SMDP Instructional Enhancement Programme: 13-24 Nov. 2006

Nagendra Krishnapura (nagendra@iitm.ac.in)

18 Nov. 2006

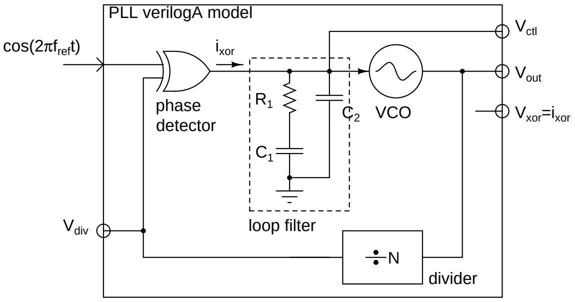

Figure 1:

Model a pll in verilogA as shown in Fig. 1. The PLL should multiply up a 10 MHz input to a 200 MHz output. Adjust the loop filter to get a unity loop gain at 0.2 MHz, a zero at 0.02 MHz, and a pole at 0.5 MHz. ($I_{cp} = 1\,\mu$A and $K_{vco} = 30$ MHz/V)

1. Reference input: The phase detector should be able to take in a sinusoidal reference (e.g. use @(cross()) to sense the zero crossings)

2. Phase detector: Use a current output XOR gate with $\pm 1\,\mu$A output

3. Loop filter: Try the following to model the loop filter a) `laplace_xx` in veriloga, b) Instantiating resistors and capacitors in the veriloga module

4. VCO: Model it as in the previous assignment, with a 200 MHz free running frequency.

5. Divider: Count the edges of the VCO to get the divided output

Run a transient simulation with `errpreset = conservative` and observe the results

Tighten the tolerances (reltol) by an order of magnitude and observe the results

Force simulations every 25 ps using `strobeperiod` and observe the results

Observe the settling of the control voltage, output waveform of the VCO, divider output, phase detector output. What is the settling time? Does the settling correspond to hand calculated value of the loop bandwidth?

Run the simulations for 50 μs and compute the power spectral density of the output using `psd` with `Hann` window. What is the level of the reference feedthrough? You need to choose an appropriate length of the output at the end of the simulation and perform dft on it.

Note: Model, simulate, and test each block individually before putting the PLL together. It'll be impossible to debug the model otherwise.