Qucs-S a maturing GPL software package for circuit simulation and compact modelling of current and emerging technology devices

Mike Brinson [1], mbrin72043@yahoo.co.uk.
Vadim Kuznetsov [2], ra3xdh@gmail.com

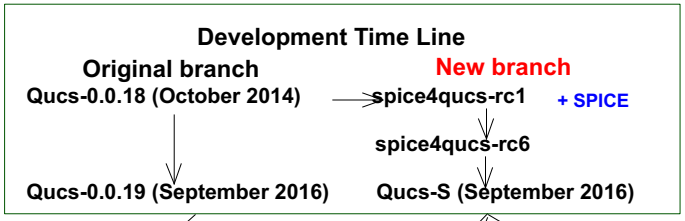
[1]Centre for Communications Technology, London Metropolitan University, UK
[2]Bauman Moscow Technical University, Russia

Presented at the 14th MOS-AK Workshop, ESSDERC/ESSCIRC,
Lausanne, September 12, 2016

Qucs-S a maturing GPL software package: Introduction to presentation

- Background and release dates for Qucs-S/RC7
- Qucs-S documentation and circuit simulation capabilities
  - Qus-S Help documentation: User manual and reference material,
  - Qucs-S SPICE style components and model libraries,
  - Qucs-S extended circuit simulation features.
- Qucs-S Equation-Defined Device (EDD) and Verilog-A compact device modelling case studies
  - The Efficient Power Corporation (EPC) GaN EPC2001 power transistor model,
  - The MIT virtual source GaN-RF HEMT 1.0.0 model.
- Qucs-S modelling tool extensions and new features
  - XSPICE "Code Model" support,
  - Qucs-S model parameter extraction controlled by Qctave script files.
- Qucs-S future developments - the way forward in 2016-2017

Qucs-S a maturing GPL software package: Background and release details



Qucs-0.0.19 is the next release in the Qucs series of software packages. This release is primarily GUI and quasator improvements plus general bug fixes.

Snapshot at
https://sourceforge.net/projects/qucs/files/qucs-binary/0.0.19-snapshots/qucs-0.0.19-rc1-win64.exe/download

New test suite for Qucs-S introduced.
A set netlist and simulation tests available.
Sources are here:
https://github.com/ra3xdh/qucs-test/tree/spice4qucs

Extended Qucs circuit simulator package, including

1. Improved Qucs GUI,
2. Access to qucsator, Ngspice, Xyce and SPICE OPUS
3. Increased number of circuit simulation types, including pole-zero analysis, distortion analysis, Harmonic Balance, steady state shooting method and Monte-Carlo statistical analysis,
4. Full range of SPICE 3f5, Ngspice, Xyce and SPICE OPUS device models,
5. SPICE style component libraries,
6. Equation-defined device modelling,
7. Verilog-A model synthesis and "Turn Key" C++ code compiler,
8. XSPICE Code model "Turn Key" compiler,
9 XSPICE analogue component library,
10. Nutmeg and Xyce script device modelling,
11. Qucs and Octave post-simulation data processing,
12. Qucs-S device parameter extraction with linked Octave packages.

Qucs-S can be found at:

https://github.com/ra3xdh/qucs/releases/tag/0.0.19S-rc7

New "Qucs-S Help" Documentation:

This is a "live" document which is updated by Qucs-S Developes as the package is improved.

Users can find it at -
https://qucs-help.readthedocs.io/en/spice4qucs/index.html

Qucs-S a maturing GPL software package: Latest release impact

uApp Explorer is the unofficial viewer for snaps and Ubuntu Touch apps.

qucs-spice

DOWNLOAD SHARE

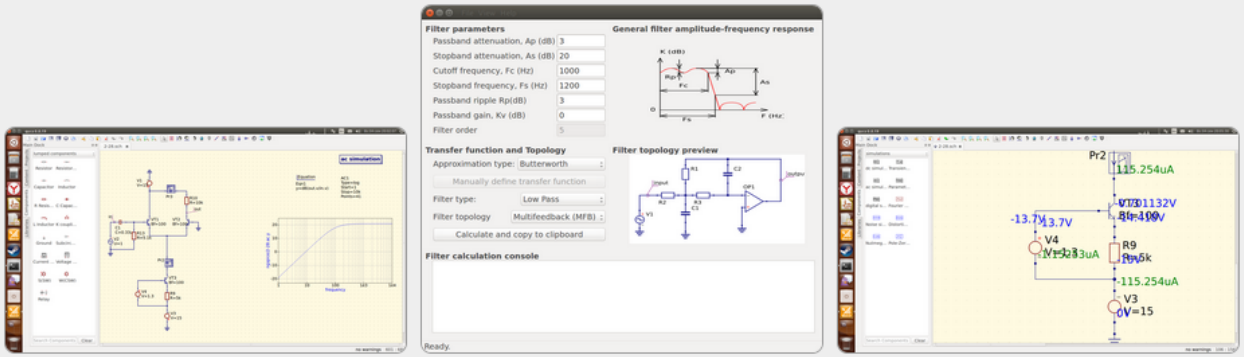
Install using snapd

0

Eldar Khayrullin

Snap Free

Screenshots



Description

Qucs is an integrated circuit simulator which means you are able to setup a circuit with a graphical user interface (GUI) and simulate the large-signal, small-signal and noise behaviour of the circuit. After that simulation has finished you can view the simulation results on a presentation page or window. Supports spice simulators.

Changelog

After install run in terminal:
snap run qucs-spice.qucs

Permissions

- Home
- Unity7

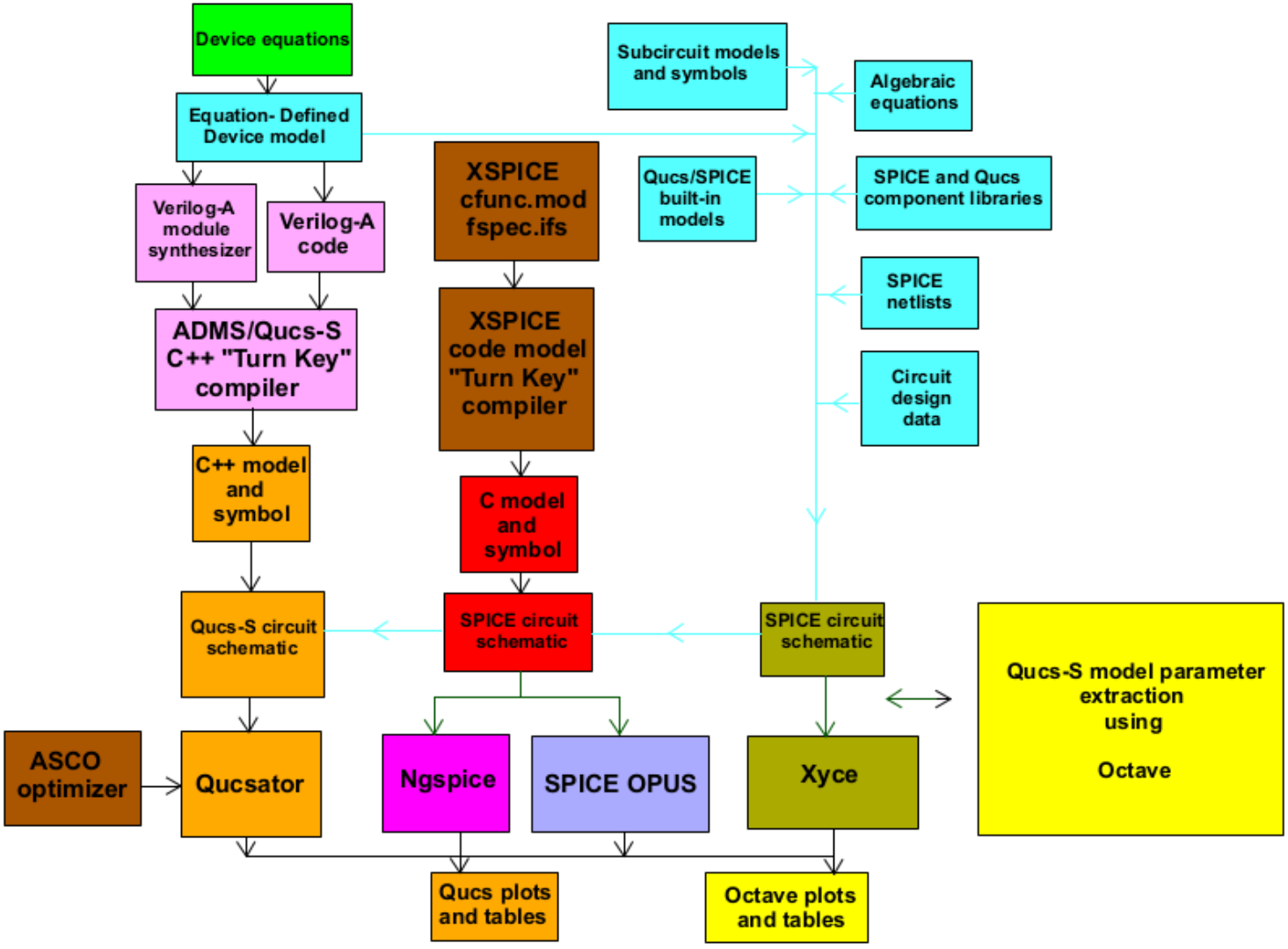
Info

Support: https://github.com/eldarkg/qucs-spice-snap/issues
Website: http://qucs.sourceforge.net

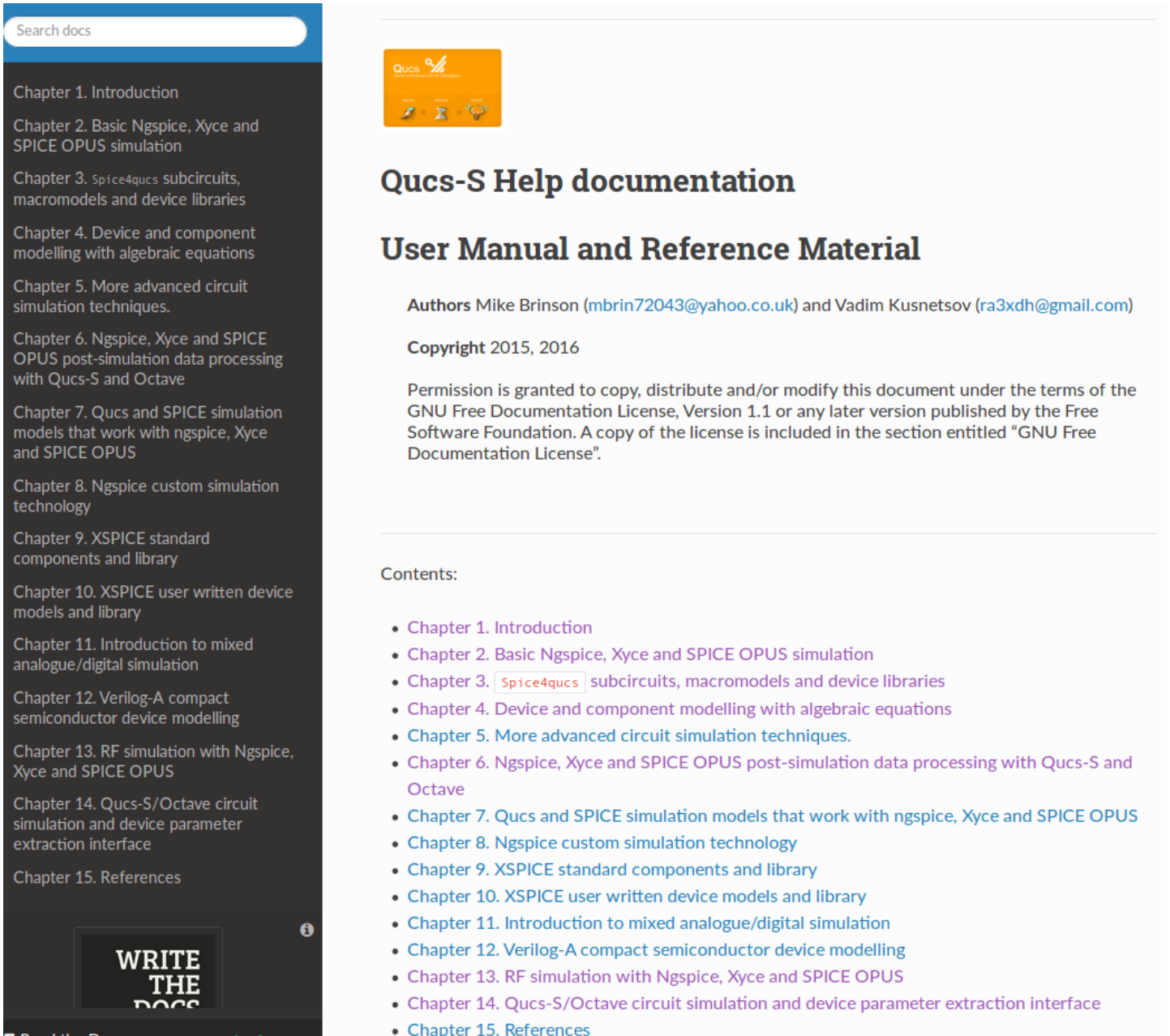
Version: 0.0.19S-RC7-snap2
Updated: Sep 5, 2016
Published: Sep 4, 2016
License: GNU GPL v2
File Size: 89.2 MB
Architectures: i386, amd64

