Manufacturing-Aware Physical Design Techniques

Puneet Sharma

ECE Department, UCSD

Advisor: Prof. Andrew B. Kahng

Aug 29, 2007

Publications

- In this theme:
  - A. B. Kahng, S. Muddu and P. Sharma, "Defocus-Aware Leakage Estimation and Control," **TCAD07**.
  - P. Gupta, A. B. Kahng, P. Sharma and D. Sylvester, "Gate-Length Biasing for Runtime Leakage Control," **TCAD06**.
  - A. B. Kahng, P. Sharma and R. O. Topaloglu, "Exploiting STI Stress for Performance," **ICCAD07**.
  - A. B. Kahng, S. Muddu and P. Sharma, "Detailed Placement for Leakage Reduction using Systematic Through-Pitch Variation," **ISLPED07**.
  - A. B. Kahng, P. Sharma and A. Zelikovsky, "Fill for Shallow Trench Isolation CMP," **ICCAD06.**
  - A. B. Kahng, S. Muddu and P. Sharma, "Impact of Gate-Length Biasing on Threshold-Voltage Selection," **ISQED06**.
  - A. B. Kahng, K. Samadi and P. Sharma, "Study of Floating Fill Impact on Interconnect Capacitance," **ISQED06**.
  - A. B. Kahng, C.-H. Park, P. Sharma and Q. Wang, "Lens Aberration Aware Timing-Driven Placement," **DATE06**.
  - P. Gupta, A. B. Kahng, S. Nakagawa, S. Shah and P. Sharma, "Lithography Simulation-Based Full-Chip Design Analyses," **SPIE06**.
  - A. B. Kahng, S. Muddu and P. Sharma, "Defocus-Aware Leakage Estimation and Control," **ISLPED05**.
  - P. Gupta, A. B. Kahng, C.-H. Park, P. Sharma, D. Sylvester and J. Yang, "Joining the Design and Mask Flows for Better and Cheaper Masks," INVITED **SPIE04**.
  - P. Gupta, A. B. Kahng, P. Sharma and D. Sylvester, "Selective Gate-Length Biasing for Cost-Effective Runtime Leakage Control," **DAC04**.

Outline

- Introduction
- Systematic Variation-Aware Techniques
  - ACLV-Aware Leakage Analysis and Optimization
  - Detailed Placement for Leakage Optimization
  - Aberration-Aware Timing Analysis
- Utilizing STI Stress in Delay Analysis and Optimization
- Other Research Contributions
- Conclusions

Process Variations: Sources & Taxonomy

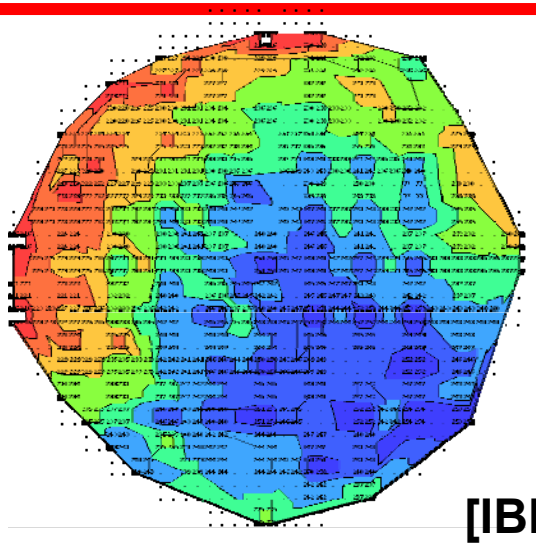
- Modern semiconductor manufacturing extremely complex and process variations unavoidable
- Sources
  - Wafer: topography, reflectivity
  - Resist: Thickness, refractive index
  - Reticle: CD error, proximity effects, defects
  - Stepper: Lens heating, defocus, dose variation, lens aberrations
  - Etch: Power, pressure, flow rate
- Taxonomy
  - Nature
    - Systematic: focus, aberration, topography, proximity
    - Random: material variations, *all difficult to model variations*
  - Spatial scale
    - Intra-die: proximity effects, topography, etch bias
    - Inter-die: focus, aberrations, stage error (wafer-to-wafer), batch-to-batch material variations (lot-to-lot)

Yield

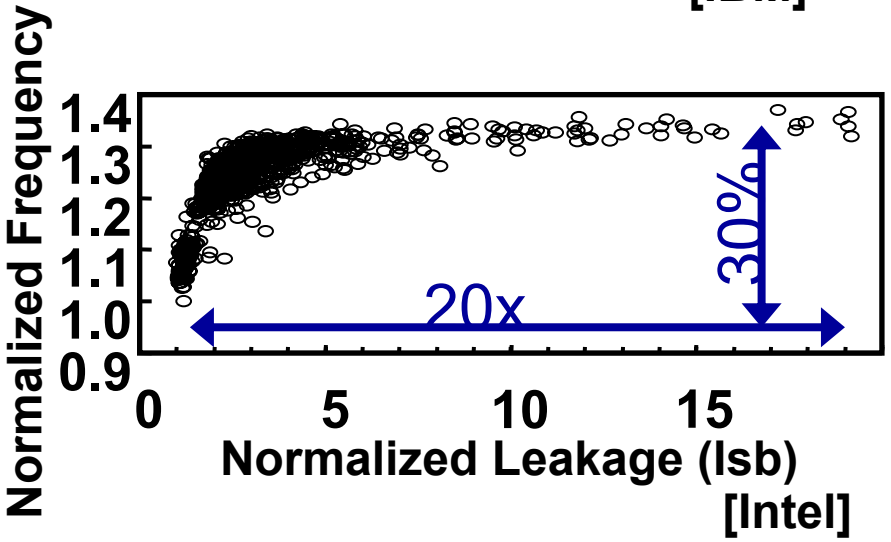
- Fraction of chips that function *and* meet performance and/or power specifications
- Two types:
  - Functional yield
  - Parametric yield
- Functional yield: chips that function (may not meet specs)
  - Causes of functional yield loss: large process variations, random defects, misprocessing
  - Examples of functional failures: short & open circuits, line-end shortening, etc.
  - Solutions to increase functional yield: design rules (today), yield models, CAA, etc.

Parametric Yield

- Parametric yield: fraction of functional chips that meet frequency and power specifications
- Parametric yield loss is caused by variability in delay and power
- Primary cause of delay and leakage variability: process variations
  - Lateral dimension variations, e.g., gate-length
  - Topography variations, e.g., interconnect height
  - Stress effects, e.g., due to different STI widths
  - Material variations, e.g., dopant concentration



[IBM]



[Intel]

***DFM*: measures taken in design to enhance yield**

Traditional DFM: corner-based models, design rules, RETs

DFM Today

- Corner-based models
  - Convey device design metrics from process to design
  - Safety margins (guardbanding) kept to ensure correctness in presence of unmodeled effects and process variations
  - Essentially give upper (or lower) bounds on design metrics
- Design rules
  - Convey manufacturing limitations from process to design
  - E.g.: min. width, min. spacing, min. and max. density
  - If design rules followed → high manufacturing yield
- Resolution enhancement techniques (RETs) + fill insertion
  - Performed after sign-off to minimize lateral dimension and topography variation
- Advantages: simplicity, easy of adoption
- **Disadvantages**:
  - Lose performance: too much guardbanding
  - Lose yield: design oblivious to process variations
  - Complex design rules: process variations depend on complex layout configs
  - High turn-around: tools less effective due to DRCs, designers under pressure to deliver on expectations, unnecessary RET
  - Predictability: RET+fill applied after sign-off

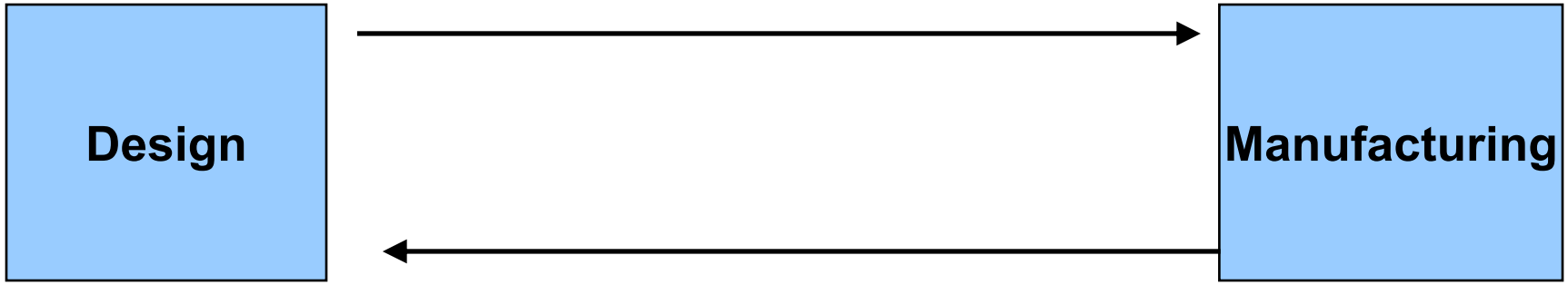
Need for Novel DFM Solutions

- Bidirectional information exchange between design and manufacturing

Pass functional intent

E.g. Apply aggressive RET for critical features only

+ Reduces cost and time to market



My Focus

Pass variation models and manufacturing limits

Better estimate variability in design, systematic variation-driven optimization, avoid patterns that cannot be manufactured

+ Improves yield (better power and performance)

How Manufacturing-Aware Design Helps?

- Reduce variability: RETs, regularity, fill insertion
  - **STI fill**
  - **Metal fill**
- Design robustness enhancement
  - Resistance to variations: gate-biasing, wire spacing, less usage of low V_{th}, logic depth, #critical paths
  - Redundancy: via-doubling, ECC
  - **Leakage & its variability control with gate-length biasing**
- Model systematic variations and utilize in analyses & optimizations
  - **ACLV-aware leakage estimation & control**
  - **Detailed placement for leakage**
  - **Aberration-aware timing analysis**
  - **Utilizing STI stress in timing analysis and optimization**
- Statistical methods: SSTA, statistical analysis and optimization of leakage

Outline

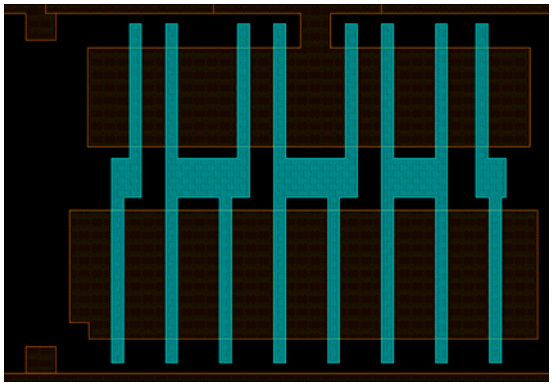
- ✓ Introduction
- Systematic Variation-Aware Techniques
  - ACLV-Aware Leakage Analysis and Optimization
  - Detailed Placement for Leakage Optimization
  - Aberration-Aware Timing Analysis
- Utilizing STI Stress in Delay Analysis and Optimization
- Other Research Contributions
- Conclusions

ACLV-Aware Leakage Analysis & Optimization

- Most significant source of leakage variability: linewidth (=gate-length) variation
  - E.g., in 90nm technology, decrease of linewidth by 10nm → leakage increases by 5X for PMOS and 2.5X for NMOS
- Traditional leakage estimation techniques model linewidth variation as random → very pessimistic
- Large fraction of linewidth variation is across-chip (ACLV: across-chip linewidth variation)
- Reality: ACLV systematically varies with defocus and pitch
- This work: (1) model systematic ACLV → (2) improve leakage estimation accuracy → (3) optimize leakage accurately
- Publications:
  - ISLPED'05, TCAD (to appear)

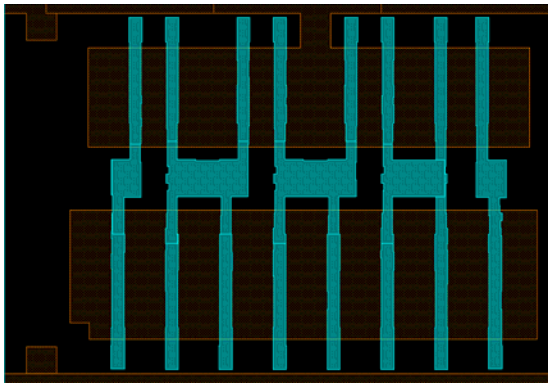
Linewidth Variation with Defocus

Defocus: Gap between wafer plane and focal plane (ideal location)



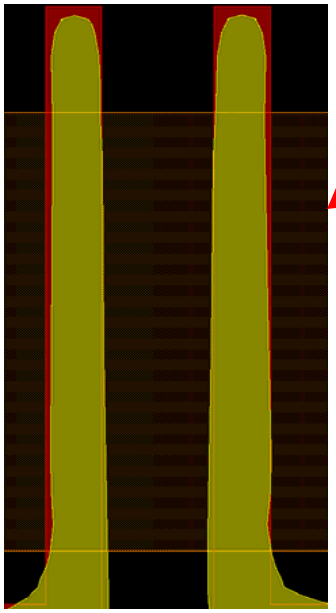
Standard cell

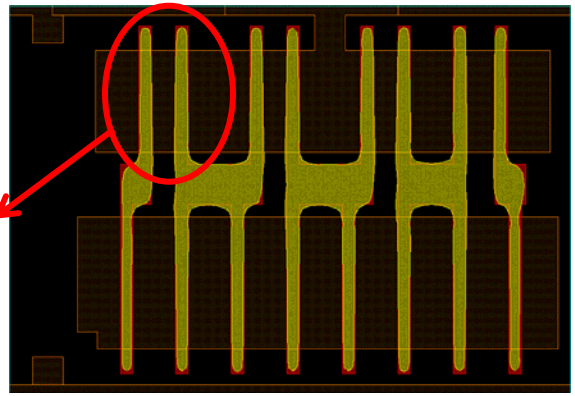
OPC at nominal defocus



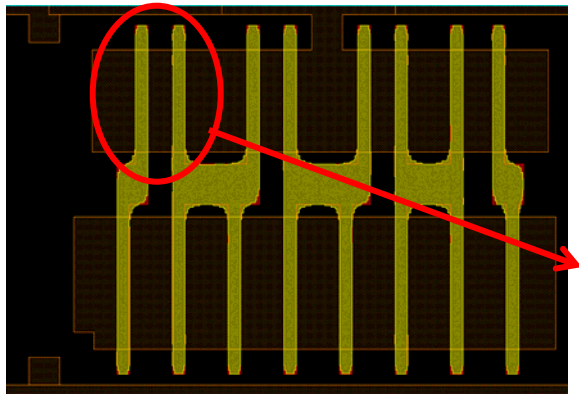
Lithography simulation at 200nm defocus

Lithography simulation at nominal defocus

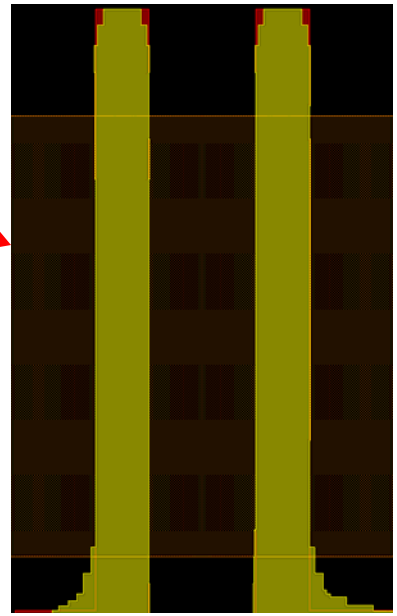




Printed polysilicon line in yellow shows LARGE deviation from drawn for 200nm defocus



Printed polysilicon line in yellow shows NO deviation from drawn for nominal defocus

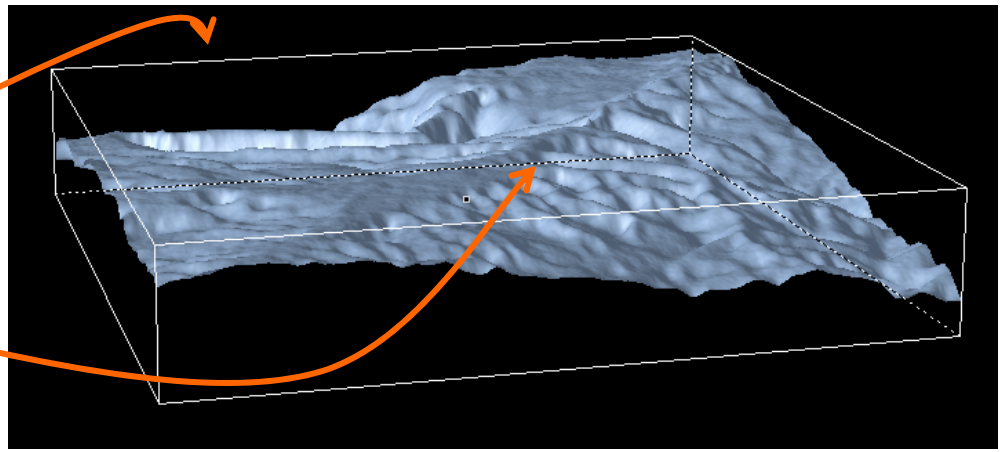


Sources of Defocus

- Defocus during lithography is caused primarily due to wafer topography variation, lens aberration and wafer plane tilt
- Wafer topography variation is caused due to chemical-mechanical polishing (CMP) anomalies during wafer processing
  - Substrate flatness, films, etc. also contribute to wafer topography

