

**VŠB - Technická univerzita Ostrava**  
**Fakulta elektrotechniky a informatiky**

**BAKALÁŘSKÁ PRÁCE**

**2015**

**Nováček Martin**

**VŠB - Technická univerzita Ostrava**  
**Fakulta elektrotechniky a informatiky**  
**Katedra elektroniky**

**Analýza vlastností a použití speciálních bloků z knihoven  
OrCAD**

**Analysis and the Use of Special Blocks from OrCAD  
Libraries**

**2015**

**Nováček Martin**

## Zadání bakalářské práce

Student: **Martin Nováček**  
Studijní program: B2649 Elektrotechnika  
Studijní obor: 2612R003 Aplikovaná elektronika  
Téma: **Analýza vlastností a použití speciálních bloků z knihoven OrCAD.**  
**Analysis and the Use of Special Blocks from OrCAD Libraries.**

Zásady pro vypracování:

1. Popište obsah knihovny ABM, uveďte možnosti jejího použití v analogových simulacích.
2. Na jednoduchých příkladech demonstруйте použití vybraných bloků.
3. Podle požadavků vedoucího proveďte simulaci jednoduchého obvodu za pomoci bloků ABM.
4. Získané znalosti shrňte do sylabu pro snadné použití ve výuce.

Seznam doporučené odborné literatury:


Dle doporučení vedoucího bakalářské práce.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.


Vedoucí bakalářské práce: **Ing. Tomáš Pavelek, Ph.D.**

Datum zadání: 01.09.2014

Datum odevzdání: 07.05.2015

  
\_\_\_\_\_  
doc. Ing. Petr Palacký, Ph.D.  
vedoucí katedry



  
\_\_\_\_\_  
prof. RNDr. Václav Snášel, CSc.  
děkan fakulty

## Prohlášení studenta

Prohlašuji, že jsem celou bakalářskou práci včetně příloh vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 7.5.2015.....



.....

Podpis studenta

## **Poděkování**

Především bych chtěl poděkovat vedoucímu mé bakalářské práce panu Ing. Tomáši Pavelkovi, Ph.D. za pomoc a nápady při tvorbě této bakalářské práce. Dále bych chtěl poděkovat panu doc. Ing. Ivovi Neborákovi, CSc. za připomínky v oblasti simulací.

## **Abstrakt**

Bakalářská práce se zabývá popisem knihovny Analog Behavioral model v programu OrCAD Pspice. V této knihovně se nachází dva typy prvků. Control systém parts a equivalent parts. Prvky z této knihovny jsou detailně rozebrány a popsány jejich vlastnosti a parametry. V další části práce jsou jednoduché příklady aplikací. Je popsán postup matematické analýzy stejnosměrného motoru za použití součástek z knihovny ABM. Tato simulace je následně porovnána se simulací v programu Matlab Simulink. Dále jsou zmíněny další možnosti simulace. V příloze je pak vytvořen Sylab popisující vytvoření PI regulátoru.

## **Klíčová slova**

OrCAD Pspice, Analog Behavioral model, simulace, matematický model, stejnosměrný motor, Matlab Simulink, PI regulátor, simulace matematických modelů v OrCAD pspice

## **Abstract**

This bachelor thesis is focused on description of Analog Behavioral model library in OrCAD Pspice programme. There are two types of components in the library. Control system and equivalent parts. The property and parameters of these parts are described and analysed minutely. The second part of the bachelor consists of simple examples of application. It describes mathematical model of DC motor with application of components from library ABM. This simulation is subsequently compared with simulation in Matlab Simulink programme. The other possibilities of simulation are also mentioned. Sylab describing formation of PI regulator is created and included in the enclosure.

## **Key words**

OrCAD Pspice, Analog Behavioral model, simulation, mathematical model, DC motor, Matlab Simulink, PI regulator, simulation mathematical model in OrCAD Pspice

## Seznam použitých symbolů

Symbol	Jednotka	Význam Symbolu
$C$	F	Kapacita
$c\phi$	V·s	Budící konstanta stroje
$F(p)$		Přenos soustavy
$f_p$	Hz	Frekvence lomu
$f_s$	Hz	Frekvence definující útlum
$I$	A	Proud
$I_a$	A	Proud kotevním obvodem
$J_c$	kg·m <sup>2</sup>	Celkový moment setrvačnosti
$J_m$	kg·m <sup>2</sup>	Moment setrvačnosti motoru
$L_a$	H	Indukčnost kotvy
$M_e$	N·m	Moment motoru
$M_z$	N·m	Moment zátěže
$\eta$	%	Účinnost
$n_n$	ot/min	Jmenovité otáčky
$p$		Laplaceův operátor
$P_n$	W	Jmenovitý výkon
$R_a$	$\Omega$	Odpor kotvy
$T_a$	s	Elektromagnetická časová konstanta
$U$	V	Napětí
$U(p)$		Obraz vstupu
$U_a$	V	Napětí kotevního obvodu DC motoru
$U_C$	V	Napětí na kapacitoru
$U_i$	V	Indukované napětí
$U_R$	V	Napětí na rezistoru
$Y(p)$		Obraz výstupu
$\omega$	s <sup>-1</sup>	Okamžitá úhlová rychlost
$\omega_n$	s <sup>-1</sup>	Jmenovitá úhlová rychlost

# 1 Obsah

1	Obsah.....	- 1 -
2	Úvod.....	- 3 -
3	Knihovna Analog Behavioral Models.....	- 4 -
3.1	Popis jednotlivých součástí.....	- 4 -
3.1.1	Control system parts.....	- 4 -
3.1.2	Equivalent parts.....	- 19 -
4	Příklady.....	- 23 -
4.1	Laplaceova transformace.....	- 23 -
4.2	Dolní propust.....	- 25 -
4.3	Horní propust.....	- 25 -
4.4	Pásmová propust.....	- 25 -
4.5	Pásmová zadrž.....	- 26 -
4.6	Zadání prvku ETABLE.....	- 27 -
5	Matematický model stejnosměrného motoru.....	- 28 -
5.1	Sestavení matematického modelu.....	- 28 -
5.2	Simulovaný stejnosměrný motor.....	- 30 -
5.2.1	Výpočet parametru pro matematický model.....	- 31 -
5.3	Simulace DC motoru v OrCAD.....	- 31 -
5.3.1	Blokové schéma motoru v OrCAD.....	- 32 -
5.3.2	Výsledné průběhy.....	- 33 -
5.4	Simulace DC motoru v Matlab Simulink.....	- 33 -
5.4.1	Blokové schéma motoru v Simulinku.....	- 34 -
5.4.2	Výsledné průběhy.....	- 35 -
5.5	Další možnost simulace.....	- 35 -
5.6	Porovnání a závěr.....	- 37 -



6	Závěr .....	- 38 -
7	Seznam příloh.....	- 40 -

## 2 Úvod

Tato bakalářská práce se bude zabývat popisem knihovny ABM v simulačním programu OrCAD. V simulačním programu OrCAD a jeho nadstavby PSpice lze využít pro simulování elektronických obvodů takzvanou knihovnu ABM (Analog Behavioral Models). Tyto prvky umožňují nahradit část elektronického schématu funkčním blokem, který obsahuje matematický popis dané části obvodu. Při složitém elektrotechnickém schématu se tato knihovna nevyužívá pouze ke zjednodušení schématu, ale také ke zkrácení doby simulace. Při použití ABM modelu se čas simulace mnohonásobně zmenší, jelikož funkce je definována matematickým popisem. ABM model se dá použít i při prvotním navrhování elektronického obvodu nebo také může nahradit část fungujícího elektronického obvodu složeného z reálných součástek. V této práci se nejprve seznámíme s popisem funkčnosti jednotlivých prvků z knihovny ABM. Tyto prvky jsou rozděleny do dvou skupin. V další kapitole si ukážeme jednoduché příklady použití jednotlivých prvků. Dále se v práci nachází matematický model stejnosměrného motoru sestavený z prvků knihovny ABM. Stejný model je nasimulován i v programu Matlab Simulink. Jednotlivé simulace jsou porovnány. V práci je zmíněna také další možnost simulace takzvaná kosimulace za pomoci bloku SLPS. V příloze je pak vložen Syllabus, ve kterém je popsána simulace PI regulátoru.

### 3 Knihovna Analog Behavioral Models

V knihovně se nachází dva základní druhy bloků. Jsou to prvky s názvem Control system parts. Tyto prvky lze spojovat bez zátěže a bez vstupního odporu. Dalším druhem jsou takzvané Equivalent parts, které se chovají buď jako E prvky, které mají napěťový výstup, nebo jako G prvky s proudovým výstupem.

V knihovně nalezneme spoustu různých funkčních bloků, které nám reprezentují různé funkce. Nalezneme zde bloky například pro základní matematické operace přes konstanty až po bloky, které nám umožní simulovat například dolní propust. Konkrétně se k jednotlivým součástkám a jejich funkcím dostaneme v následujícím textu. Tato kapitola vychází z [1], [2], [3]

#### 3.1 Popis jednotlivých součástek

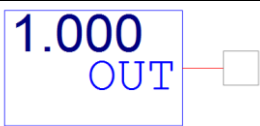
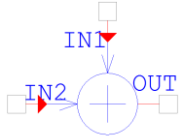
##### 3.1.1 Control system parts

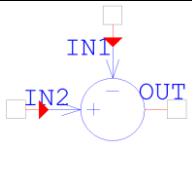
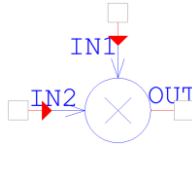
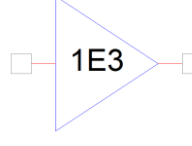
Prvky nazývané jako Control system parts obsahují jeden vstup a jeden výstup. K následnému spojování není nutno vkládat zátěže ani vstupní odpory. Napětí je měřitelné proti nulovému potenciálu (země z knihovny capsym.lib). Tyto prvky můžeme dále dělit na základní komponenty, limity, expresní funkce, matematické funkce, tabulky, Laplaceovy transformace a na Čebyševovy filtry.

##### 3.1.1.1 Základní komponenty

Mezi základní komponenty se zařazují bloky, které nám definují konstantu, sumu, násobení, zesílení, a rozdíl. Popis jednotlivých schématických značek a parametrů je zobrazen v následující tabulce č. 1

Tabulka č. 1 - přehled základních komponent

Název	Schématická značka	Parametr	Popis
CONST		VALUE	Výstupní konstantní napětí specifikované parametrem VALUE
SUM			Sčítá dva vstupní signály IN1 a IN2

<b>DIFF</b>			Odečítá dva vstupní signály IN1 a IN2
<b>MULT</b>			Násobí dva vstupní signály IN1 a IN2
<b>GAIN</b>		<b>GAIN</b>	Násobí vstupní signál parametrem GAIN

### 3.1.1.2 Limity

Limity umožňují omezit výstupní hodnotu v definovaném rozsahu. V knihovně ABM jsou k dispozici tři druhy limit.

#### LIMIT

Prvek LIMIT udržuje výstupní signál v rozmezí nastavených hodnot HI a LO.

#### Parametry:

HI- horní limit

LO- dolní limit

#### GLIMIT

Prvek GLIMIT vynásobí vstupní signál definovaným zesílením GAIN. Výstupní signál pak udržuje v rozmezích nastavených hodnot HI a LO.

#### Parametry:

GAIN- konstantní zesílení

HI- horní limit

LO- dolní limit

## SOFTLIMIT

Prvek SOFTLIMIT má stejné vlastnosti jako prvek GLIMIT s tím rozdílem že omezení není provedeno nespojitě, nýbrž spojitě. Rozdíl mezi prvkem GLIMIT a SOFTLIMIT je znázorněn na obrázku č. 3.1.1.

