Dynamic Noise Analysis: Definitions, Models and Tool

GSRA: Li Ding

Presented by: Professor Pinaki Mazumder

Department of Electrical Engineering and Computer Science
The University of Michigan, Ann Arbor MI 48109, USA
Email: mazum@eecs.umich.edu

Compiled from talks presented by Pinaki Mazumder
at DAC 2004, ECCTD 2003, ISCAS 2002, ICCAD 2002,
Sequence Design Automation, & Sun Microsystems

The Talk summarizes the following papers presented by Professor Pinaki Mazumder at the following Conferences:

- L. Ding and P. Mazumder, "Noise-Tolerant Quantum MOS Circuits Using Resonant Tunneling Devices," ***Proceedings of the European Circuit Conference: Theory and Design***, Krakow, Poland, 2003.
- L. Ding and P. Mazumder, "Modeling Cell Noise Transfer Characteristic for Dynamic Noise Analysis," ***Proceedings on IEEE Design Automation and Testing Conference in Europe (DATE)***, May 2003.
- L. Ding and P. Mazumder, "Dynamic Noise Margin: Definitions and Model***," Proceedings on IEEE International Conference on VLSI Design***, pp. 1001-1006, Jan.2004.
- L. Ding and P. Mazumder, "A Novel Technique to Improve Noise Tolerance of Dynamic Logic Circuits," ***Proceedings on IEEE/ACM Design Automation Conference,*** San Diego, June 2004.

Outline

- Overview of static noise margin
- Dynamic noise margin definitions
- Dynamic noise margin model
- DNM based noise analysis method

Possible Effects of Noise

Functionality Failure
- **Logic Level Change: Depends on Circuit Styles**
- **False State Latching**
- **Latch States Switching**

Timing Violation
- **Reduce Delay: Race-through, Double Clocking**
- **Increase Delay: Latch False Setup**

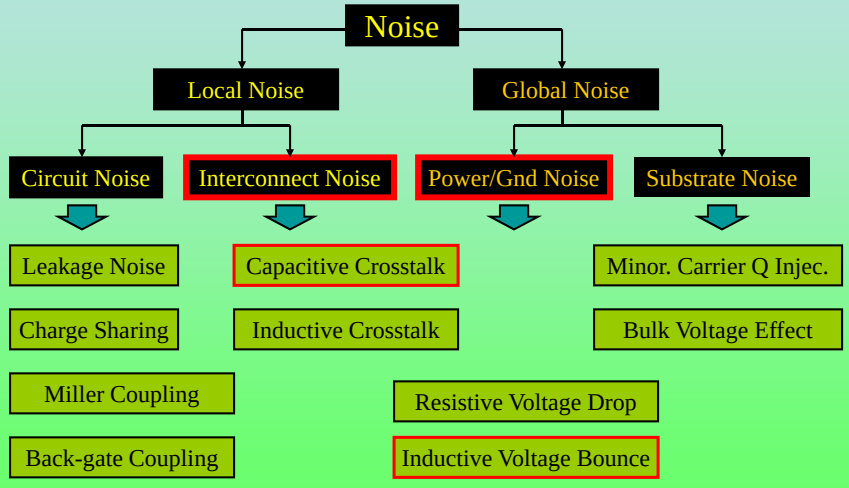
Noise in Digital Systems

- **What:** any deviation from expected state/value
- **Why:** noise margin is continuously decreasing
- **Noise margin:** max noise a circuit can survive

Year	1999	2002	2005	2008	2011	2014
Tech (um)	180	130	100	70	50	35
Density (M)	20	54	133	328	811	2000
VDD (V)	1.5-1.8	1.2-1.5	0.9-1.2	0.6-0.9	0.5-0.6	0.3-0.6

* Lower threshold voltage
* Denser interconnect
* Faster switching time
* More dynamic circuits

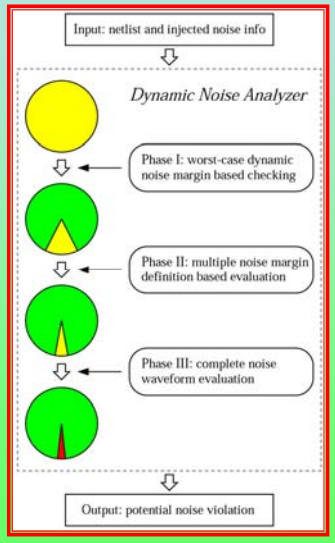
Noise Classification



Speedup Common Tasks

– Three phases in DNA

- **Majority nodes in large VLSI chips do not have excess noise**

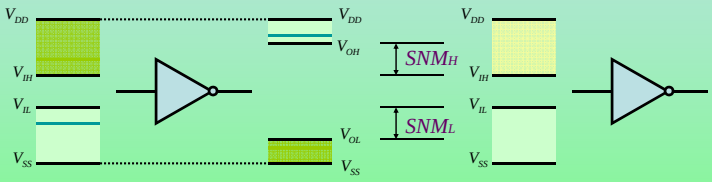


Extremely fast
Pessimistic

Very fast
Slightly pessimistic

Slow
Accurate

Overview of Static Noise Margin

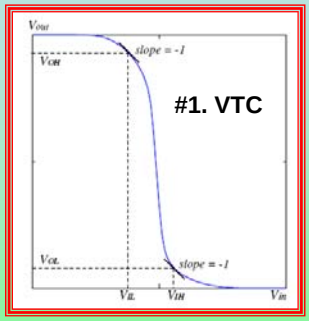


$$V_{in} \le V_{IL} \Rightarrow V_{out} \ge V_{OH}$$
$$V_{in} \ge V_{IH} \Rightarrow V_{out} \le V_{OL}$$

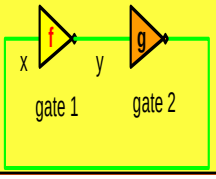
$$SNM = \min\{SNM_H, SNM_L\}$$
$$SNM_H = V_{OH} - V_{IH}$$
$$SNM_L = V_{IL} - V_{OL}$$

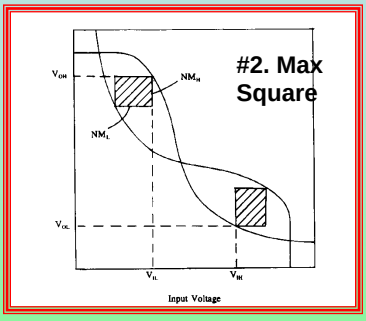
7

Static Noise Margin Criteria



Given by -1 slope points
Stable logic states of an Infinite chain of inverters





Coincidence of roots of flip-flop equation

$$x - g(f(x)) = 0$$

Small-signal closed loop gain

#3. $$\frac{\partial f}{\partial x} \cdot \frac{\partial g}{\partial y} = 1$$

Maximum square between normal and mirrored VTC

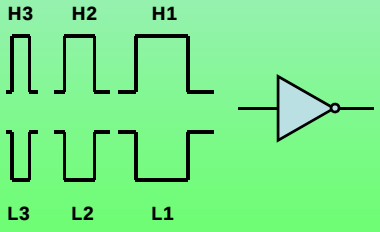
NMH·NML is Maximum (maximum product criteria)

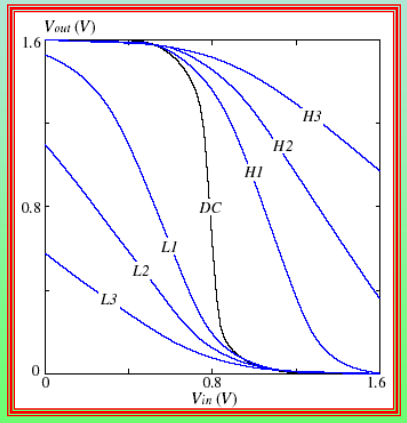
$$F_1 \equiv x - g(y) = 0$$
$$F_2 \equiv y - f(x) = 0$$
$$J = \begin{vmatrix} \frac{\partial F_1}{\partial x} & \frac{\partial F_1}{\partial y} \\ \frac{\partial F_2}{\partial x} & \frac{\partial F_2}{\partial y} \end{vmatrix} = 0$$

#4. Jacobian of Kirchhoff equation is zero

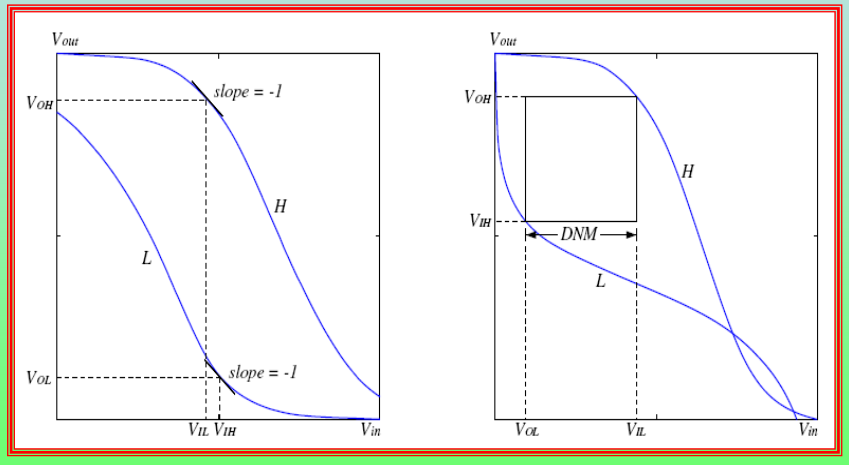
DC and AC Voltage Transfer Characteristics (VTC's)

Experiment:
scan amplitude (A) for a given width (W)





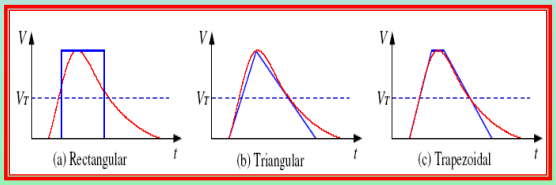
Dynamic Noise Definition



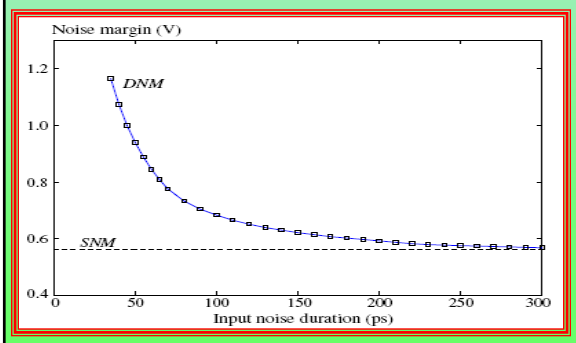
-1 Slope based Definition

Maximum Square based Definition

Dynamic Noise



Noise waveforms: rectangular, triangular and trapezoidal



DNM ≈ SNM, when input noise duration increases

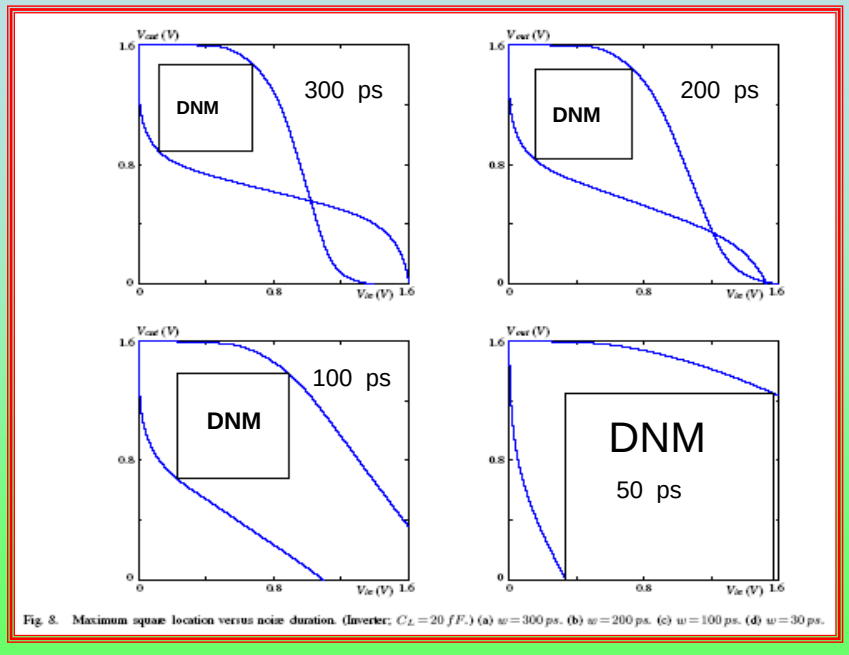
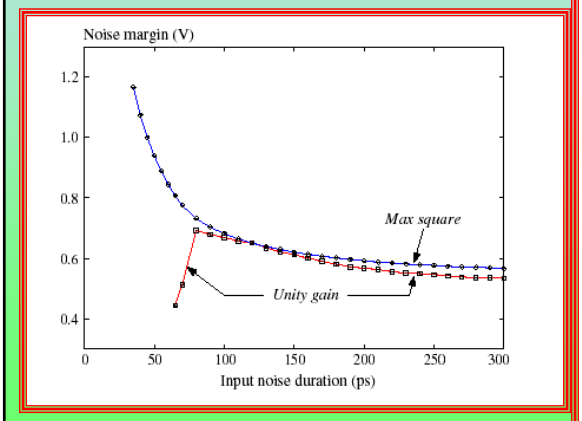
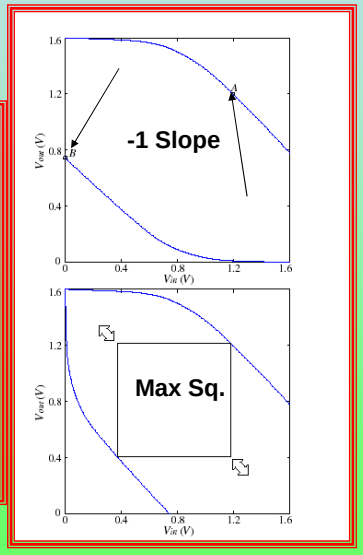


