrých. Velmi snadno lze měnit parametry obvodu a pozorovat jejich vliv. Lze předem dobře připravit experiment a ušetřit čas a peníze. Úspěšný výsledek simulace však nesmí být považován za důkaz správné činnosti obvodu - ten může podat jen experiment s reálnými součástkami.

Ačkoliv simulace mohou být vhodným doplňkem teoretického rozboru úlohy, je třeba varovat před přílišnou důvěrou v jejich výsledky. Simulační program pracuje s čísly a nerozlišuje fyzikální veličiny. Chyby nejsou v aritmetice, ale v logice algoritmů, které nemohly podchytit všechny možné "podezřelé" okolnosti. Těmi může být příliš hrubý krok iterace, příliš malé číslo, pomalá konvergence, nedokonalé modely součástek (zvláště nelineárních, polovodičových), souhra jednostranných tolerancí, atd. Ačkoliv u jednoduchých lineárních obvodů může k odhalení chyby stačit technický cit a znalost základů elektrotechniky, u složitých elektronických obvodů je nutná hluboká znalost jejich funkce.

Pro simulaci analogových obvodů jsou dnes velice rozšířené programy, vycházející z původního vzoru SPICE. Program SPICE (Simulation Program with Integrated Circuit Emphasis) vznikl na University of California, Berkeley, v 70. letech. Byl postupně zdokonalován a několika softwarovými firmami přepracován do komerční podoby. Jednou z variant, která dosáhla značného rozšíření, byl "PSpice" (SPICE pro personální počítače) firmy MicroSim Corporation. V této kapitole bude přibližně popsán profesionální programový produkt Design Center, verze 6.1, zmíněné firmy. Účelem nebude překlad příruček v plném rozsahu, nýbrž popis těch nejpodstatnějších vlastností tak, aby bylo porozuměno principům. Méně často používané nebo nepodstatné vlastnosti budou vynechány. Výsledkem by měla být schopnost účelně využívat simulační program pro běžné úlohy a rychle se orientovat ve firemních příručkách v případě komplikovaných úloh.

Při studiu této kapitoly je velmi vhodné současně experimentovat s programem. Ačkoliv se výklad opírá o profesionální verzi, která nebude hromadně dostupná, je možné pracovat s některou z demonstračních verzí, které jsou volně šiřitelné. Příkladem může být "Evaluation Version 6.1" pro DOS nebo "Evaluation Version 5.4" pro Windows od téže firmy. Tyto demonstrační verze jsou plně funkční, a ačkoliv mají proti profesionální verzi některá kvantitativní omezení (hlavně v počtech součástek v jednom schématu a v rozsahu knihoven), zachovávají si všechny podstatné funkční vlastnosti verze profesionální a pro menší úlohy jsou zcela postačující.

1. Program PSpice

Program PSpice zpracovává vstupní soubor, který obsahuje popis zapojení obvodu a příkazy k simulaci. Vstupní soubor je textový a má příponu **.cir**. Může být předem vytvořen jakýmkoliv textovým editorem. Existují též grafické editory, které umožňují nakreslit schéma s vloženými všemi potřebnými údaji pro simulaci, a které na závěr ze schématu vygenerují textový soubor ***.cir**. Použití grafického editoru přináší klady i zápory, jak bude dále ukázáno. Ve verzi pro Windows je firmou MicroSim dodáván editor schémat "Schematics". Dále mohou být volány knihovní soubory, obsahující modely součástek a podobvody (příkaz **.LIB**), a vkládané soubory (příkaz **.INC**). Knihovny součástek jsou jednak součástí programového produktu Design Center, jednak jsou dodávány výrobcem součástek.

PSpice provádí několik druhů analýz. Druh požadované analýzy se volí příslušným příkazem. Během své činnosti vytváří PSpice výstupní textový soubor s příponou **.out**, který obsahuje kopii souboru **.cir**, informace o průběhu simulace (případně o chybách) a v omezeném rozsahu i tabulky hodnot některých analýz (hlavně údaje o pracovním bodu). Dále je vytvářen rozsáhlý datový soubor s příponou **.dat**, který je v následující etapě zpracován programem pro zobrazení dat (grafický postprocesor). Ten umožňuje nejenom zobrazit výsledky ve formě grafů, ale i provádět různé matematické operace. Grafický postprocesor k programu PSpice se nazývá "Probe".

Programy **Schematics**, **PSpice** a **Probe** tak tvoří organický celek s vzájemnou návazností. To však nebrání samostatnému použití kteréhokoliv z nich.

1.1. Vstupní soubor ***.cir**

Vstupní soubor začíná *titulní řádkou*, která je programem ignorována a slouží jen pro informaci. Může být i prázdná. Další částí je tzv. "*netlist*", což je skupina deklarací, které popisují každou součástku v obvodu, její spojení v obvodu a její vlastnosti. Každá deklarace zaujímá jeden řádek; pokud je příliš dlouhá, může pokračovat na dalším "pokračovacím" řádku za znakem "+". Pokračovacích řádků může být více. Třetí částí jsou *příkazy simulátoru*. Příkazy vždy začínají tečkou. Posledním příkazem souboru ***.cir** musí být **.END**. Označuje konec definice obvodu ve vstupním souboru. V jednom souboru může být definováno několik obvodů, vždy zakončených **.END**. Analýza je pak prováděna postupně, pro každý obvod nezávisle. Výsledky jsou však ukládány jen do jednoho souboru ***.dat** a ***.out**. *Pořadí* ostatních

příkazů a deklarací je *libovolné*, je však rozumné je vhodně seskupovat pro lepší čitelnost programu. Prázdné řádky jsou ignorovány. K lepší srozumitelnosti přispívají i komentáře. *Komentář* je uvozen buď znakem "*" na první pozici v řádku, nebo ";" za deklarací či příkazem. Program nikde *nerozlišuje velká a malá písmena*. Jejich volba opět souvisí jen s lepší přehledností. Formální zápis deklarací a příkazů, používaný v následujícím textu, je tento:

<povinná položka>
 <povinná položka opakovaná> *
 [nepovinná položka]
 [nepovinná položka opakovaná]*

Jako *oddělovače* mezi položkami jsou přípustné mezery, čárky a tabulátor. **Pozor, na nesprávném místě (uprostřed položky) vložená mezera ukončí čtení položky a ta je pak interpretována chybně!**

Číselné konstanty

Číselné konstanty v PSpice mohou být vyjádřeny buď jako čísla s pohyblivou **desetinnou tečkou** (nikoliv čárkou!), nebo v exponenciálním tvaru, nebo s využitím standardních symbolů: **f** (femto), **p** (piko), **n** (nano), **u** (mikro), **m** (mili), **k** (kilo), **MEG** (Mega), **G** (Giga), **T** (Tera). Nerozlišuje se velikost písmen, je však vhodné používat běžně používané značení. **Pozor na omyly: M i m značí "mili", meg i MEG značí "Mega", u i U značí "mikro". Další znaky, následující bezprostředně (tedy bez mezery) za číslem či za jednotkou, se ignorují až k oddělovači. Je proto vhodné zde uvádět symboly fyzikálních jednotek. Je to jen pro lepší čitelnost, PSpice pracuje s čísly a pojem fyzikálních jednotek nezná.**

Příklady: 1050000 1.05E6 1.05MEGOhm ; ekvivalenty čísla 1,05.10⁶
 .01V 0.01V 1e-2 10m 10000u ; ekvivalenty čísla 10⁻²
 3F ; 3.10⁻¹⁵, nikoliv 3 Farady !

Výrazy

Výraz je vždy uzavřen ve složených závorkách {}. Proměnnými ve výrazu jsou napětí, proudy, parametry a funkce (viz další text), **time** (jen v ABM - viz Schematics). Aritmetické operátory jsou + - * / . Jsou též dovoleny matematické funkce v následující formě:

<u>Funkce:</u>	<u>Význam:</u>	<u>Poznámka:</u>
abs(x)	x	
sgn(x)	+1 při x>0, 0 při x=0, -1 při x<0	
stp(x)	1 při x>0, jinak 0	jednotkový skok
sqrt(x)	$x^{1/2}$	
exp(x)	e^x	
log(x)	ln(x)	
log10(x)	$\log_{10}(x)$	
pwr(x,y)	$ x ^y$	
pwr(x,y)	+ x ^y při x>0, - x ^y při x<0	
m(x)	abs. hodnota komplex. čísla	
p(x)	fáze	ve stupních
r(x)	reálná část x	
img(x)	imaginární část x	
sin(x)		x v radiánech
asin(x)	arcsin(x)	výsledek v radiánech
sinh(x)		
cos(x)		výsledek v radiánech
acos(x)	arccos(x)	výsledek v radiánech
cosh(x)		
tan(x)		x v radiánech
atan(x), arctan(x)	arctg(x)	výsledek v radiánech
atan2(y,x)	arctg(y/x)	výsledek v radiánech
tanh(x)		x v radiánech
min(x,y)	minimum z x,y	
max(x,y)	maximum z x,y	
limit(x,min,max)	oboustranné omezení x	
table(x,x ₁ ,y ₁ ,x ₂ ,y ₂ ,...)	zadání funkce tabulkou dvojic x _i ,y _i	s interpolací
if(t,x,y)	x při t=TRUE, y při t=FALSE	t je relační výraz s operátory jako v "C", ==,!=,>,>=,<,<=

Závorky () lze používat dle běžných pravidel. Soubor funkcí se poněkud liší od funkcí v PROBE.

Topologie obvodu

Obvod se skládá z *uzlů* a *součástek*, kterými jsou uzly propojeny. Topologie obvodu musí splňovat jisté podmínky:

- ◆ V každém obvodu musí existovat uzel **0**, což je **referenční uzel** (zem u analogových součástek).
- ◆ Každý uzel musí být **označen jménem**, žádné dva uzly nesmějí mít stejná jména.
- ◆ Do každého uzlu musí být spojeny **alespoň dvě součástky**.
- ◆ Nesmějí existovat **nezapojené součástky** ani jejich jednotlivé vývody. To lze snadno ošetřit spojením takového vývodu s uzlem **0** přes rezistor o velmi vysoké hodnotě.
- ◆ Pro každý uzel musí existovat **cesta pro stejnosměrný proud** k uzlu **0**.
- ◆ Nesmí existovat dva nebo více **paralelně zapojených zdrojů** napětí či sériově zapojených zdrojů proudu.
- ◆ Zdroj napětí nesmí být **přemostěn induktorem** bez sériového rezistoru.

Podle konvence teče proud vždy od kladnějšího uzlu k uzlu zápornějšímu. U napěťových zdrojů pak vycházejí záporné proudy.

