

# INF 5460 Electronic Noise – calculation and countermeasures

## Mandatory task number 1.

Deadline for delivery: Monday 12 of September at 12:00.

Assessment: Approved/Not approved.

Reports are submitted on an individual basis. The reports shall consist of all the schematics that are used to achieve the reported results, simulation results and text explaining what has been done, summary tables as well as an analysis of the results. The report should be submitted electronically as a pdf file. Schematics/symbols at LTspice format may be attached as well. Use white/gray background on schematics and simulation results. Avoid yellow color on curves. The report should be in ENGLISH.

### Tool

Download the free simulator LTspice/SwitchCad III from Linear Technologies and install on the machine you want. The software is available at:

<http://www.linear.com/designtools/softwareRegistration.jsp>

(You may install program and libraries under C:\<Program Files>\LTC\LTspiceIV\ .

The exact name of <Program Files> depends on your operation system. You may install it another place if you want but remember where you put it.)

## 1. Get familiar with the simulator

### LM741

Recall circuit LM741 from the library which comes with the software.  
(LTC\LTspiceIV\examples\Educational )

a) Transient Analysis.

Identify the components that belong to the feedback and which belongs to the LM741 amplifier. What gain do you expect to get theoretically? Let the DC offset on the positive input remain zero while you find the DC offset interval for the negative input where the output does not go into saturation. What is the relation between DC input, AC input, gain, and output range? Explain!

b) Set up for frequency analysis (AC analysis). What is the DC gain and Gain Bandwidth (GBW)? What is the phase margin? What is the relation between DC gain and GBW in feedback systems?

c) Perform AC analysis with varying common DC-offset (common mode) and find the approximate area where the gain is greater than -6dB of maximum level.

d) Add a load capacitor and use the STEP function to create a plot in which the capacitive load is increased to 30pF in increments of 10pF. What is the new DC gain and GBW? What is the main effect of changing the capacitor's value?

## 2. Frequency characteristics of some curves.

Draw a new schematic consisting of a voltage source and a first order low pass RC filter. The voltage source and the capacitor are grounded. Name the voltage source node "A" and the output of

the filter “B”. Start by selecting RC filter values so that it has no attenuation (i.e. very high cutoff frequency).

- a) Let the voltage source generate a sine wave with period 1 millisecond and amplitude 1V. Simulate over a period of 1 second and set maximum timestep to 1/10 period. Perform an FFT on node A. What did you expect? What is the strength of the strongest unwanted frequency component? Perform the same again with maximum timestep 1/100 of the period. What's going on? What is the strength of the strongest unwanted component? What does this say about the simulator setup?
- b) Let the voltage source generate a symmetrical square pulse with the same period as the sine. Simulate with rise and fall time equal to i) half the period (that is a triangle pulse), ii) 1/10 of the period and iii) 1/100 of the period (the signals should look as clock signals, so set  $T_{on} = 0$  for i),  $T_{on} = 0.5m - T_{rise}$  for ii) and iii)). Perform an FFT on node A and measure the strength and frequency of the strongest signal for the fundamental frequency. What's going on? What should the clock edges look like to reduce the high frequency components in the signal? (What's the downside of having gently sloping edges in eg. standard CMOS logic?)
- c) Calculate and give RC filter values so that we get a cutoff frequency around 5 times over the fundamental frequency. Use the input described in point b-iii). Perform a transient simulation and measure at node A and B. How do the curves look like now? How is the FFT output?

### 3. Decoupling capacitors

In this subtask you shall find the size and number of capacitors required to reduce the supply voltage noise to an acceptable level. You will find all the equations in the first lecture note. (Literature background can be found in Ott 11.4). Each capacitor has an inductance of 10nH. The current spikes can be modelled as a triangular shape from 0A to 1A with a rise and fall time of 5ns. We want the voltage to be stable within 5% of 1.5V. The lower frequency corner is 1MHz.

- a) What is the low frequency target impedance  $Z_t$  based on the values above?
- b) What is the number of capacitors required?
- c) What is the total capacitance? And what is the capacitance per capacitor?
- d) What is the high frequency corner decided by the current rise/fall time?
- e) Draw a schematic and do transient and frequency simulation. The schematic consist of a current source, an inductor and a capacitor. The inductor and capacitor represents all the parallel coupled capacitors. (In addition you may add a resistor in parallel with the source to empty the current between the spikes and remove the step behaviour. If it is not too small it will not influence on our noise analysis). Make sure to get the correct value on the inductor.
- f) What will be the difference if the raise/fall time is reduced to 1ns?

### 4. Parasitic capacitive coupling

#### **System Description:**

Two lines are routed in parallel over a length of 10cm. The capacitance between the lines is 0.1pF/cm. In one line (S=source) there is a signal with 5V swing that is received as noise in the second line (O = object). The receiving line has a capacitance and a resistor in parallel to ground of 10pF and 10M $\Omega$ .

#### **Task:**

- a) How much noise is captured in the receiving line? Show this both by calculation and simulation.

b) We will put a shield around the receiver. The shield is grounded and it has a capacitance with the inner conductor of 1pF/cm. For each cm with shield, we can disregard the capacitance between the object O and the source S for the same distance. Show by simulation and calculation what noise is received with a 2cm, 5cm, 9cm, and 9.9cm shield.

## 5. Artificial sources of transient analysis

Sometimes we need to create artificial noise sources to see how the circuit responds to noise. LTspice has a general voltage source "BV" (Arbitrary Behavioral Voltage Source) where the voltage may be described by functions. Two of the functions are RANDOM(i) and WHITE(i), which generate a random number within a range. For RANDOM(i) the interval is [0: 1], while for WHITE(i) it is [-0.5, 0.5]. These functions draw quasi random numbers within their intervals depending on a integer seed *i*. To generate a sequence of random numbers we may use "time" as the index. "Time" has the units in seconds and thus will generate a new number each second. If we want new numbers, for example, every millisecond, we have to use  $time*1E3$  as index. Often we need more sources and it is important to make them independent.

a) Make two BV sources *WHI1* and *WHI2* both having WHITE(time\*1E3) as function. Subtract the resulting curves in the simulation window and comment on the result.

b) Change the index in one of the sources to say WHITE(time\*1E3 + 1), subtract the resulting curves and comment on the new result.

This noise model has a limited validity due to the limited value range. Unfortunately, LTSpice does not have a noise source that generates noise that has a normal distribution (Gaussian distribution). However, we can generate a normal distributed noise with another application (e.g. Excel) and store it as a text-file. To do this we use a standard voltage source in LTspice but choose PWL FILE as the source.

c) Create a voltage source FILE1 browsing file PWLsrc\_10k\_1ms.txt (from course webpage) (10k: 10k points, 1ms: 1ms between points). Compare the curve with the curves you got with the WHITE() - function.

d) Create a new BV source FILE2 (general voltage source) and give it the function DELAY(V(FILE1),1m). Subtract the two curves and comment on the result.

**Enjoy! :)**