Automated Macromodelling of Systems with Nonlinear Blocks

Jaijeet Roychowdhury
University of Minnesota

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Automated Macromodelling

- The dream: **push-button bottom-up model generation**
  - Rob's Red Button
  - prescribed accuracy guaranteed
  - trade off **speed vs accuracy**
- Needed for design sustainability
  - complexity exceeds manual ability to keep up

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

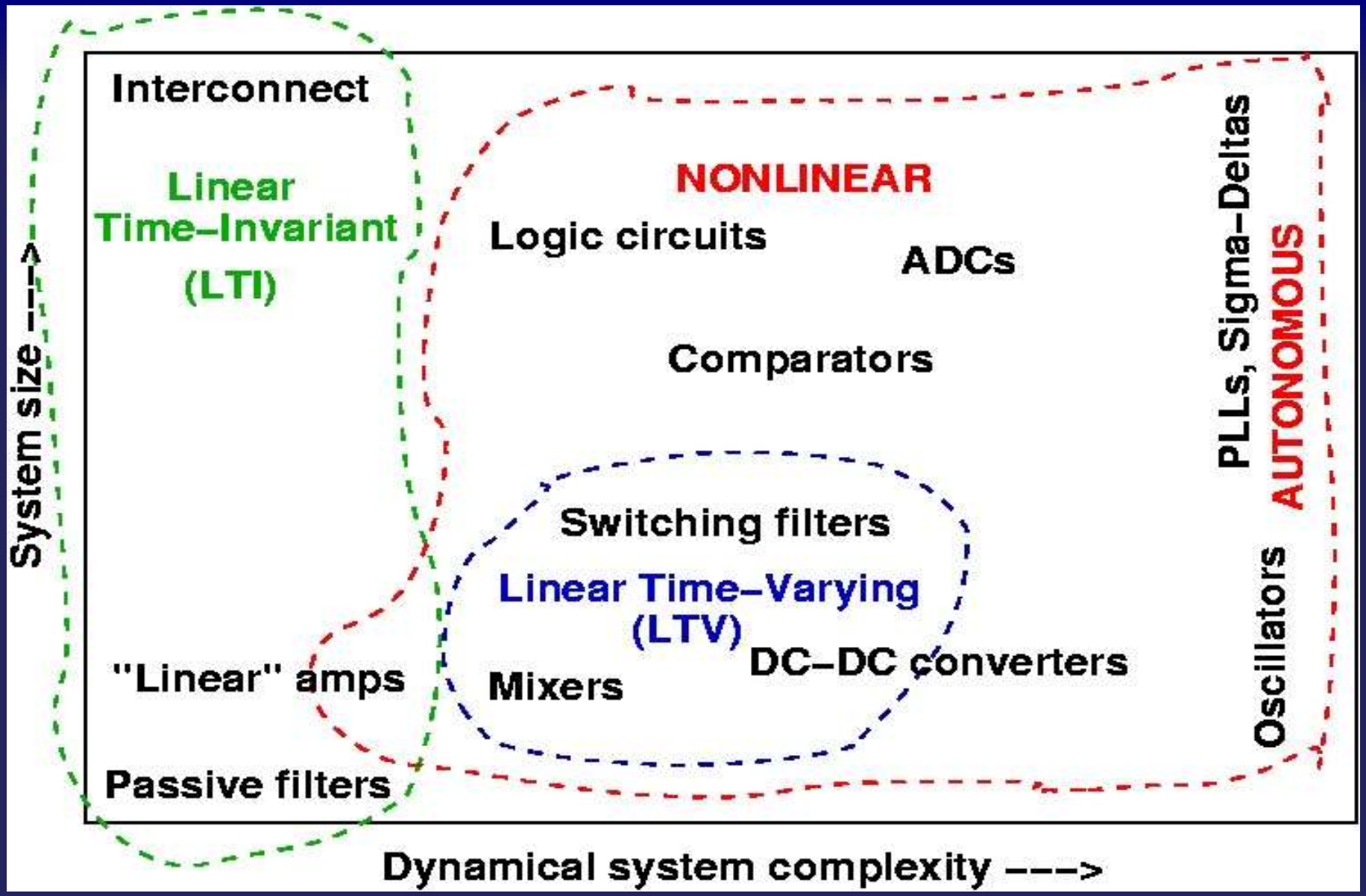
"The Gap"

"At this point, you may wonder why you should bother with behavioral libraries and calibration. Why not just submit the transistor-level design to some smart software and let it come up with a model? Unfortunately, despite some claims to the contrary, practical model synthesis is still a long way off. Attempts at this technology rely on pre-existing templates, which are unlikely to exist for leading-edge or proprietary designs. There's no pushbutton approach to analog modeling, and from all indications, this will remain the case for some time to come." - EE Times, 2001

- Perhaps not quite so bleak!
- Automated macromodel generation **is** difficult

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Why Difficult?

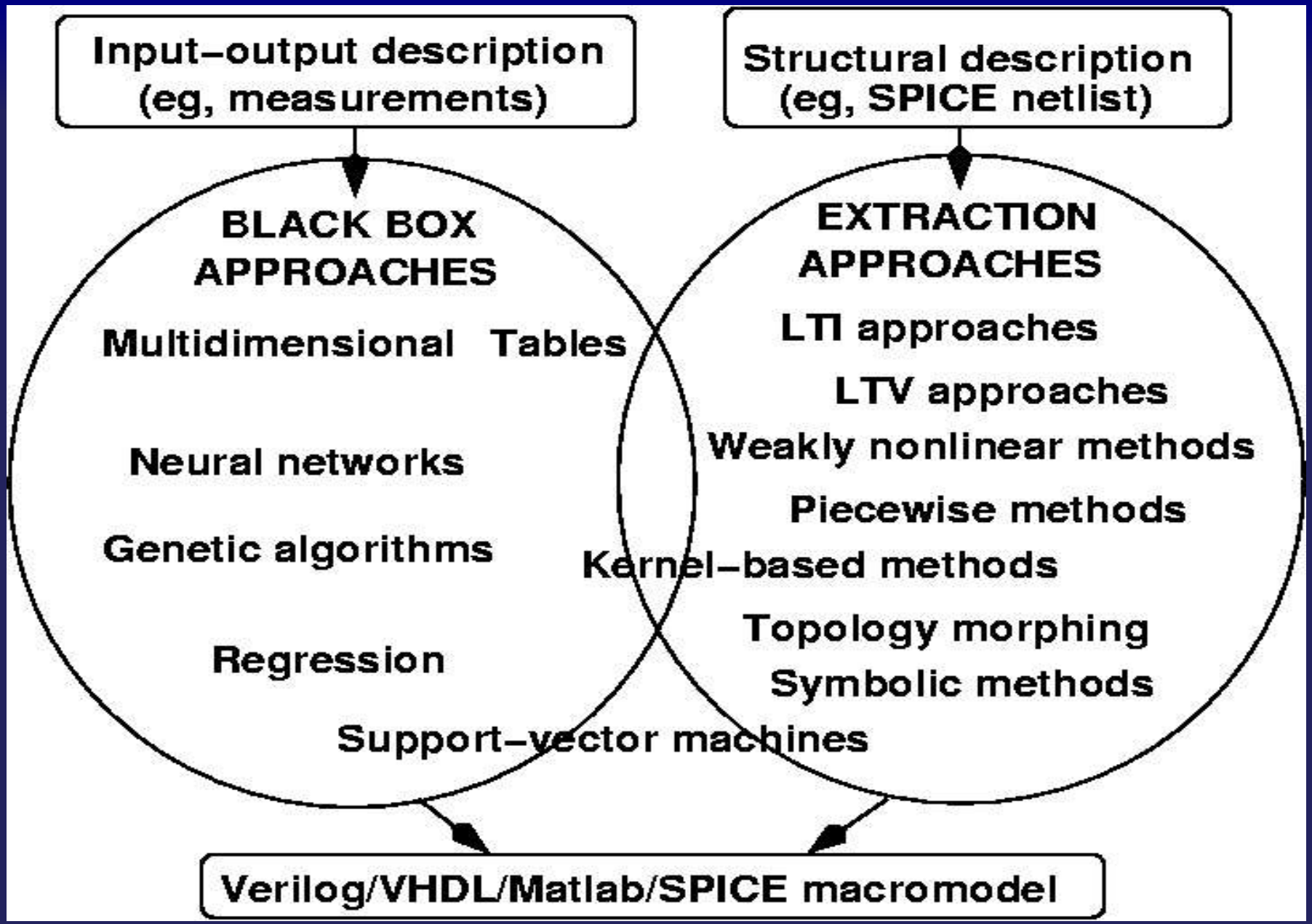


Approaches to Macromodelling

- **Black box** problems
  - samples of input-output pairs
  - from measurement and/or simulation
  - paucity of information
- **Extraction (bottom-up reduction)** problems
  - detailed circuit/simulation info available
  - eg, SPICE netlist: differential equations
  - surfeit of information
  - potential for better macromodels

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Automated MM Approaches



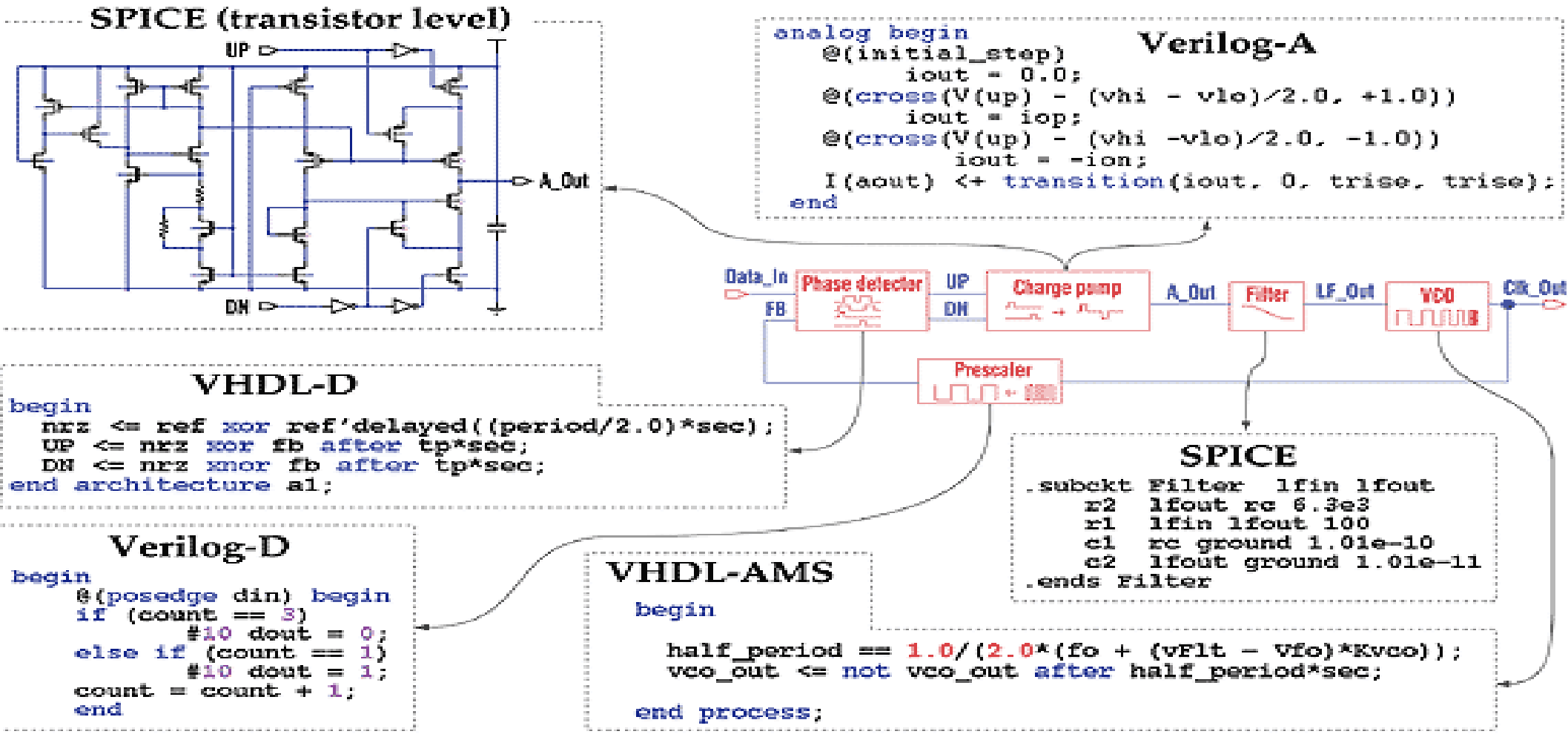
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Macromodelling Languages

- Output of macromodelling process
- AHDL languages – eg, Verilog-A, VHDL-AMS
  - the EDA choice
- Matlab/Simulink
  - widely used by designers
- SPICE(?)
- Fundamentally: (integro-)differential-algebraic equations

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Macromodelling Languages



EE Times

Multiplicity of languages, should inter-operate

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

"Algorithmic" Macromodelling Approaches

- **Mathematical algorithms** based on theory
- **Provably preserve some useful property**
  - eg, moments of transfer function
- **AWE:** first prominent method (LTI)
- Variety of nonlinear, LTV methods
- **Generally applicable** (eg, multi-physics)
  - not specific to a particular type of circuit/topology
  - eg: same oscillator MM technique works as well for a Colpitts oscillator as for a DFB laser or a grandfather clock

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Types of Algorithmic Macromodelling

- **Linear Time Invariant (LTI)**
  - application: interconnect (delay, crosstalk)
  - AWE, PVL, PRIMA, TBR
- **Linear Time Varying (LTV)**
  - mixers, sampling/switching circuits, (oscillators)
- **Weakly nonlinear (Volterra)**
  - companding ckts, amplifier/mixer gain compression
- **Strongly nonlinear (stable)**
  - everything else: comparators, switching, slewing, ...
- **Autonomous**
  - oscillators, PLLs, etc.: marginally stable

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Macromodelling Linear Time-Varying Systems

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

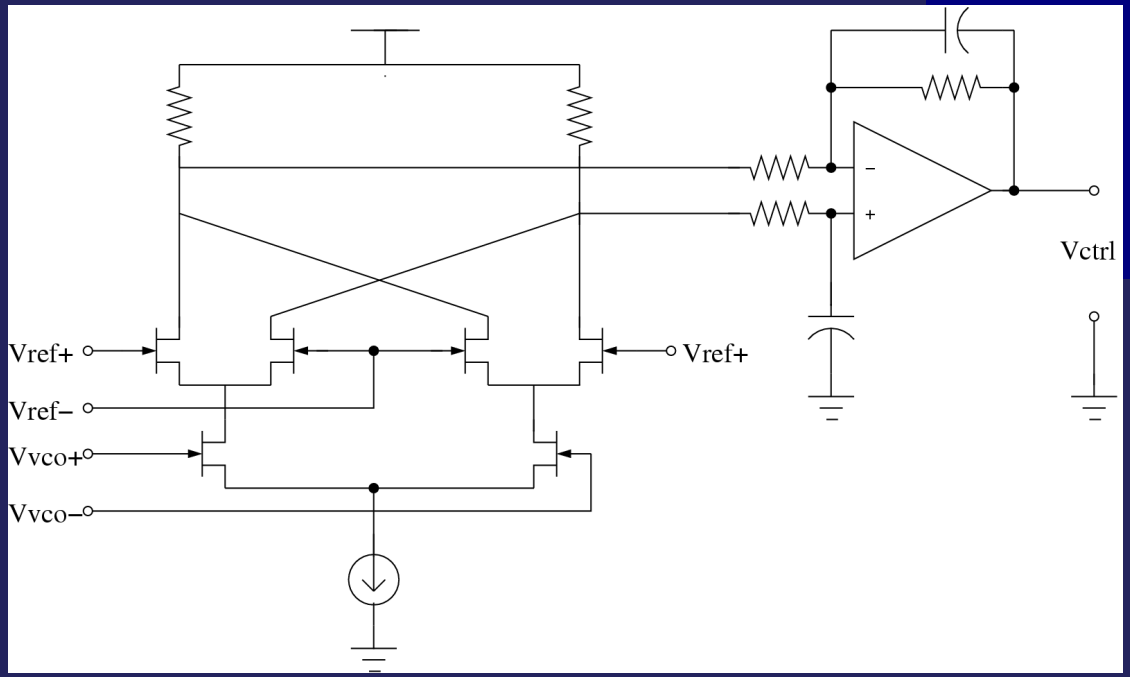
Linear Time-Varying Macromodelling

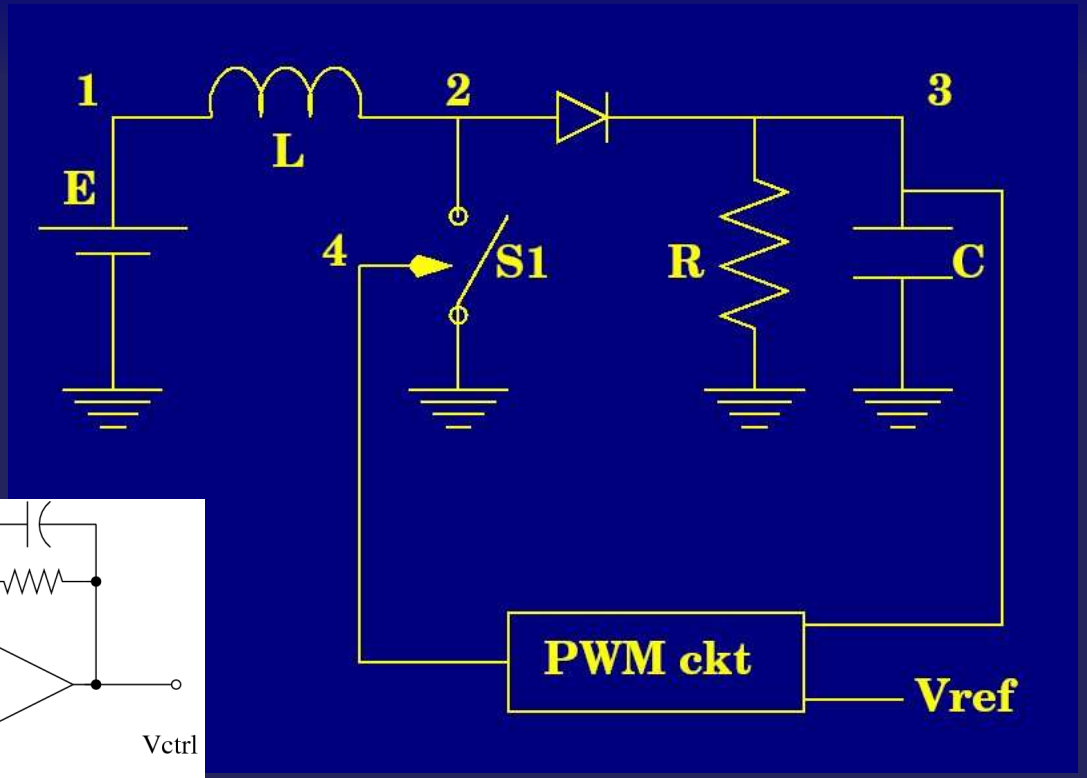
- Useful abstraction for some nonlinear systems
  - mixers, switching filters, samplers, DC/DC converters
  - leverage LTI methods
  - frequency translation, sampling captured
  - signal-path nonlinearities **not** captured
- Input-output relationship linear
  - but **not** time-invariant

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Example candidates for LTV Macromodelling

Gilbert Cell Mixer





DC-DC power converter

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Basic Principles of LTV Macromodelling

- LTV systems have time-varying (but linear) transfer functions
  - LTI transfer function H(s) --> H(t,s)
    - t captures system time variation
    - s captures input/output time variations
  - computationally useful form of H(t,s):
    - linear matrices C(t) and G(t): from transient simulation/steady state calculations

$$H(t_1, s) = d^T \left(\frac{\partial}{\partial t_1} C(t_1) + sC(t_1) + G(t_1) \right)^{-1} [b]$$

[Roychowdhury, Phillips ICCAD 98] [Roychowdhury TCAS-2 99]

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Reducing LTV Systems

- Matching moments of H(t,s)
  - Moments are w.r.t s (moments are functions of t)
  - One way to leverage LTI moment-matching MOR techniques
    - convert LTV system to artificial LTI system
    - increases number of "outputs" -- one output for each t
- Eg, for periodic LTV systems: Fourier basis
  - expand H(t,s) in Fourier series in t
  - SIMO/MIMO system, each F. harmonic is an "output"

$$H_{FD}(s) = D^T \left(sC_{FD} + J_{FD}\right)^{-1} B_{FD}$$

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Reducing LTV Systems (contd)

- Now reduce the (bigger, artificial) LTI system
  - with, eg, Krylov-subspace methods
  - get a smaller reduced system of size q
  - convert back to LTV form

$$-T_q \frac{\partial x_q}{\partial t} + x_q = r_q u(t), \quad y(t) \approx l_q^T(t) x_q(t)$$

Linear Time Invariant

Memoryless Multiplication

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

LTV Reduction: Procedure

- Input: circuit description (SPICE)
- first: get nonlinear steady state (via HB/shooting/etc)
- find time-varying matrices (often automatic) C(t), G(t)
- create large TI-matrix system (implicitly) by stacking C(t), G(t) for various “samples” of t
- run LTI model reduction methods to get a reduced model – choice of methods
- map back to LTV form
- size-vs accuracy tradeoff (as for LTI)
- LTI + memoryless-TV form: implementation convenience

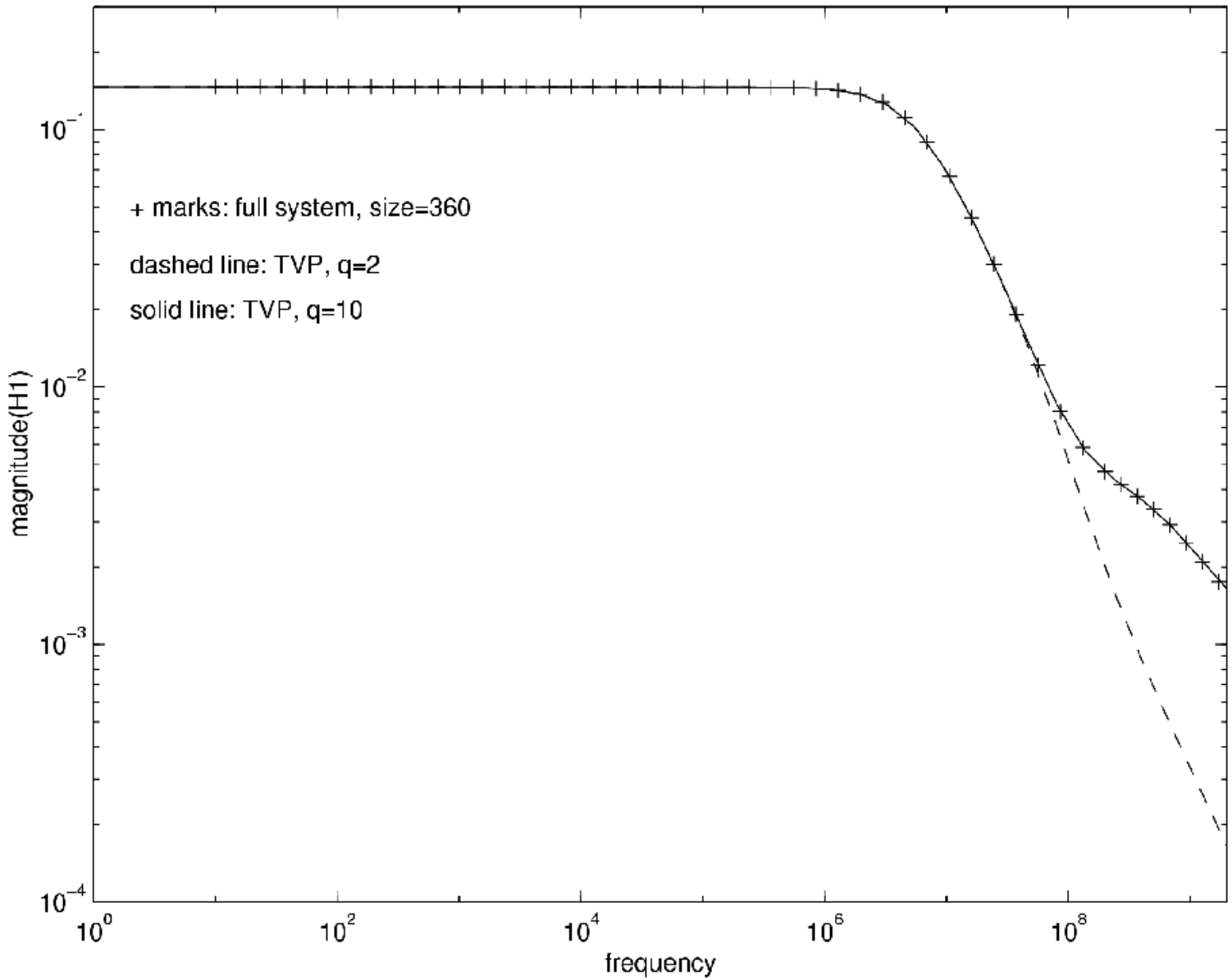
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

LTV Macromodel of RF Mixer

- I-channel mixer and buffer block (Lucent 2013 RFIC)
  - 360 nodes
  - signal upto ~80kHz, LO at 178MHz
- **LTV macromodel accuracy (Time-Varying Pade)**
  - size 2: upconversion transfer function matches to 300kHz
  - size 10: upconversion transfer function matches to **400MHz**
- **Speedup more than 500x**

