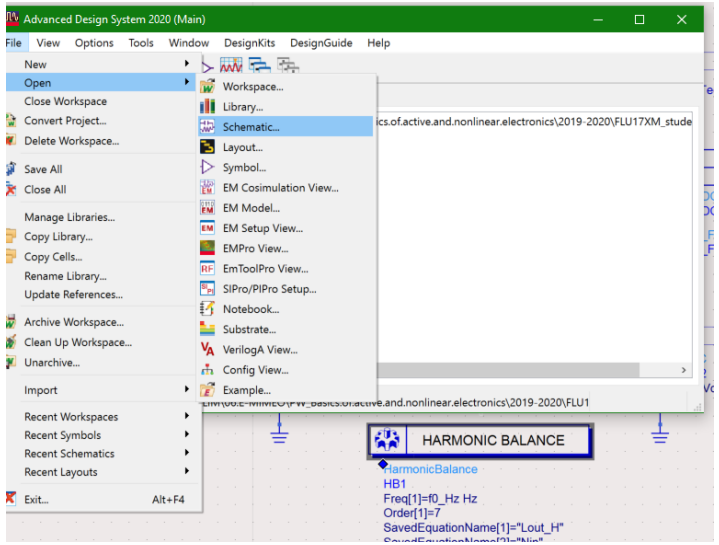
PRACTICAL WORK
PW4:
SIMULATION OF A LINEAR AMPLIFIER @ 2GHZ USING KEYSIGHT ADS AND BASED ON LUMPED COMPONENTS

- **Student Help Manual for PW 3, 4**

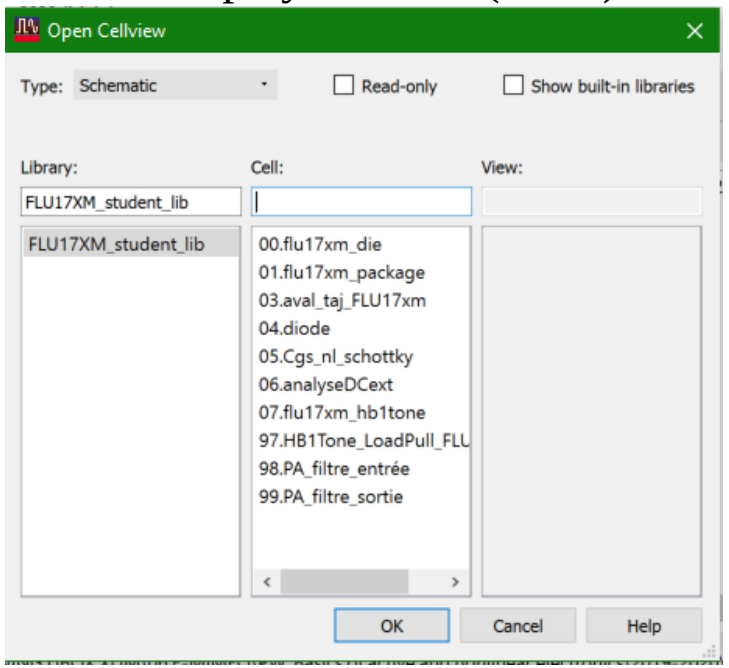
Copy to be left on site

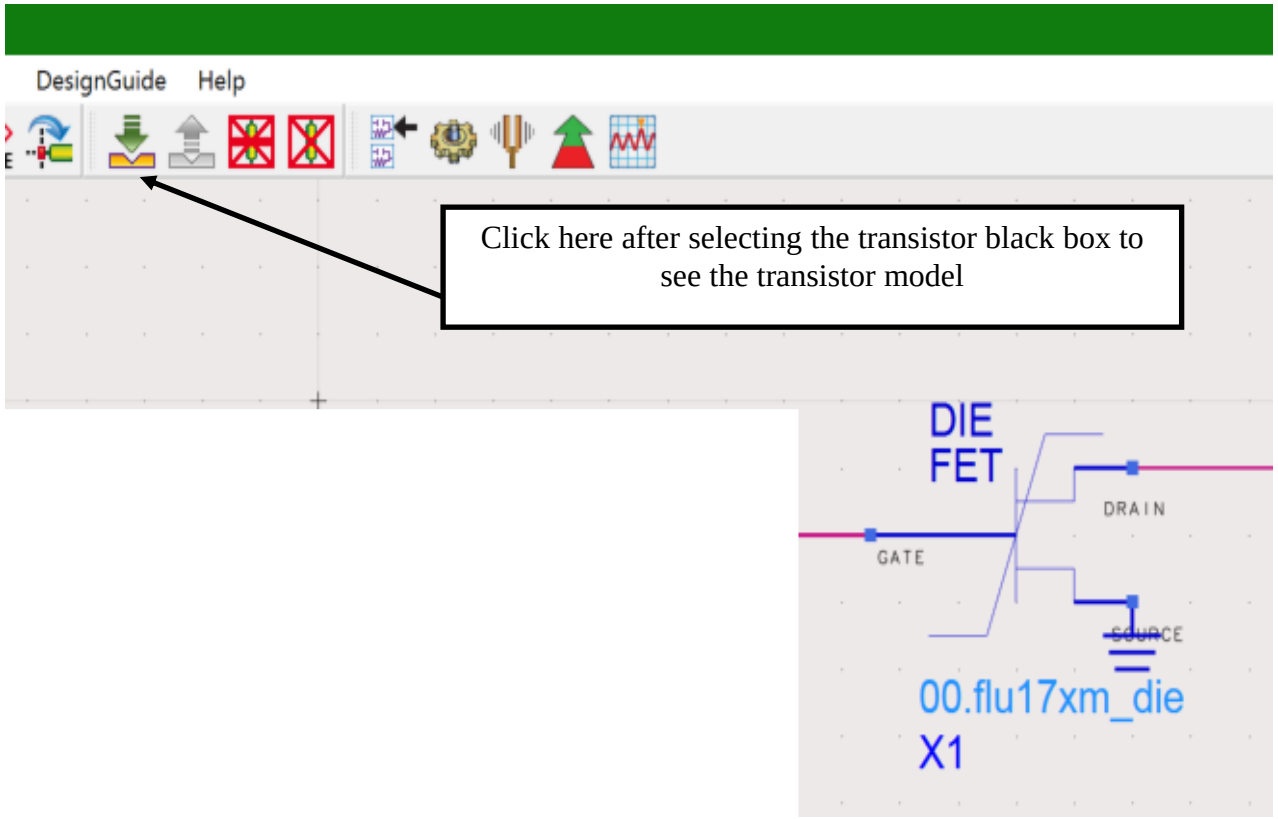
This booklet presents the architecture of the ADS project you will use.

After launching ADS, click on ***file/open Schematic*** to obtain the following list of schematics :

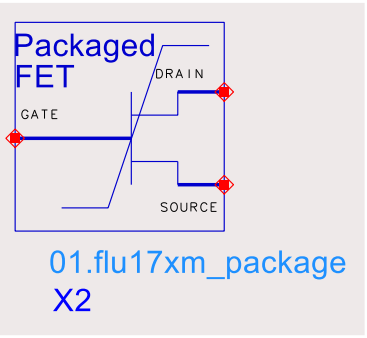


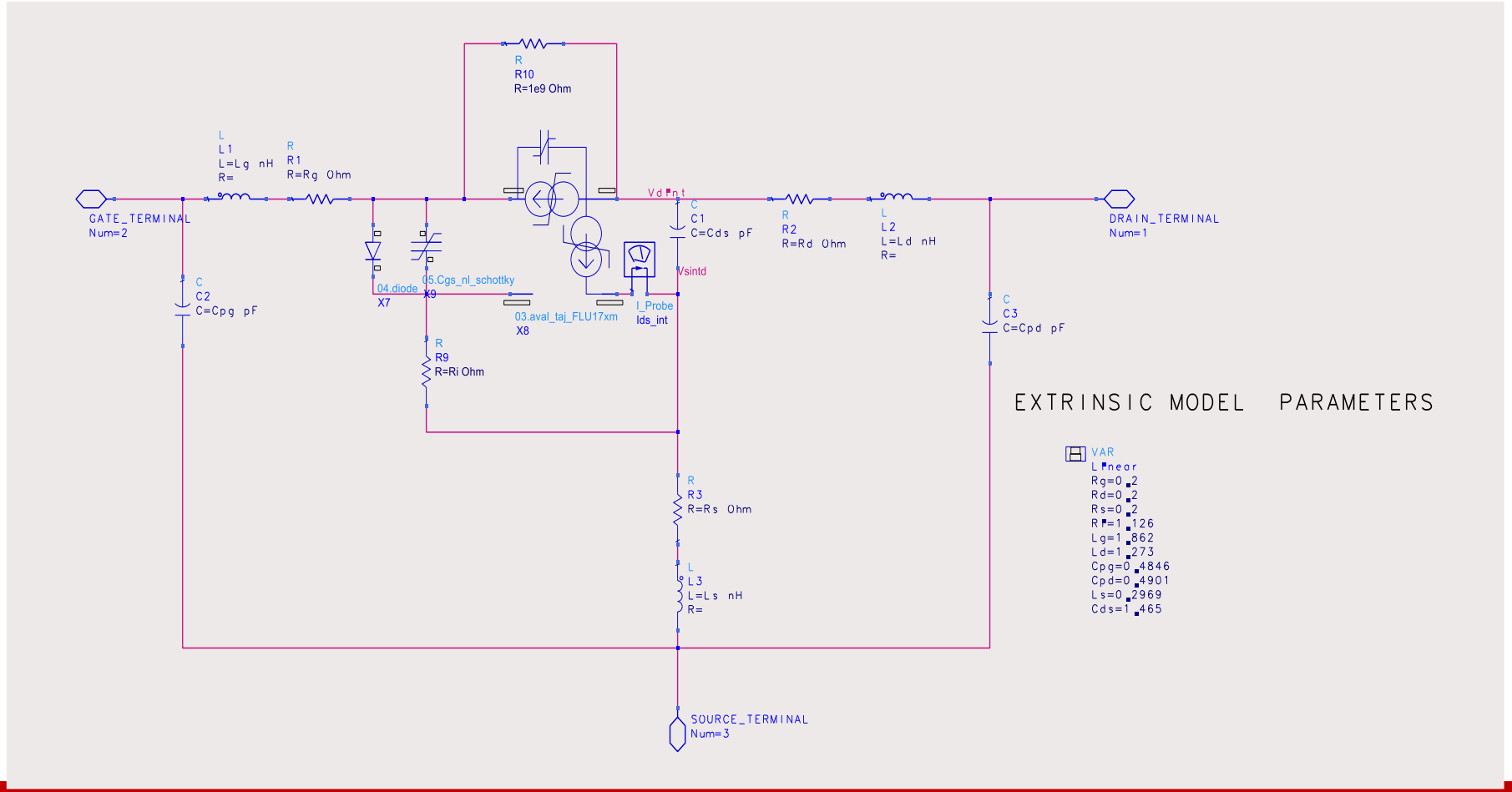
Each schematic file (*.dsn) is detailed hereafter as well as the corresponding data display window (*.dds).



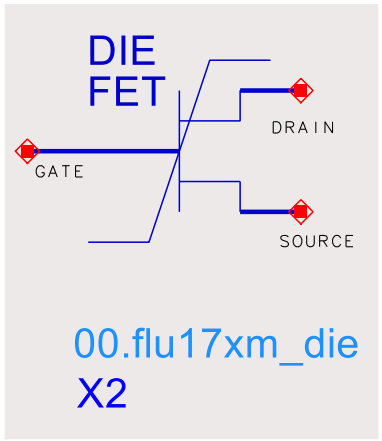


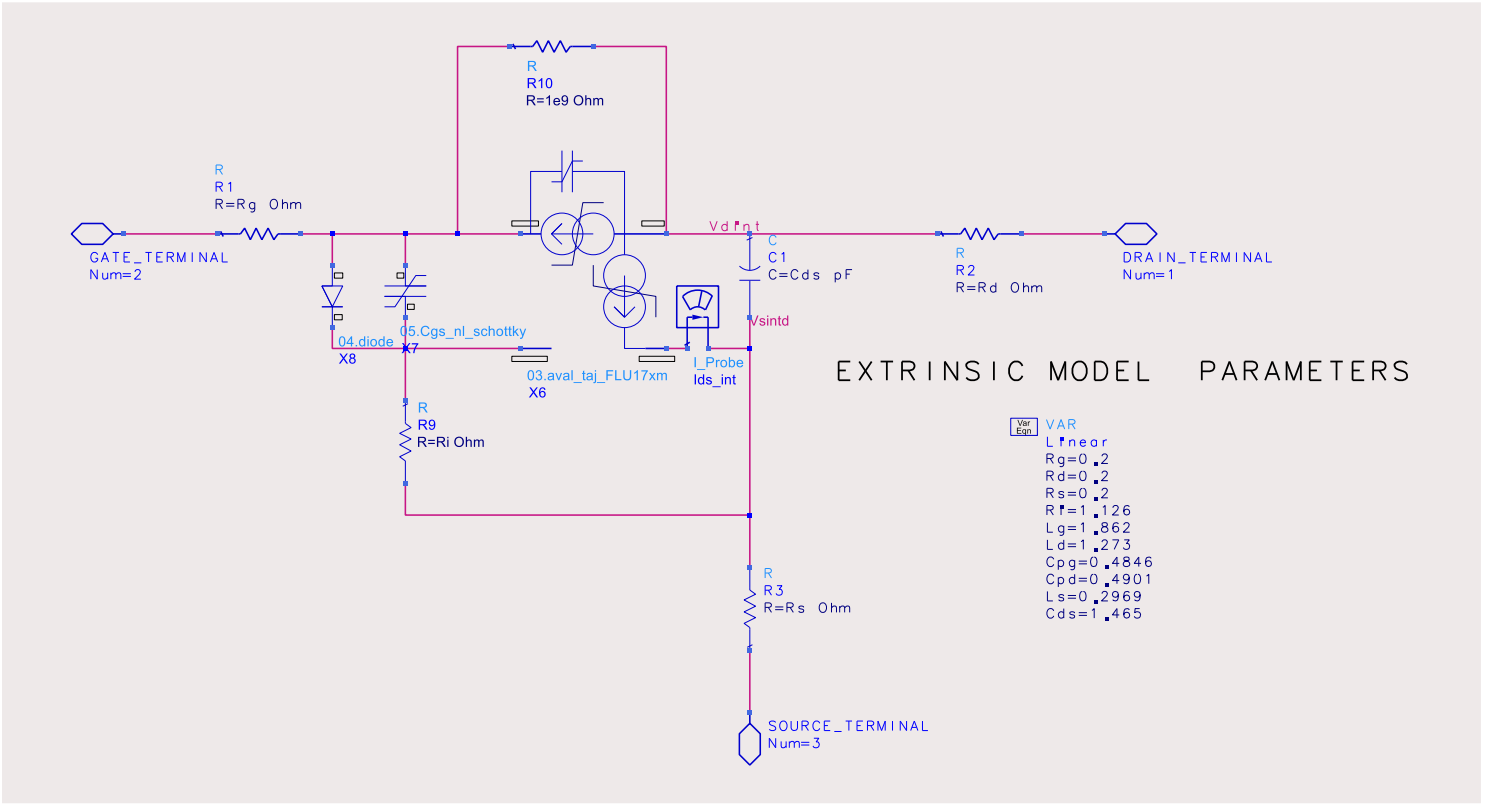
The *01.flu17xm_package.dsn* file defines the non-linear packaged model of the FET. The figure below represents the symbol associated to this model (and to this file). You can click on *Window/symbol* in the *Window* menu.



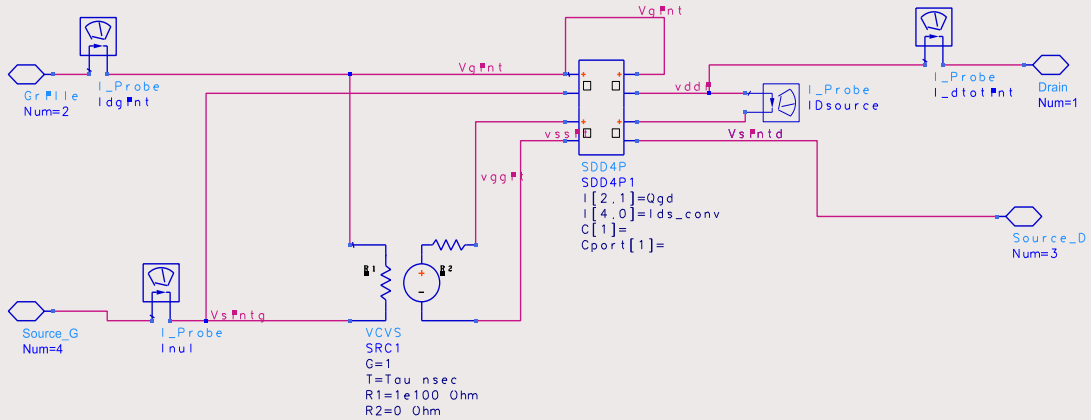


The *00.flu17xm_package.dsn* file defines the non-linear packaged model of the FET. The figure below represents the symbol associated to this model (and to this file). You can click on *Window/symbol* in the *Window* menu.



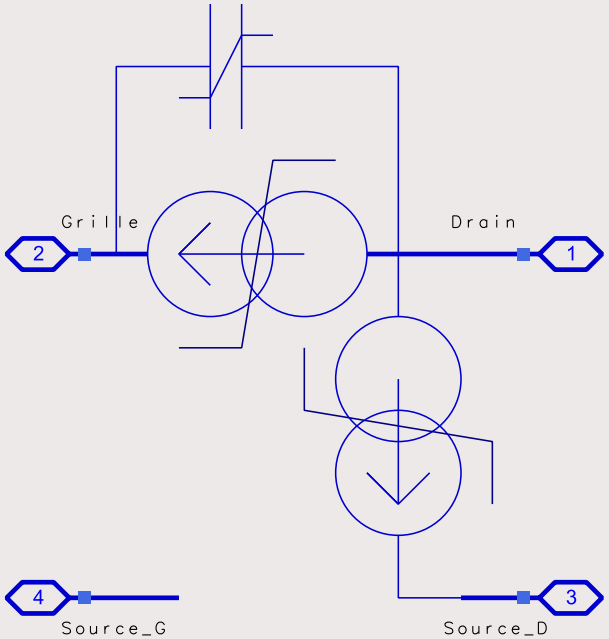


The *aval_taj.dsn* file defines the non-linear intrinsic model of the FET. The figure below represents the symbol associated to this model (and to this file). You can click on *Window/symbol* in the *Window* menu.

