
**Cairo University** **Faculty of Engineering**

# Advanced Topics in Electronics-1

**Under supervision of:**

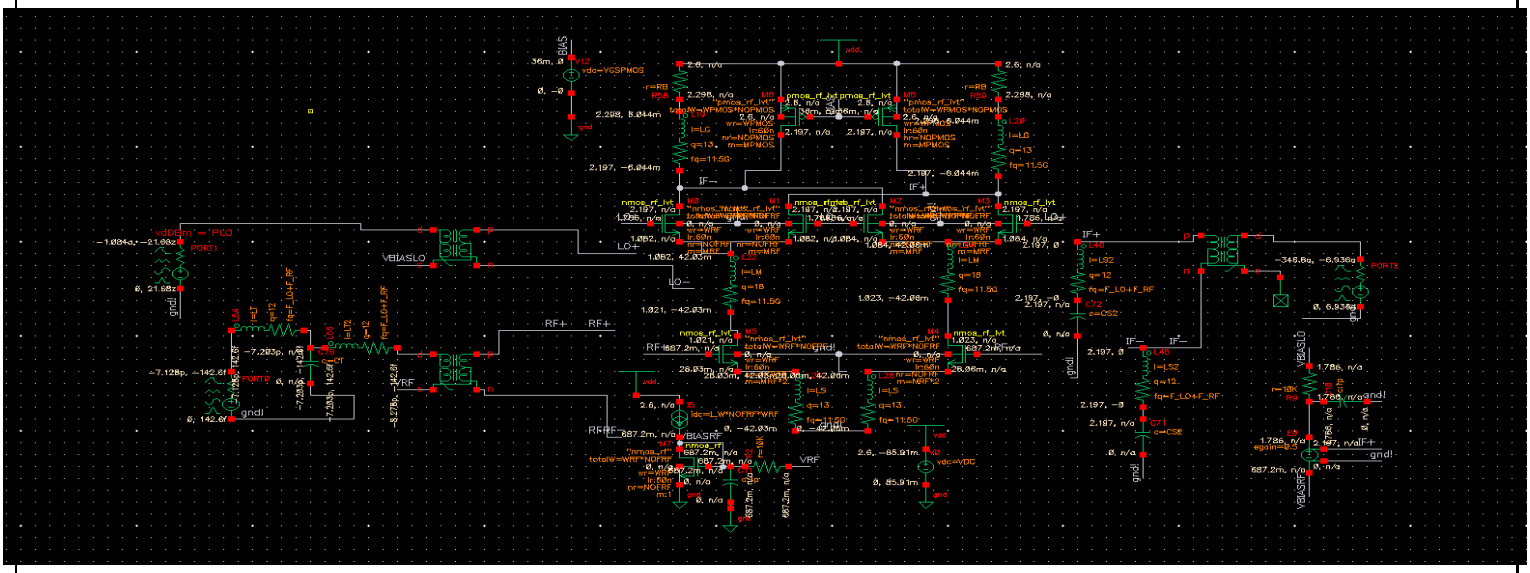
**Dr. Mohamed mobarak**

**Eng. Ahmed Atef**

**Students Names:**

**Kerolos Hanna Fayez 1200049**

## Schematic of design including the Voltages and currents

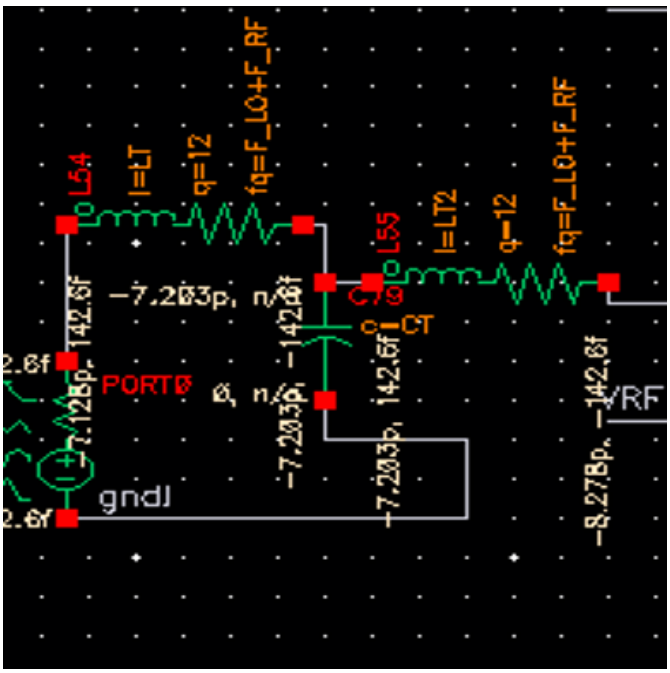


| Global Variables | |
|---|---|
| I_W | 95 |
| MRF | 6 |
| NOFRF | 32 |
| VDC | 2.6 |
| WLO | 600n |
| WRF | 600n |
| WTAIL | 600n |
| VGSPMOS | 600m |
| F_LO | F_RF-1 |
| F_RF | 12G |
| I_WLO | 50 |
| Pin | -40 |
| RD | 1 |
| WPMOS | 600n |
| NOBIAS | 4 |
| PLO | 0 |
| LG | 3n |
| LS | 120p |
| CS | 200f |
| RE | 100 |
| NPMOS | 32 |
| WP | 1 |
| MPMOS | 5 |
| IDC | 1u |
| RB | 50 |
| CB | 10f |
| RBIAS | 100 |
| NOPMOS | 32 |

