Qucs, SPICE and Modelica equation-defined modelling techniques for the construction of compact device models based on a common model template structure

M.E. Brinson
Centre for Communications Technology
London Metropolitan University
London, UK
mbrin72043@yahoo.co.uk

S. Jahn
Qucs Project Manager
Munich
Germany
stefan@lkcc.org

H.Nabijou
Centre for Communications Technology
London Metropolitan University
London, UK
h.nabijou@londonmet.ac.uk

- **Background**
- **A compact semiconductor model template for Qucs, SPICE and Modelica**
- **Building an extended compact equation-defined device equivalent circuit for the SPICE semiconductor diode model**
- **Diode performance evaluation using S parameters, DC, AC, transient simulation, AC and time domain noise analysis, plus harmonic balance analysis**
- **Verilog-A and Modelica diode models**
- **Future directions**

Presented on 8 April 2011 at the MOS-AK/GSA Workshop on "Frontiers of Compact Modeling for Advanced Analog/RF Applications", Paris, 2011

Background

- Recent trends in compact device modelling suggest that there is a growing interest in equation-defined device techniques and Verilog-A among the device modelling community
- Current releases of the popular GPL **Qucs** circuit simulator, the freely available Linear Technology **LTspiceIV**® circuit simulator and the open source Modelica Consortium **OpenModelica** simulation environment all include equation-defined features for compact semiconductor device modelling
- A primary aim of compact equation-defined device modelling is the generation of efficient high-level hardware description language models that support a wide range of circuit simulation domains, including DC, AC, transient, S-parameter, AC and time domain noise, plus harmonic balance analysis, which can be translated to Verilog-A easily
- A second equally important aim is a high level of portability for equation-defined device models across circuit simulation platforms
- Equation-defined device models based on a simple template structure encourage the development of readable and accurate simulation models
- An extension of equation-defined model parameters allows model functionality to be selected by users, reducing the number of arithmetic computations which in turn decreases simulation times
- Production versions of equation-defined device models are easily translated to Verilog-A or the Modelica simulation language prior to compiling to C++ and merging with the control and analysis C++ code sections of a simulator

A compact semiconductor model template for Qucs, SPICE and Modelica

Template	Qucs	LTspice	Verilog-A	Modelica
Header	.Def:name nodes param="xx"..	.subckt name nodes param = xx..	module name (nodes ..)	model name Interface pin descriptions
Model Initialisation Code	Eqn:EqnX param="yy"	.param = {yy}	parameter type name=yy Analog begin @(initial_model) end	parameter type name=yy
Functional body of Model	EDD VCVS CCVS VCCS CCCS C R Current contributions Noise contributions [Voltage and current noise generators] .Def.End	B E H G F C R Current contributions Noise contributions [Diode noise generators] .ends	Current contributions Noise contributions endmodule	Equation R G C Controlled sources Current contributions Interface node current and voltage equations endname
Simulation Capabilities	DC AC TRAN S parameter analysis Noise (freq and time domains) HB (single input signal) Parameter sweep Digital (VHDL, Verilog) Optimisation	.OP .DC .TRAN .TEMP .TF .STEP .noise (freq domain) .FOUR Digital (mixed-mode)		Transient
Compact Modelling Tools	1. Interactive subckts 2. ADMS Verilog-A compiler	Interactive subckts	Verilog-A to C/C++ compiler	Modelica to C/C++ compiler

NOTE: Mathematical expressions require a full range of operators and functions (similar to Verilog-A) plus some form of "if then else statement" [for example, a ? b : c or if(a, b, c) syntax]

Specification for an extended SPICE semiconductor diode model

Format	User Selection switches [0=off, 1=on]	Physical parameters	Operating region or property
SPICE		IS, N, XTI, EG,AREA TEMP, TNOM	Non-linear DC I-V characteristics
	BVSWITCH	BV, IBV	Reverse breakdown
	CdepSWITCH	CJ0, M, VJ	Depletion capacitance
	CdiffSWITCH	TT	Diffusion capacitance
	AcnoiseSWITCH	KF, AF, FFE	Small signal AC noise
EXTENDED MODEL	IrecovSWITCH	TAU	Reverse recovery characteristics
	SWSH SWF	ScaleR ScaleSH ScaleF	Large signal time domain noise

User selection switches allow different levels of model to be constructed to meet specific circuit and simulation requirements: other physical effects can be added as needed, for example very low current or high current I-V effects

SPICE semiconductor diode model equations (simplified)

1. Non-linear DC I-V characteristics

$$I_d = IST2\cdot\left(\exp\left(\frac{V_d}{N\cdot Vt(T2)}\right)-1\right)+GMIN\cdot V_d\,,\ where\ V_d = V(ANODE\,,CATHODE)\,,$$

$$IST2 = IS\cdot AREA\cdot\left(\frac{T2}{T1}\right)^{\frac{XTN}{N}}\cdot\exp\left(\left(\frac{-E_g(300)}{Vt(T2)}\right)\cdot\left(1-\frac{T2}{T1}\right)\right),\ Vt(T2)=\frac{K\cdot T2}{q}\,,$$

$$T1 = TNOM + 273.15\,K\,,\ T2 = TEMP + 273.5\,K$$

2. Reverse breakdown

$$K2=\frac{1}{N\cdot Vt(T2)}\,,\ K5 = N\cdot Vt(T2)\,,\ IBVEFF = IBV\cdot AREA\,,$$

$$IDBV = -IST2\cdot(\exp(-BV\cdot K2)-1)$$

$$BVEFF = (IBVEFF > IDBV)\ ?\ BV - K5\cdot\ln\left(\frac{IBVEFF}{IDBV}\right) : BV$$

$$I_d = -IST2\cdot(\exp(-(BVEFF - V_d)\cdot K2) - 1 + BVEFF\cdot K2)$$

3. Depletion capacitance

$$Q_{dep} = (V_d \geq 0.0)\ ?\ CJ0T2\cdot(V_d + P11\cdot V_d\cdot V_d) : P6\cdot\left(1-\left(1-\frac{V_d}{VJT2}\right)^{P7}\right)$$

$$CJ0T2 = CJ0\cdot AREA\,,\ P11=\frac{M}{(2\cdot VJ)}\,,\ P6=\frac{(CJ0T2\cdot VJT2)}{P7}\,,\ P7 = 1 - M$$

$$VJT2=\left(\frac{T2\cdot VJ}{T1}\right)-2\cdot Vt(T2)\cdot\ln\left(\frac{T2}{T1}\right)^{1.5}-\left(\left(\frac{T2 * Eg(T1)}{T1}\right)-Eg(T2)\right)$$

$$Eg(T1) = Eg - \frac{7.02\text{e-}4\cdot T1\cdot T1}{1108 + T1}\,,\ Eg(T2) = Eg - \frac{7.02\text{e-}4\cdot T2\cdot T2}{1108 + T2}$$

4. Diffusion capacitance

$$Q_{diff} = TT\cdot I_d$$

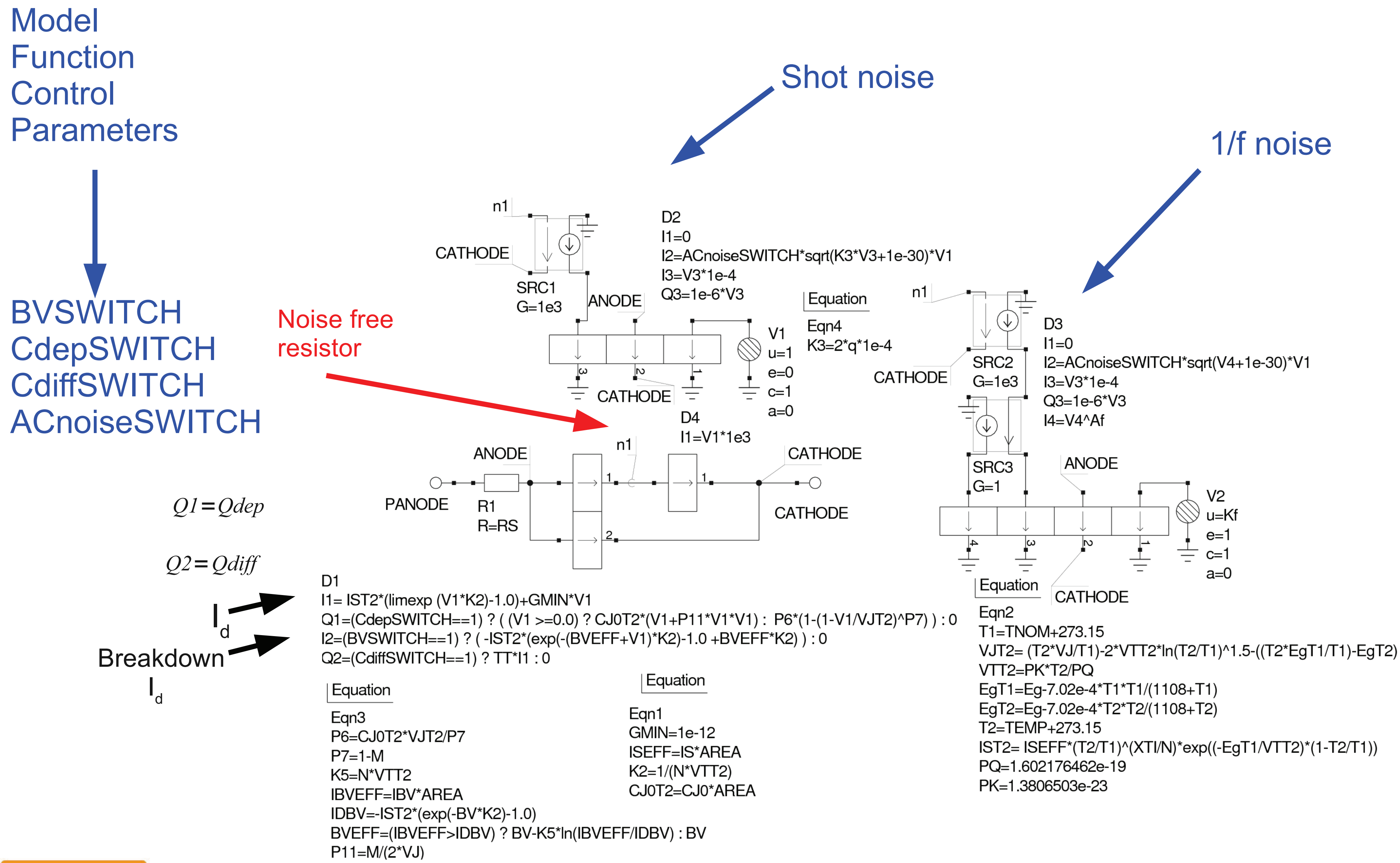
5. Noise current

$$\overline{i_d^2} = 2\cdot q\cdot I_d \Delta f + \frac{Kf\cdot I_d^{Af}}{f}\Delta f + \frac{4\cdot k.T}{RS}\Delta f$$

