

# MATLAB言語を用いた回路シミュレータの開発

東京工業大学 大学院理工学研究科 松澤・岡田研究室  
盛 健次、松澤 昭

## 1. 研究背景

- 東京工業大学では、「電氣的モデリングとシミュレーション」という講義を新設した。
- その講義では、以下の内容を教えることにより、シミュレータの原理を学ぶことを目的としている。
  - ①回路シミュレータ、②MOSFETモデル、③電磁界シミュレータ、④第一原理計算
- 回路シミュレータの原理を理解する為に、MATLAB言語を用いたプログラムを作成した。
- 線形回路(抵抗、容量、インダクタ、電圧源等を含む)から、非線形回路(ダイオード、MOSFET等)までの汎用回路シミュレータ(SPICE)の開発を教えている。

## 2. 回路シミュレータのフロー図

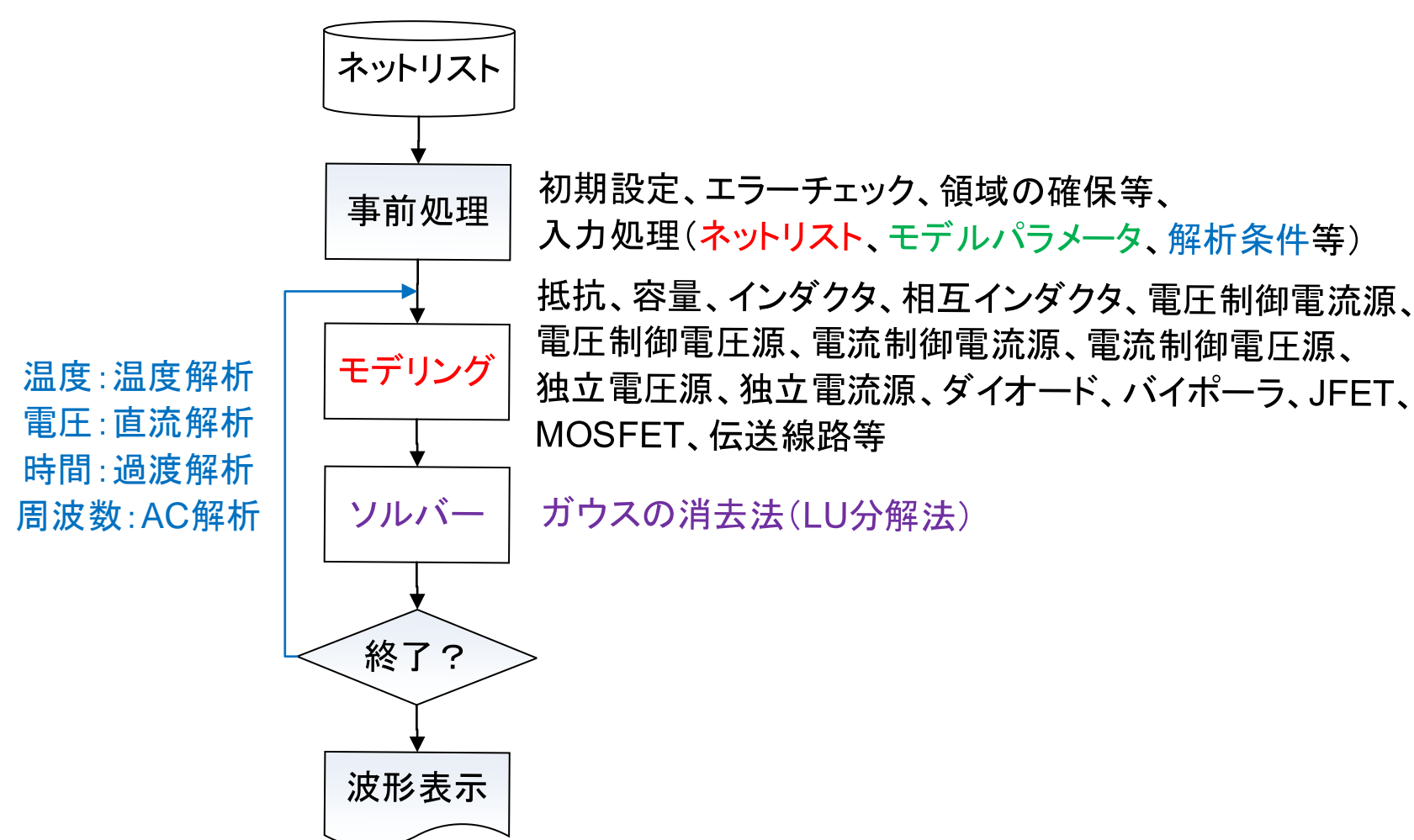


図1 回路シミュレータのフロー図

## 3. C言語とMatLab言語の比較

	C	MatLab
関数	戻り値 関数名(引数) { return(値); }	function 戻り値 = 関数名(引数) 戻り値 = 値; end
入出力	scanf(書式, 変数); printf(書式, 変数);	textscan(fid,書式, 変数); fprintf(fid,書式, 変数);
配列	#include "****.h" double int 構造体配列が可能 mallocによるメモリの自動取得	global 変数 A[] 配列 A[] セル配列 A.Name 構造体配列 A.Pnode 構造体配列 A.Nnode 構造体配列
反復	for (START; STOP; STEP) { 処理 }	for 変数 = START:STEP:STOP 処理 end
分岐	if (条件) (処理) else (処理)	if 条件 処理 elseif 条件 処理 else 処理 end

## 4. 1 2個の抵抗から成る回路

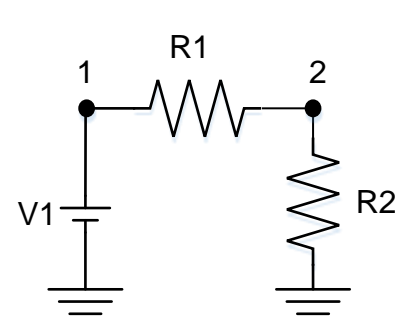