MORE INFO

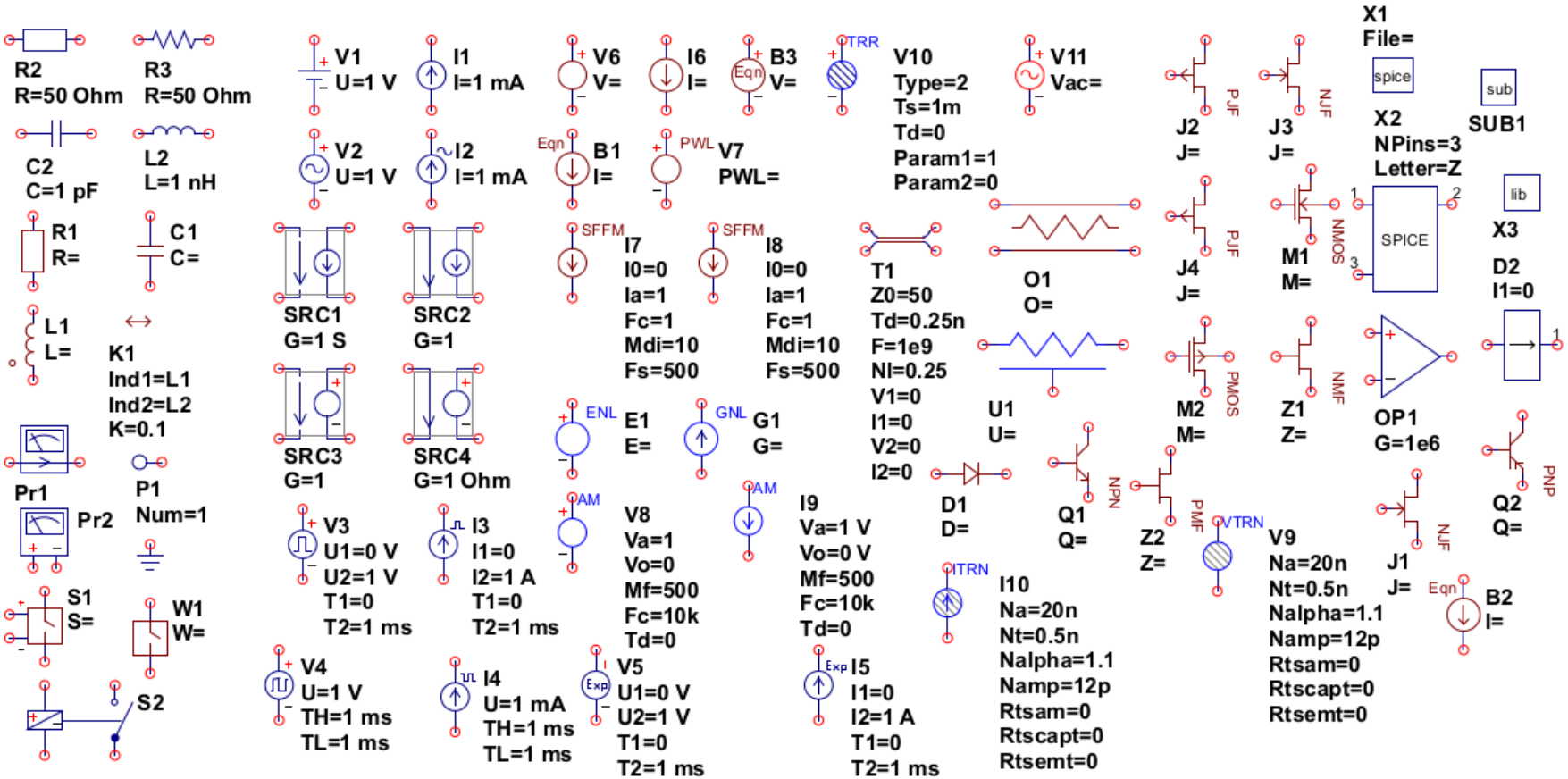
Qucs-S a maturing GPL software package: Functional block diagram



Qucs-S a maturing GPL software package: Qucs-S Help documentation



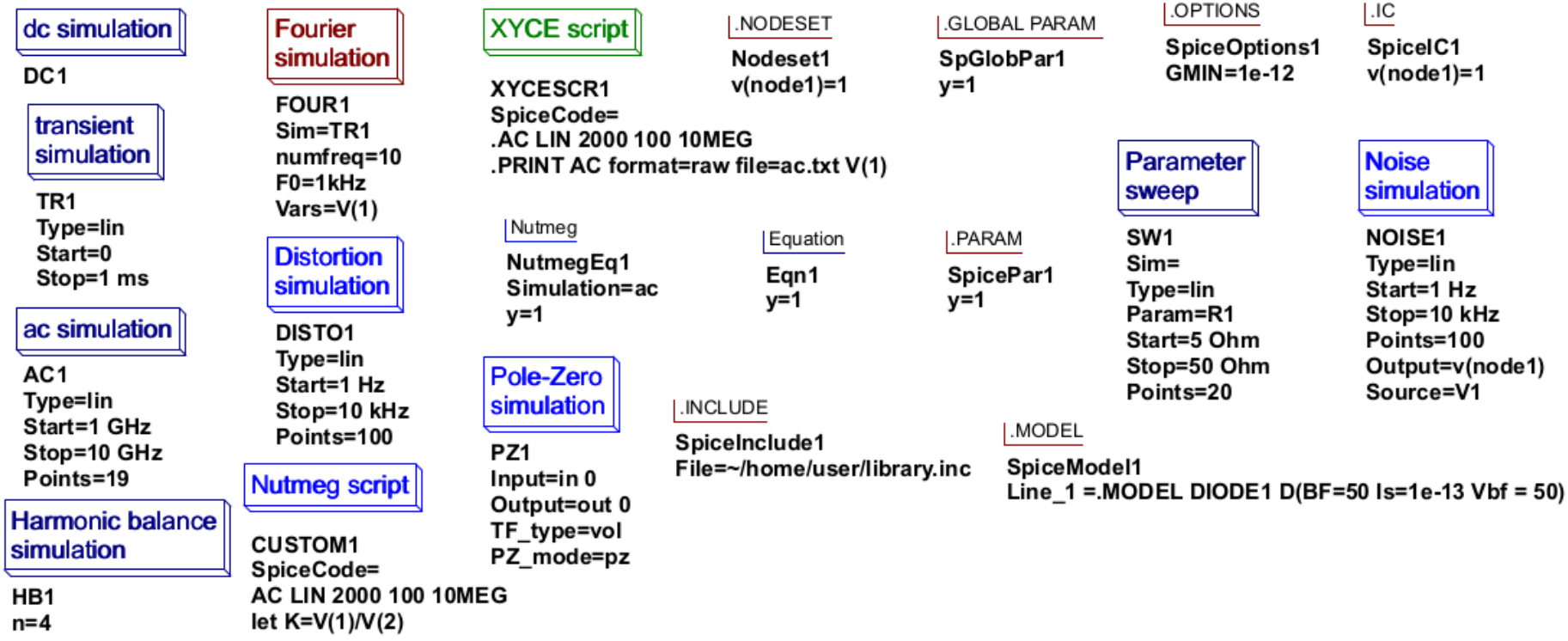
Qucs-S a maturing GPL software package: Ngspice, Xyce and SPICEOPUS built in components



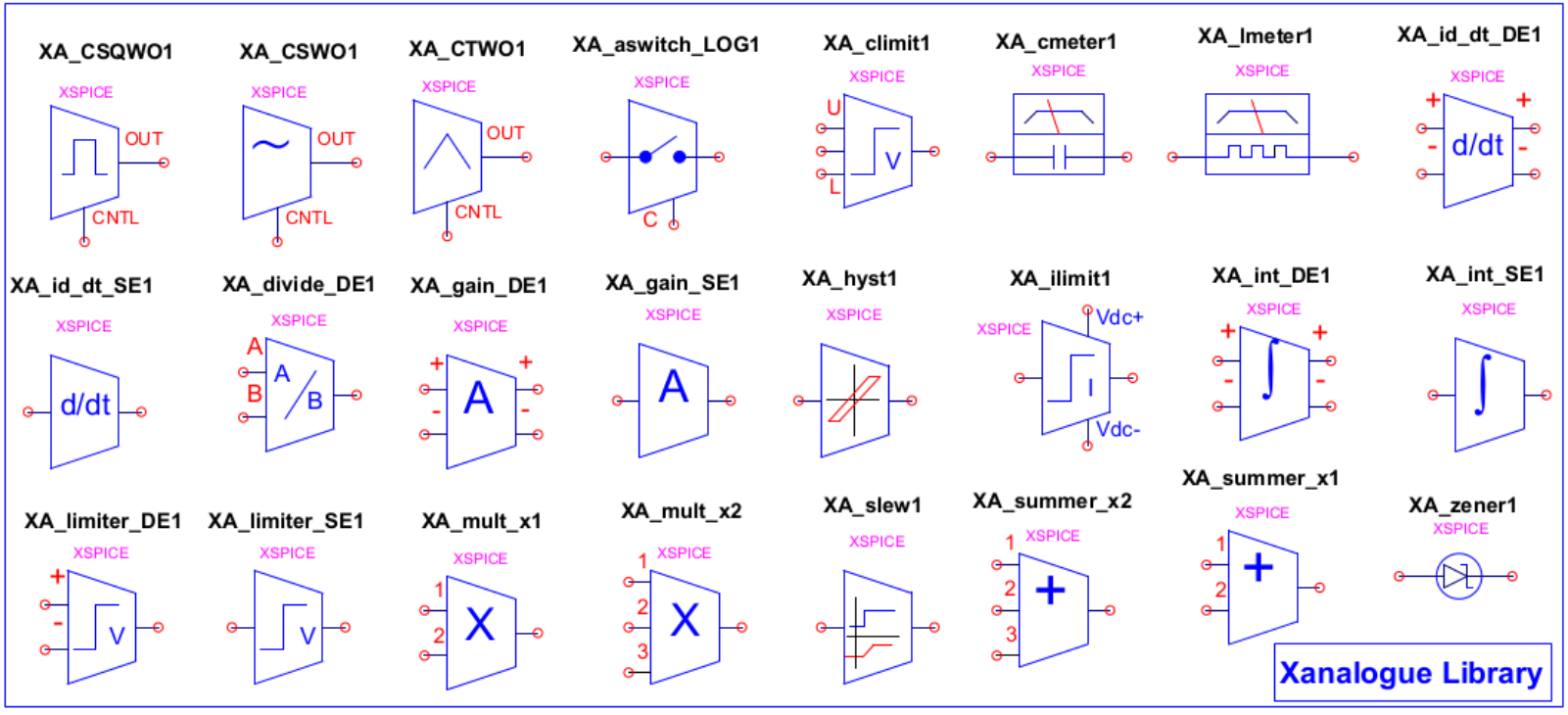
Qucs-S a maturing GPL software package: Available semiconductor device models

