Review of Register-transfer Level Design Flow and a Look at Industrial Practice

Prof. Kurt Keutzer
EECS
keutzer@eecs.berkeley.edu

Feedback

- Good
  - Entire flow of CAD
  - Real world perspective
  - Examples of algorithms
  - Class structure → problem → formulation → algorithm
  - No homework
  - Dialog with class/ "Socratic approach"
- Bad
  - No homework
  - Questions to class sometimes unclear
  - Too fast, need more details
  - Energy of lecture up and down some days

Actually, this was last year's feedback!!!

Feedback – this year

- Good
  - Entire flow of CAD
  - Real world perspective
  - Examples of algorithms
  - Good slides
  - No homework
  - Questions invited in class
- Bad
  - No homework
  - Too fast, need more details – more examples of algorithms
  - Slides not updated on webpage before class – slides don't match the lecture
  - Can't see fonts on slides
  - Don't know what to read before class

Responding to feedback

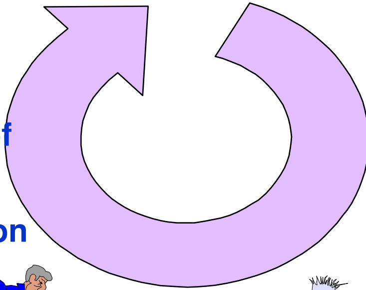
- Graduate course without an undergraduate "Intro to CAD" course – it's a challenge for all of us
  - No homework -
    - we voted – it was your call
  - Too fast, need more details –
    - A lot of material but a modest amount of out of class work
    - more examples of algorithms? Welcome to graduate school!
  - Slides not updated on webpage before class – slides don't match the lecture
    - I add new questions every year – don't want to update before class
    - Should be updated immediately after class – my bad
  - Can't see fonts on slides – get glasses, really!
  - Don't know what to read before class – check the web page – reading assignments have been there
  - Speak up!! Raise your questions/concerns *during* class

Design Process

- **Design** : specify and enter the design intent

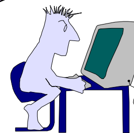
Verify:

verify the correctness of design and implementation



Implement:

refine the design through all phases



How are designers describing ICs?

Current Practice: HDL at RTL Level

```
module foobar (q,clk,s,a,b);
  input clk, s, a, b;
  output q; reg q; reg d;
always @(a or b or s) // mux
    begin
      if( !s )
        d = a;
      else if( s )
        d = b;
      else
        d = 'bx;
    end // always @ (a or b or s)

always @(clk) // latch
  begin
    if( clk == 1 )
      q = d;
    else if( clk !== 0 )
      q = 'bx;
  end // always @ (clk)

endmodule
```


Verification

- **Design** : specify and enter the design intent

Verify:

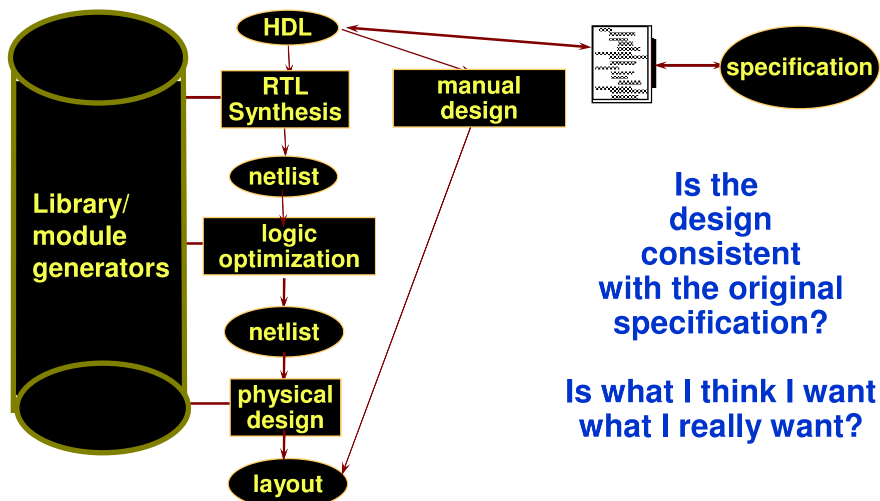
verify the correctness of design and implementation

Implement:

refine the design through all phases

What are the three phases of verification?

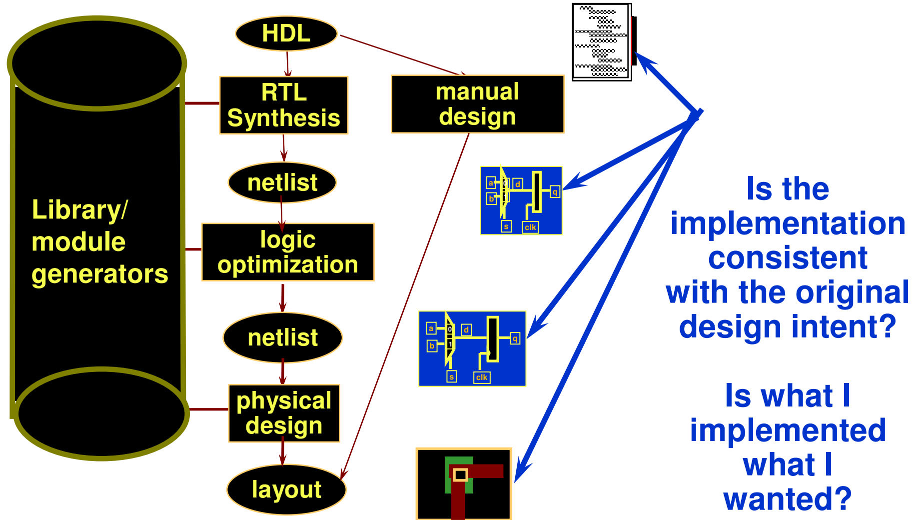
Design Verification



Is the design consistent with the original specification?

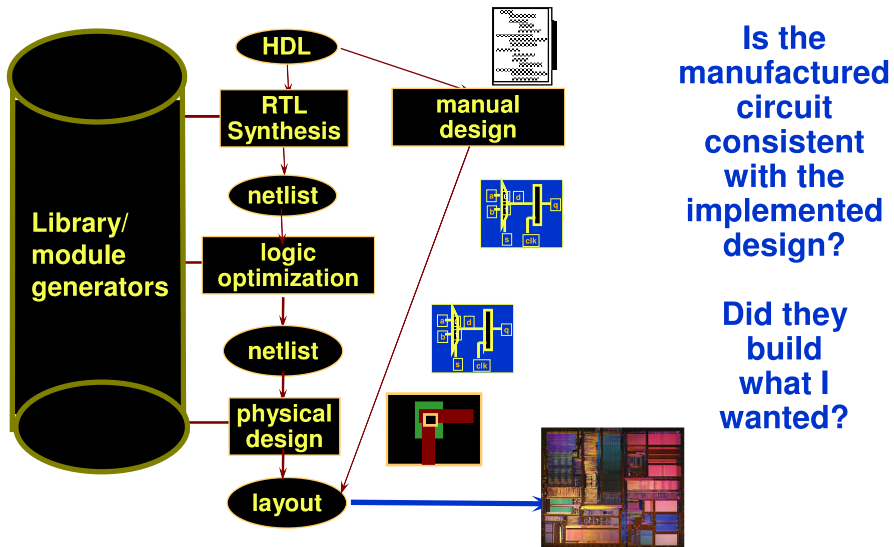
Is what I think I want what I really want?

Implementation Verification



11

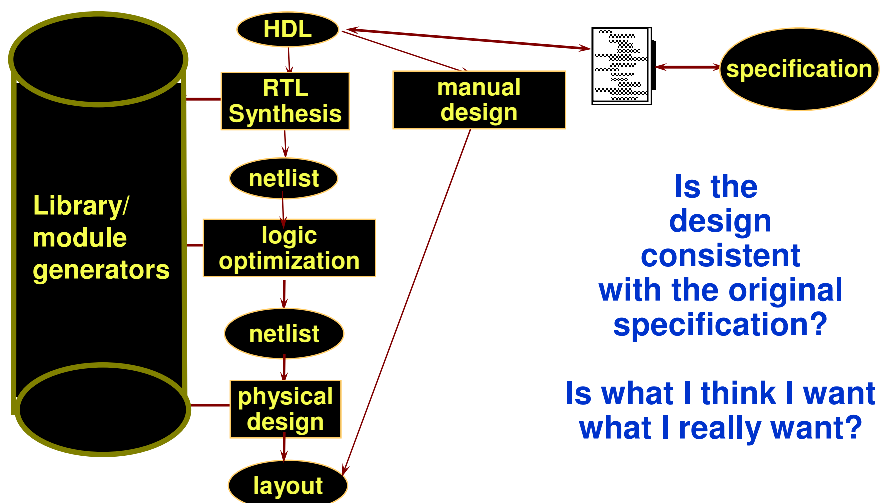
Manufacture Verification (Test)



12

What is the work horse of design verification?

Design Verification



Is the design consistent with the original specification?

