

Design for Leading-Edged Mixed-Signal ICs Analog Intellectual Property: Why, When, How

Rob A. Rutenbar

*Carnegie Mellon University
Pittsburgh, PA, USA*

rutenbar@ece.cmu.edu

<http://www.ece.cmu.edu/~rutenbar>

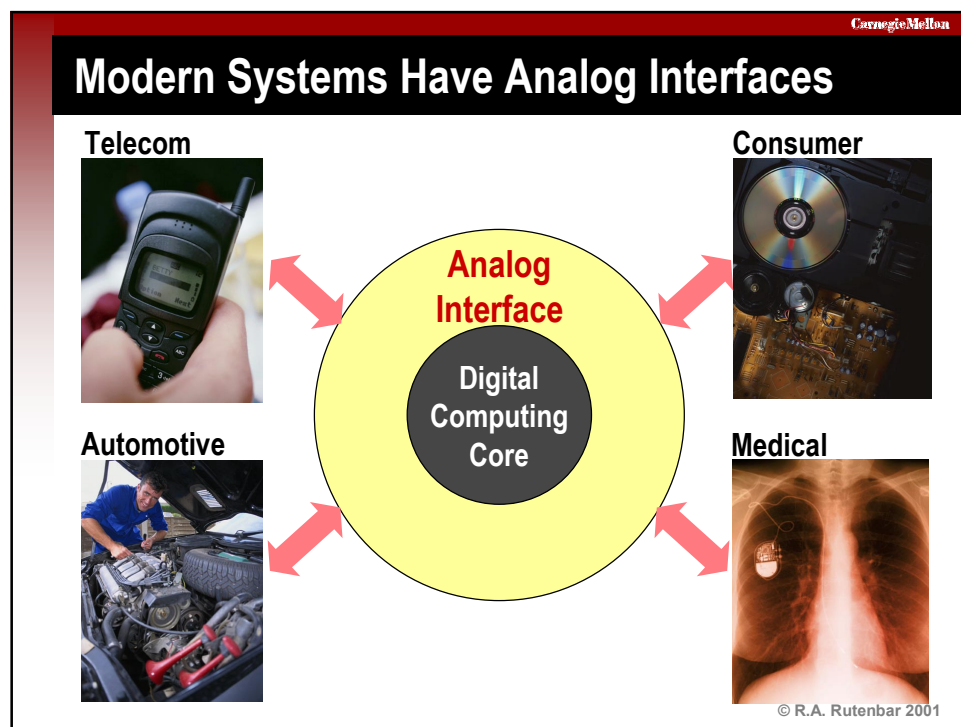


© R.A. Rutenbar 2001

Be Honest: I Say “Analog” ... You Think *This*

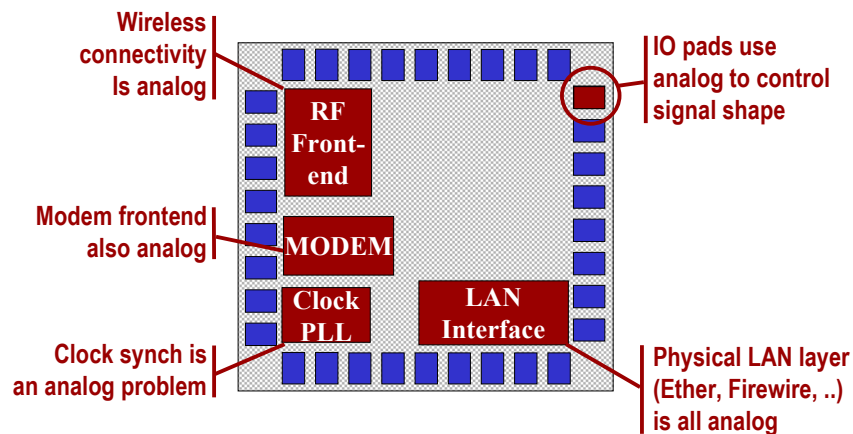


© R.A. Rutenbar 2001



Lots of Digital “Support” Functions Are Analog

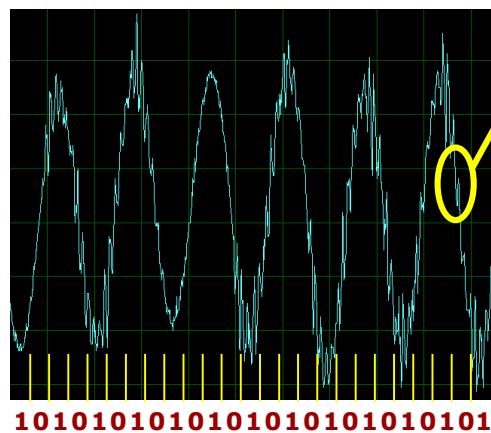
- Some obvious, some not



© R.A. Rutenbar 2001

Lots of “Digital” Signals—Aren’t

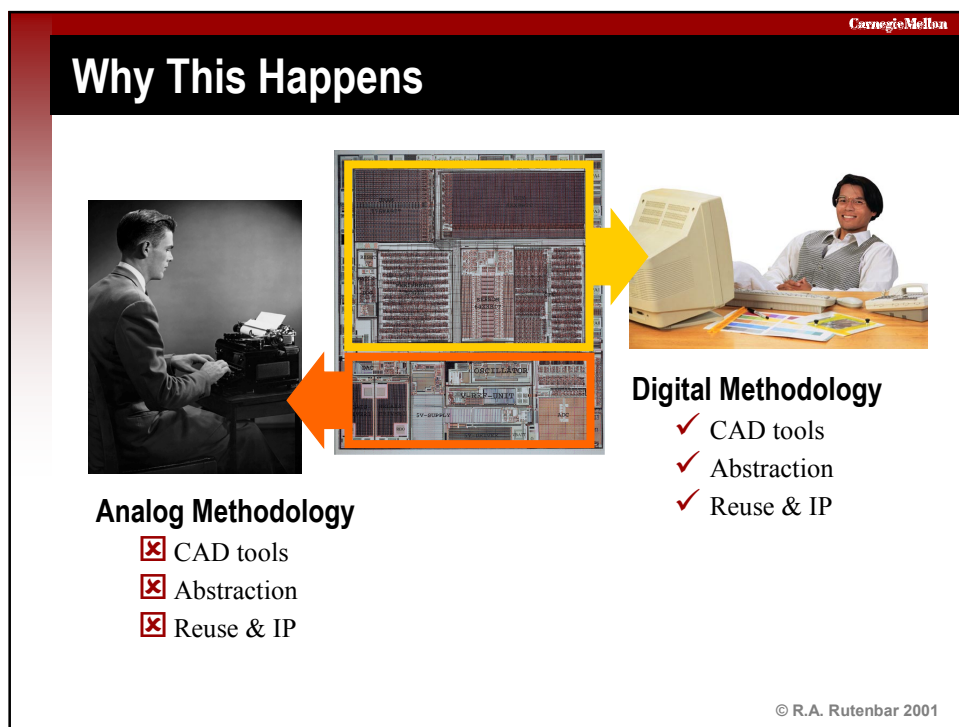
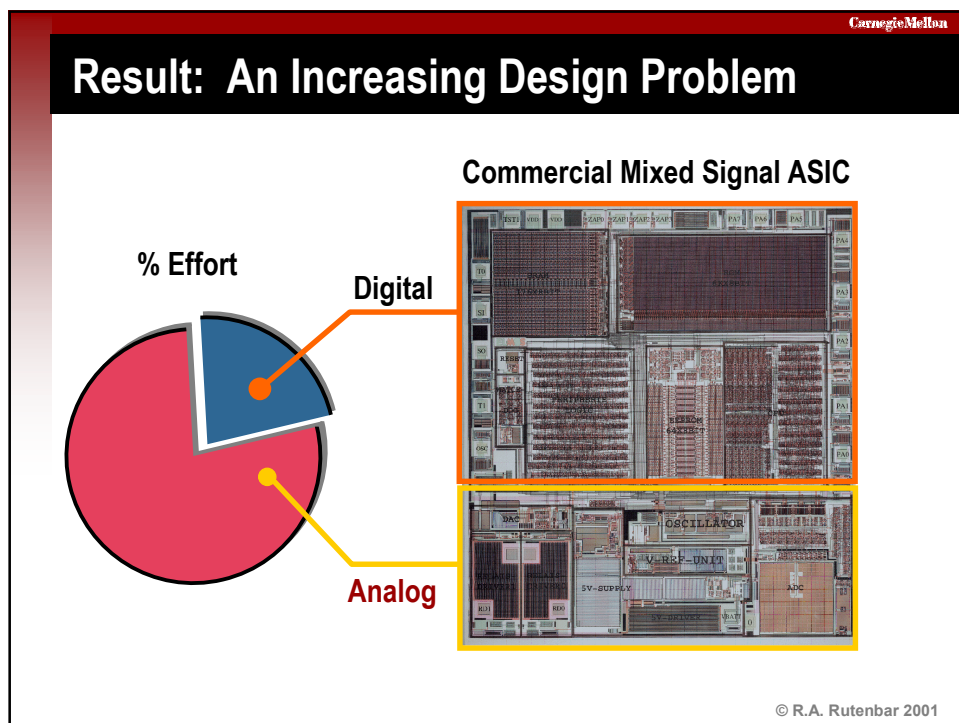
- Ex: What the bits *really* look like read off a magnetic disk

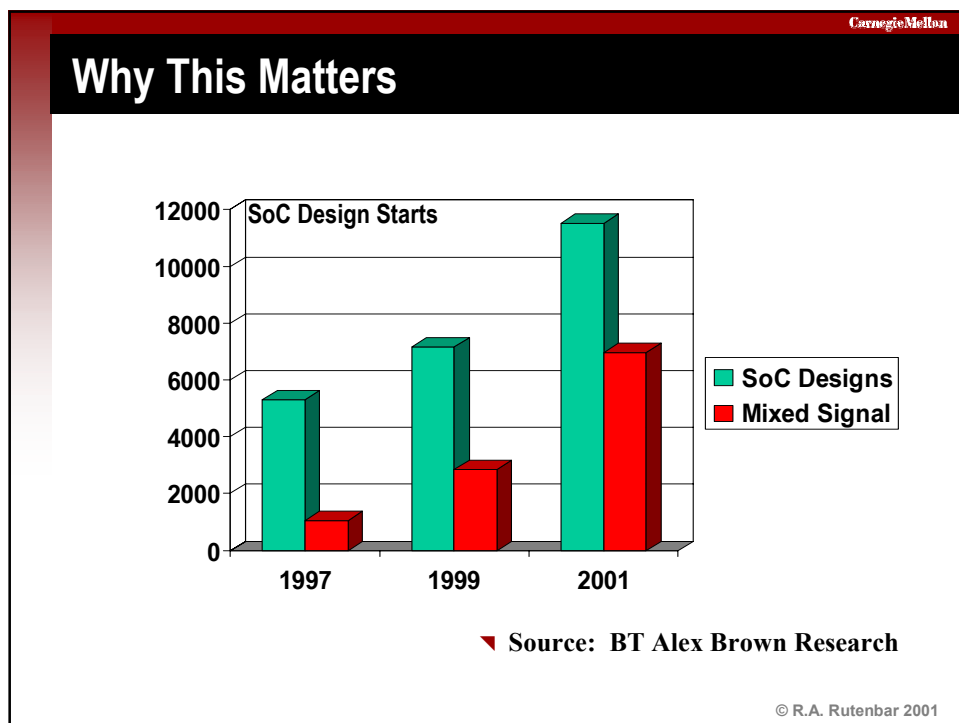


The bits
are the *bumps*
on these sine waves,
by the way...

Courtesy
Jim Bain
CMU

© R.A. Rutenbar 2001





Carnegie Mellon

Outline

- Quick tour of mixed-signal System-on-Chip (SoC) design
- Design problems & strategies for analog building blocks
- Design problems & strategies for mixed-signal chips
- Talk emphasis
 - ▼ We do all this analog design by hand, as painful full custom, today
 - ▼ That has got to change—too many opportunities, too few designers
 - ▼ What are the prospects for “buy it” or “reuse it” for analog?
 - ▼ This is the hot topic in analog today: **analog intellectual property**

© R.A. Rutenbar 2001

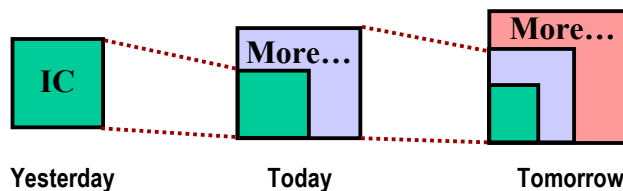
Outline

- Quick tour of mixed-signal System-on-Chip (SoC) design
- Design problems & strategies for analog building blocks
- Design problems & strategies for mixed-signal chips
- Talk emphasis
 - ▼ We do all this analog design by hand, as painful full custom, today
 - ▼ That had got to change—too many opportunities, too few designers
 - ▼ What are the prospects for “buy it” or “reuse it” for analog?
 - ▼ This is the hot topic in analog today: **analog intellectual property**

© R.A. Rutenbar 2001

CMOS Scaling: Different Impact on Analog

- Central fact of life for digital: ICs get smaller, denser, faster



- Scaling matters for analog too; but it's *different*

© R.A. Rutenbar 2001

CarnegieMellon

Analog: The Eggshell Model

■ Analog circuits don't get a lot *bigger* with scaling

▼ Analogy credited to Paul Gray of Berkeley

▼ Scaling provides more *opportunities* for analog interfaces

▼ 10K-20K analog devices/chip is common

Digital Core

Shell is the *analog* here: thin layer of interface to continuous real world

Scaling

Digital Core

More egg; not much more shell

© R.A. Rutenbar 2001

CarnegieMellon

What "More Mixed-Signal SoCs" Means

■ Larger fraction of SoCs need some analog interfaces

Yesterday

Today

Tomorrow

A

A

A

A

A

A

A

A

A

A

A

A

A

A

A

A

A

A

A

A

A

© R.A. Rutenbar 2001

©R.A. Rutenbar, 2001

7

CarnegieMellon

Example: Automotive Mixed-Signal ASIC

A die photograph of an automotive mixed-signal ASIC. The top half of the die is enclosed in a yellow rectangular border. Within this border, two large rectangular blocks are labeled 'Mem' (memory), and three smaller blocks are labeled 'Logic', 'Mem', and 'CPU'. To the right of the yellow border, a grey rectangular box contains the word 'digital'. Below the yellow border, the remaining die area contains various analog and support blocks, including a DAC, OSCILLATOR, V-REF-UNIT, 5V-SUPPLY, and ADC, which are not highlighted.

