Berkeley Model and Algorithm Prototyping Platform (MAPP)

- ☑ MAPP release website: [*MAPP.eecs.berkeley.edu*](MAPP.eecs.berkeley.edu)
  - ☑ MAPP mailing list;
  - ☑ MAPP bug tracking system;
  - ☑ MAPP git repository.
- ☑ Download and set up MAPP
- ☐ A demo of MAPP
  - ☐ prototype a device model in MAPP;
  - ☐ build a circuit;
  - ☐ run analyses.

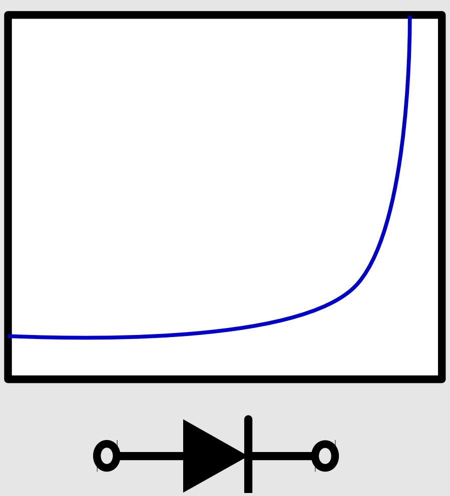
A demo of MAPP

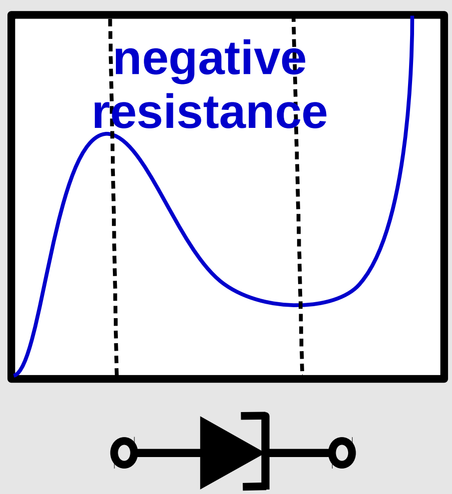
- Now that we have set up MAPP...

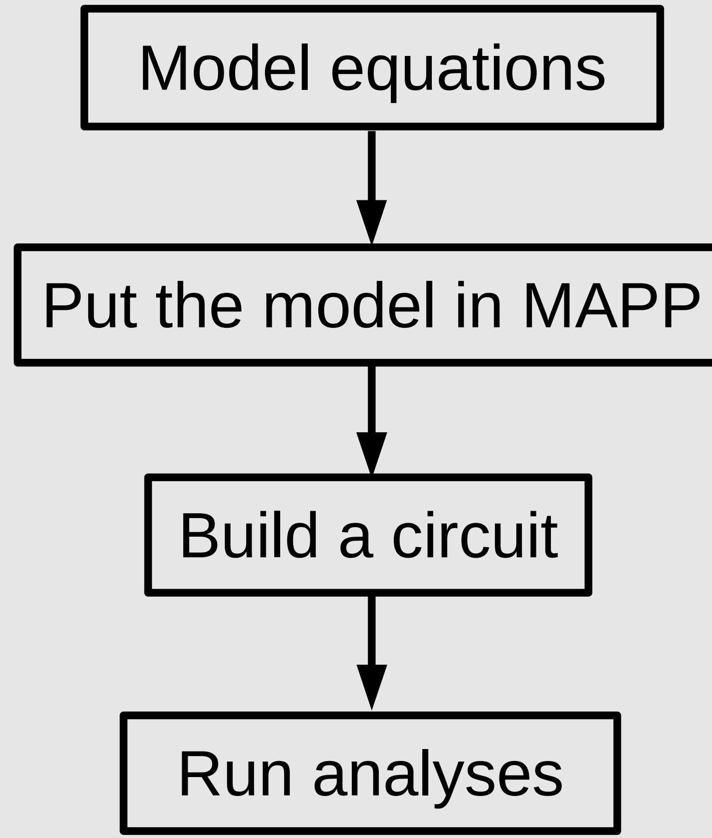
Let's prototype a device in it!