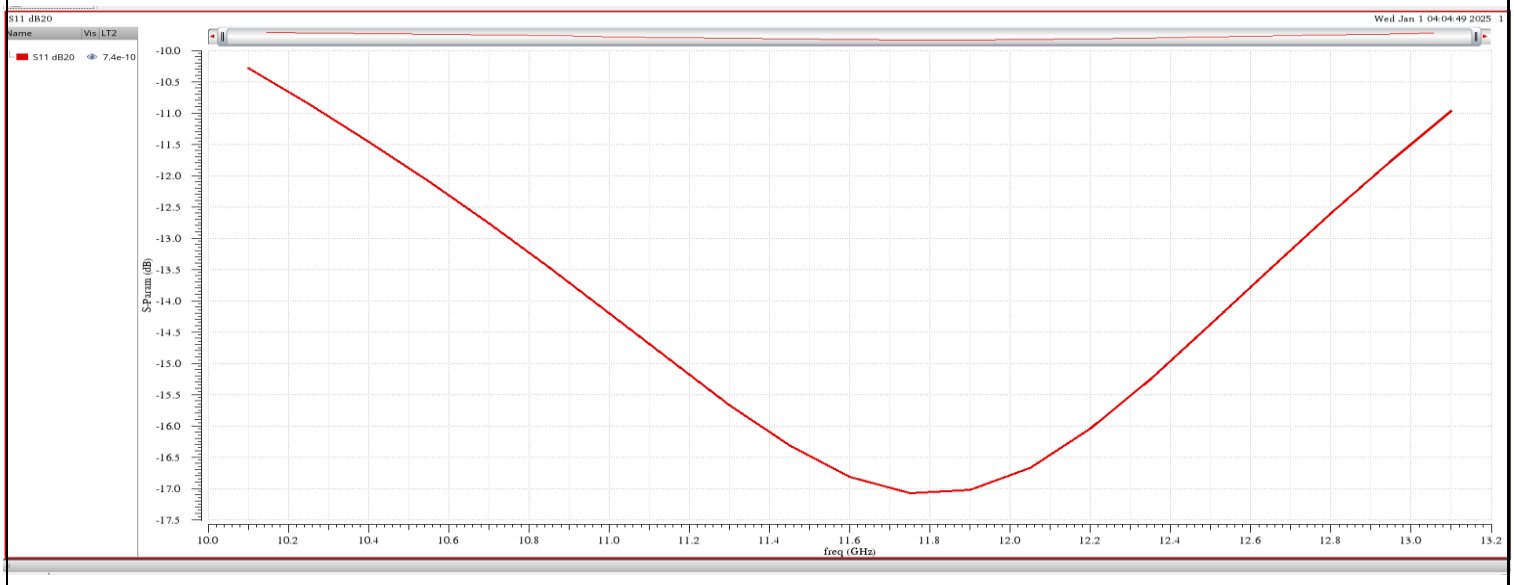
| | |
|---|---|
| RBIAS | 100 |
| NOPMOS | 32 |
| LM | 360p |
| CL | 400f |
| LE | 1 |
| NOFR | 1 |
| CS2 | 41f |
| LS2 | 1.8n |
| F_IF | 1:1G:3G |
| CP | 10f |
| VGSBIAS | 36m |
| LR | 300p |
| NOFBIAS | 32 |
| CT | 300f |
| LT | 50p |
| LT2 | 720p |
| F_RF2 | 12G |
| F_RFtwo | F_RF+1 |

All variable used in the schematic with sizing

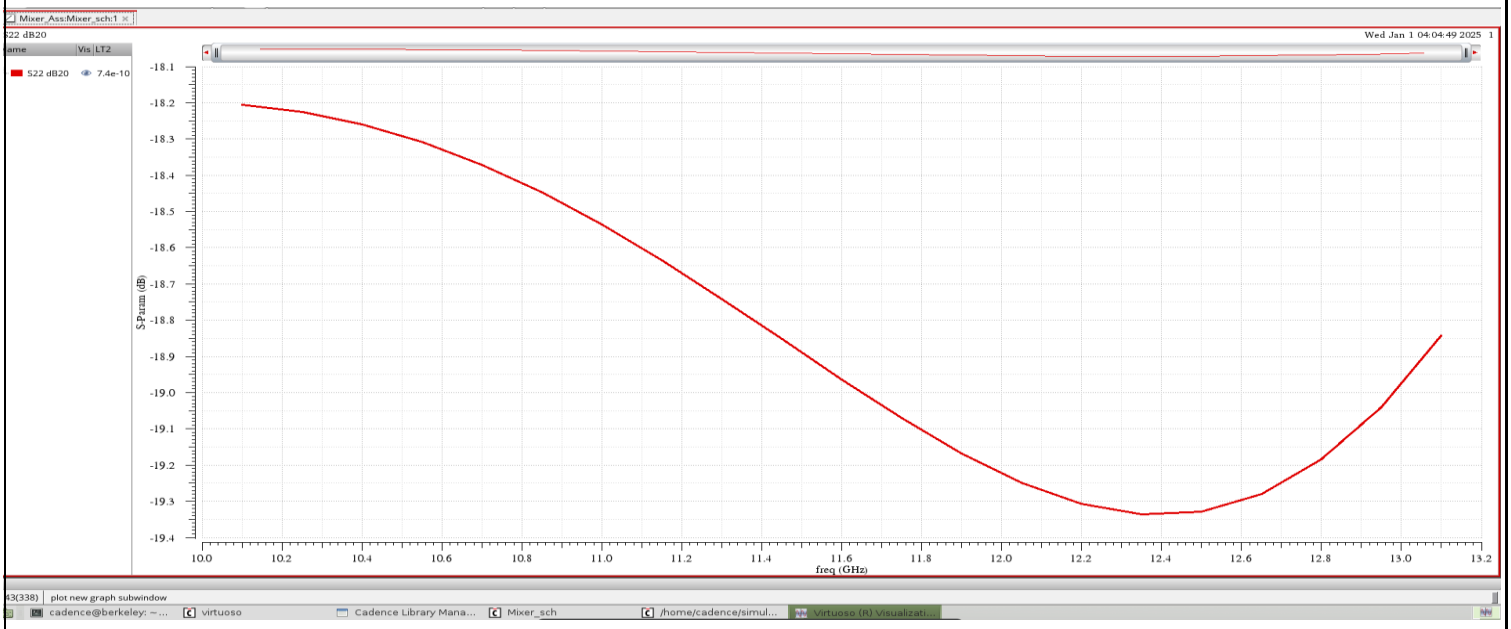
In the previous design I used current bleeding technique using a pmos to increase linearity and decrease current drop on the resistance , I used a matching network of series inductor ,shunt cap, series inductor .



