1 Eldo Tutorial - Analog IC Design

1.1 Lab setup

You may use the FPGA lab to run your Eldo simulations for this course. This simulation softwares are installed on all machines in this lab. The machine ip addresses are “10.7.8.41 to 80”, and you can run simulations remotely on these through an “ssh -X MACHINE-IP-ADDRESS” command. You may also install and run Eldo on your personal computers.

1.2 Setting up your terminal session to run Eldo

Your default login to the FPGA lab machines uses the Z-shell (zsh). Add the following lines to the .zshrc file in your home directory:

```
export LM_LICENSE_FILE=1717@10.7.9.34:1717@10.7.9.35:1717@10.7.9.36
export MGC_AMS_HOME=/tools/Mentor/AMS_2008_2_IXL
export PATH=\$PATH:\$MGC_AMS_HOME/bin:/tools/asitic
```

If you don’t have a .zshrc file, create one in your home directory (you could use vim editor) and add these lines to it. These are sourced everytime you login, but for the first time alone you will have to source it manually with a “source .zshrc” command (unless you want to log out and log back in). You are now ready to run Eldo.

1.3 Eldo Documentation

Eldo documentation is available on the FPGA lab machines in the following locations.

```
HTML: /tools/Mentor/AMS_2008_2_IXL/docs/htmldocs/eldo_ur/eldo_ur.htm
PDF: /tools/Mentor/AMS_2008_2_IXL/docs/pdfdocs/eldo_ur.pdf
```

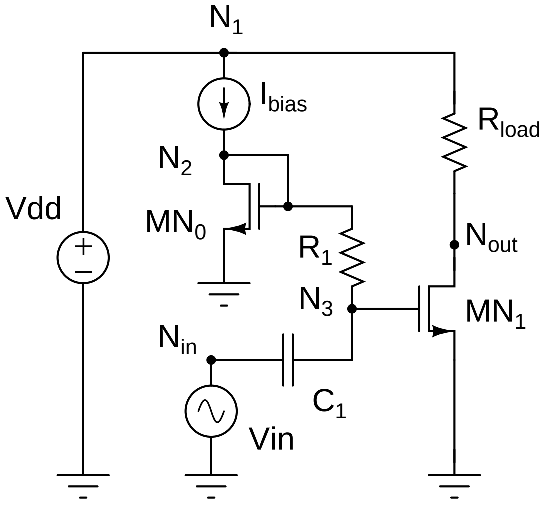
1.4 Example circuit and simulation

Download the netlist file “csamp.cir” from the class website. The netlist file is also given below:

```
*~~~~~~~~~~~~ CS amplifier ~~~~~~~~~~~~~*

.include ibm013.lib
.option compat tuning=accurate

.param LMN0=0.13u WMN0=1u LMN1=0.13u WMN1=50u
.param VIN=10m FIN=10k

*~~~~~~~~~~~~ Power Supply, Current biasing ~~~~~~~~~~~~~*
Vdd N1 0 1.2
Ibias N1 N2 100u
MN0 N2 N2 0 0 MODEL=CMOSN L={LMN0} W={WMN0}
R1 N2 N3 500k
```

```
C1 N3 Nin 100u
MN1 Nout N3 0 0 MODEL=CMOSN L={LMN1} W={WMN1}
Rload N1 Nout 200

Vin Nin 0 DC=0 AC=1 sin (0 {VIN} {FIN})

*~~~~~~~~~~~~~ Analysis ~~~~~~~~~~~~~~*
.op
.tran 0 400u
.ac dec 1k 1 10G
.noise v(Nout) Vin 1k

*~~~~~~~~~~~~~ Measurement and results ~~~~~~~~~~~~~~*
.plot tran v(Nout) v(Nin)
.plot ac vdb(Nout) vdb(Nin)
.plot noise input_PSD=PAR('db(inoise)+db(inoise)')

.end
```

This simulates the common source amplifier with resistive load. The circuit of the amplifier is shown below:



NOTE: Observe the connectivity (nodes) of the circuit and the SPICE netlist above.

1.5 Explanation of the netlist (SPICE Netlist

- To run eldo, cd into the directory where you have the csamp.cir file (If you do not understand this line, then you got to learn Linux basics first) and the in the shell type "eldo csamp.cir".
- In SPICE syntax, anything that starts with * is a comment and is ignored by the simulator.

