ELEN 6901
PLL Phase Noise/Jitter Modeling

Chun-Wei Hsu
Karthik Tripurari

Introduction of Phase Noise

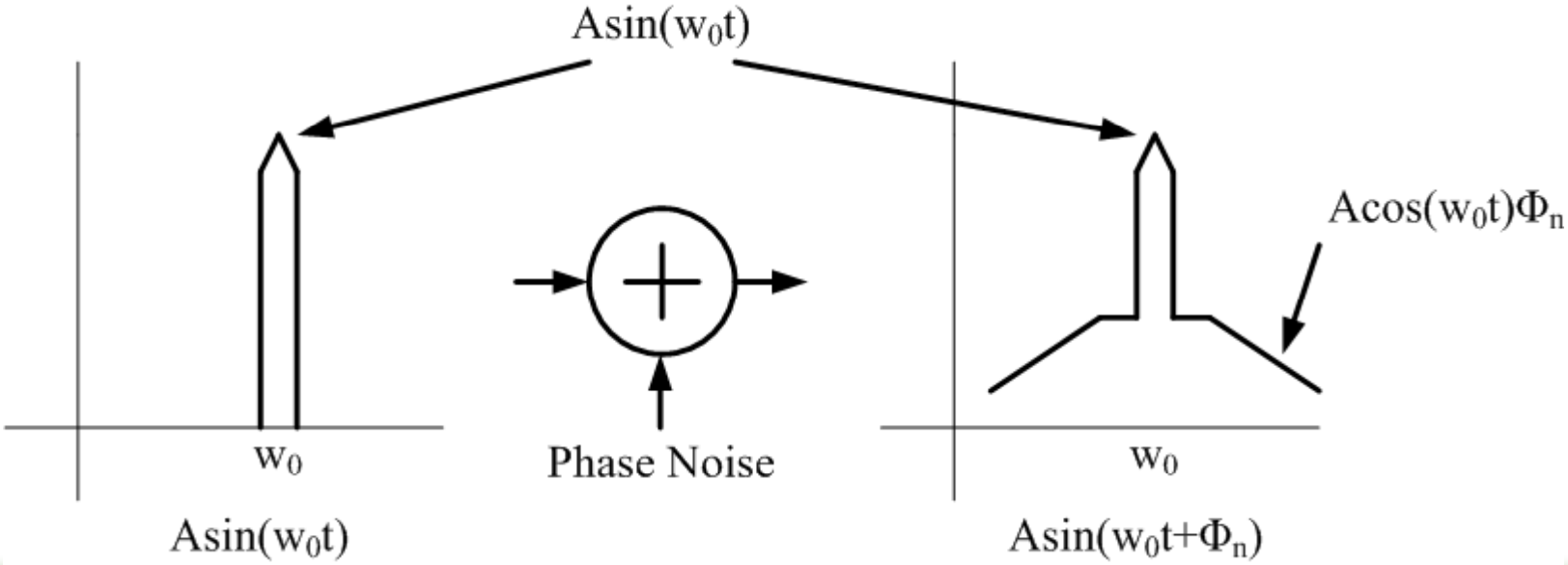
- The oscillator output *Vo* with the phase noise Φ_n

$$Vo = A\sin(w_0 t + \phi_n) = A\sin(w_0 t)\cos(\phi_n) + A\cos(w_0 t)\sin(\phi_n)$$

where *A* is the amplitude and w_0 is the frequency

- For small noise term

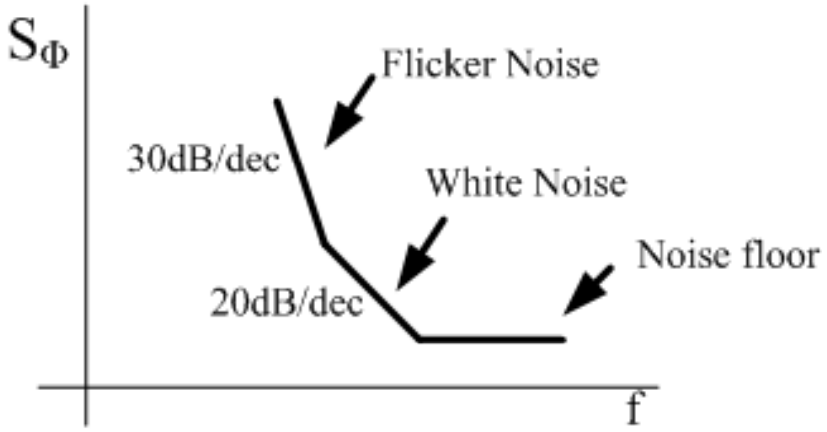
$$Vo \cong A\sin(w_0 t) + A\cos(w_0 t)\phi_n$$

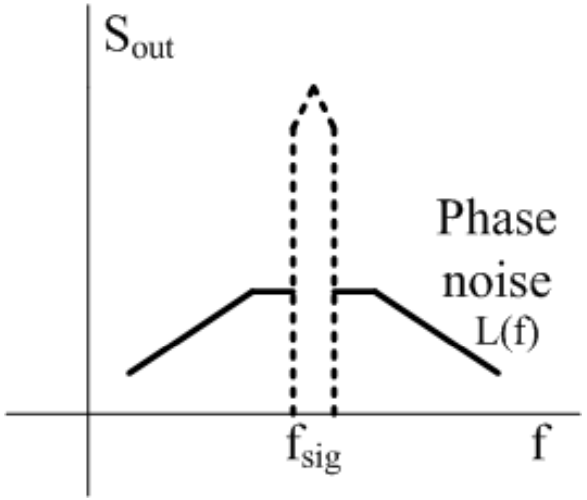


Introduction of Phase Noise

- Phase perturbation from device noise includes white and flicker noise

- Power spectral density S_Φ of phase noise Φ_n



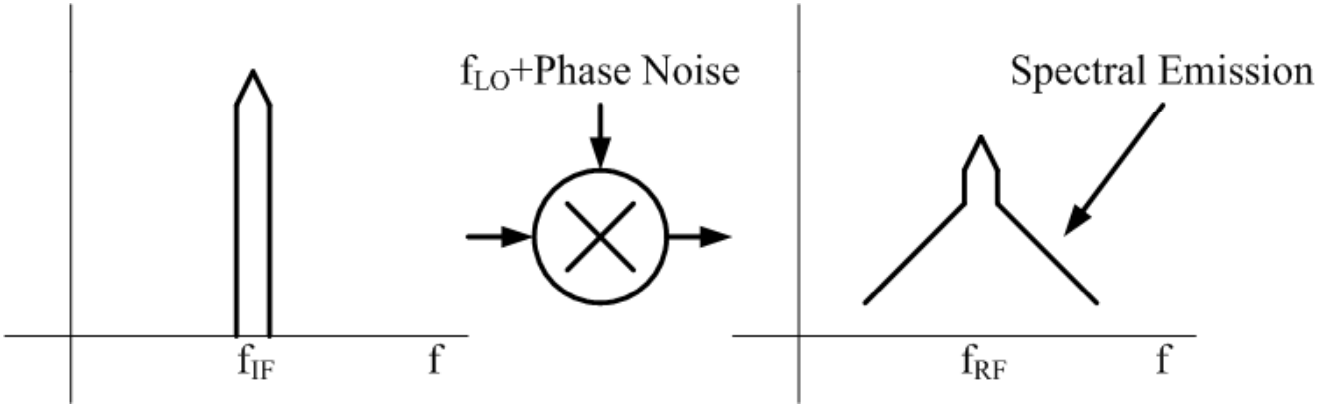


- Extracting noise parameters
  - Single-sided PSD L(f)
  - $L(\Delta f)=(1/2)S_\Phi(\Delta f)$

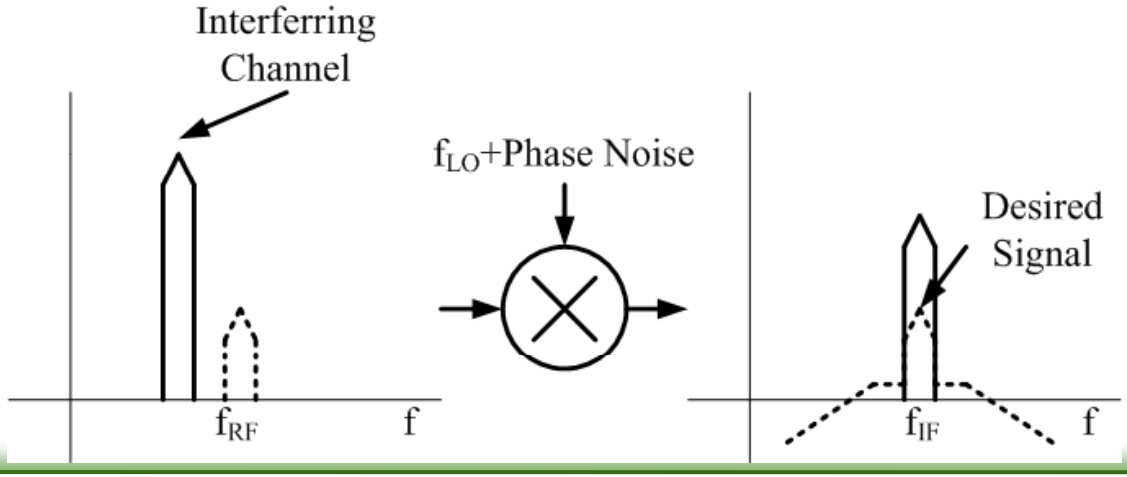
Lee, T.H.; Hajimiri, A., "Oscillator phase noise: a tutorial," Solid-State Circuits, IEEE Journal of , vol.35, no.3, pp.326-336, Mar 2000