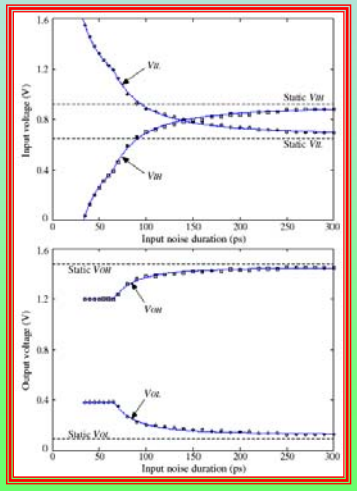
Fig. 8. Maximum square location versus noise duration. (Inverter; $C_L = 20\,fF$.) (a) $w = 300\,ps$. (b) $w = 200\,ps$. (c) $w = 100\,ps$. (d) $w = 30\,ps$.

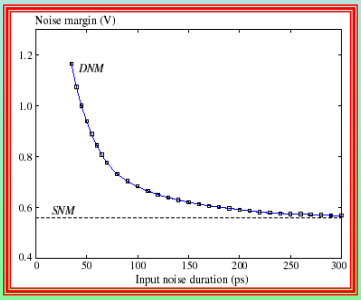
Comparison between Max Square v. -1 Slope Dynamic Noise Margins





Dynamic Noise Margin - Max. Square





- **Using Maximum Square Method to find V_{IH}, V_{IL}, V_{OH}, V_{OL} for a given pulse width W**
- **NM= V_{OH}-V_{IH}=V_{IL}-V_{OL}**

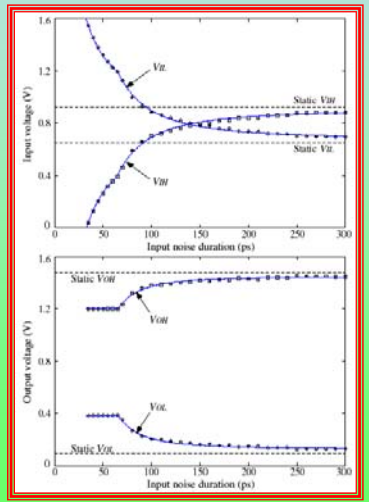
Why ρ in the Modeling

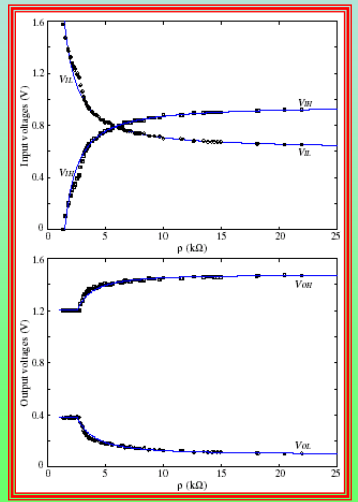
- **It is unfair to only consider the width in the DNM modeling since the time constant plays an important role for transient response of a gate to a noise.**
- **Define a new parameter**

$\rho = W/(C_L+C_o)$ $\quad V_{IH}=a_o+a_1\rho^{-a_2}$

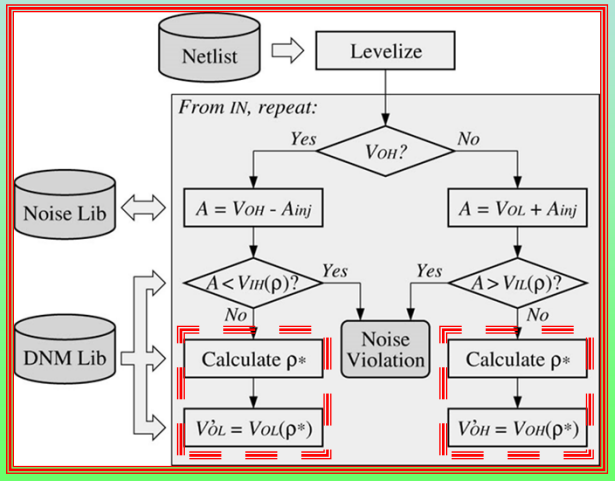
$$
\begin{aligned}
V_{IH}(\rho) &= \max\{a_0^{IH} + a_1^{IH} \cdot \rho^{-a_2^{IH}}, 0\}, \\
V_{IL}(\rho) &= \min\{a_0^{IL} + a_1^{IL} \cdot \rho^{-a_2^{IL}}, V_{DD}\}, \\
V_{OH}(\rho) &= \max\{a_0^{OH} + a_1^{OH} \cdot \rho^{-a_2^{OH}}, V_{OH}(\rho\infty)\}, \\
V_{OL}(\rho) &= \min\{a_0^{OL} + a_1^{OL} \cdot \rho^{-a_2^{OL}}, V_{OL}(\rho\infty)\},
\end{aligned}
$$

Comparison b/w Modeling & Data





Flowchart

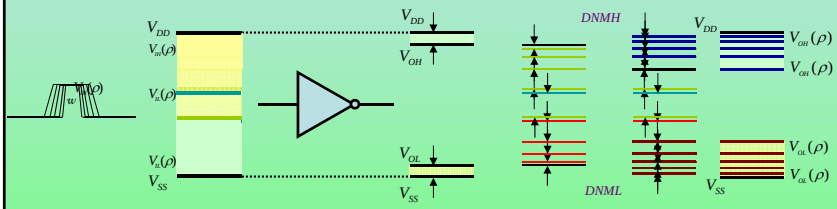


Reasons for Simplified Model

The DNM based analysis method is not efficient enough to be used in full-chip noise analysis due to the following reasons:

- **Netlist Levelization**
- **The computational effort (calculating ρ*) that is needed to reduce the pessimism of predicting high propagation noise when noise duration is small.**
- **The modeling effort of the propagated noise on the amplitude and duration of the inject noise.**

Simplified dynamic noise margin definition

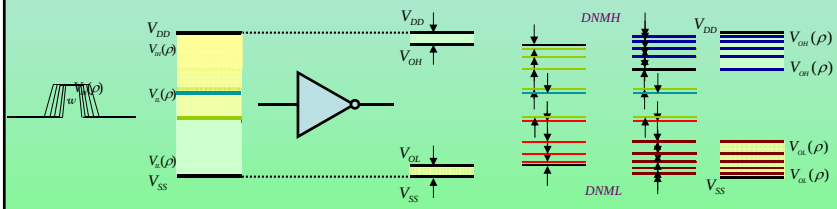


$$DNM_H(\rho) = V_{OH} - V_{IH}(\rho) \quad DNM_L(\rho) = V_{IL}(\rho) - V_{OL}$$

$$DNM(\rho) = \min\{DNM_H(\rho) - DNM_L(\rho)\}$$

V_{OH} = Constant V_{OL} = Constant

Simplified dynamic noise margin definition

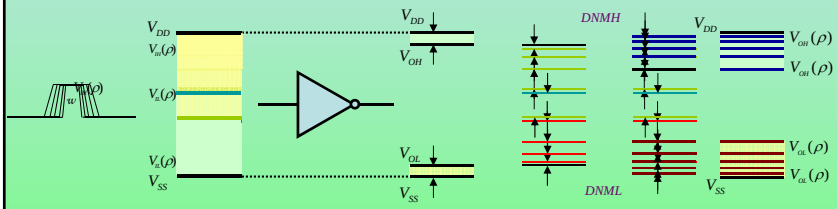


$$DNM_H(\rho) = V_{OH} - V_{IH}(\rho) \quad DNM_L(\rho) = V_{IL}(\rho) - V_{OL}$$

$$DNM(\rho) = \min\{DNM_H(\rho) - DNM_L(\rho)\}$$

V_{OH} = Constant V_{OL} = Constant

Simplified dynamic noise margin definition



$$DNM_H(\rho) = V_{OH} - V_{IH}(\rho) \quad DNM_L(\rho) = V_{IL}(\rho) - V_{OL}$$

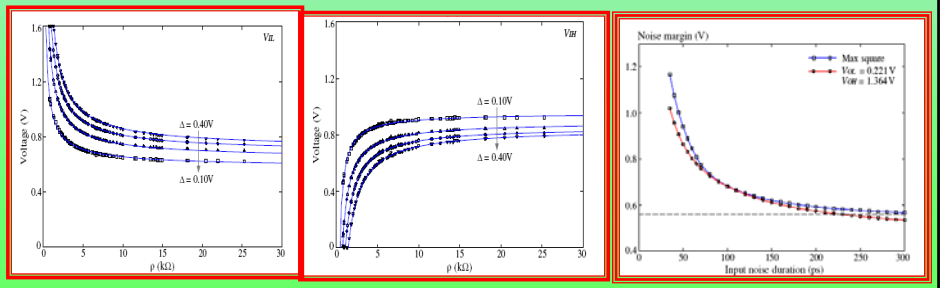
$$DNM(\rho) = \min\{DNM_H(\rho) - DNM_L(\rho)\}$$

V_{OH} = Constant V_{OL} = Constant

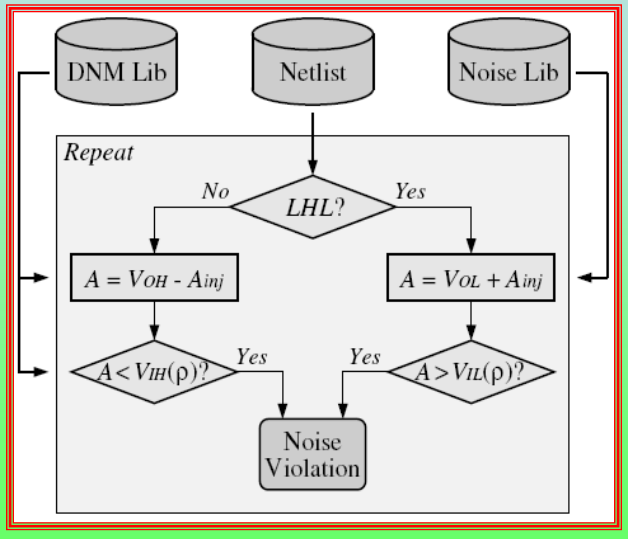
Modeling simplified dynamic noise margins

- The analytical model for simplified dynamic noise margin is similar to the non-simplified analytical model with different parameters.

$$V_{IL}(\rho) = \min\{a_0^{IL} + a_1^{IL} \cdot \rho^{-a_2^{IL}}, V_{DD}\} \quad V_{IH}(\rho) = \max\{a_0^{IH} + a_1^{IH} \cdot \rho^{-a_2^{IH}}, 0\}$$



Flowchart



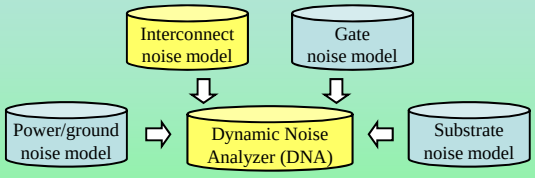
List of Current Tools

- **In Industry**
  - **SubstrateStorm (Layin): Simplex Solutions**
  - **GateScope: Magma Design Automation**
  - **Nova: IBM**
  - **Eldo/Eldo RF: Mentor Graphics**
  - **CeltIC: Cadence**
  - **Swan: IMEC, Belgium**
  - **Substrate Noise Analyst: Cadence**
  - **Pacific State Noise Analyzer: Cadence**
  - **Physical Studio: Sequence Design**

