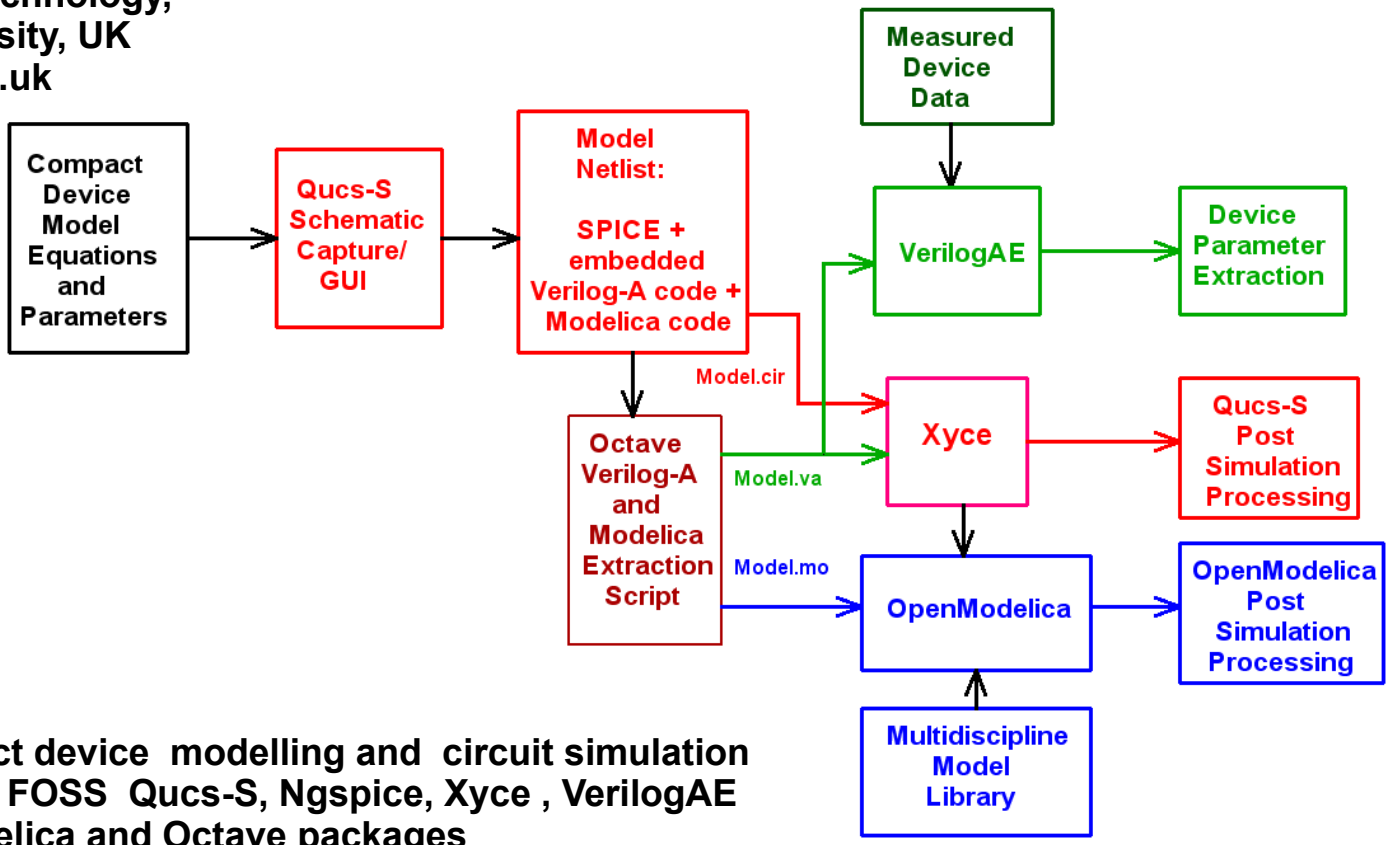
Advances in Qucs-S schematic capture for compact modelling and simulation

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



Proposed compact device modelling and circuit simulation tool-set based on FOSS Qucs-S, Ngspice, Xyce , VerilogAE OpenModelica and Octave packages