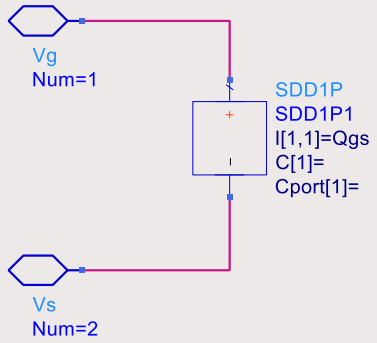


MODEL EQUATIONS AND PARAMETERS

```
VAR
Linear
Tau=0.02160

VAR
Control_Voltages
Vdf=_v4
Vgf=_v1
Vgft=_v3
Vgd=_v2
Vr=-(_v2)
Vcgd=_v2

VAR
Gate_Drain
Cgdo=1.282e-012
Vbf_cgd=0.7
K_cgd=0.8

VAR
Ids_param
E=1.099
Ke=0.07078
Mu=2422
Delta=5.970
Kg=-0.08187
Idss=0.5711
n1=6.619
n2=4.988
Vp=-1.450
Vpf=1.165
Gamma=-0.1044
Ss=-0.01723
Sl=1.014

VAR
Ids_Source
Vgfteff=(1/(2*n1))*(n2*Vgft1+Vgftlch)
Vgftlch=ln(2*cosh(n2*Vgft1))
Vgflch=ln(2*cosh(Vgf))
Vgf=Vgft-(Vpf)
Vgft1=Vgft-(Vp+Gamma*Vdf)
Ids1=ff Vgfteff < 1e-20 then 1e-20 else Idss*Vgfteff^(E+Ke*Vgft) endff
Ids2= (1+(Ss*Vdf/Idss))*exp((-(Vgflch^Delta)/Mu))
Ids3=tanh(Sl*Vdf/(Idss*(1-Kg*Vgft)))
Ids=Ids1*Ids2*Ids3

VAR
Convergence_Improvement
Ids_conv=ff Ids < 0 then 0 else Ids endff

VAR
Cgd_NONLIN
Qgd1_cgd=-2*Cgdo*Vbf_cgd*(1-Vcgd/Vbf_cgd)^0.5
Qgd2_cgd=-0.25*(Cgdo*(4*Vcgd*Vbf_cgd - 6*K_cgd*Vbf_cgd*Vcgd + Vcgd^2 - 8*Vbf_cgd^2 + 12*Vbf_cgd^2*K_cgd - 3 *Vbf_cgd^2*K_cgd^2))/(Vbf_cgd*((1 - K_cgd)^0.5)*(-1 + K_cgd))
Qgd=ff Vcgd<K_cgd*Vbf_cgd then Qgd1_cgd else Qgd2_cgd endff
```



Cgs Non linear EQUATIONS AND PARAMETERS

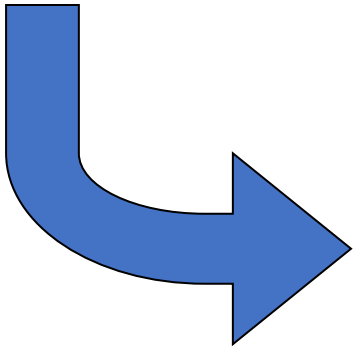


Var Eqn VAR
Control_Voltages
Vcgs=_v1

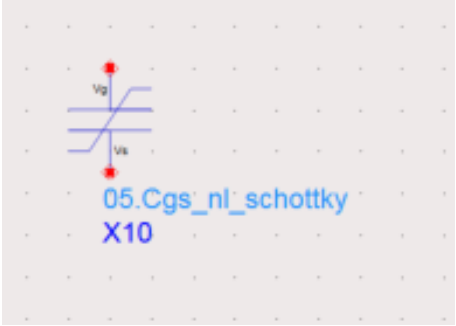
Var Eqn VAR
Gate_Source
Cgso=9.592e-012
Vbi_cgs=0.8
K_cgs=0.8

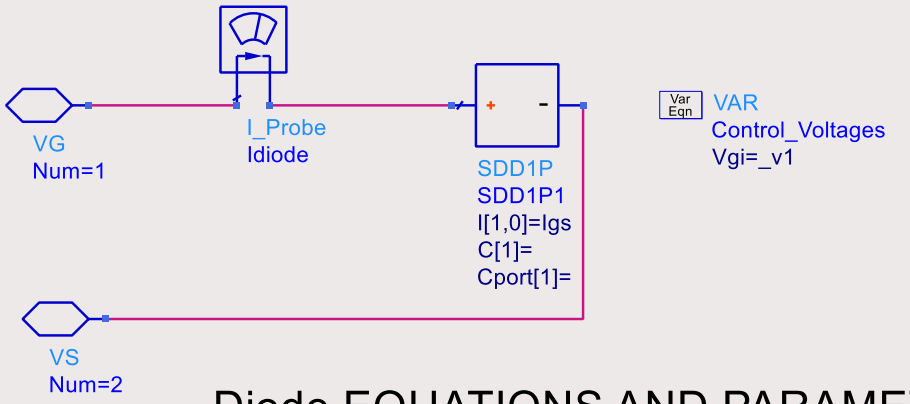
Var Eqn VAR
Cgs_NONLIN
Qgs1_cgs=-2*Cgso*Vbi_cgs*(1-Vcgs/Vbi_cgs)^0.5
Qgs2_cgs=-0.25*(Cgso*(4*Vcgs*Vbi_cgs - 6*K_cgs*Vbi_cgs*Vcgs + Vcgs^2 - 8*Vbi_cgs^2 + 12*Vbi_cgs^2*K_cgs - 3 *Vbi_cgs^2*K_cgs^2))/(Vbi_cgs*((1 - K_cgs)^0.5)*(-1 + K_cgs))
Qgs=if Vcgs<K_cgs*Vbi_cgs then Qgs1_cgs else Qgs2_cgs endif

The *Cgs_nl_schottky.dsn* file defines the non-linear model of the C_{GS} capacitance of the FET.



Associated symbol



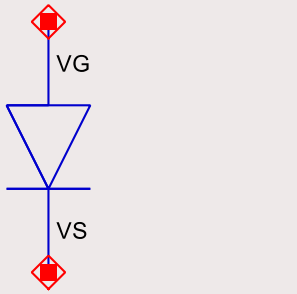


Diode EQUATIONS AND PARAMETERS

Var Eqn
VAR
IgS_NONLIN
as_igs=Alphas_igs*Vgi
bs_igs=Alphas_igs*Vgi-Ms_igs
I1_igs=Ins_igs*(exp(Ms_igs)*(1+bs_igs+bs_igs^2/2)-1)
I2_igs=if as_igs<-20 then -Ins_igs else Ins_igs*(exp(as_igs)-1) endif
Igs=if bs_igs>0 then I1_igs else I2_igs endif

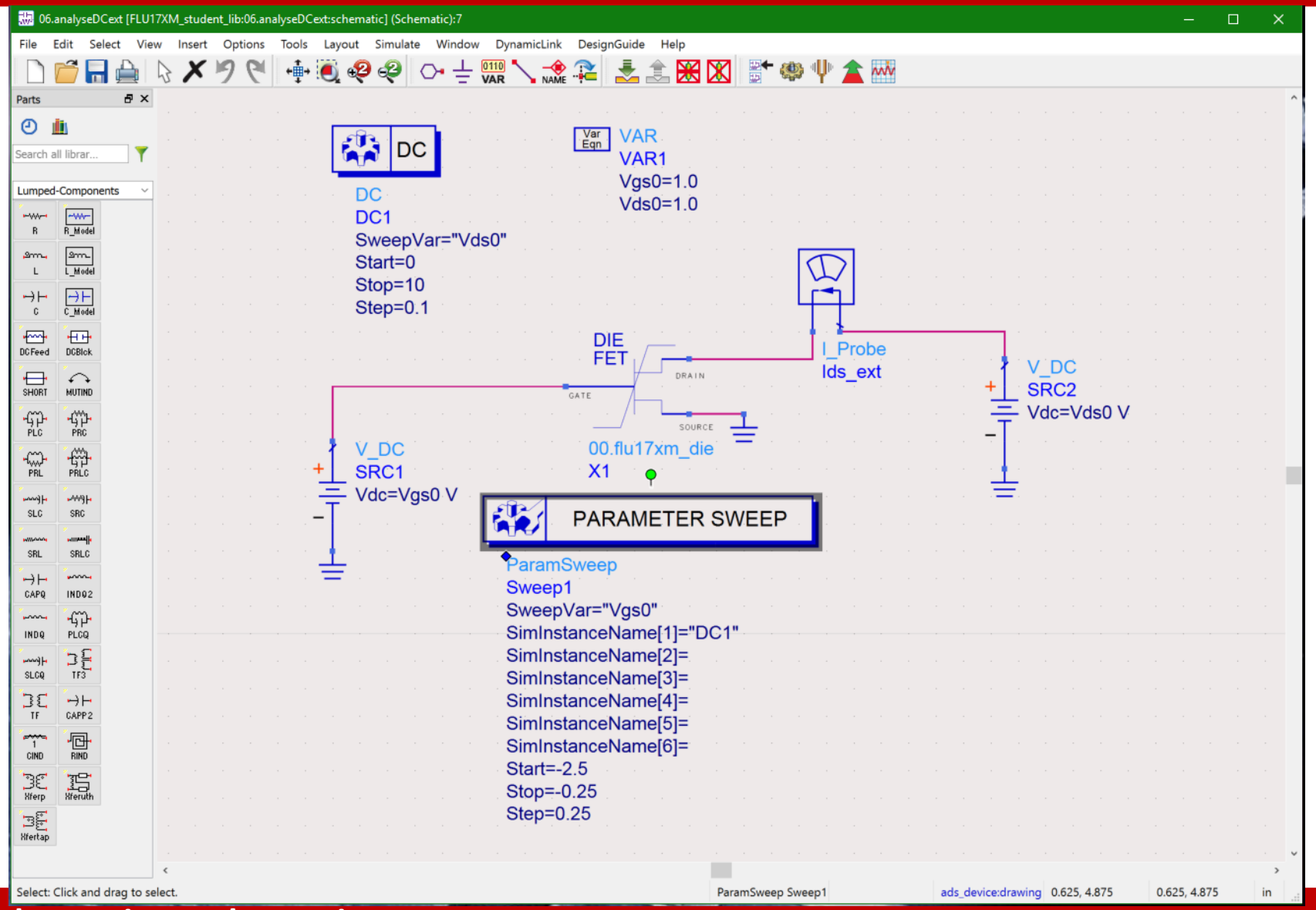
Var Eqn
VAR
Gate_Source
Ins_igs=4.822e-011
Ms_igs=40
Alphas_igs=29.1
Cgso=9.592e-012
Vbi_cgs=0.8
K_cgs=0.8

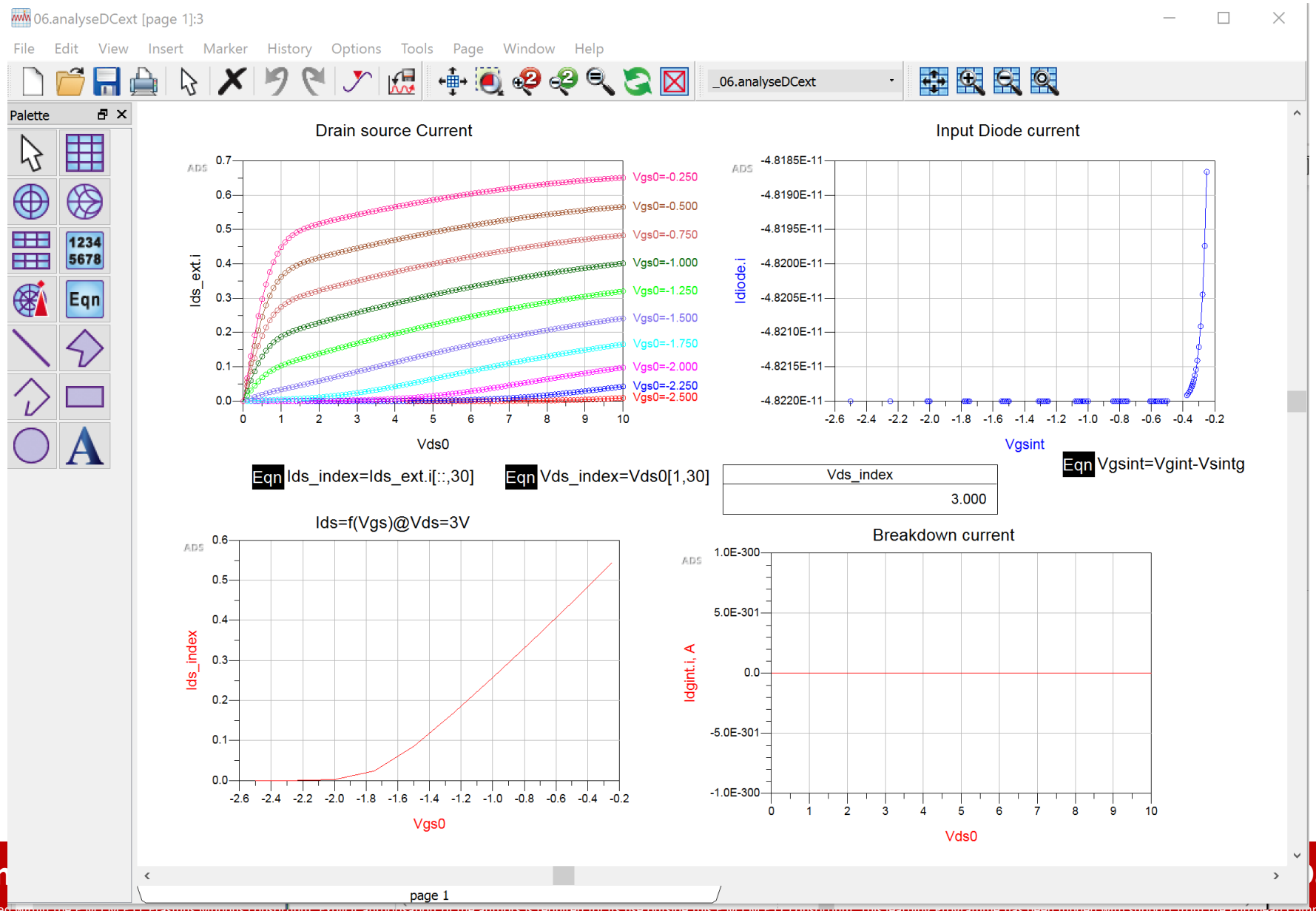
Associated symbol



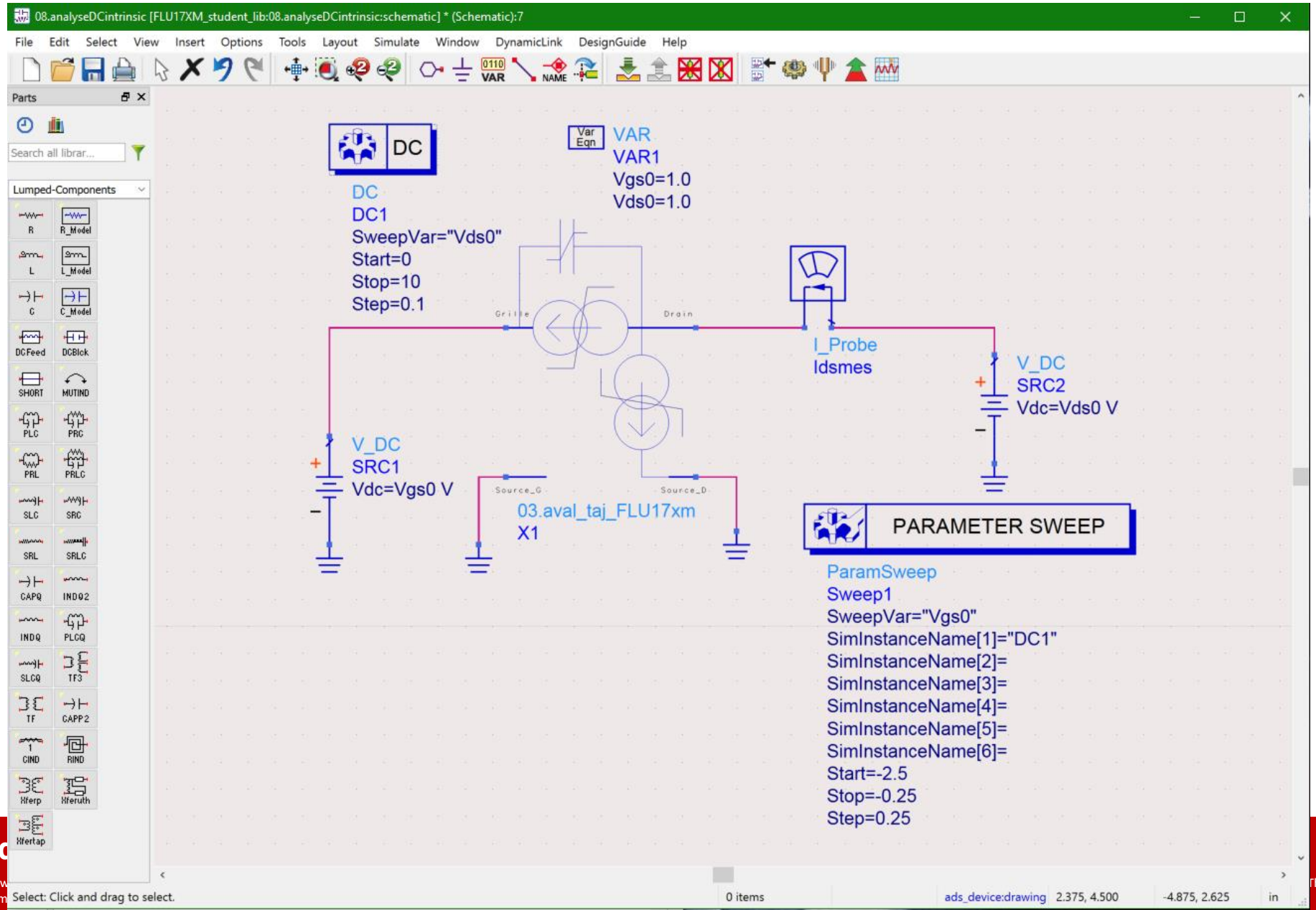
The *Diode.dsn* file defines the non-linear model of the input Diode of the FET.

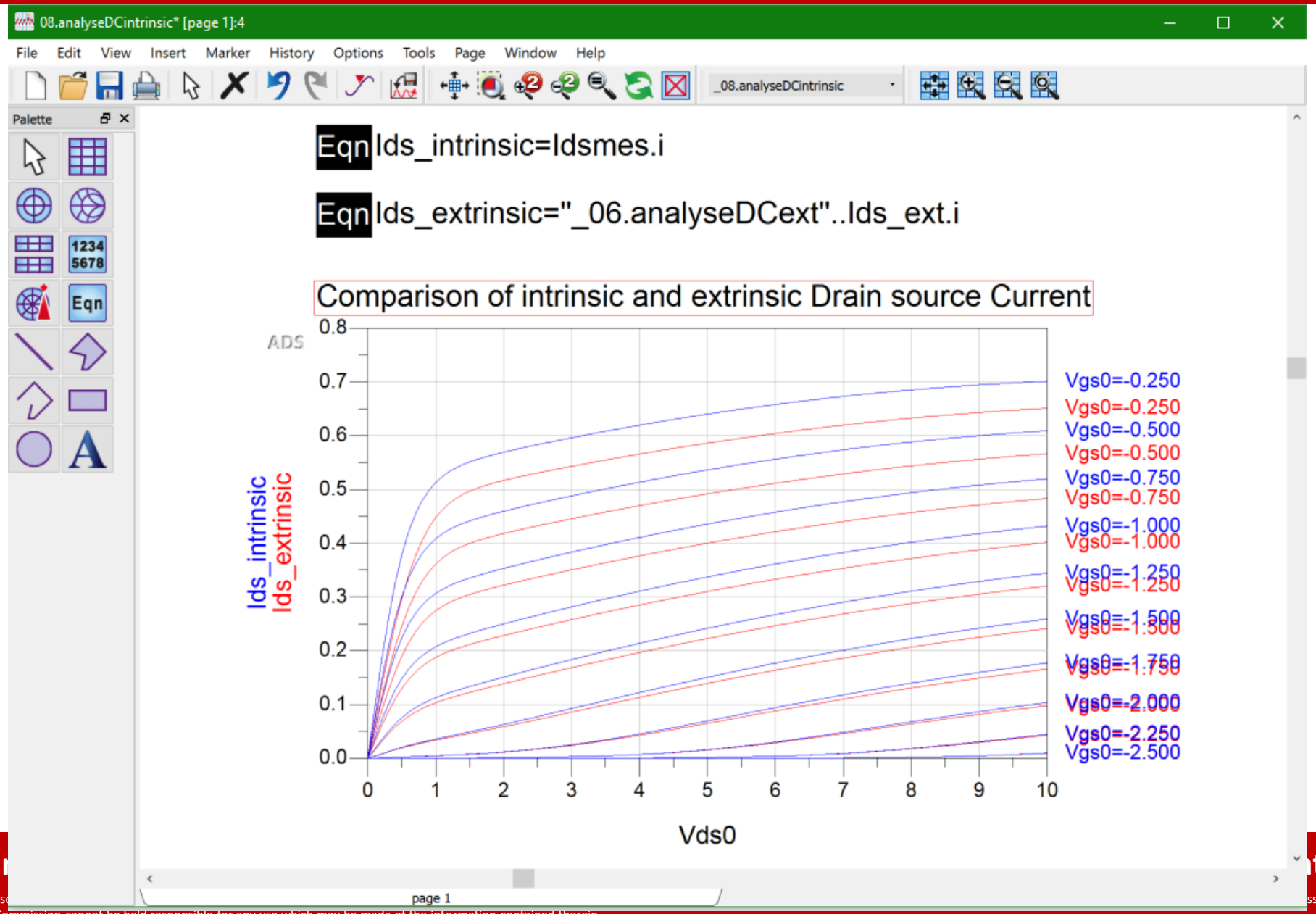
Schematic to plot the extrinsic I/V curves of the Die transistor



