Lecture 10.
Variable Domain Transformation

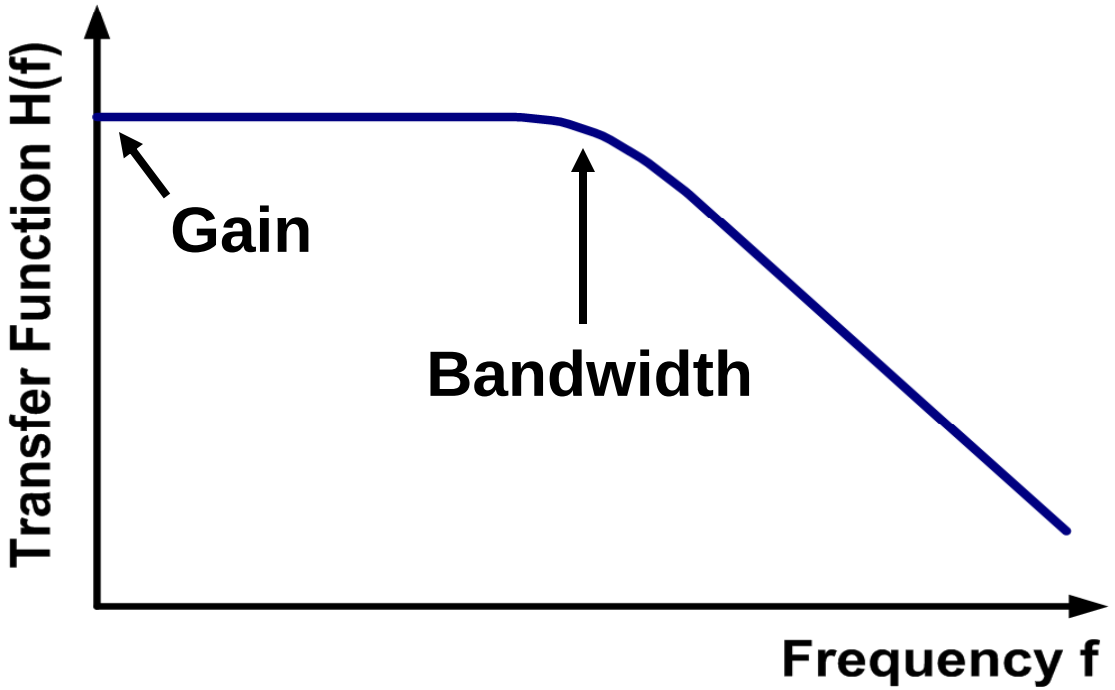
Jaeha Kim
Mixed-Signal IC and System Group (MICS)
Seoul National University
jaeha@ieee.org

Overview

- Readings
  - Jaeha Kim, et al., “Variable Domain Transformation for Linear PAC Analysis of Mixed-Signal Systems”, ICCAD 2007.
- Background
  - We have seen that PSS and PAC analyses are effective tools for analyzing RF circuits which are periodic and whose signals are narrowband. In this lecture, we will find that the PAC analysis is also useful for analyzing systems that are linear in variables other than voltage or current. For example, we can use PAC analysis to directly simulate the phase-domain transfer function of a PLL.

AC Analysis: Formal for Analog

- Most efficient in verifying linear intent
- Transfer function: complete description of the circuit's response to arbitrary small-signal inputs; thus formal

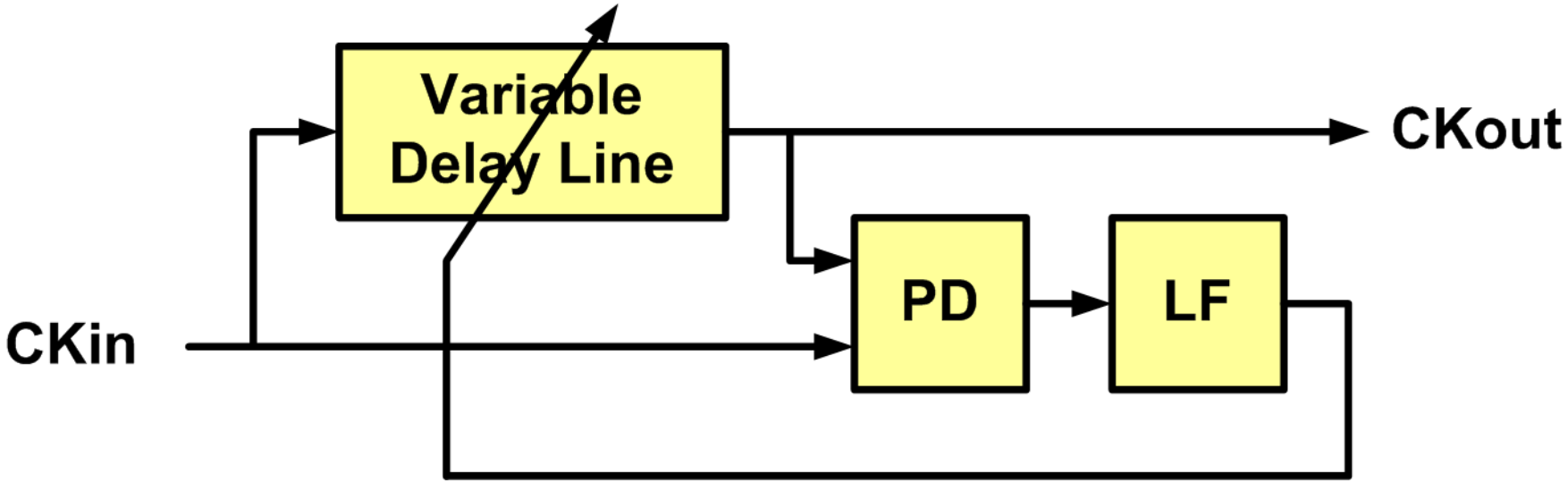


Periodic AC Analysis

- AC analysis assumes a DC op. point
  - Thus, limited to amplifiers, filters, passive networks, bandgap references, etc.
- Periodic AC (PAC) analysis linearizes circuit on a periodic steady-state (PSS)
  - Extends the reach of linear analysis to “time-varying” linear systems
  - Ex: switched-C filters, mixers, etc.
  - RF simulators: SpectreRF, ADS, …