Motivation for New Tools

- **Noise becomes a grave concern**
  - Various noise sources: wire, gate, power/ground & substrate
  - Technology shrinking and voltage scaling
  - Increased chip operating frequency
- **Available tools are inadequate**
  - Static noise analysis is pessimistic
  - Worst-case analysis is pessimistic
  - Full waveform evaluation is time consuming

Proposed Framework



DNA has three phases:
1. Worst-case noise margin based method
2. Multiple noise margins based approach
3. Complete dynamic noise evaluation

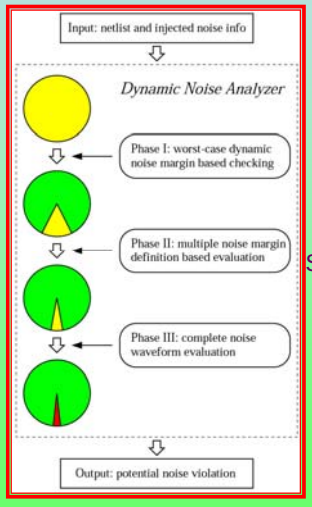
To be developed under this project

To be obtained from Third Party

Speedup Common Tasks

– Three phases in DNA

- Majority nodes in large VLSI chip do not have excess noise

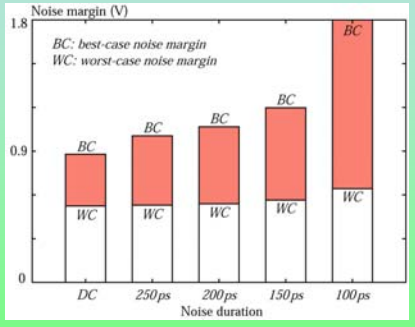


Extremely fast
Pessimistic

Very fast
Slightly pessimistic

Slow
Accurate

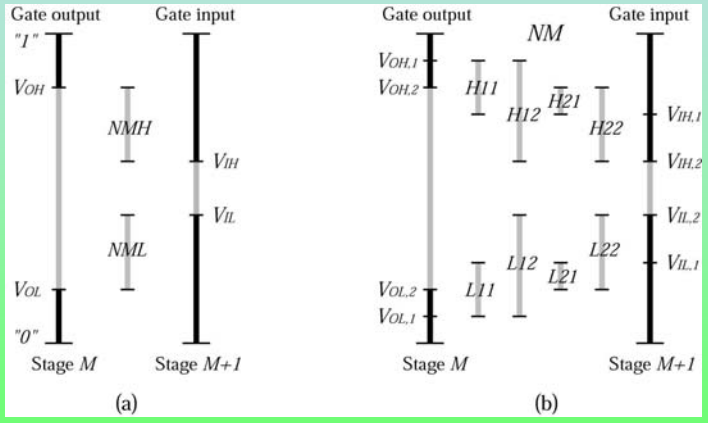
Phase II: Multiple Noise Margin Based



- WC: assume same amount of noise at all nodes
- WC noise margin can be overly pessimistic
- BC: assume noise only impinges on one node

Phase II: cont'd

Towards multiple noise margins



Conventional noise margin definition

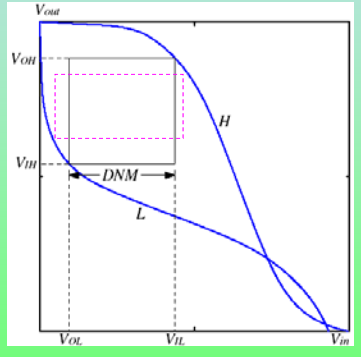
Multiple noise margin (N=2)

Multiple dynamic noise margins

- **Single DNM Definition**
  Maximum square for equal DNM_H and DNM_L
  (Black square)
- **Multiple DNMs Definition**

$$\max(V_{in}(t)) \le V_{IL}(w,k) \Rightarrow \min(V_{out}(t) \ge V_{OH}(w,k))$$
$$\max(V_{in}(t)) \ge V_{IH}(w,k) \Rightarrow \min(V_{out}(t) \le V_{OL}(w,k))$$

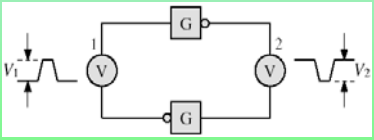
Draw another rectangle for another dynamic noise margin (Dashed rectangle)



Gate AC Transfer Curve

Noise Coverage Probability

- **Noise on infinity inverter chain = Noise on flip-flop**
- **Noise covered: those set of noise signals do not exceed dynamic noise margin of flip-flop**
- **Noise coverage probability:**

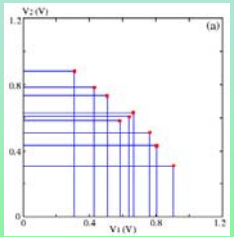


$$P_{cover} = P(\Delta V_1 \le DNM_L) \times P(\Delta V_2 \le DNM_H)$$

Noise coverage probability for single DNM and multiple DNMs

- **Noise on V1 and V2 covered by dynamic noise margin metric**

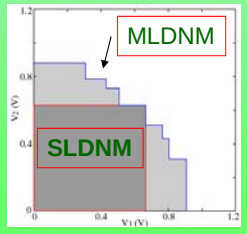
  Single DNM .vs. Multiple DNMs
  (Shaded square .vs. polygon)



- **Noise coverage probability**

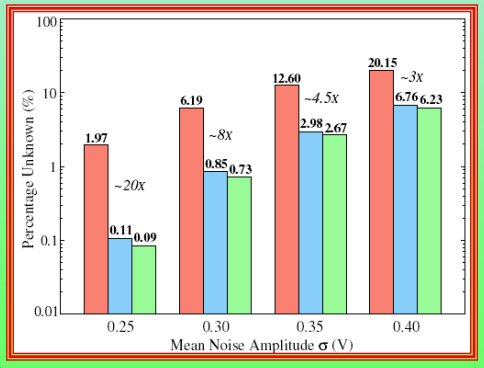
  Single DNM
  P(Noise on V1 and V2 lying in the shaped square)

  Multiple DNMs
  P(Noise on V1 and V2 lying in the polygon)

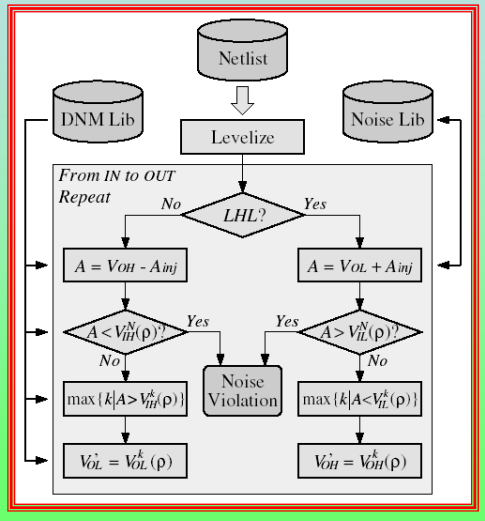


Experimental Results

- Noise amplitudes follow non-negative Guassian distribution
- The figure shows the comparison of noise coverage using single and multiple DNM definition



Multiple DNM flowchart



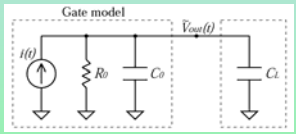
Phase III: Complete Noise Waveform Evaluation

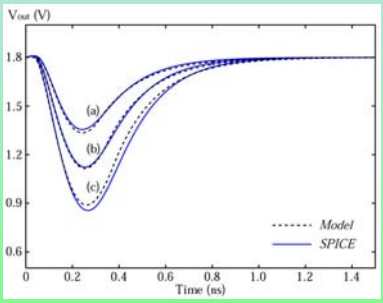
Majors tasks:

- Circuit modeling for various logic styles
- Noise waveform modeling
- Gate noise transfer characteristic modeling
- Impact of dynamic noise on sequential circuits

Phase III: cont'd

- Circuit model (dynamic gate)





- Waveform model

$$V_{in}(t) = a_0 + \sum_{i=1}^{M} a_i e^{-p_i t}$$

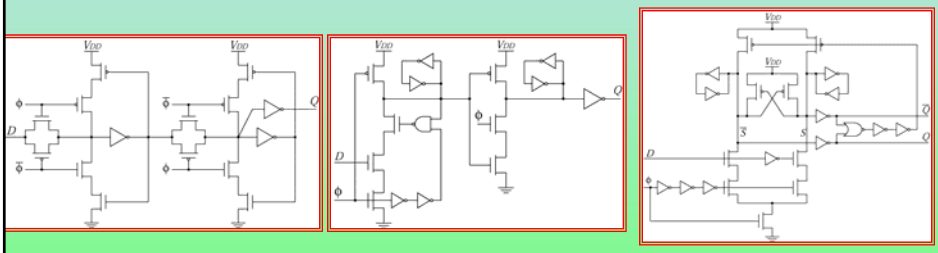
- Noise transfer characteristic model

$$V_{out}(t) = \begin{cases} 0 & t \le t_1 \\ kR_0\left(\Phi(t) - \Phi(t_1)e^{-(t-t_1)/\tau}\right) & t_1 < t \le t_2 \\ V_{out}(t_2)e^{-(t-t_2)/\tau} & t > t_2 \end{cases}$$

$$\Phi(t) = (a_0 - V_T) + \sum_{i=1}^{M} \frac{a_i}{1 - p_i\tau} e^{-p_i t}$$

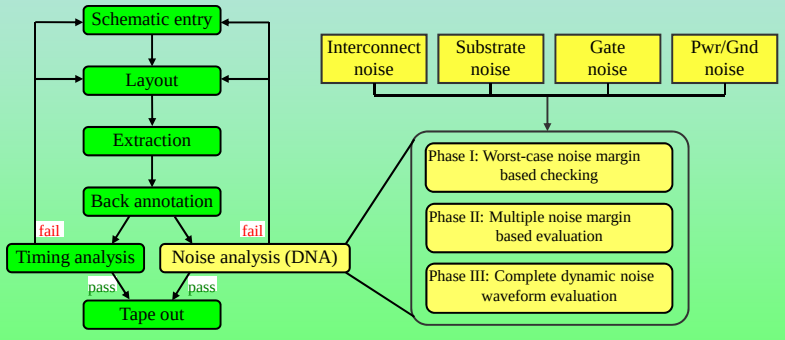
Phase III: cont'd

Sequential circuits

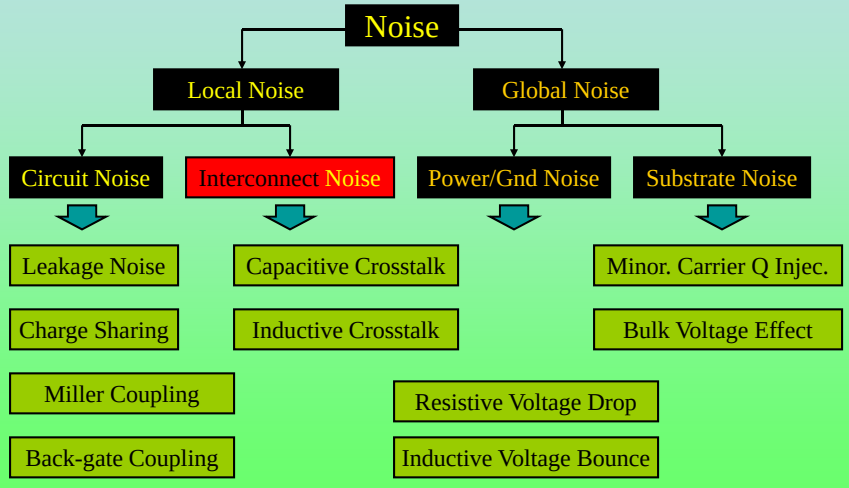


- Master-slave flip-flop
  PowerPC603
- Semidynamic flip-flop
  Sun Microsystems
- K6 edge-triggered latch
  AMD

