

# SPICE の使い方

天野 英晴

## 1 SPICE について

SPICE(Simulation Program with Integrated Circuit Emphasis) はUCB(University of California Berkeley) で開発された電子回路シミュレータである。遅延解析やノイズ解析など広い用途に用いることができる。spice が基となって作られた hspice は、世界中の半導体メーカーで利用されている。ちなみに、VDEC 経由で利用できるの、矢上に来れば利用可能である。ここでは、フリーソフトウェアの ngspice を用いる。より深く知りたい方は、<http://www.ngspice.sourceforge.net> を参照されたい。

## 2 入力ファイルを作る

まずシミュレートする回路図を書く。

次に入力ファイルを作るための準備として以下のことを行なう。

1. 電源素子を定義する。
2. 素子名をつける。
3. ノード番号をつける。
4. 接続に従ってファイルを記述する。

## 3 spice の起動から終了まで

### 3.1 起動

入力ファイルが cmosinv.cir であるとする。

```
% ngspice cmosinv.cir [Return]
*****
** ngspice-24 : Circuit level simulation program
** The U. C. Berkeley CAD Group
** Copyright 1985-1994, Regents of the University of California.
** Please get your ngspice manual from http://ngspice.sourceforge.net/docs.html
** Please file your bug-reports at http://ngspice.sourceforge.net/bugrep.html
** Creation Date: Fri Nov 16 18:08:30 JST 2012
*****

Circuit: cmos inverter

ngpice 1 ->
```

入力ファイルにエラーがある場合は、エラーのある行番号を出力する。

### 3.2 実行

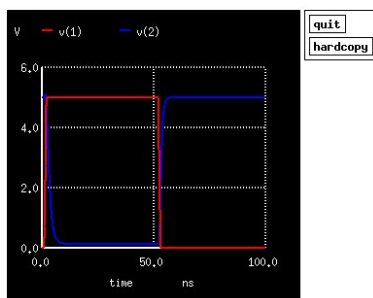
```
ngspice 1 -> run [Return]
```

### 3.3 結果の出力

plot コマンドを実行すると新しくウィンドウが開いて波形が表示される。ここでは ノード番号 1 と 2 での電圧を見ようとしている。

```
ngspice 2 -> plot v(1) v(2)
```

ウィンドウには



のような図が表示される。hardcopy ボタンでプリンタへのハードコピー、quit ボタンでウィンドウを閉じる。ちなみに、マウスの左ボタンを使って 2 点間の座標、 $x$  方向の増分と  $y$  方向の増分、傾き等を表示することができる。右ボタンを使うと部分的に拡大することができるので、レポート作成時に数値を計測する場合、この機能を利用することをお勧めする。レポートは keio.jp での提出なので、ハードコピーを取ることはお勧めしない。

- plot コマンドの代わりに print コマンドを使うと実際の数値が表示される。
- 入力ファイル中に .plot, .print という行を作っておくと (書き方はコマンドと同じ)、

```
% ngspice < 入力ファイル > 出力ファイル
```

のようにするだけで 出力ファイル中に波形が文字で表示される。

### 3.4 終了のしかた

```
ngspice 3 -> quit [Return]
```

## 4 主なコマンド一覧

コマンド名	動作
display	入力回路の情報の表示
edit	エディタの起動 (Mule)
help	ヘルプの表示
listing	入力ファイルの内容表示
source 入力ファイル名	新しい入力ファイルの読込

## 5 入力ファイルの記述方法

- 先頭が \* で始まる行はコメント
- 先頭が + で始まる行は前の行からの続き
- 値を表す時には以下の scale factor が使用できる。

記号	値
T	1e+12
G	1e+9
MEG	1e+6
K	1e+3
M	1e-3
U	1e-6
MIL	25.4e-6
N	1e-9
P	1e-12
F	1e-15

- GND は必ず節点番号 0
- 主な素子 (xxxx は任意、Nx にはノード番号を書く。オプションの記述方法は回路記述例、または help を参照のこと)

素子名	記述
電圧源	Vxxxx N+ N- V
抵抗	Rxxxx N+ N- R
容量	Cxxxx N+ N- C
インダクタンス	Lxxxx N+ N- L
ダイオード	Dxxxx N+ N- modelname
BJT	Qxxxx Ncollector Nbase Nemitter ModelName
MOS	Mxxxx Ndrain Ngate Nsource Nsubstrate ModelName

- シミュレーション制御記述 (. で始まるもの)

目的	記述
電圧変化	.dc Vxxxx Vstart Vstop Vincr
入力ファイルの終了	.end
モデルの記述	.model ModelName parameter...
シミュレーション時間	.tran UnitTime TotalTime

