# 5.5.: CALCULATING S-PARAMETERS FROM COMPLEX VOLTAGES

## or: HOW TO OBTAIN S-PARAMETERS OUT OF SPICE OUTPUTS

IC_CAP file: s_params_from_volt_basics.mdl
calc_Spar_from_volt.mdl

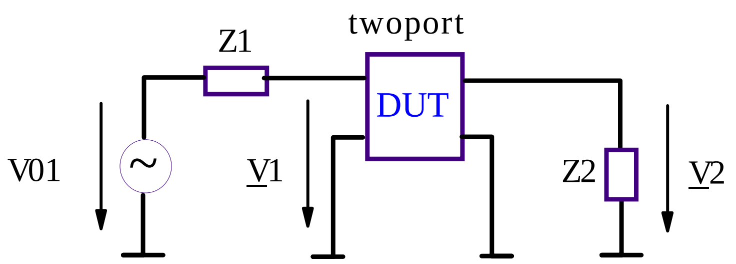
Ass a fact, most SPICE-like, time-based simulators cannot simulate S-parameters. But they can simulate the frequency behavior of circuits in magnitude and phase (or real and imaginary numbers). And IC-CAP can calculate the S-parameters out of these numbers.

How this is done is explained below:

Note: two subsequent simulations are needed in order to obtain one set of S-parameters.

1st simulation:

Z1 = Z2 = Z0
e.g.Z0 = 50 Ohm

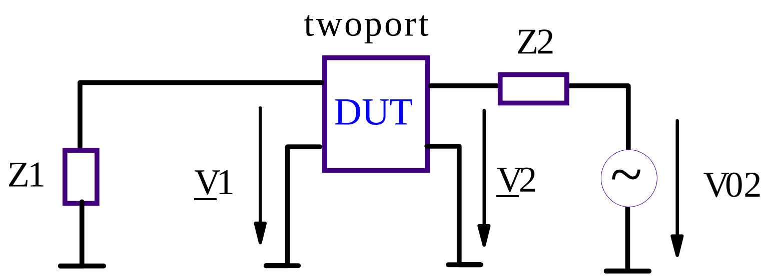



$$\underline{S}_{11} = 2 * \frac{\underline{V}_1}{\underline{V}_{01}} - 1 \qquad (1)$$

$$\underline{S}_{21} = 2 * \frac{\underline{V}_2}{\underline{V}_{01}} \qquad (2)$$

2nd simulation:

Z1 = Z2 = Z0
e.g.Z0 = 50 Ohm




$$\underline{S}_{12} = 2 * \frac{\underline{V}_1}{\underline{V}_{02}} \qquad (3)$$

$$\underline{S}_{22} = 2 * \frac{\underline{V}_2}{\underline{V}_{02}} - 1 \qquad (4)$$

This is the reason why IC-CAP has to insert the user-defined circuit (2-port DUT) as a subcircuit into a hyper-circuit and to finally send this total circuit to the simulator. In this manner, the simulator output deck will provide the complex voltages that are required by IC-CAP to calculate the S-parameters.

To speed-up simulations, IC-CAP puts these two schematics together into one big circuit. This 'big' circuit is depicted below in fig.3 using SPICE syntax. Its structure is valid for all other simulator circuit descriptions in IC-CAP, only the syntax may be different. For HP's

MDS/ADS, however, this cmplex calculations are not required, because both simulators support S-parameter simulations directly.

**T**he example shows a S-parameter simulation for a bipolar transistor with inputs VB,VE,VC,VS and frequency. For simplicity, the bias is kept constant and only the frequency is swept.

see IC-CAP file spar_from_volt.mdl

THE USER_DEFINED CIRCUIT DESCRIPTION:

```
Q1 1=C 2=B 3=E 4=S NPN
.MODEL NPN NPN
+ IS  = 2.704E-16
+ BF  = 86.16
+ NF  = 0.979
+ VAF = 86.95
```

Fig.1: The user-defined circuit for the example below (a simple bipolar transistor)
NOTE: no subcircuit description used for simplification.

