

# SPICE入門

山本健一\*

(2002年6月)

## 1 はじめに

SPICEは、電気電子回路を計算機上でシミュレートし、その動作を確認するためのプログラムであり、その機能は強力です。Unix上ではSpice3やngspice、Windows上ではPSpiceやmicro-capなどが有名で、これらは無償で提供されていますので入手・インストールできます。ここでは、電気電子工学科の端末室で使えるSpice3f5のごく初歩的な使い方と、振幅変調器で使用する回路の解析方法について説明します。SPICEは、電気電子回路の動作確認が簡単にできるなど利用価値が高いシミュレータですので、電気電子回路の学習や設計等に活用して下さい。詳しい使用法はマニュアルや解説書に譲りますが、ここではとりあえずSPICEを使ってみてみたい方のために書かれています<sup>1</sup>。本稿がSPICE導入の一助となれば幸いです。

## 2 解析の手順

### 2.1 直流回路

まず、図1(a)の回路の動作解析を行う場合について説明します。次の手順に従って、ソースファイル(回路の情報が書かれたファイル)を作成し、解析を行ってみてください。

1. 回路図内の節点(ノード)に番号をふります。(節点とは、電気電子素子(部品)の足と足の接合点のことで、プリント基板上では、同一のランドやジャンパー線につながっている場所のことを指します。図1には節点を●で示してあります<sup>2</sup>。)このときに、「グラウンドは「0」をあてるこ

\*e-mail: yamamoto@eee.u-ryukyu.ac.jp

<sup>1</sup>本ドキュメントの記述がなるべく正確になるよう努力しておりますが、ミスが皆無であることは保障いたしかねます。本ドキュメントにより生じた損害や不利益などには著者は一切責任を負いかねます。また万が一、間違いなど発見されましたら、是非御連絡下さいませお願い致します。

<sup>2</sup>図13では節点(5)に●が複数個ありますが、同一のランド上(抵抗ゼロで接続されていて電位が同じ状態になっている)に配置されるので唯一の節点番号が与えられています。

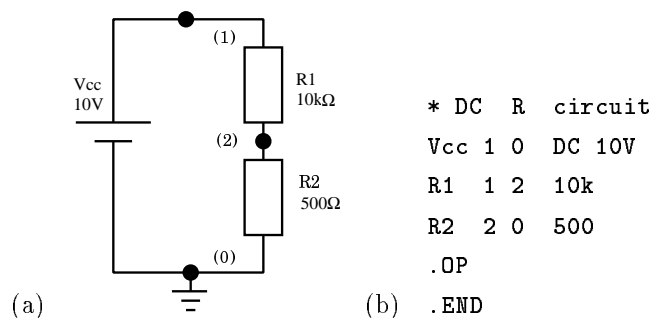


図 1: 直流回路; (a) 回路図, (b) ソースファイル

と, 「各節点番号は重複しないこと」, 「回路は必ず閉じたループになっていること」という制約がありますので, 注意が必要です。

2. 図 1(b) に書かれたソースファイルを作成します。

「mule」などを使って同図 (b) を入力し, 何か名前 (ファイルネーム) を付けて (例えば, 「dc.cir」という名前) で保存して下さい。

ここで, 各行の意味は次の通りです。1 行目は, コメント行といってあなたのメモに使うして下さい。(SPICE の動作には全く影響を与えません。また, 1 行目以降の行でも, 行の先頭に「\*」が書かれていればコメント行となります。)

(a) 2 行目は, 電圧源 (Vcc) が節点 1 と 2 に接続されていて, 直流の 10V であることを示しています。(最初の「V」が電圧源であることを示しています。「cc」は名前で, 他の電圧源と重複しなければ, どんな名前でも良いです。)

(b) 3 行目は, 抵抗 (R1) が節点 1 と 2 に接続されていて, 10kΩ であることを示しています。(最初の「R」が抵抗であることを示しています。1 は名前で, 他の抵抗値と重複しなければ, どんな名前でも良いです。)