© R.A. Rutenbar 2001

CarnegieMellon

Example: Automotive Mixed-Signal ASIC

A die photograph of an automotive mixed-signal ASIC, identical to the one above. The bottom half of the die is enclosed in a red rectangular border. Within this border, three blocks are labeled 'DAC', 'Driver', and 'Support', and another three blocks are labeled 'OSCILLATOR', 'V-REF-UNIT', and 'ADC'. To the right of the red border, a grey rectangular box contains the word 'Analog'. Above the red border, the digital blocks (Mem, Logic, Mem, CPU) are visible but not highlighted.

© R.A. Rutenbar 2001

CarnegieMellon

Example: Alcatel ISDN Chip

The image shows a top-down view of a square integrated circuit (chip) with a complex pattern of gold-colored circuitry. Five specific regions are highlighted with white rectangular boxes and labeled with grey text boxes. The 'DSP' (Digital Signal Processor) is in the top-left corner. The 'CPU Core' is in the top-right corner. The 'Logic' is in the center. The 'Analog Frontend' is in the bottom-left corner. The 'Memory' is in the bottom-right corner.

Courtesy Frank Op't Eynde, Alcatel

© R.A. Rutenbar 2001

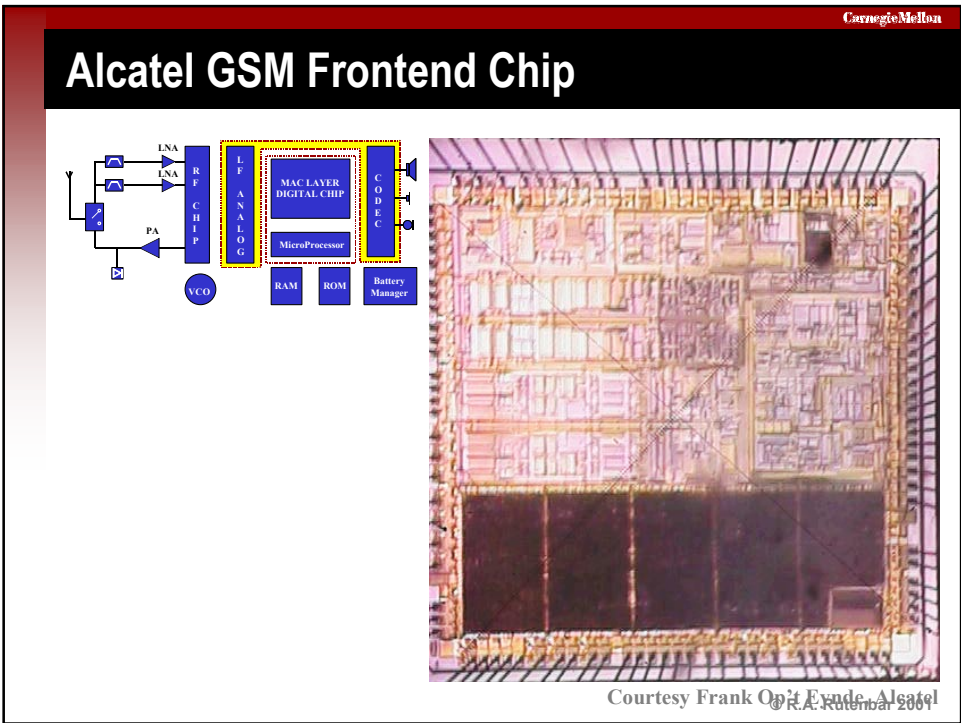
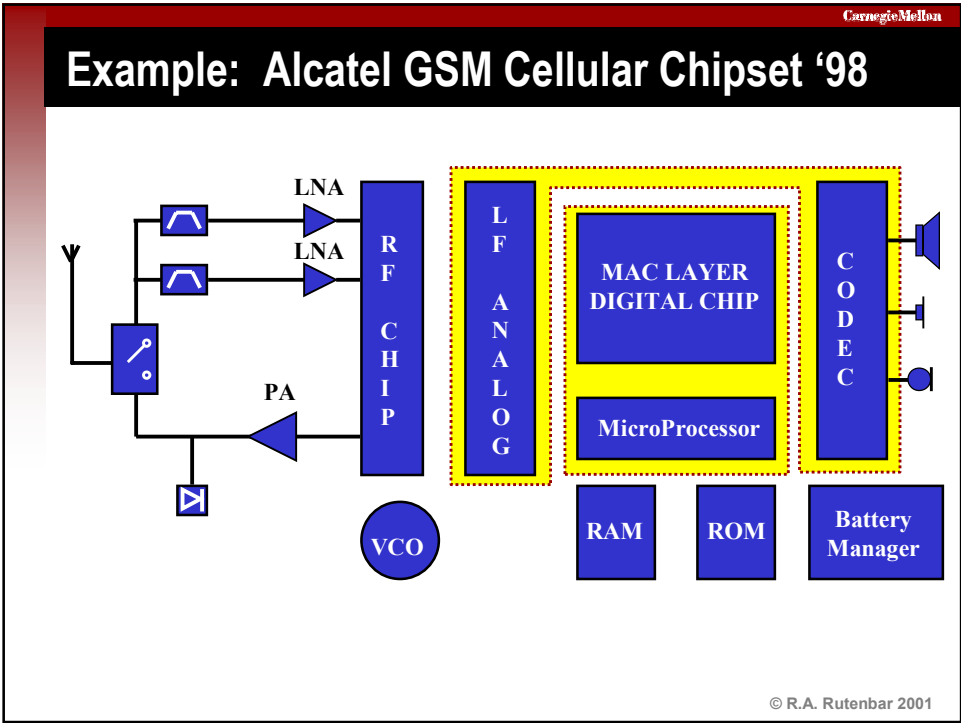
CarnegieMellon

Example: Alcatel GSM Cellular Chipset '96

The diagram is a block diagram of a GSM cellular chipset. It starts with an antenna symbol on the left connected to a switch. The switch routes signals to either an LNA (Low Noise Amplifier) or a PA (Power Amplifier). The LNA output goes to an RF CHIP, which then connects to an LF ANALOG block. The PA output goes to a speaker symbol. Below the RF CHIP is a VCO (Voltage-Controlled Oscillator). The LF ANALOG block connects to a MAC LAYER DIGITAL CHIP. Below this is a MicroProcessor. Below the MicroProcessor are RAM and ROM blocks. To the right of the MAC LAYER DIGITAL CHIP is a CODEC block, which connects to a speaker and a microphone symbol. Below the CODEC is a Battery Manager block.

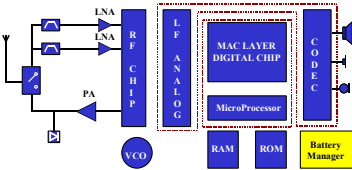
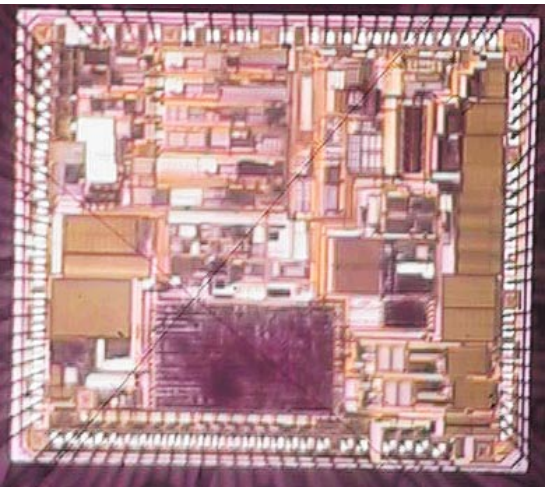
Courtesy Frank Op't Eynde, Alcatel

© R.A. Rutenbar 2001



Carnegie Mellon

Alcatel GSM Power Manager Chip

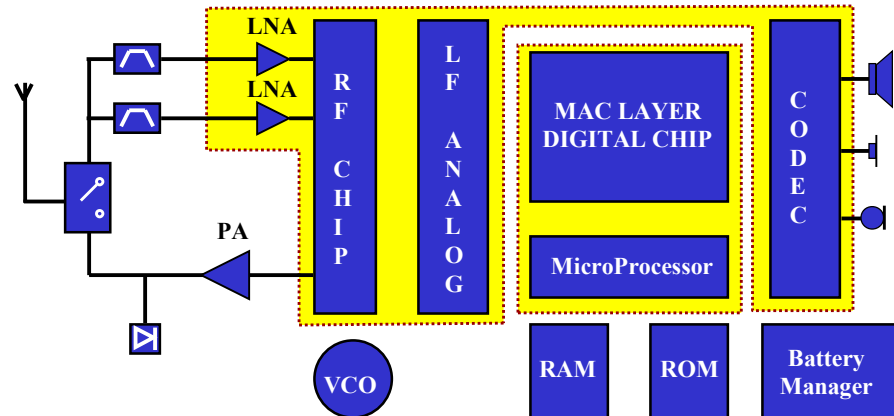



Courtesy Frank Op't Eynde, Alcatel

© R.A. Rutenbar 2001

Carnegie Mellon

Example: Alcatel GSM Telecom Chipset '00



■ Natural result of scaling is analog *integration*: fewer chips

© R.A. Rutenbar 2001

Outline

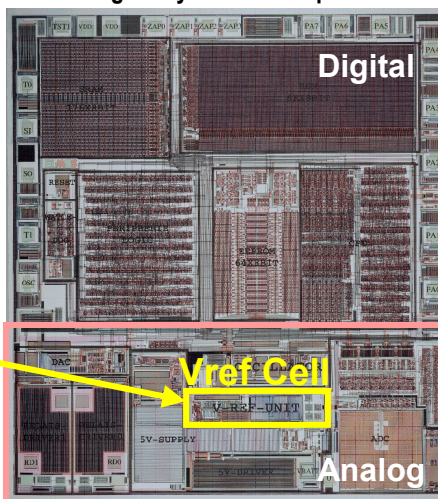
- Quick tour of mixed-signal System-on-Chip (SoC) design
- Design problems & strategies for analog building blocks
- Design problems & strategies for mixed-signal chips
- Talk “spin”
 - ▼ We do all this analog design by hand, as painful full custom, today
 - ▼ That had got to change—too many opportunities, too few designers
 - ▼ What are the prospects for “buy it” or “reuse it” for analog?
 - ▼ This is the hot topic in analog today: **analog intellectual property**

© R.A. Rutenbar 2001

Carnegie Mellon

Example of a Basic Building Block (or Cell)

Mixed-Signal System-on-Chip



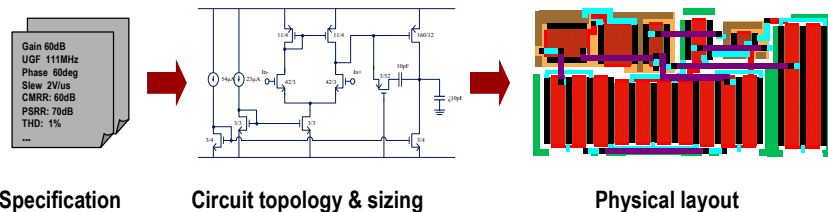
Example:
one analog cell on
analog-side of a
mixed-signal ASIC

© R.A. Rutenbar 2001

Just What Is An “Analog Building Block?”

- Typical analog cell

- ▶ ~5-100 devices (if bigger, usually use some hierarchy)
- ▶ Active devices (FET, BJT, etc) and passives (R, L, C)
- ▶ Often requires precision devices/passives for performance
- ▶ Often requires sensitive device placement, wiring

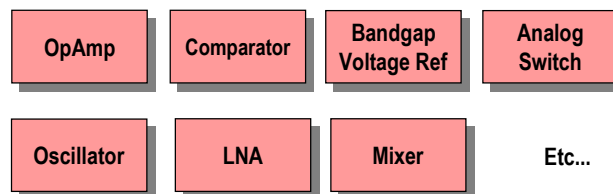


Need all 3 of these to have a “complete” cell

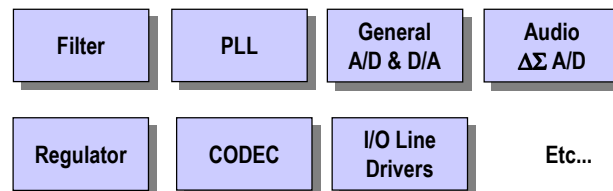
© R.A. Rutenbar 2001

Analog Cells: Common Examples

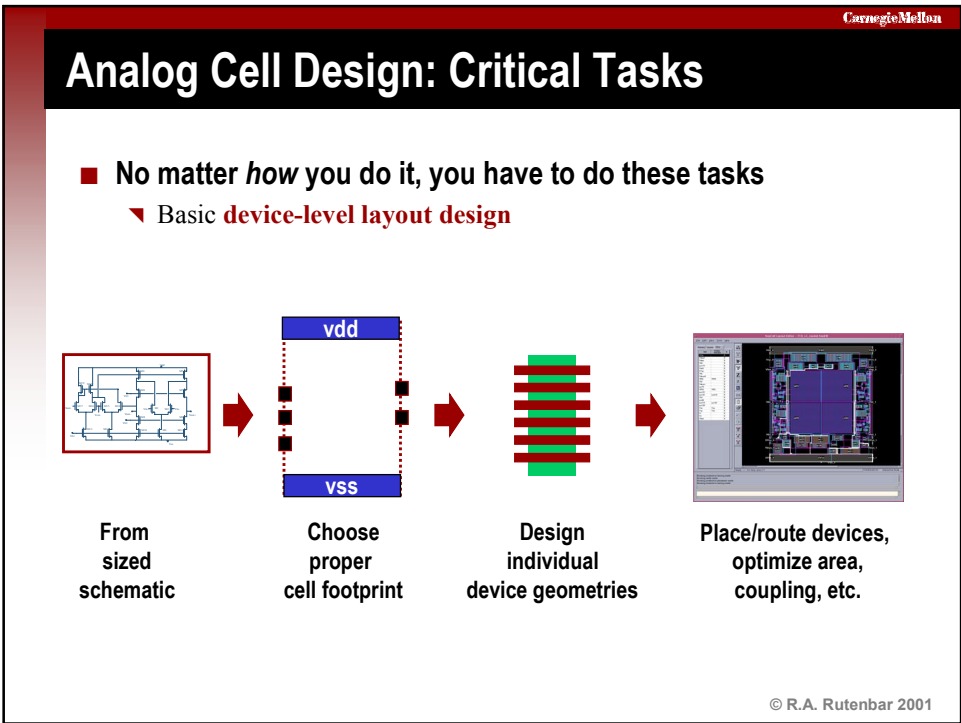
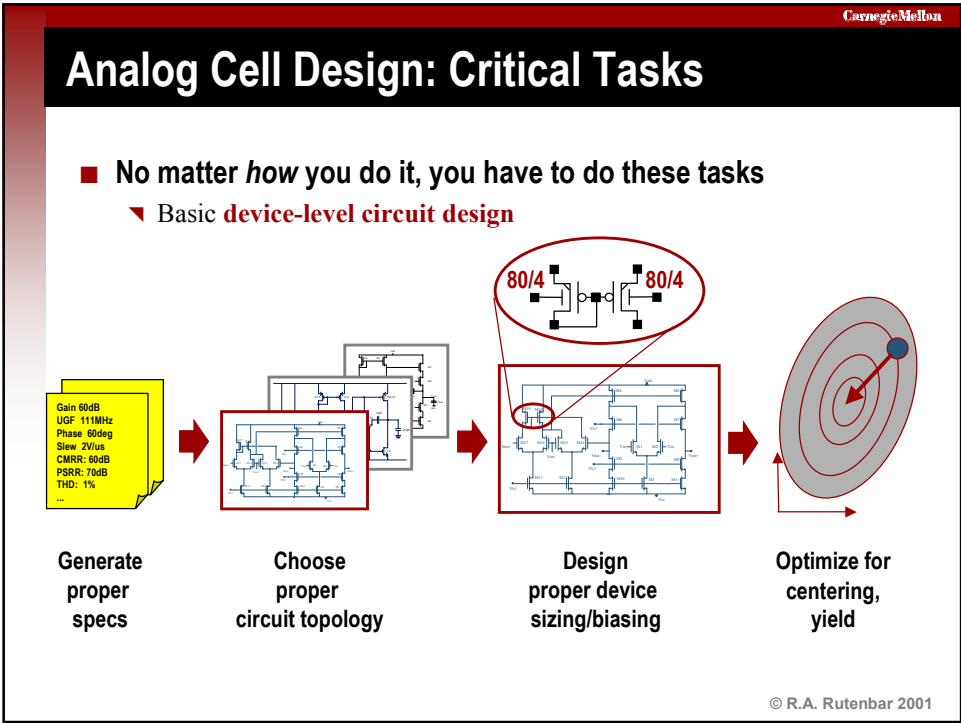
■ Common cells



■ Common subsystems composed from basic cells



© R.A. Rutenbar 2001



Why Is This Actually Difficult...?