THE USER_DEFINED SETUP DESCRIPTION LOOKS LIKE THIS:

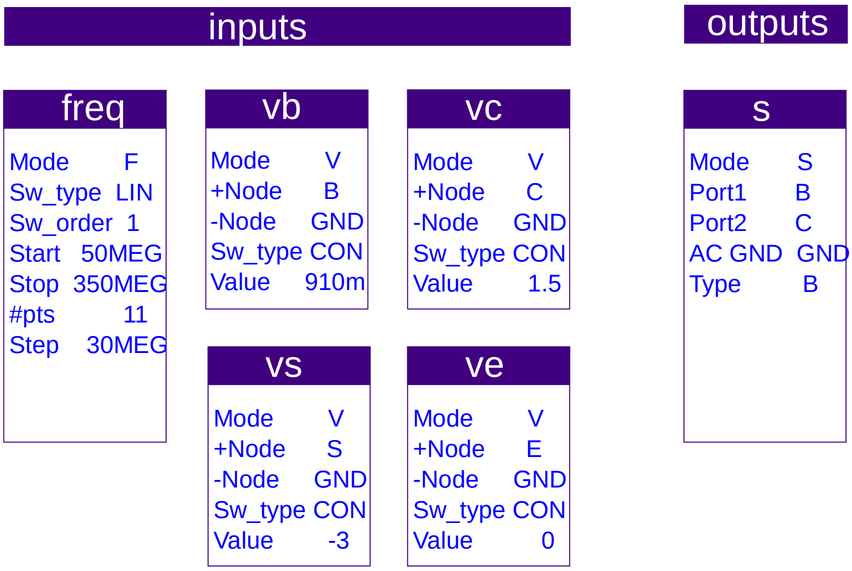



Fig.2: User-defined IC-CAP setup, from which IC-CAP generates the corresponding SPICE deck

Notes on the names of the circuit elements of the following pages:

The user-defined 'Circuit' description consists simply -as shown above- of a bipolar transistor with 4 connections E,B,C and S. Therefore nodes 1 .. 4 (or 1 .. n in a more general case) as well as 5 ... 8 (or n+1 .. 2n) of the total circuit generated by IC-CAP are used to link the user-defined circuit into the 'overhead circuit'.

Nodes 9 .. 11 (or generally 2n+1 .. 2n+3) as well as 12 .. 14 (or 2n+4 .. 2n+6) of the IC-CAP circuit are used to calculate the voltages required for equ. (1) - (4)
(or in other words to 'emulate' the 'S-parameter test set' of the network analyzer).

The simulator node names as defined by the user in the input fields determine the effective name of the elements in the IC-CAP 'main' circuit, e.g. VCGRO: a voltage from user-defined node C to GROund or LBGRO: an inductor L from user-defined node B to GROund.

The values of LxGROx and CxGROx are defined by the IC-CAP system variables TWOPORT_L and TWOPORT_C, all resistors in TPxCKT that have 50Ohms can be set to a different value when defining the IC-CAP system variale TWOPORT_Z0.