(c) 5 行目は, この回路の詳細な直流動作を求めることを命令しています。

(d) 6 行目は, ソースファイルの終了を示しています。

なお, 「1 行目がコメント行」であり, 「.END が最終行」であること以外, 行の順番に関する制約はありません。

3. spice3 を立ち上げる。

kterm 上で,

```
> spice3 dc.cir
```

と入力してください。

```
Spice 1 ->
```

という表示がでます。ここで、画面にあらわれた「Spice 1 ->」は、SPICE が立ち上がっており、SPICE はコンソール画面であなたの命令を待っています。したがって「Spice ->」がでている間は、UNIX コマンドは実行できませんので、ご注意下さい。

4. 解析させる。

```
Spice 1 -> run
```

と入力しましょう。すると、spice は「dc.cir」の解析を始め、それが終ると

```
Spice 2 ->
```

となります。

5. 結果を表示させる。

```
Spice 2 -> show all
```

と入力すると、次の表示がでます。

```
Resistor: Simple linear resistor
device      r2      r1
model       R      R
resistance  500    1e+04
           i 0.000952 0.000952
           p 0.000454 0.00907
```

```
Vsource: Independent voltage source
device      v
           dc      10
           acmag    0
           i -0.000952
           p  0.00952
```

```
Spice 3 ->
```

これらは各素子 (device) が抵抗 model=R であり、その抵抗値 (resistance)、電流値 (i)、消費電力 (p) を示し、さらに電源の電圧 (dc)、電流 (i)、消費電力 (p) を示しています。電源の電流がマイナスになっているのは、電源内部でマイナス極 (節点 0) からプラス極 (節点 1) に向けて電流が流れているためです。

次に、

```
Spice 3 -> print all
```

と入力すると、各部の電圧がプリントされます。もちろん節点 1 の電

```
kterm
> mule dc.cir &
[1] 378
> spice3 dc.cir
Program: Spice, version: 3f5, patchlevel: 1
Date built: Wed Jan 23 08:21:15 GMT 2002

Type "help" for more information, "quit" to leave.

Circuit: DC R Circuit file="dc.cir"

Spice 1 -> run
Spice 2 -> show all
Resistor: Simple linear resistor
device      r2      r1
model       R      R
resistance  500    1e+04
i  0.000952 0.000952
p  0.000454 0.00907

Vsource: Independent voltage source
device      v
dc          10
acmag       0
i  -0.000952
p   0.00952

Spice 3 ->
Spice 3 -> quit
Warning: the following plot hasn't been saved:
op1      DC R Circuit file="dc.cir", Operating Point

Are you sure you want to quit (yes)? y
Spice-3f5 done
[1] + 終了                                mule dc.cir
>
>
```

図 2: SPICE を実行した例

圧のみを知りたいければ、

```
Spice 4 -> print v(1)
```

などと、入力して下さい。ただしこのときの「v」は必ず小文字を使用して下さい。(ソースファイルでは、大文字、小文字のどちらでもかまいません。)また、節点 1 と節点 2 の間の電位差を知りたいければ、

```
Spice 5 -> print v(1,2)
```

あるいは

```
Spice 6 -> print v(1)-v(2)
```

を入力して下さい。

6. SPICE を終了させる。一つの回路 (ソースファイル) の解析が終わったら、

```
Spice 7 -> quit
```

として下さい。spice が終了し、UNIX コマンドを受けつける状態に戻っています。

念のため SPICE で作業した時のスクリーンショットの例を図 2 に示しておきますので、ご自分の画面と比較してみてください。

## 2.2 交流回路

### 2.2.1 抵抗回路

本節では交流回路の解析について説明します。解析方法は、基本的に直流回路の場合とほとんど変わりません。回路は電源が直流から交流に変わっています。それに伴って、図 3(b) に示したソースファイルを用います。ここでは「V 1 0 AC 1V」が新たにできました。「AC」は、電源が交流であることを示しています。ここで、1V というのは、振幅の最大値 (0-ピーク) であることを知っておいて下さい。(電気回路では、一般に実効値で交流の電圧や電流をよく表示しますので注意が必要です。)

もう 1 箇所「.AC DEC 10 10Hz 100kHz」が新しくできました。交流信号源の周波数を、10Hz から 100kHz まで DEC(10 の対数) 目盛で (1 DEC 内を 10 点きざみとして) 増加させることを示しています。ちなみに、ここを「.AC LIN 10 10Hz 100Hz」とすると、10Hz から 100Hz の間で線形に (10~100Hz 間のきざみ点数を 10 として) 周波数を増加させながら解析していきます。ある一定の周波数間隔で解析したいのであれば、「.AC LIN 1 100Hz 100Hz」としてください。

