

Analog Layout Synthesis: What's Missing?

Rob A. Rutenbar
Bliss Professor and Head

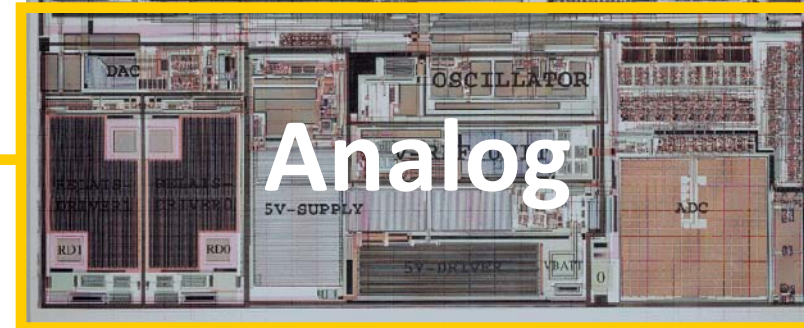
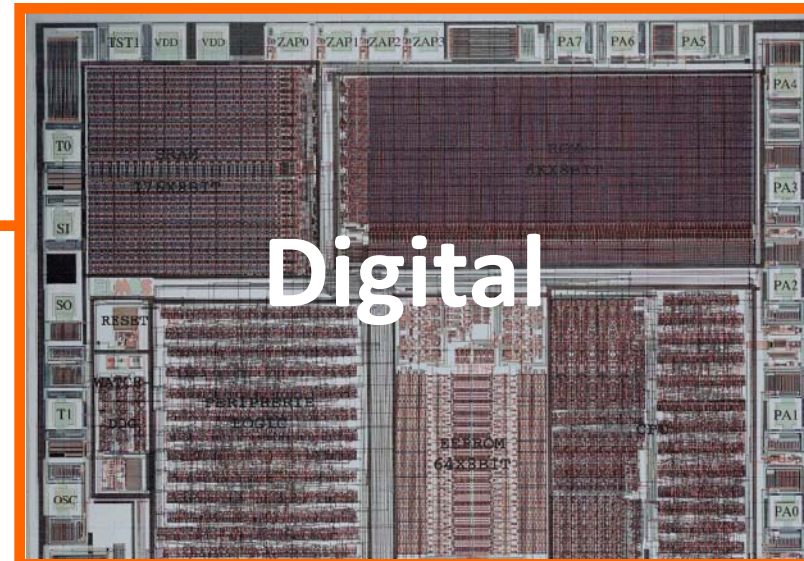
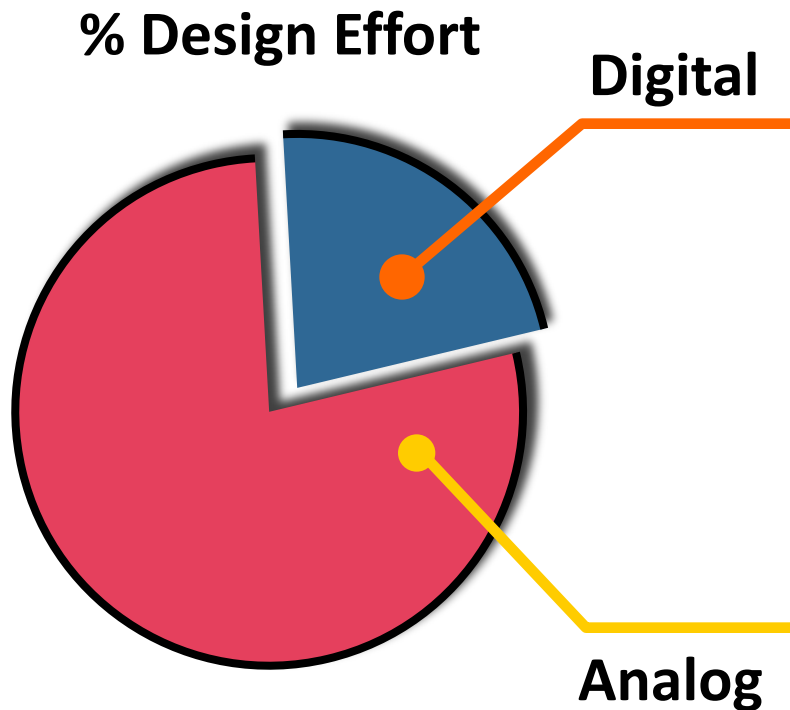


COMPUTER SCIENCE • UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN



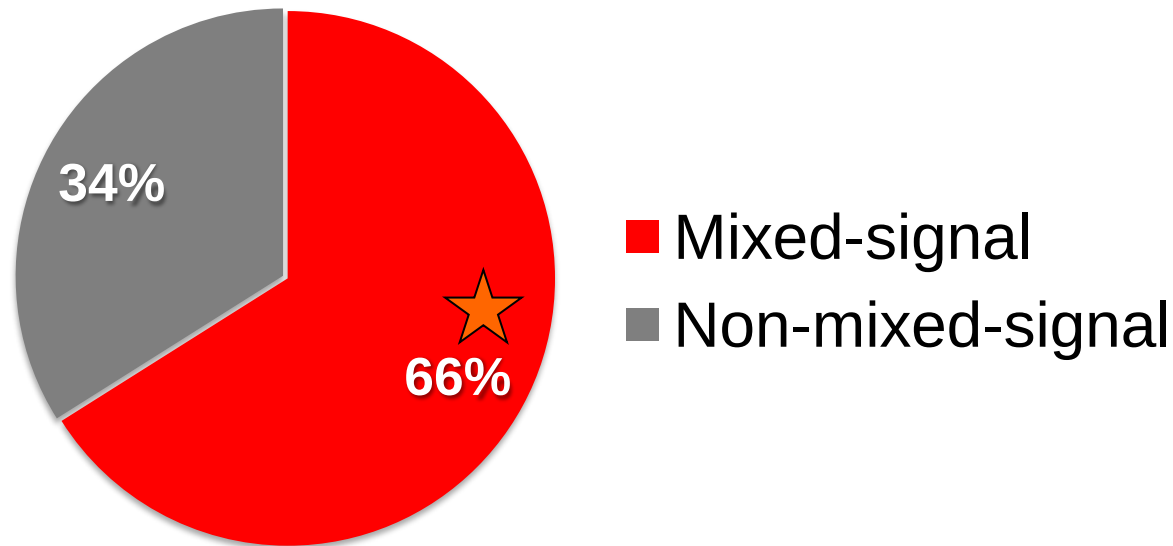
The Mixed-Signal Design Problem

Commercial Mixed Signal ASIC



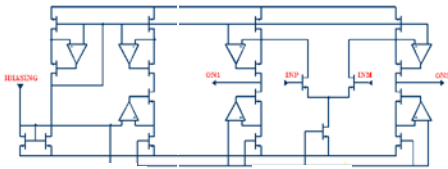
Why this Matters

- Total worldwide market for non-memory ICs in 2008: **\$167B**
 - Mixed-signal portion (*some* analog/RF) was **\$107B** in 2008; ~ **66%**
 - Projected to grow to **70+%** in 2012
 - Growth rate **higher** than overall non-memory IC marketplace