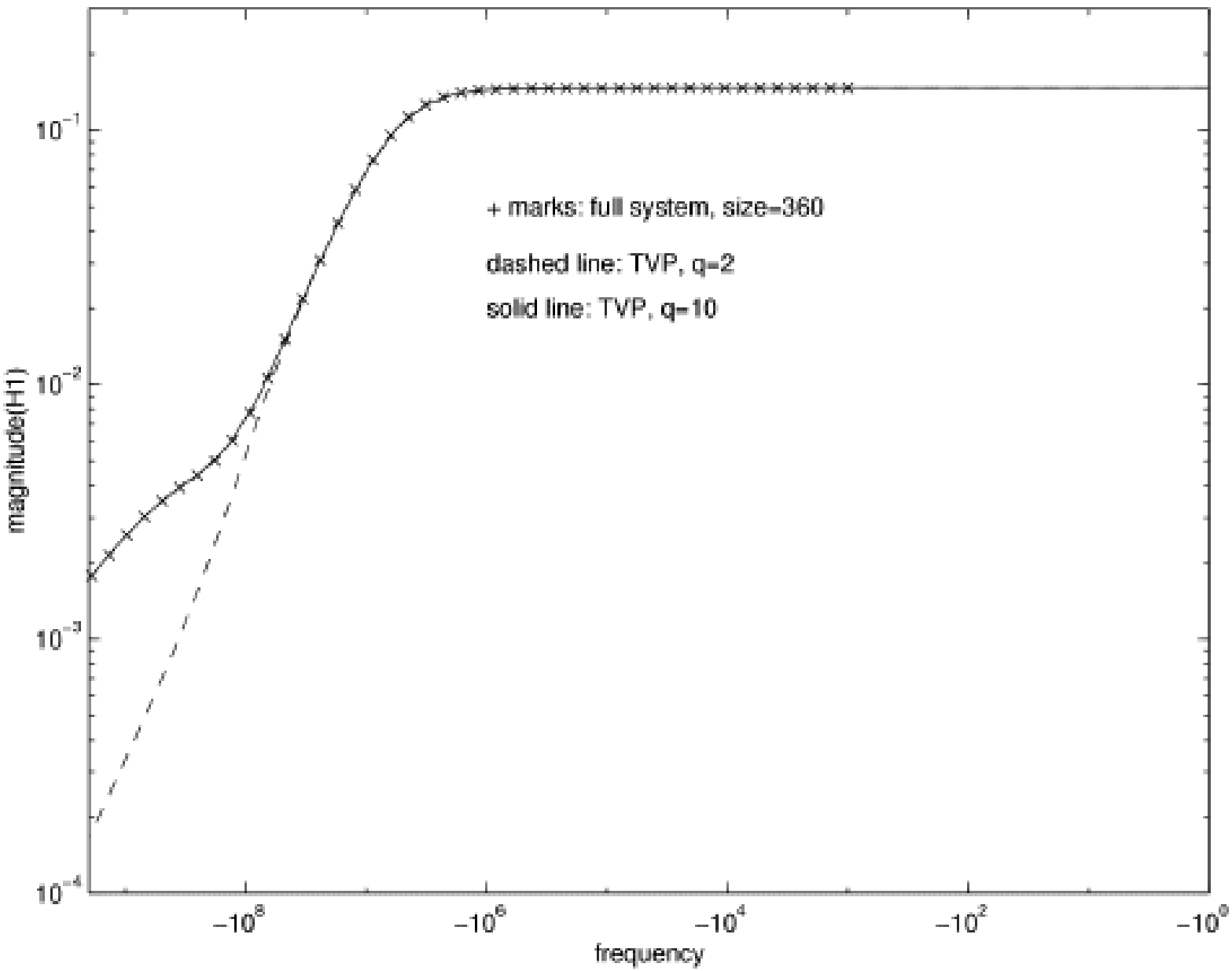
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

W2013 Upconversion Transfer Function



.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

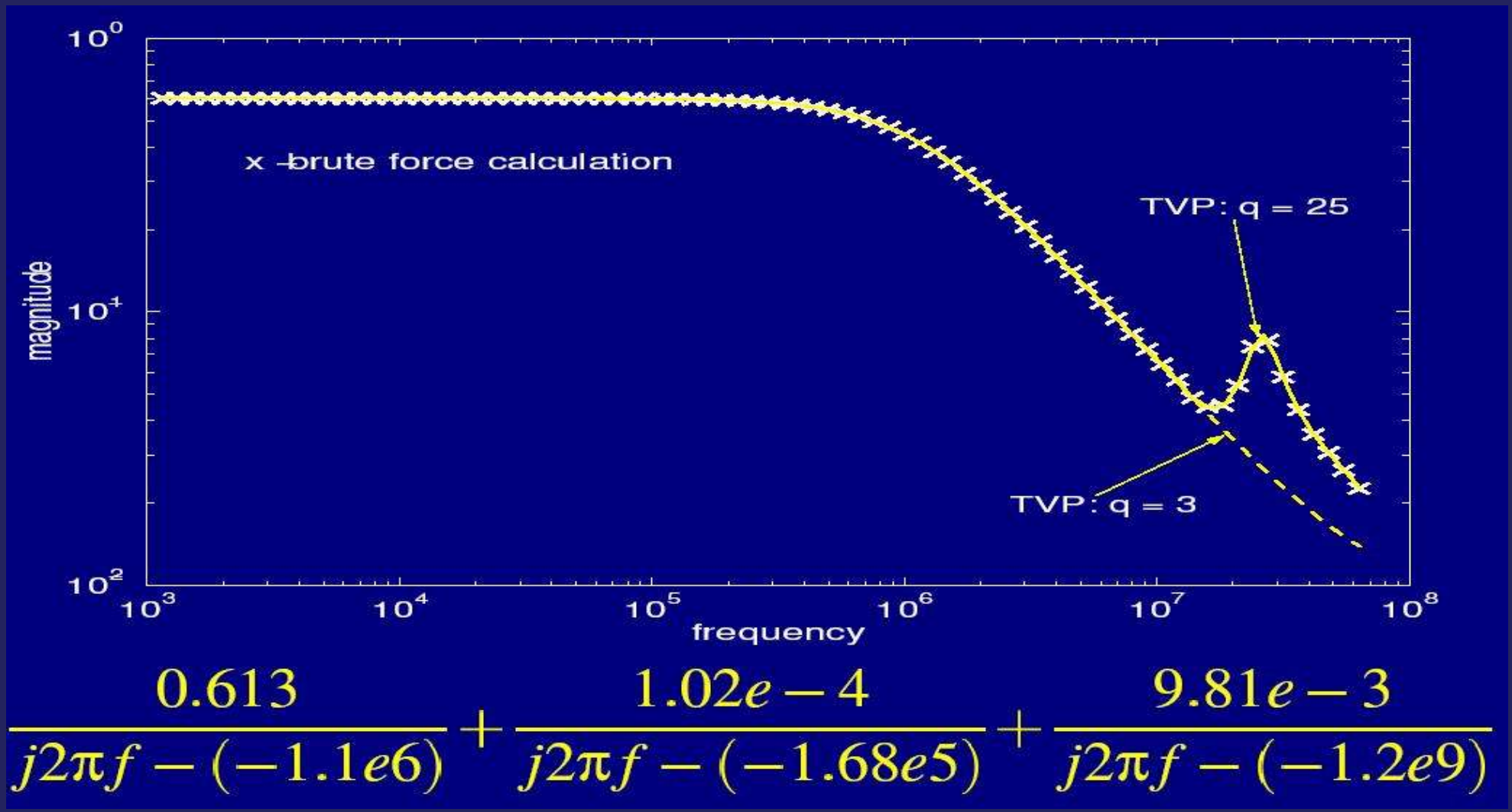
W2013 Downconversion Transfer Function



.DAC 2004 Macromodeling Tutorial. Copyright © J Roychowdhury

LTV Macromodel of Switched Capacitor Integrator Block

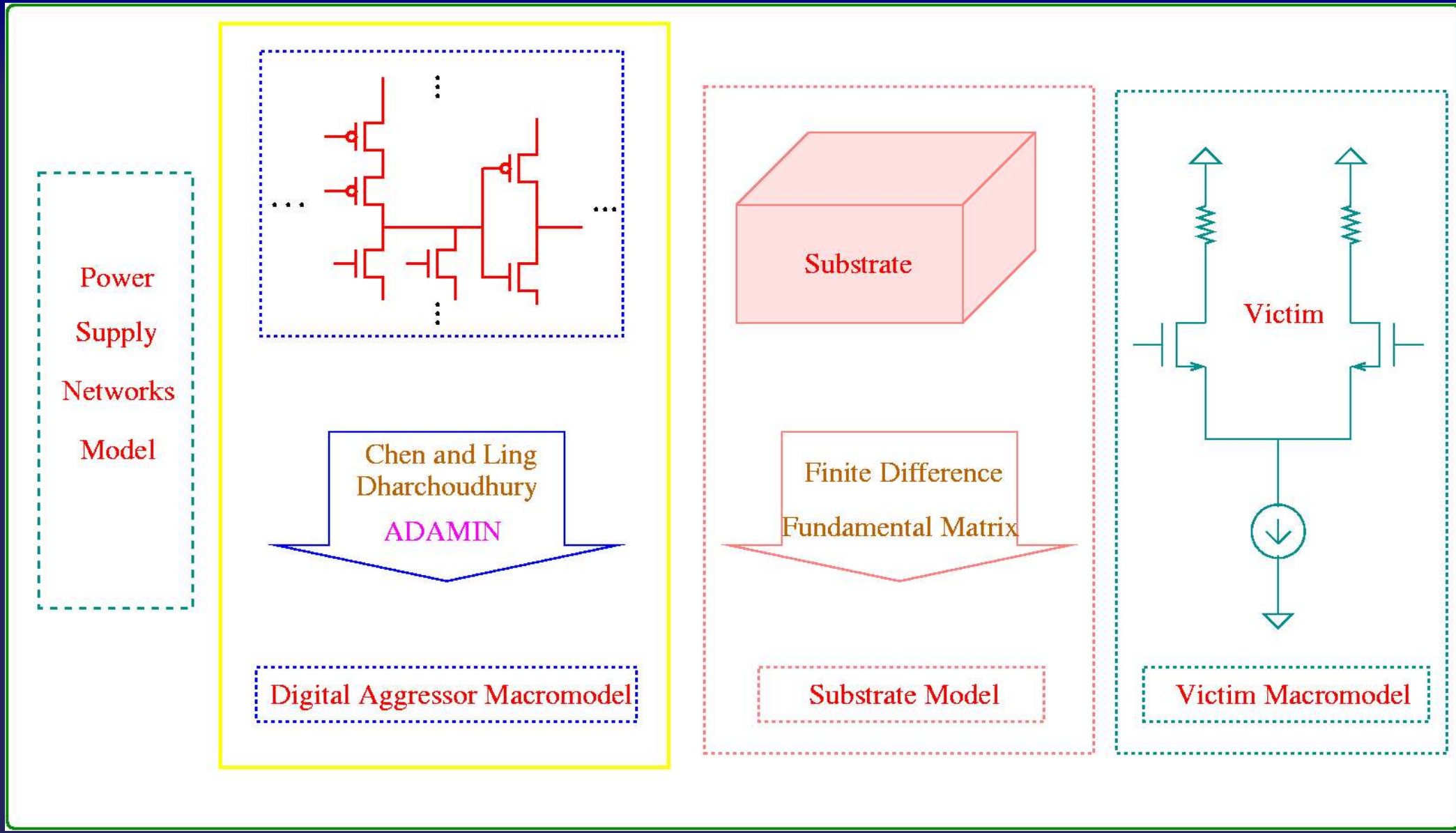
350 MOSFETS; clock=12.8MHz, signal=10KHz



$$\frac{0.613}{j2\pi f-(-1.1e6)}+\frac{1.02e-4}{j2\pi f-(-1.68e5)}+\frac{9.81e-3}{j2\pi f-(-1.2e9)}$$

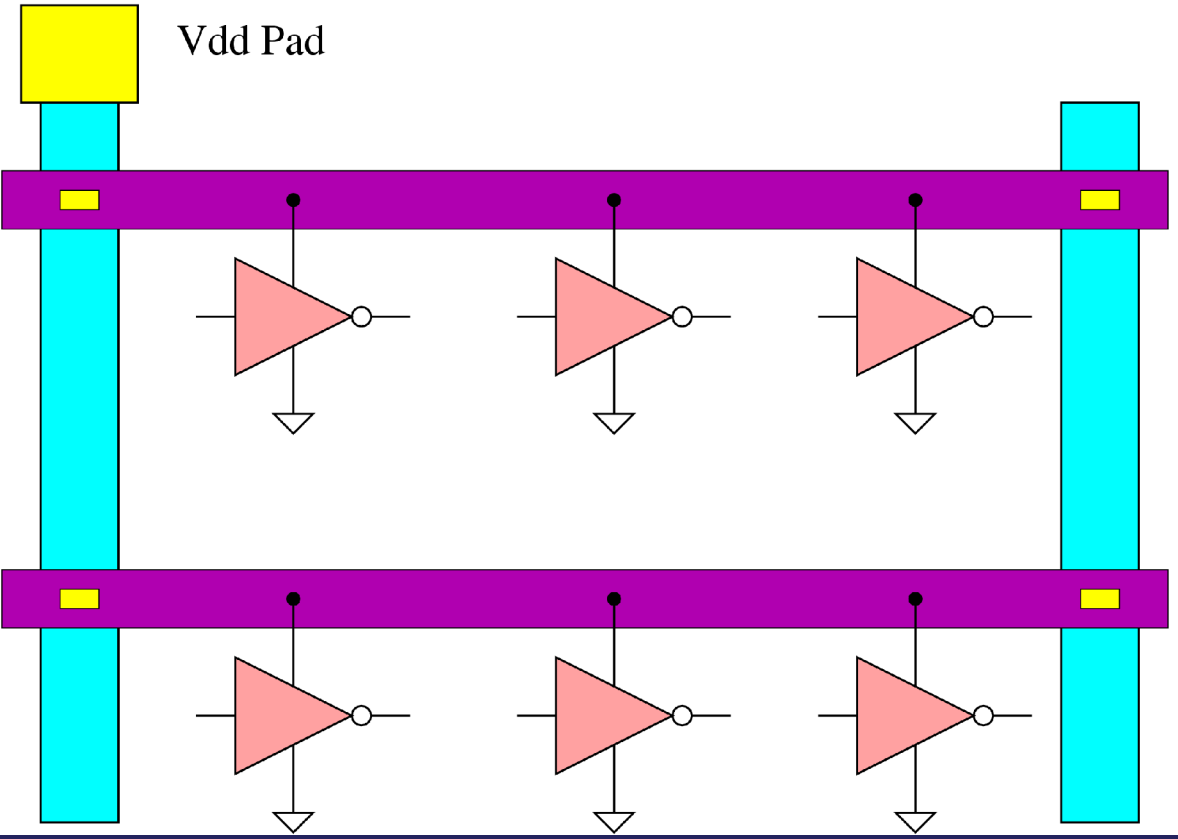
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

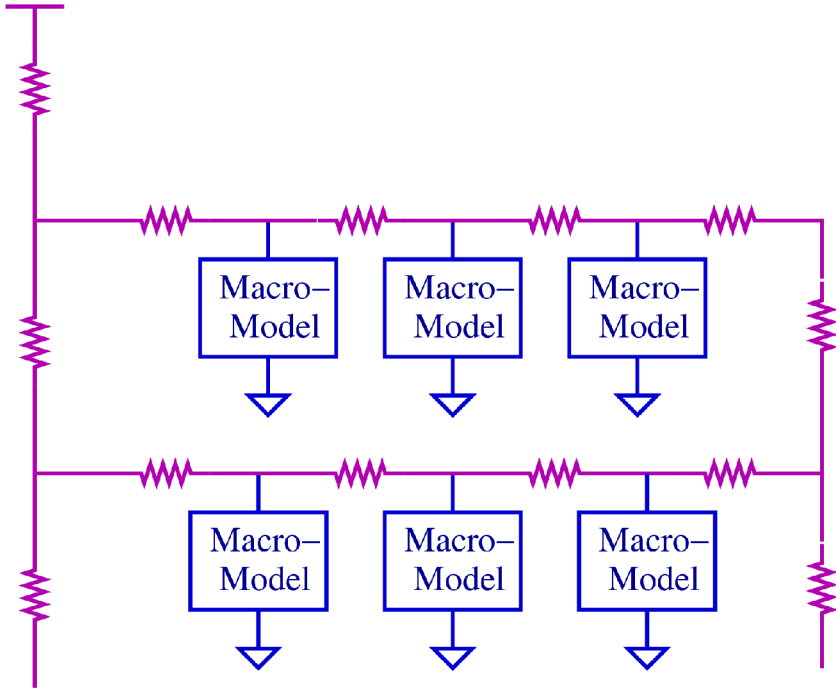
The Mixed Signal Interference Noise Problem



.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Macromodelling Digital Noise Injection



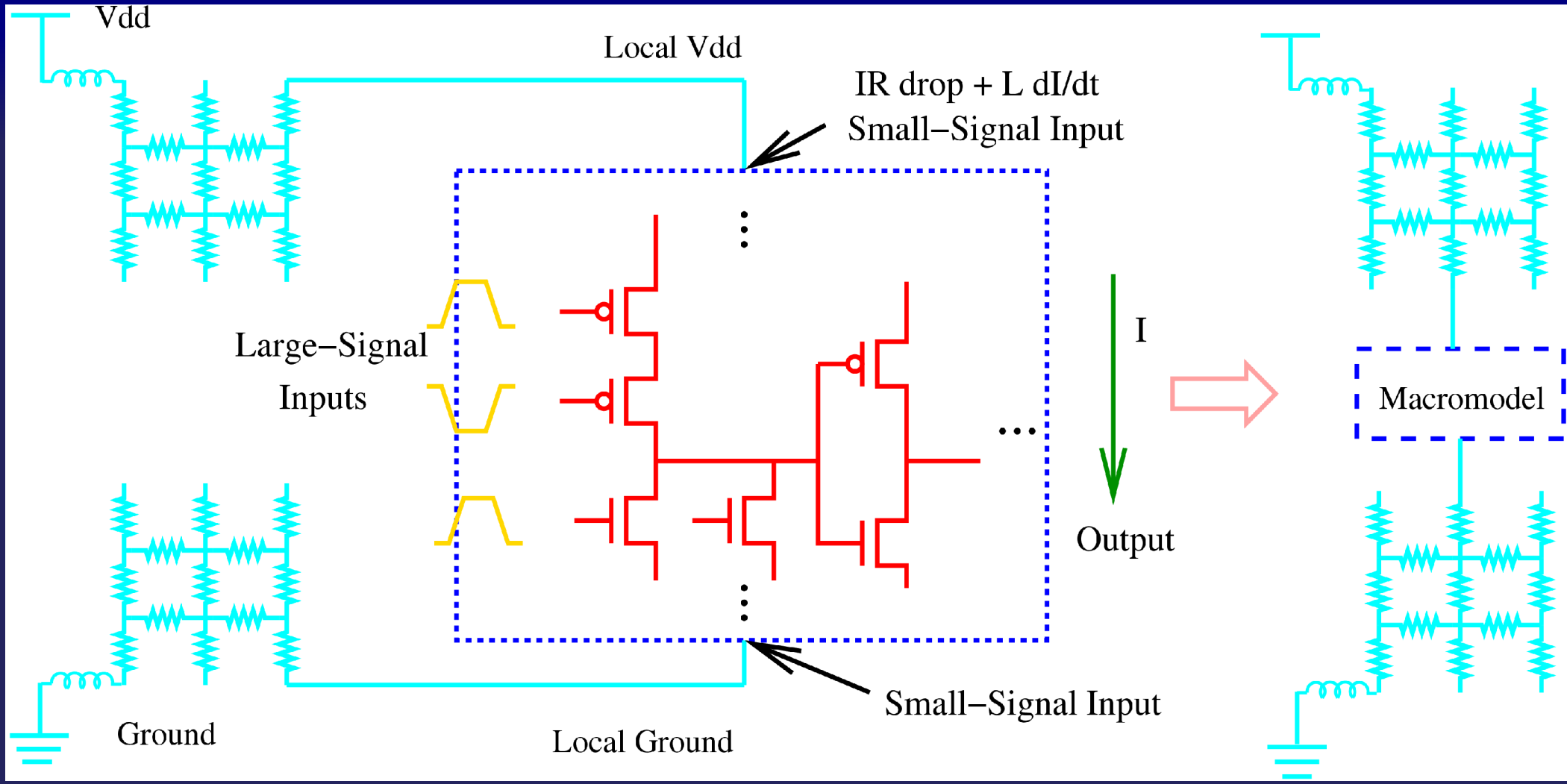


- Interested only in interference injection characteristics
  - supply and substrate injections: analog waveforms
  - digital signals: system time variation
  - LTV model captures switching behaviour well!

[Wang/Murgai/Roychowdhury DATE04]

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

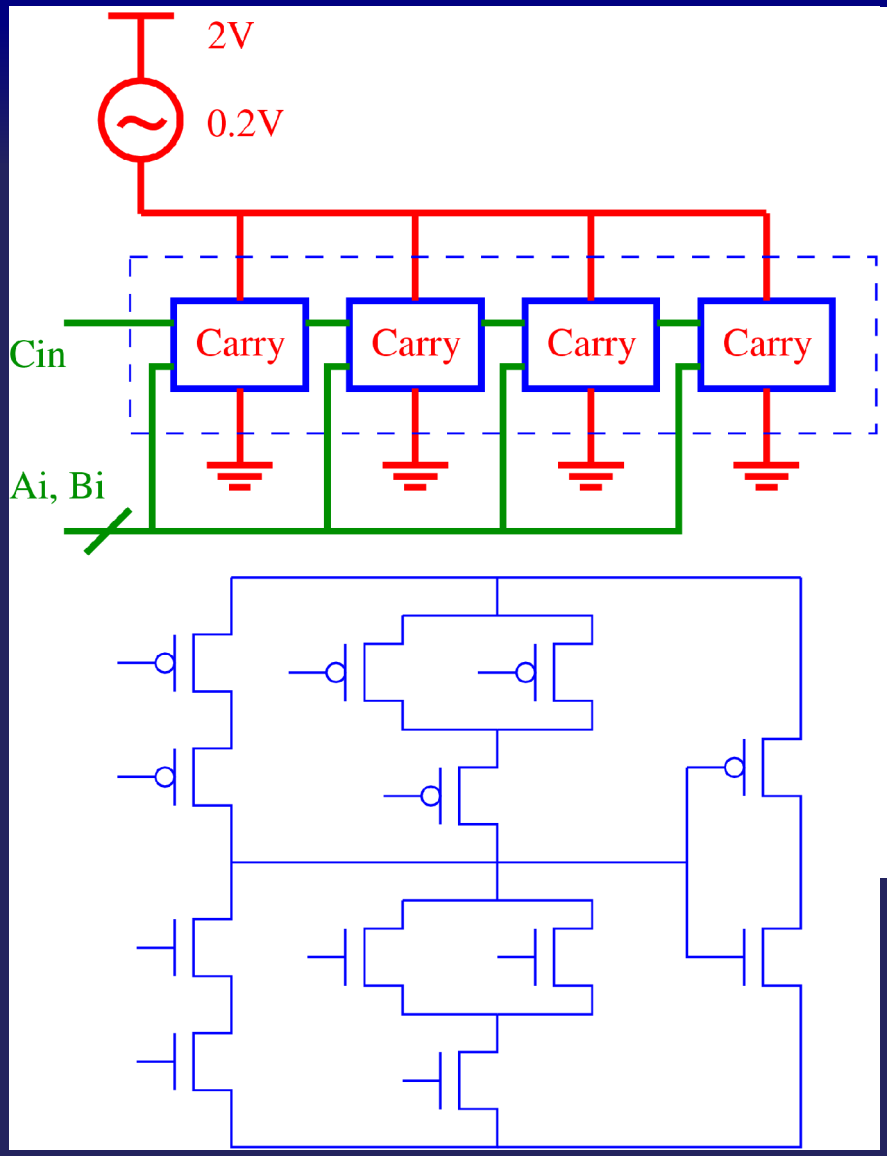
LTV Digital Aggressor Macromodels

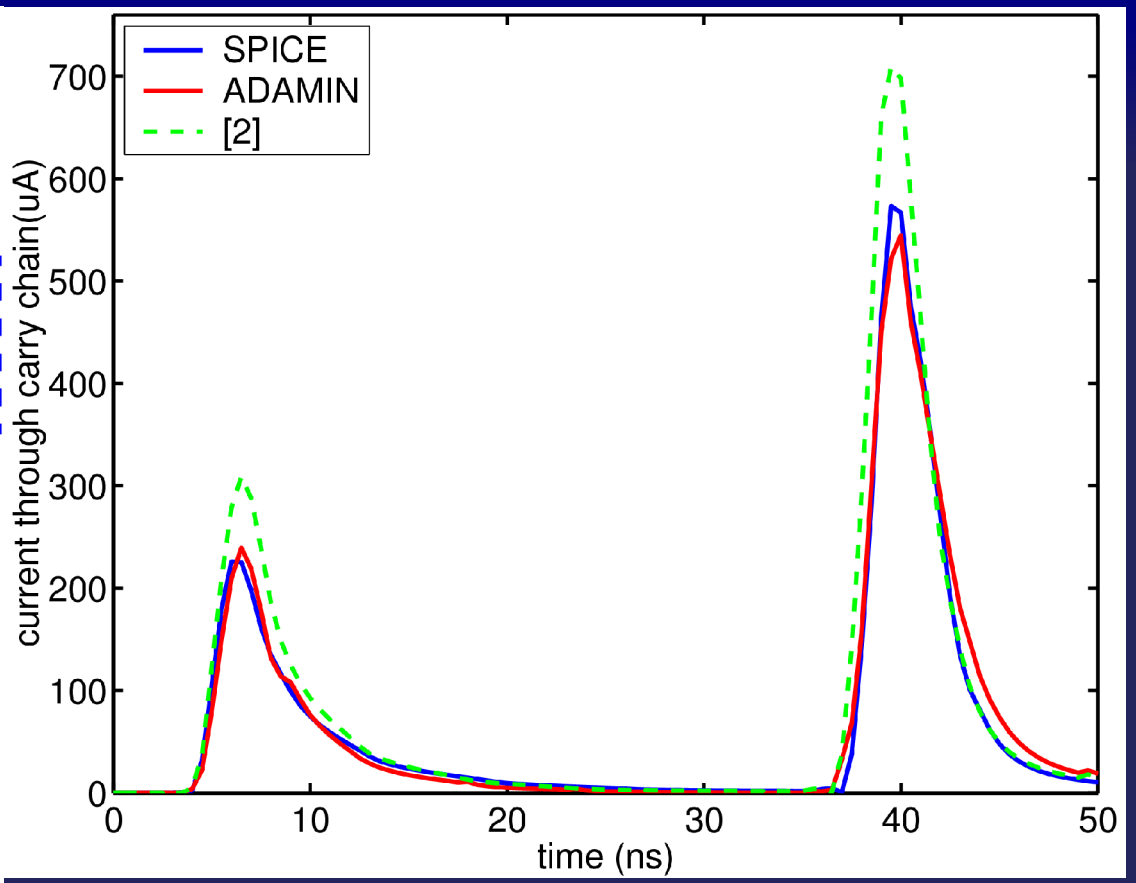


- Input: (eg) power supply voltage variation
- Output: resulting current variation
- MM: small time-varying system relating input to output