さあ、先の直流回路の場合と同様にして、「run」「print v(1)」「print all」などで実行・出力させてみてください。周波数の実数部と虚数部 (0), 電圧の実数部と虚数部が表示されます。

用いた素子が純抵抗であり、インピーダンスは周波数に依存しないので、同じ値が縦にずらっと並んで出てきましたね。

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

と入力すると、それらの結果 (周波数依存性) がグラフに出力されます。ここで、v(1) を vr(1), vi(1), vm(1), vp(1) に変えると、それぞれ節点 1 の電圧の実数部、虚数部、大きさ ( $\sqrt{vr(1)^2 + vi(1)^2}$ ), 位相角に対応する値が出力されます。ここで、v(1) は実数部 (vr) を表しており「vm ではない」ことに注意して下さい。

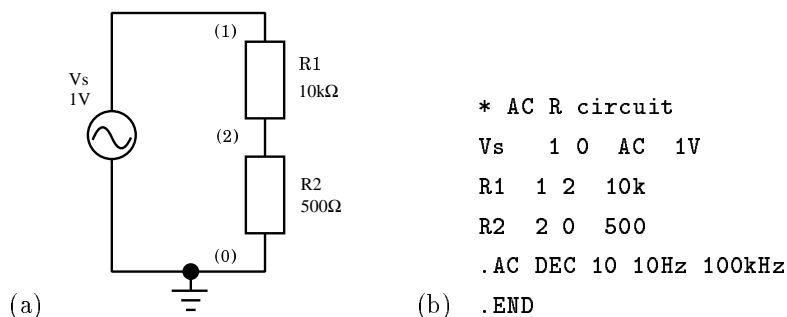


図 3: 交流回路 1 (抵抗器のみ); (a) 回路図, (b) ソースファイル

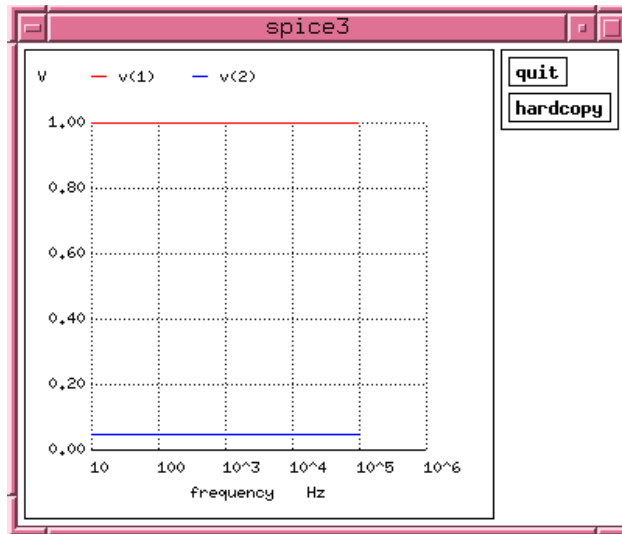


図 4: 「DEC」で実行させた結果

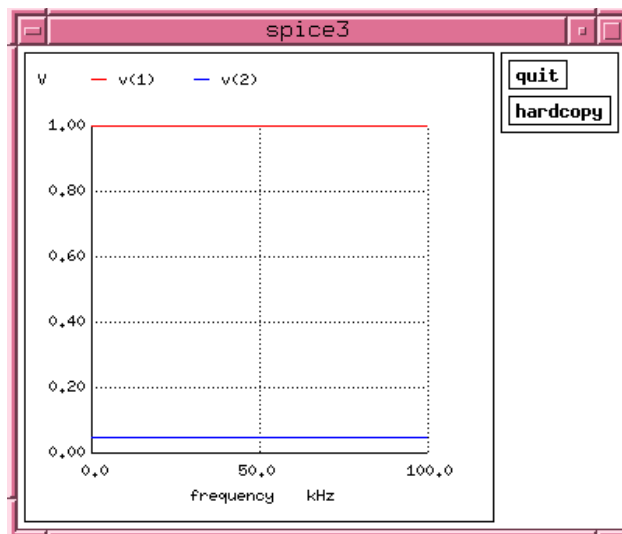


図 5: 「LIN」で実行させた結果

## 2.2.2 LCR 回路

図 6 は, LCR 直列回路とそのソースファイルです。同様に解析してみてください。ソースファイルを作成して, 「spice lcr.cir」 「run」 「plot vm(3)」 「plot vm(2,3)」 で, R1 と L1 の両端電圧の周波数依存性が図??の様に出てきましたね。