Každý vývod (svorka) součástky musí být zapojen do nějakého uzlu v síti. **Jména uzlů** definují zapojení součástky v síti. Součástky s uzly stejných jmen jsou navzájem spojeny. Jménem uzlu může být libovolný řetězec číslic a písmen v délce do 131 znaků. Některá jména jsou však rezervována (tzv. globální symboly). Je to jméno **0** (nula) vyhrazené pro referenční uzel (zem u analogových součástek), jména začínající **\$G_** pro rozvod pevných napájecích napětí a země u číslicových součástek, a některá další jména.

Jméno součástky musí začínat písmenem, které udává typ součástky (R, C, L, E, V, D, Q, M, atd. - viz přehled součástek v dalším textu). Za ním bez mezery následuje řetězec, volený uživatelem. Přípustná jsou čísla, písmena a znaky **\$ _ * / % .** Maximální délka řetězce je 131 znaků. V PSpice existují dvě kategorie součástek. Jedny jsou jím přímo modelovány a jejich jméno musí začínat příslušným písmenem, druhé jsou popsány jako podobvody v dalších knihovních souborech a jejich jméno v deklaraci musí začínat písmenem **X**. Podobvody s definicemi nových součástek si může vytvářet i sám uživatel.

Značení napětí a proudů

Napětí jsou zásadně značena jako V (Voltage) , proudy jako I (Intensity).

V(<jméno uzlu>)	; napětí uzlu proti zemi
V(<jméno uzlu 1>,<jméno uzlu 2>)	; napětí mezi uzly
V(<jméno součástky>)	; napětí na součástce se dvěma vývody
Vx(<jméno součástky>)	; napětí vývodu proti zemi u součástky s více vývody, vývody mají standardní označení
Vxy(<jméno součástky>)	; napětí mezi vývody souč. s více vývody, vývody mají standardní označení
I(<jméno součástky>)	; proud součástkou
Ix(<jméno součástky>)	; proud do vývodu součástky s více vývody, vývody mají standardní označení

Elektrody bipolárního tranzistoru jsou standardně značeny jako C,B,E , tranzistoru FET jako D (Drain - kolektor), S (Source - emitor), G (Gate - hradlo), B (Bulk - substrát). Jako nikde jinde, ani zde se nerozlišuje velikost písmen.

<u>Příklady:</u>	V(3)	; napětí uzlu 3 proti zemi
	V(3,2)	; napětí mezi uzly 3 a 2
	V(R1)	; napětí na rezistoru R1
	Vb(Q3)	; napětí na bázi tranzistoru Q3 proti zemi
	VDS(M5)	; napětí mezi kolektorem a emitorem MOS FET M5
	I(D5)	; proud diodou D5
	Ib(Q4)	; proud do báze tranzistoru Q4

Jako příklad souboru .cir a odpovídajícího schématu slouží obr. 1. Některé uzly byly v nakresleném schématu označeny čísla (0 až 7), některé textem (out1, out2). Uzel 0 (analogová zem) je společný. Příkazy jsou uvozeny tečkami (.LIB, .OP, .END). Je zřejmé, že obvod i jeho součástky jsou popsány velmi úsporným způsobem. Změny hodnot součástek nebo napětí zdrojů lze snadno provést textovým editorem jako změnu několika málo čísel. Vložení součástek nebo dalších příkazů je rovněž jednoduché.

Práce s programem PSpice

PSpice lze vyvolat buď jako samostatný program, nebo ze schematického editoru (Schematics). Jako samostatný program se ovládá velmi jednoduše - po jeho vyvolání se z menu vybere **File/Open** a soubor .cir. Program hlásí průběh simulace. Před spuštěním simulace je vhodné nejprve vyvolat kontrolu syntaxe. PSpice tuto

funkci obsahuje. Včasné odhalení chyby může ušetřit čas. Po skončení činnosti lze z menu vyvolat Probe k zobrazení výsledků, nebo prohlížet výstupní soubor .out.

Rozdilovy stupen (prvni radek se ignoruje)

```
.LIB nom.lib ;budou pouzity knihovny

R_RBIAS 1 2 20k ;rezistor 20 kΩ
R_RC1 OUT1 2 10k
R_RS2 3 0 1k
R_RS1 4 5 1k
Q_Q1 1 1 6 Q2N2222 ;bipolarni tranzistor
Q_Q4 7 1 6 Q2N2222
Q_Q2 OUT1 5 7 Q2N2222
Q_Q3 OUT2 3 7 Q2N2222
C_CLOAD OUT1 OUT2 5p ;kondenzator 5 pF
V_V1 4 0 DC 0 AC 1 SIN 0v 0.1v 5Meg 0 0 0 ;zdroj napeti
R_RC2 OUT2 2 10k
V_V2 2 0 12V
V_V3 6 0 -12V

.OP ;prikaz pro analyzu

.END ;konec souboru
```

Obr. 1. - schéma obvodu a odpovídající soubor .cir

1.2. Součástky v PSpice

PSpice obsahuje zabudované modely pro simulaci několika desítek typů součástek. Mezi nimi jsou jednoduché součástky (rezistor, kondenzátor, induktor), složitější pasivní součástky (transformátor, vedení, napěťově a proudově řízený spínač) i polovodičové součástky (bipolární i unipolární tranzistory, diody, atd.). Implicitní hodnoty v programu samozřejmě nejsou beze změny použitelné pro konkrétní typy součástek od různých výrobců, a proto lze při deklaraci součástky jmenovat i model. Ten musí být přístupný v některém z knihovnických souborů, nebo jej lze definovat zvláštním příkazem (.MODEL). Všechny odvozené modely vycházejí ze zabudovaných (standardních) modelů. V dalším textu nebudou zahrnuty informace o tvorbě modelů a jejich přizpůsobování. Potřebují se k tomu rozsáhlé údaje o vnitřních modelech. V nutném případě je třeba prostudovat patřičné příručky. K programu PSpice jsou však k dispozici knihovny modelů od výrobců polovodičových součástek, které jsou velmi rozsáhlé a jsou výrobci stále doplňovány.

Deklarace součástky

Deklarace popisuje součástku, její spojení v obvodu a její vlastnosti. V dalším textu budou uvedeny jen základní tvary deklarácí. Mnohé jsou ve skutečnosti daleko širší, v závislosti na verzi PSpice. Uvedené tvary jsou však postačující pro pochopení práce s PSpice i pro běžné úlohy. Deklarace jednoduché součástky se dvěma vývody je následující:

<jméno součástky> <jméno uzlu 1> <jméno uzlu 2> <hodnota>

U součástek se dvěma svorkami teče kladný proud od první svorky ke druhé. U složitějších součástek existuje větší počet uzlů. U deklarácí napěťových zdrojů je na prvním místě uváděna svorka s kladným napětím, na druhém se záporným napětím. Pokud nejsou uvedena napětí u napěťových zdrojů a proudy u zdrojů proudových, jsou implicitně nulové.

Hodnotami mohou být buď **číselné konstanty**, nebo **matematické výrazy** (po vyhodnocení musí dát číslo), nebo **symboly** (ty musí být na jiném místě deklarovány příkazem .PARAM), nebo **jméno modelu**.

Rezistor

R<jméno> <uzel> <uzel> [jméno modelu] <hodnota> [TC=<TC1>[,TC2]]

Model zabudovaný v PSpice zohledňuje teplotní závislost:

$$\langle \text{hodnota} \rangle \cdot (1 + TC1(T - T_{\text{nom}}) + TC2(T - T_{\text{nom}})^2),$$

kde T je teplota, jedna ze systémových proměnných PSpice, a T_{nom} (implicitně 27°C) je jmenovitá teplota. Implicitně jsou oba koeficienty nastaveny na nulu. V deklaraci však mohou být uvedeny jiné hodnoty, případně je lze změnit příkazem .MODEL. Může též být vyvolán jiný model s jinými vlastnostmi.

Příklad:

Rvyst 4 0 8	; rezistor Rvyst = 8 Ω mezi uzlem 4 a zemí
Rc 6 2 2.7k	; rezistor Rc = 2k7 mezi uzly 6 a 2
Rb1b2 6 7 Rtemp_komp 10k	; rezistor o hodnotě 10 k a speciálních vlastnostech modelu Rtemp_komp
R2 1 2 1k TC=0.015, -0.003	; rezistor s teplotní závislostí, 1 k při T _{nom}

Kondenzátor

C<jméno> <uzel> <uzel> [jméno modelu] <hodnota>
+[IC=<počáteční podmínka>]

Přiměřeně platí totéž, co u rezistoru. Implicitní model PSpice zohledňuje teplotní i napěťovou závislost, koeficienty jsou však nastaveny na nuly. Mohou být nastaveny v příkazu .MODEL. Počáteční podmínka udává napětí na kondenzátoru na začátku simulace.

Příklad: Cvaz 3 7 100pF IC=1.5V ; poč. hodnota 1,5 V s + na uzlu 3

Induktor

L<jméno> <uzel> <uzel> [jméno modelu] <hodnota>
+[IC=<počáteční podmínka>]

Přiměřeně platí totéž, co u kondenzátoru. Implicitní model PSpice je lineární a zohledňuje teplotní i napěťovou závislost, koeficienty jsou však nastaveny na nuly. Mohou být nastaveny v příkazu .MODEL, též může být volen vhodný nelineární model. Počáteční podmínka udává proud induktorem na začátku simulace.

Transformátor

V případě lineárního modelu magnetického jádra:

K<jméno> L<jméno> <L<jméno>>* <činitel vazby>

V případě nelineárního modelu magnetického jádra:

K<jméno> <L<jméno>>* <činitel vazby> <jméno modelu>

Induktory musí být deklarovány v jiných deklaracích. Činitel vazby musí být v mezích 0 až 1. V prvním případě jsou u induktorů jako hodnota deklarovány **indukčnosti**, ve druhém jsou to **počty závitů**. Ve druhém případě musí být uveden model popisující magnetické vlastnosti jádra. Smysl vinutí, který se ve schématech obvykle značí tečkami, vyplývá z pořadí uzlů v deklaracích induktorů - tečka přísluší prvnímu uzlu. Na jejich pořadí v deklaraci K nezáleží.