Diode Physical Parameters

AREA=1
N=1
IS=1e-14
XTI=3.0
TEMP=26.85
Eg=1.16
TNOM=26.85
TT=1e-9
CJ0=1e-12
VJ=1.0
M=0.5
BV=4.5
IBV=1e-3
Kf=1e-16
Af=1
RS=0.1

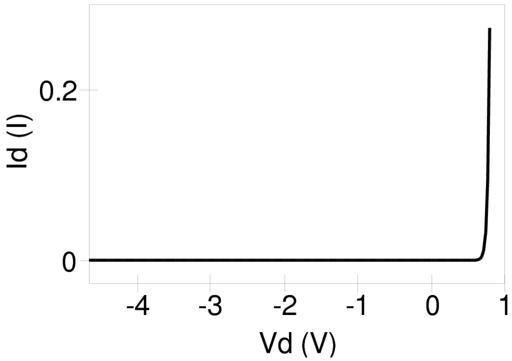
Qucs equation-defined device equivalent circuit for the SPICE semiconductor diode model

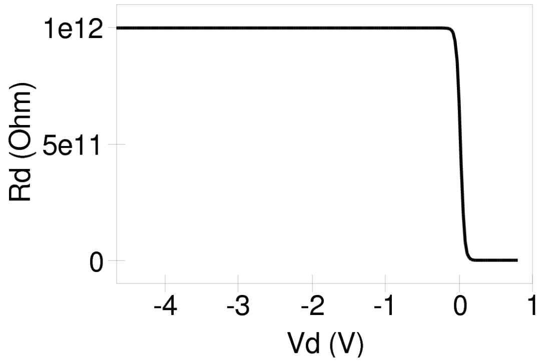


Equation EqnX blocks contain variable initialisation equations and constants. Variable expressions are converted into numerical values before a simulation starts.

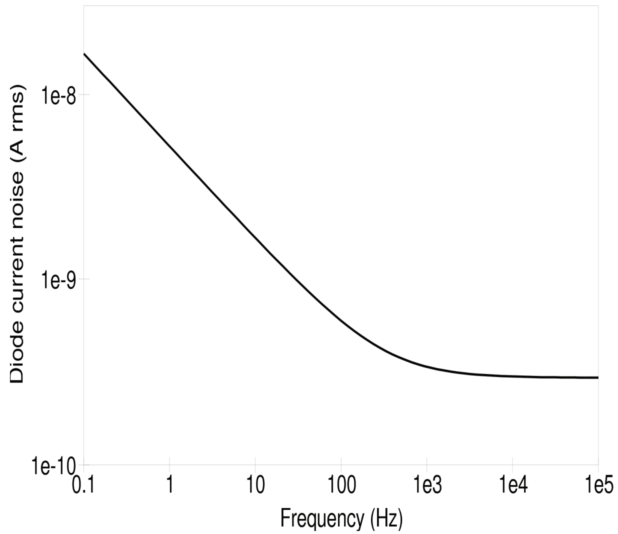
Typical SPICE diode model properties

1. I-V characteristics
[BVSWITCH=0]

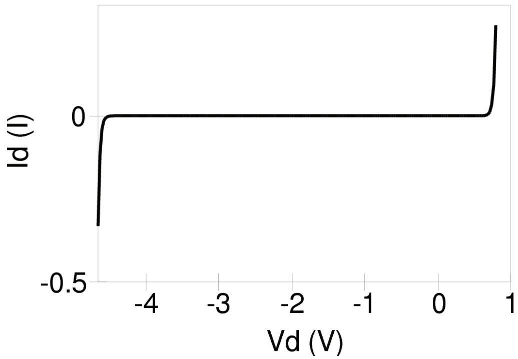


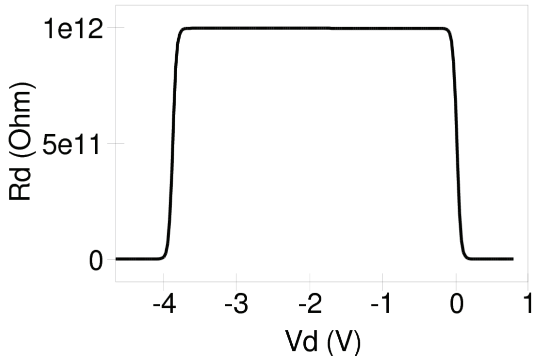


4. Flicker and shot noise
[ACnoiseSWITCH=1]

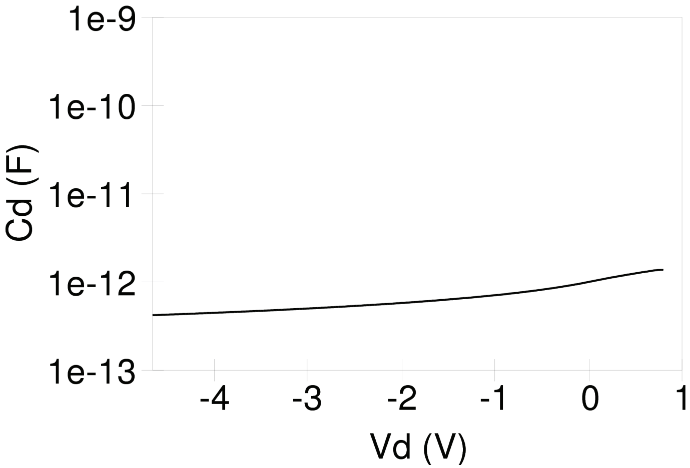


2. Reverse
breakdown
[BVSWITCH = 1]

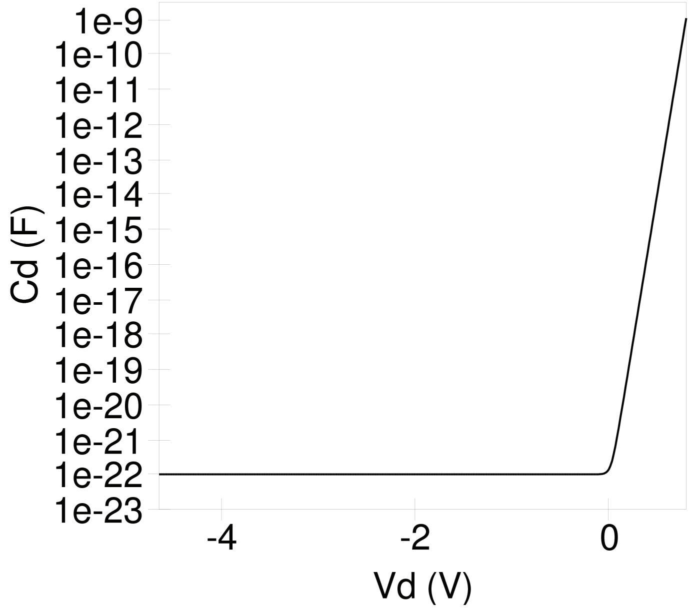




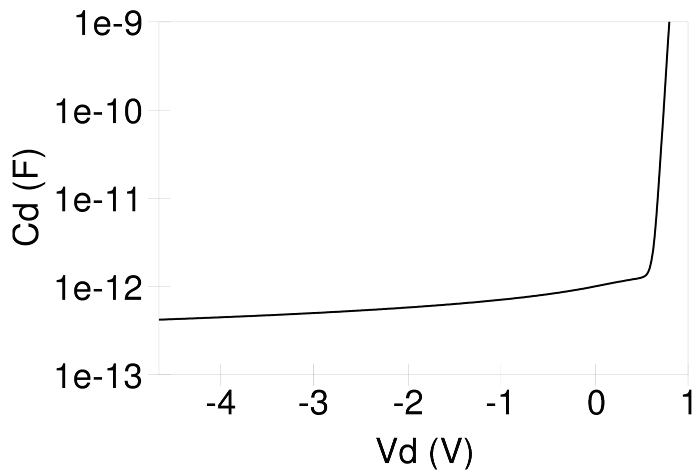
3. Capacitance



[CdepSWITCH=1, CdiffSWITCH=0]



[CdepSWITCH=0, CdiffSWITCH=1]



[CdepSWITCH=1, CdiffSWITCH=1]