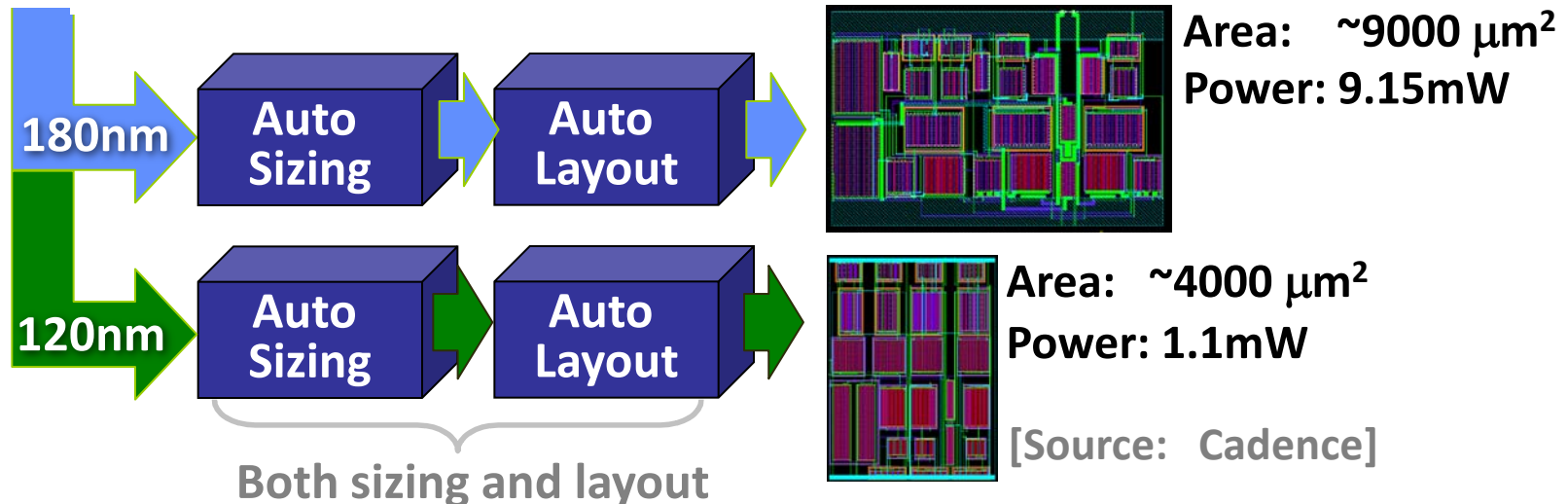


To Start Off On A Positive Note...

- Yes, there are real tools in this space, doing real circuits
 - **Help** size, optimize for perform/yield, **layout**, migrate ...
- But, tools far from perfect, lots of problems unsolved...



- STMicroelectronics result
[Shah, Dugalleix, Lemery DATE02]

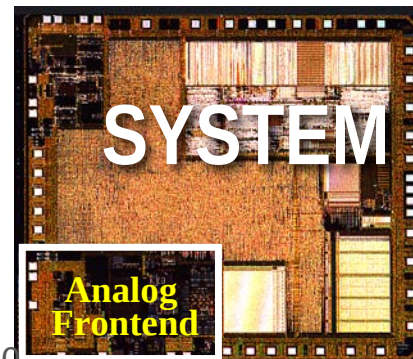
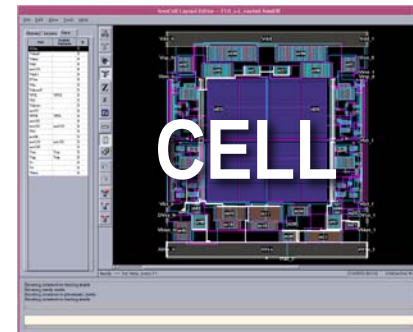
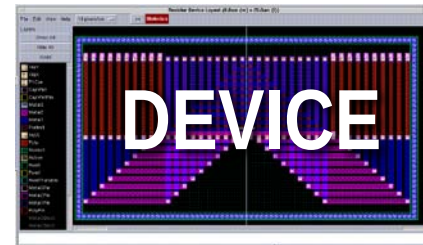


About this Talk...

- Accomplish 3 things (not entirely in sequential order):
- A very brief tour of what “analog layout” *looks like*
- Explain some facts about how/why analog layout *is different*
- Offer some ideas about what are the *open problems*

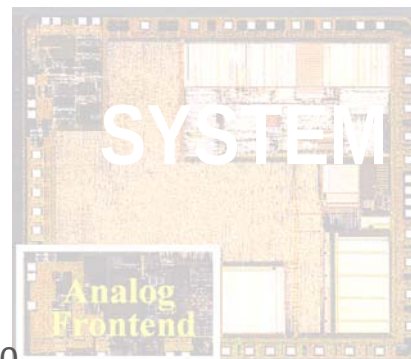
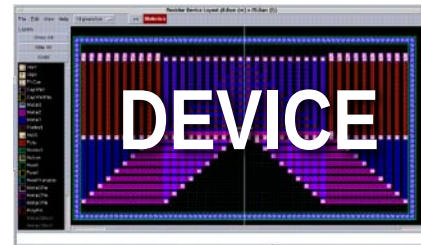
About Analog: Layout Happens at 3 Levels

- Two of these look familiar
- One of them probably looks **odd** to you...

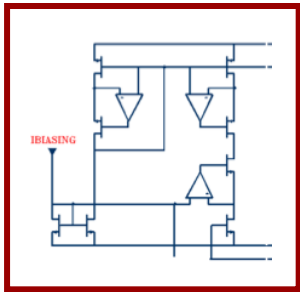


About Analog: Layout Happens at 3 Levels

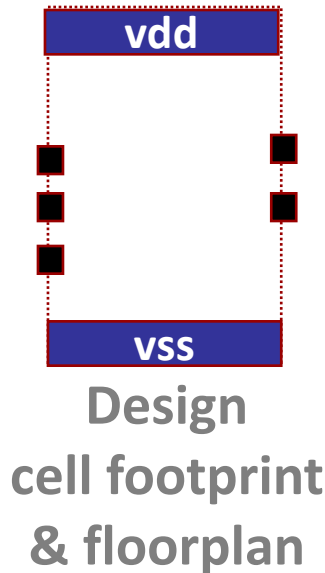
- **Devices** play a role like gate-level cells in digital
- They tend to be the “smallest” units of layout
- But, they are extremely complex, highly diverse
 - Because all analog is about electrical precision
 - Devices are how we harness, manage essential nonlinearities



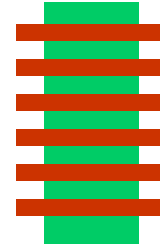
Circuit/Cell Level Layout Flow (Simplified)



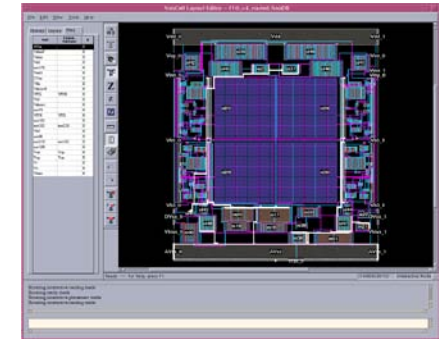
From
sized
schematic



Design
cell footprint
& floorplan



Design
individual
device geometries



Place & route devices,
optimize area,
coupling, etc.

