

Pontifícia Universidade Católica de Goiás
Escola de Ciências Exatas e da Computação
Engenharia da Computação

Documentação do Parser para o SPICE

Aluno(a): Gabriel Teixeira Andrade Sousa

Orientador: Bierley

Goiânia
2019

Sumário

1	Parametrizando	3
2	Compilação	3
3	Observações	3
4	Linha de título	4
5	.END Linha final	4
6	Comentários	4
7	Resistores	5
8	Capacitores	5
9	Indutores	6
10	Fontes Independentes	6
11	Fontes de corrente lineares controladas por tensão	7
12	Fontes de tensão lineares controladas por tensão	7
13	Fontes de corrente lineares controladas por corrente	7
14	Fontes de tensão lineares controladas por corrente	8
15	Diodos de Junção	8
16	Transistor Bipolar de Junção (TBJs)	9
17	Transistores de Efeito de Campo de Junção (TECJs)	9
18	MOSFETs	9
19	Opções (.OPTIONS)	10
20	.NODESET	11

21 .IC	11
22 .AC: Análise AC Small-Signal	12
23 .DC: Função de Transferência DC	12
24 .PZ: Analise Pólo-Zero	13
25 .SENS: Analise Sensitiva DC ou Small-Signal	13
26 .TF: Análise da Função de Transferência	14
27 .TRAN: Análise Transitória	14
28 Linhas .SAVE	15
29 Linhas .PRINT	15
30 .SUBCKT	16
31 .ENDS	16
32 Chamadas de Sub-Circuitos	17

Configurando o fonte para o arquivo .ckt

1 Parametrizando

Para que o programa analise um arquivo .ckt, é necessário que o fonte contenha o caminho do arquivo a ser analisado.

No arquivo main.cpp uma variável string de nome "filename" contém o caminho. Dessa forma deve-se colocar o caminho do arquivo desejado, como no exemplo abaixo:

```
std::string filename = "C:\\Users\\Gabriel\\ Documents\\Dev\\exemplo.ckt"
```

2 Compilação

A execução do programa ocorre no próprio terminal, assim como a compilação. Para compilar os 5 arquivos:

```
parser.h  
Circuito.h  
parser.cpp  
Circuito.cpp  
main.cpp
```

O comando que tenho utilizado é:

```
g++ parser.h Circuito.h parser.cpp Circuito.cpp main.cpp
```

3 Observações

O comando .OP eu não entendi como funciona, por isso não o implementei. Minha dúvida quanto a ele é quais são as opções/regras que vem depois do comando .OP.

Caso o parser aponte algum erro que não esteja fazendo sentido tente separar os tokens, essa função ainda está em melhoria. Exemplo:

```
.PRINT AC VM(4, 2) separe dessa forma .PRINT AC VM( 4 , 2 )
```

Titulos, comentários e .End

4 Linha de título

Exemplos:

```
POWER AMPLIFIER CIRCUIT  
TEST OF CAM CELL
```

A linha do título deve ser a primeira entrada do arquivo. O título é escrito na tela durante a execução para confirmar qual arquivo está sendo executado.

5 .END Linha final

Exemplos:

```
.END
```

O ".END" deve ser a última linha do arquivo de entrada. Note que o ponto faz parte do comando.

6 Comentários

Forma Geral:

```
* Comentários aqui
```

Exemplos:

```
* RF=1K Gain should be 100  
* Check open-loop gain and phase margin
```

Elementos do Circuito e Modelos

7 Resistores

Forma Geral:

```
RXXXXXXX N1 N2 VALUE
```

Exemplos:

```
R1 1 2 100  
RC1 12 17 1K
```

N1 e N2 são os dois nós dos elementos. VALUE é a resistência (em ohms) e só podem ter valores positivos e negativos, diferente de zero.

8 Capacitores

Forma Geral:

```
CXXXXXXX N+ N- VALUE <IC=INCOND>
```

Exemplos:

```
CBYP 13 0 1UF  
COSC 17 23 10U IC=3V
```

N+ e N- são os nós positivo e negativo dos elementos, respectivamente. VALUE é o valor da capacitância em Farads.

9 Indutores

Forma Geral:

```
LYYYYYYY N+ N- VALUE <IC=INCOND>
```

Exemplos:

```
LLINK 42 69 1UH  
LSHUNT 23 51 10U IC=15.7MA
```

N+ e N- são os nós positivo e negativo dos elementos, respectivamente. VALUE é o valor da indutância em Henries.