SPICE equation-defined device equivalent circuit for the basic semiconductor diode model

```
.subckt ModDiodeSlide8 1 2  BVSWITCH=0 CDEPSWITCH = 0 CDIFFSWITCH = 0 ACSWITCH=0
+                    IRECSWITCH=0 AREA=1.0 N=1 XTI=3.0 ISat=1e-14 TEMP=26.85
+                    TNOM=26.85 EG=1.16 BV=100 IBV=1e-30 CJ0=1p VJ=1 M=0.5
+                    TT=1n TAU=100e-9 AF=1 KF=1e-16 RS=0.01
*
.param CJ0T2 = {CJ0*AREA}
.param ISEFF = {ISat*AREA}
.param T1    = {TNOM+273.15}
.param T2    = {TEMP+273.15}
.param EGT1  = {EG-7.02e-4*T1*T1/(1108+T1)}
.param EGT2  = {EG-7.02e-4*T2*T2/(1108+T2)}
.param P7    = 1-M
.param P6    = {CJ0T2*VJ/P7}
.param P11   = {M/(2*VJ)}
.param PK    = 1.3806503e-23
.param PQ    = 1.602176472e-19
.param VTT2  = {Pk*T2/PQ}
.param VJT2  = {(T2*VJ/T1)-2*VTT2*(LN(T2/T1)**1.5)-((T2*EGT1/T1)-EGT2)}
.param IST2  = {ISEFF*((T2/T1)**(XTI/N))*EXP((-EGT1/VTT2)*(1-T2/T1))}
.param K2    = {1/(N*VTT2)}
.param K5    = {N*VTT2}
.param IBVEFF = {IBV*AREA}
.param IDBV  = {-IST2*(exp(-BV*K2)-1.0)}
.param BVEFF = {if(IBVEFF/(2*IDBV),  BV-K5*ln(IBVEFF/IDBV), BV)}
.param GMIN  = 1e-12
.param CDEPPARAM = {if(CDEPSWITCH, 1, 0)}
.param CDIFFPARAM = {if(CDIFFSWITCH, 1, 0)}
*
RS 1 n5 {RS}
* Diode I-V characteristics
B1 n5 n1 I = IST2*(exp(V(n5, n1)*K2)-1.0)+GMIN*V(n5, n1)
Vsense n1 2 DC 0
B2 1 2 I = if( {BVSWITCH},  -IST2*(exp( -( BVEFF+V(n5, n1) )*K2)-1.0+BVEFF*K2), 0 )
* Depletion capacitance
Cd 1 2  Q = CDEPPARAM*(if(x+0.501, CJ0T2*(x+P11*x*x), P6*(1-(1-x/VJT2)**P7) ))
* diffusion capacitance
Cdiff 1 2 Q = CDIFFPARAM*TT*(IST2*(exp(V(n5,N1)*K2)-1.0))
* Diode shot and 1/f noise
F1 0 n2 Vsense 1
Lnoise n2 n3 1000
cnoise n3 n4 1000
Vnoise n4 0 dc 0
f n5 n1 Vnoise {if(ACSWITCH, 1, 0)}
Dn n3 0 Dnoise
.model Dnoise D(Is=ISat N=N AF=Af KF=KF)
.ends
```

Variable expressions

Constants

Function selection

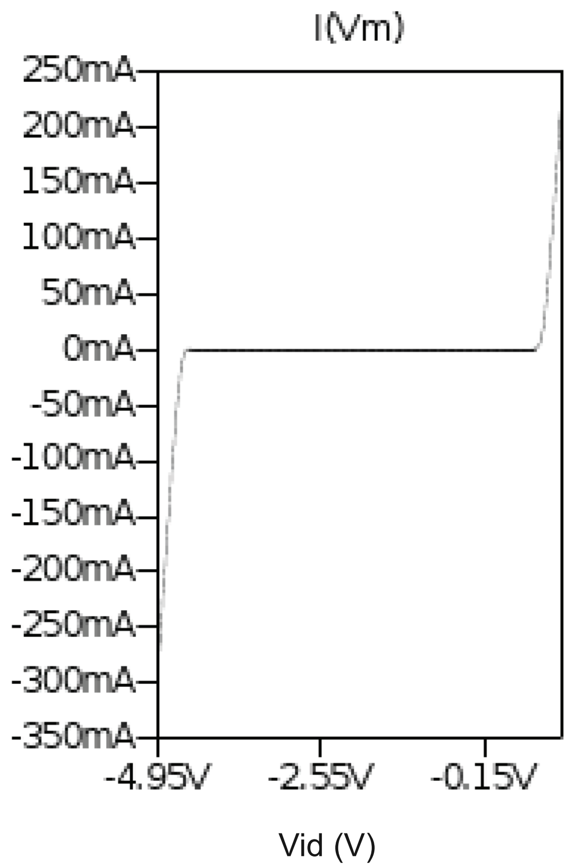
If..then..else statement

C expressed as charge Q

Noise current generated using built-in diode SPICE model

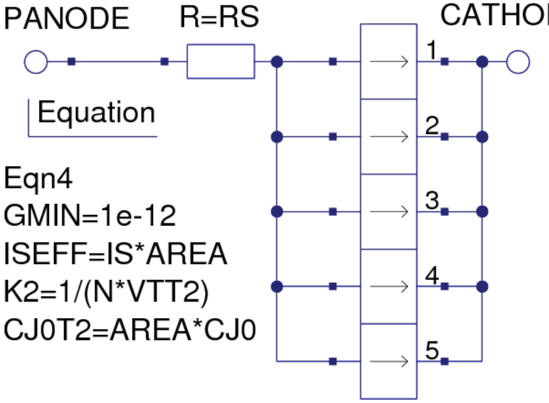
SPICE test code

```
Vid 2 0 dc 0.6
Vm 2 22 dc 0
Rs 22 3 1
X1 3 0 ModDiode BVSWITCH=1 CDEPSWITCH=0  CDIFFSWITCH=0 AREA=1 N=1 XTI=2.0
+ ISat=1e-14 TEMP=26.85 TNOM=26.85 BV=4.5 IBV=1e-3  CJ0=1p VJ=1 M=0.5 TT=1n
.op
.dc Vid -4.95 1 0.01
.end
```



Free versions of SPICE:
LTspice ®, ngspice, SPICEOPUS
NO
standardisation of extended features!

Qucs non-linear model of a step recovery diode*



Equation
```
Eqn4
GMIN=1e-12
ISEFF=IS*AREA
K2=1/(N*VTT2)
CJ0T2=AREA*CJ0
```

```
D1
I1= IST2*(limexp (V1*K2)-1.0)+GMIN*V1
Q1=(CdepPARAM==1) ? ( (V1 >=0.0) ? CJ0T2*(V1+P11*V1*V1) :  P6*(1-(1-V1/VJT2)^P7) ) : 0
I2=(BVSWITCH==1) ? (V1 <  -BV) ?   -IST2*(limexp(-(BVEFF+V1)*K2) -1 +BVEFF*K2)  : 0 : 0
Q2=(CdiffPARAM==1) ? TT*I1 : 0
I3=(BVSWITCH == 1) ? (V1 == -BV) ? IBV : 0 : 0
Q3=(IrecSWITCH == 1) ? (V1 <0.0) ? CJ0*V1 : 0 : 0
Q4=(IrecSWITCH == 1) ? (V1 > 0.0) && (V1 < FCP) ? C1*(V1+C2)^2-C3 : 0 : 0
Q5=(IrecSWITCH == 1) ? (V1 > FCP) ? CF*V1-C4 : 0 : 0
```

Q3 represents Cdep

Q4 and Q5 represent Cdiff

Equation
```
Eqn2
T1=TNOM+273.15
VJT2= (T2*VJ/T1)-2*VTT2*ln(T2/T1)^1.5-((T2*EgT1/T1)-EgT2)
VTT2=PK*T2/PQ
EgT1=Eg-7.02e-4*T1*T1/(1108+T1)
EgT2=Eg-7.02e-4*T2*T2/(1108+T2)
T2=TEMP+273.15
IST2= ISEFF*(T2/T1)^(XTI/N)*exp((-EgT1/VTT2)*(1-T2/T1))
PQ=1.602176462e-19
PK=1.3806503e-23
```

Equation
```
Eqn3
P6=CJ0T2*VJT2/P7
P7=1-M
K5=N*VTT2
IBVEFF=IBV*AREA
IDBV=-IST2*(limexp(-BV*K2)-1.0)
BVEFF=(IBVEFF>IDBV) ? BV-K5*ln(IBVEFF/IDBV) : BV
P11=M/(2*VJ)
```

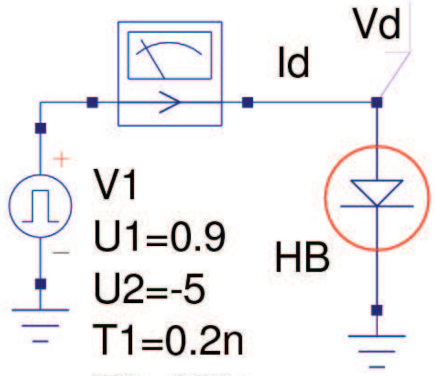
Equation
```
Eqn5
FCP=VJ
CF=TAU/RS
CM=CF-CJ0
C1=CF-CJ0/2*FCP
C2=(CJ0*FCP)/CM
C3=(CJ0*CJ0*FCP)/(2*CM)
C4=CM*FCP/2
CdepPARAM=(IrecSWITCH == 1 ) ? 0: (CdepSWITCH ==1) ? 1 : 0
CdiffPARAM=(IrecSWITCH == 1 ) ? 0: (CdiffSWITCH ==1) ? 1 : 0
```