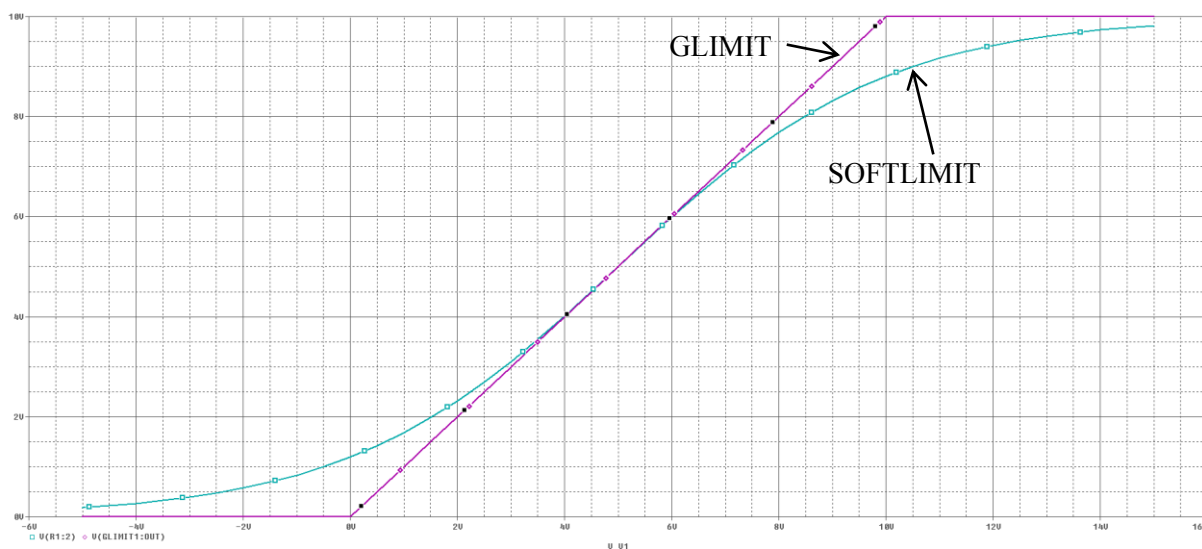
### Parametry:

GAIN- konstantní zesílení

HI- horní limit

LO- dolní limit

A, B, V- vnitřní proměnné pro definování přechodu



Obr. č. 3.1.1 - graf přechodu prvku Softlimit

### 3.1.1.3 Expresní funkce ABM

Expresní funkce nám umožňují plnit libovolnou matematickou rovnici, kterou zadáme do parametru. Rozdíly mezi jednotlivými prvky jsou v množství vstupních hodnot a v realizaci výstupní svorky (proudový nebo napěťový výstup).

#### ABM

Tento blok nedisponuje žádným vstupem, obsahuje pouze jeden výstup. Do bloku můžeme zadat hodnotu napětí, kterou chceme na výstupu. Tento blok se chová jako napěťový zdroj.

## **ABM/I**

Tento blok je obdobou bloku ABM s tím rozdílem, že tento blok má proudový výstup.

## **ABM1**

U tohoto bloku máme jeden vstup a jeden výstup. Na výstupu máme hodnotu, která odpovídá matematickému výrazu v parametru. Můžeme tedy vstupní napětí násobit, přičítat k němu konstanty a provádět základní matematické operace. Defaultně je nastavena hodnota parametru na:  $(V(\%IN) * 100)/1000$ . Tento parametr znamená, že vstupní napětí se vynásobí 100 a následně vydělí 1000.

## **ABM1/I**

ABM1/I je obdobou bloku ABM1. Má jeden vstup a dva výstupy (proudový výstup). Přivedená hodnota proudu se upraví podle zadaného parametru a přenesou se na výstup ABM1/I.

## **ABM2**

Blok plní stejnou funkci jako blok ABM1 s tím rozdílem, že na něj lze přivést dva signály, které se mohou upravovat. Tyto signály se sloučí do jednoho.

## **ABM2/I**

Má dva vstupní piny, vstupní hodnoty proudu se sloučí za pomoci matematického vztahu uvedeného v parametru.

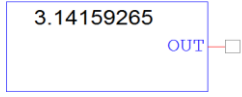
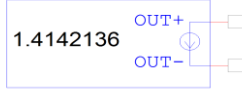
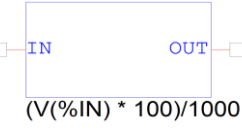
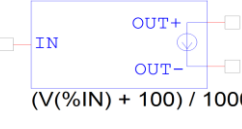
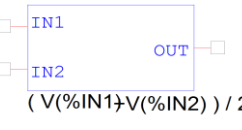
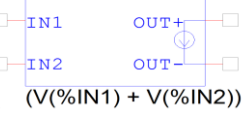
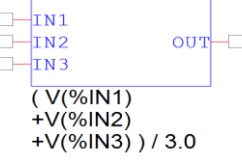
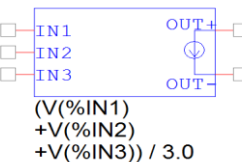
## **ABM3**

Blok má tři vstupní signály, ze kterých vznikne jeden výstupní.

## **ABM3/I**

Blok má tři vstupní signály, které se přenesou na jeden proudový výstup.

Tabulka č. 2 - přehled expresních funkcí

Název	Schématická značka	Počet vstupů	Počet výstupů	Druh výstupu
ABM		0	1	napěťový
ABM/I		0	1	proudový
ABM1		1	1	napěťový
ABM1/I		1	1	proudový
ABM2		2	1	napěťový
ABM2/I		2	1	proudový
ABM3		3	1	napěťový
ABM3/I		3	1	proudový

#### 3.1.1.4 Matematické funkce

Další části knihovny ABM jsou takzvané matematické funkce. Matematické funkce jsou založeny na druhu součástky „E“. Některé mají jeden nebo více vstupů. Vstupní signál je pak upraven definovaným matematickým výrazem. Takto upravený vstupní signál je přenesen na výstup.

##### ABS

Tento blok nám plní matematickou funkci absolutní hodnoty. Při přivedení záporného průběhu se na výstupu objeví signál stejného tvaru a stejné velikosti s kladným znaménkem. Výstupní výraz  $y = |x|$ .

##### SQRT

Jeden vstup, jeden výstup. Na výstupu je výraz  $y = \sqrt{x}$ .

##### PWR

Na výstupu je výraz  $y = |x^{EXP}|$ . Parametrem je EXP.

##### PWRS

Na výstupu je výraz  $y = x^{EXP}$ . Parametrem je EXP.

##### LOG

Přirozený logaritmus. Na výstupu je výraz  $y = \ln x$ .

##### LOG10

Logaritmus se základem 10. Na výstupu je výraz  $y = \log x$ .

##### EXP

Exponenciální funkce se základem e. Na výstupu je výraz  $y = e^x$ .

##### SIN

Vypočítává sinus, kde je na vstupu přiveden úhel v radiánech.

##### COS

Vypočítává cos, kde je na vstupu přiveden úhel v radiánech.

##### TAN

Vypočítává tg, kde je na vstupu přiveden úhel v radiánech.



## ARCTAN, ATAN

Vrací úhel v radiánech.



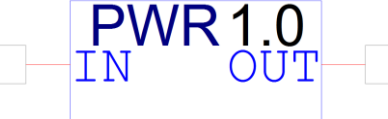
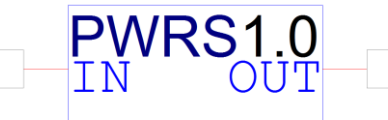

## INTEGRÁTOR INTEG








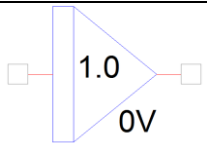
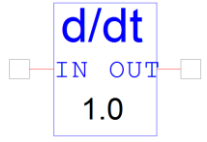
Integrátor má dva nastavitelné parametry. Parametr IC (initial condition) - počáteční stav na výstupu integrátoru a parametr GAIN - který nám udává zesílení integrátoru. Integrátor má proudový vstup. V modelu je vložen také matematický model kapacitoru, díky kterému je možné nastavit počáteční podmínku.

## DERIVÁTOR DIFFER

Derivační člen DIFFER má pouze jeden nastavitelný parametr, a to parametr GAIN - jedná se o hodnotu zesílení. Má napěťový vstup.

Tabulka č. 3 - přehled matematických funkcí

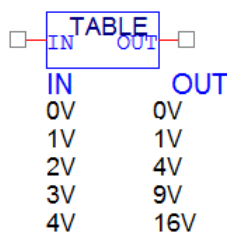
Název	Schéma	Funkce
ABS		$y =  x $
SQRT		$y = \sqrt{x}$
PWR		$y =  x^{\text{EXP}} $
PWRS		$y = x^{\text{EXP}}$
LOG		$y = \ln x$

<b>LOG10</b>		$y = \log x$
<b>EXP</b>		$y = e^x$
<b>SIN</b>		$y = \sin x$ $x = rad$
<b>COS</b>		$y = \cos x$ $x = rad$
<b>TAN</b>		$y = \tan x$ $x = rad$
<b>ARCTAN</b>		$y = \tan^{-1} x$ $y = rad$
<b>ATAN</b>		$y = \tan^{-1} x$ $y = rad$
<b>ITEG</b>		$y = GAIN \cdot \int x(t)dt + x_0$
<b>DIFFER</b>		$y = GAIN \cdot \frac{dx}{dt}$

### 3.1.1.5 Tabulky

Knihovna ABM obsahuje dva funkční bloky pro zadávání tabulek, TABLE a FTABLE. Tyto tabulky nám umožňují přenést na výstup hodnotu, která je definována pro určitou vstupní hodnotu. Příklad: Máme definovanou tabulku IN 1V, 2V; OUT 5V, 9V. Při hodnotě vstupního napětí 1V se na výstup přenesou hodnota 5V, při vstupním napětí 2V bude na výstupu hodnota 9V. To je základní princip tabulky.

#### TABLE



Obr. č. 3.1.2 - schématický značka prvku TABLE

Tabulka může obsahovat až 5 párů hodnot. Tyto hodnoty můžeme upravit buď přímo ve schématu, nebo v detailním nastavení prvku. V detailním nastavení se tyto parametry nazývají ROWn (kde  $n = 1, 2, 3, 4, 5$ ). Kdybychom potřebovali zadat více jak 5 hodnot, musíme upravit model prvku.