Imperfect wafer planarity after STI CMP

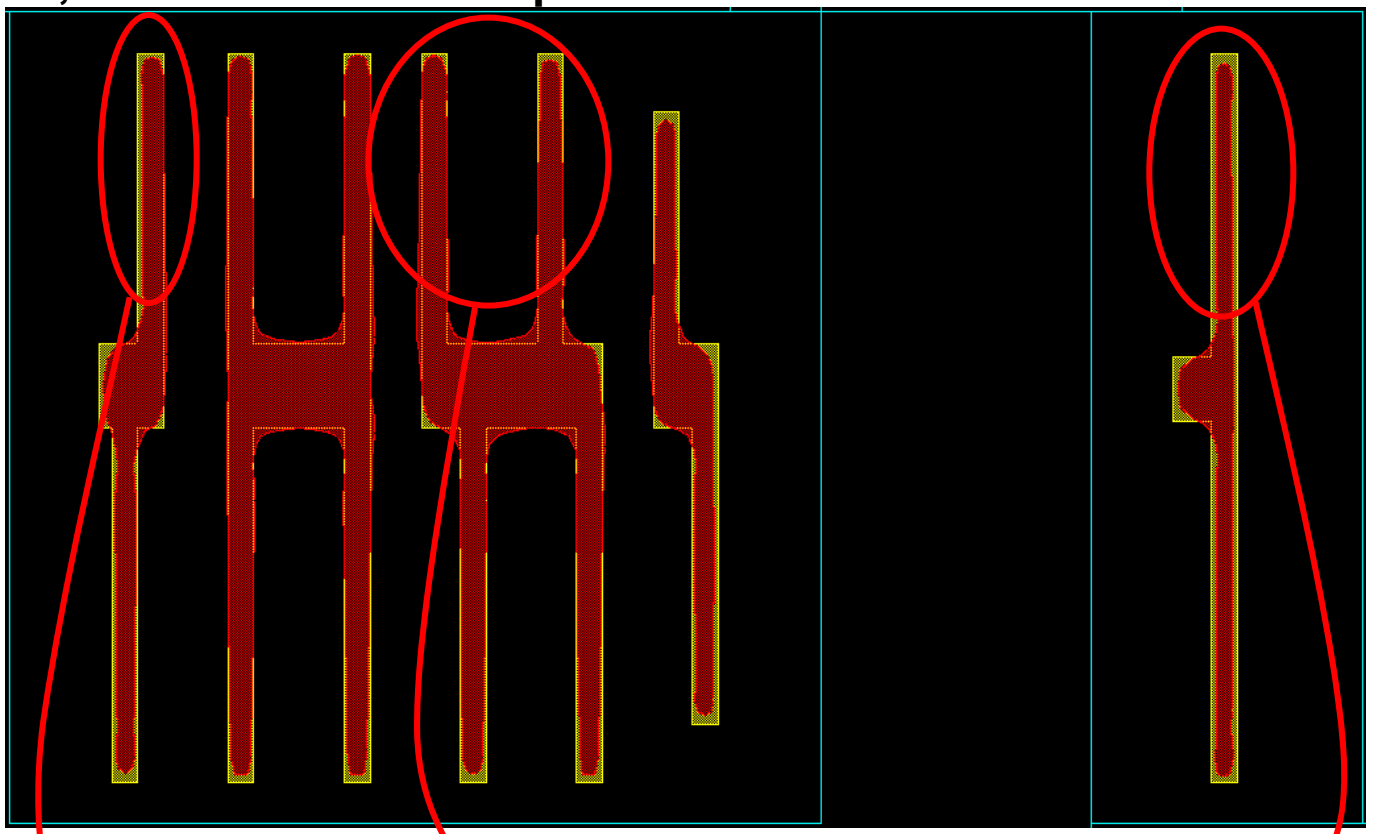
Images print at different defocus levels depending on the topography of the location



Linewidth Variation with Pitch

Pitch of a feature: its spacing with left and right neighbors
Dense pitch implies small spacing, *isolated* or *sparse* pitch implies large

Portion of a 90nm standard cell layout showing polysilicon lines in isolated, dense and self-compensated contexts



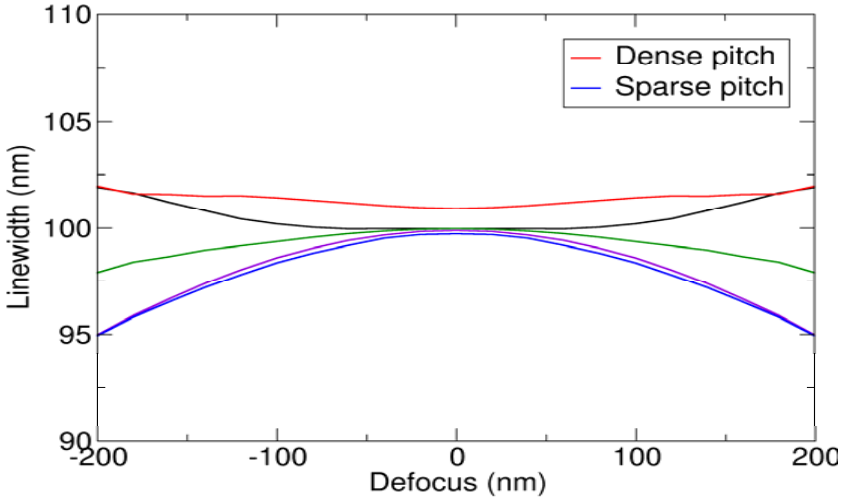
Self-compensated lines
linewidth ~ nominal

Dense lines
linewidth > nominal

Isolated lines
linewidth < nominal

Across-Chip Linewidth Variation

- Linewidth variation compensated by OPC at nominal defocus
  - At defocus levels other than nominal, linewidth varies systematically with pitch
  - For dense pitches: linewidth increases with defocus (smiling)
  - For isolated: linewidth decreases with defocus (frowning)



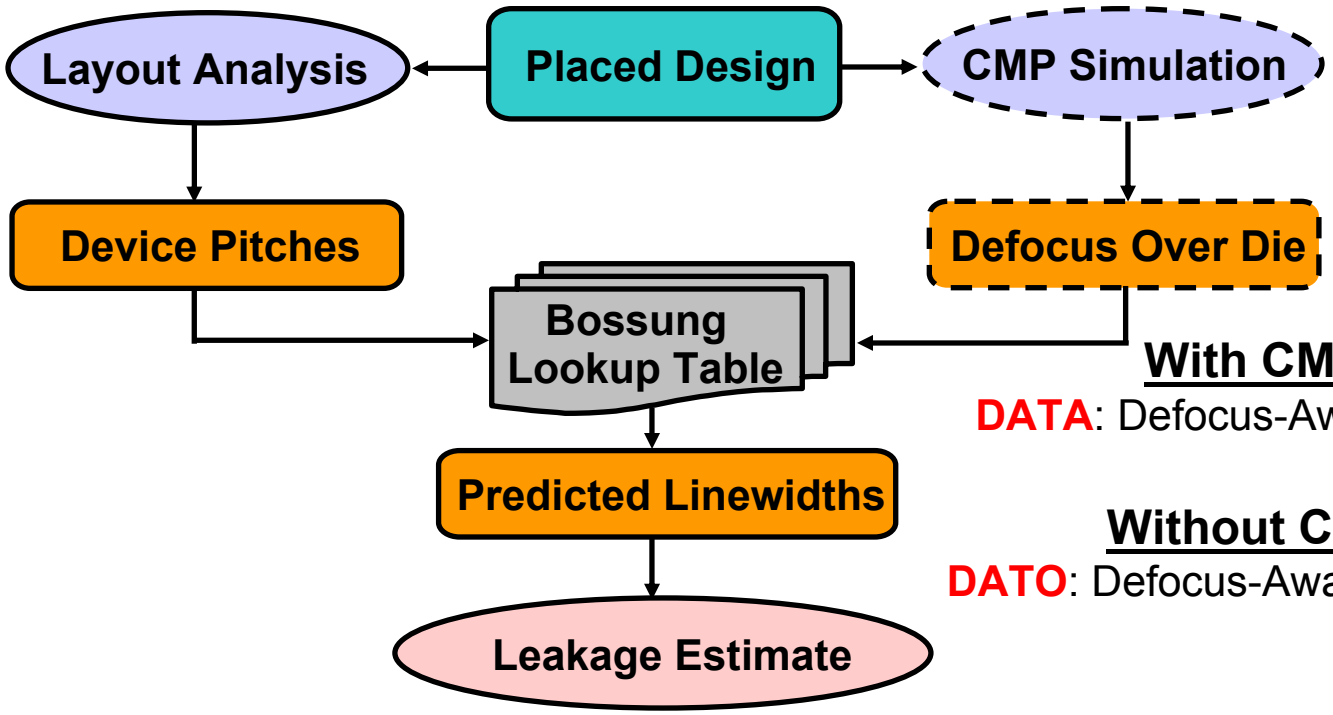
Bossung plot

  - At any given defocus level, *linewidth for dense pitches is always greater than that of isolated pitches*

- Linewidth variation with pitch and defocus is captured in *Bossung* lookup tables

Defocus-Aware Leakage Estimation Flow

- Key idea: Layout analysis → predict on-silicon linewidth → leakage estimation



With CMP Simulation

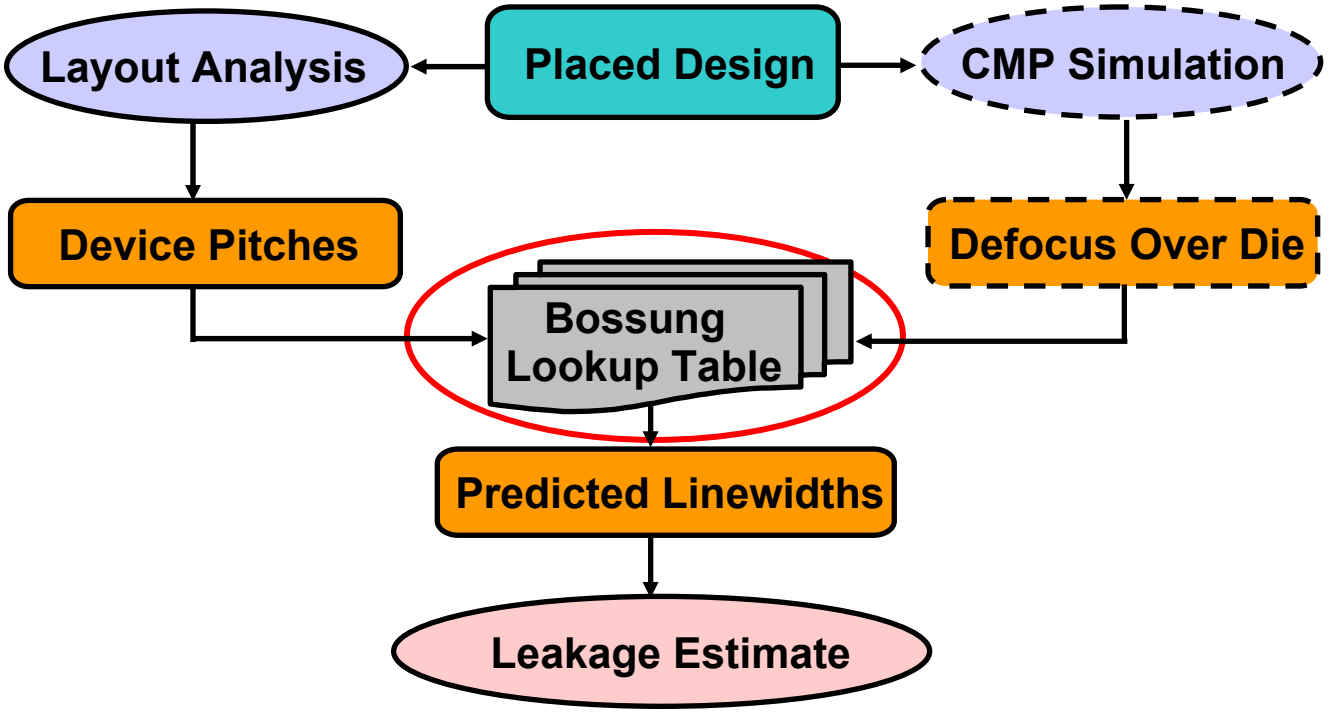
DATA: Defocus-Aware, Topography-Aware

Without CMP Simulation

DATO: Defocus-Aware, Topography-Oblivious

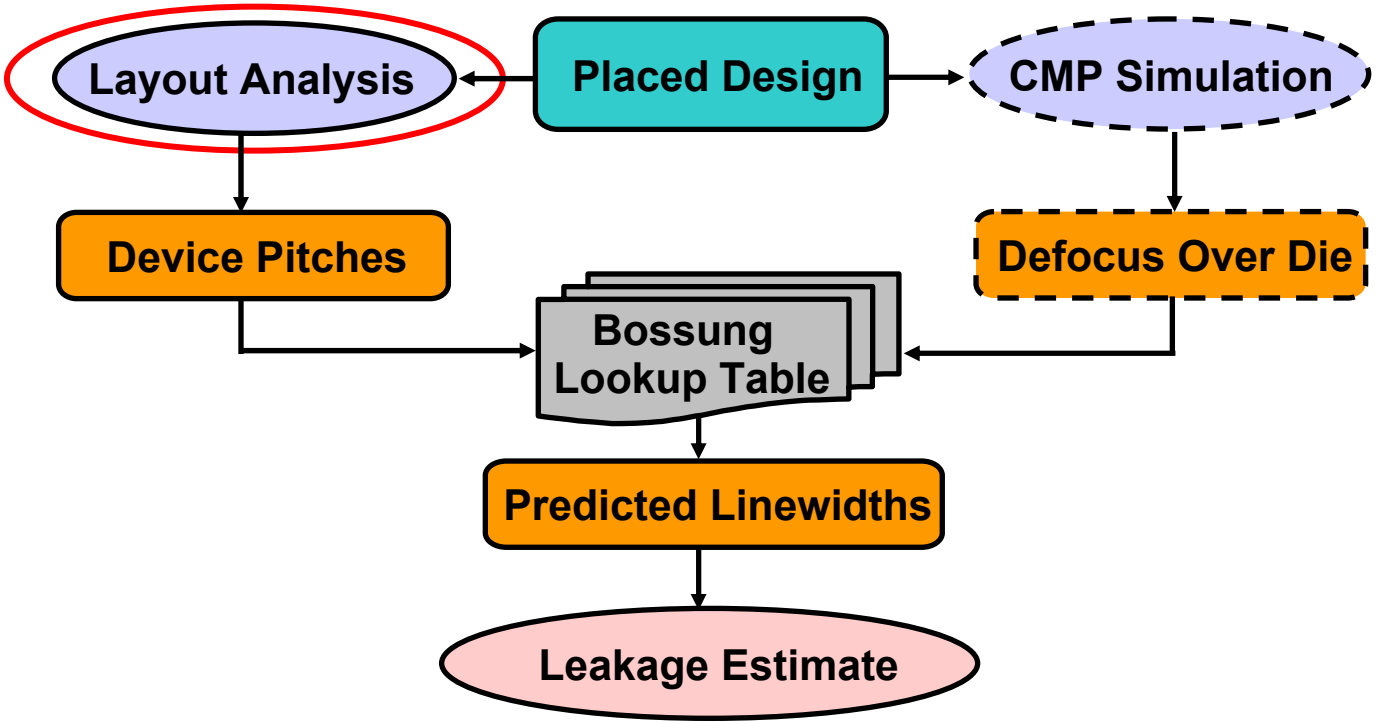
- Flow components
  - Bossung LUT creation
  - Pitch calculation
  - Cell leakage estimation

Bossung Lookup Table Creation



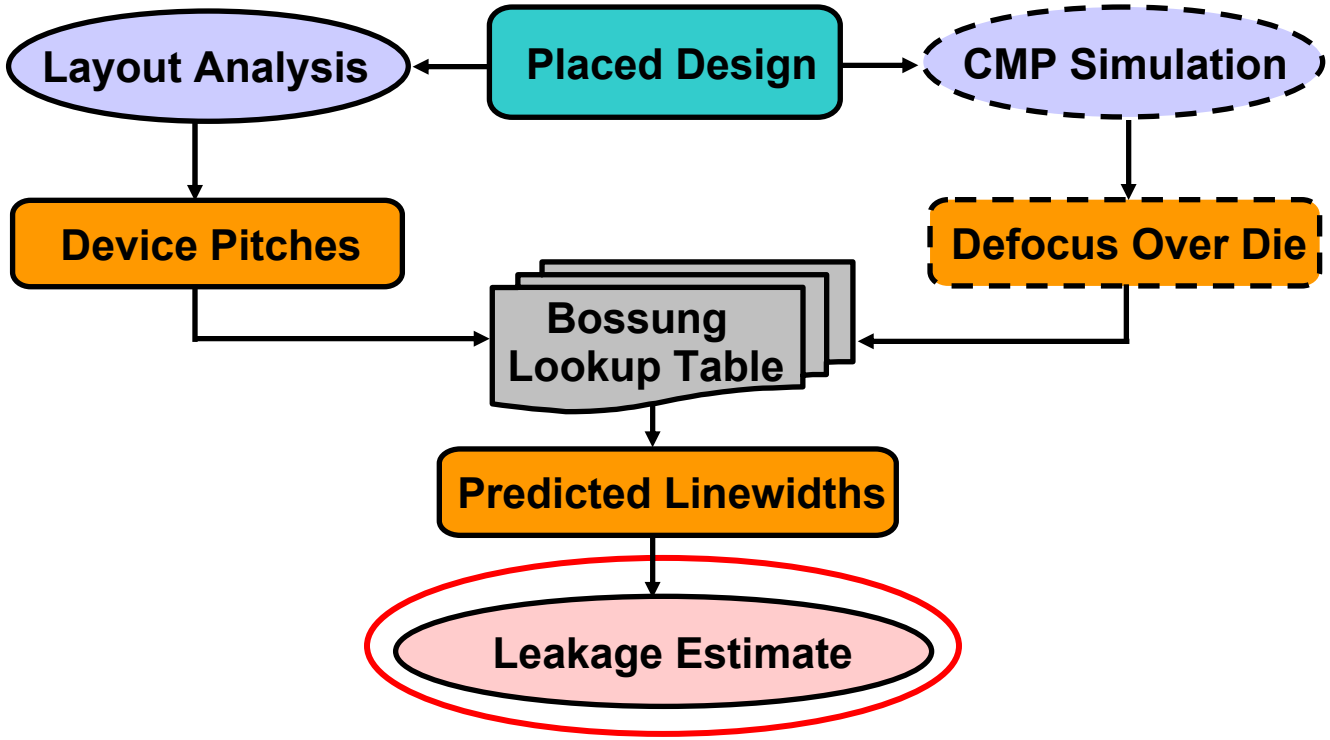
- Bossung LUT: predicts linewidth given pitch and defocus
  - Rows: pitch, Columns: defocus values, Entries: predicted linewidth
- Creation:
  - line-and-space patterns to simulate different line pitches
  - lithography simulation at different defocus values to predict linewidth
- Done once per process

Pitch Calculation



- Layout analysis: calculates pitch given placement and cell layouts
- Pitch calculated from:
  - Cell neighbor spacing and cell orientation from placement
  - Location of devices within cell from LVS information