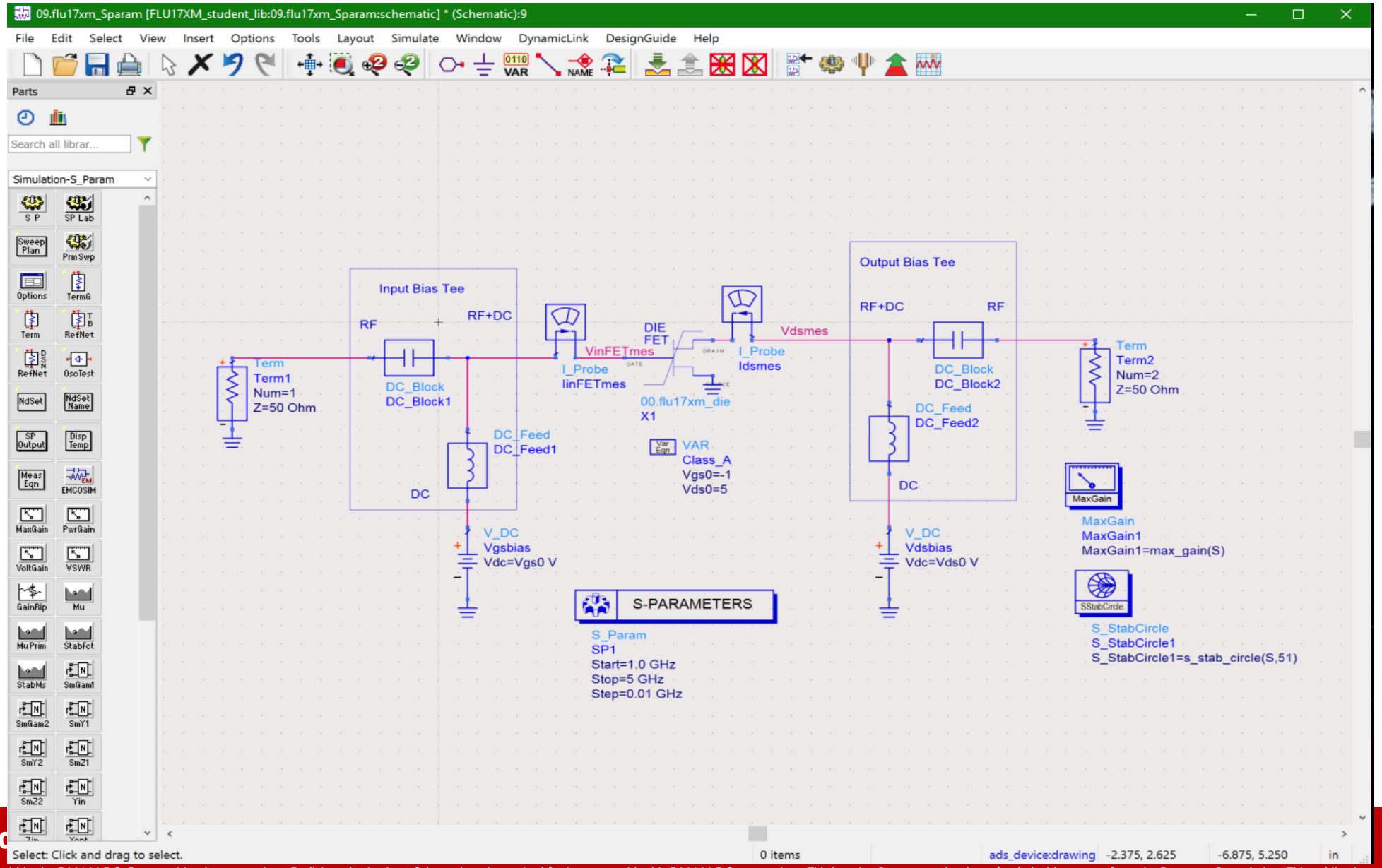


Schematic to plot the intrinsic I/V curves of the Die transistor





Schematic to plot the S parameter curves of the Die transistor



Work Frequency Selection :

Choose the value of index_freq to choose the desired frequency

Eqn index_freq=80

Eqn fwork_Hz=freq[index_freq]

fwork_Hz
1.800E9

Input Reflection

S11(f)=b1(f)/a1(f) when a2(f)=0

|S11(f)|=0 <=> matched
|S11(f)|<>0 <=> unmatched

freq	m1	indep(m1)
1.800 GHz	0.890 / -158.855	1.800E9

m1
freq=1.800GHz
S(1,1)=0.890 / -158.855
impedance = 3.013 - j9.300

S11_fwork
0.890 / -158.855

Eqn S11_fwork=S(1,1)[index_freq]

Eqn Z11calc_fwork=50*(1+S11_fwork)/(1-S11_fwork)

Z11calc_fwork	Z11calc_fwork
3.013 - j9.300	9.776 / -72.051

Eqn Cin_F=-imag(Z11calc_fwork)/(2*pi*fwork_Hz)

Eqn Rin=real(Z11calc_fwork)

Input Series Capacitance

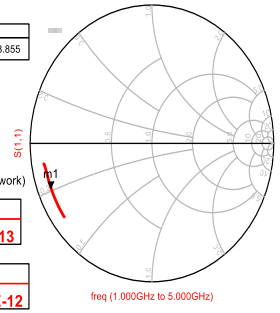
Cin_F
8.223E-10

Rin
3.013

Eqn Linmatch_H=1/(Cin_F*(2*pi*fwork_Hz)**2)

Input Series Inductance for capacitance compensation

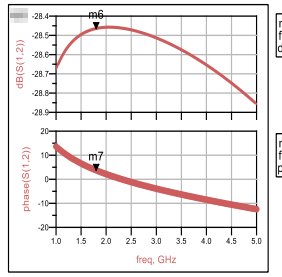
Linmatch_H
9.508E-12



Reverse transmission

S12(f)=b1(f)/a2(f) when a1(f)=0

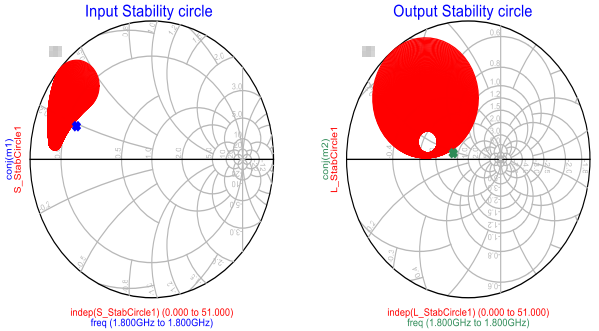
|S12(f)|<1 <=> losses
|S12(f)|>1 <=> gain
|S12(f)|<0dB <=> losses
|S12(f)|>0dB <=> gain



m6
freq=1.800GHz
dB(S(1,2))=-28.464

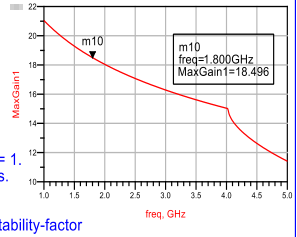
m7
freq=1.800GHz
phase(S(1,2))=3.734

Stability circles

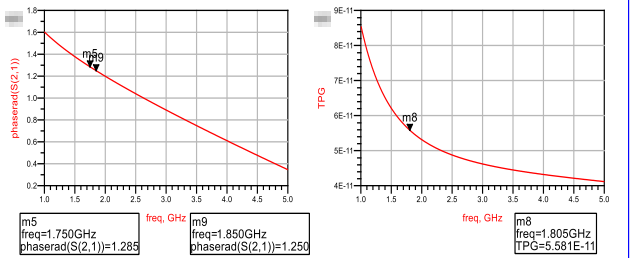


MaxGain :
Maximum available and stable gain (in dB) between the input and the measurement ports

This function should be used in a stable system, i.e. when the stability factor (k) is greater than 1. The function actually works for both k > 1 and k <= 1. If k > 1, then k is used in the max gain calculations. However, if k <= 1, then k = 1 is used in the max gain calculations.
https://www.microwaves101.com/encyclopedias/stability-factor



Group Delay Determination



m5
freq=1.750GHz
phaserad(S(2,1))=1.285

m9
freq=1.850GHz
phaserad(S(2,1))=1.250

m8
freq=1.805GHz
TPG=5.581E-11

Eqn TPG=-1/(2*pi)*diff(phaserad(S(2,1)))

TPG[index_freq]
5.581E-11

Eqn Group_delay_eval=-1/(2*pi)*(m9-m5)/(indep(m9)-indep(m5))

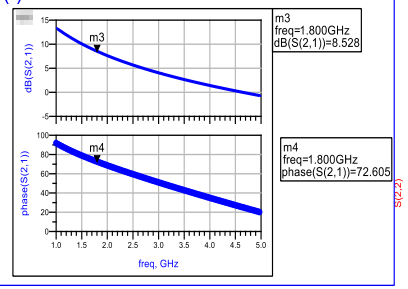
Group_delay_eval[0]
5.590E-11

Direct transmission

S21(f)=b2(f)/a1(f) when a2(f)=0

|S21(f)|<1 <=> losses
|S21(f)|>1 <=> gain
|S21(f)|<0dB <=> losses
|S21(f)|>0dB <=> gain

Phase linearity => Constant Group delay



m3
freq=1.800GHz
dB(S(2,1))=8.528

m4
freq=1.800GHz
phase(S(2,1))=72.605

Output Reflection

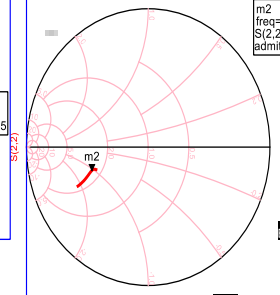
S22(f)=b2(f)/a2(f) when a1(f)=0

|S22(f)|=0 <=> matched
|S22(f)|<>0 <=> unmatched

freq	m2	indep(m2)
1.800 GHz	0.492 / -159.975	1.800E9

fwork_Hz
1.800E9

S22_fwork
0.492 / -159.975



m2
freq=1.800GHz
S(2,2)=0.492 / -159.975
admittance = 0.048 + j0.021

Eqn S22_fwork=S(2,2)[index_freq]

Eqn Y22calc_fwork=1/50*(1-S22_fwork)/(1+S22_fwork)

Y22calc_fwork	Y22calc_fwork
0.048 + j0.021	0.052 / 23.938

Eqn Routparcalc_fwork=real(1/Y22calc_fwork)

Routparcalc_fwork
17.511

Eqn Cout_F=imag(Y22calc_fwork)/(2*pi*fwork_Hz)

Cout_F
1.873E-12

Eqn Loutmatch_H=1/(Cout_F*(2*pi*fwork_Hz)**2)

Loutmatch_H
4.175E-9

Input Reflection
S11(f)=b1(f)/a1(f) when a2(f)=0

|S11(f)|=0 <=> matched
|S11(f)|<>0 <=> unmatched

freq	m1	indep(m1)
1.800 GHz	0.890 / -158.855	1.800E9

m1
freq=1.800GHz
S(1,1)=0.890 / -158.855
impedance = 3.013 - j9.300

S11_fwork
0.890 / -158.855

Eqn S11_fwork=S(1,1)[index_freq]

Eqn Z11calc_fwork=50*(1+S11_fwork)/(1-S11_fwork)

Z11calc_fwork	Z11calc_fwork
3.013 - j9.300	9.776 / -72.051

Eqn Cin_F=-imag(Z11calc_fwork)/(2*pi*fwork_Hz)

Eqn Rin=real(Z11calc_fwork)

Input Series Capacitance

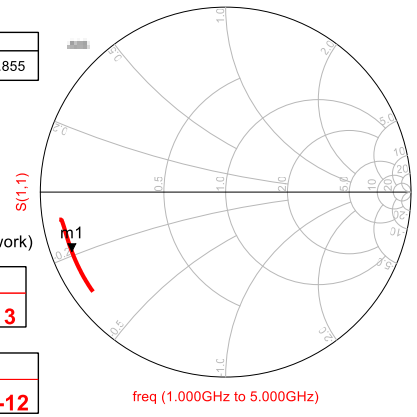
Cin_F
8.223E-10

Rin
3.013

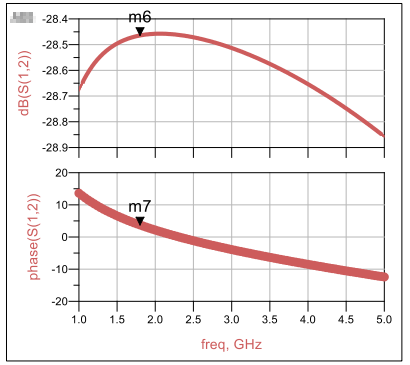
Eqn Linmatch_H=1/(Cin_F*(2*pi*fwork_Hz)**2)

Input Series Inductance for capacitance compensation

Linmatch_H
9.508E-12



Reverse transmission
S12(f)=b1(f)/a2(f) when a1(f)=0



m6
freq=1.800GHz
dB(S(1,2))=-28.464

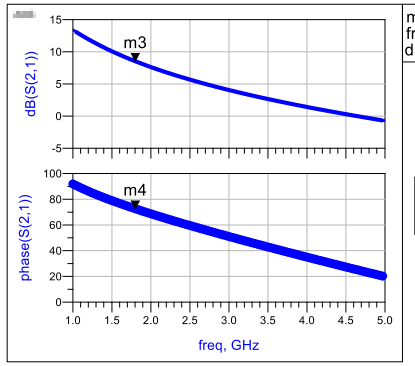
m7
freq=1.800GHz
phase(S(1,2))=3.734

|S12(f)|<1 <=> losses
|S12(f)|>1 <=> gain
|S12(f)|<0dB <=> losses
|S12(f)|>0dB <=> gain

Direct transmission
S21(f)=b2(f)/a1(f) when a2(f)=0

|S21(f)|<1 <=> losses
|S21(f)|>1 <=> gain
|S21(f)|<0dB <=> losses
|S21(f)|>0dB <=> gain

Phase linearity => Constant Group delay



m3
freq=1.800GHz
dB(S(2,1))=8.528

m4
freq=1.800GHz
phase(S(2,1))=72.605

Output Reflection
S22(f)=b2(f)/a2(f) when a1(f)=0

|S22(f)|=0 <=> matched
|S22(f)|<>0 <=> unmatched

freq	m2	indep(m2)
1.800 GHz	0.492 / -159.975	1.800E9

fwork_Hz
1.800E9

S22_fwork
0.492 / -159.975

m2
freq=1.800GHz
S(2,2)=0.492 / -159.975
admittance = 0.048 + j0.021

Eqn S22_fwork=S(2,2)[index_freq]

Eqn Y22calc_fwork=1/50*(1-S22_fwork)/(1+S22_fwork)

Y22calc_fwork	Y22calc_fwork
0.048 + j0.021	0.052 / 23.938

Eqn Routparcalc_fwork=real(1/Y22calc_fwork)

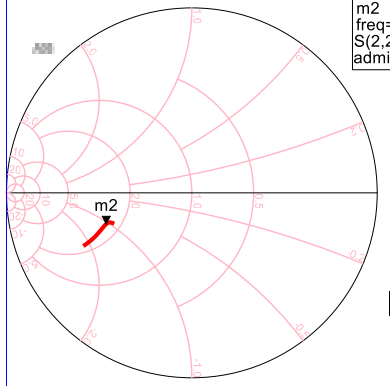
Routparcalc_fwork
17.511

Eqn Cout_F=imag(Y22calc_fwork)/(2*pi*fwork_Hz)

Cout_F
1.873E-12

Loutmatch_H
4.175E-9

Eqn Loutmatch_H=1/(Cout_F*(2*pi*fwork_Hz)**2)



Input Reflection
S11(f)=b1(f)/a1(f) when a2(f)=0

|S11(f)|=0 <=> matched
|S11(f)|<>0 <=> unmatched

freq	m1	indep(m1)
1.800 GHz	0.890 / -158.855	1.800E9

m1
freq=1.800GHz
S(1,1)=0.890 / -158.855
impedance = 3.013 - j9.300

S11_fwork
0.890 / -158.855

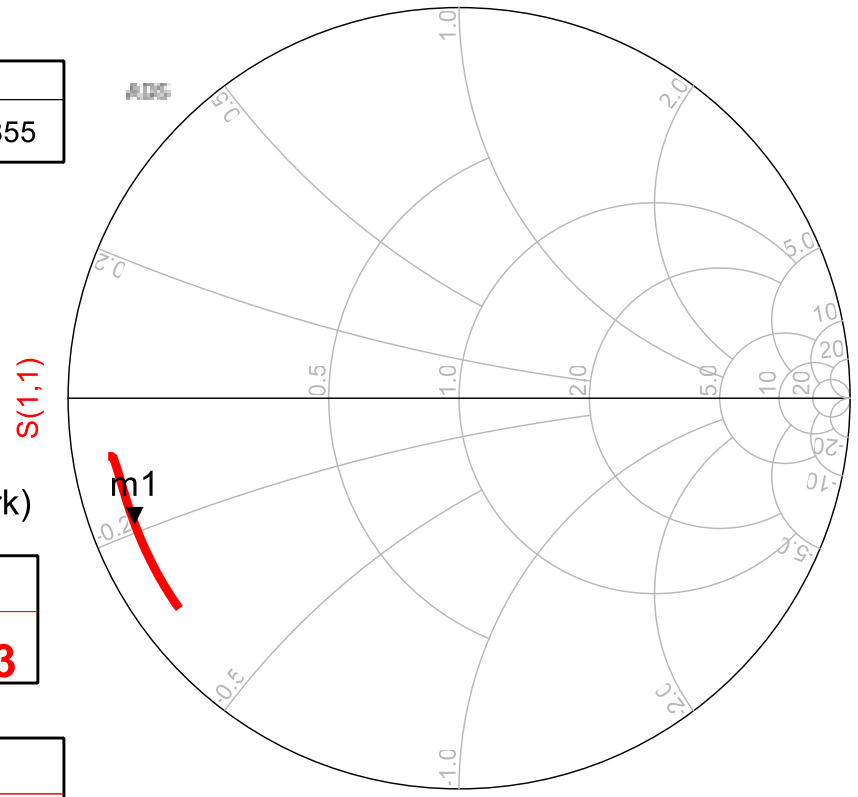
Eqn S11_fwork=S(1,1)[index_freq]

Eqn Z11calc_fwork=50*(1+S11_fwork)/(1-S11_fwork)

Z11calc_fwork	Z11calc_fwork
3.013 - j9.300	9.776 / -72.051

Eqn Cin_F=-imag(Z11calc_fwork)/(2*pi*fwork_Hz)

Eqn Rin=real(Z11calc_fwork)