Advances in Qucs-S schematic capture for compact modelling and simulation

Qucs-S/Xyce Modular Libraries: New schematic symbols and models

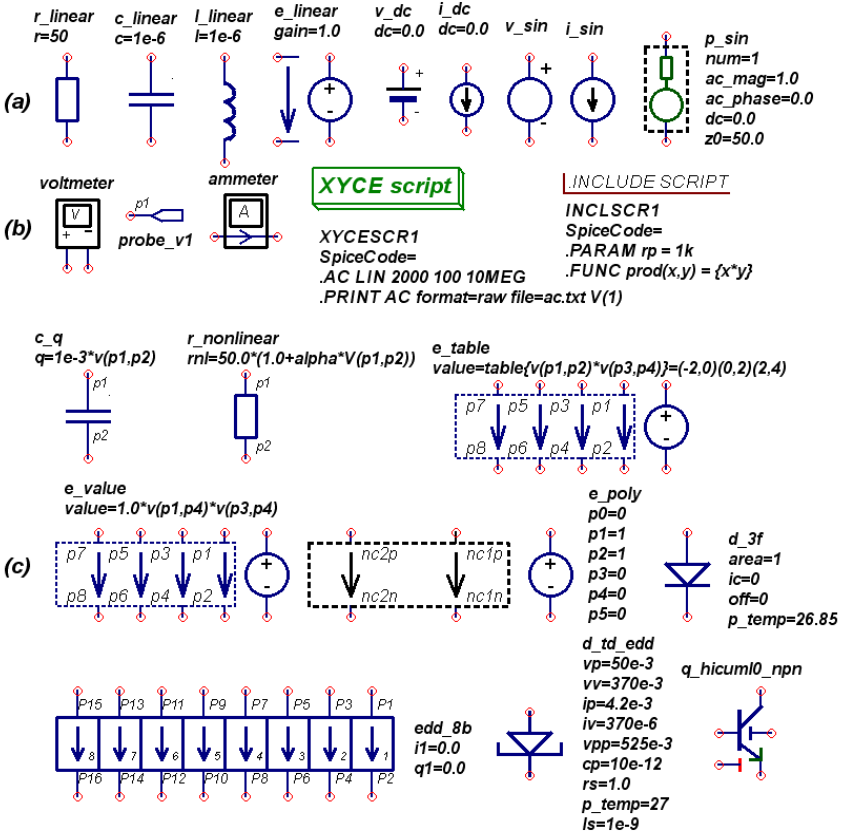
Test library for Qucs-S/Xyce

Points to note:

1. Component and device models in this library are not hard wired into Qucs-S program code;

2. Xyce SPICE dialect becomes the Qucs-S/ Xyce netlist format;

3. The functionality of each schematic symbol is set by Xyce SPICE code.



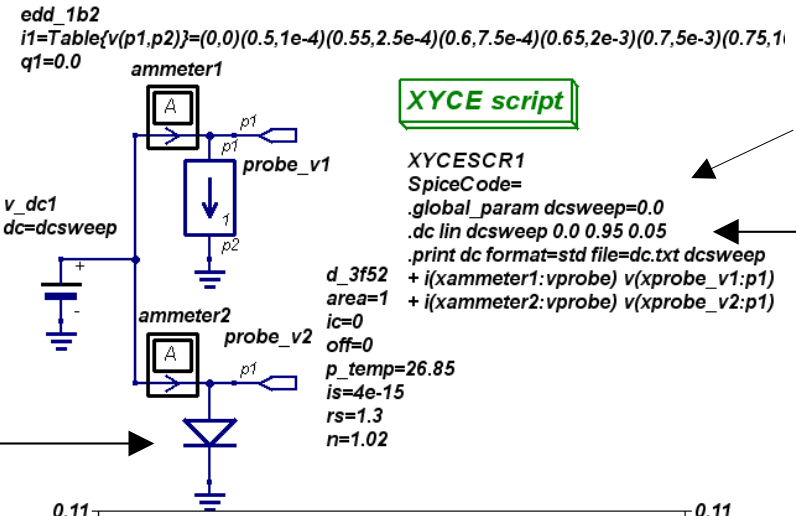
THE CONTENT OF EACH LIBRARY IS DEFINED BY USERS

Advances in Qucs-S schematic capture for compact modelling and simulation

Qucs-S/Xyce Modular Libraries: Schematic symbols and global variables; 1

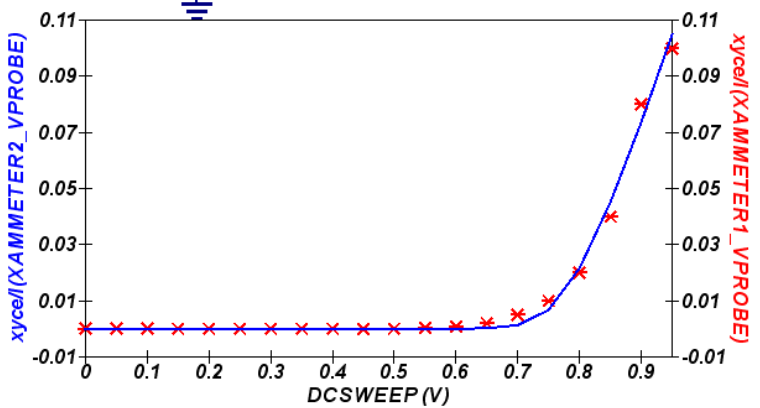
Extended EDD diode model:
Diode Id/Vd characteristic constructed from a numeric Table passed as a subcircuit parameter, allowing the diode model current to be set by the voltage across EDD internal nodes p1 and p2.

SPICE diode model



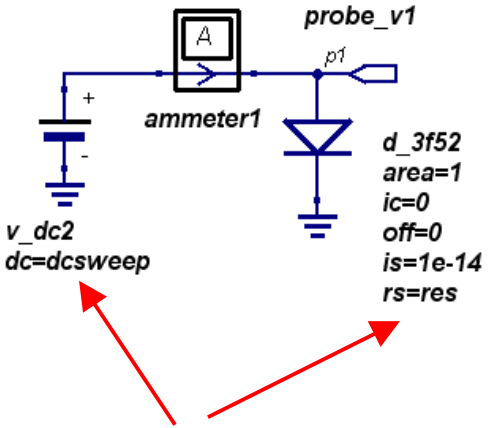
Global parameter dcsweep is Equivalent to a variable that can be changed during simulation.

dcsweep value changed by .dc



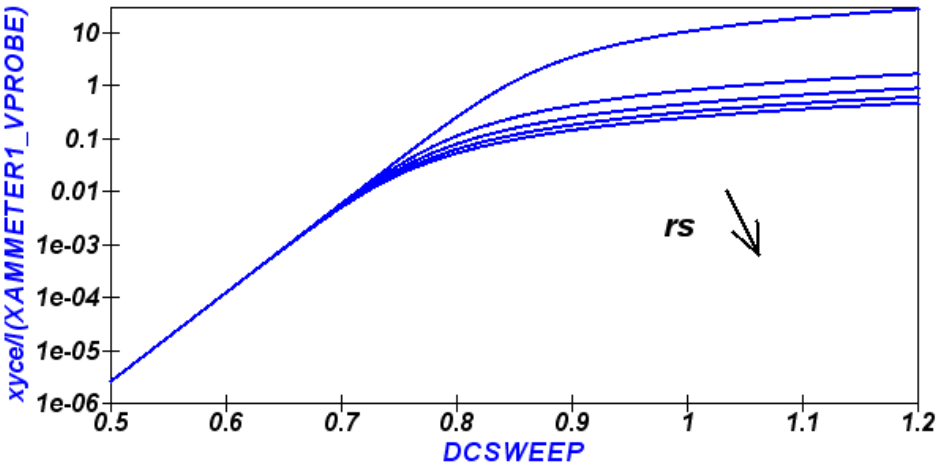
Advances in Qucs-S schematic capture for compact modelling and simulation

Qucs-S/Xyce Modular Libraries: Schematic symbols and global variables; 2



XYCE script

```
XYCESCR1
SpiceCode=
.global_param dcsweep = 0.0
.global_param res=1e-3
.dc lin dcsweep 0.5 1.2 0.02
.step lin res 0.01 1.0 0.2
.print dc format=std file=dc.txt dcsweep
+ i(xammeter1:vprobe) v(xprobe_v1:p1)
```