Tunnel Diodes (Esaki Diodes)

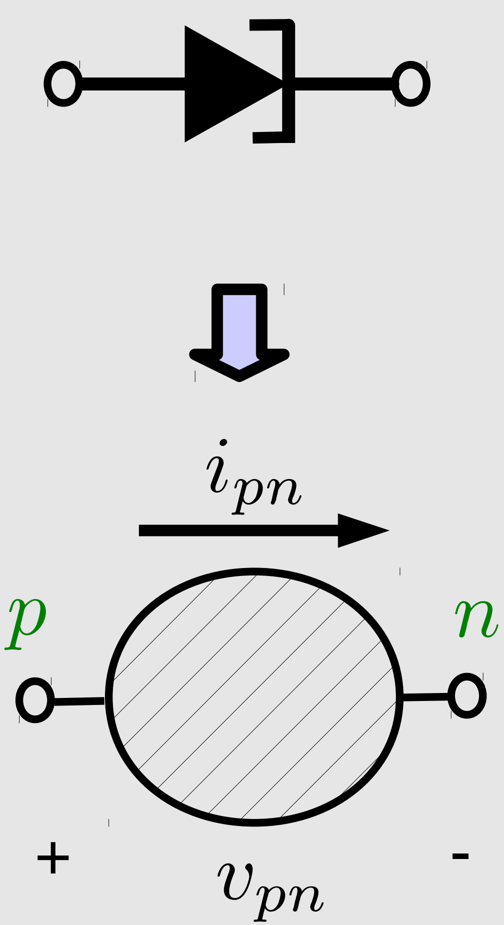
Leo Esaki

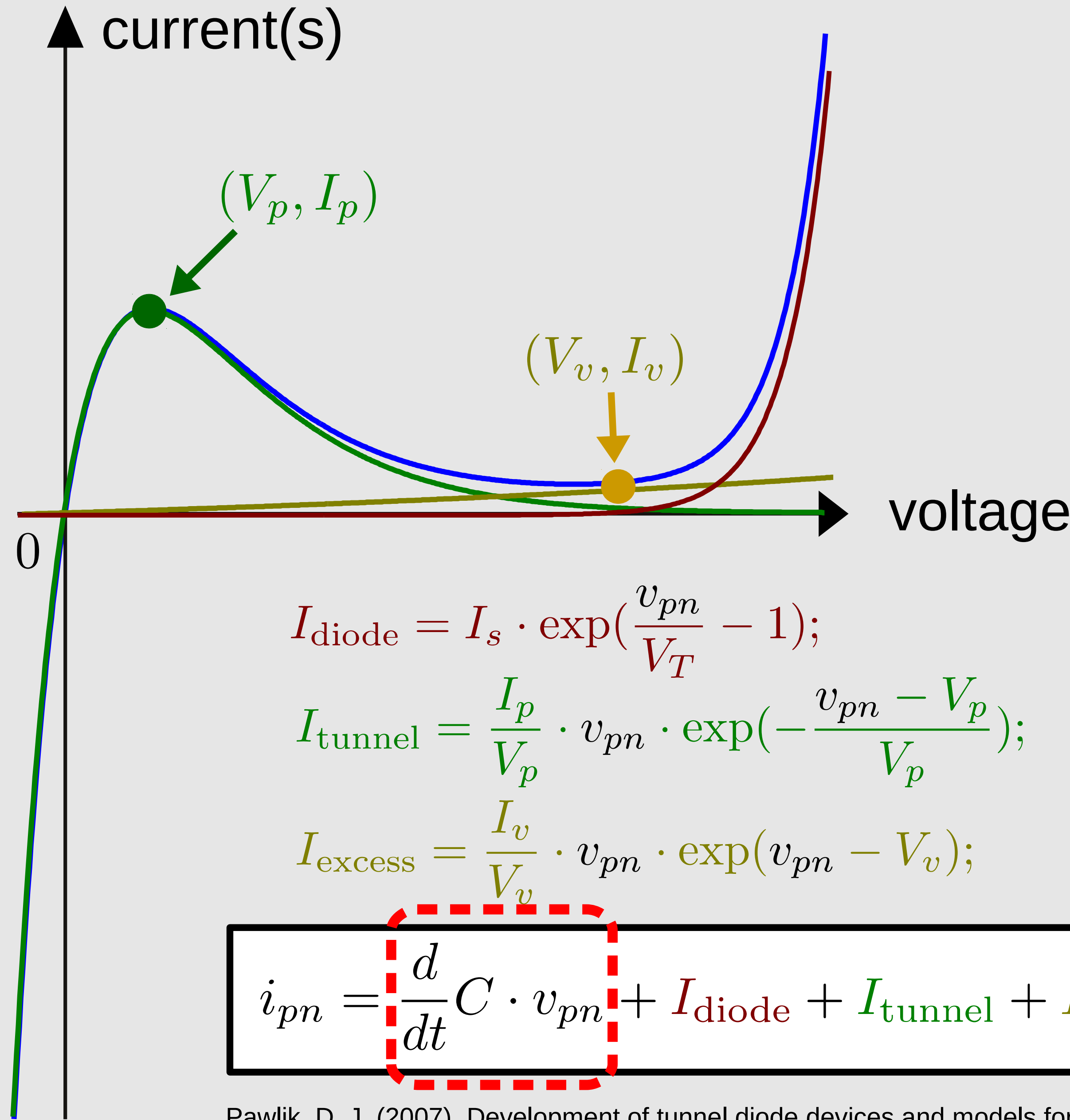






A tunnel diode model





$$I_{\text{diode}} = I_s \cdot \exp\left(\frac{v_{pn}}{V_T} - 1\right);$$

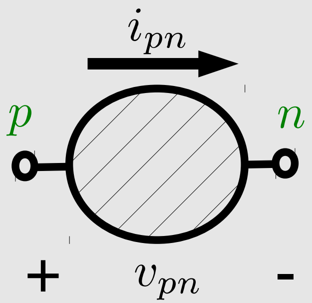
$$I_{\text{tunnel}} = \frac{I_p}{V_p} \cdot v_{pn} \cdot \exp\left(-\frac{v_{pn} - V_p}{V_p}\right);$$

$$I_{\text{excess}} = \frac{I_v}{V_v} \cdot v_{pn} \cdot \exp(v_{pn} - V_v);$$

$$i_{pn} = \frac{d}{dt} C \cdot v_{pn} + I_{\text{diode}} + I_{\text{tunnel}} + I_{\text{excess}}$$

Pawlik, D. J. (2007). Development of tunnel diode devices and models for circuit design and characterization (Doctoral dissertation, Rochester Institute of Technology).

Tunnel diode model in MAPP



$$i_{pn} = \frac{d}{dt} C \cdot v_{pn} + I_{\text{diode}} + I_{\text{tunnel}} + I_{\text{excess}} \Rightarrow i_{pn} = \frac{d}{dt} q(v_{pn}) + f(v_{pn})$$