#### Úprava modelu prvku TABLE:

V nastavení prvku přidáme nový parametr ROW6 a v řádku PSpice Template upravíme stávající řádek:  $E^{@REFDES} \%OUT 0 TABLE \{V(\%IN)\} @ROW1 ?ROW2 | \backslash n+ @ROW2 | ?ROW3 | \backslash n+ @ROW3 | ?ROW4 | \backslash n+ @ROW4 | ?ROW5 | \backslash n+ @ROW5 |$

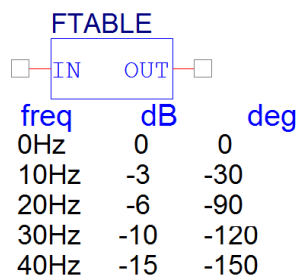
Přepíšeme na:

$E^{@REFDES} \%OUT 0 TABLE \{V(\%IN)\} @ROW1 ?ROW2 | \backslash n+ @ROW2 | ?ROW3 | \backslash n+ @ROW3 | ?ROW4 | \backslash n+ @ROW4 | ?ROW5 | \backslash n+ @ROW5 | ?ROW6 | \backslash n+ @ROW6 |$

A	
SCHEMATIC1 : PAGE1	
Color	Default
Designator	
Graphic	TABLE.Normal
ID	
Implementation	
Implementation Path	
Implementation Type	PSpice Model
Location X-Coordinate	290
Location Y-Coordinate	310
Name	INS4855
Part Reference	TABLE1
PCB Footprint	
Power Pins Visible	<input type="checkbox"/>
Primitive	DEFAULT
PSpiceOnly	TRUE
PSpiceTemplate	$E^{@REFDES} \%OUT 0 TABL$
Reference	TABLE1
ROW1	0V 0V
ROW2	1V 1V
ROW3	2V 4V
ROW4	3V 9V
ROW5	4V 16V
ROW6	
Source Library	C:\CADENCE\SPB_16
Source Package	TABLE
Source Part	TABLE.Normal
Value	TABLE

Obr. č. 3.1.3 - nastavení TABLE

## FTABLE

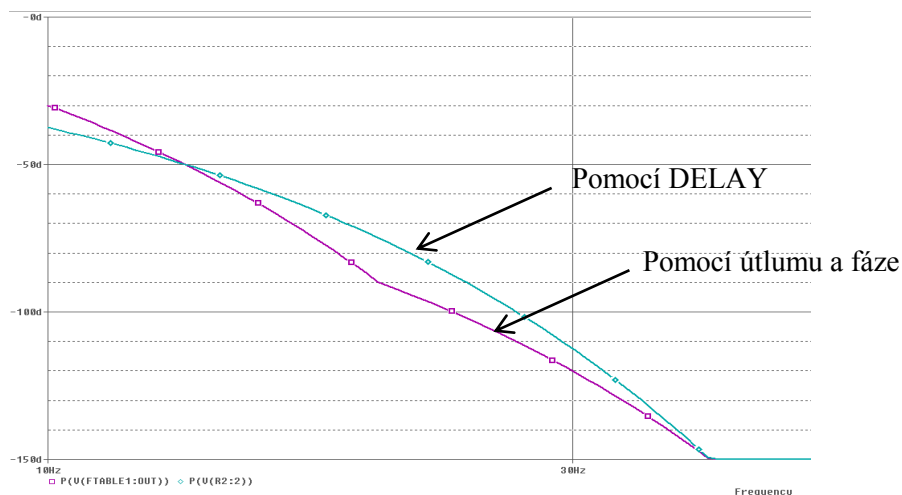


Obr. č. 3.1.4 - schématická značka prvku FTABLE

Funkce FTABLE nám umožňuje zadat frekvenční tabulku. Do tabulky zadáme konkrétní frekvenci s požadovaným útlumem a fázovým posuvem. Defaultní model FTABLE je definován pro zadání 5 frekvencí. Tyto frekvence jsou zadávány do parametru ROWn, kde  $n = 1, 2, 3, 4, 5$ . Tabulku lze zapsat také pomocí komplexních čísel, a to tak, že do parametru R\_I zadáme hodnotu YES. Tabulka se pak bude zadávat ve formátu: vstupní frekvence, reálná část, imaginární část.

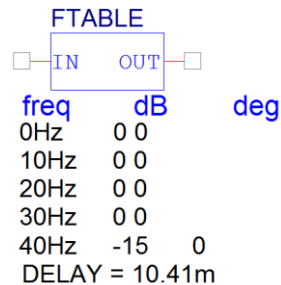
Parametr DELAY přidává ke skupině frekvenční tabulky hodnotu zpoždění. Toto zpoždění nahrazuje možnost zadávání jednotlivým frekvencím jejich útlum a fázi. Lze zadat pouze útlum v posledním parametru. Fázi určí parametr delay, který lze vypočítat dle následujícího vzorce. Tento parametr se využívá zejména v časové analýze, když nám vyskočí varování při spuštění simulace.

$$delay = \frac{\frac{fáze}{360}}{frekvence}$$



Obr. č. 3.1.5 - průběhy fáze při různých druzích zadávání

Na obrázku č. 3.1.5 je vidět rozdíl průběhu, kdy jeden průběh odpovídá FTABLU zadanému pomocí jednotlivých útlumů a fází (obr. č. 3.1.4), druhý průběh odpovídá FTABLE, kdy je použit parametr DELAY. Parametr DELAY byl vypočítán na hodnotu 10,41ms. Frekvenční tabulka byla zadána podle obr. č. 3.1.6.



Obr. č. 3.1.6 - zadání parametru za pomoci DELAY

Mezi další důležité parametry patří parametr MAGNITS, který udává, v jakých jednotkách zadáváme útlum. Jako defaultní hodnota je nastavena jednotka dB. Tuto jednotku můžeme změnit na velikost výstupního napětí tak, že do parametru zadáme „mag“. Přepočítání mezi mag a dB je zobrazeno v následující rovnici.

$$dB = 20 \cdot \log(mag)$$

Posledním parametrem, který je možno nastavit, je parametr PHAUSEUNITS. Tento parametr udává jednotky fázového posuvu. Defaultně je nastaven na stupně. Parametr je možno změnit na radiány a to tak, že do parametru vložíme hodnotu „rad“. Přepočítání mezi radiány a stupni je zobrazeno v následující rovnici.

$$rad = \frac{stupně \cdot \pi}{180}$$

Jak už jsem se zmínil, tabulka může obsahovat pouze 5 hodnot. Pro rozšíření je nutno přidat nový parametr s názvem ROW6 a upravit model prvku.

### Úprava modelu prvku FTABLE:

V řádku PSpice Template se nachází tento model:

```
E^@REFDES %OUT 0 FREQ {V(%IN)} ?R_I|R_I| ~R_I?MAGUNITS/@MAGUNITS/
?PHASEUNITS/@PHASEUNITS/ \n+@ROW1 ?ROW2| \n+ @ROW2| ?ROW3| \n+ @ROW3|
?ROW4| \n+ @ROW4| ?ROW5| \n+ @ROW5| ?DELAY|DELAY=@DELAY|
```

Přepíšeme ho na:

```
E^@REFDES %OUT 0 FREQ {V(%IN)} ?R_I|R_I| ~R_I?MAGUNITS/@MAGUNITS/  
?PHASEUNITS/@PHASEUNITS/| \n+@ROW1 ?ROW2| \n+ @ROW2| ?ROW3| \n+ @ROW3|  
?ROW4| \n+ @ROW4| ?ROW5| \n+ @ROW5| ?ROW6| \n+ @ROW6|  
?DELAY|DELAY=@DELAY|
```

V upraveném modelu jsou zvýrazněny úpravy, \n+ @ROW5| ?ROW6| bylo doplněno a následující člen \n+ @ROW5| byl přepsán na \n+ @ROW6|.

Jednotlivé frekvence se musí zadávat od nejmenší po největší. Při AC analýze se nesmí zadávat nulová frekvence, jinak dojde k chybě.

Výstupní hodnota také závisí na druhu analýzy. Při analýze stejnosměrného pracovního bodu (Bias point) a při stejnosměrné analýze (DC sweep) se na výstupu neuplatní parametry frekvenční tabulky a výstupní napětí bude odpovídat vstupní hodnotě napětí.

Tabulka č. 4 - zadávání parametru

FTABLE				FTABLE			
□ IN		OUT □		□ IN		OUT □	
freq	dB	deg		freq	dB	deg	
0Hz	0	0		0Hz	1	0	
10Hz	-3	-30		10Hz	0.707945	-0.524	
20Hz	-6	-90		20Hz	0.501187	-1.571	
30Hz	-10	-120		30Hz	0.316227	-2.094	
40Hz	-15	-150		40Hz	0.177827	-2.618	
ROW1=	0Hz	0	0	ROW1=	0Hz	1	0
ROW2=	10Hz	-3	-30	ROW2=	10Hz	0.707945	-0.524
ROW3=	20Hz	-6	-90	ROW3=	20Hz	0.501187	-1.571
ROW4=	30Hz	-10	-120	ROW4=	30Hz	0.316227	-2.094
ROW5=	40Hz	-15	-150	ROW5=	40Hz	0.177827	-2.618
DELAY=				DELAY=			
R_I=				R_I=			
MAGNITS=				MAGNITS=	mag		
PHAUSEUNITS=				PHAUSEUNITS=	rad		
FTABLE je zadání v defaultních jednotkách (dB, deg)				FTABLE je zadána ve velikosti výstupního napětí a fáze je zadána v radiánech			

### 3.1.1.6 Laplaceova transformace:

Funkce LAPLACE nám umožňuje zadat přenos v Laplaceově transformaci. Funkce LAPLACE má dva parametry. Parametr NUM pro zadání čitatele Laplaceova výrazu a parametr DENOM pro zadání jmenovatele Laplaceova výrazu.

Výstup z prvku závisí na zvoleném typu analýzy. Pro stejnosměrný pracovní bod (Bias point) a pro stejnosměrnou analýzu DC se na výstupu objeví signál se zesílením při nulové frekvenci. Toto zesílení vypočteme tak, že do Laplaceovy transformace dosadíme za operátor  $s=0$ .

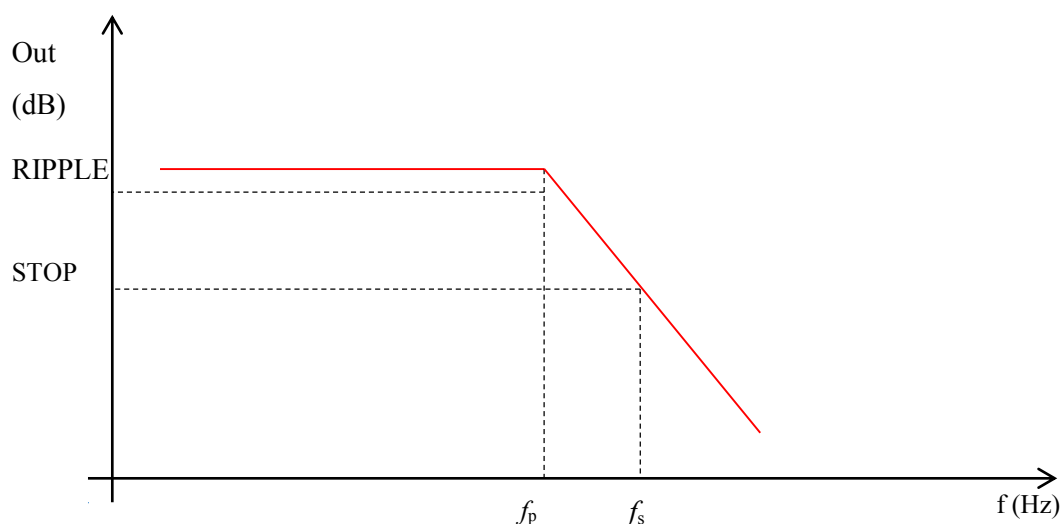
Při střídavé analýze je v transformaci provedena substituce za  $s = j\omega$ , kde  $\omega = 2\pi f$ . Při časové analýze je výstup konvolucí vstupního signálu a impulzní odezvy z Laplaceovy transformace.

### 3.1.1.7 Čebyševovy fitry

V knihovně ABM jsou prvky, kterými lze realizovat Čebyševovy filtry. Čebyševův filtr umožňuje filtrovat signál na základě nastavené frekvenční charakteristiky. Výstup filtru závisí na nastavené frekvenční charakteristice. Tyto prvky lze použít pouze při AC charakteristice, v použití DC nám nic nepropustí.

#### LOPASS (DOLNÍ PROPUST)

Dolní propust je definována dvěma zlomovými frekvencemi, které nám ohraničují pásmo propouštění a pásmo zadržené frekvence. Dolní propust nám propouští dolní frekvence, tzn., že parametr  $f_p$  (frequency pass) nám definuje, po jakou frekvenci filtr propustí. S parametrem frekvence  $f_s$  (frequency stop) souvisí parametr STOP, který nám říká, jaký bude minimální útlum při frekvenci  $f_s$ .