.global_param

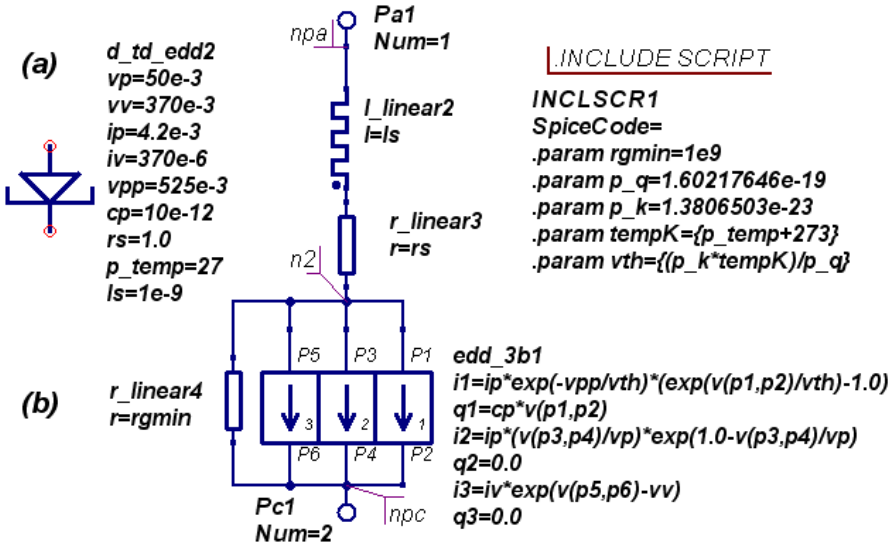
1. dcsweep
2. res

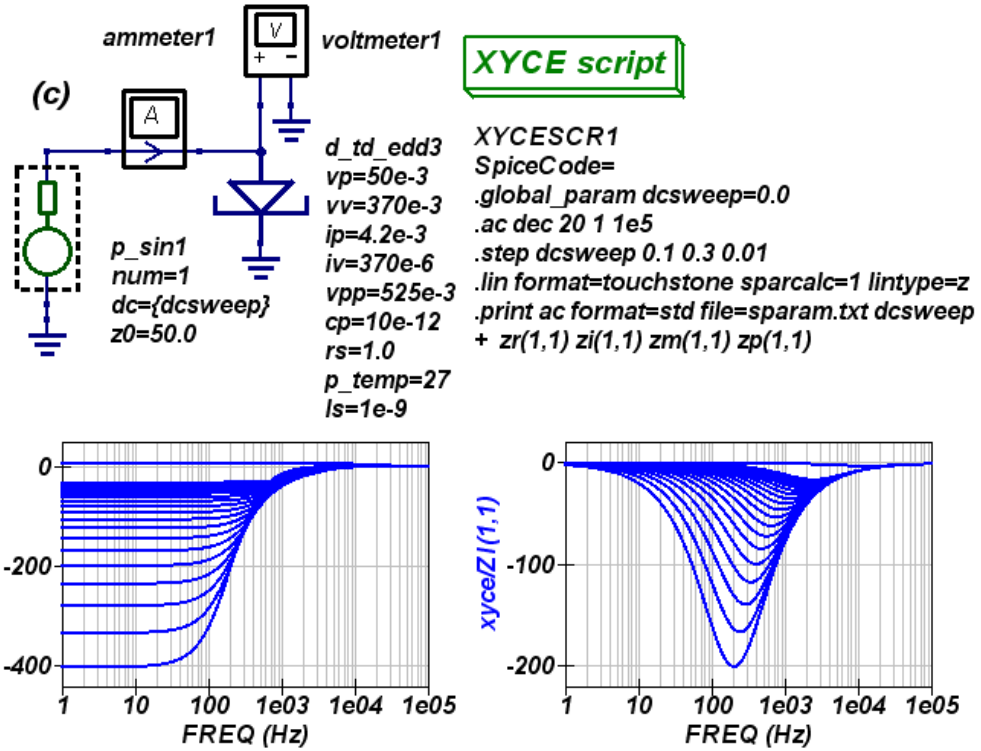
Xyce .dc and .step comands

set dcsweep and res values

Advances in Qucs-S schematic capture for compact modelling and simulation

Qucs-S/Xyce Modular Libraries: Schematic symbols and global variables; 3

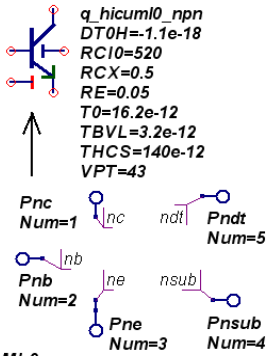




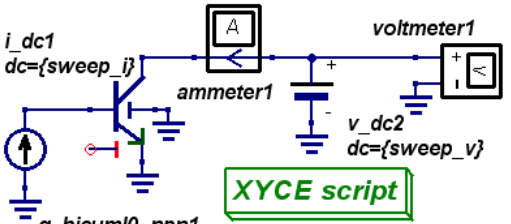
Advances in Qucs-S schematic capture for compact modelling and simulation

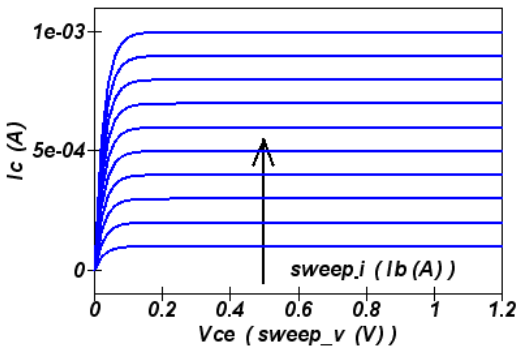
Qucs-S/Xyce Modular Libraries: CMC Verilog-A standardized device models

Xyce Verilog-A HICUML0 BJT model with default parameters - except where changed in symbol parameter list




```
INCLUDE SCRIPT
INCLSCR1
SpiceCode=
rnc nc 0 1e9
rnb nb 0 1e9
rne ne 0 1e9
rnsub nsub 0 1e9
rndt ndt 0 1e9
*
Q1 nc nb ne nsub ndt HICUML0_npn
.MODEL HICUML0_npn npn ( Level = 230
+ AF={AF} AHC={AHC} AHQ={AHQ} AJE={AJE} AJEDC={AJEDC}
+ ALCES={ALCES} ALEAV={ALEAV} ALIQFH={ALIQFH} ALIT={ALIT}
+ ALKAV={ALKAV} ALQF={ALQF} ALT0={ALT0} ALVS={ALVS}
+ AVER={AVER} CBCPAR={CBCPAR} CBEPAR={CBEPAR} CJCI0={CJCI0}