Impact of Phase Noise

- Phase Noise
  - Reduce SNR of the signal
  - Tx: Out-of band emission



  - Rx: Reciprocal mixing



Phase Jitter

- Jitter issues
  - Sampling cases
    - Degrade SNR of ADC/DAC
  - Data recovery considerations
    - Reduce the eye-opening
    - Decrease BER

- Definition of phase jitter
  - Difference between the measured time and the ideal bit period

$$\sigma^2_{phase\_jitter} = \left(\frac{1}{2\pi f_{sig}}\right)^2 \int S_\Phi df$$

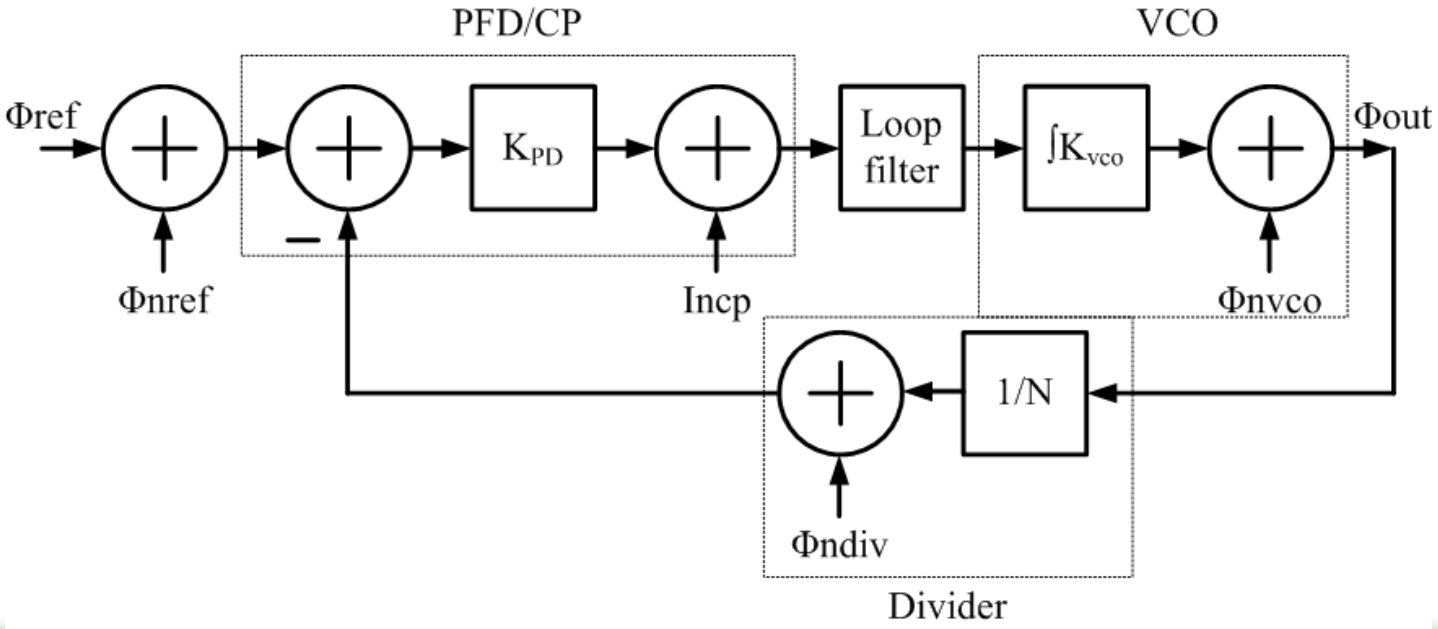
PLL Noise Modeling

- Phase domain model
  - Simple and Linear model
  - Sampling nature of PLL ignored
  - Efficient for the noise analysis when the PLL is in locked state

- Voltage domain model
  - A complete but complex model
  - No quiescent operating points and only periodic operating points
  - Describes phenomena like cycle slipping, false locking and lock capture
  - Long simulation time

Phase Domain Model

- Noise Sources
  - Reference
  - PFD/CP
  - Loop filter
  - VCO
  - Divider



CPPLL Type II 3rd Order

- Transfer function of loop filter (R-C//C_p)

$$Z(s) = \frac{(s/w_z + 1)}{s(C + C_p)(s/w_p + 1)}$$

$$w_z = \frac{1}{RC}, \quad w_p = \frac{1}{RCC_p/(C + C_p)}$$

- Transfer function of loop gain (two poles at s=0)

$$L(s) = \frac{Icp}{2\pi} Z(s) \frac{Kvco}{s} \frac{1}{N} = \frac{Icp}{2\pi} \frac{Kvco(s/w_z + 1)}{s^2(C + C_p)(s/w_p + 1)} \frac{1}{N}$$

Noise Transfer Function of Type II CPPLL

- Noise from the reference

$$Href = \frac{\Phi out}{\Phi nref} = \frac{NL(s)}{1+L(s)} = H(s)$$

where H(s) is the closed-loop transfer function of PLL

- Noise from PFD/CP

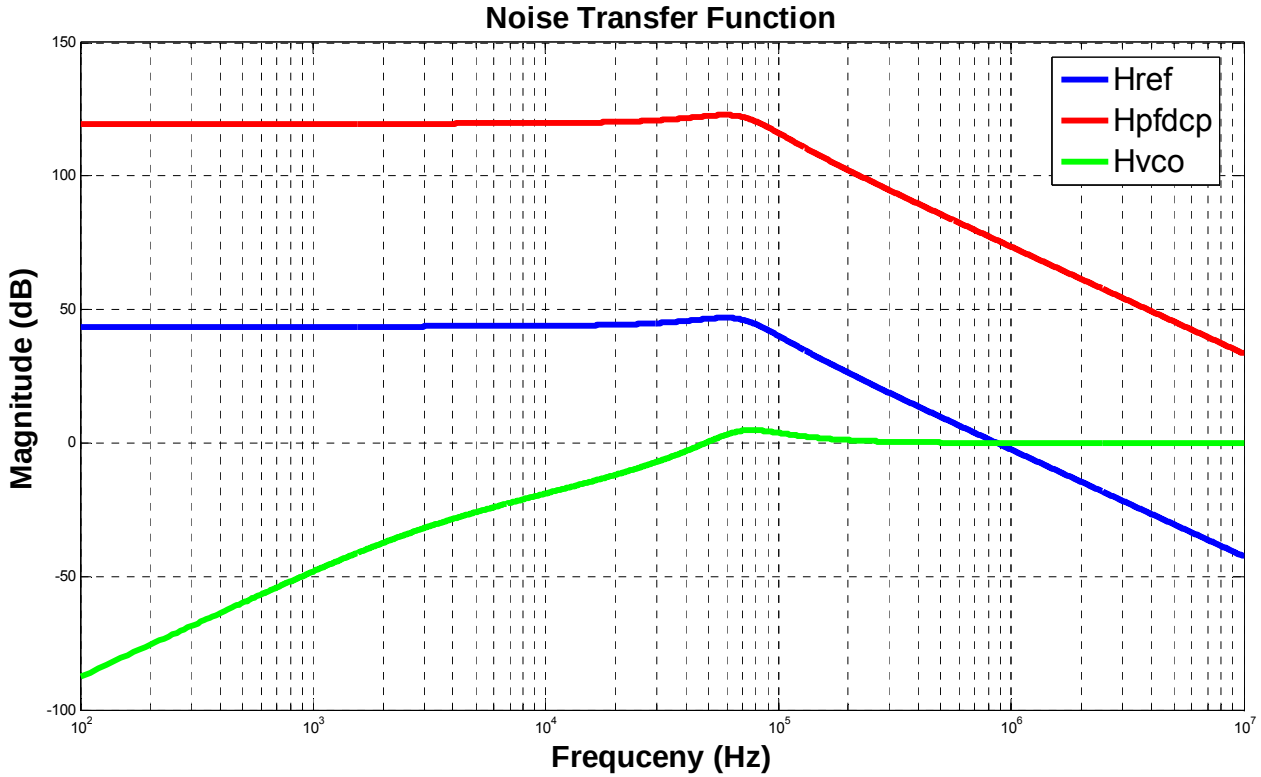
$$Hpfdcp = \frac{\Phi out}{Incp} = \frac{Z(s)}{1+L(s)} \frac{Kvco}{s}$$