Fundamental
Assumptions

Manual
& *easy*

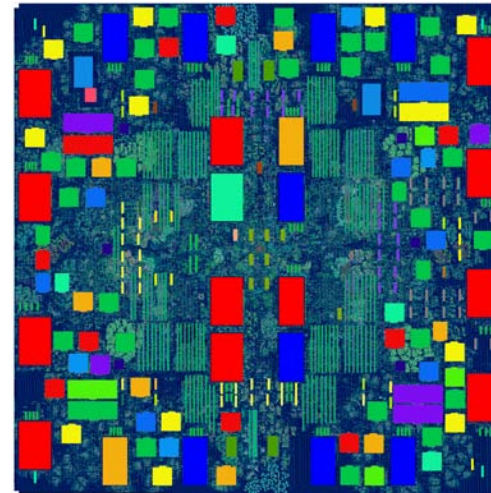
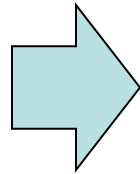
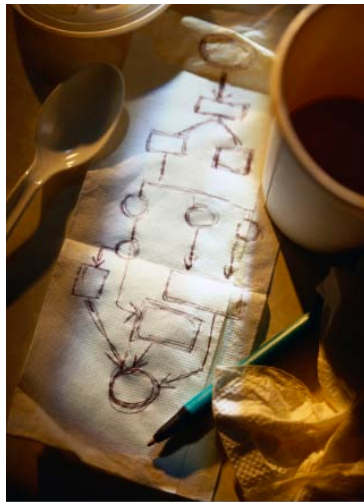
Simple
generators

Essential
algorithms

Orange assumptions are problematic...

Difference #1: Circuit Designer != Layout Designer

- Nobody is surprised that folks who write 1,000,000 lines of Verilog are **NOT THE SAME** folks who do physical design



Courtesy Juergen Koehl, IBM

- So, you should not be surprised **same is true** in analog world...

Analog Circuit Designer != Layout Designer

- (Not universally the case, but often, and it complicates things)

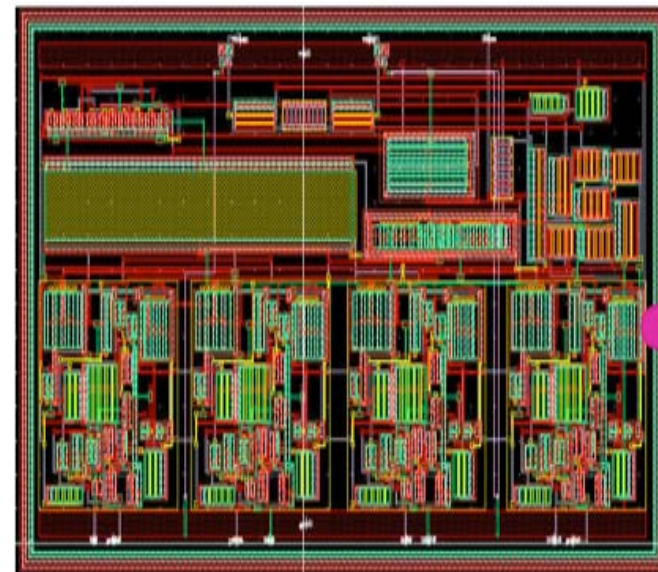
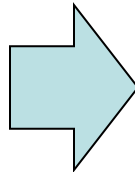


Circuit designer

An engineer

(Maybe grad degree)

Hack transistors, *creatively*



Layout designer

May be a technician, not engineer

(Maybe no degree)

Hack rectangles, *creatively*

Difference #2: Analog Space is Bifurcating

- This is a picture of Texas Instrument's new analog fab
 - They are **very** proud of it
 - It was the **only** big fab built in the US in 2009
 - It was the industry's **very first** 300mm analog fab



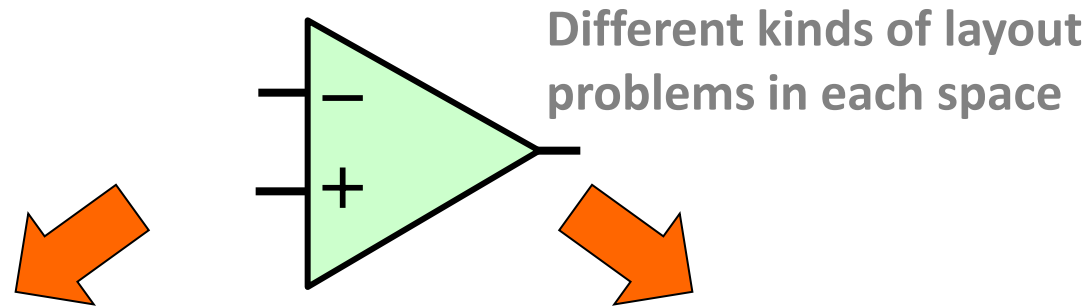
What technology node?