図2 抵抗回路

R1=R2=10Ω、V1に10V、V0をGNDに接続したとする。

```
function A=RES_load(A,NP,NN,RES)
G=1.0/RES;
if NP ~= 0 A(NP,NP)=A(NP,NP)+G; end
if NN ~= 0 A(NN,NN)=A(NN,NN)+G; end
if NP ~= 0 && NN ~= 0
A(NP,NN)=A(NP,NN)-G;
A(NN,NP)=A(NN,NP)-G;
end
end
抵抗モデル
```

```
% example01
A=[0,0,0;0,0,0;0,0,0];
B=[0;0;0];
A(1,1)=A(1,1)+1/10;
A(2,2)=A(2,2)+1/10;
A(1,2)=A(1,2)-1/10;
A(2,1)=A(2,1)-1/10;
A(2,2)=A(2,2)+1/10;
A(1,3)=A(1,3)+1.0;
A(3,1)=A(3,1)+1.0;
B(3)=B(3)+10;
A=sparse(A);
B=sparse(B);
X=A\B; % inv(A)*B;
A,B,X
```

```
% example02
A=[0,0,0;0,0,0;0,0,0];
B=[0;0;0];
R1=10;
R2=10;
V1=10;
IBR=2+1;
A=RES_load(A,1,2,R1);
A=RES_load(A,2,0,R2);
[A,B]=VSRC_load(A,B,1,0,IBR,V1);
A=sparse(A);
B=sparse(B);
X=A\B; % inv(A)*B;
A,B,X
```

```
function [A,B]=VSRC_load(A,B,NP,NN,IBR,E)
if NP == 0
A(NP,IBR)=A(NP,IBR)+1.0;
A(IBR,NP)=A(IBR,NP)+1.0;
end;
if NN == 0
A(NN,IBR)=A(NN,IBR)-1.0;
A(IBR,NN)=A(IBR,NN)-1.0;
end;
B(IBR,1)=B(IBR,1)+E;
end
```