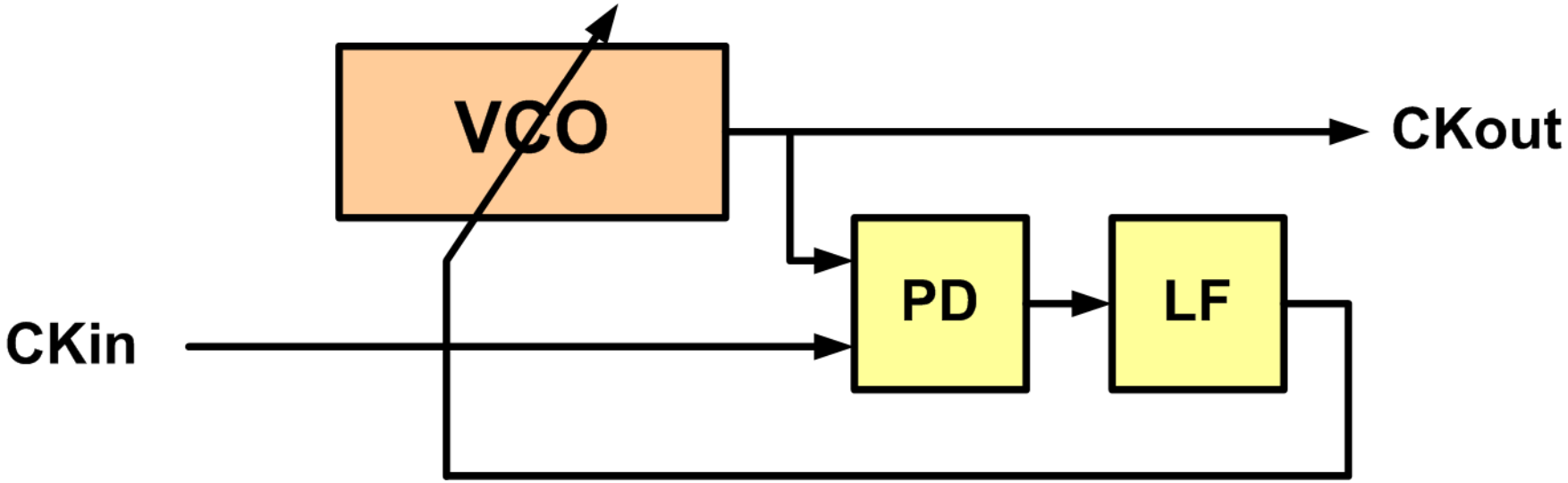
Clocking Systems

- PLL, DLL, duty-cycle corrector (DCC), …
- Intent is still linear; just not in voltage
- Example: DLL: linear in delay



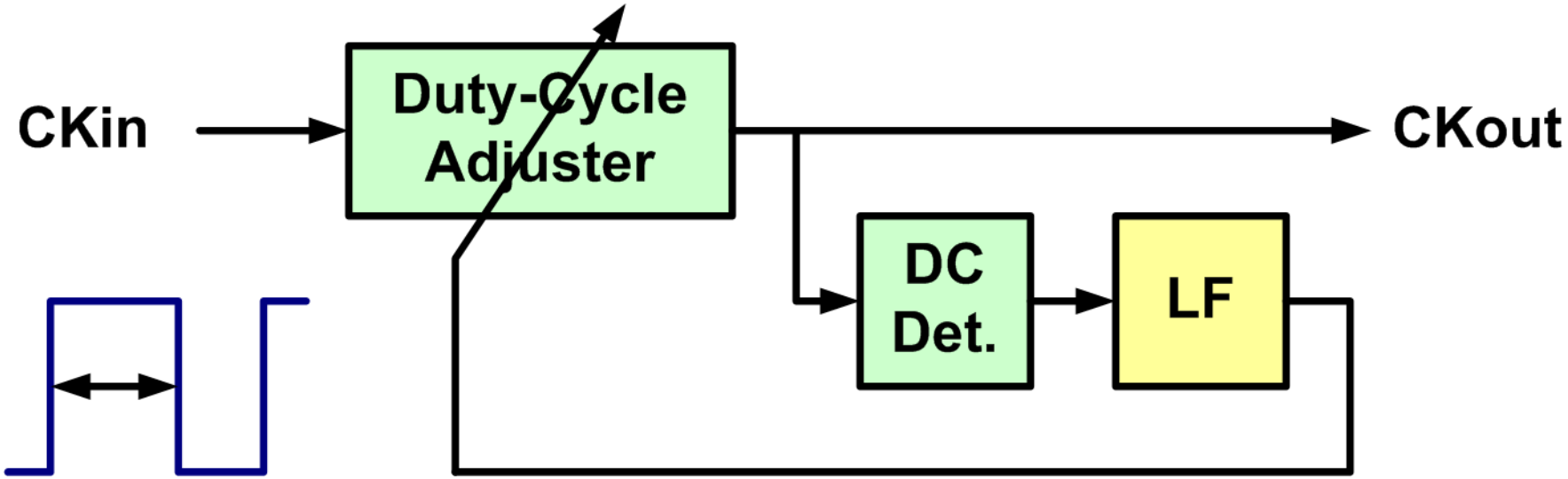
Clocking Systems

- PLL, DLL, duty-cycle corrector (DCC), …
- Intent is still linear; just not in voltage
- Example: PLL: linear in phase



Clocking Systems

- PLL, DLL, duty-cycle corrector (DCC), …
- Intent is still linear; just not in voltage
- Example: DCC: linear in duty-cycle

