Lecture 16

Design for manufacturability and new layout rules

Computer Systems Laboratory
Stanford University
ronho@vlsi.stanford.edu

Overview

- Today's topics
  - – Layout rules to ensure manufacturability
    - Metal density rules, both min and max
    - Antenna rules
    - Resolution enhancement techniques
    - Logos
  - – (Time permitting) Soft-errors and dealing with them

In your classes or jobs, most of you have used layout tools, and have had experience satisfying layout design rules, such as minimum widths, minimum spacings, or minimum surrounds. These rules get more and more complicated with each technology generation. This lecture will discuss some of the rules you may not have seen yet, why they exist, and how to deal with them.

Layout rules

- With technology scaling, the old rules get strict in different ways
  - Vias
    - Can no longer run a string of single stacked vias from M1 to M10
    - Although this is okay if you have a block of them arrayed out
  - Metals
    - Wider metals require wider spacings (a pain to memorize!)
  - Transistors
    - Can only run in a single direction, at a single spacing
    - Can no longer share diffusions between transistors of different widths

- There are some newer rules, too

Metal Density Requirements

- Density of metal usage has a min and a max constraint
  - Limits are typically 30% < density < 80% for metal
  - Poly also has density requirements, typically 15% < density < 80%
  - So does diffusion (for transistor regularity and matching)

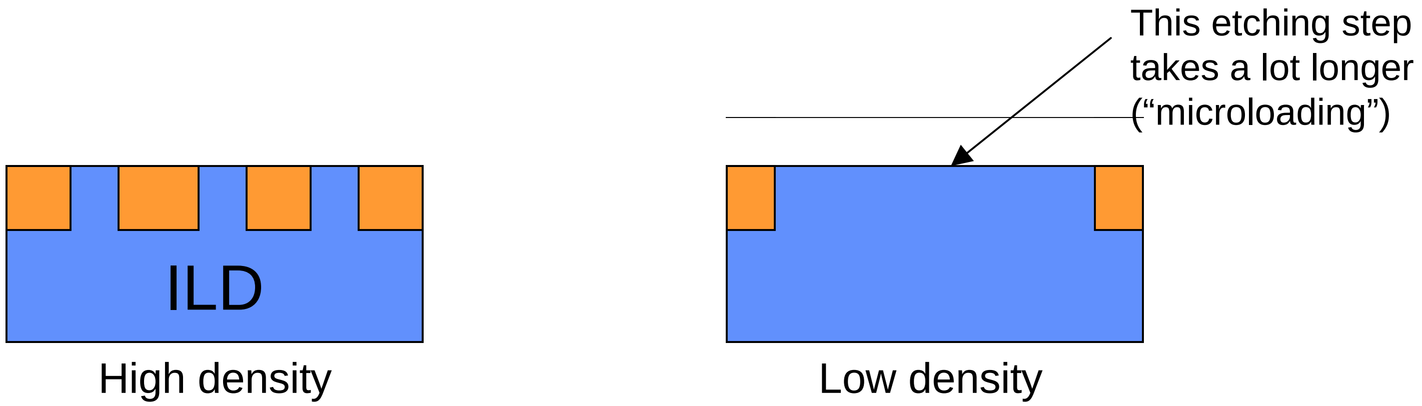
- Density needs to be checked both globally and locally
  - Globally: over the entire chip
  - Locally: over 500μmx500μm sliding windows (typically)
  - Need CAD tool support for the local checks

- Why check?
  - Density affects etch rates
  - Large variations in density can cause thermal expansion stress

Aluminum Density Rules

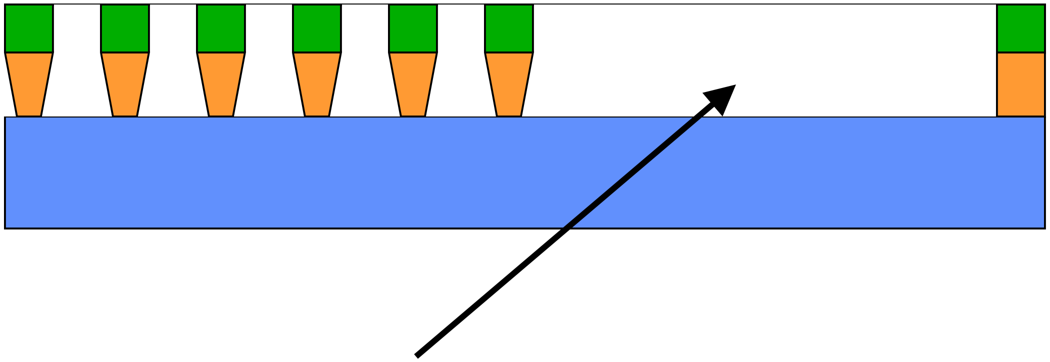
- More of a historical note, now that everybody's moved to Cu
  - Cu requires a non-conductive barrier layer (usually Tantalum)
  - Cu also suffers from worse surface scattering than Al
  - Will Al come back? (Probably not, but an interesting question)

- Wires are subtractive with Aluminum
  - Lay down a sheet and then remove what you don't need
  - Depending on density, this may take more time to etch



Aluminum Density Rules, con’t

- Total etch time set by the low-density regions
  - Need enough time to finish clearing out the metal
- If Aluminum density is too low, then etch time becomes severe
  - May over-etch a high-density region

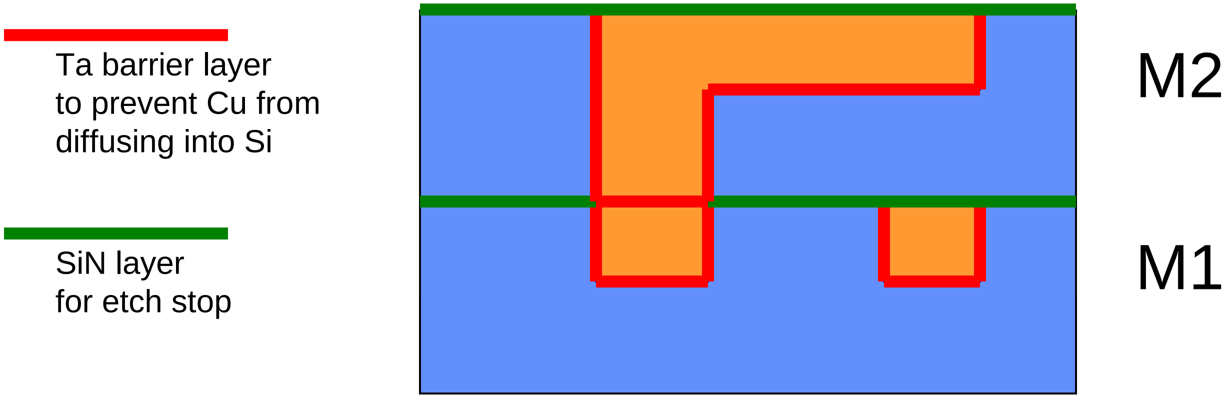


- Adding some dummy metals at the right evens out the density
  - Prevents over-etch of metal at the left
  - Use “fill cells” to increase metal density – wires tied to Vdd or Gnd

Copper Density Rules

- Copper metals created in an additive (not subtractive) process
  - Cut the openings, then “pour” in the Copper
  - A “damascene” process (as in inlaying patterns into sword blades)

- Here’s an amateur’s view of “dual-damascene” processing
  - Damascene – making a Copper metal layer
  - Dual-damascene – making a metal layer and vias at the same time

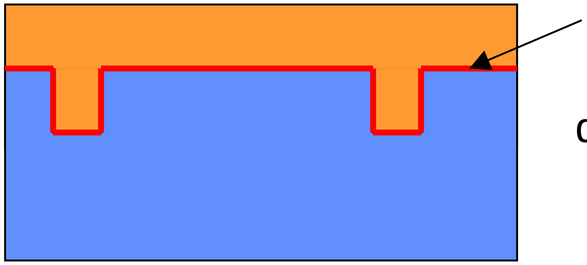


🟥 Ta barrier layer to prevent Cu from diffusing into Si

🟩 SiN layer for etch stop

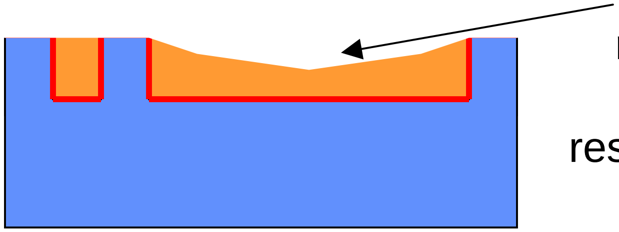
Copper Density Rules, con’t

- There are two density rules, covering both min and max density
  - Minimum density comes from the difficulty of removing Ta barrier



Barrier is tough to remove.
If you have more metal
density, there is less barrier.