```
function MOD = Tunnel_Diode_ModSpec()
  MOD = ee_model();
  MOD = add_to_ee_model(MOD, 'terminals', {'p', 'n'});
  MOD = add_to_ee_model(MOD, 'explicit_outs', {'ipn'});
  MOD = add_to_ee_model(MOD, 'parms', {'Is', 1e-12, 'VT', 0.025});
  MOD = add_to_ee_model(MOD, 'parms', {'Ip', 3e-5, 'Vp', 0.05});
  MOD = add_to_ee_model(MOD, 'parms', {'Iv', 3e-6, 'Vv', 0.3});
  MOD = add_to_ee_model(MOD, 'parms', {'C', 1e-15});
  MOD = add_to_ee_model(MOD, 'f', @f);
  MOD = add_to_ee_model(MOD, 'q', @q);
  MOD = finish_ee_model(MOD);
end
```

Is, VT, Ip, Vp,
Iv, Vv, C
vpn

```
function out = f(S)
  v2struct(S);
  I_diode = Is*(exp(vpn/VT)-1);
  I_tunnel = (Ip/Vp) * vpn * exp(-1/Vp * (vpn - Vp));
  I_excess = (Iv/Vv) * vpn * exp(vpn - Vv);
  out = I_diode + I_tunnel + I_excess;
end
```

```
function out = q(S)
  v2struct(S);
  out = C*vpn;
end
```