If IrecSWITCH == 1 then CdepSWITCH and CdiffSWITCH over-riden

Small signal AC shot and 1/f noise model not included

* EagleWare Application note 22, May 2007

Qucs test circuit for a step recovery diode



V1
U1=0.9
U2=-5
T1=0.2n
T2=400n
Tr=0.001 ns
Tf=0.001 ns

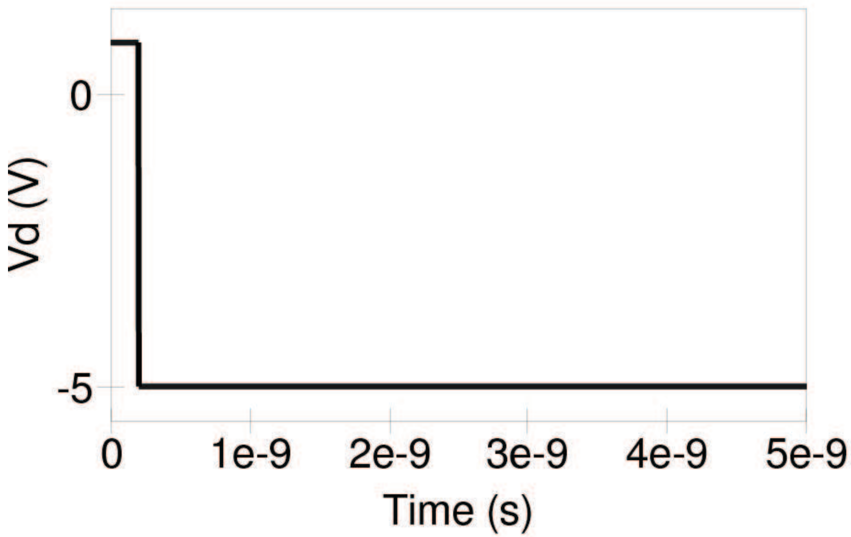
dc simulation

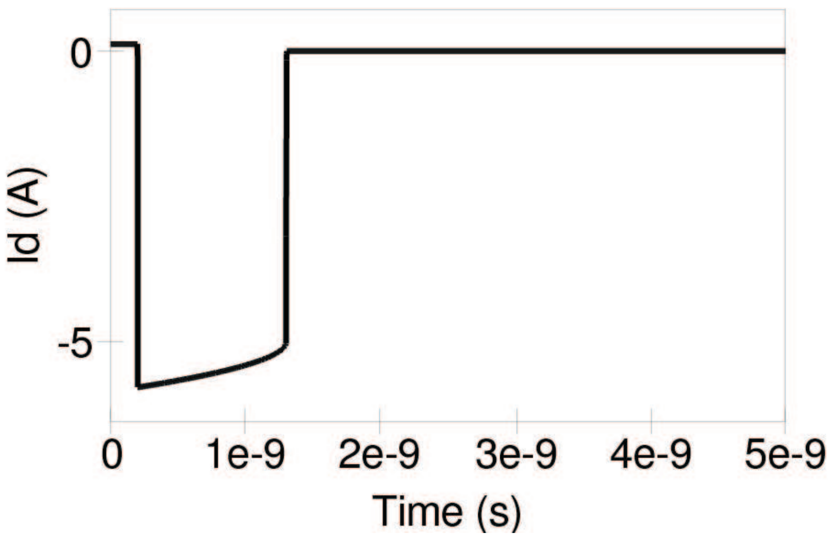
DC1

D1
AREA=1
IS=1e-14
N=1
XTI=3.0
TEMP=26.85
Eg=1.16
TNOM=26.85
TT=1n
CJ0=1e-12
VJ=1.0
M=0.5
BVSWITCH=0
CdepSWITCH=1
BV=100
CdiffSWITCH=1
IBV=1e-3
IrecSWITCH=1
TAU=10n
RS=1