- The first line in SPICE netlist file is always taken as a comment no matter whether you put * at the start or not.
- .include - anything in spice starting with . is a SPICE command. By ".include ibm013.lib" we are saying SPICE to include the "ibm013.lib" file. The file ibm013.lib must be present in the directory where you are executing the "eldo" command.
- ibm013.lib file contains the required SPICE models for the transistors. Open the file and just explore a bit. See if you can understand it (partially is also ok).
  - A simulator can be thought of a complex calculator which can solve complex equations.
  - The simulator needs a mathematical model (equation) for every circuit elements.
  - Models for most of the ideal circuit elements like, Resistor (R), Capacitor (C), Inductor (L), Voltage source (V) are inbuilt with in the simulator. We could use them directly from the simulator itself.
  - For practical circuit elements as in MOS, BJT, Practical Non linear resistor etc, we need to provide the corresponding mathematical device model to the simulator.
  - The mathematical model for the MOS transistors (from IBM Process) are captured into ibm013.lib file. This is called the model files for the transistors.
  - As we intend to use these thansistors in the circuit, we include the ibm013.lib file into the simulator.
  - If you open the ibm013.lib file you can see what type of mathematical modelling is done. You will see "T55R SPICE BSIM3 VERSION 3.1 PARAMETERS". This tells you that it is a "BSIM3 VERSION 3.1" model.
  - BSIM stands for Barkley Simulation Model. This mathematical model to capture a transistor behaviour was given by UCB hence the name BSIM.
  - The square law mathematical model for MOS transistors that you are fameliar with is the simplest model possible. It's inaccurate (in a sense it doesn't caplure all MOS device effects).
  - BSIM is more accurate model for the same. This is one of standard model used across simulators.
  - ".MODEL CMOSN NMOS ... " in the ibm013.lib file tells you that the model name for the NMOS device is CMOSN.
  - Hence, when you would like to use NMOS, you need to use the model name CMOSN.
- .option compat tuning=accurate - these are the options used by the simulator (to decide the simulation accuracy). For details look into the eldo manual.
- .param LMN0=0.13u WMN0=1u LM ... - Using .param we could set parameter values (LMN0 holds a value 0.13u). Once defined, these parameters could be used as variables in any other part of the netlist by using {parameter_name} (e.g. L={LMN0}).
- Vdd N1 0 1.2 - Any line starting with V will represent a voltage source. Here voltage source Vdd is connected between N1 and ground (0). Its value is 1.2 volts.

- Note, the syntax is “Vxx node1 node2 voltage_value” - see manual for more details.
- Ibias N1 N2 100u - Any line starting with I will represent a current source. Here current source Ibias is connected between N1 and N2.
- Note, the syntax is “Ixx node1 node2 current_value” - see manual for more details.
- Engineering notations can be used in SPICE. m=milli (1e-3), u=micro (1e-6), G=gig (1e+9) etc.
- **NOTE: Spice is case insensitive. Hence M and m are same to SPICE. So, to represent Mega (1e+6), one should use Meg and not M.**
- MN0 N2 N2 0 0 MODEL=CMOSN L=LMN0 W=WMN0 - Line starting with M is taken a MOS transistor. Now a MOS transistor is a 4-terminal device. The order for four terminals go as “Drain Gate Source Bulk”
- Note, the syntax is “Mxx drain_node gate_node source_node bulk_node MODEL=model_name L=length_of_the_gate W=width_of_the_transistor” - see manual for more details.
- R1 N2 N3 500k - A line starting with R represents a resistor. Here a resistor R1 of value 500k is connected between node N2 and N3.
- C1 N3 Nin 100u - A line starting with C represents a capacitor. Here a capacitor C1 of value 100u is connected between node N3 and Nin.
- Vin Nin 0 DC=0 AC=1 sin (0 VIN FIN) - as said earlier a line starting with V represent a voltage source. By default it is taken as a DC voltage source. But, with a little more syntax we can get any type of source (as in Sin source, Pulse source etc.). Here the source is a Sin wave source.
- Vin is connected between Nin 0.
- DC=0 means its DC voltage value is zero (means zero offset).
- AC=1 means, for AC analysis (small signal model analysis, to be discussed below), the input voltage will be taken as 1v.
- sin (0 VIN FIN) - means the source will have sinusoidal output (along with the DC output). The offset of the SIN wave is 0, amplitude is {VIN} (a parameter defined earlier), and frequency is {FIN} (a parameter defined earlier).
- .op - This commands simulator to run Operating Point simulation (as in bias point calculation).
- The output goes into a .chi file (csamp.chi). Below is a snapshot of different operating points for the given example.

```
0**** SMALL SIGNAL BIAS SOLUTION    TEMPERATURE =   25.000 DEG C

0************************************************************
```

