# Model implementation into ADS

Rüdiger Follmann

# Contents

→PN-diode, equivalent circuit and equations

→Linear part

→Non-linear part

→Linearized part

→Noise

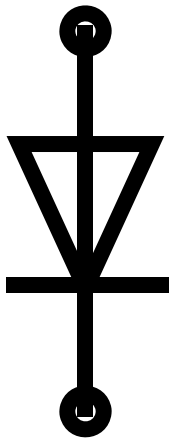
→Interface

→Consistent implementation

# PN diode

Implementation of a pn-diode model into Agilents ADS



# How to start?

A non-linear model consists of the parts

→Equivalent circuit and associated equations

→Linear part containing linear elements

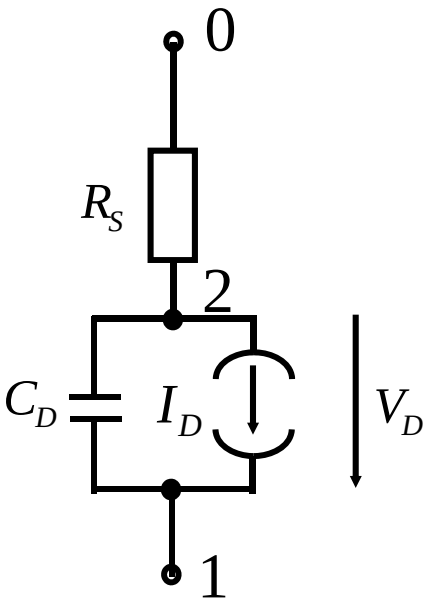
→Same with non-linear part

→Bias point dependent linearized part (Jacobian matrix)

→Bias dependent noise part

# ADS specific things

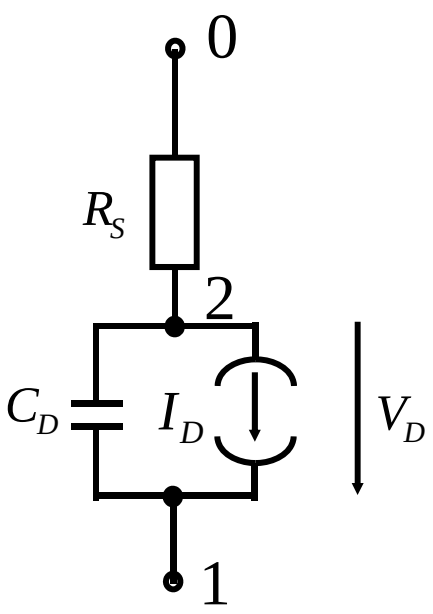



→All nodes are numbered

→Outer nodes get smallest numbers

# Equivalent circuit and equations




$$I_D(V_D) = I_S\left(e^{\frac{V_D}{V_{te}}} - 1\right) + 10^{-12}\,\frac{\mathrm{A}}{\mathrm{V}}V_D$$

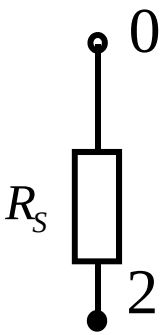
$$\frac{dI_D}{dV_D} = \frac{I_S e^{\frac{V_D}{V_{te}}}}{V_{te}} + 10^{-12}\,\frac{\mathrm{A}}{\mathrm{V}}$$

$$C_D(V_D) = TT\frac{dI_D}{dV_D} + CJ0\left(1 - \frac{V_D}{VJ}\right)^{-M}$$

$$Q_D(V_D) = \int_0^{V_D} C_D(\tilde{V}_D)\,d\tilde{V}_D = TT \cdot I_D + \frac{VJ \cdot CJ0}{1-M}\left(1 - \left(1 - \frac{V_D}{VJ}\right)^{1-M}\right)$$

# Linear part




$R_S$ is the only linear element between nodes 0-2

Admittances here

nodes

Y-matrix

```
COMPLEX y;

 y.real = 1/Rs; y.imag = 0.0;
 status = add_y_branch(userInst, 0, 2,
y);
```

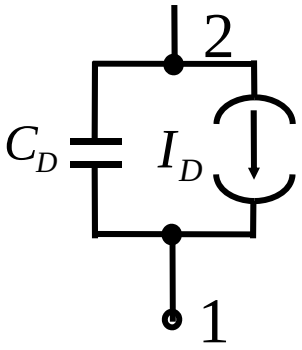
 

# Non-linear part

→Number all nodes in order to get the voltages

→Start with outer nodes

→Use code calculates charges at each node and non-linear currents out of each node




$I_D$ and $C_D$ are non-linear elements between nodes 1-2

# Linearized part

→Calculation of partial derivatives for non-linear elements with respect to the nodes to get capacitances and conductances for Jacobian matrix

→The linearized part does not influence the final harmonic balance solution

→The linearized part influences the convergence speed

# Noise

→ Noise part contains the bias dependent correlation parameters - normalized to $4KT_0$

→ This part is only needed for noise analysis

**Possible noise sources:**

Thermal noise $\left\langle i_{th}^2 \right\rangle = \frac{4kT_0}{R}\Delta f$

Shot noise $\left\langle i_{shot}^2 \right\rangle = 2qI_D\Delta f$

1/f noise $\left\langle i_{1/f}^2 \right\rangle = KF\frac{I_D^{AF}}{f^{FFE}}\Delta f$

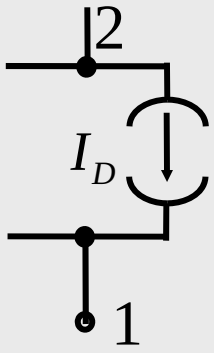
Burst or popcorn noise

$$\left\langle i_{burst}^2 \right\rangle = KB\frac{I_D^{CF}}{1+\left(\frac{f}{f_{CF}}\right)^2}\Delta f$$

# Non-linear part

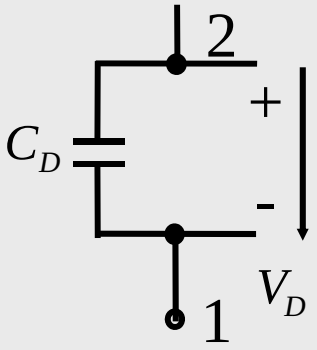
## Calculation of node currents and charges




Node 1: $i_1 = -I_D$

Node 2: $i_2 = +I_D$

Current from node:+, to node:-




Node 1: $q_1 = -Q_D$

Node 2: $q_2 = +Q_D$

Node i q

```
status =   add_nl_iq(userInst, 1, -id, -qd) &&
           add_nl_iq(userInst, 2, +id, +qd);
```