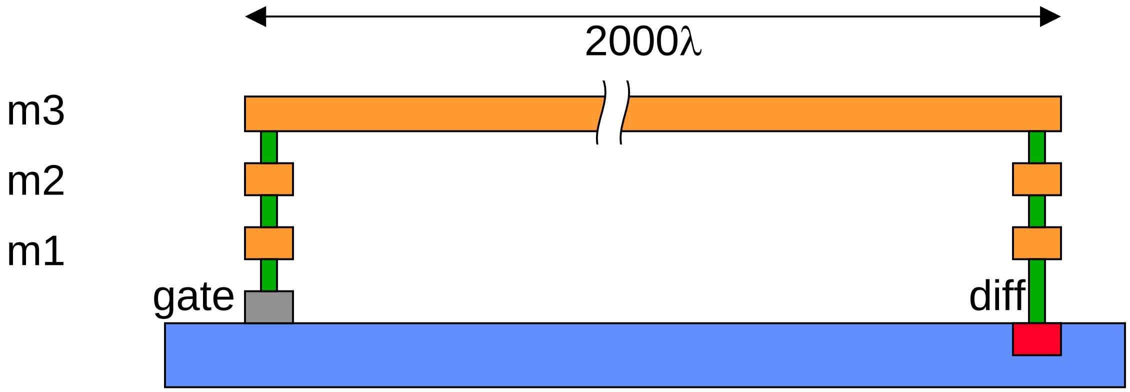
  - Maximum density comes from the softness of Cu versus Ta
    - “Selectivity” of Cu is 20x higher than it is for Ta
    - Add slots to wide metal



Softness of Cu
results in “dishing” –
this leads to higher
resistance than expected

Antenna Rules

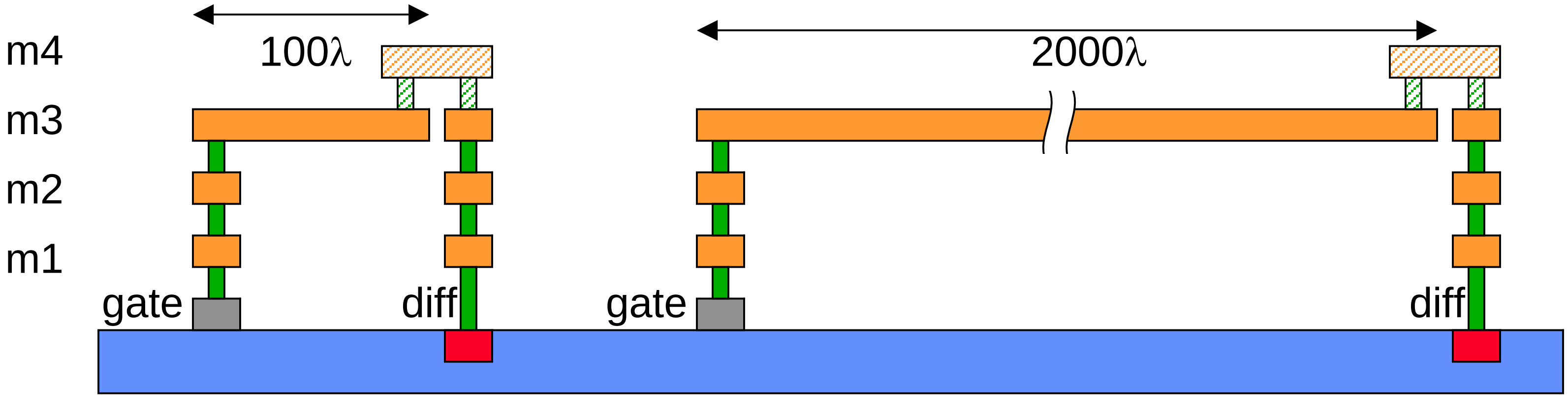
- When a metal line is fabricated, it can act as an antenna
  - Reactive ion etching causes charge to accumulate on the wire
    - Usually on the edges of wires, so the wire perimeter is key
  - If the wire is attached to an n-diffusion, charge will drain harmlessly
  - Substrate is grounded during fab, so you get leakage current



Safe: m3 wire “sees”
the diffusion, so charge
can leak away harmlessly

Antenna Rules

- However, the charge on the antenna may destroy a gate
  - This is only a problem if the wire is “long enough”
  - Although almost every wire on a chip will have a diffusion…
  - …the question is whether the gate or the diffusion is “seen” first

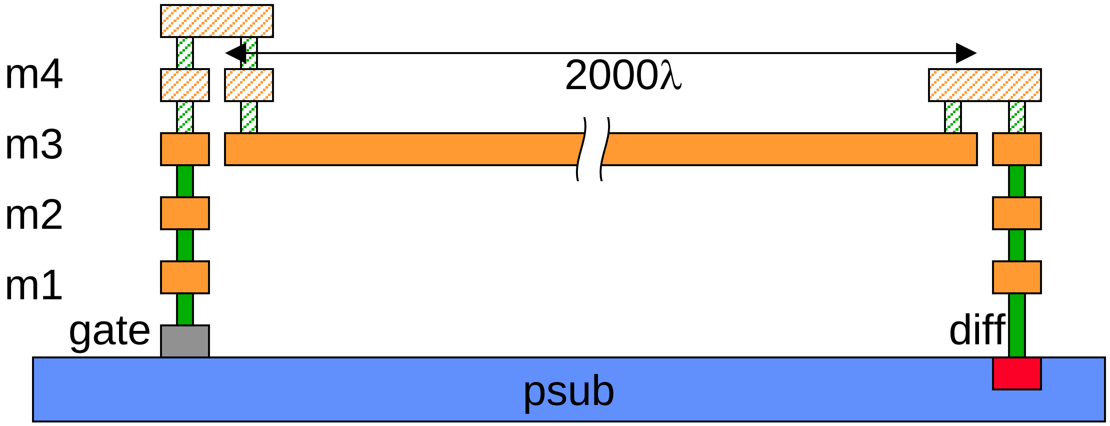


Safe: m3 is too short to accumulate very much charge; won’t kill gate

Dangerous: lots of m3; may accumulate lots of charge and then blow oxide

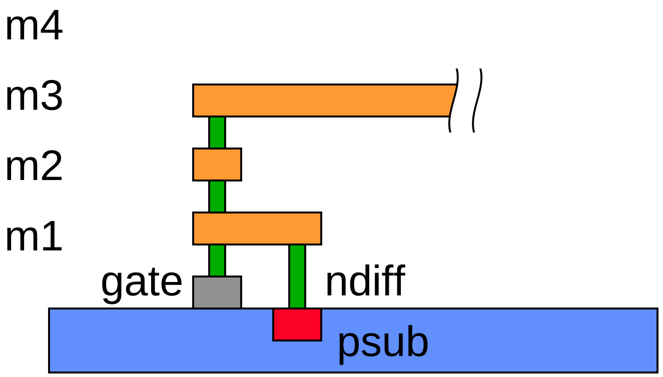
Antenna Rule Fixes

- One fix is to add bridges to the layout
  - Ensure that the long wire “sees” diffusion first, and gate second
  - Remember, wires are built from the lower layers up
  - Costs us the routing resources on the upper metal layers
    - Not a very commonplace fix for antenna violations



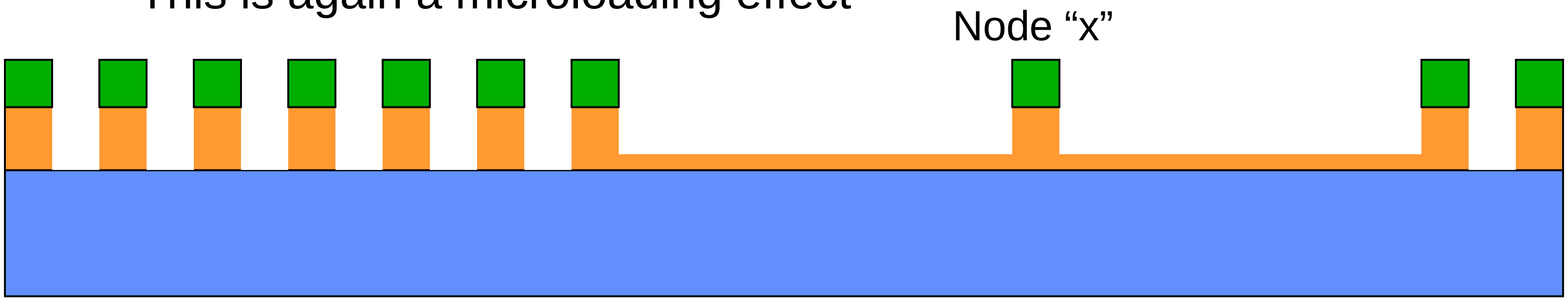
Antenna Rule Fixes, con't

- Can also add a piece of "drainage" diffusion near the gate
  - This reversed-bias diode does not affect chip operation (much)
  - During fab, it allows charge to drain away harmlessly
  - Area of diode set by ratio of wire perimeter and gate area
  - This is the common way to fix antenna violations