transient simulation

TR1
Type=lin
Start=0
Stop=5n
Points=5000
IntegrationMethod=Gear
Order=6



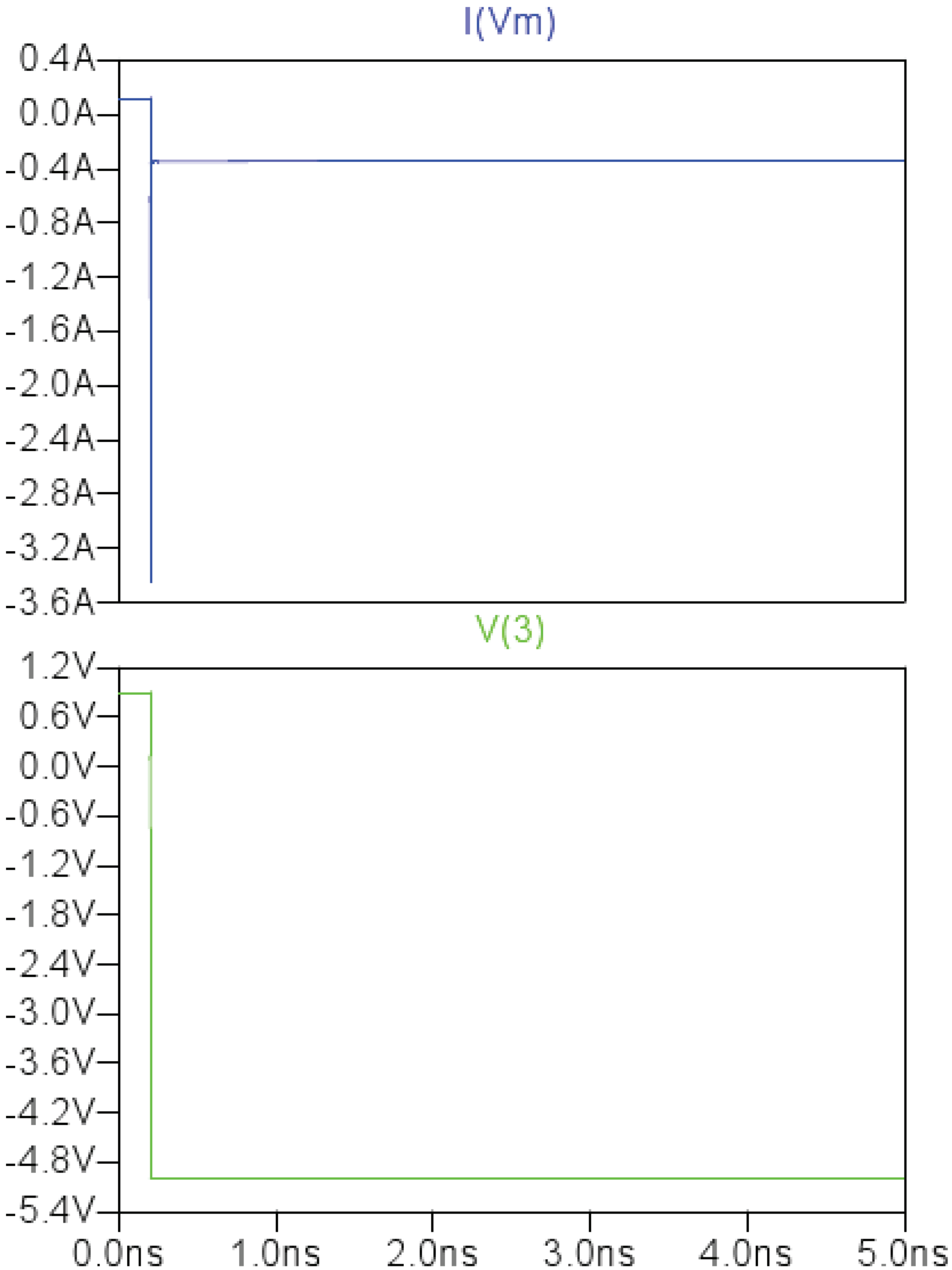


SPICE non-linear model of a step recovery diode

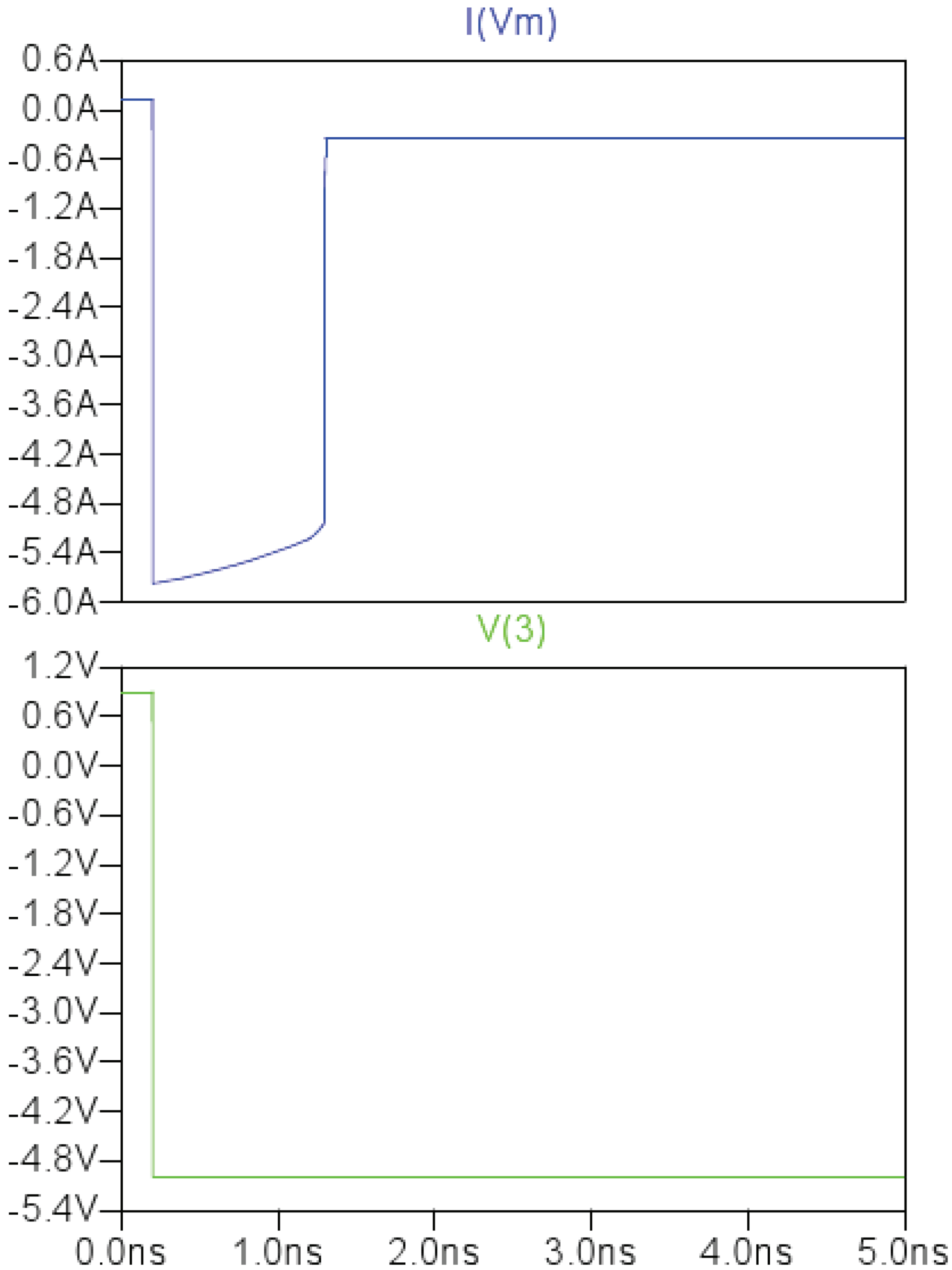
```
*SPICE Interpretive compact semiconductor diode model.
*      Based on modular compact device structure.
*
.subckt ModDiodeSlide10 1 2  BVSWITCH=0 CDEPSWITCH = 0 CDIFFSWITCH = 0
+                       IRECSWITCH=0 AREA=1.0 N=1 XTI=3.0 ISat=1e-14 TEMP=26.85
+                       TNOM=26.85 EG=1.16 BV=100 IBV=1e-30 CJ0=1p VJ=1 M=0.5
+                       TT=1n TAU=100e-9 AF=1 KF=1e-16 RS=0.01
*
.param CJ0T2 = {CJ0*AREA}
.param ISEFF = {ISat*AREA}
.param T1    = {TEMP+273.15}
.param T2    = {TNOM+273.15}
.param EGT1  = {EG-7.02e-4*T1*T1/(1108+T1)}
.param EGT2  = {EG-7.02e-4*T2*T2/(1108+T2)}
.param P7    = 1-M
.param P6    = {CJ0T2*VJ/P7}
.param P11   = {M/(2*VJ)}
.param PK    = 1.3806503e-23
.param PQ    = 1.602176472e-19
.param VTT2  = {Pk*T2/PQ}
.param VJT2  = {(T2*VJ/T1)-2*VTT2*(LN(T2/T1)**1.5)-((T2*EGT1/T1)-EGT2)}
.param IST2  = {ISEFF*((T2/T1)**(XTI/N))*EXP((-EGT1/VTT2)*(1-T2/T1))}
.param K2    = {1/(N*VTT2)}
.param K5    = {N*VTT2}
.param IBVEFF = {IBV*AREA}
.param IDBV  = {-IST2*(exp(-BV*K2)-1.0)}
.param BVEFF = {if(IBVEFF/(2*IDBV),  BV-K5*ln(IBVEFF/IDBV), BV)}
.param GMIN  = 1e-12
.param CDEPPARAM = {(1-IrecSWITCH)*if(CDEPSWITCH, 1, 0)}
.param CDIFFPARAM = {(1-IrecSWITCH)*if(CDIFFSWITCH, 1, 0)}
.param FCP       = {VJ}
.param CF        = {TAU/RS}
.param CM        = {CF-CJ0}
.param C1        = {CF-CJ0/2*FCP}
.param C2        = {(CJ0*FCP)/CM}
.param C3        = {(CJ0*CJ0*FCP)/(2*CM)}
.param C4        = {CM*FCP/2}
*
```

```
RS 1 n5 {RS}
B1 n5 n1 I = IST2*(exp(V(n5, n1)*K2)-1.0)+GMIN*V(n5, n1)
Vsense n1 2 DC 0
B2 n5 n1 I = if( {BVSWITCH},  -IST2*(exp( -( BVEFF+V(n5, n1) )*K2)-1.0+BVEFF*K2), 0 )
* Depletion capacitance
Cd n5 n1  Q = CDEPPARAM*(if(x+0.501, CJ0T2*(x+P11*x*x), P6*(1-(1-x/VJT2)**P7) ))
* diffusion capacitance
Cdiff n5 n1 Q = CDIFFPARAM*TT*(IST2*(exp(V(n5,N1)*K2)-1.0))
CQ3   n5 n1 Q = IrecSWITCH*if(x+0.5 <= 0.5, CJ0*x, 0)
CQ4   n5 n1 Q = IrecSWITCH*if((x+0.5 > 0.5) & (x+0.5 <= FCP+0.5), C1*(x+C2)**2-C3, 0)
CQ5   n5 n1 Q = IrecSWITCH*if(x+0.5 > FCP+0.5, CF*x-C4, 0)
.ends
*
Vtran 2 0 dc 0 pulse(0.9 -5 0.2n 0.001n 0.001n 400n 800n)
Vm  2 3 dc 0
X1 3  0 ModDiodeSlide10  BVSWITCH=0 CDEPSWITCH=1 CDIFFSWITCH=1
+                   IRECSWITCH=1 AREA=1 N=1 XTI=3.0 ISat=1e-14 TEMP=26.85
+                   TNOM=26.85 EG=1.16 BV=4.5 IBV=1e-3 CJ0=1p VJ=1 M=0.5
+                   TT=1e-9 TAU=10e-9 AF=1 KF=1e-16 RS=1
.op
.tran 0.0001n 5n
.end
```