- Noise from VCO

$$Hvco = \frac{\Phi out}{\Phi nvco} = \frac{1}{1+L(s)}$$

Noise Transfer Function

- PLL parameters
  - Fout=2.4GHz, Fref=16MHz
  - Icp=1mA, Kvco=2π*300MHz
  - R=300Ω, C=200nF, Cp=10nF



VCO Model

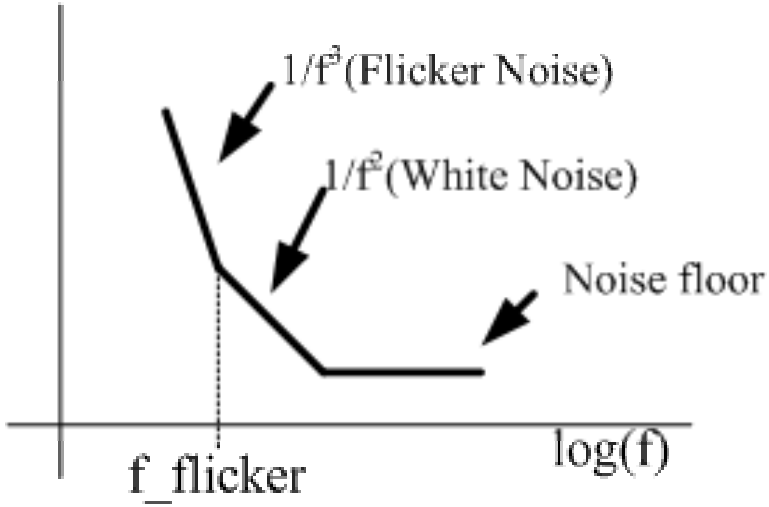
- Fundamental function

$$\Phi = \int f dt = \int (K_{vco} \times V_c) dt$$

- Noise model
  - Flicker/White noise added to the frequency

$$f = K_{vco} \times V_c + v_n(t)$$

  - $1/f^3$ and $1/f^2$ regions of the phase noise plot



White Noise Generating in Verilog-A

- white_noise(PSD,”name”)
  - PSD is Power Spectral Density in units of V^2/Hz or A^2/Hz
  - “name” identifies the source while analyzing the noise simulation results

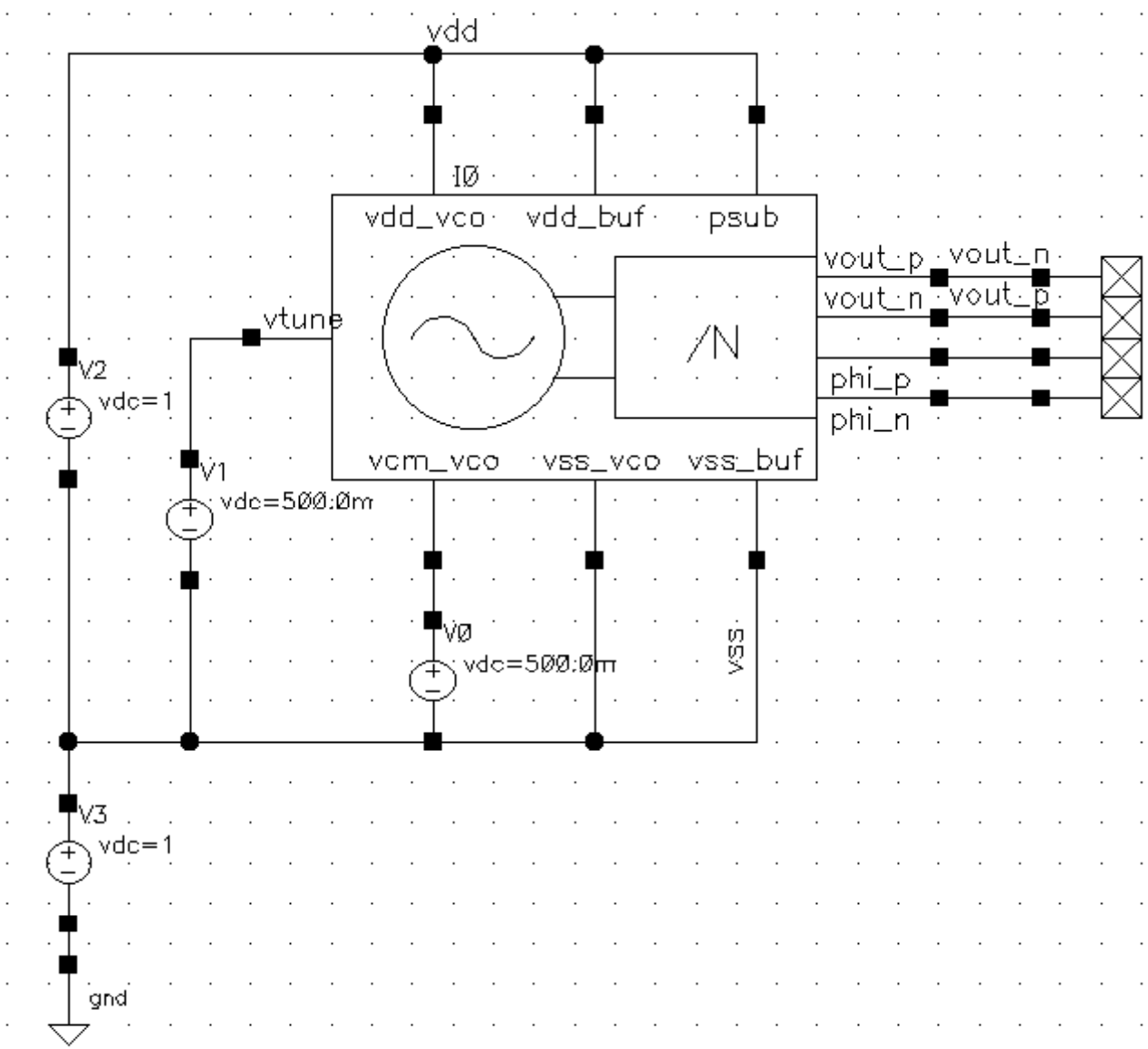
- Examples
  - V(res) <+ white_noise(4*`P_K*$temperature*Rs,”thermal”)
  - I(diode) <+ white_noise(2*`P_Q*Id,”shot”)

Flicker Noise Generating in Verilog-A

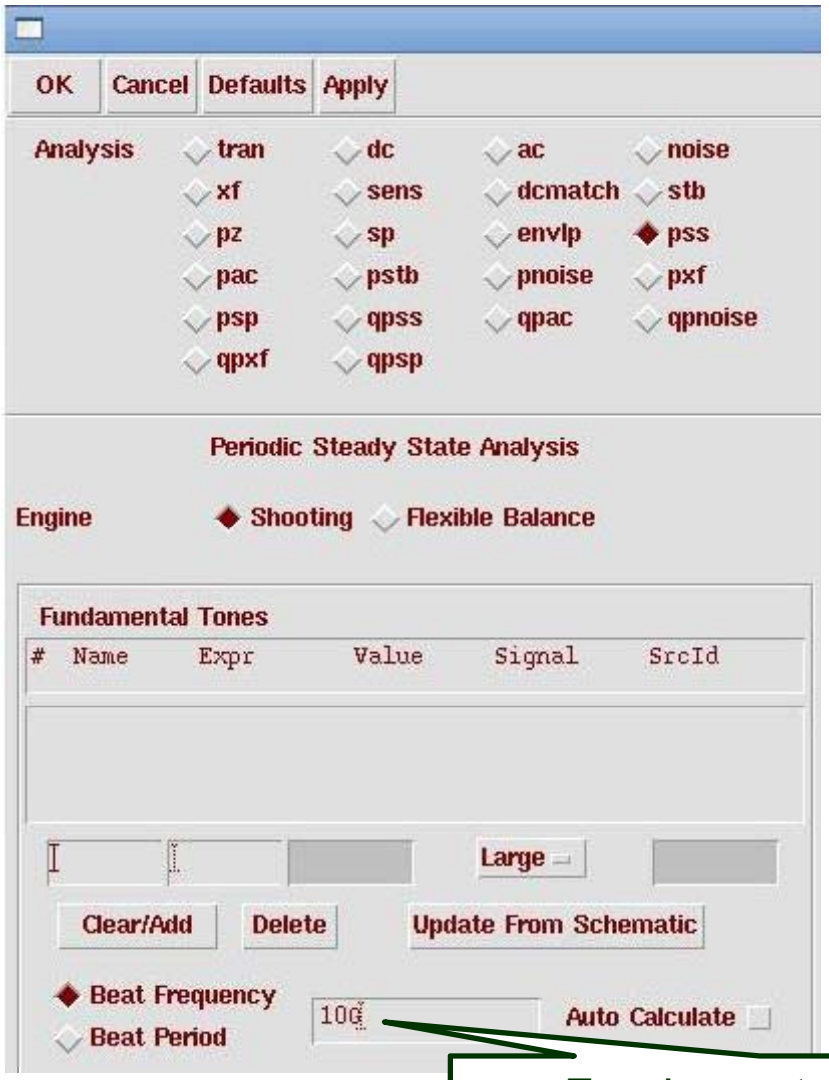
- flicker_noise(pow,n,"name")
  - pow is Power Spectral Density of the source at 1Hz in units of V^2/Hz or A^2/Hz
  - n is the order of the frequency i.e. $1/f^n$ is modeled
  - "name" is to identify the source while analyzing the noise simulation results