+ CJCX0={CJCX0} CJE0={CJE0} CJS0={CJS0} CTH={CTH} DT0H={DT0H}
+ DVGBE={DVGBE} EAVL={EAVL} F1VG={F1VG} F2VG={F2VG} FBC={FBC}
+ FGEO={FGEO} FIQF={FIQF} FLNQS={FLNQS} FLSH={FLSH} GTE={GTE}
+ IBCS={IBCS} IBES={IBES} IQF={IQF}IQFH= {IQFH} IQR={IQR} IRES={IRES}
+ IS={IS} ISCS={ISCS} IT_MOD={IT_MOD} ITSS={ITSS} KAVL={KAVL}
+ KF={KF} KIQFH={KIQFH} KT0={KT0} MBC={MBC} MBE={MBE} MCF={MCF}
+ MCR={MCR} MRE={MRE} MSC={MSC} MSF={MSF} RBI0={RBI0} RBX={RBX}
+ RCI0={RCI0} RCX={RCX} RE={RE} RTH={RTH} T0={T0} TBVL={TBVL}
+ TEF0={TEF0} TEF_TEMP={TEF_TEMP} TFH={TFH} THCS={THCS}
+ TNOM={TNOM} TR={TR} TYPE={TYPE} VCES={VCES} VDCI={VDCI}
+ VDCX={VDCX} VDE={VDE} VDEDC={VDEDC} VDS={VDS} VEF={VEF}
+ VER={VER} VGB={VGB} VGC={VGC} VGE={VGE} VGS={VGS} VLIM={VLIM}
+ VPT={VPT} VPTCI={VPTCI} VPTCX={VPTCX} VPTS={VPTS}
+ VR0C={VR0C} VR0E={VR0E} ZCI={ZCI} ZCX={ZCX} ZE={ZE} ZEDC={ZEDC}
+ ZETABET={ZETABET} ZETACI={ZETACI} ZETACT={ZETACI}
+ ZETAIQF={ZETAIQF} ZETARBI={ZETARBI} ZETARBX={ZETARBX}
+ ZETARCX={ZETARCX} ZETARE={ZETARE}
+ ZETARTH={ZETARTH} ZETAVER={ZETAVER} ZETAVGBE={ZETAVGBE} )
```