共振点が 160Hz あたりにあるので, 解析指定を「.AC LIN 100 100Hz 300Hz」に変えるとその付近の詳しい変化を図 5 の様に見ることが出来ます。

さて, ここまでは, 波形の振幅値のみを見てきましたね。交流の波形そのもの(時間に対する電圧や電流の変化)は, 見ることができるのでしょうか? できます。そのために, 「.TRAN」(過渡解析)が用意されています。図 7 のようにソースファイルを作成してください。まず電源の部分が, 「sin(0 1 100Hz)」と変わっていますね。これは, 「100Hz の正弦波で 1V の振幅, バイアス(直流分)は 0」という意味です。また「.AC .....」のところが, 「.TRAN 0.1ms 100ms 0ms 0.01ms」に変わっていますね。これは, 「時刻 0 から 100ms まで, 0.01ms 間隔でその値を解析(計算)し, 0.1ms 間隔で表示せよ。」という意味です。さあ, SPICE の画面で, 「run」 「plot v(3)」などとしてみてください。節点 3 の電圧波形が時刻に合わせて変化している交流波形が見れましたね。他の部分の波形も出力させて遊んでください。

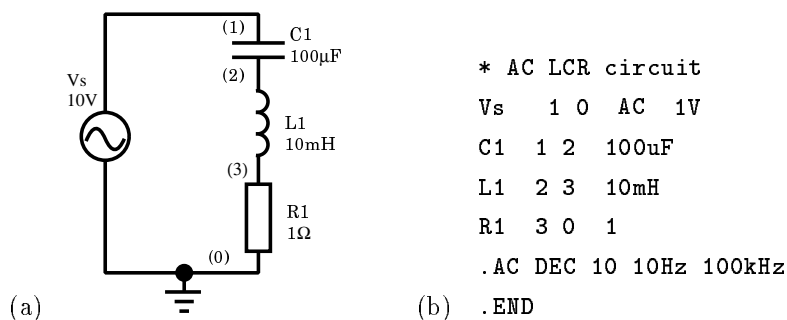


図 6: 交流回路 2(LCR 回路); (a) 回路図, (b) ソースファイル

```

* TRAN LCR circuit
Vs 1 0 sin(0 1 100Hz)
C1 1 2 100uF
L1 2 3 10mH
R1 3 0 1
.TRAN 0.1ms 100ms 0ms 0.01ms
.END

```

図 7: 過渡解析用のソースファイル(LCR 回路)