## 4. 2 12個の抵抗から成る回路

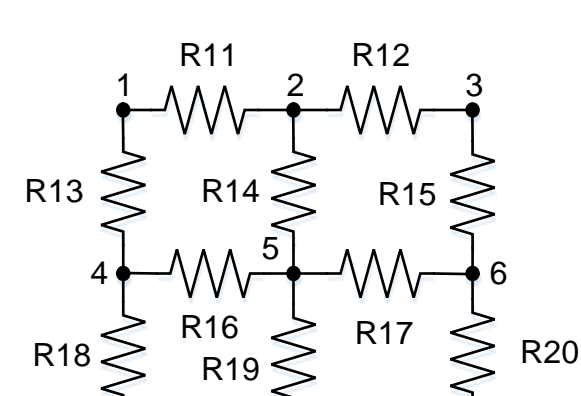


図3 12個の抵抗回路網

R11=R12=10Ω  
R16=R17=10Ω  
R21=R22=10Ω、  
R13=R14=R15=20Ω、  
R18=R19=R20=20Ω、  
V1に10V、V0をGNDに接続したとする。

```
% example03
N=9;
A=zeros(N,N);
B=zeros(N,1);
R11=10; R12=10; R16=10;
R17=10; R21=10; R22=10;
R13=20; R14=20; R15=20;
R18=20; R19=20; R20=20;
V1=10;
IBR=N;
A=RES_load(A,1,2,R11);
A=RES_load(A,2,3,R12);
A=RES_load(A,1,4,R13);
A=RES_load(A,2,5,R14);
A=RES_load(A,3,6,R15);
A=RES_load(A,4,5,R16);
A=RES_load(A,5,6,R17);
A=RES_load(A,4,7,R18);
A=RES_load(A,5,8,R19);
A=RES_load(A,6,0,R20);
A=RES_load(A,7,8,R21);
A=RES_load(A,8,0,R22);
[A,B]=VSRC_load(A,B,1,0,IBR,V1);
A=sparse(A);
B=sparse(B);
X=A\B; % inv(A)*B;
A,B,X
```

```
% example04
N=9;
A=zeros(N,N);
B=zeros(N,1);
R11=10; R12=10; R16=10; R17=10; R21=10; R22=10;
R13=20; R14=20; R15=20; R18=20; R19=20; R20=20;
fprint('example04.txt');
K=12;
DATA1=[R11;R12;R13;R14;R15;R16;
R17;R18;R19;R20;R21;R22];
DATA2=[1,2,3,1,4,2,5,3,6,4,5;
5,6,4,7,5,8,6,0,7,8,0];
DATA3=[10,0;10,0;20,0;20,0;20,0;10,0;
10,0;20,0;20,0;20,0;10,0,10,0];
V1=10;
% 解析
N=9; A=zeros(N,N); B=zeros(N,1);
IBR=N;
for i=1:K
Node1=DATA2(i,1);
Node2=DATA2(i,2);
R_val =DATA3(i);
A=RES_load(A,Node1,Node2,R_val);
end
[A,B]=VSRC_load(A,B,1,0,IBR,V1);
A=sparse(A);
B=sparse(B);
X=A\B; % X=inv(A)*B;
A,B,X
```

```
% example05
fprint('example05.txt');
DATA={ 'resistor','R11','N1','N2',10,0; 'resistor','R12','N2','N3',10,0; 'resistor','R13','N1','N4',20,0; 'resistor','R14','N2','N5',20,0;
'resistor','R15','N3','N6',20,0; 'resistor','R16','N4','N5',10,0; 'resistor','R17','N5','N6',10,0; 'resistor','R18','N4','N7',20,0;
'resistor','R19','N5','N8',20,0; 'resistor','R20','N6','0',20,0; 'resistor','R21','N7','N8',10,0; 'resistor','R22','N8','0',10,0;
'vsource','V1','N1','0',10,0};
K=length(DATA);
[DATA,JUNODE]=ERRCHK(DATA,K); % エラーチェック
JUNODE=[];
[NN,N]=SETUP(DATA,K,JUNODE); % セットアップ
% 解析
A=zeros(N,N); B=zeros(N,1);
IBR=NN;
for i=1:K
switch DATA(i,1)
case 'resistor'
N1=DATA(i,3);
N2=DATA(i,4);
RV=DATA(i,5);
A=RES_load(A,N1,N2,RV);
case 'vsource'
IBR=IBR+1;
N1=DATA(i,3);
N2=DATA(i,4);
EV=DATA(i,5);
[A,B]=VSRC_load(A,B,N1,N2,IBR,EV);
end
end
X=A\B; % X=inv(A)*B;
A,B,X
```