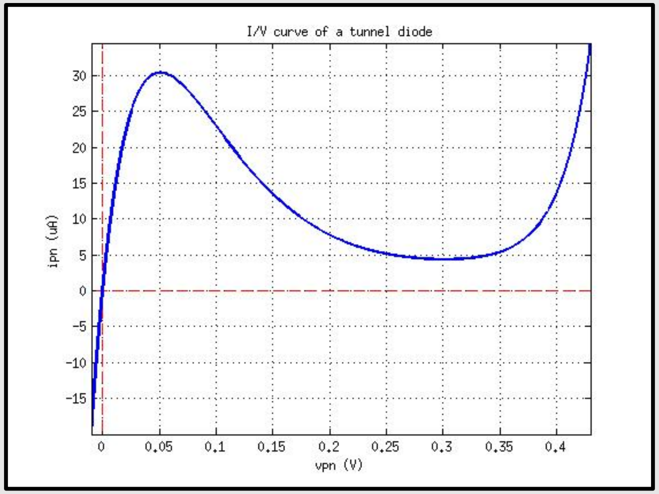
Tunnel diode model in MAPP

1. evaluate model functions:

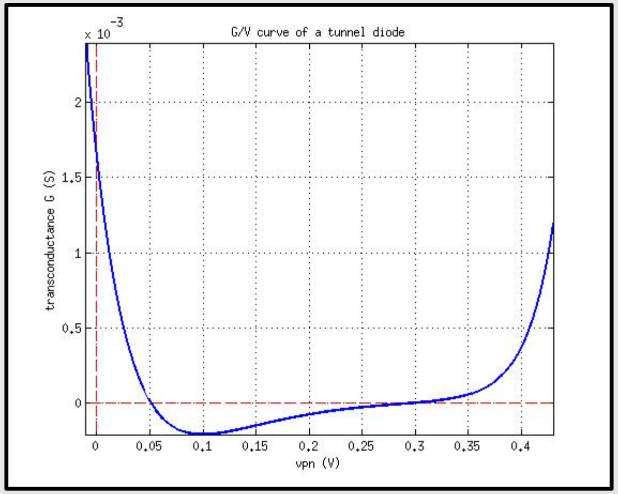
```
MOD = Tunnel_Diode_ModSpec;

% Evaluate ipn at vpn = 0.1V:
S = ee_model_parm2struct(MOD);
S.vpn = 0.1;
ipn = MOD.fe_of_S(S)
```

2. test the model standalone:

```
plotIV_Tunnel_Diode_ModSpec;
```



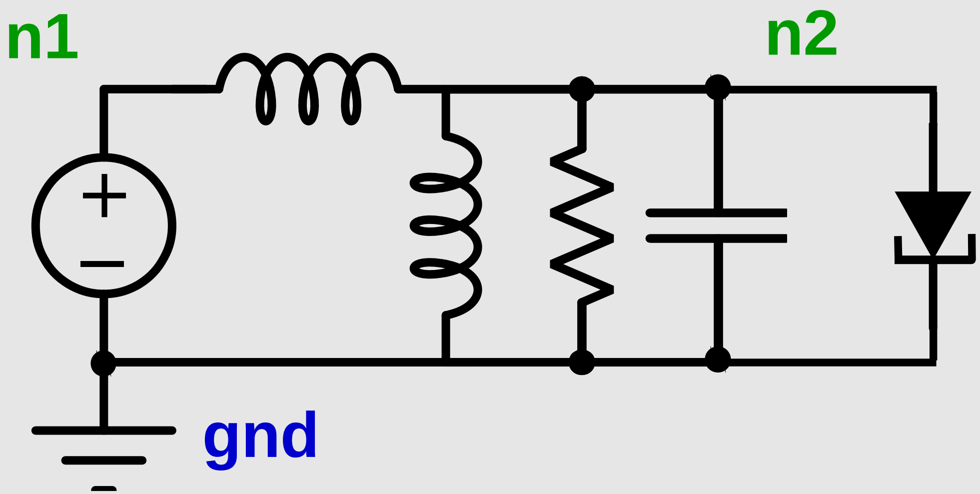

```
plotGV_Tunnel_Diode_ModSpec;
```

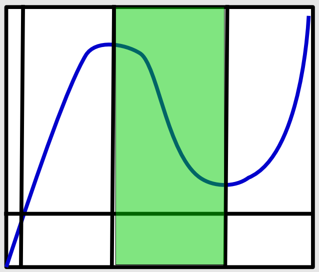


Berkeley Model and Algorithm Prototyping Platform (MAPP)

- ☑ MAPP release website: MAPP.eecs.berkeley.edu
  - ☑ MAPP mailing list;
  - ☑ MAPP bug tracking system;
  - ☑ MAPP git repository.
- ☑ Download and set up MAPP
- ☐ A demo of MAPP
  - ☑ prototype a device model in MAPP;
  - ☐ build a circuit;
  - ☐ run analyses.