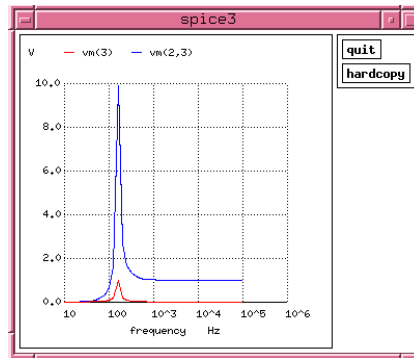


図 8: 10Hz から 100kHz まで解析させた例

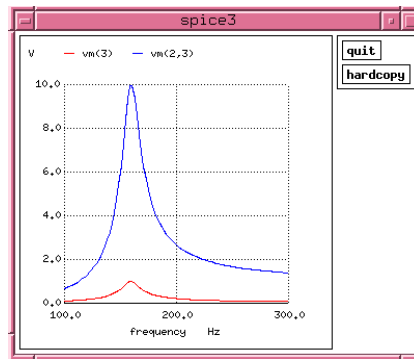


図 9: 100Hz から 300Hz まで解析させた例

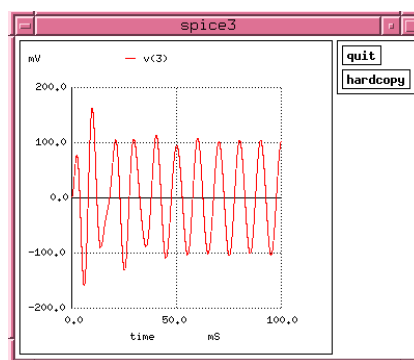


図 10: 過渡解析させた例

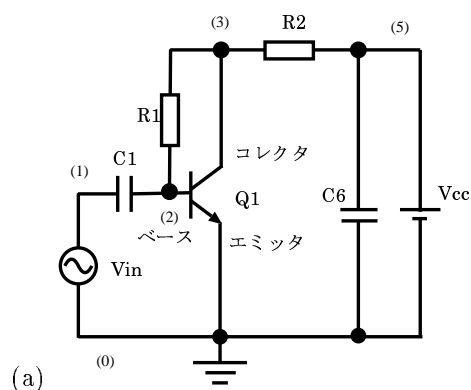


## 2.3 トランジスタ回路

図 11 は、トランジスタ増幅とそのソースファイルです。この回路で新たにトランジスタが登場しました。ご存知のようにトランジスタには足が3本あって、コレクタ、ベース、エミッタという名前が付いています。それぞれの足の名前と回路図記号の対応は同図中に書いてあります。

ソースファイルでトランジスタは「Q」という名前で始まることになっています。それぞれの足の節点は、「コレクタ」「ベース」「エミッタ」の順番で記述することになっています。

この素子の特性(素子固有の定数)は、「.MODEL」の行に記述してあります。2行に分かれていますので、そのまま入力してください。ここで指定してある定数の意味は以下の通りです。



```
(a)
*Transistor amp., file_name="amp.cir"
Vin 1 0 sin(0 0.01 100)
C1 1 2 4.7uF
R1 2 3 470k
R2 3 5 4.7k
Q1 3 2 0 QC1815
C6 5 0 0.47uF
Vcc 5 0 9V
.OP
*****2SC1815
.MODEL QC1815 NPN (IS=4E-14 BF=170 BR=3.6 VA=100 IK=0.25
+ RB=50 RC=0.76 CJC=4.8p CJE=12p TF=0.63n TR=25n)
.TRAN 0.01ms 5.1ms 0s 0.01ms
(b) .END
```

図 11: トランジスタ増幅回路; (a) 回路図, (b) ソースファイル

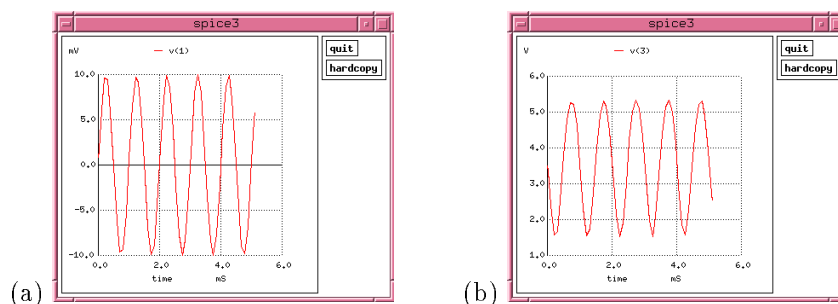


図 12: 解析結果; (a) 入力波形 – plot v(1), (b) 出力波形 – plot v(3)

QC1815 NPN 「Q」はトランジスタを意味しており「C1815」は名前です。この名前は何でも良いのですが、ここではトランジスタ 2SC1815 の特性を表現していますので C1815 という名前にしてあります。「NPN」は NPN 型トランジスタを意味しています。

IS 飽和電流

BF 順方向電流増幅率  $\beta_F$

BR 逆方向電流増幅率  $\beta_R$

VA 順方向アーリー電圧 [V]

IK 順方向  $\beta$  高電流ロールオフに対する折れ曲がり点 [A]

RB ベース抵抗 [ $\Omega$ ]

RC コレクタ抵抗 [ $\Omega$ ]

CJC B-C 間接合容量 ( $V_{bc} = 0$ )[F]

CJE B-E 間接合容量 ( $V_{be} = 0$ )[F]