Input Series Capacitance

Cin_F
8.223E-10

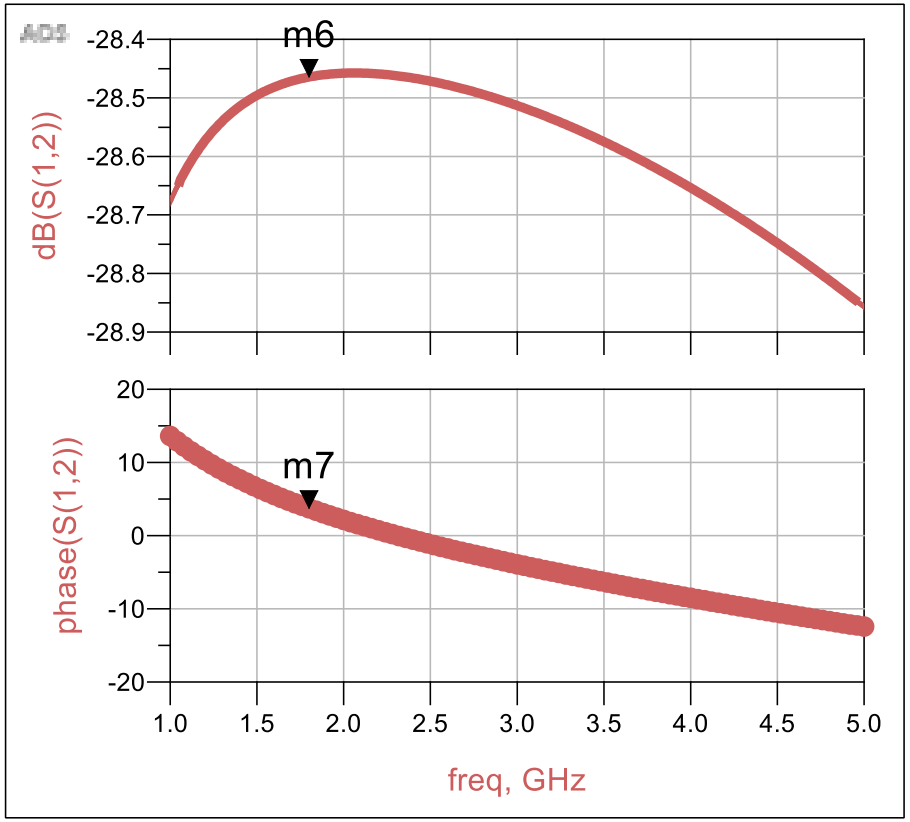
Rin
3.013

Eqn Linmatch_H=1/(Cin_F*(2*pi*fwork_Hz)**2)

Input Series Inductance for capacitance compensation

Linmatch_H
9.508E-12

Reverse transmission
S12(f)=b1(f)/a2(f) when a1(f)=0



m6
freq=1.800GHz
dB(S(1,2))=-28.464

m7
freq=1.800GHz
phase(S(1,2))=3.734

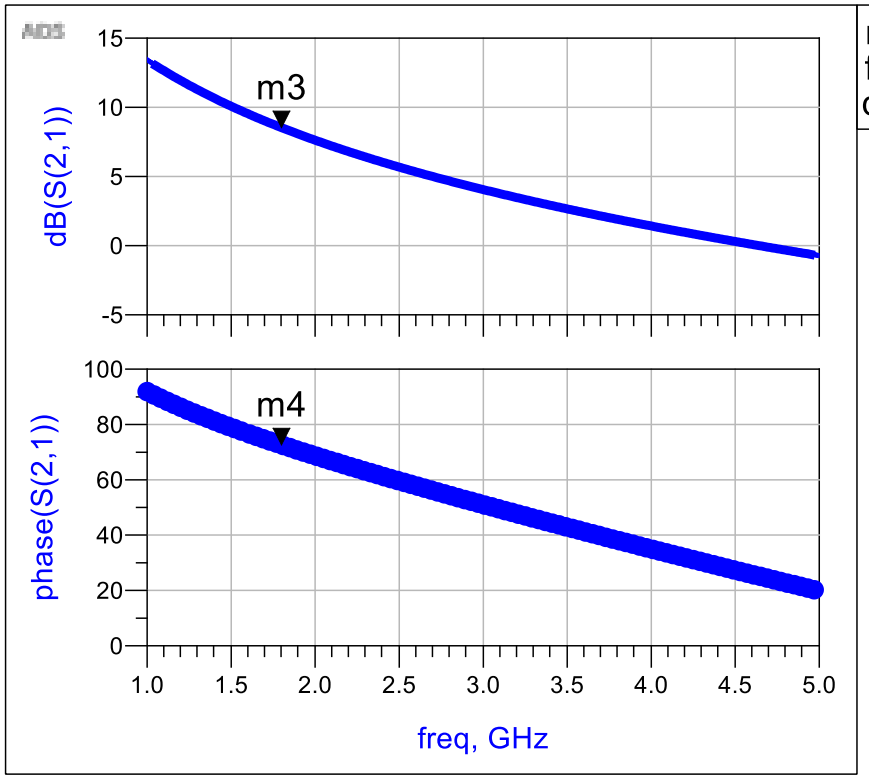
|S12(f)|<1 <=> losses
|S12(f)|>1 <=> gain
|S12(f)|<0dB <=> losses
|S12(f)|>0dB <=> gain

Direct transmission
S21(f)=b2(f)/a1(f) when a2(f)=0

|S21(f)|<1 <=> losses
|S21(f)|>1 <=> gain

|S21(f)|<0dB <=> losses
|S21(f)|>0dB <=> gain

Phase linearity => Constant Group delay



m3
freq=1.800GHz
dB(S(2,1))=8.528

m4
freq=1.800GHz
phase(S(2,1))=72.605

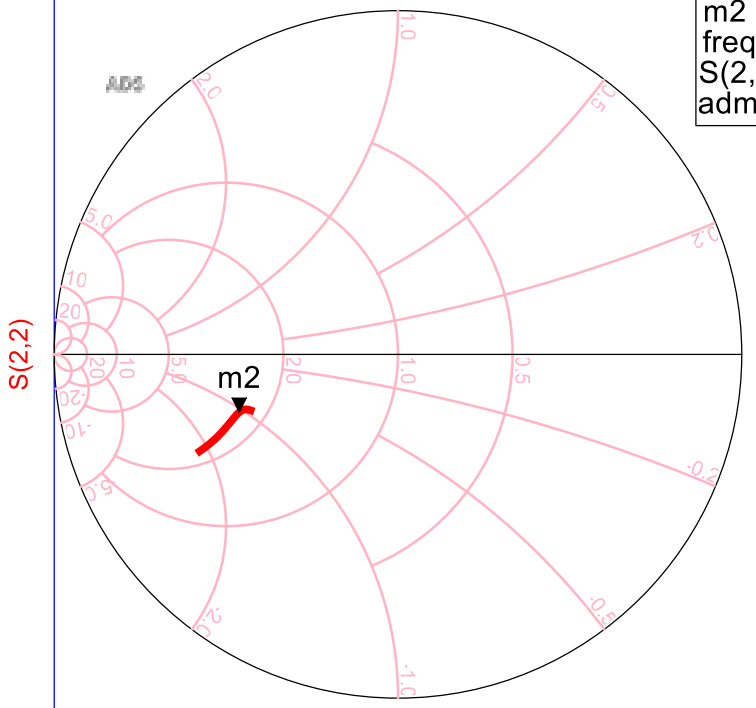
Output Reflection
S22(f)=b2(f)/a2(f) when a1(f)=0

|S22(f)|=0 <=> matched
|S22(f)|<>0 <=> unmatched

freq	m2	indep(m2)
1.800 GHz	0.492 / -159.975	1.800E9

fwork_Hz
1.800E9

S22_fwork
0.492 / -159.975



m2
freq=1.800GHz
S(2,2)=0.492 / -159.975
admittance = 0.048 + j0.021

Eqn S22_fwork=S(2,2)[index_freq]

Eqn Y22calc_fwork=1/50*(1-S22_fwork)/(1+S22_fwork)

Y22calc_fwork	Y22calc_fwork
0.048 + j0.021	0.052 / 23.938

Eqn Routparcalc_fwork=real(1/Y22calc_fwork)

Routparcalc_fwork
17.511

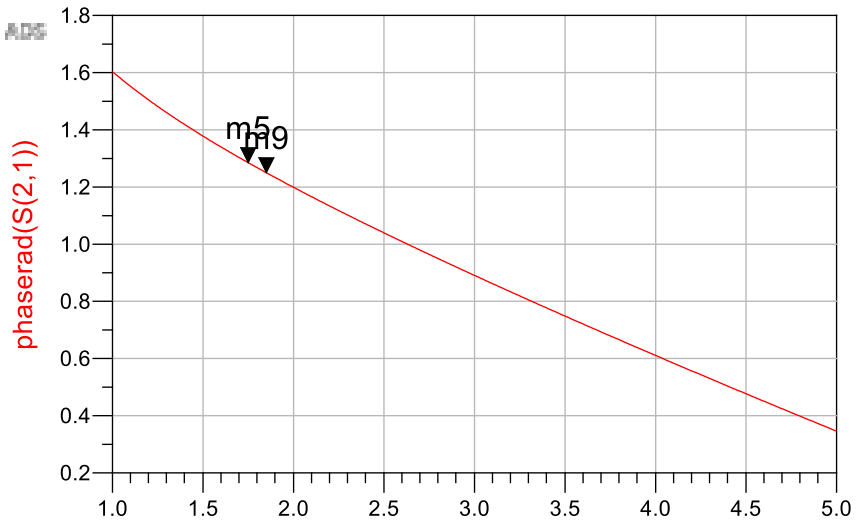
Eqn Cout_F=imag(Y22calc_fwork)/(2*pi*fwork_Hz)

Cout_F
1.873E-12

Eqn Loutmatch_H=1/(Cout_F*(2*pi*fwork_Hz)**2)

Loutmatch_H
4.175E-9

Group Delay Determination

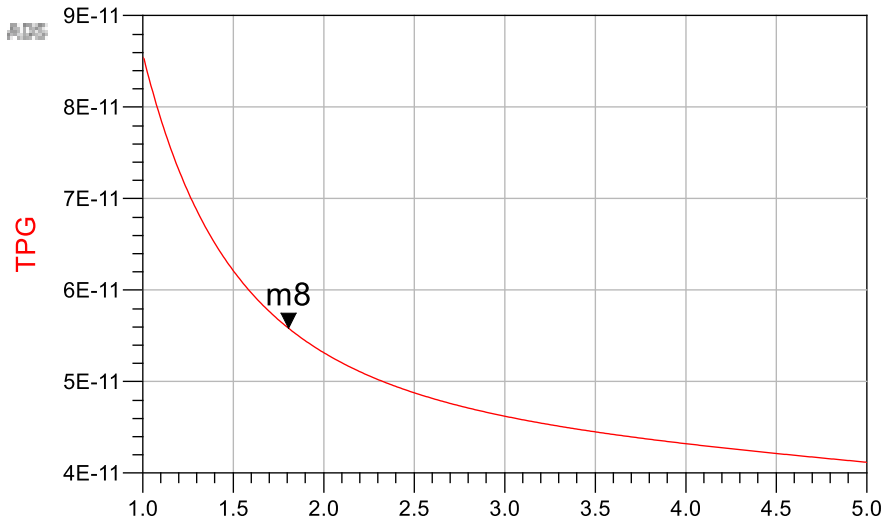


m5
freq=1.750GHz
phaserad(S(2,1))=1.285

m9
freq=1.850GHz
phaserad(S(2,1))=1.250

Eqn Group_delay_eval=-1/(2*pi)*(m9-m5)/(indep(m9)-indep(m5))

Group_delay_eval[0]
5.590E-11

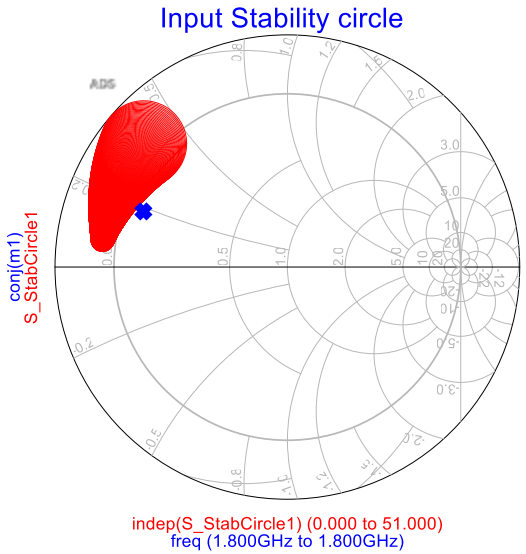


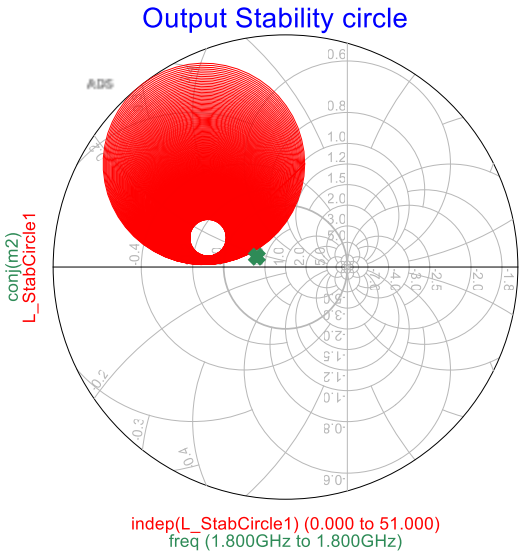
m8
freq=1.805GHz
TPG=5.581E-11

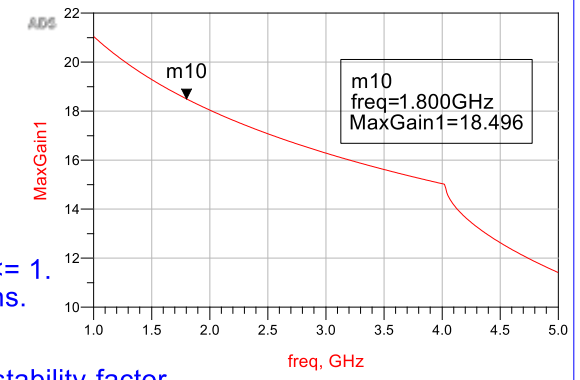
Eqn TPG=-1/(2*pi)*diff(phaserad(S(2,1)))