Tensões e Correntes de Fontes

10 Fontes Independentes

Forma Geral:

```
VXXXXXXXX N+ N- <DC/TRAN VALUE>  
VXXXXXXXX N+ N- COMPLEX( R, I)  
IXXXXXXXX N+ N- <DC/TRAN VALUE>  
IXXXXXXXX N+ N- COMPLEX( R, I)
```

Exemplos:

```
VCC 10 0 DC 6  
VIN 13 2 0.001  
ISRC 23 21 2  
IIN1 1 5 COMPLEX( 255.333, 0)
```

N + e N- são os nós positivos e negativos, respectivamente.
DC / TRAN é o valor da análise de corrente contínua e dc da fonte.

Fontes Linearmente Dependentes

11 Fontes de corrente lineares controladas por tensão

Forma Geral:

GXXXXXXX N+ N- NC+ NC- VALUE

Exemplos:

G1 2 0 5 0 0.1MMHO

N + e N- são os nós positivos e negativos, respectivamente. O fluxo atual é do nó positivo, através da fonte, para o nó negativo. NC + e NC- são os nós de controle positivo e negativo, respectivamente. VALUE é a transcondutância (em ohms).

12 Fontes de tensão lineares controladas por tensão

Forma Geral:

EXXXXXXXX N+ N- NC+ NC- VALUE

Exemplos:

E1 2 3 14 1 2.0

N+ é o nó positivo e N- é o nó negativo. NC+ e NC- são os nós de controle positivo e negativo, respectivamente. VALUE é o ganho de tensão.

13 Fontes de corrente lineares controladas por corrente

Forma Geral:

FXXXXXXX N+ N- VNAME VALUE

Exemplos:

F1 13 5 VSENS 5

N+ e N- são os nós positivos e negativos, respectivamente. O fluxo atual é do nó positivo, através da fonte, para o nó negativo. VNAME é o nome de uma fonte de tensão através da qual a corrente de controle flui. A direção do fluxo positivo da corrente de controle é do nó positivo, através da fonte, para o nó negativo do VNAME. VALUE é o ganho de corrente.

14 Fontes de tensão lineares controladas por corrente

Forma Geral:

HXXXXXXXX N+ N- VNAME VALUE

Exemplos:

HX 5 17 VZ 0.5K

N+ e N- são os nós positivos e negativos, respectivamente. VNAME é o nome de uma fonte de tensão através da qual a corrente de controle flui. A direção do fluxo positivo da corrente de controle é do nó positivo, através da fonte, para o nó negativo do VNAME. VALUE é a transresistência (em ohms).

Transistor e Diodos

15 Diodos de Junção

Forma Geral:

DXXXXXXXX N+ N- MNAME <AREA> <OFF> <IC=VD> <TEMP>

Exemplos:

DBRIDGE 2 10 DIODE1
DCLMP 3 7 DMOD 3.0 IC=0.2

N+ e N- são os nós positivos e negativos, respectivamente. MNAME é o nome do modelo, AREA é o fator de área e OFF indica uma condição inicial (opcional) no dispositivo para análise dc. Se o fator de área for omitido, será assumido um valor de 1.0.

16 Transistor Bipolar de Junção (TBJs)

Forma Geral:

QXXXXXXXX NC NB NE MNAME

Exemplos:

Q23 10 24 13 QMOD
Q50A 11 26 4 20 MOD1

NC, NB e NE são os nós do coletor, base e emissor, respectivamente. MNAME é o nome do modelo, AREA é o fator de área e OFF indica uma condição inicial (opcional) no dispositivo para a análise dc.

17 Transistores de Efeito de Campo de Junção (TECJs)

Forma Geral:

JXXXXXXXX ND NG NS MNAME

Exemplos:

J1 7 2 3 JM1

ND, NG e NS são os nós de dreno, porta e fonte, respectivamente. MNAME é o nome do modelo.

18 MOSFETs

Forma Geral:

MXXXXXXXX ND NG NS NB MNAME

Exemplos:

```
M1 24 2 0 20 TYPE1  
M31 2 17 6 10 MODM  
M1 2 9 3 0 MOD1
```

ND, NG, NS e NB são os nós de dreno, porta, fonte e volume (substrato), respectivamente. MNAME é o nome do modelo.

Variáveis do Simulador

19 Opções (.OPTIONS)

Forma Geral:

```
.OPTIONS OPT1 OPT2 ... (or OPT=OPTVAL ...)
```

Exemplos:

```
.OPTIONS RELTOL=.005 TRTOL=8
```

A linha de opções permite ao usuário redefinir o controle do programa e as opções do usuário para fins específicos de simulação. Opções adicionais para o Nutmeg também podem ser especificadas e entram em vigor quando Nutmeg lê o arquivo de entrada. As opções especificadas para Nutmeg por meio do comando 'set' também são passadas para o SPICE3 como se especificadas em uma linha .OPTIONS.