DNA In Typical VLSI Chip Design Flow

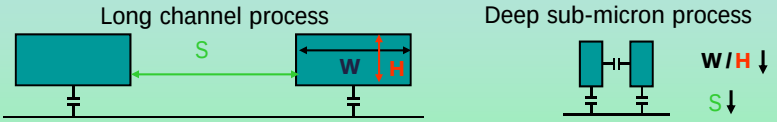


Noise Classification

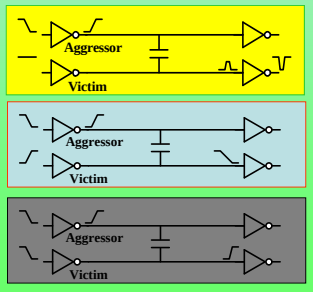


Crosstalk Noise Estimation Using Effective Coupling Capacitance

Crosstalk Noise Issues



Problem: signal nets are no longer independent

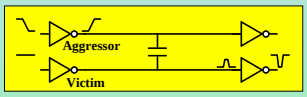


Voltage glitch – signal integrity

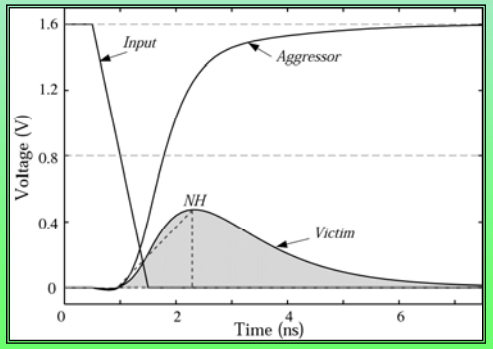
More delay – slow system clock

Less delay – race problem

Typical Crosstalk Waveform



Signal integrity problem



Main Issues:

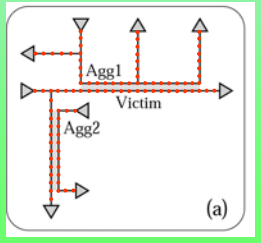
- Maximum noise voltage
- Noise pulse width
- Noise pulse area
- Noise peak time

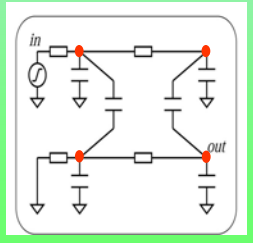
Crosstalk Noise Modeling

Problem: SPICE simulation is computationally expensive

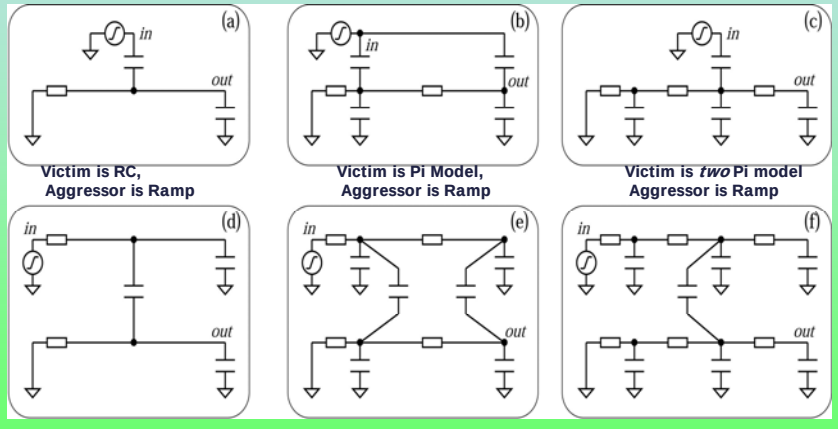
Solutions:

- ✓ Reduced order modeling (PRIMA) – post-layout simulation after back annotation (slower, but accurate)
- ✓ Template circuit based – layout synthesis (faster, approximate)





Six Common Template Circuits



Victim is RC, Aggressor is Ramp

Victim is Pi Model, Aggressor is Ramp

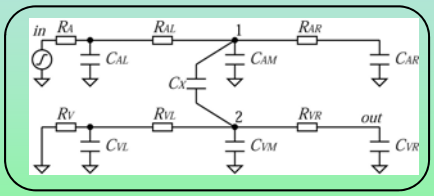
Victim is *two* Pi model Aggressor is Ramp

Victim is RC, Aggressor is RC

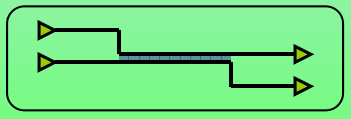
Victim is Pi Model Aggressor is Pi Model

Victim is *two* Pi Model Aggressor is *two* Pi Model

Proposed Template Circuit



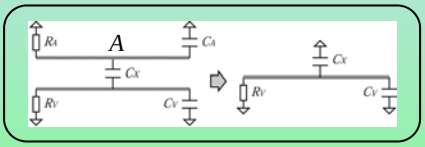
Victim is *two* Pi Model
Aggressor is *two* Pi Model



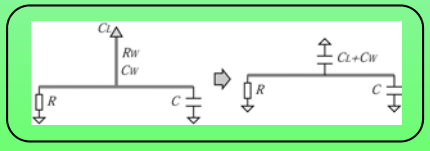
- Treat aggressor and victim equally
- Models the location of coupling
- Easy to extract model parameter

Previous Methods for Multiple Nets

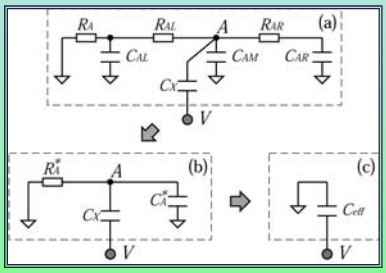
* Multiple aggressors
  * Aggressors are replaced by coupling capacitance

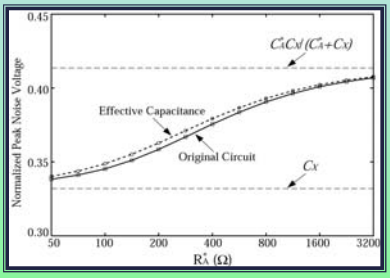


* Nets are trees
  * Branches are replaced by total load capacitance



Aggressor Net Reduction



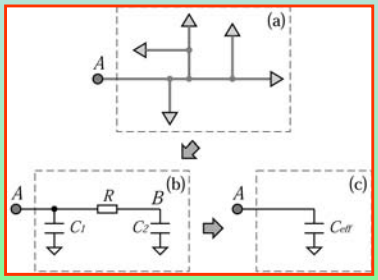


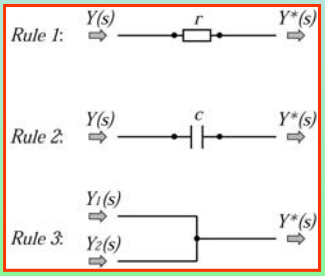
$$R_A^* = R_A + R_{AL},\quad C_A^* = \frac{R_A^2}{(R_A + R_{AL})^2} C_{AL} + C_{AM} + C_{AR}$$

Effective capacitance:

$$C_{eff} = \left(1 - \frac{R_A^* C_X}{t_r}\left(1 - e^{-t_r/(R_A^*(C_A^* + C_X))}\right)\right) \cdot C_X$$

Tree Branch Reduction





$$y_1^* = y_1$$
$$y_2^* = y_2 - r y_1^2$$
$$y_3^* = y_3 - 2r y_1 y_2 + r^2 y_1^3$$

$$y_1^* = c$$
$$y_2^* = -c^2 / y_0$$
$$y_3^* = c^2 (y_1 + c) / y_0^2$$

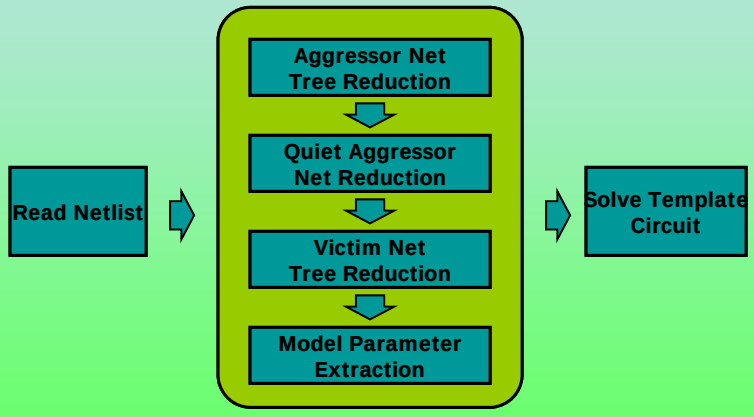
$$y_1^* = y_{1,1} + y_{2,1}$$
$$y_2^* = y_{1,2} + y_{2,2}$$
$$y_3^* = y_{1,3} + y_{2,3}$$

$$C_1 = y_1 - \frac{y_2^2}{y_3},\ C_2 = \frac{y_2^2}{y_3},\ R = -\frac{y_3^2}{y_2^3}.$$

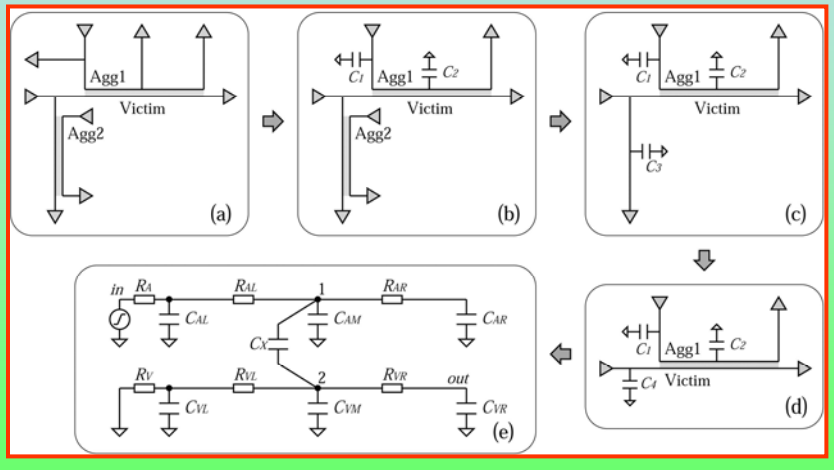
Effective capacitance:

$$C_{eff} = C_1 + \left(1 - \frac{RC_2}{t_r}\left(1 - e^{-t_r/RC_2}\right)\right) \cdot C_2$$

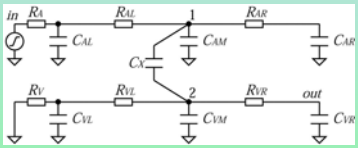
Netlist Reduction Flowchart



Netlist Reduction Example



Template Circuit Modeling



Transfer function:

$$H(s)=\frac{a_5 s^5+\cdots+a_1 s+a_0}{b_6 s^6+b_5 s^5+\cdots+b_1 s+1}$$

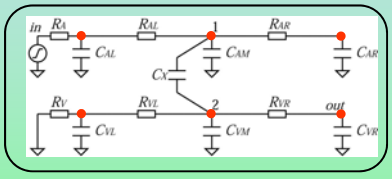
$$H(s)=\sum_{i=1}^{6}\frac{r_i}{s-p_i}$$

Victim waveform:

$$V_{out}(t)=\begin{cases}\sum_{i=1}^{6}\left(-\frac{r_i(1+p_i t)}{t_r p_i^2}+\frac{r_i e^{p_i t}}{t_r p_i^2}\right) & t\le t_r\\ \sum_{i=1}^{6}\left(-\frac{r_i e^{p_i (t-t_r)}}{t_r p_i^2}+\frac{r_i e^{p_i t}}{t_r p_i^2}-\frac{r_i}{p_i}\right) & t>t_r\end{cases}$$

- No analytical expression for p_i and r_i
- Even more difficult for noise peak, noise width, etc.