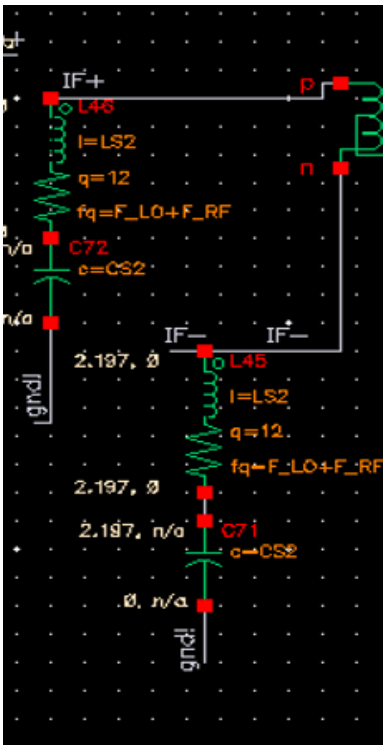

To achieve a S11 under -10 , the following graph was drawn against RF frequency



The S22 was achieved by a resistive load of 50 ohm , choking inductor to increase gain.




It was required to have IF=RF-LO , so I used a shunt inductor and shunt capacitor at IF outputs to filter the RF+LO.



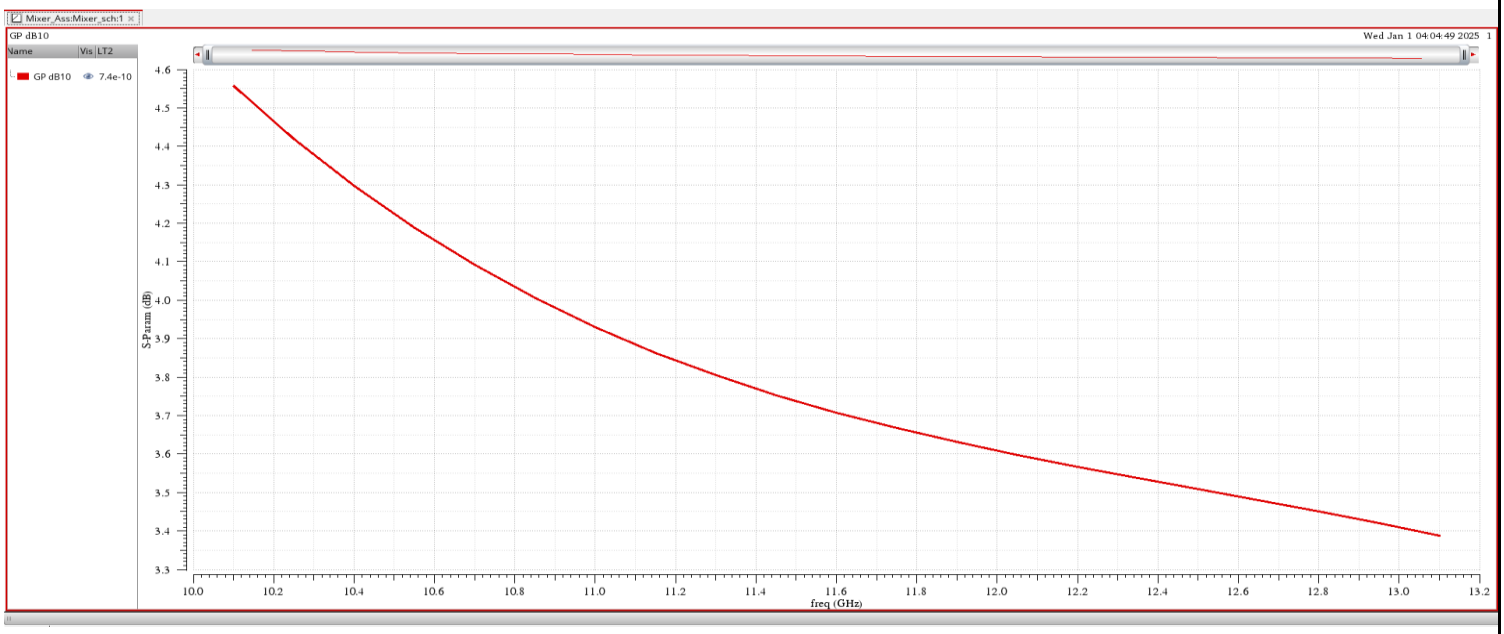
| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Mixer_Ass:Mixer_sch:1 | NFmin | expr | getData("NFmin" ?result "hbsp") | point | ☐ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | H21 dB20 | expr | db(hp(2 1 ?result "hbsp")) | point | ☐ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | GP dB10 | expr | db10(gp(sp(1 1 ?result "hbsp") sp(1 2 ?result "hbsp") sp(2 1 ?result "hbsp") sp(2 2 ?result "hbsp"))) | point | ☑ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | Gmax dB10 | expr | db10(gmax(sp(1 1 ?result "hbsp") sp(1 2 ?result "hbsp") sp(2 1 ?result "hbsp") sp(2 2 ?result "hbsp"))) | point | ☐ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | NF | expr | getData("NF" ?result "hbsp") | point | ☑ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | S22 dB20 | expr | db(spm('hbsp 2 2)) | point | ☑ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | S11 dB20 | expr | db(spm('hbsp 1 1)) | point | ☑ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | Gmin | expr | gmin(getData("Gopt" ?result "hbsp") getData("Bopt" ?result "hbsp") zref(1 ?result "hbsp")) | point | ☐ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | | expr | xval(value(db10(getData("Fmin" ?result "sp_noise")) 1.3e+10)) | point | ☐ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | | expr | waveVsWave(?x xval(getData("NFmin" ?result "hbsp")) ?y getData("NFmin" ?result "hbsp")) | point | ☐ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | | expr | waveVsWave(?x (xval(getData("NFmin" ?result "hbsp")) - 1.2e+10) ?y getData("NFmin" ?result "hbsp")) | point | ☐ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | p(/PORT0/PLUS /net019); hb_mt dBmP | expr | dbm(pvi("hb "/net019" 0 "/PORT0/PLUS" 0)) | point | ☐ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | p(/PORT0/PLUS /net019) F_RF=14G; hb_mt dBmP | expr | dbm(value(pvi("hb "/net019" 0 "/PORT0/PLUS" 0) 'F_RF 1.4e+10)) | point | ☐ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | ZM 3 reOhms | expr | real(zm(3 ?result "hbsp")) | point | ☐ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | ZM 2 reOhms | expr | real(zm(2 ?result "hbsp")) | point | ☐ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | ZM 1 reOhms | expr | real(zm(1 ?result "hbsp")) | point | ☑ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | ZM 3 imOhms | expr | imag(zm(3 ?result "hbsp")) | point | ☐ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | ZM 2 imOhms | expr | imag(zm(2 ?result "hbsp")) | point | ☐ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | S23 mag | expr | spm('hbsp 2 3) | point | ☐ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | S33 mag | expr | spm('hbsp 3 3) | point | ☐ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | S33 dB20 | expr | db(spm('hbsp 3 3)) | point | ☐ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | ipnCurves | expr | ipnVRICurves(v("/net037" ?result "hb_mt_fi") '(2 -1) '(1 0) ?iport i("/PORT2/PLUS" ?result "hb_mt_fi") ?epoint -40) | point | ☐ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | Input Referred IP3 Point | expr | ipnVRI(v("/net037" ?result "hb_mt_fi") '(2 -1) '(1 0) ?iport i("/PORT2/PLUS" ?result "hb_mt_fi") ?epoint -40) | point | ☐ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | 1st Order freq | expr | "cadar(setof(x harmonicFreqList(?result \"hb_mt_fi\") equal(car(x) '(1 0))))" | point | ☐ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | | signal | /PORT2/PLUS | point | ☐ | ☑ | |
| Mixer_Ass:Mixer_sch:1 | compressionCurves | expr | compressionVRICurves(v("/net037" ?result "hb_mt_fi") '(1 0) ?iport i("/PORT2/PLUS" ?result "hb_mt_fi") ?epoint -40 ?gcomp 1) | point | ☐ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | Input Referred 1dB Compression Point | expr | compressionVRI(v("/net037" ?result "hb_mt_fi") '(1 0) ?iport i("/PORT2/PLUS" ?result "hb_mt_fi") ?epoint -40 ?gcomp 1) | point | ☐ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | Input Referred IP3 Points | expr | ipnVRI(v("/net037" ?result "hb_mt_fi") '(2 -1) '(1 0) ?iport i("/PORT2/PLUS" ?result "hb_mt_fi") ?epoint -20 ?psweep nil) | point | ☐ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | p(/PORT2/PLUS /net037) h=1,0; hb_mt dBmP | expr | dbm(pvi("hb "/net037" 0 "/PORT2/PLUS" 0 '((1 0)))) | point | ☐ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | Input-referred | expr | _drplCompressionPointCurve("Input-referred") | point | ☑ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | Input Referred 1dB Compression Points | expr | compressionVRI(v("/net037" ?result "hb_mt_fi") '(0 1) ?iport i("/PORT2/PLUS" ?result "hb_mt_fi") ?epoint -30 ?gcomp 1) | point | ☐ | ☐ | |
| Mixer_Ass:Mixer_sch:1 | p(/PORT2/PLUS /net037) Pin=-30; hb_mt dBmP | expr | dbm(value(pvi("hb "/net037" 0 "/PORT2/PLUS" 0) 'Pin -30)) | point | ☐ | ☐ | |