■ Common misperceptions here

- ▼ Based mostly on familiarity with digital cells, digital libraries, and with digital design scenarios

■ Myth of “limited size”

- ▼ “Hey--only 50 transistors? How **hard** can that be to design?”
- ▼ “I don’t see people obsessing over NAND gate design!”

■ Myth of “limited libraries”

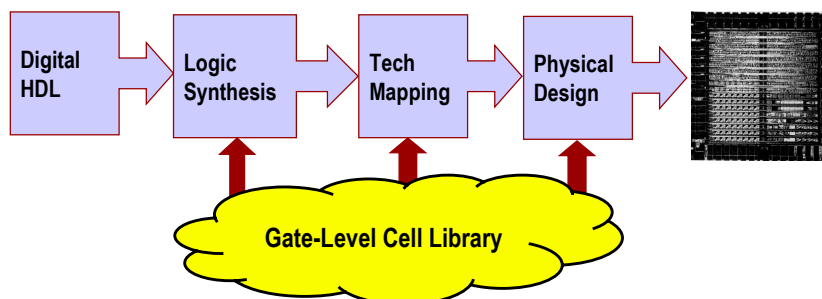
- ▼ “There’s not much analog on chip, and it’s mostly understood functions like A/D and D/A, so why not just design all the required cells **once**, put them in a library, reuse them?”

© R.A. Rutenbar 2001

Reminder: Cell-Based Digital Design

■ Digital ASIC design

- ▼ Often **starts** from assumed library of cells (maybe some cores too)
- ▼ Supports changes in cell-library; assumed part of methodology
- ▼ Cell libraries heavily **reused** across different designs

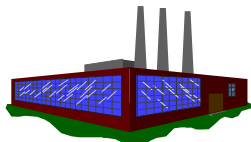


© R.A. Rutenbar 2001

Cell-Based Design Strategies: Digital

■ Where do digital cells come from?

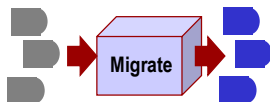
Foundries:
Optimized for
this fab



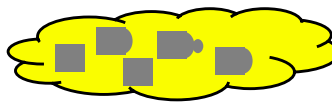
3rd Party IP:
Emphasize
portability, quick use



Migration Tools:
Old cells -> new cells



Manual, Custom Design:
Proprietary or custom library



© R.A. Rutenbar 2001

Cell-Based Design Strategies: Analog



■ Where do analog cells come from?

- ▼ Mainly **manual** design
- ▼ Often, manual **redesign**
- ▼ Almost **no** reuse

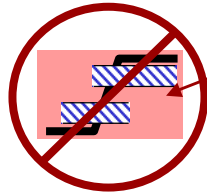
■ Why is this?

© R.A. Rutenbar 2001

Analog Cells: Strong Fab Dependence

- No digital abstraction to “hide” process

- ▼ No logic levels, noise margins, etc, on analog cells



Can't hide behind nice 1s and 0s...

- Exploits physics of fab process, instead of avoiding it

- ▼ Individual devices designed to achieve precise behaviors
 - ▼ Especially true with precision passive devices, which might require separate process steps (eg, double poly for capacitors)
 - ▼ Circuits sensitive to all aspects of device/interconnect behavior, even modest changes due to simple dimensional shrinks

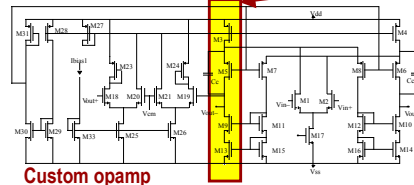
© R.A. Rutenbar 2001

Analog Cells in Digital Processes

- For SoC designs, want analog in standard digital process

- Common problems

- ▼ Low supply voltages preclude some circuit topologies



4-high gate stack works fine in 2 μ m, fails in deep submicron due to lack of ΔV_{GS}

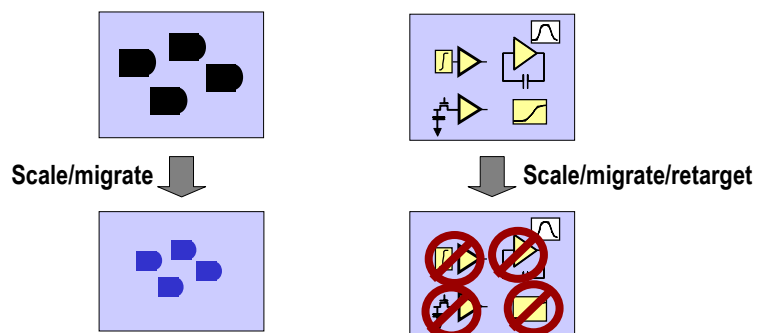
- ▼ Precision structures may be hard/impossible to build if special layers are unavailable (eg, poly-poly capacitor)
 - ▼ Digital processes do not characterize devices for analog uses, eg, models do not capture subthreshold ops, matching, etc

© R.A. Rutenbar 2001

Result: Analog Cells Resist Scaling/Migration

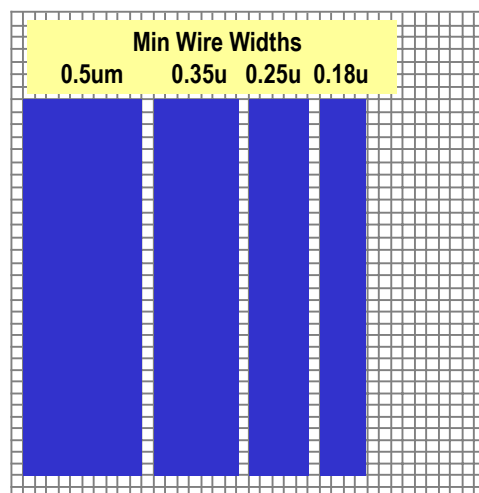
■ Analog cells manipulate precise electrical quantities

- ▼ Depend on precise physical parameters, precise device geometry
- ▼ Scale or migrate: process changes, so must **redo** circuit and layout
- ▼ Retarget circuit function: specs change (even a *little*), must **redo** ckt



© R.A. Rutenbar 2001

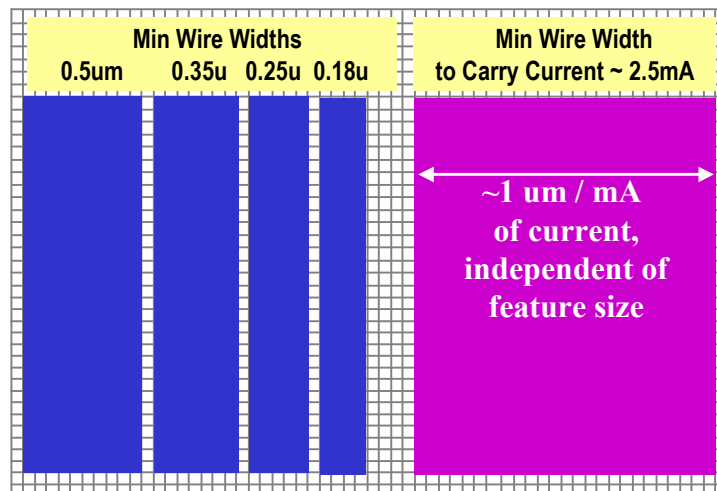
Note: Feature Size Is Scaling...



© R.A. Rutenbar 2001

Note: ... Electrical Interface Specs May *Not*

- Example: currents in critical wires affects min allowed width



© R.A. Rutenbar 2001

Major Impact: Analog is *Less* Library-able

- Cell design difficulty, libraries

- ▼ OK, so, maybe it's hard to design an analog cell.
- ▼ So, why not just **design it once**, add to lib, reuse it?

- Problem: leverage not same for analog libraries

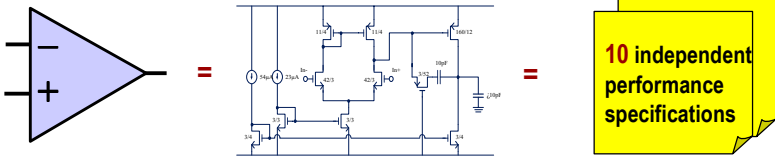
- ▼ How big is a digital library? Big enough to get all necessary logic functions, IO variants, timing variants, drive strengths, to first order

$$\left\{ \begin{array}{l} \text{Logic} \\ \text{functions} \end{array} \right\} \times \left\{ \begin{array}{l} \text{Fanin \& fanout} \\ \text{variants} \end{array} \right\} \times \left\{ \begin{array}{l} \text{Timing,} \\ \text{latch/FF,} \\ \text{scan} \\ \text{variants} \end{array} \right\} \times \left\{ \begin{array}{l} \text{Drive} \\ \text{strength} \\ (1X, 2X \\ 4X, 8X) \\ \text{variants} \end{array} \right\} = \sim 1k-2k \text{ cells}$$

© R.A. Rutenbar 2001

Analog Cell Libraries: Dimensionality

■ Problem: **many** continuous specs for analog cells



➔ **X** $\left\{ \begin{array}{l} \text{Spec=LOW} \\ \text{Spec=HIGH} \\ \text{variants} \\ \text{for ALL} \\ \text{combinations} \end{array} \right\} = \sim 1000 \text{ variants for just this cell}$

■ Can't just build a practical-size, **universal** analog library

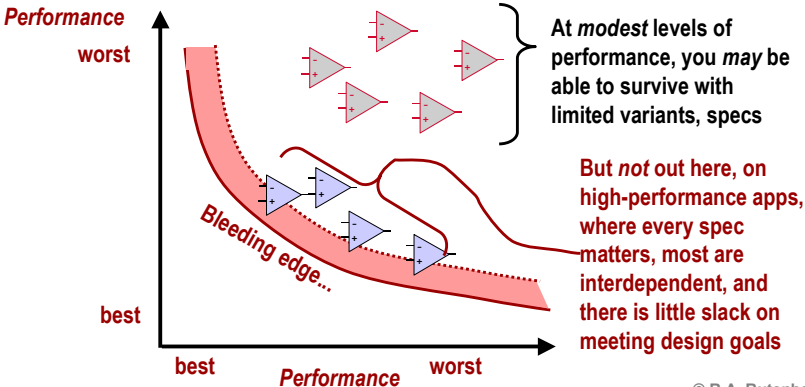
© R.A. Rutenbar 2001

Analog Cell Libraries: Dimensionality

■ Dimensionality: Reality check

- ▼ OK, do you really need all 1000 of those variants?
- ▼ Can't we make do with just a **few**--like we do for digital gates?

■ Maybe: depends on your application



At modest levels of performance, you may be able to survive with limited variants, specs

But *not* out here, on high-performance apps, where every spec matters, most are interdependent, and there is little slack on meeting design goals

© R.A. Rutenbar 2001

CarnegieMellon

Analog Cells: Design & Reuse Strategies

- 2 major issues
 - ▼ How do I make it easier to **design** this cell in the first place?
 - ▼ How do I avoid designing it again? Can I **reuse** it, wrap/buy it as **IP**?
- Design: focuses at 3 levels
 - ▼ Devices, cells, cores
- IP/reuse: focuses on 3 strategies
 - ▼ Hard, firm, soft IP strategies

© R.A. Rutenbar 2001

CarnegieMellon

Analog Cells: Design & Reuse Strategies

- Simple taxonomy

		IP/REUSE		
		hard	firm	soft
DESIGN	device	Libraries of difficult, exotic device layouts	Parametric device layout generators	--
	cell	Libs of generic cell layouts for specific fab	Parametric templates for schematic, layout	Analog ckt synthesis and layout synthesis
	core	Libs of useful block layouts for specific fab	Parametric templates for useful cores	Mixed-signal system assembly

Focus is on layout reuse

Focus is on reusable circuit & layout templates

Focus is on synthesis, from spec to ckt to layout

© R.A. Rutenbar 2001

CarnegieMellon

Analog Cell Design & Reuse

■ What are people *most commonly* doing right now?

▼ (Actually, they're mostly designing *by hand*, one device at a time...)

		IP/REUSE		
		hard	firm	soft
DESIGN	device	Libraries of difficult, exotic device layouts	Parametric device layout generators	--
	cell	Libs of generic cell layouts for specific fab	Parametric templates for schematic, layout	Analog ckt synthesis and layout synthesis
	core	Libs of useful block layouts for specific fab	Parametric templates for useful cores	Mixed-signal system assembly

© R.A. Rutenbar 2001

CarnegieMellon

First, Look at Device-Level Issues

■ Question: why the emphasis on *individual* devices...?