Příklad: L1 1 0 10uH
L2 2 0 20uH
K12 L1 L2 1 ; lineární model, hodnoty u induktorů jsou indukčnosti

Příklad: L1 5 9 20
L2 3 8 15
K1 L1 L2 K528T500_3C8 ; nelineární model, hodnoty u induktorů jsou závity

Dioda

D<jméno> <uzel anoda> <uzel katoda> <jméno modelu>

Příklad: D3 1 4 D1N3063 ; dioda z knihovny součástek

Bipolární tranzistor

Q<jméno> <uzel kolektor> <uzel báze> <uzel emitor> <jméno modelu>

Příklad: Q7 5 4 0 Q2N3901 ; tranzistor z knihovny součástek

Tranzistor JFET

J<jméno> <uzel kolektor> <uzel hradlo> <uzel emitor> <jméno modelu>

Tranzistor MOSFET

M<jméno> <uz. kol.> <uz. hrad.> <uz. em.> <uzel substrát> <jméno modelu>

Napětím řízený spínač

S<jméno> <+uzel výstup> <-uzel výstup> <+uzel řízení> <-uzel řízení>
+<jméno modelu>

Řídicím napětím je ovládán odpor mezi výstupními svorkami. Chování spínače je popsáno zabudovaným modelem VSWITCH, který je relativně jednoduchý. Je definován odpor v sepnutém stavu ($R_{ON}=1\Omega$), v rozepnutém stavu ($R_{OFF}=1\text{ M}\Omega$), řídicí napětí pro úplné sepnutí ($V_{ON}=1\text{V}$) a řídicí napětí pro úplné rozepnutí ($V_{OFF}=0\text{V}$). Mezi V_{OFF} a V_{ON} se odpor mění plynule (na však lineárně) mezi svými krajními hodnotami. Na základě prvku S lze po vhodné změně parametrů definovat např. relé, analogové spínače, apod.

Příklad: S12 13 17 2 0 SMOD ; spíná uzly 13-17, řízen napětím
; mezi uzly 2-0, model SMOD

Proudem řízený spínač

W<jméno> <+uzel výstup> <-uzel výstup> <řídící zdroj V> <jméno modelu>

Činnost je obdobná, jako u spínače S. Pouze řídicí veličina je získána jinak. Je to proud, protékající některou větví obvodu. Informaci o proudu lze získat tak, že se do obvodu vloží napěťový zdroj - PSpice totiž počítá všechny proudy zdrojů. Tento pomocný zdroj může mít nastaveno nulové napětí. Proto v deklaraci je položka *řídící zdroj V*.

Příklad: W12 7 8 Vpom WMOD ; řídicí veličinou je proud zdrojem Vpom

Zpoždovací linka beze ztrát

T<jméno> <uzel A> <uzel A> <uzel B> <uzel B> [jméno modelu]
+Z0=<hodnota> TD=<hodnota>

T<jméno> <uzel A> <uzel A> <uzel B> <uzel B> [jméno modelu]
+Z0=<hodnota> F=<hodnota> [NL=<hodnota>]

Tato součástka slouží jako ideální zpoždovací linka o impedanci Z_0 . Jeden konec linky je značen A, druhý B. Směr přenosu není určen. Prvý uzel A je spojen s prvním uzlem B, druhý s druhým. V první variantě deklarace je zpoždění zadáno přímo jako TD. Ve druhé variantě je zadán kmitočet F a činitel zkrácení vlnové délky NL. V modelu je implicitní hodnota $NL=0.25$; jestliže vyhovuje, nemusí se NL zadávat.

Příklad: T1 1 2 3 4 Z0=220 TD=115ns ; všechny jsou ekvivalentní
 T2 1 2 3 4 Z0=220 F=2.25MEG
 T3 1 2 3 4 Z0=220 F=4.5MEG NL=0.5

Zpoždovací linka se ztrátami

T<jméno> <uzel A> <uzel A> <uzel B> <uzel B> [jméno modelu]
 + LEN=<hodnota> R=<hodnota> L=<hodnota> G=<hodnota> C=<hodnota>

Zde LEN je elektrická délka vedení, R,L,G,C jsou hodnoty na jednotku délky.

Příklad: T1 1 2 3 4 LEN=1 R=.311 L=.378 G=6.28u C=67.3p

Nezávislý zdroj napětí

V<jméno> <+uzel> <-uzel> +[<[DC] napětí>]
 +[<AC> <napětí> [fáze ve stupních]]
 +[specifikace přechodného děje]+[STIMULUS=<jméno budícího průběhu>]

Tento zcela univerzální zdroj může být naprogramován tak, že pokrývá potřeby všech druhů analýz. Podle momentálního druhu analýzy pak PSpice vybírá patřičnou specifikaci. Může být stejnosměrným zdrojem (řádka 1), střídavým zdrojem (řádka 2), zdrojem tvarových kmitů (řádka 3). Průběh kmitu může být též vybrán ze souboru stimulů (řádka 3). Jako přechodný děj pro .TRAN analýzu (*specifikace přechodného děje*) musí být zvolen jeden z následujících:

EXP (v1 v2 td1 tc1 td2 tc2)	;viz obr. 2.
PULSE (v1 v2 td tr tf pw per)	;viz obr. 3.
PWL (t1,v1) (t2,v2) ...	;viz obr. 4.
PWL REPEAT FOR <n> (t1,v1) (t2,v2) ... ENDREPEAT	;n-kráte
PWL REPEAT FOREVER (t1,v1) (t2,v2) ... ENDREPEAT	;periodická
SFFM (voff vampl fc mod fm)	;viz obr. 5.
(offset amplituda kmitočet_nosné modulační_index kmitočet_modulace)	
SIN (voff vampl freq td df phase)	;viz obr. 6.
(offset amplituda kmitočet zpoždění činitel_tlumení fáze_ve_stupních)	

Všechny průběhy jsou jednorázové, kromě PULSE, který je opakovaný, a PWL (Piece-Wise Linear), které mohou být opakované. U PWL a SIN nemusí být první bod definován při nulovém čase (t₁ u PWL, t_d u SIN). PSpice doplní hodnoty před tímto bodem tak, že hodnota v prvním definovaném bodě je protažena doleva až do času 0. U PWL je tato hodnota dána souřadnicemi prvního bodu, u SIN závisí na fázi. Hodnota za posledním bodem u PWL je protažena do nekonečna.

Obr. 2. Průběh napětí zdroje EXP

Obr. 3. Průběh napětí zdroje PULSE

Obr. 4. Průběh napětí zdroje PWL

Obr. 5. Průběh napětí zdroje SFFM

Obr. 6. Průběh napětí zdroje SIN

Zdroj se specifikací AC neslouží jako zdroj střídavého napětí pro napájení obvodu, nýbrž výhradně jako zdroj signálu pro střídavou analýzu (příkaz .AC - viz další text). Tam se nastavují kmitočty, proto v této deklaraci se údaj o kmitočtu nevyskytuje. Pro generaci sinusového signálu pro časovou analýzu (příkaz .TRAN) a pro napájení obvodů se musí zdroj specifikovat jako SIN.

Příklady: V3 26 77 DC .02 AC 1 SIN(0 .1 1.5MEG 0 0 90) ; připraven pro
; .DC, .AC, .TRAN
VPULS 1 0 PULSE(-1mV 1mV 2ns 2ns 2ns 50ns 100ns)
V1 1 2 PWL REPEAT FOR 5 (1,0)(2,1)(3,3) ENDREPEAT

Nezávislý zdroj proudu

I <jméno> <+uzel> <-uzel> stejně jako u napěťového zdroje

Vše platí stejně, jako u napěťového zdroje.

Napěťově řízený zdroj napětí

E<jméno> <+uzel výstupní> <-uzel výstupní> <řízení....>

Zdroj má dvě výstupní a (až na výjimku POLY) dvě vstupní (řídící) svorky. Řízení může být několikerého typu:

<+uzel řídicí> <-uzel řídicí> <zesílení>
 VALUE = {<matematický výraz>}
 TABLE {<výraz pro řízení>} = <(vstup,výstup)> *
 POLY (<počet vstupů>) <<+uzel řídicí> <-uzel řídicí>> * <koefficient> *
 FREQ {<výraz pro řízení>} = <(kmitočet, zesílení_v_dB, fáze_ve_stupních)> *
 +[DELAY = <zpoždění>]
 CHEBYSHEV {<výraz pro řízení>} = <[LP][HP][BP][BR]>
 +<kmitočet zlomu> * <útlum v dB> *
 LAPLACE {<výraz pro řízení>} = {<výraz pro transformaci>}

Matematický výraz může využívat libovolné proměnné v obvodu. Výraz pro řízení popisuje vztah pro získání řídicího napětí. Nejběžnější je prostá specifikace některého uzlového napětí proti zemi V(uzel) nebo napětí mezi uzly V(uzel1,uzel2). Typ POLY umožňuje zavést větší počet řídicích signálů. Zde počet vstupů značí počet vstupních (řídících) signálů a musí se shodovat s počtem párů řídicích uzlů. Výstupní napětí je dáno polynomem, jehož koeficienty jsou uvedeny na konci deklarace. Typ FREQ je zadáván tabulkou. Typ CHEBYSHEV umožňuje realizovat

Čebyševovy filtry typu Low-Pass, High-Pass, Band-Pass, Band-Reject. Zadává se *útlum*, ne zesílení. Výraz *pro transformaci* udává Laplaceův obraz v proměnné *s* (místo u nás běžného *p*).

Zdroj E je velmi univerzální. Může reprezentovat např. napěťový zesilovač, nelinearity zadané buď matematickým vztahem nebo tabulkou (ta může být zadána jen hrubě, program provádí interpolaci), umožňuje tvarovat frekvenční charakteristiku i odezvu v časové oblasti. Dává široké možnosti uplatnění při modelování systémů.

Příklady: EZES 2 0 1 0 4E3 ;zesílení 4000
 E2 2 0 VALUE={2+V(3,4)*time*time} ;funkce času
 E3 2 0 TABLE{V(3,4)*100}={(-10,-10) (0,0) (100,5)}
 E10 2 0 FREQ{V(3)}=(0,0,0) (5kHz,-20,720)
 E20 2 0 LAPLACE{V(C3)}={s/(1+s*s)}
 ENELIN 100 101 POLY(2) 3 0 4 0 0 13.6 0.2 0.005 ; 2 vstupy, nelineární
 ELOWPASS 5 0 CHEBYSHEV {V(10)}=LP 800 1.2k 0.1dB 50dB

Napěťově řízený zdroj proudu

G<jméno> <+uzel výst. proud> <-uzel výst. proud> <řízení>

Vše platí obdobně, jako u napěťového zdroje. Místo zesílení se uvede transkonduktance.

Vedle uvedených dvou řízených zdrojů existuje ještě proudově řízený zdroj proudu **F** a proudově řízený zdroj napětí **H**, ale s podstatně omezenějšími možnostmi řízení (přibližně odpovídající 1. způsobu u zdroje E).

1.3. Příkazy PSpice

Příkazy tvoří třetí složku vstupního souboru *.cir. V dalším textu budou probrány pouze nejdůležitější příkazy. Lze je rozdělit do několika skupin:

- Pro odkazy na jiné soubory:
.INC, .LIB
- Pro modifikaci součástek nebo vlastností obvodu:
.SUBCKT, .MODEL, .PARAM, .IC
- Pro usnadnění zápisu:
.FUNC
- Pro řízení výstupu:
.PROBE
- Pro provádění analýz:
.OP, .DC, .TF, .AC, .TRAN, .FOUR, .STEP, .TEMP, .SENS,
.MC, .WCASE

V některých příkazech se vyskytuje specifikace "*výstupní proměnná*". Jedná se o napětí v uzlech, napětí na součástkách a proudy součástkami.