Checking Antenna Violations

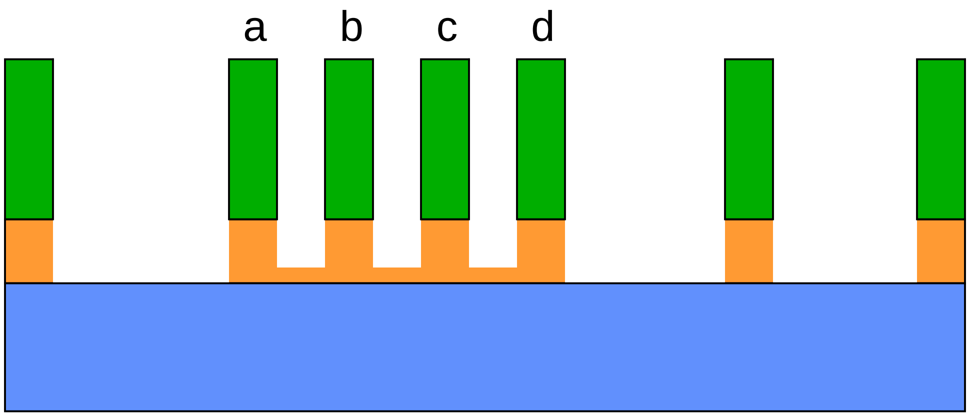
- Check antennas incrementally (only M1, then add M2, etc…)
- Beware of differential etch rates (mostly for Al)
  - Slower etching may cause more perimeter for a metal wire
  - This is again a microloading effect



- Due to microloading, a large "island" of metal exists at node X
  - Only temporary, but if node X was close to antenna limits already…
  - And if the "bridged" wires are also close to antenna limits…
  - Then node X could have its gate destroyed

Checking Antenna Violations, con't

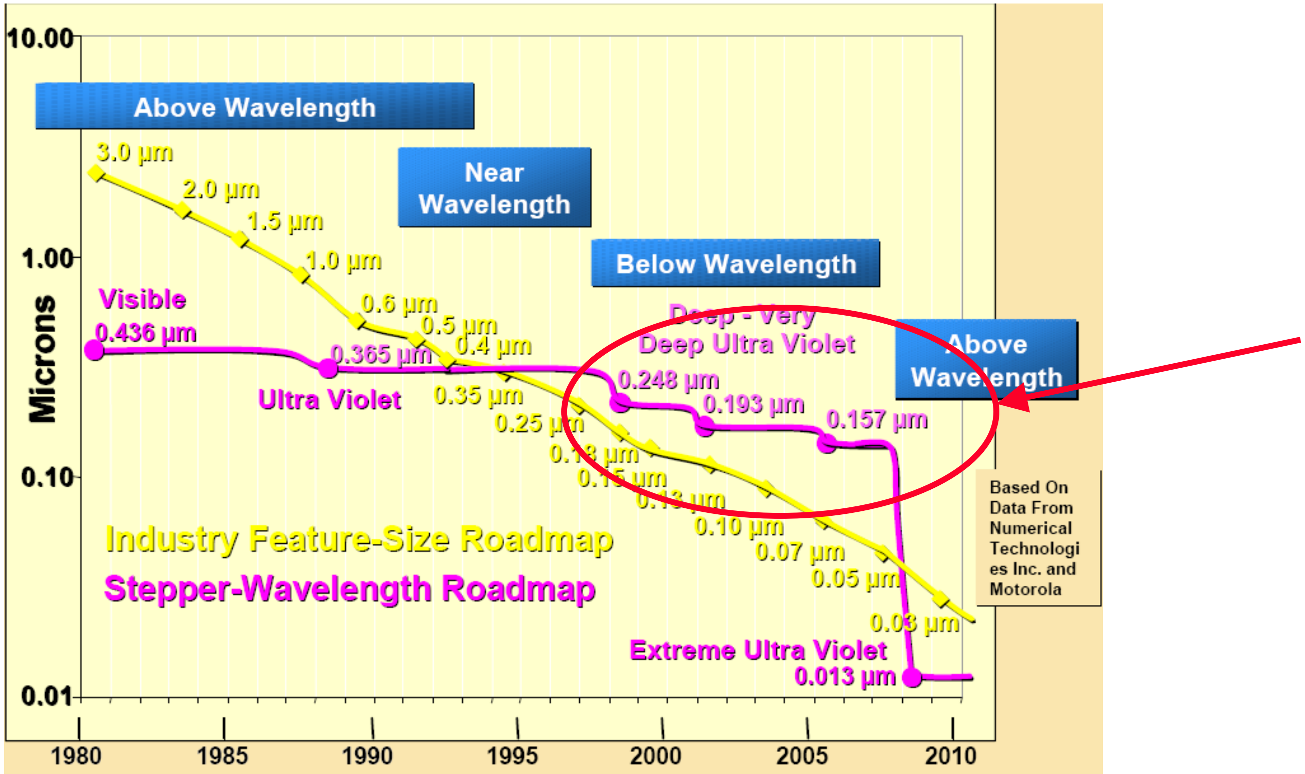
- You can also have unequal etching effects from proximity
  - Called e$^-$ shading, and more a problem if resist is tall and skinny
  - Etchant particles don't enter the "troughs" as easily
  - Differential in etch rates cause islands to once again appear



- Here, node "c" has an "etch-island" that includes a, b, and d
  - Again, only temporarily, but can cause antenna problems

Lithography Effects

- Lithography uses light to image the features on a chip
  - Essentially, a **very** expensive light projector and a stencil
  - The wavelength of the light sets the feature sizes attainable



Um... Light is **coarser** than the features that we need!

(A function of the high cost of steppers.)

Source: Grobman, DAC 2001

Dealing With Resolution Shortfalls

- Three principal methods used today – all described later
  - Phase-shifted masks (PSM)
  - Off-axis illumination (OAI)
  - Optical proximity correction (OPC)
- These increase the ability to “focus” light onto the wafers

- PSM and OAI increase the difference in light intensity at edges
  - Sharpen the optical transitions between light and dark

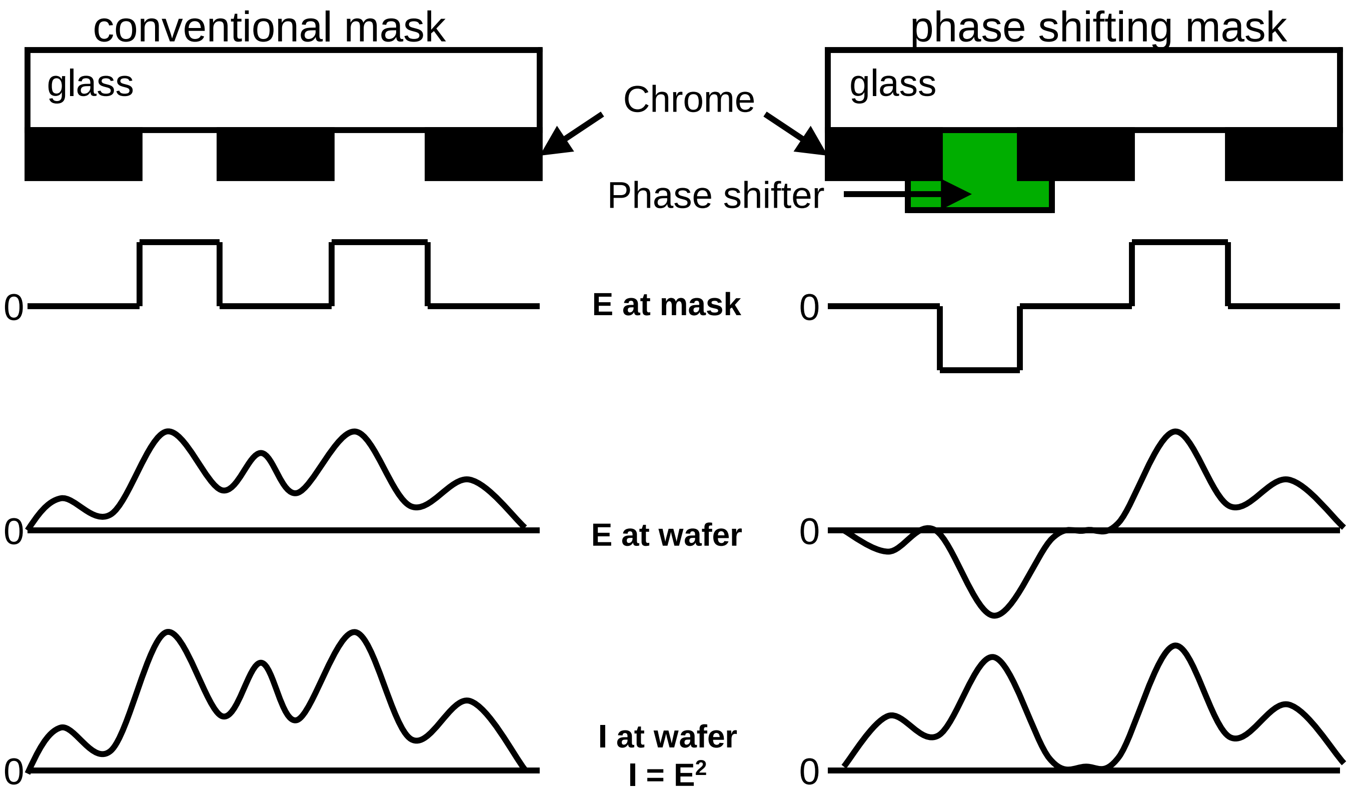
- OPC performs a spatial pre-emphasis function
  - Lack of optical focus is essentially a lossy channel