Advances in Qucs-S schematic for capture compact modelling and simulation

Qucs-S/Xyce Modular Libraries: Tunnel diode netlist code

Xyce model SPICE subcircuits

```
* C:/Users/mbrin/.qucs-s/mixdes2021_prj/TDexample.sch

.SUBCKT Xyce_voltmeter  gnd p1 p2
rp p1 p2 1e9
gout 0 pton p1 p2 1.0
rout pton 0 1.0
.ENDS

.SUBCKT Xyce_ammeter  gnd p1 p2
vprobe p1 p2 dc 0.0
.ENDS

.SUBCKT Xyce_p_sin  gnd p1 p2 num=1 ac_mag=1.0 ac_phase=0.0
+ dc=0.0 z0=50.0 tran_vo=0.0 tran_va=1.0 tran_frequency=1000.0
+ tran_phase=0.0
port p1 p2 dc {dc} port={num}  z0={z0}  ac {ac_mag} {ac_phase}
+ sin {tran_vo} {tran_va} {tran_frequency} {tran_phase}
.ENDS

.SUBCKT Xyce_r_linear  gnd p1 p2 r=50 m=1 tc1=0.0 tc2=0.0 tce=0.0
+ p_temp=26.58 dtemp=0.0
r_linear p1 p2 {r} m={m} tc1={tc1} tc2={tc2} tce={tce}
+ temp={p_temp} dtemp={dtemp}
.ENDS

.SUBCKT Xyce_l_linear  gnd p1 p2 l=1e-6 m=1.0 tc1=0.0 tc2=0.0
+ p_temp=26.58
l_linear p1 p2 {l} m={m} tc1={tc1} tc2={tc2} temp={p_temp}
.ENDS

.SUBCKT Xyce_edd_3b  gnd p1 p2 p3 p4 p5 p6 i1=0.0 q1=0.0
+ i2=0.0 q2=0.0 i3=0.0 q3=0.0
bb1   p1 p2  i=i1
cb1   p1 p2 q=q1
*
bb2   p3 p4 i=i2
cb2   p3 p4 q=q2
*
bb3   p5 p6 i=i3
cb3   p5 p6 q=q3
.ENDS
```

Subcircuit X call statements

```
.SUBCKT Xyce_d_td_edd  gnd npa npc vp=50e-3 vv=370e-3 ip=4.2e-3
+  iv=370e-6 vpp=525e-3 cp=10e-12 rs=1.0 p_temp=27 ls=1e-9
.param rgmin=1e9
.param p_q=1.60217646e-19
.param p_k=1.3806503e-23
.param tempK={p_temp+273}
.param vth={(p_k*tempK)/p_q}
Xr_linear1 0  _net0 n2 Xyce_r_linear r={RS} m=1 tc1=0.0
+ tc2=0.0 tce=0.0 p_temp=26.58 dtemp=0.0
Xl_linear1 0  npa _net0 Xyce_l_linear l={LS} m=1.0 tc1=0.0
+  tc2=0.0 p_temp=26.58
Xr_linear2 0  n2 npc Xyce_r_linear r={RGMIN} m=1 tc1=0.0
+ tc2=0.0 tce=0.0 p_temp=26.58 dtemp=0.0
Xedd_3b1 0  n2 npc n2 npc n2 npc Xyce_edd_3b
+  i1={IP*EXP(-VPP/VTH)*(EXP(V(P1,P2)/VTH)-1.0)}
+  q1={CP*V(P1,P2)}
+  i2={IP*(V(P3,P4)/VP)*EXP(1.0-V(P3,P4)/VP)}
+  q2=0.0
+  i3={IV*EXP(V(P5,P6)-VV)}
+ q3=0.0
.ENDS

Xvoltmeter1 0  _net0 0 Xyce_voltmeter
Xammeter1 0  _net1 _net0 Xyce_ammeter{-}
Xp_sin1 0  _net1 0 Xyce_p_sin num=1 ac_mag=1.0
+ ac_phase=0.0 dc={DCSWEEP} z0=50.0 tran_vo=0.0 tran_va=1.0
+ tran_frequency=1000.0 tran_phase=0.0
Xd_td_edd3 0  _net0 0 Xyce_d_td_edd vp=50E-3
+ vv=370E-3 ip=4.2E-3 iv=370E-6 vpp=525E-3
+ cp=10E-12 rs=1.0 p_temp=27 ls=1E-9

.global_param dcsweep=0.0
.ac dec 20 1 1e5
.step dcsweep 0.1 0.3 0.01
.lin format=touchstone sparcalc=1 lintype=z
.print ac format=std file=sparam.txt dcsweep
+  zr(1,1) zi(1,1) zm(1,1) zp(1,1)
.END
```