## 抵抗回路網の汎用回路シミュレータ

```
% example06
fprint('example06.txt');
[DATA,K]=SPICE_ReadIn('SPICE_exp.txt');
[Data,K]=SPECTRE_ReadIn('SPECTRE_exp.txt');
[DATA,JUNODE]=ERRCHK(DATA,K); % エラーチェック
[NN,N]=SETUP(DATA,K,JUNODE); % セットアップ
% 解析
A=zeros(N,N); B=zeros(N,1);
IBR=NN;
[A,B]=LOAD(DATA,K,A,B,IBR);
X=A\B; % X=inv(A)*B;
A,B,X
```

```
function [A,B]=LOAD(DATA,K,A,B,IBR)
for i=1:K
switch DATA(i,1)
case 'resistor'
N1=DATA(i,3); N2=DATA(i,4); RV=DATA(i,5);
A=RES_load(A,N1,N2,RV);
case 'vsource'
IBR=IBR+1;
N1=DATA(i,3); N2=DATA(i,4); EV=DATA(i,5);
[A,B]=VSRC_load(A,B,N1,N2,IBR,EV);
end
end
end
```

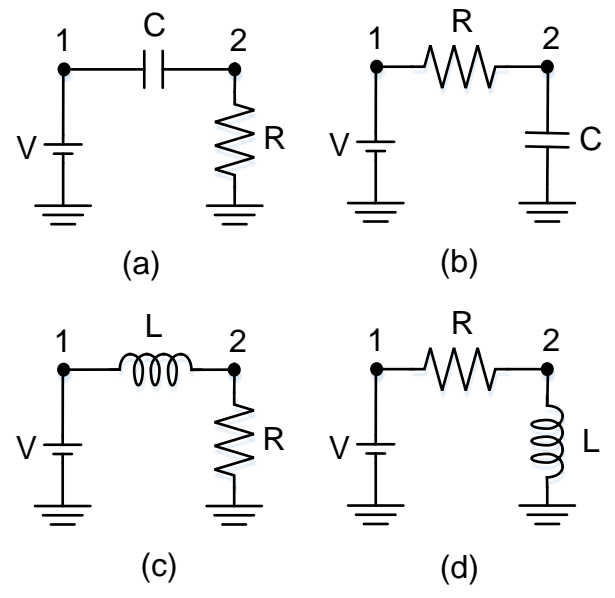
```
function [DATA,y]=SPICE_ReadIn(filename)
fid=fopen(filename);
DATA=[];
y=0;
tline=fgets(fid);
while ischar(tline)
tline=fgets(fid);
if strcmp(tline,'R',1) == 1 y=y+1; DATA=INP2R(DATA,tline); end
if strcmp(tline,'V',1) == 1 y=y+1; DATA=INP2V(DATA,tline); end
end
fclose(fid);
end
```

```
function DATA = INP2R(DATA,tline)
[Rname,tline] = strtok(tline);
[nnam1,tline] = strtok(tline);
[nnam2,tline] = strtok(tline);
[R_val,tline] = strtok(tline);
node1=str2num(nnam1);
node2=str2num(nnam2);
RVAL=str2double(R_val);
DATA = [DATA; {'resistor',Rname,node1,node2,RVAL}];
end
```

```
function DATA = INP2V(DATA,tline)
[Vname,tline] = strtok(tline);
[nnam1,tline] = strtok(tline);
[nnam2,tline] = strtok(tline);
[V_val,tline] = strtok(tline);
node1=str2num(nnam1);
node2=str2num(nnam2);
VVAL=str2double(V_val);
DATA = [DATA; {'vsource',Vname,node1,node2,VVAL}];
end
```