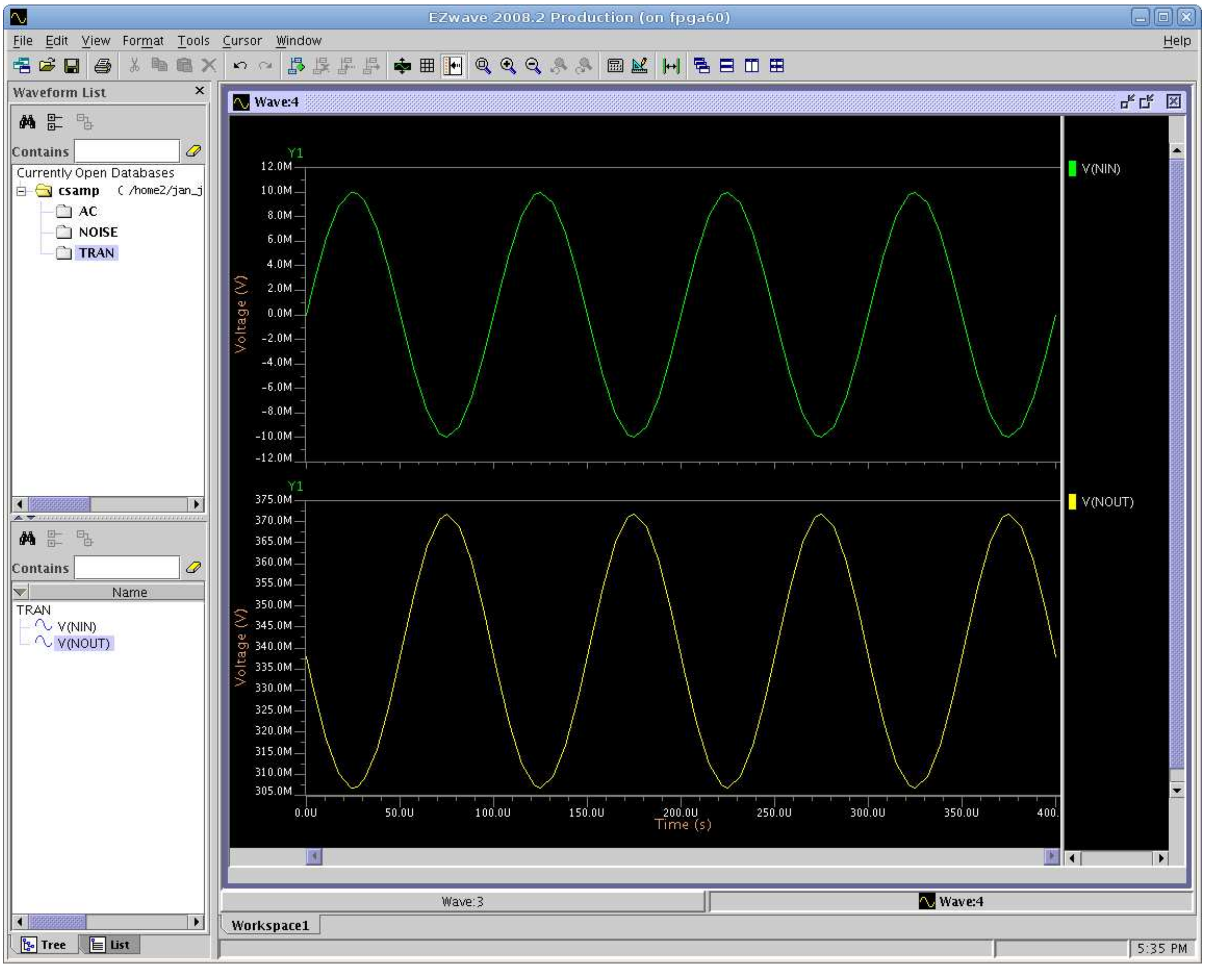
```
NODE         VOLTAGE     NODE     VOLTAGE    NODE    VOLTAGE
N1           1.2000E+00  N2       6.7537E-01  N3      6.7537E-01
NIN          0.0         NOUT     3.3800E-01


   VOLTAGE SOURCE CURRENT

NAME      CURRENT      VOLTAGE      POWER
VDD       -4.4100E-03   1.2000E+00  -5.2920E-03
IBIAS      1.0000E-04   5.2463E-01   5.2463E-05


0****  OPERATING POINT INFORMATION   TEMPERATURE =   25.000 DEG C

0*****************************************************************

               MN0          MN1

MODEL          CMOSN        CMOSN
ID              1.0000E-04   4.3100E-03
Ibd            -1.6270E-30  -3.6238E-30
Ibs             0.0          0.0
VGS             6.7537E-01   6.7537E-01
VDS             6.7537E-01   3.3800E-01
VBS             0.0          0.0
VTH             4.2274E-01   4.3030E-01
VDSAT           1.3186E-01   1.3360E-01
GM              5.5263E-04   2.5106E-02
GDS             3.2823E-05   2.7180E-03
RON             3.0467E+04   3.6792E+02
GMB             6.8003E-05   3.5951E-03
Cdd             4.7874E-16   2.4523E-14
Cdg            -8.5306E-16  -4.3636E-14
Cds             4.2053E-16   2.1831E-14
Cdb            -4.6205E-17  -2.7179E-15
Cgd            -4.7835E-16  -2.4411E-14
Cgg             1.9338E-15   9.7658E-14
Cgs            -1.3842E-15  -7.0214E-14
Cgb            -7.1293E-17  -3.0335E-15
Csd             1.2987E-19   1.3286E-16
Csg            -8.5306E-16  -4.3636E-14
Css             8.9913E-16   4.6221E-14
Csb            -4.6205E-17  -2.7179E-15
Cbd            -5.2013E-19  -2.4541E-16
Cbg            -2.2770E-16  -1.0386E-14
Cbs             6.4520E-17   2.1624E-15
```