Dealing with resolution shortfalls

- Increase the numerical aperture of the imaging light
  - Basically, widen the angle of light that can be collected to target
  - This will capture longer paths (higher diffraction orders)
  - Analogous to higher order terms in a fourier expansion
- Immersion lithography buries the lens in water or high-index fluid
- Eventually, we will move to Extreme UV light (13nm)
  - But this is pretty challenging in itself
  - Everything absorbs 13nm (air, and even to a partial extent, mirrors)
  - Aimed at the 22nm generation and smaller

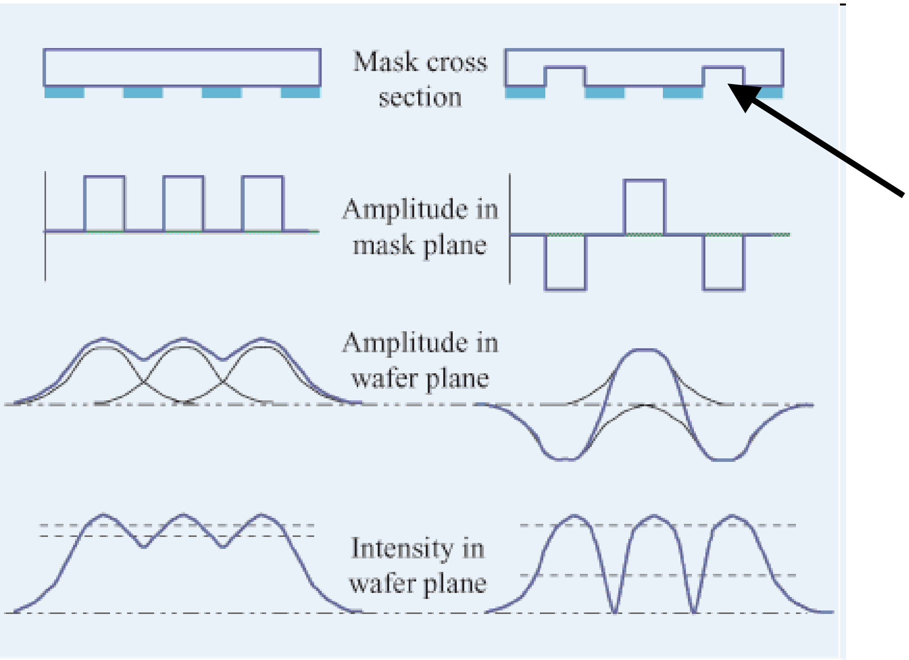
PSM

- Increase intensity peaks by inverting light phase selectively
  - Phase shifter material delays edge to give 180º out-of-phase light



Source: Kahng, DAC 1999

Another View of PSM



Phase shifting can also be done using a thinner mask section, not by adding a layer.

Figure 13

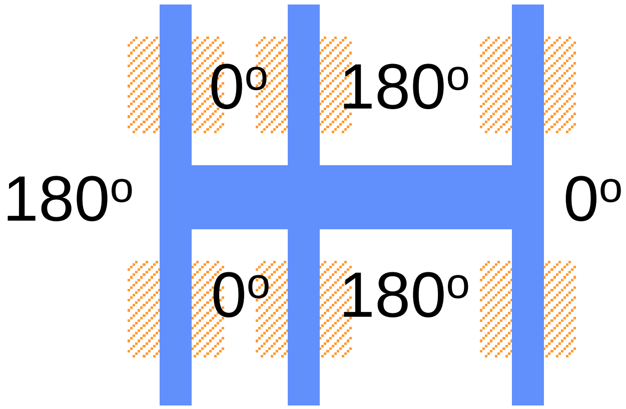
Schematic diagram comparing conventional binary mask lithography (left) and phase-shifted-mask lithography (right). The path-length difference in alternate patterns in the phase-shifted mask causes light with amplitudes of equal magnitude but opposite sign to be transmitted through neighboring mask openings.

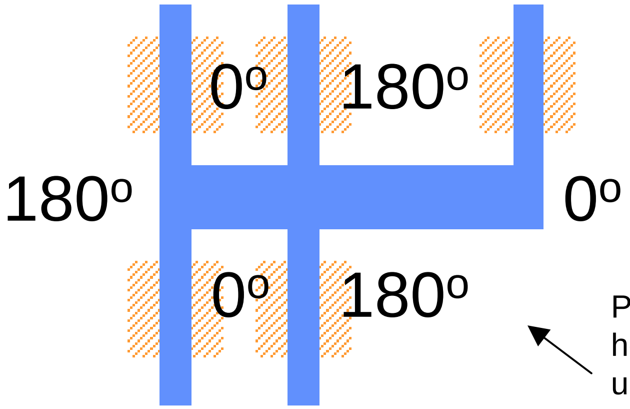
Source: Liebman, IBM JRD, 2001

PSM and Layout

- Generally only do PSM on poly, maybe on contact
  - The mask is expensive (3x); save it for the really fine feature layers

- PSM requires every feature to have two “sides” (0^{o} and 180^{o})
  - This is a two-color map problem
  - Of course, two-color maps are not generally solvable…

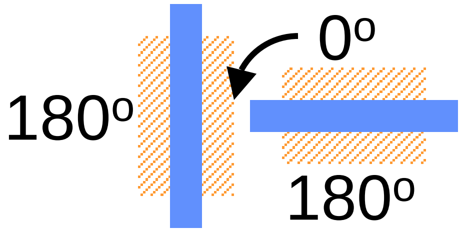




Phase conflict here will create an unwanted line; need “trim mask” to kill it

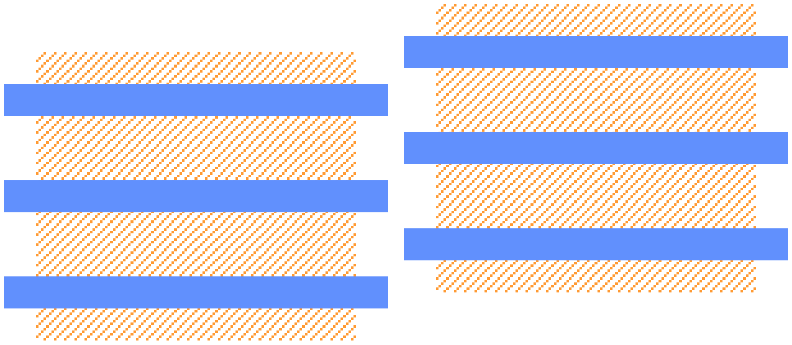
PSM and Example Problem Layouts

- You cannot abut orthogonal gates (“Tee-junction gates”)
  - You need a 0º section to the right of the vertical gate



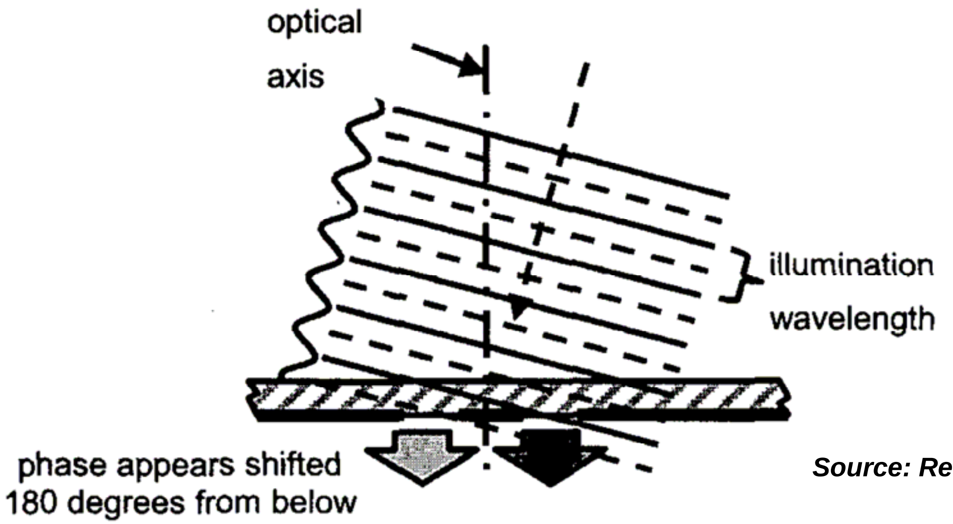
Probably not a huge problem, given that transistors all must run in the same direction today.