# Jacobian matrix

$$\frac{\partial I_1}{\partial V_1} = -1 \cdot \left( -\frac{dI_D}{dV_D} \right) = \frac{dI_D}{dV_D} \qquad \frac{\partial I_1}{\partial V_2} = -\frac{dI_D}{dV_D}$$

$$\frac{\partial I_2}{\partial V_1} = -\frac{dI_D}{dV_D} \qquad \frac{\partial I_2}{\partial V_2} = \frac{dI_D}{dV_D}$$

Real part

$$\frac{\partial Q_1}{\partial V_1} = C_D \qquad \frac{\partial Q_1}{\partial V_2} = -C_D$$

$$\frac{\partial Q_2}{\partial V_1} = -C_D \qquad \frac{\partial Q_2}{\partial V_2} = C_D$$

Imaginary part

partial derivatives g c + -

$V_D = V_2 - V_1$

```
status =    add_nl_gc(userInst, 1, 1, +gd, +cd) &&
            add_nl_gc(userInst, 1, 2, -gd, -cd) &&
            add_nl_gc(userInst, 2, 1, -gd, -cd) &&
            add_nl_gc(userInst, 2, 2, +gd, +cd);
```

# Noise

```
COMPLEX thermal, dNoise;

thermal.imag = dNoise.imag = 0.0;

thermal.real = 1.0/Rs*DEV_TEMP/NOISE_REF_TEMP;

dNoise.real = 2 * CHARGE * id;

dNoise.real = kf * pow(id, AF)*pow(omega/TWOPI, -FFE);

dNoise.real /= FOUR_K_T0;

status =      add_n_branch(userInst, 0, 2, thermal) &&
              add_n_branch(userInst, 1, 2, dNoise);
```

Thermal noise

Shot noise

1/f noise

**Attention: All noise currents are normalized to $4KT_0$**

# Interface

nonlin_ele.h

```
// name      numPars     params     compute_y     post_analysis      senior Info
//      numExtNodes     pre_analysis     compute_n     devDef      tranDef

 {"PNDIODE",  0,  size(PNDIODE),  PNDIODE,  NULL,  NULL,  NULL, NULL,
NULL, NULL, NULL, NULL}
```

nonlin_def.h

```
static UserNonLinDef PNDIODE =
{
  0,                /* numIntNodes   */
  NULL,             /* analyze_lin() */
  sdiode_nl,        /* analyze_nl()  */
  sdiode_ac,        /* analyze_ac()  */
  NULL,             /* modelDef      */
  NULL              /* analyze_ac_n  */
};
```

# Interface

nonlin_typ.h

```
PNDIODE[] =
{ /* P-N junction diode */
        {"AREA",     REAL_data},
        {"IS",       REAL_data},
        {"RS",       REAL_data},
        {"N",        REAL_data},
        {"TT",       REAL_data},
        ...
}
```

nonlin_fun.h

```
* NON-LINEAR PN-DIODE MODEL */
EXTERN_FUNCTION( extern boolean  pndiode_lin, (UserInstDef *userInst, double omega));
EXTERN_FUNCTION( extern boolean  pndiode_nl,  (UserInstDef *userInst, double *vPin));
EXTERN_FUNCTION( extern boolean  pndiode_ac,  (UserInstDef *userInst, double *vPin, double omega));
EXTERN_FUNCTION( extern boolean  pndiode_ac_n,(UserInstDef *userInst, double *vPin, double omega));
```

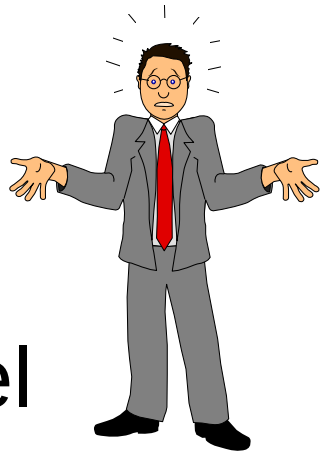
 

# Interface

→AEL programming language is the GUI for the simulator

# Summary



→Different parts of a non-linear model

→Nonlinear equations and their derivatives

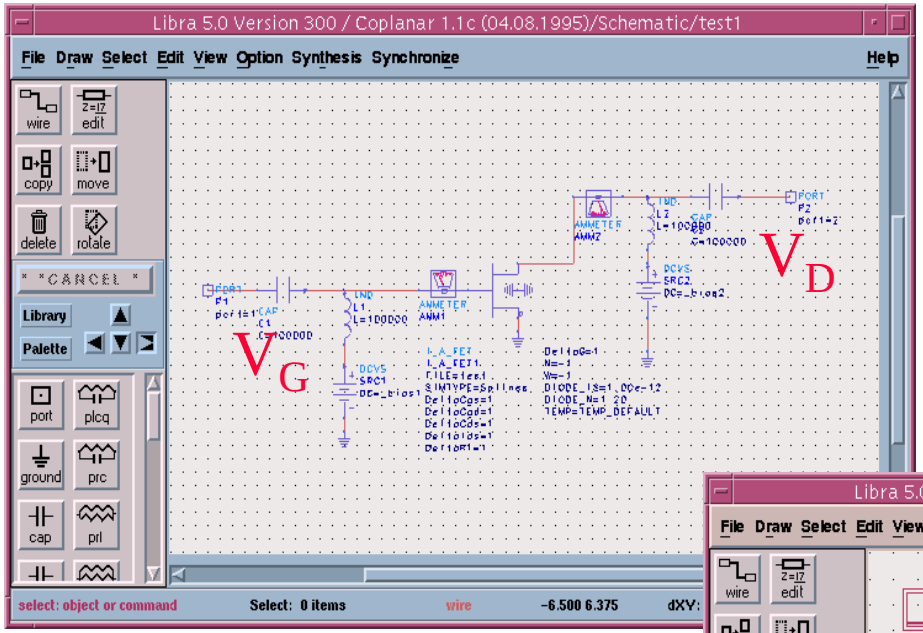
→Implementation into circuit simulation software