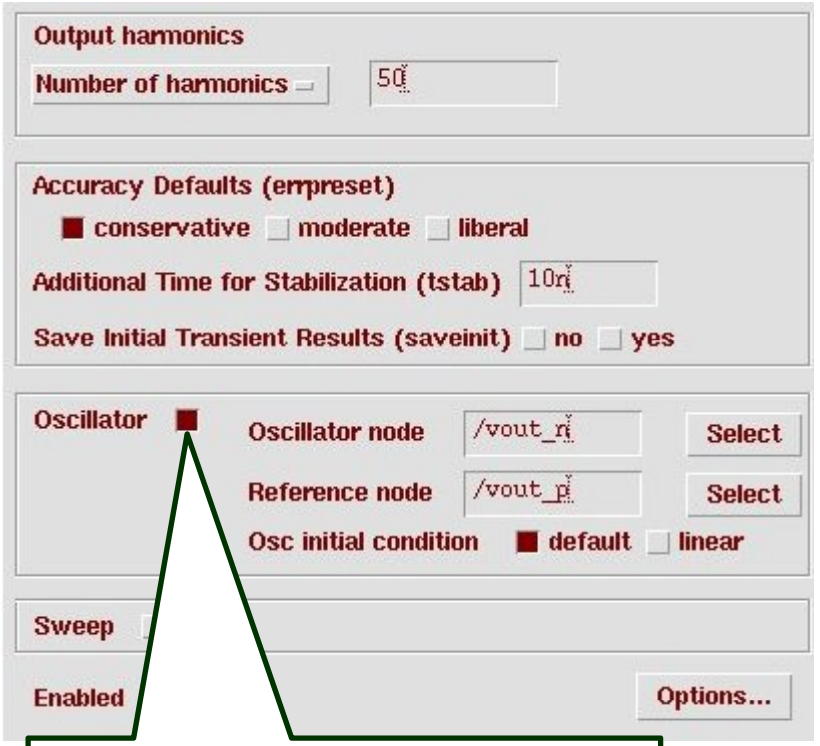
- Example of 1/f roll-off noise
  - f_flicker: corner frequency
  - wn: Desired white noise level
  - V(out) <+ flicker_noise(wn*f_flicker, 1, "fn")

VCO/Divider Test Bench



VCO PSS Simulation Setup

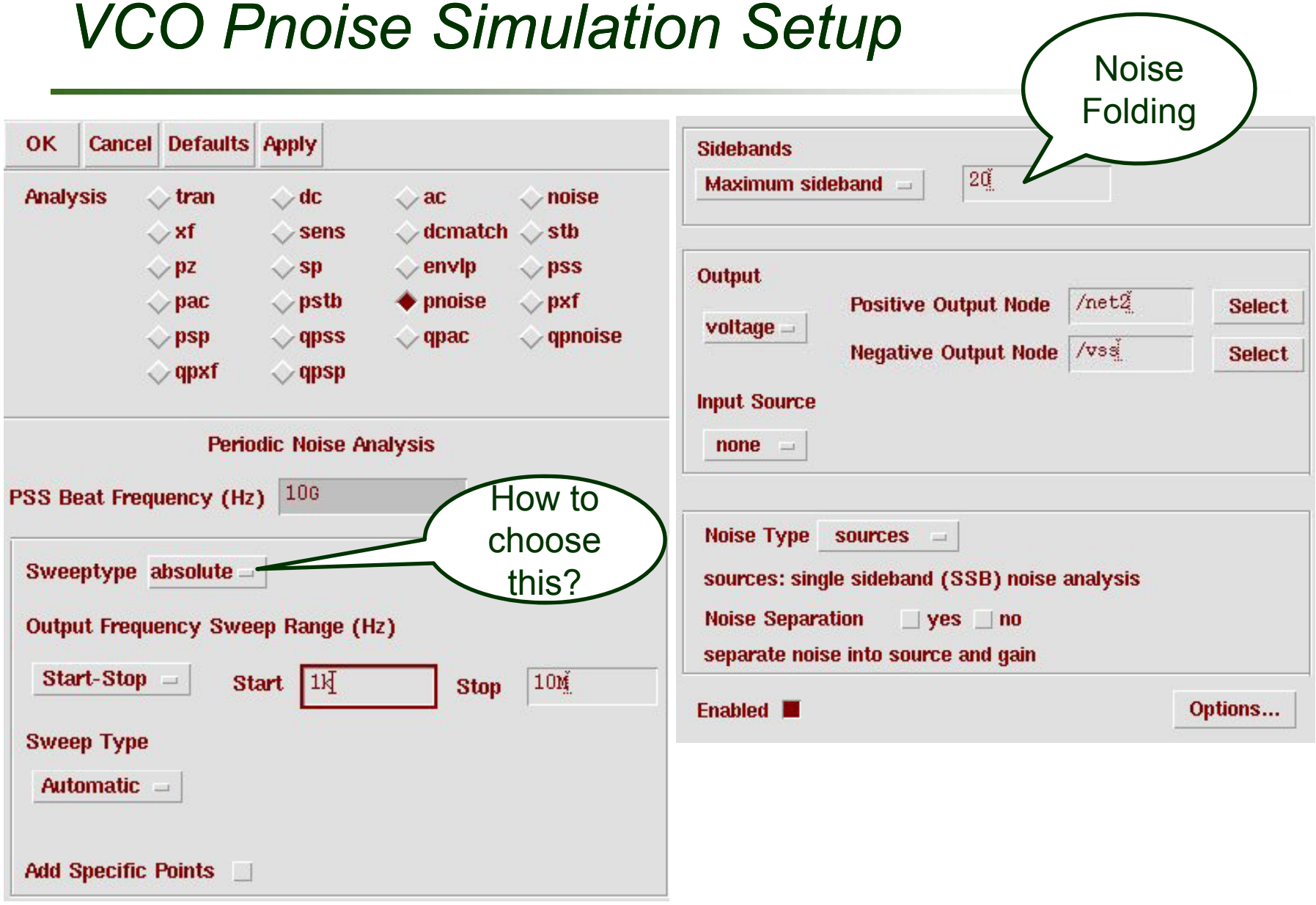




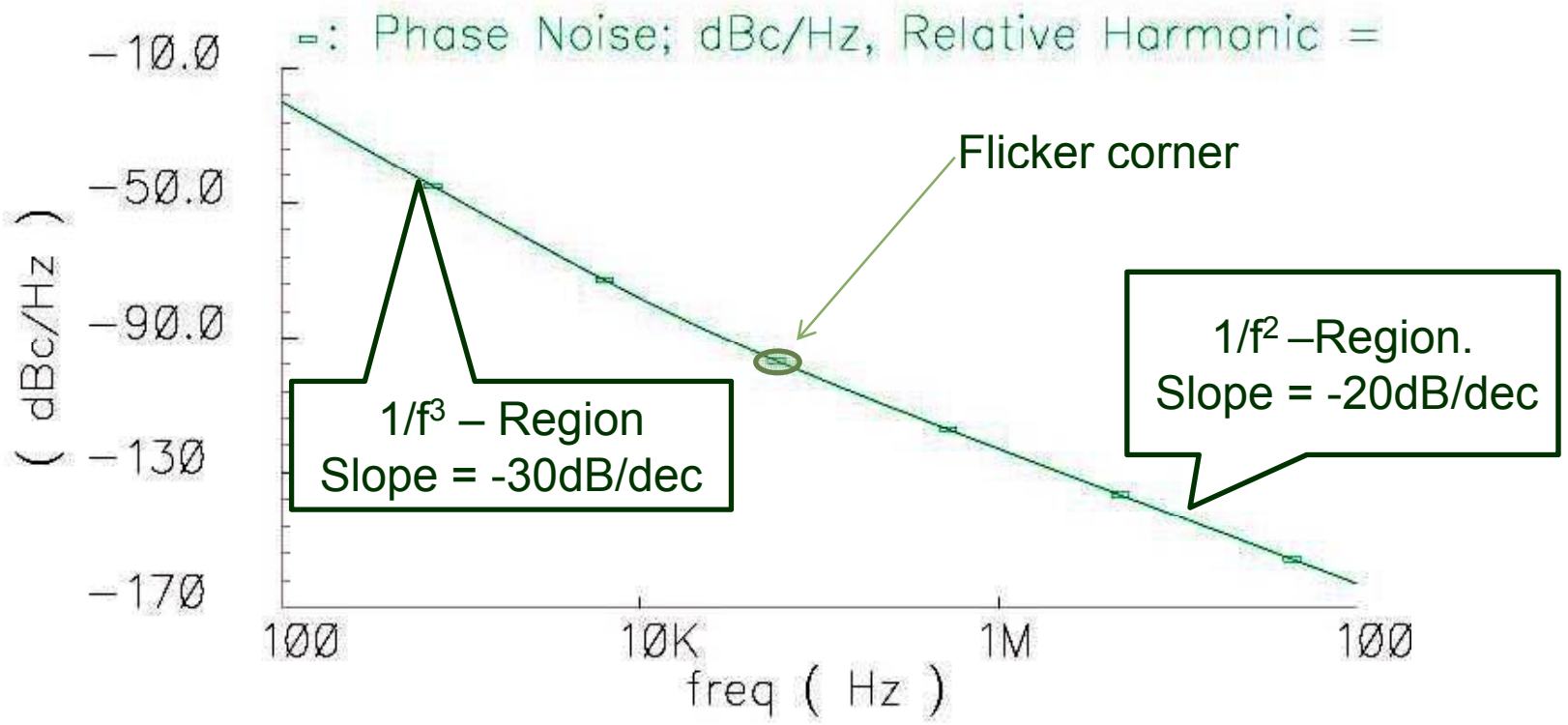
Check this for any autonomous osciallator