node numbers valid for data/bjt_npn.mdl file

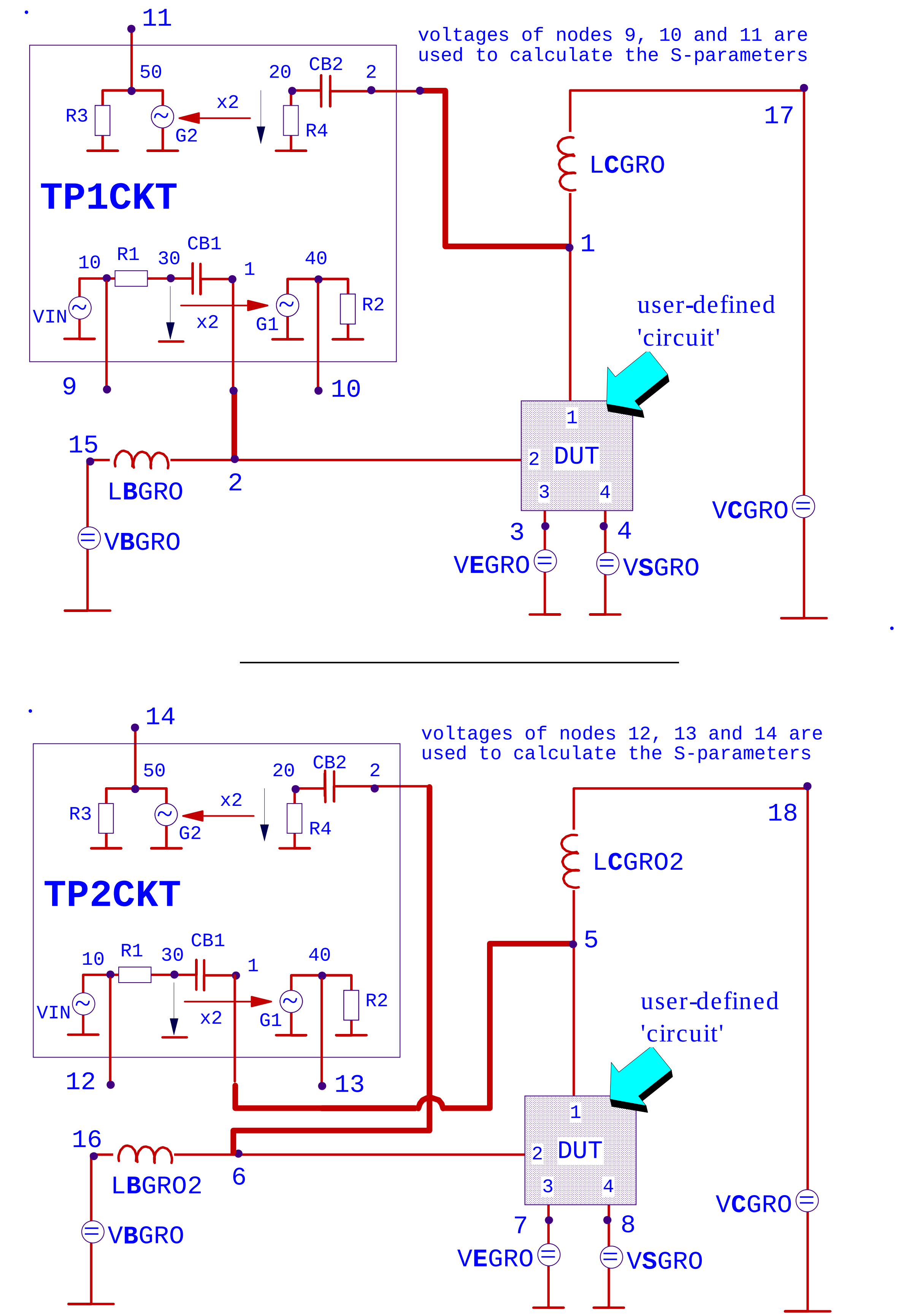



Fig.3: Spice circuit generated by IC-CAP in order to be able to calculate S-parameters out of complex voltages.

## The corresponding UCB SPICE input file generated by IC-CAP and sent to SPICE:

only this first line is the user-defined circuit description!
all the rest has been added by IC-CAP

```
.MODEL npn NPN  IS=2.704E-16  BF=86.16  NF=0.979  VAF=86.95

QCKT  1 2 3 4 npn

QCKT2 5 6 7 8 npn

* START SOURCES

.SUBCKT TP1CKT 1 2 10 40 50
CB1 1 30 1e-05
R1 30 10 50
R2 0 40 1
R3 0 50 1
CB2 2 20 1e-05
R4 20 0 50
G1 0 40 30 0 2
G2 0 50 20 0 2
VIN 10 0 AC 1
.ENDS

.SUBCKT TP2CKT 1 2 10 40 50
CB1 30 2 1e-05
R1 10 30 50
R2 40 0 1
R3 50 0 1
R4 20 0 50
CB2 1 20 1e-05
G1 0 40 30 0 2
G2 0 50 20 0 2
VIN 10 0 AC 1
.ENDS

XTP1CKT 2 1 9 10 11 TP1CKT
XTP2CKT 6 5 12 13 14 TP2CKT
LBGRO 2 15 0.0001
VBGRO 15 0 DC 0.91
LBGRO2 6 16 0.0001
VBGRO2 16 0 DC 0.91
LCGRO 1 17 0.0001
VCGRO 17 0 DC 1.5
LCGRO2 5 18 0.0001
VCGRO2 18 0 DC 1.5
VEGRO 3 0 DC 0
VEGRO2 7 0 DC 0
VSGRO 4 0 DC -3
VSGRO2 8 0 DC -3
* END SOURCES

.AC LIN 11 5e+07 3.5e+08
.PRINT AC V(10) V(9) V(11) V(14) V(13) V(12)
.END
```