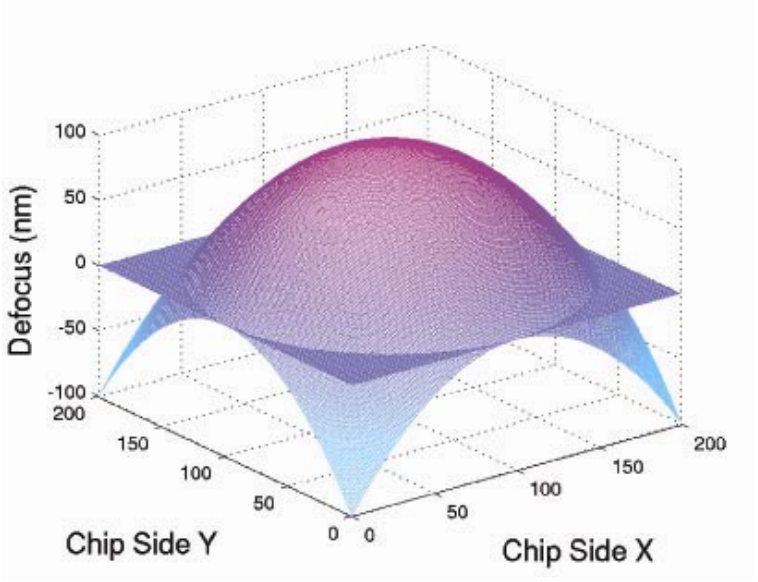
Cell Leakage Estimation



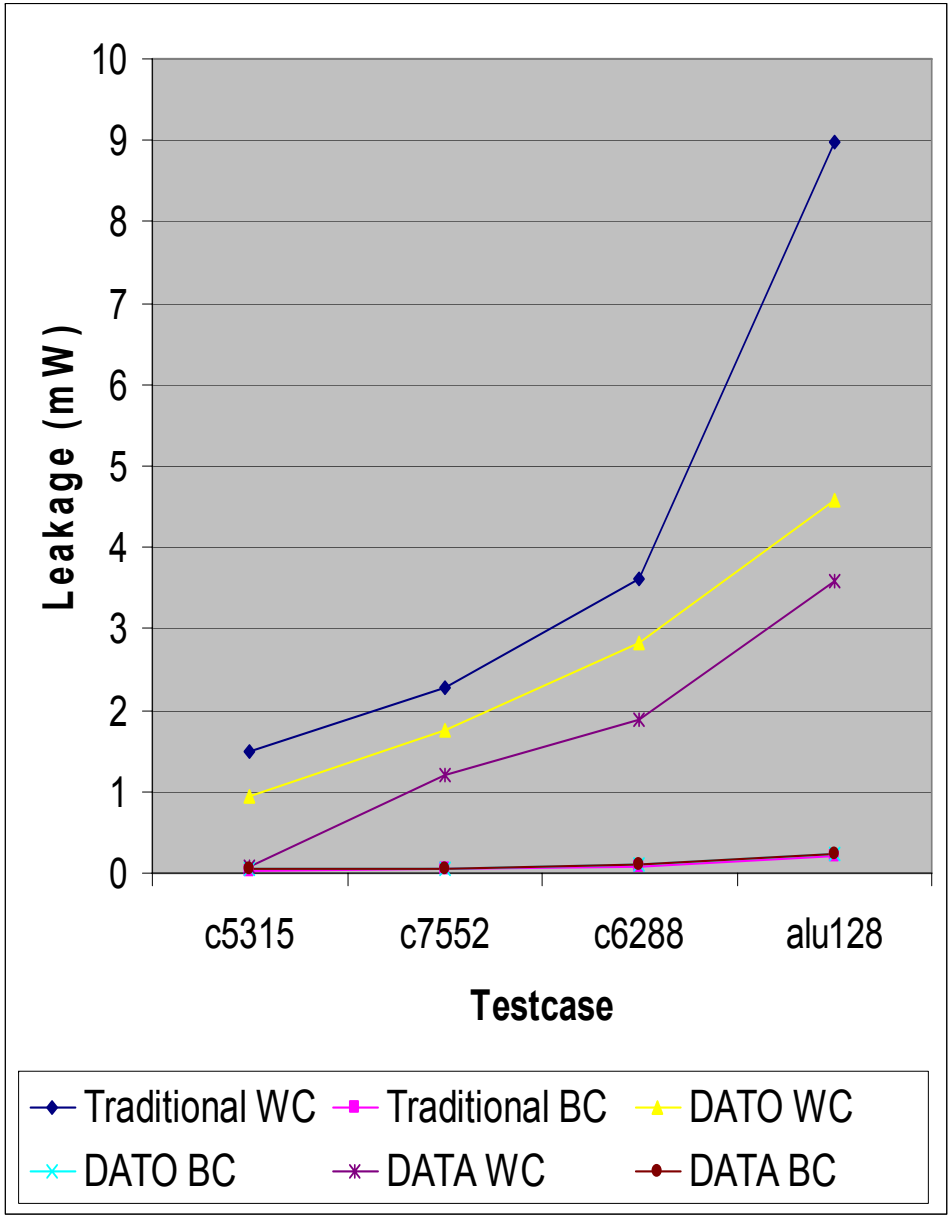
- Leakage estimation: calculates cell leakage from linewidths of devices in it followed by design leakage
- Approach:
  - Cell leakage for each input state estimated by finding leaking devices by logic simulation within cell
    - Leakage of stacked devices neglected
  - Device (NMOS and PMOS) leakage for a given gate length and width from table characterized with SPICE

Experimental Setup

- Testcases: *c5315* (2077 cells), *c6288* (4776 cells), *c7552* (3155 cells), *alu128* (11724 cells)
- Cell library (20 cell) characterization with *BPTM BSIM3* device models, *Synopsys HSPICE*, and *Cadence SignalStorm*
- Synthesis with *Synopsys Design Compiler* with tight delay constraints. Placement with *Cadence SoC Encounter*.
- OPC, litho-simulation and scattering-bar insertion with *Mentor Calibre* using industry-strength recipes for 100nm linewidth and 193nm stepper.
- Topography used: +100nm at die center, quadratically decreases to -100nm at die corners



Leakage Estimation Results



WC: Worst Case
BC: Best Case

DATO:
Defocus-Aware, Topography-Oblivious
Defocus Gaussian random with
μ=0nm, 3σ=200nm

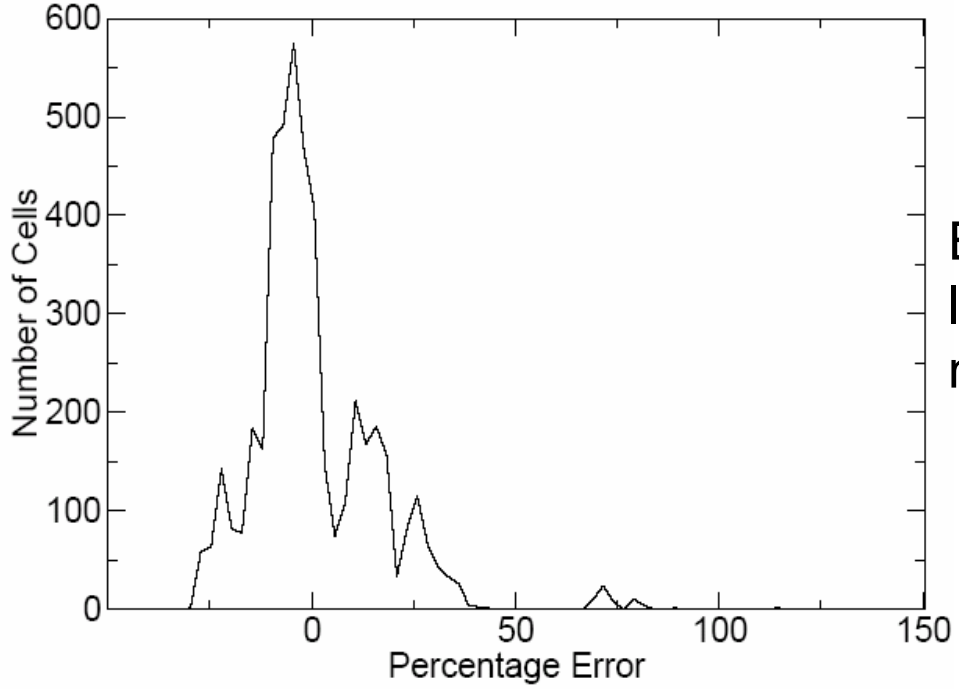
DATA:
Defocus-Aware, Topography-Aware
Defocus Gaussian random with
μ=predicted topography height
3σ=100nm

Spread Reduction

c5315: 56%
c7552: 49%
c6288: 49%
alu128: 62%

Per-Instance Leakage Estimation

- Ability to predict leakage for each cell instance



Error distribution of traditional leakage estimation for c6288 at nominal process corner

(Negative error → Traditional estimate is higher)

→Can drive leakage reduction techniques like V_{Th} assignment, input vector control, gate-length biasing

(optimize cells that are more leaky)

Gate-Length Biasing

- Proposed by us in DAC04 and TCAD06 to reduce leakage and its variability
- Key idea: exploit V_{Th} roll-off by increasing gate-length of non timing-critical devices
- Increasing gate-length of a cell, increases its delay, may cause other cells to become critical

  →Optimization problem: selection of cells to bias
- We proposed a sensitivity-based greedy optimization

> Sensitivity of cell p = $\xi_p = \Delta L_p \times s_p$
> ΔL_p : Leakage reduction of cell p upon biasing
> s_p : Timing slack of cell p after biasing it

- Bias cells in decreasing order of sensitivity
  - Requires sensitivity updates and timing violation checks

Defocus-Aware Gate-Length Biasing

- We add defocus-awareness to gate-length biasing
- Sensitivity-based greedy opt. in gate-length biasing

Sensitivity of cell p = $\xi_p = \Delta L_p \times s_p$
ΔL_p : Leakage reduction of cell p upon biasing
s_p : Timing slack of cell p after biasing it

- Defocus aware sensitivity function:

$\xi_p = \langle \Delta L_p \rangle \times s_p$
$\langle \Delta L_p \rangle$: Expected leakage reduction of cell p

- Expected leakage reduction computation:

$\langle \Delta L_p \rangle = \sum_t \langle \Delta L_{pt} \rangle$ $\langle \Delta L_{pt} \rangle$: Exp. leakage reduction of device t of cell p
$\Delta L_{pt} = f(l_{pt})$ l_{pt} : gate-length
$l_{pt} = g(D_{pt}, P_{pt})$ D_{pt} : defocus; P_{pt} : pitch
$\langle \Delta L_{pt} \rangle = \sum_t \sum_D f(g(D_{pt}, P_{pt})).P(D_{pt})$ P : probability defocus is D_{pt}

- We assume defocus (D) to be Gaussian random
  - Topography-oblivious: μ=0nm, 3σ=200nm
  - Topography-aware: μ=topography height, 3σ=100nm

Results

Leakage after traditional and defocus-aware gate-length biasing

Circuit	Traditional Gate-Length Biasing			Defocus-Aware Gate-Length Biasing			Leakage Reduction		
	WC (mW)	Nom (mW)	BC (mW)	WC (mW)	Nom (mW)	BC (mW)	WC (%)	Nom (%)	BC (%)
c5315	3.948	0.855	0.326	3.838	0.838	0.321	2.78	2.01	1.63
c6288	9.363	1.923	0.730	8.958	1.861	0.712	4.33	3.23	2.56
c7552	6.678	1.350	0.507	6.212	1.280	0.485	6.98	5.17	4.21
alu128	21.258	4.908	1.907	19.968	4.663	1.827	6.07	4.99	4.19

- Optimization for nominal corner and topography mentioned earlier
- Modest leakage reductions from 2-7%
- 10% optimization runtime increase

Outline

- ✓ Introduction
- Systematic Variation-Aware Techniques
  - ✓ Defocus-Aware Leakage Analysis and Optimization
  - ➢ Detailed Placement for Leakage Optimization
  - Aberration-Aware Timing Analysis
- Utilizing STI Stress in Delay Analysis and Optimization
- Other Research Contributions
- Conclusions

Does Placement Affect Power?

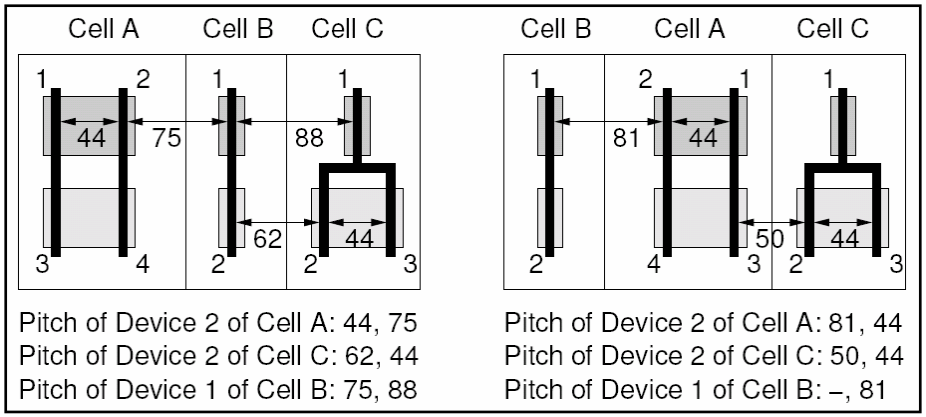
- Conventional wisdom: Placement changes wirelength which affects dynamic power
  - Longer wires → more C → more CV^2 power
  - Longer wires → larger loading → more internal power
- We show placement affects leakage power
  - Placement selects neighbors

    → neighbors of a cell determine the patterns to print

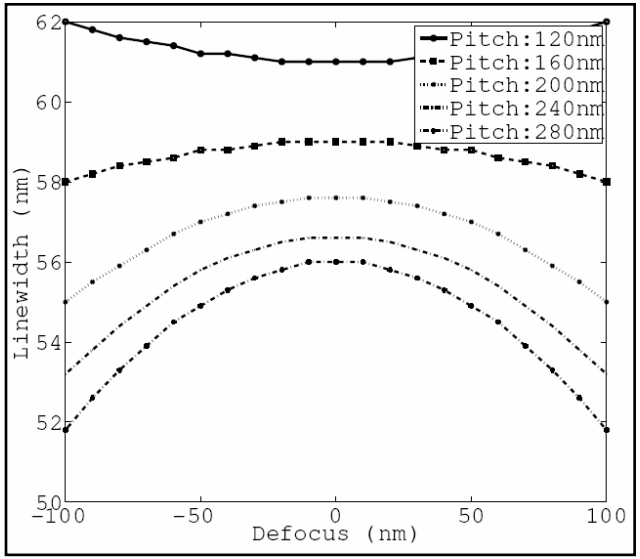
    → pattern-dependent lithography errors affect on-silicon gate-length

    → leakage depends on gate-length
- Publication: ISLPED07 (to appear)

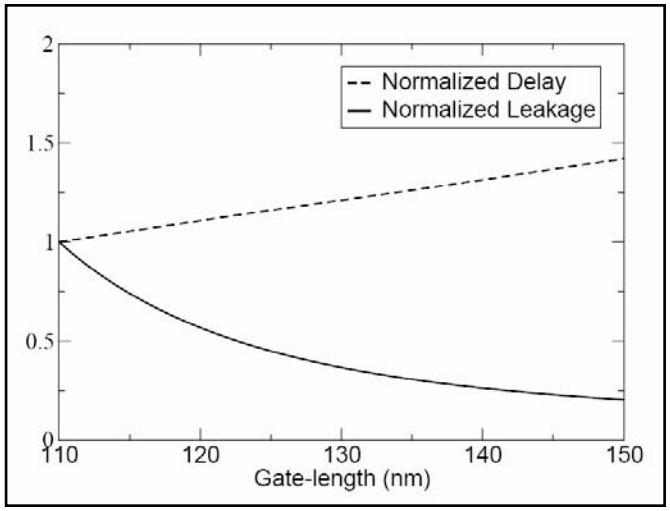
Placement → Leakage



Placement affects device pitches



Pitch systematically affects gate length



Gate length affects leakage

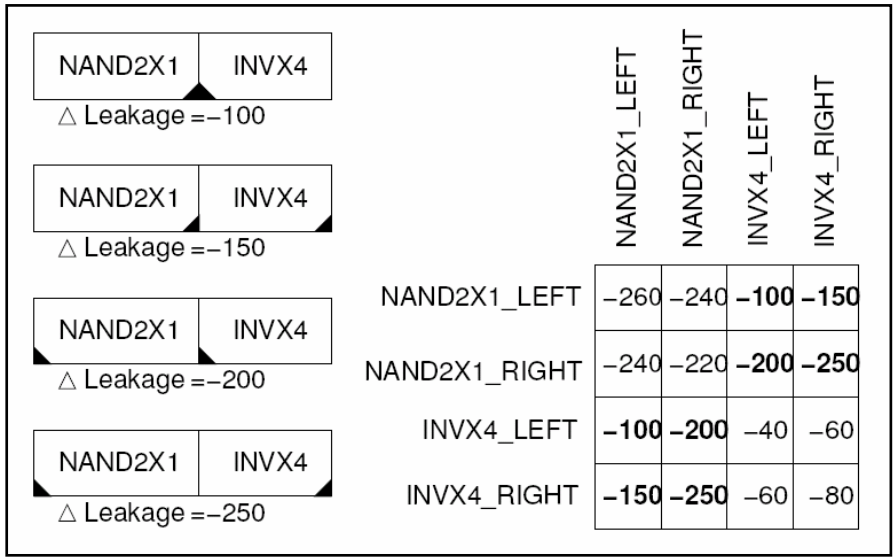
Placement affects leakage

Detailed Placement for Leakage

- *Detailed Placement*: refinement step which performs small-range perturbations to generate a new optimized placement
  - Typically for wirelength and timing
- Affects pitches (and consequently leakage) by three knobs:
  - Neighbor selection
  - Orientation
  - Cell-to-cell spacing
- Our approach:
  - Step 1: Capture impact of placement on leakage
  - Step 2: Utilize information in detailed placement

Capturing Placement Impact on Leakage

- Goal:
  - Construct a matrix of leakage cost



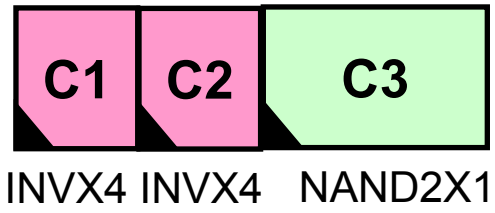
	NAND2X1_LEFT	NAND2X1_RIGHT	INVX4_LEFT	INVX4_RIGHT
NAND2X1_LEFT	–260	–240	**–100**	**–150**
NAND2X1_RIGHT	–240	–220	**–200**	**–250**
INVX4_LEFT	**–100**	**–200**	–40	–60
INVX4_RIGHT	**–150**	**–250**	–60	–80

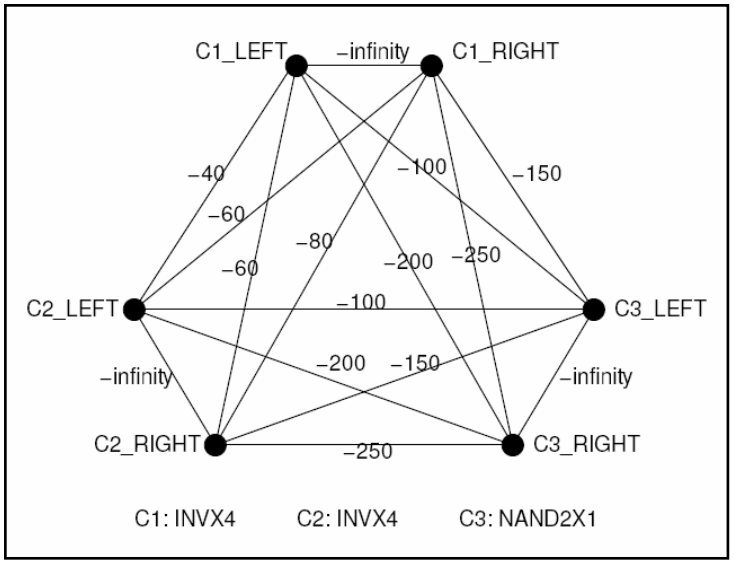
- Our approach:
  - Compute pitch when two cells abut
  - Predict linewidth from computed pitch from Bossung plot
  - Calculate device leakage from linewidths
  - Calculate cell leakage from leakage of its devices
- Alternative approaches
  - Litho-simulate abutted cells → calculate device and cell leakages
  - On-silicon measurements