# 4. 3 抵抗、容量、インダクタによる回路



R=10Ω, C=1pF, L=1nH, V=10V, V0をGNDに接続したとする。

```

CR circuit
C1 N1 N2 1.0E-12
R1 N2 0 10.0
V1 N1 0 10.0
.TRAN 1E-12 5E-11
.END

RC circuit
R1 N1 N2 10.0
C1 N2 0 1.0E-12
V1 N1 0 10.0
.TRAN 1E-12 5E-11
.END

LR circuit
L1 N1 N2 1.0E-9
R1 N2 0 10.0
V1 N1 0 10.0
.TRAN 1E-11 5E-10
.END

RL circuit
R1 N1 N2 10.0
L1 N2 0 1.0E-9
V1 N1 0 10.0
.TRAN 1E-11 5E-10
.END
    
```

```

% example07
fprintf(1,'example07\n');
[DATA,K,ANAL]=SPICE_ReadIn('SPICE_exp07_2.txt');
%
[DATA,JUNODE]=ERRCHK(DATA,K); % エラーチェック
[NN,N]=SETUP(DATA,K,JUNODE); % セットアップ
%
% TRAN解析
TSTEP=ANAL{1,2}; TSTOP=ANAL{1,3}; 過渡解析条件
A=zeros(N,N); B=zeros(N,1);
X0=zeros(N,1);
IBR=NN; XX=[]; YY=[];
for t=0.0:TSTEP:TSTOP
[A,B]=LOAD(DATA,K,A,B,X0,IBR,TSTEP); % モデル
X=A\B; % X=inv(A)*B;より、正確で速い。 ソルバー
A=zeros(N,N); B=zeros(N,1); X0=X;
IBR=NN; XX=[XX,t]; YY=[YY,X];
end
plot(XX,YY(1,:),'-',XX,YY(2,:),'-');
    
```

```

% example08
fprintf(1,'example08\n');
[DATA,K,ANAL]=SPICE_ReadIn('SPICE_exp08_1.txt');
%
[DATA,JUNODE]=ERRCHK(DATA,K); % エラーチェック
[NN,N]=SETUP(DATA,K,JUNODE); % セットアップ
%
% AC解析
AANAL = ANAL{1,1}; AC解析条件
TYPE = ANAL{1,2}; NUM1 = ANAL{1,3};
FSTART= ANAL{1,4}; FSTOP = ANAL{1,5};
if TYPE == 'DEC'
FINCR=exp(log(10)*NUM1);
FSTEP=(FSTOP/FSTART)/FINCR;
end
if TYPE == 'OCT'
FINCR=exp(log(2)*NUM1);
FSTEP=(FSTOP/FSTART)/FINCR;
end
A=zeros(N,N); B=zeros(N,1);
X0=zeros(N,1);
IBR=NN; XX=[]; YY=[];
for f=FSTART:FSTEP:FSTOP
[A,B]=ACLOAD(DATA,K,A,B,IBR,f); % ACモデル
X=A\B; % X=inv(A)*B;より、正確で速い。 ソルバー
A=zeros(N,N); B=zeros(N,1); X0=X;
IBR=NN; XX=[XX,f]; YY=[YY,X];
end
semilogx(XX,YY(1,:),'-',XX,YY(2,:),'-');
    
```

```

function [A,B]=LOAD(DATA,K,A,B,X0,IBR,H)
for i=1:K
switch DATA{i,1}
case 'resistor'
NP=DATA{i,3}; NN=DATA{i,4}; RES=DATA{i,5};
A=RES_load(A,NP,NN,RES); % 抵抗モデル
case 'capacitor'
NP=DATA{i,3}; NN=DATA{i,4}; CAP=DATA{i,5};
[A,B]=CAP_load(A,B,X0,NN,CAP,H); % 容量モデル
case 'inductor'
IBR=IBR+1;
NP=DATA{i,3}; NN=DATA{i,4}; IND=DATA{i,5};
[A,B]=IND_load(A,B,X0,NN,IBR,IND,H);
case 'vsource'
IBR=IBR+1;
NP=DATA{i,3}; NN=DATA{i,4}; VOL=DATA{i,5};
[A,B]=VSRC_load(A,B,NP,NN,IBR,VOL); % 独立電圧源モデル
end
end
    
```

```

function [A,B]=ACLOAD(DATA,K,A,B,IBR,f)
for i=1:K
switch DATA{i,1}
case 'resistor'
Pnode=DATA{i,3}; Nnode=DATA{i,4}; RES=DATA{i,5};
A=RES_load(A,Pnode,Nnode,RES); % 抵抗モデル
case 'capacitor'
Pnode=DATA{i,3}; Nnode=DATA{i,4}; CAP=DATA{i,5};
A=CAP_acload(A,Pnode,Nnode,CAP,f); % 容量ACモデル
case 'inductor'
IBR=IBR+1;
Pnode=DATA{i,3}; Nnode=DATA{i,4}; IND=DATA{i,5};
A=IND_acload(A,Pnode,Nnode,IBR,IND,f);
case 'vsource'
IBR=IBR+1;
Pnode=DATA{i,3}; Nnode=DATA{i,4}; VOL=DATA{i,5};
[A,B]=VSRC_load(A,B,Pnode,Nnode,IBR,VOL); % 独立電圧源モデル
end
end
    
```

```

function DCTRAN(DATA,K,NN,N,ANAL, i)
XX=[]; YY=[];
%
% TRAN解析
%
TSTEP=ANAL{i,2}; TSTOP=ANAL{i,3}; 過渡解析条件
A=zeros(N,N); B=zeros(N,1);
X0=zeros(N,1); IBR=NN;
for t=0.0:TSTEP:TSTOP
[A,B]=LOAD(DATA,K,A,B,X0,IBR,TSTEP); % モデル
X=A\B; % X=inv(A)*B;より、正確で速い。 ソルバー
A=zeros(N,N); B=zeros(N,1);
X0=X; IBR=NN;
XX=[XX,t]; YY=[YY,X];
end
plot(XX,YY(1,:),'-',XX,YY(2,:),'-');
end
    
```

```

function ACAN(DATA,K,NN,N,ANAL, i)
XX=[]; YY=[];
%
% AC解析
%
TYPE = ANAL{i,2}; NUM1 = ANAL{i,3}; AC解析条件
FSTART= ANAL{i,4}; FSTOP = ANAL{i,5};
if TYPE == 'DEC' FINCR=exp(log(10)*NUM1); end
if TYPE == 'OCT' FINCR=exp(log(2)*NUM1); end
FSTEP=(FSTOP/FSTART)/FINCR
A=zeros(N,N); B=zeros(N,1); X0=zeros(N,1);
IBR=NN;
for f=FSTART:FSTEP:FSTOP
[A,B]=ACLOAD(DATA,K,A,B,IBR,f); % ACモデル
X=A\B; % X=inv(A)*B;より、正確で速い。 ソルバー
A=zeros(N,N); B=zeros(N,1);
X0=X; IBR=NN;
XX=[XX,f]; YY=[YY,X];
end
semilogx(XX,YY(1,:),'-',XX,YY(2,:),'-');
end
    
```