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Supply-noise Induced Currents: Carry Chain

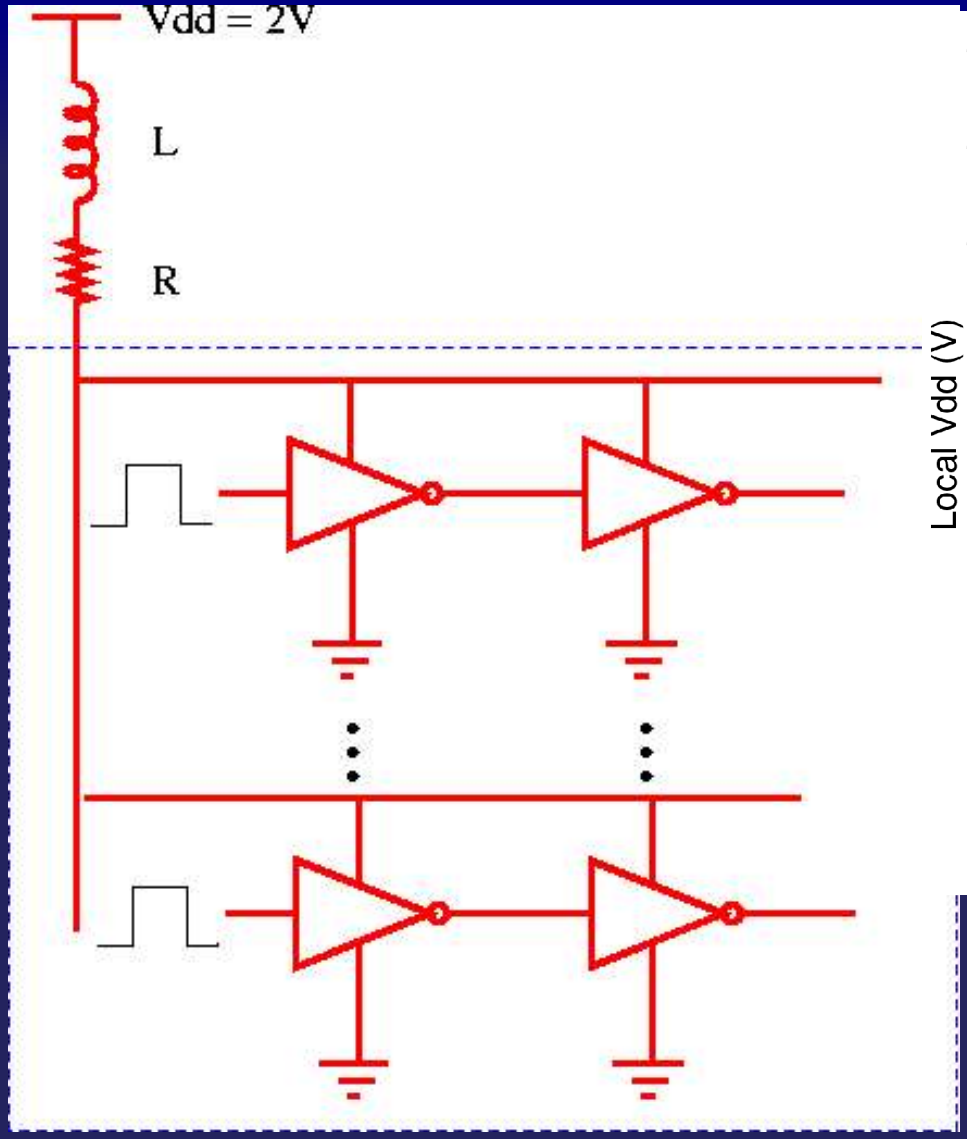


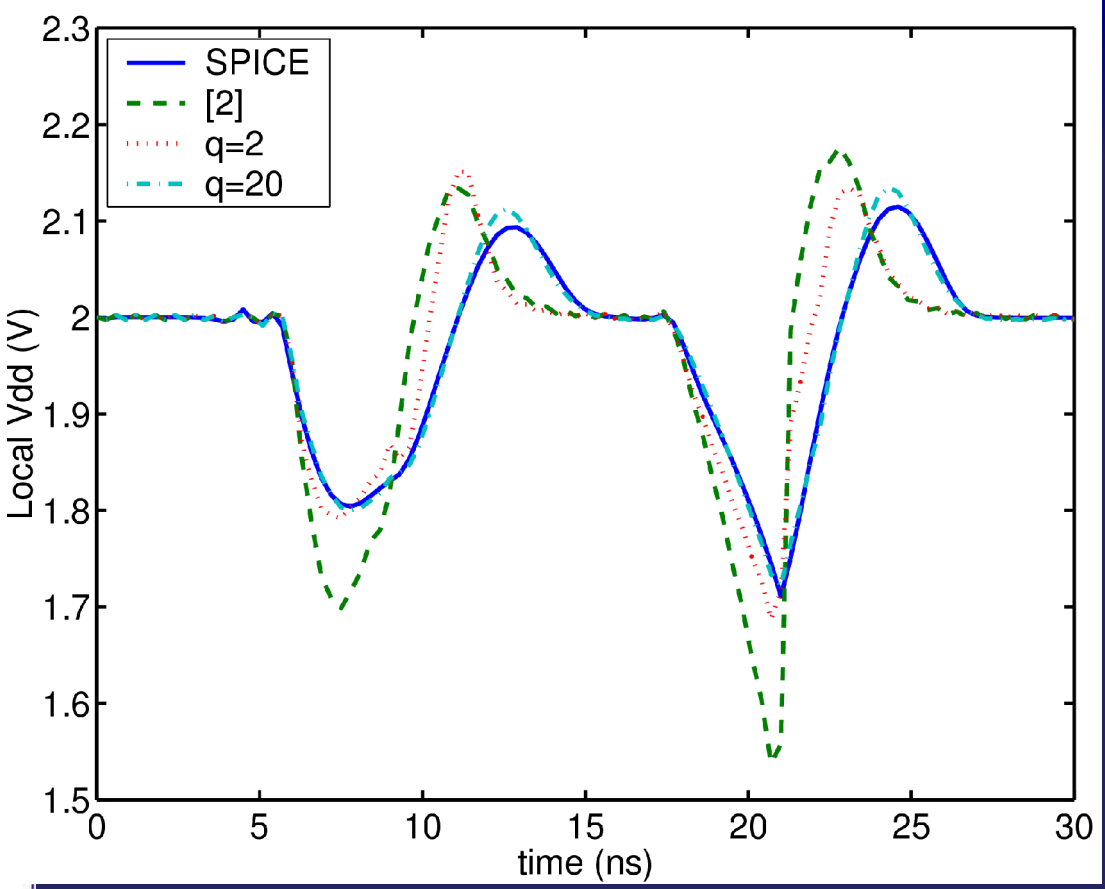


- Considerable accuracy improvement over current-source macromodel [Dharchoudhury 98]

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

“System” Simulation with Inductive Supply





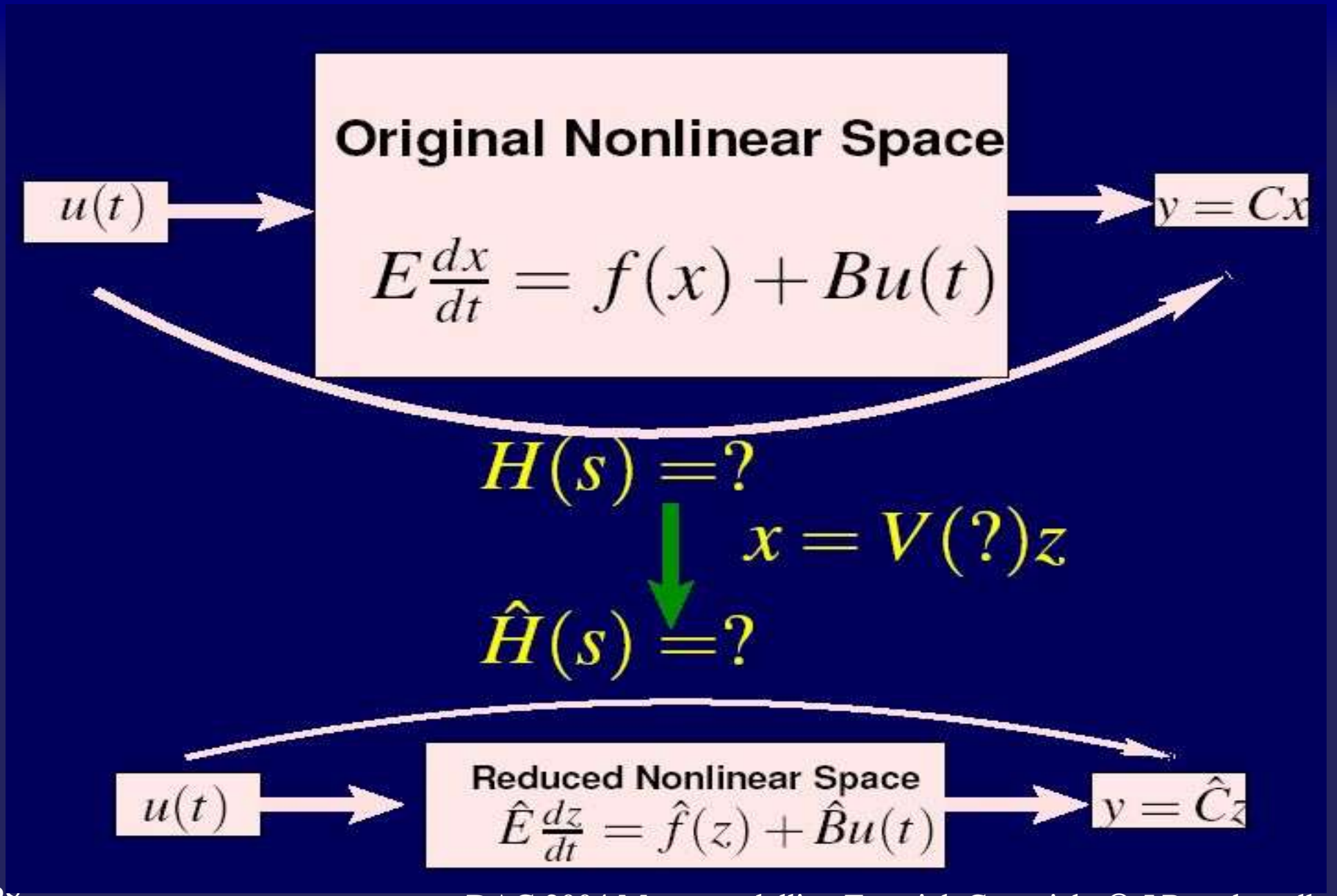
- size 2 MM: already better
- size 20: much more accurate
- 17000x speedup for 8000 gates

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Macromodelling Nonlinear Systems

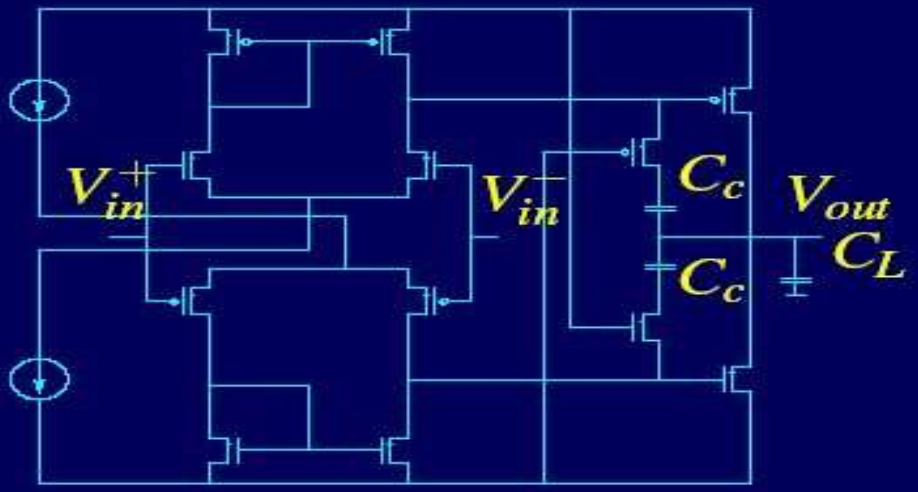
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

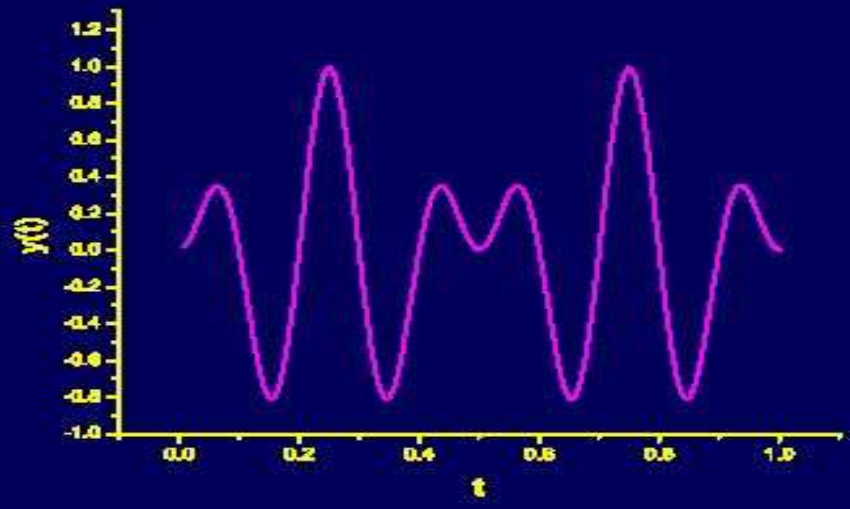
Nonlinear Macromodel Generation

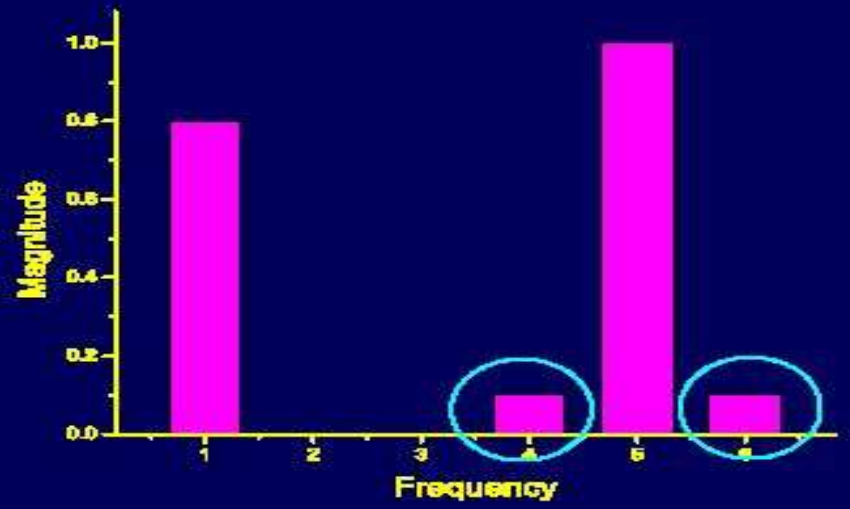


.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Weakly Nonlinear Systems





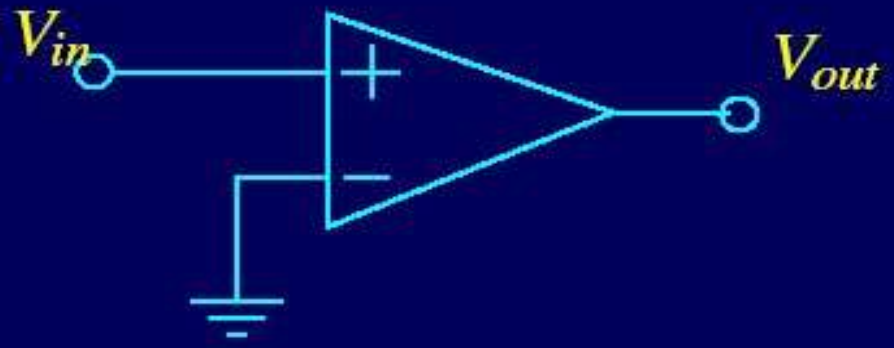


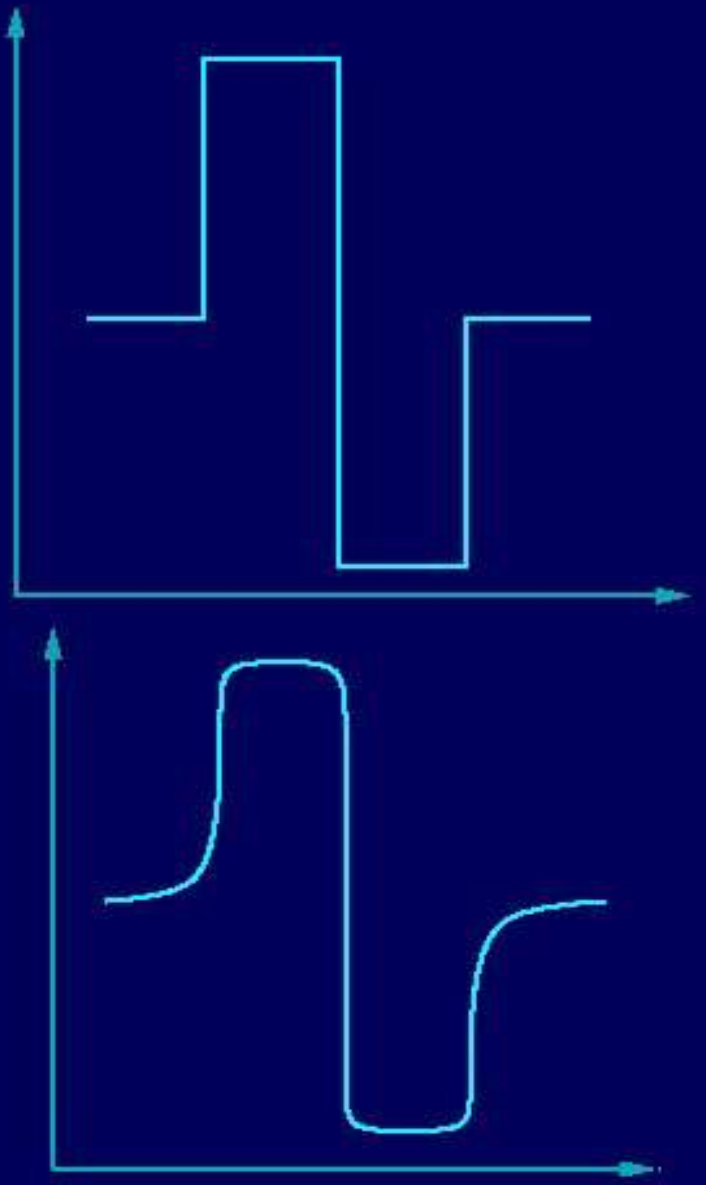
- Distortion, IM important!
- Must capture small distortion/IM

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Strong Nonlinearities

Comparators, switching mixers





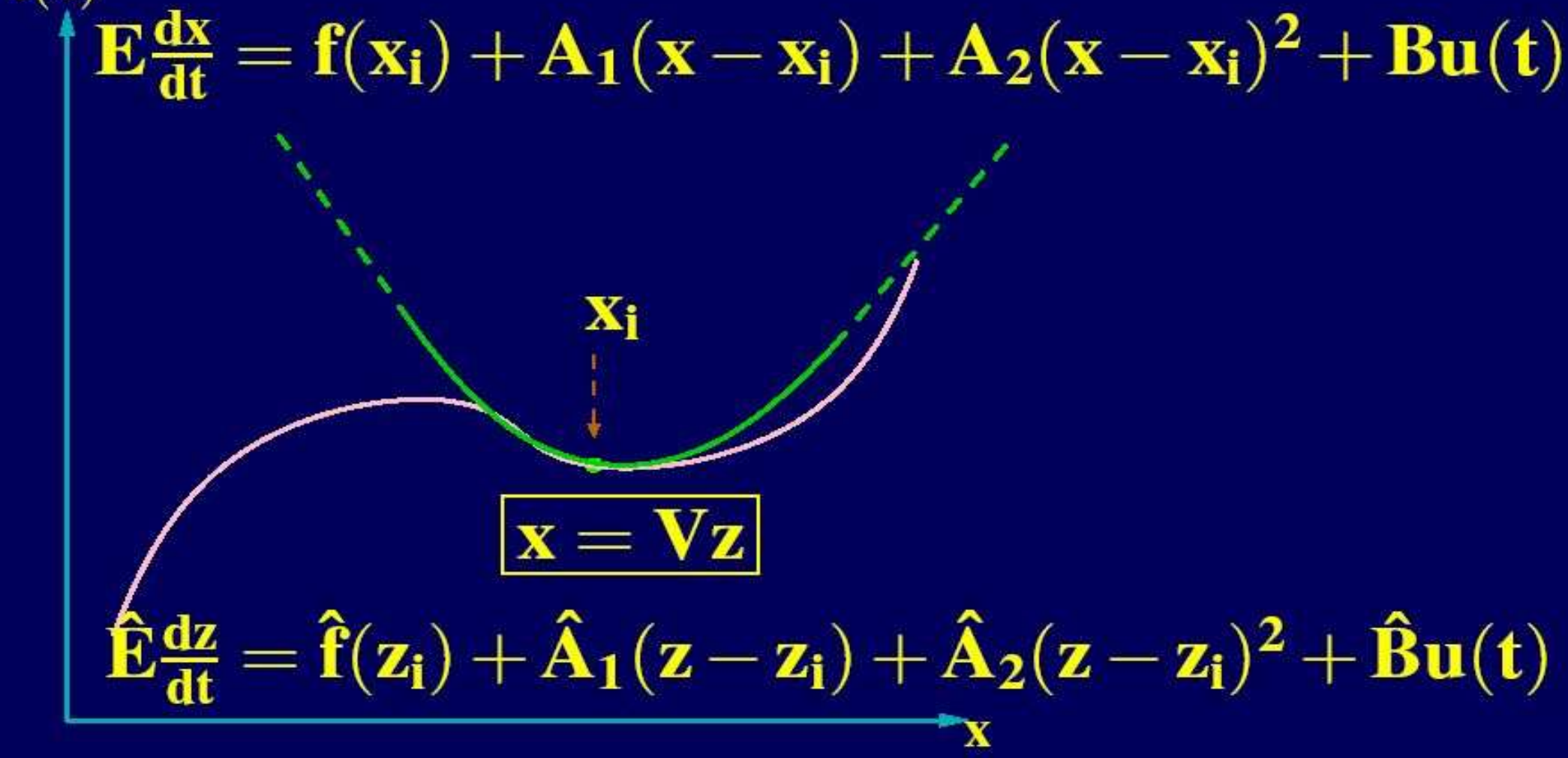
- Large signal clipping
- Must capture strong nonlinearities

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Polynomial (Volterra) Reduction

[Roychowdhury TCAS-2 99] [Phillips DAC 00] [Li/Pilleggi DAC 04]

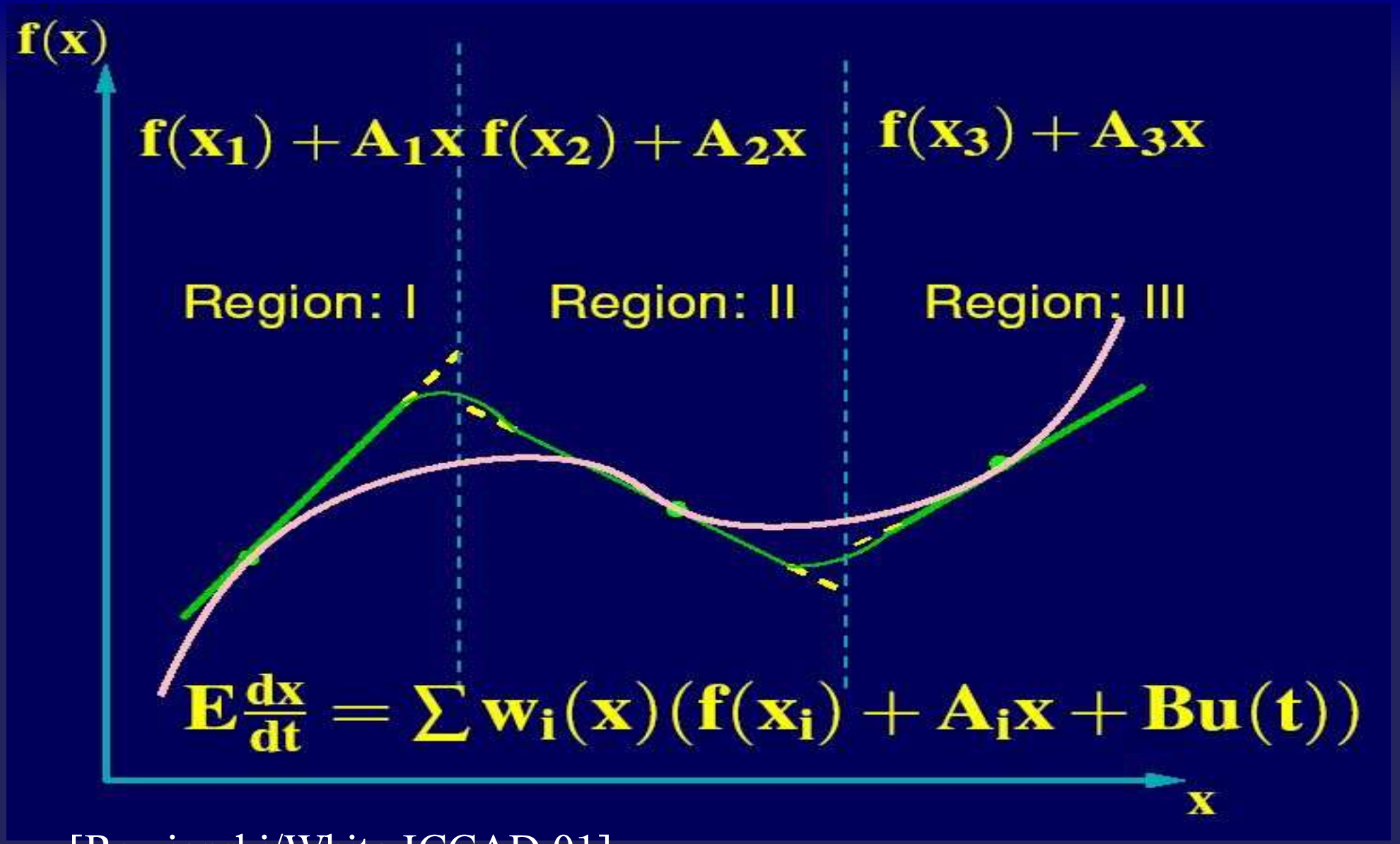
$$E\frac{dx}{dt} = f(x_i) + A_1(x - x_i) + A_2(x - x_i)^2 + Bu(t)$$