→Noise sources

→Jacobian matrix

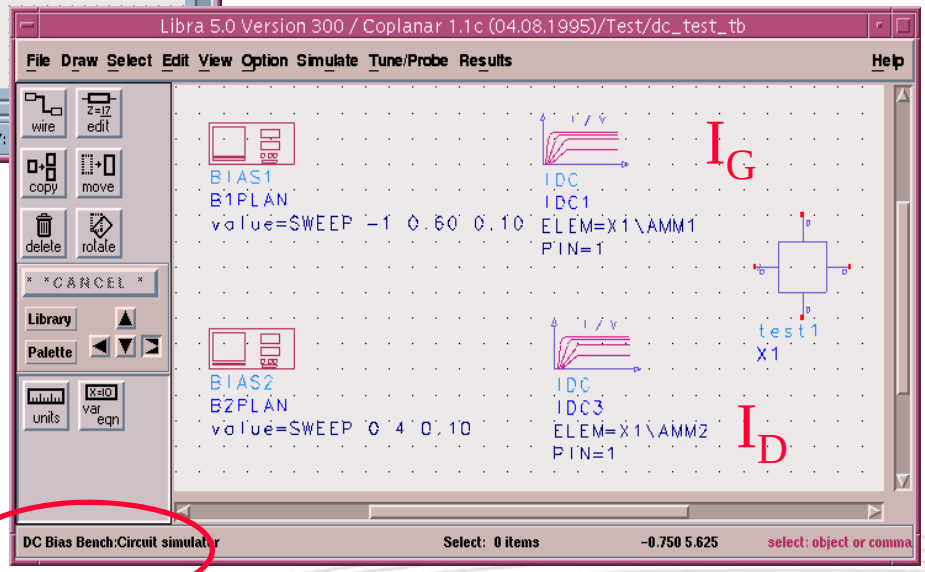
 

# DC tests




All tests are shown for a transistor device. The are valid for diode devices as well.

Perform a DC analysis first!



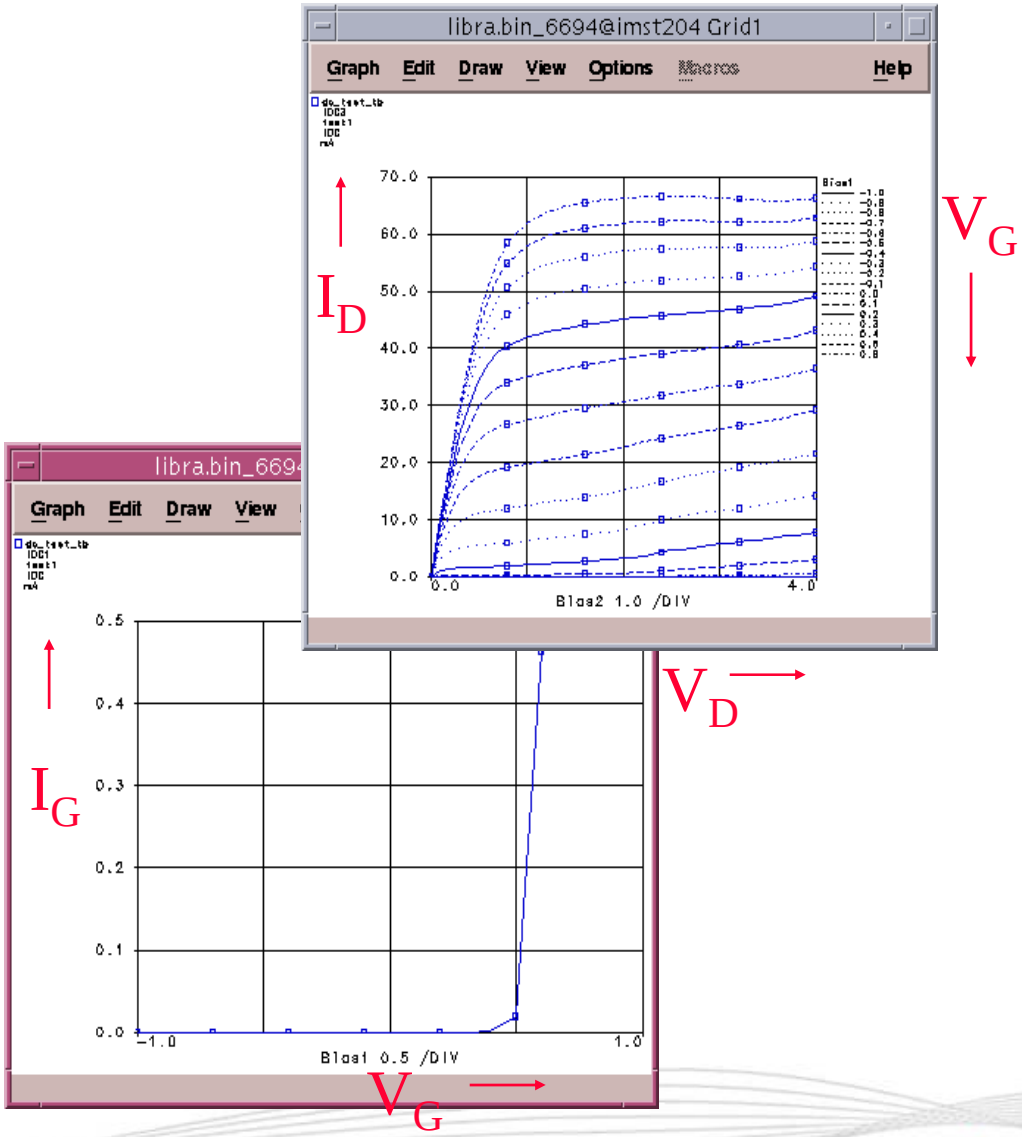

Only currents are calculated

DC testbench

# DC tests

- → IV output curves OK?
- → IV input curves OK?
- → Curves typical for element?
- → Show measurements similar behaviour?

