

# Tutorial do Simulador Elétrico SPICE

Renato P. Ribas – Abril/2001

O simulador elétrico SPICE permite a avaliação de circuitos eletrônicos descritos textualmente, onde estão inseridos os comandos de descrição do circuito e de análise requerida:

Para criação do arquivo texto de descrição é importante seguir algumas regras básicas, respeitando a sintaxe de descrição de cada componente (elementos passivos, ativos e fontes) assim como a sintaxe dos comandos que devem ser usados para uma determinada simulação.

**Regra 1** – Primeira linha do arquivo sempre é usada como título, e não tem influencia nenhuma na simulação. Um comando importante colocado na primeira linha será desconsiderado e usado como título do arquivo.

**Regra 2** – Unidades aceitas pelo SPICE:

f	$10^{-15}$	e-15
p	$10^{-12}$	e-12
n	$10^{-9}$	e-9
u	$10^{-6}$	e-6
m	$10^{-3}$	e-3
k	$10^3$	e3
meg	$10^6$	e6
g	$10^9$	e9
t	$10^{12}$	e12

Obs.: pode-se usar também a letra ‘e’ e o expoente ao invés da unidade, ex.:  $4 \times 10^{-12} = 4p$  ou  $4e-12$ .

**Regra 3** – Os nós ou nodos do circuito a ser descrito devem ser nomeados ou numerados (este é preferencial) antes de descrever cada elemento.

**Regra 4** – O número 0 (zero) sempre é usado para indicar o terra, massa ou *ground* do circuito.

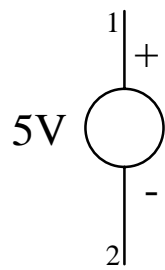
**Regra 5** – Descrição de fontes de tensão sempre são nomeadas começando pela letra ‘V’ e de fontes de corrente pela letra ‘I’. Algumas fontes:

**DC** – Vnome nó\_1(+) nó\_2(-) valor\_dc. Ex.: VCC 1 0 5 (fonte de tensão entre nós 1 (+) e 0 (-) de 5V)

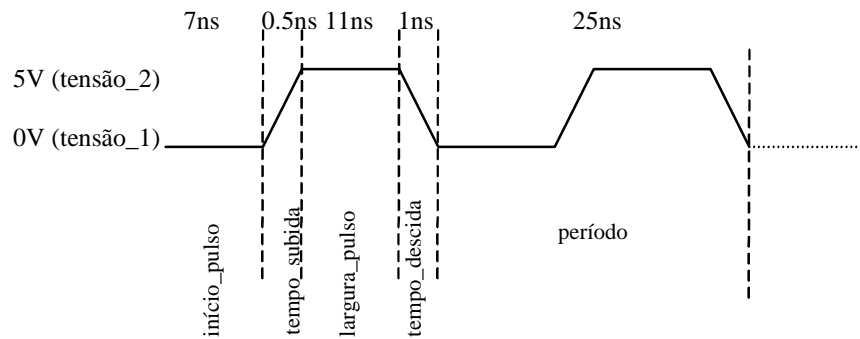
**Senoidal** – Vnome nó\_1(+) nó\_2(-) sin(offset amplitude frequência tempo\_início)

**Quadrada** – Vnome nó\_1(+) nó\_2(-) pulse(tensão\_1 tensão\_2 início\_pulso tempo\_subida tempo\_descida largura \_pulso período)

**Linear** (quadrada/triangular) – Vnome nó1(+) nó2(-) pwl(tempo\_1 tensão\_1 tempo\_2 tensão\_2 tempo\_3 tensão\_3 ... tempo\_n tensão\_n)



Ex.: Vx 1 2 5



Ex.: Vx 1 2 pulse(0 5 7n 0.5n 1n 11n 25n)

Figura 1 - Exemplo de sinais de tensão DC e Quadrada.

**Regra 6** – Descrição de elementos passivos:

**Resistor** – Rnome nó1(+) nó2(-) valor\_resistência

**Capacitor** – Cnome nó1(+) nó2(-) valor\_capacitância

**Indutor** - Lnome nó1(+) nó2(-) valor\_indutância

Obs.: Nunca devem haver dois elementos com mesmo nome. Sugestão: numerar os elementos como R1, R2, C1, C2, C3,...