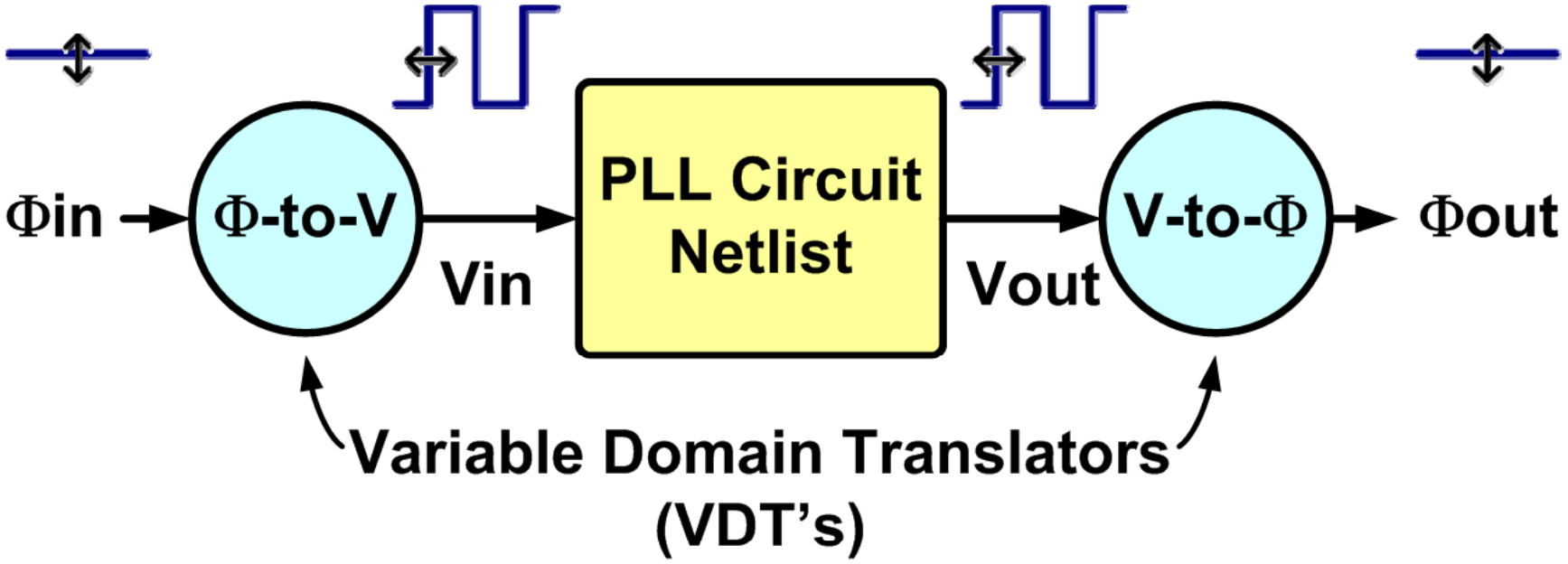


Problem Statement

- Can we do linear analysis on PLL, DLL, & DCC using PAC analysis?
- Must do in phase, delay, and duty-cycle
- But, simulators do AC/PAC in V/I only

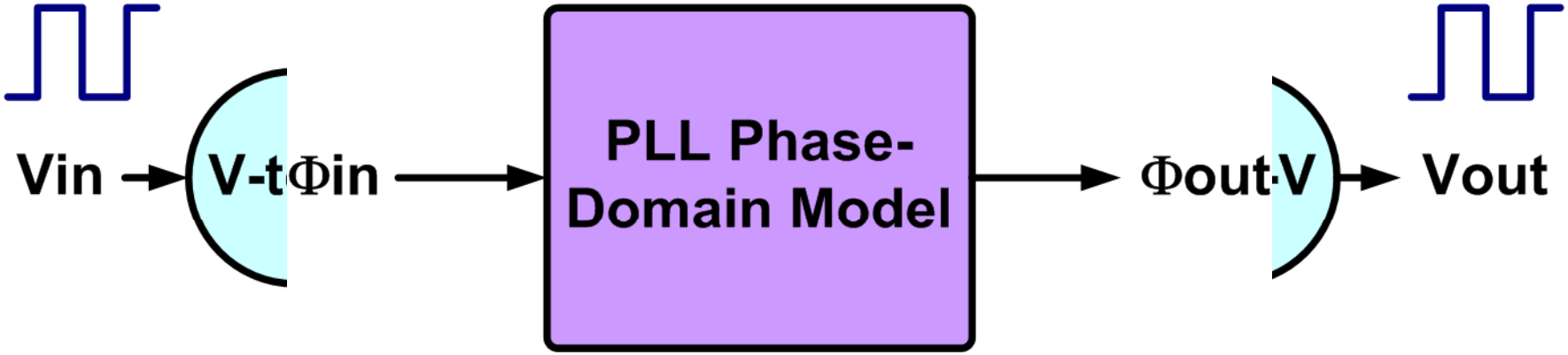
Variable Domain Transformation

- Variable Domain Translators (VDTs): convert between V $\leftrightarrow \Phi$, f, D, and DC
- Perform PAC analysis in new variables



Domain Transformation for Modeling

- Model a PLL as linear in phase domain
  - Based on weakly nonlinear models
- Use VDTs to convert to voltage domain



Variable Domain Translators

- Implemented in Verilog-A
- Ex: Phase-to-V translator

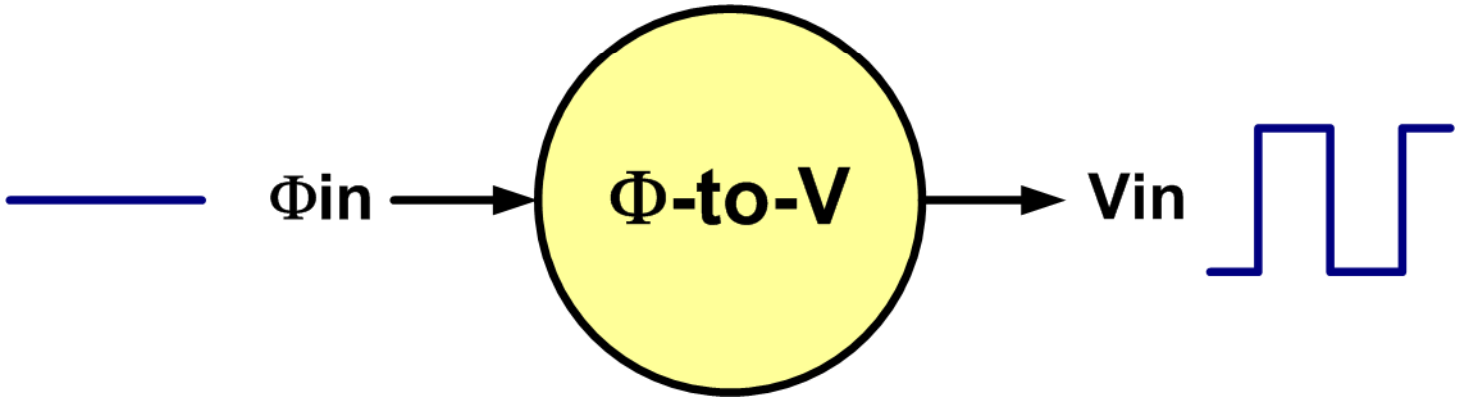
```
module phase2v (phin, vout);
input phin;
```

BUT IT DOES NOT WORK!

```
analog begin
   V(sine) <+ cos(2*`M_PI*freq*$abstime + V(phin));
   if (V(sine) > 0.0) V(vout) <+ Vhigh;
   else V(vout) <+ Vlow;
end

endmodule
```

Requirements for VDTs (1)