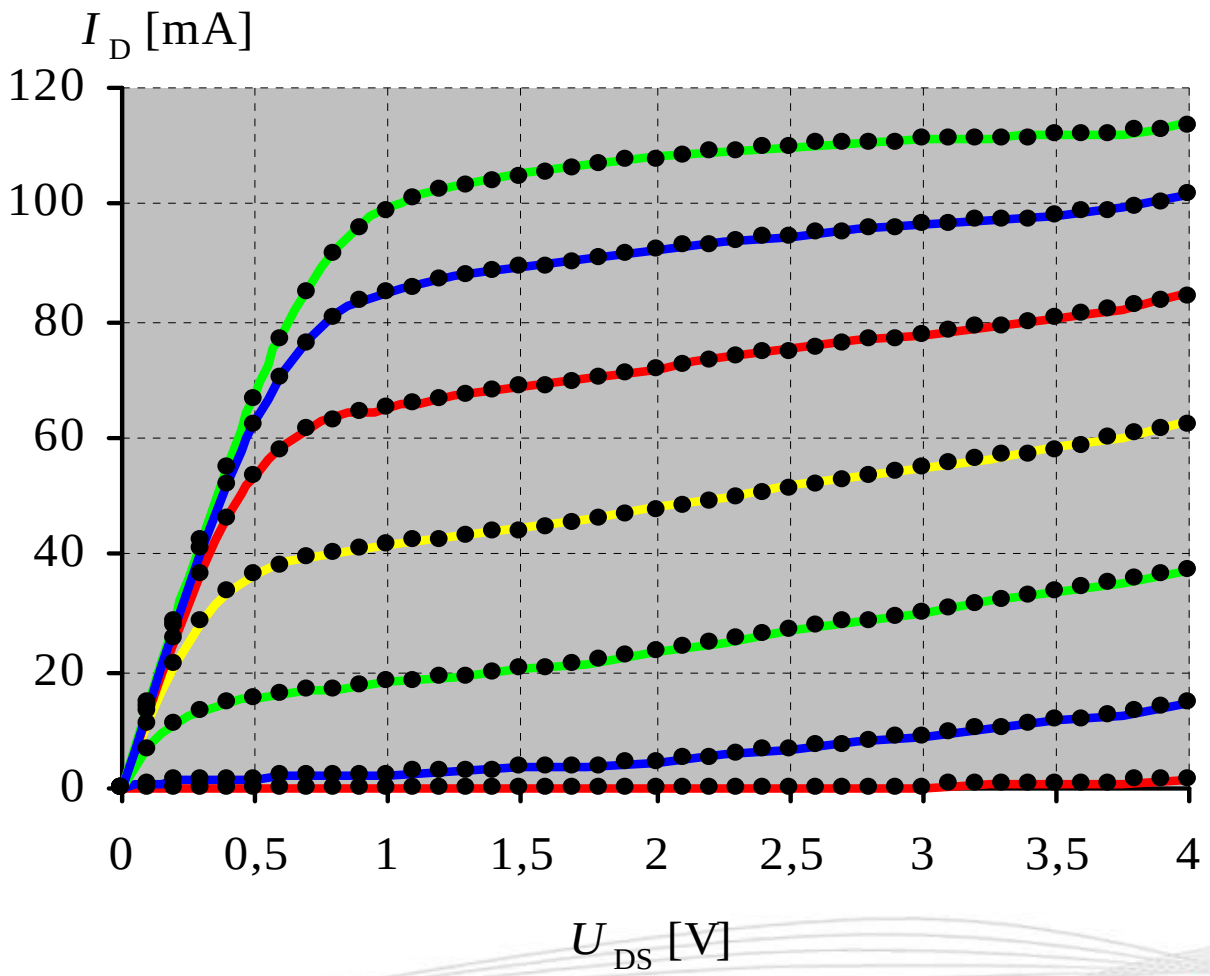


# DC tests

T4x50µm HEMT, simulation vs. measurements

Show simulation and measurements similar behaviour?

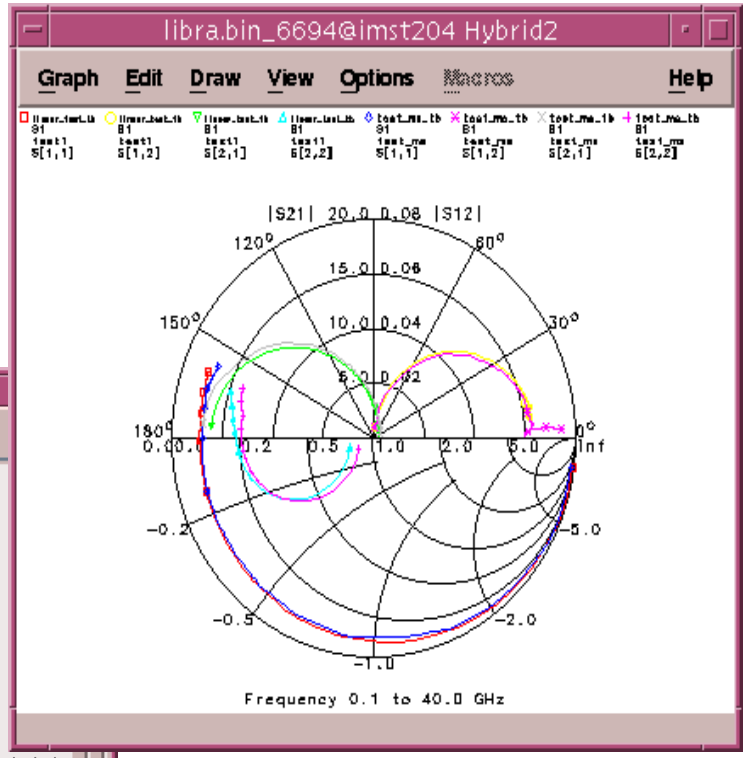


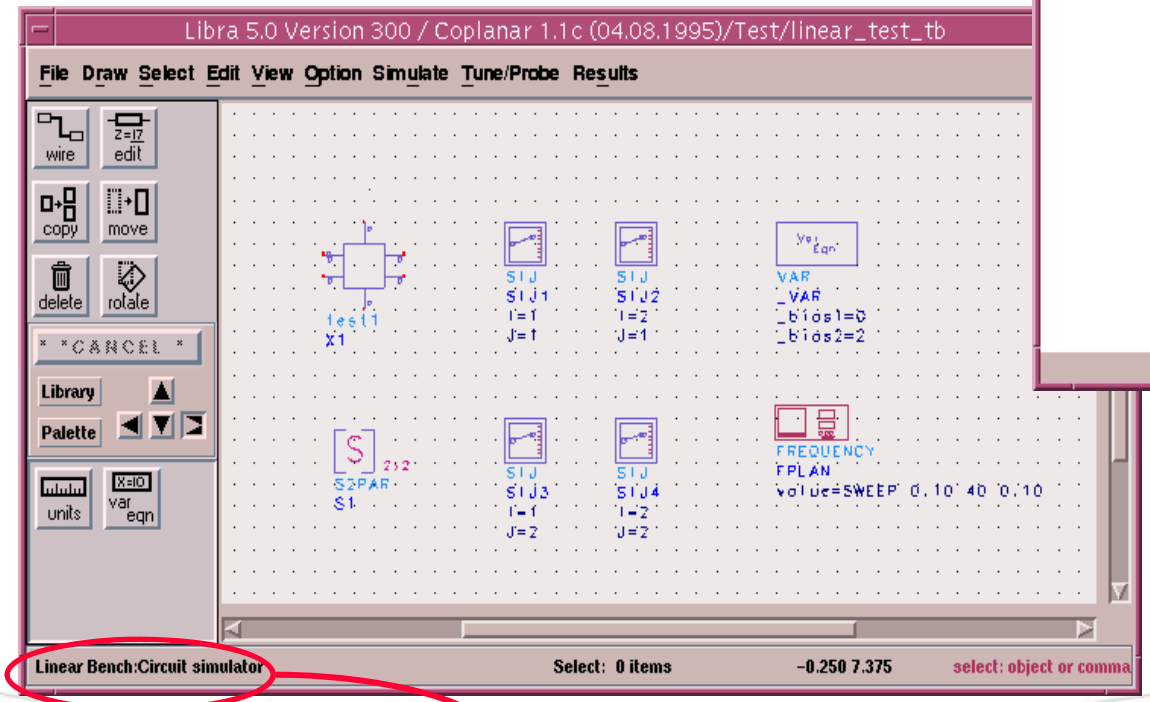

 

# RF Tests

Are measured and simulated S-parameters similar ?