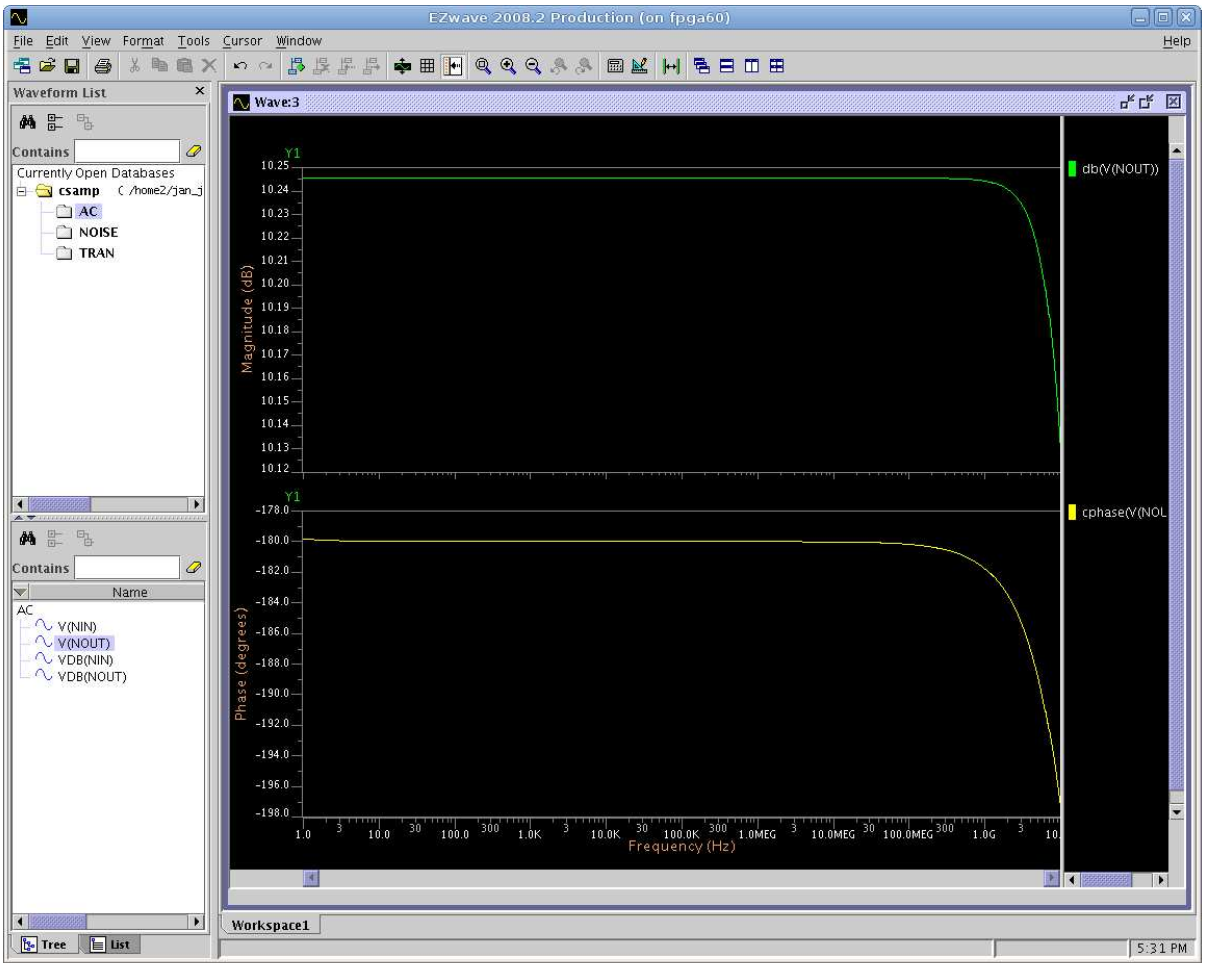
```
Cbb              1.6370E-16   8.4693E-15
PHI              8.5885E-01   8.5885E-01
VBI              1.0154E+00   1.0154E+00
Region          saturation   saturation
VTH_D            2.5263E-01   2.4507E-01

                R1           RLOAD
V+               6.7537E-01   1.2000E+00
V-               6.7537E-01   3.3800E-01
ID               0.0          4.3100E-03

                C1
V+               6.7537E-01
V-               0.0
```