Vložení souboru

.INC <"*jméno souboru*">

Ve vkládaném souboru nesmí být první řádka bez označení komentáře.

Odkaz na knihovny

.LIB ["*jméno souboru*"]

Slouží k odkazu na modely nebo podobvody v knihovním souboru. Jméno souboru musí být úplné, včetně přípony. Knihovní soubor smí obsahovat jen definice modelů nebo podobvodů. Není-li uvedeno jméno souboru, PSpice získá informaci o všech knihovních souborech prostřednictvím textového souboru "nom.lib".

Podobvody

.SUBCKT<*jméno*> [*seznam vývodů*] [PARAMS: <*parametr=hodnota*>*]
..... tělo definice (popis schématu uvnitř podobvodu)
.ENDS

Umožňují sestavovat schéma z mnohonásobně využitých menších celků. V podobvodu může být vložen další podobvod. Seznam vývodů zahrnuje označení (čísla) všech vstupů a výstupů. Parametry umožňují podle potřeby modifikovat činnost jednou definovaného podobvodu tím, že hodnoty z definice budou změněny.

Příklad: .SUBCKT Op_Amp n p output
 Rin n p 10Meg
 Rout c output 100
 Egain c 0 p n 1E5
 .ENDS

Do schématu se podobvod vkládá jako jiná součástka. Jeho jméno vždy začíná písmenem X:

X<jméno> [*seznam vývodů*] <jméno podobvodu>
 +[PARAMS: <parametr=hodnota>*]

X<jméno> bude označení součástky v novém schématu, seznam vývodů bude odpovídat značení (číslování) v novém schématu. Jména parametrů musí být stejná, jako v definici podobvodu, ale jejich hodnoty jsou podle potřeby v novém schématu změněny. Tak lze opakovaně využívat podobvod v jednom schématu, vždy jako novou vkládanou součástku s patřičným očíslováním vývodů a s potřebnými hodnotami. V podobvodu jsou přípustné i vnitřní definice funkcí, modelů a parametrů - ty jsou však pak lokální a z vnější úrovně neviditelné. Objekty definované na globální úrovni jsou uvnitř podobvodu viditelné (srovnej s viditelností lokálních a globálních proměnných v jazyce C).

Příklad: XOA1 3 2 3 Op_Amp

Model

.MODEL <jméno modelu> [AKO: <jméno referenčního modelu>]
 + <typ modelu>([<parametr> = <hodnota> [tolerance]])*

Příkaz definuje množinu parametrů, charakterizujících vlastnosti konkrétní součástky. Umožňuje tak jejich přizpůsobení. Zde *jméno modelu* je dáno uživatelem, *typ modelu* se vztahuje ke standardnímu modelu v PSPICE, jako *parametry* se udávají jen ty, které je třeba nově specifikovat. AKO (z angl. "A Kind Of") a *jméno referenčního modelu* odkazuje na nějaký jiný, již přizpůsobený model, který se tím bere za základ.

Parametry

.PARAM <jméno parametru = konstanta nebo {výraz}> *

V mnoha případech je výhodné místo číselné hodnoty použít symbolickou hodnotu, parametr. Mohou se tak přiřadit shodné hodnoty několika součástkám najednou a měnit proměnné při opakované (parametrické) analýze, tj. při vyšetřování vlivu postupné změny některé veličiny. Definice parametru:

Příklad: .PARAM VCC=12V VEE= -12V
 .PARAM pasmo={100kHz/3}

Po definici se parametr stává globální hodnotou pro celé schéma a je na většině míst použitelný místo číselných hodnot. Pokud někde nahrazuje hodnotu (původně konstantní), uzavírá se do závorek {jméno parametru}.

Příklad:
 R20 2 0 {promenny_odpor} ; odpor udany parametrem

 .PARAM promenny_odpor = 1k ;takto je velky

Parametr se též může postupně měnit při opakované simulaci, což je základem parametrické analýzy.

Počáteční podmínky

.IC <V(<uzel [,<uzel>])=<napětí>>*

Někdy je pro analýzu .AC a .TRAN nutno nastavit počáteční podmínky jako napětí v uzlu nebo mezi uzly. Při analýze .DC je příkaz ignorován.

Příklad: .IC V(2)=10 V(2,3)= -1

Funkce

.FUNC <jméno>([seznam argumentů]) {<výraz>}

Pro zjednodušení a zpřehlednění zápisu lze definovat funkce. PSpice s nimi v podstatě provádí substituci textových řetězců (srovnej s #define v jazyce "C"). Ve vlastním programu tak lze různé konstanty nahradit symboly, zjednodušit zápis složitých a opakovaných výrazů, apod. Jméno se nesmí shodovat s předdefinovanými funkcemi. Ve výrazu, který se vždy uzavírá do závorek {}, se

může vyskytovat jiná (již definovaná) funkce. Počet argumentů v seznamu se musí shodovat s počtem ve výrazu. Je-li funkce bez argumentu, nesmějí závorky () chybět. Soubor s definicemi funkcí může být vložen pomocí .INC .

Příklad: .FUNC E(x) {exp(x)}
 .FUNC pi {3.14}
 .FUNC MIN3(A,B,C) {MIN(A,MIN(B,C))}

Výběr výsledků pro Probe

.PROBE [*výstupní proměnná*]*

Pokud nejsou uvedeny žádné výstupní proměnné, budou do souboru *.dat zapsány simulované hodnoty všech uzlových napětí a proudy všemi vývody všech součástek. To může vést na velmi rozsáhlé soubory, z nichž se pak využije jen malá část. Jsou-li uvedeny výstupní proměnné, budou se zapisovat jen data, která se jich týkají.

Příklad: .PROBE V(1) V(4,5) I(R2)
 .PROBE

Analýza pracovního bodu

.OP

Pracovní bod, tj. stav při ustálené hodnotě napětí a proudů, je počítán vždy, ale bez příkazu .OP jsou do výstupního souboru *.out zapsána jen uzlová napětí. Při přítomnosti příkazu .OP jsou vypsané linearizované parametry (tj. pro malé signály) všech nelineárních řízených zdrojů a všech polovodičových součástek.

Stejnoseměrná analýza

.DC [LIN] <proměnná> <počátek> <konec> <krok>
 + [<specifikace vnější smyčky>]
.DC DEC <proměnná> <počátek> <konec> <počet bodů>
 + [<specifikace vnější smyčky>]
.DC OCT <proměnná> <počátek> <konec> <počet bodů>
 + [<specifikace vnější smyčky>]
.DC <proměnná> LIST <hodnota> *
 + [<specifikace vnější smyčky>]

Výpočet pracovního bodu, opakovaný při proměnné hodnotě jednoho nebo dvou parametrů obvodu ("sweep analysis"). Změna proměnné může probíhat lineárně (označení LIN není nutné), logaritmicky po dekádách (DEC) nebo po oktávách (OCT) - při logaritmickém průběhu se neudává krok, ale počet hodnot v dekádě či oktávě. Průběh lze též naprogramovat po jednotlivých bodech (LIST). Pro LIN je možný průběh zdola nahoru i shora dolů (ale krok se uvádí vždy jako kladný), u DEC a OCT jen zdola nahoru. Pro LIST nejsou žádná omezení.

Mohou se měnit i dvě různé proměnné. Pak se musí uvést i *specifikace vnější smyčky*. Způsob změny druhé proměnné je nezávislý na způsobu změny první proměnné. Pořadí změn proměnných je takové, že na proběhnutí první proměnné (vnitřní smyčka) všemi hodnotami udělá druhá proměnná (vnější smyčka) jeden krok. Při změně dvou proměnných se do souboru *.dat pro zobrazení programem Probe předávají skupiny křivek. Lze tak vykreslit různé soustavy charakteristik, apod.

Jako *proměnná* zde může vystupovat jméno nezávislého zdroje V... nebo I..., proměnná TEMP (teplota), parametr modelu nebo globální parametr. V případě proměnného parametru modelu se na místě <proměnná> uvádí:

<<typ modelu v PSpice> <jméno modelu> <(proměnný parametr v modelu)>>

V případě proměnného globálního parametru, definovaného v souboru *.cir jako .PARAM *jméno*, se bude jednat o současné změny stejné proměnné (*jméno*), použité na několika místech programu. Na místě <proměnná> se uvádí:

<PARAM <jméno globálního parametru>>

<u>Příklady:</u>	.DC VIN -.25 .25 .05	; LIN vynecháno
	.DC LIN I2 5mA -2mA 0.1mA	; LIN uvedeno
	.DC VCE 0V 10V .5V IB 0mA 1mA 50uA	; kolektorové charakteristiky
	.DC RES RMOD(R) 0.9 1.1 .001	; model rezistoru
	.DC DEC NPN QFAST(IS) 1E-18 1E-14 5	; logaritmická změna, 5 bodů
		; v dekádě, parametr tranzist.
	.DC TEMP LIST 0 20 27 50 80 100	; změna teploty
	.DC PARAM Vsupply 7.5 15 .5	

Přenosová funkce

.TF <výstupní proměnná> <vstupní proměnná>

Obvod je linearizován v okolí pracovního bodu. Počítá se poměr změn výstupní proměnné ku vstupní proměnné, vstupní odpor a výstupní odpor. Výsledek je zapsán jen do souboru *.out.

Příklad: .TF V(5) VIN

Střídavá analýza

.AC <průběh> <počet bodů> <počátek> <konec>

Obvod je linearizován v okolí pracovního bodu. Jako *průběh* musí být uvedena jedna z možností LIN, DEC, OCT. Ve všech případech je počáteční kmitočet nižší než koncový kmitočet. Mění se současně kmitočet u všech nezávislých zdrojů napětí a proudu, deklarovaných s příznakem AC. U LIN značí *počet bodů* celkový počet frekvencí, krok je spočten jako konstantní. U DEC a OCT je to počet frekvencí v dekádě či oktávě. Počítána je amplituda i fáze, takže v Probe lze zobrazit fázovou i amplitudovou charakteristiku, a též z nich počítat další průběhy (např. reálné i imaginární složky, apod.).

Pozor !!! - Analýza .AC se provádí na linearizovaném modelu, takže nezáleží na amplitudě u zdrojů napětí nebo proudu. Nelze usuzovat na zkreslení při velkých amplitudách.