Xyce control and data output statements

Device count summary	
B level 1	3
C level 1	3
G level 1	1
L level 1	1
R level 1	4
V level 1	2
Total Devices	14

Advances in Qucs-S schematic capture for compact modelling and simulation

Qucs-S/Xyce/Modelica: Multi-HDL schematic capture and code generation; 1

R2
R=
R_Line 2=
R_Line 3=
R_Line 4=
R_Line 5=

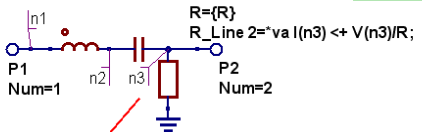
Symbol and SPICE component type

Multiple additional information lines: these allow long strings of SPICE and other HDL data to be passed to Xyce netlists to form a component specification

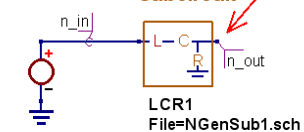
Qucs-S component template : built into C++ GUI code

Subcircuit: internal multi-HDL code (Xyce SPICE and Verilog-A)

```
.INCLUDE SCRIPT
INCLSCR1
SpiceCode=
*va `include "disciplines.vams"
*va `include "constants.vams"
*va module NGenSub1 n1, n2;
*va inout n1, n2;
*va electrical n1, n2, n3;
*va parameter real R=1.5;
*va parameter real C=100e-6;
*va parameter real L=100e-9;
*va analog begin
L1
L={L}
L_Line 2=*va V(n1, n2) <+ ddt(L*I(n1,n2));
C1
C={C}
C_Line 2=*va I(n2, n3) <+ ddt(C*V(n2,n3));
R1
R={R}
R_Line 2=*va I(n3) <+ V(n3)/R;
```



Xyce subcircuit



V1
V=dc 0 ac 1

LCR1
File=NGenSub1.sch
R=1.5
L=100u
C=100n

XYCE script

```
XYCESCR1
SpiceCode=
.AC lin 1000 4e4 6e4
.PRINT AC format=std file=ac.txt
+ V(n_in)  V(n_out) Vm(n_in) Vm(n_out)
+ VP(n_in) VP(n_out)

*va end
*va endmodule
```

Qucs-S schematic drawing

SPICE netlist generated by Qucs-S GUI for Xyce simulation

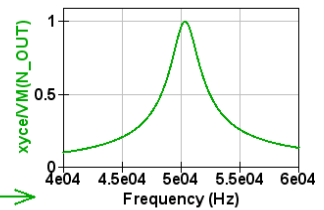
```
* Qucs 0.0.22  C:/Users/mbrin/.qucs-s/LMETResSemSlides_prj/testLCRXyce.sch
.SUBCKT NGenSub1 _net0 n3 R=1 L=100n C=0.1u

*va `include "disciplines.vams"
*va `include "constants.vams"
*va module NGenSub1 n1, n2;
*va inout n1, n2;
*va electrical n1, n2, n3;
*va parameter real R=1.5;
*va parameter real C=100e-6;
*va parameter real L=100e-9;
*va analog begin

C1 n2  n3 {C}
*va I(n2, n3) <+ ddt(C*V(n2,n3));
R1 0  n3 {R}
*va I(n3) <+ V(n3)/R;
L1 n1  n2 {L}
*va V(n1, n2) <+ ddt(L*I(n1,n2));

.ENDS
V1 n_in  0 dc 0 ac 1
XLCR1 n_in n_out NGenSub1 R=1.5 L=100U C=100N

.AC lin 1000 4e4 6e4
.PRINT AC format=std file=ac.txt
+ V(n_in)  V(n_out) Vm(n_in) Vm(n_out)
+ VP(n_in) VP(n_out)
*va end
*va endmodule
.END
```