0.25 micron linear BiCMOS

**5 full nodes *behind*
leading-edge digital**

Source: Bill Krenik, Chief Technologist, TI SLL

Bifurcating Analog Space



■ “Fully depreciated” analog

- Need the function, cheap
- Don’t need SOC integration
- Don’t need 10M gates of logic

■ PRO

- Cheap, fewer nm effects

■ CON

- Can’t integrate lots of gates

■ “Fully scaled” analog

- Need function AND integration
- Essential for high-volume parts in cheap digital processes

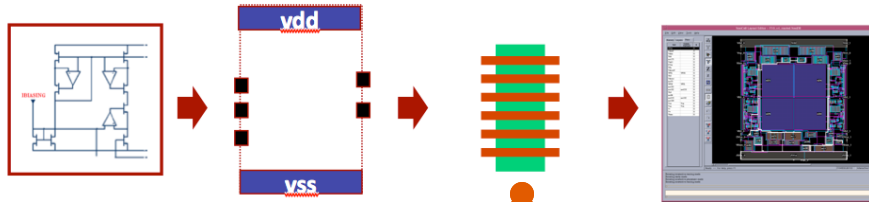
■ PRO

- Easy access to 10M+ gates

■ CON

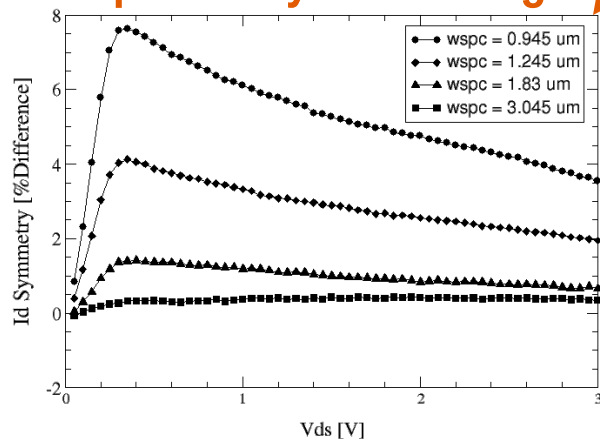
- Nanometer grief is worse here

Difference #3: Nanometer Grief Hurts More

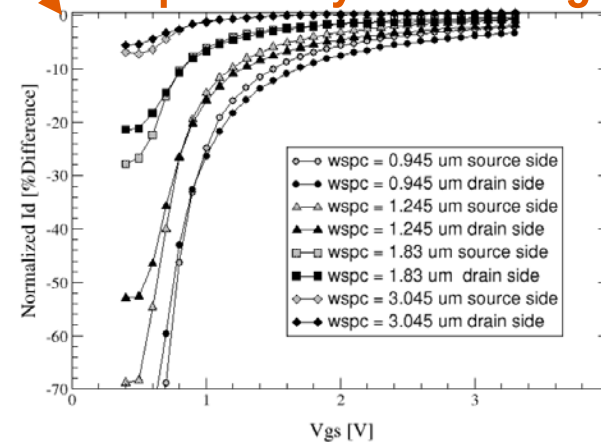


Source: Cao and McAndrew, ICCAD 2007 tutorial, and
P. G. Drennan, M. L. Kniffin, and D.
R. Locascio, CICC 2006

I_D vs V_{DS} Src/Drain asymm
vs proximity to well edge



I_D vs V_{GS}
vs proximity to well edge



- Since analog is always about manipulating electrical quantities in precise ways, the nm effects hurt more
- Interesting result: Lots of focus on device-level automation

Opportunity: Device-Level Automation

- **Shapes complexity**
 - 1 schematic device → **many** physical devices (“fingers”)
 - **N** schematic devices → **single** physical layout structures
 - **Complex** spatial constraints (symmetries, matching, wells)
 - **Routing** problem is integral to the device gen/placement too
- **Physics complexity**
 - As **nm** effects worsen, need optimization to do these **right**

- **Example: Cadence MODGEN**

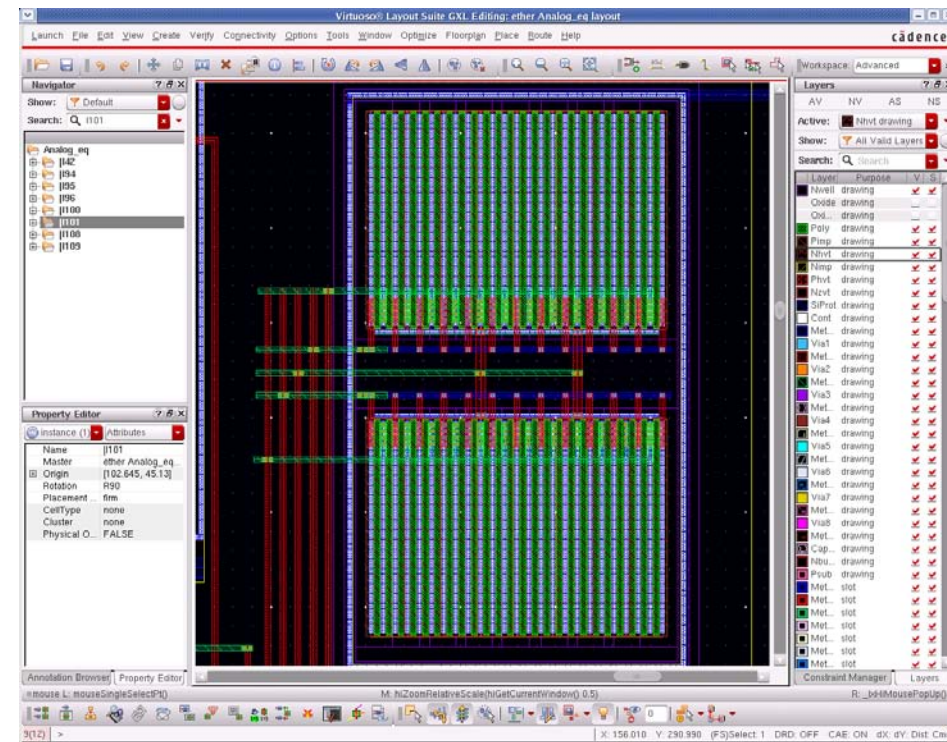
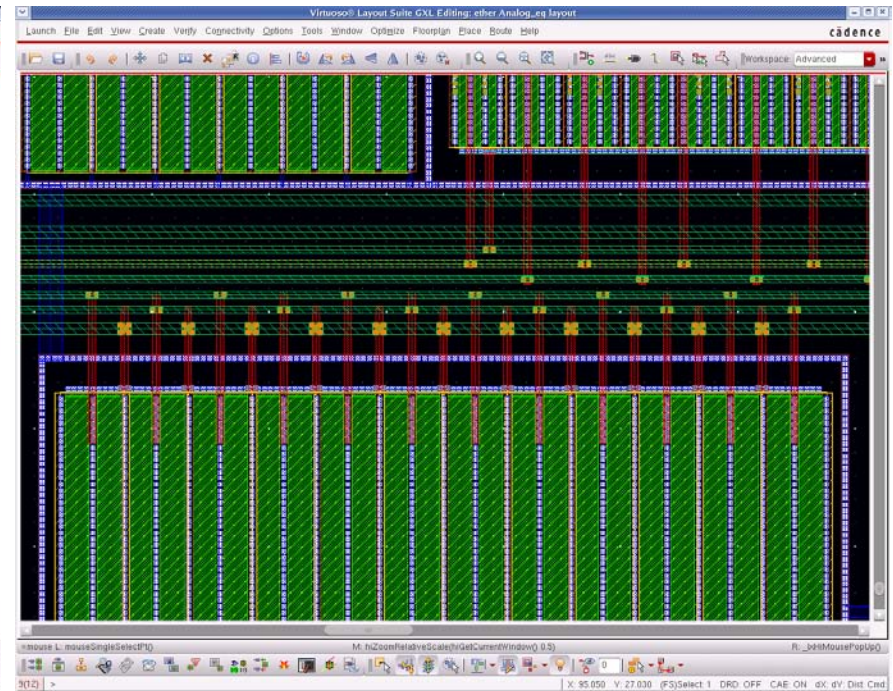
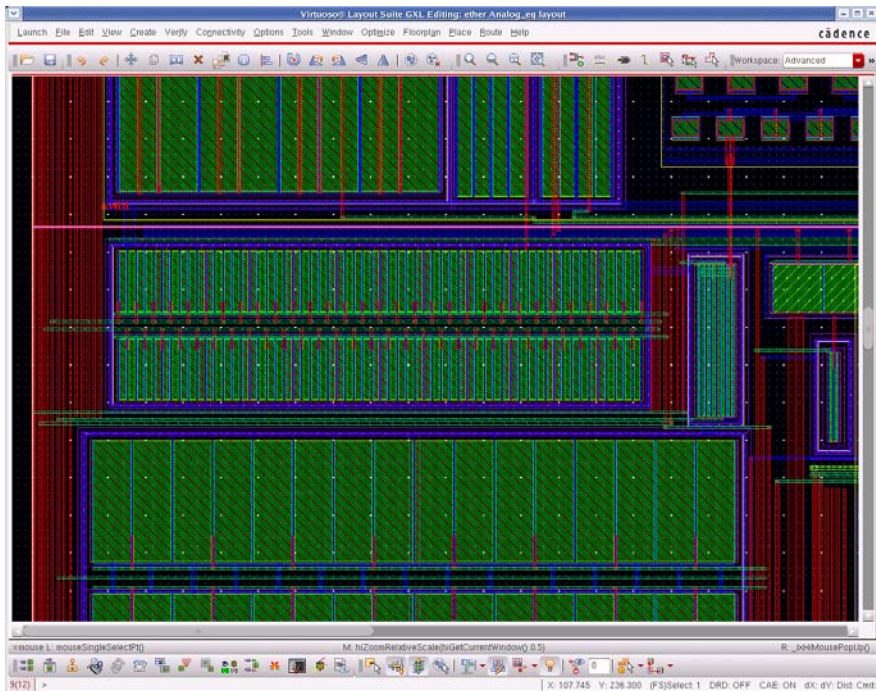