Příklady: .AC LIN 101 100kHz 200Hz
 .AC OCT 10 1kHz 16kHz
 .AC DEC 20 1MEG 100MEG

Přechodová analýza

.TRAN [/OP] <časový krok pro výpis> <konečný čas>
 +[počátek výpisu [mez časového kroku]][UIC]

Při časové analýze se uplatňuje proměnná TIME, která se mění od 0 do "*konečný čas*". "*Časový krok pro výpis*" se týká výpisu do výstupního souboru, nikoliv kroku pro výpočty. Hodnoty až do času "*počátek výpisu*" nejsou vypisovány, ale počítány jsou. Výpočty používají adaptivní algoritmus, přičemž časový krok je přizpůsobován reakci obvodu. Jsou-li děje pomalé, jeho maximální hodnota je implicitně 1/50 z hodnoty *konečný čas*, ale může být stanovena kratší, podle hodnoty "*mez časového kroku*". Jsou-li děje rychlé, je časový krok určován automaticky.

.STEP <proměnná> LIST <hodnota> *

Příkaz slouží pro opakovanou analýzu při postupné změně jedné proměnné. Význam položek je shodný s příkazem .DC. V ostatních příkazech pro analýzu se mohou přitom měnit jiné proměnné. Do souboru *.dat pro Probe se pak запиše soustava křivek. Tento příkaz nemá význam při analýze .DC, neboť příkaz .DC nabízí stejné možnosti. Ale např. současně s příkazem .AC lze studovat vlivy některých obvodových parametrů (tj. proměnných) na frekvenční charakteristiky, apod.

Příklad: .STEP PARAM F_stred 9.5kHz 10.5kHz 50Hz

Nastavení teploty

.TEMP <teplota ve ⁰C>*

Modely polovodičových i některých dalších součástek v PSpice respektují teplotní závislosti. Implicitně je nastavena teplota 27 ⁰C, příkazem .TEMP ji však lze změnit. Může se uvést i několik teplot, pak je celá analýza prováděna postupně pro všechny uvedené teploty a do výstupního souboru je předávána skupina křivek. Příkaz je tak obdobný příkazu .STEP, ale pro teplotu je jednodušší.

Příklad: .TEMP 0 27 125 ; tentýž výsledek jako .STEP TEMP LIST 0 27 125

Citlivostní analýza

.SENS <výstupní proměnná> *

Obvod je linearizován kolem pracovního bodu a pro každou vyjmenovanou výstupní proměnnou je vypočítána citlivost na změnu všech parametrů rezistorů, nezávislých zdrojů, spínačů, diod a bipolárních tranzistorů. Výsledky jsou vypsány formou tabulky do souboru *.out. Výpis může být velmi rozsáhlý!

Statistická analýza (Monte Carlo)

.MC <počet průběhů> <analýza> <výstupní proměnná> <funkce> [volba]*

Analýza chování obvodu při náhodných změnách parametrů součástek. Proběhne mnohonásobná analýza, počet opakování je dán číslem *počet průběhů*, položka *analýza* udává typ analýzy (může být jen DC, AC, TRAN - zde vždy bez tečky!). Požadovaná analýza musí být specifikována příkazem .DC nebo .AC nebo .TRAN na jiném místě vstupního souboru *.cir. Sleduje se jeden výstup podle položky *výstupní*

proměnná. První analýza je provedena při jmenovitých hodnotách parametrů součástek, při každém dalším opakování analýzy jsou pozmeněny parametry u všech vybraných součástek. Počet opakování je omezen na 2000, pokud jsou výsledky ukládány do výstupního textového souboru **.out*, a na 400, pokud jsou ukládány do **.dat* a předány k zobrazení programu Probe. Ten je zobrazí jako skupinu křivek. Při náhodných změnách hodnot několika parametrů současně je nemožné podchytit všechny možné kombinace, nelze se spolehnout na to, že byl skutečně analyzován nejnepříznivější či nejpříznivější případ. Výsledky mají statistickou povahu a jsou tím průkaznější, čím větší je počet opakování - ten však nemůže být příliš velký, neboť analýza je pak velmi zdoluhavá. Zřejmě se musí volit kompromis. Aby se alespoň částečně omezil velký rozsah tabulek v **.out*, používá se třídící *funkce* v několika možných variantách. Výsledek každého opakování analýzy se nezaznamenává v celém průběhu (např. frekvenční charakteristika *.AC* nebo přechodný děj *.TRAN*), nýbrž jako jediné číslo, které charakterizuje odchylky od průběhu při jmenovitých hodnotách. Možné varianty jsou:

YMAX	...	absolutní hodnota největší odchylky
MAX	...	maximální hodnota celého průběhu
MIN	...	minimální hodnota celého průběhu
RISE_EDGE <práh>	...	hodnota nezávisle proměnné (frequency, time) při protnutí prahové hodnoty výstupní proměnná zdola nahoru
FALL_EDGE <práh>	...	totéž při protnutí shora dolů

Pro spolupráci s Probe třídící funkce nemá význam, neboť do souboru **.dat* jsou zaznamenány vždy celé průběhy každého opakování analýzy (proto je přípustný počet opakování mnohem menší), některá z výše uvedených variant však musí být vybrána (lze např. zadat YMAX). Pro detailní informaci je k dispozici tabulka hodnot v souboru **.out*, a zde je zadání třídící funkce důležité. Z možných variant položky [volba] má při využití Probe význam hlavně tvar OUTPUT ALL, kdy se ukládají všechna opakování. Bez uvedení OUTPUT je uložen jen výsledek analýzy při jmenovitých hodnotách !

Příklad: `.MC 20 AC VP(13,5) YMAX OUTPUT ALL`
; analýza AC musí být definována příkazem `.AC` někde na jiném řádku

Při opakování analýzy jsou u vybraných součástek **náhodně** obměňovány specifikované parametry. U těchto součástek musí být nově definován model příkazem `.MODEL`, jehož součástí je i specifikace tolerancí vybraných parametrů. Podle udané tolerance jsou při analýze parametry měněny zabudovanými softwarovými generátory náhodných čísel. V definici modelů lze stanovit, má-li mít

každý parametr svůj nezávislý generátor, nebo zda některé parametry u celé skupiny součástek mají být pozměňovány současně. Tím lze jednak simulovat situaci, kdy součástky jsou zcela nezávislé, a jednak situaci, kdy více součástek je vyrobeno integrovanou technologií na jednom substrátu, a tedy s přibližně shodnými změnami parametrů. Je možná i kombinace obojího. Rozdělení pravděpodobnosti generovaných čísel je implicitně nastaveno jako rovnoměrné (v mezích udané tolerance parametru), lze je ale případně změnit na Gaussovo nebo i na uživatelem definované.

Je zřejmá důležitost **modelů součástek**. Všimneme si proto podrobněji příkazu:

.MODEL <jméno modelu> [AKO: ...] <typ modelu>
+ ([<parametr> = <hodnota> [tolerance]]*)

Součástka, jejíž hodnota se má měnit, se vytvoří na základě některého ze standardních typů modelů, zabudovaných v PSpice: CAP, IND, RES, D, NPN, PNP, NJF (j-FET s N kanálem), PJF (j-FET s P kanálem), NMOS, PMOS, CORE (nelineární mag. jádro), VSWITCH, ISWITCH. Výčet není úplný, obsahuje ale nejčastější součástky. Nové jméno se musí lišit od typu modelu. U jednoduchých součástek (R, C, L, ...) je i model jednoduchý a parametrem je jen samotná hodnota součástky. U složitých součástek (D, Q, J, M, ...) jsou modely velmi složité, lze měnit řadu parametrů (např. proudový zesilovací činitel tranzistoru, apod.). Detaily je nutné vyčíst z příruček programu. Jako *parametr* se uvádí jen ten, který se má měnit (ostatní se přebírají automaticky ze standardního modelu), a to včetně tolerancí. Větší počet součástek (např. R) může používat tentýž model, pokud mají mít stejně definované tolerance. Pro různě definované tolerance musí být vytvořeny **různé** modely.

Tolerance může být uvedena jako: DEV[/gen][/dist] <hodnota> [%]
pro případ **nezávislé** součástky. *Gen* je číslo generátoru náhodných čísel. Pokud není číslo uvedeno, jsou generátory automaticky přiřazovány postupně, takže každé součástce s proměnným parametrem přísluší jiný generátor, a změny součástek jsou tudíž vzájemně nezávislé. Rozdělení pravděpodobnosti *dist* je UNIFORM (implicitní, nemusí se uvádět), GAUSS nebo uživatelem definované. *Hodnota* tolerance se udává v % nebo v absolutní hodnotě.

Tolerance může být uvedena i jako: LOT[/gen][/dist] <hodnota> [%]
pro případ **skupiny závislých** součástek se shodnými změnami parametrů. Pokud není číslo *gen* uvedeno, patří všechny součástky s tímto modelem do stejné skupiny a mění se shodně.

Uvedením různých čísel generátorů *gen* lze součástky záměrně seskupovat do více skupin se společnými změnami parametrů. Musí samozřejmě být vytvořen patričný počet modelů. Generátory pro DEV (max. 10) jsou vždy jiné než generátory pro LOT (max. 10). Kombinace DEV a LOT je možná a odpovídá skupině součástek s parametry, měnicími se souhlasně, ale i s jistými vzájemnými rozdíly (např. integrované pole tranzistorů, destičkové rezistory, apod.).

Příklad:

```
.MODEL RLOAD1 RES (R=1 DEV 5%)
.MODEL RLOAD2 RES (R=1 DEV 10%)      ; jiná tolerance, jiný model
.MODEL RLOAD3 RES (R=1 DEV 5% LOT 5%)
.....
; definice součástky s proměnným parametrem ve schématu:
R41 12 4 10k      ; běžná součástka, nebude se měnit
R16 3 0 RLOAD1 1.5k ; součástka se bude měnit, zde musí být model !
```

Poznámka k hodnotě součástky v souboru *.cir a v modelu:

U jednoduchých součástek (R, C, L, ...) se v definici součástky (popis schématu) uvádí hodnota. Ta je násobkem základní jednotky, definované v modelu součástky, kterým je buď model zabudovaný v PSpice, nebo model nový, vytvořený uživatelem. Aby se skutečná hodnota shodovala s hodnotou v definici součástky, doporučuje se v modelu uvádět hodnotu "1" (tj. 1Ω, 1F, 1H, ...)

Analýza nejhoršího případu (Worst Case)

.WCASE <analýza> <výstupní proměnná> <funkce> [volba]*

Proběhne mnohonásobná analýza, nejprve se jmenovitými hodnotami, a pak s proměnnými parametry. Na rozdíl od Monte Carlo však je v každém opakování změněn jen jeden parametr, a to o velmi malou hodnotu. Program tak postupně zjistí citlivosti na jednotlivé parametry. Počet opakování je tedy pevně dán, nelze jej měnit, a je mnohem menší než u MC. Na základě zjištěných citlivostí a jejich znamének je na závěr provedena poslední analýza s **krajními hodnotami** tolerancí volenými tak, že při třídící funkci YMAX nebo MAX je dosaženo maximální hodnoty výstupní proměnné, a při MIN minimální hodnoty výstupní proměnné.