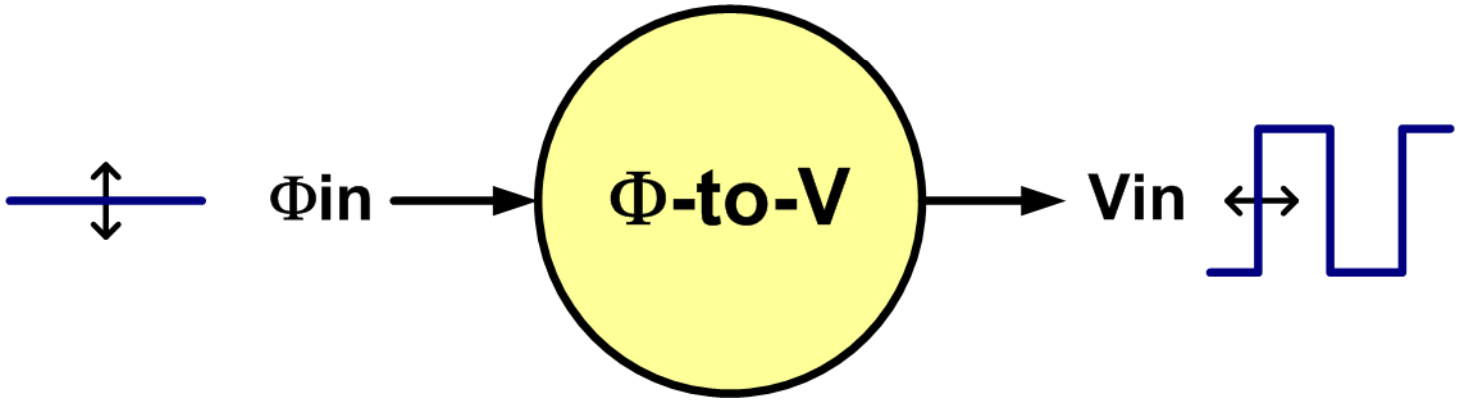
- Correct large-signal, transient response
- Required for PSS and TRAN analyses



$$\mathbf{g}(\mathbf{v}, \dot{\mathbf{v}}) = \boldsymbol{\varphi}$$

Requirements for VDTs (2)

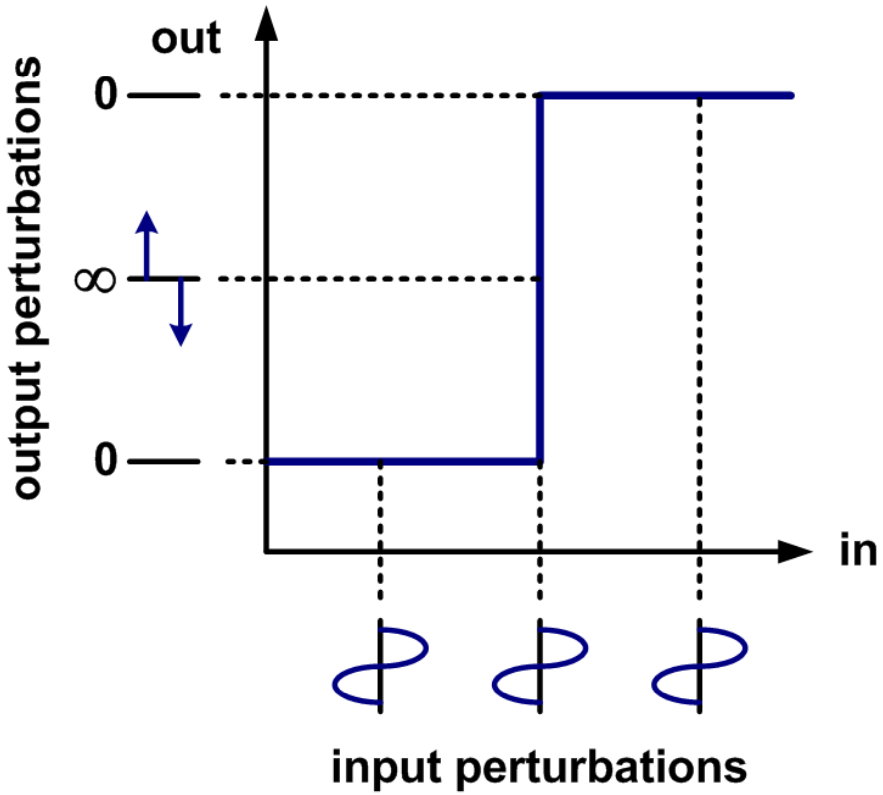
- Correct propagation of perturbations
- Required for PAC and PNOISE analyses

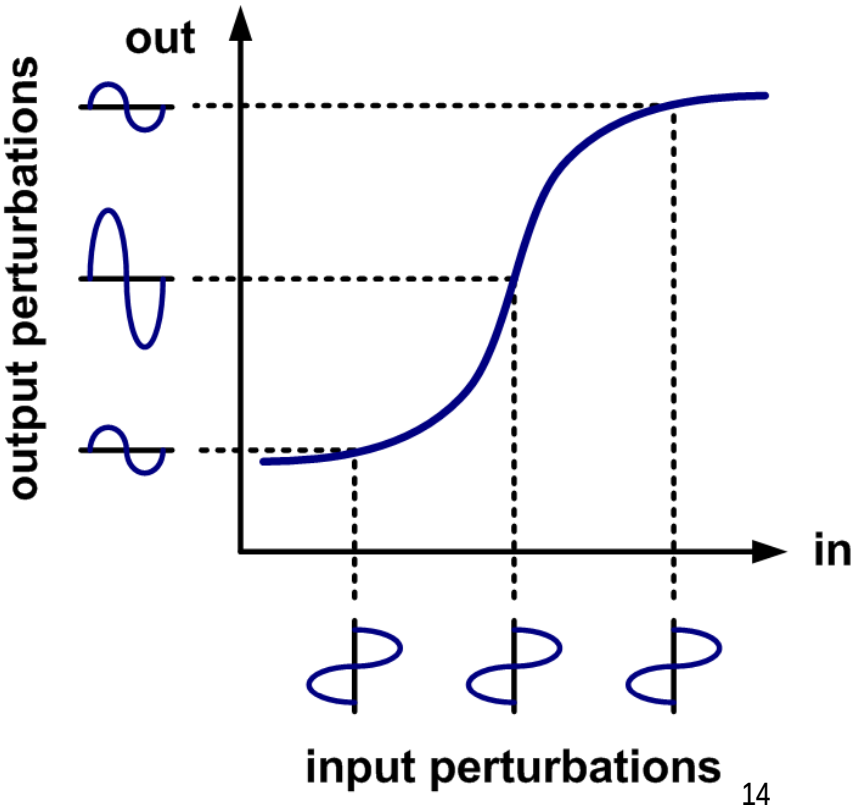


$$\left.\frac{\partial \mathbf{g}}{\partial \mathbf{v}}\right|_{\mathbf{v_S}} \boldsymbol{\delta}\mathbf{v} + \left.\frac{\partial \mathbf{g}}{\partial \dot{\mathbf{v}}}\right|_{\dot{\mathbf{v}}_S} \boldsymbol{\delta}\dot{\mathbf{v}} = \boldsymbol{\delta}\boldsymbol{\varphi}$$

Ideal Slicer Blocks Perturbation

- Hard-decision elements: if-statement, sign(), event-triggers, etc.