## 6 回路記述例

### CMOS Inverter (cmosinv.cir)

```
CMOS Inverter

* Circuit Description

m1 2 1 3 3 pmos1
m2 2 1 0 0 nmos1
c1 2 0 0.005pf

vcc 3 0 5v
vin 1 0 pulse ( 5 0 1ns 1ns 1ns 40ns 80ns)

* MOS FET Model
.model nmos1 nmos vto=1.0 kp=3.1e-5 gamma=0.37 phi=0.65 lambda=0.02 rd=1.0
+ rs=1.0 cbd=20fF cbs=20fF is=1.0e-15 pb=0.87 cgso=4.0e-11 cgdo=4.0e-11
+ cgbo=2.0e-10 rsh=10.0 cj=2.0e-4 mj=0.5 cjsw=1.0e-9 js=1.0e-8 nsub=4.0e15
+ nss=1.0e10 ld=0.8u uo=700 kf=1.0e-26 af=1.2
.model pmos1 pmos vto=1.0 kp=3.1e-5 gamma=0.37 phi=0.65 lambda=0.02 rd=1.0
+ rs=1.0 cbd=20fF cbs=20fF is=1.0e-15 pb=0.87 cgso=4.0e-11 cgdo=4.0e-11
+ cgbo=2.0e-10 rsh=10.0 cj=2.0e-4 mj=0.5 cjsw=1.0e-9 js=1.0e-8 nsub=4.0e15
+ nss=1.0e10 ld=0.8u uo=700 kf=1.0e-26 af=1.2

* Simulation unit time
.tran 0.1ns 100ns
.end
```

上記の記述は図 1 に示す単純な CMOS Inverter の記述例である。

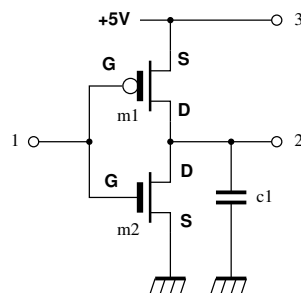


図 1: CMOS Inverter

この記述は、回路の結線自体は単純だが、モデルのパラメータが多数設定されている。これは MOS FET をシミュレーションする上で必要な値で、半導体のプロセス技術、レイアウトによる各部の面積などによって決定される。ここではこの値の細かい意味にはこだわらず（実はよくわからないのが普通）、全ての FET にこの値を使うことにする。一般的にプロセスが決まると、製造元からパラメータを貰うことができ、レイアウトツールがレイアウトに応じて自動的に抽出してくれる。また、この記述では、電圧源としてパルスが用いられている。

```
vin 1 0 pulse ( 5 0 Xns Yns Zns Kns Jns)
```

これは、節点 1 と 0 の間に振幅 5v-0v のパルスを生じさせる記述で、それぞれの数値は以下の意味を持つ。

- X: パルスの変化するスタートまでの時間
- Y: パルスの立ち下がり時間
- Z: パルス立ち上がり時間
- K: High レベル (最初の 5v) の幅
- J: 全体のパルス幅

## DTL Inverter(dtlinv.cir)

BJT を用いた回路例として DTL Inverter を紹介する。

```
DTL inverter
* Circuit description
q1 5 4 0 mod1
d1 2 1 diode
d2 2 3 diode
d3 3 4 diode
r1 6 2 1k
r2 6 5 1k

vcc 6 0 5v
vin1 1 0 pulse (5 0 1ns 1ns 1ns 400ns 800ns)

* BJT Model
.model mod1 npn is=1.0e-15 ise=1.0e-13 isc=1.0e-13 cje=2pf cjc=2pf
.model diode d

*Simulation Unit time
.tran 0.1ns 1000ns
.end
```

これは図 3 に示すように授業で紹介した通りの単純な DTL インバータである。

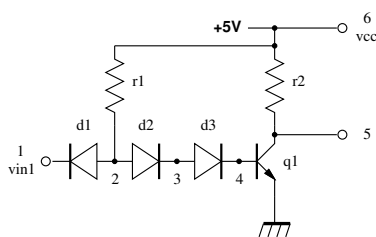


図 2: DTL Inverter

同様にシミュレーションして結果を確認せよ。結果のグラフを出したら横軸に注意。DTL が遅いのがわかる。

### レポート課題

- CMOS NAND 回路を構成し、シミュレーションし、遅延時間のおおよそを測定せよ。非常に短く測定不能の場合は 0 としても良い。
- 下の図の TTL の Inverter 回路 (1 入力の NAND) を構成し、シミュレーションせよ。遅延時間を測定し、DTL Inverter と比較して、どのように速くなったか？なぜかを簡単に説明せよ。

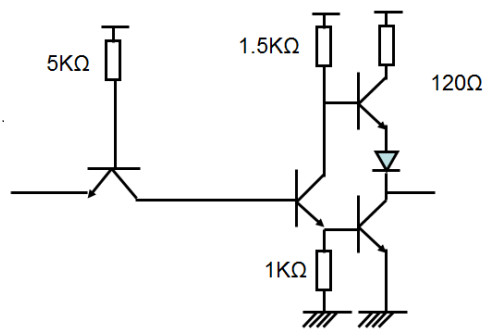


图 3: TTL Inverter