Jednotlivé položky příkazu mají význam obdobný jako u MC. Jistým nedostatkem analýzy .WCASE je to, že nezaručuje nalezení skutečného extrému, pokud třídící funkce není monotónní. Extrém totiž nemusí vždy nastávat při krajních hodnotách tolerancí, ale někde jinde.

1.4. Standardní modely součástek v PSpice

Parametry, které lze v deklaraci modelu zadat, jsou vyznačeny tučným písmem a podtržením. Není nutné zadávat všechny. V PSpice jsou definovány jejich implicitní hodnoty, které vyhovují pro nejběžnější účely.

Jméno součástky: **C**
Typ modelu: **CAP**

Model kondenzátoru. Respektuje závislost kapacity na napětí a teplotě. Zadaná *hodnota* se změní podle vztahu pro kapacitu:

$$C = \langle \text{hodnota} \rangle \cdot \underline{\mathbf{KC}} \cdot (1 + \underline{\mathbf{VC1}} \cdot U + \underline{\mathbf{VC2}} \cdot U^2) [1 + \underline{\mathbf{TC1}}(T - T_{\text{nom}}) + \underline{\mathbf{TC2}}(T - T_{\text{nom}})^2]$$

kde T_{nom} je jmenovitá teplota (implicitně 27°C), T je momentální teplota, KC je násobitel kapacity. Implicitní hodnoty jsou 0, kromě $KC=1$.

Jméno součástky: **L**
Typ modelu: **IND**

Model induktoru. Respektuje závislost indukčnosti na proudu a teplotě. Zadaná *hodnota* se změní podle vztahu pro indukčnost:

$$L = \langle \text{hodnota} \rangle \cdot \underline{\mathbf{KL}} \cdot (1 + \underline{\mathbf{IL1}} \cdot I + \underline{\mathbf{IL2}} \cdot I^2) [1 + \underline{\mathbf{TC1}}(T - T_{\text{nom}}) + \underline{\mathbf{TC2}}(T - T_{\text{nom}})^2]$$

kde T_{nom} je jmenovitá teplota (implicitně 27°C), T je momentální teplota, KL je násobitel indukčnosti. Implicitní hodnoty jsou 0, kromě $KL=1$.

Jméno součástky: **R**
Typ modelu: **RES**

Model rezistoru. Respektuje závislost odporu na teplotě. Zadaná *hodnota* se změní podle vztahu pro odpor:

$$R = \langle \text{hodnota} \rangle \cdot \underline{\mathbf{KR}} \cdot [1 + \underline{\mathbf{TC1}}(T - T_{\text{nom}}) + \underline{\mathbf{TC2}}(T - T_{\text{nom}})^2]$$

kde T_{nom} je jmenovitá teplota (implicitně 27°C), T je momentální teplota, KR je násobitel odporu. Implicitní hodnoty jsou 0, kromě $KR=1$.

Jméno součástky: D
Typ modelu: D

Model diody. Popisuje statické i dynamické vlastnosti. Model má celkem 29 parametrů, je třeba nahlédnout do příručky k programu.

Jméno součástky: Q
Typy modelu: NPN, PNP

Model bipolárního tranzistoru NPN a PNP. Popisuje statické i dynamické vlastnosti. Model má celkem 59 parametrů, je třeba nahlédnout do příručky k programu. Prakticky jediným parametrem, který lze bez podrobné znalosti fyziky polovodičů použít, je **BF**, ("beta forward"), proudový zesilovací činitel. Implicitně je nastaven na hodnotu 100.

Jméno součástky: J
Typy modelu: NJF, PJF

Model unipolárního tranzistoru J-FET s kanálem N a s kanálem P. Popisuje statické i dynamické vlastnosti. Model má celkem 25 parametrů, je třeba nahlédnout do příručky k programu.

Jméno součástky: M
Typy modelu: NMOS, PMOS

Model unipolárního tranzistoru MOS FETs kanálem N a s kanálem P. Popisuje statické i dynamické vlastnosti. Lze zvolit jeden z 5 modelů. Modely mají až 70 parametrů, je třeba nahlédnout do příručky k programu.

Jméno součástky: S, W
Typy modelu: VSWITCH, ISWITCH

"VSWITCH" je model napětově řízeného spínače. Vnitřní odpor spínače se mění v závislosti na řídicím napětí. Je-li řídicí napětí větší než horní mez **VON**, je odpor spínače minimální, **RON**, a dále se již nemění. Je-li řídicí napětí menší než dolní mez **VOFF**, je odpor spínače maximální, **ROFF**, a dále se již nemění. Implicitní hodnoty jsou $VON=1\text{V}$, $VOFF=0$, $RON=1\Omega$, $ROFF=1\text{M}\Omega$. Mezi oběma mezemi je závislost odporu na řídicím napětí nelineární. Vztah je uveden v příručce k programu.

Spínač je velmi užitečná součástka a zpravidla postačí pracovat s jeho dvěma krajními stavy. Nedoporučuje se nastavovat VON příliš blízko k VOFF. Mohou nastat problémy s konvergencí, nebo alespoň zdlouhavý průběh simulace.

Pro "ISWITCH" platí přiměřeně totéž, je však řízen proudem. Implicitní hodnoty jsou ION=1mA, IOFF=0, RON=1Ω, ROFF=1MΩ.

Jméno součástky: K
Typ modelu: CORE

Model nelineárního magnetického materiálu, využíván v jedné z definic součástky "K". Detaily je třeba nalézt v příručce k programu.

Jméno součástky: T
Typ modelu: TRN

Model vedení. Detaily je třeba nalézt v příručce k programu.

2. Program Probe

Probe je program, určený k zobrazení výsledků simulace programem PSpice. Plně tak nahrazuje osciloskop, ale nabízí navíc ještě další funkce. Podobnost s osciloskopem bude zdůrazněna tím, že simulované průběhy budou v dalším textu nazývány "stopami". Vstupní informace je programu předávána v souboru *.dat.

Probe lze vyvolat jako samostatný program, nebo může být vyvolán z PSpice, nebo ze Schematics - odtud dokonce automaticky po úspěšně ukončené analýze.

Hlavní menu obsahuje položky **File, Edit, Trace, Plot, View, Tools, Window, Help**. Dále budou v maximální stručnosti probrány ty operace, které postačují k ovládní a plnému využití programu. Jsou seřazeny v pořadí, ve kterém se zahajuje práce s Probe. Systematický popis ovládní je uveden v příručkách programu. Je velmi vhodné tuto partii studovat při spuštění programu.

Pokud se zadávají proměnné a výrazy z klávesnice do příkazové řádky, nerozlišují se malá a velká písmena.

File/....:

Obsahuje většinou běžné příkazy, známé z jiných programů pod Windows.

Edit/...:

Umožňuje přesouvat jednotlivé stopy mezi grafy, přesouvat a editovat případné vložené objekty v grafech. Funkce jsou obdobné, jako v textových editorech (Word). Objekty lze též přesunout do textového editoru. Jedná-li se o vybranou stopu, přesune se číselná tabulka hodnot.

Trace/...

/Add - Výběr proměnných k zobrazení. V řádce Trace Commands lze použít i matematické výrazy. Proměnné jsou pak vypsány pod grafem zleva doprava v pořadí zadávání. Pokud chceme dodatečně výběr již v zobrazení zrušit, označí se myší jméno proměnné na řádku pod osou x a stiskne se DEL.

Plot/...

/X Axis Settings - V poli **Data Range** lze definovat měřítko a počátek. Volba **Auto Range** dává hodnoty, přepsané ze zadání analýzy v PSpice. V poli **Scale** se volí lineární nebo logaritmické měřítko. Jako proměnná X je v analýze .TRAN automaticky dosazována proměnná "time", v analýze .AC proměnná "frequency". Lze vyvolat **Axis Variable**, tam se volí jiná proměnná pro osu X. V řádce **X Axis Variable** lze zapsat i výraz. V poli **Processing Options** se může volit **Fourier Analysis**. Pak je prostřednictvím Fourierovy

transformace převeden časový děj do kmitočtové oblasti (osa "time" se změni na "frequency") a naopak.

/Add Y Axis - Přidá další svislou osu. Pak se musí v **Trace/Add** vybrat proměnná k zobrazení (ne opačně!). Svislé osy jsou pak uspořádány tak, že prvně zadaná je nejvíce vlevo. Proměnné jsou vypsány pod grafem po skupinách podle čísla osy. Pro pole **Data Range** a **Scale** platí totéž, jako v případě osy X. V **Axis Title** lze zadat nápis k ose Y, bude shora dolů.

/Delete Y Axis - Zruší svislou osu, označenou v grafu (>>), i přiřazené stopy (!). Značka se přesouvá myší.

/Add Plot - Přidá další graf nad původní. Pak se musí zadat proměnná k zobrazení.

/Delete Plot - Zruší vybraný graf. Označení SEL>>, přesouvá se myší.

/Unsync Plot - Umožní jiné měřítko na ose X. Až do té doby je stejné, jako u prvního grafu. Po tomto příkazu je třeba pro zvolený druhý graf použít příkaz **Plot/X Axis Settings**.

Tools/...

/Options - Mark Data Points umožní zobrazit body, ve kterých byly průběhy vypočítány.

/Cursor/Display - Zobrazení/zrušení kurzorů. Kurzor 1 se táhne po stopě levým tlačítkem myši, kurzor 2 pravým. Objeví se tabulka se souřadnicemi průsečíku horizontální a vertikální čáry kurzoru, a s rozdílem souřadnic x a y mezi oběma kurzory (kurz. 1 - kurz. 2). Pokud je v jednom obrázku více stop, umístí se kurzor na patřičnou stopu tím, že se myší vybere značka u jedné z proměnných ze seznamu pod osou x.

/Cursor/Peak - Vyhledání maxim, minim, atd.

/Label - Vložení objektů do grafu pro jeho lepší čitelnost. **Mark** (jen při zapnutém kurzoru) vloží do grafu souřadnice bodu, na který ukazuje kurzor. Jsou-li kurzory dva, vloží se souřadnice toho, který se naposled pohnul. Objekty mohou být vymazány (jen při zrušené funkci kurzoru) myší a DEL.

/Display Control - Zde je možno vyvolat nastavení Probe takové, jaké bylo při předcházejícím ukončení programu. K tomu je třeba vybrat **Last Session**. Jsou zachovány všechny grafy, včetně případných vložených objektů. Nastavení lze uložit i v průběhu práce a později se k němu vrátit. Stačí otevřít **Tools/Display Control**, napsat v řádce **New Name** jméno nastavení a vyvolat **Save**. Musí ale být otevřen ten soubor .dat, pro který bylo před tím nastavení uloženo. Všechna nastavení se postupně střádají do souboru **probe.dat**. Nepožívaná nastavení lze ze souboru vypustit.