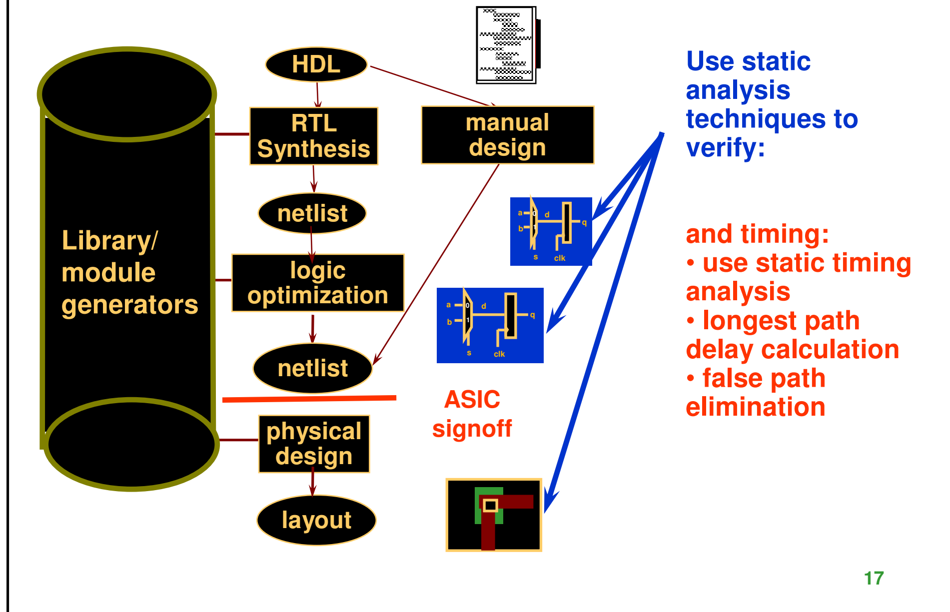
Is what I think I want what I really want?

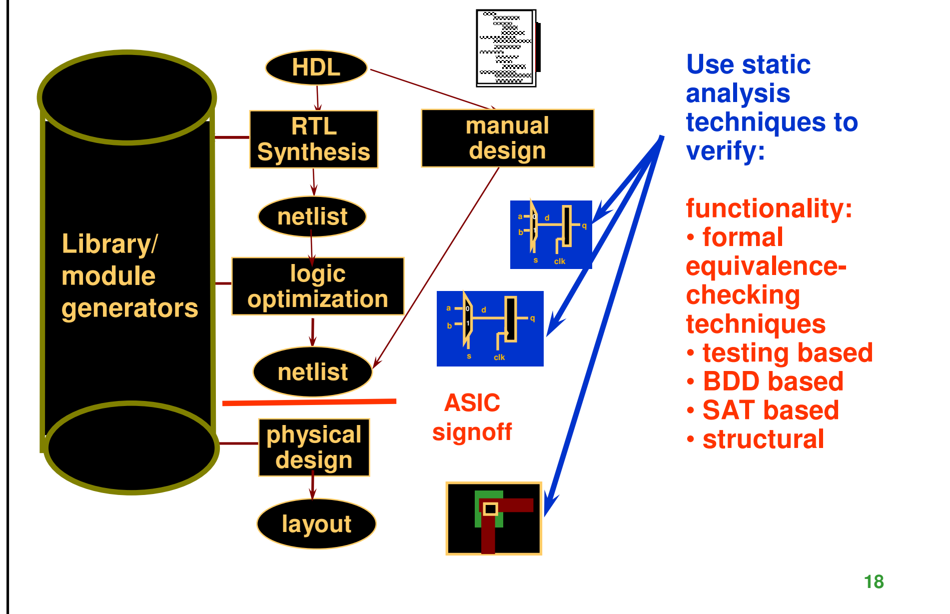
Types of software simulators

- Circuit simulation
  - Spice, Advice, Hspice
  - Timemill + Ace, ADM
- Event-driven gate/RTL/Behavioral simulation
  - Verilog - VCS, NC-Verilog, Turbo-Verilog, Verilog-XL
  - VHDL - VSS, MTI, Leapfrog
- Cycle-based gate/RTL/Behavioral simulation
  - Verilog - Frontline, Speedsim
  - VHDL - Cyclone
- Domain-specific simulation
  - SPW, COSSAP
- Architecture-specific simulation

What are the work horses of implementation verification?

- What are the key techniques employed?

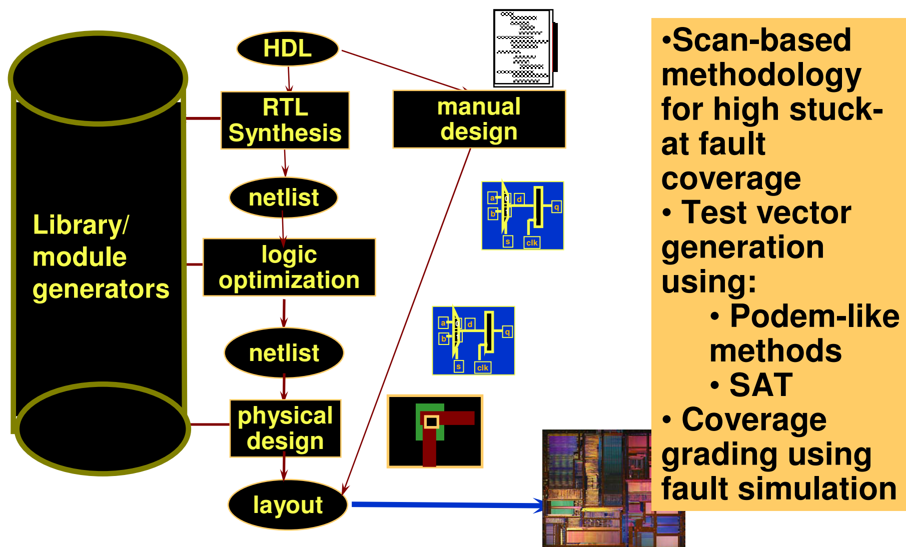

Static Sign-off - 1
HDL
RTL Synthesis
manual design
netlist
logic optimization
netlist
Library/ module generators
physical design
layout
ASIC signoff
Use static analysis techniques to verify:
and timing:
• use static timing analysis
• longest path delay calculation
• false path elimination
17


Static Sign-off - 2
HDL
RTL Synthesis
manual design
netlist
logic optimization
netlist
Library/ module generators
physical design
layout
ASIC signoff
Use static analysis techniques to verify:
functionality:
• formal equivalence-checking techniques
• testing based
• BDD based
• SAT based
• structural
18

What are the work horses of mfgr test?

- What are the key techniques employed?

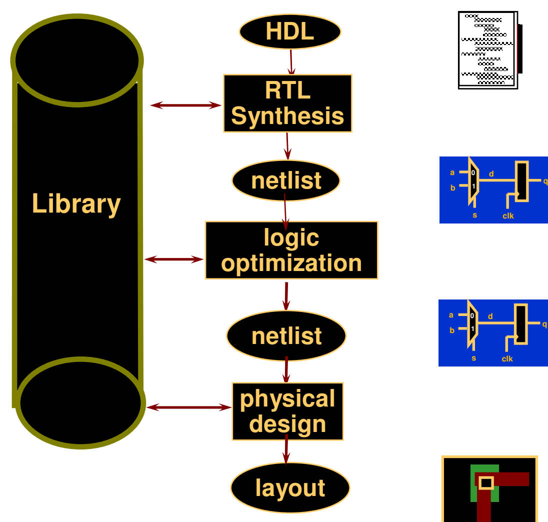
Manufacture Verification (Test)



- Scan-based methodology for high stuck-at fault coverage
- Test vector generation using:
  - Podem-like methods
  - SAT
- Coverage grading using fault simulation

What kind of flow is used to design most ASICs?

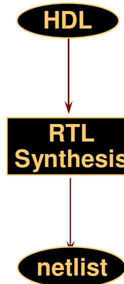
RTL Synthesis Flow

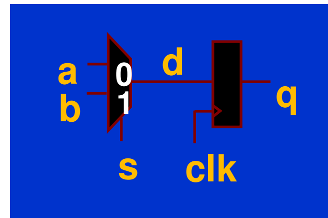


RTL Synthesis

```
module foobar (q,clk,s,a,b);
  input clk, s, a, b;
  output q; reg q; reg d;

  always @(a or b or s) // mux
    begin
      if(~s )
        d = a;
      else if( s )
        d = b;
      else
        d = 'bx;
    end // always @ (a or b or s)
```

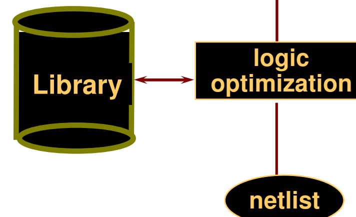


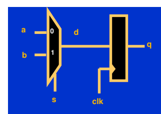


translate HDL source into netlist

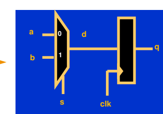
Logic Optimization

- Perform a variety of transformations and optimizations
  - Structural graph transformations
  - Boolean transformations
  - Mapping into a physical library
- What are the key algorithms?





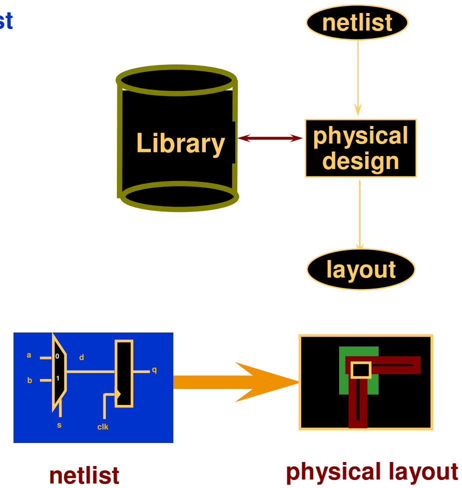
pre-optimized



smaller, faster
less power

Physical Design

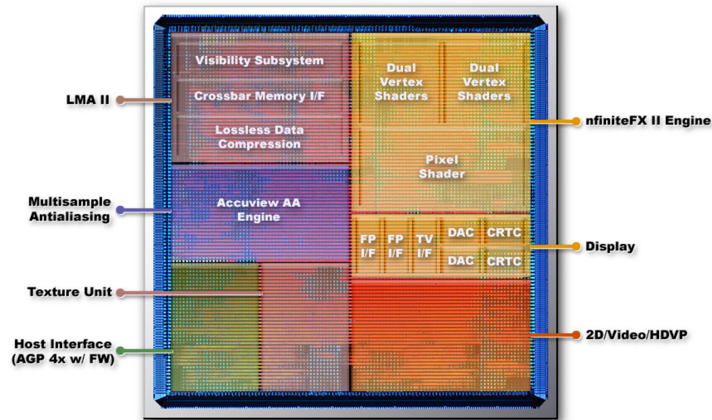
- Transform sequential circuit netlist into a physical circuit
  - *place* circuit components
  - *route* wires
  - transform into a mask
- Or for FPGA’s
  - *place* look-up tables
  - *route* wires
- What are the key algorithms?