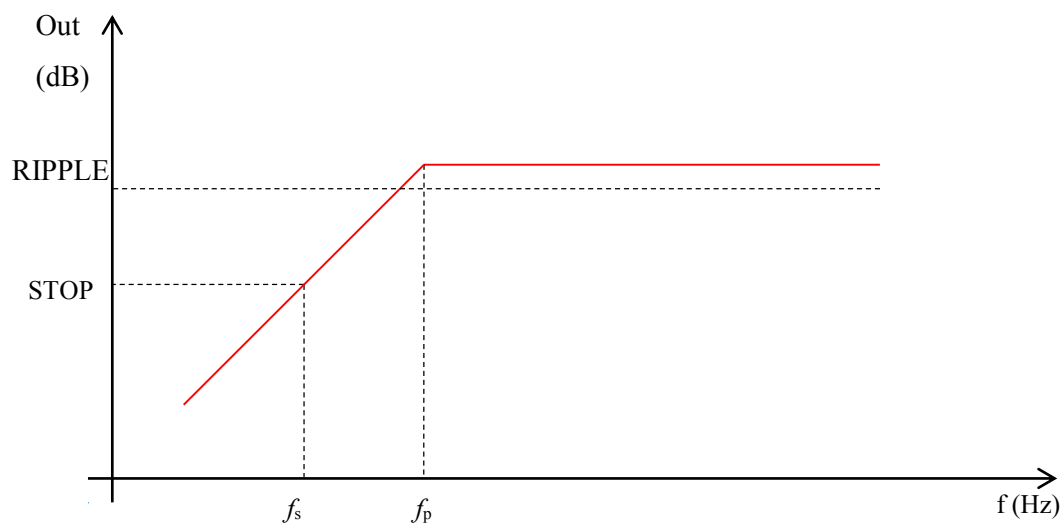


Obr. č. 3.1.7 - schématický náčrt dolní propusti

#### HIPASS (HORNÍ PROPUST)

Horní propust je definována stejnými parametry jako propust dolní. Frekvenci  $f_p$  určíme, od které frekvence filtr propouští. S frekvencí  $f_s$  souvisí parametr STOP, který říká, jaký je minimální útlum při této frekvenci.

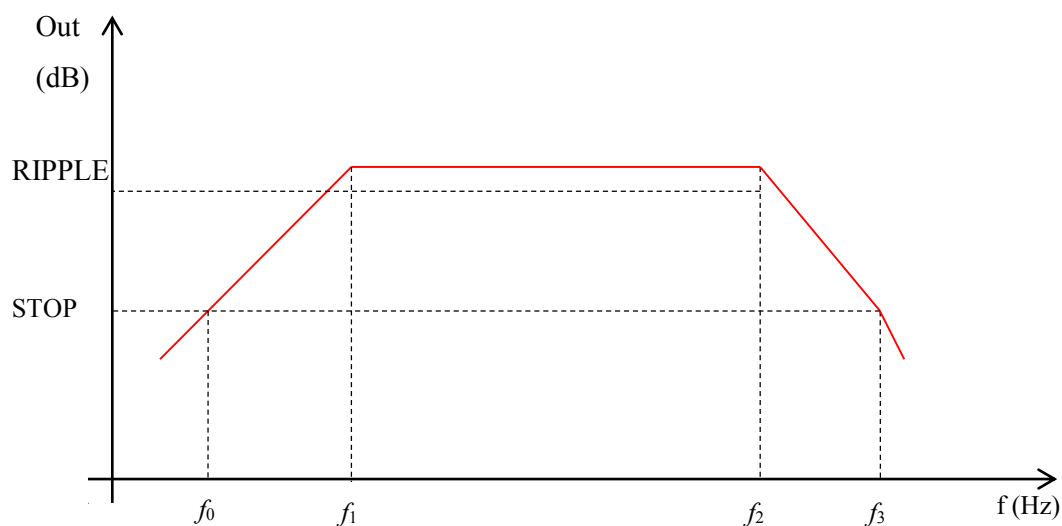




Obr. č. 3.1.8 - schématický náčrt horní propusti

### BANDPASS (PÁSMOVÁ PROPUST)

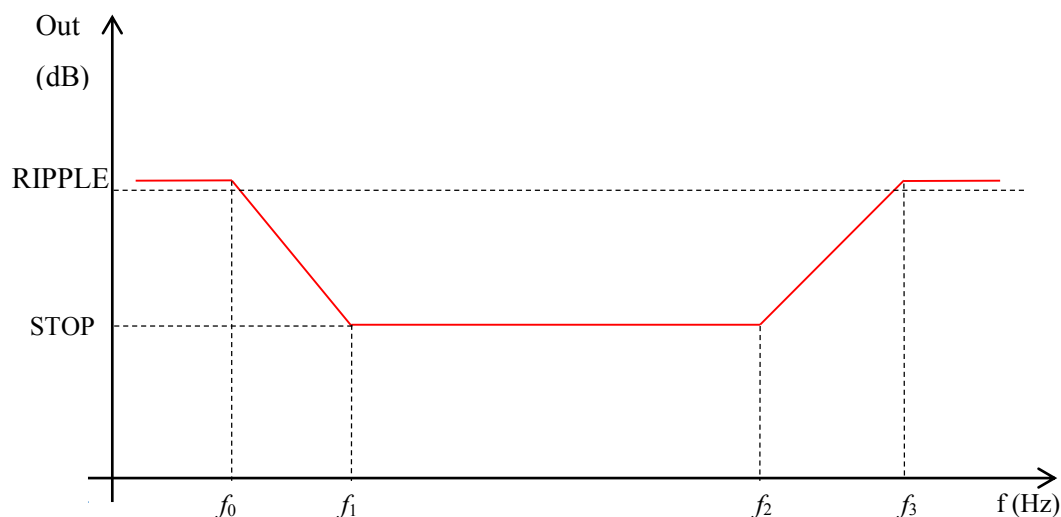
Pásmový filtr je definován šesticí parametrů. Dva parametry nám definují útlum. Maximální přípustný překmit je v pásmu propouštění definován parametrem RIPPLE. Minimální požadovaný útlum v zadržovaném pásmu nám udává parametr STOP. Frekvence  $f_0$  nám udává hodnotu před kterou je útlum roven minimálně hodnotě STOP. Frekvence  $f_1$  a  $f_2$  nám ohraničují pásmo propouštění a frekvence  $f_3$  nám definuje pásmo útlumu.



Obr. č. 3.1.9 - schématický náčrt pásmové propusti

## BANDREJ (PÁSMOVÁ ZÁDRŽ)

Pásmová zadrž je definována stejnými parametry jako pásmová propust. Parametr RIPPLE definuje maximální možný překmit v pásmu propouštění a parametr STOP minimální útlum v zadržovaném pásmu. Frekvencemi  $f_0$  až  $f_3$  nastavíme mezní frekvence.



Obr. č. 3.1.10 - schématický náčrt pásmové zadrž

### 3.1.2 Equivalent parts

Druhou velkou skupinou prvků jsou takzvané equivalent parts. Tyto prvky odpovídají struktuře E a G zařízení. Mají tedy dva piny pro vstup a dva pro výstup. Typ výstupu se liší v závislosti na typu prvku.

Prvky můžeme rozdělit buď jako E a G zařízení. E zařízení poskytují napěťový výstup, G prvky poskytují proudový výstup nebo zařízení obsahující přenosové funkce, které mohou mít na vstupu smíšený signál proudový i napěťový. U takovýchto zařízení už neexistuje rozdíl mezi napěťově řízenými a proudově řízenými prvky.

Každý E a G prvek z knihovny ABM.OLB je definován jako šablona poskytující specifickou funkci. Další parametry, které jsou v modelu definovány, můžeme upravit. Základní šablonu nelze upravovat přímo. Můžeme upravovat pouze hodnoty, které jsou definované v šabloně. Šablonu je možno upravit v nastavení parametru prvku. Řádek obsahující šablonu se nazývá PSpice Template.

## STRUKTURA ŠABLONY

Základ struktury pro E a G prvky vypadá následovně. [1]

E <name> <connecting nodes> <abm keyword> <ABM function>

G <name> <connecting nodes> <abm keyword> <ABM function>

<name> jméno vodiče, na které je zařízení připojeno

<connecting nodes> specifikuje pár uzlů mezi kterými bude výstupní napětí

<ABM keyword> definuje, která přenosová funkce je použita. Mohou být použity tyto přenosové funkce: VALUE pro matematickou funkci, TABLE pro tabulku, FREQ pro frekvenční tabulku LAPLASE pro Laplaceovou transformaci a CHEBYSHEV pro Čebyševův filtr

<ABM function> specifikace dané funkce například matematický výraz

### Příklad Pspice template

ESUM

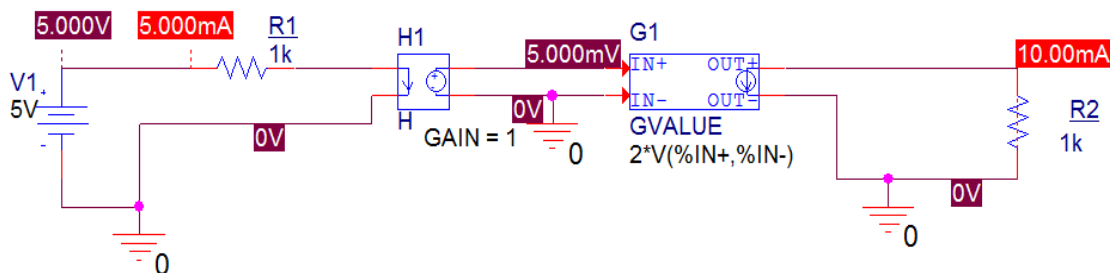
```
E^@REFDES %OUT+ %OUT- VALUE {V(%IN1+,%IN1-)+V(%IN2+,%IN2-)}
```

Šablona začíná písmenem E, které nám říká, že se bude jednat o prvek s napět'ovým výstupem. Název prvku není v této šabloně definován. Další prvek definuje pár uzlů, na kterých se objeví výstupní hodnota. Prvek VALUE nám říká, že se bude jednat o matematickou funkci, která je definována výrazem ve složených závorkách. Prvek tedy sčítá rozdíl napětí mezi vstupy IN1+ a IN1- s napětím IN2+ a IN2-.

### EVALUE A GVALUE

Tyto dva prvky nám na výstupu vypočítávají výraz, který je uveden v parametru EXPR. V parametru EXPR mohou být použity konstanty a další parametry, jako napětí, proud a čas. Napětí může být definováno buď přímo na vodiči V(2) (napětí na vodiči 2), anebo jako napětí mezi dvěma vodiči V(1,2) (napětí mezi vodičem 1 a 2). Proud musí být definován přes zdroj napětí.

### **Příklad připojení proudu na vstup:**



*Obr. č. 3.1.11 - schéma zapojení proudového vstupu*

Když chceme připojit na vstup proudový signál, musí být tento signál odvozen od napěťového zdroje. V tomto příkladu usilujeme o to, mít na výstupu dvakrát zesílený proud protékající rezistorem R1. K převodu na napětí musíme použít převodník proud na napětí, který nalezneme v knihovně ANALOG pod názvem H. Tento prvek nám převede proud protékající rezistorem R1 na napětí, které vstupuje do součástky GVALUE, kde se vynásobí dvěma. Rezistorem R2 nám prochází dvojnásobný proud než na rezistoru R1.

### **EMULT, GMULT, ESUM A GSUM**

Prvky EMULT a GMULT generují na výstupu vynásobený signál ze dvou vstupů. EMULT poskytuje napěťový výstup a GMULT proudový výstup. Obdobně jsou na tom prvky ESUM a GSUM, které na výstupu generují součet dvou vstupních signálů.

### **ETABLE A GTABLE**

Tyto prvky používají převodní funkci popsanou pomocí tabulky. Prvky ETABLE a GTABLE jsou vhodné pro zadání naměřených dat.