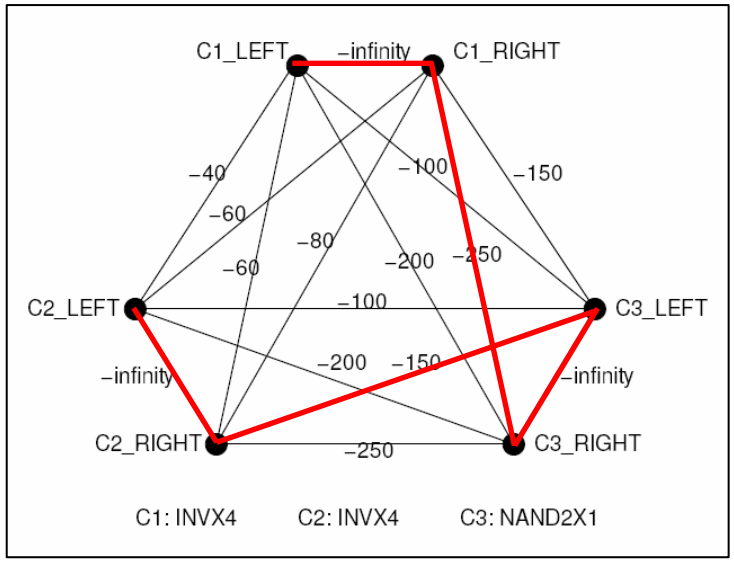
Single-Row, No-Whitespace Optimization

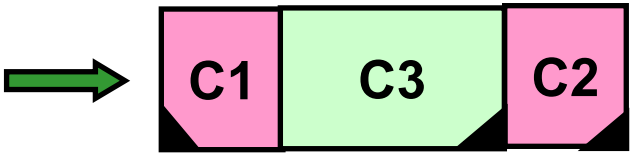
- Optimization done in small windows (single row)
  - Design partitioned into windows, cells in each window optimized
- Goal: order cells and select the ones to flip to minimize leakage cost
- We transform the problem to the famous traveling salesman problem
  - Node ≡ each side of each cell (so #nodes = 2 × #cells)
  - Complete graph with edge weight = leakage cost matrix entry
- Tour gives ordering and selects cells to be flipped
  - All nodes (≡ each side) ordered to minimize cost (= sum of leakage cost when two edges touch)
  - Additional constraint: two edges of the same cell must occur consecutively in tour → we assign -∞ weight
- We use multi-fragment greedy heuristic to solve TSP

Illustration of the Optimization



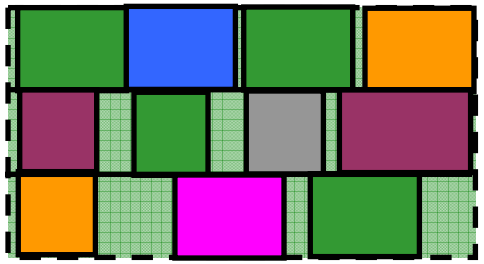






Whitespace and Multiple Rows

- Fillers are inserted in whitespace
  - Approach:
    - Compute number of white spaces (=N)
    - N FILLx1 cells can be inserted → Add N vertices to TSP
    - Merge consecutive fill cells (e.g., 2 FILLx1 → FILLx2)
- Multiple rows
  - We exhaustively partition the set of cells into rows
  - Number of partitions can be extremely large
    - Prune number of partitions using row capacity constraints
    - Best single-row results cached

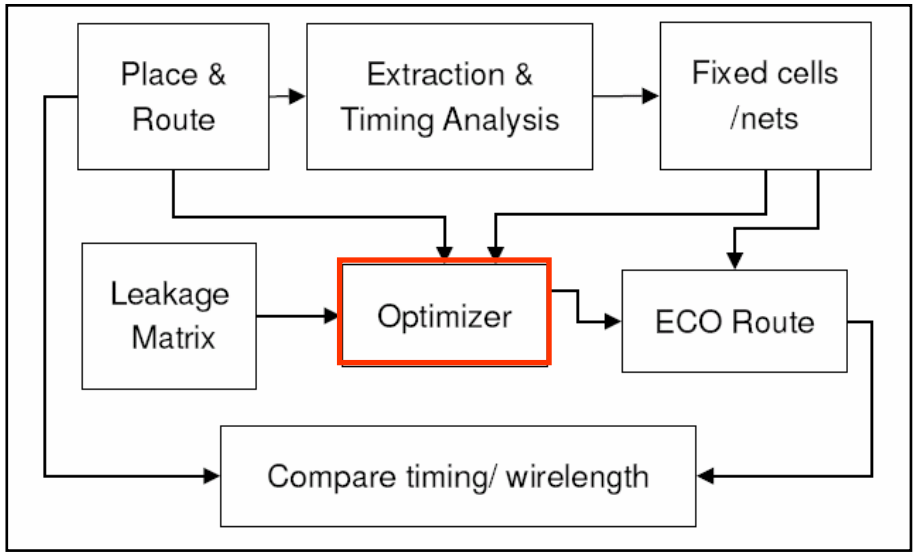


Minimizing Wirelength and Timing Impact

- Wirelength increase bad
  - Increases congestion, degrades routability
  - Increases dynamic power
- Smaller window size causes smaller wirelength increases

  → we increase window size progressively in phases
  - accept solution of a phase if it improves upon that of last phase
- Timing impact minimized
  - Critical cells marked as don't-touch
  - All cells connected to nets of critical cells also marked as don't-touch
  - Incremental routing performed with nets of critical cells marked as don't-touch

Experimental Study



- Technology: 65nm dual-V_{Th}
- Tools: RTL Compiler, SoC Encounter, OpenAccess
- Testcases: AES (80% util), AES (85% util), DES (73% util)

Results

Final Window Size	Proposed Technique				
	Leakage Reduction (%)	Wirelength Impact (%)	Max. Frequency Impact (%)	Dynamic Power Impact (%)	Runtime (s)
4μ×1 row	2.91	+0.72	+0.33	+0.13	5.18
6μ×1 row	4.16	+2.39	-0.41	+0.31	8.72
8μ×1 row	5.08	+4.94	-1.18	+0.45	14.64
4μ×2 rows	5.21	+3.86	+0.50	+0.36	37.90
6μ×2 rows	6.41	+8.14	-0.49	+0.61	301.35
2μ×3 rows	4.02	+2.08	+0.46	+0.25	23.83
4μ×3 rows	6.44	+7.12	-0.41	+0.67	1964.09
6μ×2 rows†	7.45†	+12.33†	-5.62†	+0.92†	284.34†

Results for testcase AES (80% utilization)

- † identifies results with our measures to minimize wirelength and timing bypassed
- As expected, larger windows
  - improve leakage, but;
  - increase wirelength and dynamic power.

Outline

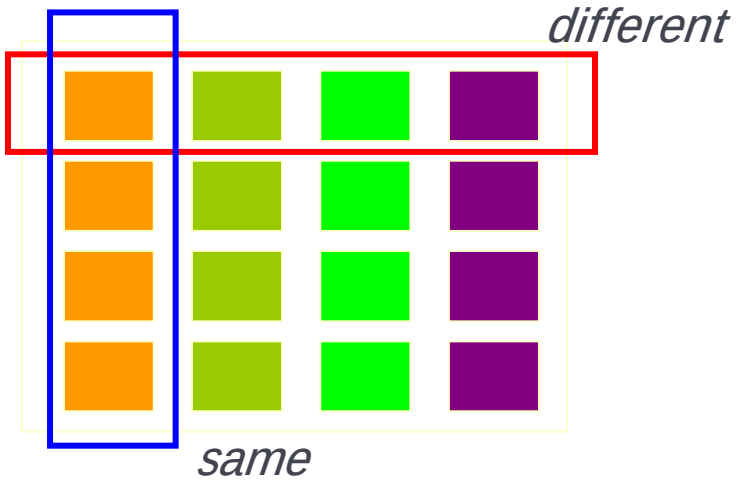
- ✓ Introduction
- Systematic Variation-Aware Techniques
  - ✓ Defocus-Aware Leakage Analysis and Optimization
  - ✓ Detailed Placement for Leakage Optimization
  - ➢ Aberration-Aware Timing Analysis
- Utilizing STI Stress in Delay Analysis and Optimization
- Other Research Contributions
- Conclusions

Lens Aberration

- Lens aberrations: image distortions due to imperfect lens
  - Variety of effects on lithographic imaging → shifts in image position, image asymmetry, reduction of process window
- Zernike aberration coefficients
  - represent wavefront aberrations (36 terms)
  - Coma → image asymmetry, pattern-dependent image shift
  - Astigmatism → CD difference between horizontal, vertical lines
  - Spherical → changes best DOF between dense/isolated patterns
- *Lens field*: wafer area exposed in one shot
- Aberration (and Zernike's coefficients) vary with position in the lens field → gate length varies with position in lens field

Impact on Gate Length (CD)

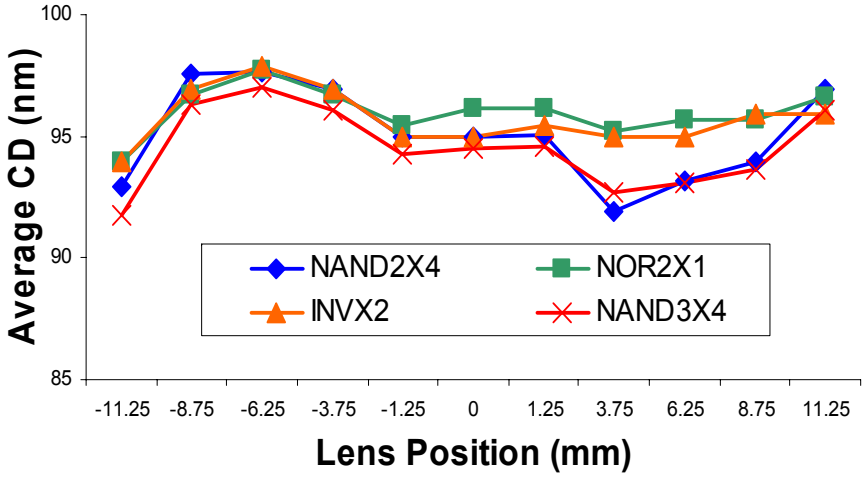
- Slit scans from one side of the field to another
  - Zernike coefficients vary with position in the lens field
    - → CD varies along horizontal direction
    - → CD constant along vertical direction



Reticle (mask) map

- Impact on average CD varies with location in lens field

  → Average CD 93nm – 97nm for NAND2X4

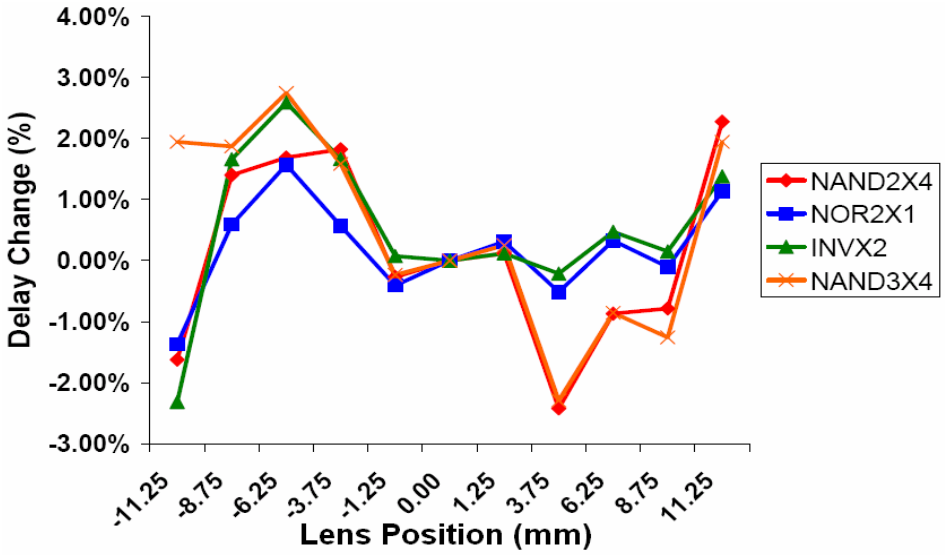


- Different devices in a cell affected differently → CD skew induced
- Gate delay depends on CD → Gate delay depends on position in lens field

Impact on Gate Delay

- Impact on average cell delay varies with location in lens field

→ NAND2X4 delay varies between -2% and 2%



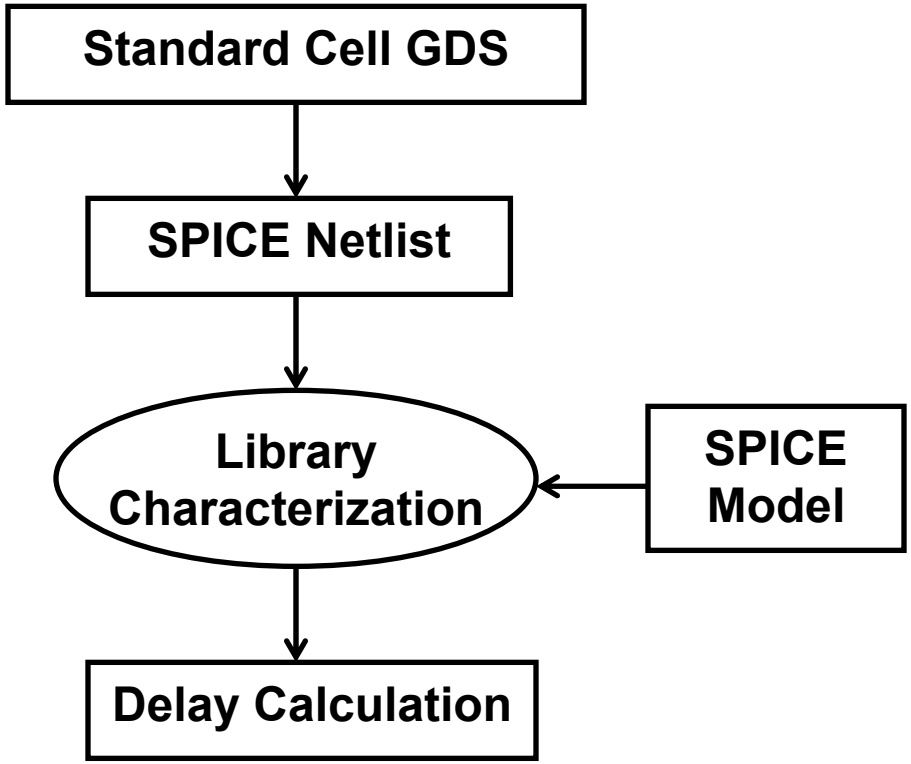
- Input capacitance and slews increase with CD

→ **Predictable "fast" and "slow" regions due to aberration**

→ **Account for delay variations induced by aberration in timing analysis**

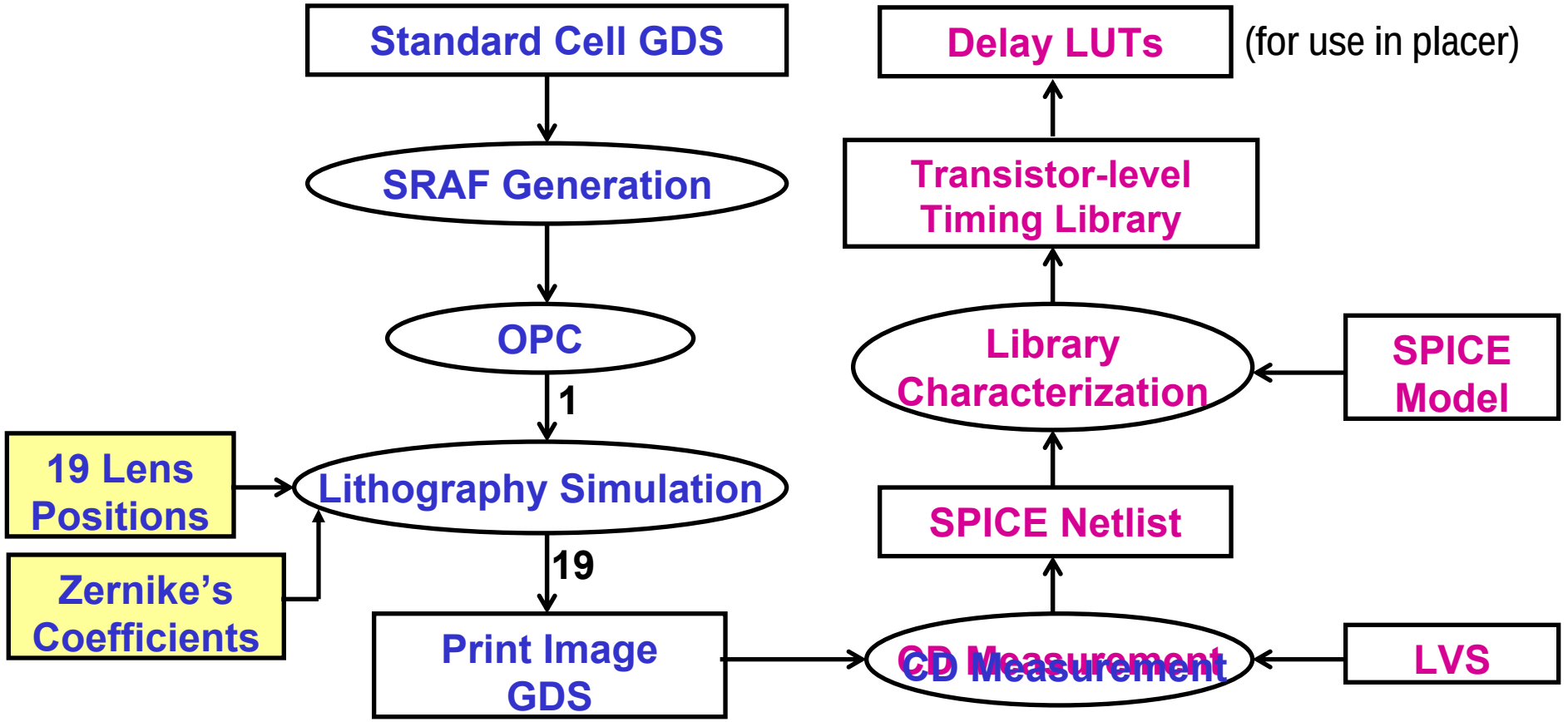
→ **Also: place setup-critical cells in the fast regions, and place hold-critical cells in the slow regions (DATE06)**

Standard Timing Analysis Flow



- Problem: With aberration, two instances of the same master should have different timing models !