Fundamental Oscillation Frequency of the oscillator

VCO Pnoise Simulation Setup



VCO Simulation Results



Alternate VCO Model for Phase Noise

```
module vco(in, out);
input in; output out;
voltage in;
phase out;
parameter real gain = 1 from (0:inf);     // transfer gain, Kvco (Hz/V)
parameter real n = 0 from [0:inf);        // white output phase noise at 1 Hz (rad^2/Hz)
parameter real fc = 0 from [0:inf);       // flicker noise corner frequency (Hz)

analog begin
    Theta(out) <+ 2*`M_PI*gain*idt(V(in));
    Theta(out) <+ flicker_noise(n, 2, "wpn") + flicker_noise(n*fc, 3, "fpn");
end
endmodule
```

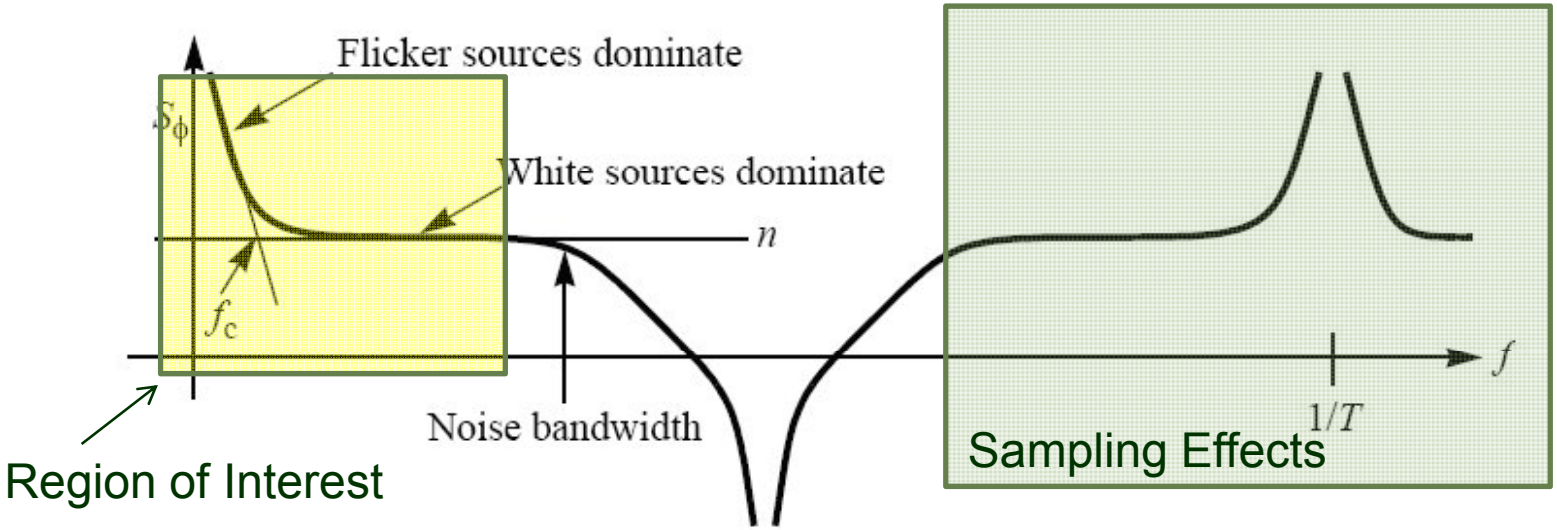
White noise ($1/f^2$)

V(out) instead of Theta(out). Discipline of "out" should be electrical/ voltage

Flicker noise ($1/f^3$)