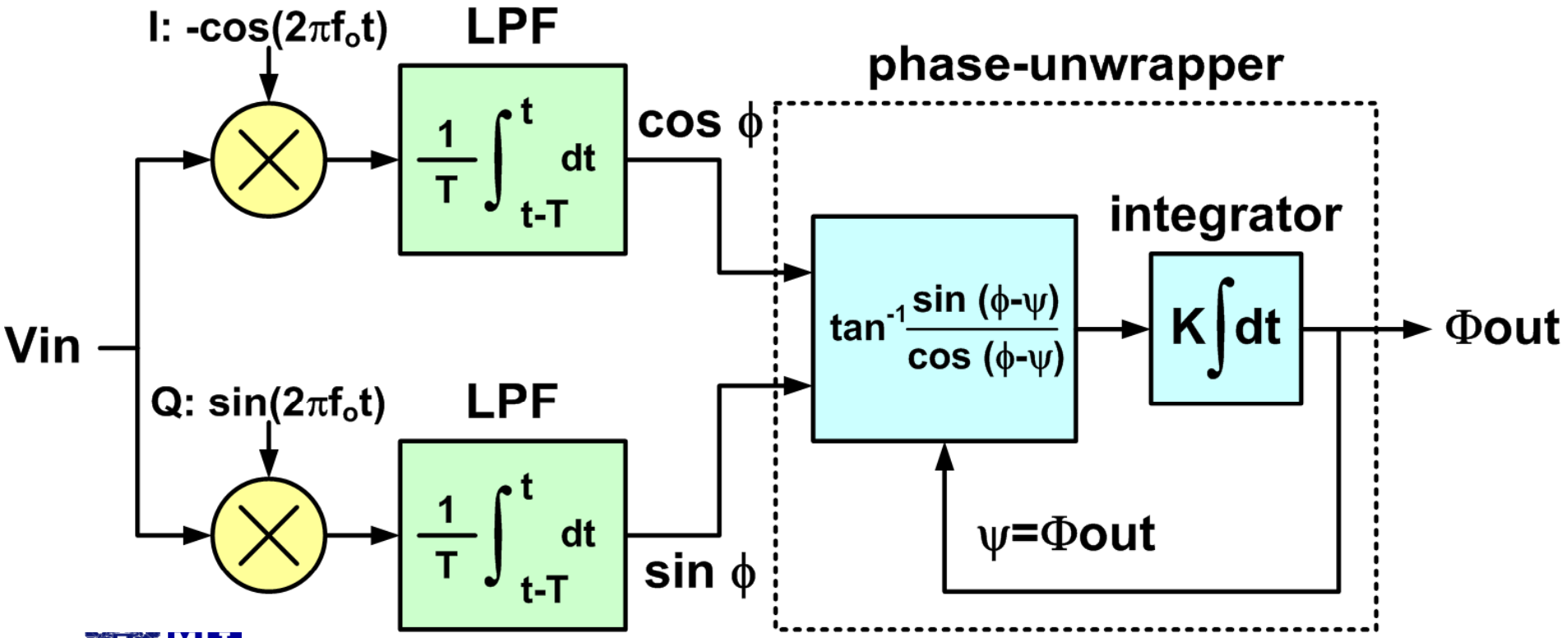
Revised Phase-to-V Translator

- Use tanh() instead of the ideal slicer
- alpha $\propto$ clock edge rate

```
module phase2v (phin, vout);
input phin;
output vout;

// parameter definitions omitted for brevity

analog begin
   V(sine) <+ cos(2*`M_PI*freq*$abstime + V(phin));
   V(vout) <+ Voffset +
              Vamplitude*tanh(alpha*V(sine));
end

endmodule
```