Ok – now what do companies really do?

- What do they build?
- How do real designers put things together?
- Where does the time go?
- And what challenges do they face?

Nvidia GeForce4 Ti Architecture

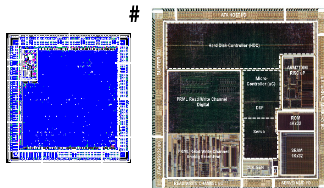


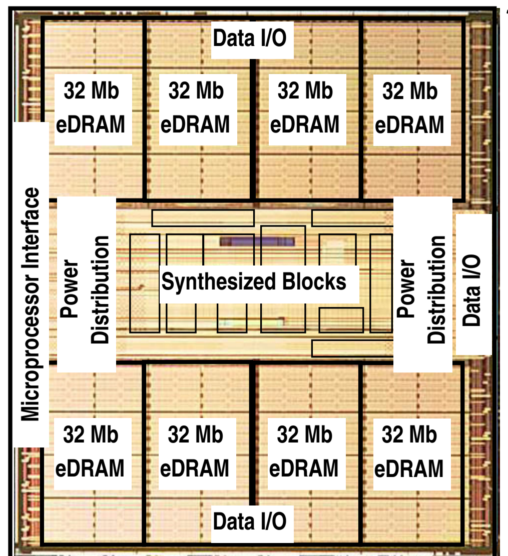
• 0.15u, 8LM, 63M txtr, 505 signals/801 balls, 18W

A. Khan – Cadence - 2002

- 0.5um → 0.18um
- 5x5 mm^2 → 21.7x21.3 mm^2
- ~0.8M → 287.5M transistors
- 3LM → 6LM (8LM)
- ~100 → 150 MHz (622 MHz)

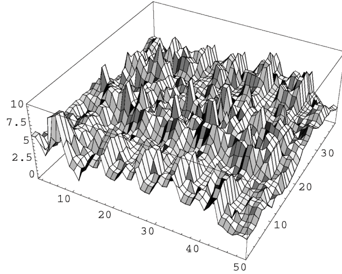
#



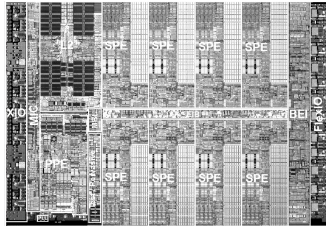


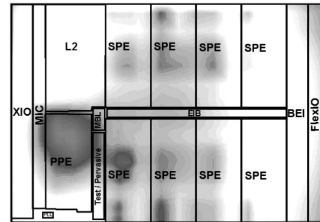
#: Cirrus Logic, Inc. IC, 3Ci™
*: Sony Computer Entertainment, Inc. & Sony Corporation
Graphics Synthesizer®I-32. Copyright 2000 Sony Computer Entertainment, Inc.

The Design and Implementation of a First-Generation CELL Processor (IBM, Sony, Toshiba)



- **1 64b PPE + 8 SPEs; typ. clock 4GHz+**
- **234M transistors**
- **3 clock networks**
- **12ps skew (top-2 metal grids)**
  - **850 individually-tuned elements**
- **Design verified for valid operation with thermal transients**

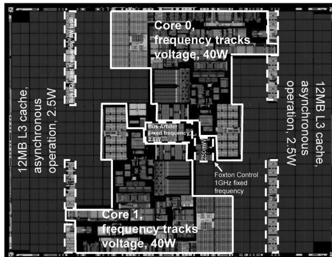




PPE; Power Processor Element; SPE: ssynergistic Processor Elements;

D. Pham, The Design and Implementation of a First-Generation CELL Processor," IEEE International Solid-State Circuits Conference, February 2005

The Implementation of a 2-core Multi-Threaded Itanium®-Family Processor (Intel)



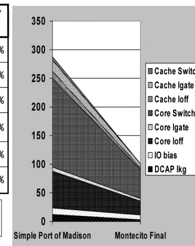
Domain	Voltage	Frequency	Power
Cores	.8 – 1.2V	Tracks voltage	80W total
Cache	.9-1V	Self-timed	5W total
IO term	1.2V	Bus	8W total
Fixed	1.15V	Fixed	6W total

	FET count	low Vt
Core logic	57M	1.7%
Core caches	106.5M	0
L3 Cache	1550M	0
Bus logic & IO	6.7M	0.3%
Total	1.72G	.06%

- **Power reduction primary design priority**
  - **130W single uP**
  - **2 uPs, 90nm, 2+GHz, 26.5MB cache (L1-L3); 1.72B transistors, 21.5x27.7 sq. mm**
    - **300W → 100W**
- **Manage power to a dynamically-adjustable limit (Max. performance/watt)**
  - **Multi-L, V_T; real-time clock, VDD control (ammeter, thermal monitors)**

Feature	Frequency Gain @ 100W
Manage to application power vs. max power	12%
Adapt Vcc to optimal value for each part	5%
Manage junction temperature to the minimum possible	3%
Adapt frequency to Vcc to operate at frequency(V_{avg}) vs. frequency(V_{min})	7%
Optimize circuits for low switching and low leakage	11%
Cache power optimization (low V, asynch. etc.)	5%

With $P \sim F^3$:
$P = P_{orig}/(1+.12+.05+.03+.07+.11+.05)^3 = P_{orig}/2.92$

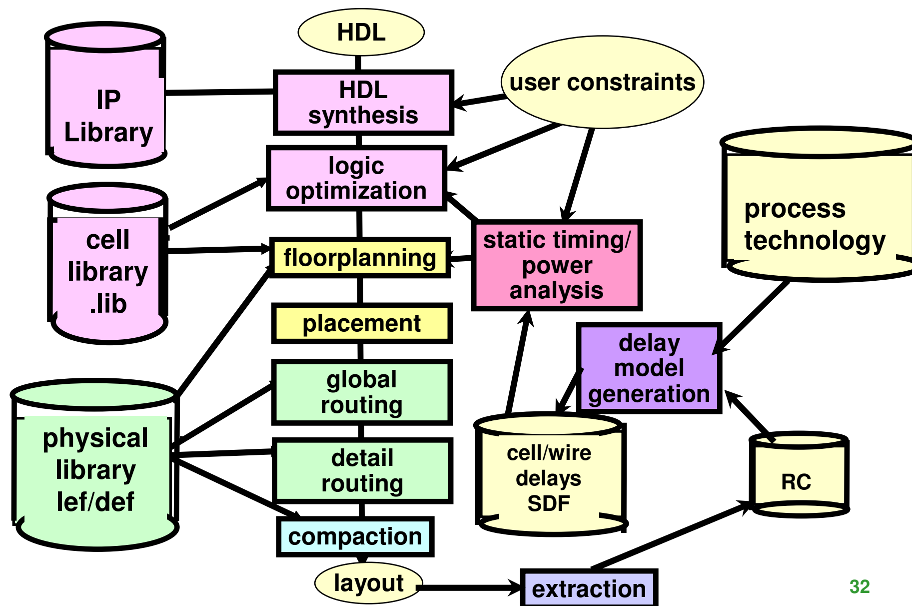


S. Naffziger, "The Implementation of a 2-core Multi-Threaded Itanium®-Family Processor," IEEE International Solid-State Circuits Conference, February 2005

Ok – now what do people really do?

- What do they build?
- How do real designers put things together?
- Where does the time go?
- And what challenges do they face?

EDA Design Flow for Implementation



NVIDIA Design Flow

External Reqmnts
Architecture & Algorithms
Class Definitions
Micro-architecture
C-models
RTL
Drivers
Logic Verification
Validation
Test Reqmnts
Floor Plan
Emulation
Test Insertion
TapeOut
Place and Route
Phys Opt
Foundry Reqmnts
DRC/LVS

Architecture
ASIC
Physical Design
Software
Marketing
Operations
Sales

Stargen Tool Flow

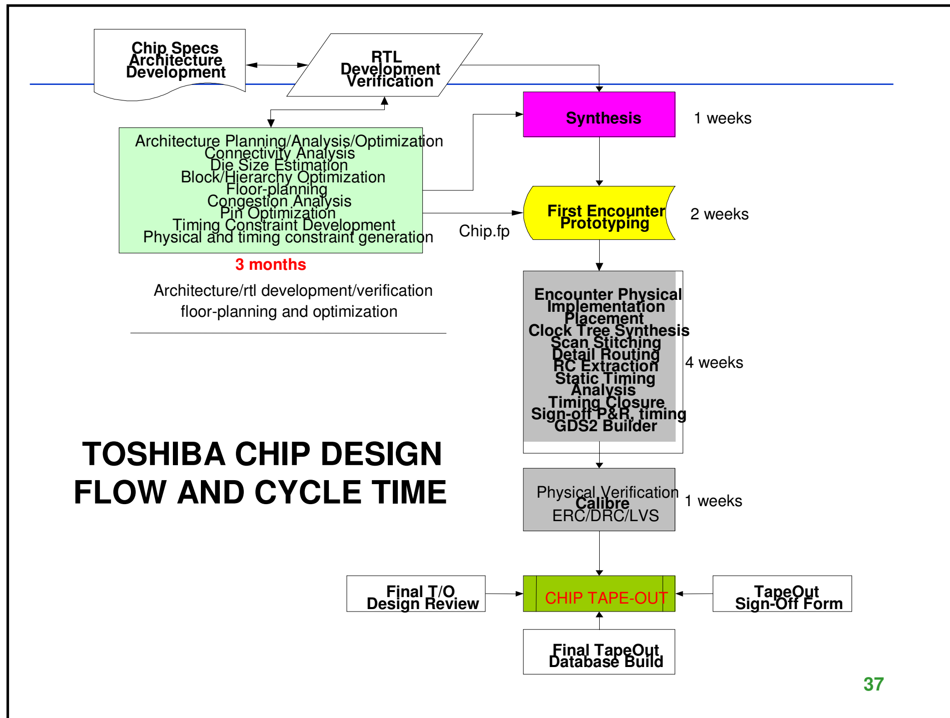
Verilog RTL Coding Timing fixes
Synthesis, Scan (DC, DFT Compiler)
RTL DRC (hdl-lint)
RTL/Gate Simulation (VCS)
Static Timing (Primetime)
DFT audits
RTL Floorplanning (SOCPrototype)
lefs
black boxes
pad order
early power planning
Jupiter def
.pdef
Verilog gates/RTL
Physical Synthesis (Physical Compiler)
Stargen
Timing violations
.pdef
.pdef
Netlist Floorplanning (Jupiter)
Detailed power planning
Pad ring placement
Ram placement
Equivalency (Formality)
Physical Synthesis (Physical Compiler)
Scan chain specification
CTS, Incr Placement, Route (Astro)
.sdf, Verilog
Backend Vendor
ATPG/Test Vectors (FastScan or Tetramax)
Extraction (StarRC-XT)
.spef, Verilog
.GDSII
Test Program (Fluence)
LVS/DRC (Mentor Calibre)
Signal Integrity Analysis (Astro/PrimetimeSI)
Tester (Agilent 93K)