Aberration-Aware Timing Flow



- Cell variant created in library for each lens position
- Two main steps:
  - **Construct litho models → get simulated gate CDs of each instance**
  - **Generate timing library models of all masters for different locations**
- Timing library used along with placement in STA

Aberration-Aware Timing vs. Traditional Timing

- Testcases

Design	Utilization (%)	Chip Side (mm)	#Cells	#Nets
AES	60	0.50	17304	17465
JPEG	60	1.41	118321	125036

- Timing analysis results

Circuit	Traditional STA Delay (ns)	#Columns	Aberration-Aware STA	
			Delay (ns)	Δ (ps)
AES	2.845	1	2.790	55
		2	2.827	18
		3	2.840	5
JPEG	3.727	1	3.634	93
		2	3.713	14
		3	3.699	28

Modest improvement in analysis

Maybe useful for large, high-speed designs

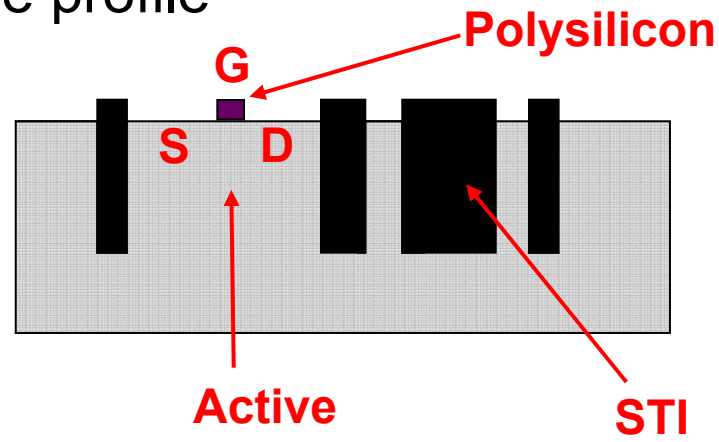
#columns: number of chip copies placed horizontally in the lens field
Larger designs → fewer #columns

Outline

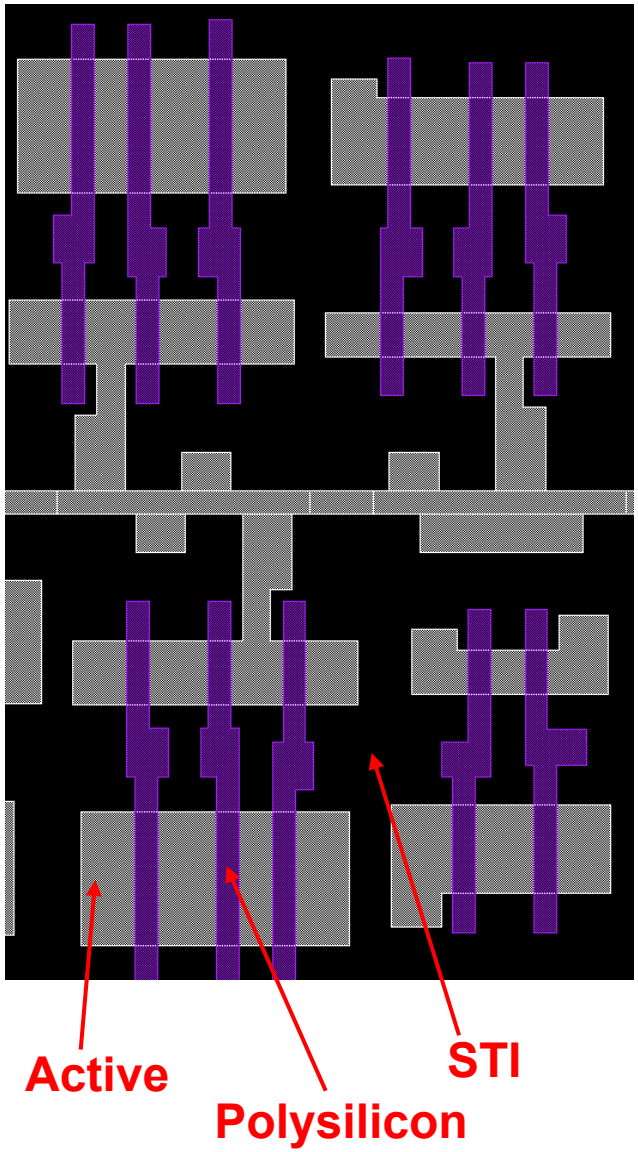
- ✓ Introduction
- ✓ Systematic Variation-Aware Techniques
  - ✓ Defocus-Aware Leakage Analysis and Optimization
  - ✓ Detailed Placement for Leakage Optimization
  - ✓ Aberration-Aware Timing Analysis
- ➢ Utilizing STI Stress in Delay Analysis and Optimization
- ▪ Other Research Contributions
- ▪ Conclusions

What Is STI Stress?

- STI surrounds devices to electrically isolate them
- Side profile



- STI pushes active region inwards
  - Smaller active region gets pushed more
    → more stress
  - Larger STI region pushes more
    → more stress
- Stress enhances PMOS mobility and performance, degrades NMOS mobility and performance



Present BSIM Stress Modeling

- Stress partially modeled since BSIM 4.3.0
  - Parameters SA, SB, SC added
  - Parameters capture gate to STI separation

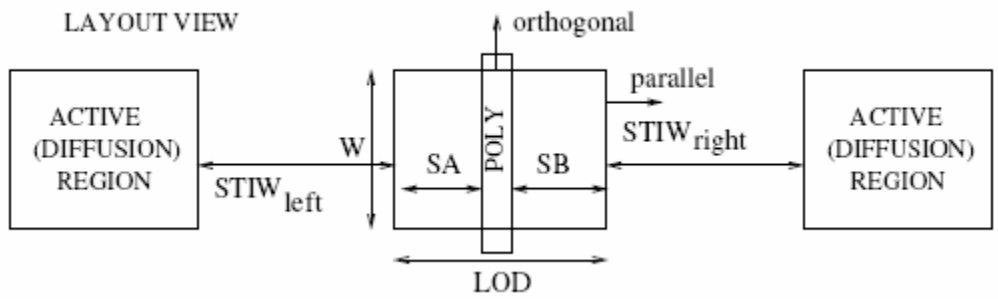
```
.subckt  INVX1  A Z
 MM1 D G S B  NCH SA=0.2u SB=0.2
 MM2 D G S B  PCH SA=0.19u SB=0.19u
   :
.ends
```

```
.model  NCH    NMOS (
  *Other stress parameters defined
   :
)
```

- Stress effect due to STI width (STIW) not modeled
  - STI width determined by placement
    → Cannot be calculated at cell netlist level in characterization
    → New flow needed to annotate STIW information from placement
  - Smaller in extent than due to gate-active edge separation
- We multiply BSIM mobility by our correction factor *mob*
- We construct a *mob* model using Sentaurus process simulations
- We plug in our *mob* model as a function of STI width parameters set from placement

STI Stress Compact Modeling

- Process simulation until gate deposition using Synopsys Sentaurus
- Simulations performed for different STI widths on left and right, SA, SB



- Simulated stress converted to mobility using [Smith65] and normalized
- Mobility models derived using curve-fitting to simulated data

NMOS

$$MOB_{L,R} = \zeta + (1 - (STIW_{L,R}/2)^{\alpha})/S\{A,B\}^{\beta}$$
$$MOB = [MOB_L * MOB_R]^{0.26}$$

PMOS

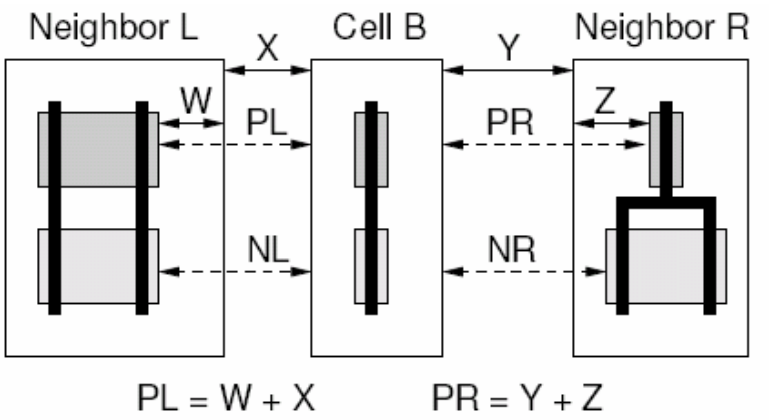
$$MOB_{L,R} = \zeta + ((STIW_{L,R}/2)^{\alpha})/S\{A,B\}^{\beta}$$
$$MOB = [MOB_L * MOB_R]^{0.14}$$

	ζ	α	β
NMOS	1.03	0.076	0.48
PMOS	0.49	0.48	0.57

- Several other STI heights and stress liners simulated → STI width changes by <10% → STI width effects significant for most processes

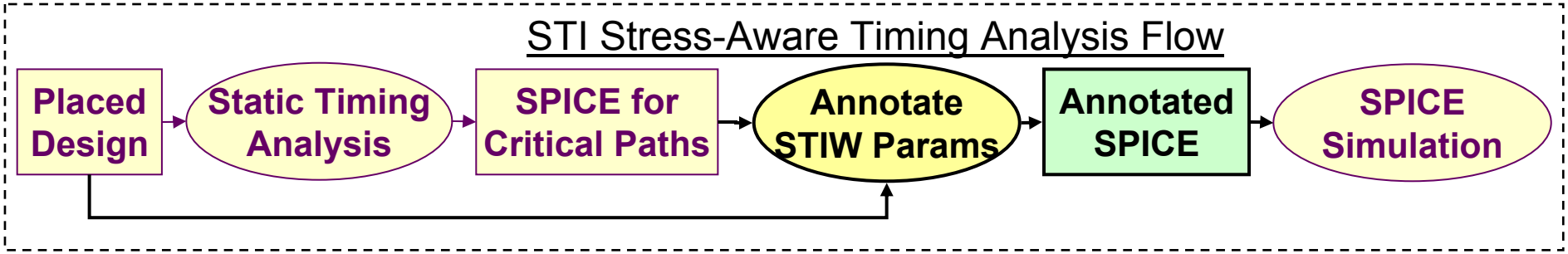
STI Stress-Aware Timing Analysis

- Added STIW parameters:
  - PL, PR, NL, NR
  - PL: Distance between cell boundary and positive active region edge of left neighbor




```
* Critical path 00001
 X01 N1 N2 INVX1  PL=0.08u PR=4.08u NL=0.06u NR=4.06u
 X02 N2 1 N2 NAND2X1  PL=5.0u PR=5.0u NL=5.0u NR=5.0u
 X03 N3 N4 BUFFX1  PL=2.1u PR=5.0u NL=2.04u NR=5.0u
   :
```

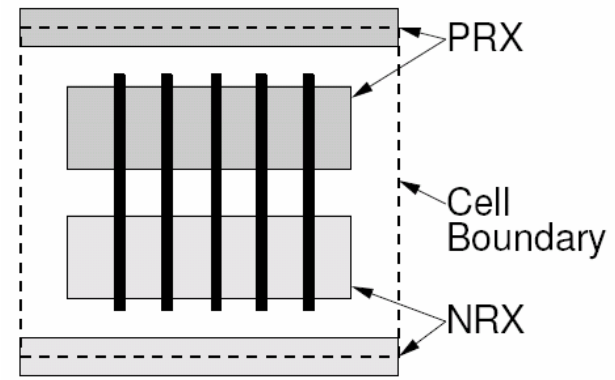
```
.subckt  INVX1  A Z
 .param PMOB = Our_PMOS_Model (PL, PR, NL, NR)
 .param NMOB = Our_NMOS_Model (PL, PR, NL, NR)
 MM1 D G S B  NCH SA=0.2u SB=0.2  MOB=NMOB
 MM2 D G S B  PCH SA=0.19u SB=0.19u  MOB=PMOB
   :
.ends
```



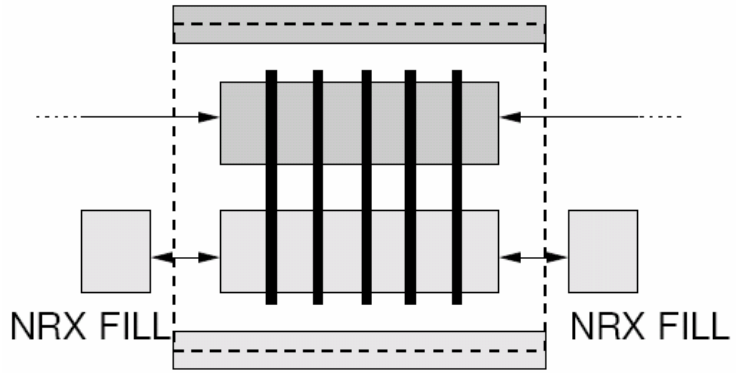
- We use this flow to evaluate our optimization

Optimization: Exploiting Stress for Performance

- Goal: engineer STIW such that stress speeds PMOS and NMOS
- High stress improves PMOS → increase STIW for PMOS
- Low stress improves NMOS → decrease STIW for NMOS
- Knobs to alter STIW
  - Active layer fill insertion
  - Placement perturbation
- Active layer fill insertion



Generic Cell Layout

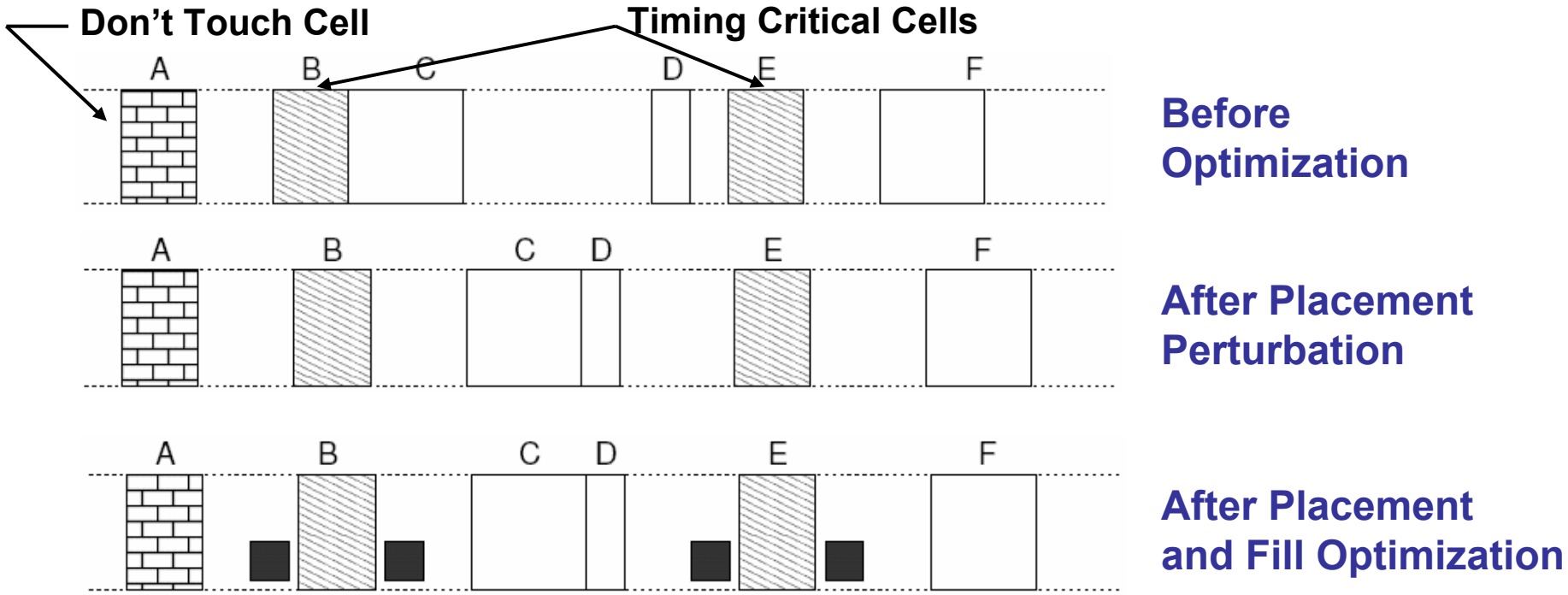


After Active Layer Fill

- Fill inserted next to N-diffusion (NRX) → Small STIW for NMOS
- No fill next to P-diffusion (PRX) → Large STIW for PMOS

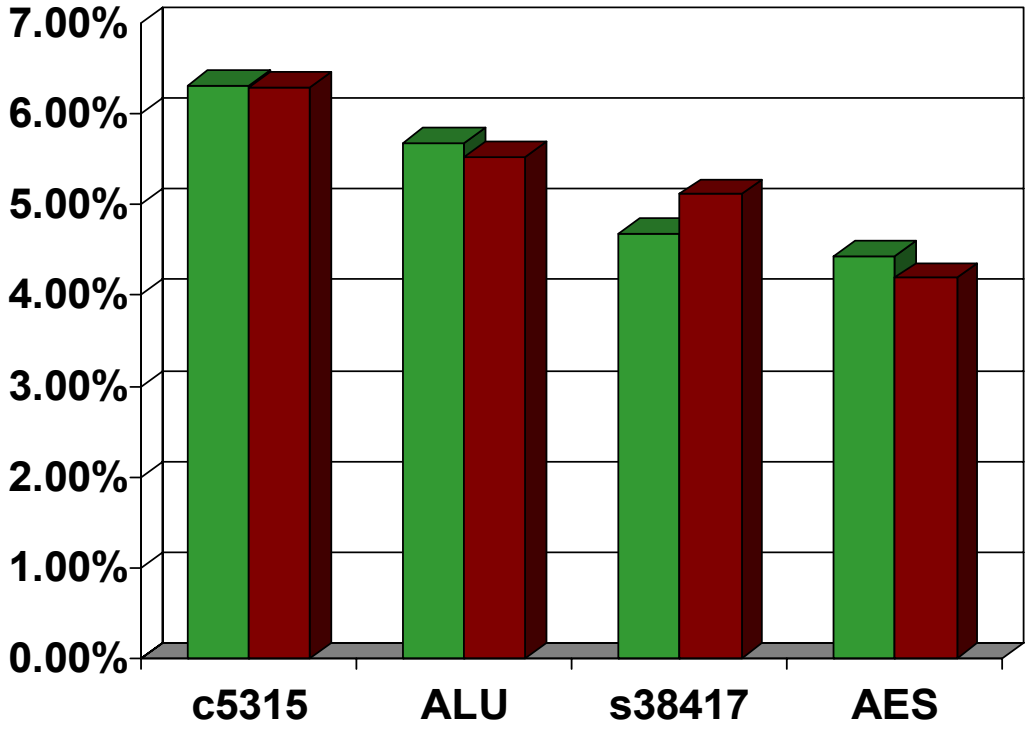
Placement Perturbation

- Increase spacing for timing critical cells
  - Increases PMOS STIW → better PMOS speed
  - Create spacing for active fill for NMOS that was previously not possible → better NMOS speed
- Minimize adverse timing impact of placement perturbation
  - Don't modify locations of critical cells, their routes, clock tree, etc.