Do parametru TABLE zadáváme hodnoty tabulky a to tak, že zadáváme vždy dvojici čísel do kulatých závorek, které jsou oddělené čárkou. První číslo nám reprezentuje hodnotu na vstupu a druhé číslo nám reprezentuje hodnotu, která bude na výstupu. Při zadání více hodnot stačí vložit za první závorku závorku druhou. Pro hodnoty, které nejsou definované, se tabulka linearizuje. V parametru EXPR je definována vstupní hodnota, která se porovnává s parametrem TABLE.

### **EFTABLE, GFTABLE**

Frekvenční tabulka ETABLE a GTABLE má stejné parametry pro zadávání jako frekvenční tabulka FTABLE s tím rozdílem, že definici tabulky definujeme způsobem, který je shodný s tabulkou ETABLE.

Hodnoty tabulky zadáváme do parametru TABLE, a to v následujícím pořadí. První číslo je frekvence, druhé číslo útlum a třetí číslo fáze. V parametru EXPR je opět definováno vstupní napětí.

### **ELAPLASE, GLAPLASE**

Rozdíl mezi funkcemi ELAPLASE, GLAPLASE a funkcí LAPLASE je takový, že u funkcí ELAPLASE a GLAPLASE je zadána Laplaceova transformace v kompletní podobě (v případě LAPLASE to bylo za pomoci num a den). Další rozdíl je v tom, že v těchto funkcích si můžeme nastavit, kde se má snímat vstupní signál a také si ho můžeme upravit (např. vynásobit konstantou).

V parametru EXPR je opět definováno vstupní napětí. Do parametru XFORM. vkládáme Laplaceovou transformaci i s operátory s.

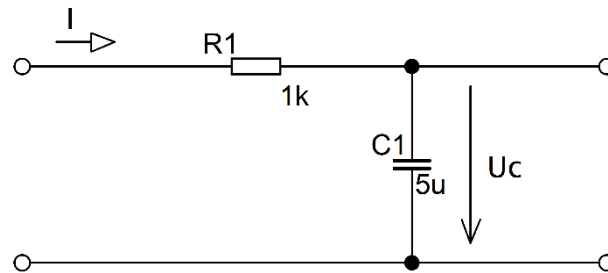
Výstup ze zařízení opět záleží na typu analýzy. Jedná se o stejný princip jako u prvku LAPLASE.

## 4 Příklady

### 4.1 Laplaceova transformace

Zobrazte pomocí Laplaceovy transformace amplitudovou a fázovou charakteristiku dolní propusti.

**Schéma zapojení:**



Obr. č. 4.1.1 - schéma zapojení dolní propusti

Nejdříve si musíme odvodit přenos dolní propusti.

$$i(t) = i_c(t) = C \cdot \frac{du_c(t)}{dt} \quad (4.1.1)$$

$$i(t) = C \cdot y'(t) \quad (4.1.2)$$

$$u(t) = u_R(t) + u_C(t) \quad (4.1.3)$$

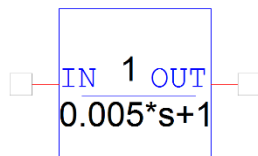
$$u(t) = RC \cdot y'(t) + y(t) \quad (4.1.4)$$

$$U(p) = RC \cdot p \cdot Y(p) + Y(p) \quad (4.1.5)$$

$$F(p) = \frac{U(p)}{Y(p)} = \frac{1}{RC \cdot p + 1} \quad (4.1.6)$$

Přenos dolní propusti pro  $R = 1000\Omega$  a  $C = 5\mu\text{F}$ :

$$F(p) = \frac{1}{0.005p + 1} \quad (4.1.7)$$

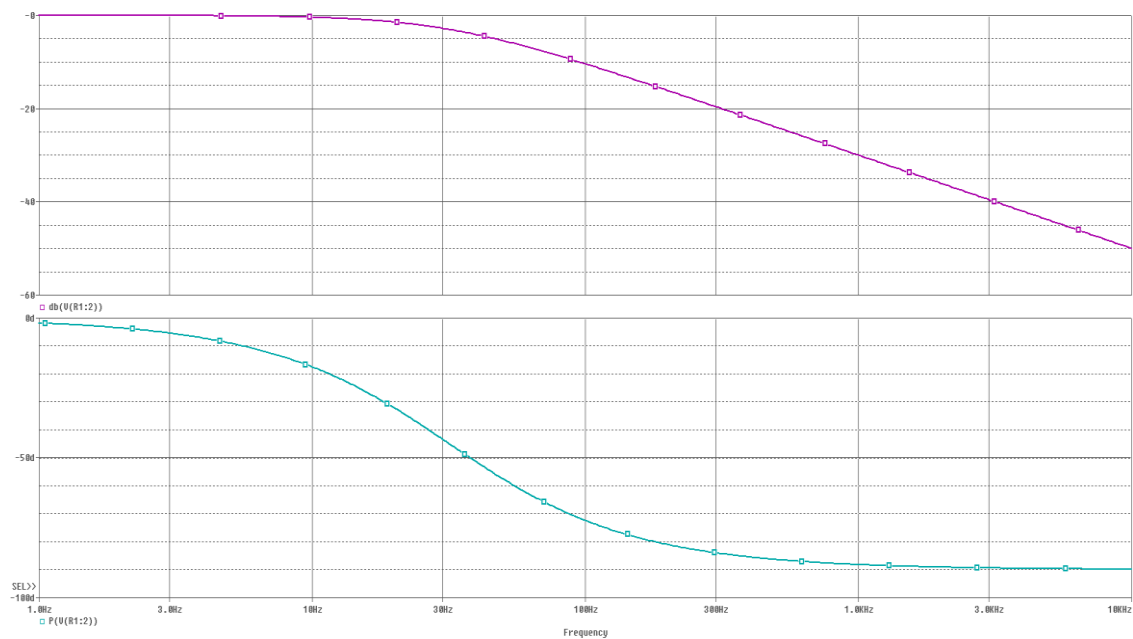


Obr. č. 4.1.2 - nastavení prvku LAPLASE

NUM=1

DENOM=0.005\*s+1

**Výsledné průběhy:**

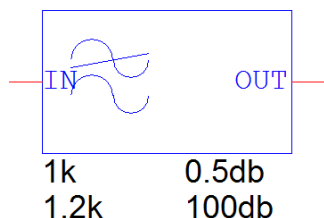


Obr. č. 4.1.3 - amplitudová a fázová charakteristika dolní propusti

Zlomovou frekvenci vypočítáme z časové konstanty obvodu  $f_0 = \frac{1}{\tau \cdot 2\pi} = \frac{1}{RC \cdot 2\pi} = 31.85\text{Hz}$ . Při zlomové frekvenci se útlum rovná  $-3\text{dB}$ . Filtr funguje správně.

## 4.2 Dolní propust

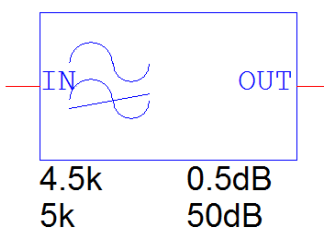
Realizujte dolní propust, která zadrží frekvence nad 1kHz. Při 1,2KHz bude minimální útlum 100dB. V pásmu propouštění bude maximální útlum 0.5dB.



Obr. č. 4.2.1 - nastavení prvku LOWPASS

## 4.3 Horní propust

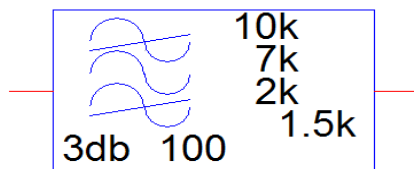
Realizujte horní propust, která bude tlumit frekvence do 5kHz. Při frekvenci 4kHz bude útlum minimálně 50dB. V pásmu propouštění je maximální útlum 0.5dB.



Obr. č. 4.3.1 - nastavení prvku HIPASS

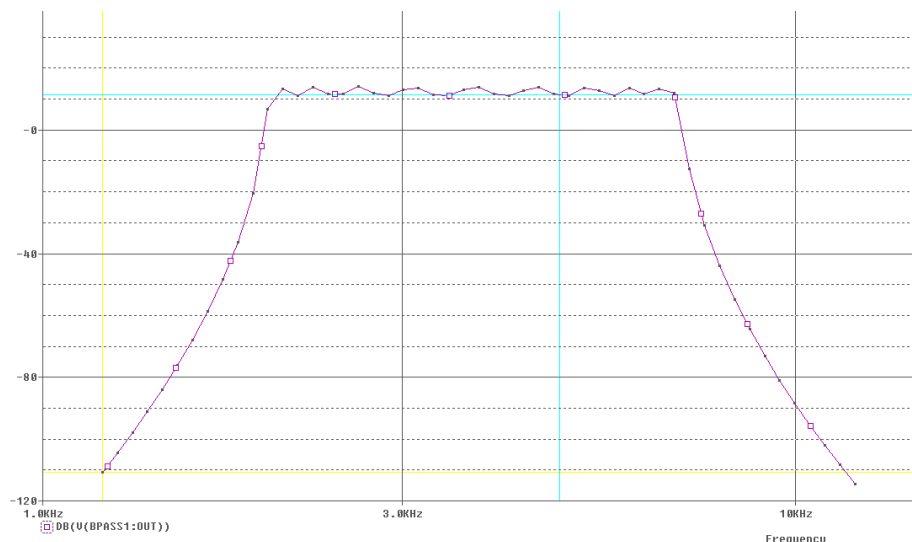
## 4.4 Pásmová propust

Realizujte pásmovou propust, která bude propouštět frekvenci od 2k do 7k, maximální překmit v propouštěcím pásmu bude 3db. Minimální útlum pod frekvenci 1.5k a nad frekvenci 10k bude 100db.



Obr. č. 4.4.1 - nastavení prvku BANDPASS

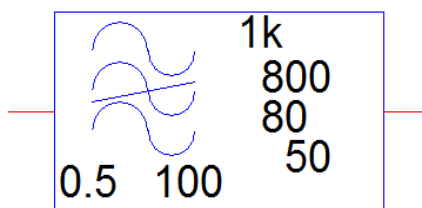




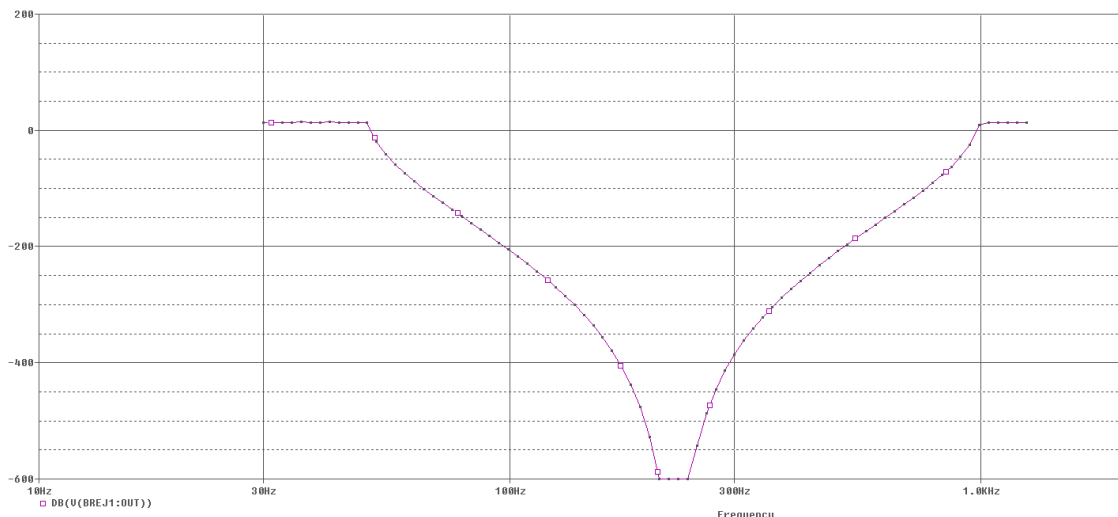
Obr. č. 4.4.2 - amplitudová charakteristika pásmové propusti

## 4.5 Pásmová zadrž

Realizujte pásmovou zadrž, která bude zadržovat frekvenci od 50Hz do 1KHz, maximální překmit v pásmu propouštění bude 0.5dB. Minimální útlum při frekvenci 80Hz do frekvence 800Hz bude 100dB.