Tools in Stargen Flow

RTL creation, checking, debugging	Emacs, vi, Everest HDL-Lint, Novas Debussy
Verilog simulation/testbench	Synopsys VCS, VERA
RTL floorplanning	Icinergy SOCPrototype
Logic synthesis Physical synthesis	Synopsys Design Compiler Synopsys Physical Compiler
Static timing analysis	Primetime
Design for test analysis and scan chain insertion	Synopsys DFT Compiler or Mentor DFT Advisor
Gate netlist floorplanning	Synopsys (Avanti) Jupiter
Clock tree synthesis, routing	Synopsys (Avanti) Astro
Extraction	Synopsys (Avanti) Star-RCXT
Signal integrity	PrimetimeSI, AstroXTalk, AstroRail
DRC/LVS	Mentor Calibre
Equivalency checking	Synopsys Formality
Memory BIST	TBD Tools
ATPG, IEEE1149	TBD Tools

Ok – now what do people really do?

- What do they build?
- How do real designers put things together?
- Where does the time go?
- And what challenges do they face?



Current Status of RTL Design Flow

- Current RTL design flow is able to produce
  - 10 - 100M logical gate ASIC platforms:
  - Significant portions of high speed microprocessors - e.g. Alpha, Pentium Pro > 1M gate-equivalents > 1GHz.
  - Rapid turnaround ASIC → structured ASIC
  - 6000 (down from 8500) IC designs/year go through this flow
  - Not providing productivity improvement needed to keep up with Moore's Law
- Successful flow, but stalled out – why?

Ok – now what do people really do?

- What do they build?
- How do real designers put things together?
- Where does the time go?
- And what challenges do they face?

What happens in a process generation?

- In the transition from one process generation (e.g. .18u) to another (e.g. .13u)
- Critical dimensions shrink by a scaling factor *S,* typically $S = 2^{1/2}$
  - Has a "squaring" effect on density $S^2 = 2$ – I.e. same number of transistors in half the area
  - Hence Moore's Law – 2X density increase every 18 months
  - Ideally has a similar effect on performance 1.3 * 1.3 = 1.7
- Vdd reduces by a scaling factor *U* – more on this later
- As Vdd reduces Vth reduces in order to get performance gains

- And what's the point of all this?

Defining a technology node – "half pitch"

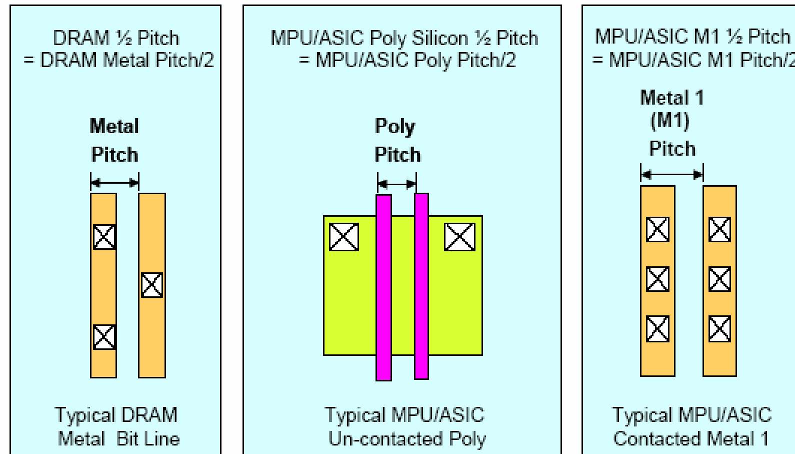
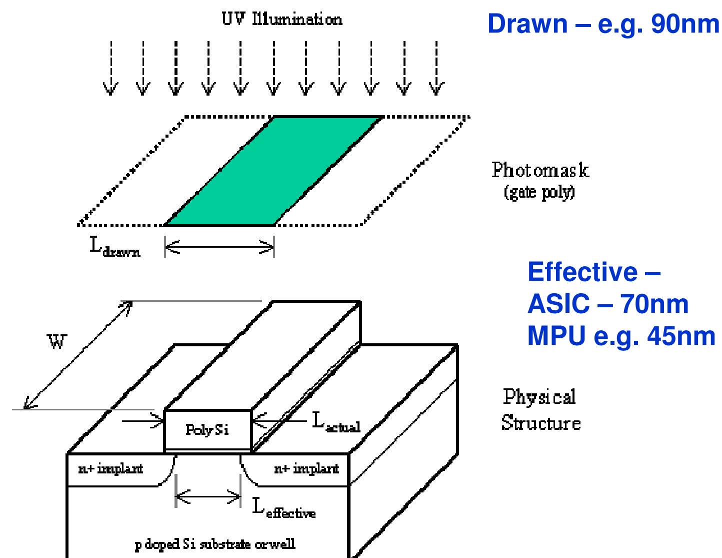


Figure 4 Definition of Metal Half Pitch

Definition of "half-pitch" – ITRS Roadmap

Drawn and Effective



Drawn – e.g. 90nm

Effective –
ASIC – 70nm
MPU e.g. 45nm

Technology Scaling Models

- **Full Scaling (Constant Electrical Field)**

  ideal model — dimensions and voltage scale together by the same factor *S*

- **Fixed Voltage Scaling**

  most common model until 1990's
  only dimensions scale, voltages remain constant

- **General Scaling**

  most realistic for today's situation —
  Voltages and dimensions scale with different factors – *U* and *S* respectively

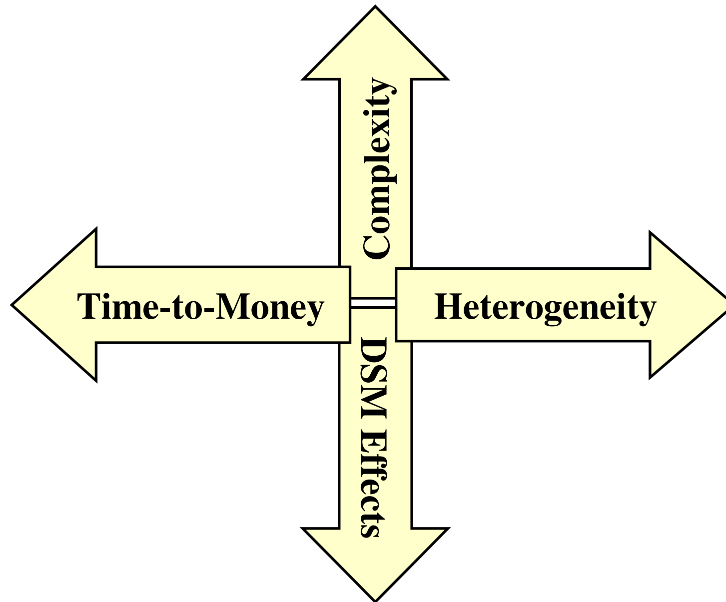
Scaling Relationships for Long Channel Devices

Parameter	Relation	Full Scaling	General Scaling	Fixed Voltage Scaling
W, L, t_{ox}		1/S	1/S	1/S
V_{DD}, V_T		1/S	1/U	1
N_{SUB}	V/W_{depl}^2	S	S^2/U	S^2
Area/Device	WL	$1/S^2$	$1/S^2$	$1/S^2$
C_{ox}	$1/t_{ox}$	S	S	S
C_L	$C_{ox}WL$	1/S	1/S	1/S
k_n, k_p	$C_{ox}W/L$	S	S	S
I_{av}	$k_{n,p} V^2$	1/S	S/U^2	S
t_p (intrinsic)	$C_L V / I_{av}$	1/S	U/S^2	$1/S^2$
P_{av}	$C_L V^2 / t_p$	$1/S^2$	S/U^3	S
PDP	$C_L V^2$	$1/S^3$	$1/SU^2$	1/S

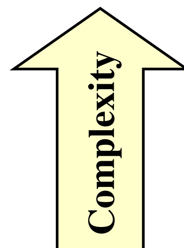
Table 3.1: Scaling Relationships for Long Channel Devices

Weste & Harris – CMOS VLSI Design

Result: A Quadruple-Whammy



Increasing Device and Context Complexity



- Exponential increase in device complexity—increasing with Moore's law (or faster)!
- System context in which devices are deployed (e.g. cellular radio) are increasing in complexity as well exponential increases in design productivity

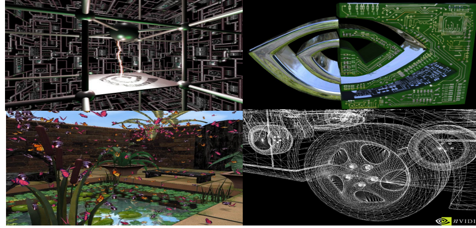
We have exponentially more transistors!