Image provided by ©2010 Cadence Design Systems, Inc.

All rights reserved worldwide

Opportunity: Device-Level Automation

- If you push analog polygons for a living, you *love* these tools
 - Automation doesn't threaten creativity; helps get hard stuff right



Images provided by ©2010 Cadence Design Systems, Inc.

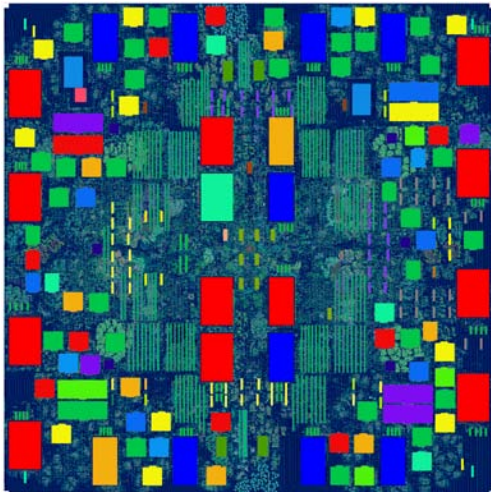
All rights reserved worldwide

© Rob A. Rutenbar 2010

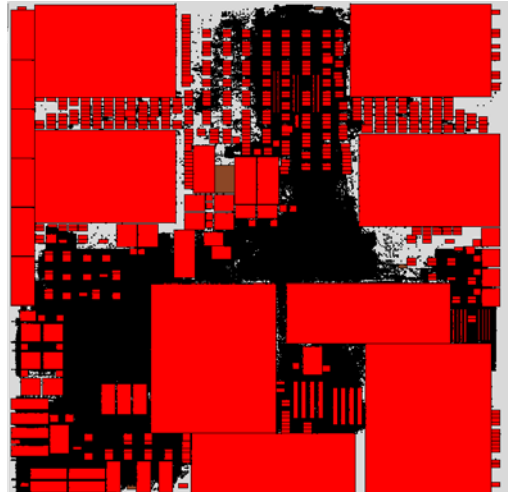
Slide 15

Difference #4: Layout Style Variations

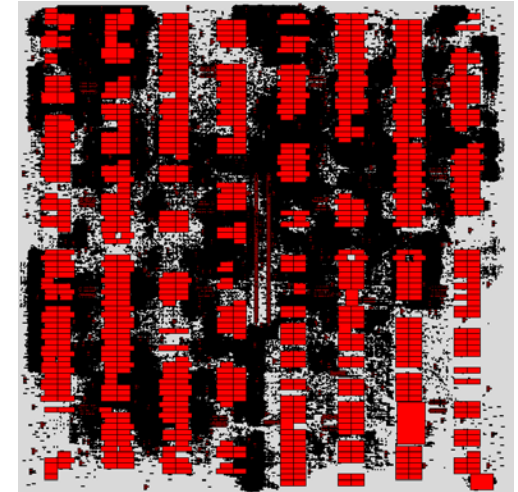
- Only a *slight* simplification to say all digital ASICs look alike
 - Lots of **gates in rows**, in between lot of **macros** (SRAMs, etc)
 - Some diversity in the spread of sizes of macro; some regularity