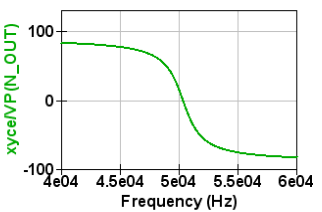
Xyce SPICE netlist with Verilog-A embedded HDL code

Xyce

Note: lines with a * in column 1 are treated by Xyce as documentation COMMENTS

Qucs-S post-simulation data visualization





Advances in Qucs-S schematic capture for compact modelling and simulation

Qucs-S/Xyce/Modelica: Multi-HDL schematic capture and code generation; 2

```
* Qucs 0.0.22  C:/Users/mbrin/.qucs-s/LMETResSemSlides_prj/testLCRXyce.sch
.SUBCKT NGenSub1 _net0 n3 R=1 L=100n C=0.1u

*va `include "disciplines.vams"
*va `include "constants.vams"
*va module NGenSub1 n1, n2;
*va inout n1, n2;
*va electrical n1, n2, n3;
*va parameter real R=1.5;
*va parameter real C=100e-6;
*va parameter real L=100e-9;
*va analog begin

C1 n2  n3 {C}
*va I(n2, n3) <+ ddt(C*V(n2,n3));
R1 0  n3 {R}
*va I(n3) <+ V(n3)/R;
L1 n1  n2 {L}
*va V(n1, n2) <+ ddt(L*I(n1,n2));

.ENDS
V1 n_in  0 dc 0 ac 1
XLCR1 n_in n_out NGenSub1 R=1.5 L=100U C=100N

.AC lin 1000 4e4 6e4
.PRINT AC format=std file=ac.txt
+ V(n_in)  V(n_out) Vm(n_in) Vm(n_out)
+ VP(n_in) VP(n_out)
*va end
*va endmodule
.END
```

NGenSub1.cir

```
function ExtractVa(ModuleName);
% An Octave function to extract a Verilog-A module from a
% Xyce/Ngspice netlist.
% (C) 2021 Mike Brinson: Published under GNU General
%  Public License V2 or later.
% ============== Initialize variables =====================
netlistName = strcat( ModuleName, ".cir");
VaModuleName= strcat( ModuleName, ".va");
fidread=fopen(netlistName,"r"); fidwrite=fopen(VaModuleName,"w");
% ============== Extract Verilog-A module ==================
line=fgetl(fidread);
while 1
if strfind(line,"*va")
  line=[line(4:end)];
  fprintf(fidwrite,"%s\n",line);
endif
line=fgetl(fidread);
if feof(fidread)
%   if ischar(line) fprintf(fidwrite,"%s\n",line); endif
   break
endif
endwhile
fclose(fidread); fclose(fidwrite);
display("Verilog-A module extraction finished.\n");
return
```

Octave Verilog-A module extraction function

```
`include "disciplines.vams"
 `include "constants.vams"
module NGenSub1 n1, n2;
inout n1, n2;
electrical n1, n2, n3;
parameter real R=1.5;
parameter real C=100e-6;
parameter real L=100e-9;
analog begin
I(n2, n3) <+ ddt(C*V(n2,n3));
I(n3) <+ V(n3)/R;
V(n1, n2) <+ ddt(L*I(n1,n2));
end
endmodule
```

NGenSub1.va

Advances in Qucs-S schematic capture for compact modelling and simulation

Qucs-S/Xyce/Modelica: Multi-HDL schematic capture and code generation; 3

The tunnel diode model revisited