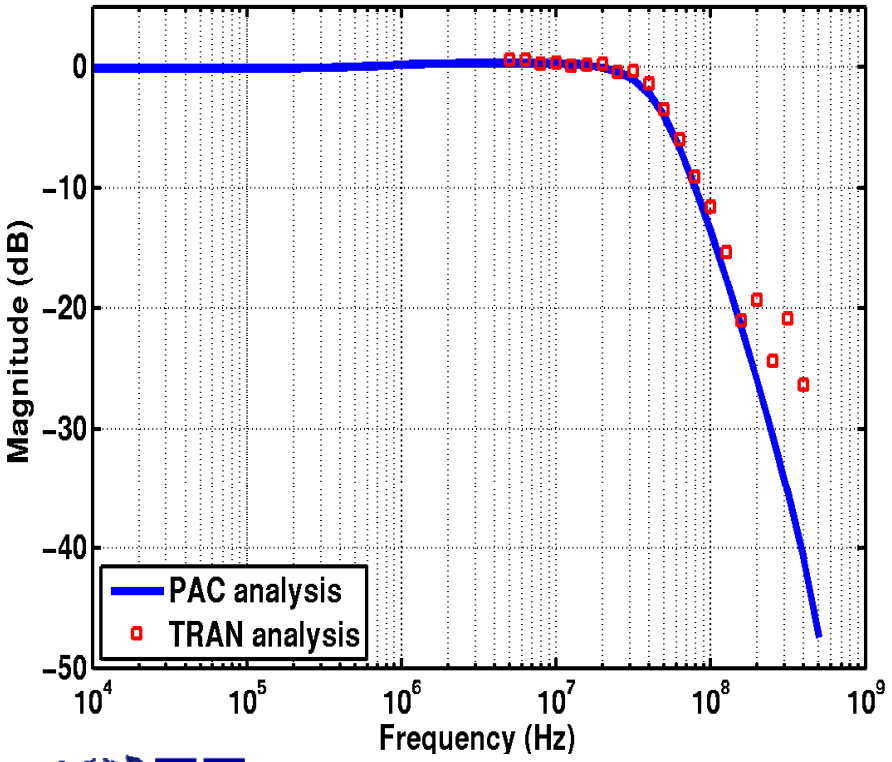
V-to-Phase Translator

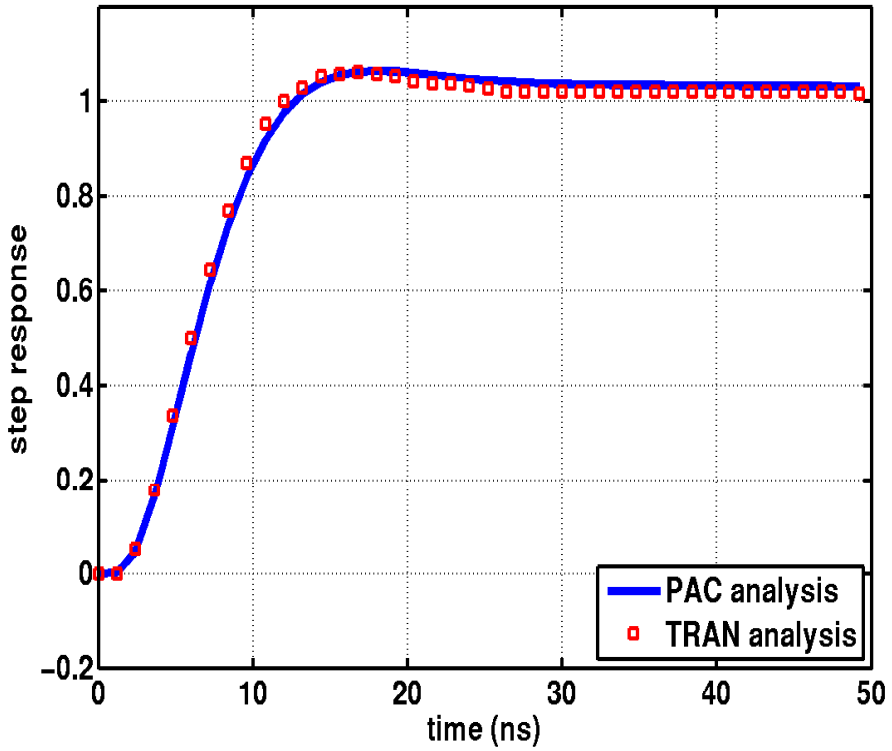
- Basically a phase detector
- I/Q-demodulation extracts the phase



PLL: Phase Transfer Analysis

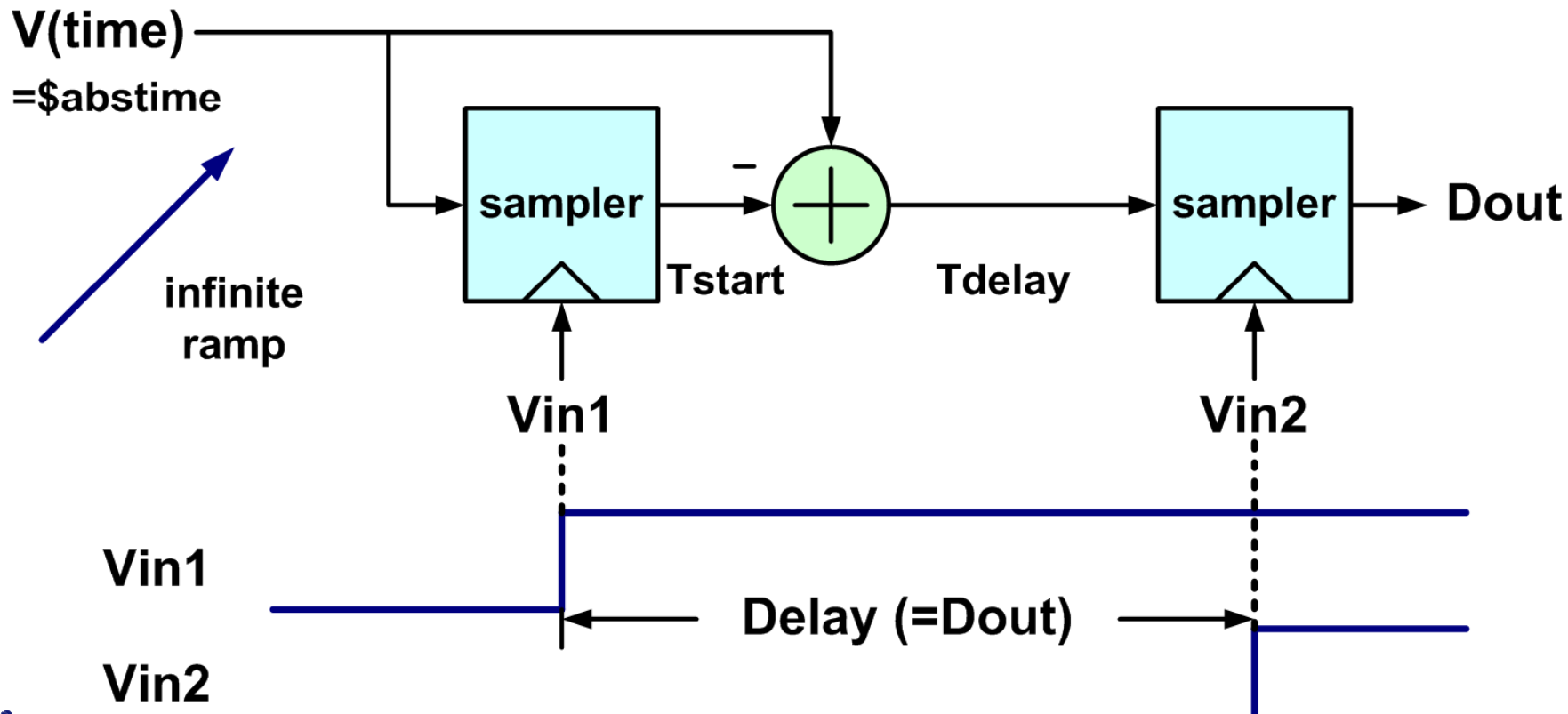
- Compared to 2 transient-based results
  - Frequency sweep of sinusoidal excitation
  - Step response and estimate the system