Courtesy Juergen Koehl, IBM

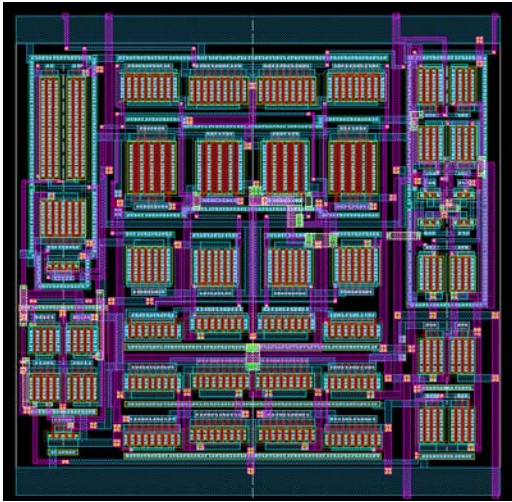


Courtesy Zhong Xiu, CMU

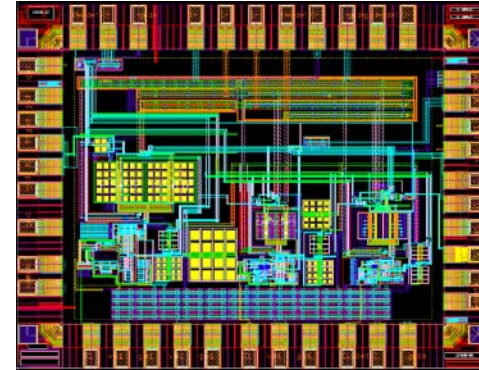


Courtesy Zhong Xiu, CMU

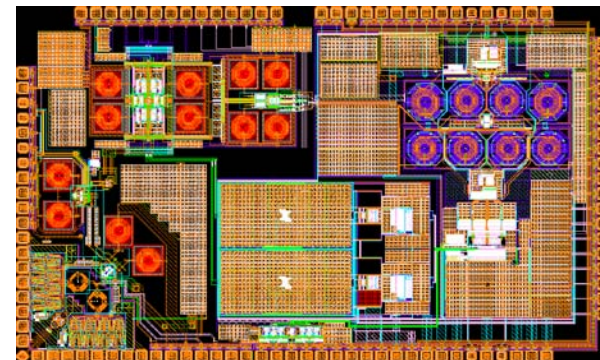
Layout Style Variations in Analog: *Wide...*



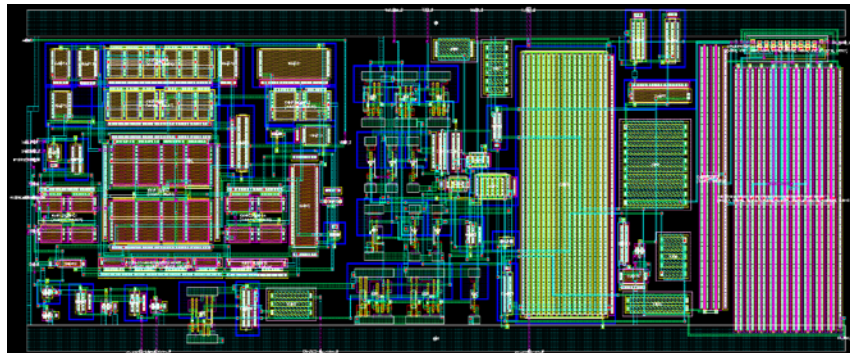
Source: Cadence



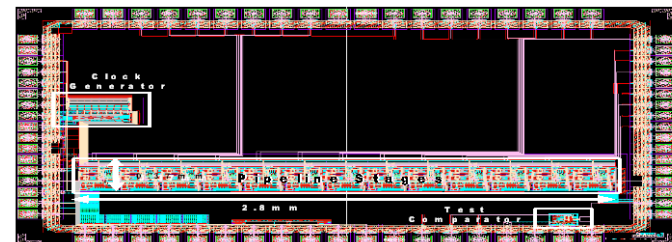
Source: B. Tsang, Y. Chiu, B. Nikolić UCB



Source: P. Gray, UCB



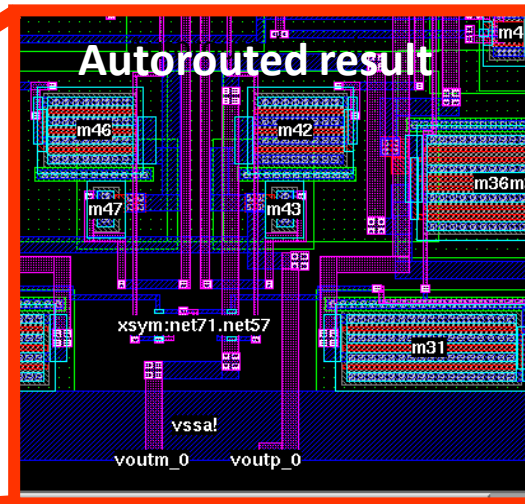
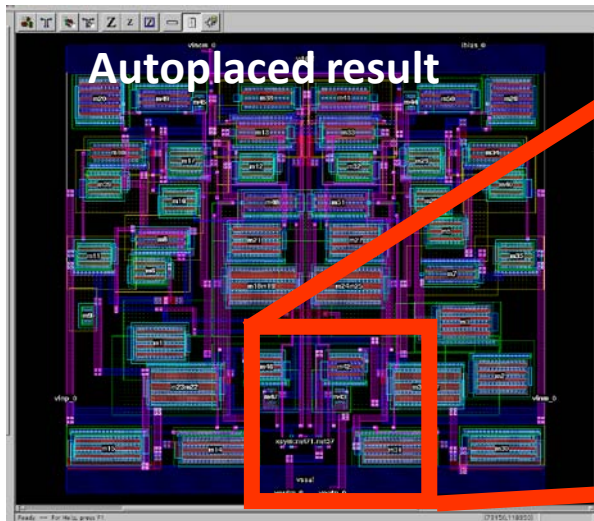
Source: Cadence



Source: H.-S. Lee, C. Sodini, MIT

First-Gen Layout Algorithms....

- Essential formulation was: **Floorplanning + Routing**
 - Devices have large, variable shape (since FETS fold many ways)
 - Pack the shapes, then route the shapes
 - Minimize wirelen+area, while respecting constraints (symmetry)



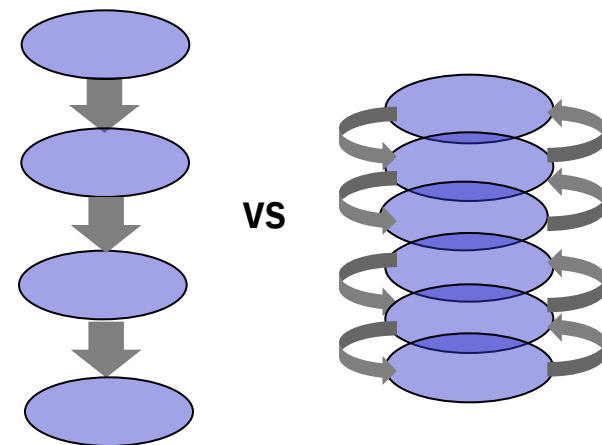
Source: Cadence

- Necessary, but *not sufficient*...

What Did We Not Get (Entirely) Right...?