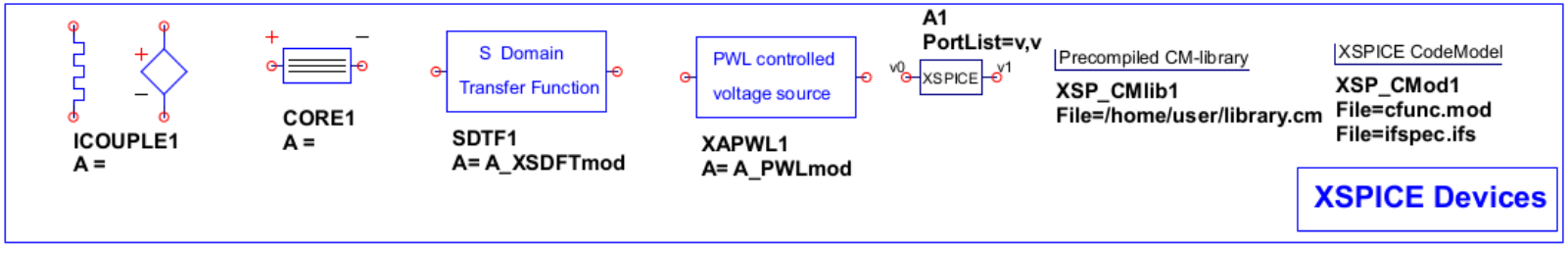
Type	Description	Level	Ngspice	Xyce	SPICEOPUS
D	Legacy	1	X	X	X
		2		X	
		3			X
BJT	Legacy	1	X	X	X
	VBIC	10,11,12		X	
	FBH HBI_X	23		X	
	MEXTRAM	504,505			X
JFET		1	X	X	X
		2	X	X	X
MESFET		1 (Statz)	X	X	X
		2 (Ytterdel)	X		
MOSFET	Legacy	1	X	X	X
		2	X		X
		3	X	X	X
		4(BSIM1)	X		X
		5(BSIM2)	X		X
		6	X	X	X
	BSIM3v2	47			X
		8	X		
		9		X	
	BSIM3v3	53			X
		49	X		
	BSIM4	60			X
		14	X	X	
		54	X		
	BSIM3SOIv1	55			X
	BSIMOIv2	56	X		X
		58,55,57	X		
		10	X	X	
	STAGSOI3	57			X
	UFSOI	58			X
	SOI3	60	X		
	UFET	7			X
	EKVv2p6	44	X		X
	HISIM2	61,68	X		
	HISIM_HV	62,63	X		
	VDMOS	18		X	
	BSIM 6p1	77		X	
	PSP 103p1	103		X	

Qucs-S a maturing GPL software package: Simulation control icons

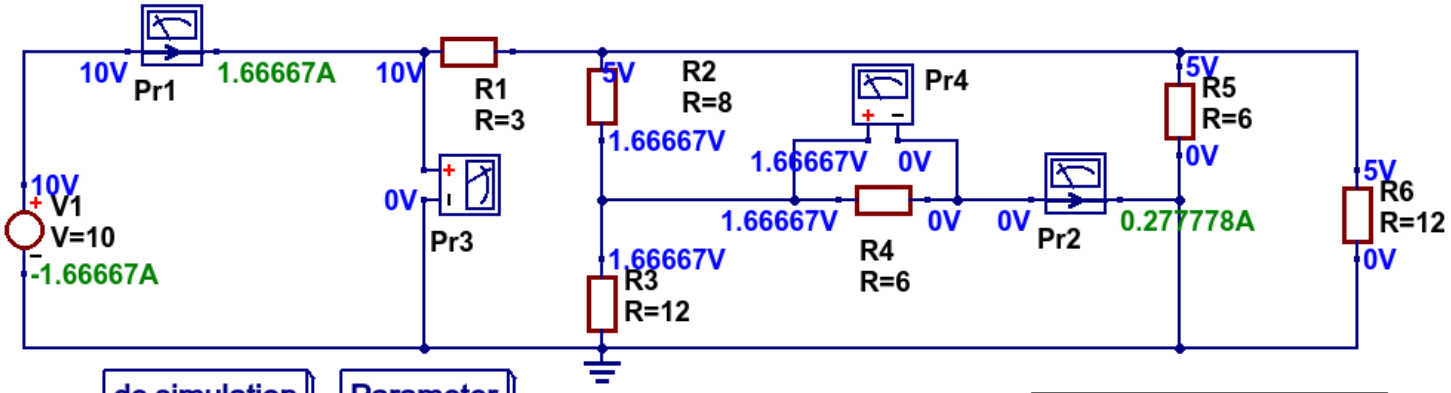


Qucs-S a maturing GPL software package: XSPICE analogue component models





Qucs-S a maturing GPL software package: Qucs-S extended circuit simulation Part 1. SPICE .OP to visual DC by pressing key "F8"



dc simulation

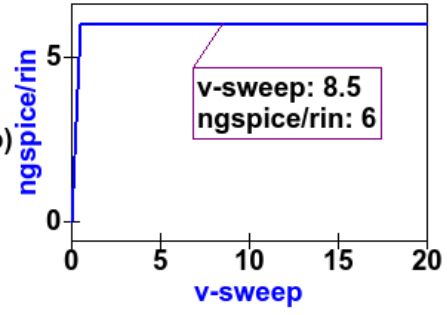
DC1

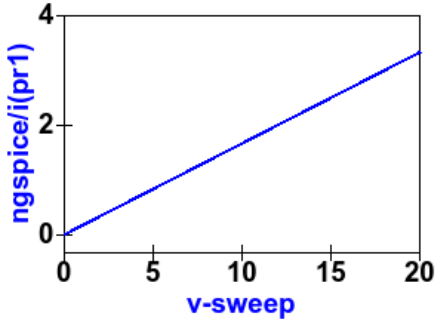
Parameter sweep

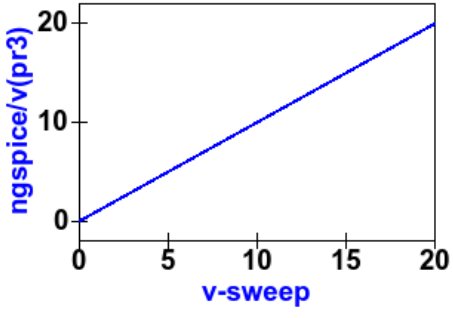
SW1
Sim=DC1
Type=lin
Param=V1
Start=0
Stop=20
Points=40