Source: www.designers-guide.org

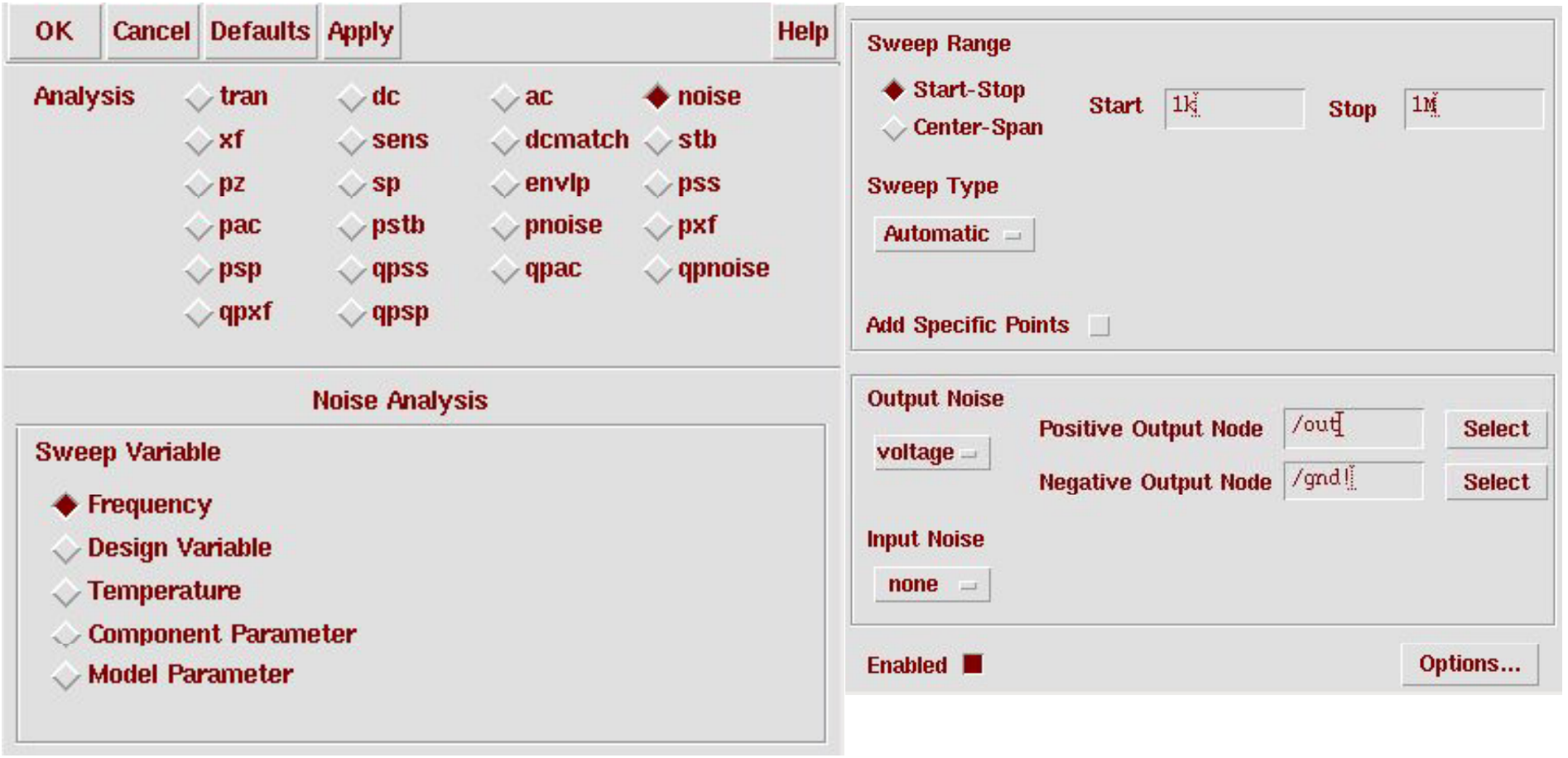
Divider Verilog-A Model



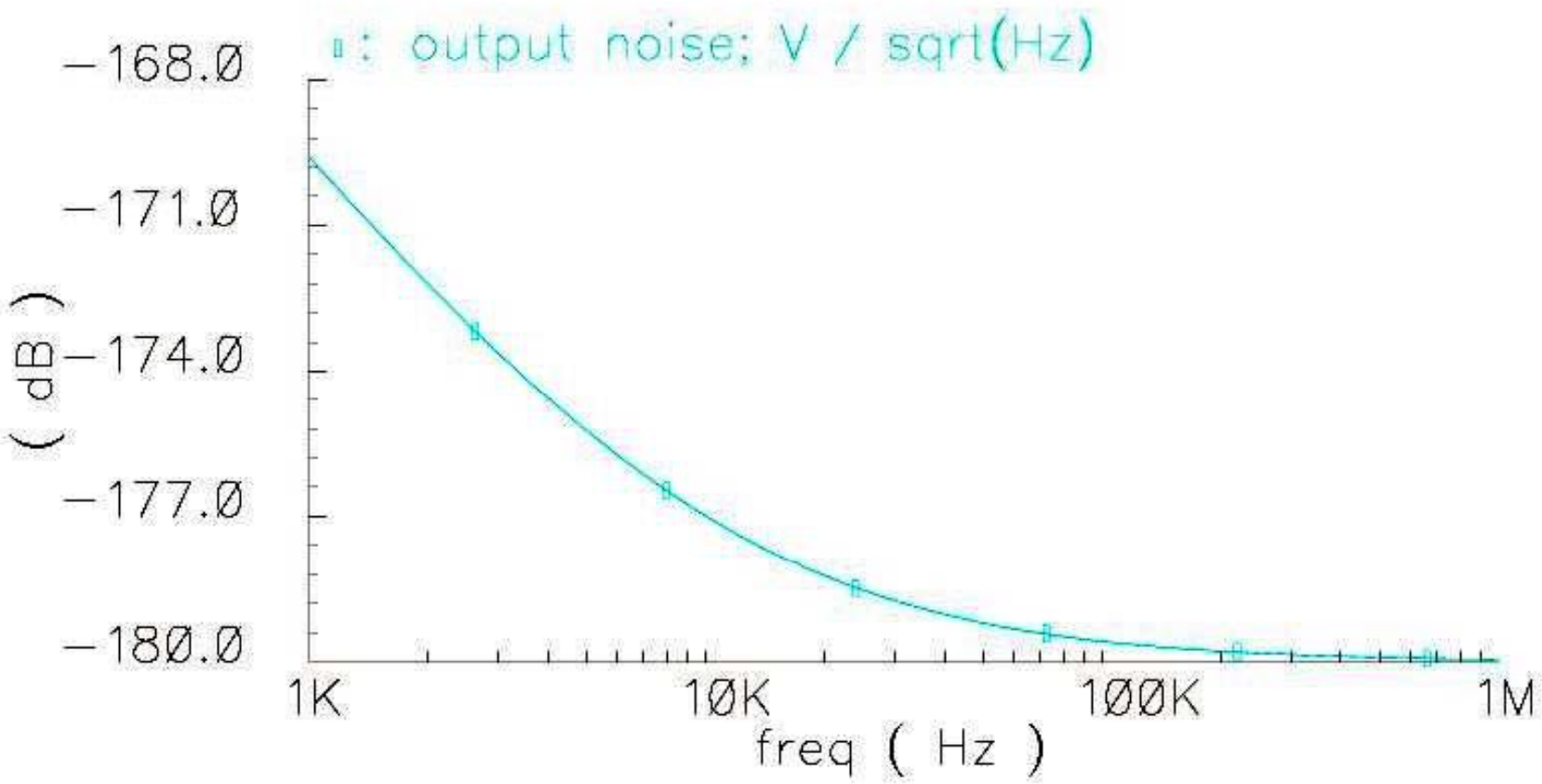

```
module divider(in, out);
input in; output out;
phase in, out;
parameter real ratio = 1 from (0:inf);      // divide ratio
parameter real n = 0 from [0:inf);          // white output phase noise (rads^2/Hz)
parameter real fc = 0 from [0:inf);         // flicker noise corner frequency (Hz)

analog begin
    Theta(out) <+ Theta(in) / ratio;
    Theta(out) <+ white_noise(n, "wpn") + flicker_noise(n*fc, 1, "fpn");
end
endmodule
```