Scope of design growing exponentially

Lines of Verilog grows with Moore's Law

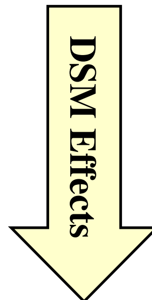
GPU	Technology	Transistors	Frequency	Placeable Instances	Flops	Models C vs V	Directed Arch Tests	GDS2 File size	Power
Gen1	0.25u	9M	125MHz	1M	~60K	90K/300K	300	800KB	6 W
Gen2	0.18u	25M	250MHz[1]	1.5M	~200K	100K/300K	6000	2GB	10 W
Gen3	0.15u	57M	350MHz[1]	3M	~500K	400K/800K	25000	4.5GB	12 W
Gen4	0.13u	~120M	450MHz[1]	5.5M	~750K	700K/1.3M	50000	8GB	15W

1 - dual edge clocking



Chris Malachowsky - NVidia

Deep Submicron Effects



The design of each transistor is getting more difficult due to physical effects associated with very small (deep submicron) geometries.

Important Deep Submicron Effects

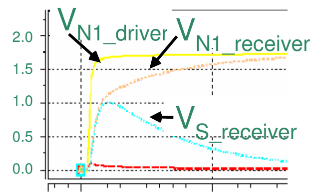
1. Rising relative delay of interconnect
2. Cross-coupled capacitance
3. IR drop
4. Power
   - 4a. Dynamic power consumption
   - 4b. Static power and leakage
5. Electromigration
6. Variability
7. Reliability

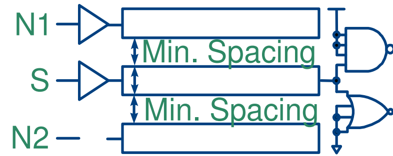
Important Deep Submicron Effects

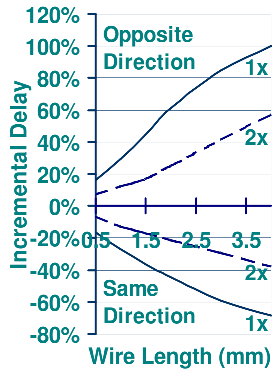
1. Rising relative delay of interconnect
2. Cross-coupled capacitance
3. IR drop
4. Power
   - 4a. Dynamic power consumption
   - 4b. Static power and leakage
5. Electromigration
6. Variability
7. Reliability

Signal integrity

A. Khan - Cadence







- Crosstalk/Crosscoupled capacitances can not only cause delay degradation in static cmos digital circuits – it can cause errors in:
  - Analog circuits
  - Dynamic CMOS circuits
- Solutions as before:
  - Shielding
  - Better analysis
  - Signal encoding

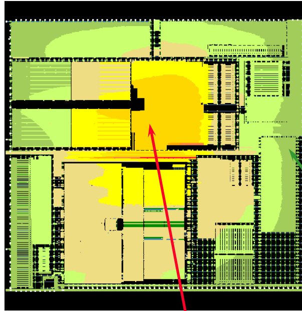
- All 3 signals switch concurrently
- 1x, 2x: 1-grid, 2-grid spacing

Important Deep Submicron Effects

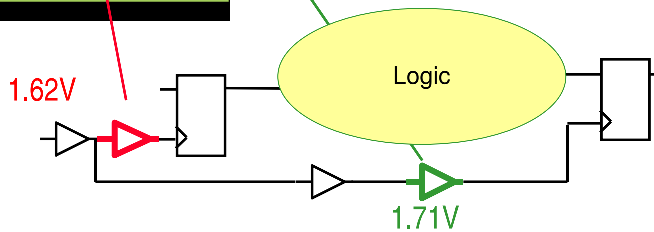
1. Rising relative delay of interconnect
2. Cross-coupled capacitance
3. IR drop
4. Power
   - 4a. Dynamic power consumption
   - 4b. Static power and leakage
5. Electromigration
6. Variability/Reliability

Power, timing inter-related: Instance IR drop impact on delay

A. Khan - Cadence



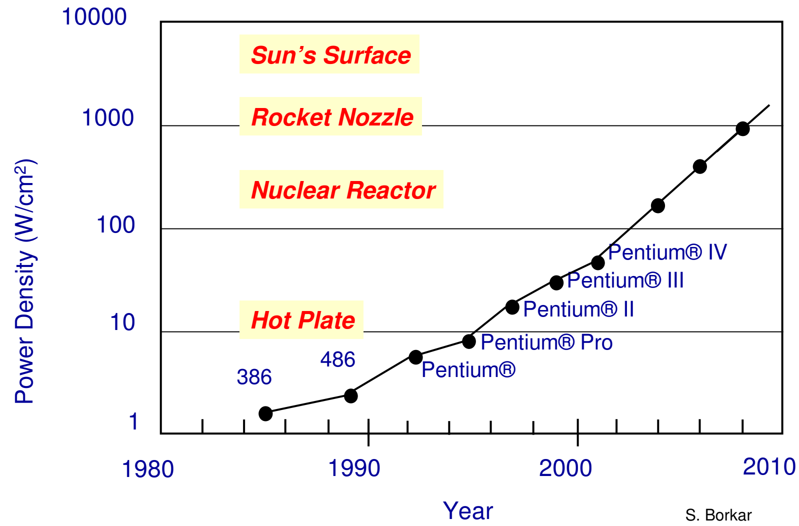
- Buffers get different VDD voltage
- Causes timing closure problems if not accounted for
  - Additional failed paths
  - Race conditions
- Working solution – better analysis tools



Important Deep Submicron Effects

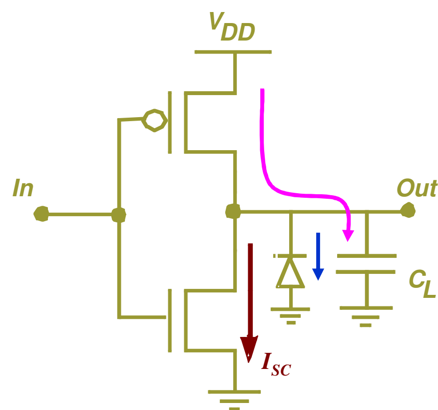
1. Rising relative delay of interconnect
2. Cross-coupled capacitance
3. IR drop
4. Power
   - 4a. Dynamic power consumption
   - 4b. Static power and leakage
5. Electromigration
6. Variability/Reliability

Power dissipation going forward



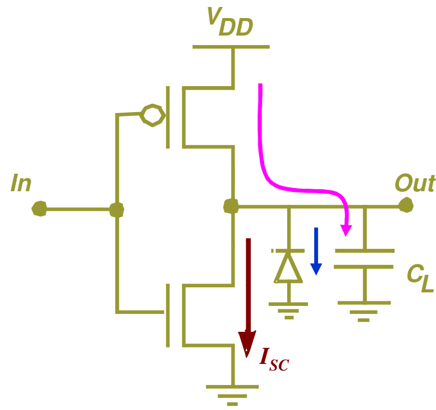
S. Borkar

Power Ingredients



- Dynamic Dissipation
  $P_{dyn} = C_L V_{DD} V_{sw} f$
- Short-Circuit Currents
  $P_{sc} = V_{DD} I_{sc}$
- Static Dissipation
  $P_{stat} = V_{DD} I_{leak}$

Dynamic Power Dissipation



- Dynamic Dissipation

  $P_{dyn} = C_L V_{DD} V_{sw} f$

How do these scale?

C_L

V_{DD}

V_{SW}

f

Scaling Relationships for Long Channel Devices

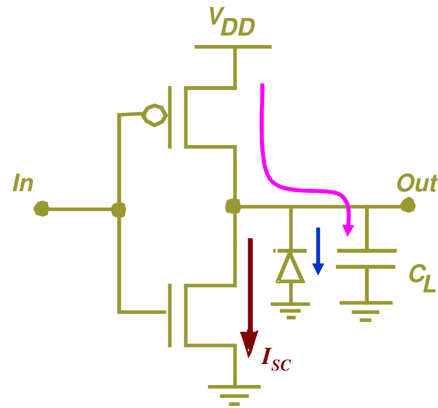
Weste & Harris – CMOS VLSI Design

Parameter	Relation	Full Scaling	General Scaling	Fixed Voltage Scaling
W, L, t_{ox}		1/S	1/S	1/S
V_{DD}, V_T		1/S	1/U	1
N_{SUB}	V/W_{depl}^2	S	S^2/U	S^2
Area/Device	WL	$1/S^2$	$1/S^2$	$1/S^2$
C_{ox}	$1/t_{ox}$	S	S	S
C_L	$C_{ox}WL$	1/S	1/S	1/S
k_n, k_p	$C_{ox}W/L$	S	S	S
I_{av}	$k_{n,p} V^2$	1/S	S/U^2	S
t_p (intrinsic)	$C_L V / I_{av}$	1/S	U/S^2	$1/S^2$
P_{av}	$C_L V^2 / t_p$	$1/S^2$	S/U^3	S
PDP	$C_L V^2$	$1/S^3$	$1/SU^2$	1/S

Table 3.1: Scaling Relationships for Long Channel Devices

f – scaling faster than 1/S- why?

Dynamic Power Dissipation



- Dynamic Dissipation

  $P_{dyn} = C_L V_{DD} V_{sw} f$

How do these scale?