Stress-Aware vs. Traditional Timing Analysis

- Circuit-level stress-aware vs. traditional timing analysis
  - Traditional timing analysis worst-cases stress effects for correctness
  - Stress-aware analysis models stress → correct and less pessimistic

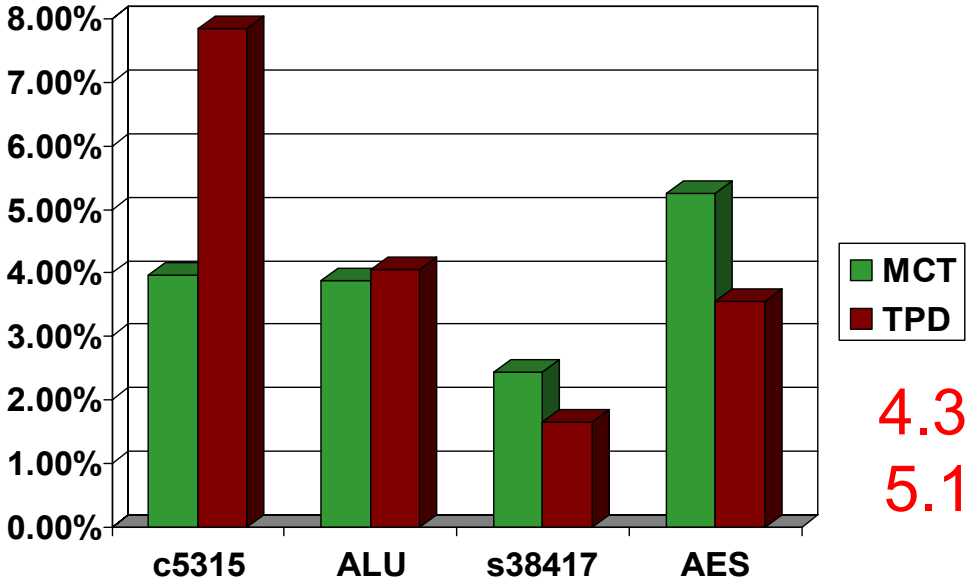


MCT: Minimum cycle time

TPD: Top paths delay (= sum of delays of top 100 critical paths)

5.75% smaller MCT on average

Results: Delay Optimization

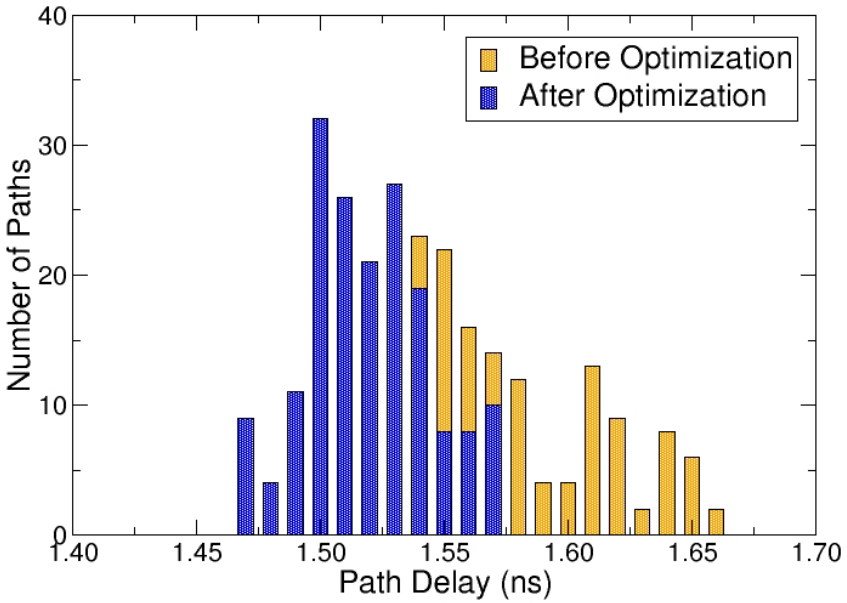


Smaller delay reduction for *s38417* because >50% cells are flops and marked don't-touch.

Negligible wirelength increase (<0.67%)

4.37% average reduction in MCT

5.15% average reduction in TPD



Path delay histogram for test case AES

Outline

- ✓ Introduction
- ✓ Systematic Variation-Aware Techniques
  - ✓ Defocus-Aware Leakage Analysis and Optimization
  - ✓ Detailed Placement for Leakage Optimization
  - ✓ Aberration-Aware Timing Analysis
- ✓ Utilizing STI Stress in Delay Analysis and Optimization
- ➢ Other Research Contributions
- ▪ Conclusions

Other Contributions Under This Theme

- Gate-length biasing
  - Motivation: leakage and its variability are critical concerns
  - Key idea: exploit V_{Th} roll-off by increasing gate-length of non timing-critical devices
  - Reduces leakage and its variability significantly
- STI fill for CMP
  - Motivation: Imperfect CMP causes device failure, device latch-up, and leakage. Expensive reverse etchback used to rectify
  - High nitride density and low oxide-density variation addresses above shortcomings
  - Oxide deposited over nitride with a slanting sidewall → nitride features determine nitride and oxide density
  - Key idea: size and shape nitride features to control nitride and oxide density
  - Proposed fill insertion to maximize nitride density and minimize oxide density variation
  - Results from CMP simulation showed superior post-CMP topography and planarization window
- Impact of floating fill on interconnect capacitance
  - Motivation: floating fill affects capacitance of wires and no reliable full-chip extraction methods exist
  - Studied impact of floating fill on neighboring interconnects on the same layer
  - Performed field solver simulations to understand the capacitive impact of different fill sizes and configurations
  - Proposed fill insertion guidelines to reduce capacitive effect

Conclusions

- DFM: measures taken in design to enhance yield
- Focus of my work: manufacturing-aware physical design techniques to increase yield by:
  - Reducing process variations
  - Improving design robustness
  - Accounting for systematic variations in analyses and optimizations
- Looking forward
  - Manufacturing technology will improve, but process variability as a percentage will not reduce → novel DFM methods will be needed
  - Several challenges exist to adoption of novel DFM methods (e.g., acquisition of variational data)
  - Traditional DFM will continue to be crucial
  - Techniques to reduce variability and enhance robustness will be deployed first, followed by statistical and systematic variation-aware methods.

Thank You!

- Questions?

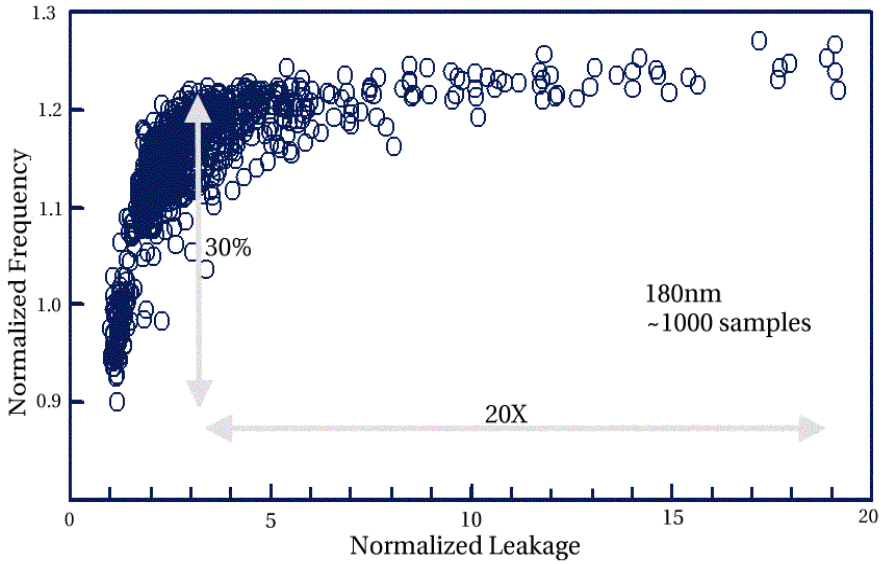
Backup

- Gate-Length Biasing for Leakage Control

Leakage and Leakage Variability

- Contribution of leakage to total power increasing
- Near-exponential dependence of leakage on gate-length

  → Even small gate-length variability → Large leakage variability



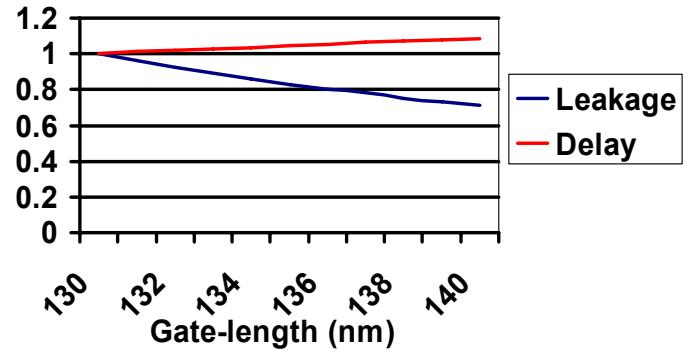
Proposed gate-length biasing to control leakage and its variability.

Publications:

- Gate-biasing in *TCAD06* and *DAC04*
- Its impact on V_{th} selection in *ISQED06*

Gate-Length Biasing

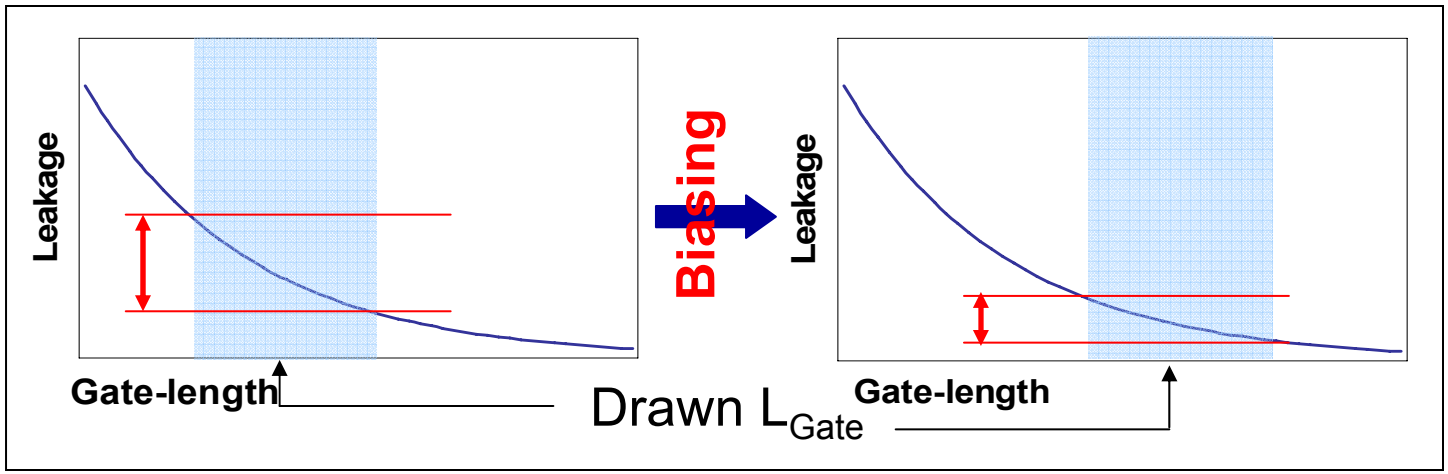
- Key Idea 1: *Slightly* increase (bias) the L_{Gate} of devices
- Reduce leakage?



Variation of leakage and delay (each normalized to 1) for an NMOS device in an industrial 130nm technology

Impact on Leakage and Delay

- Key Idea 2: Bias only non- timing-critical devices → No loss in circuit performance
- Reduce leakage variability?

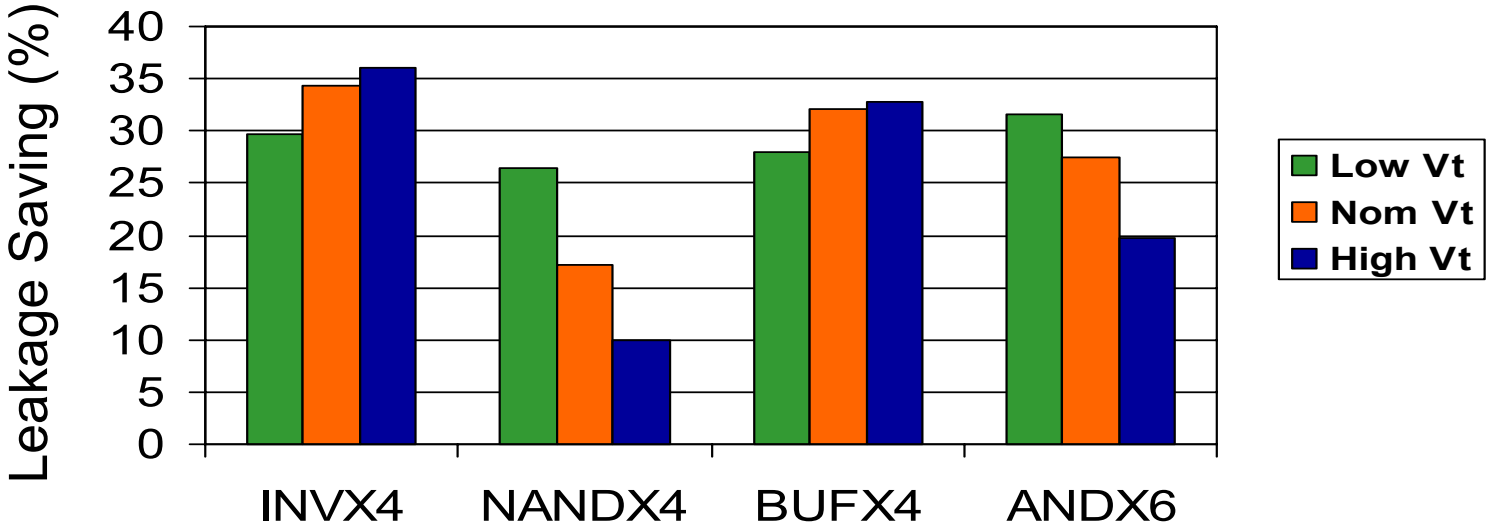


Impact on Leakage Variability

How Much to Bias?

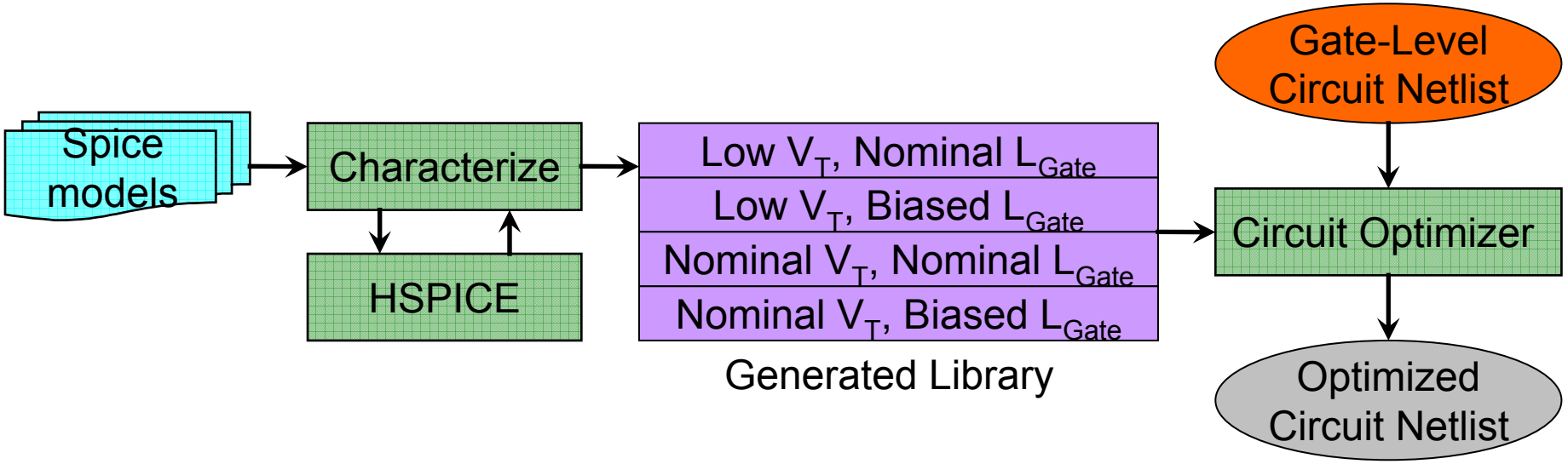
- We propose small bias < layout grid resolution
  - Little reduction in leakage beyond 10% bias while delay degrades linearly
  - Preserves pin compatibility: layout swappable

    → Technique applicable as post-P&R step
  - No additional process steps
- Cell-level leakage reduction