- You cannot interdigitate fingers (e.g., to duck the poly-poly rule)
  - 0º and 180º sections collide



OAI

- Another way to achieve phase interference is to angle the light
  - You get characteristics very similar to PSMs
  - Mask generation a little simpler, because only need a pair of slits

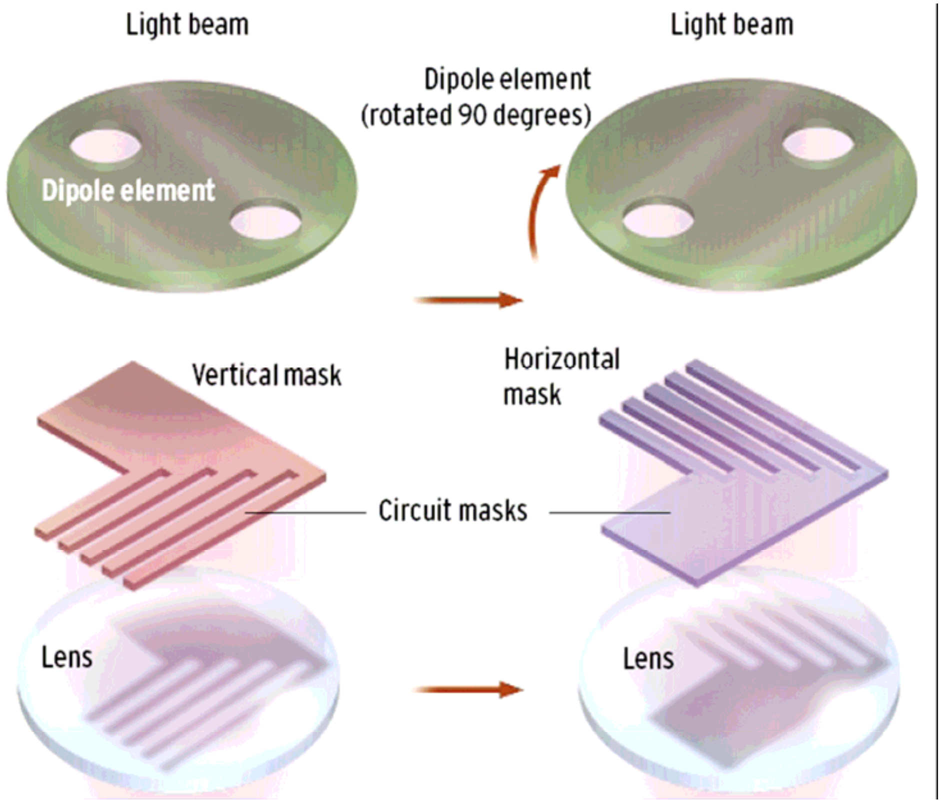


Source: Reiger, DAC 2001

- Constraints include unidirectional layout
  - Some widths will not pattern (destructive interference) – need grids!

Applying OAI to make fine patterns

- A prettier picture of OAI
  - Orthogonal lines need different dipole mask rotations



Source: Schellenberg, Spectrum 2003

More Alphabet Soup: OPC

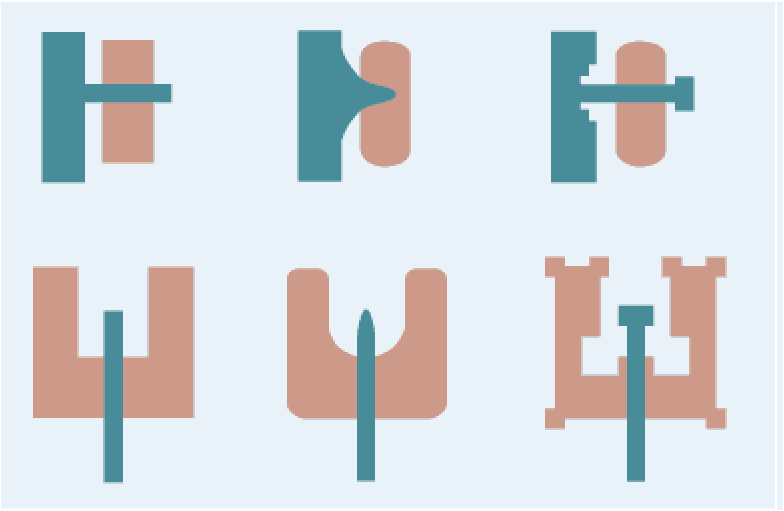
- Optics that image features on a wafer act like a lossy channel
  - Low-pass filter on the spatial resolution

- You want to image very sharp edges, but you don't
  - Rectangles end up imaging as blobby ovals
  - Sharp corners end up as sloppy turns
  - Layout is no longer WYSIWYG

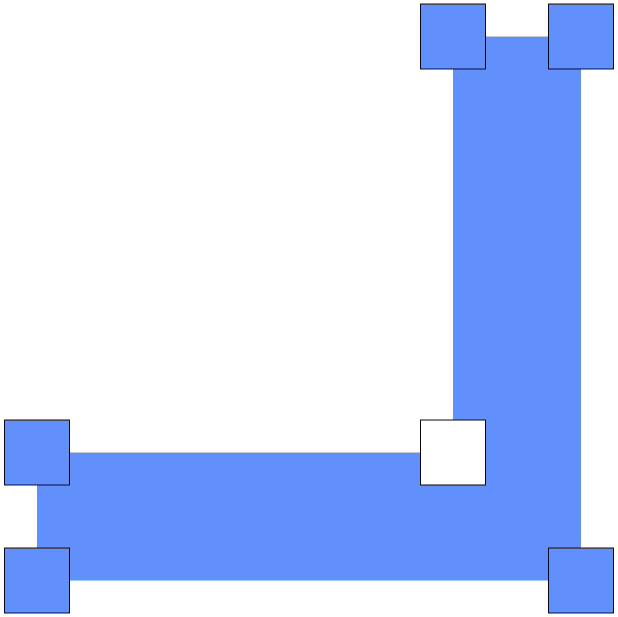
- OPC predistorts the high-frequency spatial components
  - Just like we pre-emphasize high-freq time components in datacom
  - Not done by the designers directly; done in back-end flow
  - Your masks end up looking very different from drawn layout
  - Tends to explode the database size