C_L - 1/S

V_{DD} – Full 1/S, Fixed 1, General U

f - S ASIC, >>S for uP

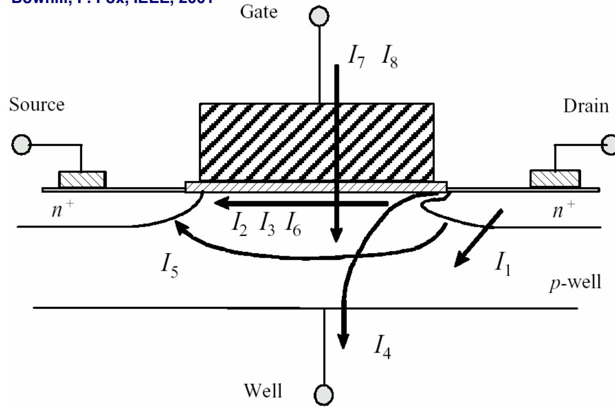
All adds up to higher power density for high-performance designs

Important Deep Submicron Effects

1. Rising relative delay of interconnect
2. Cross-coupled capacitance
   - 2a. Delay degradation in static CMOS
   - 2b. Signal integrity in Analog and Dynamic CMOS
3. IR drop
4. Power
   - 4a. Dynamic power consumption
   - 4b. Static power and leakage
5. Electromigration
6. Variability/Reliability

Leakage current components

These Slides are derived from *Design of High-Performance Microprocessor Circuits*, A. Chandrakasan, W. Bowhill, F. Fox, IEEE, 2001



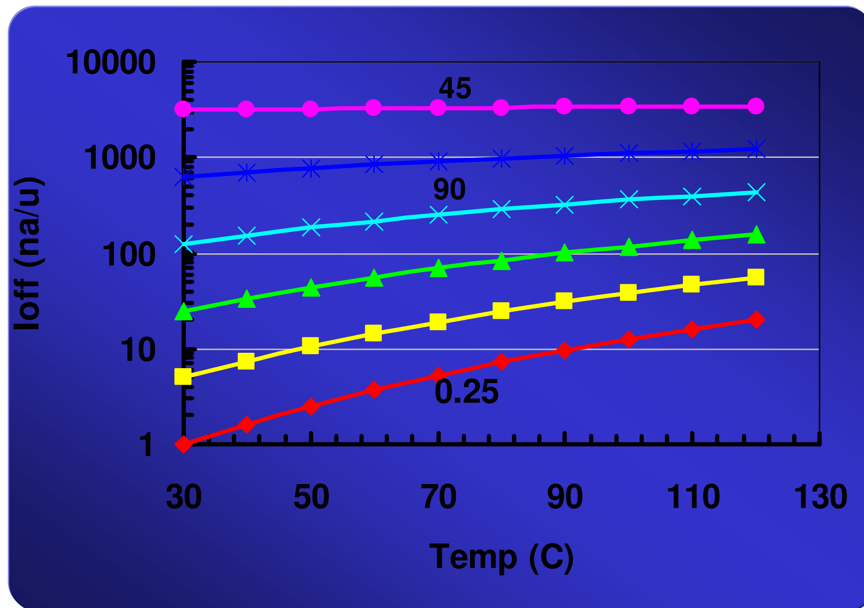
1: pn junc reverse bias
2: weak inversion
3: DIBL –drain induced barrier lowering
4: GIDL –gate induced drain leakage
5: Punchthrough
6: Narrow width effect
7: gate oxide tunneling
8: hot carrier injection

Leakage Current Trends

$$I_{off} = I_0 \exp\left(\frac{-qV_t}{mkT}\right)$$

- **Leakage current *exponentially* increases with reduction in threshold voltage V_t**
- **Leakage projected to grow to 40% of total power on future microprocessors**

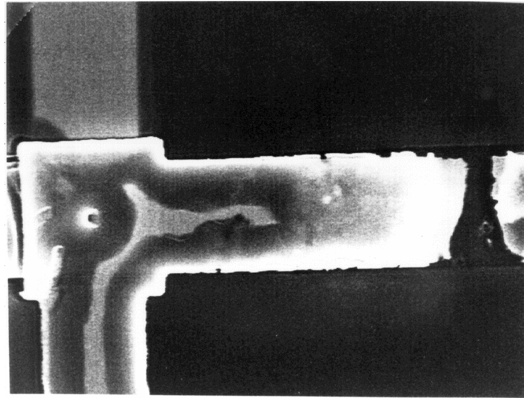
Subthreshold Leakage – Borkar Intel



Important Deep Submicron Effects

1. Rising relative delay of interconnect
2. Cross-coupled capacitance
3. IR drop
4. Power
   - 4a. Dynamic power consumption
   - 4b. Static power and leakage
5. Electromigration
6. Variability
7. Reliability

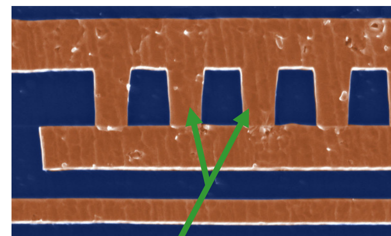
Electromigration (1)



Limits dc-current to 1 mA/μm

Solution – Materials: Copper

- With cladding and other effects, Cu ~ 2.2 μΩ-cm vs. 3.5 for Al(Cu) ⇒ 40% reduction in resistance
- Electromigration improvement; 100X longer lifetime (IBM, IEDM97)
  - Electromigration is a limiting factor beyond 0.18 μm if Al is used (HP, IEDM95)



Important Deep Submicron Effects

1. Rising relative delay of interconnect
2. Cross-coupled capacitance
3. IR drop
4. Power
   - 4a. Dynamic power consumption
   - 4b. Static power and leakage
5. Electromigration
6. Variability – covered in earlier lecture
7. Reliability – covered in earlier lecture

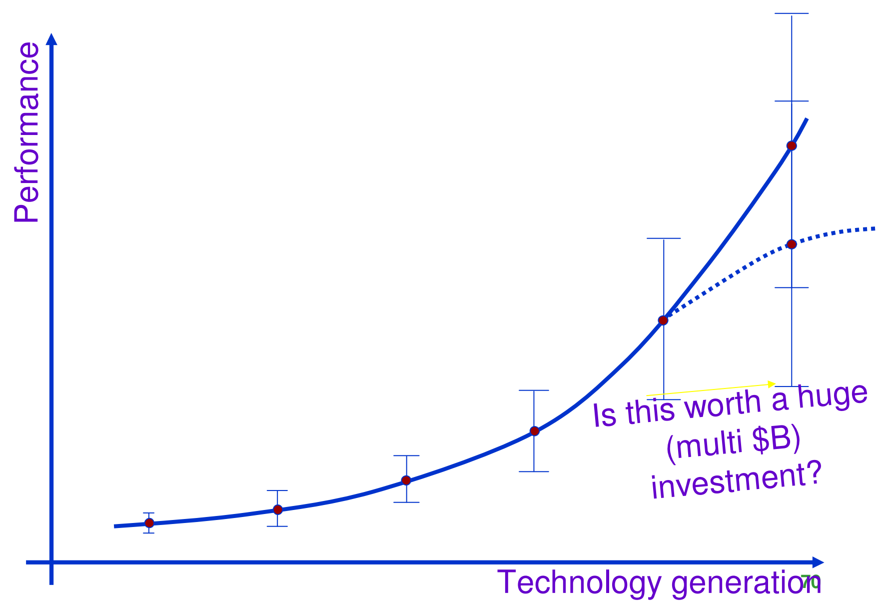
DSM Effects

- During the golden era of ASIC (3.5u to .18u) there was little concern for physical effects during design.
  - Area
  - Delay
  - Power
- Were all relatively predictable and insurable
- With smaller geometries (.18u and below) physical effects began to demand greater and greater attention in design. Basic design parameters were neither predictable or insurable
- But moving higher in the design hierarchy (RTL → Behavioral → "System-level") requires building on top of predictable layers
- As a result the design process has been stalled at the RTL level as we try to bring netlist implementation back to the predictability and reliability that we had 20 years ago!
- Focus of research in the RTL design flow has been on managing these DSM effects

Process Variability

- Increased device variation
- Increased interconnect variation
- Result
  - Slower/unpredictable speed of digital circuits
  - Greater uncertainty about parameter values
  - Over-conservatism
  - Reduced signaling robustness
  - Increased clock skew

The march of technology

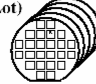


Where does it Come from?

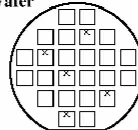
- Inter-Die Variation
- Intra-Die Variation

Lot–to–Lot

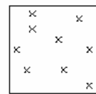
Wafer–to–Wafer
(or within Lot)

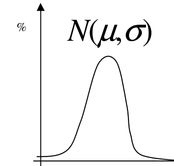


Within Wafer



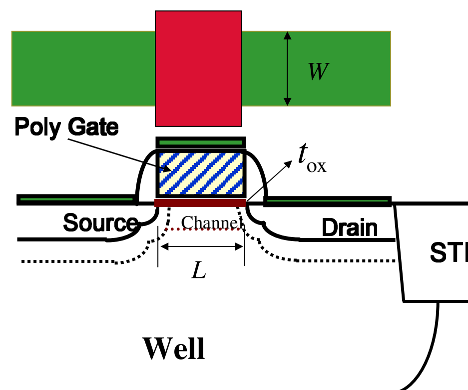
Intradie





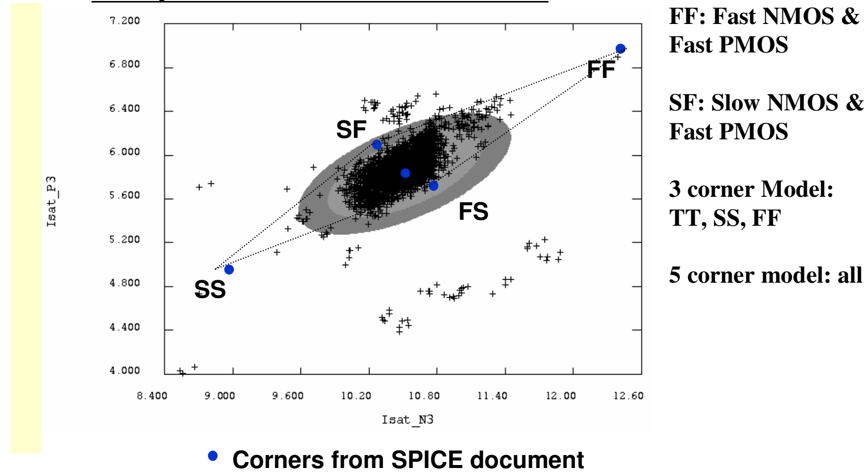
Device Parameter Variations

- W, L variations
  - Due to photolithography proximity effect or etching
  - Layout density dependent
  - Location dependent
- Tox variation
  - Well controlled by a product spec.
- Vth variation
  - Due to doping