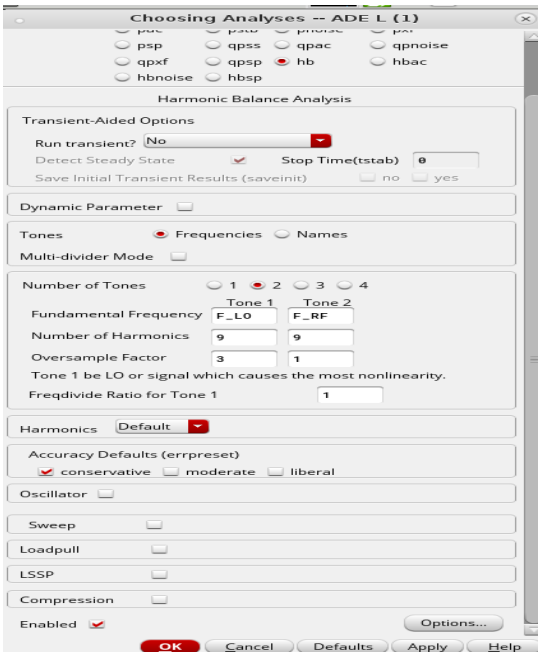
test bench used for runs

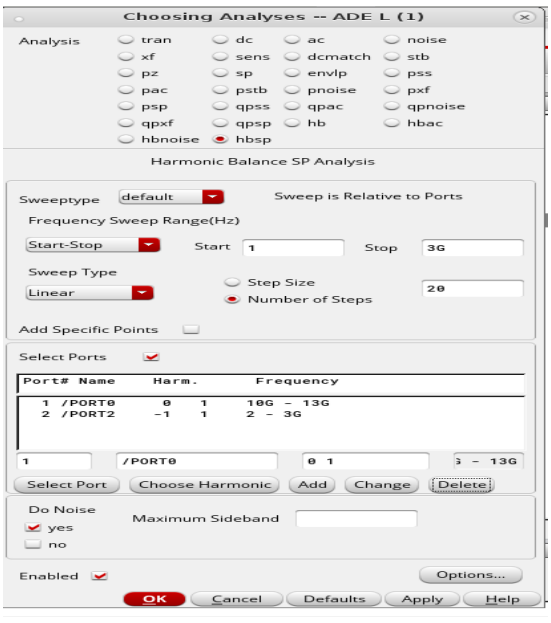
## 1- Plot gain, IIP3, IP1dB, and NF versus RF freuqnecy at IF frequency=100MHz

It was required to have IF = 100MHZ so a sweep OF F_RF = 10.1G:1G:13.1G
With F_LO = F_RF – 0.1G .