On biasing from 130nm to 136nm

Methodology



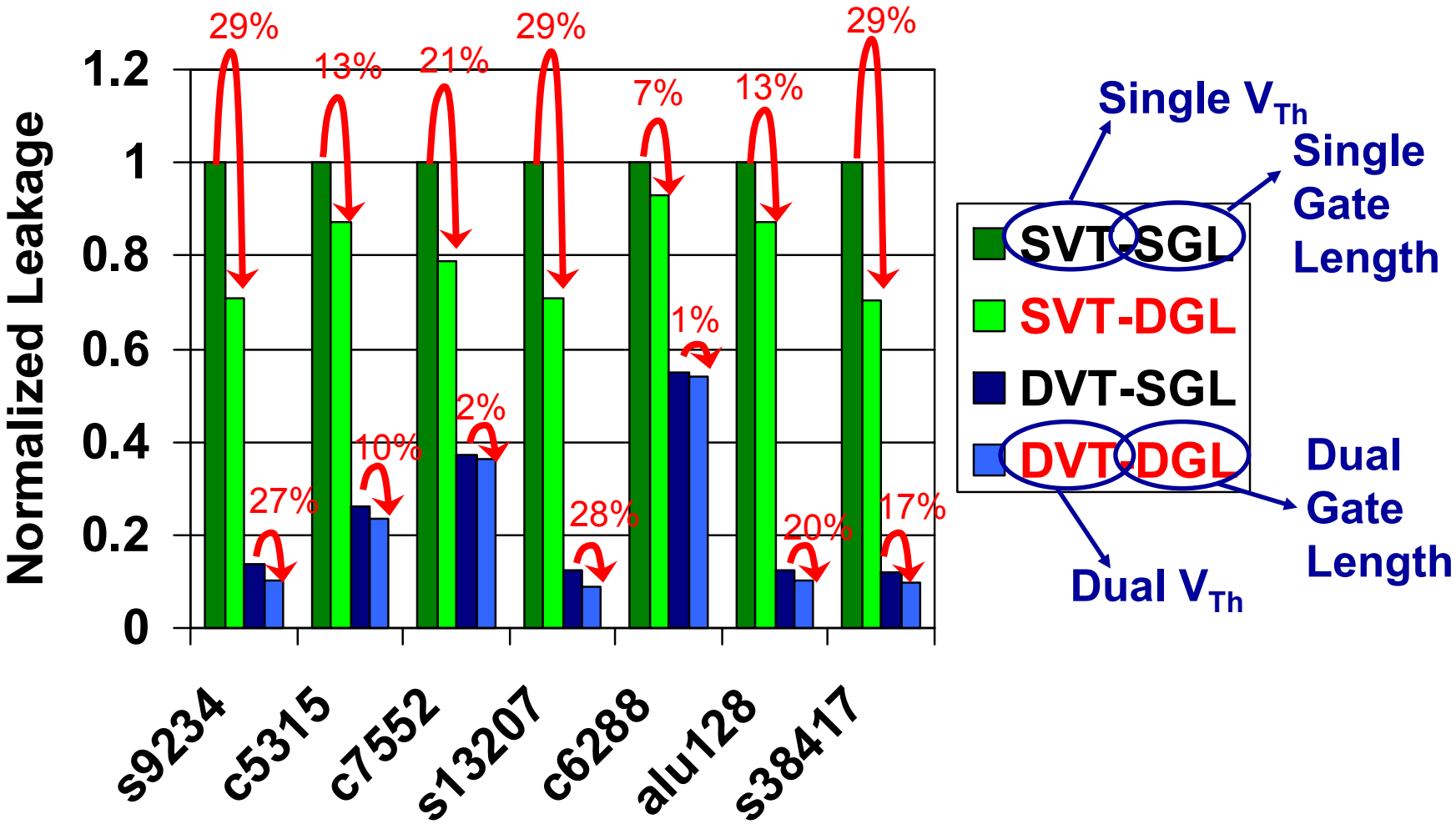
- Extend a cell library with biased L_{Gate} versions of all cells
- Optimize circuit for leakage by using biased L_{Gate} versions for non-critical *cells*

Leakage Optimizer

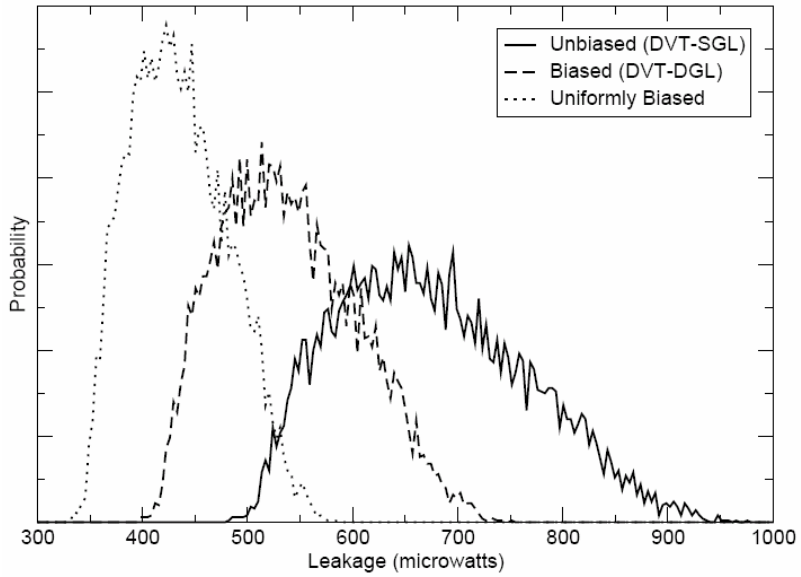
- Off-the-shelf sizing tools (e.g., SNPS DC) do not work well
  - Tradeoffs involved different from traditional cell (width) sizing
- First approach: sensitivity-based downsizing
  - Start with a netlist with no timing violations
  - Downsize (i.e., delay increases, leakage decreases) iteratively
    - Order by sensitivity
    - Check timing after each or a couple of downsizing moves
- Enhancements
  - Transistor-level optimization
  - Bulk moves (=simultaneous downsizing of a group of cells)
    - Cells in different pipeline stages
    - Cells at same topological level
- Lagrangian relaxation
  - Constraints and objectives expressed as convex functions
  - Iteratively improving solution (modeling inaccuracy improves each iteration)
  - Better quality but more runtime

Results: Leakage Reduction

- Dual-V_{Th} is the mainstream leakage reduction technique
- Assess leakage reduction with and without dual-V_{Th} technique



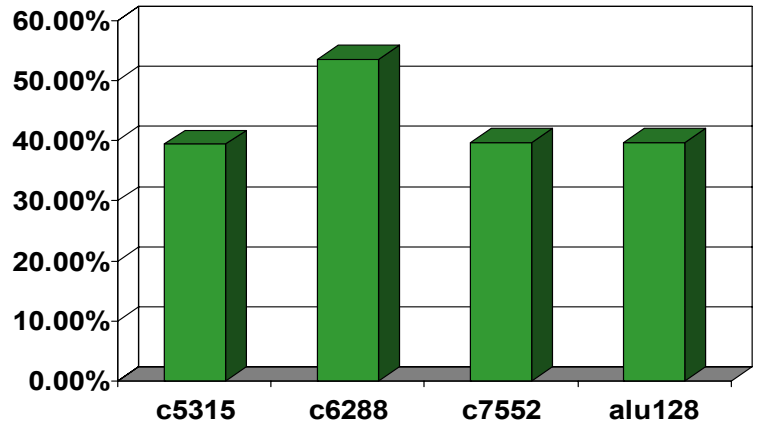
Results: Leakage Variability



Leakage variability estimated with 10,000 Monte-Carlo simulations on **alu128**

$\sigma_{WID} = \sigma_{DTD} = 3.3nm$

(Variations in gate-length assumed to be Gaussian w/ zero correlation)



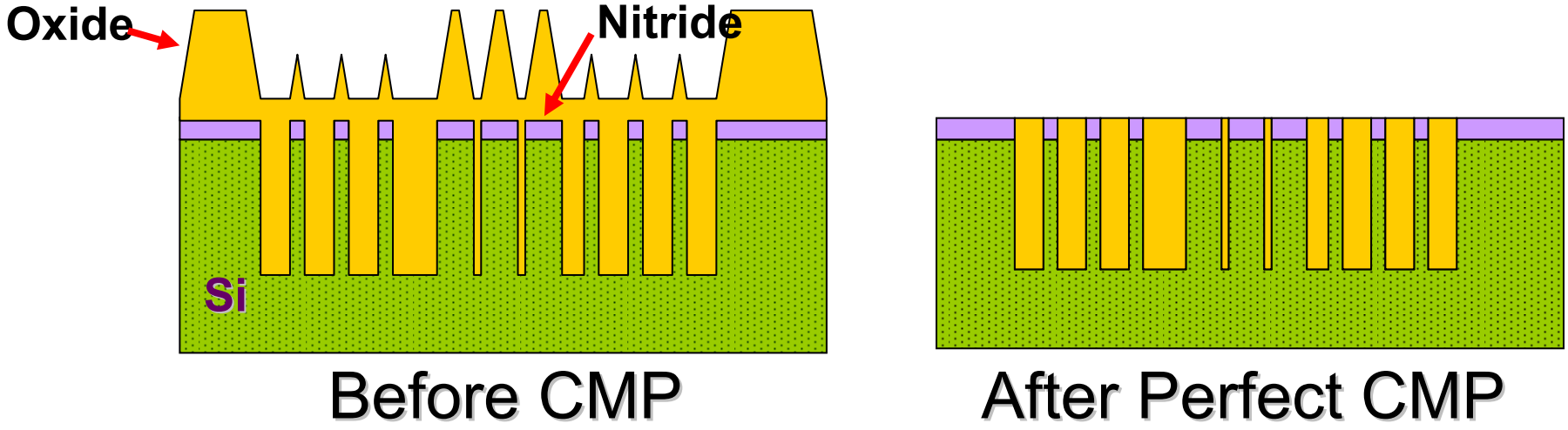
Percentage Reduction in Leakage Spread

Backup

- STI Fill for CMP

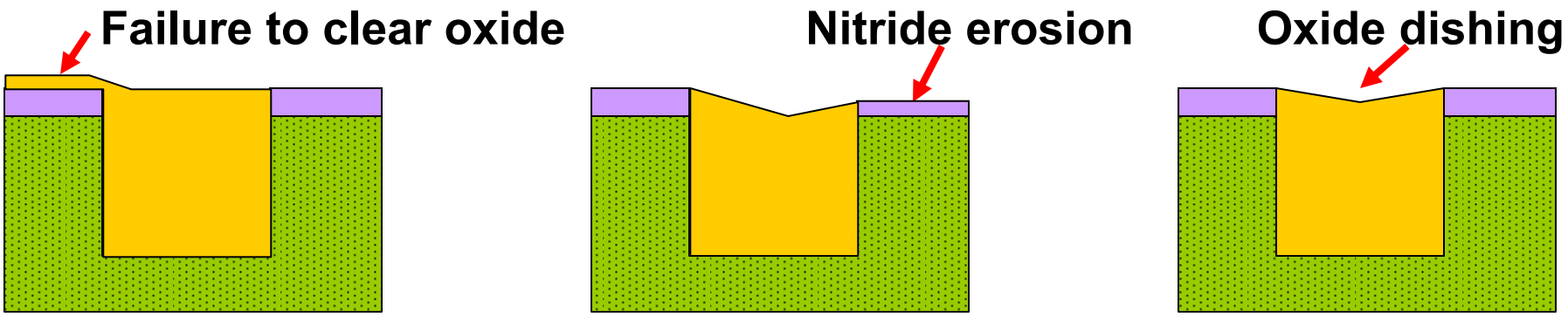
CMP for STI

- STI mainstream CMOS isolation technology
- In STI, substrate trenches filled with oxide surround devices or group of devices that need to be isolated
- Relevant process steps:
  - Diffusion (OD) regions covered with nitride
  - Trenches created where nitride absent and filled with oxide
  - Chemical Mechanical Polishing (CMP) to planarize and remove excess oxide over nitride *(overburden oxide)*



- CMP goal: Perfectly planar nitride and trench oxide surface

CMP is Not Perfect



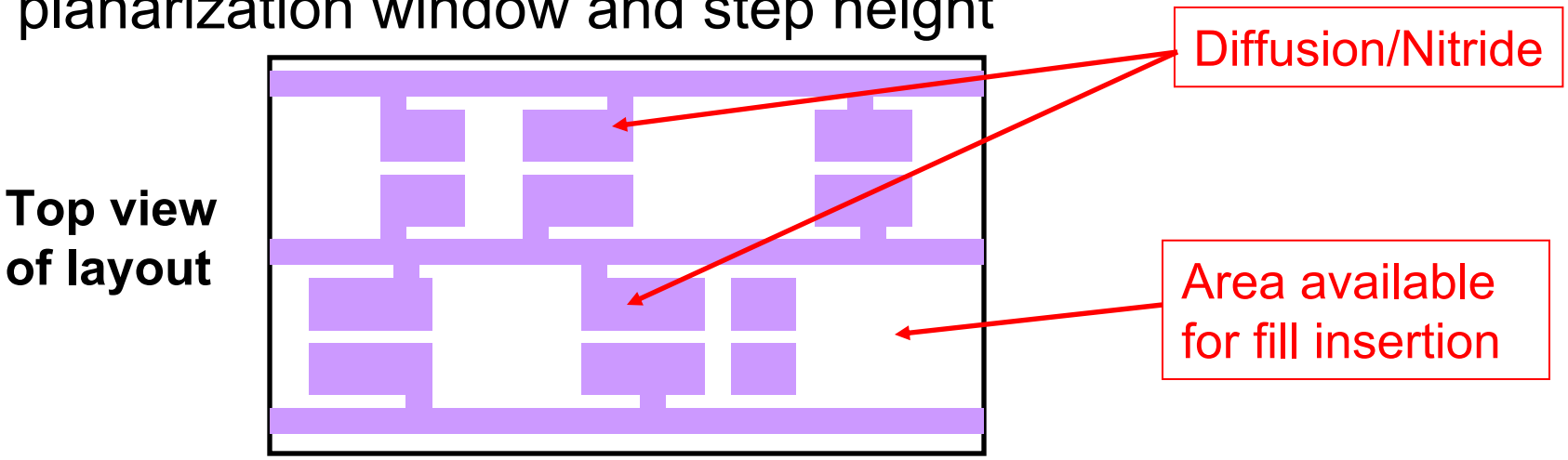
Key Failures Caused by Imperfect CMP

- *Planarization window:* Time window to stop CMP
  - Stopping sooner leaves oxide over nitride
  - Stopping later polishes silicon under nitride
  - Larger planarization window desirable
- *Step height:* Oxide thickness variation after CMP
  - Quantifies oxide dishing
  - Smaller step height desirable
- CMP quality depends on nitride and oxide density

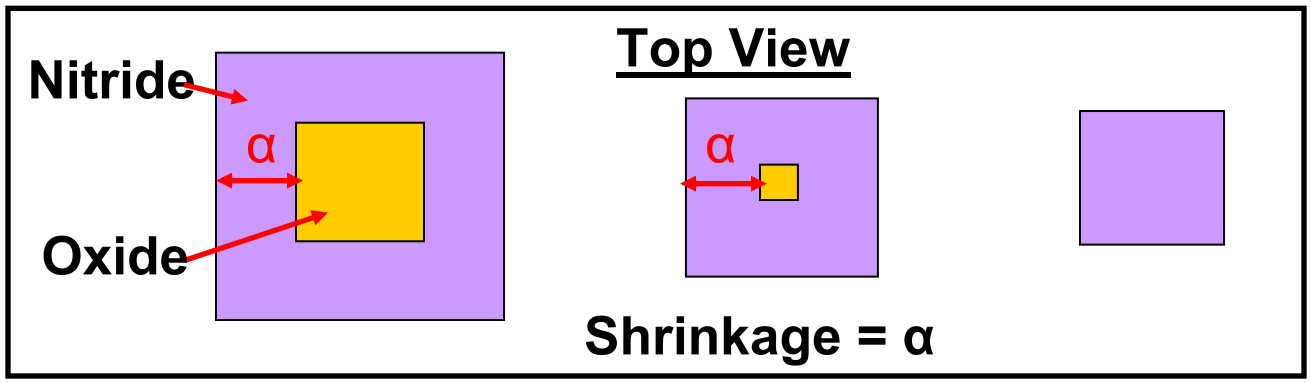
→ Control nitride and oxide density to enlarge planarization window and to decrease step height

Fill Insertion

- CMP is pattern dependent → Fill insertion improves planarization window and step height



- *Deposition bias:* Oxide over nitride deposited with slanted profile → Oxide features are "shrunk" nitride features



- *Size* and *shape* fill to control nitride and oxide density

Objectives for Fill Insertion

- Primary goals:
  - Enlarge planarization window
  - Minimize step height i.e., post-CMP oxide height variation
- Minimize oxide density variation

  → Oxide uniformly removed from all regions

  → Enlarges planarization window as oxide clears simultaneously
- Maximize nitride density

  → Enlarges planarization window as nitride polishes slowly

Objective 1: Minimize oxide density variation
Objective 2: Maximize nitride density

Dual-Objective Problem Formulation

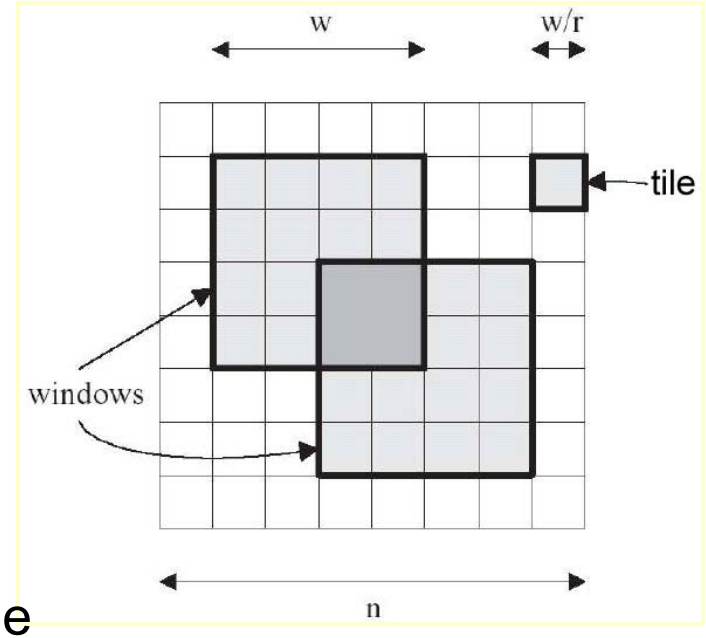
- Dummy fill formulation
  - Given:
    - STI regions where fill can be inserted
    - Shrinkage α
  - Constraint:
    - No DRC violations (such as min. spacing, min .width, min. area, etc.)
  - Objectives:
    1. minimize oxide density variation
    2. maximize nitride density

Density Variation Minimization with LP

- Minimize oxide density variation
  - Use previously proposed LP-based solution
  - Layout area divided into n x n tiles
  - Density computed over sliding windows (= w x w tiles)
  - Inputs:
    - *min.* oxide density ($|Oxide_{Min}|$) per tile

      → To compute: shrink design's nitride features by α
    - *max.* oxide density ($|Oxide_{Max}|$) per tile

      → To compute: insert max. fill, shrink nitride features by α
  - Output: *target* oxide density *(*$|Oxide_{Target}|$*)* per tile
  - Dual-objective → single-objective (nitride density) problem with oxide density constrained to $|Oxide_{Target}|$



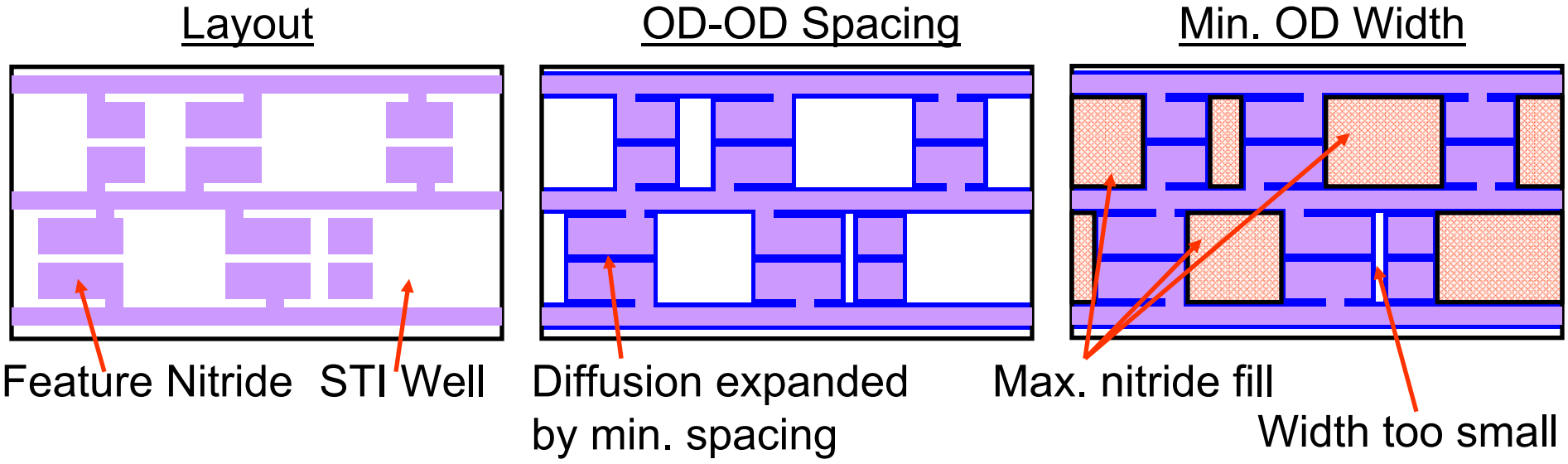
Nitride Maximization Problem Formulation