- **Good for small distortion, Poor for large swing**

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

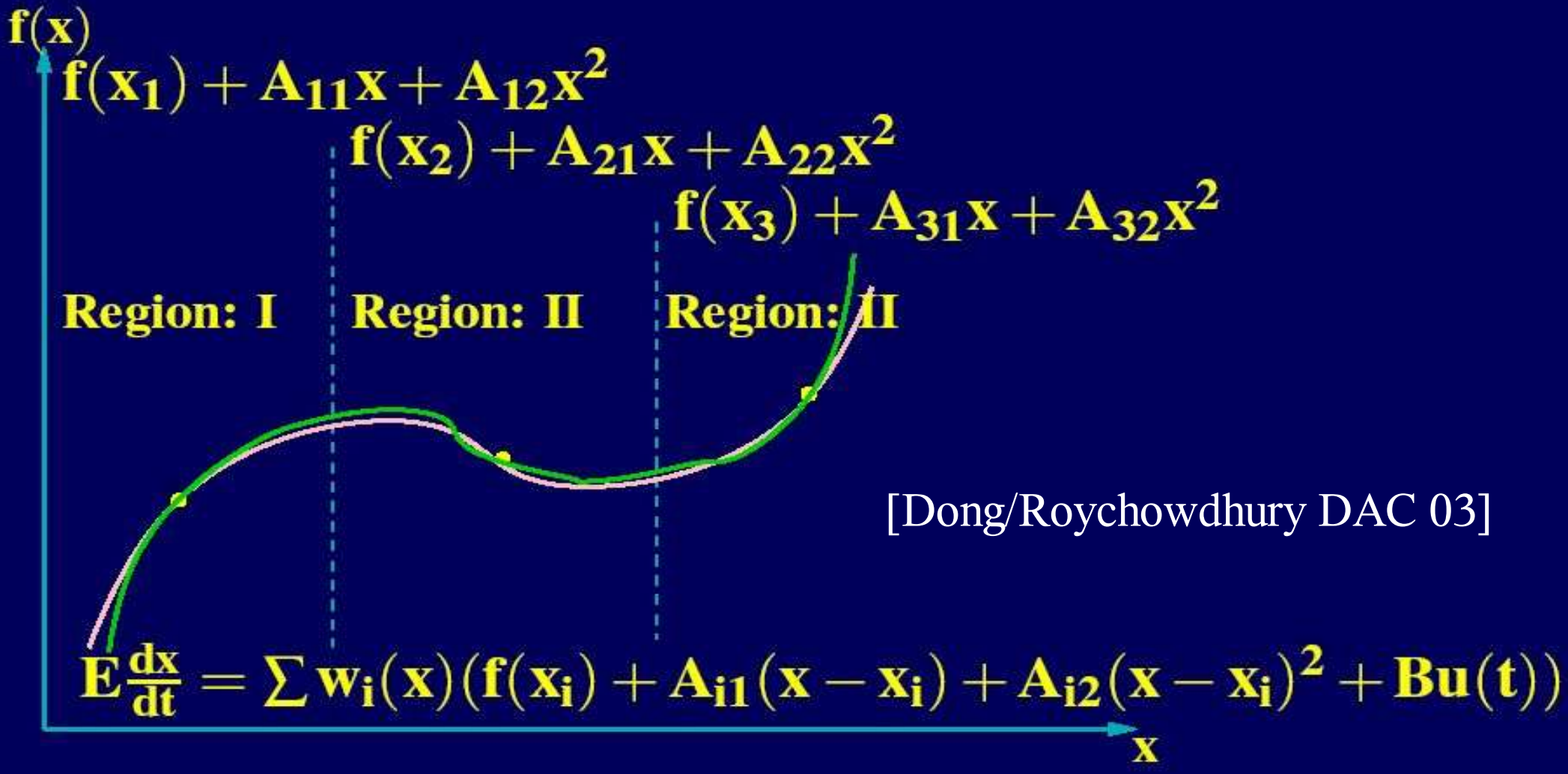
Trajectory Piecewise Linear MM



[Rewienski/White ICCAD 01]

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Piecewise Polynomial Macromodelling



[Dong/Roychowdhury DAC 03]

$$E\frac{dx}{dt} = \sum w_i(x)\left(f(x_i) + A_{i1}(x - x_i) + A_{i2}(x - x_i)^2 + Bu(t)\right)$$

- Good for small distortion, also good for large swing

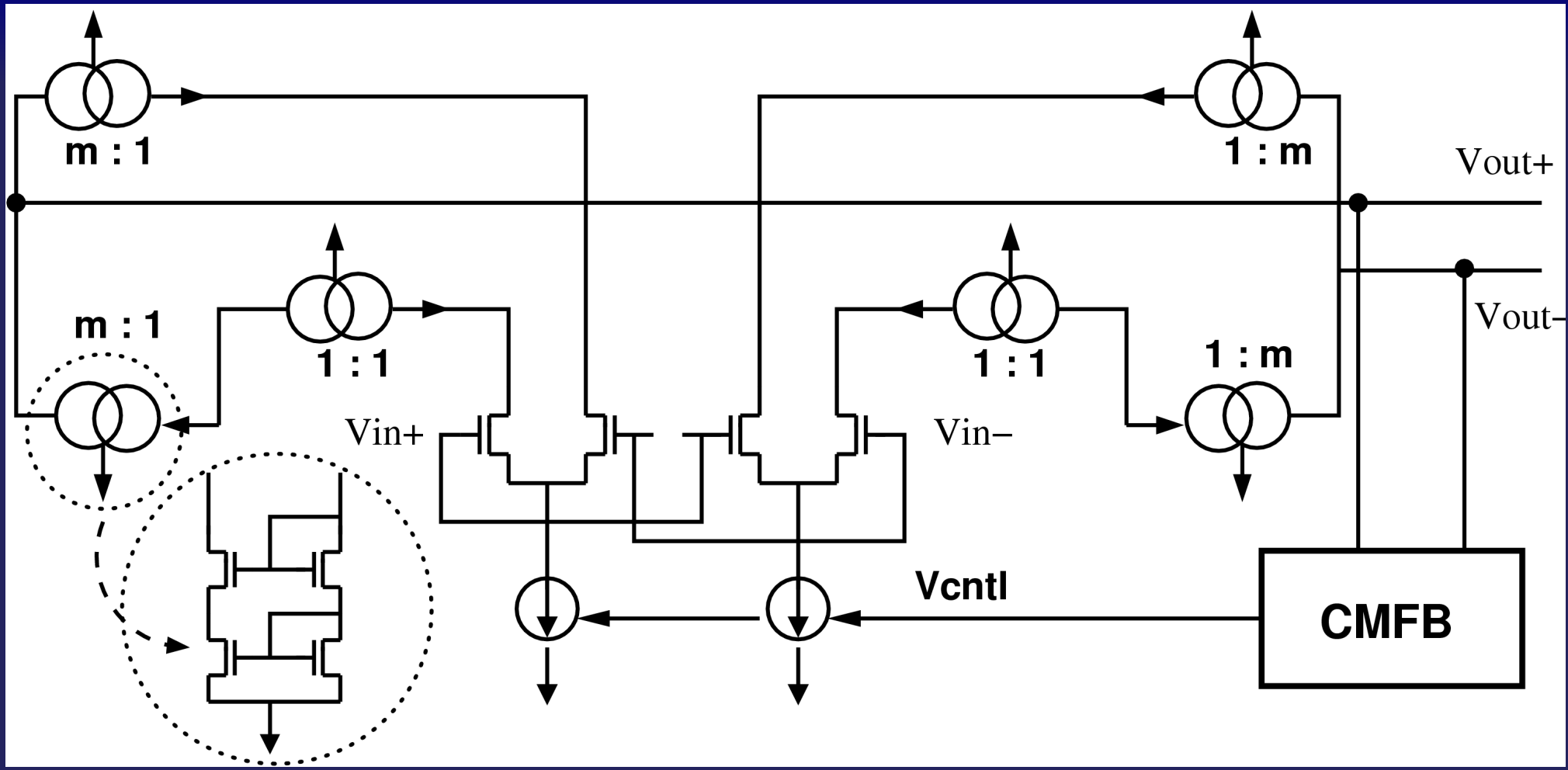
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Drop-in Replacement Macromodels

- Design process typically runs many simulations
  - DC (sweep), AC, small-signal distortion, transient
  - time- and frequency-domain analyses
- Would like one extracted macromodel to work for all analyses
  - ie, a drop-in replacement for the original
  - PWP-generated macromodels: good candidates

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

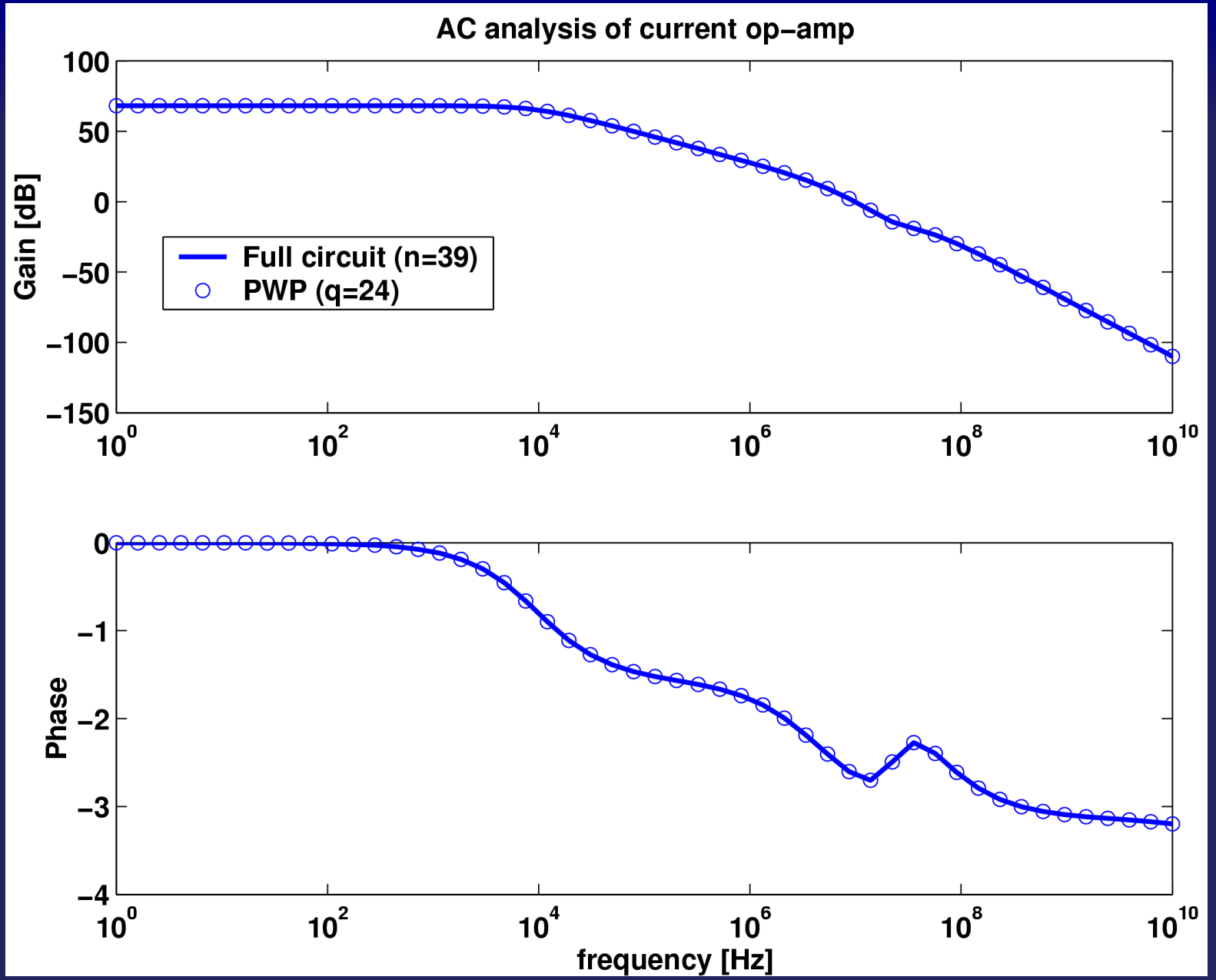
Current Mirror Op-Amp



Original: ~50 MOSFETs. Macromodel size: 19, 27 regions

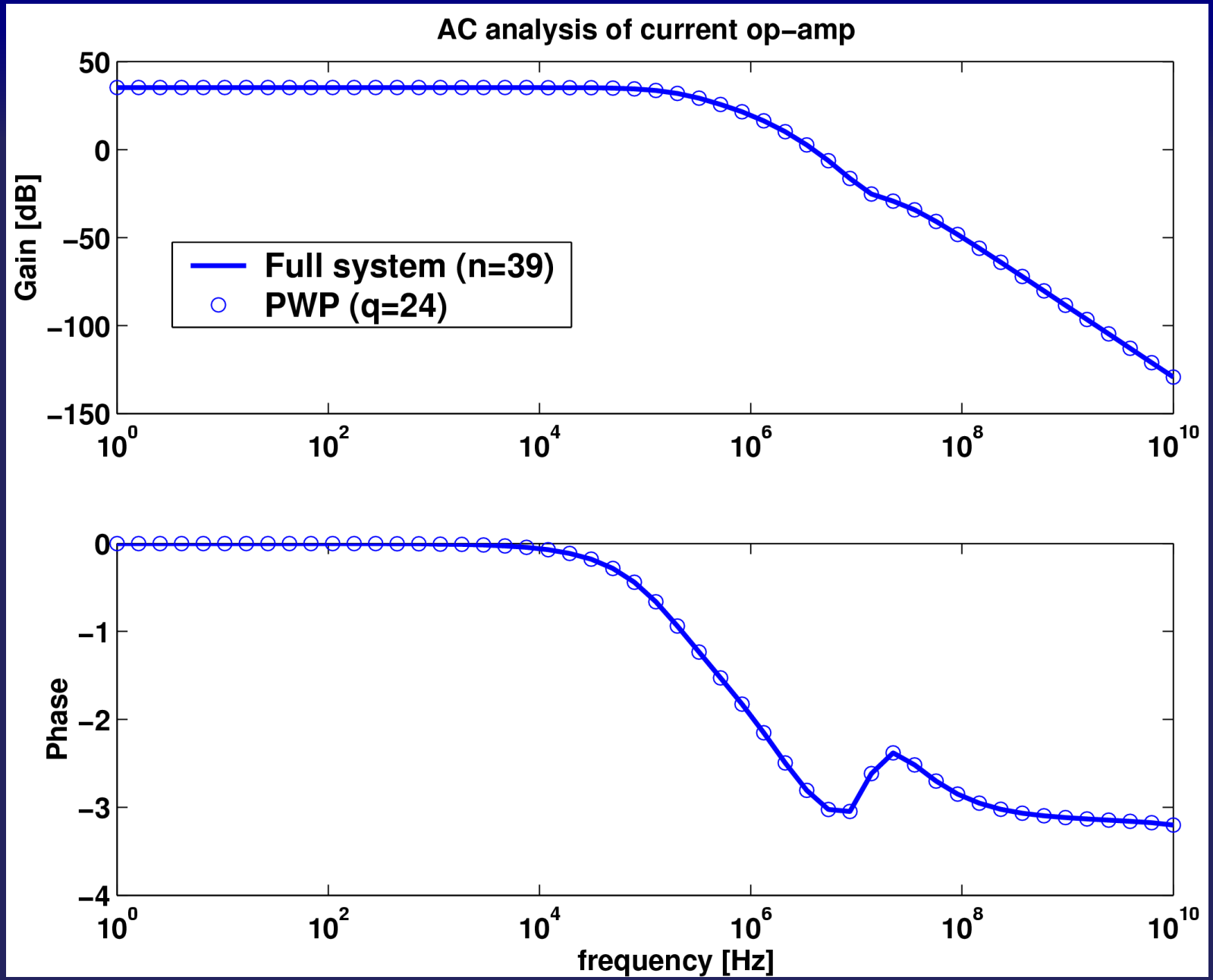
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Macromodel vs Original: AC Sweep 1



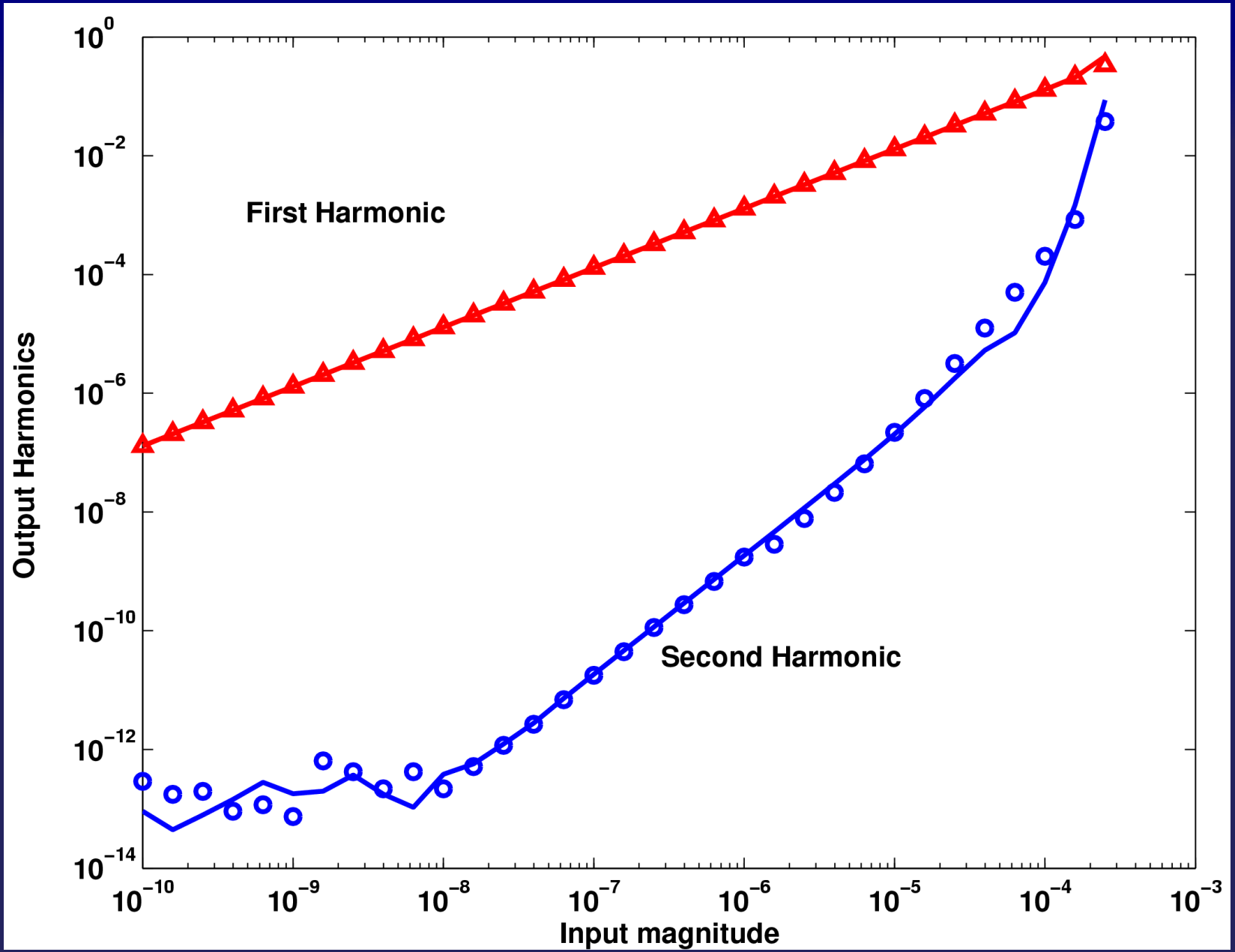
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Macromodel vs Original: AC Sweep 2



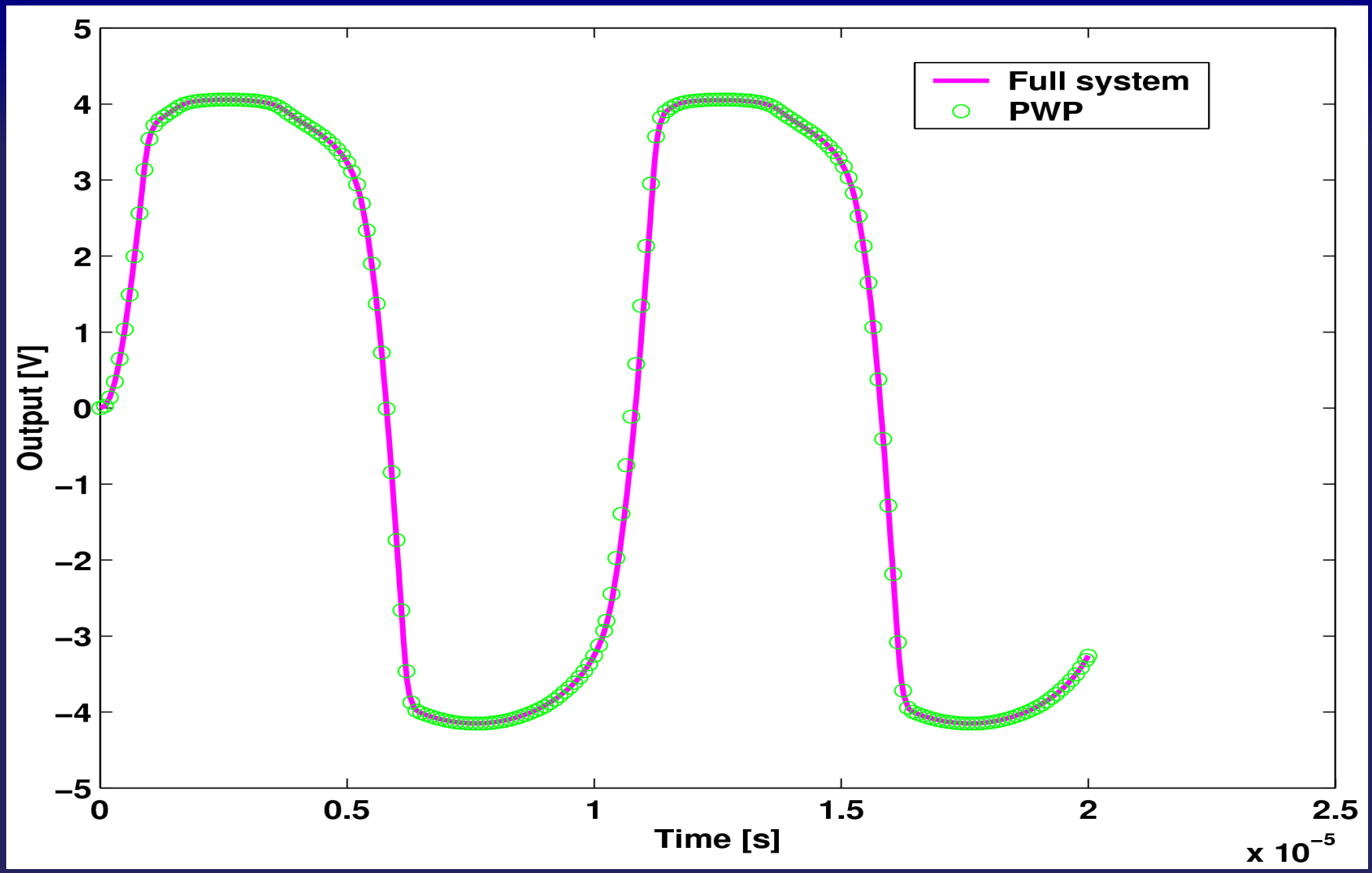
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Macromodel vs Original: Distortion