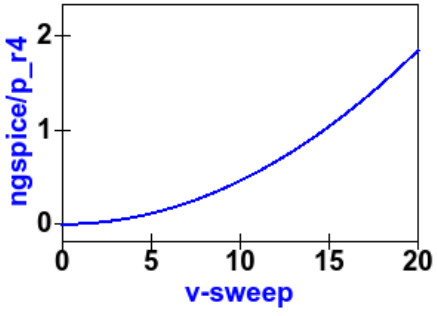
Nutmeg

NutmegEq1
Simulation=dc
Rin=V(pr3) / (vpr1#branch +1p)
P_R4=V(pr4) * vpr2#branch

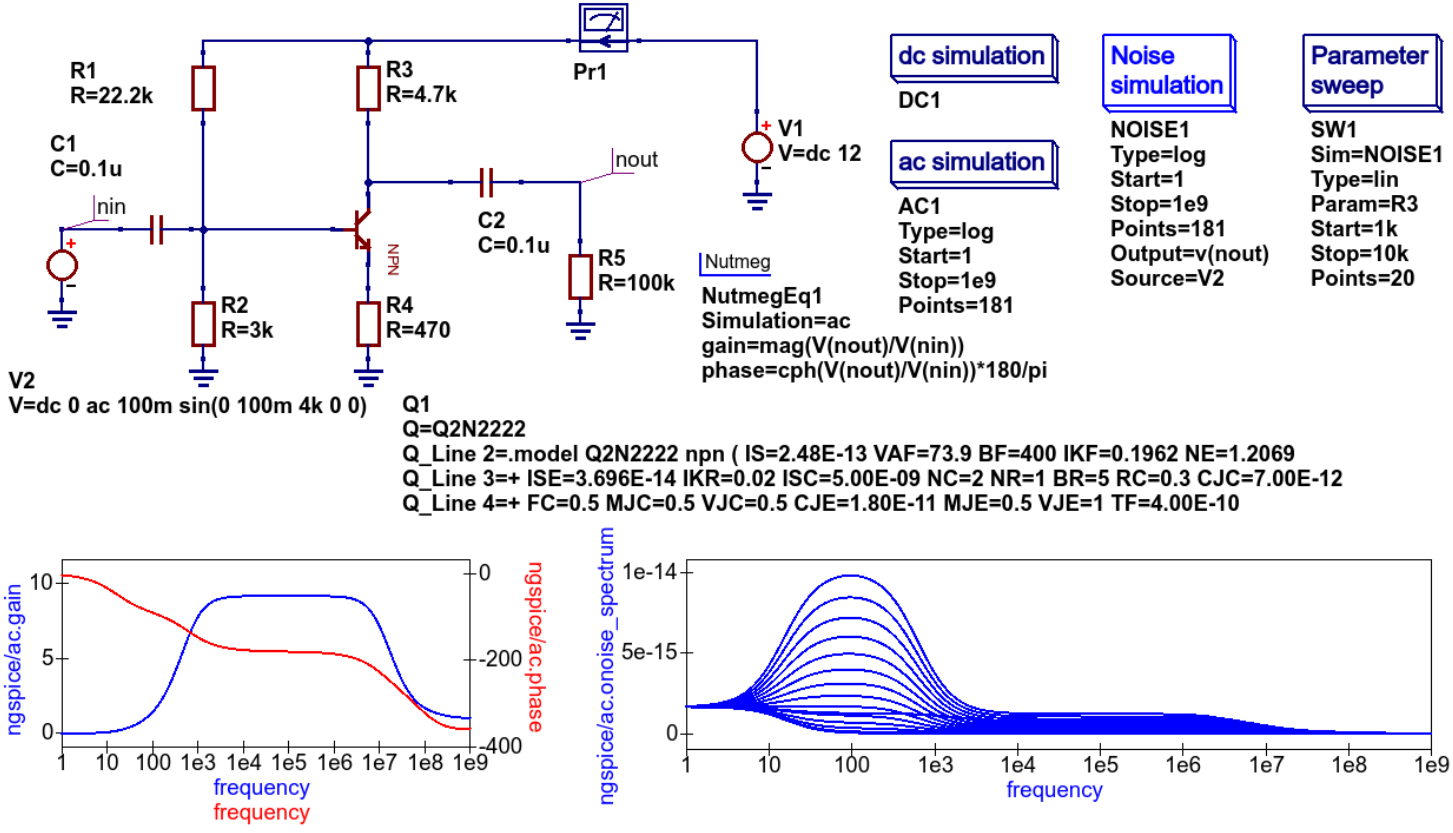




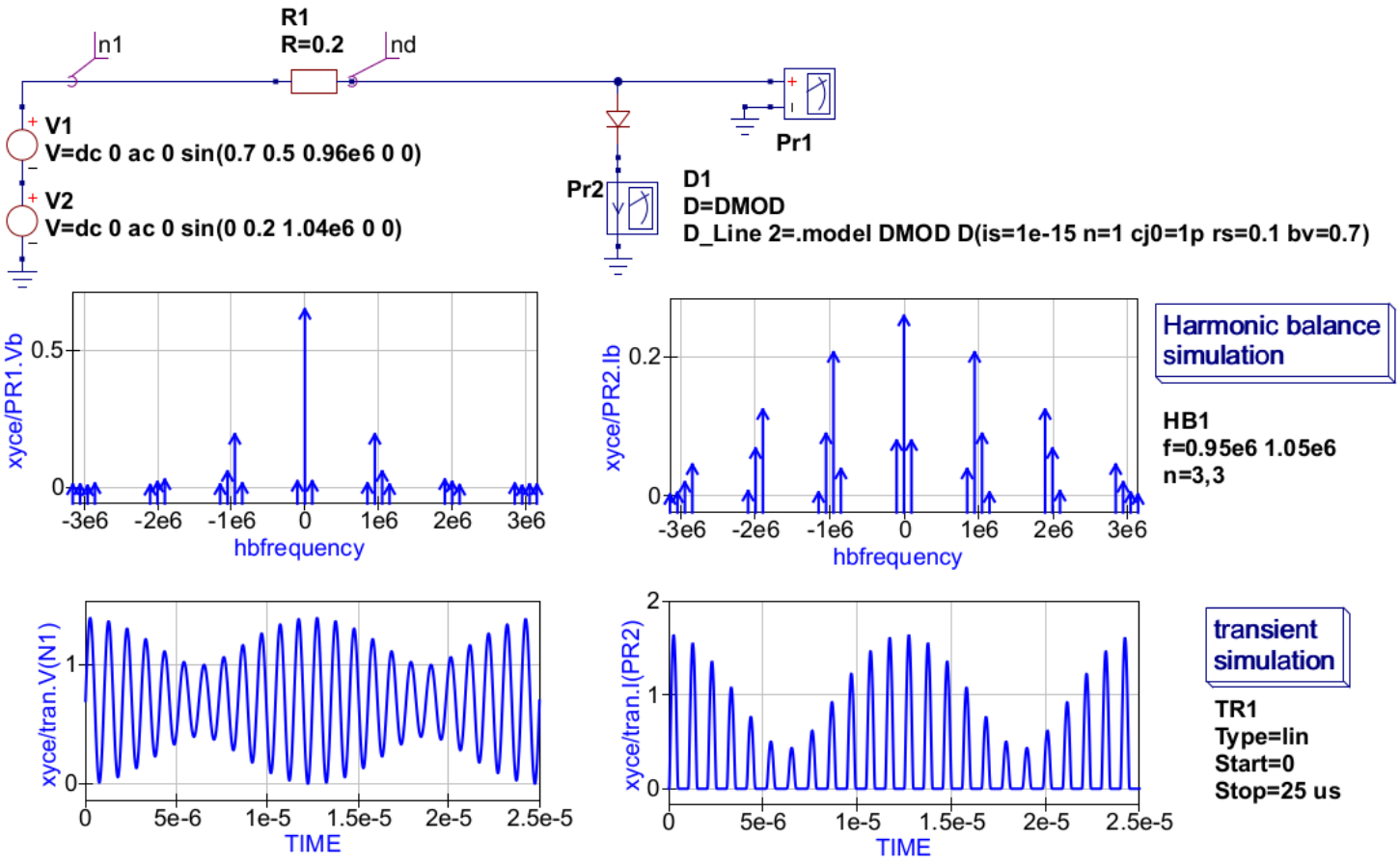




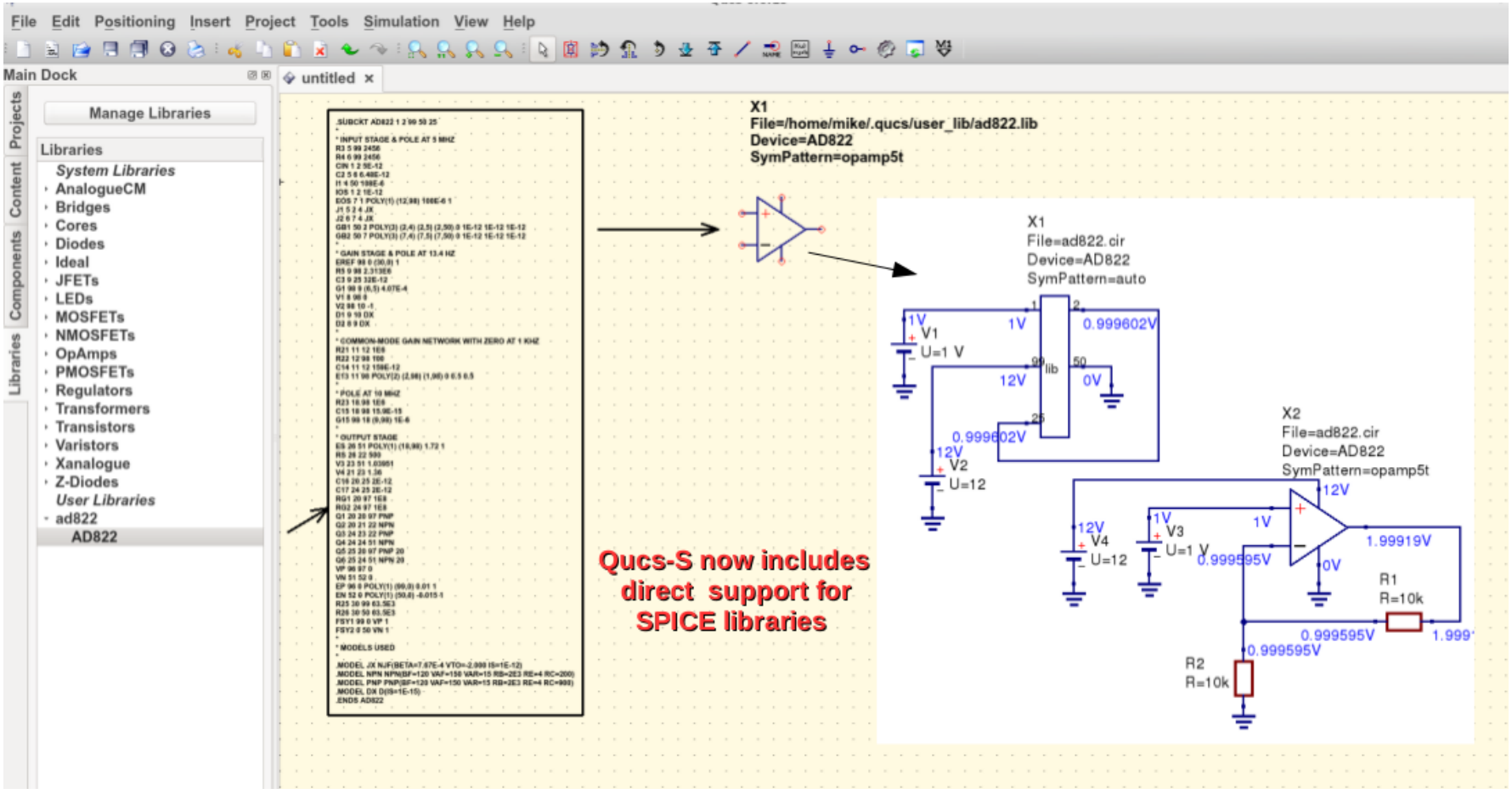
Qucs-S a maturing GPL software package: Qucs-S extended circuit simulation Part 2. Noise spectral response



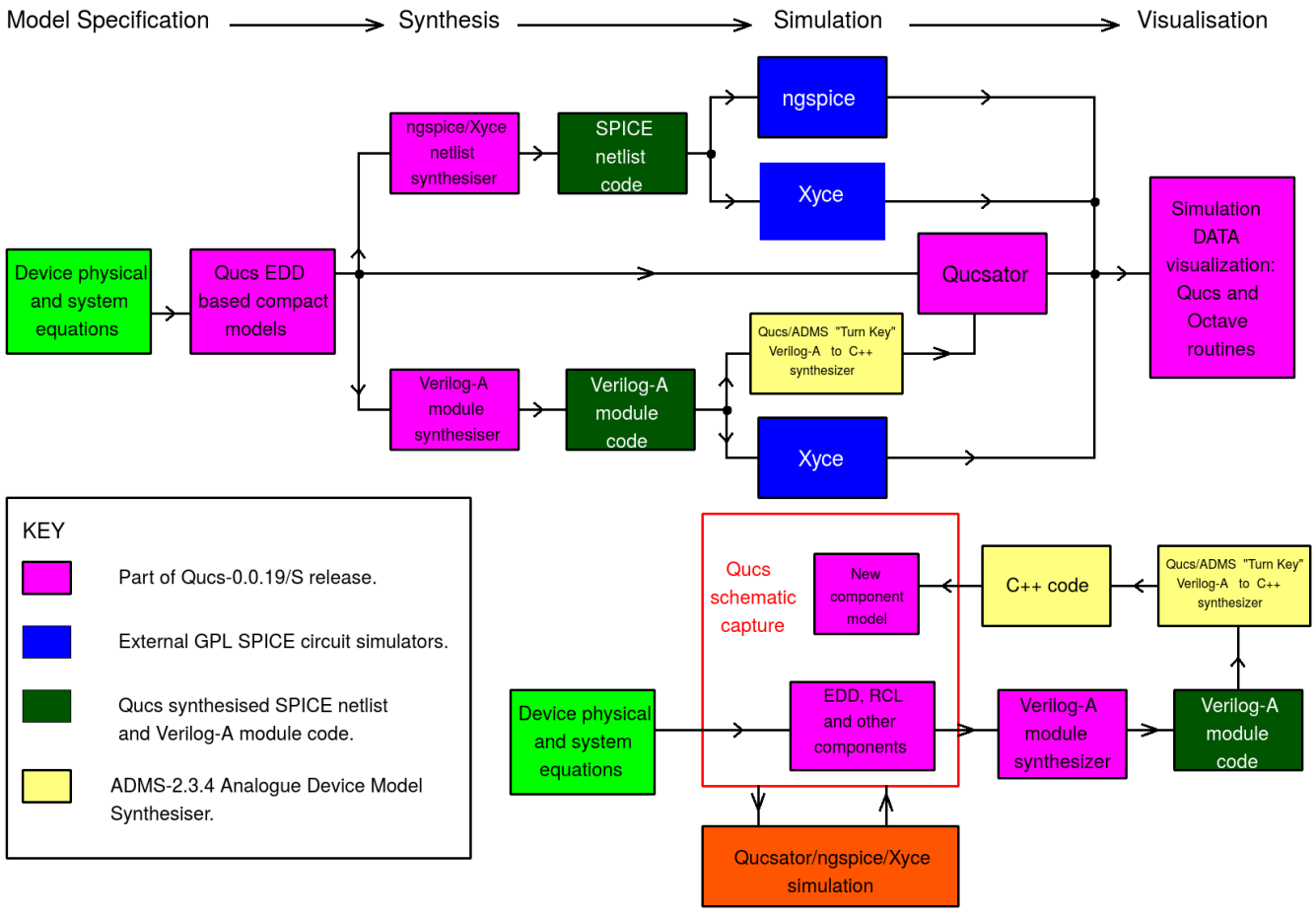
Qucs-S a maturing GPL software package: Qucs-S extended circuit simulation Part 3. Multi-tone Harmonic balance analysis



Qucs-S a maturing GPL software package: Qucs-S extended circuit simulation Part 4. Direct support for SPICE libraries

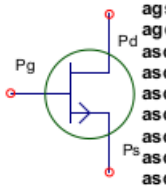


Qucs-S a maturing GPL software package: Compact device modelling tools Part 1. Equation-Defined behvioural modelling and Verilog-A model code synthesis

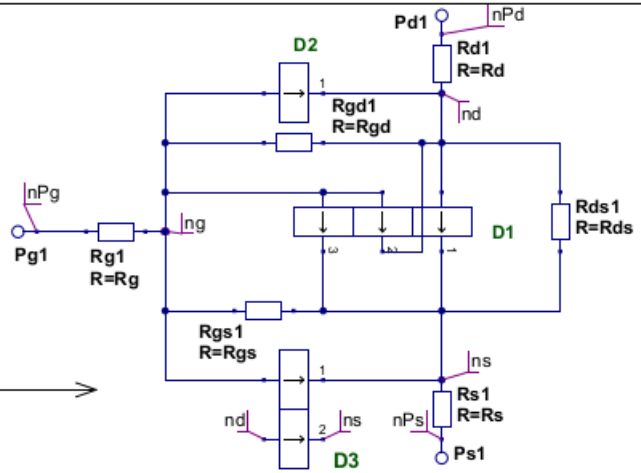


Qucs-S a maturing GPL software package: Compact device modelling Part 2. Qucs-S EDD model of the Efficient Power Corporation (EPC) GaN EPC2001 power transistor

Qucs-S EDD subcircuit model of a EPC2001 enhancement GaN power transistor -See Application Note: An005 Circuit Simulation using EPC Device Models, Efficient Power Conversion Corporation,Copyright 2011, www.epc-co.com.



GaN_EPC1
Temp=25
A1=41.7998
alTc=5.486028e-3
k2=2.259866
k3=1.2e-1
aWg=1077
rpara=4.463059e-3
x0_0=0.75
x0_1=1.10
ax0Tc=0.75e-4
dgs1=4.4e-7
dgs2=2.6e-13
dgs3=0.8
dgs4=0.23
ags1=8.6952e-10
agd1=1.4182e-11
asd1=3.3621e-10
asd2=6.308e-10
asd3=1.2803e1
asd4=2.269
asd5=2.518e-10
asd6=-4.0599e1
asd7=2.0638e1
agd2=2.1475e-10
agd3=-3.803
agd4=5.9551
ags2=5.3168e-10
ags3=1.997
ags4=2.8377e-1
ags5=-1.4751e-10
ags6=-7.5163
ags7=7.121



Equation

Eqn1
EQ1=A1*(1-alTc*(Temp-25))
Rgs=1e11/aWg
Rgd=1e11/aWg
Rds=1e11/aWg
Rd=0.75*rpara
Rs=0.25*rpara
Rg=0.6*1077/aWg
EQ2=1.0+ax0Tc*(Temp-25)*(Temp-25)
EQ3=0.5*aWg

D1
I1=(V1 > 0) ? EQ1*ln(1.0+exp((V3-k2)/k3))*V1/(1.0+max((x0_0+x0_1*V3/EQ2), 0.5)*V1) : -EQ1*ln(1.0+exp((V2-k2)/k3))*(-V1)/(1.0+max((x0_0+x0_1*V2/EQ2), 0.5)*(-V1))
Q1=asd1*V1 + (asd2*asd4*ln(1+exp((V1-asd3) / asd4)) + asd5*asd7*ln(1 + exp((V1-asd6)/asd7)))
I2=0
I3=0