SPICE step recovery diode test waveforms

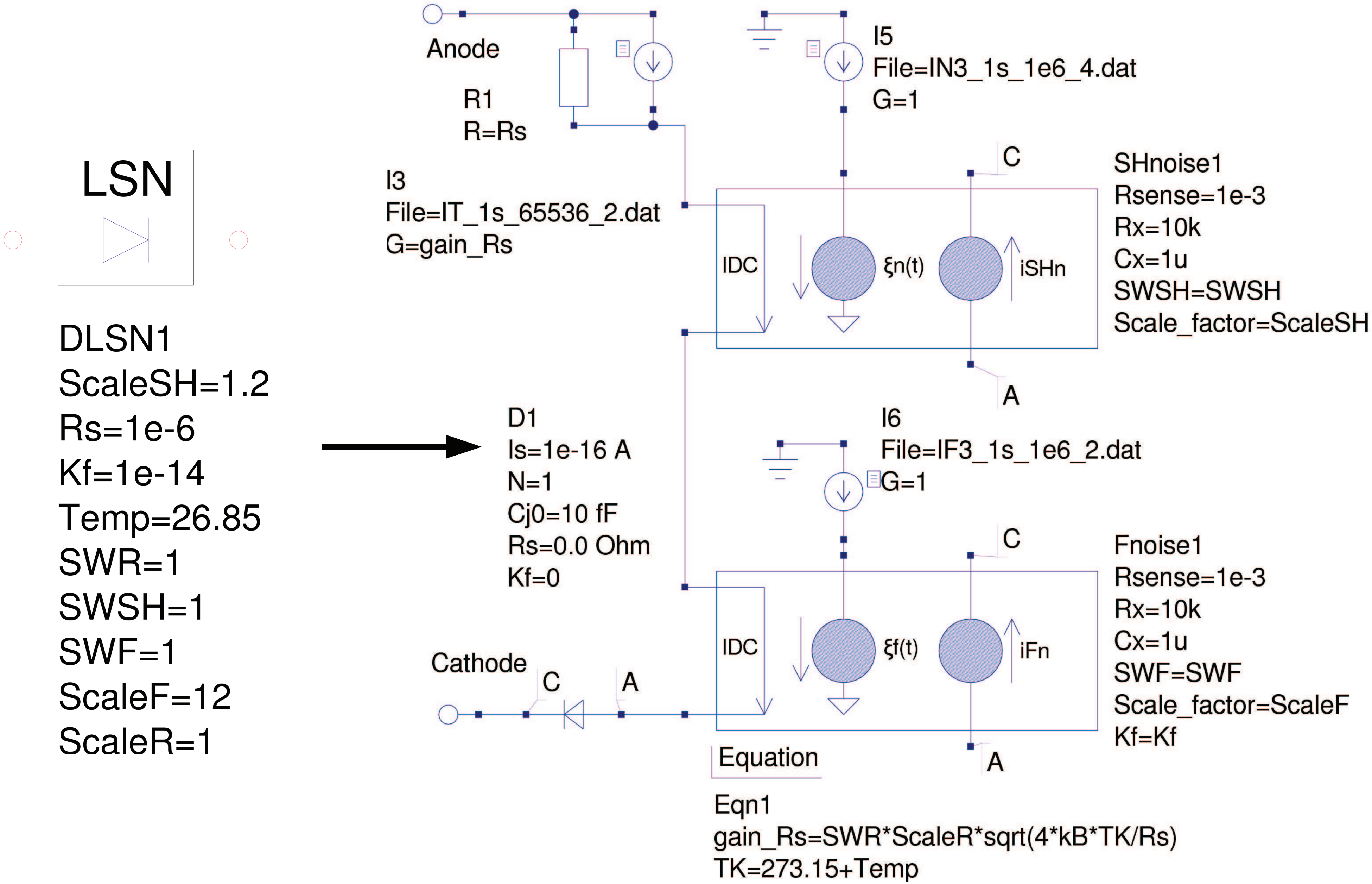


IrecSWITCH=0



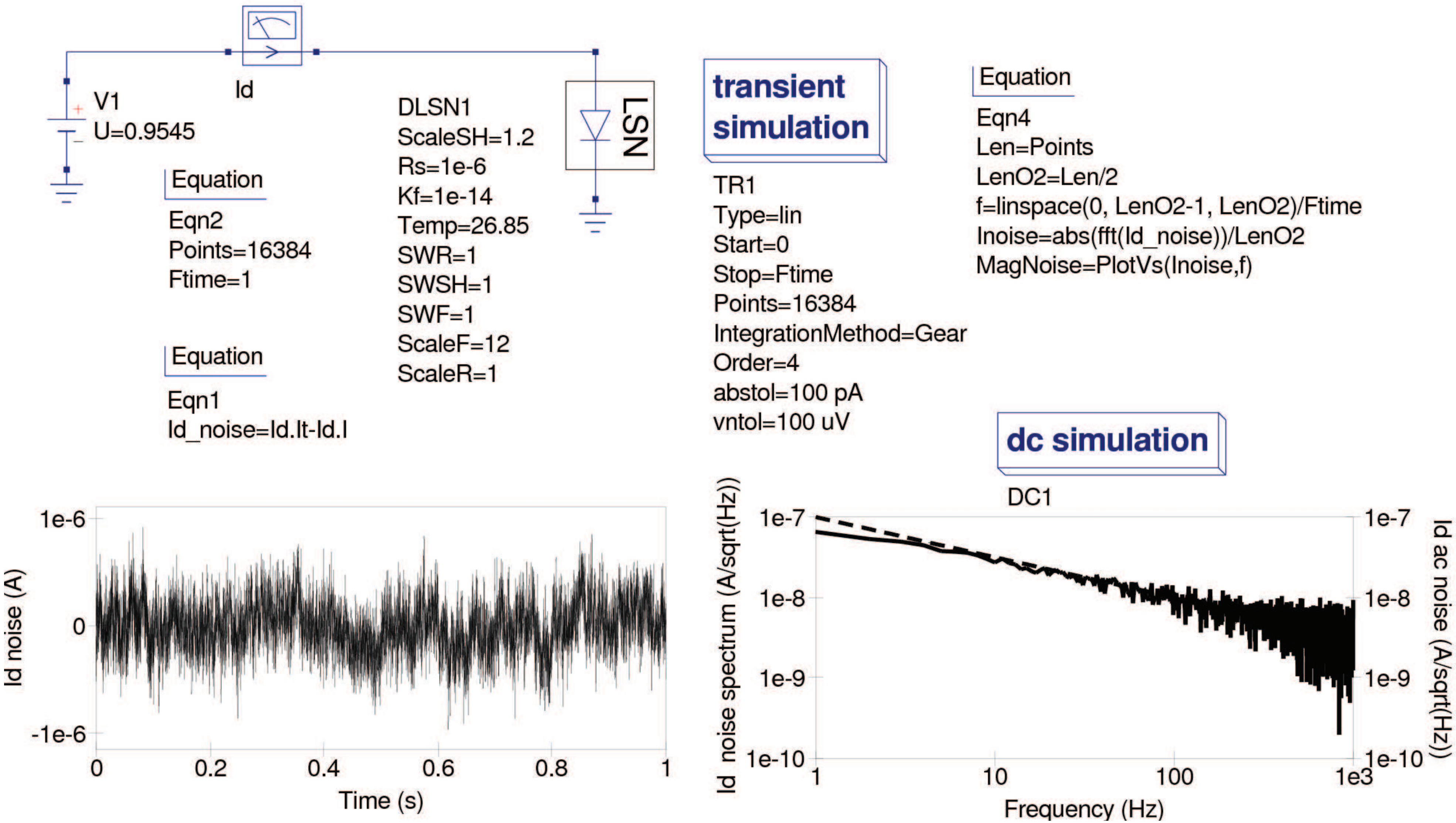
IrecSWITCH=1

Large signal time domain diode noise model*



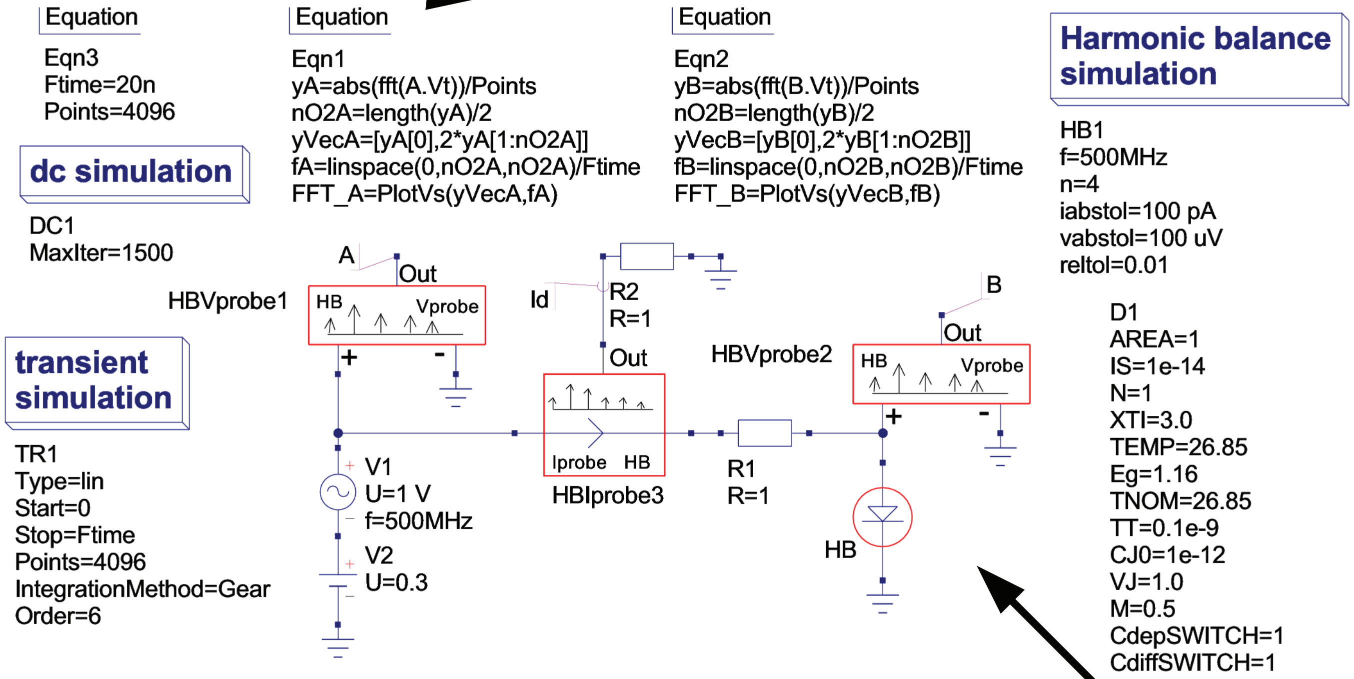
* Brinson M.E, Jahn S.and Nabijou H., “A tabular approach to modelling and simulating device and Circuit noise in the time domain, Int. J. Numer. Model., Published online 3 Feb. 2011, DOI: 10.1002/jnm.801.

Large signal time domain diode noise model continued

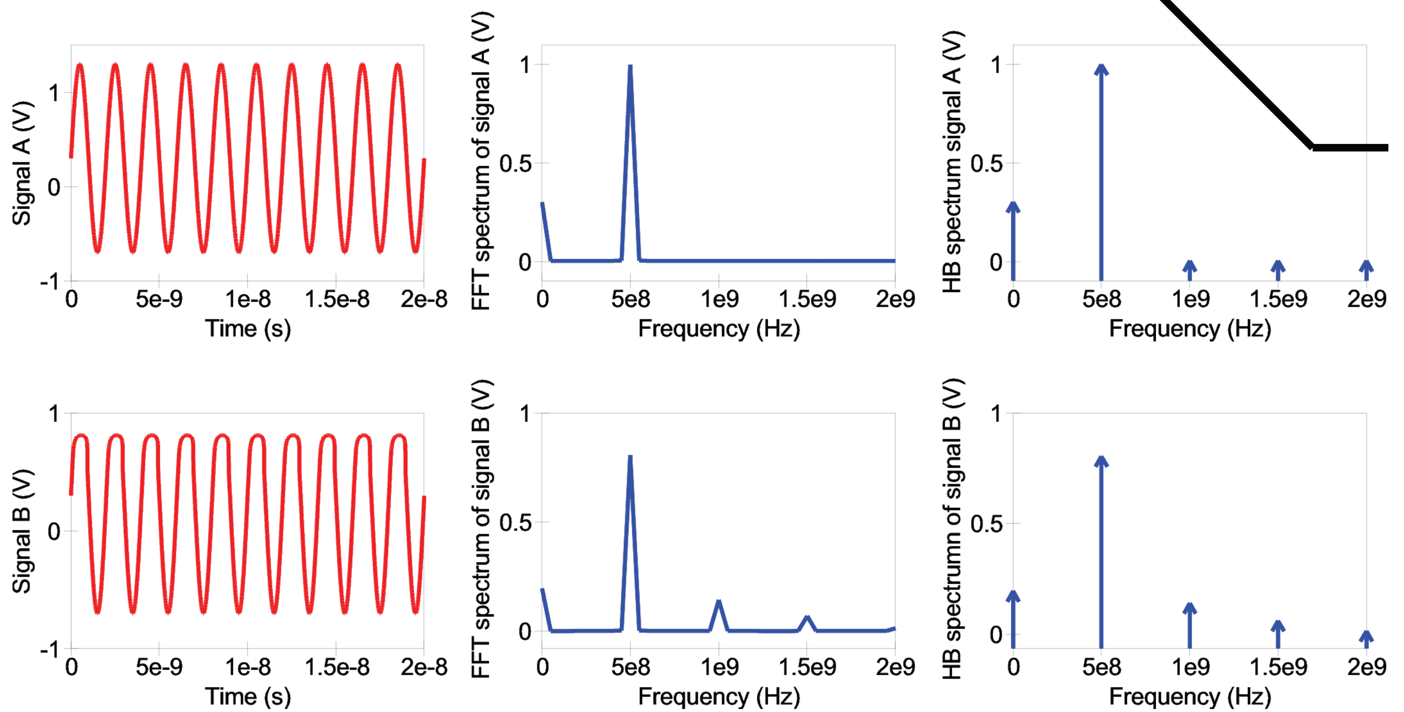


Harmonic balance (HB) simulation of extended diode model

Equation Blocks
Eqn1 and Eqn2
are FFT templates for
Qucs 0.0.15



Models marked HB indicate that they are suitable for harmonic balance simulation.