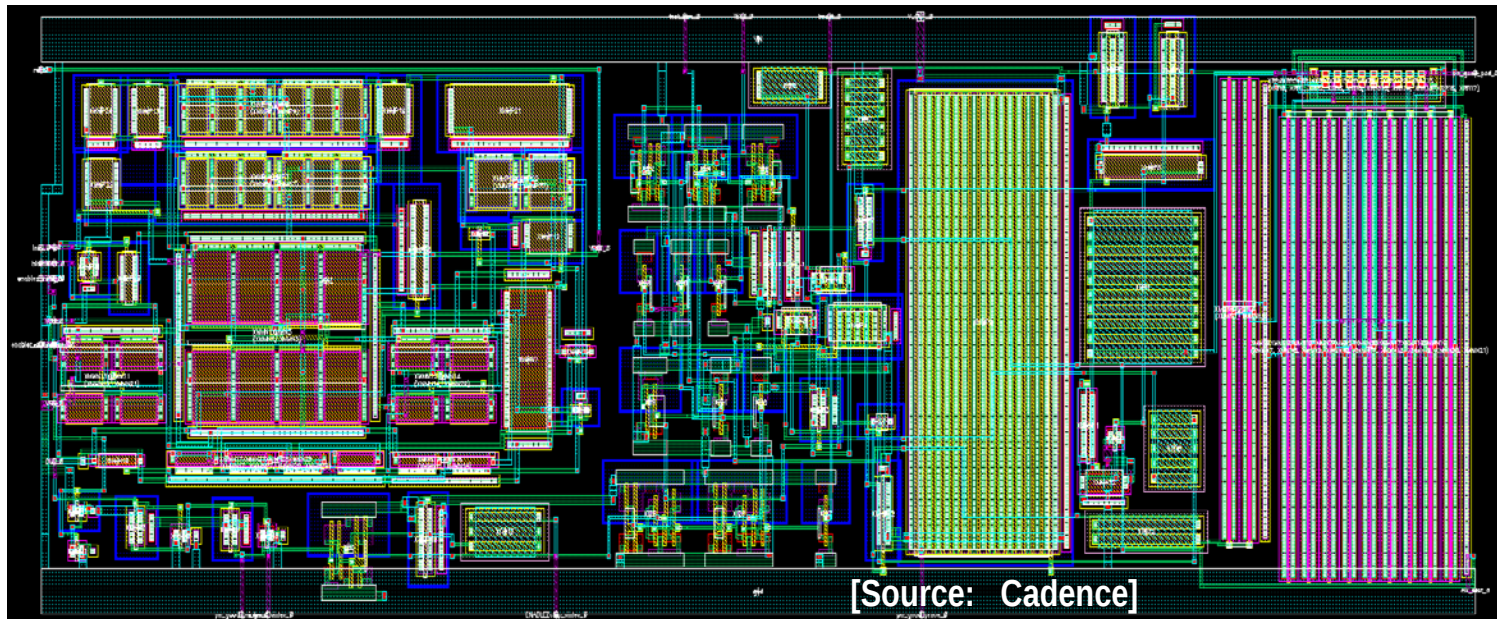
- **Constraint extraction and tradeoff management**
 - Critical stuff in real designs often **never written down**
 - Exists **implicitly** in design group's legacy portfolio and human resources
- **Organic integration: devices, place, power, route**
not done sequentially
 - Design steps less independent, less sequential than digital
 - Usually optimizing across N steps **simultaneously**

*More
comments
in your
code!*



Constraint Extraction/Mgt: Industrial Example

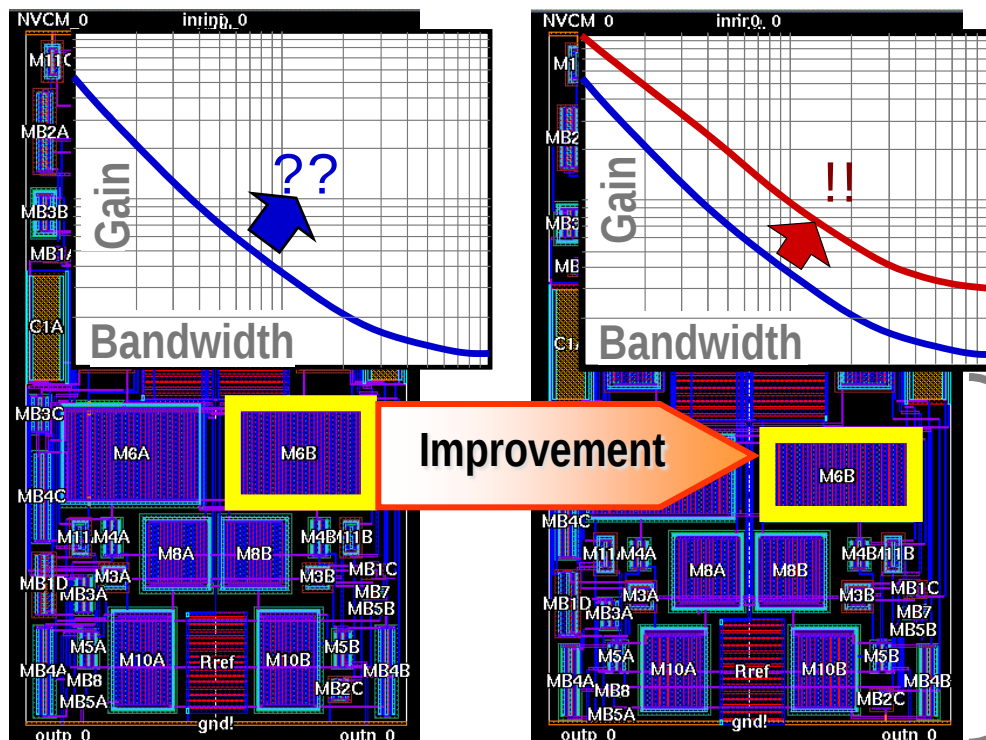
- **Proprietary CMOS comparator block**
 - Lots of critical electrical / geometric constraints – none explicit on schematic, all extracted (arduously) from designer interaction