TPG[index_freq]
5.581E-11

Stability circles



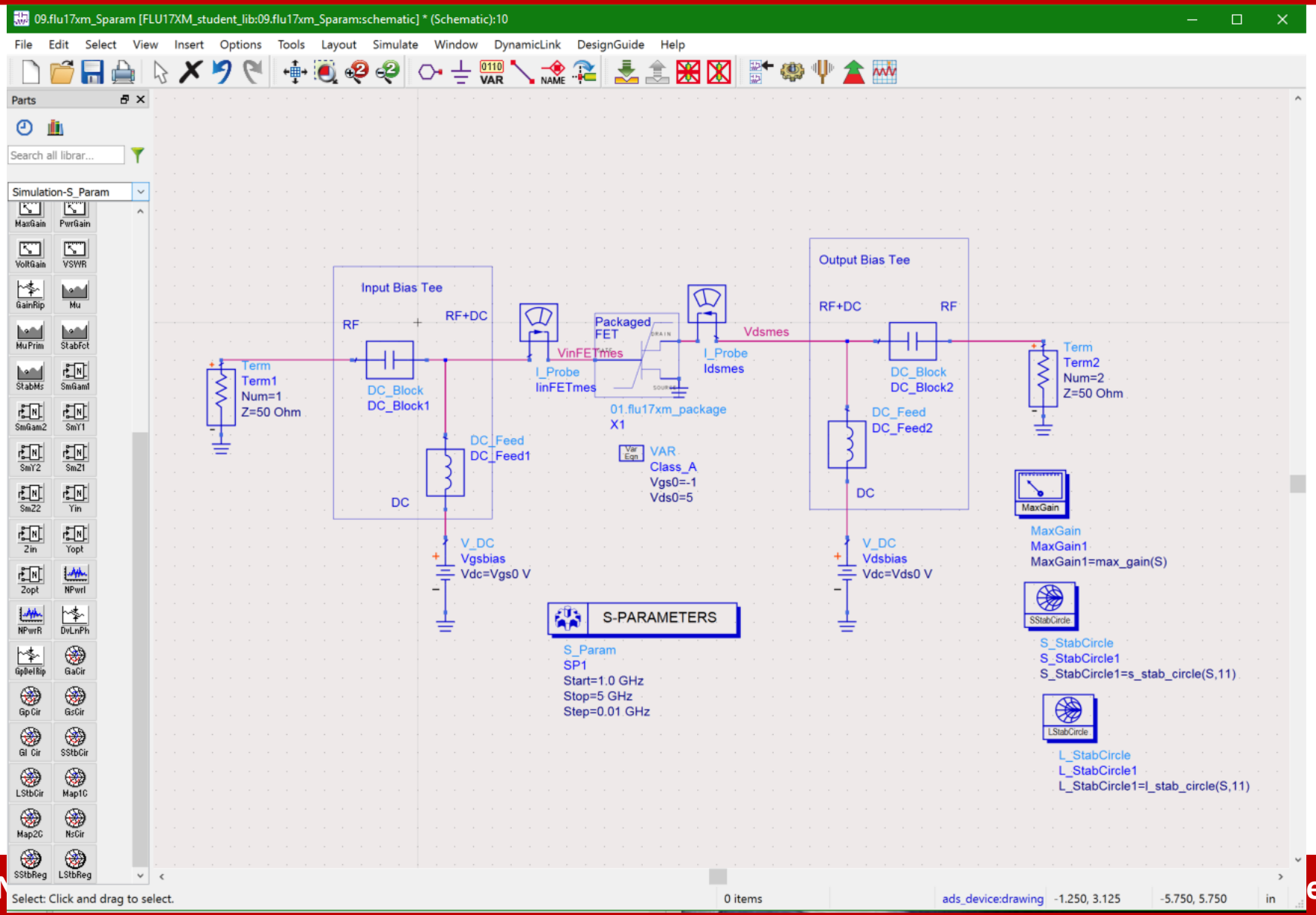


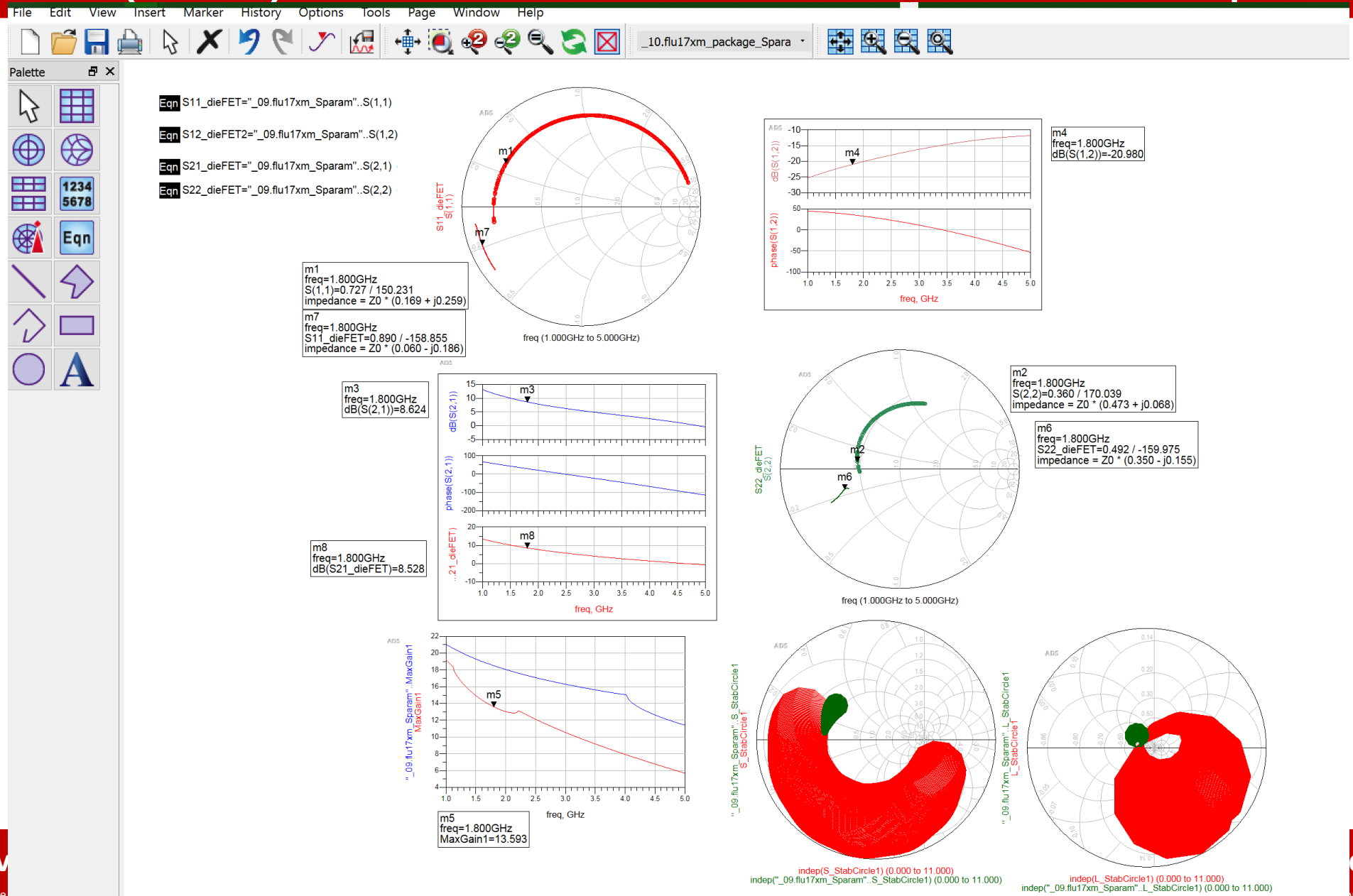


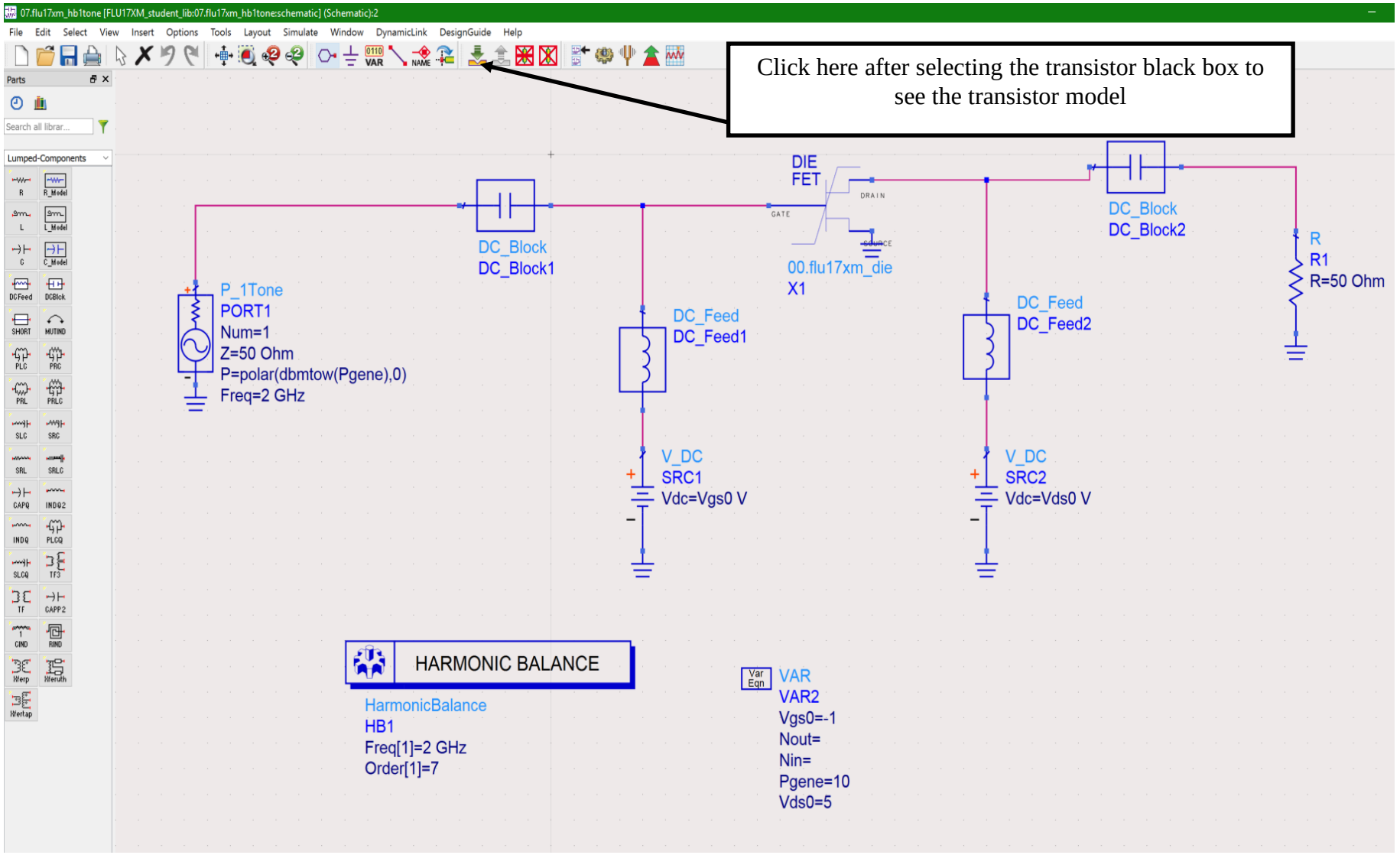
MaxGain :
Maximum available and stable gain (in dB) between the input and the measurement ports

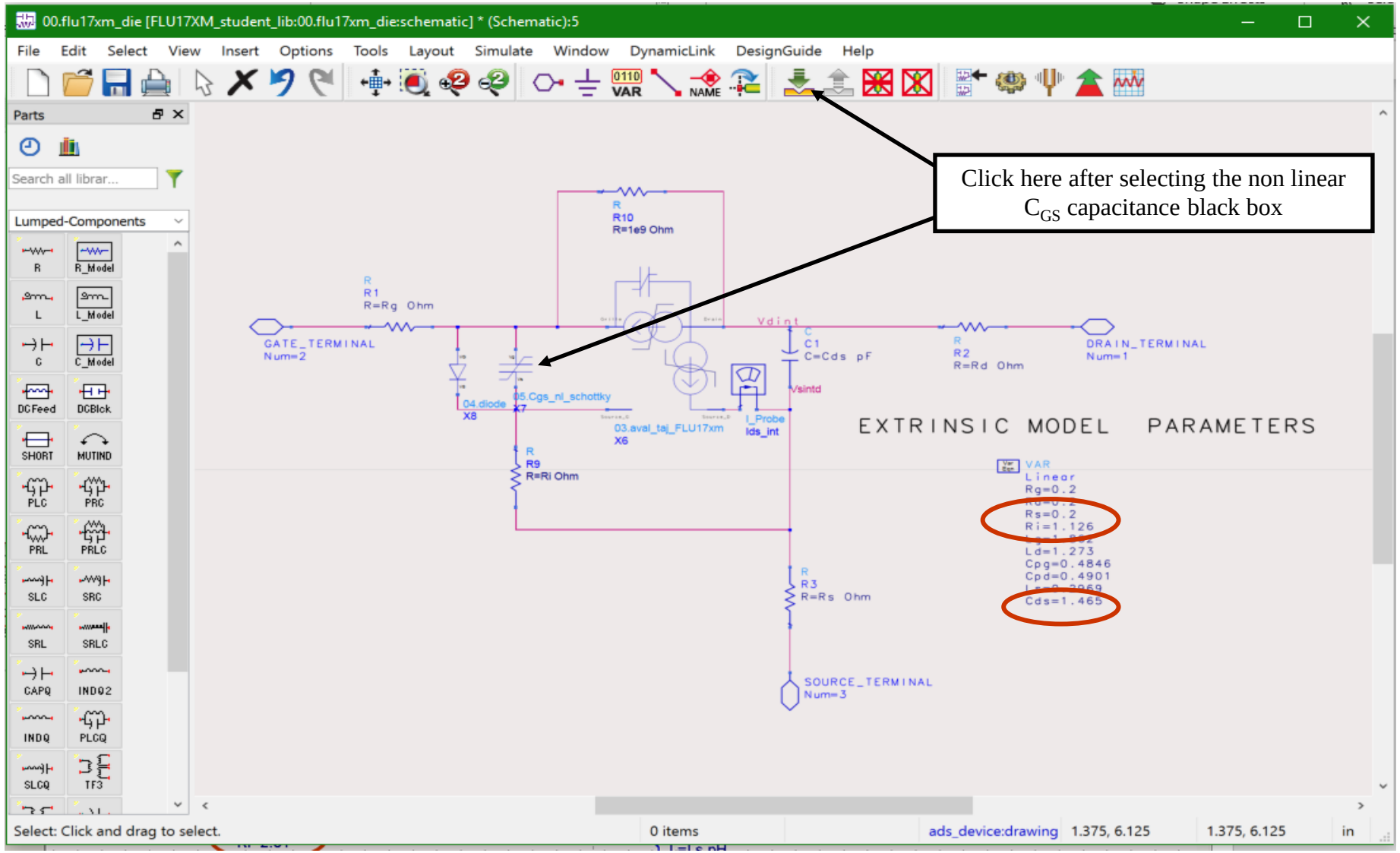
This function should be used in a stable system, i.e. when the stability factor (k) is greater than 1. The function actually works for both k > 1 and k <= 1. If k > 1, then k is used in the max gain calculations. However, if k <= 1, then k = 1 is used in the max gain calculations.
https://www.microwaves101.com/encyclopedias/stability-factor

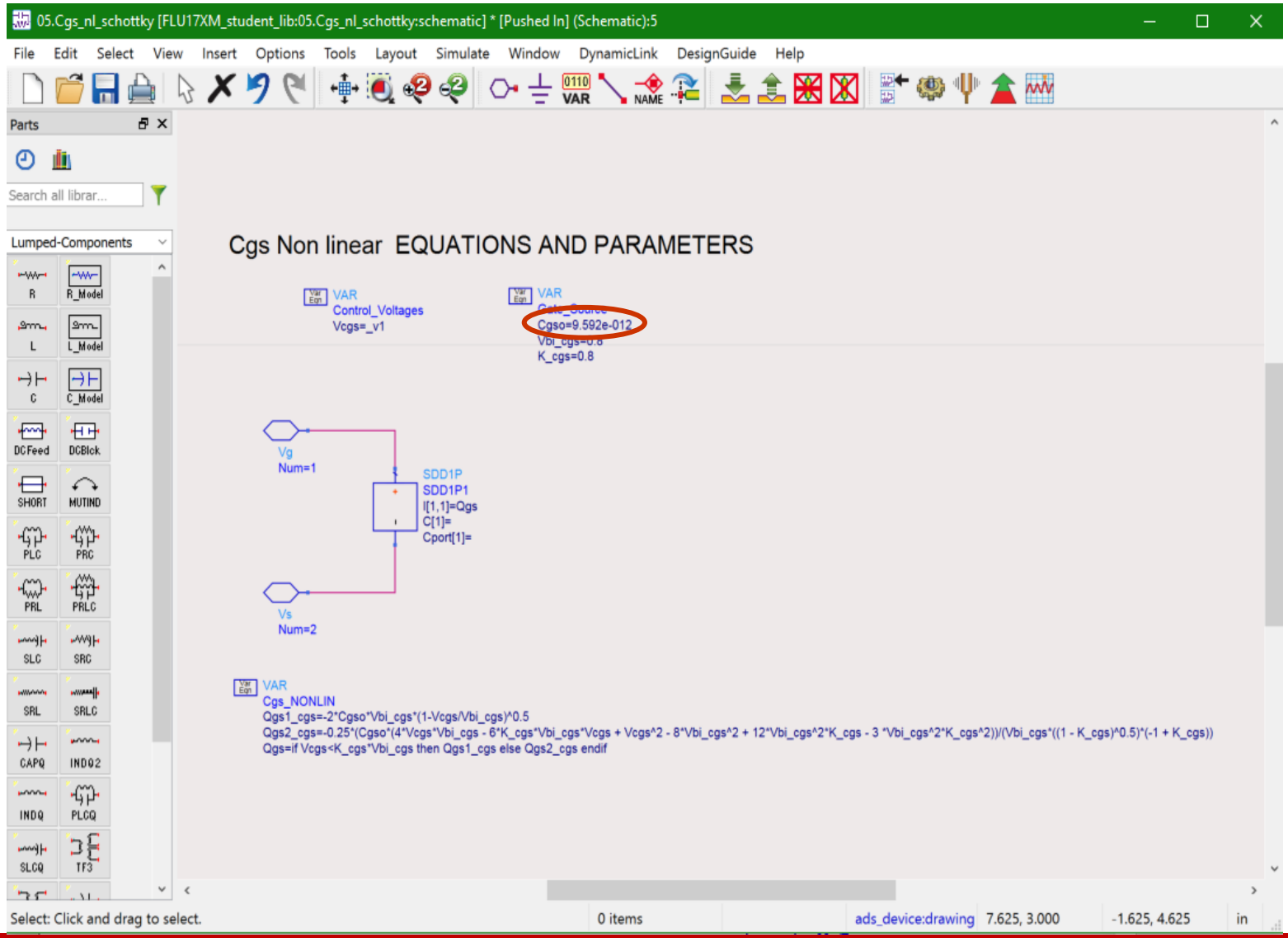
Schematic to plot the S parameter curves of the Packaged transistor

