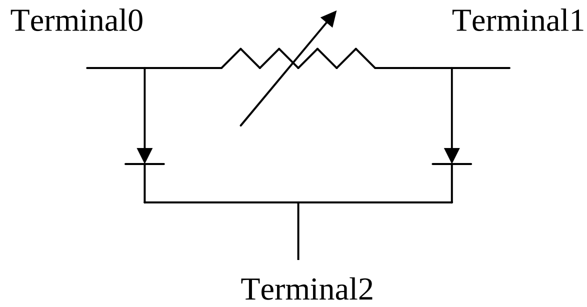
Physical resistor model – n subtype **ResistorPhyN**



Description:
This element implements a semiconductor resistor based on the n subtype of the Cadence physical resistor model.

Form: resistorPhyN:<instance name> n_0 n_1 n_2 <parameter list>

instance name is the model name
n_0 is the positive element terminal (Terminal0),
n_1 is the negative element terminal (Terminal1),
n_2 is the substrate terminal (Terminal2).

Parameters:

Parameter	Type	Default value	Required?
r: Resistance (ohms)	DOUBLE	1E+9	no
coeff0: Constant term of conductance polynomial	DOUBLE	1	no
coeff1: First order coefficient of conductance polynomial	DOUBLE	0	no
coeff2: Second order coefficient of conductance polynomial	DOUBLE	0	no
coeff3: Third order coefficient of conductance polynomial	DOUBLE	0	no
coeff4: Fourth order coefficient of conductance polynomial	DOUBLE	0	no
coeff5: Fifth order coefficient of conductance polynomial	DOUBLE	0	no
polyarg: Polynomial model argument type	BOOLEAN	1 (TRUE)	no
tc1: Linear temperature coefficient of resistor (1/C)	DOUBLE	0	no
tc2: Quadratic temperature coefficient of resistor (1/C^2)	DOUBLE	0	no
tnom: Parameter measurement temperature (K)	DOUBLE	300	no

tdev: Device operating temperature (K)	DOUBLE	300	no
is: Saturation current (A)	DOUBLE	1E-14	no
n: Emission coefficient	DOUBLE	1	no
ibv: Current magnitude at the reverse breakdown voltage (A)	DOUBLE	1E-10	no
bv: Junction reverse breakdown voltage (V)	DOUBLE	0	no
fc: Coefficient for forward-bias depletion capacitance	DOUBLE	0.5	no
cj0: Zero-bias junction capacitance (F)	DOUBLE	0	no
vj: Junction built-in potential (V)	DOUBLE	1.0	no
m: Junction grading coefficient	DOUBLE	0.5	no
tt: Transit time (s)	DOUBLE	0	no
area: Diode area multiplier	DOUBLE	1	no
rs: Diode series resistance (ohms)	DOUBLE	0	no

Example:

resistorPhyN:r2 2 3 0 r=1000.0 coeff0=1.0 coeff1=0.1 coeff2=0.0 coeff3=0.002 coeff4=0.0
+ coeff5=0.00004 polyarg=0 tc1=0.0 tc2=0.0 tnom=300.0 tdev=300.0 is=1E-14 n=1.0
+ ibv=1.0E-10 bv=0.0 fc=0.5 cj0=1.0E-10 vj=1.0 m=0.5 tt=0.0 area=1.0 rs=0.0

Model Documentation:

For polyarg=true:

The controlling voltage for the resistance is:

$$V = ((V(t0) – V(t2)) + (V(t1)-V(t2))) / 2$$

and the resistance is:

$$R(V) = r / (coeff0 + coeff1*V + coeff2*V^2+coeff3*V^3+coeff4*V^4+coeff5*V^5)$$

For polyarg=false:

The controlling voltage for the resistance is:

$$V = V(t0) - V(t1)$$

and the resistance is:

$$R(V) = r / (coeff0 + 1/2*coeff1*V + 1/3*coeff2*V^2 + 1/4*coeff3*V^3 + 1/5*coeff4*V^4 + 1/6*coeff5*V^5)$$

Note that the code does not prevent a negative resistance value; care should be taken in selecting coefficients to ensure that the resulting resistance is positive for all anticipated values of the controlling voltage.

Resistance as a function of temperature is:

$$R(tdev) = R(tnom) * (1 + tc1*(tdev-tnom) + tc2*(tdev-tnom)^2)$$

References:
This model is based on a description of the Cadence Spectre physical resistor model found at http://www.uta.edu/ronc/cadence/ResistorModels.pdf. Code for diodes was taken from SPDiode model written by Carlos E. Christoffersen.

Sample Netlist:

```
**** resistorPhyN transient characteristic ****

* This choice of conductance coefficients should result in positive resistor
* values for Vctrl down to about -5V for polyarg = true or false.

.tran2 tstop=4E-6 tstep=2E-8

res:r1 1 2 r = 1000.0
resistorPhyN:r2 2 3 0 r=1000.0 coeff0=1.0 coeff1=0.1 coeff2=0.0 coeff3=0.002
coeff4=0.0
+ coeff5=0.00004 polyarg=0 tc1=0.0 tc2=0.0 tnom=300.0 tdev=300.0 is=1E-14 n=1.0
+ ibv=1.0E-10 bv=0.0 fc=0.5 cj0=1.0E-10 vj=1.0 m=0.5 tt=0.0 area=1.0 rs=0.0
res:r3 3 0 r = 1000.0
vpulse:vbias 1 0  v1=0 v2=-3.0 td=0 tr=0 tf=0 pw=1E-6 per=2E-6

.out write term 1 vt in "n_tran_vt1.out"
.out write term 2 vt in "n_tran_vt2.out"
.out write term 3 vt in "n_tran_vt3.out"

.end
```