Tunnel diode in a circuit





```
clear cktnetlist;
% ckt name
cktnetlist.cktname = 'Tunnel Diode LC oscillator';

% nodes (names)
cktnetlist.nodenames = {'n1', 'n2'};
cktnetlist.groundnodename = 'gnd';

cktnetlist = add_element(cktnetlist, Tunnel_Diode_ModSpec(), ...
             'D1', {'n2', 'gnd'});
cktnetlist = add_element(cktnetlist, resModSpec(), 'R1', {'n2', 'gnd'}, 1e9);
cktnetlist = add_element(cktnetlist, indModSpec(), 'L1', {'n1', 'n2'}, 1e-6);
cktnetlist = add_element(cktnetlist, capModSpec(), 'C1', {'n2', 'gnd'}, 1e-12);

cktnetlist = add_element(cktnetlist, vsrcModSpec(), 'V1',...
             {'n1', 'gnd'}, {}, {{'E', {'dc', 0.2}}});
```

Run analyses

1. convert cktnetlist to DAE:

```
DAE = MNA_EqnEngine(cktnetlist);
```

2. run DC operating point analysis:

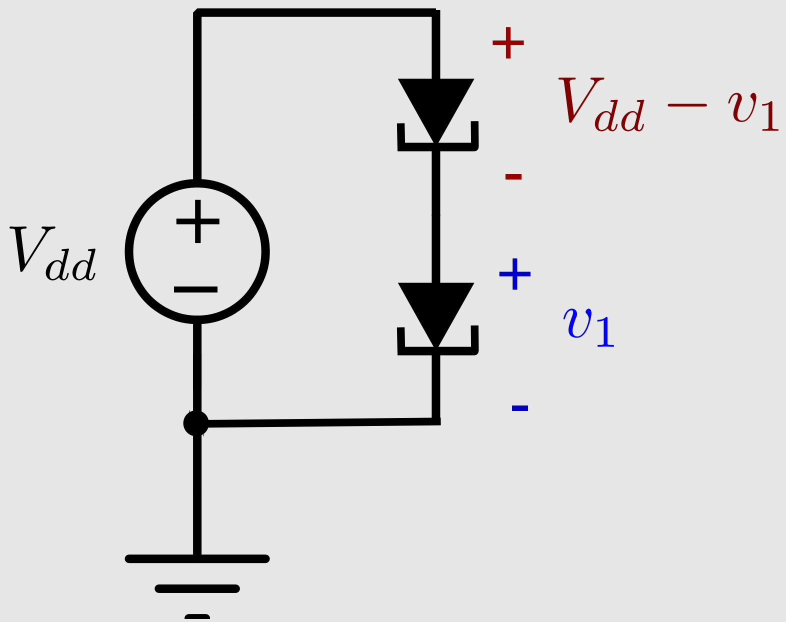
```
dcop = dot_op(DAE);
dcop.print(dcop);
```