Condições Iniciais

20 .NODESET

Forma Geral:

```
.NODESET V(NODNUM)=VAL V(NODNUM)=VAL ...
```

Exemplos:

```
.NODESET V(12)=4.5 V(4)=2.23
```

A linha Nodeset ajuda o programa a encontrar a solução transitória inicial ou dc, fazendo um passo preliminar com os nós especificados, mantidos nas tensões especificadas. A restrição é então liberada e a iteração continua na verdadeira solução. A linha .NODESET pode ser necessária para convergência em circuitos biestáveis ou estáveis.

21 .IC

Forma Geral:

```
.IC V(NODNUM)=VAL V(NODNUM)=VAL ...
```

Exemplos:

```
.IC V(11)=5 V(4)=-5 V(2)=2.2
```

A linha IC é para definir condições iniciais transitórias. Ele tem duas interpretações diferentes, dependendo se o parâmetro UIC está especificado na linha de controle .TRAN. Além disso, não se deve confundir essa linha com a linha .NODESET. A linha .NODESET serve apenas para ajudar na convergência dc e não afeta a solução de polarização final.

Análises

22 .AC: Análise AC Small-Signal

Forma Geral:

```
.AC DEC ND FSTART FSTOP  
.AC OCT NO FSTART FSTOP  
.AC LIN NP FSTART FSTOP
```

Exemplos:

```
.AC DEC 10 1 10K  
.AC DEC 10 1K 100MEG  
.AC LIN 100 1 100HZ
```

DEC representa variação de dezenas, e ND é o número de pontos por dezena. OCT significa variação de oitava e NO é o número de pontos por oitava. LIN significa variação linear e NP é o número de pontos. FSTART é a frequência inicial e FSTOP é a frequência final. Se essa linha estiver incluída no arquivo de entrada, o SPICE executará uma análise AC do circuito na faixa de frequência especificada.

23 .DC: Função de Transferência DC

Forma Geral:

```
.DC SRCNAM VSTART VSTOP VINCR [SRC2 START2 STOP2 INCR2]
```

Exemplos:

```
.DC VIN 0.25 5.0 0.25  
.DC VDS 0 10 .5 VGS 0 5 1  
.DC VCE 0 10 .25 IB 0 10U 1U
```

A linha DC define a fonte da curva de transferência dc e os limites de varredura (novamente com os capacitores abertos e os indutores em curto). SRCNAM é o nome de uma fonte de tensão ou corrente independente. VSTART, VSTOP e VINCR são os valores inicial, final e incremental, respectivamente.

24 .PZ: Analise Pólo-Zero

Forma Geral:

```
.PZ NODE1 NODE2 NODE3 NODE4 CUR POL  
.PZ NODE1 NODE2 NODE3 NODE4 CUR ZER  
.PZ NODE1 NODE2 NODE3 NODE4 CUR PZ  
.PZ NODE1 NODE2 NODE3 NODE4 VOL POL  
.PZ NODE1 NODE2 NODE3 NODE4 VOL ZER  
.PZ NODE1 NODE2 NODE3 NODE4 VOL PZ
```

Exemplos:

```
.PZ 1 0 3 0 CUR POL  
.PZ 2 3 5 0 VOL ZER  
.PZ 4 1 4 1 CUR PZ
```

CUR representa uma função de transferência do tipo (tensão de saída) / (corrente de entrada) enquanto VOL representa uma função de transferência do tipo (tensão de saída) / (tensão de entrada). POL significa apenas análise de pólos, ZER apenas para análise zero e PZ para ambos. Esse recurso é fornecido principalmente porque, se houver uma não convergência na localização de pólos ou zeros, pelo menos o outro poderá ser encontrado. Finalmente, NODE1 e NODE2 são os dois nós de entrada e NODE3 e NODE4 são os dois nós de saída.

25 .SENS: Analise Sensitiva DC ou Small-Signal

Forma Geral:

```
.SENS OUTVAR  
.SENS OUTVAR AC DEC ND FSTART FSTOP  
.SENS OUTVAR AC OCT NO FSTART FSTOP  
.SENS OUTVAR AC LIN NP FSTART FSTOP
```

Exemplos:

```
.SENS V(1,OUT)  
.SENS V(OUT) AC DEC 10 100 100k  
.SENS I(VTEST)
```

A sensibilidade do OUTVAR a todos os parâmetros diferentes de zero é calculada quando a análise SENS é especificada. OUTVAR é uma variável de circuito (tensão do nó ou corrente de derivação da fonte de tensão). A primeira forma calcula a sensibilidade do valor do ponto de operação DC da OUTVAR. A segunda forma calcula a sensibilidade dos valores AC do OUTVAR.