- Dummy fill formulation
  - Given:
    - STI regions where fill can be inserted
    - Shrinkage α
  - Constraint:
    - No DRC violations (such as min. spacing, min .width, min. area, etc.)
    - *Target oxide density ($|Oxide_{Target}|$)*
  - Objectives:
    - maximize nitride density

Case Analysis Based Solution

- Given $|Oxide_{Target}|$, insert fill for max. nitride density
- Solution (for each tile) based on case analysis
  - Case 1: $|Oxide_{Target}| = |Oxide_{Max}|$
  - Case 2: $|Oxide_{Target}| = |Oxide_{Min}|$

  } More common due to nature of LP
  - Case 3: $|Oxide_{Min}| < |Oxide_{Target}| < |Oxide_{Max}|$
- Case 1 → Insert max. nitride fill
  - Fill nitride everywhere where it can be added
  - Min. OD-OD (diffusion-diffusion) spacing ≈ 0.15μ
  - Min. OD width ≈ 0.15μ
  - Other OD DRCs: min. area, max. width, max. area

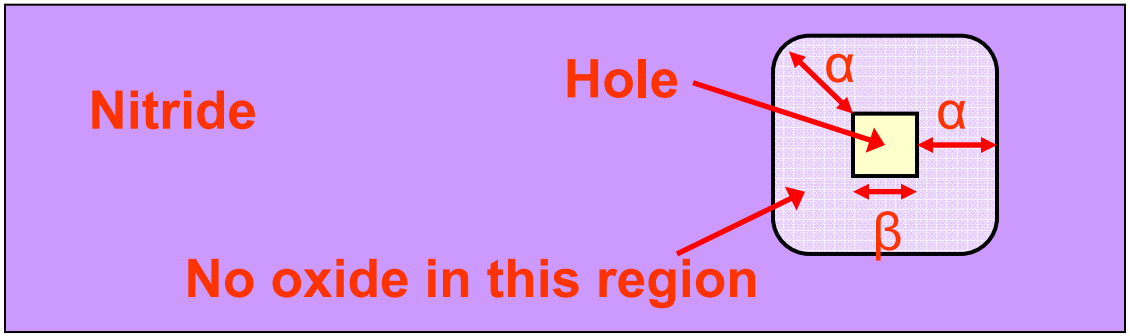


Case 2: $|Oxide_{Target}| = |Oxide_{Min}|$

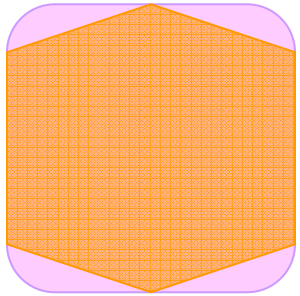
- Need to insert fill that does not increase oxide density
- Naïve approach: insert fill rectangles of shorter side < α
- Better approach: perform max. nitride fill then dig square holes of min. allowable side β

  → Gives higher nitride:oxide density ratio

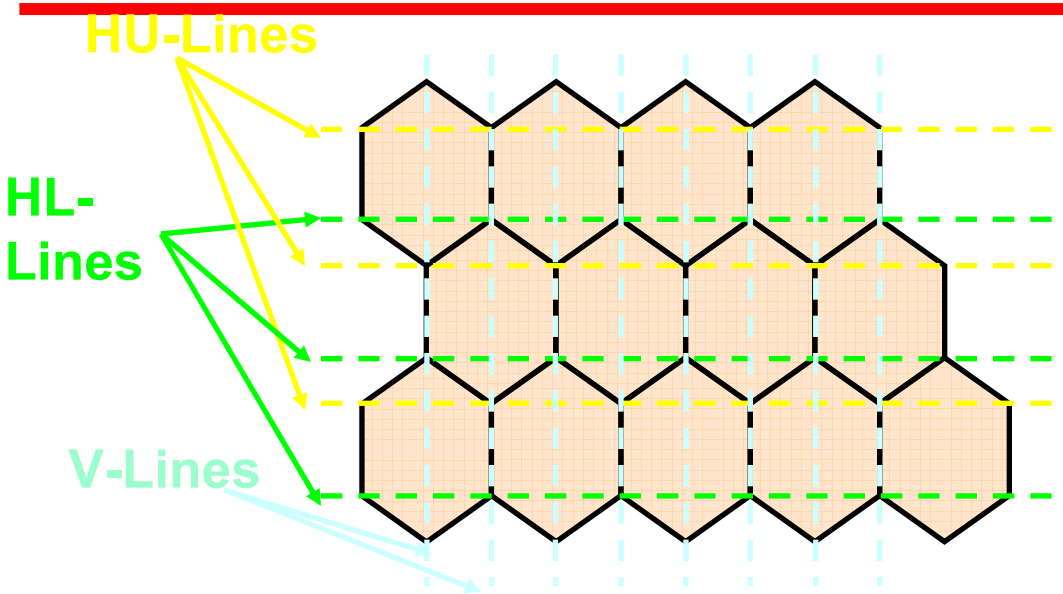
Top View

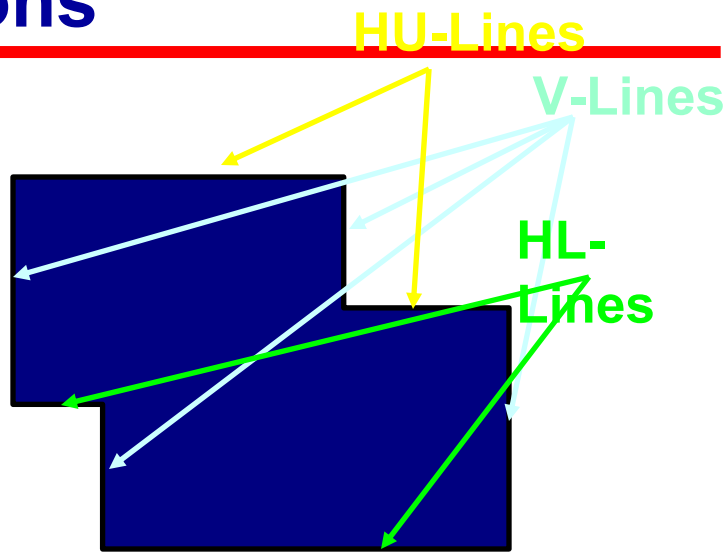


- No oxide density in rounded square around a hole
- *Cover* nitride with rounded squares → no oxide density
- Covering with rounded squares difficult → approximate rounded squares with inscribed hexagons
- Cover rectilinear max. nitride with min. number of hexagons → proposed a new algorithm



Covering Bulk Fill with Hexagons





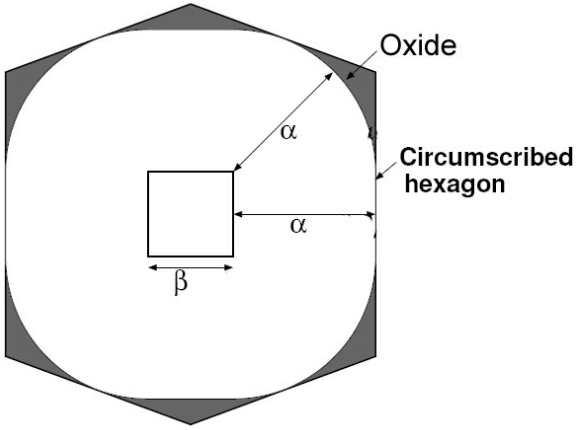
Key observation: At least one V-Line and one of HU- or HL- Lines of the honeycomb must overlap with corresponding from polygon
Proof: In paper. (Can displace honeycomb to align one V-Line and one of HU- or HL-Line without needing additional hexagons.)

Approach: Select combinations of V- and HL- or HU- Lines from polygon, overlap with honeycomb and count hexagons. Select combination with min. hexagons. Also flip polygon by 90º and repeat.
Complexity: |Polygon V-Lines| x (|Polygon HL-Lines| + |Polygon HU-Lines|) x |Polygon area|

→ Cover max. nitride fill with hexagons, create holes in hexagon centers

Case 3: $|Oxide_{Min}| < |Oxide_{Target}| < |Oxide_{Max}|$

- Holes give high nitride:oxide density
  → insert max. nitride fill and create holes to reduce oxide density
- OK for nitride fill to contribute to oxide density
  → approximate rounded squares by *circumscribed* hexagons



$$Outloss = \frac{\text{Oxide Area}}{\text{Nitride Area}}$$

- When max. nitride is covered with circumscribed hexagons, oxide density increases
  - If oxide density (=outloss x max. nitride area) $< |Oxide_{Target}|$ → increase oxide density by filling some holes
  - If oxide density $> |Oxide_{Target}|$ → decrease oxide density by partially using Case 2 solution

Solution Summary

- Divide layout into tiles
- Calculate $|Oxide_{Min}|$ and $|Oxide_{Max}|$
- Run LP-based fill synthesis for oxide variation minimization → Get $|Oxide_{Target}|$
- If $|Oxide_{Target}| = |Oxide_{Max}|$ (i.e., max. oxide needed)
  → Add max. nitride fill
- If $|Oxide_{Target}| = |Oxide_{Min}|$ (i.e., add no more oxide)
  → Add max. nitride fill
  → Calculate *inscribed* hexagon size based on α and β
  → Cover max. nitride fill with hexagons
  → Create square holes in the center of hexagons
- If $|Oxide_{Min}| < |Oxide_{Target}| < |Oxide_{Max}|$ (i.e., general case)
  → Add max. nitride fill
  → Calculate *circumscribed* hexagon size based on α and β
  → Cover max. nitride fill with hexagons
  → Create square holes in the centers of hexagons
  → If oxide density *lower* than needed → fill some holes
  → If oxide density *higher* than needed → Use *inscribed* hexagons in some region

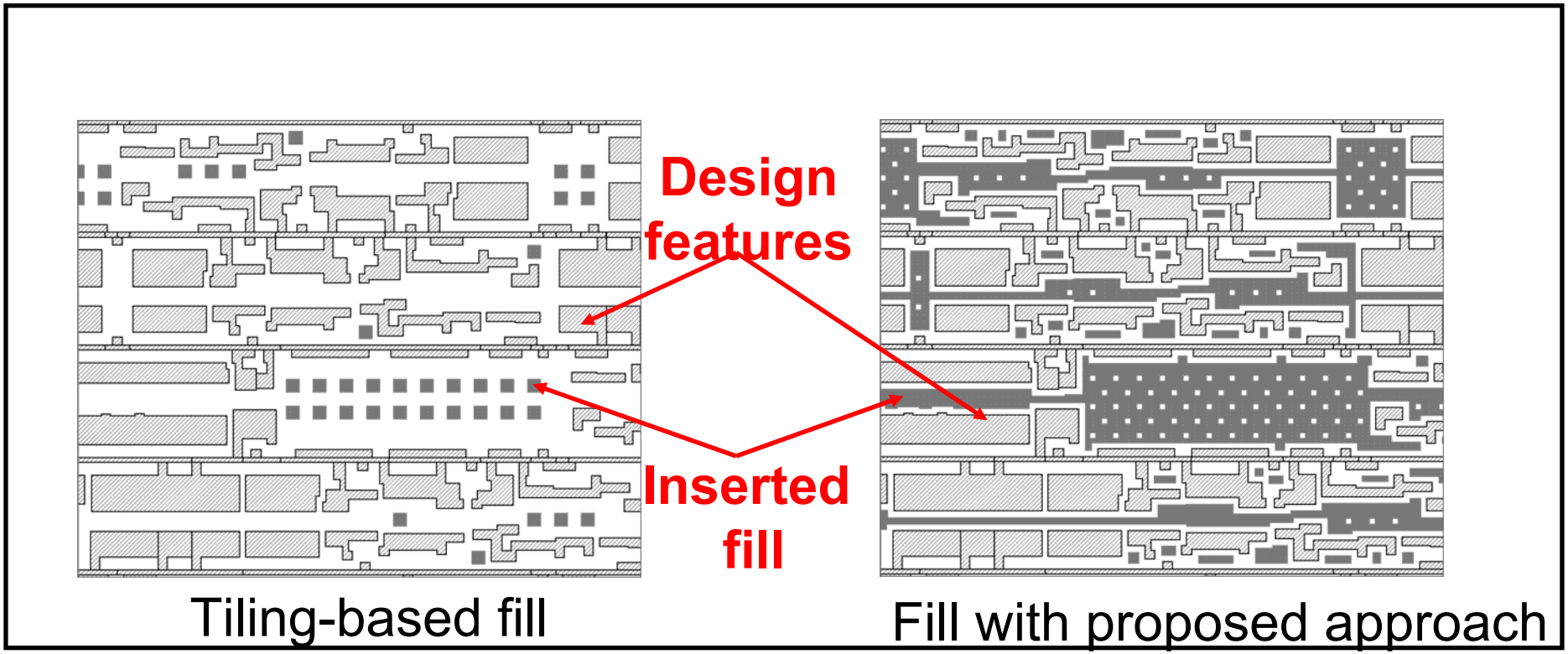
Experimental Setup

- Two types of studies
  - Density analysis
  - Post-CMP topography assessment using CMP simulator
- Comparisons between:
  - Unfilled
  - Tile-based fill (DRC-correct fill squares inserted)
  - Proposed fill
- Our testcases: 2 large designs created by assembling smaller ones
  - “Mixed”: RISC + JPEG + AES + DES

    2mm x 2mm, 756K cells
  - “OpenRisc8”: 8-core RISC + SRAM

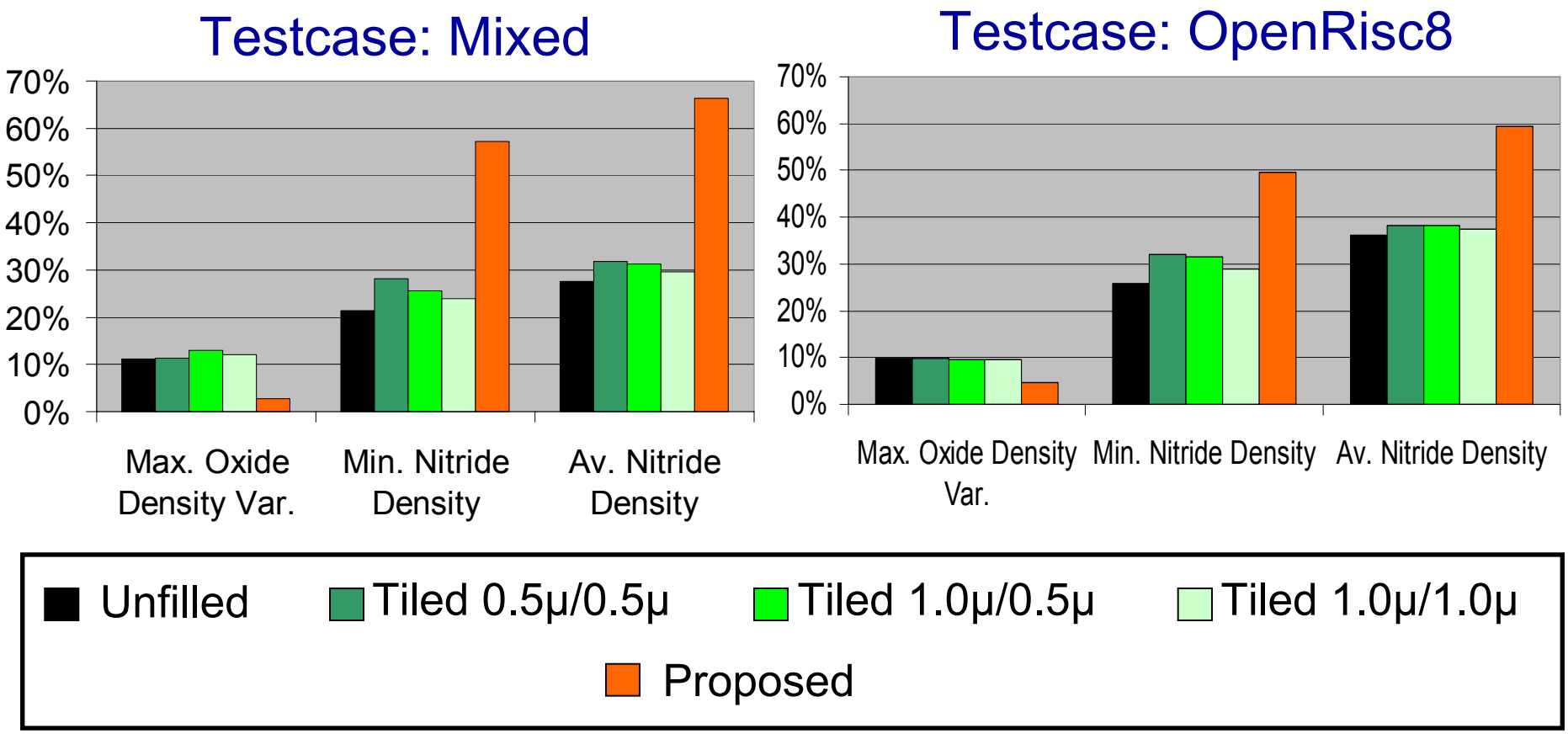
    2.8mm x 3mm, 423K cells + SRAM

Layout After Fill Insertion



+ Higher nitride density
+ Smaller variation in STI well size → less variation in STI stress

Density Enhancement Results



+ Significantly higher nitride density
+ Lower oxide density variation

Post-CMP Topography Assessment

Testcase	Fill Approach	Final Max. Step Height (nm)	Planarization Window (s)
Mixed	Unfilled	142	45.3
	Tiled 0.5µ/0.5µ	143	46.5
	Proposed	**129**	**53.6**
OpenRisc8	Unfilled	146	42.7
	Tiled 0.5µ/0.5µ	144	44.7
	Proposed	**133**	**50.4**

+ Smaller step height → less oxide height variation
+ Larger planarization window