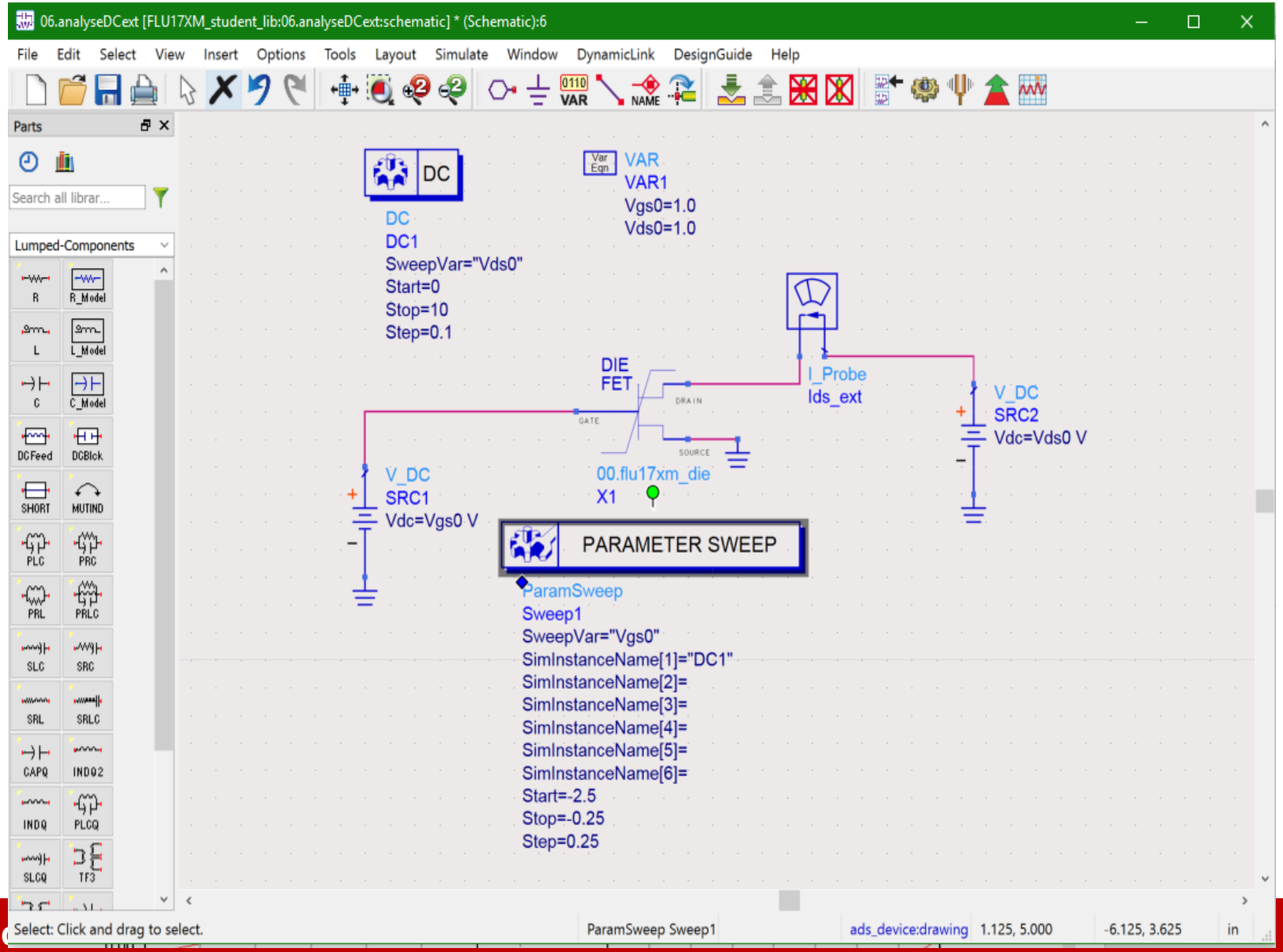


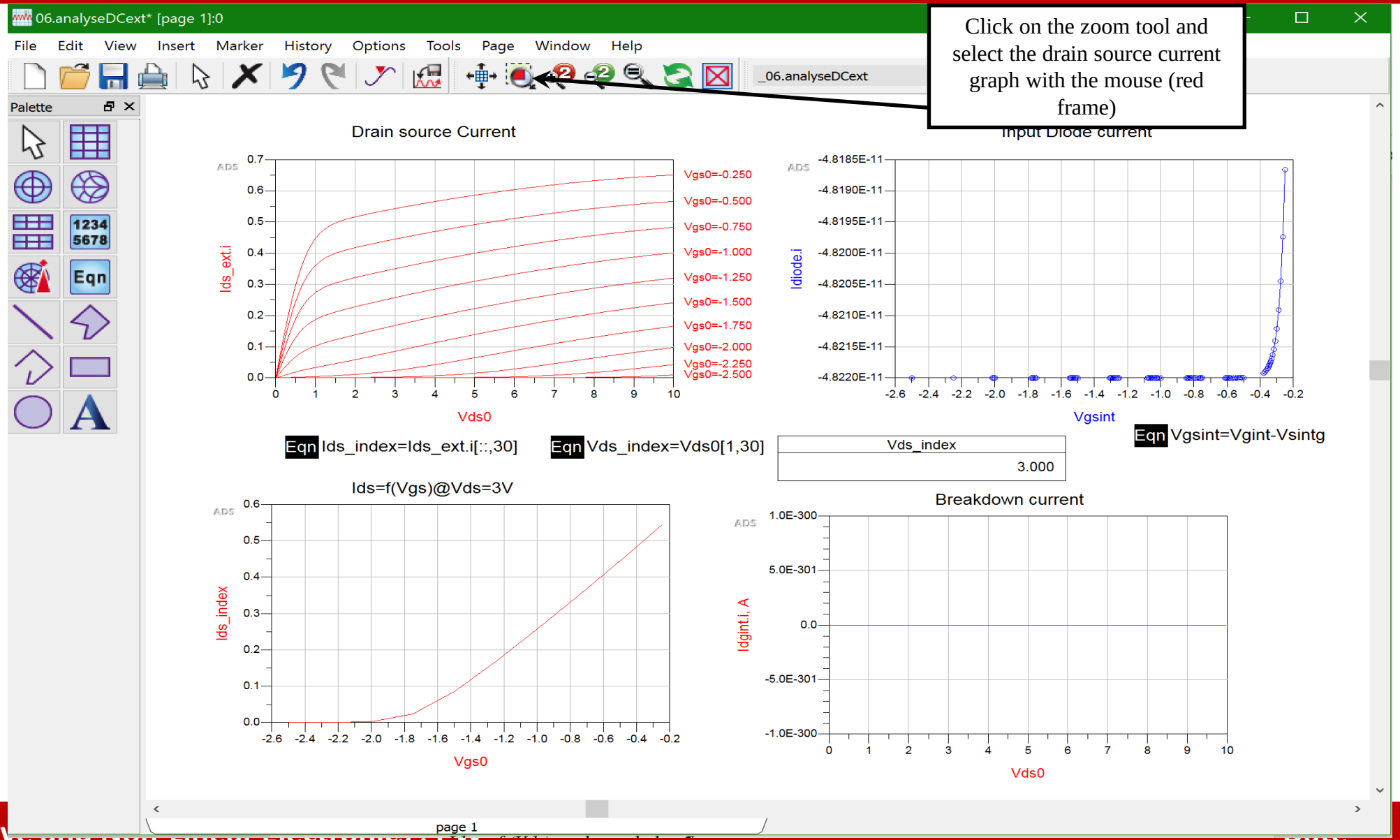




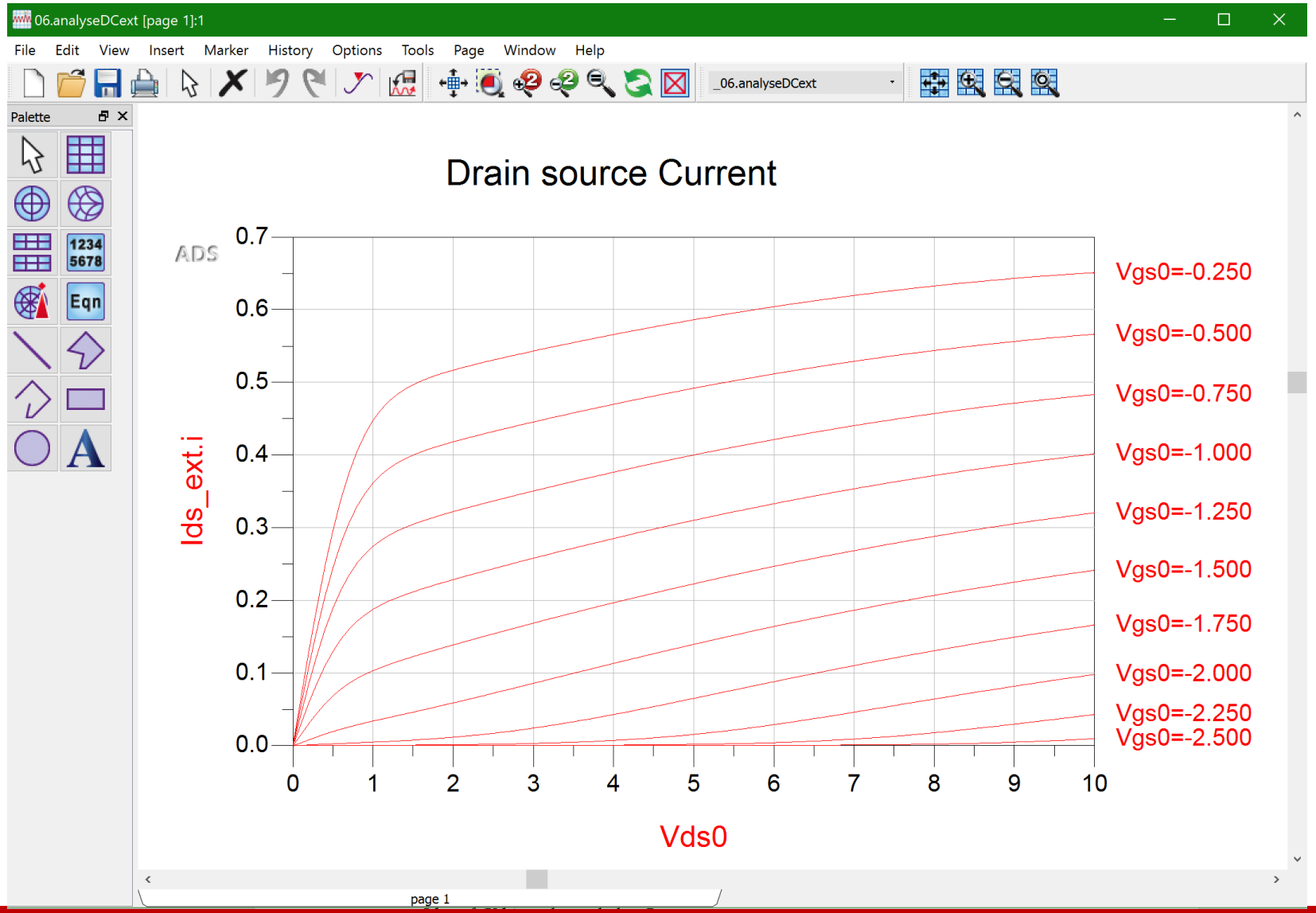




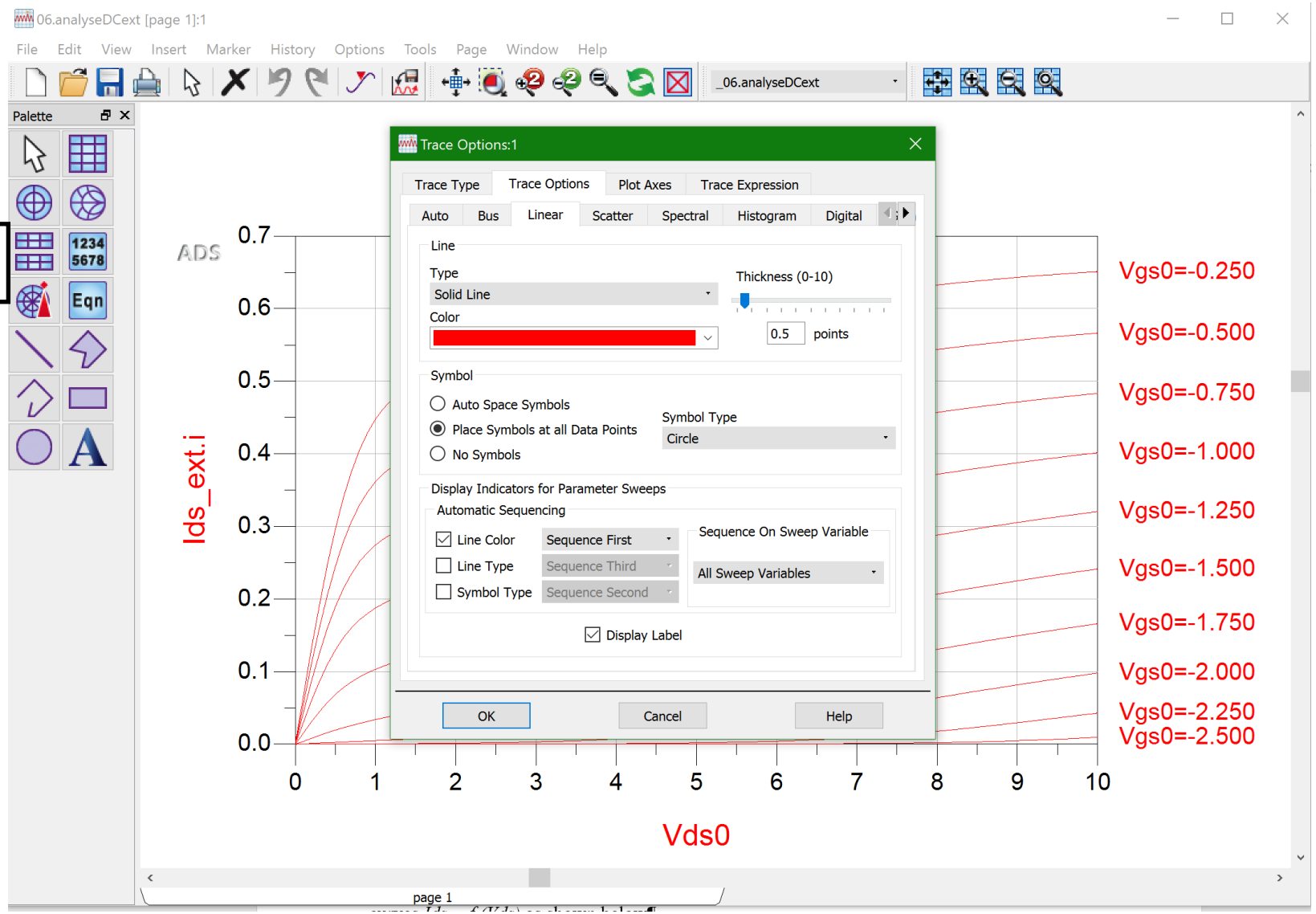


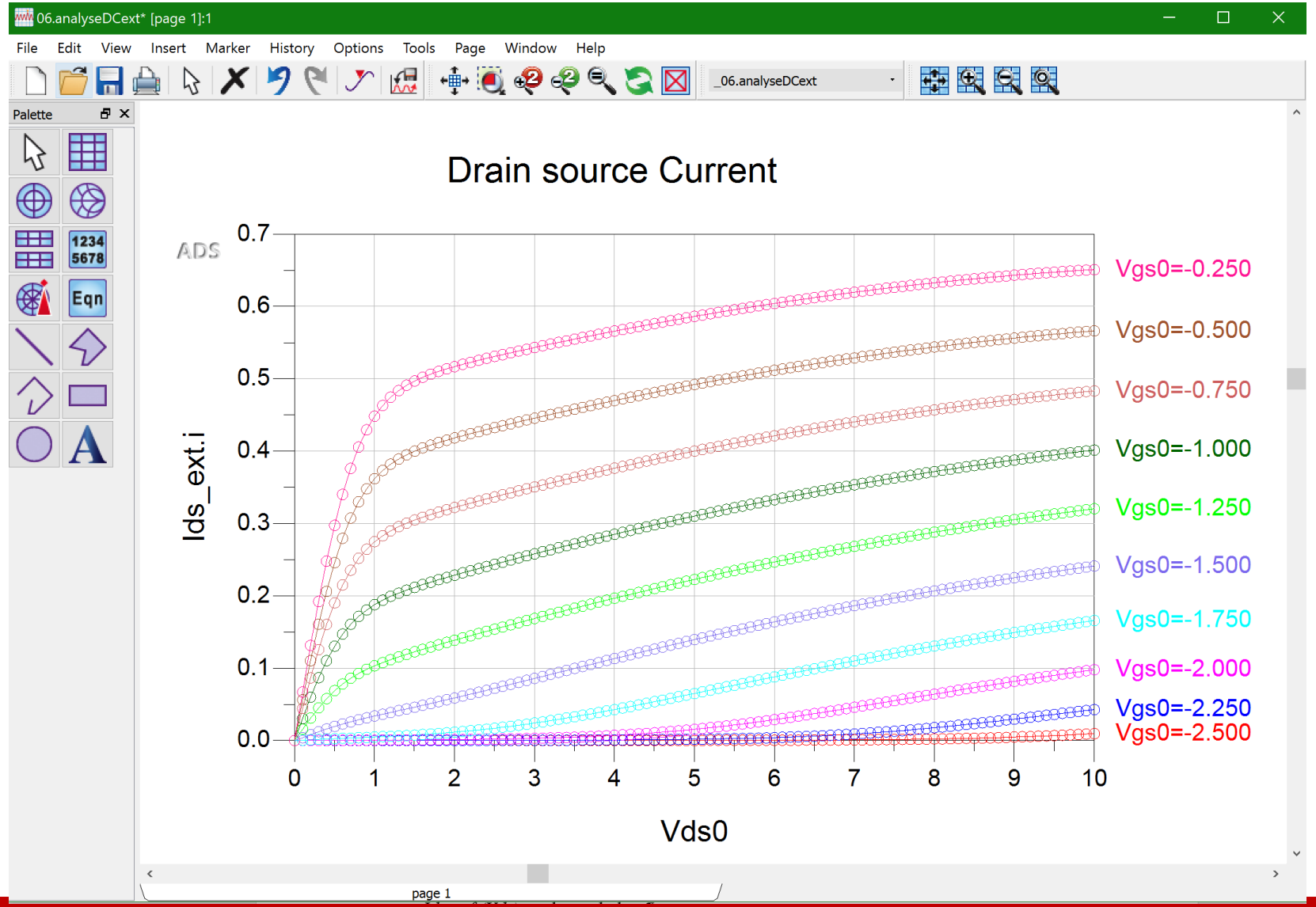


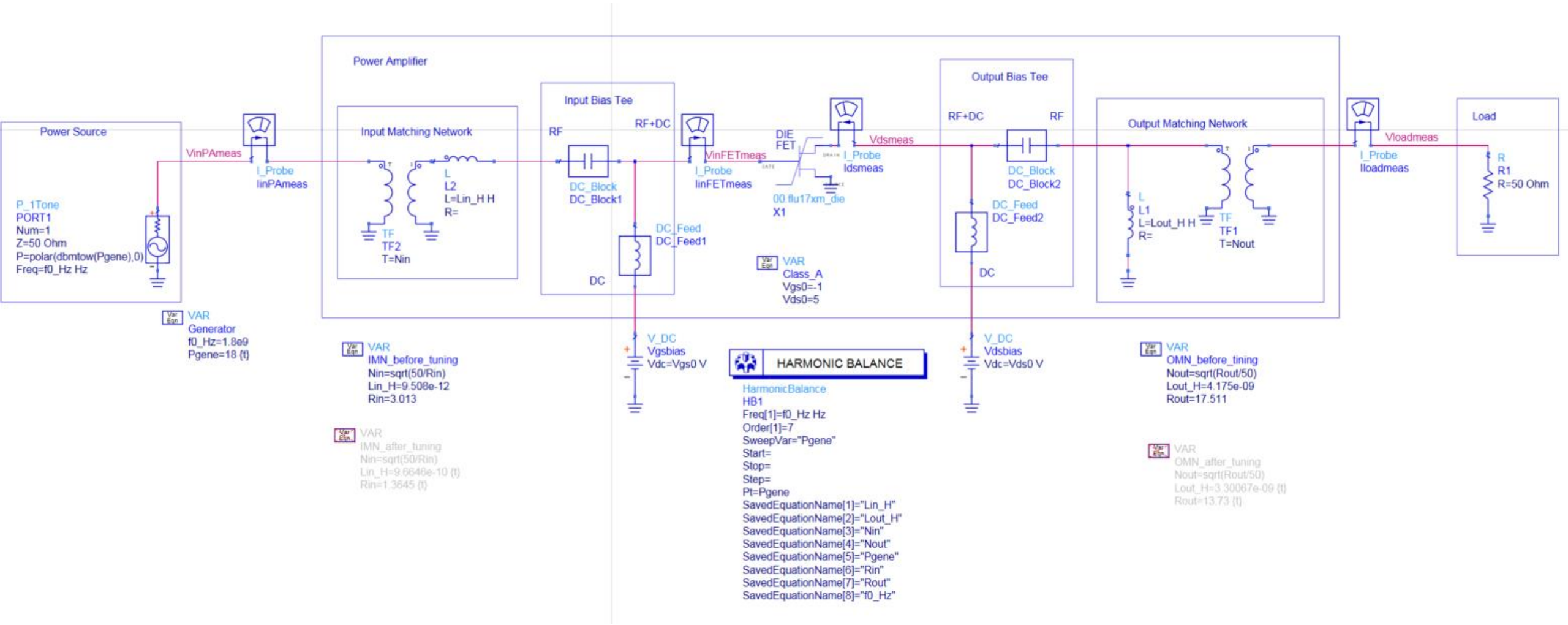
Click on the zoom tool and select the drain source current graph with the mouse (red frame)



Double Click on the curves to open the trace options window





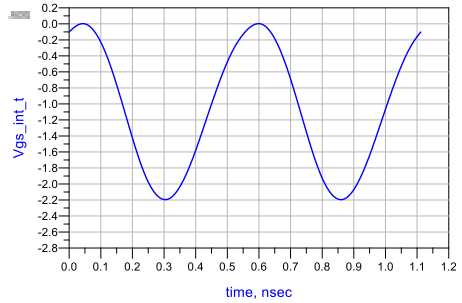


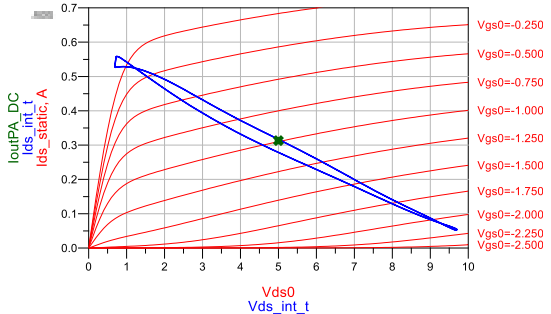
Eqn VinPA_f0=VinPAmeas[1]
Eqn VoutPA_f0=Vloadmeas[1]
Eqn IinPA_f0=IinPAmeas.i[1]
Eqn IoutPA_f0=Iloadmeas.i[1]
Eqn PinPA_f0_mW=0.5*1000*real(VinPA_f0*conj(IinPA_f0))
Eqn PinPA_f0_dBm=10*log10(PinPA_f0_mW)
Eqn PoutPA_f0_mW=0.5*1000*real(VoutPA_f0*conj(IoutPA_f0))
Eqn PoutPA_f0_dBm=10*log10(PoutPA_f0_mW)
Eqn PDC_mW=1000*(VinPA_DC*IinPA_DC+VoutPA_DC*IoutPA_DC)
Eqn GPA_f0=PoutPA_f0_mW/PinPA_f0_mW
Eqn GPA_f0_dB=10*log10(GPA_f0)
Eqn PaddPA_f0_mW=PoutPA_f0_mW-PinPA_f0_mW
Eqn PaddPA_f0_dBm=10*log10(PaddPA_f0_mW)
Eqn Drain_Efficiency=PoutPA_f0_mW/PDC_mW*100
Eqn PAE=PaddPA_f0_mW/PDC_mW*100
Eqn VinPA_DC=mag(VinFETmeas[0])
Eqn VoutPA_DC=mag(Vdsmeas[0])
Eqn IinPA_DC=mag(IinFETmeas.i[0])
Eqn IoutPA_DC=mag(Idsmeas.i[0])

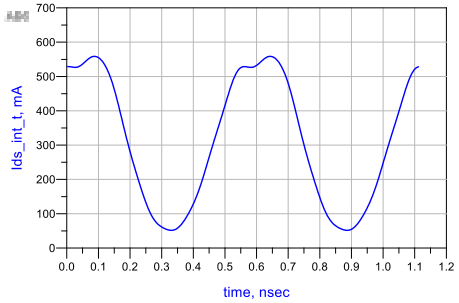
Eqn ZinFET_f0=VinFETmeas[1]/IinFETmeas.i[1]
Eqn ZinPA_f0=VinPAmeas[1]/IinPAmeas.i[1]
Eqn GamainFET_f0=(ZinFET_f0-50)/(ZinFET_f0+50)
Eqn GamainPA_f0=(ZinPA_f0-50)/(ZinPA_f0+50)

Pgene	PinPA_f0_mW	...tPA_f0_mW	GPA_f0	PDC_mW	...in_Efficiency	PAE
18.000	9.413	614.277	65.261	1565.602	39.236	38.635

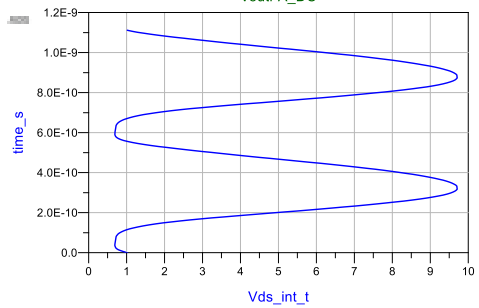
Pgene	...PA_f0_dBm	...tPA_f0_dBm	GPA_f0_dB	PDC_mW	...in_Efficiency	PAE
18.000	9.737	27.884	18.147	1565.602	39.236	38.635

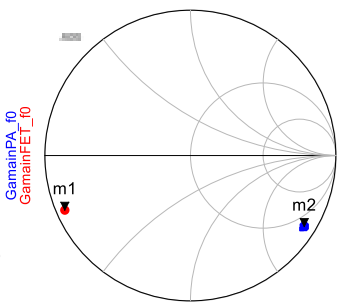






Eqn Ids_static="_06.analyseDCext"..Ids_int.i
Eqn time_s=indep(Vds_int_t[0,::])
Eqn Vds_int=Vdint-Vsintd
Eqn Vds_int_t=ts(Vds_int)
Eqn Vgs_int_t=ts(Vgint-Vsintg)
Eqn Ids_int_t=ts(Ids_int.i)





Pgene	ZinFET_f0	...ainFET_f0
18.000	1.572 - j10....	0.941 / -156...

Pgene	ZinPA_f0	...mainPA_f0
18.000	26.090 - j17...	0.922 / -31....