## 線形回路の汎用シミュレータ

```

% example09
%
fprintf(1,'example09\n');
%
SPICE 'SPICE_exp07_1.txt'
SPICE 'SPICE_exp07_2.txt'
SPICE 'SPICE_exp07_3.txt'
SPICE 'SPICE_exp07_4.txt'
SPICE 'SPICE_exp08_1.txt'
SPICE 'SPICE_exp08_2.txt'
SPICE 'SPICE_exp08_3.txt'
SPICE 'SPICE_exp08_4.txt'
SPICE 'SPICE_exp09_1.txt'
SPICE 'SPICE_exp09_2.txt'
SPICE 'SPICE_exp09_3.txt'
SPICE 'SPICE_exp09_4.txt'

function SPICE(filename)
%
% READIN
%
[DATA,K,ANAL]=SPICE_ReadIn(filename);
%
[DATA,JUNODE]=ERRCHK(DATA,K); % エラーチェック
[NN,N]=SETUP(DATA,K,JUNODE); % セットアップ
%
% 解析
%
for i=1:AK
AKIND= ANAL{1,1};
if strcmp('TRAN',AKIND,4) == 1
DCTRAN(DATA,K,NN,N,ANAL, i); % 過渡解析
elseif strcmp('AC',AKIND,3) == 1
ACAN(DATA,K,NN,N,ANAL, i); % AC解析
end
end
end
    
```

### 抵抗素子 抵抗モデル

$$\begin{bmatrix} G_0 & -G_0 \\ -G_0 & G_0 \end{bmatrix} \cdot \begin{bmatrix} V_{N+} \\ V_{N-} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

### 容量素子 容量モデル

$$\begin{bmatrix} G_0 & -G_0 \\ -G_0 & G_0 \end{bmatrix} \cdot \begin{bmatrix} V_{N+} \\ V_{N-} \end{bmatrix} = \begin{bmatrix} I_0 \\ -I_0 \end{bmatrix}$$

```

function A=RES_load(A,NP,NN,RES)
G=1.0/RES;
if NP == 0
A(NP,NP)=A(NP,NP)+G;
end;
if NN == 0
A(NN,NN)=A(NN,NN)+G;
end;
if NP == 0 && NN == 0
A(NP,NN)=A(NP,NN)-G;
A(NN,NP)=A(NN,NP)-G;
end;
end
    
```

```

function A=CAP_load(A,B,X,NP,NN,CAP,H)
G0=CAP/H; I0=G0*(X(NP)-X(NN));
if NP == 0
A(NP,NP)=A(NP,NP)+G0;
B(NP,1)=B(NP,1)+I0;
end;
if NN == 0
A(NN,NN)=A(NN,NN)+G0;
B(NN,1)=B(NN,1)-I0;
end;
if NP == 0 && NN == 0
A(NP,NN)=A(NP,NN)-G0;
A(NN,NP)=A(NN,NP)-G0;
end;
end
    
```