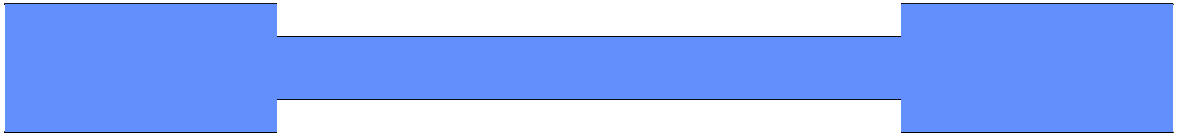
OPC Examples

- Pre-distortion of the image mostly at corners and along lengths



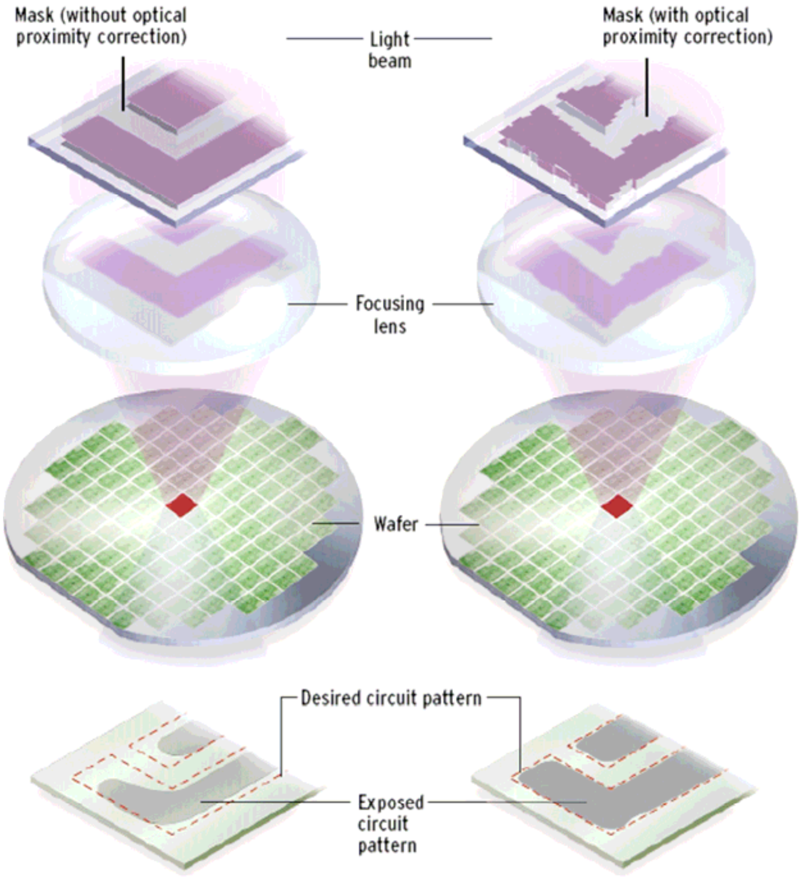
Source: Liebman, IBM JRD, 2001



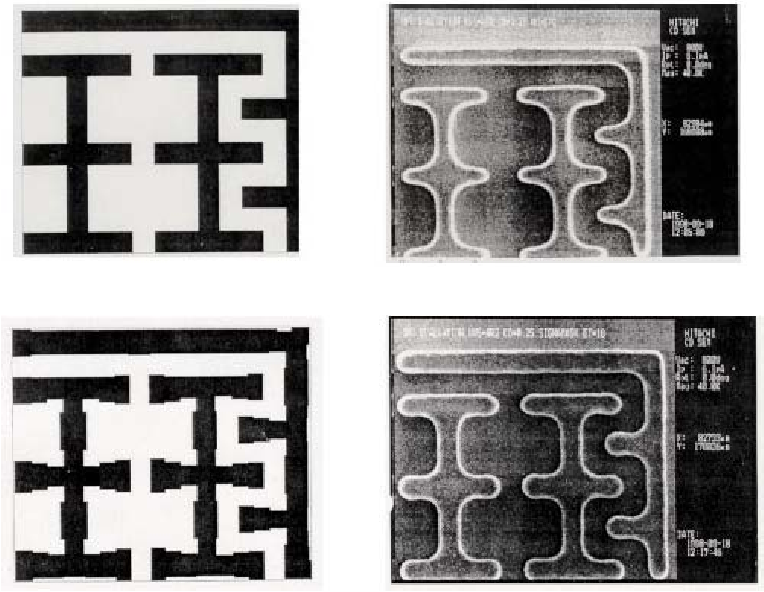


OPC Examples, con't

- These features are called serifs, "ears," and "dogbones"

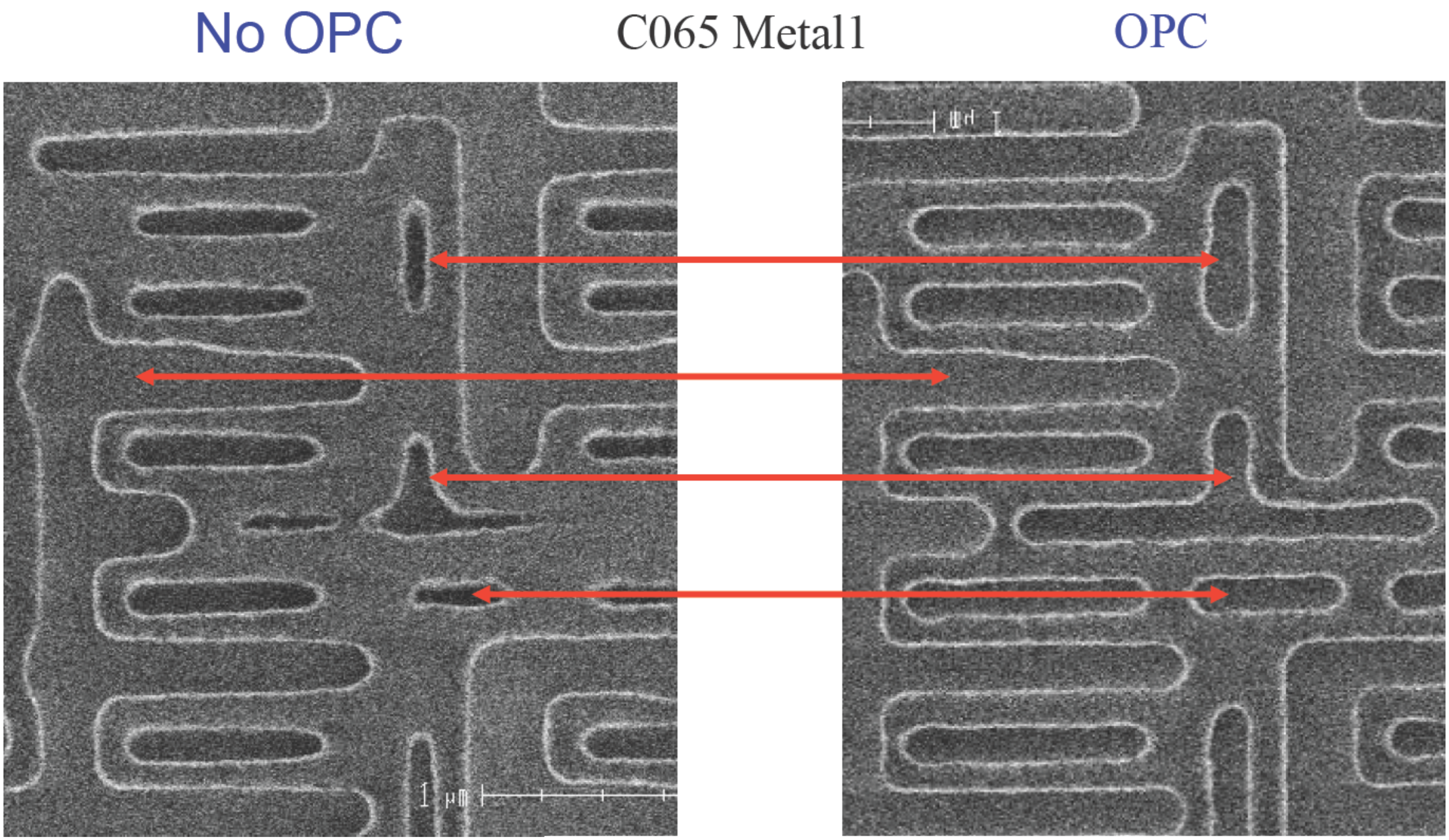

Mask (without optical proximity correction)
Light beam
Mask (with optical proximity correction)
Focusing lens
Wafer
Desired circuit pattern
Exposed circuit pattern

Source: Schellenberg, Spectrum 2003



Source: Schellenberg, SPIE 1999

More OPC examples

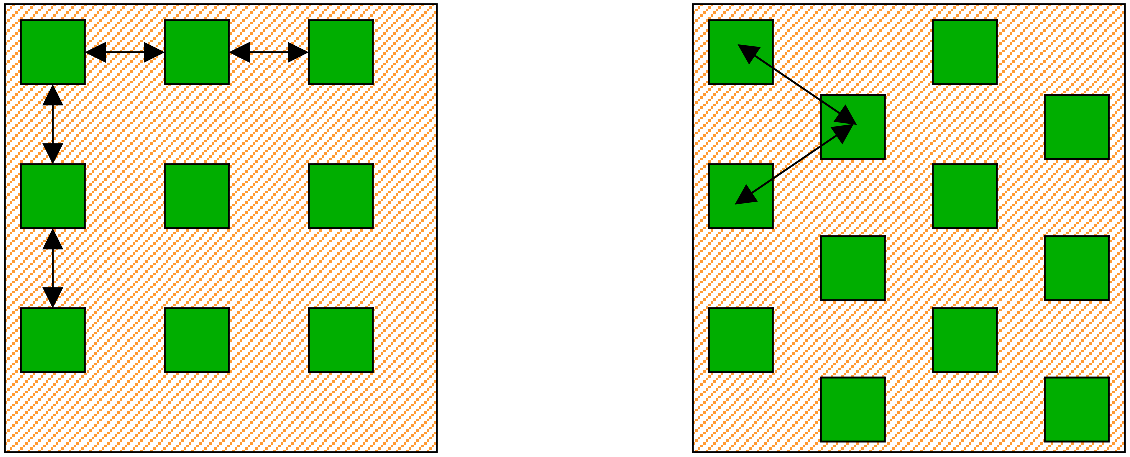


Source: Tesesco, 2006

Via Spacing Rules

- Key point is that focusing light is difficult (even with OPC)
  - Square features end up more like circles

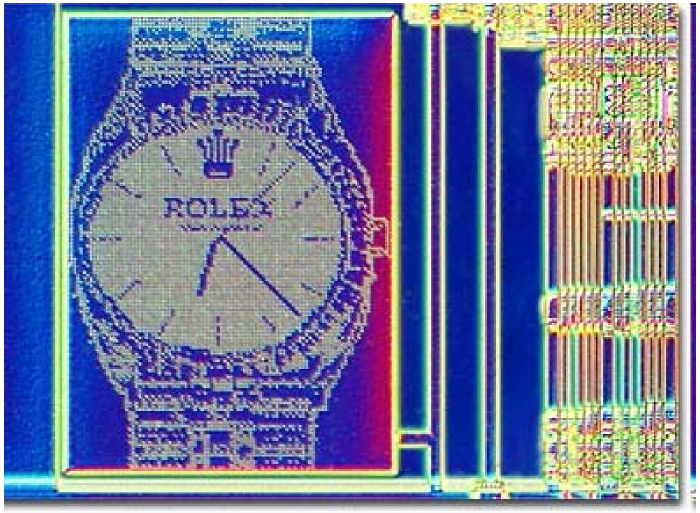
- Layout rules will reflect the lack of hard edges on features
  - Via spacing rules will not be edge-to-edge any more
  - Via spacing rules will instead be center-to-center
  - Most efficient packing will use staggered rows



Other Manufacturability Rules

- Watch out for your logos!
  - Your boss will be *peeved* if your logo breaks manufacturability
  - Keep it far from any critical circuits
  - Maintain metal density in your logos
  - But they are fun… (more logos at the end…)