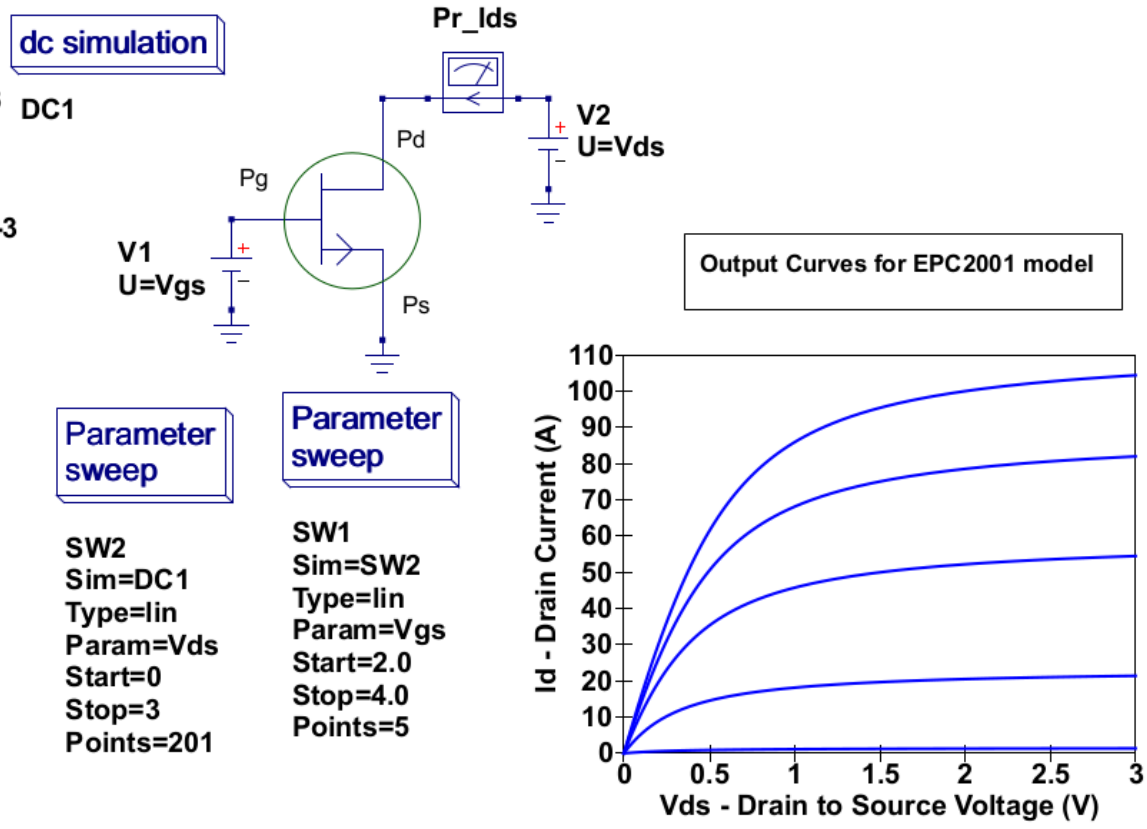
D2
I1=(V1 > 10) ? (EQ3/1077*(dgs1*(exp(10.0/dgs3)-1)+dgs2*(exp(10/dgs4)-1))) : (EQ3/1077*(dgs1*(exp(V1/dgs3)-1)+dgs2*(exp(V1/dgs4)-1)))
Q1=agd1*V1 + (0.5*ags2*ags4*ln(1+exp((V1-ags3) / ags4)) + agd2*agd4*ln(1 + exp((V1-agd3)/agd4)))

D3
I1=(V1 > 10) ? (EQ3/1077*(dgs1*(exp(10.0/dgs3)-1)+dgs2*(exp(10/dgs4)-1))) : (EQ3/1077*(dgs1*(exp(V1/dgs3)-1)+dgs2*(exp(V1/dgs4)-1)))
Q1=ags1*V1 + (0.5*ags2*ags4*ln(1+exp((V1-ags3) / ags4)) + ags5*ags7*ln(1 + exp((V2-ags6)/ags7)))

Ported from the EPC mathematical model common to LTSPICE, PSPICE, TSPICE and Spectra netlist models.

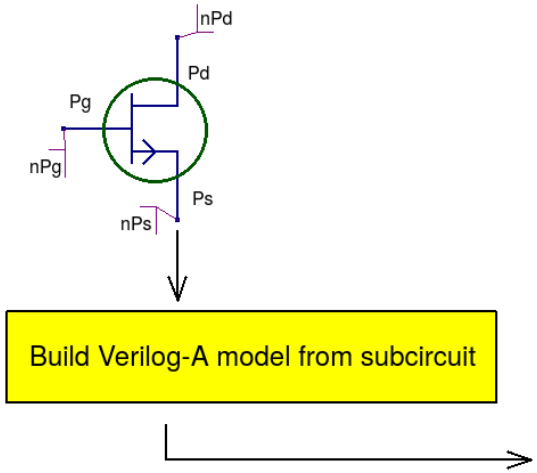
Qucs-S a maturing GPL software package: Compact device modelling Part 3. Qucs-S EDD model of the Efficient Power Corporation (EPC) GaN EPC2001 power transistor; DC test bench and typical output simulation curves

EPC20011
Temp=25
A1=41.7998
alTc=5.486028e-3
k2=2.259866
k3=1.2e-1
aWg=1077
rpara=4.463059e-3
x0_0=0.75
x0_1=1.10
ax0Tc=0.75e-4
dgs1=4.4e-7
dgs2=2.6e-13
dgs3=0.8
dgs4=0.23
ags1=8.6952e-10
agd1=1.4182e-11
asd1=3.3621e-10
asd2=6.308e-10
asd3=1.2803e1
asd4=2.269
asd5=2.518e-10
asd6=-4.0599e1
asd7=2.0638e1
agd2=2.1475e-10
agd3=-3.803
agd4=5.9551
ags2=5.3168e-10
ags3=1.997
ags4=2.8377e-1
ags5=-1.4751e-10
ags6=-7.5163



Qucs-S a maturing GPL software package: Compact device modelling Part 4. Synthesis of Verilog-A code for the Efficient Power Corporation (EPC) GaN EPC2001 power transistor

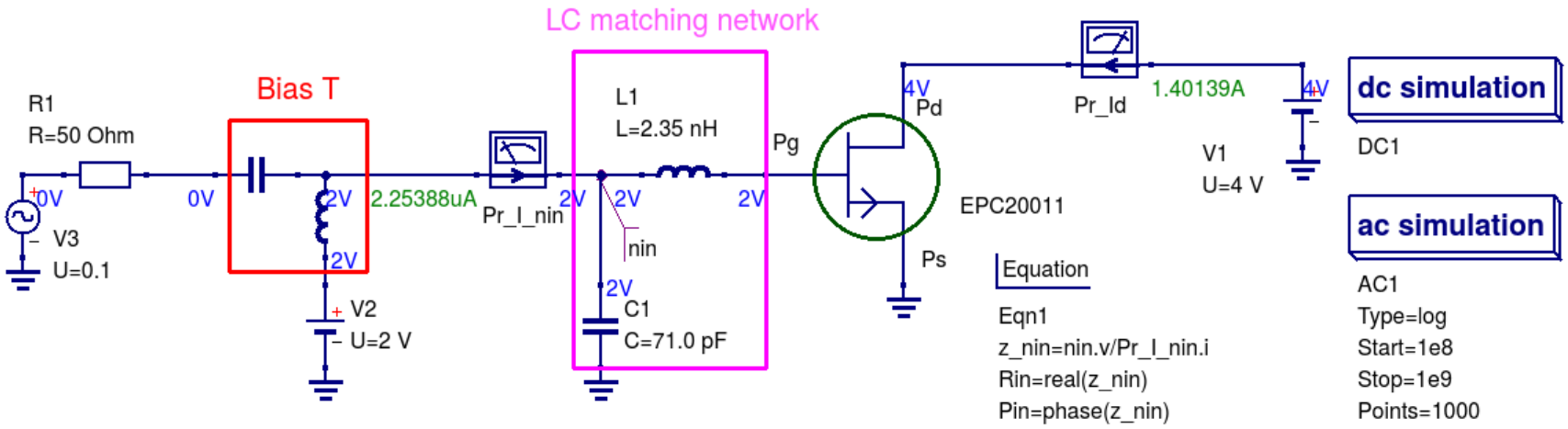
```
EPC20011
Temp=25
A1=41.7998
alTc=5.486028e-3
k2=2.259866
k3=1.2e-1
aWg=1077
rpara=4.463059e-3
x0_0=0.75
x0_1=1.10
ax0Tc=0.75e-4
dgs1=4.4e-7
dgs2=2.6e-13
dgs3=0.8
dgs4=0.23
ags1=8.6952e-10
agd1=1.4182e-11
asd1=3.3621e-10
asd2=6.308e-10
asd3=1.2803e1
asd4=2.269
asd5=2.518e-10
asd6=-4.0599e1
asd7=2.0638e1
agd2=2.1475e-10
agd3=-3.803
agd4=5.9551
ags2=5.3168e-10
ags3=1.997
ags4=2.8377e-1
ags5=-1.4751e-10
ags6=-7.5163
ags7=7.121
```