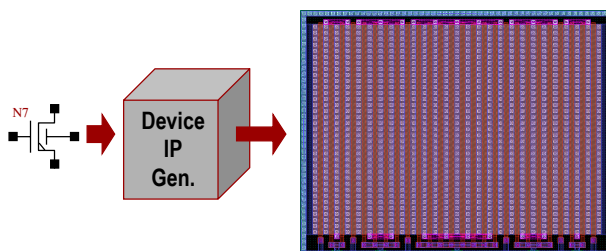
		IP/REUSE		
		hard	firm	soft
DESIGN	device	Libraries of difficult, exotic device layouts	Parametric device layout generators	--
	cell	Libs of generic cell layouts for specific fab	Parametric templates for schematic, layout	Analog ckt synthesis and layout synthesis
	core	Libs of useful block layouts for specific fab	Parametric templates for useful cores	Mixed-signal system assembly

© R.A. Rutenbar 2001

Analog Device IP

■ Basic idea

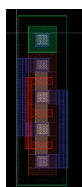
- ▼ Analog cells require “difficult” device structures
- ▼ May need large devices, aggressive matching, unusual precision
- ▼ Can save device layouts in a library, or more commonly...
- ▼ ... write **layout generators**; may be provided by your foundry
- ▼ Implementations vary: can use commercial frameworks (Mentor GDT, Cadence PCELL), or write your own (C++, JAVA, etc)



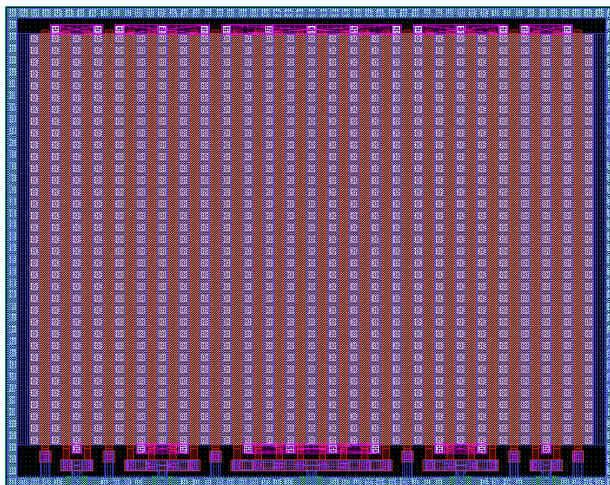
© R.A. Rutenbar 2001

Device-Level IP: What “Large” Means

Digital FET



Analog FET



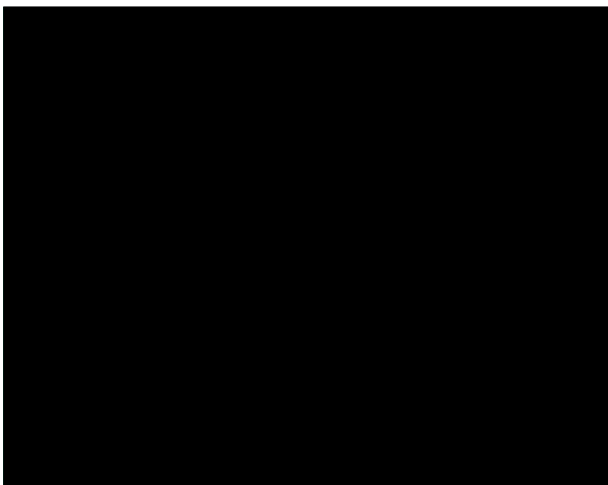
© R.A. Rutenbar 2001

Device-Level IP: Limited Porosity

Also,
precision devices
almost never
allow wires
over the top,
to minimize
potential coupling.

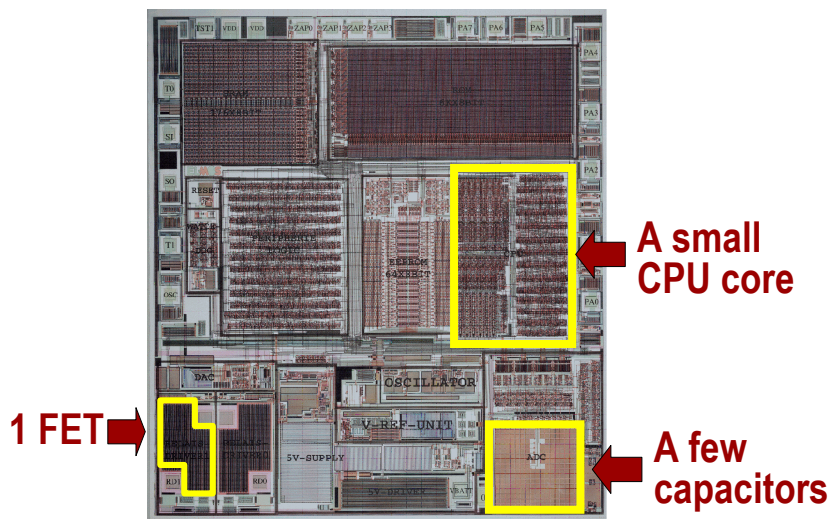
This whole
object is
*blocked for
upper metals.*

Analog FET



© R.A. Rutenbar 2001

Large Can Mean Very Large, Too

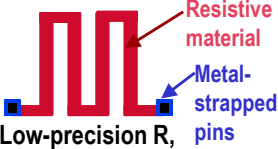


© R.A. Rutenbar 2001

CarnegieMellon

Example: Analog Precision Tricks for Devices

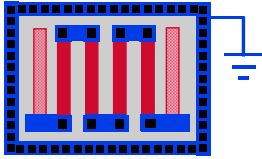
■ Consider a resistor which uses a resistive poly layer



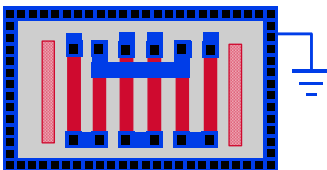
Low-precision R, poly snake resistor



Higher-precision R, poly bars with all-metal interconnect



High-precision R, add dummy bars at ends, well and guard ring

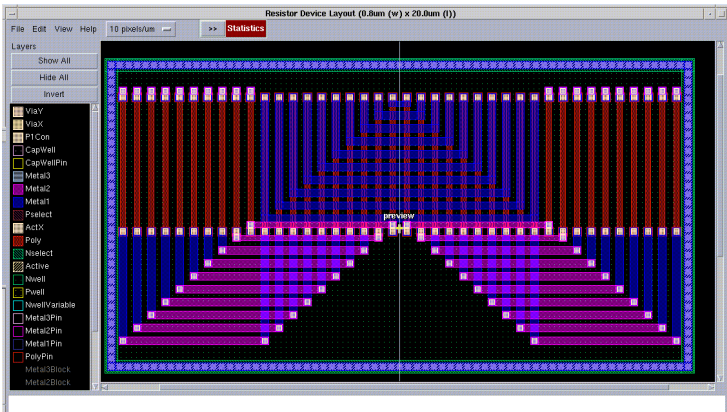


Interdigitated pair of precise-ratio 2:1 resistors

© R.A. Rutenbar 2001

CarnegieMellon

Industrial Ex: Precision Interdig Resistor Array



Courtesy Neoliner

© R.A. Rutenbar 2001

©R.A. Rutenbar, 2001

25

Next, Look at Hard Analog IP

- Question: how much can you *reuse* complete layouts?

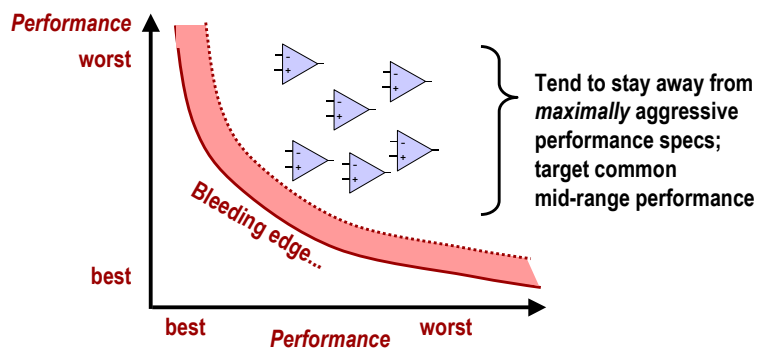
		IP/REUSE		
		hard	firm	soft
DESIGN	device	Libraries of difficult, exotic device layouts	Parametric device layout generators	--
	cell	Libs of generic cell layouts for specific fab	Parametric templates for schematic, layout	Analog ckt synthesis and layout synthesis
	core	Libs of useful block layouts for specific fab	Parametric templates for useful cores	Mixed-signal system synthesis

© R.A. Rutenbar 2001

Hard Analog Cell IP

■ Basic idea

- ▼ Hard IP (layouts) for common, generic cell functions
- ▼ Performance ranges estimated to target common application areas (eg, audio, video, LAN, IO driver, etc)
- ▼ Available from some foundries; also some 3rd party IP shops who design for standard digital fabs



© R.A. Rutenbar 2001

CarnegieMellon

Hard Analog Cell IP: Analysis

■ Pro

▼ Again, makes it easy to do some simple functions

■ Con

▼ Unlike digital libraries, **unlikely** that 100% of needed cells available

▼ And, cell portfolio will differ significantly from vendor to vendor

Your mixed signal ASIC

Vendor 1 Coverage

Vendor 2 Coverage

Vendor 3 Coverage

Sorry, this requires **custom** analog-- more design effort, impact on design risk

© R.A. Rutenbar 2001

CarnegieMellon

Focus Now on Design & Synthesis

■ OK, suppose you can't just buy the analog cells you need; what can you do to help **design** them faster, better?

		IP/REUSE		
		hard	firm	soft
DESIGN	device	Libraries of difficult, exotic device layouts	Parametric device layout generators	--
	cell	Libs of generic cell layouts for specific fab	Parametric templates for schematic, layout	Analog ckt synthesis and layout synthesis
	core	Libs of useful block layouts for specific fab	Parametric templates for useful cores	Mixed-signal system synthesis

© R.A. Rutenbar 2001

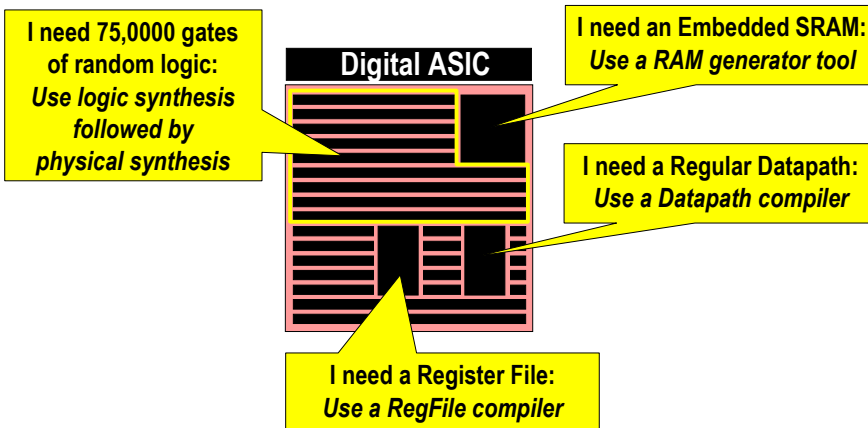
Cell-Level Strategies

- Aside from doing everything manually, are there options?
- Template-based design
 - ▼ If you keep designing the same cells, for similar ranges of performance, try to capture central characteristics as a template
 - ▼ Parameters fill in the template, change resulting design
- Analog synthesis
 - ▼ For more general case, specify critical performance constraints (electrical, geometric, etc)
 - ▼ Synthesis tool uses numerical/geometric search to create circuit to match your design goals

© R.A. Rutenbar 2001

Analogy from Digital World

- How do people put big ASICs together today?
 - ▼ In big pieces, compiling & synthesizing the chunks as needed

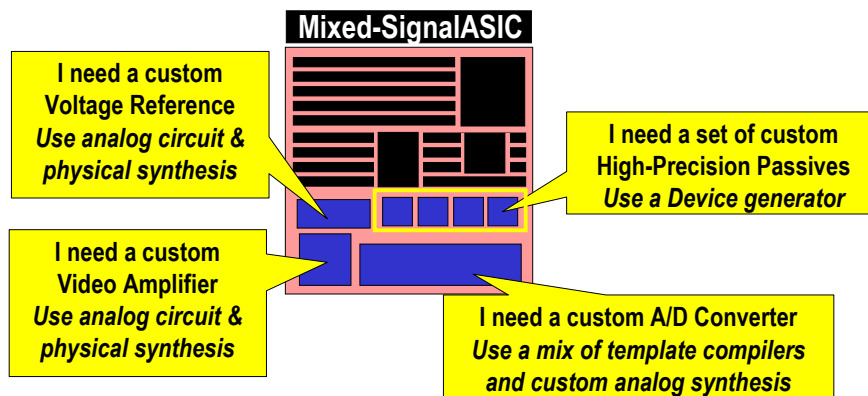


© R.A. Rutenbar 2001

On the Analog Side of a Mixed-Signal SoC...

■ We want the **same** sort of functionality

- ▼ Synthesis: for the very custom cells that determine analog performance
- ▼ Templates: for the less custom, more regular stuff left over

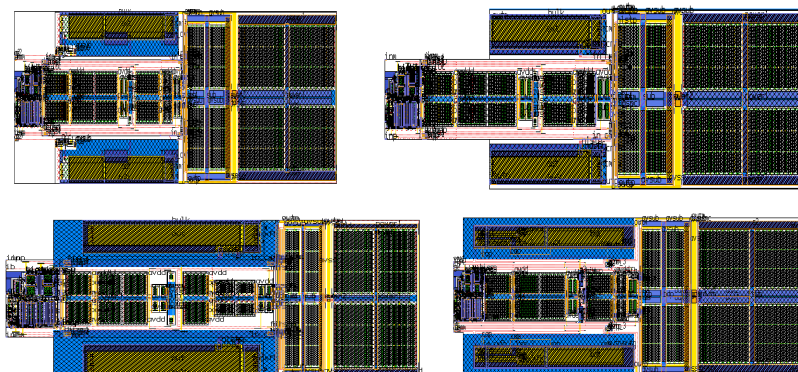


© R.A. Rutenbar 2001

Template Example: CMOS Analog Cells

■ Manually capture regularities as procedures for high-use cells

- ▼ Can mix device generators, cell generators, compaction ideas, etc.
- ▼ Still requires significant manual setup & maintenance investment



Courtesy Koen Lampaert, Conexant

© R.A. Rutenbar 2001

Carnegie Mellon

Template Example: RF Components

■ Optimizes LC-oscillators from specs to layout [Deranter DAC'00]

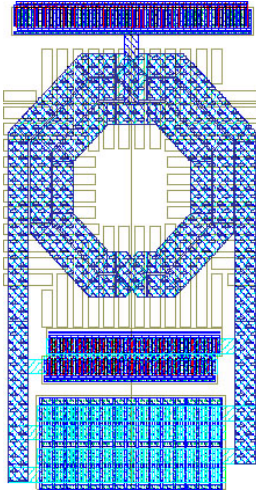
▶ Simulated annealing in combination with circuit simulations and some equations

▶ FEM simulations to characterize inductor coils

▶ Auto template-based generation of VCO layout

Parameter	Low resistive sub CMOS 0.35μm	High resistive sub BiCMOS 0.65μm
Ls	1.26 nH	2.3 nH
Rs	6.5 Ω	5.2 Ω
Rad, W, Turns	109 μm, 40 μm, 2	141 μm, 5 μm, 2
Power	32 mW	8.2 mW