Dominant-Pole Approximation (1P)



Transfer function:

$$H(s)=\frac{a_5 s^5+\cdots+a_1 s}{b_6 s^6+b_5 s^5+\cdots+b_1 s+1}$$

Dominant-Pole approximation:

$$H_1(s)\approx\frac{a_1 s}{b_1 s+1}$$

Victim waveform:

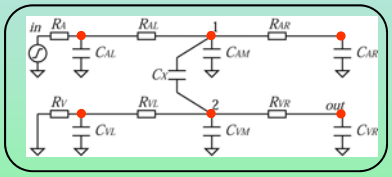
$$V_{out}(t)=\begin{cases}\frac{t_X}{t_r}\left(1-e^{-t/t_T}\right) & t\le t_r\\ \frac{t_X}{t_r}\left(e^{-(t-t_r)/t_T}-e^{-t/t_T}\right) & t>t_r\end{cases}$$

$$t_X=C_X(R_V+R_{VL})$$

$$t_V=C_{VL}R_V+(C_{VM}+C_X)(R_V+R_{VL})+C_{AL}R_A+(C_{AM}+C_X)(R_A+R_{AL})+C_{AR}(R_A+R_{AL}+R_{AR})+C_{VR}(R_V+R_{VL}+R_{VR})$$

- Noise peak and width can be analytically obtained
- Predicts noise peaks at time t_r
- Derivative of noise waveform not continuous
- Not very accurate

Double-Pole Approximation (2P)



Victim 1P approximation:

$$H(s)=\frac{a_2s^2+a_1s+a_0}{b_3s^3+b_2s^2+b_1s+1}$$

$$H_1(s)\approx\frac{a_1s}{b_1s+1}$$

Aggressor time constant:

$$t_A=C_{AL}R_A+(C_{AM}+C_{X,eff}+C_{AR,eff})(R_A+R_{AL})$$

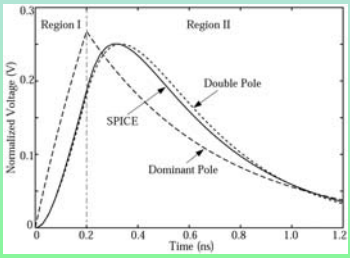
Victim waveform:

$$V_{out}(t)=\begin{cases}\dfrac{t_X}{t_r}\left(1+\dfrac{t_A}{t_V-t_A}e^{-t/t_A}-\dfrac{t_V}{t_V-t_A}e^{-t/t_V}\right) & t\le t_r\\ \dfrac{t_X}{t_r}\left(\dfrac{t_A}{t_V-t_A}(e^{-t/t_A}-e^{-(t-t_r)/t_A})-\dfrac{t_V}{t_V-t_A}(e^{-t/t_V}-e^{-(t-t_r)/t_V})\right) & t>t_r\end{cases}$$

$$t_X=C_X(R_V+R_{VL})$$

$$t_V=C_{VL}R_V+(C_{VM}+C_X)(R_V+R_{VL})+C_{VR}(R_V+R_{VL}+R_{VR})$$

Comparison of models



- 1P model predicts wrong noise peak time
- 1P model waveform is not smooth
- 2P model matches SPICE very well

$$t_{peak}=t_r+\frac{t_Vt_A}{t_V-t_A}\ln\left(\frac{1-e^{-t_r/t_A}}{1-e^{-t_r/t_V}}\right)$$

Maximum noise voltage (based on 5000 random circuits):

Method	Dominant pole	Double pole
Average error	8.3%	2.3%
Cases < 5% error	37.3%	92.6%
Cases < 10% error	66.8%	99.9%
3 sigma error	26%	8%

Experiments on Industrial Circuits

- 30 noise-prone industrial circuits in 0.15 micron tech
- Average number of aggressors: 5
- Average number of RC elements: 128
- Average victim wire length: 2.1 mm

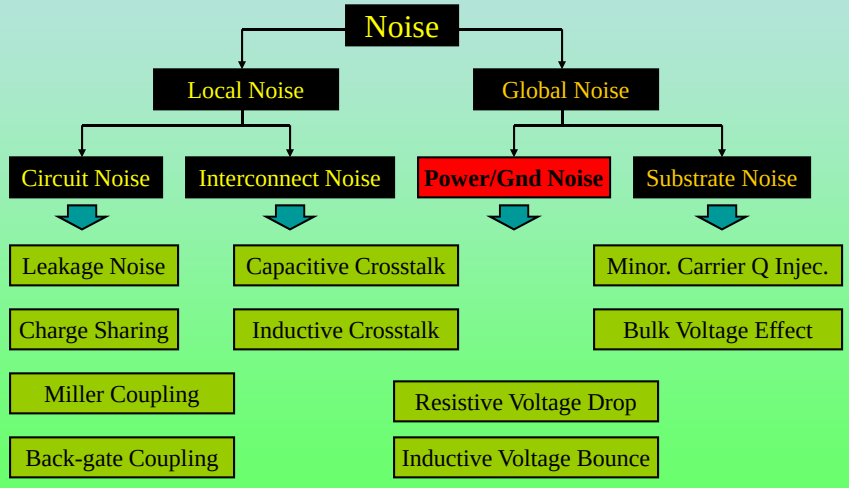
Results of the Proposed Method (2-Pole, 6-nodes template)

- Average peak noise error: Reduced from 11.7% (aggressors nets and branches are reduced by previous naïve methods) to 2.7%
- Maximum peak noise error: Reduced from 21.3% (previous naïve methods) to 7.8%

Conclusions

- ✓ Method for quiet aggressor nets reduction
- ✓ Method for tree reduction
- ✓ Double-pole model for template circuit
- ✓ Efficient crosstalk noise estimation framework

Noise Classification



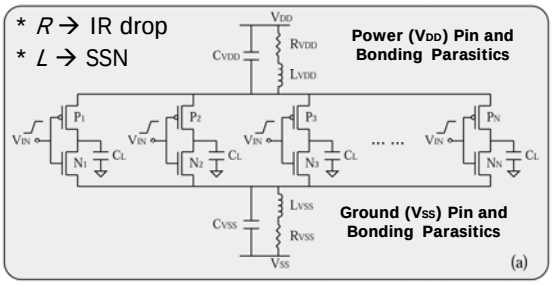
Outline

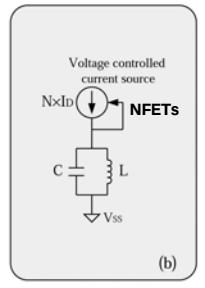
- ✓ Motivation of the research
- ✓ Equivalent circuit for SNN modeling
- ✓ ASDM for MOS devices
- ✓ SSN analysis and formulation
- ✓ Effect of parasitic capacitance

Motivations

- ✓ SSN is caused by parasitic inductance at power/ground network
- ✓ SSN is a serious problem in VDSM VLSI chips
  1. Generate glitches on the power/ground wires
  2. Increase delay
  3. Cause output signal distortion
  4. Reduce overall margin of a system
- ✓ Simple formulation is desired for SSN estimation
- ✓ Previous formulations are not adequate

Equivalent Circuit for SSN Modeling



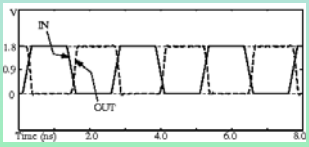


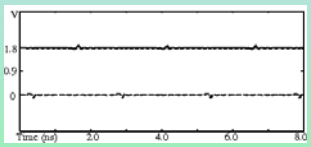
Ground (V_{SS}) Pin and Bonding Parasitics

1. Assume there are *N* identical drivers
2. Drivers switch simultaneously
3. Drivers switch at the same direction
4. Large capacitance load at the drivers
5. Since CL is large and NFETs are in Saturation, they can be replaced by Current Sources

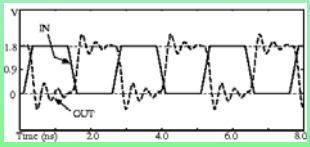
Typical Waveforms

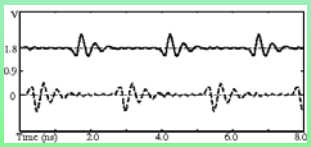
L=1.0nH
1 driver



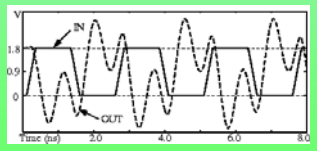


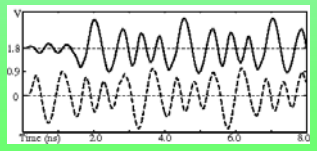
L=2.0nH
5 drivers





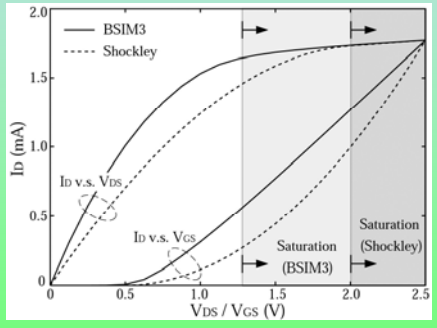
L=5.0nH
10 drivers





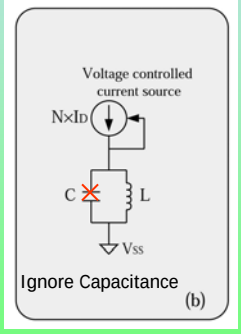
MOSFET Modeling

Long channel MOSFET model (Shockley's Model):



Short channel MOSFET model
(Sakurai & Newton's α-Power Law Model)

SSN Analysis – Inductance Only



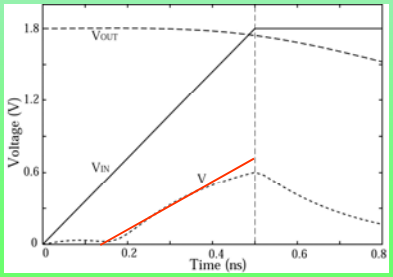
α-Power Law Model:

$$V = NL\frac{dI_D}{dt}$$
$$I_D = K(V_{IN} - V - V_{TH})^{\alpha}$$

* No analytical solution!
* How do we proceed?

Previous Works

- ✓ Sethinathan and Prince, JSSC91 (long channel model, linear)
- ✓ Vemuru, TPCK96 (α-power model, Taylor)
- ✓ Jou *et al*., CICC98 (α-power model, Taylor2)
- ✓ Song, Ismail, *et al*., ISCAS99 (α-power model, linear+Taylor)

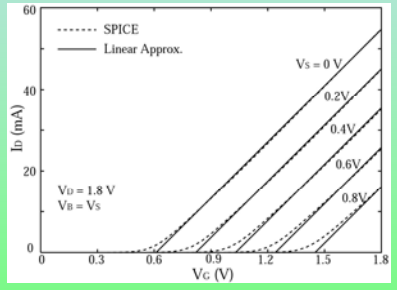


$$V = NL\frac{dI_D}{dt}$$
$$I_D = K(V_{IN} - V - V_{TH})^{\alpha}$$

* Why approximation?
* Why α-power law model?

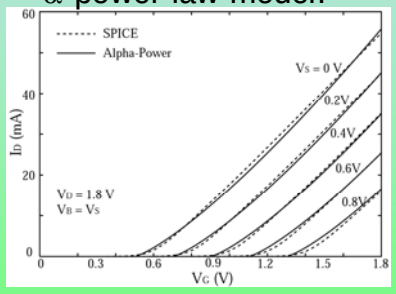
Application-Specific Device Modeling

Linear model:



$$I_D = K(V_G - \gamma \cdot V_S - V_0)$$

α-power law model:



$$I_D = K(V_G - V_S - V_{TH})^{\alpha}$$

Extracted Model Parameters

Process*	V_{DD} (V)	Type	K (mA/V)	V_0 (V)	γ
0.18um	1.8	NFET	0.46	0.61	1.06
		PFET	0.26	0.79	1.12
0.25um	2.5	NFET	0.30	0.72	1.08
		PFET	0.22	0.92	1.08
0.35um	3.3	NFET	0.21	0.89	1.07
		PFET	0.18	1.08	1.08

** TSMC processes, available through MOSIS*

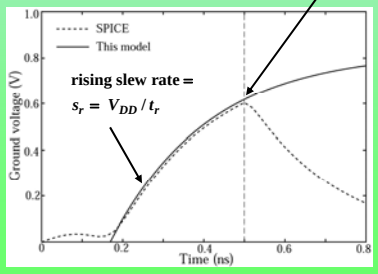
Simple SSN Formulation

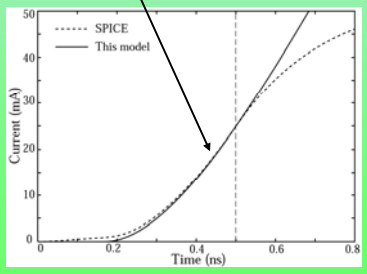
$$V = L\frac{dI_D}{dt}$$
$$I_D = NK(V_G - \gamma \cdot V_S - V_0)$$

$$\frac{dV}{dt} + \frac{V}{NLK\gamma} = \frac{s_r}{\gamma}$$

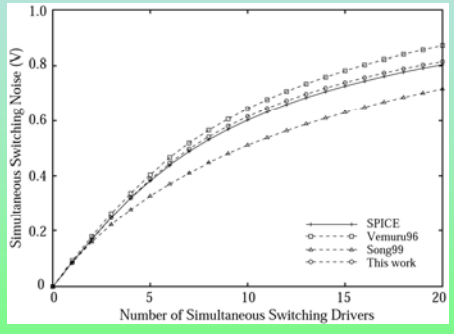
Solution to 1st order ODE:

$$V(t) = NLs_rK\left(1 - e^{-\frac{t-t_0}{\gamma NLK}}\right)$$
$$V_m = NLs_rK\left(1 - e^{-\frac{V_{DD}-V_0}{\gamma NLs_rK}}\right)$$
$$I_D(t) = K\left(s_r t - V_0 - \gamma NLs_rK\left(1 - e^{-\frac{t-t_0}{\gamma NLK}}\right)\right)$$