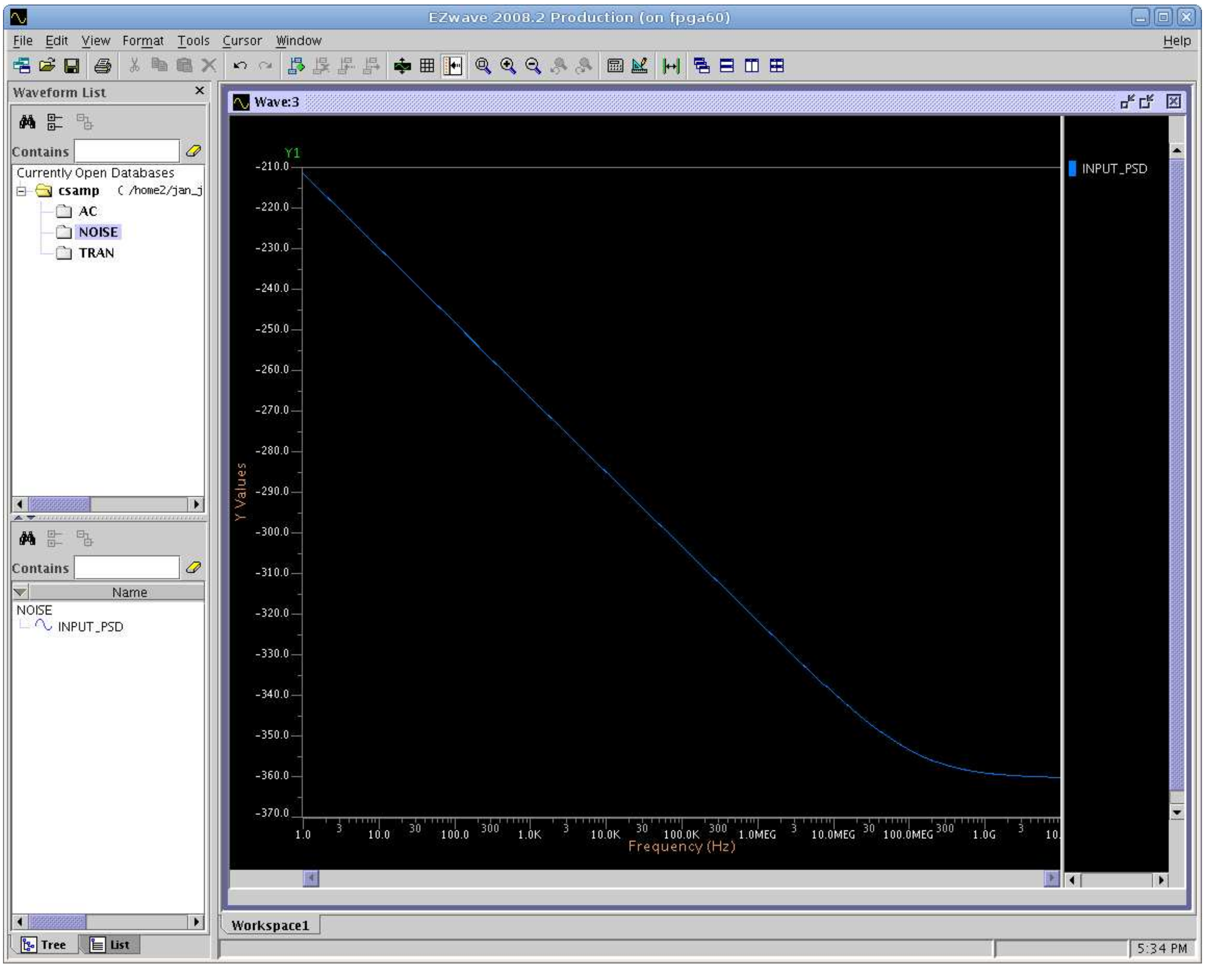
- As it can be seen that all the operating point values are listed.
- .tran 0 400u - This tells simulation to run a transient simulation for 400u Sec. and save the data from 0 Sec to 400u Sec.
- A transient simulation means, for every value of the input signal the simulator will solve KCL and KVL and will give node voltages and branch currents.
- .ac dec 1k 1 10G - This tells simulation to run an ac analysis and store the output from 1Hz to 10GHz with 1k data points every decayed.
- An ac analysis is a small signal analysis. The simulator linearises the circuit around its operating point and then find the over all transfer function of the small signal system. Once it has the transfer function it calculates the frequency response and phase response.
- .noise v(Nout) Vin 1k - puts in the noise model for every device and finds out the input referred and output referred noise.
- inoise parameter gives the input referred noise and onoise gives the output referred noise.
- .plot tran v(Nout) v(Nin) - tells the simulator to plot the results of transient simulation for node voltage of Nout and node voltage of Nin. This writes into .wdb and .swd files.
- ezwave is the viewer for these .wdb and .swd files.
- **ezwave file_name.wdb - will open the waveform viewer (ezwave csamp.wdb).**
- .plot ac vdb(Nout) vdb(Nin) - tells the simulator to plot the results of ac simulation for node voltage of Nout and node voltage of Nin in db format. This writes into .wdb and .swd files.
- .plot noise input_PSD=PAR(’db(inoise)+db(inoise)’) - tells the simulator to plot the results of noise simulation. This writes into .wdb and .swd files. NOTE: PSD is square of the input referred voltage/current noise spectral density.