Pgene (18.000 to 18.000)

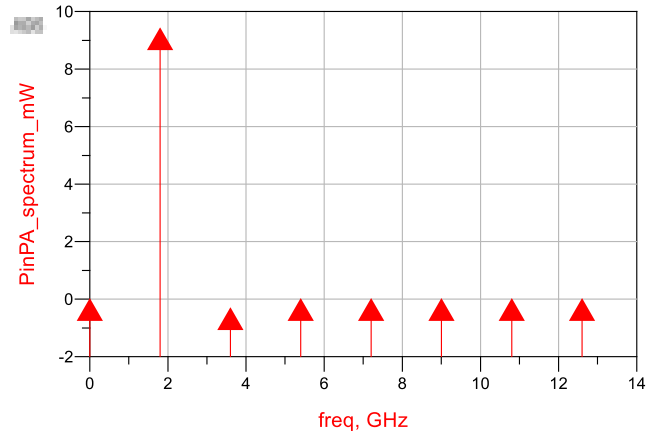
m1
Pgene=18.000
GamainFET_f0=0.941 / -156.472
impedance = Z0 * (0.031 - j0.208)

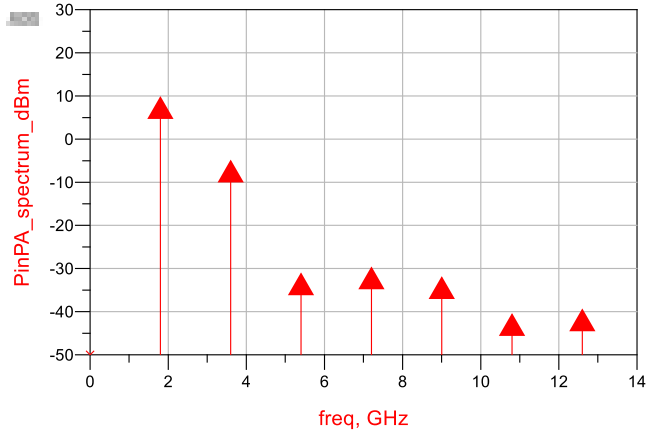
m2
Pgene=18.000
GamainPA_f0=0.922 / -31.974
impedance = Z0 * (0.522 - j3.417)

freq	PinPA_spectrum_mW
0.0000 Hz	0.000
1.800 GHz	9.413
3.600 GHz	-0.316
5.400 GHz	-0.001
7.200 GHz	-0.001
9.000 GHz	-0.001
10.80 GHz	-8.805E-5
12.60 GHz	-1.122E-4

Eqn PinPA_spectrum_mW=0.5*1000*real(VinPAmeas*conj(IinPAmeas.i))

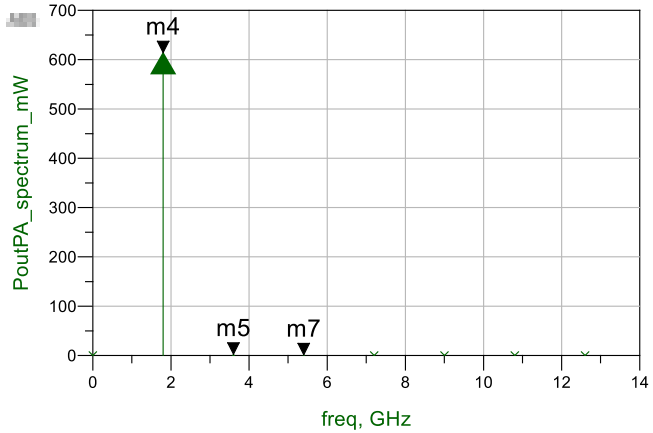
Eqn PinPA_spectrum_dBm=10*log10(mag(PinPA_spectrum_mW))





Eqn PoutPA_spectrum_mW=0.5*1000*real(Vloadmeas*conj(Iloadmeas.i))

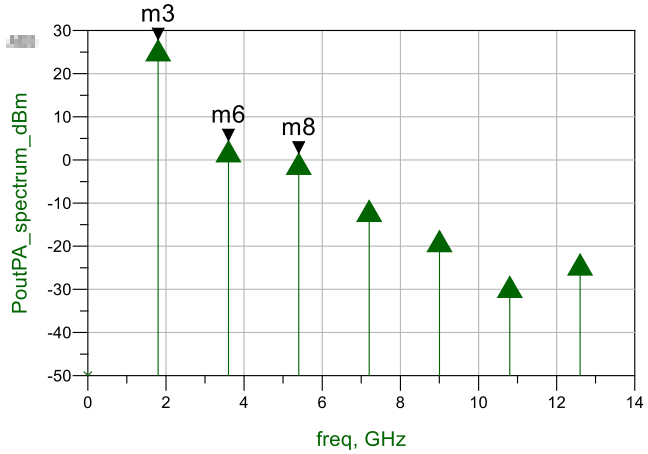
Eqn PoutPA_spectrum_dBm=10*log10(PoutPA_spectrum_mW)



m4
freq=1.800GHz
PoutPA_spectrum_mW=614.277
Pgene=18.000

m5
freq=3.600GHz
PoutPA_spectrum_mW=2.789
Pgene=18.000

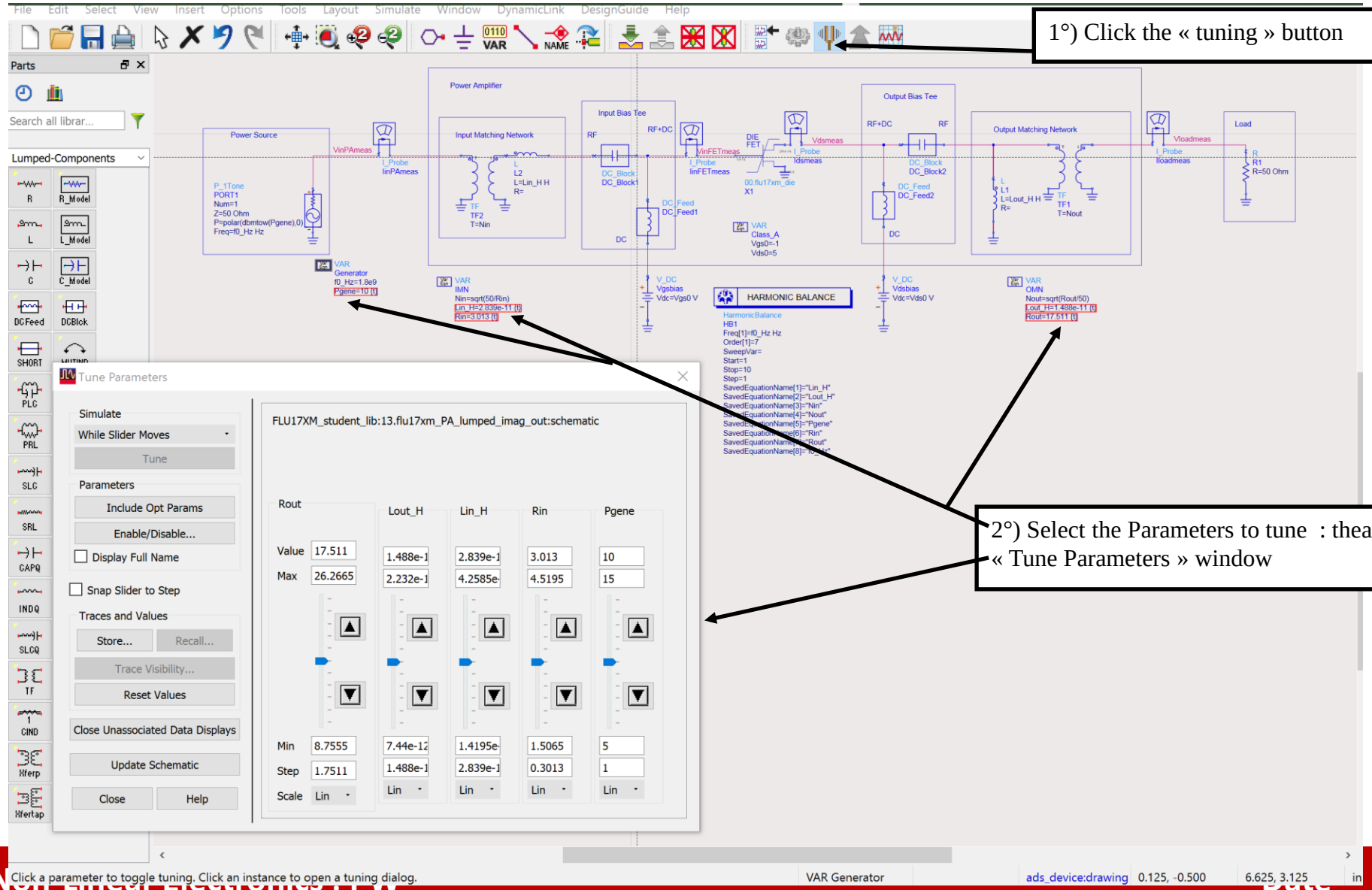
m7
freq=5.400GHz
PoutPA_spectrum_mW=1.437
Pgene=18.000



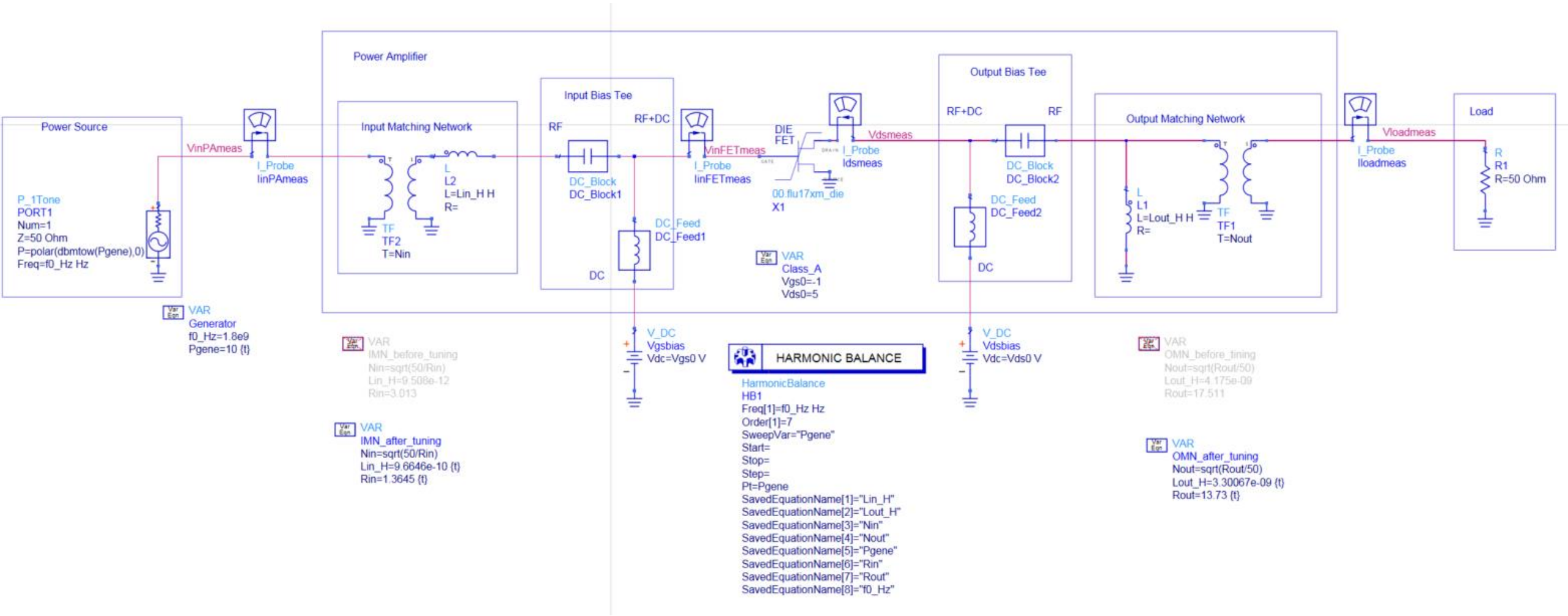
m3
freq=1.800GHz
PoutPA_spectrum_dBm=27.884
Pgene=18.000

m6
freq=3.600GHz
PoutPA_spectrum_dBm=4.454
Pgene=18.000

m8
freq=5.400GHz
PoutPA_spectrum_dBm=1.574
Pgene=18.000



After the tuning of the 5 parameters



Results after the tuning of the 5 parameters to maxipise the output power of the PA.

Method to tune the parameters

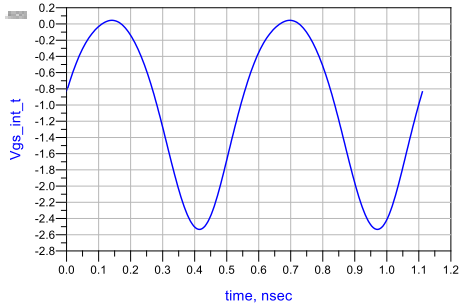
1) Tune first Lout_H
2) Tune Secondly Rout
3) Modify Pgene if required
4) Tune again Lout_H
5) Tune again Rout
6) Tune then Lin_H to cancel Imag(Zin)
7) Tune Secondly Rin (To match Zin=50Ω)
8) Modify Pgene if required
9) Tune again Lout_H if required
10) Tune again Rout if required
11) Tune again Lin_H to cancel Imag(Zin)
12) Tune again Rin (To match Zin=50Ω)
13) Repeat again step 8 to 12 to maximise the Gain or the output power or the PAE

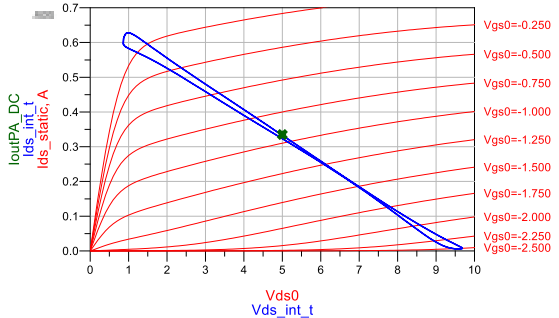
```
Eqn VinPA_f0=VinPAmeas[1]
Eqn VoutPA_f0=Vloadmeas[1]
Eqn VinPA_DC=mag(VinFETmeas[0])
Eqn VoutPA_DC=mag(Vdsmeas[0])
Eqn IinPA_f0=IinPAmeas.i[1]
Eqn IoutPA_f0=Iloadmeas.i[1]
Eqn IinPA_DC=mag(IinFETmeas.i[0])
Eqn IoutPA_DC=mag(Idsmeas.i[0])
Eqn PinPA_f0_mW=0.5*1000*real(VinPA_f0*conj(IinPA_f0))
Eqn PinPA_f0_dBm=10*log10(PinPA_f0_mW)
Eqn PoutPA_f0_mW=0.5*1000*real(VoutPA_f0*conj(IoutPA_f0))
Eqn PoutPA_f0_dBm=10*log10(PoutPA_f0_mW)
Eqn PDC_mW=1000*(VinPA_DC*IinPA_DC+VoutPA_DC*IoutPA_DC)
Eqn GPA_f0=PoutPA_f0_mW/PinPA_f0_mW
Eqn GPA_f0_dB=10*log10(GPA_f0)
Eqn PaddPA_f0_mW=PoutPA_f0_mW-PinPA_f0_mW
Eqn PaddPA_f0_dBm=10*log10(PaddPA_f0_mW)
Eqn Drain_Efficiency=PoutPA_f0_mW/PDC_mW*100
Eqn PAE=PaddPA_f0_mW/PDC_mW*100
```

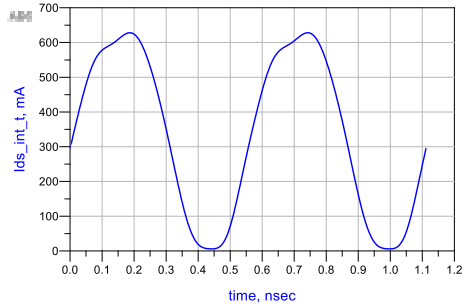
```
Eqn ZinFET_f0=VinFETmeas[1]/IinFETmeas.i[1]
Eqn ZinPA_f0=VinPAmeas[1]/IinPAmeas.i[1]
Eqn GamainFET_f0=(ZinFET_f0-50)/(ZinFET_f0+50)
Eqn GamainPA_f0=(ZinPA_f0-50)/(ZinPA_f0+50)
```

Pgene	PinPA_f0_mW	...tPA_f0_mW	GPA_f0	PDC_mW	...in_Efficiency	PAE
10.000	10.000	733.199	73.320	1675.090	43.771	43.174

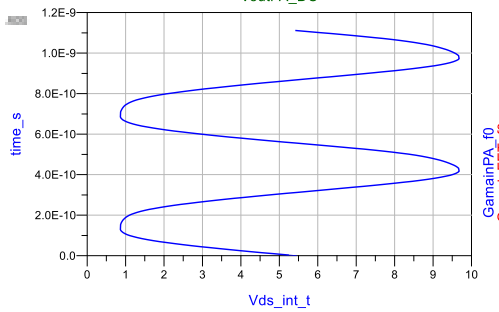
Pgene	...PA_f0_dBm	...tPA_f0_dBm	GPA_f0_dB	PDC_mW	...in_Efficiency	PAE
10.000	10.000	28.652	18.652	1675.090	43.771	43.174

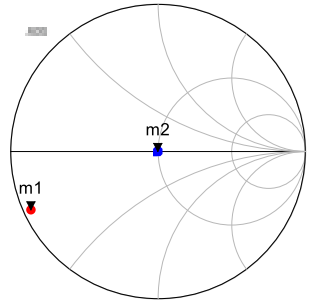







```
Eqn Ids_static="_06.analyseDCext"..Ids_int.i
Eqn time_s=indep(Vds_int_t[0,::])
Eqn Vds_int=Vdint-Vsintd
Eqn Vds_int_t=ts(Vds_int)
Eqn Vgs_int_t=ts(Vgint-Vsintg)
Eqn Ids_int_t=ts(Ids_int.i)
```





Pgene	ZinFET_f0	...ainFET_f0
10.000	1.359 - j10....	0.949 / -155...

Pgene	ZinPA_f0	...mainPA_f0
10.000	49.810 + j0....	0.002 / 177....