Courtesy Georges Gielen, K. U. Leuven

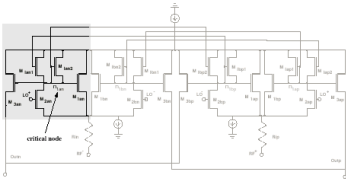


© R.A. Rutenbar 2001

Carnegie Mellon

Template Example: RF Components

■ RF mixer, circuit & layout optimized together, [Gielen ICCAD01]



One quarter of the mixer core

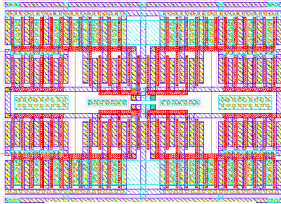
M1an M3an M3bp M1bp

M2an M1an1.2 M1bp1.2 M2bp

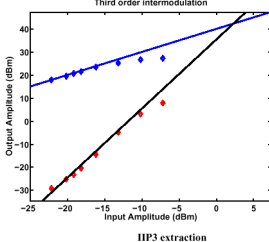
M2ap M1ap1.2 M1bn1.2 M2bn

M1ap M3ap M3bn M1bn

Complete mixer floorplan, with quarter-piece above highlighted



An optimized layout



Third order intermodulation

Output Amplitude (dBm)

Input Amplitude (dBm)

IP3 extraction

Courtesy Georges Gielen, K. U. Leuven

© R.A. Rutenbar 2001

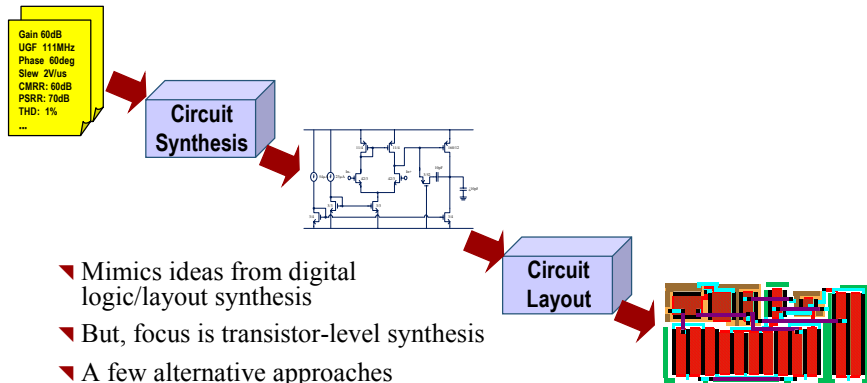
©R.A. Rutenbar, 2001

30

More General Attack: Analog Synthesis

■ Basic idea

- ▼ **Circuit synthesis:** transform cell spec into sized/biased schematic
- ▼ **Layout synthesis:** transform device-level netlist into laid-out cell

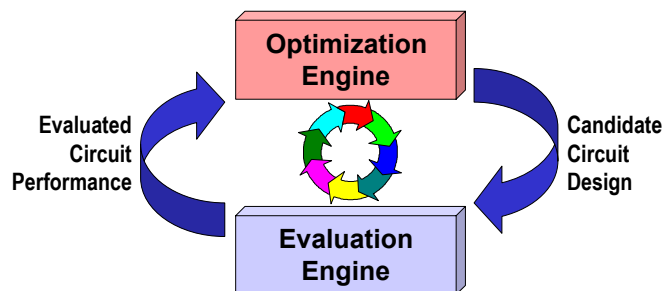


- ▼ Mimics ideas from digital logic/layout synthesis
- ▼ But, focus is transistor-level synthesis
- ▼ A few alternative approaches

© R.A. Rutenbar 2001

Cell-Level Synthesis: Framework

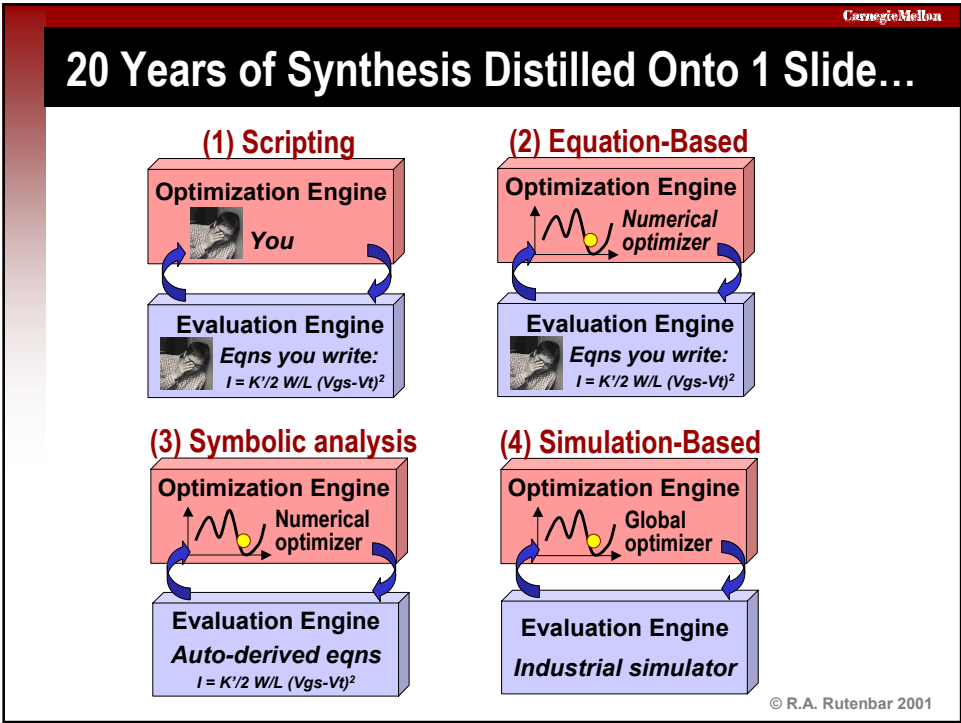
■ Most approaches have this overall structure



■ Uses heuristic or numerical search

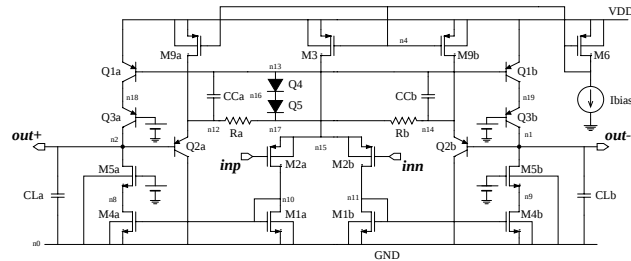
- ▼ **Optimization engine:** proposes candidate circuit solutions
- ▼ **Evaluation engine:** evaluates quality of each candidate
- ▼ **Cost-based search:** cost metric represents “goodness” of design

© R.A. Rutenbar 2001



Symbolic Analysis Example

- Katholieke Univ. Leuven, ISAAC/SYMBA tool [Gielen JCTh'95]



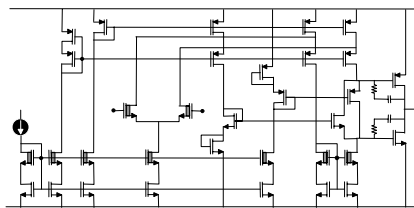
$$A_{V0} = \frac{g_{m,M2}}{g_{m,M1}} \frac{g_{m,M4}}{\left(\frac{g_{o,M4}g_{o,M5}}{g_{m,M5} + g_{mb,M5}} + \frac{G_a + g_{o,M9} + g_{o,Q2}}{\beta_{Q2}} \right)}$$

Courtesy Georges Gielen, KUL

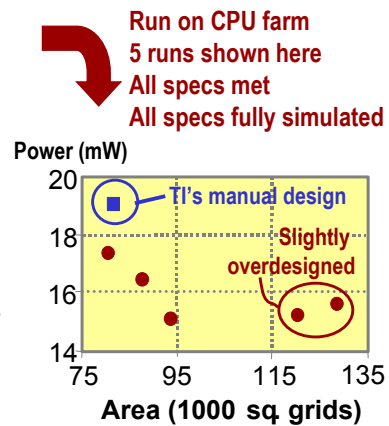
© R.A. Rutenbar 2001

Simulation Based Example: Cells from TI

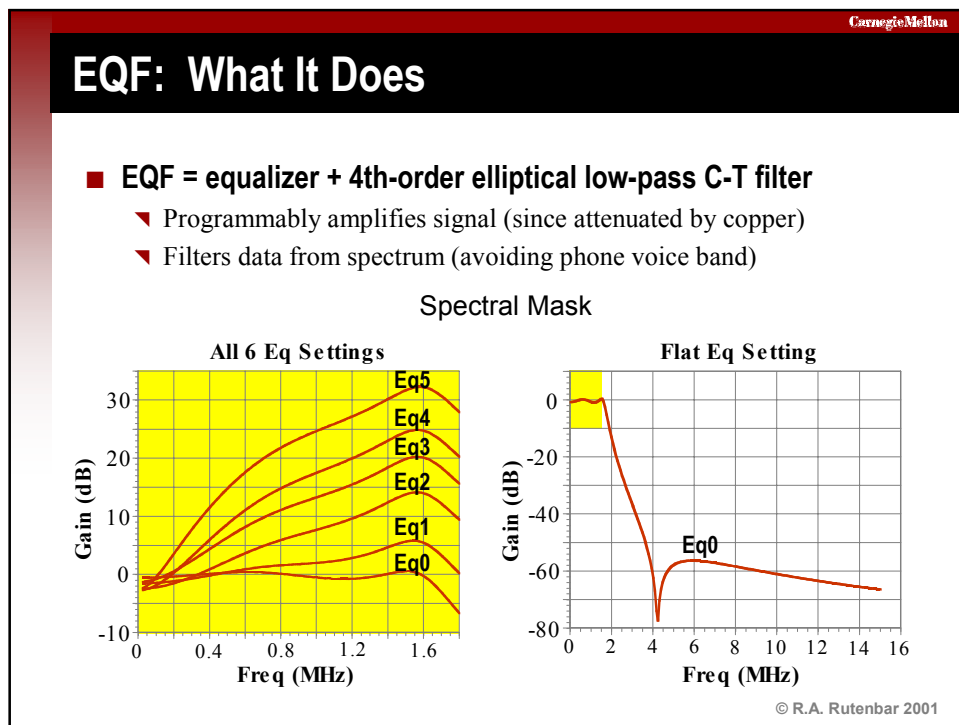
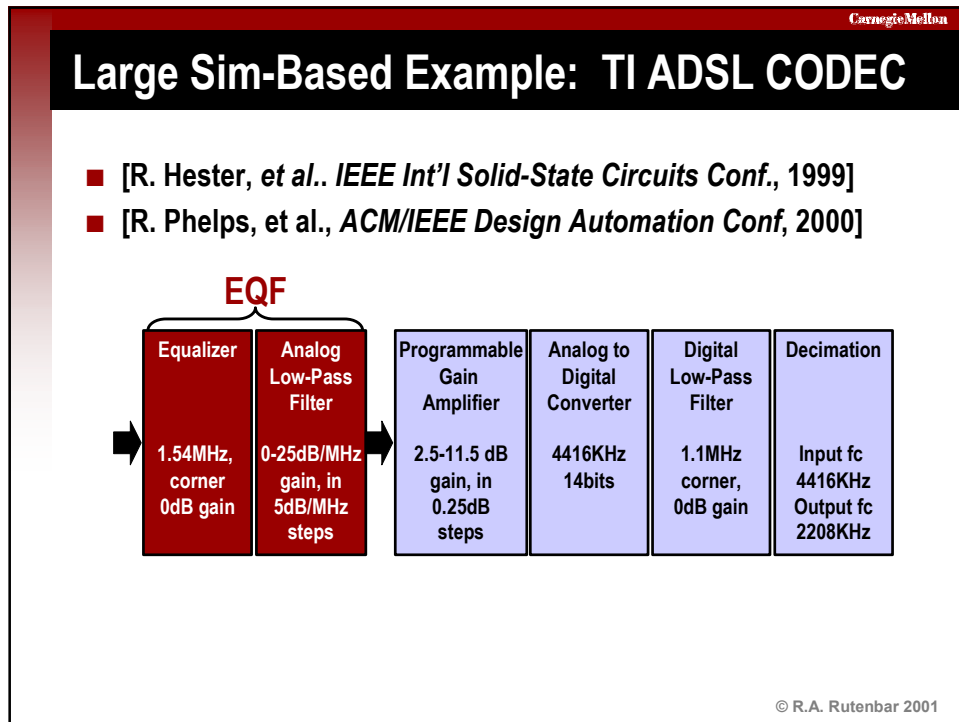
- Done using CMU ANACONDA tool [Phelps CICC99]