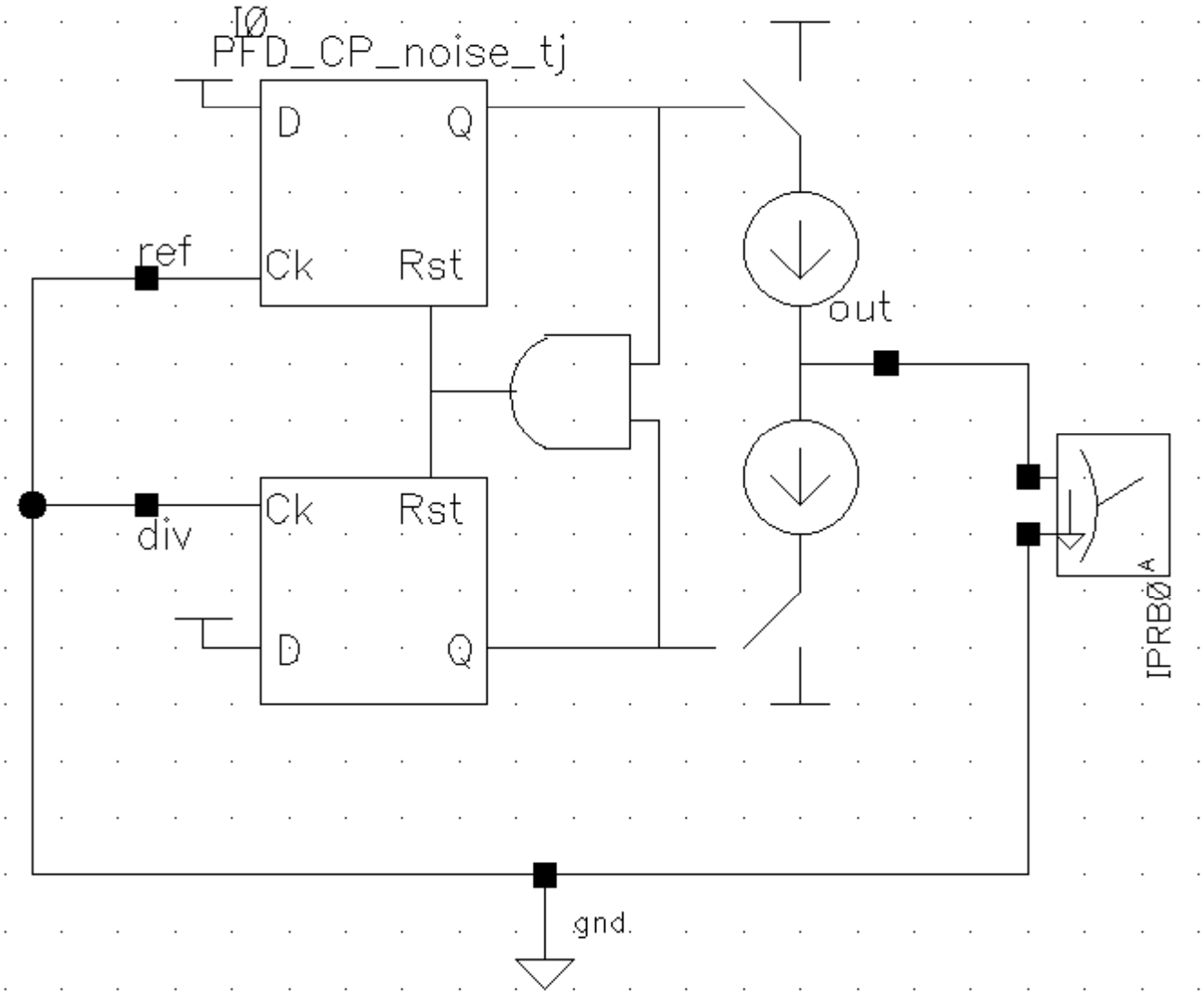
Divider Noise Simulation Setup



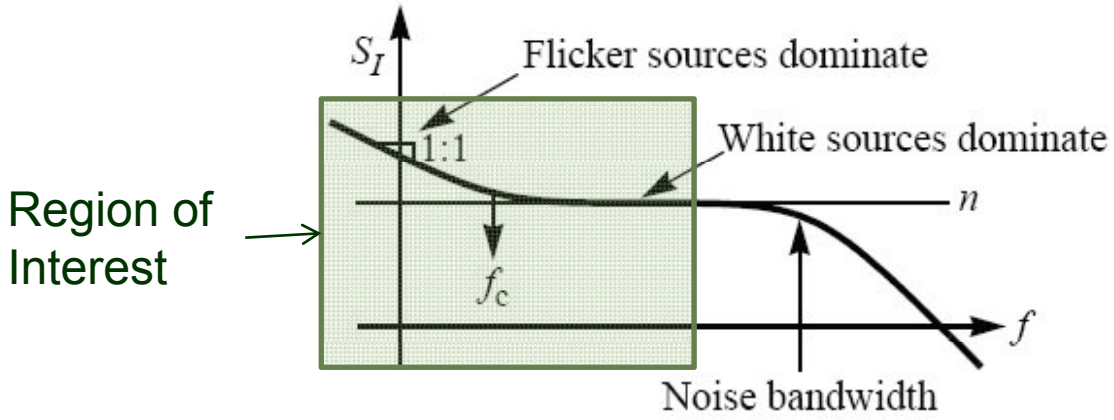
Divider Simulation Result



PFD/CP Test Bench



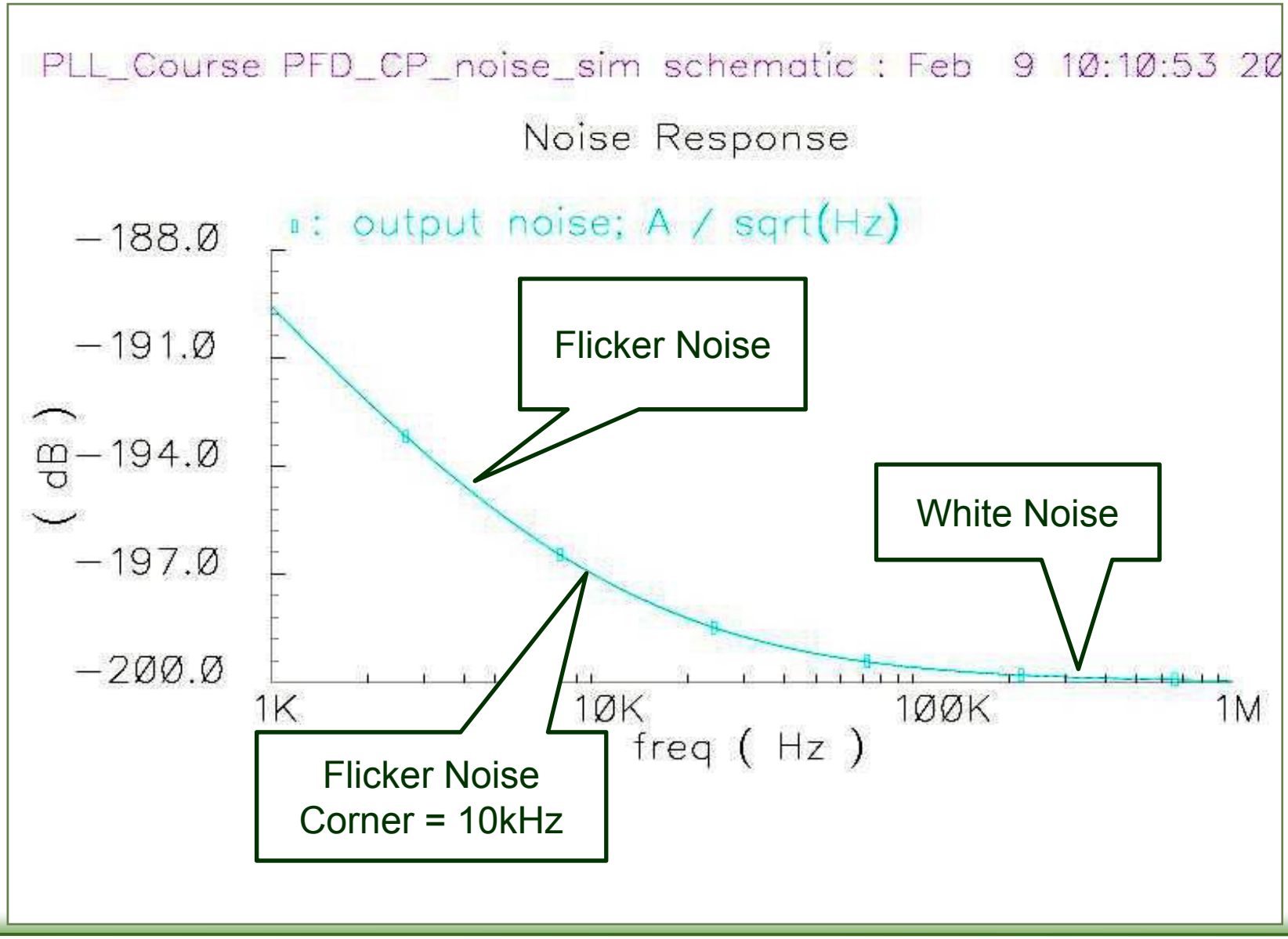
PFD/CP Verilog-A Model



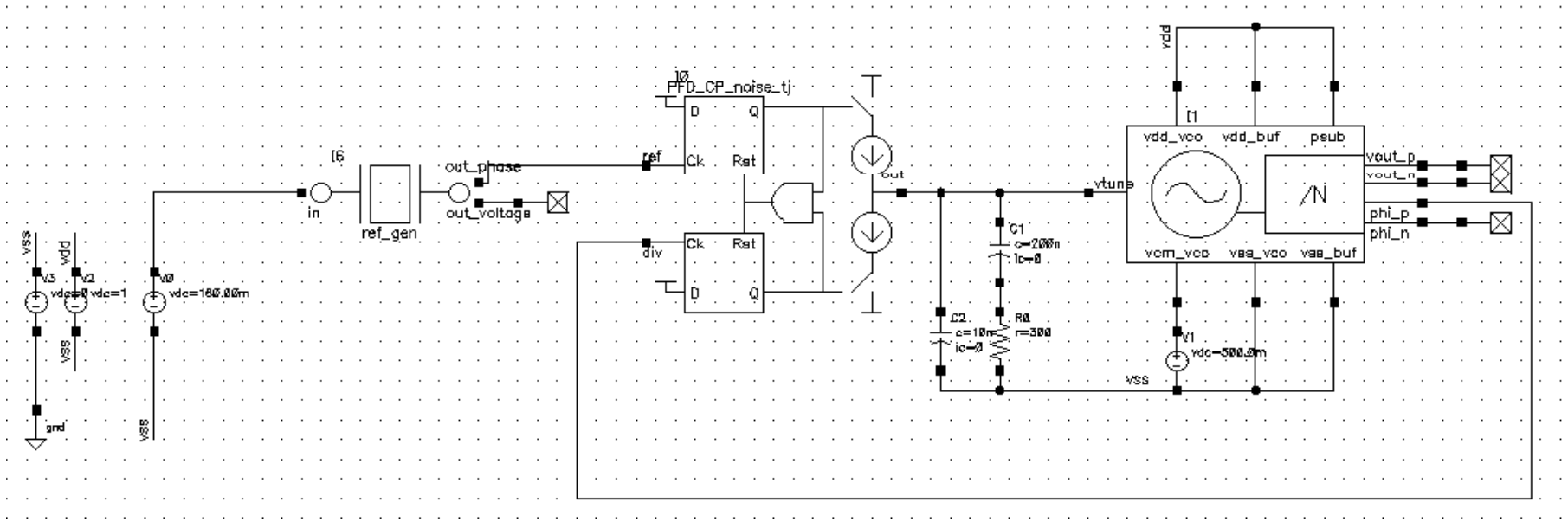

```
module phaseDetector(pin, nin, out);
input pin, nin; output out;
phase pin, nin;
electrical out;
parameter real gain = 1 from (0:inf);      // transfer gain (A/cycle)
parameter real n = 0 from [0:inf);         // white output current noise (A^2/Hz)
parameter real fc = 0 from [0:inf);        // flicker noise corner frequency (Hz)

analog begin
    I(out) <+ –gain * Theta(pin,nin) / (2*`M_PI);
    I(out) <+ white_noise(n, “wpn”) + flicker_noise(n*fc, 1, “fpn”);
end
endmodule
```