Verilog-A code for an extended diode model

```
`include "disciplines.vams"
`include "constants.vams"
//
module ModDiodeMOSAK (PA, CA);
inout PA, CA;
electrical PA, AN, CA;
 parameter integer BVSWITCH=0 from [0 : 1];        parameter integer CDEPSWITCH=0 from [0 : 1];
 parameter integer CDIFFSWITCH=0 from [0 : 1];    parameter integer ACSWITCH=0 from [0 : 1];
 parameter integer IRECSWITCH=0 from [0 : 1];
 parameter real AREA = 1.0 from [1.0 : inf];            parameter real N = 1.0 from [1.0 : inf];
 parameter real XTI = 3.0 from [1.0 : inf];             parameter real ISat = 1e-14 from [1e-30 : inf];
 parameter real TEMP = 26.85 from [-100.0 : inf];    parameter real TNOM = 26.85 from [-100 : inf];
 parameter real EG = 1.16 from [1 : inf];               parameter real BV = 100 from [0 : inf];
 parameter real IBV = 1e-3 from [1e-20 : inf];          parameter real CJ0 = 1e-12 from [1e-30 : inf];
 parameter real VJ = 1.0 from [0 : inf];                parameter real M = 0.5 from [1e-2 : inf];
 parameter real TT = 1e-9 from [1e-20  inf];            parameter real TAU = 100e-9 from [1e-20 : inf];
 parameter real AF = 1 from [0.1 : inf];                parameter real KF = 1e-16 from [1e-30 : inf];
 parameter real RS = 0.01 from [1.0e-6 : inf];
real CJ0T2,ISEFF,T1,T2,EGT1, EGT2, P7, P6, P11, PK, PQ, VTT2, VJT2, IST2, K2, K5, IBVEFF, IDBV, BVEFF;
real GMIN, CdepPARAM, CdiffPARAM, FCP, CM, C1, CF, C2, C3, C4, Id;
//
analog begin
// Equation initialization
@(initial_model)
begin
 CJ0T2=CJ0*AREA; ISEFF=ISat*AREA; T1=TNOM+273.15; T2=TEMP+273.15; EGT1=EG-7.02e-4*T1*T1/(1108+T1);
 EGT2=EG-7.02e-4*T2*T2/(1108+T2); P7=1-M;  P11=M/(2*VJ);PK=1.3806503e-23;PQ=1.602176472e-19;
 VTT2=PK*T2/PQ; VJT2=(T2*VJ/T1)-2*VTT2*(pow(ln(T2/T1), 1.5))-((T2*EGT1/T1)-EGT2);
 IST2=ISEFF*pow(T2/T1, XTI/N)*exp((-EGT1/VTT2)*(1-T2/T1)); K2=1/(N*VTT2); K5=N*VTT2;
 IBVEFF=IBV*AREA; P6=CJ0T2*VJT2/P7;
 IDBV=-IST2*(limexp(-BV*K2)-1.0);GMIN=1e-12;
 if (IBVEFF > IDBV) BVEFF=BV-K5*ln(IBVEFF/IDBV); else BVEFF=BV;
 if (IRECSWITCH==1) CdepPARAM=0;  else if (CDEPSWITCH==1)  CdepPARAM=1;  else CdepPARAM=0;
 if (IRECSWITCH==1) CdiffPARAM=0; else if (CDIFFSWITCH==1) CdiffPARAM=1; else CdiffPARAM=0;
 FCP=VJ; CF=TAU/RS; CM=CF-CJ0; C1=CF-CJ0/2*FCP; C2=(CJ0*FCP)/CM; C3=(CJ0*CJ0*FCP)/(2*CM); C4=CM*FCP/2;
end
```

Verilog-A code for an extended diode model

RS → (current contributions)

Diode I and dQ/dt terms →

Diode step recovery I →

Diode small signal I noise →

```
// Current contributions
I(PA,AN) <+ V(PA,AN)/RS;
Id = IST2*(limexp(V(AN,CA)*K2)-1.0)+GMIN*V(AN,CA);
I(AN,CA)    <+ Id;
if (CdepPARAM == 1)
     if (V(AN,CA) >= 0.0) I(AN,CA) <+ ddt(CJ0T2*(V(AN,CA)+P11*V(AN,CA)*V(AN,CA)));
     else I(AN,CA) <+ ddt(P6*(1-pow(1-V(AN,CA)/VJT2, P7)));
else I(AN,CA) <+ 0.0;
if ( BVSWITCH == 1)
      if (V(AN,CA) < -BV) I(AN,CA) <+ -IST2*(limexp(-(BVEFF+V(AN,CA))*K2-1+BVEFF*K2));
      else I(AN,CA) <+ 0.0;
else I(AN,CA) <+ 0.0;
if (CdiffPARAM == 1) I(AN,CA) <+ ddt(TT*Id); else I(AN,CA) <+ 0.0;
if ( BVSWITCH == 1)
     if (V(AN,CA) == -BV) I(CA,AN) <+ IBV; else I(AN,CA) <+ 0.0;
else I(AN,CA) <+ 0.0;
if (IRECSWITCH == 1)
     if (V(AN,CA) <=0.0) I(AN,CA) <+ ddt(CJ0*V(AN,CA)); else I(AN,CA) <+ 0.0;
else I(AN,CA) <+ 0.0;
if (IRECSWITCH == 1)
     if ((V(AN,CA) > 0.0) && (V(AN,CA) < FCP)) I(AN,CA) <+ ddt(C1*pow(V(AN,CA)+C2, 2)-C3);
     else I(AN,CA) <+ 0.0;
else I(AN,CA) <+ 0.0;
if ( IRECSWITCH == 1)
     if (V(AN,CA) >= FCP) I(AN,CA) <+ ddt(CF*V(AN,CA)-C4);
     else I(AN,CA) <+ 0.0;
else I(AN,CA) <+ 0.0;
if (ACSWITCH == 1) begin
   I(PA,AN) <+ white_noise((4*PK)/RS,"thermal");
   I(AN,CA) <+ white_noise(2*PQ*Id, "shot");
   I(AN,CA) <+ flicker_noise(KF*pow(Id,AF), 1, "flicker");
end
end
endmodule
```

Modelica code for a basic diode model

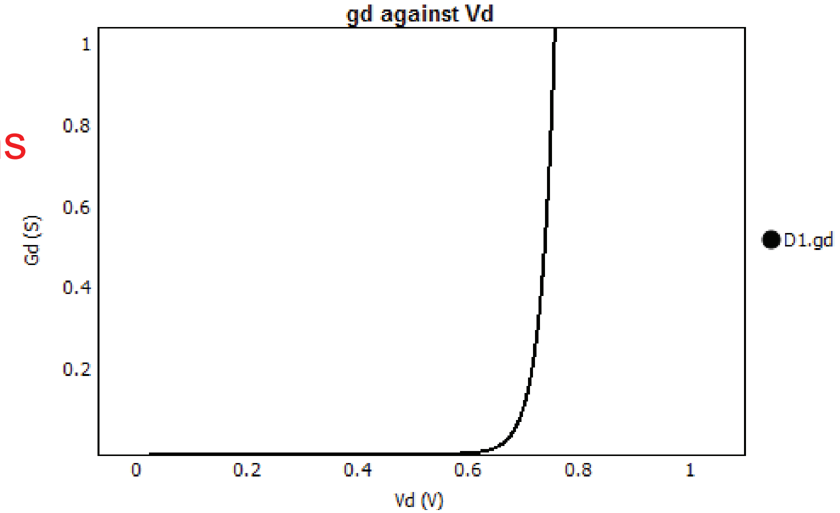
```
model ModDiode
Modelica.Electrical.Analog.Interfaces.NegativePin Cathode;  Modelica.Electrical.Analog.Interfaces.PositivePin Anode;
parameter Integer CdepSWITCH = 0;  parameter Integer CdiffSWITCH = 0; parameter Integer AREA = 1;
parameter Real N = 1.0; parameter Real XTI = 3.0; parameter Real ISat = 1e-14;
parameter Real TEMP = 26.85; parameter Real TNOM = 26.85; parameter Real EG = 1.16; parameter Real BV = 100;
parameter Real IBV = 0.001; parameter Real CJ0 = 1e-12; parameter Real VJ = 1.0; parameter Real M = 0.5;
parameter Real TT = 1e-12; parameter Real CJ0EFF = CJ0 * AREA; parameter Real T1 = TNOM + 273.15;
parameter Real T2 = TEMP + 273.15; constant Real PQ = 1.602176462e-19; constant Real PK = 1.3806503e-23;
constant Real A = 0.000702; constant Real B = 1108.0;
constant Real MAXEXP = 80.0 "Maximum argument in exp function before linearisation - same function as limexp in Verilog-A";
constant Real GMIN = 1e-12;
parameter Real CJ0EFF = CJ0 * AREA;
parameter Real CJ0EFFT2 = CJ0EFF * (1 + M * 0.0004 * (T2 – T1)); parameter Real T1 = TNOM + 273.15;
parameter Real T2 = TEMP + 273.15; parameter Real VtT2 = (PK * T2) / PQ;
parameter Real EGT1 = EG - (A * T1 * T1) / (B + T1); parameter Real EGT2 = EG - (A * T2 * T2) / (B + T2);
parameter Real IST2 = ISat * AREA * (T2 / T1) ^ (XTI / N) * exp((-EGT1 / VtT2) * (1 - T2 / T1));
parameter Real K2 = 1.0 / (N * VtT2);  parameter Real K3 = IST2 * K2;
Real v,i,gd,Cdep,Cdiff,CT,i1,i2,i3;
equation
 v = Anode.v – Cathode.v; 0 = Anode.i + Cathode.i;i = Anode.i;
i1 = if v / VtT2 > MAXEXP then IST2 * (exp(MAXEXP) * (1.0 + v / VtT2 - MAXEXP) - 1.0) + GMIN*v else IST2*(exp(K2 v) - 1.0) + GMIN*v;
 gd = if v / VtT2 < MAXEXP then K3 * exp(K2 * v) + GMIN else K3 * exp(MAXEXP) + GMIN;
Cdep = if CdepSWITCH == 1 then if v > 0.0 then CJ0EFF * (1 + (M * v) / VJ) else CJ0EFF * (1 - v / VJ) ^ (-M) else 1e-20;
 Cdiff = if CdiffSWITCH == 1 then if v >= 0.0 then TT * gd else 1e-20 else 1e-20;  CT = Cdep + Cdiff;
i2 = Cdep * der(v); i3 = Cdiff * der(v);  i = i1 + i2 + i3;
end ModDiode;
```