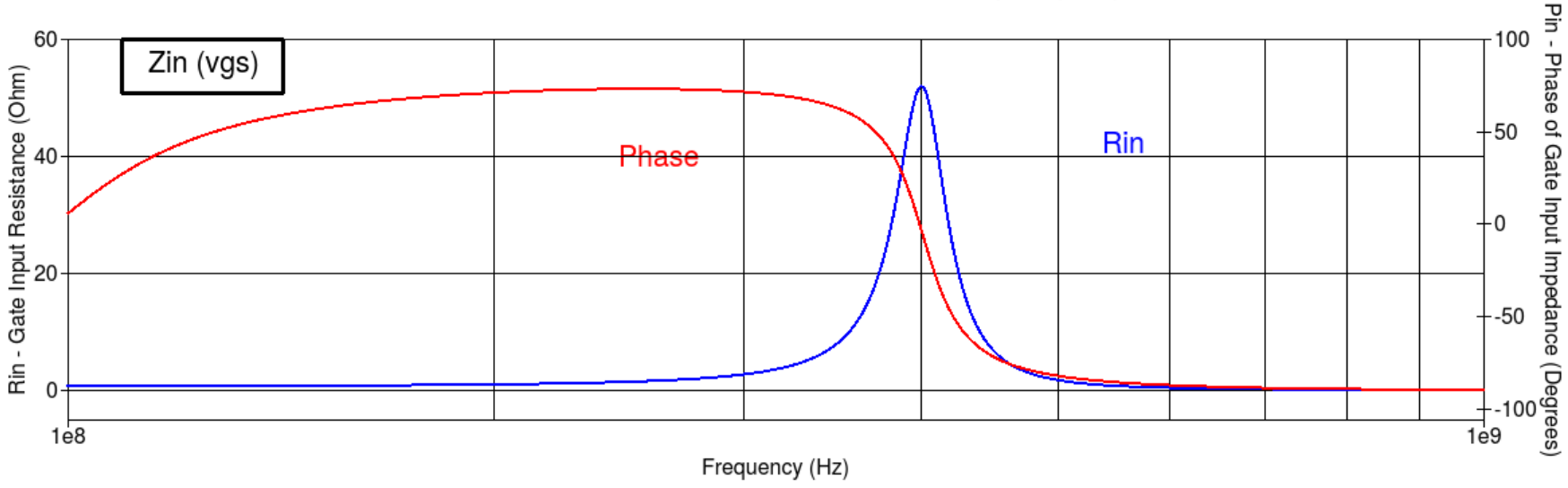


EPC2001 Synthesized Verilog-A module code

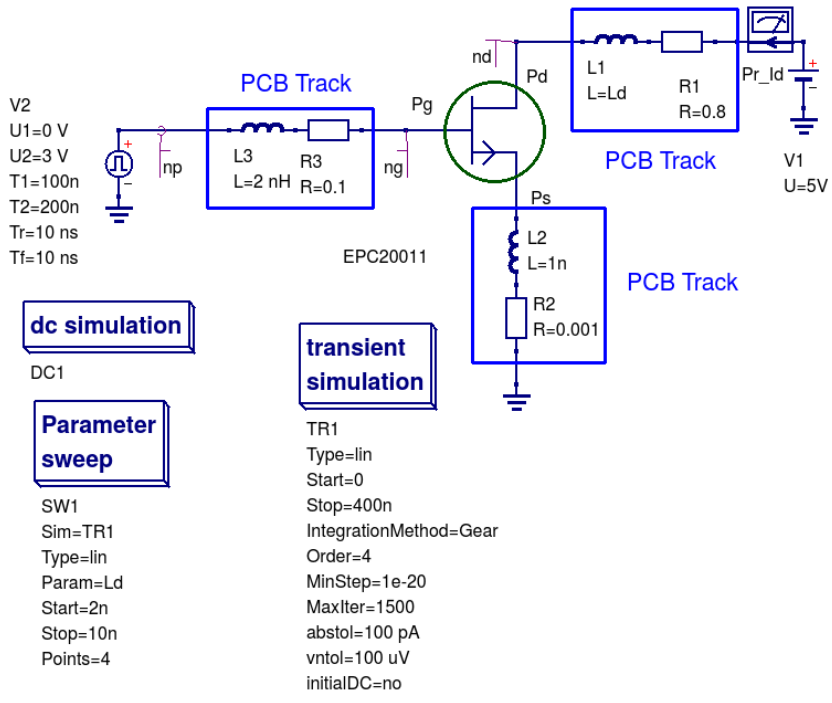
```
`include "disciplines.vams"
`include "constants.vams"
module EPC2001(nPs, nPd, nPg);
inout nPs, nPd, nPg;  electrical ng, ns, nd, nPs, nPd, nPg;
parameter real Temp=25;          parameter real A1=41.7998;     parameter real alTc=5.486028e-3;
parameter real k2=2.259866;      parameter real k3=1.2e-1;      parameter real aWg=1077;
parameter real rpara=4.463059e-3; parameter real x0_0=0.75;     parameter real x0_1=1.10;
parameter real ax0Tc=0.75e-4;    parameter real dgs1=4.4e-7;    parameter real dgs2=2.6e-13;
parameter real dgs3=0.8;         parameter real dgs4=0.23;      parameter real ags1=8.6952e-10;
parameter real agd1=1.4182e-11;  parameter real asd1=3.3621e-10;
parameter real asd2=6.308e-10;   parameter real asd3=1.2803e1;  parameter real asd4=2.269;
parameter real asd5=2.518e-10;   parameter real asd6=-4.0599e1; parameter real asd7=2.0638e1;
parameter real agd2=2.1475e-10;  parameter real agd3=-3.803;    parameter real agd4=5.9551;
parameter real ags2=5.3168e-10;  parameter real ags3=1.997;     parameter real ags4=2.8377e-1;
parameter real ags5=-1.4751e-10; parameter real ags6=-7.5163;   parameter real ags7=7.121;
real EQ1, Rgs, Rgd, Rds, Rd, Rs, Rg, EQ2, EQ3;
analog begin
@(initial_model)  begin
EQ1=A1*(1-alTc*(Temp-25)); Rgs=1e11/aWg; Rgd=1e11/aWg; Rds=1e11/aWg; Rd=0.75*rpara;
Rs=0.25*rpara; Rg=0.6*1077/aWg; EQ2=1.0+ax0Tc*(Temp-25)*(Temp-25); EQ3=0.5*aWg;
end
I(ng,ns) <+ V(ng,ns)/( Rgs );    I(ng,ns) <+ white_noise( 4.0*`P_K*( 26.85 + 273.15) / ( Rgs ), "thermal" );
I(ng,nd) <+ V(ng,nd)/( Rgd );    I(ng,nd) <+ white_noise( 4.0*`P_K*( 26.85 + 273.15) / ( Rgd ), "thermal" );
I(ns,nd) <+ V(ns,nd)/( Rds );    I(ns,nd) <+ white_noise( 4.0*`P_K*( 26.85 + 273.15) / ( Rds ), "thermal" );
I(nd,ns) <+ (V(nd,ns)>0) ?  EQ1*ln(1.0+exp((V(ng,ns)-k2)/k3))*V(nd,ns)/(1.0+max((x0_0+x0_1*V(ng,ns)/EQ2),0.5)*V(nd,ns))
                         :  -EQ1*ln(1.0+exp((V(ng,nd)-k2)/k3))*(-V(nd,ns))/(1.0+max((x0_0+x0_1*V(ng,nd)/EQ2),0.5)*(-V(nd,ns)));
I(nd,ns) <+ ddt( asd1*V(nd,ns)+(asd2*asd4*ln(1+exp((V(nd,ns)-asd3)/asd4))+asd5*asd7*ln(1+exp((V(nd,ns)-asd6)/asd7))) );
I(ng,nd) <+ (V(ng,nd)>10) ? (EQ3/1077*(dgs1*(exp(10.0/dgs3)-1)+dgs2*(exp(10/dgs4)-1)))
                          :  (EQ3/1077*(dgs1*(exp(V(ng,nd)/dgs3)-1)+dgs2*(exp(V(ng,nd)/dgs4)-1)));
I(ng,nd) <+ ddt( agd1*V(ng,nd)+(0.5*ags2*ags4*ln(1+exp((V(ng,nd)-ags3)/ags4))+agd2*agd4*ln(1+exp((V(ng,nd)-agd3)/agd4))) );
I(nPs,ns) <+ V(nPs,ns)/( Rs );
I(nPs,ns) <+ white_noise( 4.0*`P_K*( 26.85 + 273.15) / ( Rs ), "thermal" );    I(nd,nPd) <+ V(nd,nPd)/( Rd );
I(nd,nPd) <+ white_noise( 4.0*`P_K*( 26.85 + 273.15) / ( Rd ), "thermal" );   I(nPg,ng) <+ V(nPg,ng)/( Rg );
I(nPg,ng) <+ white_noise( 4.0*`P_K*( 26.85 + 273.15) / ( Rg ), "thermal" );
I(ng,ns) <+ (V(ng,ns)>10) ? (EQ3/1077*(dgs1*(exp(10.0/dgs3)-1)+dgs2*(exp(10/dgs4)-1)))
                          :  (EQ3/1077*(dgs1*(exp(V(ng,ns)/dgs3)-1)+dgs2*(exp(V(ng,ns)/dgs4)-1)));
I(ng,ns) <+ ddt( ags1*V(ng,ns)+(0.5*ags2*ags4*ln(1+exp((V(ng,ns)-ags3)/ags4))+ags5*ags7*ln(1+exp((V(nd,ns)-ags6)/ags7))) );
end
endmodule
```

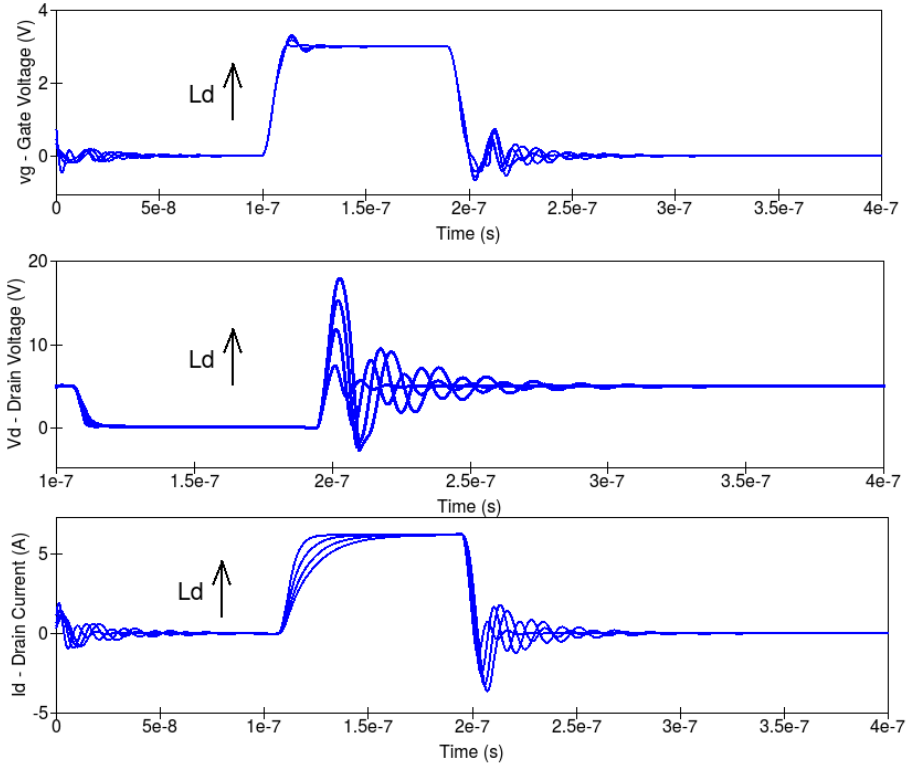
Qucs-S a maturing GPL software package: Compact device modelling Part 5. AC gate matching network, test bench and typical simulation results





Qucs-S a maturing GPL software package: Compact device modelling Part 6. Switching response test bench and typical simulation results





Qucs-S a maturing GPL software package: Verilog-A modeling of the MIT virtual source GaN-RF HEMT 1.0.0: Problems simulating with ADMS; workarounds and typical simulationdData - Part 1. Introduction

- The Analogue Device Model Synthesizer (ADMS) version 2.3.5 is used by Qucs/Qucs-S, Ngspice, Xyce and Gnucap GPL circuit simulators.
- ADMS is based on a subset of Verilog-A HDL selected for compact device modelling.
- Although the Verilog-A HDL is standardised there is no guarantee that individual simulator implementations allow the same dialect of Verilog-A for modelling purposes, for example Qucs/Qucs-S Verilog-A models can include component noise while Ngspice does not implement thermal, shot or flicker noise.
- Normally emerging technology Verilog-A compact models have to be modified, often by hand, to compile without error: specific areas which can cause problems are
  - Internal node collapsing,
  - Voltage limiting,
  - Setting initial conditions,
  - Model equations that include complex combinations of analogue functions,
  - Thermal effects due to power dissipation.

Qucs-S a maturing GPL software package: Verilog-A Modeling of the MIT Virtual Source GaN-RF HEMT 1.0.0: Problems simulating with ADMS; workarounds and typical simulation data - Part 2. Model parameter statement error workarounds

- ADMS parameter statements DO NOT ALLOW reference to previously defined model parameters.

X ADMS synthesis/compile error

```
parameter real  vxord      = 1.30e7     from [0:inf);         // Source injection velocity [cm/s]
parameter real  VtOrd      = -2.0;           // Threshold voltage of drain access transistor[V]
parameter real  Cgrd       = 5.0e-7     from (0:inf);         // Drain access areal capacitance [F/cm2]
parameter real  delta1rd   = 1.3        from [0:inf);         // DIBL for drain access transitor
parameter real  delta2rd   = 0.30       from [0:inf);         // DIBL for drain access transitor
parameter real  Vdibsat    = 2.0        from [0:inf);         // DIBL for drain access transitor
parameter real  Srd        = 0.35       from [0:inf);         // Subthreshold slope for drain access transitor [V/Dec]
parameter real  zeta       = 0.0        from [0:inf);         // Self heating parameter (scalable)
parameter real  betard     = 1.3        from [0:inf);         // Linear to saturation transition parameter
parameter real  vthetard   = 0.05       from [0:inf);         // Scattering: velocity reduction parameter with Vg
parameter real  ndrd       = 0*0.80     from [0:inf);         // Punchthrough factor affects slope change in subthreshold

parameter real  vxors      = vxord      from [0:inf);         // Source injection velocity [cm/s]
parameter real  VtOrs      = VtOrd;                  // Threshold voltage of drain access transistor[V]
parameter real  Cgrs       = Cgrd       from (0:inf);         // Drain access areal capacitance [F/cm2]
parameter real  delta1rs   = delta1rd   from [0:inf);         // DIBL for drain access transitor
parameter real  delta2rs   = delta2rd   from [0:inf);         // DIBL for drain access transitor
parameter real  Srs        = Srd        from [0:inf);         // Subthreshold slope for drain access transitor [V/Dec]
parameter real  vthetars   = 0.05       from [0:inf);         // Scattering: velocity reduction parameter with Vg
parameter real  ndrs       = ndrd       from [0:inf);         // Punchthrough factor affects slope change in subthreshold
parameter real  betars     = betard     from [0:inf);         // Linear to saturation transition parameter
```

OK

```
parameter real  vxord      = 1.30e7     from [0:inf);
parameter real  VtOrd      = -2.0;
parameter real  Cgrd       = 5.0e-7     from (0:inf);
parameter real  delta1rd   = 1.3        from [0:inf);
parameter real  delta2rd   = 0.30       from [0:inf);
parameter real  Vdibsat    = 2.0        from [0:inf);
parameter real  Srd        = 0.35       from [0:inf);
parameter real  zeta       = 0.0        from [0:inf);
parameter real  betard     = 1.3        from [0:inf);
parameter real  vthetard   = 0.05       from [0:inf);
parameter real  ndrd       = 0.80       from [0:inf);

parameter real  VtOrs      = -2.0;
parameter real  Vxors      = 1.30e7;
parameter real  Cgrs       = 5.0e-7     from (0:inf);
parameter real  delta1rs   = 1.3        from [0:inf);
parameter real  delta2rs   = 0.30       from [0:inf);
parameter real  Srs        = 0.35       from [0:inf);
parameter real  vthetars   = 0.05       from [0:inf);
parameter real  ndrs       = 0.80       from [0:inf);
parameter real  betars     = 1.3        from [0:inf);
```

Qucs-S a maturing GPL software package: Verilog-A Modeling of the MIT Virtual Source GaN-RF HEMT 1.0.0: Problems simulating with ADMS; workarounds and typical simulation data - Part 3. Removing $V(n) <+ statements$

- ADMS DOES NOT ALLOW voltage contributions of the form $V(n) <+ I(n)R$, where R is a resistance in Ω,
- OR statements of the form $V(n) <+ 0.0$,
- Resistors, for example 0.001, are used to short nodes (node collapsing), with $I(n) <+ V(N)/0.001$.