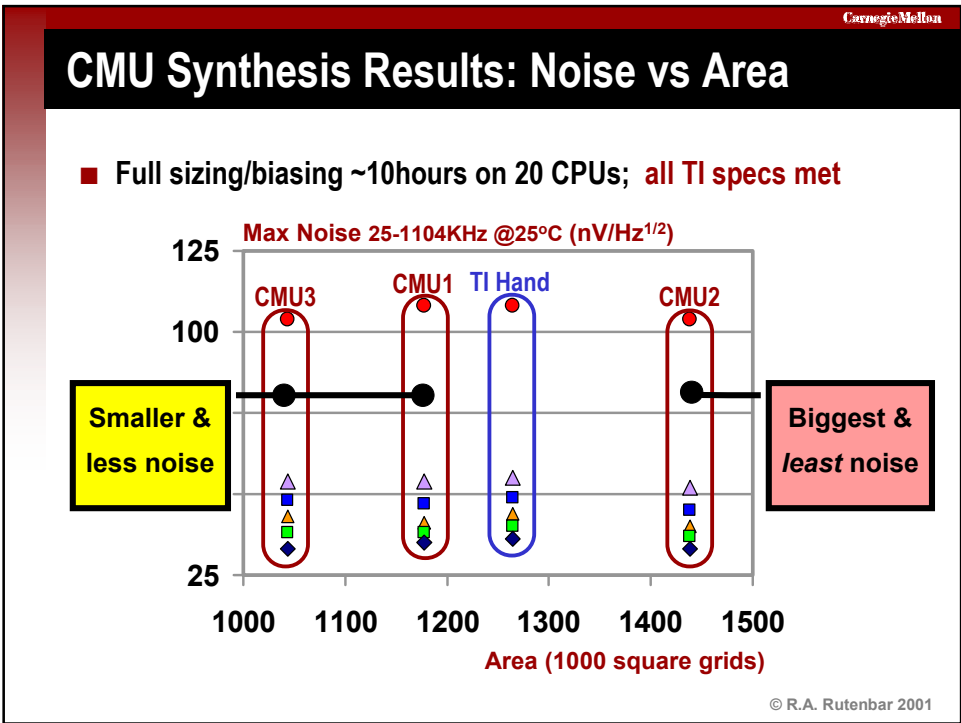
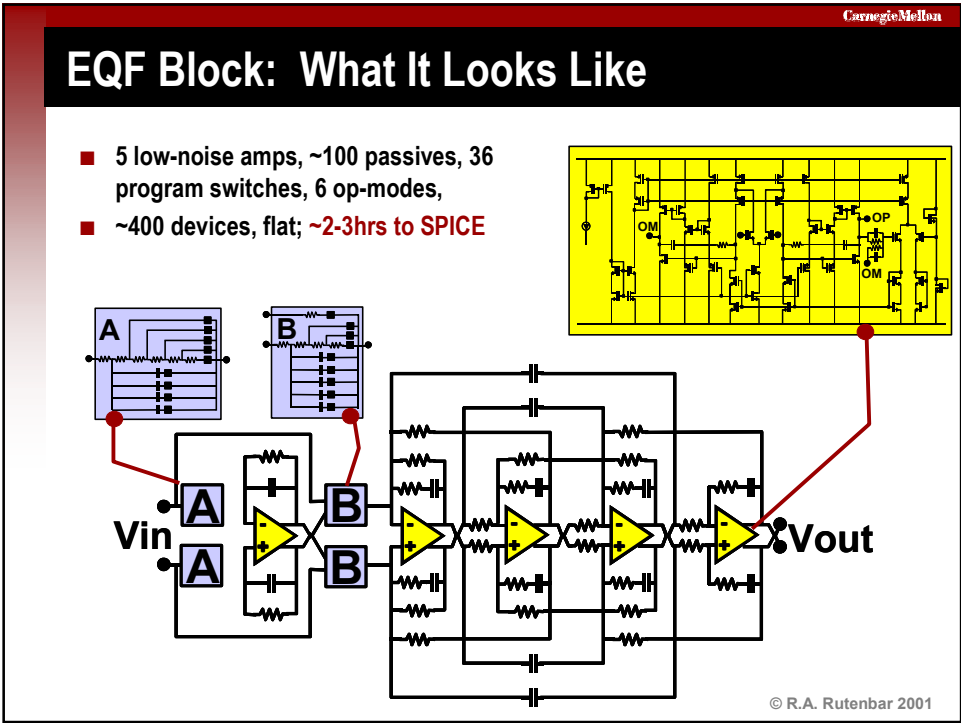


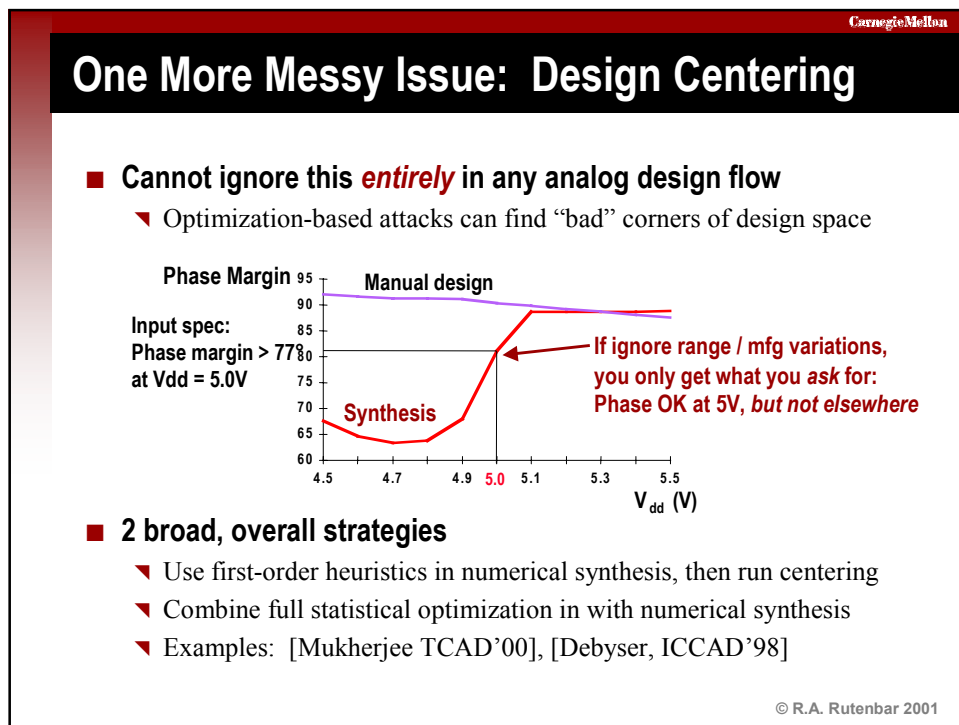
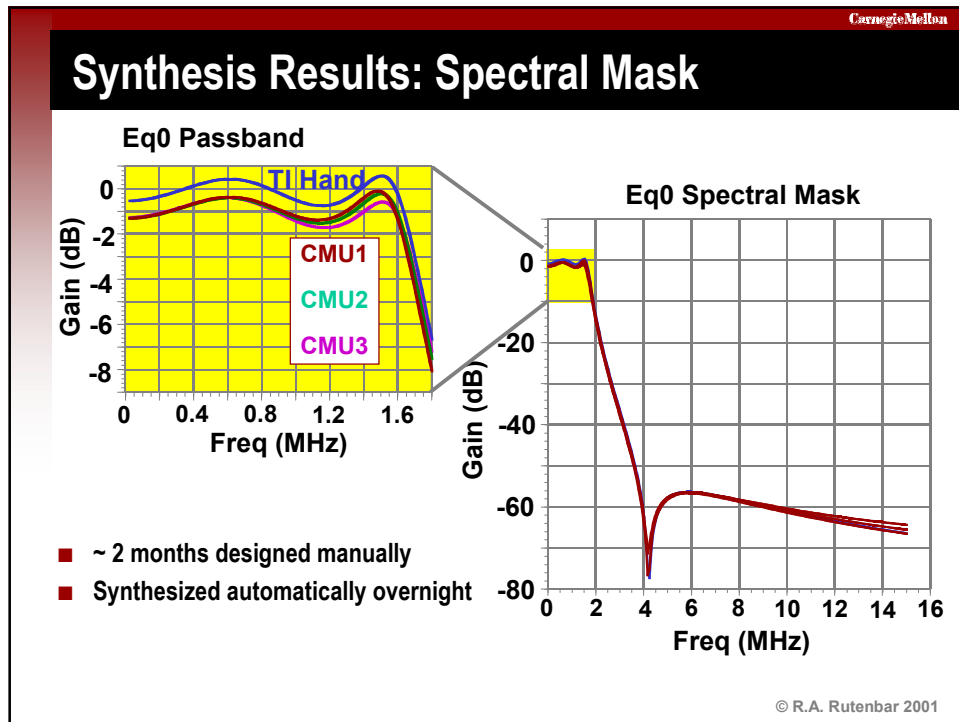
- ▼ **Folded cascode opamp, high-drive output stage**
 - ▼ 33 devs, 2 Rs, 2 Cs; 0.8um CMOS
- ▼ **Difficult goals**
 - ▼ High drive amplifier, 5Ωload
 - ▼ Nominal THD, 0.1%
 - ▼ 1kHz, 2.6V p-p input voltage



© R.A. Rutenbar 2001







Example: Centering Heuristics in Synthesis

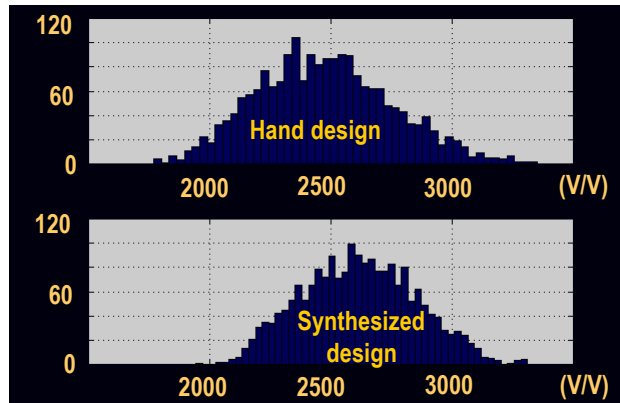
■ Simple designer-derived constraints in ANACONDA synthesis

- Require matched devices to be “big”; sensitive devices to be “far enough” into desired region of operation (eg, 250mV above V_T)

Example
Monte Carlo
spread for a small
TI opamp

3σ process,
+/-10% supply
& temp. variation

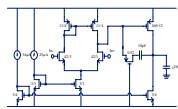
Plots show
low-frequency
gain for
manual, auto
designs



© R.A. Rutenbar 2001

Cell-Level Analog Layout Synthesis

■ Basic task



From schematic +
geometric constraints
to physical layout

■ Major strategies

- Enhanced polygon-editing
- Analog compaction & templates
- Physical synthesis: full device-level custom place/route

© R.A. Rutenbar 2001

Carnegie Mellon

Analog-Specific Optimizations: Place/Route

■ Placement symmetric and diffusion merging

No symmetry
No merging

Symmetry
No Merging

Symmetry
Merging

■ Routing: differential symmetric and coupling avoidance

Wiring task with
Obstacle

No symmetry
No crosstalk

Symmetry
No crosstalk

Symmetry
Crosstalk

[Cohn, JSSC91]

© R.A. Rutenbar 2001

Carnegie Mellon

Small Physical Synthesis Example: Close-up

■ Commercial tools emerging

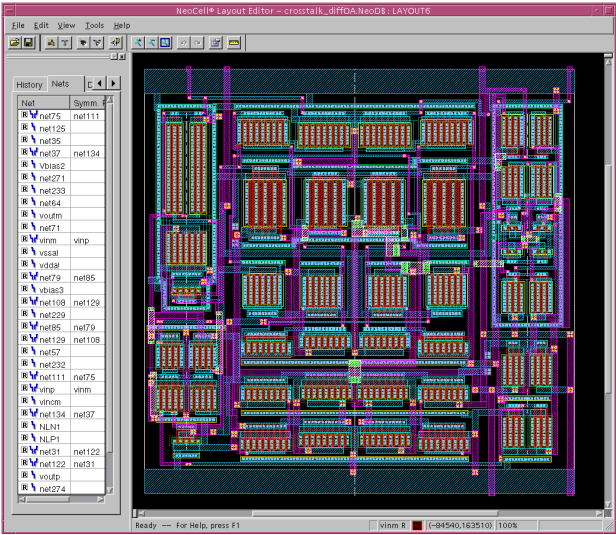
■ This example

▼ Neoliner's NeoCell

▼ CMOS

▼ ~50 devices

▼ Layout < 1 hr



Courtesy Neoliner

© R.A. Rutenbar 2001

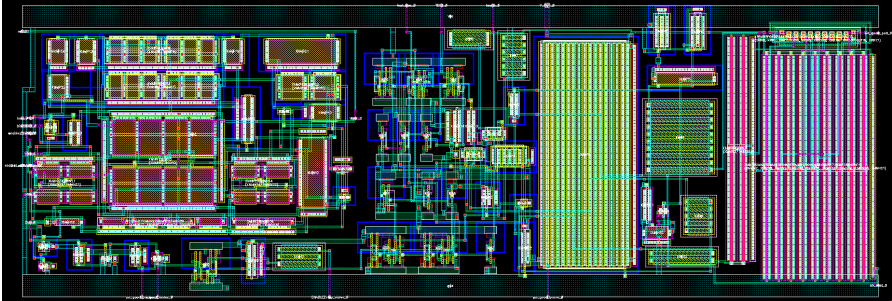
©R.A. Rutenbar, 2001

38

CarnegieMellon

Large Physical Synthesis Example

- Proprietary CMOS comparator auto-layout from NeoCell

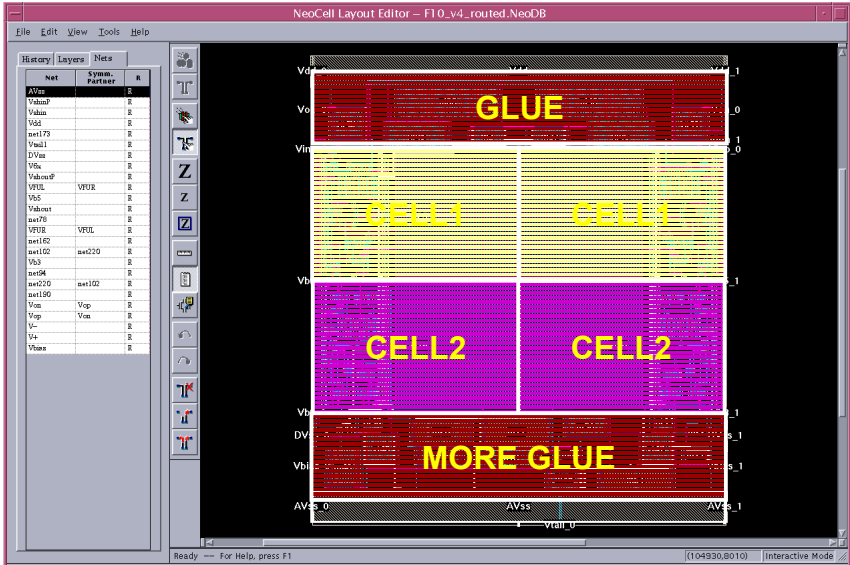


Courtesy Neolinear

© R.A. Rutenbar 2001

CarnegieMellon

Subsystem Example: Cells + Glue Circuits



Courtesy RocketChips and Neolinear

© R.A. Rutenbar 2001

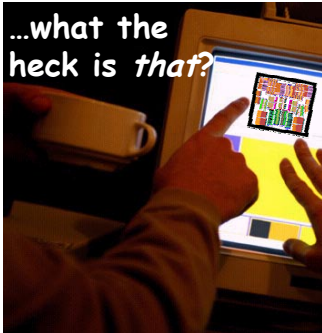
Carnegie Mellon

Historically—Why has this been so Hard?