26 .TF: Análise da Função de Transferência

Forma Geral:

```
.TF OUTVAR INSRC
```

Exemplos:

```
.TF V(5, 3) VIN  
.TF I(VLOAD) VIN
```

A linha TF define a saída e entrada de small-signal para a análise dc de small-signal. OUTVAR é a variável de saída de small-signal e INSRC é a fonte de entrada de small-signal. Se essa linha estiver incluída, o SPICE calcula o valor dc do small-signal da função de transferência (saída / entrada), resistência de entrada e resistência de saída. Para o primeiro exemplo, o SPICE computaria a razão de V (5, 3) para VIN, a resistência de entrada de small-signal no VIN e a resistência de saída de small-signal medida nos nós 5 e 3.

27 .TRAN: Análise Transitória

Forma Geral:

```
.TRAN TSTEP TSTOP <TSTART <TMAX>
```

Exemplos:

```
.TRAN 1NS 100NS  
.TRAN 1NS 1000NS 500NS  
.TRAN 10NS 1US
```

TSTEP é o incremento de impressão ou plotagem para a saída da texto de linha. Para uso com o pós-processador, TSTEP é o incremento de computação sugerido. TSTOP é a tempo final e TSTART é a tempo inicial. Se TSTART for omitido, presume-se que seja 0. A análise transitória sempre começa no tempo zero. No intervalo ltzero;,TSTART>, o circuito é analisado (para atingir um estado estacionário), mas nenhuma saída é armazenada. No intervalo ltTSTART;,TSTOP>, o circuito é analisado e as saídas são armazenadas. TMAX é o tamanho máximo que a SPICE usa; por padrão, o programa escolhe TSTEP ou (TSTOP-TSTART)/50.0, o que for menor. O TMAX é útil quando se deseja garantir um intervalo de computação menor que o incremento da impressora, TSTEP.

Grupo de Saída

28 Linhas .SAVE

Forma Geral:

```
.SAVE vector vector vector ...
```

Exemplos:

```
.SAVE i(vin) input output  
.SAVE @m1[id]
```

Os vetores listados na linha .SAVE são registrados em arquivo RAW para uso posterior com spice3 ou nutmeg (nutmeg é apenas uma parte da análise de dados do spice3, sem a capacidade de simular). Os nomes de vetores padrão são aceitos. Se nenhuma linha .SAVE for fornecida, o conjunto padrão de vetores será salvo (tensões dos nós e correntes de ramificação da fonte de tensão).

29 Linhas .PRINT

Forma Geral:

```
.PRINT PRTYPE <V(N1<,N2>)> <I(VXXXXXXX)> ...
```

Exemplos:

```
.PRINT TRAN V(4) I(VIN)  
.PRINT DC V(2) I(VSRC) V(23, 17)  
.PRINT AC VM(4, 2) VR(7) VP(8, 3)
```


Subcircuitos

30 .SUBCKT

Forma Geral:

```
.SUBCKT subnam N1 ltN2; N3 ...>
```

Exemplos:

```
.SUBCKT OPAMP 1 2 3 4
```

Uma definição de circuito é iniciada com uma linha `.SUBCKT`. `SUBNAM` é o nome do subcircuito e `N1`, `N2`, ... são os nós externos, que não podem valer zero. O grupo de linhas de elementos que seguem imediatamente a linha `.SUBCKT` define o subcircuito. A última linha em uma definição do subcircuito é a linha `.ENDS`. As linhas de controle podem não aparecer dentro de uma definição de subcircuito; no entanto, as definições de subcircuito podem conter qualquer outra coisa, incluindo outras definições de subcircuito, modelos de dispositivo e chamadas de subcircuito.

31 .ENDS

Forma Geral:

```
.ENDS ltSUBNAM;>
```

Exemplos:

```
.ENDS OPAMP
```

A linha "Ends" deve ser a última para qualquer definição de subcircuito. O nome do subcircuito, se incluído, indica qual definição de subcircuito está sendo finalizada; se omitido, todos os subcircuitos definidos são finalizados. O nome é necessário apenas quando as definições de subcircuitos aninhados estão sendo feitas.

32 Chamadas de Sub-Circuitos

Forma Geral:

```
XXXXXXXX N1 ltN2; N3 ...> SUBNA
```

Exemplos:

```
X1 2 4 17 3 1 MULTI
```

Os subcircuitos são usados no SPICE especificando pseudoelementos começando com a letra X, seguidos pelos nós do circuito a serem usados na expansão do subcircuito.