### 1-plot gain



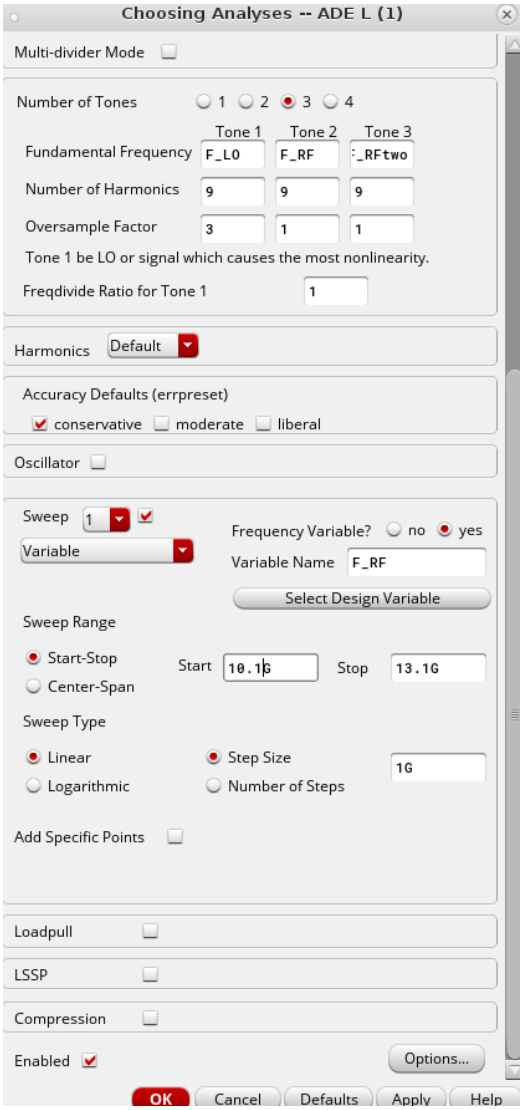




**Configuration used for test noise figure and gain.**

## 2- plot NF


## 3- plot IIP3

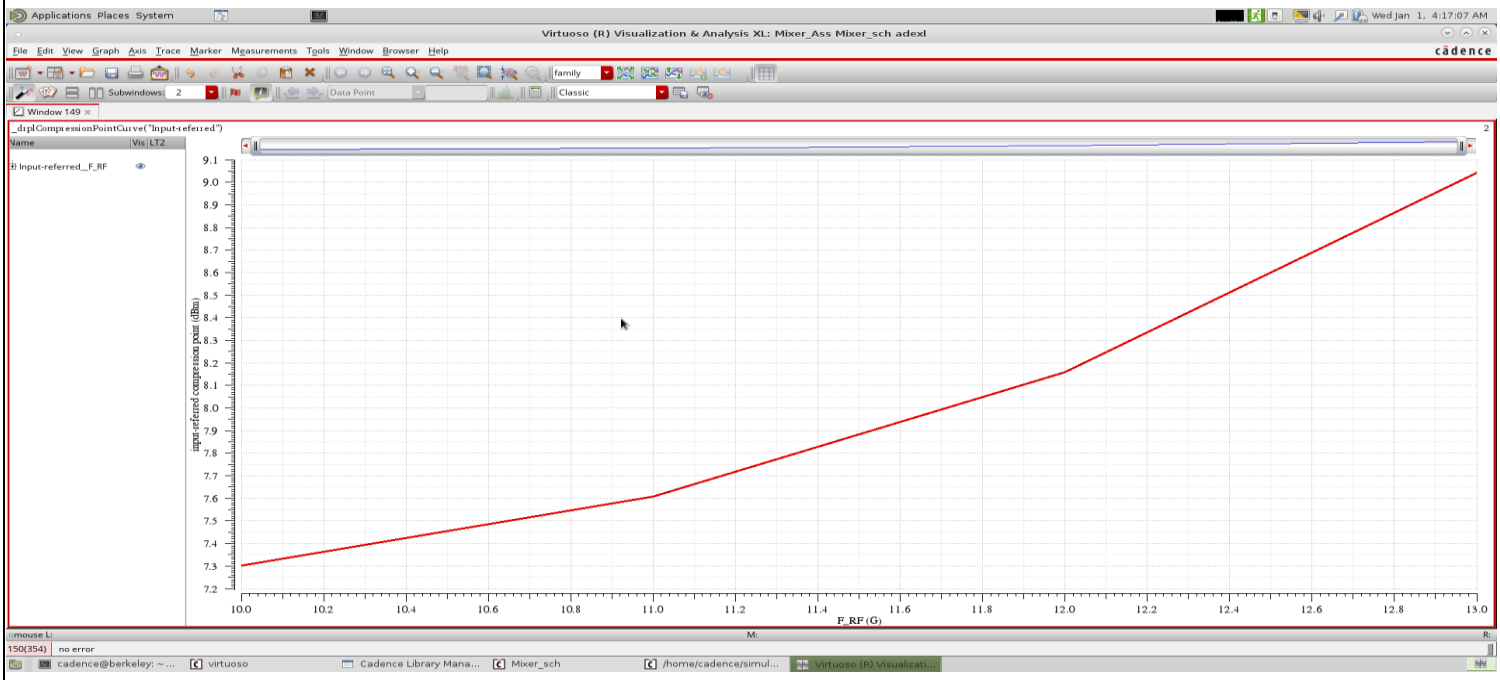


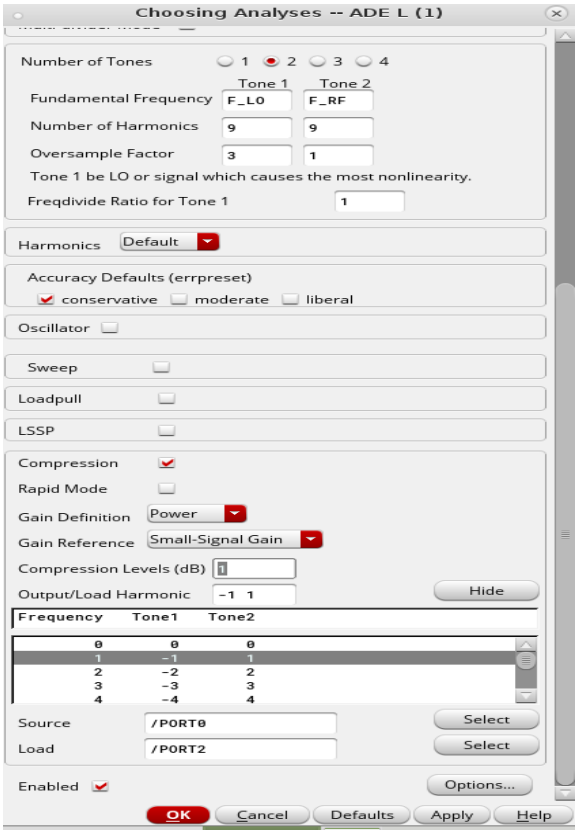

**3 tone test was used to get IIp3 where**

**F_LO=F_RF-0.1G**

**F_RFtwo=F_RF+0.1G**

## 4-plot P1dB

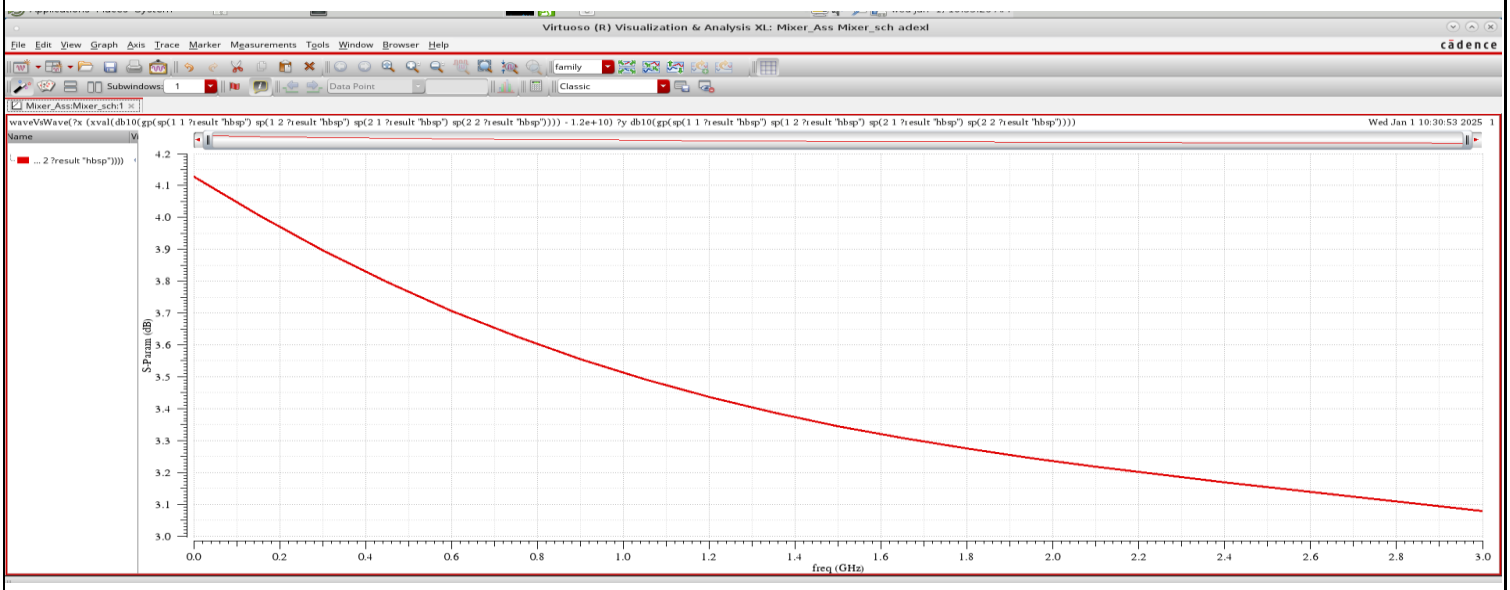

Virtuoso (R) Visualization & Analysis XL: Mixer_Ass Mixer_sch adexl
Window 149
_dtplCompressionPointCurve("Input-referred")
Input-referred__F_RF
input-referred compression point (dBm)
F_RF (G)


Choosing Analyses -- ADE L (1)
Number of Tones 1 2 3 4
Tone 1
Tone 2
Fundamental Frequency F_LO F_RF