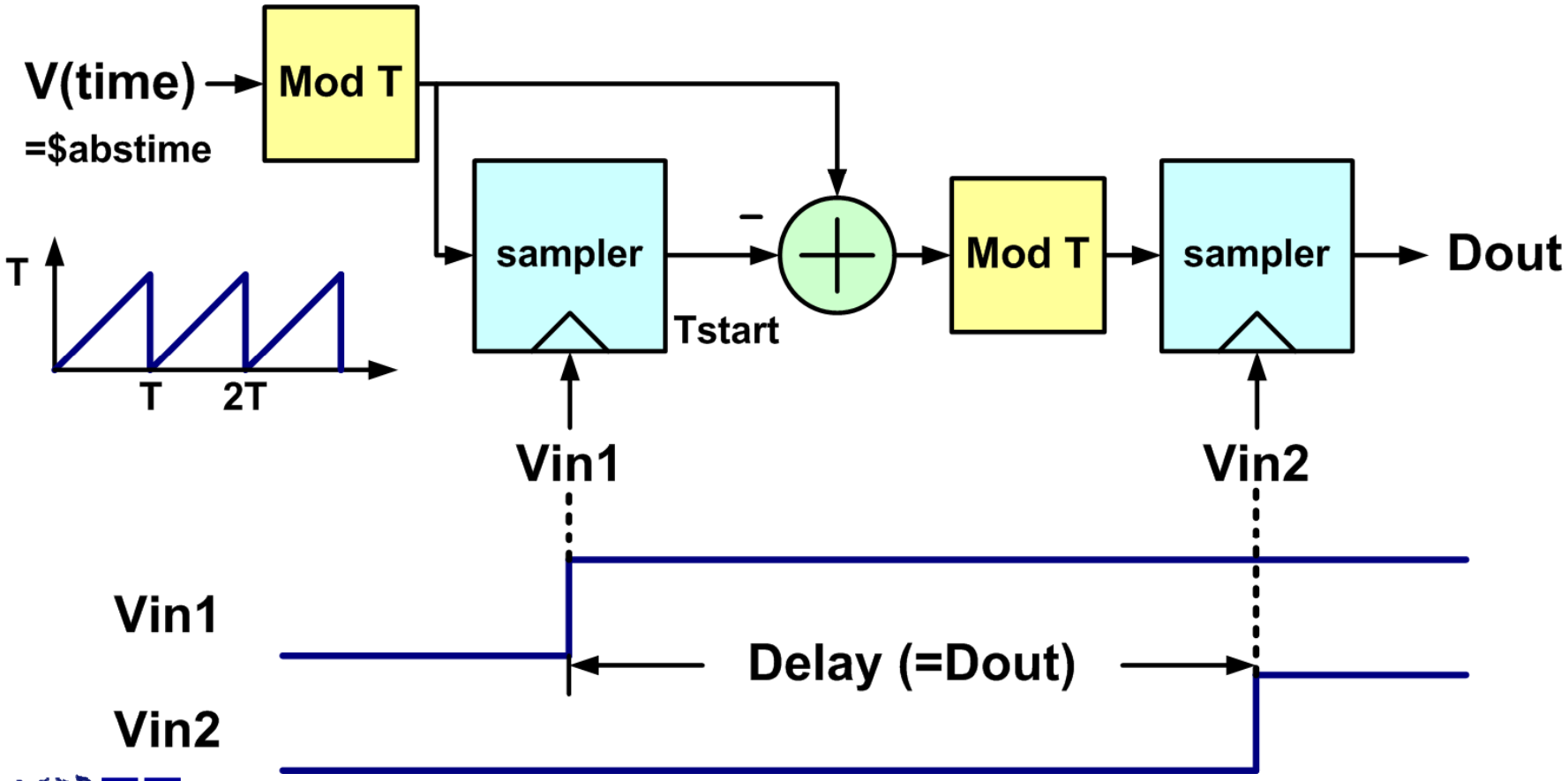
V-to-Delay Translator

- Sample the time signal V(time) to record the beginning & end of the delay



V-to-Delay Translator

- Insert modulo T to create periodic SS

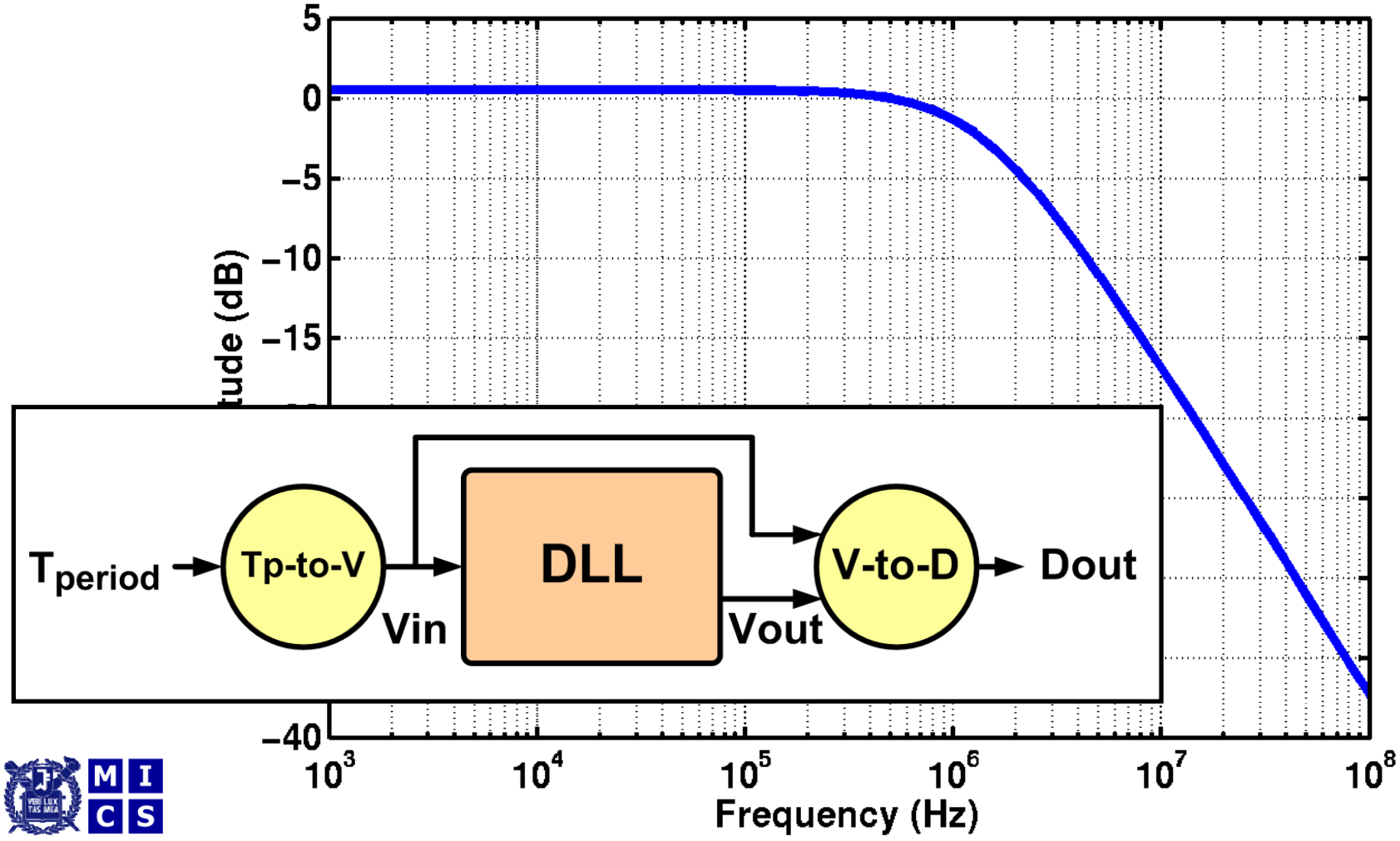


Track-and-Hold Model

```
module track_hold (in, out, clk);
input in, clk;
output out;

// parameter definitions omitted for brevity

analog begin
   @(cross(V(clk)));
   Rsw = 1.0 + 1.0e+12/(1+limexp(V(clk)/maxslope));
   I(hold) <+ (V(hold) - V(in)) / Rsw;
   I(hold) <+ C_hold * ddt(V(hold));

   V(out) <+ V(hold);
   V(out) <+ I(out) * 1.0e+12;
end

endmodule
```

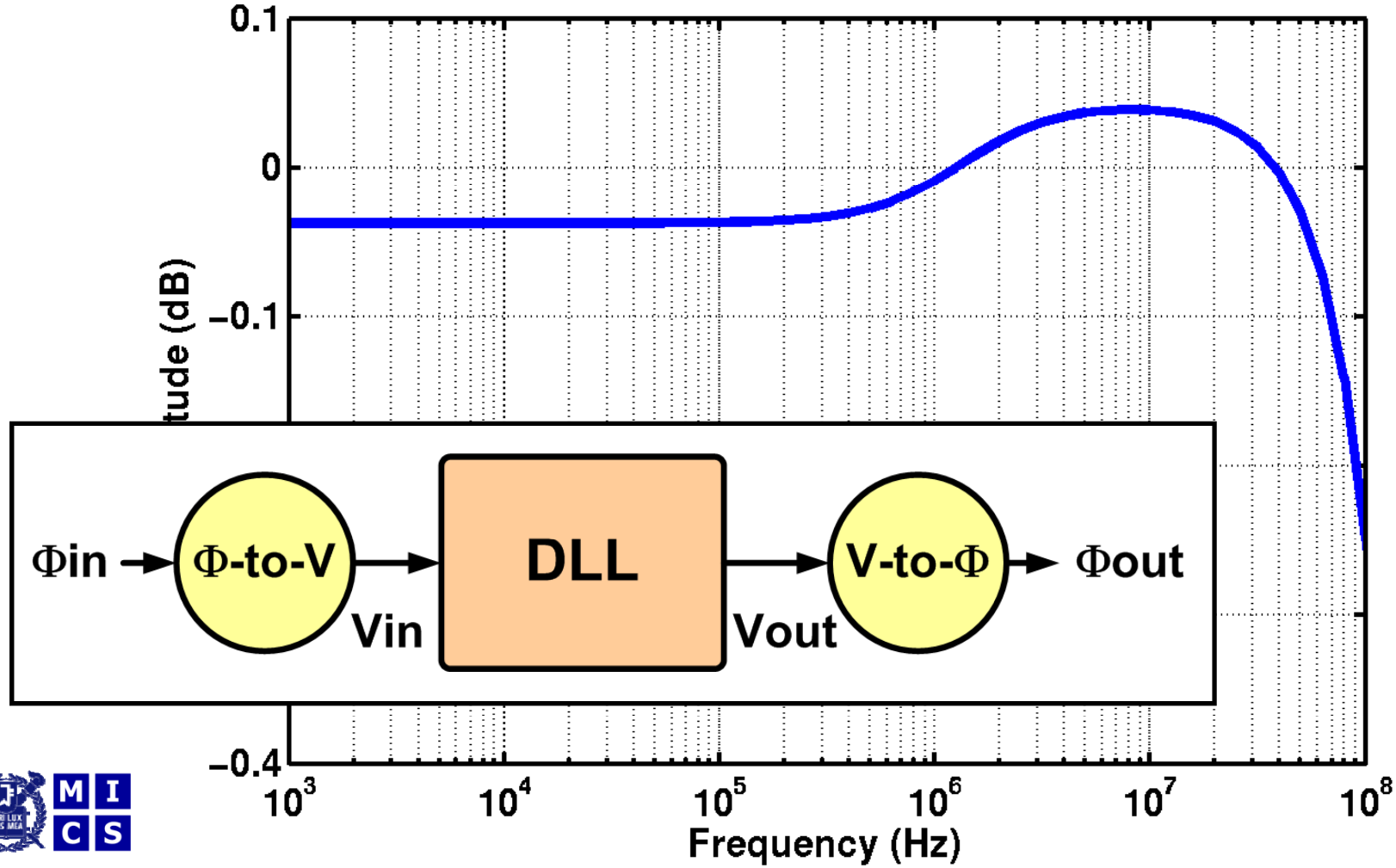
DLL: Delay Transfer Analysis

- Common design spec for DLL (e.g. BW)



DLL: Phase Transfer Analysis

- A DLL may *amplify* jitter (Type-I DLL)



Benchmark Summary

Test Circuit	Transistor Count	Simulation Time		
		Proposed	Step Resp.	20-pt Sinusoid. Sweep
PLL	341	72 sec	1496 sec	30540 sec
DLL	411	66 sec	1045 sec	22220 sec
DCC	88	41 sec	174 sec	3730 sec

- 4~20x speed up vs. step response and 50~90x vs. 20-point sinusoidal sweep
- 3.6GHz Intel Xeon with 4GB memory

Conclusions

- Most analog circuits are designed with linear intent
  - And most efficiently verified with AC sim
  - But may need to change the variables
- With variable domain translators, linear PAC analysis can be performed in variables other than voltage or current
  - Phase, frequency, delay, and duty-cycle
  - To verify the linear intent of PLL, DLL, DCC