.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

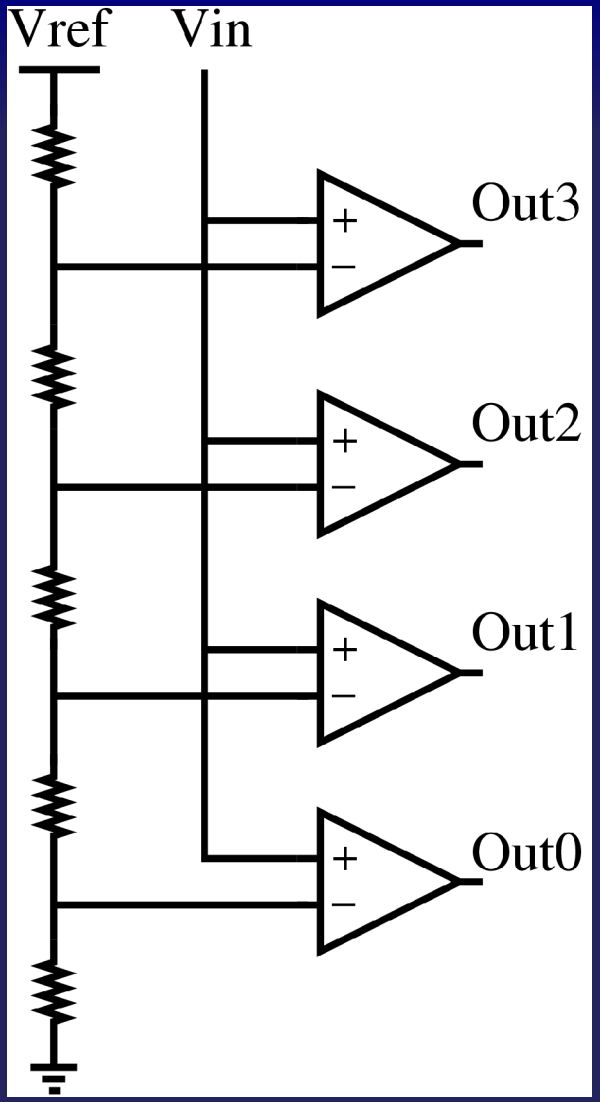
Macromodel vs Original: Transient

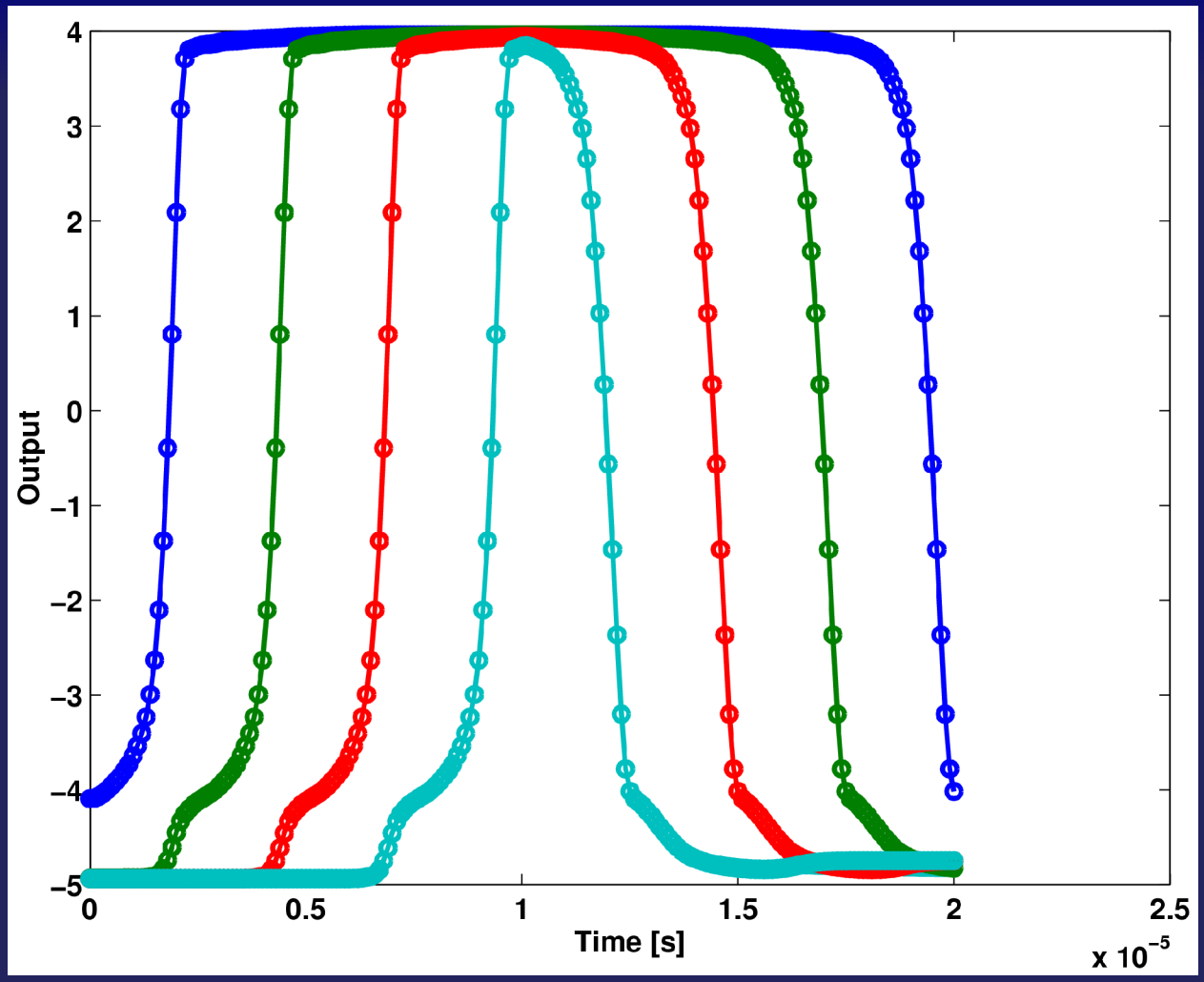


- ***speedup: 41x over original***

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Small ADC: System-level Simulation





Excellent match between original and macromodel

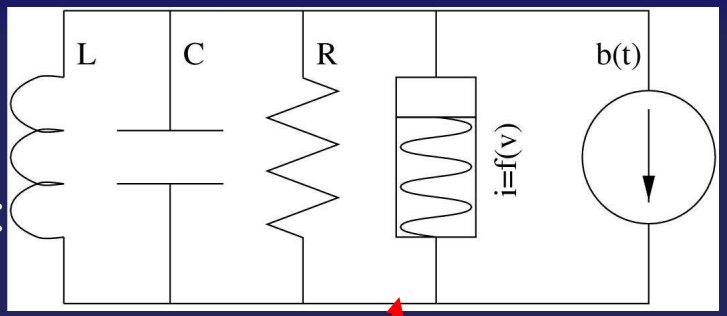
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Macromodelling Oscillatory Systems

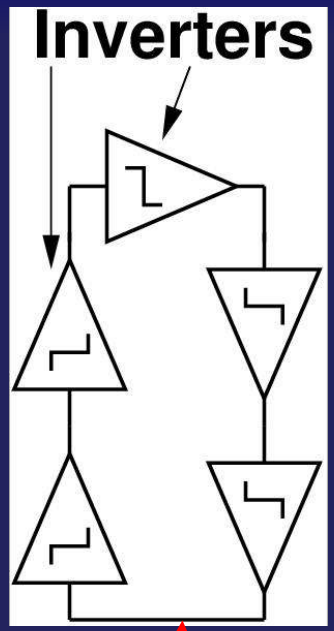
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Oscillators

- Oscillators are critical in communication systems:
  - LC oscillators
  - Ring oscillators
  - Used everywhere:
    - VCOs, PLLs
    - CDR ckts
    - synchronization loops
  - ***Very slow*** to simulate
  - ***Noise prediction problematic***
- Needed:
  - Accurate/fast oscillator macromodelling capability
  - Accurate oscillator jitter/phase noise prediction



-ve feedback LC oscillator



Ring oscillator

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

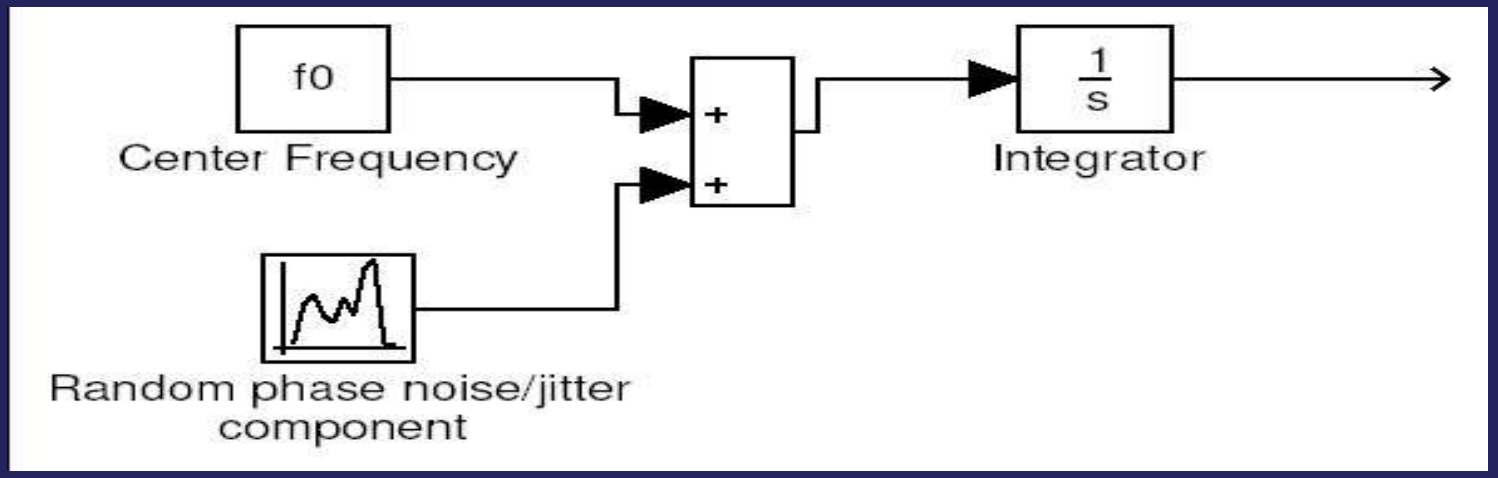
Why Oscillators are a Special Simulation Challenge

- Computation/size/accuracy: much greater than amps/mixers
- Even 1-transistor oscillators (eg, UHF oscs, >100GHz)
  - long startups, tiny timesteps needed
- On-chip RF: 100s to 1000s of transistors
  - VERY challenging to simulate
- Macromodelling offers dramatic speedup
  - Even for 1-transistor oscillator
- Oscillators feature complex phenomena: injection locking
  - oscillator's frequency “locks” to frequency of external input
  - if frequencies close enough, even if input is very small
  - can take extremely long to simulate
  - universal phenomenon: grandfather clocks, fireflies flashing, etc

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Limitations of Linear Macromodels

- Input to output relationship: **linear**



- **Does not capture important phase phenomena correctly**
  - Eg, injection locking, noise spectrum
  - Manually generated (tedious, error-prone, ...)

[Stensby] [Kundert 02]

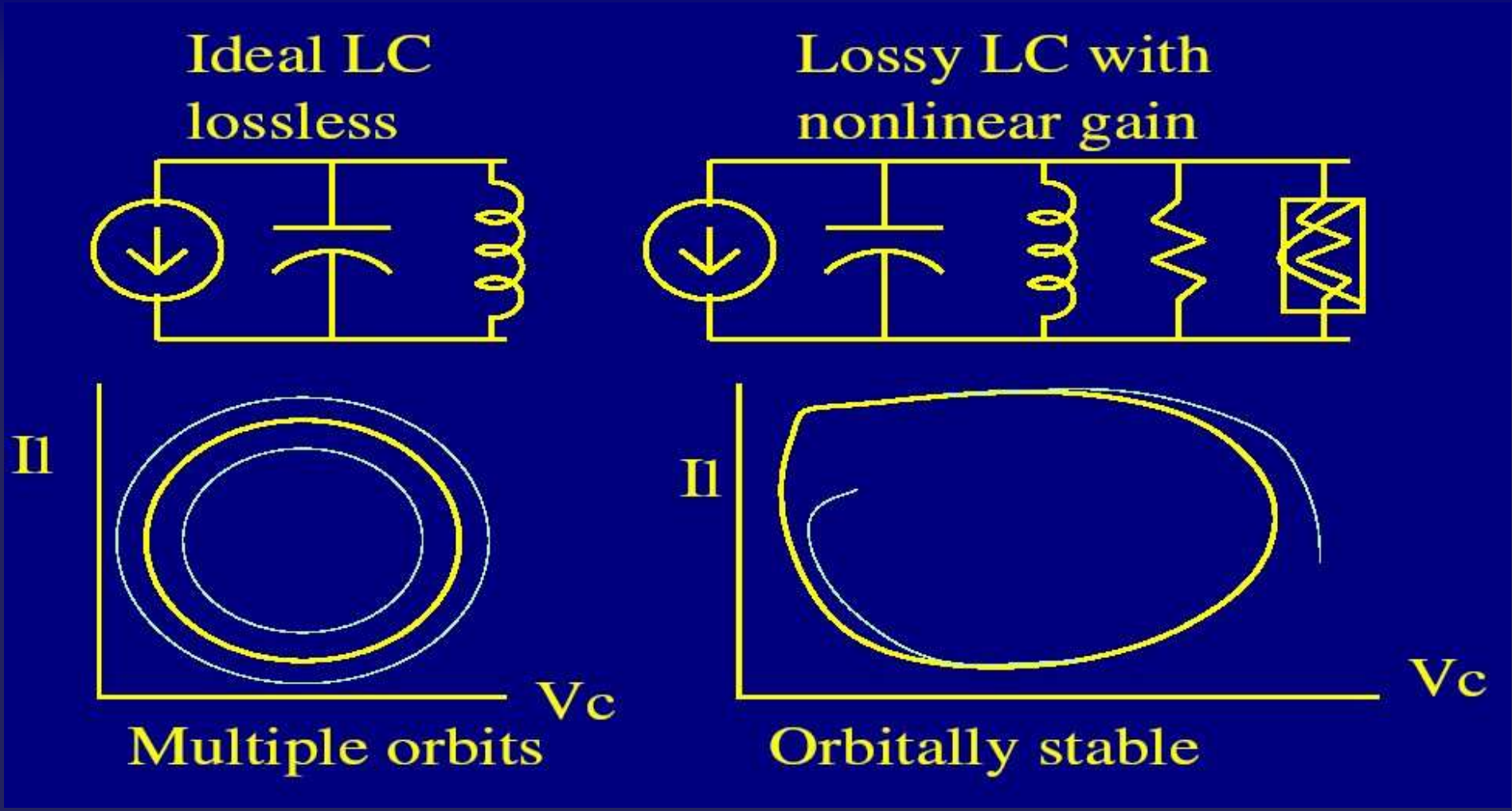
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Automated Nonlinear Oscillator Macromodelling

- Nonlinear: **accurate** I/O capture
  - Injection locking, phase noise spectrum, ...
- Automated generation
  - **SPICE in, macromodel out (Verilog-A, MATLAB, SPICE, etc)**
- Small size
  - **Very fast to simulate compared to full SPICE oscillator circuit**

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

“Ideal” vs Orbitally Stable Oscillators



Nonlinearity cannot be ignored – fundamental to oscillator operation

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Quantifying Oscillator Response

How does the oscillator (VCO) respond to "inputs"?

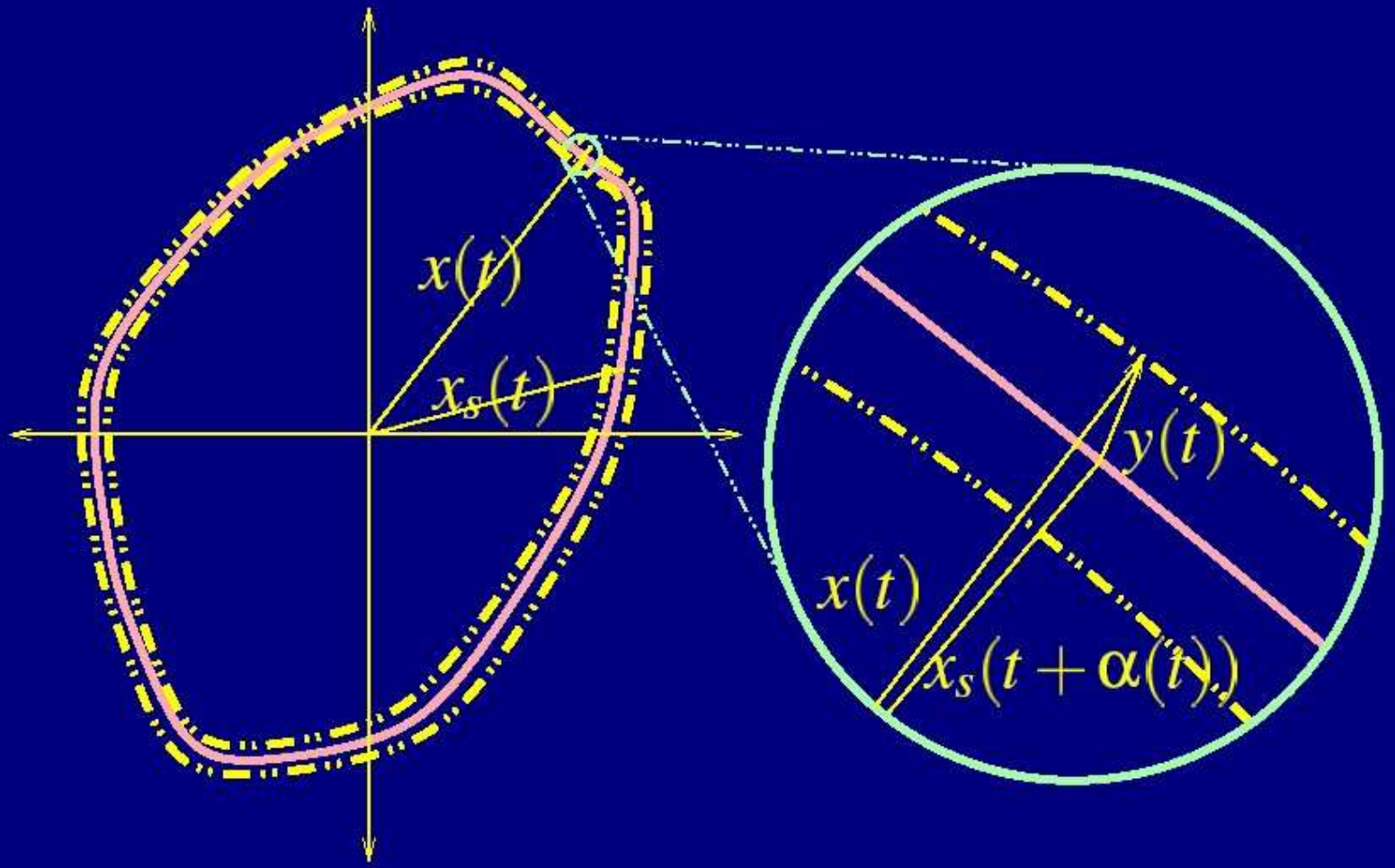
$$\dot{x}(t) = f(x) + \underbrace{b(t)}_{\text{input perturbation}}$$

- No perturbation $\Rightarrow$ perfect periodic solution $x_s(t)$
- Small $b(t)$ perturbation:

$$x(t) = x_s(t + \underbrace{\alpha(t)}_{\text{growing phase error}}) + \underbrace{y(t)}_{\text{small}}$$

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Oscillator: Response to "Inputs"



- Phase error $\alpha(t)$ *shifts track increasingly along limit cycle*
- $y(t)$ creates deviations from limit cycle that *remain small*

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Nonlinear Differential Equation for Phase

$$\dot{\alpha}(t) = v_1(t + \alpha(t)) \cdot b(t)$$

- *Scalar, nonlinear ODE* governs $\alpha(t)$
- $v_1(\cdot)$ is the *Perturbation Projection Vector (PPV)*
- *Projection* of noise perturbation onto PPV determines phase error growth
- PPV is *not obviously related* to anything!
  - periodic Floquet eigenvector of time-varying (linearized) adjoint system
  - PPV can be found from *purely LPTV analysis*
- But: periodicity of PPV makes α equation nonlinear

[Demir Mehrotra Roychowdhury 97, 01]

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

The Perturbation Projection Vector

- $v_1(t)$: "transfer function" relating "input" to oscillator phase response
  - termed the PPV: Perturbation Projection Vector
  - procedure for calculating the PPV is not obvious
    - but computationally efficient
  - In general, PPV does NOT equal the tangent vector of the phase plane plot
    - ie, not equal to the "ISF" [Hajimiri 98]

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

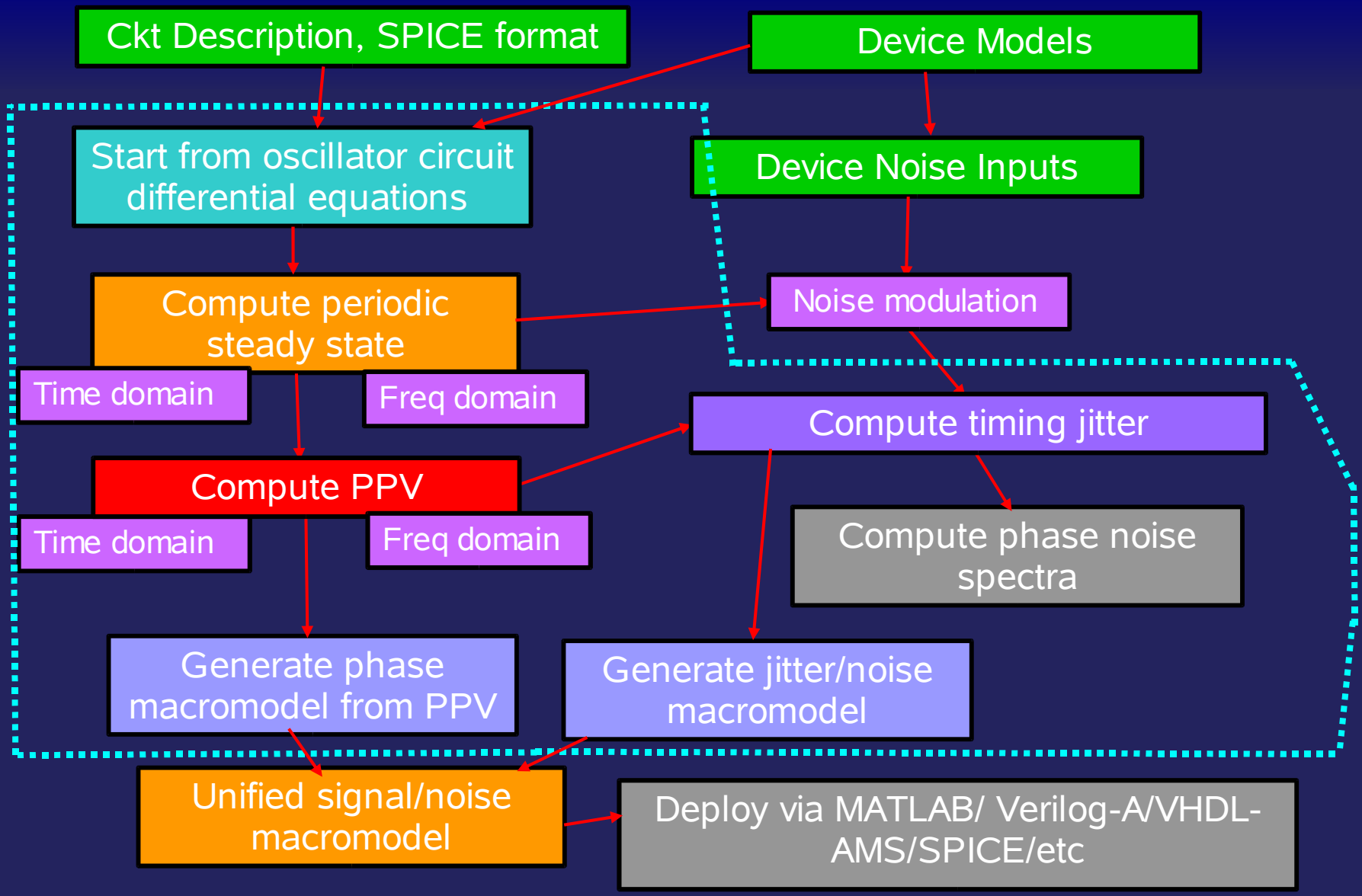
Computing the PPV

- PPV can be computed efficiently from oscillator steady-state quantities
  - first: find the periodic steady-state of oscillator
    - using, eg, HB, shooting, etc.
  - then obtain the same G(t) and C(t) used in LTV reduction
  - form a large block matrix A from "samples" of G(t) and C(t)
  - perform one single linear matrix solution with A
    - can be performed efficiently for large oscillators

[Demir Roychowdhury TCAD 03]

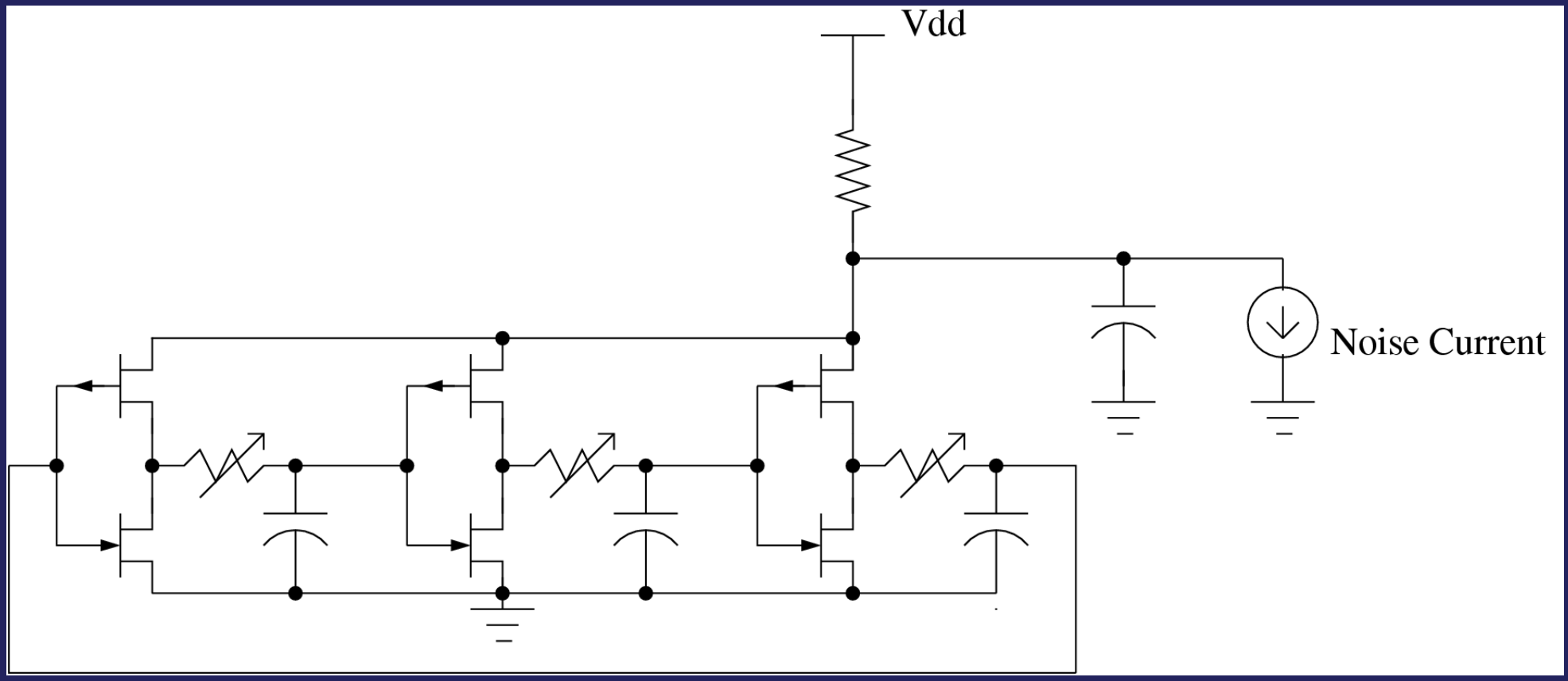
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Oscillator Macromodelling Steps