Number of Harmonics 9 9
Oversample Factor 3 1
Tone 1 be LO or signal which causes the most nonlinearity.
Freqdivide Ratio for Tone 1 1
Harmonics Default
Accuracy Defaults (errpreset)
conservative moderate liberal
Oscillator
Sweep
Loadpull
LSSP
Compression
Rapid Mode
Gain Definition Power
Gain Reference Small-Signal Gain
Compression Levels (dB) 1
Output/Load Harmonic -1 1
Hide
Frequency Tone1 Tone2
0 0 0
1 -1 1
2 -2 2
3 -3 3
4 -4 4
Source /PORT0
Load /PORT2
Select
Select
Enabled
Options...
OK
Cancel
Defaults
Apply
Help

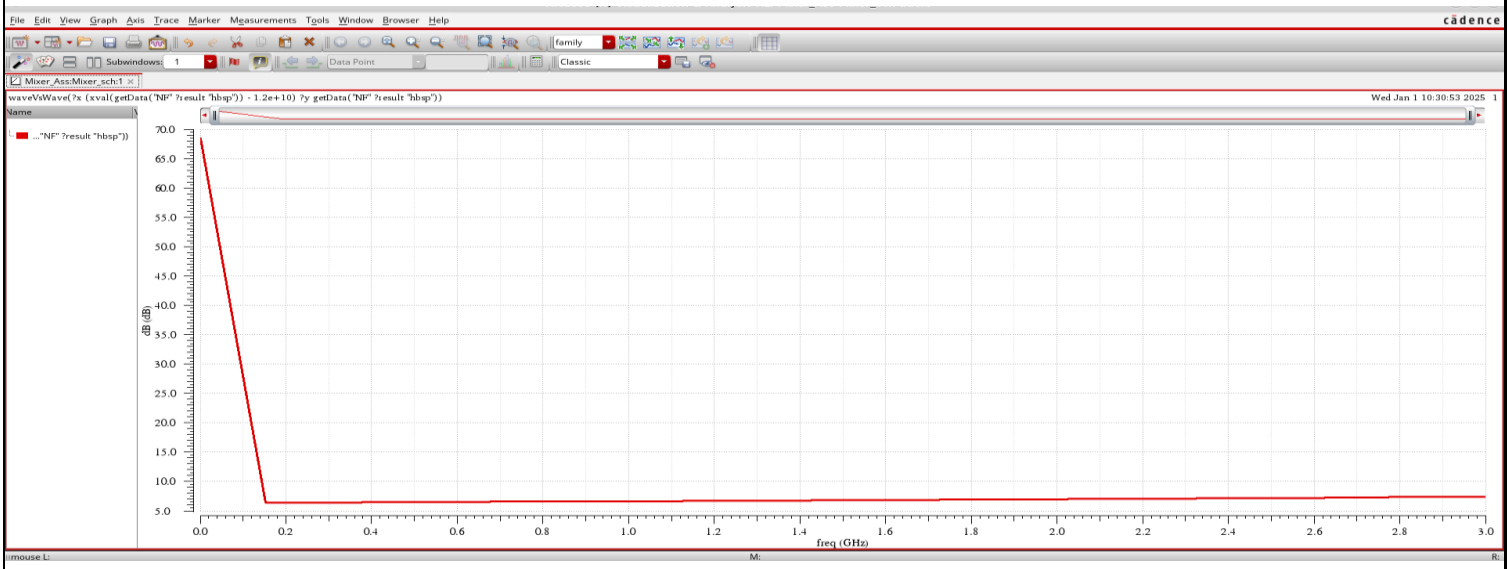
## 2- Plot gain, IIP3, IP1dB, and NF versus IF frequency at LO frequency = 12GHz