Obr. č. 4.5.1 - nastavení prvku BANDREJ



Obr. č. 4.5.2 - amplitudová charakteristika pásmové zadržče

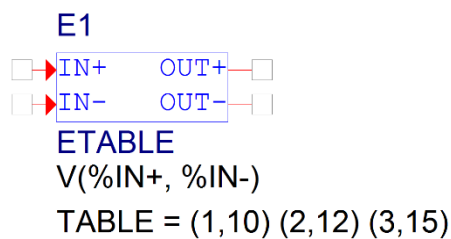
## 4.6 Zadání prvku ETABLE

Při vstupním napětí  $U_1 = 1V$  zobraz na výstupu napětí  $U_2 = 10V$ , pro  $U_1 = 2V$  bude  $U_2 = 12V$  a pro napětí  $U_1 = 3V$  bude výstupní napětí  $U_2 = 15V$ . Vstupní napětí bude mezi vstupními svorkami IN+ a IN-.

**Parametr TABLE zadáme následovně:**

TABLE (1,10) (2,12) (3,15)

EXPR V(%IN+,%IN-)



Obr. č. 4.6.1 - nastavení ETABLE

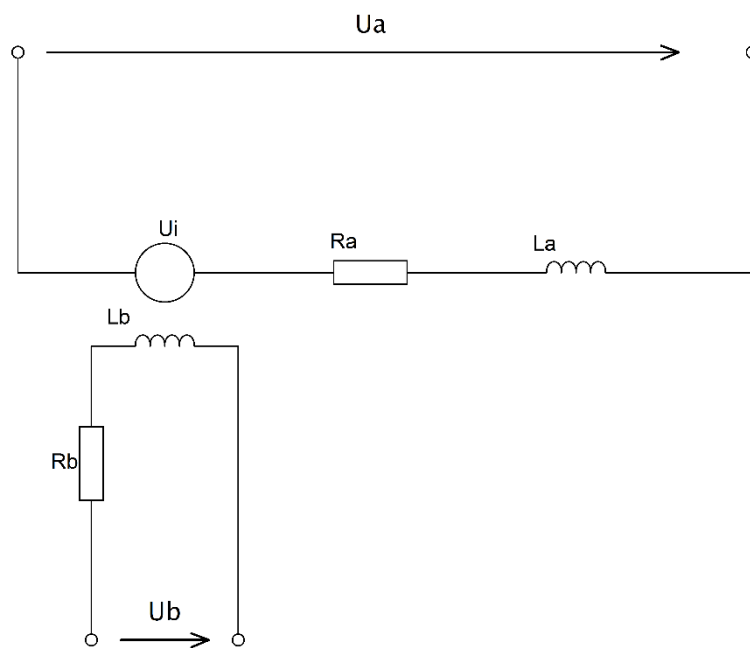
## 5 Matematický model stejnosměrného motoru

Vedoucím práce byla zadána simulace stejnosměrného motoru s cizím buzením pro ověření použití základních prvků. Motor bude simulován v programu Orcad Pspice a pro ověření také v programu Matlab Simulink.

Abychom mohli simulovat elektronický obvod, musíme si nejdříve nastavit jakési zjednodušující předpoklady. Při nenastavení zjednodušujících předpokladů se může stát, že matematický model bude příliš složitý a také velmi náročný na výpočet. U stejnosměrných motorů si zavedeme několik zjednodušujících předpokladů. Budeme uvažovat konstantní parametry motoru (indukčnost kotvy  $L_a$ , odpor kotvy  $R_a$ , a konstrukční konstantu  $c\phi$ ). Také budeme zanedbávat reakci kotvy.

### 5.1 Sestavení matematického modelu

Pro sestavení modelu pro simulaci DC motoru musíme vycházet z matematického popisu.



Obr. č. 5.1.1 - schéma zapojení DC motoru

- Základní rovnice

$$U_a = U_i + R_a I_a + L_a \frac{di_a}{dt} \quad (5.1.1)$$

$$T_a = \frac{L_a}{R_a} \quad (5.1.2)$$

$$u_i = c\phi\omega_m \quad (5.1.3)$$

$$M_e = c\phi i_a \quad (5.1.4)$$

$$M_e - M_z = J_c \frac{d\omega_m}{dt} \quad (5.1.5)$$

Tyto základní rovnice nahradíme Laplaceovou transformací

$$\begin{aligned} U_a(p) &= U_i(p) + R_a I_a(p) + L_a \cdot p \cdot I_a(p) = U_i(p) + R_a I_a(p) \cdot \left(1 + p \cdot \frac{L_a}{R_a}\right) \\ &= U_i(p) + R_a I_a(p) \cdot (1 + p \cdot T_a) \end{aligned} \quad (5.1.6)$$

$$U_i(p) = c\phi\omega_m(p) \quad (5.1.7)$$

$$M_e(p) = c\phi I_a(p) \quad (5.1.8)$$

$$M_e(p) - M_z(p) = J_c \cdot p \cdot \omega_m(p) \quad (5.1.9)$$

- Z Laplaceových transformací si vyjádříme přenosové funkce

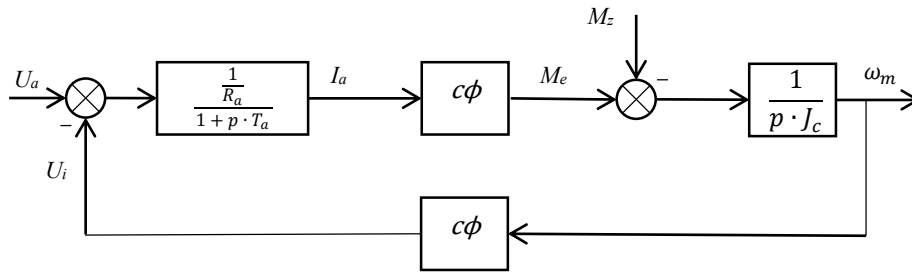
$$F_a(p) = \frac{I_a(p)}{U_a(p) - U_i(p)} = \frac{I_a(p)}{R_a I_a(p) \cdot (1 + p \cdot T_a)} = \frac{1}{1 + p \cdot T_a} \quad (5.1.10)$$

$$F_{c\phi}(p) = \frac{U_i(p)}{\omega_m(p)} = c\phi \quad (5.1.11)$$

$$F_{c\phi}(p) = \frac{M_e(p)}{I_a(p)} = c\phi \quad (5.1.12)$$

$$F_m(p) = \frac{\omega_m(p)}{M_e(p) - M_z(p)} = \frac{\omega_m(p)}{p \cdot J_c \cdot \omega_m} = \frac{1}{p \cdot J_c} \quad (5.1.13)$$

- Z přenosových funkcí složíme výsledný model



Obr. č. 5.1.2 - Matematický model stejnosměrného motoru

## 5.2 Simulovaný stejnosměrný motor

Motor, který budeme simulovat, je stejnosměrný motor s permanentními magnety od firmy AMER (MP 80S/2). Tento motor má následující parametry:

$$P_n = 400 \text{ W}$$

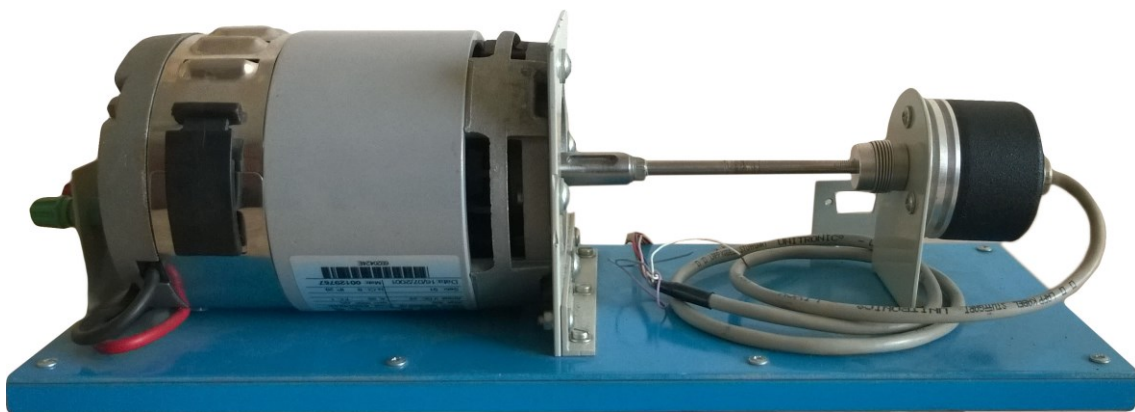
$$U_a = 24 \text{ V}$$

$$n_n = 2800 \text{ ot/min}$$

$$I_a = 20 \text{ A}$$

$$R_a = 0,115 \Omega$$

$$L_a = 0,306 \text{ mH}$$



Obr. č. 5.2.1 - fotografie simulovaného motoru

### 5.2.1 Výpočet parametru pro matematický model

$$\eta = \frac{P_n}{I_a \cdot U_a} \cdot 100 = \frac{400}{20 \cdot 24} \cdot 100 = 0,833 \% \quad (5.2.1.1)$$

$$T_a = \frac{L_a}{R_a} = \frac{0,306 \cdot 10^{-3}}{0,115} = 2,661 \cdot 10^{-3} \text{ s} \quad (5.2.1.2)$$

$$\omega_n = \frac{2 \cdot \pi \cdot n_n}{60} = \frac{2 \cdot \pi \cdot 2800}{60} = 293,22 \text{ s}^{-1} \quad (5.2.1.3)$$

$$c\phi = \frac{U_n - R_a \cdot I_a}{\omega_n} = \frac{24 - 0,115 \cdot 20}{293,22} = 0,074 \text{ Vs} \quad (5.2.1.4)$$

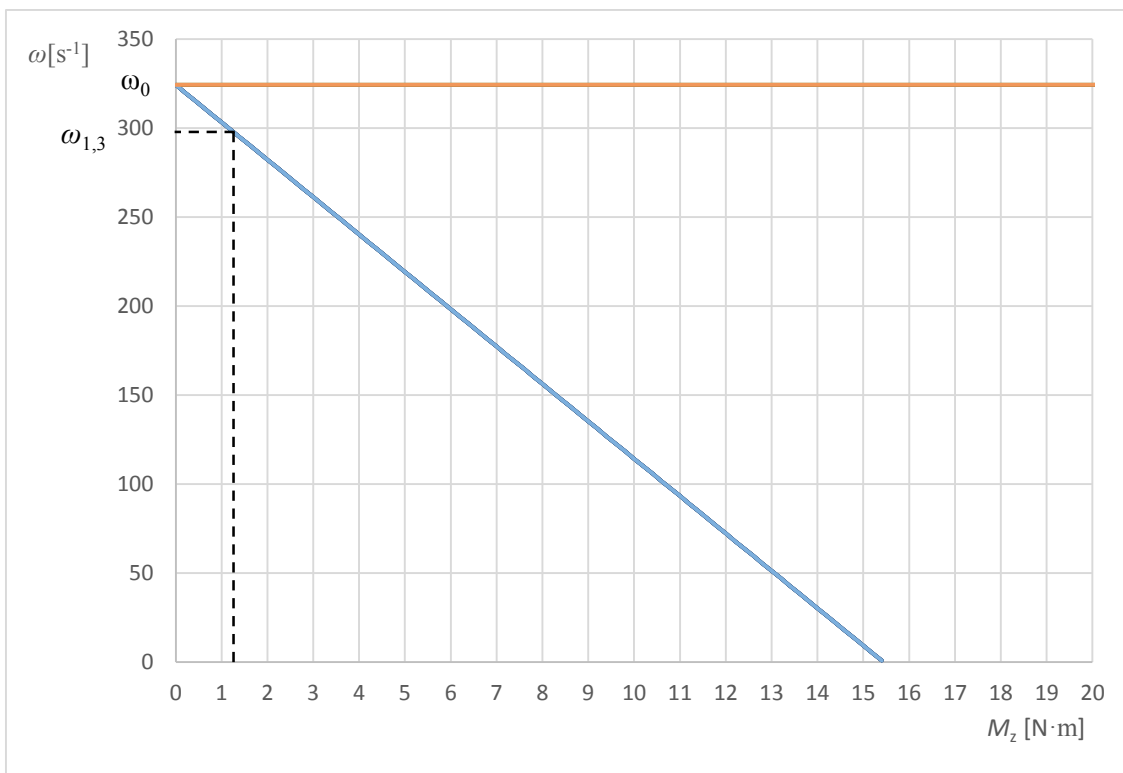
$$J_m = J_c = 0,00125 \text{ kgm}^2 \quad (5.2.1.5)$$

$$\omega = \frac{U_a - R_a \cdot I_a}{c\phi} = \frac{U_a}{c\phi} - \frac{R_a \cdot I_a}{c\phi} = \frac{U_a}{c\phi} - \frac{R_a \cdot M}{(c\phi)^2} = 324,32 - 21M \text{ s}^{-1} \quad (5.2.1.6)$$

$$M_n = M_z = \frac{P_n}{\omega_n} = \frac{400}{293,22} = 1,3 \text{ Nm} \quad (5.2.1.7)$$