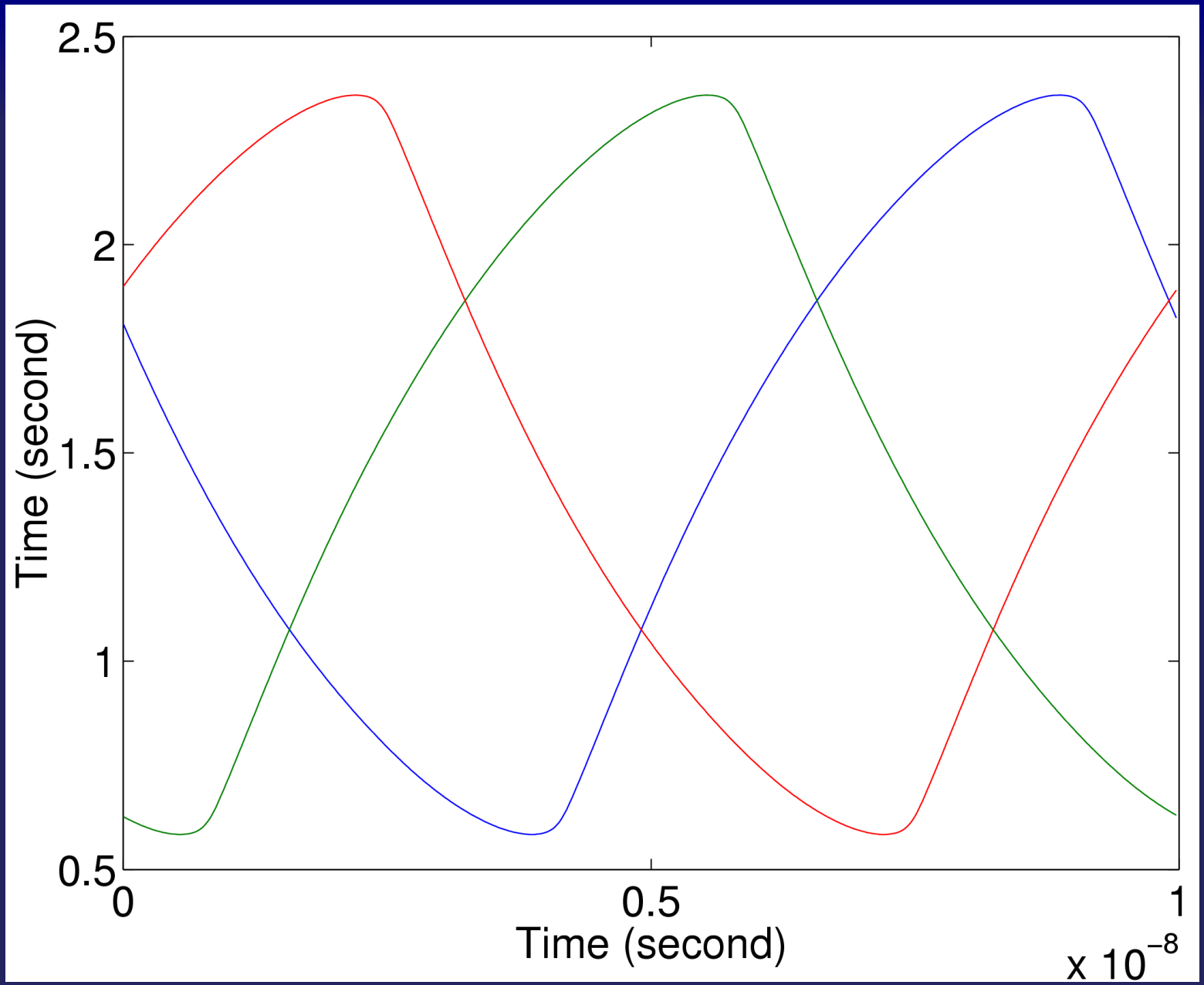
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Example: Ring-Oscillator based VCO



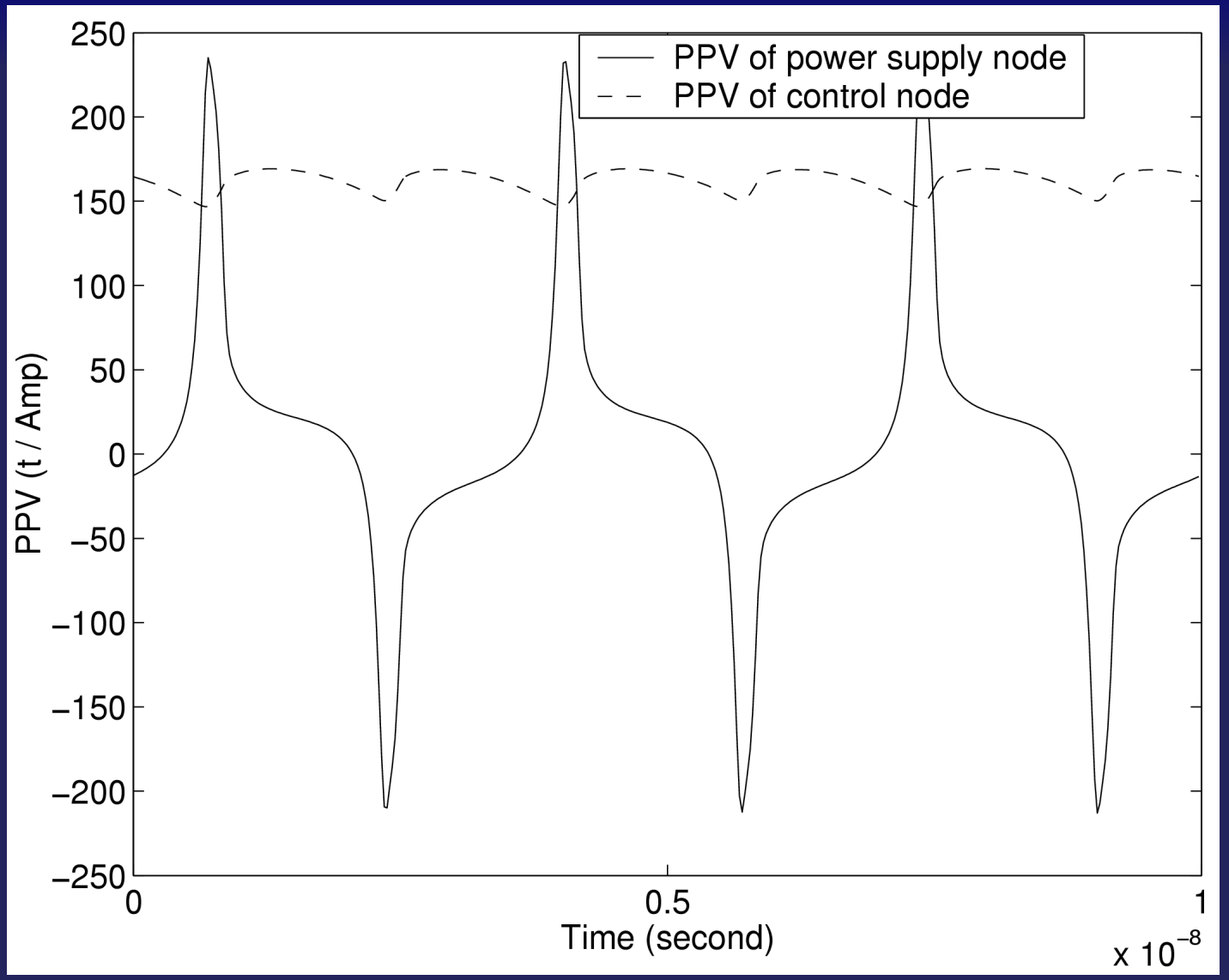
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Ring Oscillator VCO: Steady State Oscillation



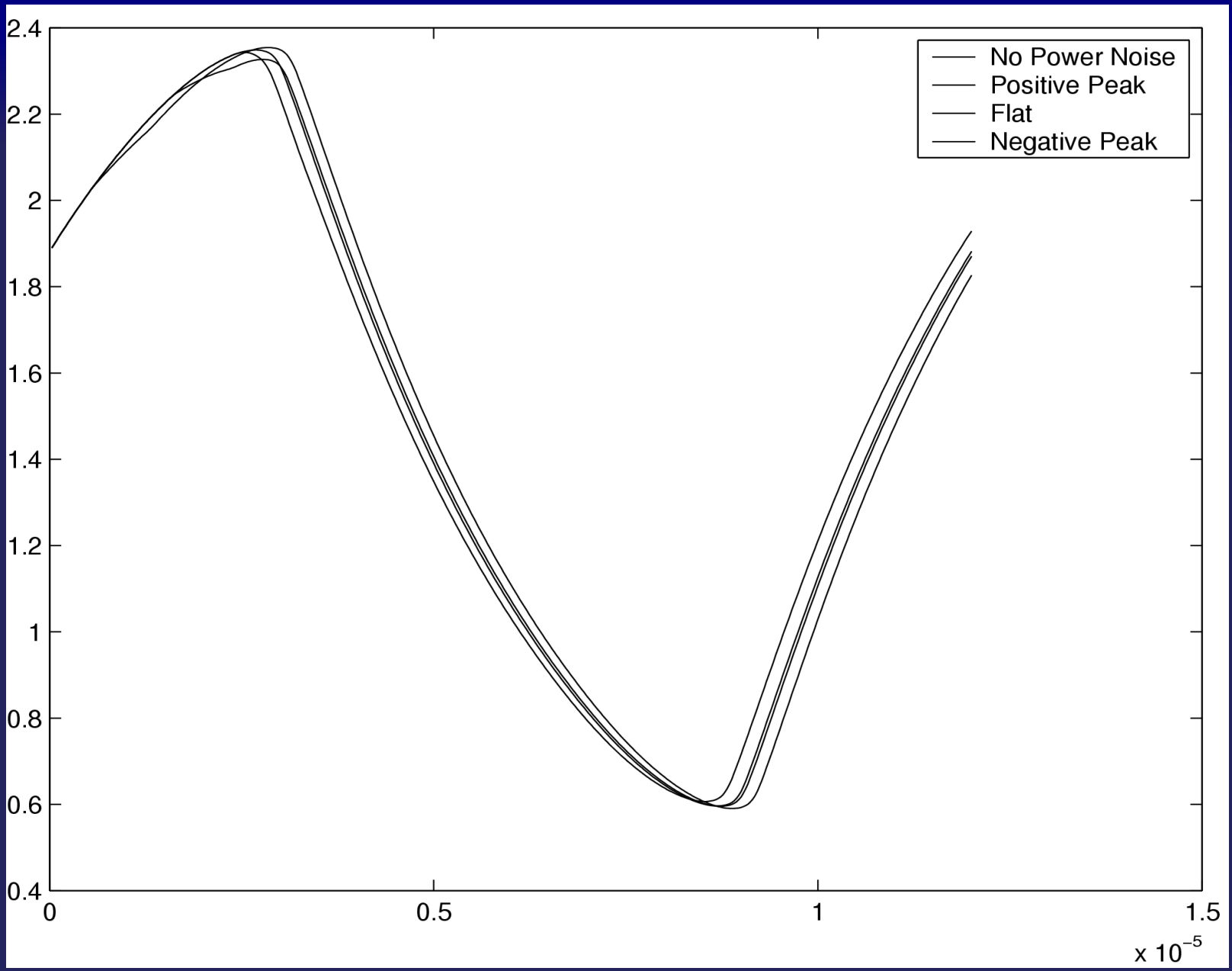
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

VCO Perturbation Projection Vectors: Control Node and Power Supply Components



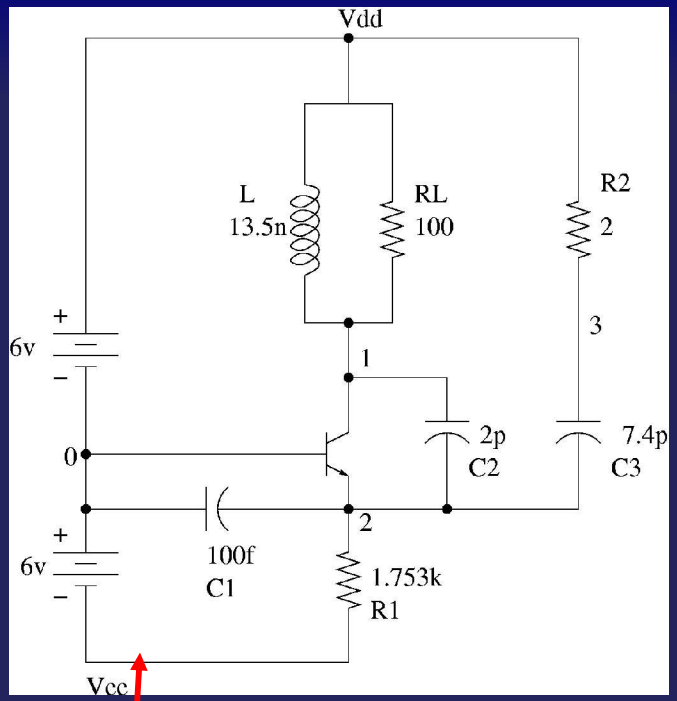
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

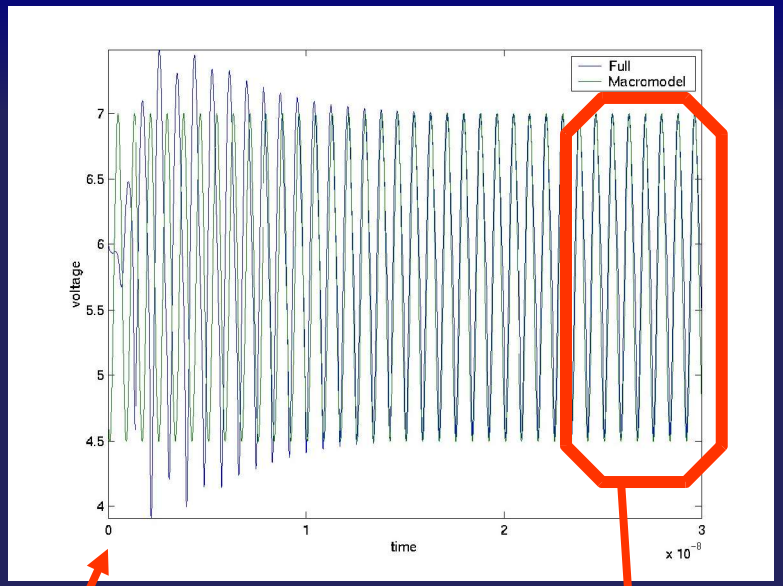
Ring Oscillator VCO: Shifts due to Supply Noise

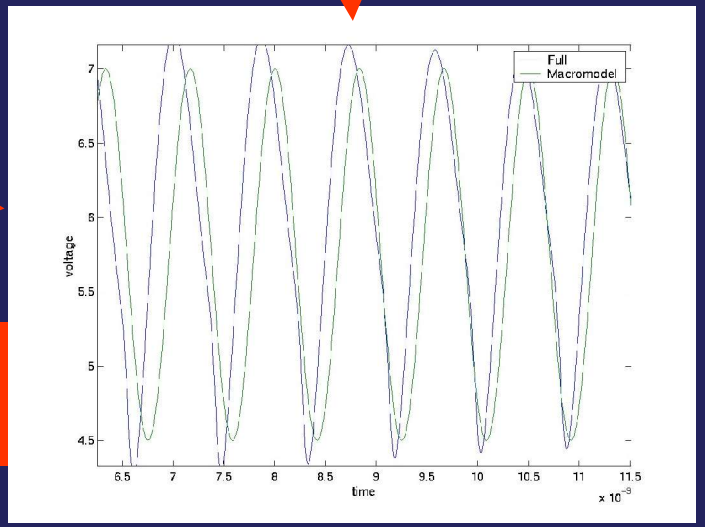


.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Capturing Injection Locking in a Colpitts (LC) Oscillator







Colpitts oscillator

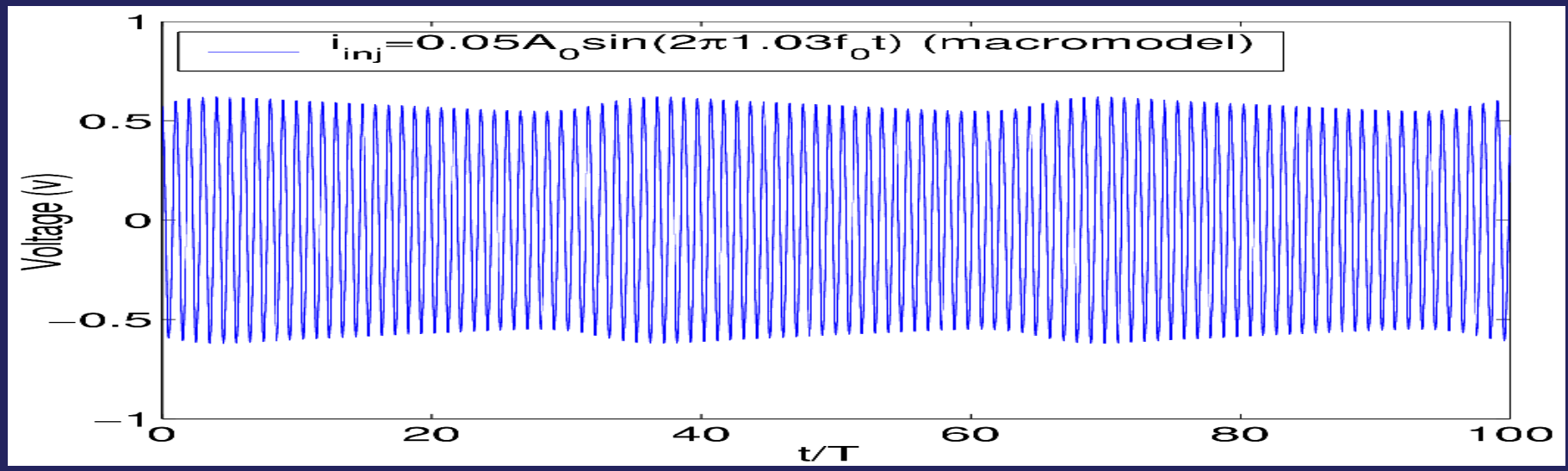
Full simulation vs macromodel

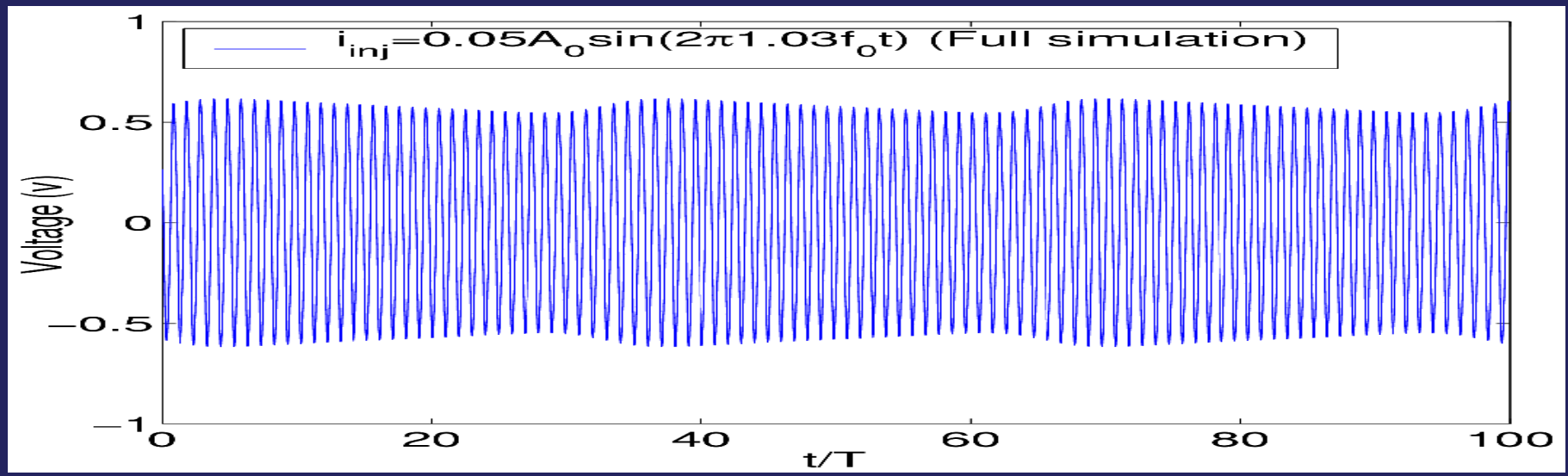
- ***89x speedup over original***

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Capturing Amplitude Changes

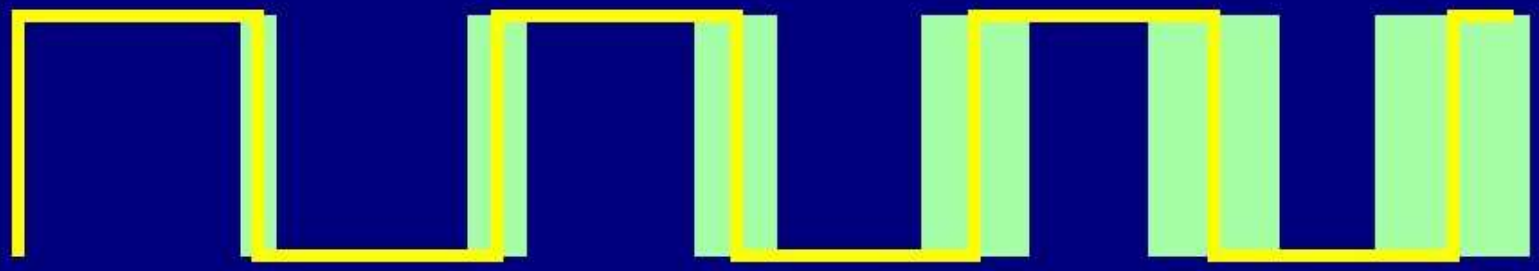
- ***nonlinear phase + amplitude components (via LTV reduction)***



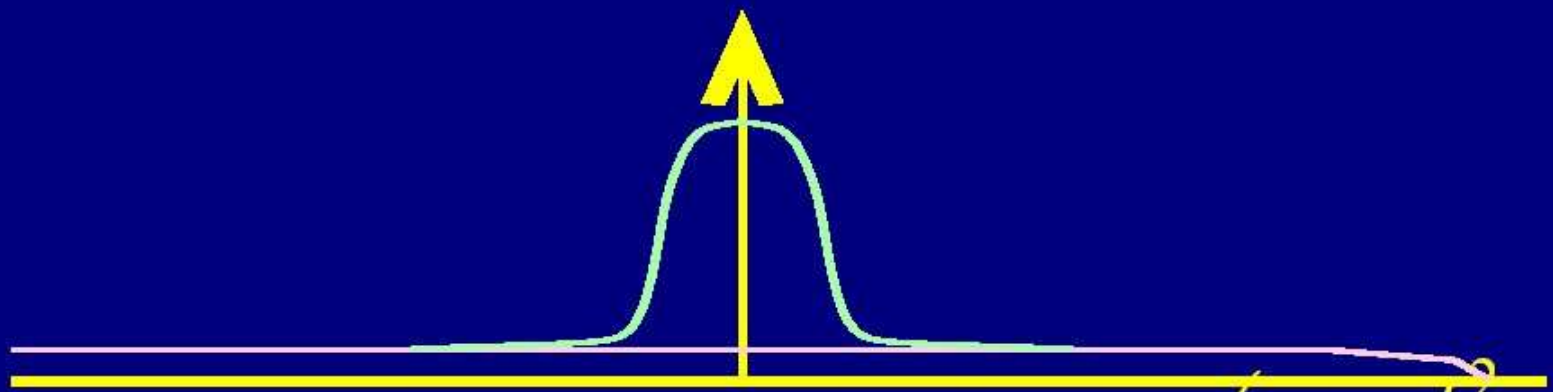


.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

PPV useful for Phase Noise/Jitter MMing



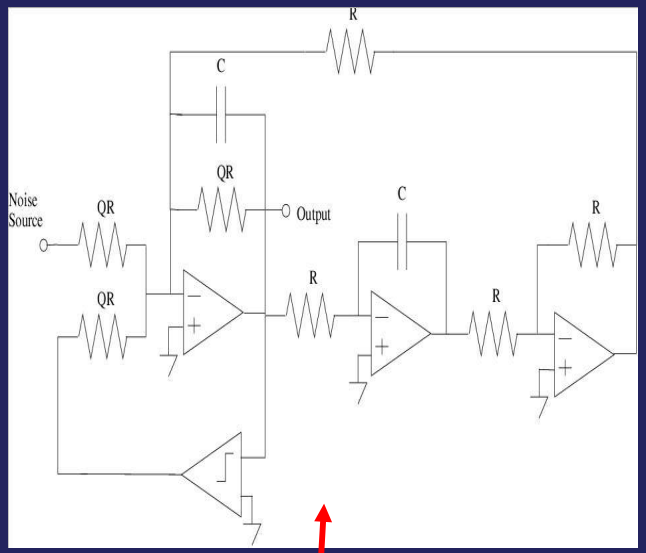
- Timing jitter variance (per cycle) = cT



- *Lorenzian spectrum:* $\mathcal{L}(f_m) = 10 \log_{10} \left(\frac{f_0^2 c}{\pi^2 f_0^4 c^2 + f_m^2} \right)$