TF 順方向通過時間 (順方向トランジット時間)[s]

TR 逆方向通過時間 (逆方向トランジット時間)[s]

そして 2 行目行頭の「+」は「直前の行のつづき」という意味です。これらの定数の意味は、ここでは割愛しますので電子回路で勉強してください。

「run」「show all」としてみてください。動作点の情報などが書かれています。意味は電子回路学で習いますので、今は分からなくても良いです。

次に、過渡解析で波形を見てみましょう。「plot v(1)」で入力波形、「plot v(3)」で出力波形がそれぞれ図 12 のように見れます。位相が  $180^\circ$  ずれており、それぞれの縦軸のスケールの違いに注意すればちゃんと電圧も増幅されていることが分かります。周波数や入力電圧を変えて、遊んでみて下さい。ひょっとすると、単純な正弦波だけでなく、トランジスタ回路の飽和現象等で波形の形が崩れたりするのが観察できるかもしれません。

### 3 振幅変調 (AM) 回路の動作解析

振幅変調 (AM) 回路の例を図 13 に示します。これも図 14 のソースファイルを用いて過渡解析を行って下さい。解析に少し時間がかかるかもしれませんが、少し我慢して待って下さい。不安なら「Ctrl-c」で解析を一旦中断し、「.TRAN」行の解析時間を短くしてから再トライしてもいいですね。

解析が終了したら「plot v(8)」として結果を表示させてみてください。図 15(a) のように信号 (被変調波) のエンベロープ (包絡線) がみれましたか? 波形の一部を「マウスの右ボタン」を使って拡大してみましょう。同図 (b) の様に高い周波数を持つ交流波形 (搬送波) がびっちり詰まっているのが分かりますね。

ところで、同図中に点線で描かれた素子 (パルス源とコンデンサ) についてですが、これらの素子は、実際の回路には出てきません。しかし、SPICE などの回路解析シミュレータではそのままの回路では発振回路として動作させることはできません。それは、発振が回路内の微弱な雑音に起因して起こるからです。この雑音を正確に表現するのは難しいのですが、パルス波は全ての周波数成分を持つ波形です (フーリエ解析で学習してください。) から、ここではそれを用いました。

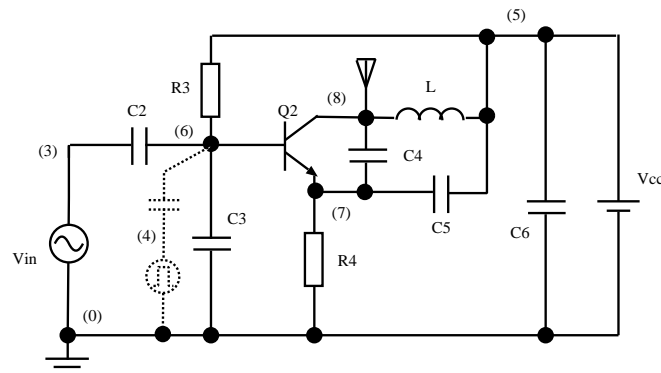


図 13: 変調回路

```

AM gen.
Vin 3 0 sin(0 1 10000)
C2 6 3 4.70uF
Vdummy 4 0 PULSE(0V 1V 10ns 0 0 100ns 1000ms)
Cd 4 6 470pF
R3 5 6 100k
C3 6 0 0.0047uF
R4 7 0 1k
Q2 8 6 7 QC1815
C4 8 7 1000pF
C5 7 5 1000pF
C6 5 0 0.47uF
L 8 5 200uH
Vcc 5 0 9V
*****2SC1815
.MODEL QC1815 NPN (IS=4E-14 BF=170 BR=3.6 VA=100 IK=0.25
+      RB=50 RC=0.76 CJC=4.8p CJE=12p TF=0.63n TR=25n)
.TRAN 2ns 200000ns 0s 1ns
.END