Comparison with Previous Works



Prior works ⇨ Modeling Error ✚ Eqn. Deriv. Error

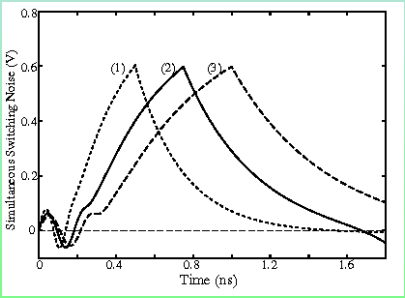
This work ⇨ Modeling Error

A Figure of Merit – *H*

$$V_m = NLs_r K\left(1 - e^{-\frac{V_{DD}-V_0}{\gamma NLs_r K}}\right)$$

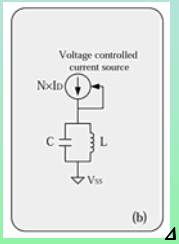
$$H = NLs_r$$

$$V_m = HK\left(1 - e^{-\frac{V_{DD}-V_0}{\gamma HK}}\right)$$



- ✓ *H* is the only *circuit-related* parameter
- ✓ *H* depends *equally* on three variables: *N*, *L*, and s_r

SSN Modeling with Capacitance



$$V = L\frac{dI_L}{dt}$$

$$I_L = NK(V_G - \gamma \cdot V_S - V_0) - C\frac{dV}{dt}$$

$$C\frac{d^2V}{dt^2} + NK\gamma\frac{dV}{dt} + \frac{1}{L}V = NKs_r$$

$\Delta = N^2K^2\gamma^2 - 4C/L \Leftrightarrow$ As $N \uparrow \Rightarrow$ underdamp $\rightarrow$ critical damp $\rightarrow$ overdamp

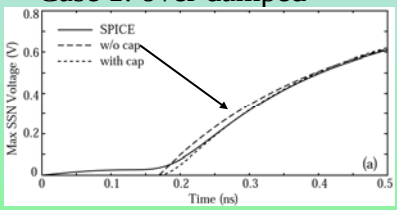
For a Fixed Technology as No. Of Drivers Increases

$N \uparrow$

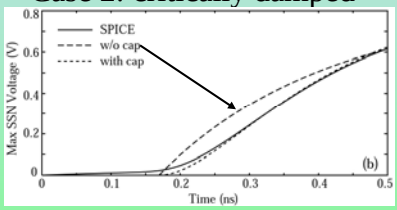
Case	Condition	Description	Maximum SSN voltage formula
1	$\Delta > 0$	over damped	$NLKs_r\left(1 - \frac{\lambda_2}{\lambda_2-\lambda_1}e^{-\lambda_1 t_{r,0}} + \frac{\lambda_1}{\lambda_2-\lambda_1}e^{-\lambda_2 t_{r,0}}\right)$,
2	$\Delta = 0$	critically damped	$NLKs_r\left(1 - (1+\lambda t_{r,0})e^{-\lambda t_{r,0}}\right)$,
3a	$\Delta < 0, t_{r,0} \le \pi/\omega$	under damped	$NLKs_r\left(1 - e^{-\lambda t_{r,0}}\left(cos\omega t_{r,0} + \frac{\lambda}{\omega}sin\omega t_{r,0}\right)\right)$, High speed
3b	$\Delta < 0, t_{r,0} > \pi/\omega$	under damped	$NLKs_r\left(1 + e^{-\frac{\lambda}{\omega}\pi}\right)$, Low speed
where $t_{r,0} = t_r(1 - V_0/V_{DD})$, $\lambda = NK\gamma/(2C)$, $\omega = \sqrt{4C/L - N^2K^2\gamma^2}/(2C)$, $\lambda_1 = (NK\gamma - \sqrt{N^2K^2\gamma^2 - 4C/L})/(2C)$, $\lambda_2 = (NK\gamma + \sqrt{N^2K^2\gamma^2 - 4C/L})/(2C)$.			

Transient Waveforms

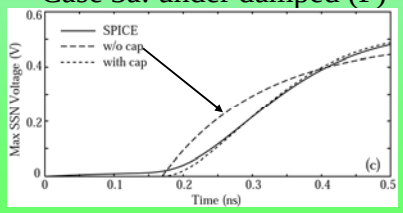
Case 1: over damped



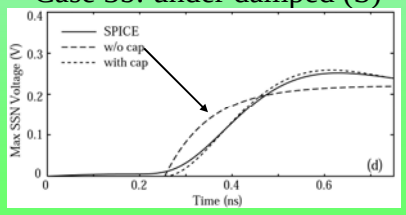
Case 2: critically damped



Case 3a: under damped (F)

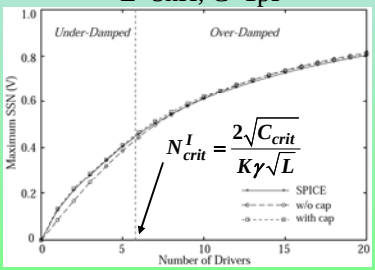


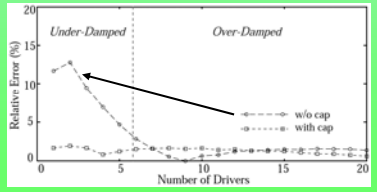
Case 3b: under damped (S)



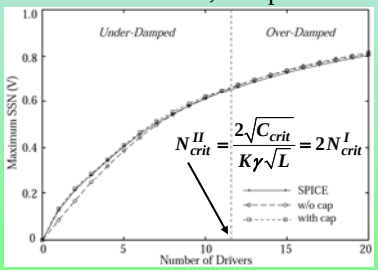
Comparison of Peak Noise Voltage

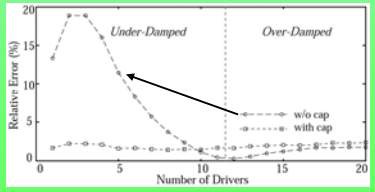
Single ground pad:
L=5nH, C=1pF





Two ground pads:
L=2.5nH, C=2pF





Contributions

- ✓ Simple & accurate modeling of SSN for chip output drivers
- ✓ Parasitic capacitance effect discussed and modeled
- ✓ Idea of application-specific device modeling is generic

THE END

List of Tools

- In Research Papers
  - ClariNet
  - Harmony / Global Harmony

SubstrateStorm

- Previously called Layin (created by SnakeTech)
- Modeling, noise analysis for RF, analog, mixed-signal IC designs
- Characterization of CMOS, BiCMOS, and bipolar processes using lightly doped, epitaxial or silicon-on-insulator (SOI) bulks
- In addition to its path through an IC well, it can also find the frequencies at which the noise enters the substrate cells

SubstrateStorm

- FEM-like analysis
- 3D mesh to model IC substrate
  - Vertical gridlines: doping profiles
  - Surface grid: IC layout
- Inputs
  - IC layout information
  - Technology characterization (CMOS, BiCMOS etc.)
- Modeling accuracy: upto 20% of actual Si

GateScope

- Created by Moscape (now part of Magma)
- Goals
  - Analyze design for noise problems
  - Locate noise violations
  - Generates data to automate repair
- Employs assertion-based technology to identify and correct noise problems caused by cross-coupling effects in ASIC designs at 0.18 µm and below

GateScope

- Noise affecting functionality + timing
- Whole-chip noise detection run followed by highly detailed analysis
- Works initially at gate level to isolate ‘noisy’ circuits
- Info from static timing analysis to determine which signals are likely to switch at the same time and whether crosstalk interference will cause stable levels to switch
- Then detects aggressor-victim combos and descends to transistor level
- Deterministic transient analysis to methodically eliminate false errors on multimillion gate designs.

Nova

- From IBM
- Full-chip power supply noise analysis tool
- can simultaneously analyze resistive IR drop and inductive delta-I noise on a full-chip scale
- Designers can
  - easily identify the hot spots,
  - optimize decoupling capacitor placement,
  - minimize power supply noise,
  - preserve signal integrity,
  - improve circuit performance

Eldo / Eldo RF

- From Mentor Graphics
- Both are fast transistor-level structure simulators driven by a Spice netlist
  - Include noise analysis tools
- Eldo RF supports phase noise analysis
- Description available is for the circuit simulation tool. Nothing specific about noise analysis.

CeltIC

- From Cadence
- Identifies nets with low noise immunity to avert potential noise-related problems and lethal silicon failures before tapeout.
- Accurately calculates the impact of noise on both the delay and functionality of cell-based designs.
- Performs SoC noise analysis and generates repairs back into place-and-route.

CeltIC

- Key features and benefits
  - Prevents silicon respins due to noise related functional failures
  - Accurately accounts for crosstalk effects on timing
  - Improves yield by fixing nets with low noise immunity
  - Reduces design iterations via early detection of signal integrity problems
  - Isolates and repairs crosstalk-induced functional and delay failures
  - Calculates the impact of noise on delay and slew for feedback to STA

CeltIC

- Key features and benefits (contd.)
  - Reduces SI closure iterations by filtering false failures by over 10 to 100X versus other crosstalk analyzers
  - Predicts functional, timing, and yield problems resulting from bootstrap and overshoot/undershoot noise
  - Performs accurate glitch propagation to verify noise immunity with no additional overhead characterization
  - Performs internal timing window convergence to reduce pessimism
  - Automates noise library creation for cells, memories, I/Os, and custom macros
  - Handles multimillion SoC designs flat or hierarchically using ECHO models

Swan

- Substrate-noise Waveform Analysis from Interuniversity Microelectronics Centre
- Substrate noise analysis on SOCs where large digital circuits generate ground bounce
- Uses macromodels to analyze noise
  - Adapts techniques for low-ohmic devices to study of high-ohmic substrates

Swan

- Swan consists of two parts
  - Standard cell-library characterization
    - Record substrate-noise generation and power-supply current related to switching activity for all standard cells.
    - Once-only for given technology and cell library
  - Substrate-noise waveform computation
    - Switching events from gate-level VHDL sim
    - Combine switching-noise generation model & switch data to get waveform and ground bounce.

Substrate Noise Analyst

- Substrate noise analyzer for RF, analog, and
- mixed-signal ICs from Cadence
- It provides a silicon-accurate model for substrate coupling effects to enable chip integration
- Captures full-chip noise effects using static and dynamic techniques to model switching noise
- Accelerates noise simulation on sensitive analog/RF circuits by utilizing RC reduction
- Reduces noise coupling through isolation analysis using graphical visualization of surface noise distribution

Substrate Noise Analyst

- Other features
  - Accurate 3D substrate modeling
    - Advanced semiconductor physics based
    - Minimum of 80% accuracy across various techs
  - Advanced digital noise modeling
    - New static noise modeling techniques
  - Simulation netlist
    - Generates substrate RC network connected to bulk terminals of selected devices

Pacific Static Noise Analyzer

- Analyzes the combined impact of major noise sources including crosstalk, IR drop, and propagated noise on the design
- Prevents functional chip failures due to noise in custom digital circuits
- Improves chip yield by identifying noise sensitivity circuitry
- Calculates crosstalk impact on timing to assist static timing signoff

Pacific Static Noise Analyzer

- Other features
  - Advanced circuit and interconnect noise analysis
    - non-linear transistor-level transient simulation engine and
    - advanced RLCK network solver for efficiently analyzing large coupled networks
    - Interconnect noise due to crosstalk combined with circuit noise due to charge-sharing, leakage, IR drop, overshoot, undershoot, and noise propagating from previous logic stages

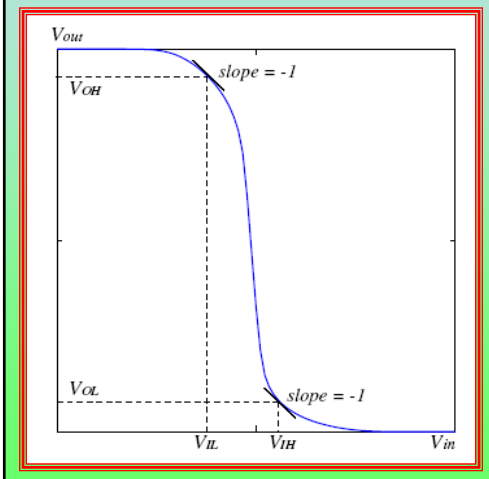
Pacific Static Noise Analyst

- Other features (contd.)
  - Noise stability
    - Uses sensitivity-based noise stability metric to determine noise immunity of every node
    - metric helps localize failures near their source and guarantees adequate noise immunity
    - circuit cross-section with the appropriate waveform stimulus required to replicate the problem