Transistor: Gain, pinch-off S-parameters OK?



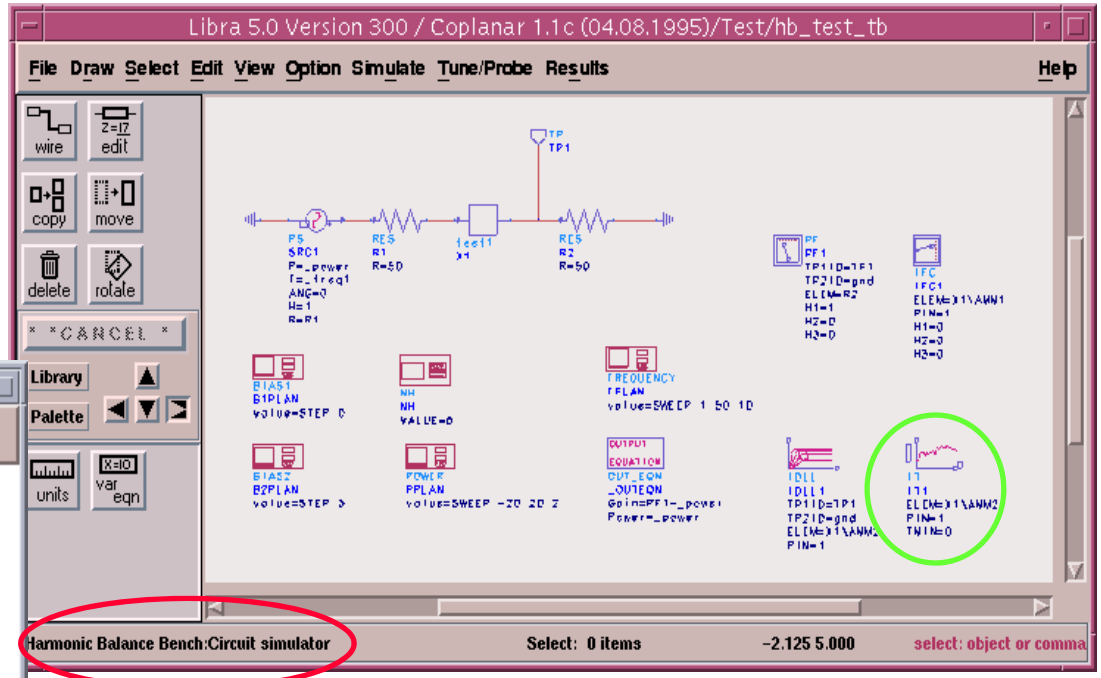

Diode: Forward and reverse S-parameters OK?