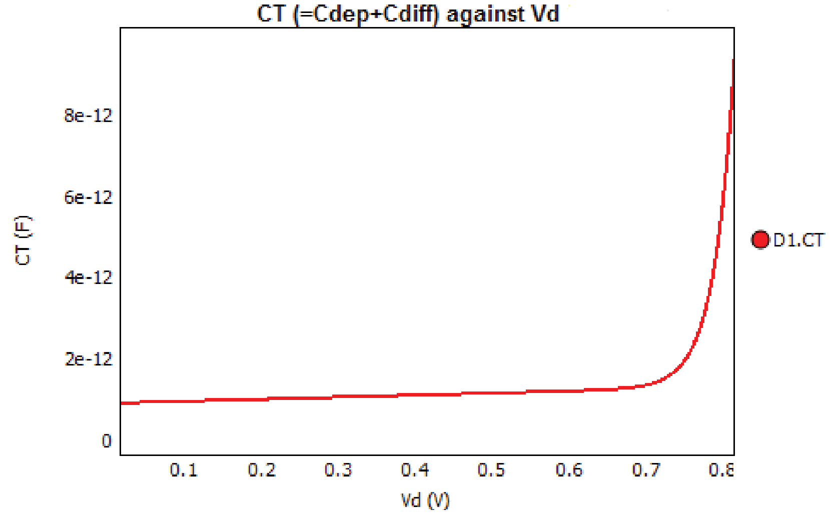
Device parameters

Model initialisation code

Equivalent to limexp

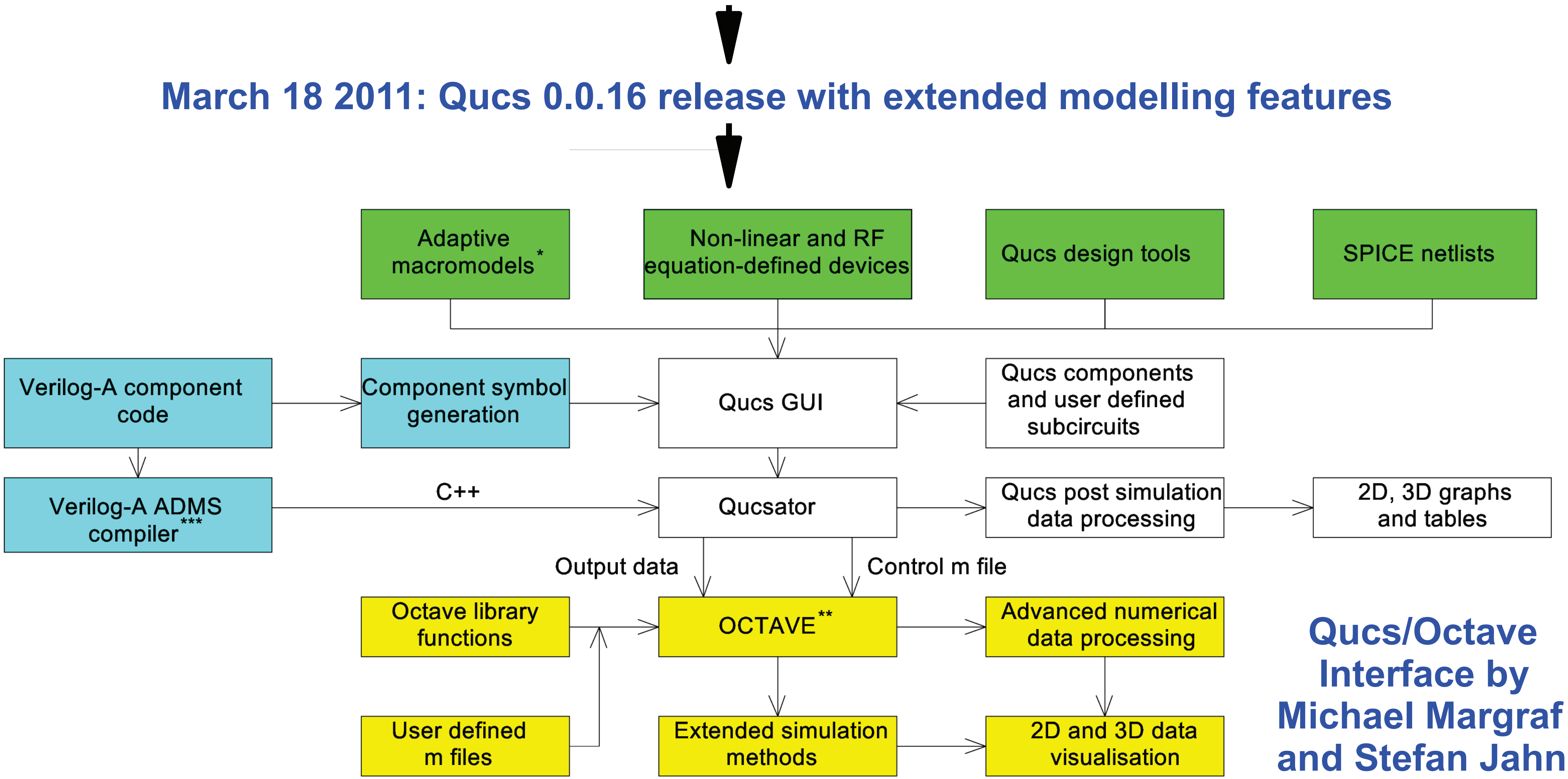
Model equations





Future directions for Qucs and compact equation-defined device modelling

March 18 2011: Qucs 0.0.16 release with extended modelling features



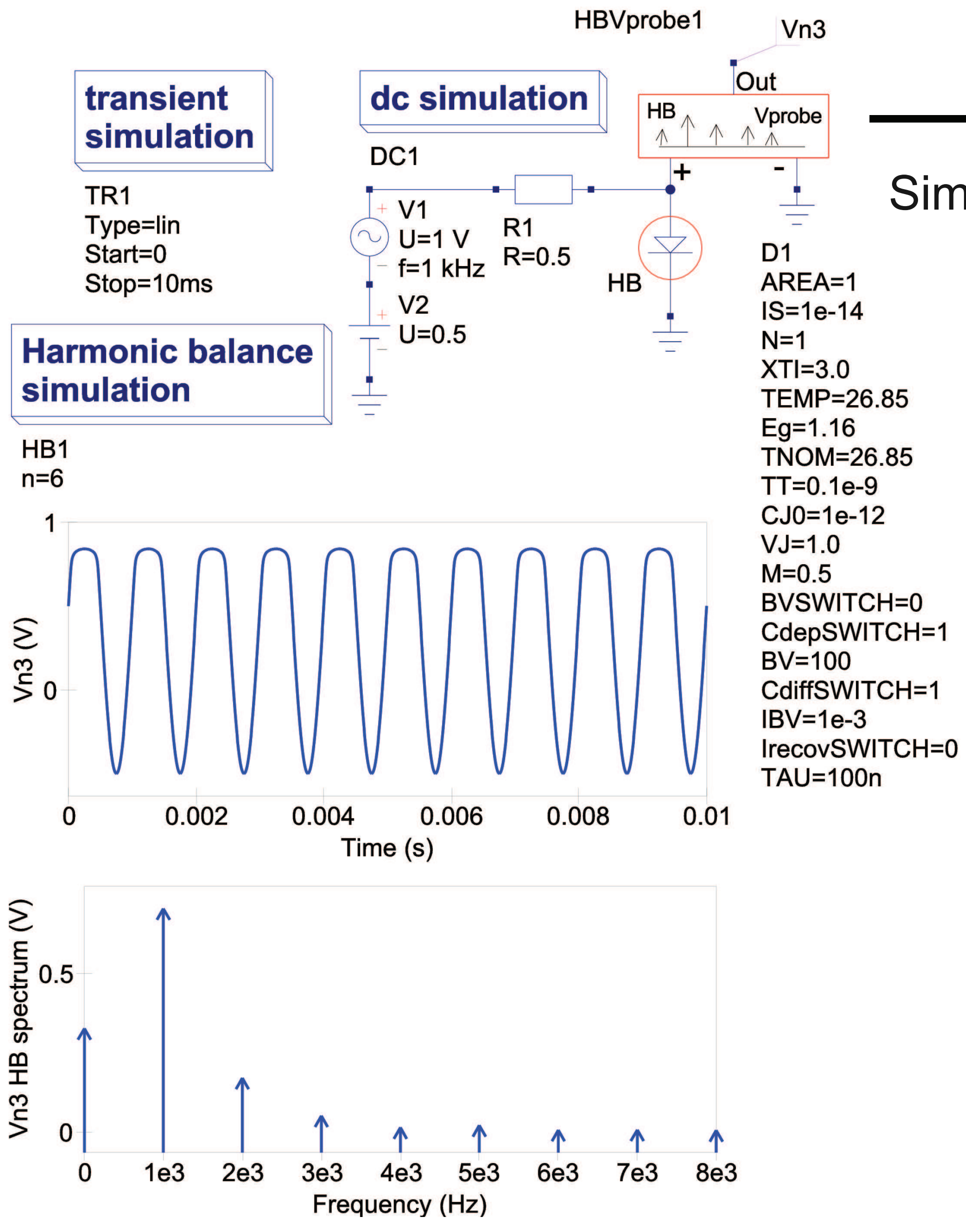
Qucs/Octave Interface by Michael Margraf and Stefan Jahn

* Brinson M.E. and Nabijou H., **Adaptive subcircuits and Verilog-A macromodels as integrated design and analysis blocks in Qucs circuit simulation**, International Journal of Electronics. In press 2011.

** Octave: A GPL numerical analysis package, www.gnu.org/software/octave

*** ADMS:Verilog-A compiler. http://mot-adms.sourceforge.net/

Final example



Simulation data

```
% MOS-AK Paris 2011- Slide 20
[data] = loadQucsDataSet("Fexample.dat")
showQucsDataSet(data)
[PLVn3] = getQucsVariable(data,"Vn3.Vb");
[Hbfreq]  = getQucsVariable(data,"hbfrequency");
clf();
stem(Hbfreq,abs(PLVn3), "-8")
title("HB amplitude spectrum against frequency");
xlabel("Frequency (Hz)");
ylabel("HB amplitude spectrum (V)");
print('RCx2.ps', '-dpsc' );
```

Octave