Pacific Static Noise Analyst

- Other features (contd.)
  - Full-chip hierarchical analysis
    - Uses high-level noise abstractions
    - Supports UDN model (incomplete or non-digital blocks) and ECHO model (transistor level)
  - SOI noise analysis
    - Account for floating-body and parasitic effects

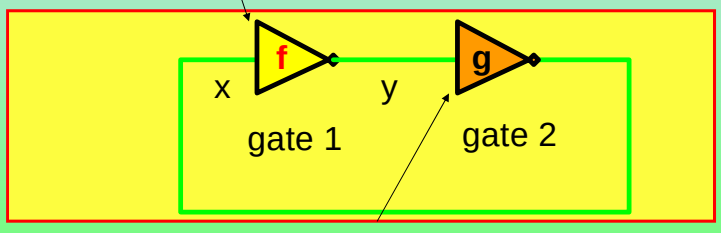
Definition of V_{OH} and V_{OL}



- Given by -1 slope points
- Stable logic states of an infinite chain of inverters or a bistable inverter pair

Static Noise Margin Criteria

- Odd gate
  $V_{out} = f(V_{in})$



- Even gate
  $V_{out} = g(V_{in})$

Noise Margin Criteria

- **Coincidence of roots of flip-flop equation**

$$x - g(f(x)) = 0$$

- **Small-signal closed loop gain**

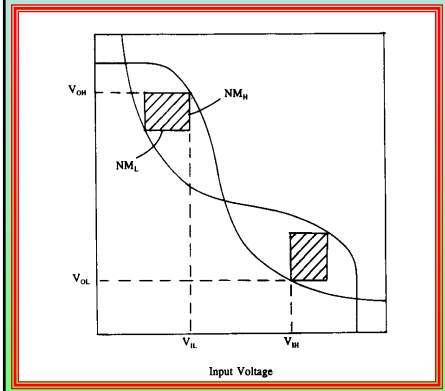
$$\frac{\partial f}{\partial x} \cdot \frac{\partial g}{\partial y} = 1$$

Noise Margin Criteria

- **Jacobian of Kirchhoff equation is zero**

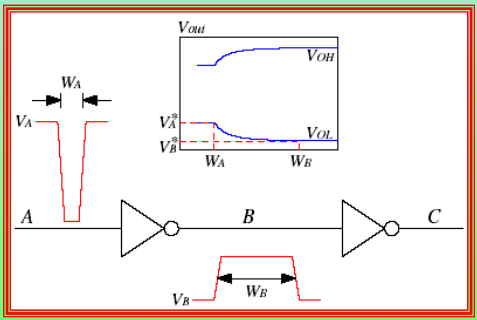
$$F_1 \equiv x - g(y) = 0$$
$$F_2 \equiv y - f(x) = 0$$
$$J = \begin{vmatrix} \frac{\partial F_1}{\partial x} & \frac{\partial F_1}{\partial y} \\ \frac{\partial F_2}{\partial x} & \frac{\partial F_2}{\partial y} \end{vmatrix} = 0$$

Static Noise Margin Criteria



- Maximum square between normal and mirrored VTC
- $NM_H \cdot NM_L$ is Maximum (maximum product criteria)

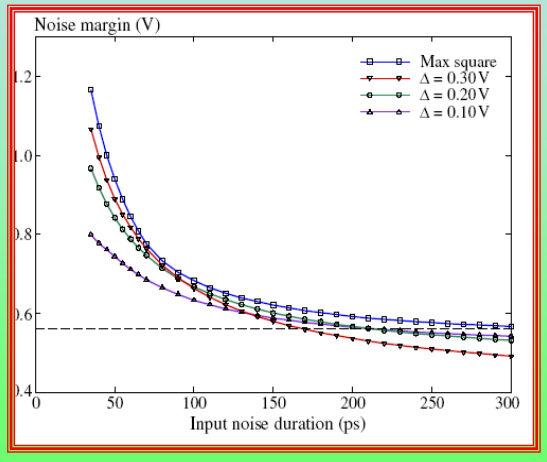
Checking DNM Only is not Enough



- No noise violation if only
  - Check V_{IH} for V_A
  - Check V_{IL} for V_B

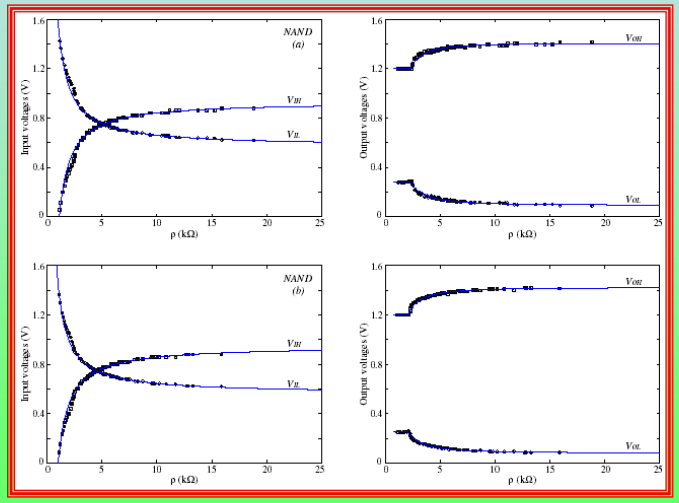
  for individual stage
- Noise violation if consider the impact of last stage so need to
  - Record V_{IH}, V_{IL}, V_{OH}, V_{OL}

Simplified dynamic noise margin definition

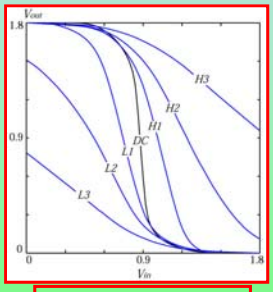


Comparison of dynamic noise margin value

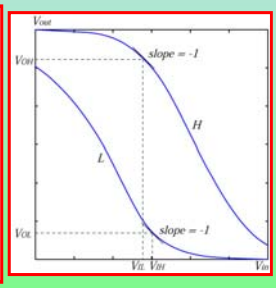
Modeling DNM for NAND



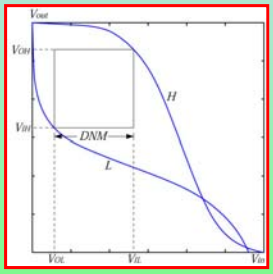
Phase I: Worst-Case Noise Margin Based



Voltage transfer characteristic

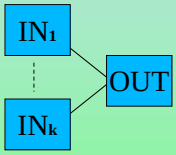


Unity gain based definition



Maximum square based definition

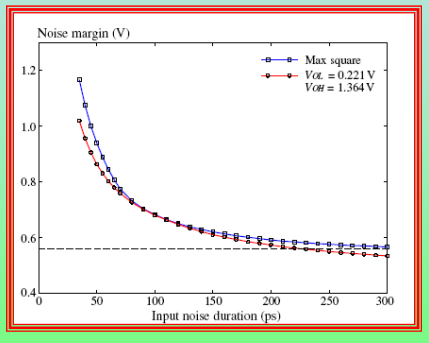
Comparison of Basic DNA Tool with Static Timing Analysis Tools

	DNA	STA
	**Propagate Noise Using Superposition or Maximization**	**Propagate Delay Using Maximization**
Gate	**Dynamic Noise Margin Modeling of Gates**	**Gate Delay Modeling**
Criteria	**Dynamic Noise Margin**	**Delay Constraints**
Baseline Algorithm	**Similar Algorithms**	
	Conditional Propagation	**False Path Elimination**

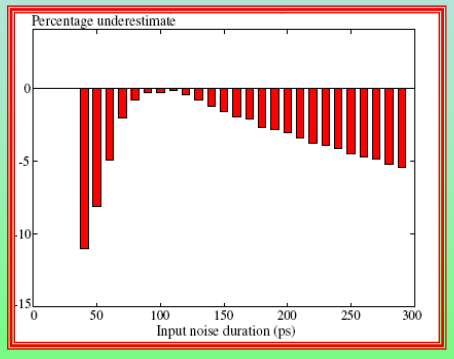
DNA Organization and Tasks

- **DNM Library**
- **Noise Library**
- **Noise Evaluation Engine**
- **Basic Longest/ Shortest Path Algorithm**
- **Interconnect Templates**
- **Simple Algorithm For Constraints Optimization As a Plus**

Simplified dynamic noise margin definition



Comparison of dynamic noise margin values



Percentage that the simplified method underestimates

Improving Dynamic CMOS Circuit Noise Tolerance Using Resonant Tunneling Devices

GSRA: Li Ding

Presented by: Prof. Pinaki Mazumder

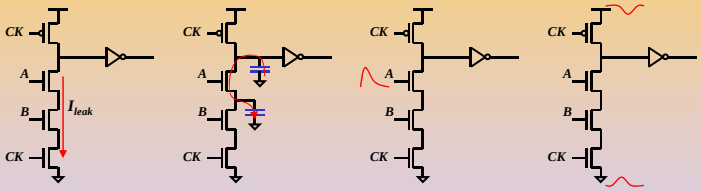
Department of Electrical Engineering and Computer Science

University of Michigan, Ann Arbor

Design Automation and Test in Europe (DATE)

Noise In Digital Systems

- What is Noise? – Any deviation from expected state/value



- Why Important? – Noise margin is continuously decreasing

Year	2001	2004	2007	2010	2013	2016
Tech (um)	130	90	65	45	32	22
Density (M)	38.6	77.2	154.3	1718	3532	7208
VDD (V)	1.1-1.2	1.0-1.1	0.7-0.9	0.6-0.8	0.5-0.7	0.4-0.6

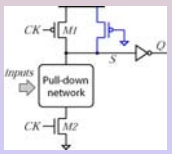
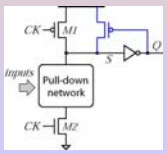
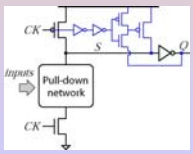
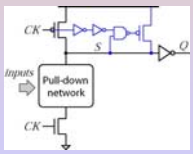
Three Central Questions

- How much noise is generated?
- Will noise affect chip functionality?
- How to ensure chip functionality?
  - Generate less noise
  - Make circuits more noise tolerant

Dynamic CMOS logic gates are employed in High-speed Circuits
Achilles hill: Low Noise Immunity

Existing NT Design Techniques - I

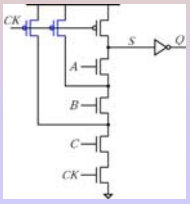
- Employing keeper
- Precharging internal nodes
- Raising source voltage
- Constructing complementary p-network

Always-on keeper Krambeck 82	Feedback keeper Oklobdzija 85	HS feedback keeper Anis 00	Conditional keeper Alvandpour 01
			
DC power consumption		Leakage noise Only	Leakage noise Only

Existing NT Design Techniques - II

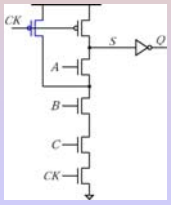
- Employing keeper
- Precharging internal nodes
- Raising source voltage
- Constructing complementary p-network

Precharge all nodes
Lee 86



Area overhead
Charge sharing only

Partial precharge
Pretorius 86

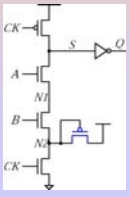


Charge sharing only

Existing NT Design Techniques - III

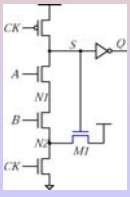
- Employing keeper
- Precharging internal nodes
- Raising source voltage
- Constructing complementary p-network

PMOS pull-up
D'Souza 96



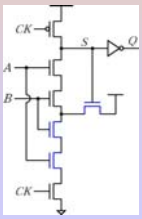
DC power consump

NMOS pull-up
Schorn 98



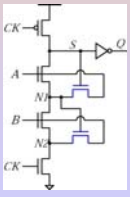
DC power consump

Mirror technique
Wang 99



Area overhead
Delay overhead

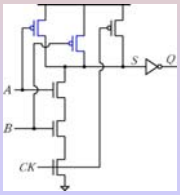
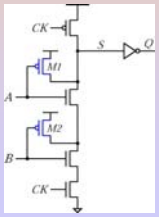
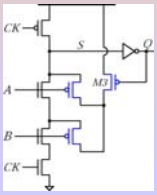
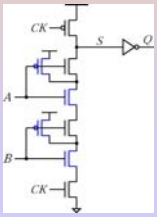
Twin transistor
Balamurugan 99



Area overhead
Only for some logics

Existing NT Design Techniques - IV

- Employing keeper
- Precharging internal nodes
- Raising source voltage
- Constructing complementary p-network

Comp. P-network Murabayashi 95	CMOS inverter Covino 97	Gated CMOS inverter Evans 98	Triple transistor Bobba 99
			
Area overhead	Area overhead Only for some logics	Area overhead Only for some logics	Area overhead Delay overhead

Metrics for NT-Technique Evaluation

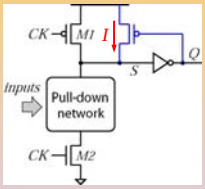
- Noise criterion – Improve tolerance against all noise types
- Functionality criterion – Suitable for all circuit functions
- Power criterion – No DC power consumption
- Area criterion – Limited circuit area overhead
- Delay criterion – Limited circuit delay overhead

Comparison of Existing Techniques

		Noise	Func	Power	Area	Delay
Always-on keeper	6			●		
Feedback keeper	1					
HS feedback keeper	2	●				
Conditional fdbk keeper	2	●				
Precharge internal nodes	3	●			●	
Partial precharge	2	●				
PMOS pull-up	3			●		
NMOS pull-up (fdbk)	3			●		
Mirror technique	5				●	●
Twin transistor	3		●		●	
Complementary p-network					●	
CMOS inverter	4		●		●	
Gated CMOS inverter	4		●		●	
Triple transistor	5				●	●

Passed all criteria

The Dilemma



- High noise tolerance → large keeper strength (➔W/L ratio)
- High performance → small keeper strength

Conflicting goals: Speed v. Area for noise optimization

- Dilemma : How to choose the size of the keeper?