/Copy to Clipboard - Je užitečné pro přenos obrázků prostřednictvím Schránky do textového editoru pod Windows.

Probe umožňuje provádět matematické operace s analogovými proměnnými při zadávání stop v příkazovém řádku v **Trace/Add**. Operandy jsou:

- Proměnné, předávané z PSpice.
- Konstanty, zadávané jako v PSpice.
- Čísla stop ve tvaru #1, #2, atd. Číslo odpovídá pořadí, v jakém byly voleny proměnné v grafu (a v tomto pořadí jsou také pod grafem vypsány zleva doprava). Pokud je více os Y, vztahuje se číslo k seznamu pod momentálně vybranou osou (ukazuje značka >>). Pokud je více grafů, vztahuje se číslo k seznamu v momentálně vybraném grafu (ukazuje značka SEL>>).

Aritmetické operátory jsou + - * / . Závorky () lze používat dle běžných pravidel. Jsou dovoleny matematické funkce v následující formě:

Funkce:	Význam:	Poznámka:
abs(x)	x	
sgn(x)		
sqrt(x)	$x^{1/2}$	
exp(x)	e^x	
log(x)	ln(x)	
log10(x)	$\log_{10}(x)$	
m(x)	abs. hodnota komplexního čísla	
p(x)	fáze	výsledek ve stupních
r(x)	reálná část x	
img(x)	imaginární část x	
g(x)	skupinové zpoždění x	výsledek v sekundách
pwr(x,y)	$ x ^y$	
sin(x)		x v radiánech
cos(x)		"
tan(x)		"
atan(x), arctan(x)		výsledek v radiánech
avg(x)	střední hodnota	
avgx(x,d)	střední hodnota	
rms(x)	efektivní hodnota	
db(x)	přepočet na dB	
min(x)	minimum reálné části x	
max(x)	maximum reálné části x	
d(x)	derivace x podle proměnné osy X	
s(x)	integrál z x podle proměnné osy X	

3. Program Schematics

Program Schematics slouží k vytváření schémat a jejich transformaci do souborů, předávaných PSpice - přede vším *.cir. Schéma je uloženo v souboru *.sch. Soubor *.cir obsahuje jen příkazy. Vlastní popis obvodu se všemi deklaracemi je v souboru *.net (netlist), vkládaném jako .INC *.net. Konverze mezi jmény součástek a jejich vývodů ve Schematics, a jmény součástek a uzlů v PSpice je v souboru *.als. Příkazem **Analyse/Simulate** jsou všechny soubory generovány automaticky a předány PSpice. Soubory lze upravovat běžným textovým editorem a zpracovat samostatně vyvolaným PSpice mimo prostředí Schematics.

Výhodou grafického editoru je odstranění omylů v číslování součástek a v syntaxi deklarací a příkazů. Též je současně vytvářena výkresová dokumentace. Nevýhodou je existence velkého množství grafických symbolů, kterými se musí vyjadřovat variabilita některých deklarací v PSpice. Výmluvným příkladem jsou zvláště zdroje E a G, jejichž různé naprogramování (jednou řádkou deklarace !) vede na několik desítek grafických symbolů.

Pro dobré porozumění a plné využití PSpice je i v prostředí grafického editoru nutné znát syntaxi deklarací a příkazů. Je užitečné sledovat obsah souboru *.out ve vztahu k různým variantám schématu, k použitým součástkám a k volbě analýz.

Dále jsou v maximální stručnosti uvedeny ty operace, které postačují k ovládní a plnému využití programu. Systematický popis ovládní je uveden v příručkách programu. Je velmi vhodné tuto partii studovat při spuštění programu.

3.1. Základní operace se Schematics

Při tvorbě schématu lze všechny funkce ovládat myší pomocí menu, pro rychlou práci je ale vhodnější využívat současně i klávesnici a některé příkazy zadávat z ní. Schéma se skládá z objektů - součástek, spojů vodičem, spojů sběrnicí, textů, atd. Objekty lze vkládat a schéma lze kdykoliv dodatečně pozměňovat (editovat). Objekty k editaci je nutné označit. Jednoduché objekty, jako součástky, spoje, text, se označí kliknutím 1x levým tlačítkem myši. Označený objekt bude vyznačen červeně. Skupina objektů se označí buď postupným výběrem jednotlivých objektů při stisknutí klávese Shift, nebo tak, že pohybem myši při stisknutém levém tlačítku se vytvoří rámeček. Objekty v něm (jen jsou-li uvnitř celé !) jsou vyznačeny červeně. K posouvání objektů není zvláštní příkaz, využívá se myš. Objekt se označí a při stisknutém levém tlačítku se táhne.

File/... Operace se soubory.

/New, Open, Close, Save (Ctrl+S), Save as, mají zcela stejný význam, jako u jiných programů pod Windows. **Save** uloží schéma do souboru *.sch . Je možné otevřít několik souborů a přepínat je prostřednictvím **Window**.

/Print - Zde lze nastavit všechny parametry pro tisk. Důležitá je volba **Auto-fit** nebo **User-definable**. V prvním případě bude měřítko obrázku pozměněno automaticky tak, že se optimálně vyplní tisknutá stránka. Ve druhém případě lze podle potřeby volit měřítko. Je přitom třeba vzít v úvahu nastavení rozměru kreslicí plochy v **Options/Page Size**.

/Current Errors - Zobrazí chybová hlášení z posledního běhu programu.

Edit/... Editace schématu.

/Copy (Ctrl+C), Paste (Ctrl+V), Cut (Ctrl+X), Del, pracuje s vybranými objekty jako běžné textové editory (Word).

/Undelete (Ctrl+U) - Velmi užitečné.

/Rotate (Ctrl+R) - Otočení vybraného objektu o 90⁰ doleva. Platí pro jednoduché objekty i pro rámeček.

/Flip (Ctrl+F) - Zrcadlové otočení vybraného objektu.

Copy to Clipboard - Je užitečné pro přenos schématu prostřednictvím Schránky do textového editoru pod Windows. Oblast k přenesení musí být předem vybrána rámečkem.

/Attributes (též **2x kliknout** na levé tlačítko nad objektem) - Umožní změnu hodnot a vlastností vybraného objektu. Změna je platná jen pro editované schéma, nezanese se do knihovního souboru součástek. Výběr objektu k editaci je nejlépe provádět myší. Pokud je objektem součástka, může se editovat jako celek tím, že se při výběru nastaví kurzor na její střed (pak se objeví tabulka se všemi příznaky součástky, počátečními podmínkami, atd.), nebo lze editovat jen jednotlivé příznaky (hodnoty, označení, ...) tak, že se kurzor nastaví na ně.

/Label (též **2x kliknout** na levé tlačítko nad objektem) - Umístí označení k vybranému vodiči, sběrnici, vývodu. Umožní též editaci již existujícího označení.

/Model - umožní změnit model pro součástku, předem vybranou kurzorem. V otevřeném dialogovém okně volba **Change Model Reference** umožní změnit jméno modelu, volba **Edit Instance Model (Text)** pak vytvořit nový model přímo jako text příkazu .MODEL . Po **OK** program uloží model do knihovny modelů, vytvořené speciálně pro toto schéma.

Draw/... Vytváření schématu.

/Get New Part (Ctrl+G) - Vložení součástky. Na výzvu se buď může rovnou napsat jméno součástky, je-li známo (R, C, atd.), nebo se volí Browse. Pak se volí jedna z knihoven součástek (*.slb) a v ní se součástka najde. Táhne se myší a umístí levým tlačítkem. Příkaz trvá až do ukončení pravým tlačítkem, lze umístit více totožných součástek.

/Wire (Ctrl+W) - Propojování vodičem. Levým tlačítkem myši se udávají začátky jednotlivých úseků, pravým se příkaz ukončí. Vodič lze připojit ke druhému tak, že se na něm začne nebo ukončí. Vodiče lze klást křížem bez propojení. Mají-li být v křížení propojeny, stiskne se při protínání levé tlačítko myši.

/Draw Bus (Ctrl+B) - Propojování sběrnicí. Platí stejná pravidla, jako u Wire. Vodič lze připojit ke sběrnici tak, že se na ní začne nebo ukončí.

/Text (Ctrl+T) - Vložení textu.

/Space (mezera) - Opakování posledního příkazu.

View/... Zobrazení.

/Redraw (Ctrl+L) - Obnovení kresby. Po některých editačních příkazech se porušily čáry, rastr, atd.

/In (Ctrl+I) - Zvětšení obrázku. Na výzvu je třeba umístit kurzor na budoucí střed obrázku. K dalšímu zvětšování lze opakovaně stisknout **Space** (mezera).

/Out (Ctrl+O) - Zmenšení obrázku. Platí totéž, jako u **In**.

/Fit (Ctrl+N) - Schéma přibližně zaplní zobrazenou stránku.

Options/... Volby.

/Page Size - Nastavení rozměru kreslicí plochy (nikoliv rozměru obrázku z tiskárny!).

/Editor Configuration - volba **Library Settings** umožní upravovat soubor knihoven symbolů .slb, lokálních nebo globálních (s "*"). Seznam knihoven bude uložen do inicializačního souboru msim.ini ve Windows. Tam jej lze případně upravit také pomocí textového editoru. Volba **Fonts** umožní upravit tvar a rozměr znaků na obrazovce i při tisku schématu.

Analysis/... Volby analýzy.

/Annotate - Nové očíslování součástek po předchozích editacích schématu a rušení součástek.

/Electrical Rule Check - Kontrola formálních chyb zapojení. Nenajde všechny možné chyby, ale je rychlá. Úplnou kontrolu chyb provádí až PSpice.

/Simulate (F11) - Vyvolá PSpice. Ten po úspěšné simulaci vygeneruje textový soubor *.out a binární soubor *.dat, který je předáván Probe. Při vhodném nastavení v Probe Setup může být Probe vyvolán po simulaci automaticky.

/Run Probe (F12) - Vyvolá Probe.

/Probe Setup - Zde nejdůležitější volby jsou **Show All Markers** (v Probe budou automaticky zobrazeny všechny stopy dle rozmístěných měřících bodů), **Show Selected Markers** (stopy bude nutné vybrat), **At Markers Only** (v souboru *.dat budou jen data v měřících bodech) a **All** (*.dat je v plném rozsahu, zpravidla zbytečně velkém).

/Setup - Zde je možné volit jednotlivé druhy analýz a zadávat pro ně bližší specifikace. V otevřeném dialogovém okně lze volit druhy analýzy, a to jednu nebo více současně. Volbou se otevře další dialogové okno s hodnotami, které odpovídají syntaxi příkazu PSpice pro analýzu - viz předchozí text. Před volbou Parametric je třeba ve schématu vytvořit parametry (viz další text).