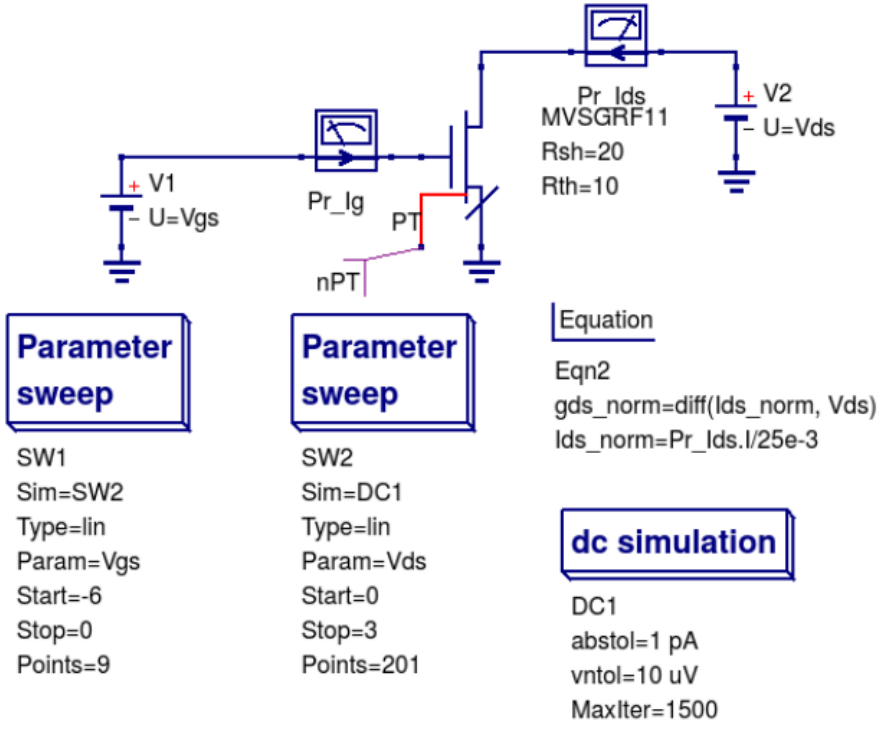
X ADMS synthesis/compile error

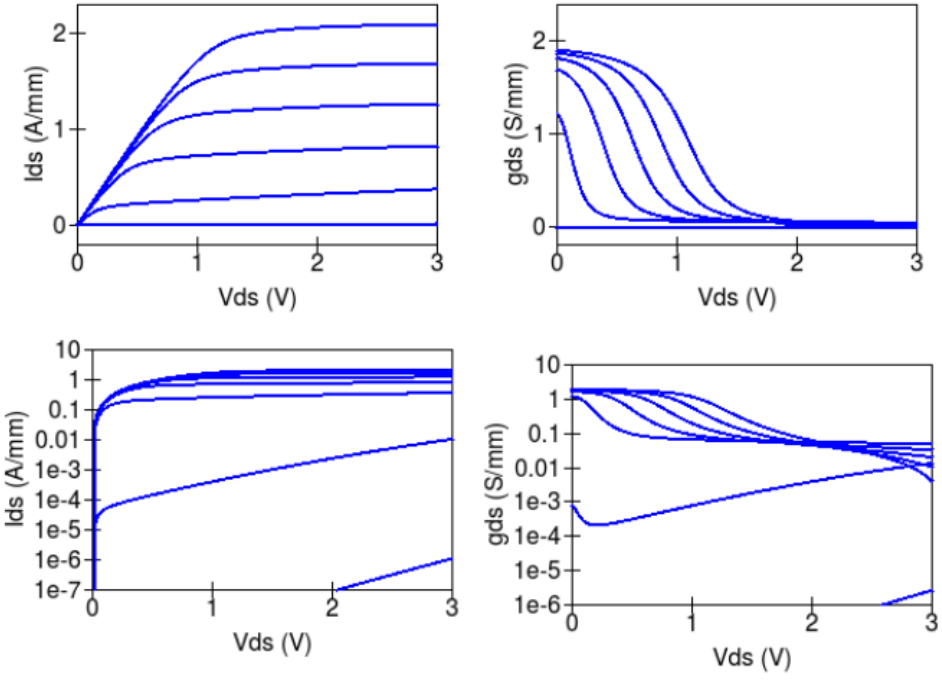
```
if (Rsh > 1e-3 && Ls > 0)
    I(si,src) <+ Idsrs;
else
   V(src,si) <+ 0;
//Source side contact resistance
if (Rc > 0) begin
  I(src,s)    <+ V(src,s) / ( Rc / Wg );
end else begin
  V(src,s)    <+ 0;
end
```

OK

```
if (Rsh > 1e-3 && Ls > 0)
    I(si,src) <+ Idsrs;
else
    I(si,src) <+ V(si, src)/1e-3;
//Source side contact resistance
if (Rc > 0) begin
  I(src,s)    <+ V(src,s) / ( Rc / Wg );
end else begin
  I(src,s)    <+ V(src,s)/1e-3;
end
```

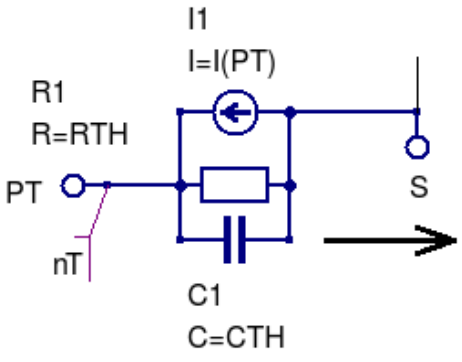
Qucs-S a maturing GPL software package: Verilog-A Modeling of the MIT Virtual Source GaN-RF HEMT 1.0.0: Problems simulating with ADMS; workarounds and typical simulation data - Part 4. DC characteristics





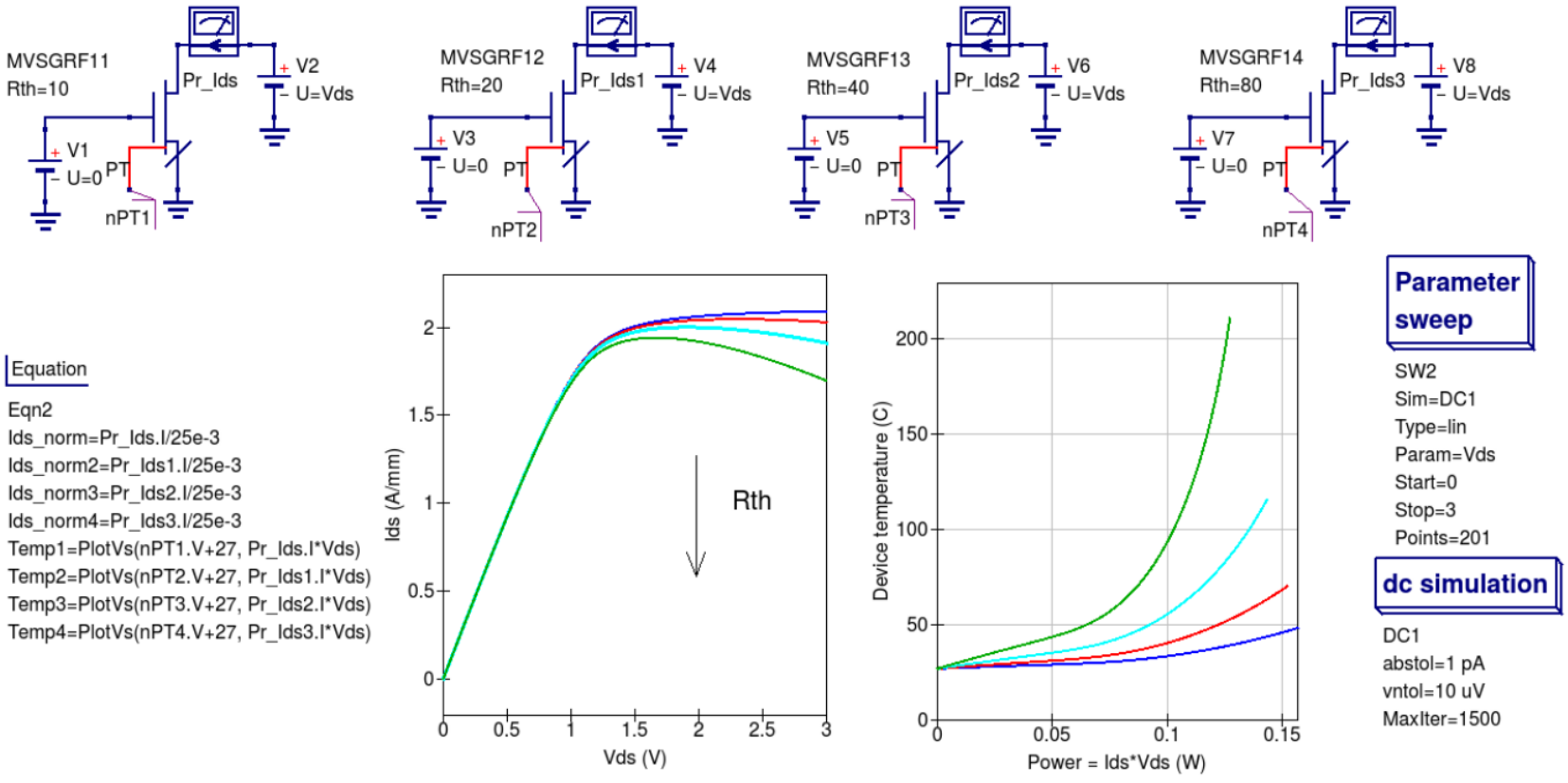
Qucs-S a maturing GPL software package: Verilog-A Modeling of the MIT Virtual Source GaN-RF HEMT 1.0.0: Problems simulating with ADMS; workarounds and typical simulation data - Part 5. Simulating thermal self-heating effects induced by internal power dissipation

- The ADMS dialect of Verilog-A does not implement the $pwr(dt)$ statement,
- Device self-heating is often modelled with a parallel RC network where the volt drop across the RC combination represents the change in device temperature due to internal power dissipation,
- $Tth = RthPd + Temp(Pd = 0)$, where Tth is the device temperature at power dissipation $Pd(W)$.




```
// Self-heating
    I(PT)  <+ ddt( Cth * V(PT) );
    I(PT)  <+ -( I(di,si) * V(di,si) + I(d,drc) * V(d,drc) + I(src,s) * V(src,s) + V(drc,di) * I(drc,di) + V(src,si) * I(src,si) );
    I(PT)  <+ V(PT)/Rth;
```

Qucs-S a maturing GPL software package: Verilog-A Modeling of the MIT Virtual Source GaN-RF HEMT 1.0.0: Problems simulating with ADMS; workarounds and typical simulation data - Part 6. Variation of thermal resistance *Rth* and its effect on DC characteristics

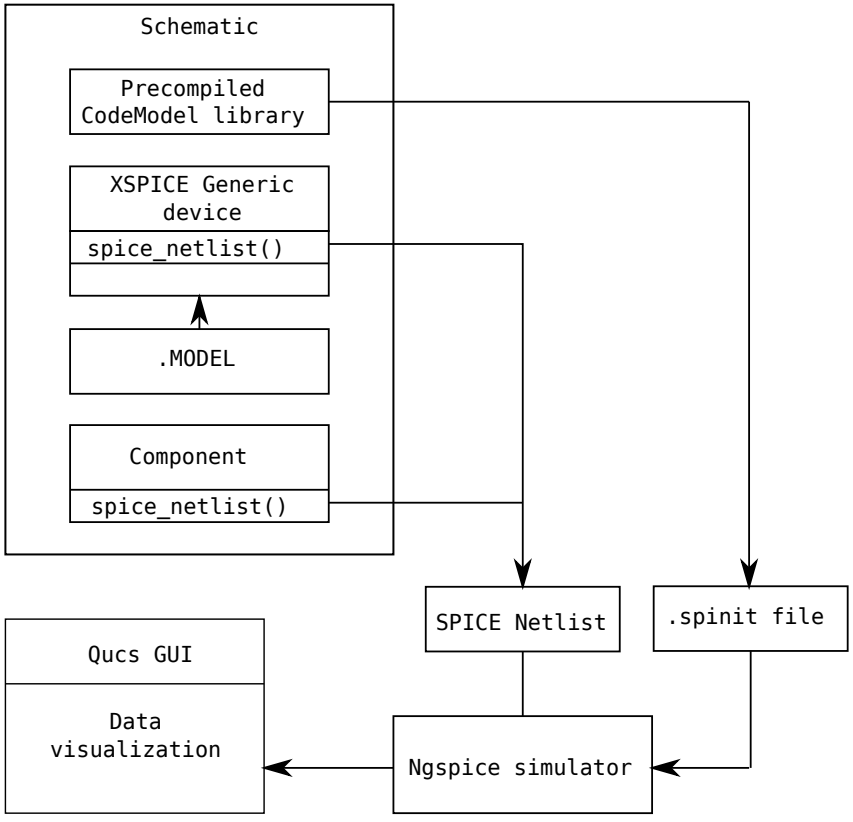


Qucs-S a maturing GPL software package: Modelling tool additions and new features

- Qucs-S includes for the first time a turn-key XSPICE "code level modelling" package for use with the Ngspice and SPICE OPUS circuit simulators,
- Qucs-S has also been extended to include a new Qucs/Octave integrated tool set for compact device model and circuit macromodel parameter extraction. The technique employed is based on data fitting and optimization using measured, or manufacturers published device data, compared against simulated circuit data.