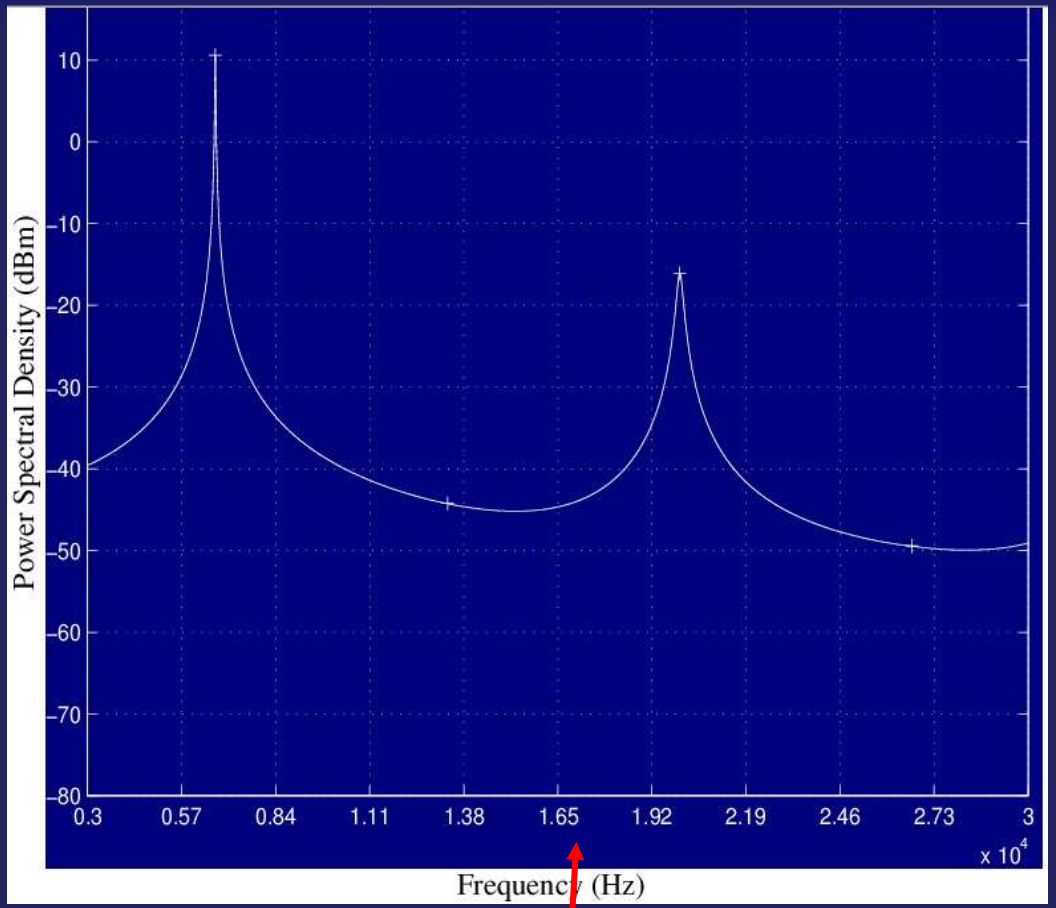
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Oscillator Phase Noise Spectrum



Tow-Thomas oscillator

[Toth TCAS-1 1992]

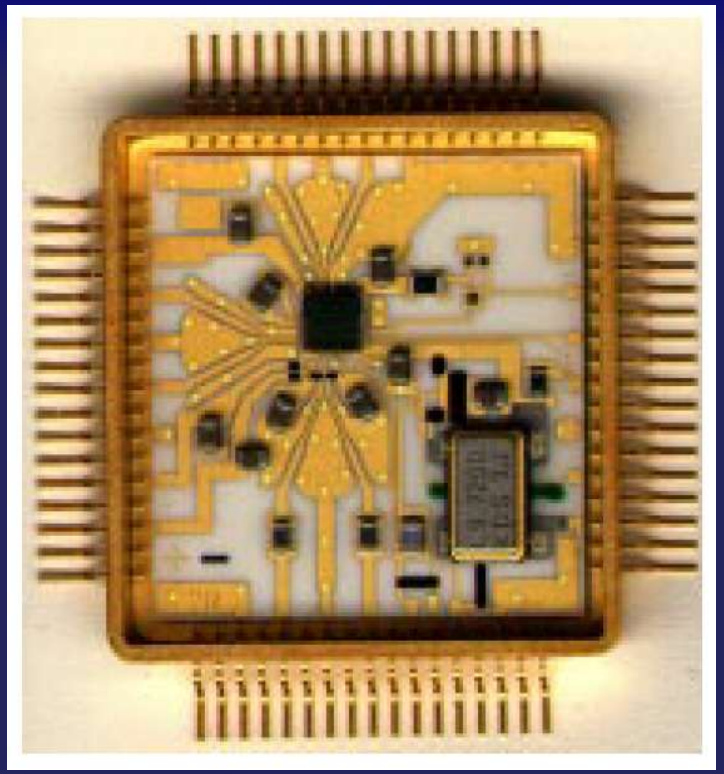


Phase noise spectrum

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

PLLs: Commodity, High-Margin

Rick Walker, HP/Agilent



Rick Walker, HP/Agilent

1.25Gbps Ethernet xcvr
<< $10

2.44Gbps SONET CDR
$500

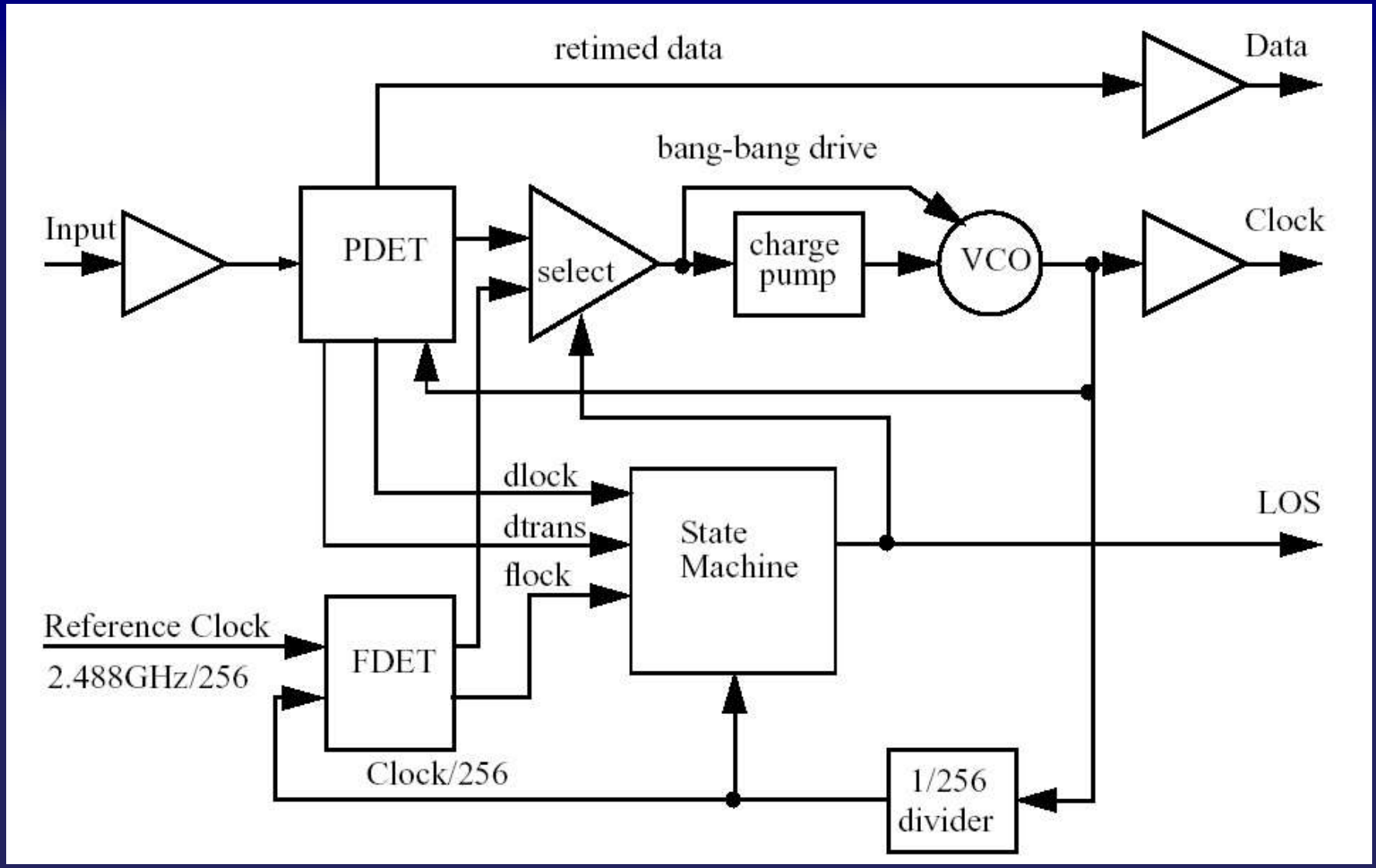
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

PLL Applications

- Clock and data recovery (CDR)
  - synchronization: Costas Loop
  - applications: communication systems, disk-drive read-channels
- Frequency synthesizers
  - integer-N, fractional-N synthesizers
  - direct digital to GPSK modulation

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

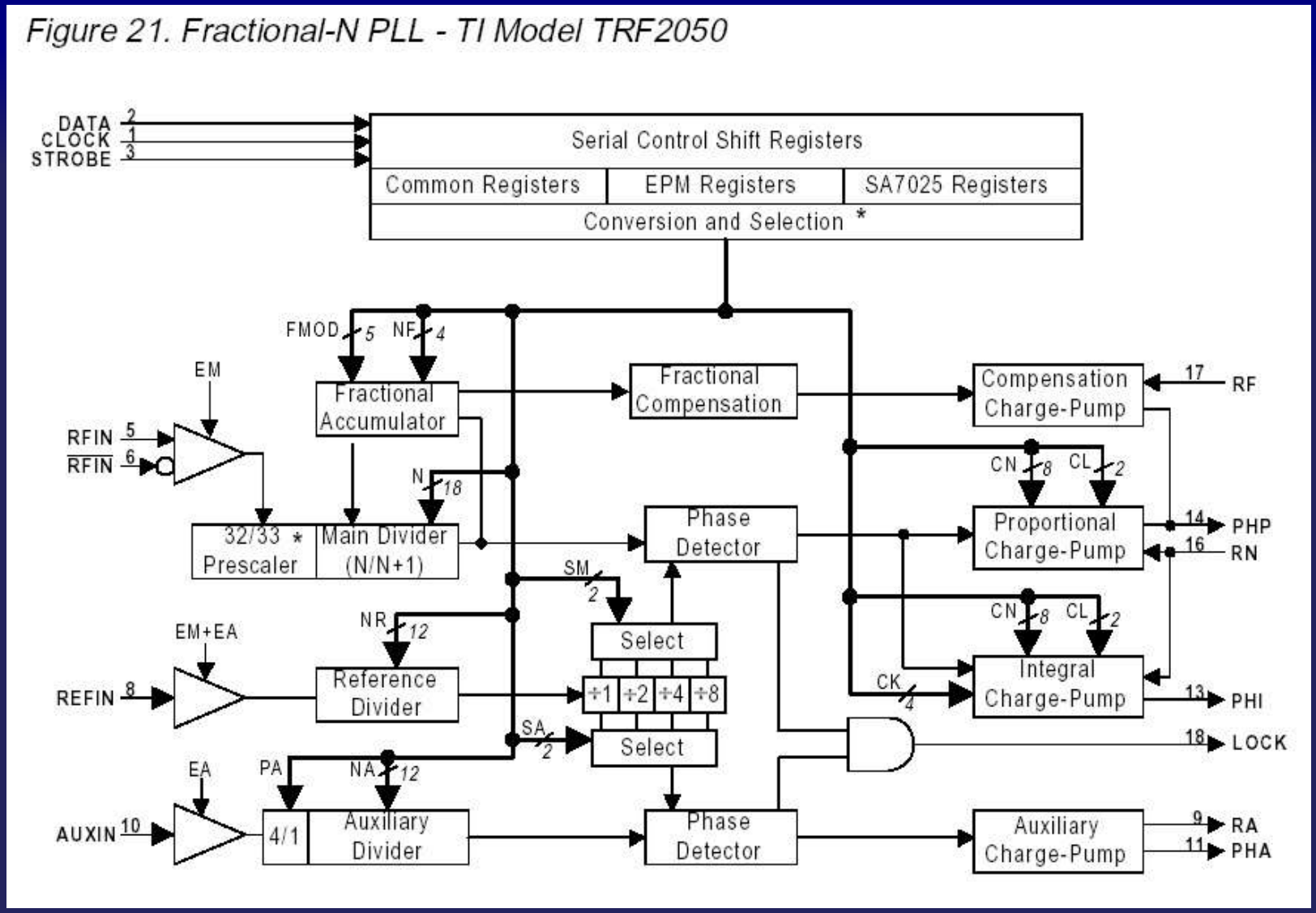
Clock and Data Recovery (CDR) ckt



Mixed-signal PLL: very hard to simulate at ckt level

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

TI TRF-2050 Fractional-N PLL



Texas Instruments

Large; complex; many interconnected functional blocks

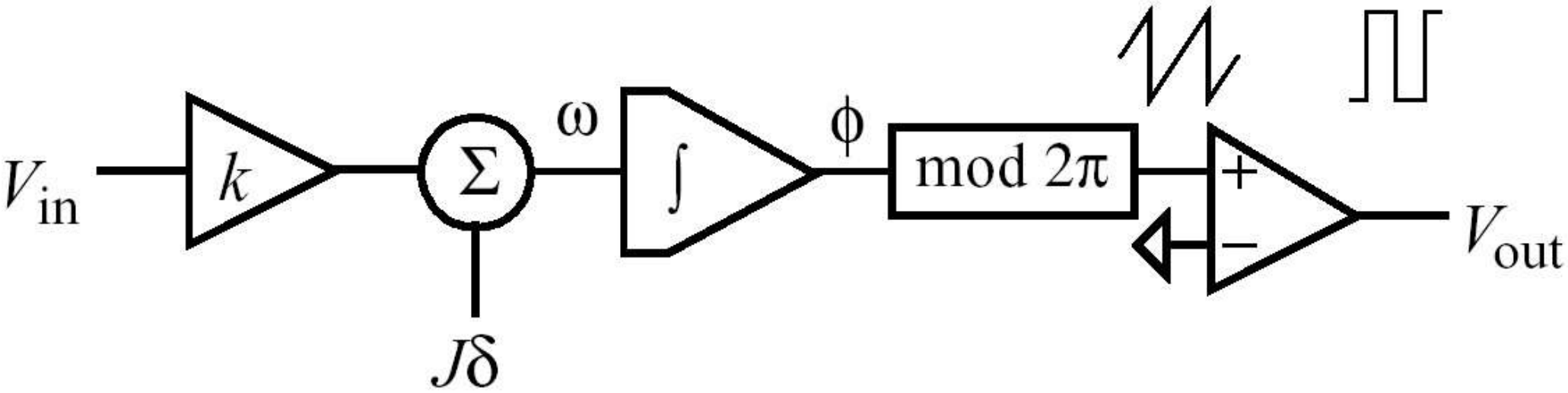
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

PLL Simulation: Difficulties

- SPICE: Inaccurate + extremely time-consuming
  - biggest single culprit: VCO
    - 100K cycles of VCO typical for PLL simulation
  - combination of digital (PFD, dividers) and complicated analog (charge pump, VCO)
  - noise characterization (jitter) critical and difficult
  - capture/lock-in/freq-hopping dynamic phenomena
  - Injection-locking: 1st order PLL
- Full system with PLLs: very difficult today
  - Current macromodel-based approaches ad-hoc, manual, cannot capture dynamics, nonlinearity well

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Linear Macromodelling of VCOs/PLLs



Ken Kundert, Cadence

[Kundert 02]

- Manually generated VCO macromodel
- Output VCO phase = integral of input control
- Linear => cannot capture nonlinear phenomena (injection locking)
- Good for intuition, hand calculations, noise trends
- Can be grossly wrong for jitter caused by power grid/substrate

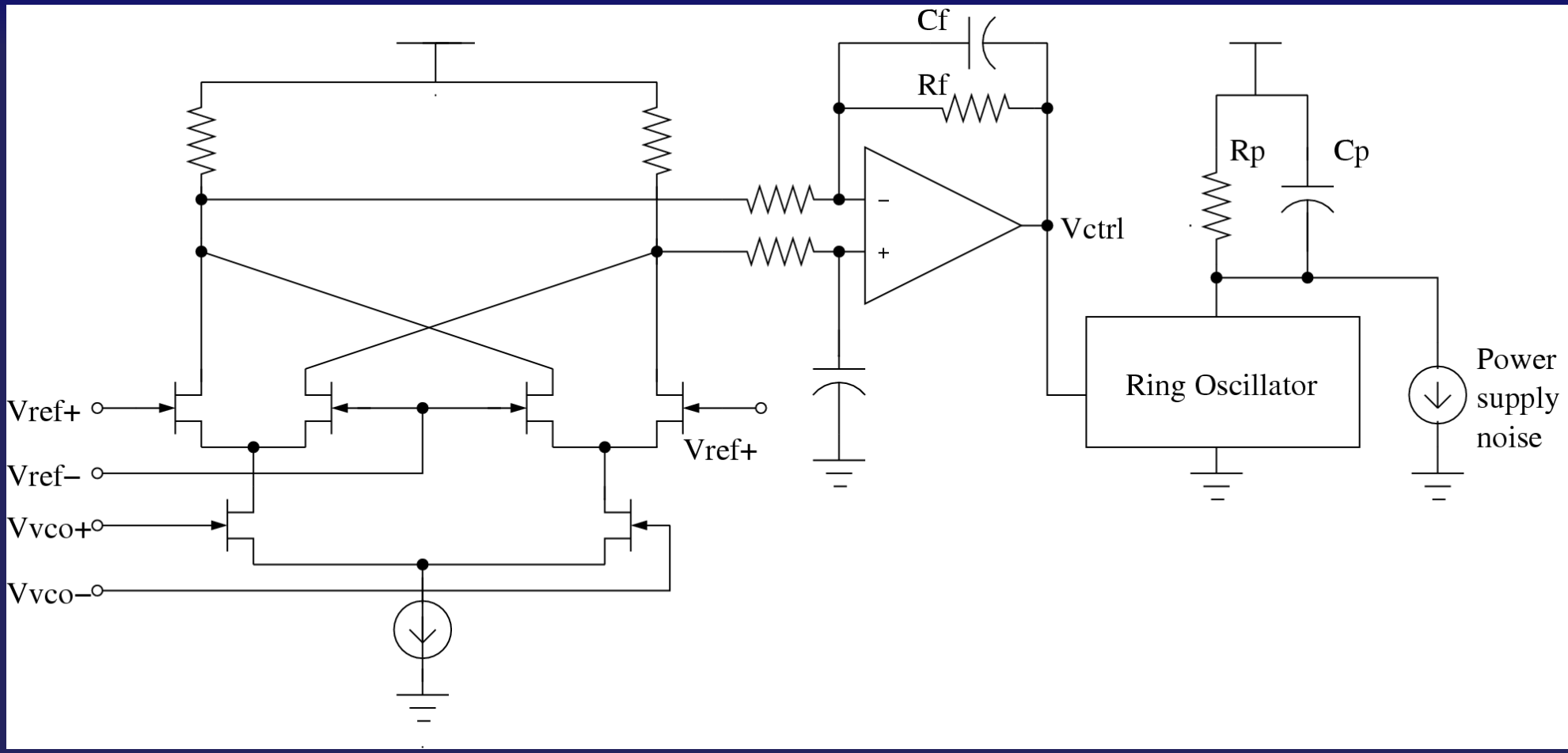
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

System Simulation with VCO Macromodels: PLLs

- Use nonlinear phase macromodel of VCO in phase macromodel for PLL [Lai/Roychowdhury 04]
  - PPV "extracted" by algorithm
  - amplitude components could also be used
- Replace other components (PFD, LPF, divider) also with macromodels if necessay
  - often, PFD and LPF small: keep full SPICE ckt
  - divider macromodel => simple scaling of phase
- Single VCO macromodel to capture desired and undesired influences

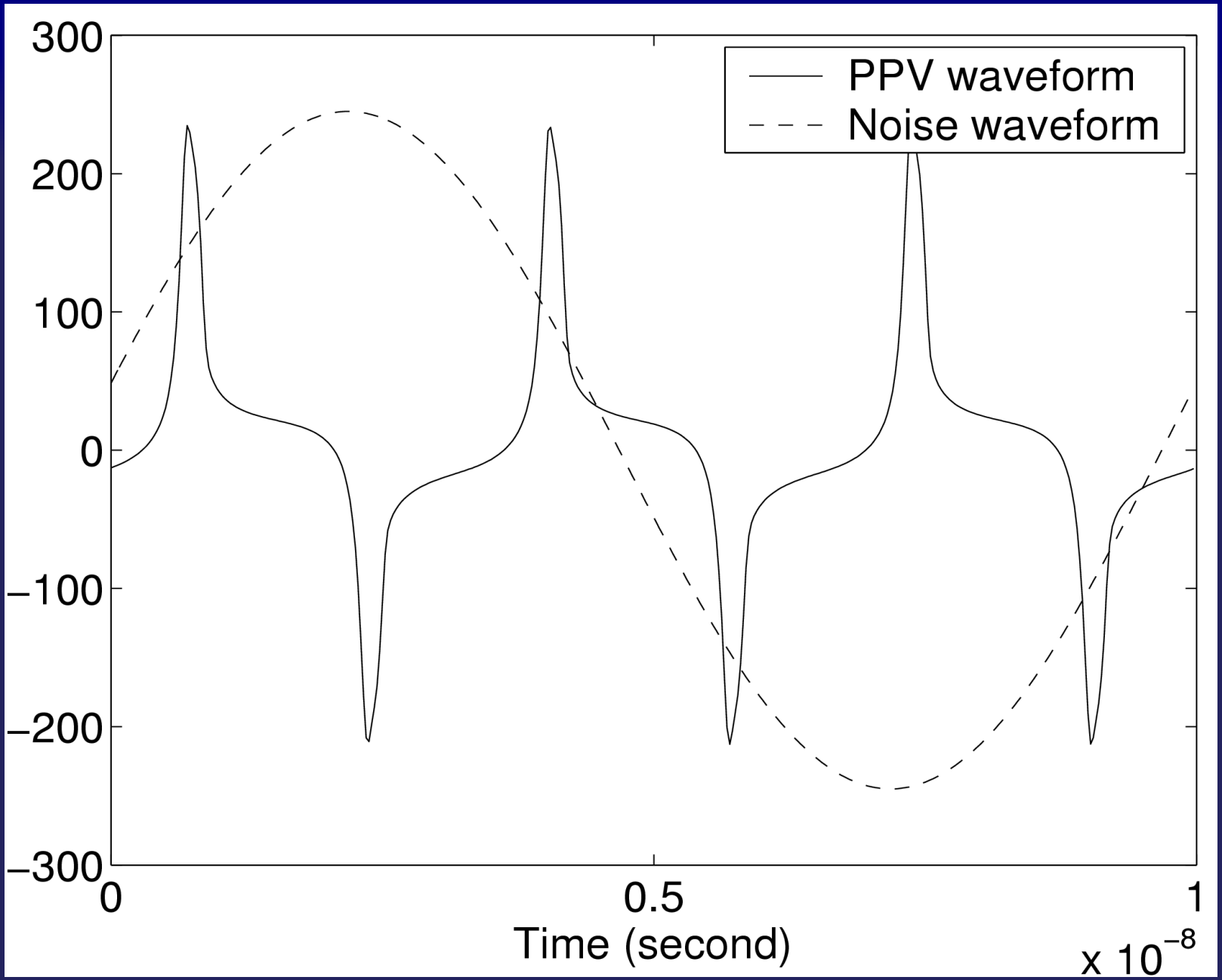
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Simple Ring-Oscillator-based Phase-Locked Loop



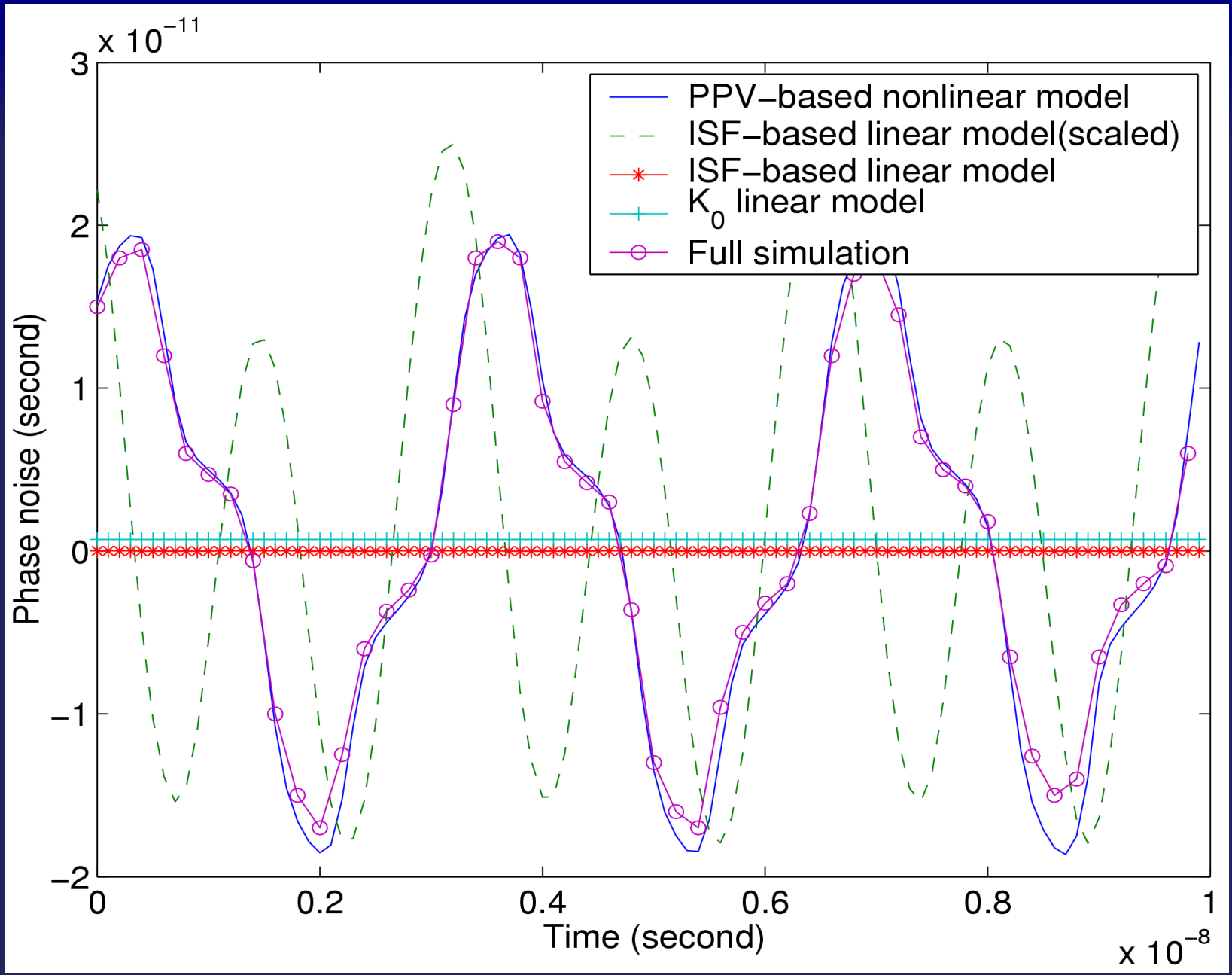
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Expected Impact of 1st harmonic Supply Noise