### 独立電圧源素子

$$\begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & -1 \\ 1 & -1 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_{N+} \\ V_{N-} \\ I_E \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ E \end{bmatrix}$$

### インダクタ素子 インダクタモデル

$$\begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & -1 \\ 1 & -1 & -R_0 \end{bmatrix} \cdot \begin{bmatrix} V_{N+} \\ V_{N-} \\ I_L \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -V_0 \end{bmatrix}$$

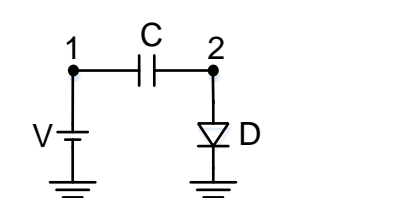
```

function [A,B]=VSRC_load(A,B,NP,NN,IBR,E)
if NP == 0
A(NP,IBR)=A(NP,IBR)+1.0;
A(IBR,NP)=A(IBR,NP)+1.0;
end;
if NN == 0
A(NN,IBR)=A(NN,IBR)-1.0;
A(IBR,NN)=A(IBR,NN)-1.0;
end;
B(IBR,1)=B(IBR,1)+E;
end
    
```

```

function [A,B]=IND_load(A,B,X,NP,NN,IBR,IND)
R0=IND/H; V0=R0*(X(NP)-X(NN));
if NP == 0
A(NP,IBR)=A(NP,IBR)+1.0;
A(IBR,NP)=A(IBR,NP)+1.0;
end;
if NN == 0
A(NN,IBR)=A(NN,IBR)-1.0;
A(IBR,NN)=A(IBR,NN)-1.0;
end;
A(IBR,IBR)=A(IBR,IBR)-R0;
B(IBR,1)=B(IBR,1)-V0;
end
    
```