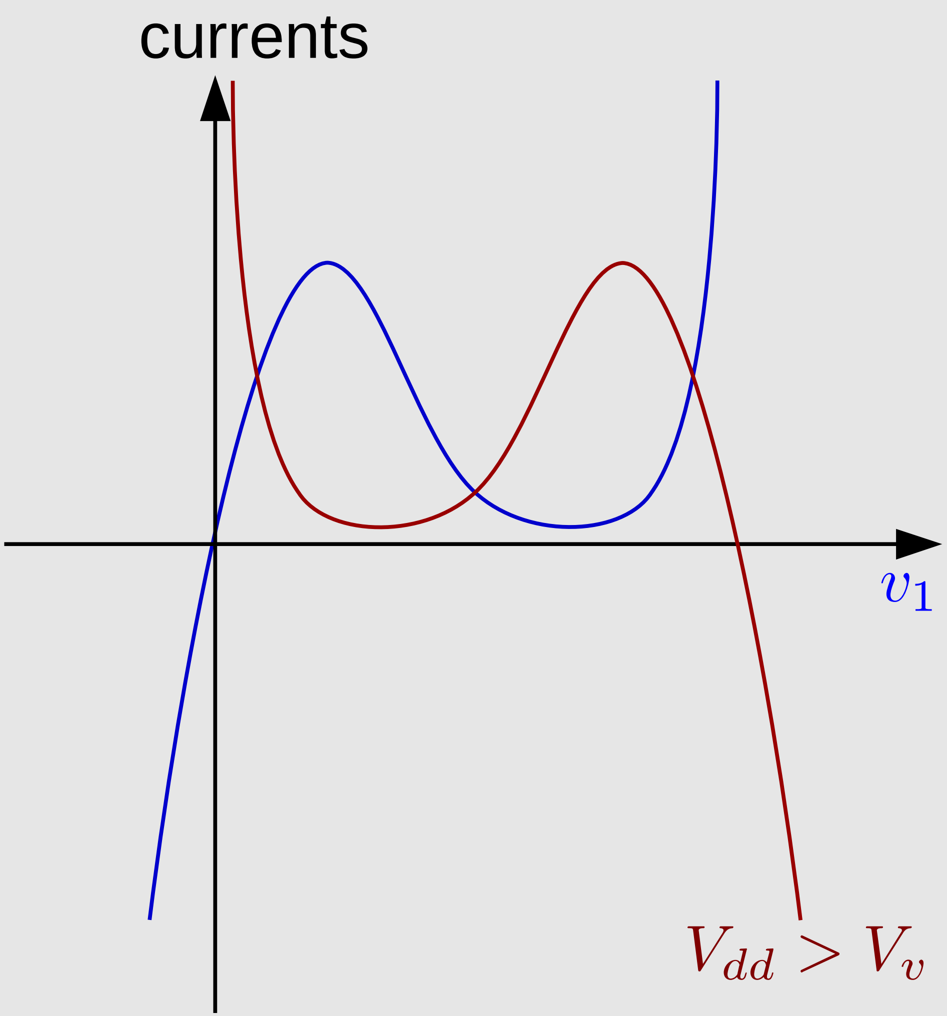
3. run transient simulation:

```
xinit = zeros(DAE.nunks(DAE), 1);
xinit(2) = 0.3;
tstart = 0; tstep = 0.1e-9; tstop = 50e-9;
tranObj = dot_transient(DAE, xinit, tstart, tstep, tstop);
tranObj.plot(tranObj);
```

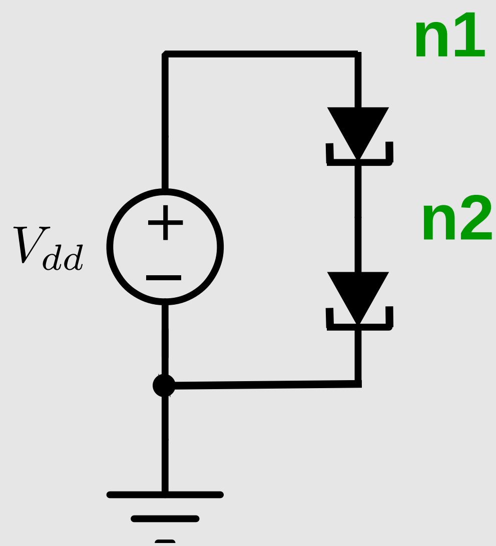
more demos: use Sparse Tableau equation engine
use Harmonic Balance

Goto pair circuit





Goto pair circuit




```
clear cktnetlist;
% ckt name
cktnetlist.cktname = 'Goto pair';

% nodes (names)
cktnetlist.nodenames = {'n1', 'n2'};
cktnetlist.groundnodename = 'gnd';

cktnetlist = add_element(cktnetlist, Tunnel_Diode_ModSpec(), ...
             'D1', {'n1', 'n2'});
cktnetlist = add_element(cktnetlist, Tunnel_Diode_ModSpec(), ...
             'D2', {'n2', 'gnd'});

cktnetlist = add_element(cktnetlist, vsrcModSpec(), 'Vdd',...
             {'n1', 'gnd'}, {}, {{'E', {'dc', 0.5}}});
```

Berkeley Model and Algorithm Prototyping Platform (MAPP)

- ☑ MAPP release website: MAPP.eecs.berkeley.edu
  - ☑ MAPP mailing list;
  - ☑ MAPP bug tracking system;
  - ☑ MAPP git repository.
- ☑ Download and set up MAPP
- ☑ A demo of MAPP
  - ☑ prototype a device model in MAPP;
  - ☑ build a circuit;
  - ☑ run analyses.