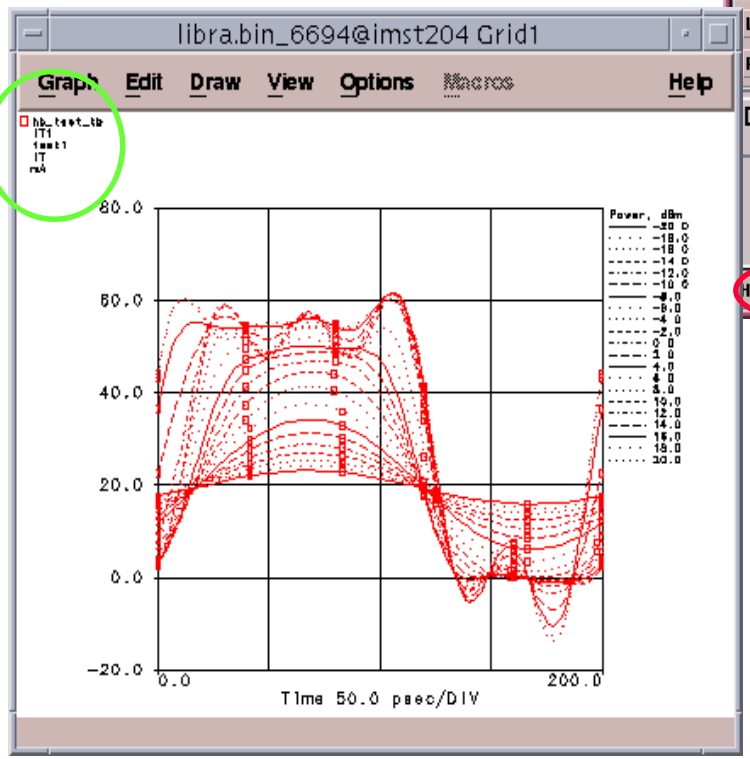
Lineares testbench

# HB tests

Simulate a power sweep



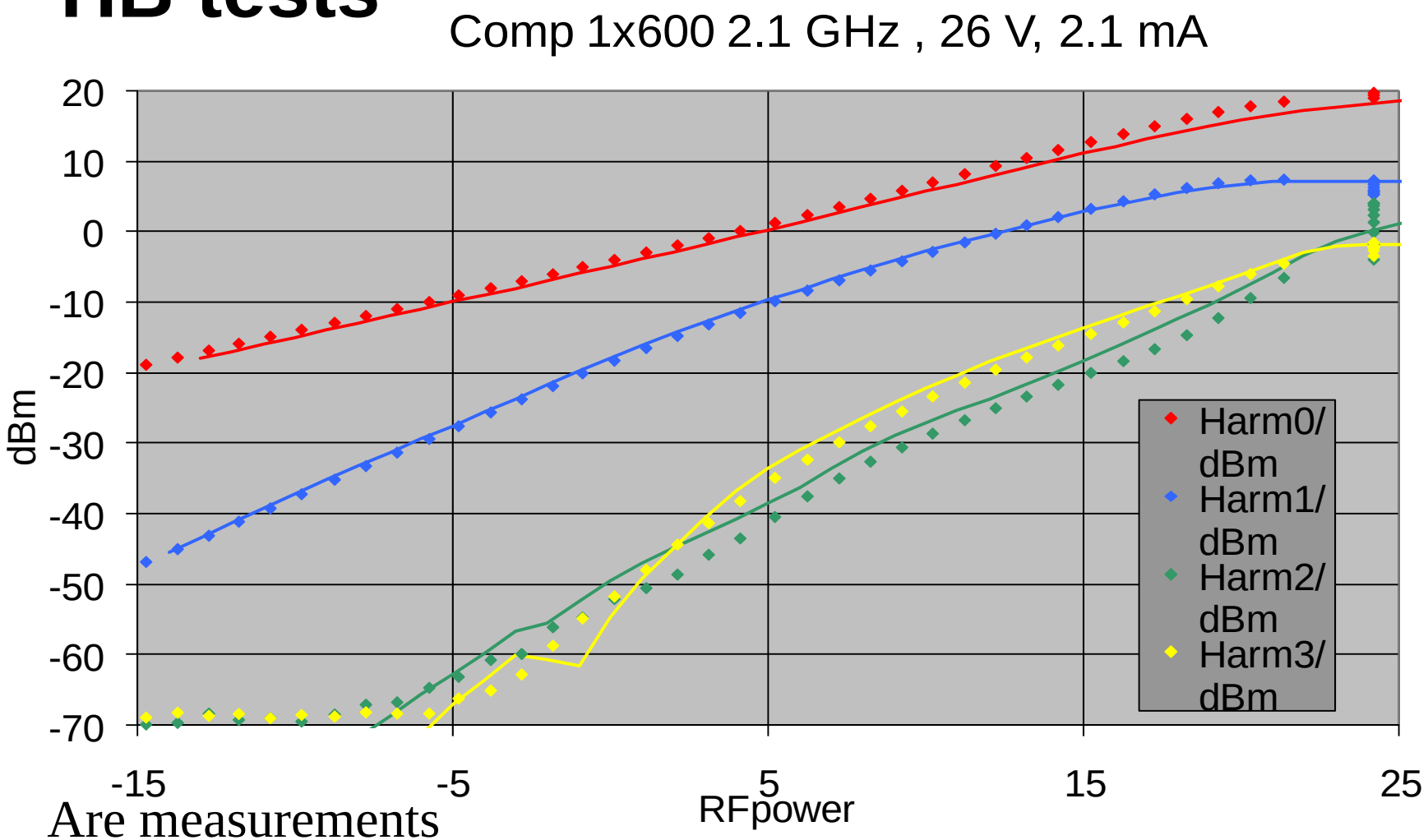




Harmonic Balance testbech

Is the behaviour as expected?
Are the nonlinearities dominant for high input power

# HB tests

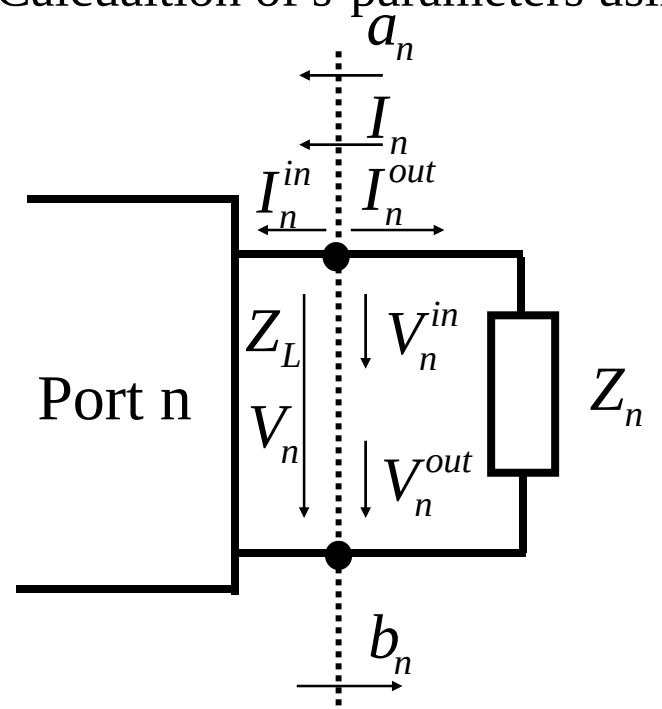



Are measurements and simulations similar ?

How good is the quality of the model ?

# Consistent implementation

Calcualtion of s-parameters using voltage and current definition




## Waves in und out of port n

$$a_n = \frac{V_n^{in}}{\sqrt{Z_{Ln}}} = I_n^{in}\sqrt{Z_{Ln}}$$

$$b_n = \frac{V_n^{out}}{\sqrt{Z_{Ln}}} = I_n^{out}\sqrt{Z_{Ln}}$$

## Currents and voltages

$$u_n = \frac{V_n}{\sqrt{Z_{Ln}}} = \frac{V_n^{in} + V_n^{out}}{\sqrt{Z_{Ln}}} = a_n + b_n$$

$$i_n = I_n\sqrt{Z_{Ln}} = \left(I_n^{in} - I_n^{out}\right)\sqrt{Z_{Ln}} = a_n - b_n$$

$$\Rightarrow \quad a_n = \frac{1}{2}(u_n + i_n), \quad b_n = \frac{1}{2}(u_n - i_n) \quad \Rightarrow \quad \|b\| = \|s\| \cdot \|a\|$$