```

图 14: AM 变调回路

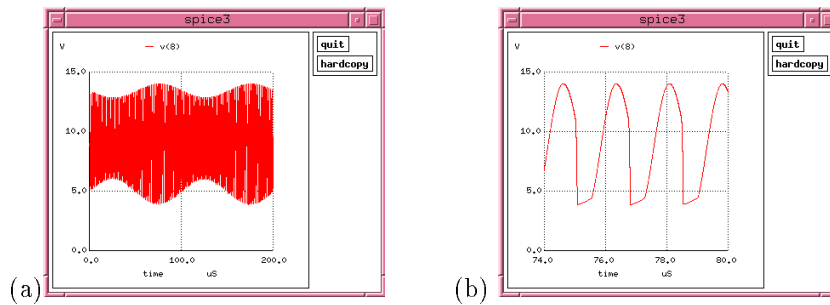


图 15: 解析结果

## 4 まとめ

以上が、SPICE の使用方法の基本です。これら以外の素子や、解析方法も多々あります。参考文献をあげておきますので、もっと詳しく知りたい方は参考にしてください。

代表的な素子についてまとめておきますので、参考にしてください。ここで、「[」と「]」で囲ってある部分は、必要なければ記述しなくても良い項目です。「{」と「}」で囲われている部分は、その内部のどれか1つの要素があれば良い項目です。

### 4.1 素子

**C: コンデンサ** C[名前] +ノード -ノード 静電容量

**D: ダイオード** D[名前] +ノード -ノード モデル名

**I: 電流源** I[名前] +ノード -ノード {[[DC] 電流値] [AC 電流  
振幅 [位相]] [過渡仕様]}

**過渡仕様** パルス波 PULSE(初期電流 パルス電流 遅延時間  
立ち上がり時間 下降時間 パルス幅 周期)

FM 変調波 SFFM(オフセット電流 電流振幅 キャリ  
ヤ周波数 変調指数 信号周波数)]

正弦波 SIN(オフセット電流 電流振幅 周波数 [遅延  
時間 [ダンピングファクタ]])

**J: j-FET** J[名前] ドレインノード ゲートノード ソースノード モデ  
ル名

**K: コイルの結合** K[名前] L コイル名 1 L コイル名 2 結合係数

**L: コイル** L[名前] +ノード -ノード インダクタンス値

**M: MOS-FET** M[名前] ドレインノード ゲートノード ソース  
ノード モデル名

**Q: トランジスタ** コレクタノード ベースノード エミッタノード モデ  
ル名

**R: 抵抗** R[名前] +ノード -ノード 抵抗値

**V: 電圧源** V[名前] +ノード -ノード {[[DC] 電圧値] [AC 電圧値  
[位相]] [過渡仕様]}

**過渡仕様** パルス波 PULSE(初期電圧 パルス電圧 遅延時間  
立ち上がり時間 下降時間 パルス幅 周期)

FM 変調波 SFFM(オフセット電圧 電圧振幅 キャリ  
ヤ周波数 変調指数 信号周波数]

正弦波 SIN(オフセット電圧 電圧振幅 周波数 [遅延  
時間] [ダンピングファクタ])

#### 4.1.1 大きさ指定について

本文の中で「k」や「u」といった大きさを指定する表示を使用しました。こ  
こで、その意味を表1にまとめておきます。ここで、「MEG」とすべきところ  
を「M」としてしまうミスが多くみられますので、注意しておきます。(「M」  
や「m」は共に  $10^{-3}$  です。)

表 1: 大きさの表し方

シンボル	値	対数表示
F, f	$10^{-15}$	1E-15
P, p	$10^{-12}$	1E-12
N, n	$10^{-9}$	1E-9
U, u	$10^{-6}$	1E-6
M, m	$10^{-3}$	1E-3
K, k	$10^3$	1E3
MEG, meg	$10^6$	1E6
G, g	$10^9$	1E9
T, t	$10^{12}$	1E12

## 参考文献

- [1] John Keown : 「SPICE による電子回路設計 (回路シミュレータ PSpice 入門)」, 東京電機大学出版局 (2001).
- [2] 小高明夫, 佐藤邦夫: 「[改訂版] SPICE による電子回路の基礎」, 東海大学出版会 (1995).
- [3] 黒田徹: 「はじめてのトランジスタ回路設計 — 回路を設計製作して SPICE で検証! —」, CQ 出版社 (2002).
- [4] <http://infopad.eecs.berkeley.edu/icdesign/SPICE/>
- [5] トランジスタ技術 「特集: 初めての回路シミュレーション」, CQ 出版社 (2002. 5).