/Library and Include Files - Zde se specifikují soubory, které budou zapsány v příkazech .INC a .LIB v souboru *.cir. Soubory označené jako "*" budou platné pro všechna schémata (globální), bez označení jen pro momentálně zpracovávané schéma (lokální). Soubor "nom.lib*" obsahuje odkazy na všechny dostupné knihovny. Je textový a může být rozšiřován a upravován podle požadovaného nebo dostupného souboru knihoven.

/Examine Output - Zobrazí výstupní soubor *.out, ve kterém je popis schématu, posloupnost příkazů pro simulaci, výsledky výpočtu pracovního bodu a případná chybová hlášení.

Markers/... Měřící body.

Usnadňují následnou analýzu výsledků v grafické formě programem Probe. Nastavení měřících bodů se přenáší do souboru *.cir jako příkaz .PROBE s výstupními proměnnými dle vložených měřících bodů. Bez tohoto příkazu počítá PSpice napětí ve všech uzlech a proudy všemi součástkami, takže výsledný soubor *.dat může být velmi rozsáhlý a nastavení zobrazení v programu Probe velmi pracné. Pokud je předem jasné, co má být v obvodu analyzováno, nastaví se příslušné měřící body a do Probe je předáván podstatně omezený soubor hodnot. Měřící bod se nastaví myší tak, aby ukazoval na spoj či uzel, ne na součástku.

/Mark Voltage/Level (Ctrl+M) - Vložení měřícího bodu, napětí proti zemi.

/Mark Voltage Differential - Vložení měřícího bodu, rozdíl napětí.

/Mark Current into Pin - Vložení měřícího bodu, proud.

/Mark Advanced - Následuje nabídka několika dalších typů měřících bodů, vesměs pro AC analýzu.

3.2. Tvorba schémat pomocí Schematics

Pro snadné určení výstupních veličin (zvláště napětí), které jsou v některých příkazech pro analýzu nutné (.TF, .SENS), je vhodné označit uzly srozumitelnými symboly (label), případně čísly. Pokud uživatel sám neoznačí uzly, Schematics je očísluje, čísla však nejsou viditelná. Vhodnost měřících bodů pro omezení rozsahu a času simulace a pro automatické zobrazení v Probe již byla zmíněna. Číslování uzlů a nově vkládaných součástek je automatické, přírůstkové. Při editaci schématu, kdy se často ruší některé součástky, však dochází k mezeře v původním očíslování. Pro PSpice to sice není na závadu, pro účely technické dokumentace (a pro další využití schématu v jiných programech) to však vadí. Příkazem Analysis/Annotate se číslování upraví bez mezer.

Součástky pro vložení se vybírají buď přímo jmény pomocí klávesnice (což je pro nejpoužívanější typy ten nejrychlejší způsob), nebo z knihoven symbolů *.slb přes **Browse**. Knihovny jsou značně rozsáhlé a hledání zdržuje. Všeobecně platí, že každá součástka má svůj symbol v některé z knihoven *.slb. Pokud se nejedná o standardní součástku se zabudovaným modelem v PSpice (R,C, ...), je ještě definována jako model nebo podobvod v příslušné knihovně *.lib. Většinou se jedná o polovodičové součástky. Počet symbolů je však větší, než počet součástek. Je to tím, že jedna a tatáž velice univerzální součástka (např. E, G, V, I) se pro různé druhy analýz potřebuje v mnoha různých variantách. Všechny alternativní symboly jsou přitom nakonec přeloženy do téměř stejné deklaráce v netlistu.

Dále existují tzv. "pseudosoučástky". Vyjadřují příkazy nebo jejich části, které nemají obraz v reálné součástce. Příkladem jsou již zmíněné měřící body, což jsou části příkazu .PROBE. Velmi potřebný je symbol parametru, nutný pro parametrickou analýzu - viz další text.

Ke spojení analogových zemí bez kreslení vodičů slouží symbol analogové země "AGND" (překládaný do netlistu jako uzel 0). Ke spojení vzdálených bodů bez vodičů může sloužit symbol svorky "BUBBLE" nebo "GLOBAL". Shodně pojmenované svorky (Edit/Label/*jméno*) jsou považovány za spojené do uzlu a v netlistu je uzel pojmenován jako *jméno*.

Některé často používané součástky

Pasivní součástky (v analog.slb):

R, C, L, XFORM_LINEAR (transformátor), T, TLOSSY (vedení se ztrátami)

Globální uzly, propojení (v port.slb):

AGND	Analogová zem
EGND	Zem kostry
BUBBLE, GLOBAL	Propojení
IF_IN, IF_OUT	Spojení s podobvodem

Modifikace zdrojů, odvozené od součástky V a I (v souboru source.slb):

VSRC	Odvozen od zdroje "V". Zdroj pro všeobecné použití, DC, AC, TRAN, pro napájení, atd. Kromě tohoto zcela univerzálního zdroje existují ještě mnohé další s omezenějšími možnostmi, ale jednodušším zadáním	
VDC	Zdroj ss. napětí pro napájení, analýzu .DC	
VAC	Zdroj pro analýzu .AC	
VEXP	Viz deklaraci "V". Exponenciální, pro .TRAN	
VPULSE	"	
VPWL	"	
VPWL_ENH	"	PWL s opakováním
VPWL_FILE	"	PWL, zadání ze souboru
VSFPM	"	
VSIN	"	
ISRC	Založen na zdroji "I". Další obdobně jako VSRC.	

Pseudosoučástky (v special.slb)

PARAM	Parametr
-------	----------

Zavedení parametru

Je nutné před volbou parametrické analýzy - viz příkaz .PARAM.

1. Do schématu se **vloží pseudosoučástka** PARAM. Dvojitým kliknutím levým tlačítkem nad objektem se otevře dialogové okno, ve kterém se vyplní NAME jako jméno parametru a VALUE jako hodnotu parametru. Lze vyplnit až tři parametry.
2. Ve všech součástkách, jejichž hodnotou má být parametr, se **označení hodnoty** změně na {*jméno parametru*}.
3. Pak lze **nastavit zadání** pro parametrickou analýzu.

Vytvoření podobvodu

Práce s podobvodou je velmi užitečnou vlastností PSpice. V textovém souboru je vytvoření podobvodu jednoduchou záležitostí (viz popis příkazu .SUBCKT), ale v grafickém editoru je třeba provést několik akcí:

1. **Vytvořit schéma** podobvodu, vstupy a výstupy opatřit svorkami IF_IN, IF_OUT.
2. **Uložit schéma** (File/Save).
3. **Vygenerovat podobvod** (Tools/Create Subcircuit).
4. **Vytvořit symbol** k danému podobvodu (File/Symbolize/...). Pokračuje dialogem, je vyžadováno jméno podobvodu a pak volba **knihovního souboru** *.slb k uložení symbolu. Lze buď vybrat již existující soubor, nebo jmenovat nový, který bude automaticky založen.
5. **Zpřístupnit nový knihovní soubor** (Options/Editor Configuration/Library Settings /Browse - následuje výběr knihovny a OK-Add Local a OK-OK). V Library Settings lze též odstranit již nepoužívané knihovny (Delete). Soubor na disku tím není dotčen, takže při případném omylu nedojde ke zničení knihovny - jen je třeba ji znovu zpřístupnit.

Při sestavování schématu se podobvody vybírají z knihovny *.slb (příkazy Draw/Get New Part) **jako kterékoliv jiné součástky**.

Práce s modely

Program umožňuje vytvářet a editovat nové modely na základě modelů již existujících. Modifikace se děje na kopiích modelů, vybíraných z knihoven. Nové modely tedy neporuší platnost modelů původních. Pro analýzy .MC a .WCASE jsou určeny pseudosoučástky s proměnnými parametry, jejichž symboly jsou v knihovně **breakout.slb**. Jimi se ve schématu nahradí původní pevné součástky (B, C, D, J, K, L, M, Q, R, S, T, W). Modely, zabudované v PSpice, totiž **nejsou přístupné** - modely součástek "breakout" jsou sice jejich kopiemi, ale jsou přístupné. Pomocí editoru modelů (Edit/Model) pak vytvoříme **nově pojmenovaný model**, založený na modelu "breakout" součástky. Principiálně je sice možné změnit podle potřeby i samotný model "breakout" součástky, ale ve schématu budeme pravděpodobně mít více součástek s různě proměnnými parametry a s různými tolerancemi. Pro ty budeme potřebovat **více modelů s různými jmény**. Tudíž je nutné tvořit nové modely, každý s jiným jménem, založené na modelu "breakout".

Funkční bloky pro modelování systémů

Tyto pseudosoučástky jsou souhrnně označovány jako ABM (Analog Behavioral Modeling) a jsou dostupné v knihovně abm.slb. Vesměs se jedná o **modifikace řízených zdrojů E** nebo **G** - viz jejich deklarace. Jedná se o velmi univerzální součástky. V textovém souboru *.cir postačí na takovou součástku jedna vhodně volená deklarace E či G, v grafickém editoru je zapotřebí několik desítek různých symbolů.

Pomocí řízených zdrojů lze modelovat analogové obvody tak, že místo detailního schématu se některé části nahradí modely o vhodném chování v časové i frekvenční oblasti. Lze tak sestavovat schémata zcela obecných regulačních obvodů, vyšetřovat časové odezvy zpětnovazebních systémů, filtrů, a pod. Kombinace součástek v PSpice s bloky ABM je možná s tím malým omezením, že vstupní proměnnou bloků ABM je napětí.

Práce s bloky ABM v prostředí Schematics je obdobná, jako s jinými součástkami - stejným způsobem se vybírají, umísťují a propojují. Stejně zásady platí pro uzly, měřící body a společnou zem. Po umístění lze blok editovat v otevřeném dialogovém okně stejně jako jinou součástku.

Bloky, realizující matematické funkce podle seznamu na začátku kapitoly:

ABS, ARCTAN, ATAN, COS, EXP, LIMIT, LOG, LOG10, PWR, PWRS, SIN, SQRT, TABLE, TAN.

Dále je zavedeno několik nových funkcí:

CONST	konstanta
SUM	sčítačka, 2 vstupy
MULT	násobička, 2 vstupy
GAIN	ideální zesilovač
DIFF	odčítačka, 2 vstupy
GLIMIT	zesilovač s omezením na výstupu
SOFTLIM	omezovač s pozvolným nasazením omezení, tanh(x)
DIFFER	ideální derivátor
INTEG	ideální integrátor

Bloky, realizující filtry (Čebyševovy) a obecnou frekvenční charakteristiku:

LOPASS, HIPASS, BANDPASS, BANDREJ, FTABLE.

Blok pro Laplaceovu transformaci:

LAPLACE