## S matrix

## Example

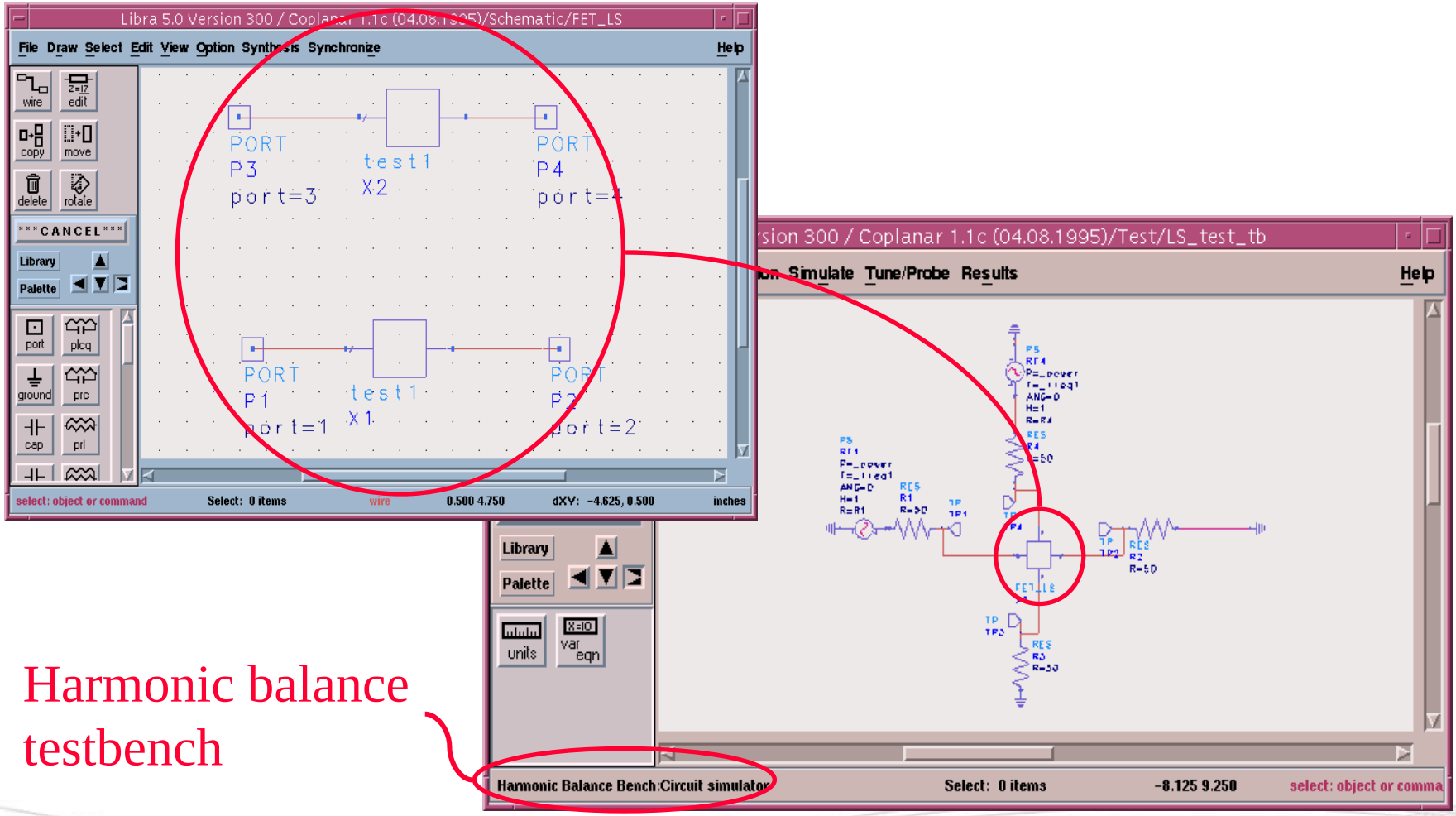
$$s_{11} = \left.\frac{b_1}{a_1}\right|_{a_{2=0}}$$