Device Worst Case Model

Comparison Corner vs Real Data



FF: Fast NMOS & Fast PMOS

SF: Slow NMOS & Fast PMOS

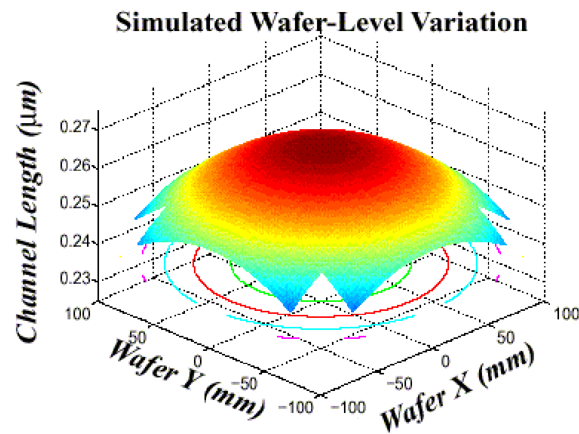
3 corner Model: TT, SS, FF

5 corner model: all

- Corners from SPICE document

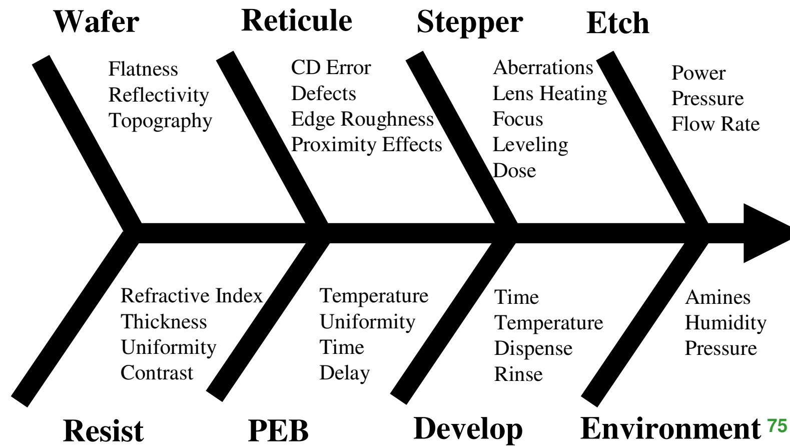
Within Wafer Channel Length Variation

- The most important source of variation is Lgate variation
  - Determines speed of the transistor through effective channel length



Sources of CD Variation

- Among lithography people, Lgate is known as CD – the critical dimension (because it's so important)

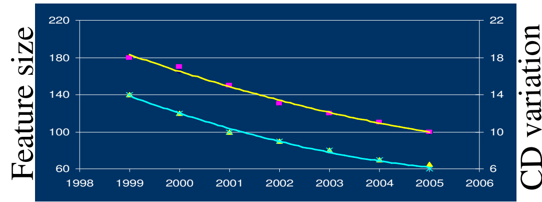


SIA thinks that CD should be allowed a 10% error budget – why?

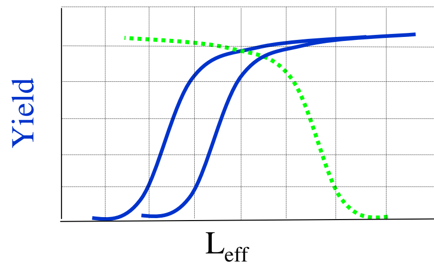
Table 39a Lithography Technology Requirements—Near Term

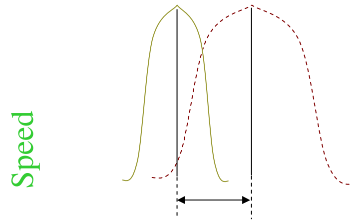
Year Technology Node	1999 180 nm	2000	2001	2002 130 nm	2003	2004	2005 100 nm
DRAM							
Half pitch (nm)	180	165	150	130	120	110	100
Contacts (nm)	200	185	170	150	145	140	130
Overlay (nm, mean + 3 sigma)	65	58	52	45	42	38	35
CD control (nm, 3 sigma, post-etch)	18	17	15	13	12	11	10
MPU							
Half pitch	230	210	180	160	145	130	115
Gate length (nm, in resist)	140	120	100	90	80	70	65
Gate length (nm, post-etch)	140	120	100	90	80	70	65
Contacts (nm, in resist)	230	210	180	160	145	130	115
Gate CD control (nm, 3 sigma, post-etch)	14	12	10	9	8	7	6

Every nm of CD spread reduction means about $10 of more revenue per CPU chip….



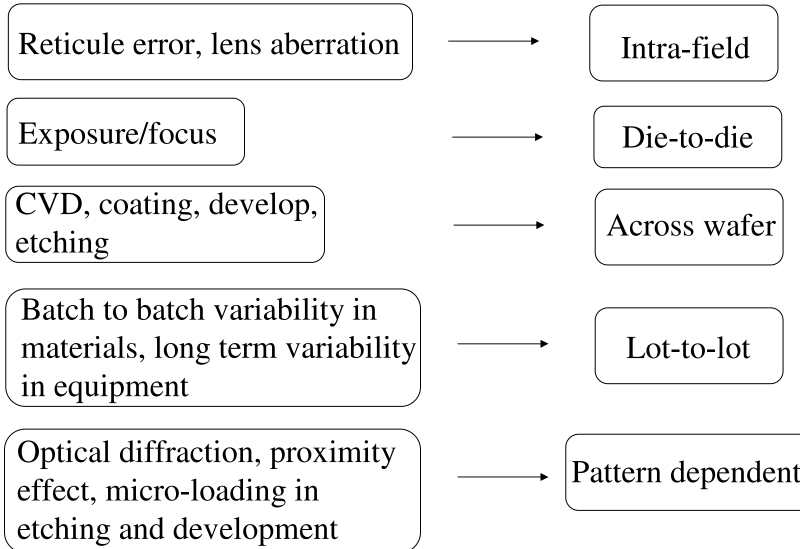
Source: ITRS 1999





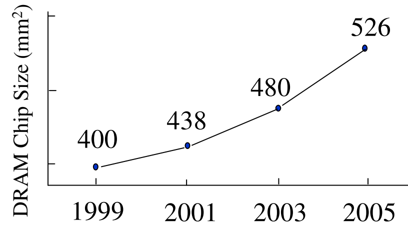
$7.5/nm/chip for the 0.25μm technology

Decomposition of Sources of Variability



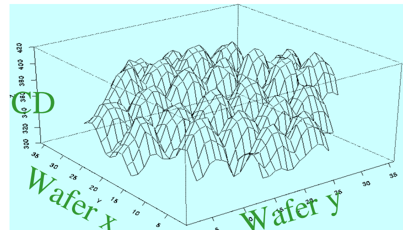
Why is Intra-field Variation Critical?

field size increases by 12%/year to accommodate 59% more components per year by Moore's Law.



Source: ITRS 1999

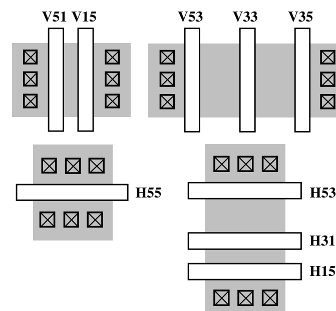
- Spatial variability is mainly *systematic* instead of random.
- It *could* be compensated by process optimization or circuit design, or special mask design.



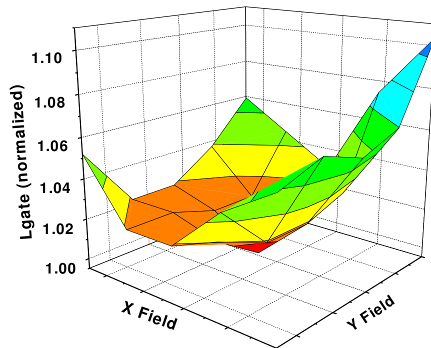
Source: BCAM previous work

Lgate Varies Depending on Local Layout Patterns

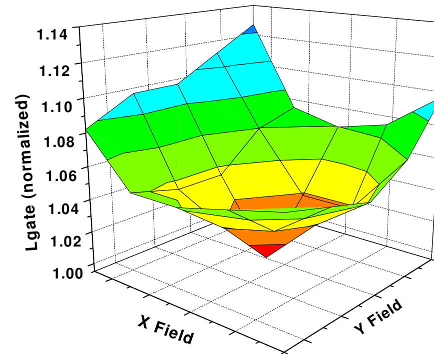
- **For full characterization, gates classified by their local layout patterns**
- **Gate are classified by:**
  - ***A) Distance* to neighboring gate (proximity effect)**
  - ***B) Left vs right* neighbor position (coma effect)**
  - ***C) Vertical vs horizontal orientation***



Spatial Maps for Different Gate Categories



Category V53

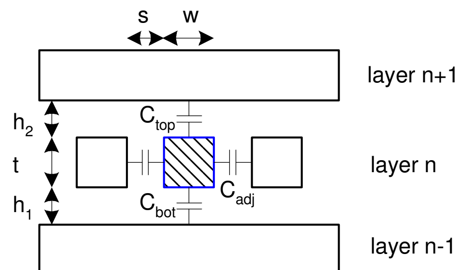


Category V33

- Two spatial profiles are statistically different
- Separate models need to be used at the CAD level