# 5. 非線形回路のプログラミング



C=50pF, RS=10Ω, IS=135pA, N=1.02, BV=9.03, V=1V, V0をGNDに接続したとする。

```

DIODE circuit
C1 N1 N2 50.0E-12
D1 N2 0 DMOD
V1 N1 0 1.0
.TRAN 1E-9 1E-6
.END

% example10
%
global fwid
fwid=fopen('test.txt','w');
%fwid=1;
fprintf(fwid,'*****\n');
fprintf(fwid,' example10 \n');
fprintf(fwid,'*****\n');
%
%SPICE 'SPICE_exp10_0.txt'
%SPICE 'SPICE_exp10_1.txt'
%SPICE 'SPICE_exp10_2.txt'
%SPICE 'SPICE_exp10_3.txt'
%
fclose(fwid);
    
```

```

function SPICE(filename)
global MODE MODEDC INITF
global VNAM VSTART VSTOP VSTEP
global TSTEP TSTOP
global ACANAL TYPE NUM1 FSTART FSTOP
flag=0;
flag=SPICE_INIT(flag); % 初期値
[DATA,K,ANAL,AK]=SPICE_ReadIn(filename); % READIN
%
[DATA,JUNODE]=ERRCHK(DATA,K); % エラーチェック
[NN,N]=SETUP(DATA,K,JUNODE); % セットアップ
%
% 解析
%
DC解析、過渡解析、AC解析等
for i=1:AK
AKIND=ANAL{1,1};
switch AKIND
case 'OP'
MODE = 1; MODEDC = 1; INITF=1;
DCTRAN(DATA,K,NN,N);
case 'DC'
MODE = 1; MODEDC = 3; INITF=1;
VNAM =ANAL{i,2}; VSTART=ANAL{i,3};
VSTOP =ANAL{i,4}; VSTEP =ANAL{i,5};
TSTEP=0; TSTOP=0;
DCTRAN(DATA,K,NN,N);
case 'TRAN'
MODE = 2; INITF=1;
VNAM='0';
TSTEP =ANAL{i,2}; TSTOP =ANAL{i,3};
DCTRAN(DATA,K,NN,N);
case 'AC'
MODE = 3; INITF=1;
ACANAL= ANAL{i,1};
TYPE = ANAL{i,2}; NUM1 = ANAL{i,3};
FSTART= ANAL{i,4}; FSTOP = ANAL{i,5};
ACAN(DATA,K,NN,N);
end
end
end
    
```

```

function DCTRAN(DATA,K,NN,N)
global fwid
global MODE MODEDC INITF FLAG
global VNAM VSTART VSTOP VSTEP
global TSTEP TSTOP TSTART
XX=[]; YY=[];
X0=zeros(N,1);
switch MODE
case 1 % DC解析
t=0; QC=0; QDO=0; VDO=0;
switch MODEDC
case 1 % DC動作点解析
v=0; ITL=30;
[X,QC,QDO,VDO]=ITER8(DATA,K,NN,N,X0,t,TSTEP,VNAM,v,QC,QDO,VDO,ITL);
case 3 % DC伝達曲線の解析
ITL=30;
for v=VSTART:VSTEP:VSTOP
FLAG=0;
[X,QC,QDO,VDO]=ITER8(DATA,K,NN,N,X0,t,TSTEP,VNAM,v,QC,QDO,VDO,ITL);
end
end
case 2 % TRAN解析
v=0; QC=0; QDO=0; VDO=0; ITL=5;
for t=0.0:TSTEP:TSTOP
FLAG=1;
[X,QC,QDO,VDO]=ITER8(DATA,K,NN,N,X0,t,TSTEP,VNAM,v,QC,QDO,VDO,ITL);
end
end
plot(XX,YY(1,:),'-',XX,YY(2,:),'-');
grid on
end
    
```

```

function [X,QC,QDO,VDO]=ITER8(DATA,K,NN,N,X0,t,TSTEP,VNAM,v,QC,QDO,VDO,ITL)
global fwid
global RELTOL VNTOL
for loop = 1:ITL
A=zeros(N,N); B=zeros(N,1); IBR=NN;
[A,B,QC,QDO,VDO]=LOAD(DATA,K,A,B,X0,IBR,TSTEP,VNAM,v,QC,QDO,VDO,loop);
X=A\B;
noncon=0;
if loop == 1
for j=1:NN
Vold=X0(j); Vnew=X(j);
TOL=RELTOL*max(abs(Vold),abs(Vnew))+VNTOL;
if abs(Vold-Vnew) > TOL noncon=noncon+1; end
end
end
X0=X;
end
end
function [A,B,QC,QDO,VDO]=LOAD(DATA,K,A,B,X0,IBR,VNAM,v,QC,QDO,VDO,loop)
global fwid
for i=1:K
switch DATA{i,1}
case 'resistor'
A=RES_load(A,DATA,i);
case 'capacitor'
[A,B,QC]=CAP_load(A,B,X0,DATA,i,QC,H);
case 'inductor'
IBR=IBR+1;
[A,B]=IND_load(A,B,X0,DATA,i,IBR,H);
case 'vsource'
IBR=IBR+1;
[A,B]=VSRC_load(A,B,DATA,i,IBR,VNAM,v);
case 'diode'
[A,B,QDO,VDO]=DIO_load(A,B,X0,DATA,i,v,QDO,VDO,loop,H);
end
end
end
    
```

```

function [A,B,QC]=CAP_load(A,B,X0,DATA,i,QC,H)
global MODE MODEDC INITF FLAG
NP=DATA{i,3};
NN=DATA{i,4};
CAP=DATA{i,5};
%
% 電圧ベクトルの計算
%
if NP == 0 && NN == 0
VCAP=X0(NP)-X0(NN);
else NP == 0
VCAP=X0(NP);
else NN == 0
VCAP=X0(NN);
end
%
% 電流ベクトル
%
if FLAG == 1 QC=CAP*VCAP; FLAG=0; end
[GC,CC]=INTEGR8(QC,CAP,H);
fprintf(1,'CAP -- GC=%e CC=%e\n',GC,CC);
if NP == 0 B(NP,1)=B(NP,1)+CC; end
if NN == 0 B(NN,1)=B(NN,1)-CC; end
end;
end
    
```