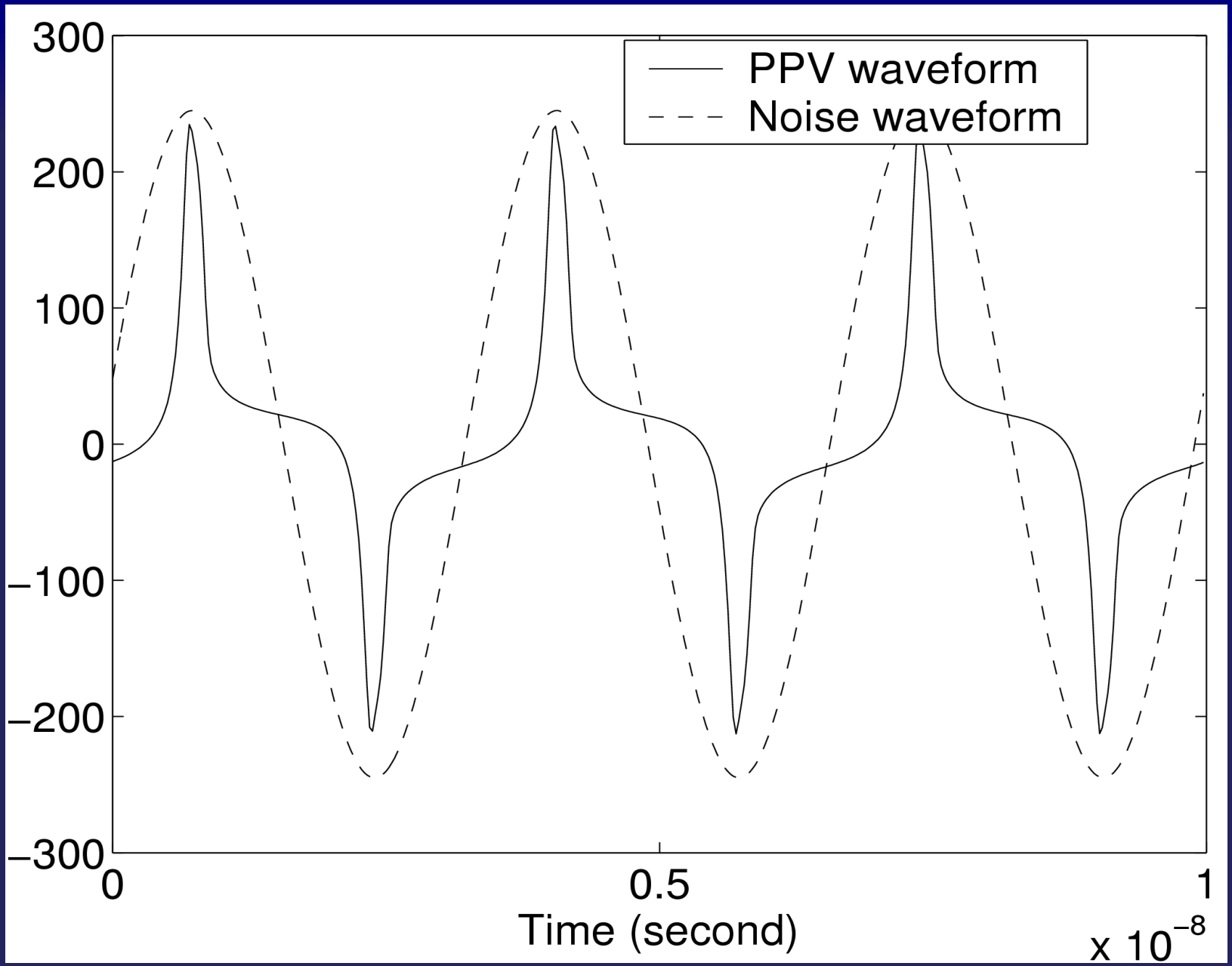
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

PLL Phase Reponse to Periodic Supply Noise



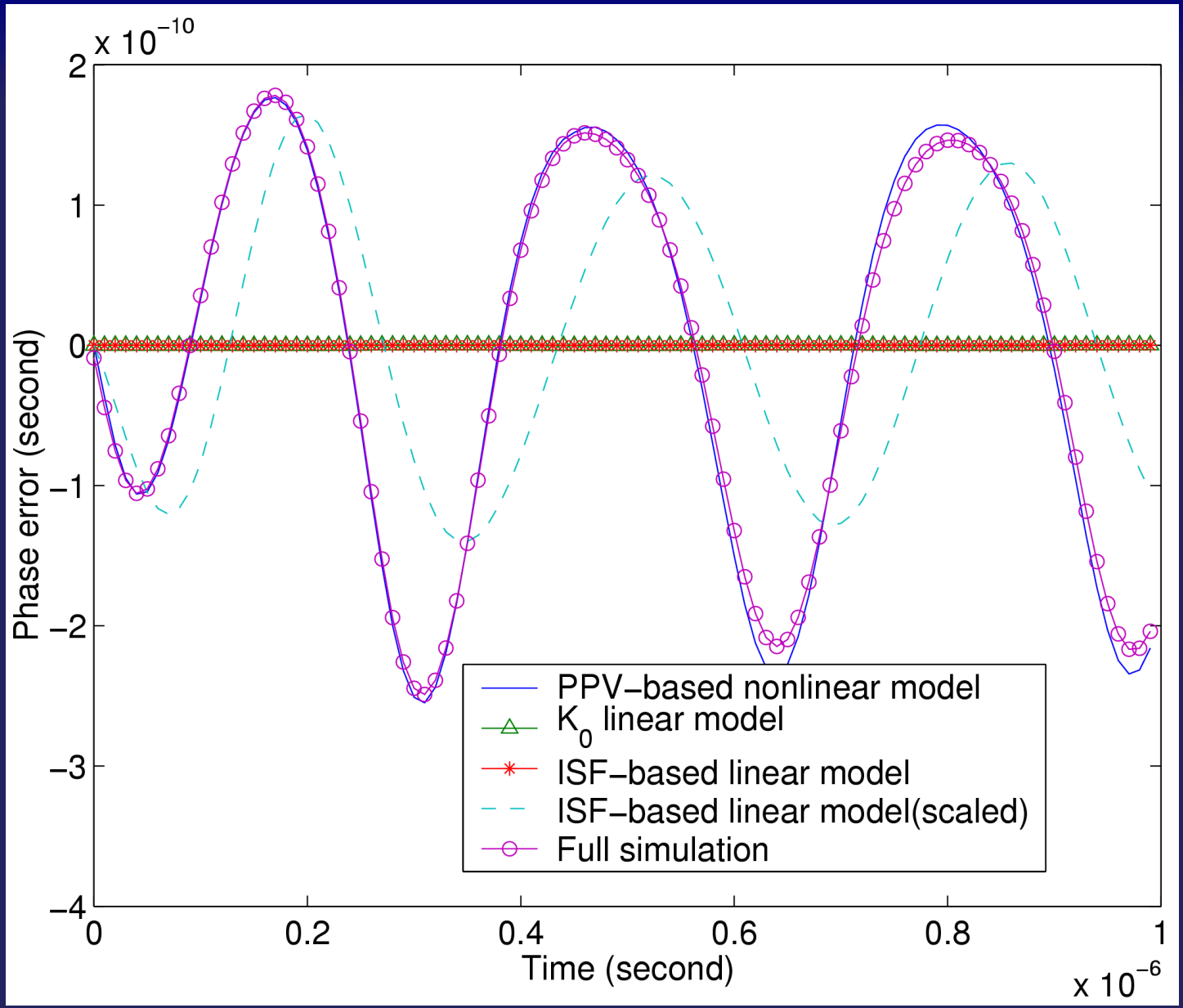
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Expected Impact of 3rd Harmonic Supply Noise



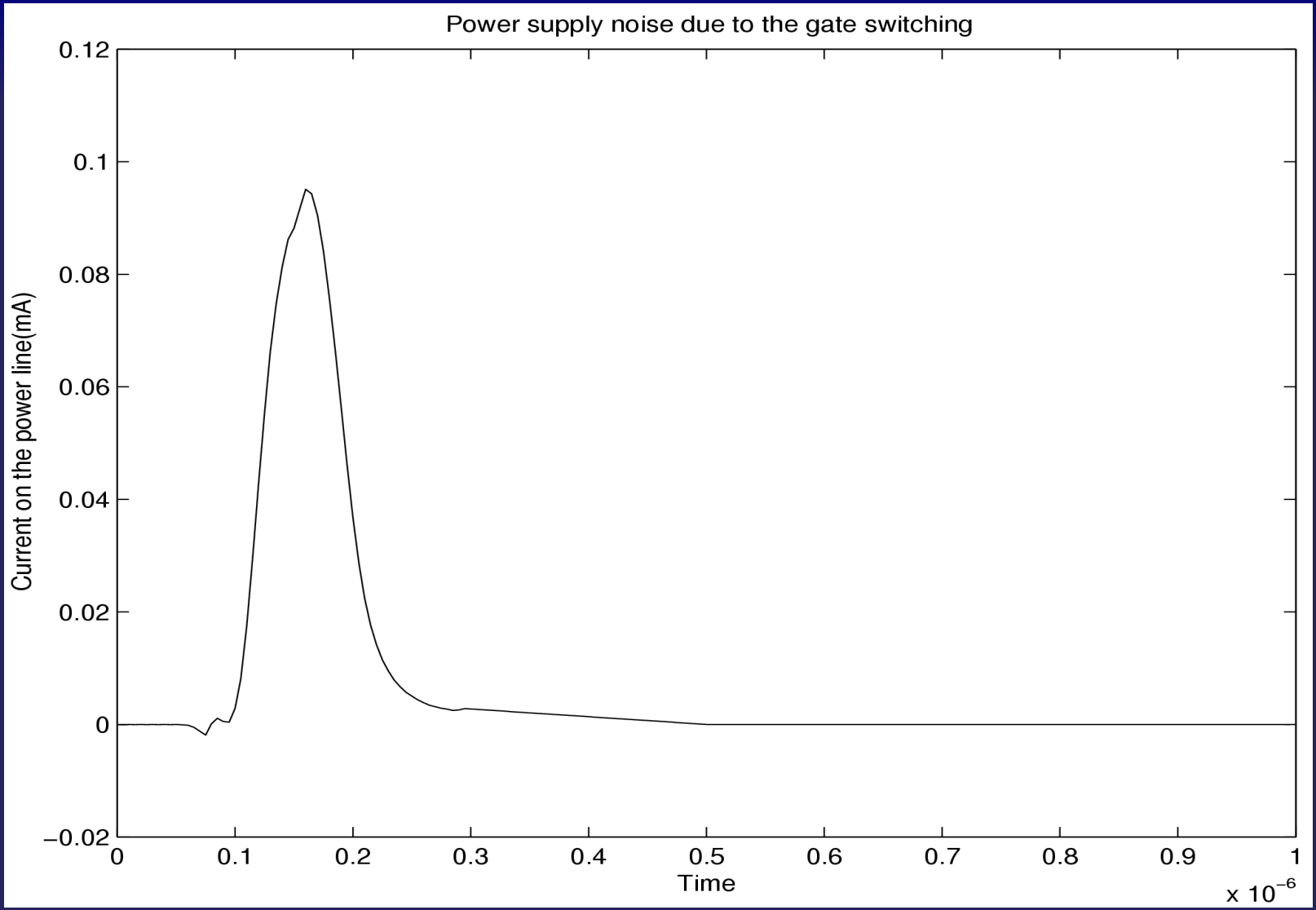
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

3rd Harmonic Supply Noise: Phase Macromodel vs Full PLL simulation



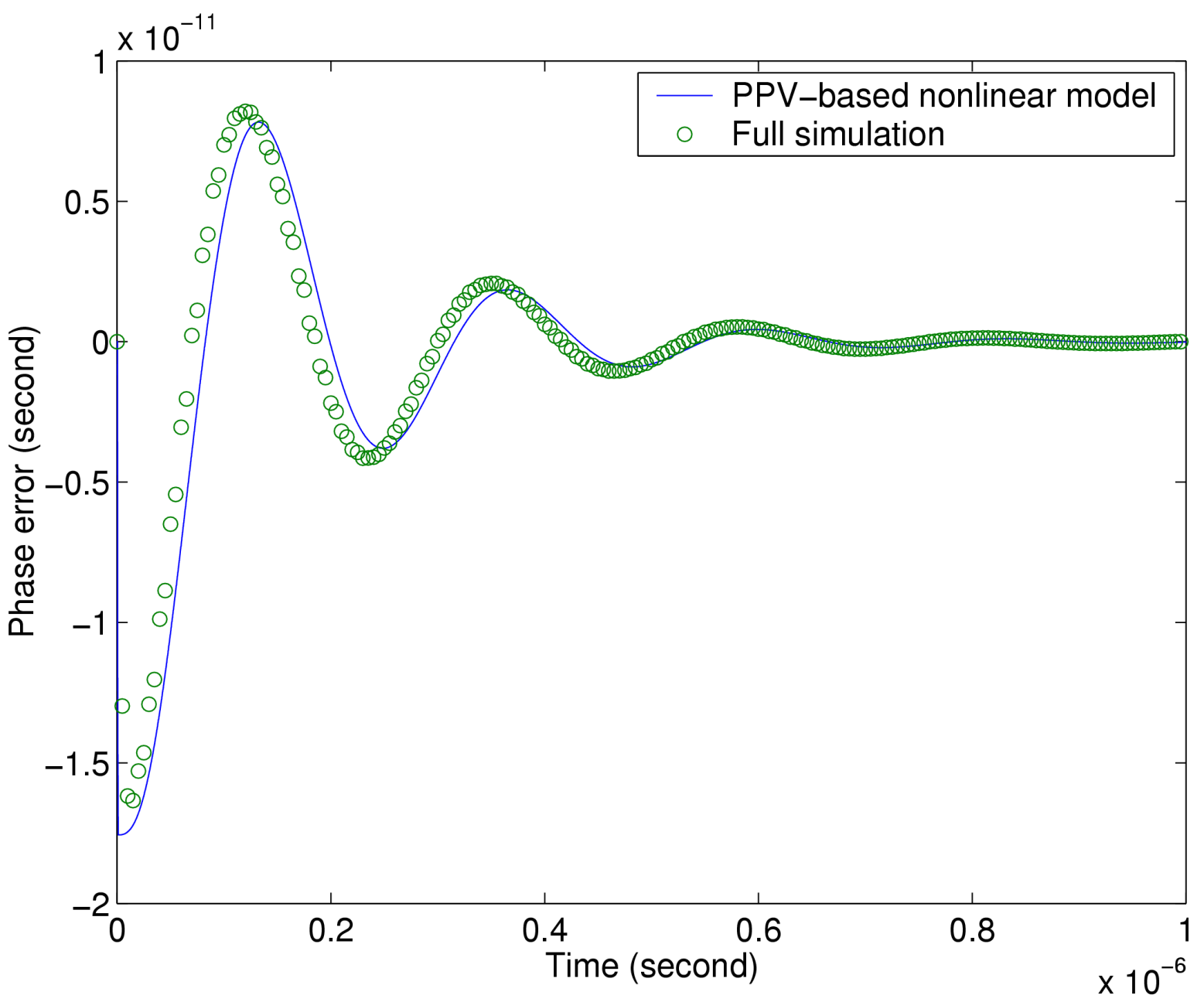
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Power Supply Spike due to Single Switching



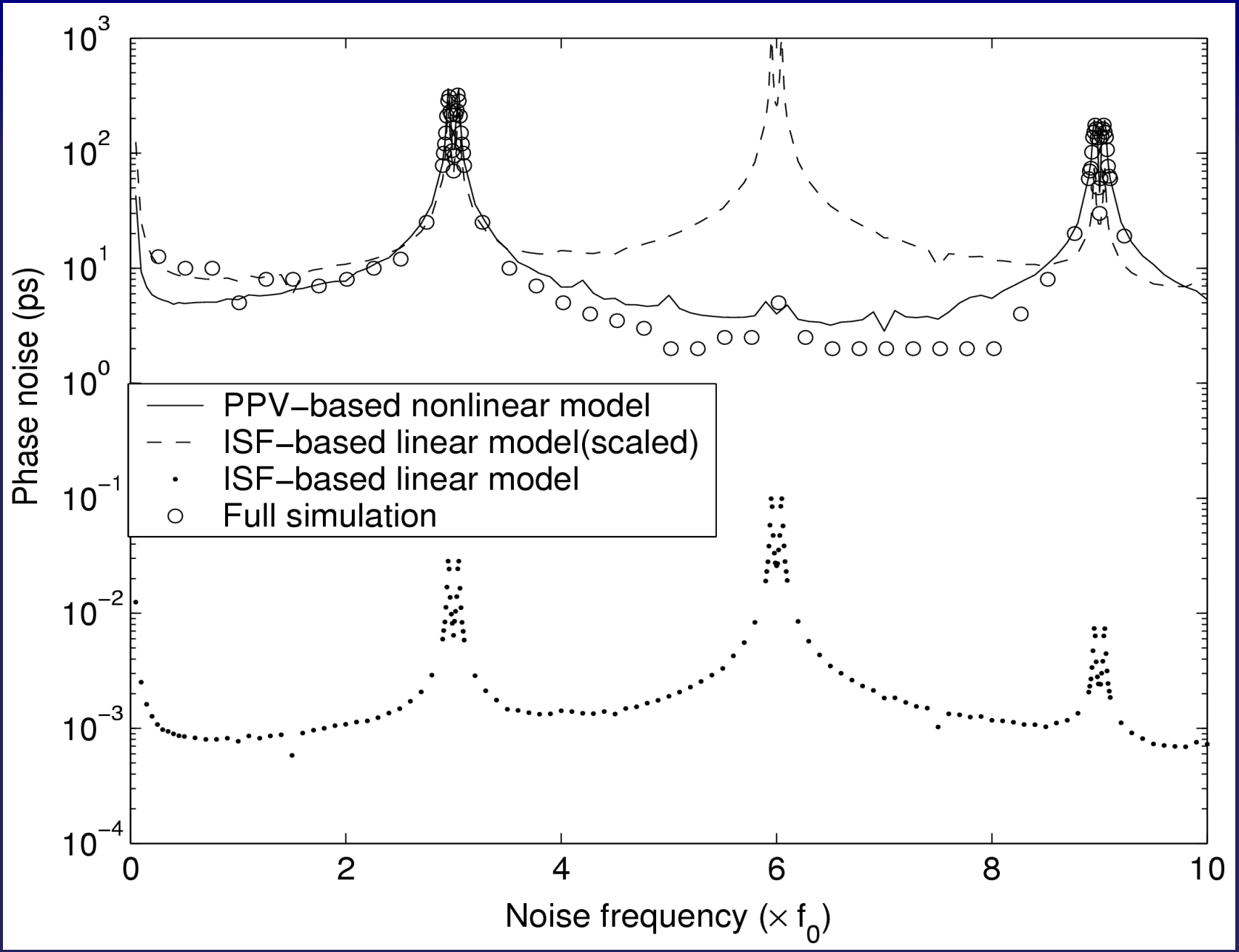
.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Phase Response of PLL to Supply Noise Spike



.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Sinusoidal Supply Noise: Effect on PLL Phase



.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

Conclusion

- Coming "soon": automated nonlinear macromodelling for...
  - op-amps, mixers, switching filters, comparators
  - oscillators, VCOs (any kind: LC/ring/relaxation/etc)
  - large digital aggressor blocks: interference macromodels
  - bottom-up "extracted" macromodels: much more accurate, second-order effects, ...
- Use of core macromodels for system simulation
  - ADCs/DACs/Sigma-Deltas
  - PLLs/Sigma-Deltas (incl jitter and noise)
  - SOCs
  - MATLAB/Simulink/Verilog-A/VHDL-AMS/etc
- Automated Macromodelling: the ONLY sustainable methodology for effective CAD support of nano-era analog, mixed-signal and RF design

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

References

- J. Roychowdhury, “Reduced-order modeling of time-varying systems,” IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing, vol. 46, no. 10, pp. 1273-1288, October 1999.
- J Roychowdhury, "Reduced-order modeling of linear time-varying systems," in Proc. ICCAD, 1998, pp. 53-61.
- J. Phillips, "Model reduction of time-varying linear systems using approximate multipoint Krylov-subspace projectors," in Proc. ICCAD, Nov. 1998.
- A. Dharchowdhury, R. Panda, D. Blaauw, R. Vaidyanathan, B. Tutuianu, and D. Bearden, “Design and analysis of power distribution networks in PowerPCTM microprocessors,” in Proc. of IEEE DAC, Anaheim, CA, June 15-19, 1998, pp. 738-743.
- Zhe Wang, R. Murgai, R., and J. Roychowdhury, "Automated, accurate macromodelling of digital aggressors for power/ground/substrate noise prediction", Proc DATE 2004, pp 824--829.
- M. v. Heijningen, M. Badaroglu, S. Donnay, M. Engels, and I. Bolsens, “High-level simulation of substrate noise generation including power supply noise coupling,” in Proc. of IEEE DAC, Los Angeles, CA, June 5-9, 2000, pp. 738-743.
- H. H. Chen and D. D. Ling, “Power supply noise analysis methodology for deep-submicron VLSI design,” in Proc. of IEEE DAC, Anaheim, CA, June, 1997, pp. 638-643.
- K. L. Shepard and D. J. Kim, “Static noise analysis for digital integrated circuits in partially-depleted silicon-on-insulator technology,” in Proc. of IEEE DAC, Los Angeles, CA, June 5-9, 2000, pp. 239-242.

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

References

- B. R. Stanisic, N. K. Verghese, and R. A. Rutenbar, “Addressing substrate coupling in mixed-mode ic's: Simulation and power distribution systhesis,” IEEE Journal of Solid-State Circuits, vol. 29, no. 3, pp. 226-238, March 1994.
- R. Gharpurey and R. G. Meyer, “Analysis and simulation of substrate coupling in integrated circuits,” Int. J. Circuit Theory and Applications, vol. 23, pp. 281-394, August 1995.
- R. Gharpurey and R. G. Meyer, “Modeling and analysis of substrate coupling in integrated circuits,” IEEE Journal of Solid-State Circuits, vol. 31, no. 3, pp. 344-353, March 1996.
- N. K. Verghese, D. J. Allstot, and M. A. Wolfe, “Verification techniques for substrate coupling and their application to mixed-signal ic design,” IEEE Journal of Solid-State Circuits, vol. 31, no. 3, pp. 354-365, March 1996.
- D. K. S. et al, “Experimental results and modeling techniques for substrate noise in mixed-signalintegrated circuits,” IEEE Journal of Solid-State Circuits, vol. 28, no. 4, pp. 420-430, April 1993.
- Y. Chen and J. White, “A Quadratic Method for Nonlinear Model Reduction”, Proc. ICMSM 2000, pp 477-40.
- J. Phillips, “Projection Frameworks for Model Reduction of Weakly Nonlinear Systems”, Proc. DAC 2000.
- M. Rewienski and J. White, “A Trajectory Piecewise Linear Approach to Model Reduction and Fast Simulation of Nonlinear Circuits and Micromachined Devices”, Proc ICCAD 2001.

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

References

- P. Li, L. Pilleggi, “NORM: Compact model order reduction of weakly nonlinear systems”, Proc DAC 2003.
- N. Dong, J. Roychowdhury, "Piecewise polynomial nonlinear model reduction" Proc. Design Automation Conference, 2003, pp 484—489.
- K. Kundert, “Predicting the Phase Noise and Jitter of PLL-based frequency synthesizers”, in the Designer's Guide (www.designersguide.com), Nov 2002.
- J.L. Stensby, “Phase-locked Loops: Theory and Applications”, CRC Press, New York, 1997.
- R. Adler, “A Study of Locking Phenomena in Oscillators”, Proc. IRE, Vol 34, pp 351-357, June 1946.
- A. Demir, A. Mehrotra and J. Roychowdhury, “Phase noise in oscillators: a unifying theory and numerical methods for characterization”, IEEE Trans. Ckts Syst I – Fundamental Theory and Applications, vol 47, no 5, pp 655-674, May 2000.
- P. Vanassche, G. Gielen, W. Sansen, “Behavioural modelling of coupled harmonic oscillators”, IEEE Trans. Computer-aided Design, vol 22, pp 1017-1026, Aug 2003.
- M. Gardner, “Phase-Lock Techniques”, Wiley, New York, 1966, 1979.
- A. Hajimiri and T. Lee, “A general theory of phase noise in electrical oscillators”, IEEE J. Solid State Ckts, vol 33, no 2, Feb 1998.
- A. Demir, A. Mehrotra, and J. Roychowdhury, “Phase noise and timing jitter in oscillators”, Proc CICC, May 1998.

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury

References

- J. Roychowdhury, “Automated Macromodel Generation for Electronic Systems”, IEEE Behavioral Modeling and Simulation Workshop, Oct 2003.
- A. Demir and J. Roychowdhury, “A Reliable and Efficient Procedure for Oscillator PPV Computation, with Phase Noise Macromodelling Applications”, IEEE Transactions on Computer-Aided Design, Feb 2003.
- Z. Wang and J. Roychowdhury, “Macromodelling of Digital Libraries for Substrate Noise Analysis”, IEEE International Symposium on Circuits and Systems, May 2004.

.DAC 2004 Macromodelling Tutorial. Copyright © J Roychowdhury