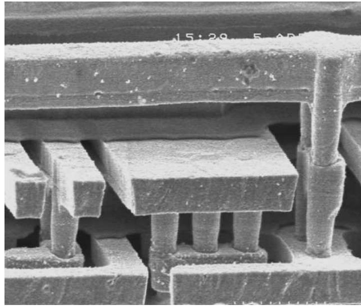
Interconnect Parameter Variations

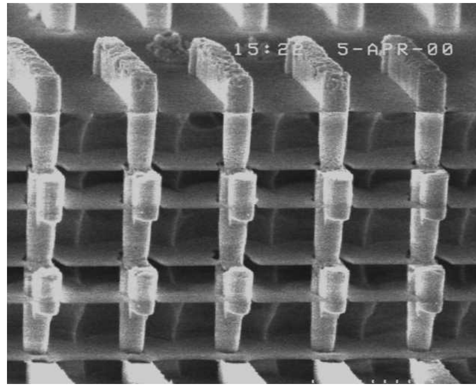
- **Line width(w), spacing(s)**
  - **Due to photolithography proximity effect or etching**
  - **Layout dependent**
  - **Location dependent**
- **Metal thickness(T)**
  - **Due to erosion, dishing**
  - **Layout density dependent**
- **Inter-layer Dielectric (ILD) thickness(H)**
  - **Due to CMP**
- **Dielectric Constant(ro)**



All layer Cu/Low-K Interconnect

Current process technology for interconnect with multiple layers of metal/dielectric

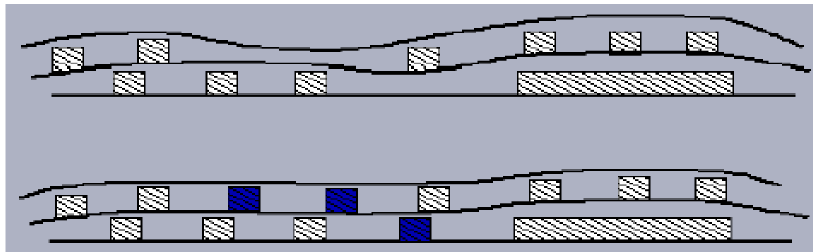




From TSMC 0.13um technology

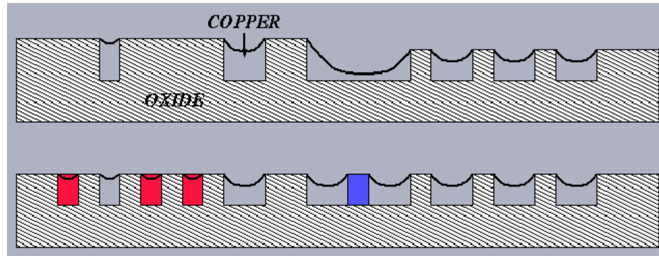
Planarity of Al Metal CMP Processes

- Chemical-mechanical polish (CMP) rate is different for sparse and dense areas
- Tiling: adds new features in sparse areas to ensure better planarity
- Design problem: determine location and amount of dummy features needed to achieve a planarity



(Grobman, DAC2001)

Planarity In Copper CMP Processes

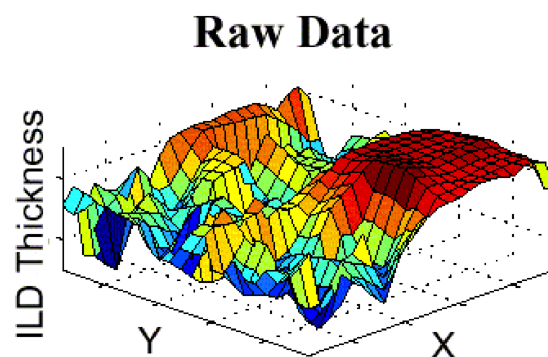


- For Cu processes have two problems
  - Oxide erosion
  - Copper 'dishing'

(Grobman, DAC2001)

Intra-Die ILD Thickness Variation

- Within die variation



ILD Thickness Variation

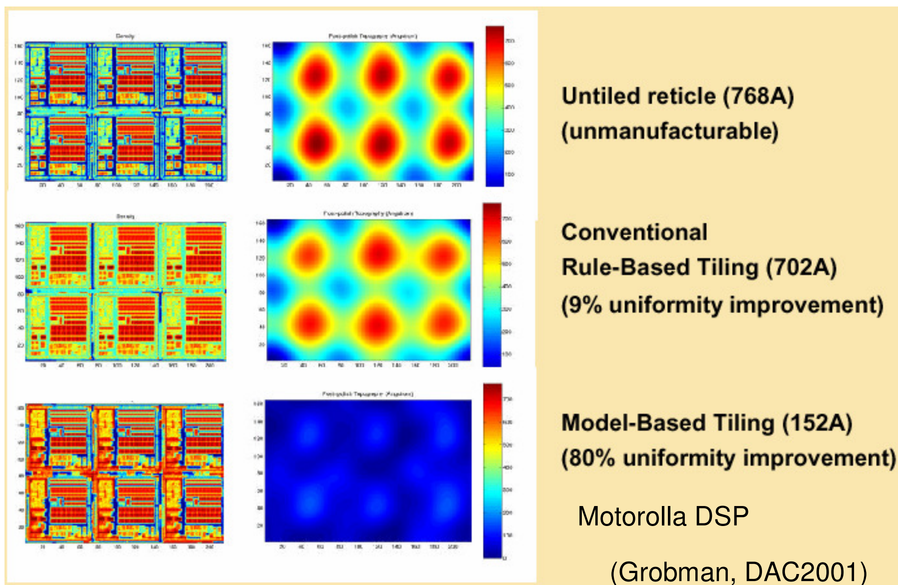
- TSMC specs on ILD Variation
  - Variation is up to 20%
  - Modest (3% ?) impact on timing

Dielectric layers

Dielectric	Thickness	%Var	Dielectric constant	Comments
FOX	3500 □	± 17.1%	3.9	
ILD	7000 □	± 21.4%	4.0	See NOTE 1.
IMD1a IMD1b	11300 □ 2000 □	± 20% ± 3%	3.7 4.2	See NOTE 1.
IMD2a IMD2b	11300 □ 2000 □	± 20% ± 3%	3.7 4.2	See NOTE 1.
IMD3a IMD3b	11300 □ 2000 □	± 20% ± 3%	3.7 4.2	See NOTE 1.
IMD4a IMD4b	11300 □ 2000 □	± 20% ± 3%	3.7 4.2	See NOTE 1.
IMD5a IMD5b	11300 □ 2000 □	± 20% ± 3%	3.7 4.2	See NOTE 1.
PASS1 PASS2	10000 □ 7000 □	± 10% ± 10%	4.2 7.9	See NOTE 2. Conformal material.

NOTE 1. The dielectric layers of ILD, IMD1b, IMD2b, IMD3b, IMD4b and IMD5b outside the metal are overetched 1000□

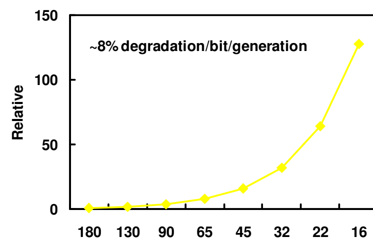
Tiling for Better Planarity



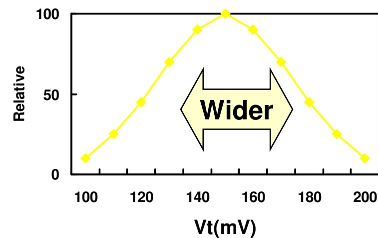
Important Deep Submicron Effects

1. Rising relative delay of interconnect
2. Cross-coupled capacitance
3. IR drop
4. Power
   - 4a. Dynamic power consumption
   - 4b. Static power and leakage
5. Electromigration
6. Variability
7. Reliability

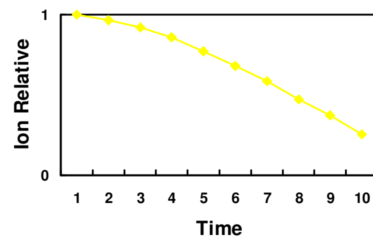
Reliability



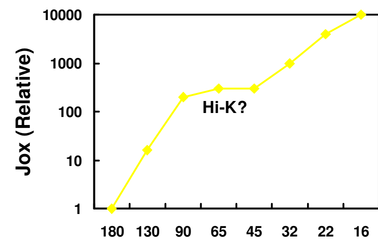
Soft Error FIT/Chip (Logic & Mem)



Extreme device variations



Time dependent device degradation

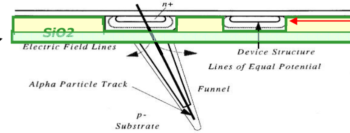


Burn-in may phase out…?
Chip infant mortality?

SER Mitigation in Microprocessors

Mike Leary – AMD

- Manufacturing Techniques
  - **SOI technology (eg, AMD Opteron™ processor) reduces SER compared to bulk**
    - **Charge generated below BOX cannot be collected**
  - **Eliminate alpha-producing materials from chip environment**
    - **BPSG**
    - **Low-alpha package materials and underfill**
    - **Low-alpha lead for C4 bumps reduces emissivity by >1000X**



- Design Techniques
  - **Parity: Detects SEU upsets**
  - **ECC: Corrects single bit fails; detects dual-bit fails.**
    - **Reduces SER by several orders of magnitude in the protected arrays.**
    - **Typically, over 90% of SER-susceptible bit are either parity or ECC-protected in Server processors**
  - **Cache Line Interleaving: Physical separation of logically adjacent bits to greatly reduce multibit fails**
  - **Scrubbing: Cache scrubbing in background to do background correction of SEU**
  - **Hardening: selective nodes made SEU-resistance by device size tweaks**
- Operational Techniques
  - **For mission-critical applications, operate at highest voltage possible**
    - **SER is strongly voltage dependent**
  - **Try to avoid high-altitude operation! In fact, try to operate in deep mines!**
    - **At airline altitudes, Neutron flux is ~300X higher than sea level**