HP PA7100 CPU (“Rolex” FPU)

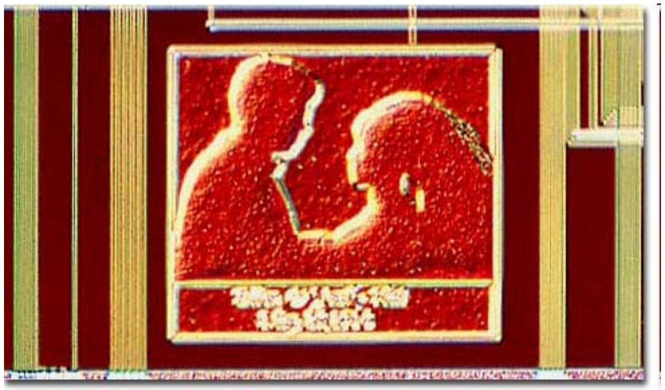


MIPS R12000 (taped out 7/97)

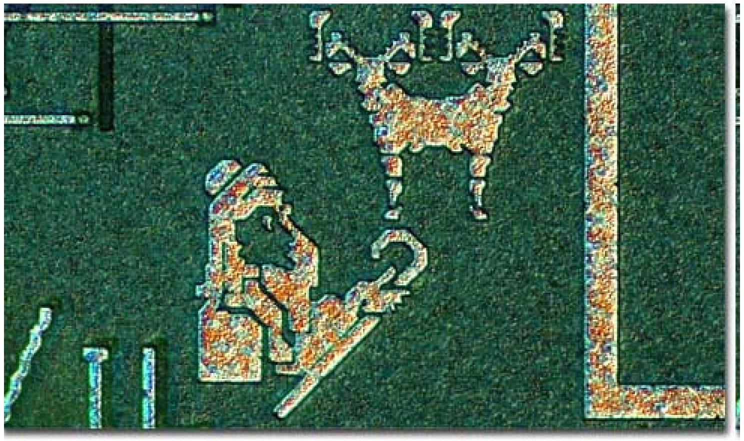
Source: microscope.fsu.edu

More Logos

- Sometimes mask designers can be romantic
  - MIPS R10000, “Ellen & Yeuk-Hai, May 25, 1996”



- Sometimes they can be silly
  - Intel 8207 memory controller
  - Shepherd picture
    - Tending a two-ported “RAM”

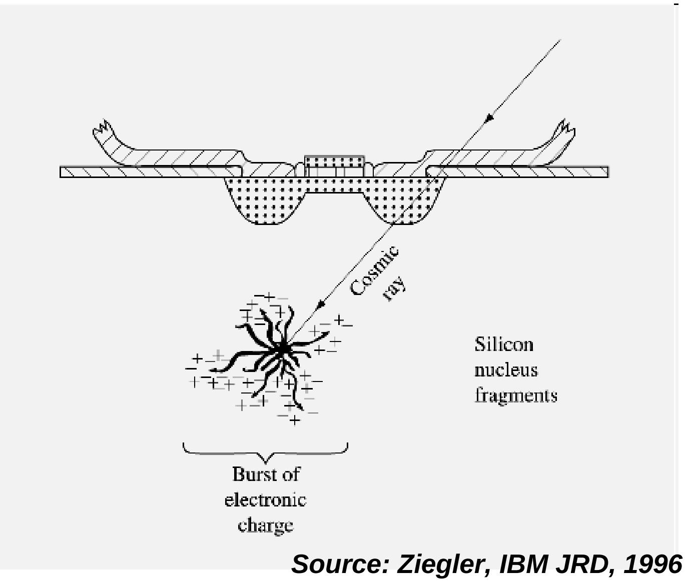


Source: microscope.fsu.edu

Noise From Outside Sources

- Some soft errors are caused by radiation from the outside
  - Cosmic rays
  - Alpha particles

- Radiation strikes the chip and smacks into Silicon lattice
  - Generates a flood of h^+ and e^-
  - Charge flows into diffusions
  - Can upset the state of nodes
  - If charge exceeds Q_{crit}



Source: Ziegler, IBM JRD, 1996

- In general, soft errors include
  - EMI, signal integrity
  - Not just radiation…

Where Does Radiation Come From?

- Alpha particles
  - From contaminants in chip packaging (ceramics, solder, epoxy)
  - Less of a problem with lead-free solder (or need “Roman lead”)
  - Energy ~ 5-10 MeV; easy to shield, except they’re right *there*

- Thermal neutrons
  - Low-energy cosmic neutrons (under 25meV) interact with ^{10}B
  - Fission results in alpha particles
  - BPSG (BoroPhosphoSilicate Glass) has ^{10}B
  - Used to be used in planerization for chips – bad news!

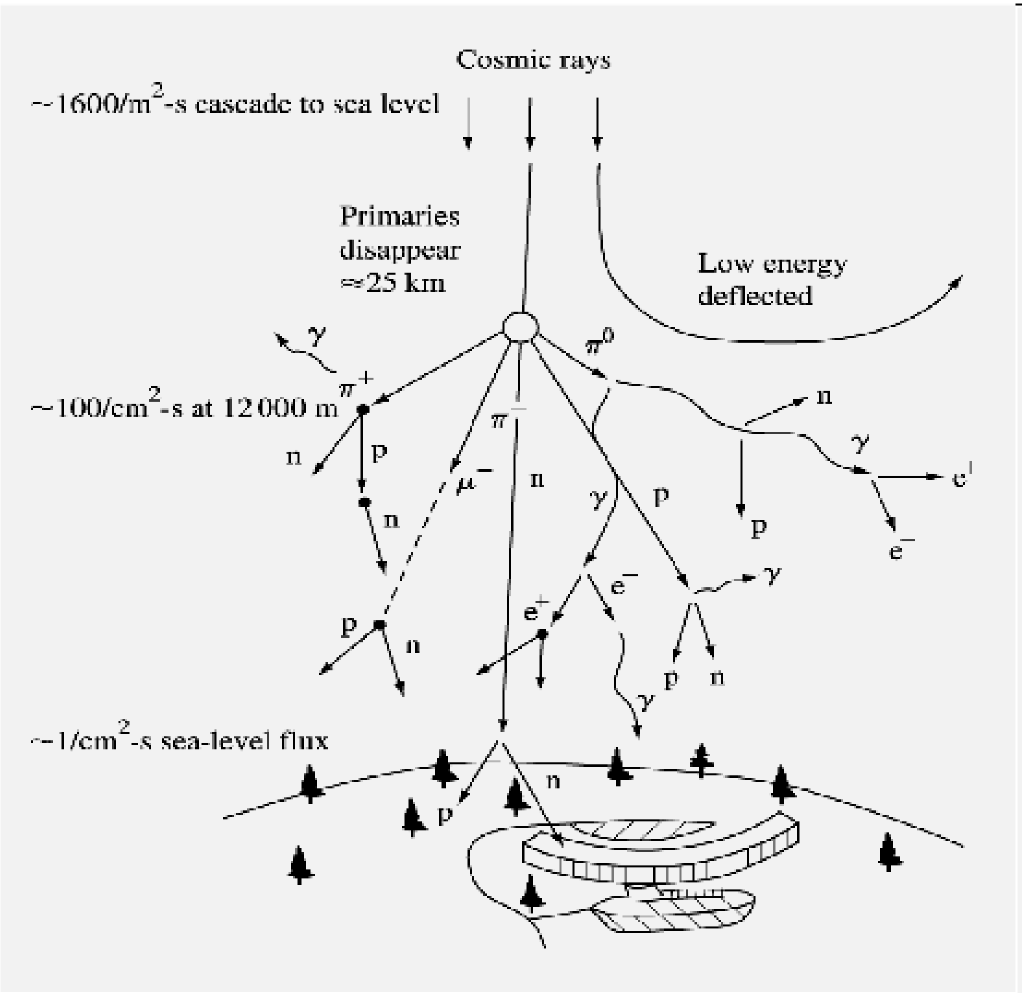
- Cosmic ray neutrons
  - From the Big Bang (really) or supernovas, energy up to GeVs

Source of Cosmic Rays

- Rays generate neutrons
  - Deflection
    - Magnetosphere
  - Collisions
    - Atmosphere nuclei

- Much worse up high
  - Airplanes or satellites

- Also depend on latitude
  - NY, NY is standard



Cosmic rays
~1600/m^2-s cascade to sea level
Primaries disappear ≈25 km
Low energy deflected
γ
π^+
π^0
~100/cm^2-s at 12 000 m
π^-
n
p
μ^-
γ
p
n
e^+
e^-
γ
p
n
γ
e^+
e^-
~1/cm^2-s sea-level flux
n
p