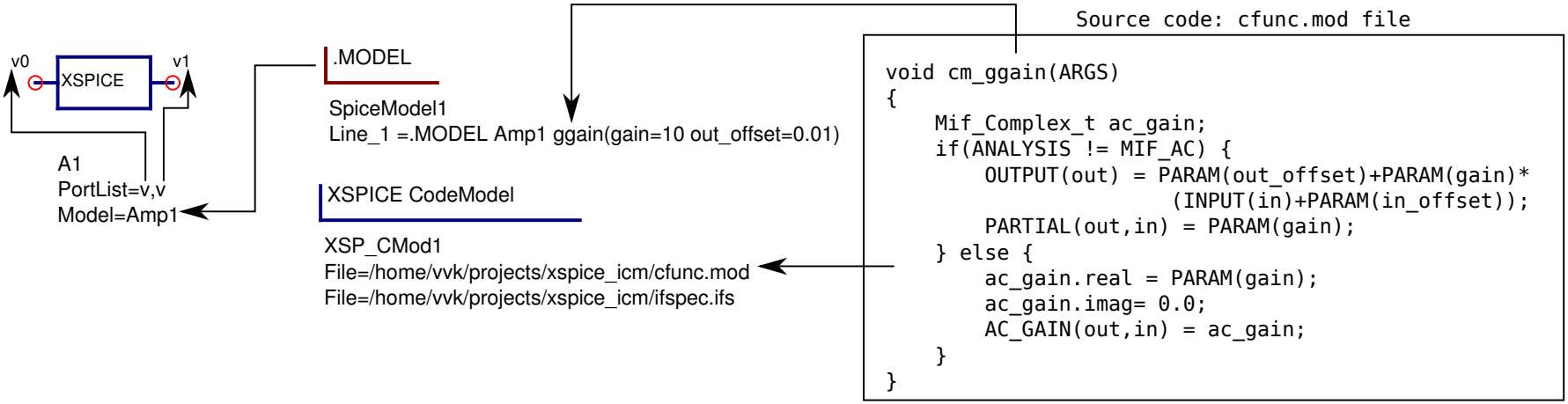
Qucs-S a maturing GPL software package: XSPICE "Code Model" support subsystem

- The XSPICE generic device component is the foundation for
  - Precompiled XSPICE device (*.cm) library support, and
  - Dynamic XSPICE "Code Model" compilation system which allows Code Model sources to be attached to a schematic and compiled automatically at simulation time.

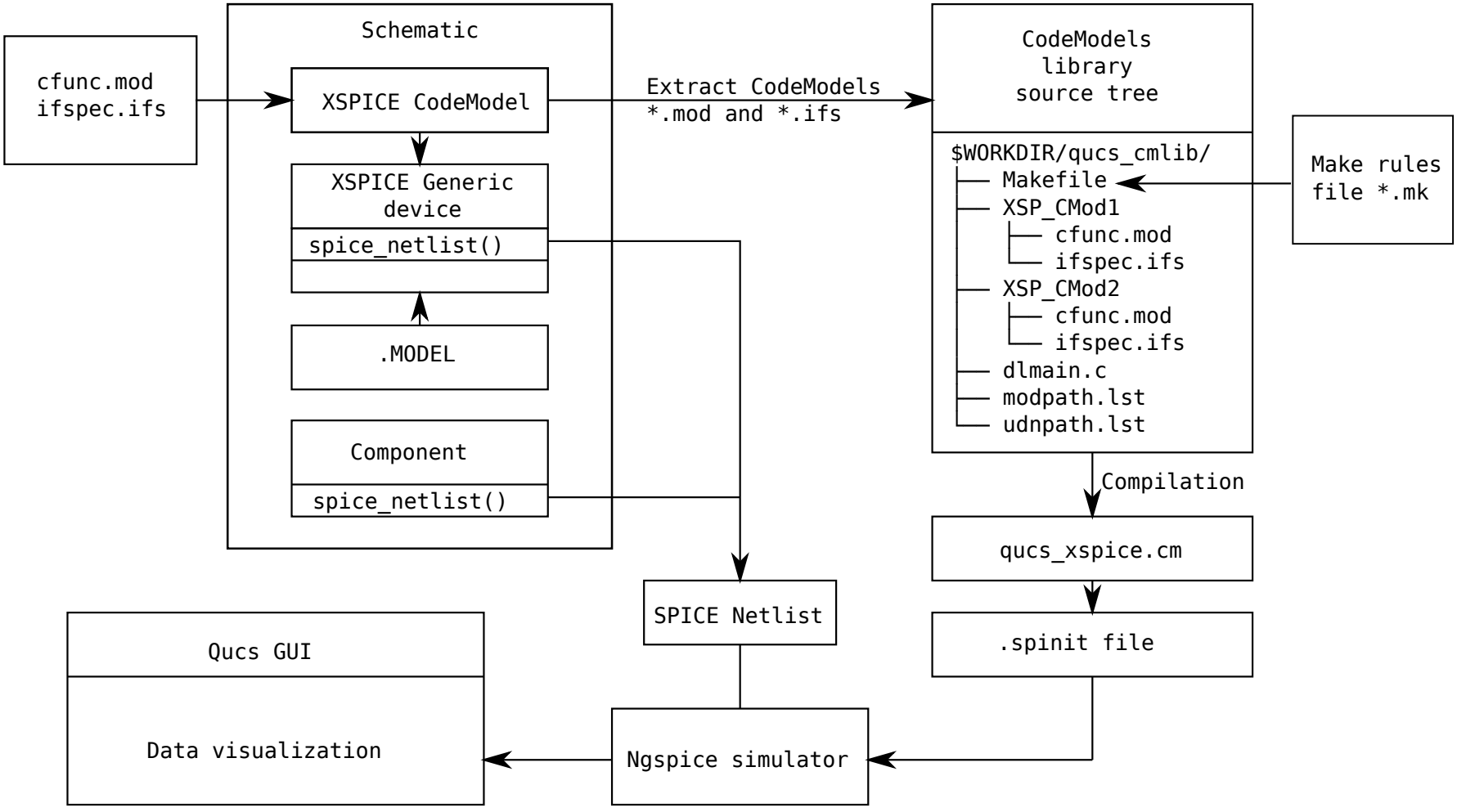


Qucs-S a maturing GPL software package: XSPICE Generic Device component

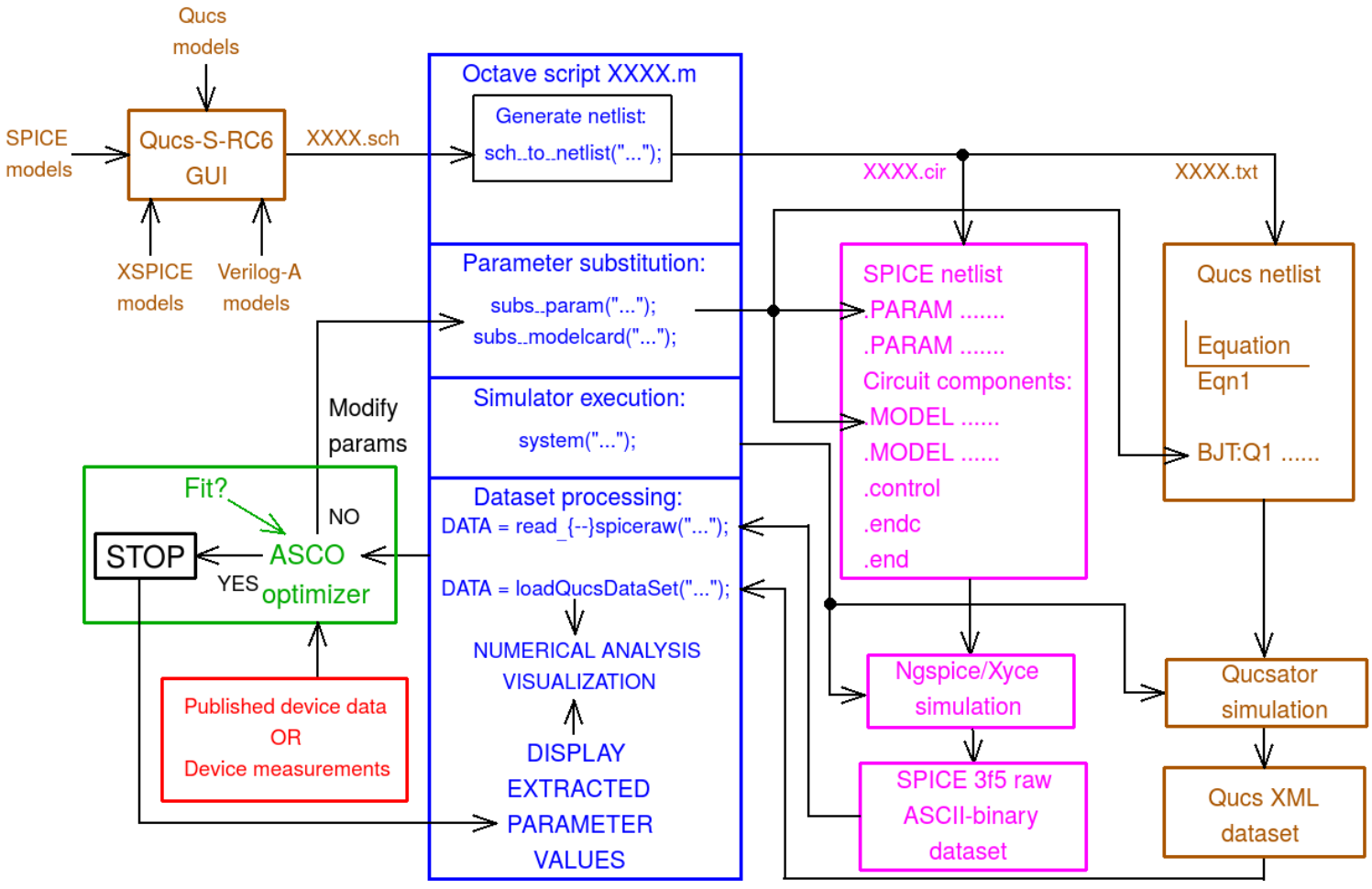
- The XSPICE generic device component is a building block for the construction of user-defined A-devices. It is defined by a comma separated port list, plus XSPICE port designators. These are attached to a SPICE .MODEL statement



Qucs-S a maturing GPL software package: XSPICE Turn-Key Model Generation; compiler system dataflow diagram



Qucs-S a maturing GPL software package: Ngspice/Xyce/SPICEOPUS device parameter extraction from maufacturers data, or measurements, controlled by Octave Script Files; structure diagram



Qucs-S a maturing GPL software package: Future developments - the way forward in 2016-2017

- Qucs-S is a relatively stable circuit simulator and compact modelling tool, incorporating the best features of Qucs, Ngspice. Xyce and SPICE OPUS. The next development phase will concentrate on the following:
  - Testing the package and improving the "feature coverage" by the Qucs-S test suit,
  - Adding the missing sections to the "Qucs-S Help" documentation,
  - Introduction of XSPICE and Xyce digital models as the first step towards making Qucs-S a true mixed-mode electronic system and circuit simulation and modelling package,
  - Improvements to XSPICE CodeModel development system, including new XXX.mod and XXX.ifs templates generated from model schematics,
  - Continue development of the Qucs-S built-in libraries.
- It is expected that the next development phase will last around one year with a series of release candidates published at regular intervals.

Qucs-S a maturing GPL software package: Endnote - A brief look into the future

Qucs-S True mixed-mode simulation and modelling

ANALOGUE and DIGITAL device and IC models

Ngspice and SPICE OPUS XSPICE digital IC models → **In the next development phase**

Xyce behavioural digital IC models

Transistor level digital models

Qucsator Ngspice SPICE OPUS Xyce