**Regra 7** – Descrição de elementos ativos. Os elementos ativos sempre exigem a descrição do modelo tecnológico com os parâmetros de processo de fabricação. Se os parâmetros não estão descritos os valores defaults são assumidos.

**Diodo** – Dnome nó1(+) nó2(-) nome\_modelo

**Transistor Bipolar** – Qnome coletor base emissor nome\_modelo

**Transistor MOS** – Mnome dreno gate fonte substrato nome\_modelo (w=? l=?)

**Regra 8** – Comando para análise transiente:

**.trans** passo\_aquisição tempo\_final

**Regra 9** – Obtenção das tensões nos nós, da análise transiente:

**.print tran** V(nó) V(nó) ...

**Regra 10** – Fim de arquivo

**.end**

**Regra 11** – Linha de comentário começa por ‘\*’

\* Esta linha é um comentário e não é considerada pelo simulador.

## 1) Executar o SPICE :

Para rodar o SPICE em modo batch, use janela do DOS Prompt:

```
C:\diretório> spice3 -b arquivo_descricao > arquivo_saida
```

## 2) Visualizar os resultados:

Os resultados da simulação podem ser observados na tabela gerada dentro do arquivo\_saida.

Esses dados também podem ser visualizados em um aplicativo para traçar gráficos: duas ferramentas desenvolvidas na UFRGS estão disponíveis para isso: 'Caipirinha' (ou 'Grafer') e 'VisualWave' (ou 'Gebalvisu').

### 2.1) Caipirinha:

```
C:\diretório> grafer arquivo_saida
```

ou

```
C:\diretório> grafer (depois... File → Open → arquivo_saida)
```

### 2.2) VisualWave (VW):

Dois clicks no icone **Gebalvisu.jar**

ou

usar a linha de comando: **java -jar Gebalvisu.jar**

(se nenhuma das duas alternativas funcionar então será necessário instalar o J2SDK 1.3 - Java 2 Standard Development Kit, ou superior, ou J2RE 1.3 - Java 2 Runtime Environment, ou superior.)

## Arquivo de Descrição:

A criação do arquivo de descrição do circuito para simulação pode seguir uma ordem de forma a manter a clareza do arquivo. Sugere-se:

- 1) Título
- 2) Descrição das fontes
- 3) Descrição dos elementos passivos
- 4) Descrição dos modelos ativos
- 5) Comando para chamada dos modelos (.model .....) )
- 6) Comando para definição do tempo de simulação (.tran .....) )
- 7) Comando para definição das tensões dos nós a serem fornecidas (.print tran .....) )
- 8) Comando final (.end)

### Exemplo 1:

Titulo: Circuito Resistivo (AP02)

```
* fontes
vcc 1 0 dc 7.5

* circuito
r1 1 2 1k
r2 2 0 560
r3 2 3 470
r4 3 0 470

* comandos
.tran 0.1m 100m
.print tran v(1) v(2) v(3)
.end
```

### Exemplo 2:

Titulo: Circuito RC Diferenciador (AP03)

```
vin 1 0 sin(0 1 10k 0 1e10)

c1 1 2 100p
r1 2 0 47k

.tran 0.1m 10m
.print tran v(1) v(2)
.end
```

### Exemplo 3:

Titulo: Circuito RC Diferenciador e Integrador (AP03)

```
*vin 1 0 sin(0 1 1k 0)
vin 1 0 pulse(0 10 0 0 0 1m 2m)
*vin 1 0 pulse(0 10 0 0.9m 0.9m 0.1m 2m)

c1 1 2 100p
r1 2 0 47k

r2 1 3 47k
c2 3 0 100n

.tran 0.001m 10m
```

```
.print tran v(1) v(2) v(3)
.end
```

#### **Exemplo 4:**

Titulo: Circuito Logica de Diodos (AP04)

```
* fontes de tensao
vcc 1 0 dc 5
vin1 10 0 pulse(0 5 0 0 0 10n 20n)
vin2 20 0 pulse(0 5 5n 0 0 10n 20n)
vin3 30 0 pulse(0 5 0 0 0 20n 40n)
vin4 40 0 pulse(0 5 0 0 0 40n 80n)

r1 1 2 100
r2 3 0 1k
d1 2 10 diodo
d2 2 20 diodo
d3 2 30 diodo
d4 40 3 diodo
d5 2 3 diodo

.model diodo d
.tran 0.01n 100n
.print tran v(2) v(3) v(10) v(20) v(30) v(40)
.end
```