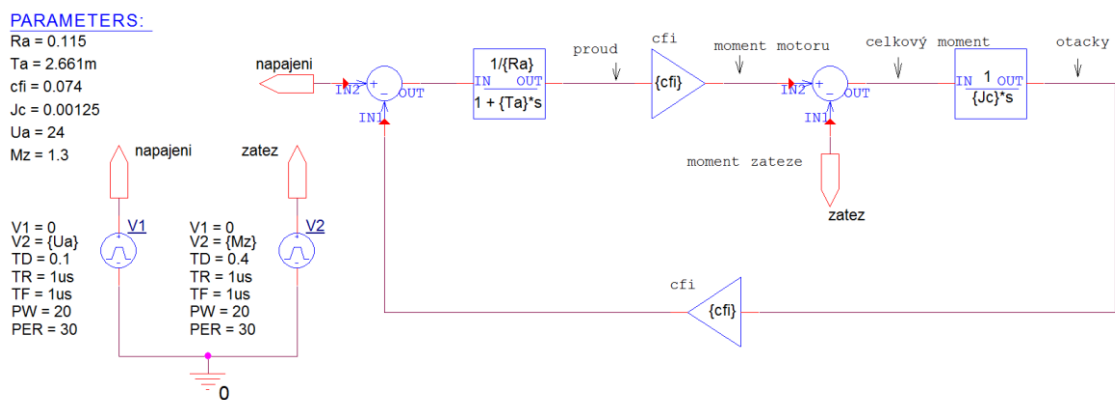
### 5.3 Simulace DC motoru v OrCAD

K simulaci jsem použil prvky z knihovny ABM. Tyto prvky náleží do skupiny control system parts. Byly použity prvky transfer function, diff a gain. Pro simulaci zdroje byl použit pulzní zdroj o amplitudě 24 V. Při navrhování regulační struktury lze použít místo tohoto zdroje například matematický model PWM modulace. Pro simulaci hodnoty zátěžného momentu  $M_z$  byl taktéž použit pulzní zdroj. Tento zdroj je také možné nahradit matematickým modelem zátěže, kterou motor bude pohánět. Simulace je provedena tak, že nejdříve necháme motor naběhnout bez zátěže na jeho jmenovité otáčky, po naběhnutí připojíme zátěž  $M_z$ . Z mechanické charakteristiky, která je zobrazena na obrázku č. 5.3.1, můžeme vyčíst, jakou úhlovou rychlostí se bude motor pohybovat ve stavu, kdy k němu není připojena zátěž a také jak se bude měnit rychlost ve vztahu se zvyšujícím se zatěžovacím momentem. Zátěž  $M_z$  bude mít velikost jmenovitého momentu motoru  $M_n$ . V simulaci zobrazíme průběh rychlosti, proudu a celkového momentu.



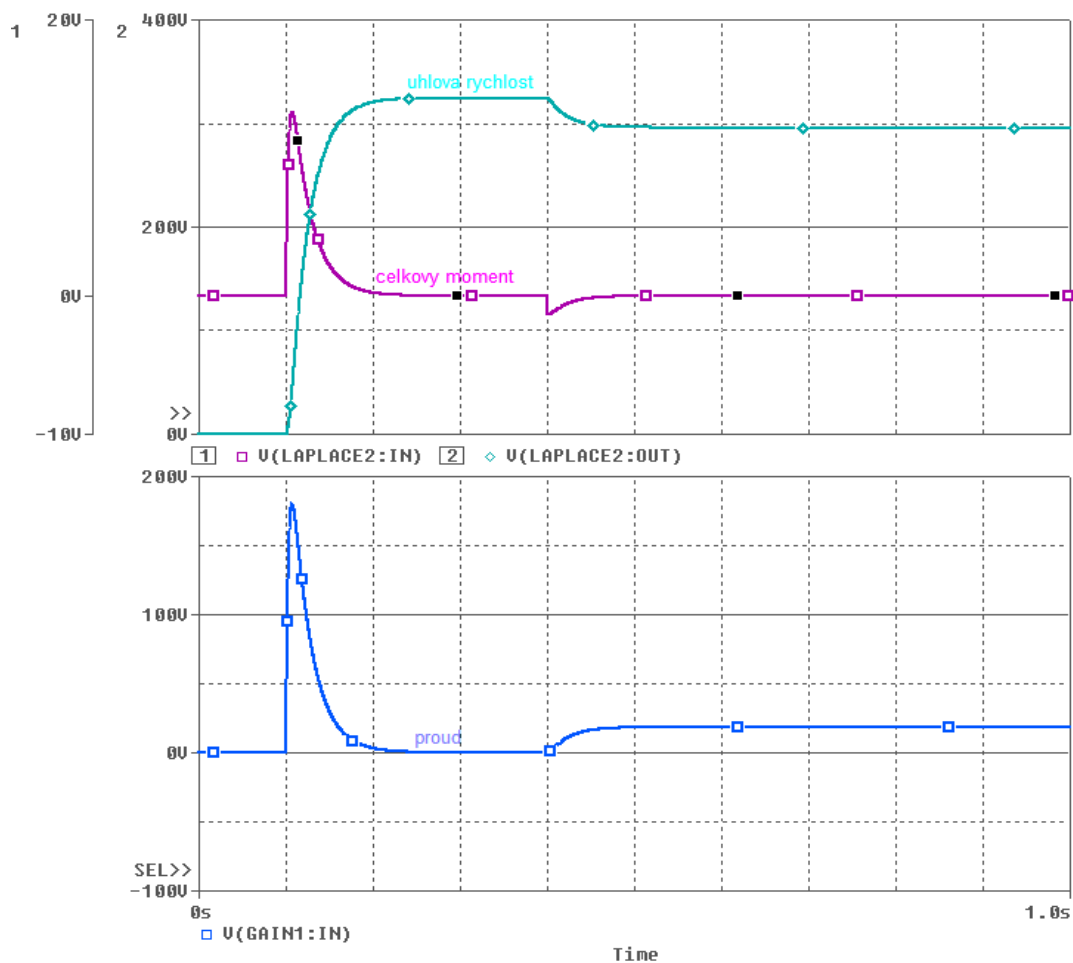
Obr. č. 5.3.1 - mechanická charakteristika DC motoru

### 5.3.1 Blokové schéma motoru v OrCAD



Obr. č. 5.3.2 - blokové schéma matematického modelu v programu OrCAD

### 5.3.2 Výsledné průběhy



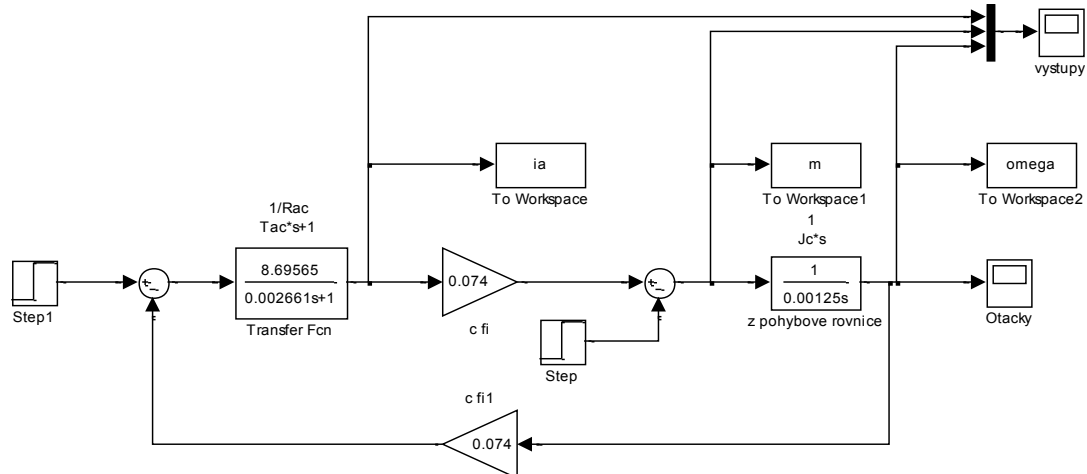
Obr. č. 5.3.3 - simulované průběhy celkového momentu, úhlové rychlosti a proudu

### 5.4 Simulace DC motoru v Matlab Simulink

Pro ověření správnosti simulace byla provedena simulace téhož motoru v programu Matlab Simulink. Pro vytvoření modelu byly použity tyto prvky: Step, Transfer function, Gain, Sum, Scope. Pro následné zpracování simulovaných hodnot v programu Matlab byl použit blok To workspace, který uloží simulované průběhy do prostředí Matlab.