- **Opportunity: “Low-hassle” constraint harvesting/mining from good designs**

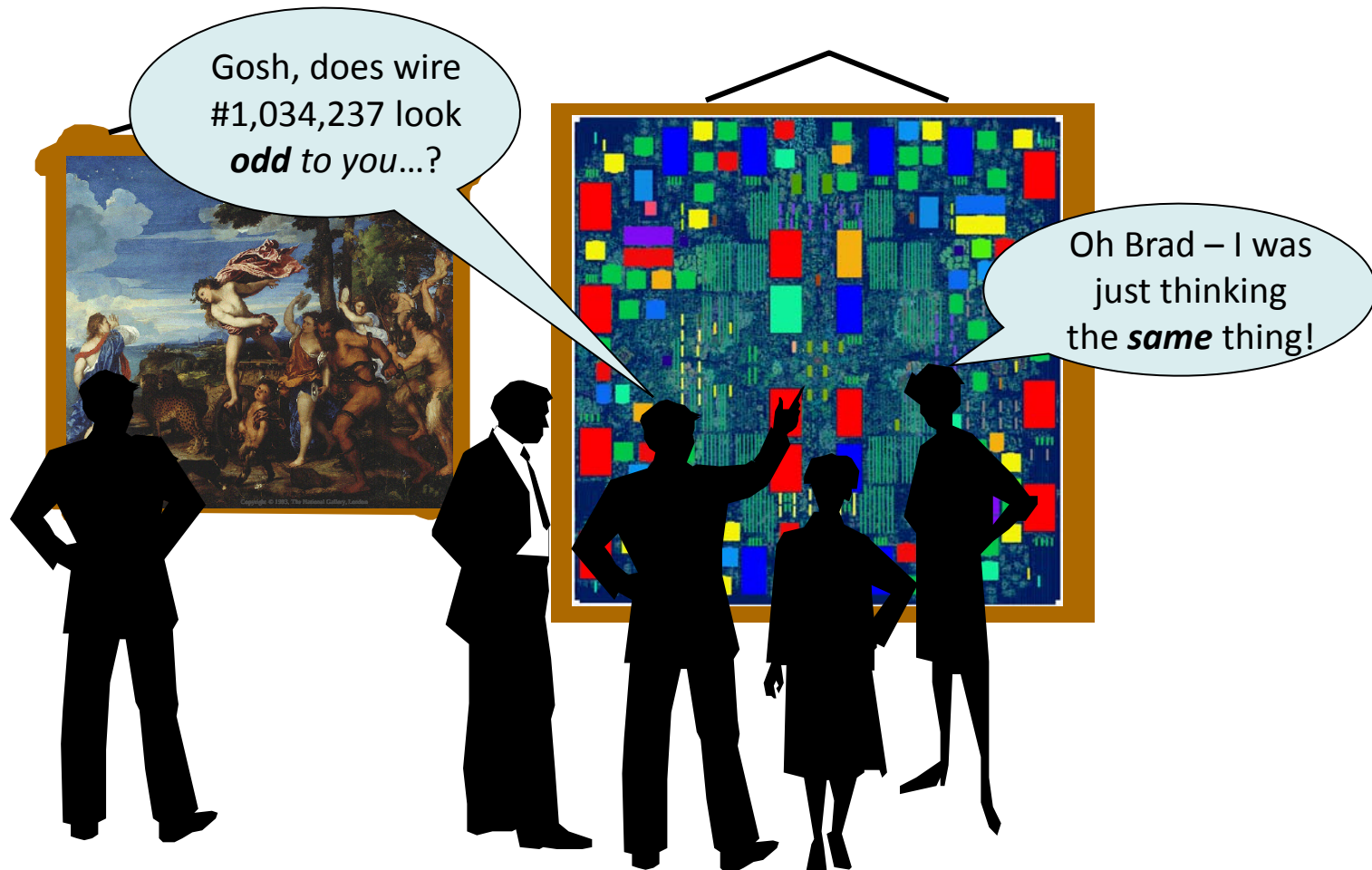
Opportunity: Every Step In Every Flow: *Fast, Incremental, and Deterministic*

- Need very fast “what if...” for **all** electrical/geometric steps
- This is **not** how today’s “deep optimizer” algorithms are done
- Req for **fast+deterministic** is also a huge challenge



Difference #5: Aesthetic Engineering

- This does *not* happen with you lay out 50M digital gates...



Aesthetic Engineering: Two Nuances

- Entire designs often fit on *one* screen
 - People pay attention to things they can grasp in one look
- Aesthetics is often a *surrogate for correctness*
 - Not everything that we'd like to check has a robust script

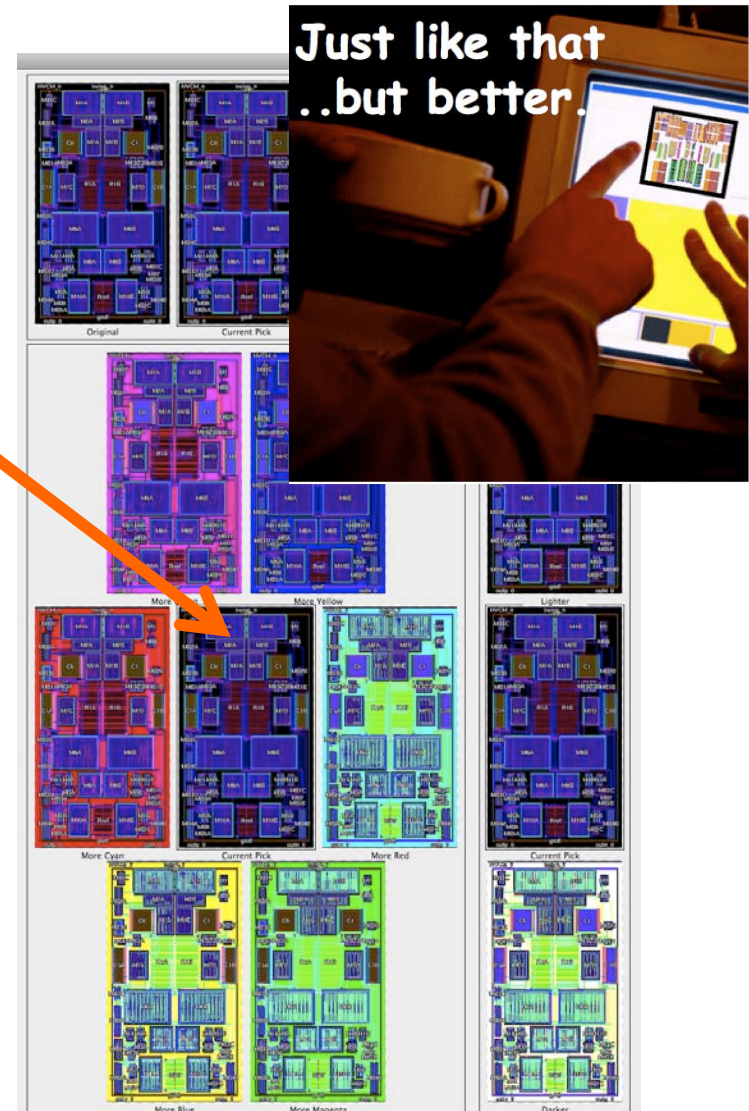
Hey, why is *that*
Fold in *that*
device, right *there*?

...and I really
don't like the
look of *that* via!



Opportunity: Incremental Tools + New Use Models

- Adobe Photoshop offers an interesting vision of this
 - This is “**Image variations**”
 - A palette of *incremental changes* to base image
- Can I do this for **analog layout**?
 - For critical analog metrics?
 - *Shorter wires?*
 - *Straighter signal path?*
 - *Simpler power routing?*
 - *More like schematic?*
 - *More critical signal isolation?*
 - *Farther from well-edge?*
 - *Etc etc?*



Summary

- **My own personal journey**
 - Publishing 20 years ago
 - Commercializing 10 years ago
 - Happy to see real use today...
- **But *lots* left to do**
 - Organic, integrated place, route, power, integrity, etc
 - Everything incremental, everything simultaneous
 - Hassle-free constraint mgt
 - Aesthetic engineering