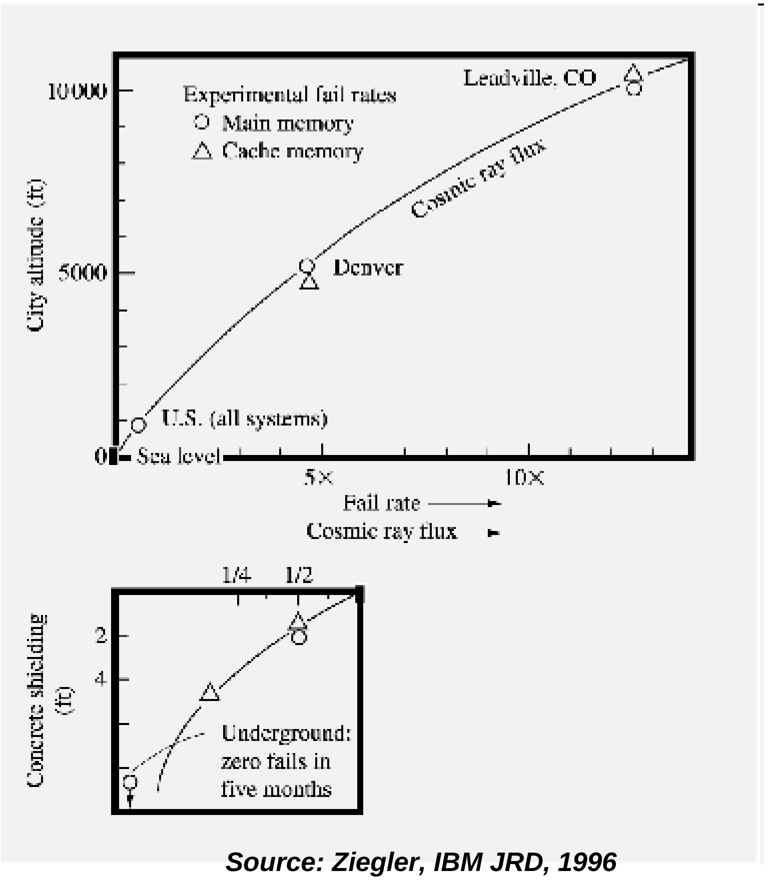
Source: Ziegler, IBM JRD, 1996

Cosmic Ray Flux Sensitivity to Altitude

- Predicted and measured flux and failure rates on testchips

- Flux depends on altitude
  - Increases as 1.3^{altitude}
  - Altitude in 1000s of feet
  - Denver, CO at 5000 ft
    - Expect 3.7x more flux

- Not easy to shield from them
  - 3 ft of concrete: 50% less flux



Source: Ziegler, IBM JRD, 1996

Selected History of Soft Errors

- May & Woods at Intel tracked down first alpha particle problem
  - 1979, saw failure rates in their 2017 chips (16Kb DRAM)
  - Tracked down to a ceramic package factory on Green River, CO
  - Downstream from an old uranium mine, water was radioactive

Source: May&Woods, IEEE TransElectronDevices, 1979

- Late 80's, IBM wanted to measure cosmic ray flux
  - Literature dominated by the UN "Quiet Sun Years" study (1965-68)
  - Canadian Atomic Energy Authority had built a mobile trailer for this
    - Comprehensive measurements, lots of exotic measuring equipment
    - But all the authors of the studies were retired or deceased – trailer lost!
  - "Somewhere on Hawaii" – IBM engineer spent months looking for it
    - Found, abandoned on Haleakala, with a few bullet holes in the side
  - Cleaned, restored, used for lots of IBM's test data (like previous pg)

Source: Ziegler, IBM JRD, 1996

Failure Rates for Soft Errors

- Numbers vary all over the place, depending on who you ask
  - Many memory vendors claim lowest FIT rates for their products
- DRAM FIT rates around 1 FIT/Mb (give or take 10x) today
  - And falling as technology scales
  - DRAM node cap is staying constant(-ish)
- SRAM FIT rates around 1000 FIT/Mb (give or take 10x)
  - And staying constant (falling very slowly) as technology scales
- But both are generally "don't care" events, because of ECC
  - Unless the memory isn't ECC, as in small registers
  - Beware the asynchronous FIFO register: deadlock on soft error!

What is ECC?

- Error-Correcting Codes recode data words in a code space
  - Any valid data word is “far” from any other valid data word
  - Any error(s) in a data word will move it in the code space
  - We can recognize the bad data word – it’s closest to its real value

- ECC can be single-error, double-error, and so on

- Example of single-bit ECC on a 3-bit data word
  - Data word = (b1,b2,b3); check bits = (c1,c2,c3)
  - c1 = b1 XOR b3
  - c2 = b2 XOR b3
  - c3 = b1 XOR b2

	b	c
a	000	000
b	001	110
c	010	011
d	011	101
e	100	101
f	101	011
g	110	110
h	111	000

ECC Example, con’t

- Draw this as 4 Karnaugh maps

Each code (“a” .. “h”) is far from others

“aX” lists the distance=1 codes for “a”; a single error will move “a” to one of the “aX” codes.

Any single error is correctable back to the valid code word

b1b2 = 00

	b3c1			
c2c3	00	01	11	10
00	a	aX	bX	aX
01	aX	eX	dX	
11	cX		bX	fX
10	aX	bX	b	bX

b1b2 = 01

	b3c1			
c2c3	00	01	11	10
00	aX		dX	hX
01	cX	dX	d	dX
11	c	cX	dX	cX
10	cX	gX	bX	

b1b2 = 10

	b3c1			
c2c3	00	01	11	10
00	aX	eX		hX
01	eX	e	eX	fX
11	fX	eX	fX	f
10		gX	bX	fX

b1b2 = 11

	b3c1			
c2c3	00	01	11	10
00	hX	gX	hX	h
01		eX	dX	hX
11	cX	gX		fX
10	gX	g	gX	hX

ECC, con’t

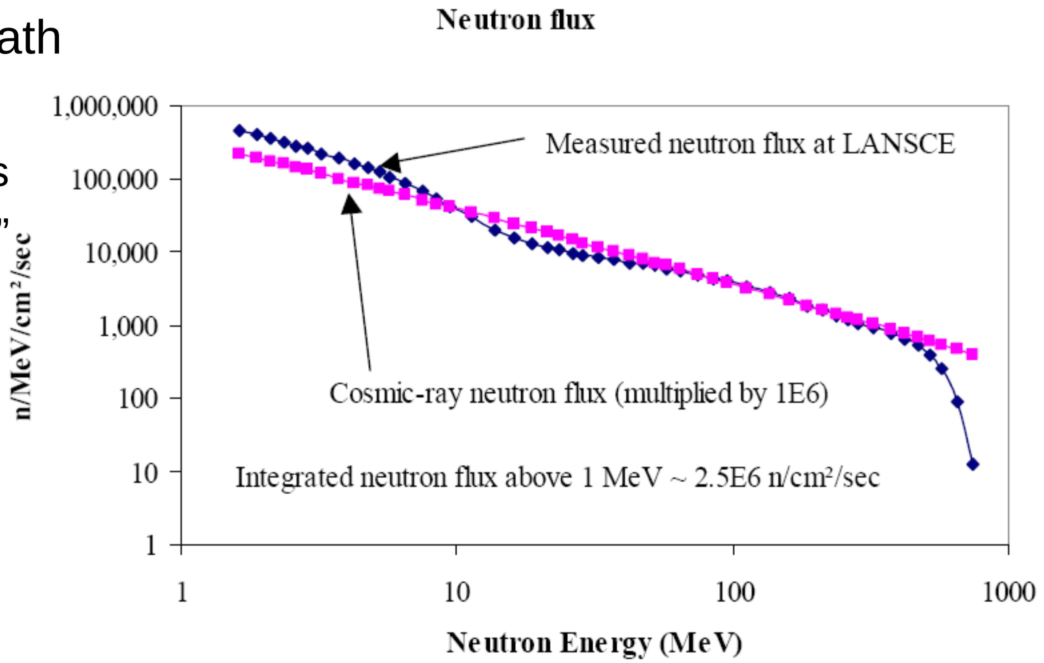
- Key point is that Hamming distance between valid codes = 3
  - Allows single-bit ECC
  - Valid code distance of 4 allows for double error detection
  - Valid code distance of 5 allows for double error correction etc. etc.

- You can do this for more data bits, of course
  - 64 data bits require 7 code bits for single-bit ECC
  - n code bits can check 2^n-(1+n) data bits
  - (32 bits require 6 code bits… more “efficient” to do 64b ECC!)

Measuring Soft Error Rates

- Can’t simply wait for errors to happen
  - Unless you set up a big lab in Leadville, CO – but that’s still slow

- Model cosmic rays using big particle accelerator (LANL)
  - Put system in beam path
  - Can stack them up
    - Multiple experiments
    - Beam isn’t “used up”



Source: iROC technologies

Solutions to Soft Errors

- In order of increasing cost…

- Use ECC in arrays and interleave entries
  - Single strikes can’t affect multiple rows

- Use static circuits instead of dynamic circuits
  - Static circuits can recover from minor charge injection

- “Harden” your latches or circuits
  - Double state, or add “ballast” capacitance

- Multiple CPUs can run in lock-step
  - Check architectural state at interface