Mediocre
analog point tools



Ad hoc, incomplete
capture of design intent



Too much art,
not enough science



- With new synthesis/analysis tools, improved methodologies, & improved *attitudes* about design—stage set for radical changes

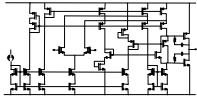
© R.A. Rutenbar 2001

Carnegie Mellon

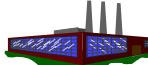
New Idea: Analog IP = Capture + Synthesis

- Commercial example from Neolinear NeoCircuit/NeoCell flow

Unsize commercial
diff-amp cell



0.6um proprietary
CMOS fab



TSMC 0.35um
CMOS fab



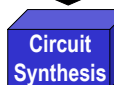
Circuit Synthesis



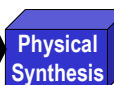
Physical Synthesis

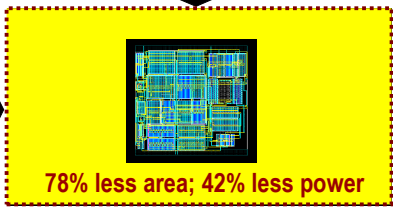


Circuit Synthesis



Physical Synthesis





78% less area; 42% less power

© R.A. Rutenbar 2001

CarnegieMellon

Outline

■ Quick tour of mixed-signal System-on-Chip (SoC) design

■ Design problems & strategies for analog building blocks

■ Design problems & strategies for mixed-signal chips

■ Talk emphasis

- ▼ We do all this analog design by hand, as painful full custom, today
- ▼ That has got to change—too many opportunities, too few designers
- ▼ What are the prospects for “buy it” or “reuse it” for analog?
- ▼ This is the hot topic in analog today: **analog intellectual property**

© R.A. Rutenbar 2001

CarnegieMellon

What's Left to Do: Chip-Level Design

■ OK, you design/buy/synthesize all your cells...then what?
Chip-level design. (...and, problems don't get easier)

		IP/REUSE		
		hard	firm	soft
DESIGN	device	Libraries of difficult, exotic device layouts	Parametric device layout generators	--
	cell	Libs of generic cell layouts for specific fab	Parametric templates for schematic, layout	Analog ckt synthesis and layout synthesis
	core	Libs of useful block layouts for specific fab	Parametric templates for useful cores	Mixed-signal system assembly

© R.A. Rutenbar 2001

Carnegie Mellon

Hard Analog Core IP (= Mixed-Signal IP)

- Recent commercial idea
 - ▼ Don't focus on basic cells, focus on **bigger mixed-signal cores**
 - ▼ Industry standards **fix** many specs; target big ASIC foundries
 - ▼ Interesting technical (& business) issues here

IP/REUSE			
	hard	firm	soft
device	Library of difficult-to-use device layouts	Parametric device layout generators	—
cell	Library of generic cell layouts for specific fab	Parametric templates for automatic layout	Analog cell synthesis and layout synthesis
core	Library of useful block layouts for specific fab	Parametric templates for useful cores	Mixed-signal system assembly

© R.A. Rutenbar 2001

Carnegie Mellon

Template-Based System Layout Example

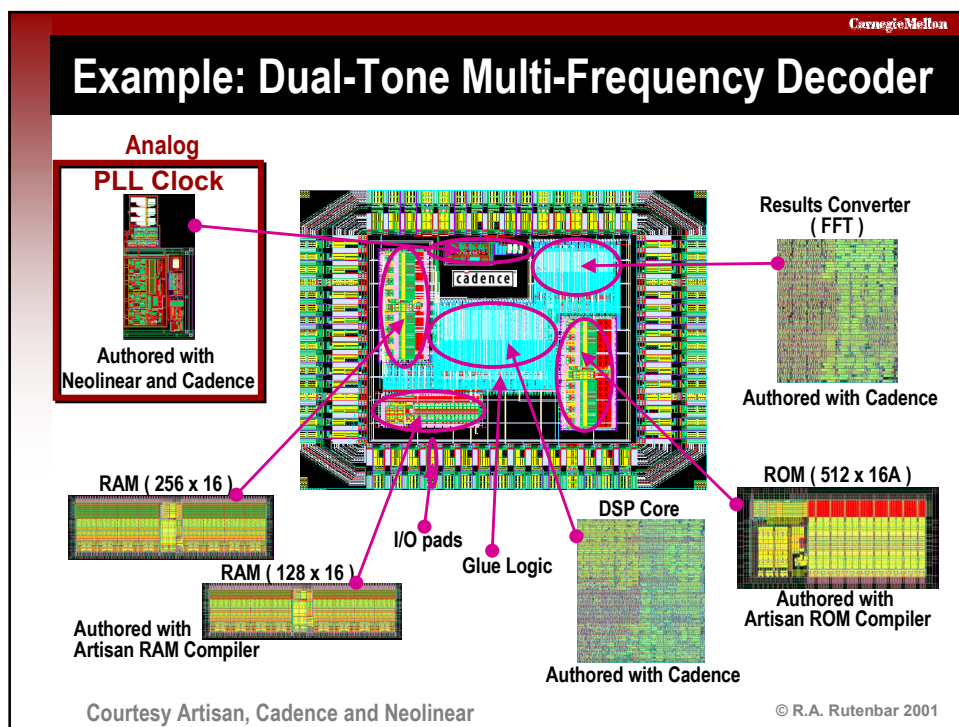
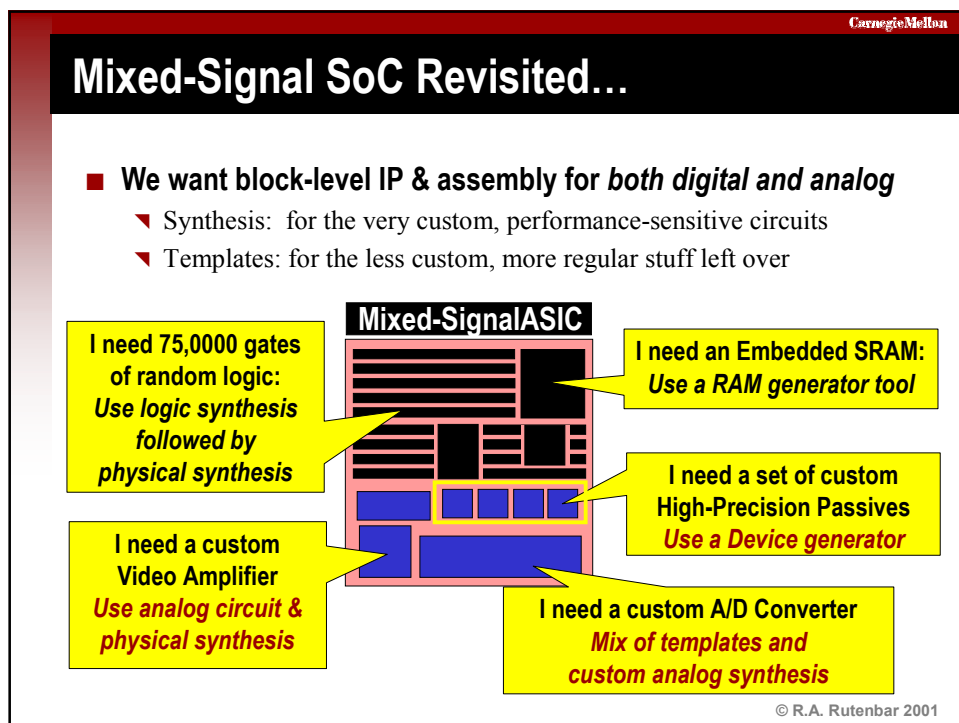
- Analogy: just like digital datapath generators
 - ▼ Can exploit analog regularities you know; procedurally generate

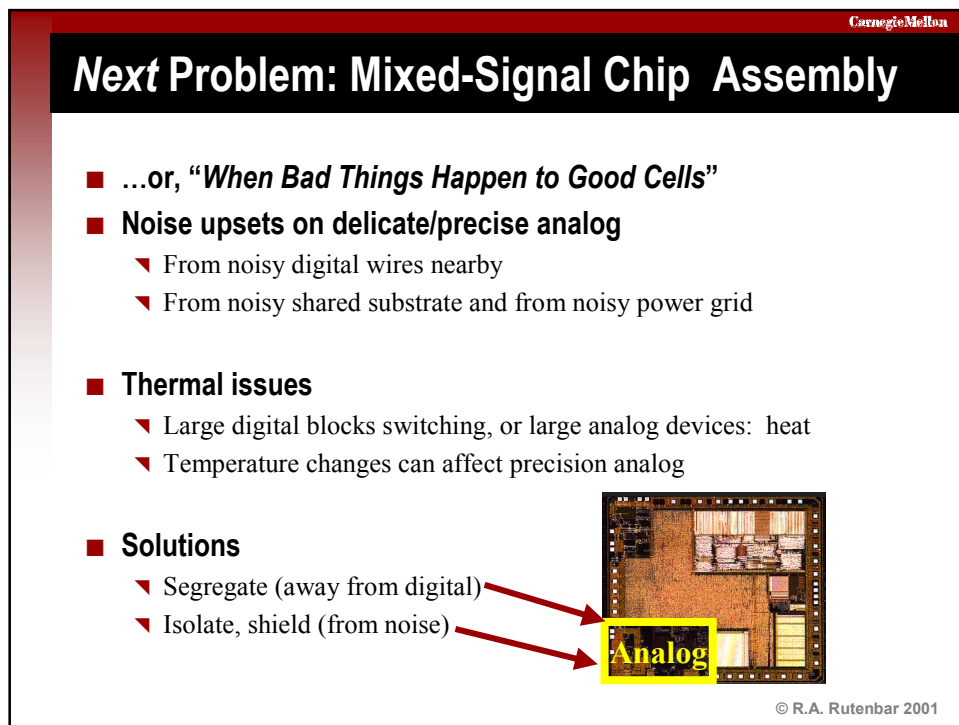
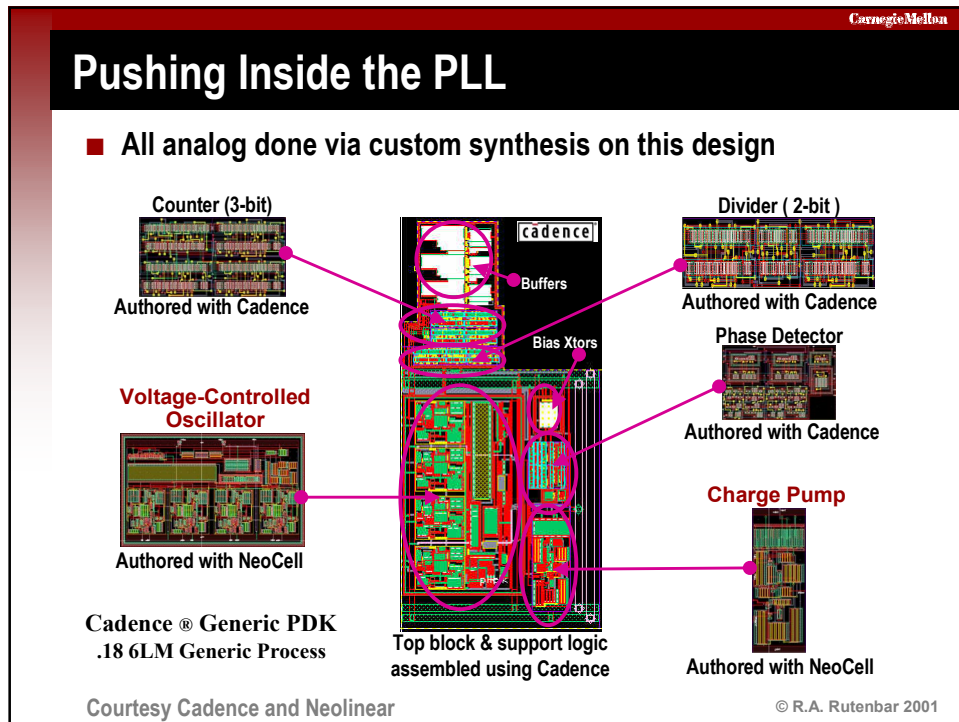
14-bit 150-Ms/s 0.5um CMOS DAC

[ISSCC'99]

J. Vandenbussche, G. Van der Plas, A. Van den Bosch, W. Daems, G. Gielen, M. Steyaert, W. Sansen

Courtesy Georges Gielen, K.U. Leuven © R.A. Rutenbar 2001





Carnegie Mellon

Noise At Mixed-Signal Chip Level

- Coupled through supply rails and common substrate
 - ▼ Precise analog biasing easily vulnerable to voltage upset

© R.A. Rutenbar 2001

Carnegie Mellon

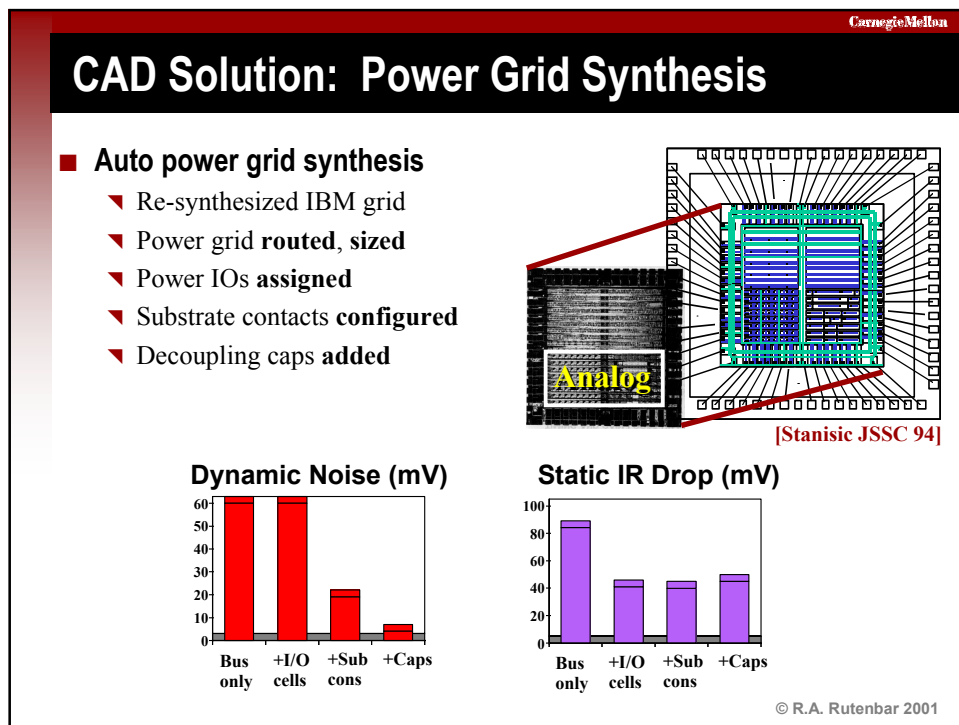
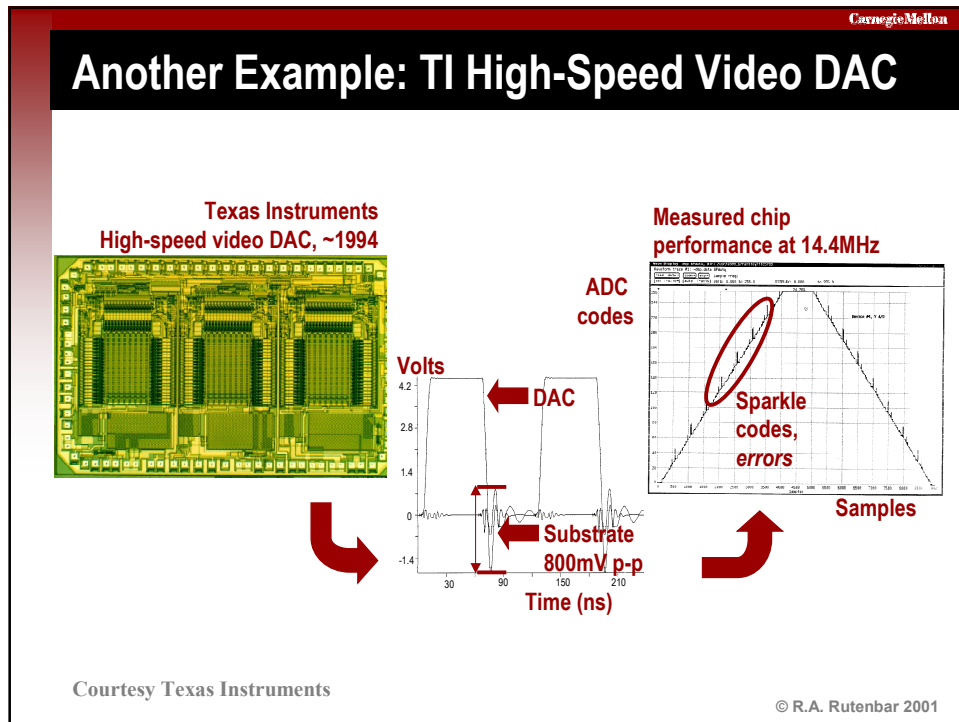
One Assembly Example: IBM Data Channel

- Digital switching is the source of (almost) all evil for analog

Measurements from IBM disk data channel;
Substrate noise spec 4mV -- exceeded

Courtesy Bob Stanicic/Tim Schmerbeck, IBM

© R.A. Rutenbar 2001



Mixed-Signal Chip-Level Assembly Today

■ Embarrassingly ad hoc

- ▼ Lots of guessing (and lots of praying) about floorplan, global signal routing, block-level isolation structures, etc
- ▼ Often vastly over-conservative; sometimes just plain wrong
- ▼ Often takes a few silicon spins to iron out (“few” may mean 5-10 at RF and higher frequencies)

■ Where the action is

- ▼ Full-chip and package extraction and simulation for noise coupling
- ▼ Smarter circuit design methodologies for noise immunity (think “echo cancelation”, but replace “echo” with “substrate noise”...)