- input_PSD=PAR('db(inoise)+db(inoise)') - PAR tells simulator to consider fillowing item within bracket as parameter. Hence, inoise in db gets added with inoise in db and the result is stored into input_PSD. Finally input_PSD gets plotted.
- .end - ends the simulation.

1.6 Simulation Results




EZwave 2008.2 Production (on fpga60)
File Edit View Format Tools Cursor Window Help
Waveform List
Contains
Currently Open Databases
csamp (/home2/jan_j
AC
NOISE
TRAN
Contains
Name
AC
V(NIN)
V(NOUT)
VDB(NIN)
VDB(NOUT)
Tree
List
Wave:3
Y1
10.25
10.24
10.23
10.22
10.21
10.20
10.19
10.18
10.17
10.16
10.15
10.14
10.13
10.12
Magnitude (dB)
db(V(NOUT))
Y1
-178.0
-180.0
-182.0
-184.0
-186.0
-188.0
-190.0
-192.0
-194.0
-196.0
-198.0
Phase (degrees)
cphase(V(NOU
1.0
3
10.0
30
100.0
300
1.0K
3
10.0K
30
100.0K
300
1.0MEG
3
10.0MEG
30
100.0MEG
300
1.0G
3
10.
Frequency (Hz)
Workspace1
5:31 PM


EZwave 2008.2 Production (on fpga60)
File
Edit
View
Format
Tools
Cursor
Window
Help
Waveform List
Contains
Currently Open Databases
csamp (/home2/jan_j
AC
NOISE
TRAN
Contains
Name
NOISE
INPUT_PSD
Tree
List
Wave:3
Y1
-210.0
-220.0
-230.0
-240.0
-250.0
-260.0
-270.0
-280.0
-290.0
-300.0
-310.0
-320.0
-330.0
-340.0
-350.0
-360.0
-370.0
Y Values
1.0
3
10.0
30
100.0
300
1.0K
3
10.0K
30
100.0K
300
1.0MEG
3
10.0MEG
30
100.0MEG
300
1.0G
3
10.
Frequency (Hz)
INPUT_PSD
Workspace1
5:34 PM