Pgene (10.000 to 10.000)

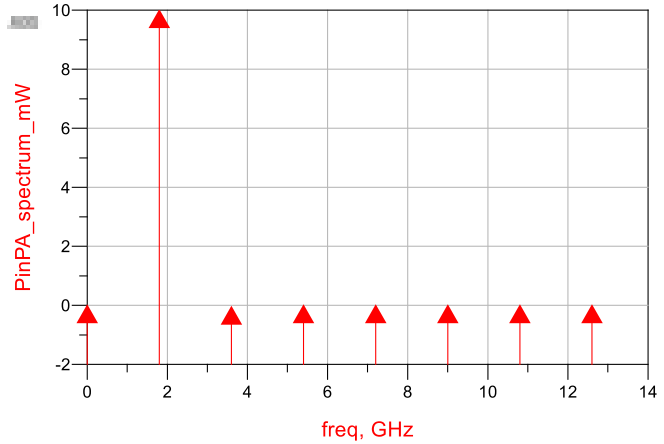
m1
Pgene=10.000
GamainFET_f0=0.949 / -155.321
impedance = Z0 * (0.027 - j0.219)

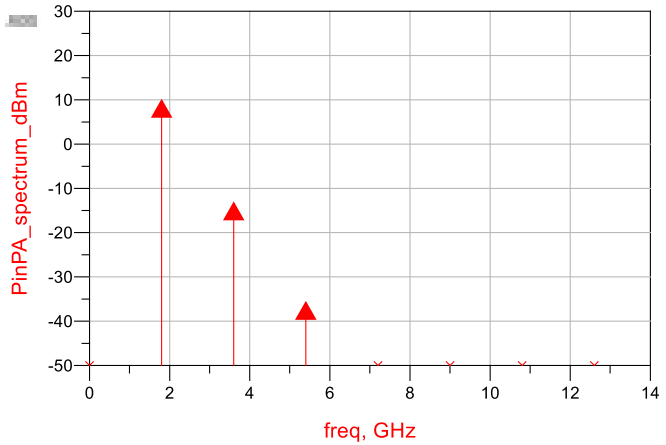
m2
Pgene=10.000
GamainPA_f0=0.002 / 177.310
impedance = Z0 * (0.996 + j1.780E-4)

freq	PinPA_spectrum_mW
0.0000 Hz	0.000
1.800 GHz	10.000
3.600 GHz	-0.048
5.400 GHz	-2.714E-4
7.200 GHz	-9.894E-7
9.000 GHz	-1.955E-6
10.80 GHz	-1.466E-7
12.60 GHz	-8.569E-12

Eqn PinPA_spectrum_mW=0.5*1000*real(VinPAmeas*conj(IinPAmeas.i))

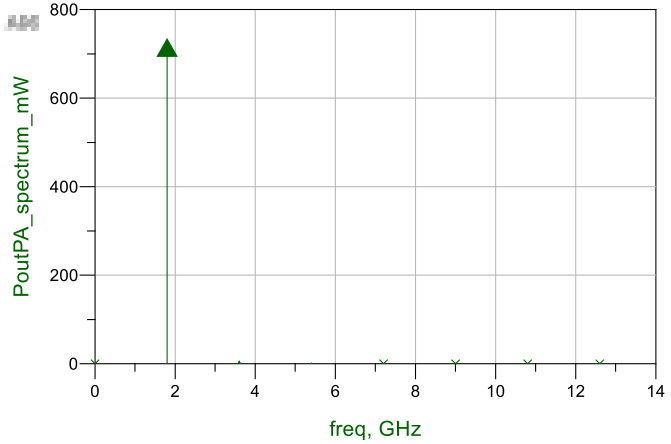
Eqn PinPA_spectrum_dBm=10*log10(mag(PinPA_spectrum_mW))

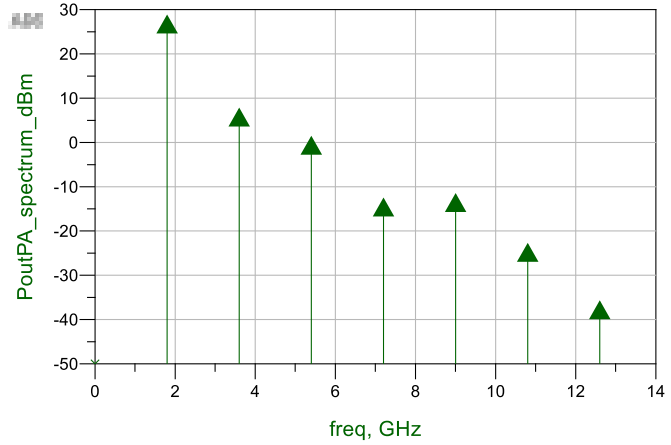




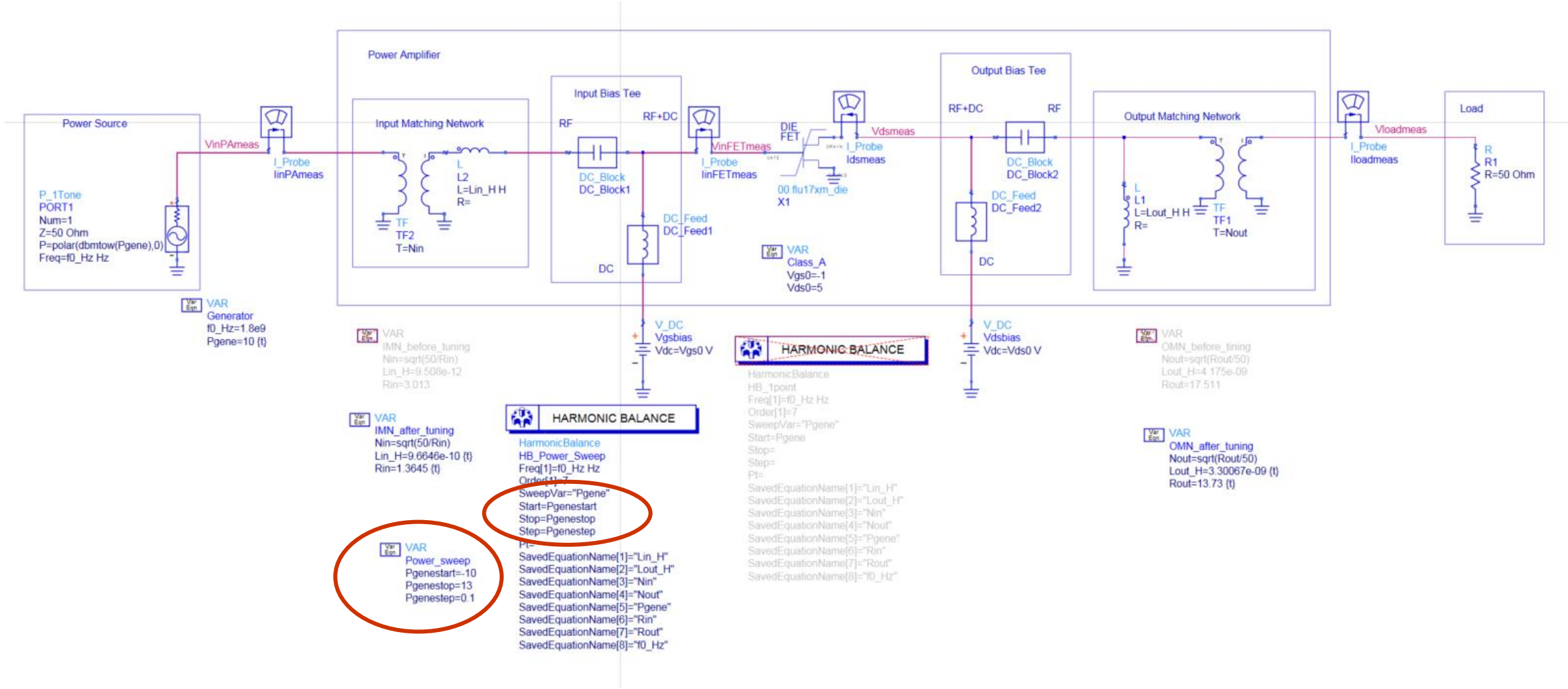
Eqn PoutPA_spectrum_mW=0.5*1000*real(Vloadmeas*conj(Iloadmeas.i)

Eqn PoutPA_spectrum_dBm=10*log10(PoutPA_spectrum_mW)





With a power sweep of the generator Power



Eqn VinPA_f0=VinPAmeas[1]
Eqn VoutPA_f0=Vloadmeas[1]
Eqn VinPA_DC=mag(VinFETmeas[0])
Eqn VoutPA_DC=mag(Vdsmeas[0])
Eqn IinPA_f0=IinPAmeas.i[1]
Eqn IoutPA_f0=Iloadmeas.i[1]
Eqn IinPA_DC=mag(IinFETmeas.i[0])
Eqn IoutPA_DC=mag(Idsmeas.i[0])

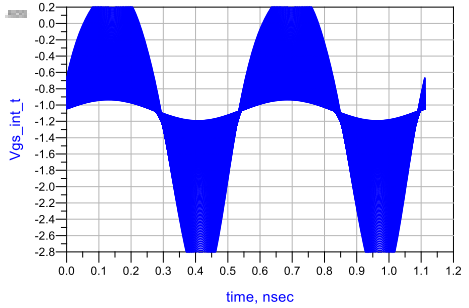
Eqn PinPA_f0_mW=0.5*1000*real(VinPA_f0*conj(IinPA_f0))
Eqn PinPA_f0_dBm=10*log10(PinPA_f0_mW)
Eqn PoutPA_f0_mW=0.5*1000*real(VoutPA_f0*conj(IoutPA_f0))
Eqn PoutPA_f0_dBm=10*log10(PoutPA_f0_mW)
Eqn PDC_mW=1000*(VinPA_DC*IinPA_DC+VoutPA_DC*IoutPA_DC)

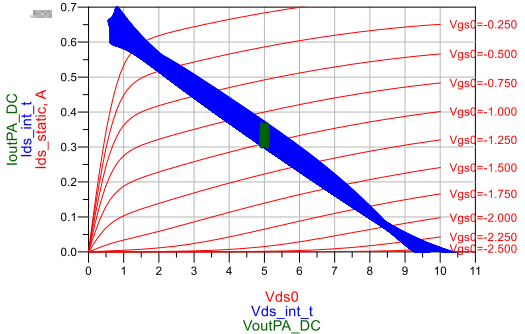
Eqn GPA_f0=PoutPA_f0_mW/PinPA_f0_mW
Eqn GPA_f0_dB=10*log10(GPA_f0)
Eqn PaddPA_f0_mW=PoutPA_f0_mW-PinPA_f0_mW
Eqn PaddPA_f0_dBm=10*log10(PaddPA_f0_mW)
Eqn Drain_Efficiency=PoutPA_f0_mW/PDC_mW*100
Eqn PAE=PaddPA_f0_mW/PDC_mW*100

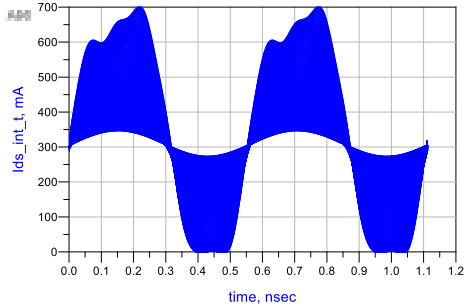
Eqn ZinFET_f0=VinFETmeas[1]/IinFETmeas.i[1]
Eqn ZinPA_f0=VinPAmeas[1]/IinPAmeas.i[1]
Eqn GamainFET_f0=(ZinFET_f0-50)/(ZinFET_f0+50)
Eqn GamainPA_f0=(ZinPA_f0-50)/(ZinPA_f0+50)

Pgene	PinPA_f0_mW	...tPA_f0_mW	GPA_f0	PDC_mW	...in_Efficiency	PAE
-10.000	0.099	9.234	92.811	1551.479	0.595	0.589

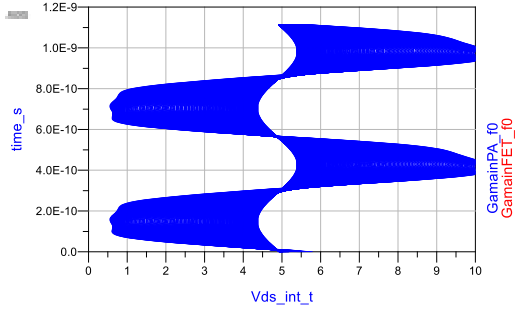
Pgene	...PA_f0_dBm	...tPA_f0_dBm	GPA_f0_dB	PDC_mW	...in_Efficiency	PAE
-10.000	-10.022	9.654	19.676	1551.479	0.595	0.589

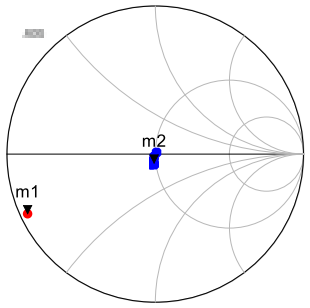






Eqn Ids_static="_06.analyseDCext"..Ids_int.i
Eqn time_s=indep(Vds_int_t[0,::])
Eqn Vds_int=Vdint-Vsintd
Eqn Vds_int_t=ts(Vds_int)
Eqn Vgs_int_t=ts(Vgint-Vsintg)
Eqn Ids_int_t=ts(Ids_int.i)





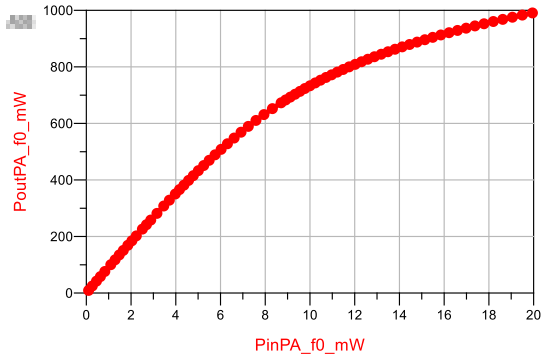
Pgene	ZinFET_f0	...ainFET_f0
-10.000	1.332 - j11....	0.951 / -154...
-9.900	1.332 - j11....	0.951 / -154...

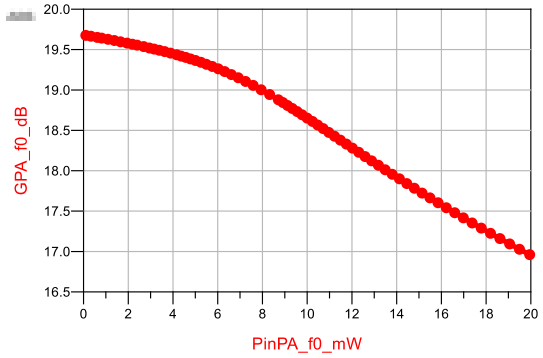
Pgene	ZinPA_f0	...mainPA_f0
-10.000	48.798 - j6....	0.071 / -95....
-9.900	48.798 - j6....	0.071 / -95....

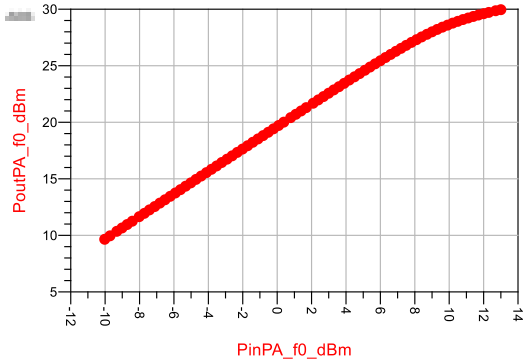
Pgene (-10.000 to 13.000)

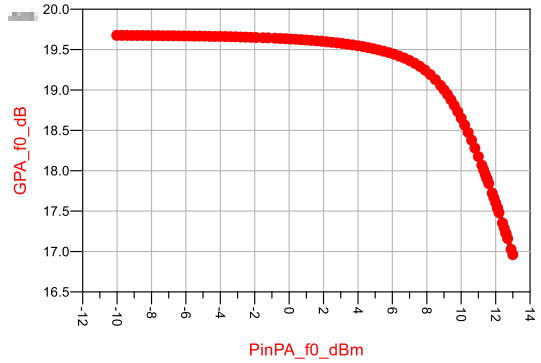
m1
Pgene=-1.879E-14
GamainFET_f0=0.950 / -154.969
impedance = Z0 * (0.027 - j0.222)

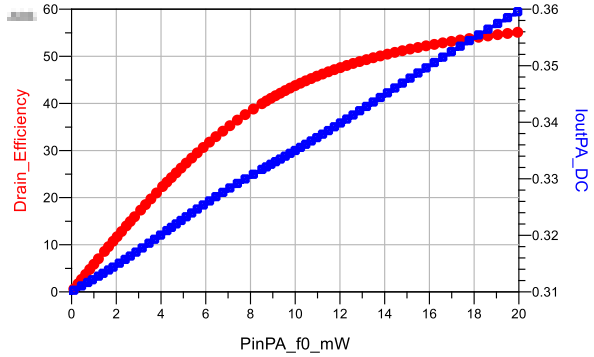
m2
Pgene=-1.879E-14
GamainPA_f0=0.061 / -97.558
impedance = Z0 * (0.977 - j0.118)

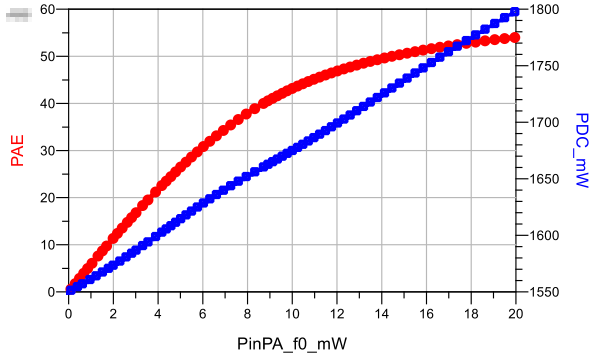


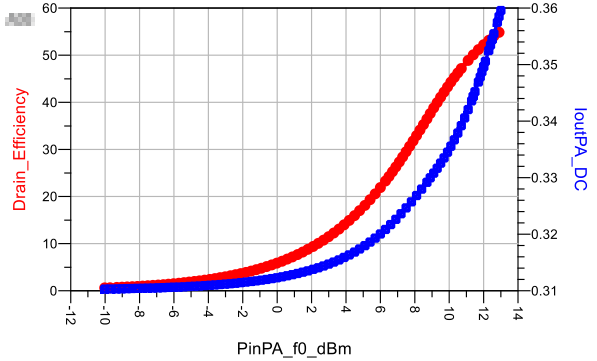


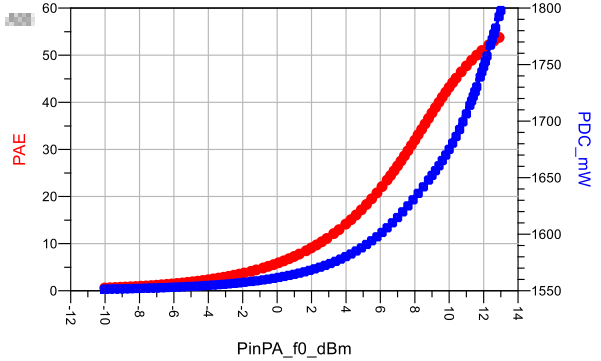








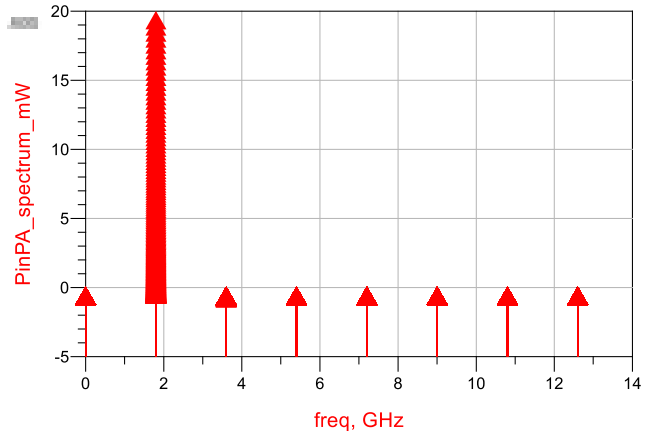


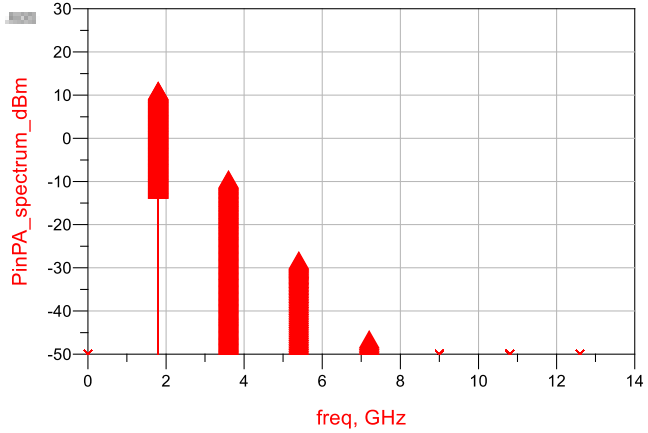


freq	PinPA_spectrum_mW
Pgene=-10.000	
0.0000 Hz	0.000
1.800 GHz	0.099
3.600 GHz	-4.957E-6
5.400 GHz	-8.445E-11
7.200 GHz	-2.161E-14
9.000 GHz	-3.669E-17
10.80 GHz	-3.145E-20
12.60 GHz	-1.887E-23
Pgene=-9.900	
0.0000 Hz	0.000
1.800 GHz	0.102
3.600 GHz	-5.190E-6
5.400 GHz	-9.050E-11
7.200 GHz	-2.370E-14
9.000 GHz	-4.117E-17
10.80 GHz	-3.611E-20
12.60 GHz	-2.220E-23
Pgene=-9.800	
0.0000 Hz	0.000
1.800 GHz	0.104
3.600 GHz	-5.435E-6
5.400 GHz	-9.698E-11
7.200 GHz	-2.600E-14
9.000 GHz	-4.620E-17
10.80 GHz	-4.145E-20
12.60 GHz	-2.610E-23

Eqn PinPA_spectrum_mW=0.5*1000*real(VinPAmeas*conj(IinPAmeas.i))

Eqn PinPA_spectrum_dBm=10*log10(mag(PinPA_spectrum_mW))





Eqn PoutPA_spectrum_mW=0.5*1000*real(Vloadmeas*conj(Iloadmeas.i)

Eqn PoutPA_spectrum_dBm=10*log10(PoutPA_spectrum_mW)