© R.A. Rutenbar 2001

Conclusions

■ Analog circuits: here to stay

- ▼ In an SoC world, big systems need to talk to the external world
- ▼ The world is analog (...get used to it); analog does this communication

■ Mixed-signal design realities

- ▼ Analog cells != digital cells
- ▼ Not as easily library-able; don't scale; don't migrate
- ▼ Tightly bound to fab process, difficult precision requirements
- ▼ Chip level assembly is nasty

■ Design strategies

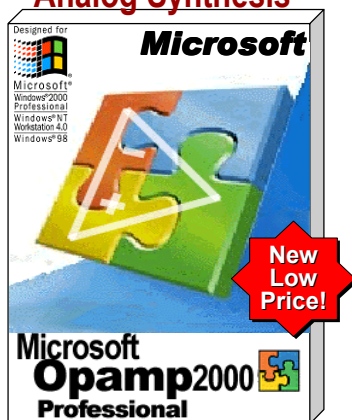
- ▼ Less art, more science: better methodologies, real synthesis tools
- ▼ Analog IP: design for migrating, retargeting is the next big thing

© R.A. Rutenbar 2001

Where all this Analog IP Stuff is Heading

Analog folks want *IP / reuse*, too

Analog Synthesis



Analog IP



© R.A. Rutenbar 2001

Select References

■ General Analog CAD Survey

- ▼ R. A. Rutenbar, "Analog Design Automation: Where are We? Where are we Going?", *Proc. IEEE CICC*, May 1993.
- ▼ L. R. Carley, G. Gielen, R. A. Rutenbar, W. Sansen, "Synthesis Tools for Mixed-Signal ICs: Progress on Frontend and Backend Strategies," *Proc. ACM/IEEE DAC*, June 1996.
- ▼ G. G. E. Gielen, R. A. Rutenbar, "Computer Aided Design of Analog and Mixed-Signal Integrated Circuits", *Proceedings of the IEEE*, December 2000.
- ▼ **Computer Aided Design of Analog and Mixed-Signal ICs**, B. Antao, G. Gielen, R. Rutenbar, editors, *IEEE Press*, to appear 2001.

■ IP Issues

- ▼ Steve Ohr, "Analog IP Slow to Start Trading", *EETimes*, Issue 1053, March 22 1999.
<http://www.eet.com> (Steve Ohr covers analog design/EDA for EETimes)
- ▼ K.C. Murphy, "A Time for Analog Design", *Electronic News Online*, August 2 1999.
<http://www.electronicnews.com/enews/BackIssues/BackIssues.asp>
- ▼ <http://www.vsia.com> -- Virtual Socket Interface Alliance working on specs for interchange of analog IP

© R.A. Rutenbar 2001

Select References

■ Analog Synthesis

- ▼ M.G.R. DeGrauwe et. al, "IDAC: An Interactive Design Tool for Analog CMOS Circuits", *IEEE Journal of Solid-State Circuits*, December 1987.
- ▼ H.Y. Koh, C.H. Sequin, P.R. Gray, "OPASYN: A Compiler for CMOS Operational Amplifiers", *IEEE Transactions on CAD*, Feb. 1990.
- ▼ R. Harjani, R.A. Rutenbar and L. Richard Carley, "OASYS: A Framework for Analog Circuit Synthesis", *IEEE Transactions on CAD*, Dec. 1989.
- ▼ G. Gielen, Walscharts, W. Sansen, "Analog circuit design optimization based on symbolic analysis and simulated annealing", *IEEE Journal of Solid-State Circuits*, June 1990.
- ▼ J. P. Harvey, M.I. Elmasry and B. Leung, "STAIC: An Interactive Framework for Synthesizing CMOS and BiCMOS Analog Circuits", *IEEE Transactions on CAD*, Nov. 1992.
- ▼ P.C. Maulik, L.R. Carley and R.A. Rutenbar, "Integer Programming Based Topology Selection of Cell-Level Analog Circuits", *IEEE Transactions on CAD*, April 1995.
- ▼ B. Antao and A. Brodersen, "ARCHGEN: Automated Synthesis of Analog Systems", *IEEE Transactions on VLSI Systems*, June 1995.
- ▼ W. Kruiskamp and D. Leenaerts, "DARWIN: CMOS Opamp Synthesis by Means of a Genetic Algorithm", *Proc. 32nd ACM/IEEE DAC*, pp. 433-438, 1995.
- ▼ G. Gielen, et al., "An analog module generator for mixed analog/digital ASIC design," *John Wiley International Journal of Circuit Theory and Applications*, Vol. 23, pp. 269-283, July-August 1995.
- ▼ J. Crols, S. Donnay, M. Steyaert and G. Gielen, "A high-level design and optimization tool for analog RF receiver front-ends," *Proc. IEEE/ACM ICCAD*, pp. 550-553, November 1995.

© R.A. Rutenbar 2001

Select References

■ Analog Synthesis, cont.

- ▼ F. Medeiro, F. Fernandez, R. Dominguez-Castro, A. Rodriguez-Vazquez, "A Statistical Optimization Based Approach for Automated Sizing of Analog Cells", *Proc. ACM/IEEE ICCAD*, 1994.
- ▼ E.S. Ochotta, R. A. Rutenbar and L.R. Carley, "Synthesis of High-Performance Analog Circuits in ASTRX/OBLX", *IEEE Transactions on CAD*, March 1996.
- ▼ M. Hershenson, S. Boyd, T. Lee, "GPCAD: a Tool for CMOS Op-Amp Synthesis", *Proc. ACM/IEEE ICCAD*, pp. 296-303, 1998.
- ▼ M. Krasnicki, R. Phelps, R. Rutenbar, L. R. Carley, "MAELSTROM: Efficient Simulation-Based Synthesis for Custom Analog Cells", *Proc ACM/IEEE DAC*, June 1999.
- ▼ R. Phelps, M. Krasnicki, R. Rutenbar, L. R. Carley, J. Hellums, "ANACONDA: Robust Synthesis of Analog Circuits Via Stochastic Pattern Search", *Proc. IEEE CICC*, May 1999.
- ▼ R. Phelps, M. Krasnicki, R. Rutenbar, L. R. Carley, J. Hellums, "A Case Study of Synthesis for Industrial-Scale Analog IP: Redesign of the Equalizer/Filter Frontend for an ADSL CODEC", *Proc. ACM/IEEE DAC*, June 2000.
- ▼ R. Phelps, M. Krasnicki, R. Rutenbar, L. R. Carley, J. Hellums, ANACONDA: Simulation-Based Synthesis of Analog Circuits Via Stochastic Pattern Search, *IEEE Transactions on CAD*, June 2000.
- ▼ T. Mukherjee, L.R. Carley and R.A. Rutenbar, "Efficient Handling of Operating Range and Manufacturing Line Variations in Analog Cell Synthesis," *IEEE Transactions on CAD*, Aug 2000.
- ▼ G. Debyser, G. Gielen, "Efficient analog circuit synthesis with simultaneous yield and robustness optimization", *Proc. IEEE/ACM ICCAD*, pp. 308-311, November 1998.
- ▼ C. De Ranter, et al., "CYCLONE: automated design and layout of RF LC-oscillators," *Proc. ACM/IEEE DAC*, June 2000.

© R.A. Rutenbar 2001

Select References

■ Symbolic Analysis

- ▼ G. Gielen, H. Walscharts, W. Sansen, "ISAAC: A Symbolic Simulator for Analog Integrated Circuits," *IEEE Journal of Solid-State Circuits*, Vol 24, No. 6, pp. 1587-1597, Dec 1989
- ▼ F. Fernandez, A Rodriguez-Vazquez, J. Huertas, "Interactive AC Modeling and Characterization of Analog Circuits Via Symbolic Analysis," *Kluwer Journal on Analog Integrated Circuits and Signal Processing*, Vol. 1, pp. 183-208, November 1991.
- ▼ J. Starzyk, A. Konczykowska, "Flowgraph Analysis of Large Electronic Networks," *IEEE Transactions on Circuits and Systems*, Vol. 33, No. 3, pp 302-315, March 1986.
- ▼ B. Li, D. Gu, "SSCNAP: A Program for Symbolic Analysis of Switched Capacitor Circuits," *IEEE Transactions on CAD*, Vol. 11, No. 3, pp. 334-340, March 1992.
- ▼ P. Wambacq, F. Fernandez, G. Gielen, W. Sansen, A. Rodriguez-Vazquez, "Efficient symbolic generation of approximated small-signal characteristics of analog integrated circuits," *IEEE Journal of Solid-State Circuits*, pp. 327-330, March 1995.
- ▼ Q. Yu and C. Sechen, "A Unified Approach to the Approximate Symbolic Analysis of Large Analog Integrated Circuits", *IEEE Trans. Circuits and Sys-I*, vol.43, pp 656-669, August 1996
- ▼ Q. Yu and C. Sechen, "Efficient Approximation of Symbolic Network Functions Using Matroid Intersection Algorithms," *IEEE Transactions on CAD*, vol. 16, no. 10. pp. 1073-1081, Oct. 1997.
- ▼ C. Shi, X. Tan, "Symbolic Analysis of Large Analog Circuits with Determinant Decision Diagrams," *Proc. ACM/IEEE ICCAD*, pp. 366-373, 1997.
- ▼ P. Wambacq, G. Gielen, P. Kinget, W. Sansen, "High-frequency distortion analysis of analog integrated circuits," *IEEE Trans. Circuits and Sys-I*, Vol. 46, No. 3, pp. 335-345, March 1999

© R.A. Rutenbar 2001

Select References

■ Analog Layout

- ▼ J. Rijmenants, J.B. Litsios, T.R. Schwarz, M.G.R. Degrauwe, "ILAC: An Automated Layout Tools for Analog CMOS Circuits," *IEEE Journal of Solid-State Circuits*, Vol. 24, No. 4, pp. 417-425, April 1989.
- ▼ J.M. Cohn, D.J. Garrod, R.A. Rutenbar, L.R. Carley, "KOAN/ANAGRAM II: New Tools for Device-Level Analog Placement and Routing," *IEEE Journal of Solid-State Circuits*, Vol. 26, No. 3, March 1991.
- ▼ U. Choudhury, A. Sangiovanni-Vincentelli, "Automatic Generation of Parasitic Constraints for Performance-Constrained Physical Design of Analog Circuits," *IEEE Transactions on CAD*, Vol. 12, No. 2, pp. 208-224, February 1993.
- ▼ E. Malavasi, E. Felt, E. Charbon and A. Sangiovanni-Vincentelli, "Automation of IC Layout with Analog Constraints," *IEEE Transactions on CAD*, vol. 15, no. 8, August 1996.
- ▼ K. Lampaert, G. Gielen, W. Sansen, "A Performance-Driven Placement Tool for Analog Integrated Circuits," *IEEE Journal of Solid-State Circuits*, Vol. 30, No. 7, pp. 773-780, July 1995.
- ▼ E. Malavasi, E. Felt, E. Charbon, A. Sangiovanni-Vincentelli, "Symbolic Compaction with Analog Constraints," *John Wiley International Journal on Circuit Theory and Applications*, Vol.23, No.4, pp. 433-452, July/Aug. 1995
- ▼ E. Malavasi, D. Pandini, "Optimum CMOS Stack Generation with Analog Constraints," *IEEE Transactions on CAD*, Vol. 14, No. 1, pp. 107-12, Jan. 1995.
- ▼ B. Basaran, R.A. Rutenbar, "An O(n) Algorithm for Optimum CMOS Device Stacking with Analog Constraints," *Proc. ACM/IEEE DAC*, June 1996.
- ▼ G. Van der Plas, J. Vandenbussche, G. Gielen, W. Sansen, "Mondriaan: a tool for automated layout synthesis of array-type analog blocks," *proceedings IEEE CICC*, pp. 485-488, May 1998.

© R.A. Rutenbar 2001

Select References

■ Analog Layout

- ▼ Mitra, R.A. Rutenbar, L.R. Carley, D.J. Allstot, "Substrate-Aware Mixed-Signal Macrocell Placement in WRIGHT," *IEEE Journal of Solid-State Circuits*, Vol. 30, No. 3, pp. 269-278, March 1995.
- ▼ S. Mitra, S. Nag, R.A. Rutenbar, and L.R. Carley, "System-Level Routing of Mixed-Signal ASICs in WREN," *Proc. ACM/IEEE ICCAD*, November 1992.
- ▼ B.R. Staniscic, N.K. Verghese, R.A. Rutenbar, L.R. Carley, D. J. Allstot, "Addressing Substrate Coupling in Mixed-Mode ICUs: Simulation and Power Distribution Synthesis," *IEEE Journal of Solid-State Circuits*, Vol. 29, No. 3, March 1994.
- ▼ G. Gielen et al, "A Layout Aware Synthesis Methodology for RF Circuits," to appear ACM/IEEE ICCAD, Nov 2001.