Known Bugs:
None

Credits:

Name	Affiliation	Date	Links
ECE718 Student	NC State University	May 2003	www.ncsu.edu

Description of Model

The goal here was to write a semiconductor resistor model based on a description of the Cadence Spectre physical resistor model. The Cadence physical resistor model consists of three subtypes: n, p, and poly. Each was implemented in fREEDA as a separarate model: resistorPhyN, resistorPhyP, and resistorPhyPoly. The poly subtype consists of a voltage-dependent resistor and a fixed capacitor between each of the resistor terminals and the substrate. The n and p subtypes consist of a voltage-dependent resistor and a diode between each of the resistor terminals and the substrate. For the n subtype, the diodes' anodes are connected to the resistor, while in the p subtype, the diodes' anodes are connected to the substrate. These diodes include junction capacitance.

There are three subtypes for this model, n, p, and poly. Each is contained in a separate file. Subtype n is called resistorPhyN, subtype p is called resistorPhyP, and subtype poly is called resistorPhyPoly. For all three subtypes, the voltage-dependent resistance is determined by coefficients of a fifth-order conductance polynomial and a nominal resistor value, which are model parameters. For the poly subtype, the fixed capacitor value is a model parameter. For the n and p subtypes, a fixed capacitance is not included, but junction capacitance is included in the diodes. Some items which are included in the Cadence model, including noise and device size parameters, were left out of the fREEDA models.

Summary of Operation

For the poly subtype, parameterization is not necessary, and the state variables are terminal voltages and their derivatives. Resistance is determined from the terminal voltages, then terminal currents are calculated. The use of diodes in the n and p subtypes forces re-parameterization as in the SPDiode model. In the n and p subtypes, diode voltages and currents are calculated first from the state variables. The terminal voltages are then set to the diode voltage plus drop across the diode series resistance (or the negative, for subtype p). From the terminal voltages, resistance is calculated, then terminal currents are calculated.

Calculation of Resistor Value

The controlling voltage for the resistor value is selected with the polyarg parameter. In addition to determining the controlling voltage, polyarg also determines how the resistance is calculated from the controlling voltage.

When polyarg is set to false (or "diff" in the documentation on which this model is based), the controlling voltage for the resistance is the voltage across the resistor:

$$V = V(t0) – V(t1).$$

and the current through the resistor is

$$I(V) = (V / r) * (coeff0 + 1/2*coeff1*V + 1/3*coeff2*V^2+...)$$

Though the document on which this model is based does not describe how the expression for current was derived, it seems that for polyarg=false, R is defined as:

$$R = dV / dI$$

so

$$I = integral (dV / R(V))$$
$$I = integral ((coeff0 + coeff1*V + coeff2*V^2 + ...)*dV / r)$$
$$I = (coeff0*V + 1/2*coeff1*V^2 + 1/3*coeff2*V^3 + ...) / r$$
$$I = (V / R(inst)) * (coeff0 + 1 /2*coeff1*V + 1/3*coeff2*V^2 + ...).$$

When polyarg is set to true (or "sum" in the documentation on which this model is based), the controlling voltage for the resistance is the average voltage between the resistor terminals and the substrate.

$$V = ((V(t0) – V(t2)) + (V(t1)-V(t2))) / 2.$$

When polyarg is set to true, current through the resistor is

$$I(V) = (V / R(inst)) * (coeff0 + coeff1*V + coeff2*V^2+...)$$

Given the result above, it seems that for polyarg=true, R is defined as:

$$R = V / I$$

It should be noted that there is no provision in the code to prevent a negative resistor value (depending on the conductance polynomial coefficients, this can occur for some negative values of the controlling voltage). Coefficients must be selected carefully to avoid negative resistor values for any anticipated value of the controlling voltage.

Diodes

The code for the diodes was taken from the SPDiode model. The non-charge-conserving portion of the code was used. For the poly subtype, parameterization is not necessary, and the state variables are terminal voltages and their derivatives. Resistance is determined from the terminal voltages, then terminal currents are calculated. The use of diodes in the n and p subtypes forces re-parameterization as in the SPDiode model. In the n and p subtypes, diode voltages and currents are calculated first from the state variables. The terminal voltages are then set to the diode voltage plus drop across the diode series resistance (or the negative, for subtype p). From the terminal voltages, resistance is calculated, then terminal currents are calculated.

Temperature effects

The phyres model was not implemented as a thermal model. However, a fixed device operating temperature can be specified in the parameter list. The parameter tdev is the device operating temperature, and tnom is the temperature at which device parameters are assumed to have been measured.

Resistance as a function of temperature is:

$$R(tdev) = R(tnom) * (1 + tc1*(tdev-tnom) + tc2*(tdev-tnom)^2)$$

For linear capacitance in the poly subtype, capacitance as a function of temperature is:

$$C(tdev) = C(tnom) * (1 + tc1c*(tdev-tnom) + tc2c*(tdev-tnom)^2)$$

For junction capacitance in the n and p subtypes, temperature coefficients are not used. Instead, any changes in capacitance that occur over temperature are due to tdev being used in diode calculations.