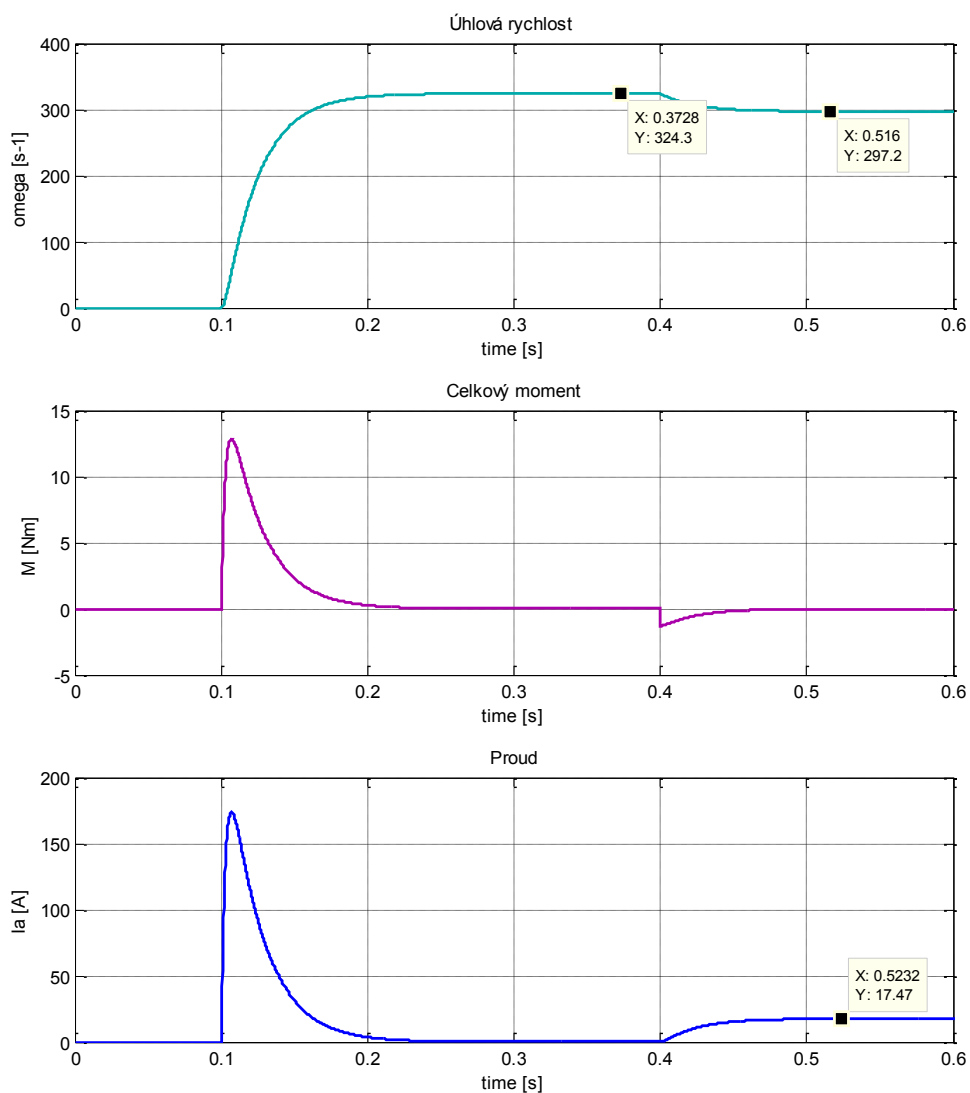


### 5.4.1 Blokové schéma motoru v Simulinku



Obr. č. 5.4.1 - blokové schéma v Matlab Simulink

## 5.4.2 Výsledné průběhy

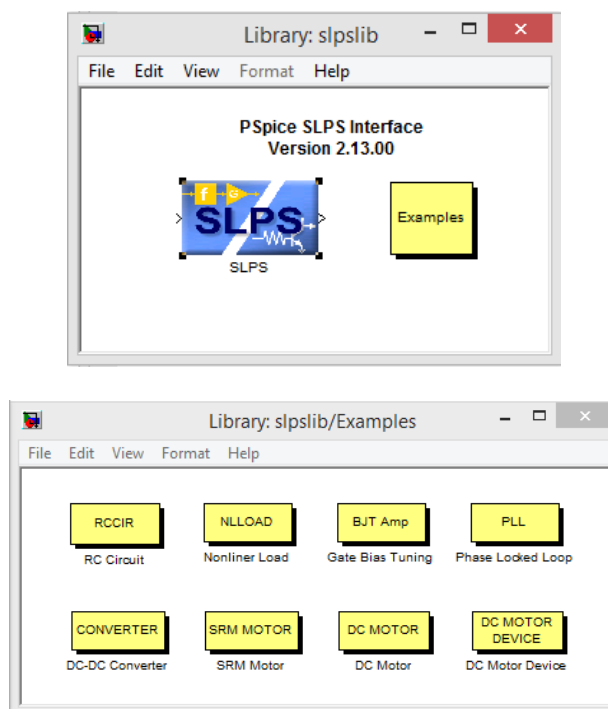


Obr. č. 5.4.2 - simulované průběhy celkového momentu, úhlové rychlosti a proudu.

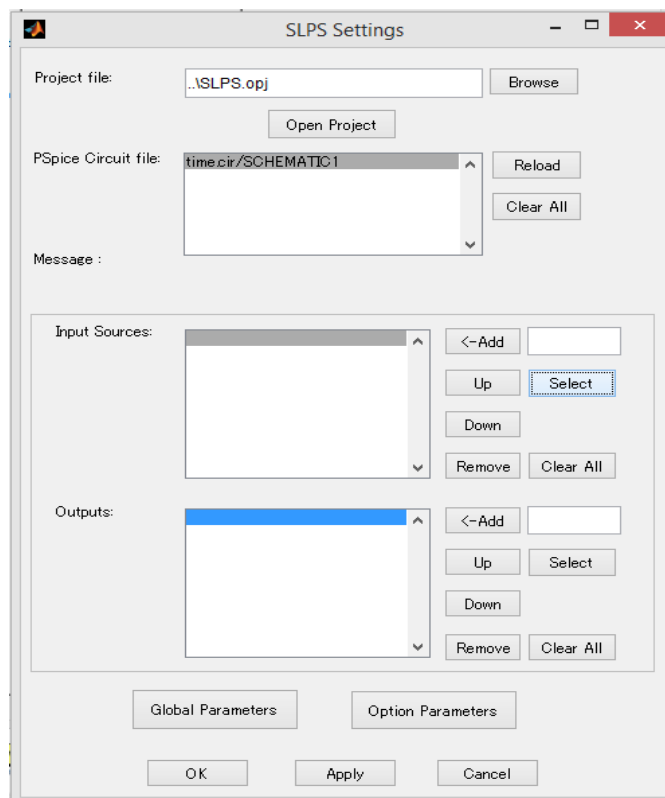
## 5.5 Další možnost simulace

Následující text vychází z [4], [5]. Další možností spolupráce mezi programy OrCad a Matlab Simulink je takzvaná kosimulace. Jedná se o možnost vložení elektrického obvodu s diskrétními součástkami do prostředí Matlab Simulink. Takováto možnost simulování vede k větší přesnosti simulace, která umožňuje simulaci reálného zapojení. Může se jednat o poslední

stádium simulování před výrobou daného systému. Pro simulaci je potřeba mít v počítači nainstalovány programy Matlab Simulink v 32bit verzi (nelze použít pro 64-bit Matlab). Dále pak produkt Cadence OrCad minimálně verze 16.6. Samozřejmě nesmí chybět ani simulační program PSpice A/D. Před vytvořením modelu v prostředí Simulink je potřeba v Matlabu vložit PSpice SLPS path (v Matlabu, file, set path). Poté vytvoříme simulovaný model pro vložení SLPS bloku, který se využívá pro načtení netlistu z Orcadu. Okno s nabídkou SLPS funkcí vyvoláme zadáním příkazu slpslib v comand window Matlabu. Otevrou se dvě okna (obr. č. 5.5.1). Blok SLPS vložíme přetažením do vytvořeného modelu. SLPS blok vložíme do vytvořené struktury (připojíme vstup a výstup). Následovně provedeme nastavení samotného bloku SLPS. Po otevření bloku se ukáže tabulka (obr. č. 5.5.2). Vybereme cestu k projektu vytvořeného v OrCad. Nastavíme vstupy a výstupy. Vstupy a výstupy je možno zadat ručně nebo je vybrat pomocí kontextového okna, které otevřeme tlačítkem select.



Obr. č. 5.5.1 - načtení SLPS bloku



Obr. č. 5.5.2 - nastavení SLPS

## 5.6 Porovnání a závěr

Simulované průběhy se nijak od sebe neliší. Při výběru simulačního programu si musíme rozmyslet, co budeme chtít simulovat. V případě simulace chování určité části obvodu, u které známe matematický popis, se vyplatí použít simulační program Matlab Simulink. V případě, kdy potřebujeme simulovat např. střídač složený z diskrétních součástek a potřebujeme k němu připojit zátěž, využijeme raději program Orcad Capcure. Výhodou programu Orcad je to, že dokáže simulovat kombinaci diskrétních součástek a matematického přenosu. Další alternativou je použití tzn. Kosimulace, která umožňuje vložení netlistu z Orcadu do Matlabu. Posledním rozdílem je pak určitý rozdíl v uživatelském prostředí. Jako výhodu simulace v programu OrCad bych považoval možnost nastavení parametrů ( $T_a$ ,  $R_a$ ,  $M_z$ ) pro uživatele přívětivější, kdy s použitím prvku properties nemusíme zadávat hodnoty přímo do bloku, ale můžeme si je zobrazit v seznamu vedle. Nevýhodou pak jsou výstupní průběhy a jejich možnost úpravy a přenesení do textového dokumentu. V programu Matlab Simulink je možné si simulované veličiny přenést do prostředí Matlab, kde je můžeme později používat k dalším účelům. Také je zde možnost jednodušší úpravy grafu a snadnější přenesení do kancelářského softwaru.

## 6 Závěr

Úkolem této bakalářské práce bylo popsat základní bloky knihovny Analog Behavioral model ze simulačního programu OrCAD.

V první části jsem se zabýval rozdělením a popisem jednotlivých bloků z knihovny ABM. V následující kapitole byla demonstrována činnost na jednoduchých analogových zapojeních. Při ukázce jsem použil např. simulaci dolní propusti za pomoci Laplaceova přenosu. Dále pak různé druhy filtrů, na kterých byly předvedeny možnosti jejich nastavení. V další části jsem se zabýval simulací stejnosměrného motoru s cizím buzením, a to nejdříve z pohledu rozebrání matematického modelu stejnosměrného motoru a určením potřebných parametrů. V dalším kroku byla provedena simulace tohoto matematického modelu v programu OrCAD za pomoci základních bloků z knihovny ABM. Následovala shodná simulace v programu Matlab Simulink. Dosažené výsledky byly porovnány a vyhodnoceny. Zmínka přišla i na další možnost simulace za pomoci SLPS bloku. V příloze byl zpracován SYLAB, ve kterém jsou popsány a využity základní bloky z knihovny ABM. Tyto bloky jsou aplikovány na PI regulátor s omezením integrační složky a výstupního signálu.

Tato bakalářská práce ukazuje, že za pomoci simulačního programu OrCAD je možno simulovat chování systému jako celku a také to, že je možné pro simulaci využít kombinaci matematického a analogového popisu, a to jak z pohledu simulování celého systému v prostředí OrCAD za pomoci diskrétních součástek a prvku z knihovny ABM, tak simulování v prostředí Matlab Simulink za pomoci bloku SLPS (kosimulace).

Tato bakalářská práce byla pro mě určitě přínosná. Naučil jsem se používat základní matematické bloky jak v OrCADu, tak v Matlab Simulinku. Poznal jsem také to, jak je matematický popis důležitý a pro návrh a simulaci složitějších systémů zkrátka nepostradatelný. Objevil jsem mnoho možností programu OrCAD a Matlab Simulink, o kterých jsem dříve nevěděl.

## Literatura

- [1] CADENCE DESIGN SYSTEMS. *PSpice® User Guide: product version 16.6*. USA, říjen 2012.
- [2] TOBIN, Paul. *PSpice for circuit theory and electronic devices*. 1st ed. San Rafael?: Morgan, 2007, s. 35-42. ISBN 1-59829-156-4.
- [3] FITZPATRICK, Dennis. *Analog design and simulation using OrCAD Capture and PSpice*. 1st ed. Amsterdam: Newnes, 2011, s. 165-171. ISBN 978-0-08-097095-0.
- [4] CADENCE DESIGN SYSTEMS. *SLPS User Guide: product version 16.6*. říjen 2012. USA.
- [5] BC. MARAČEK, Lukáš. *Rozšířené možnosti analýz v prostředí PSpice*. Plzeň, 2012. Diplomová Práce. Západočeská univerzita v Plzni. Vedoucí práce Ing. Jiří Stifter, Ph.D.
- [6] VÍTEČEK, Antonín a Miluše VÍTEČKOVÁ. *Zpětnovazební řízení mechatronických systémů*. 1. vyd. Ostrava: VŠB - Technická univerzita Ostrava, 2013, 200 s. ISBN 978-80-248-3232-6.

## **7 Seznam příloh**

I. Sylab