The Idea

- Keeper strength for gate speed

$$I_{sp} = \frac{2}{V_{DD}} \int_0^{V_{DD}/2} I(V) \cdot dV \downarrow$$

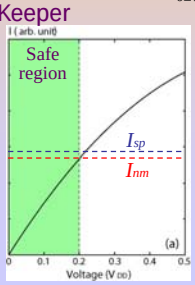
- Keeper strength for noise tolerance

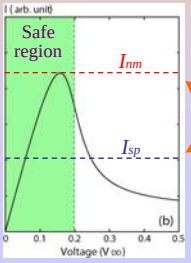
$$I_{nm} = \max_{0 \le V \le V_D} (I(V)) \uparrow$$

With PMOS Keeper

With NDR Keeper

Poor Tradeoff

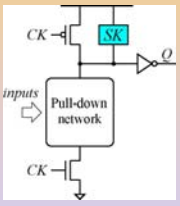


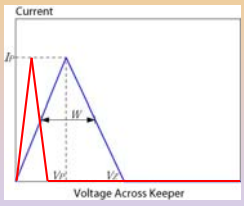


Excellent Tradeoff

Smart Keepers

Smart Keeper: NDR keeper that aggressively explores difference between keeper strengths for speed and for noise tolerance





Maximum input noise voltage

$$V_{max} = \underbrace{\frac{I_P}{I_0}(V_{DD} - V_T)}_{\text{Large } I_P} + V_T$$

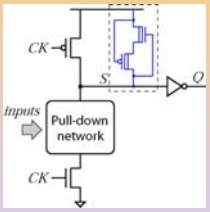
Gate delay

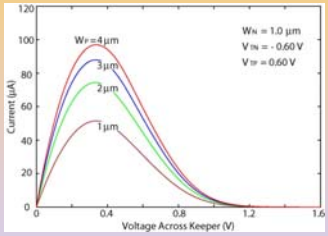
$$t_d = \left(\frac{V_T}{2V_{DD}} t_r + \frac{CV_{DD}}{2I_0}\right) + \underbrace{\left(\frac{C}{I_P} \ln\left(\frac{I_0}{I_0 - I_P}\right) - \frac{C}{I_0}\right) \cdot W}_{\text{Small } W}$$

Design Goal of NDR: High Peak Current (I_P), Low Valley Voltage (V_V)

Circuit Implementations - I

Smart Keeper Design 1: SK1

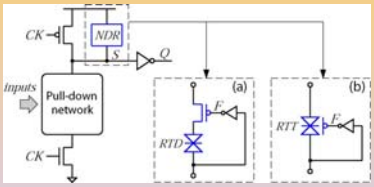


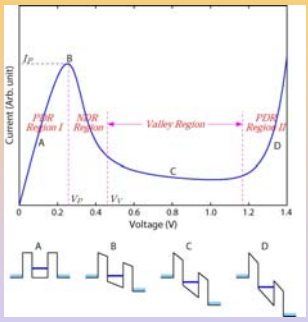


- Two cross-coupled MOS transistors
- **Depletion** mode transistors are needed

Circuit Implementations - II

Smart Keeper Design 2: SK2





- RTD → provide folded-back I-V characteristic
- MOS devices → completely shut down the keeper

Processes in Double-Barrier Tunneling

Fermi-Dirac Statistics

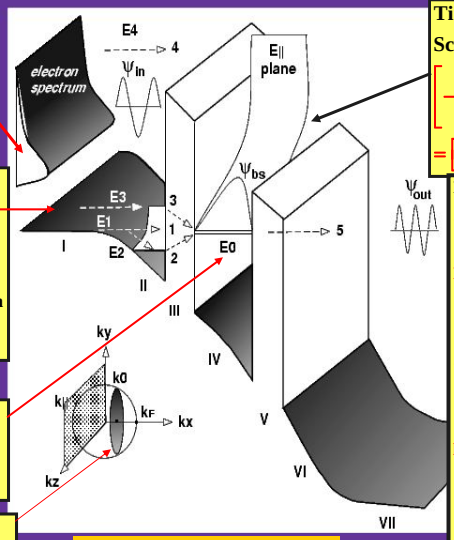
$E_{3D} = E_C + \frac{\hbar^2 k_x^2}{2m^*} + \frac{\hbar^2 k_\perp^2}{2m^*}$,

where E_C is the conduction band energy

$k_\perp$ = Transverse Momentum $= \sqrt{k_x^2 + k_y^2}$

$$E_{2D} = E_0 + \frac{\hbar^2 k_\perp^2}{2m^*}$$

Electrons having momenta in the disk tunnel through, where $k_0^2 = 2m^*(E_0 - E_C)/\hbar^2$



$E_0 = E_1$
$= E_2 + h\nu$ (Phonon Recombination)
$= E_3 - h\nu$ (Phonon Emission)

Proceedings of the IEEE, 1998

Time-independent effective mass Schrodinger Equation

$$\left[-\frac{\hbar^2}{2}\nabla\left(\frac{1}{m^*}\nabla\right) + V(\vec{r})\right]\psi_n(\vec{r}) = \left[E(\vec{k}) - E_n(0)\right]\psi_n(\vec{r})$$

Device Current Density:

$$J_{tot} = -q\hbar\sum_k W(k).$$

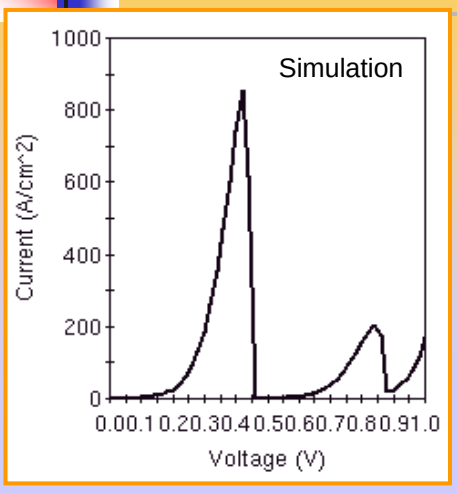
$$\mathrm{Im}\left[\psi_k^*(x)\frac{1}{m^*(x)}\frac{\partial\psi_k(x)}{\partial x}\right]\Delta k$$

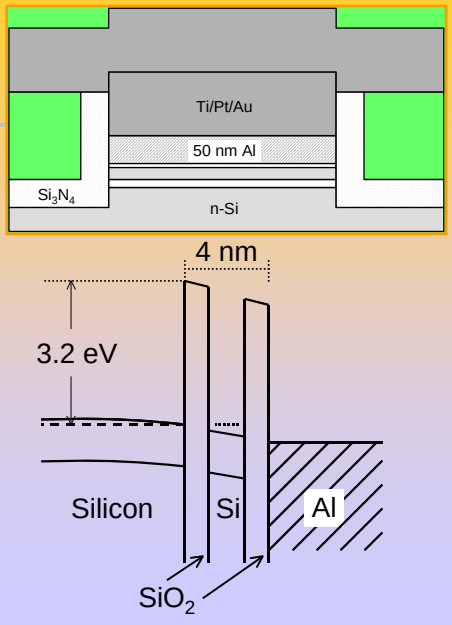
$$J_{tot} = \frac{qm^* k_B T}{2\pi^2\hbar^3}\int_{E_c}^{\infty} T(E_x).$$

$$\log\left[\frac{1+\exp\left(\frac{E_F - E_x}{k_B T}\right)}{1+\exp\left(\frac{E_F - E_x - qV_A}{k_B T}\right)}\right]dE_x$$

$$T(E) = \frac{\left(\frac{\Gamma}{2}\right)^2}{\left[E - (E_r - eV/2)\right]^2 + \left(\frac{\Gamma}{2}\right)^2}$$

$SiO_2/Si/SiO_2$ RTD

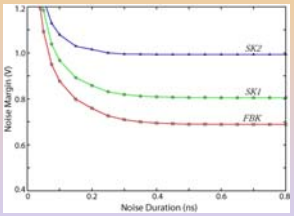




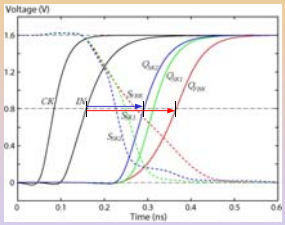
Simulation Results

- OR8 gate

Noise rejection curves



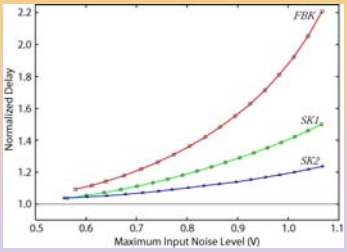
Transient waveform

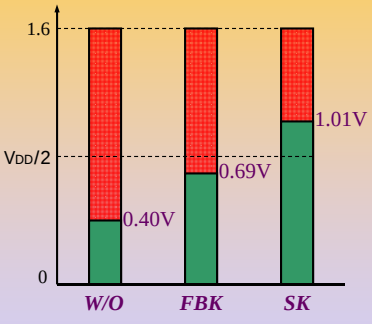


FBK : feedback keeper
SK1 : cross-coupled FET keeper
SK2 : RTD-based smart keeper

CK : clock signal
IN : input signal
S : internal dynamic node
Q : output signal

Speed and Noise Immunity

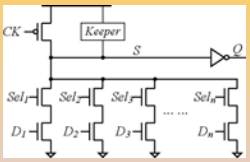




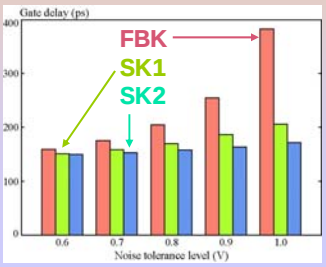
- More noise-tolerant than static CMOS gates!

Wide Fan-in Multiplexers

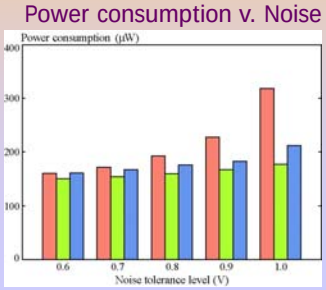
- Wide fan-in 16:1 multiplexers
- 0.18-μm process tech
- Supply voltage : 1.6 V



Gate Delay v. Noise

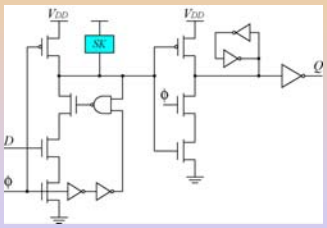


Power consumption v. Noise

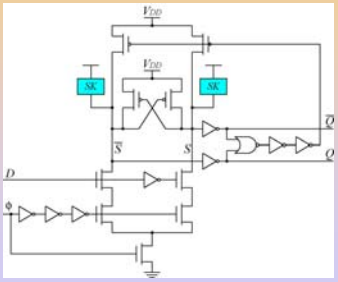


Application to Other Dynamic Circuits

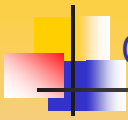
- Smart keepers can be used in any digital circuits with dynamic node



SDFF (Sun Microsystems)



K6ETL (AMD)



Conclusions

- Smart keepers – novel class of NT techniques
- Small (and constant) area overhead
- Small performance (speed & power) overhead
- Very suitable for post-layout fixing