## F_LO=12GHZ & F_RF = 12G:1G:15G

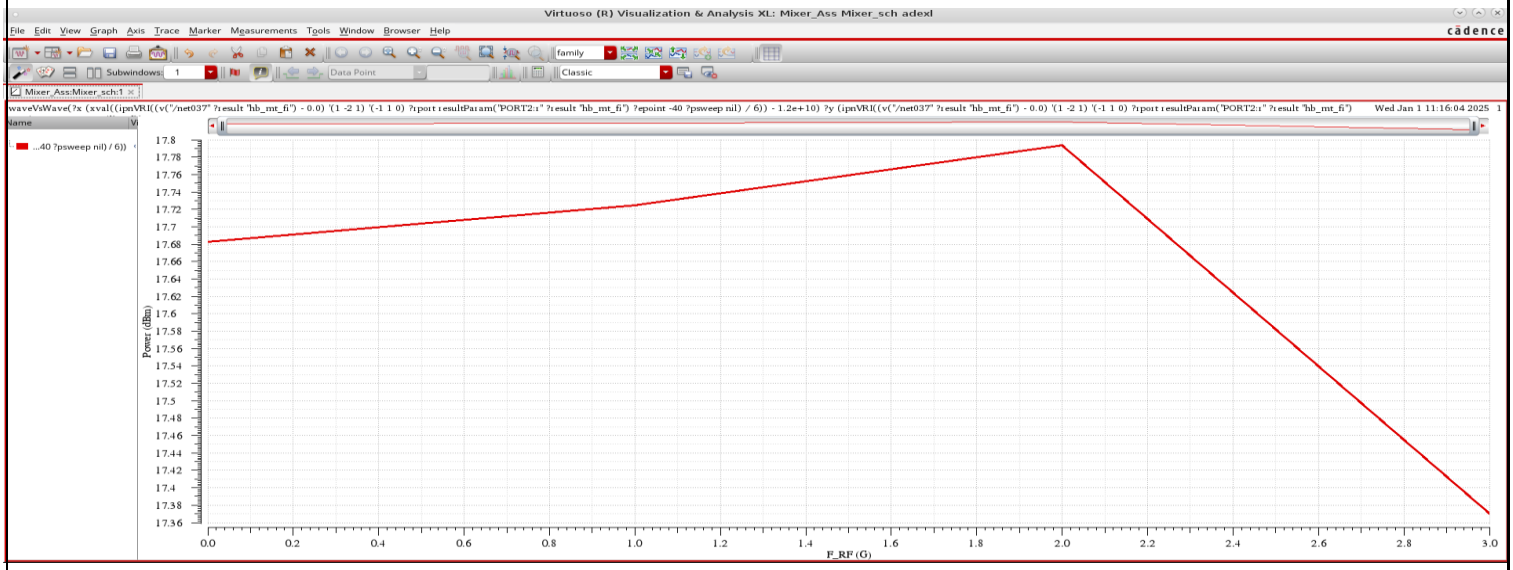
### 1-plot gain



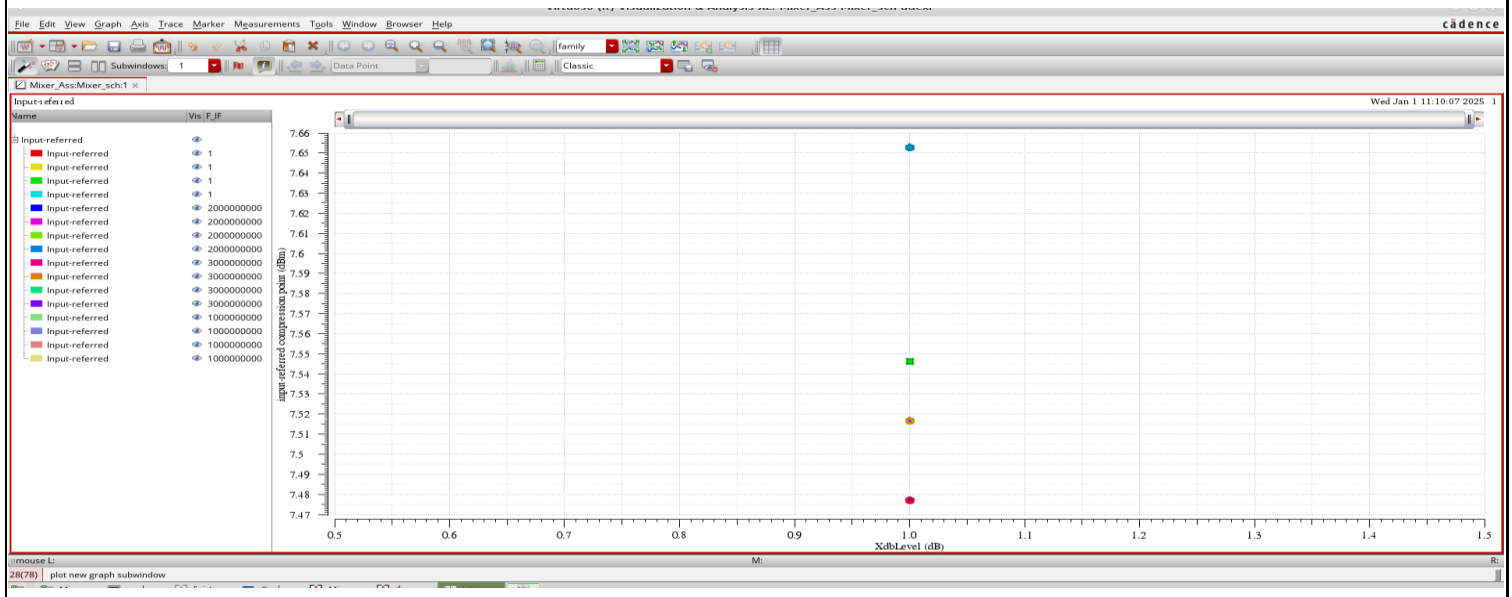

### 2- plot NF



## 3- plot IIP3

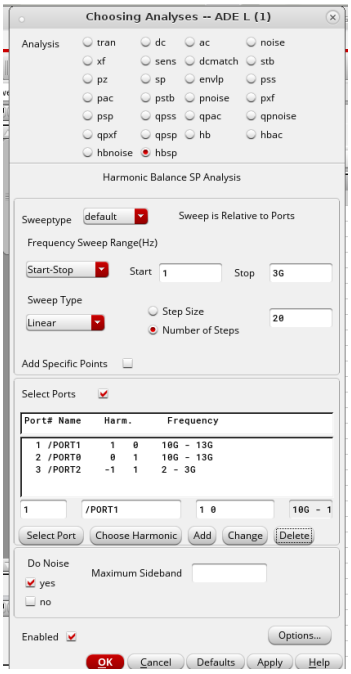



## 4-plot P1dB

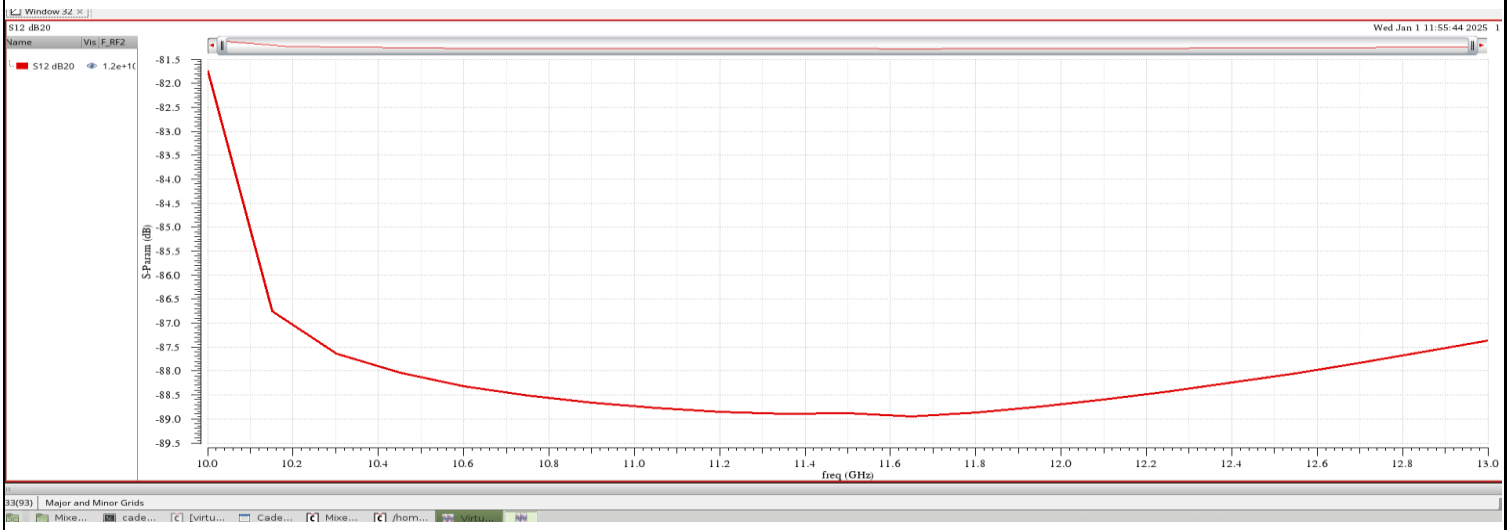


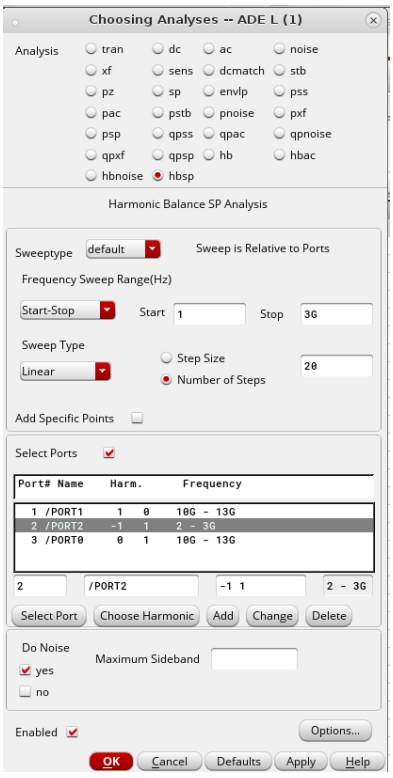
**3- Plot LO to RF isolation and LO to IF isolation versus LO frequency, during this simulation assume LO transistor sizes are different by 1% (i.e. increase width of one side of LO transistor by 1%)**

LO to RF isolation :

LO to IF isolation :



| | Spec | achieved |
|---|---|---|
| Conversion Gain | 3 | >3 |
| NF | 15db | <10 |
| IIP3 | 17dbm | 17-18 |
| LO to RF Isolation | 30db | -60 |
| LO to IF Isolation | 40db | -80 |
| IP1dB | 7dbm | >7 |