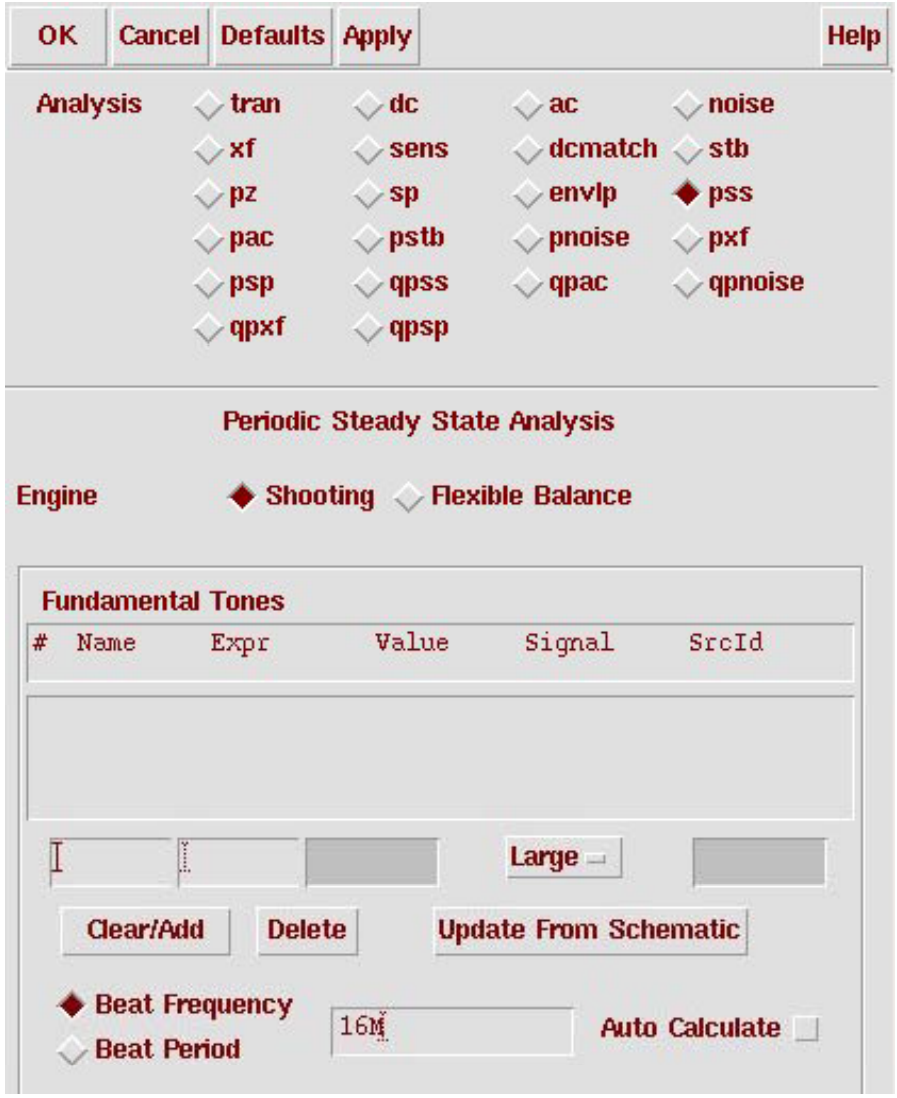
PFD Phase Noise Simulation Result

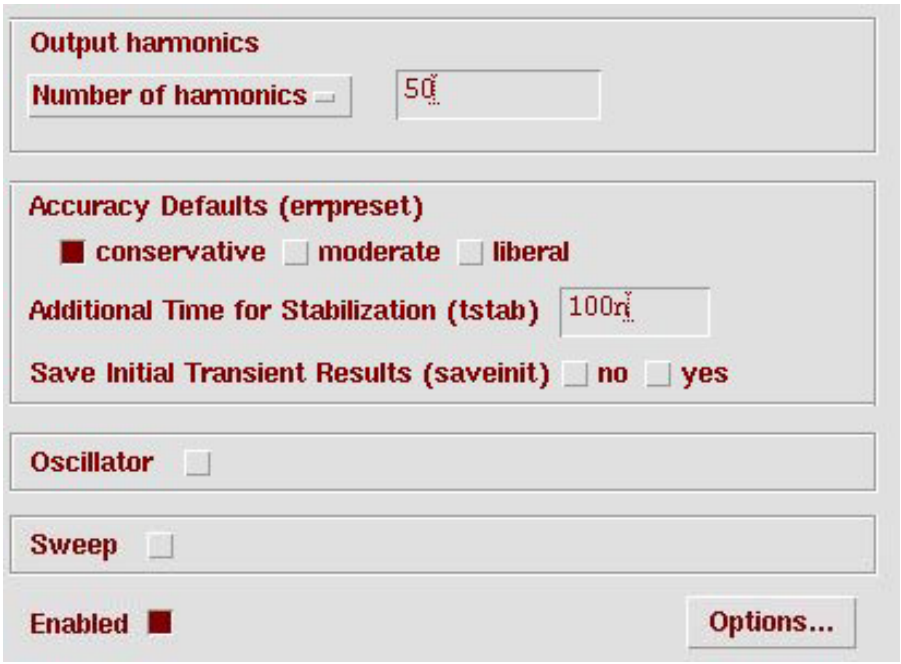


PLL Simulation Test Bench



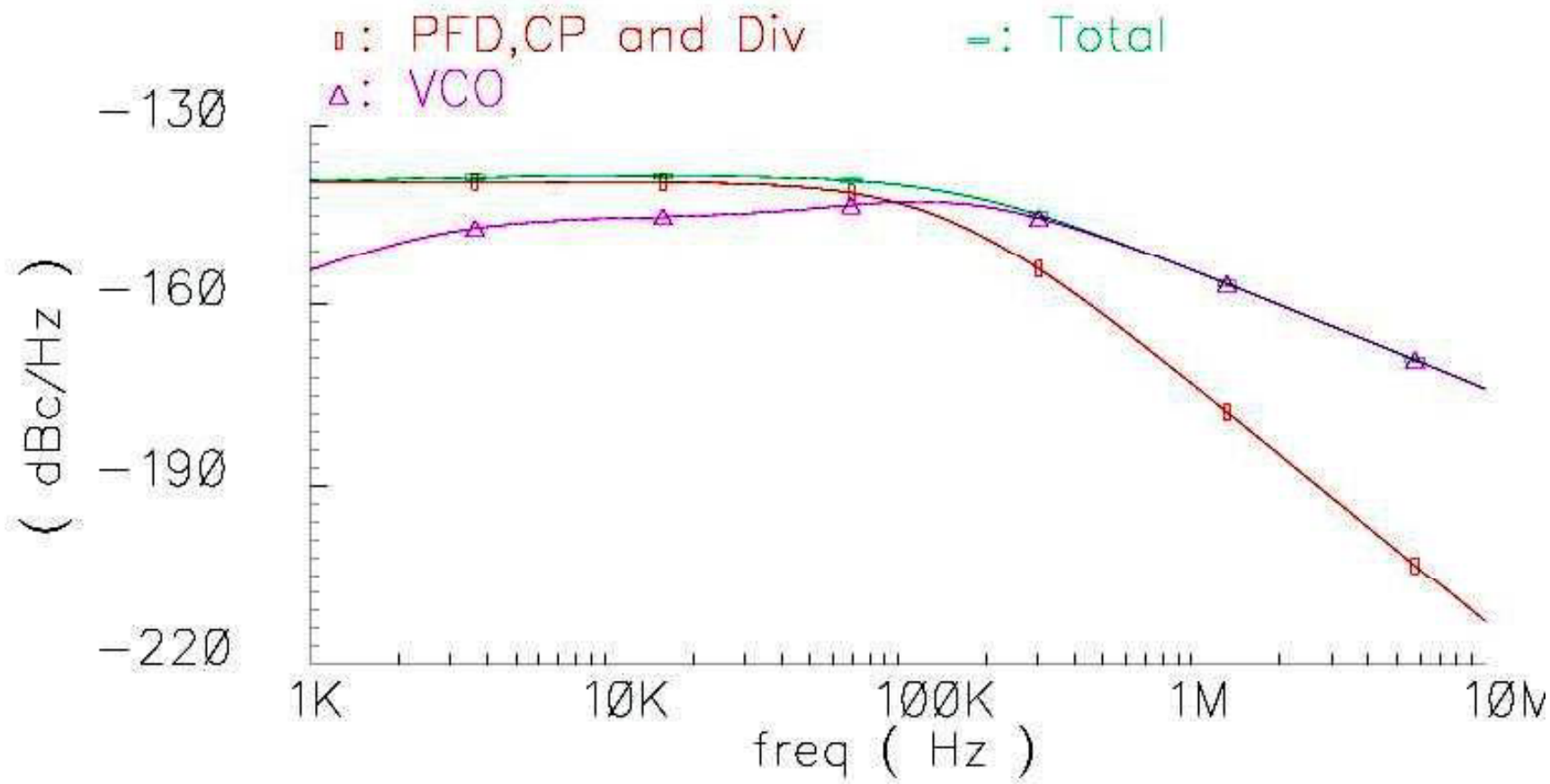
PLL PSS Analysis Setup





- PLL is a driven system so the oscillator option should be turned OFF

PLL Simulation Results



Summary

- Phase noise and its effects on RF transceiver systems were introduced
- Phase domain and voltage domain models were compared
- Verilog-A models for the different building blocks of a PLL were discussed
- Simulation setup for running phase noise simulation was demonstrated

Acknowledgements

- We would like to thank Prof. Peter Kinget for the helpful discussions on the content of this presentation and Shih-An for teaching us techniques to improve the simulation speed

- The Verilog-A code used for this presentation and those we have ever written are derived from the ones in www.designers-guide.org