$$s_{11,dB} = 20 \cdot \log\frac{u_1 - i_1}{u_1 + i_1}$$

# Consistent implementation

Calculation of small signal S-parameters using a large signal test bench

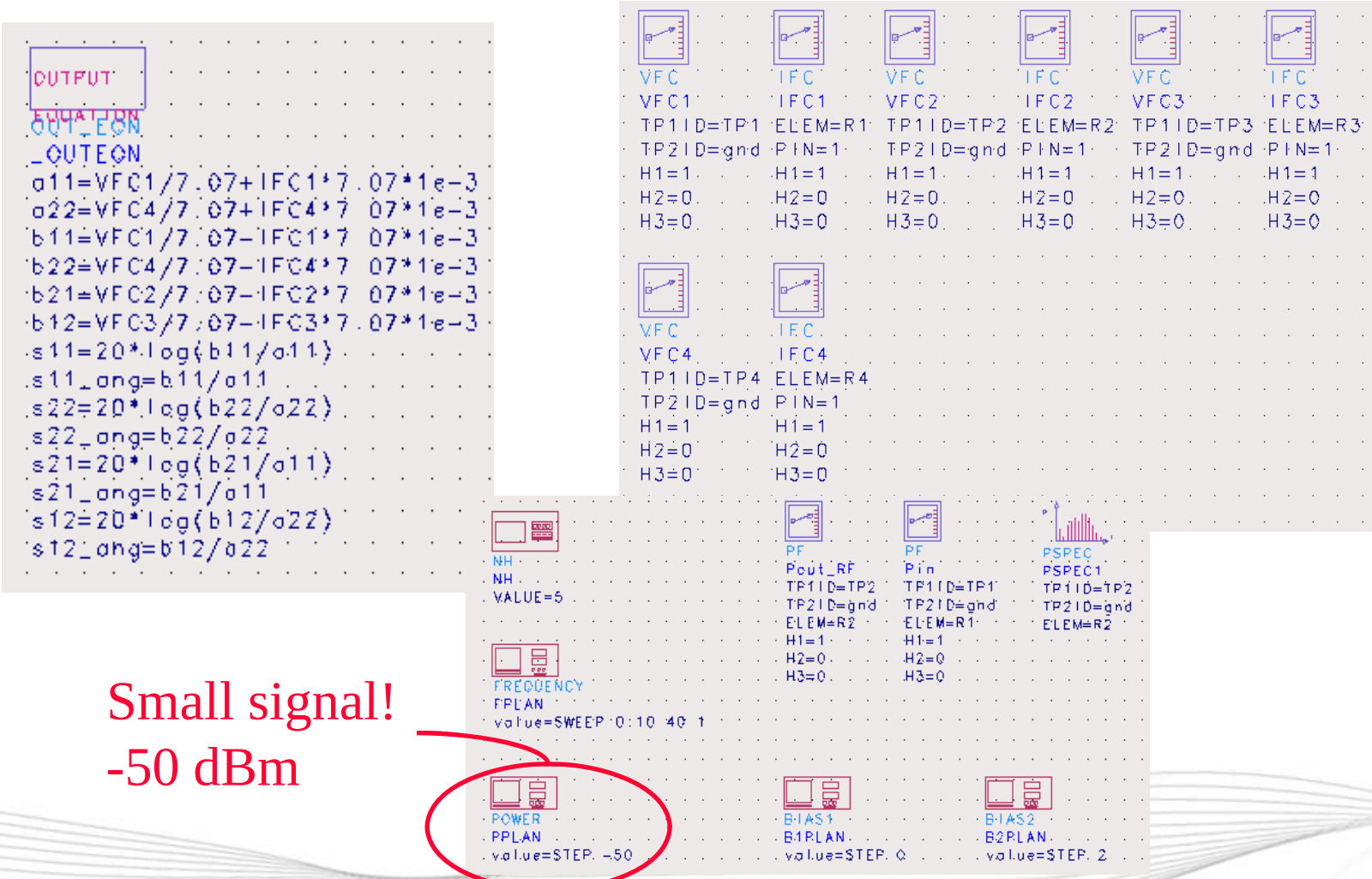



Harmonic balance testbench

# Consistent implementation

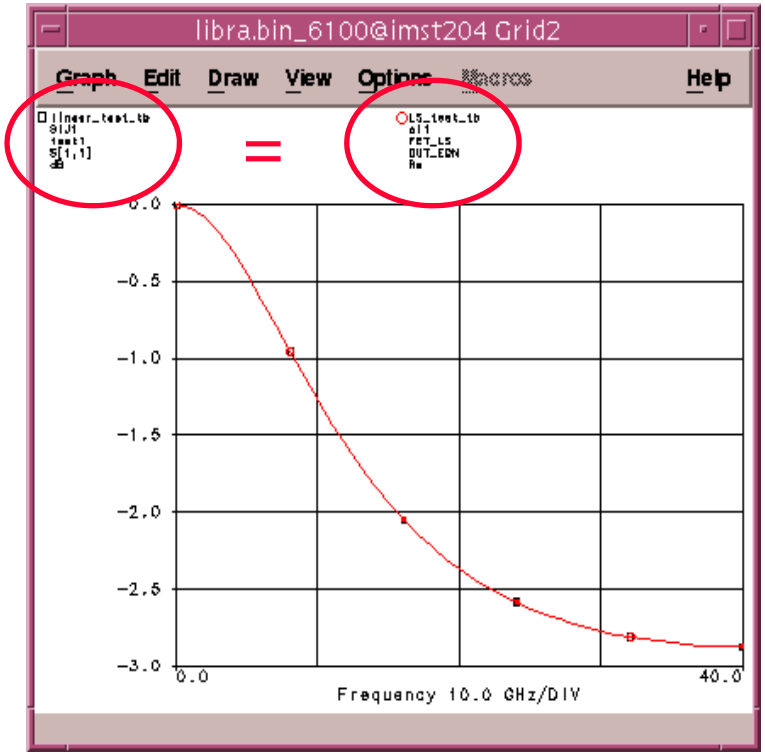
Calculation of small signal s-parameters using a large signal testbench




Small signal!
-50 dBm

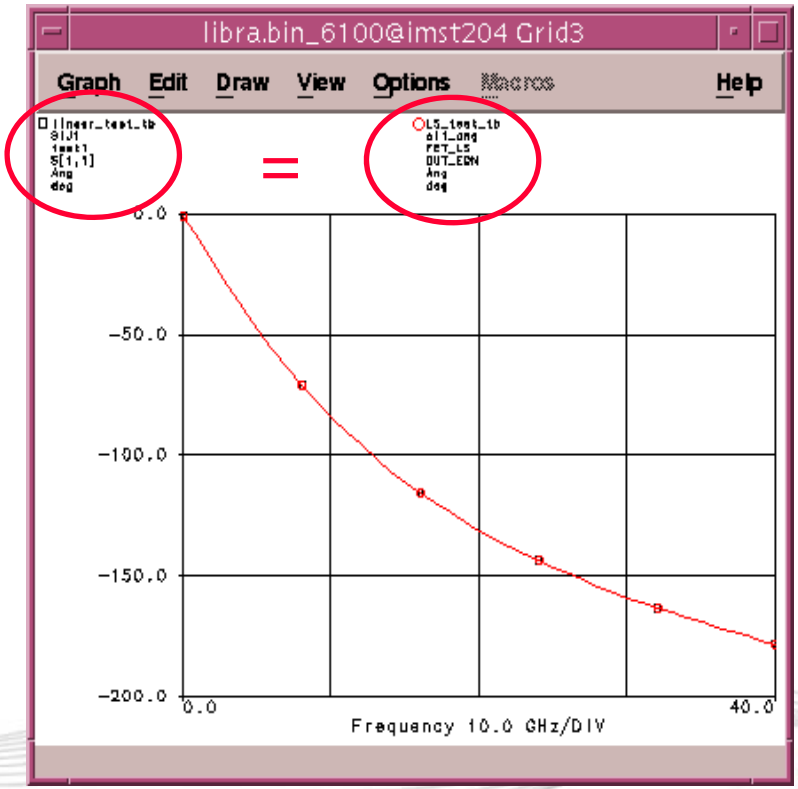
 

# Consistent implementation




Mag($s_{11}$), simulated using Small signal and large signal testbench for very low input power (-50 dBm)

Phase($s_{11}$), simulated using Small signal and large signal testbench for very low input power (-50 dBm)

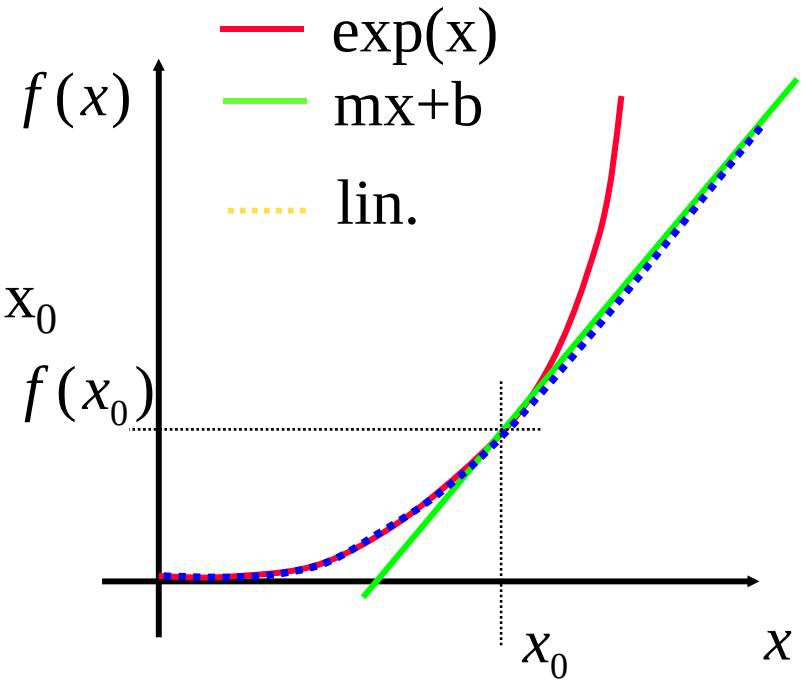


# Needful things

Linearize exp() function for better conversion and avoid errors such as numerical overflows

$$f(x)=\begin{cases}\exp(x) & x \le x_0 \\ mx+b & x > x_0\end{cases}$$




Function must be continuous at $x_0$

$$f_1(x_0) = f_2(x_0) \qquad \left.\frac{df_1}{dx}\right|_{x_0} = \left.\frac{df_2}{dx}\right|_{x_0}$$

$$\Rightarrow \quad \exp(x_0) = mx_0 + b$$

$$m = \exp(x_0)$$

$$\Rightarrow \quad b = \exp(x_0)(1 - x_0) \Rightarrow f_2(x) = \exp(x_0)(x + 1 - x_0)$$

# Summary

→Perform different tests (DC, RF, harmonic balance)

→Typical results for a FET

→Calcualtion of small signal s-parameters using a large signal testbench

→Linearization of functions