Designer's Guide Consulting

Analog, Mixed-Signal & RF Verification

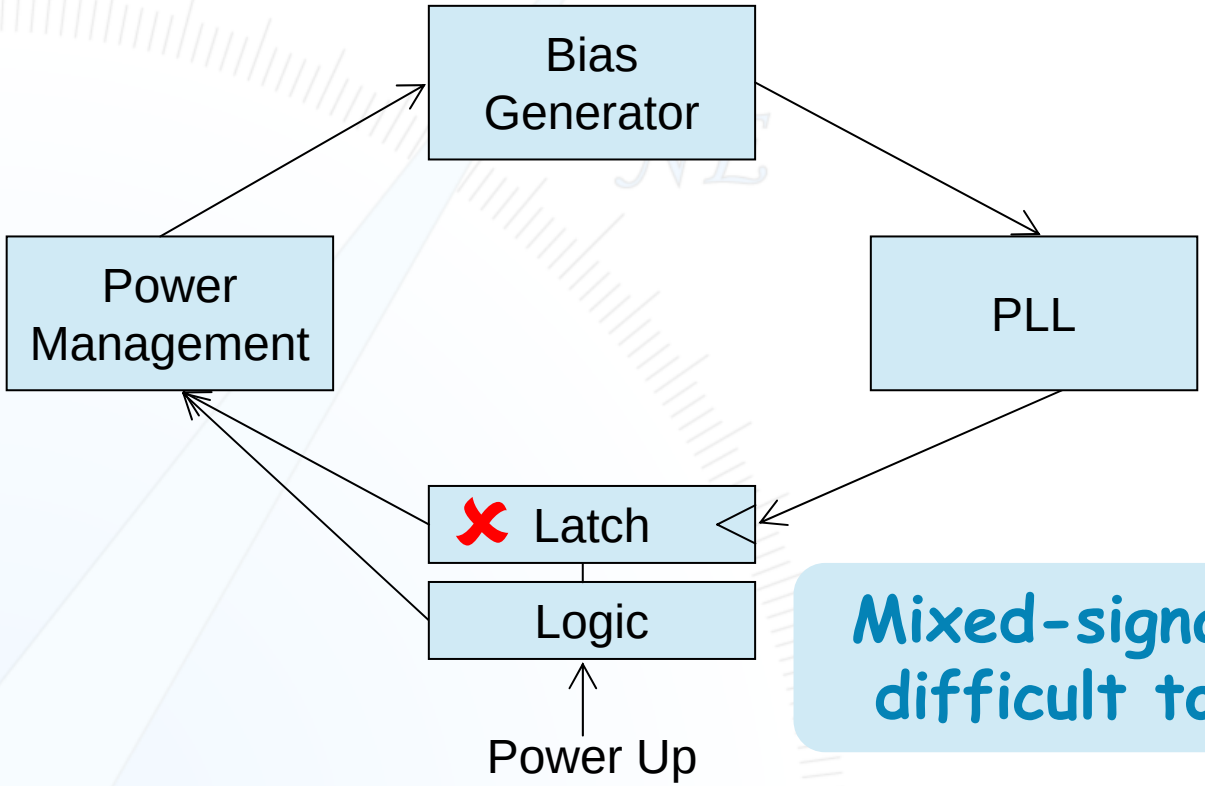
Beyond Analysis to Analog & Mixed-Signal Verification

BMAS Friday Keynote

Henry Chang

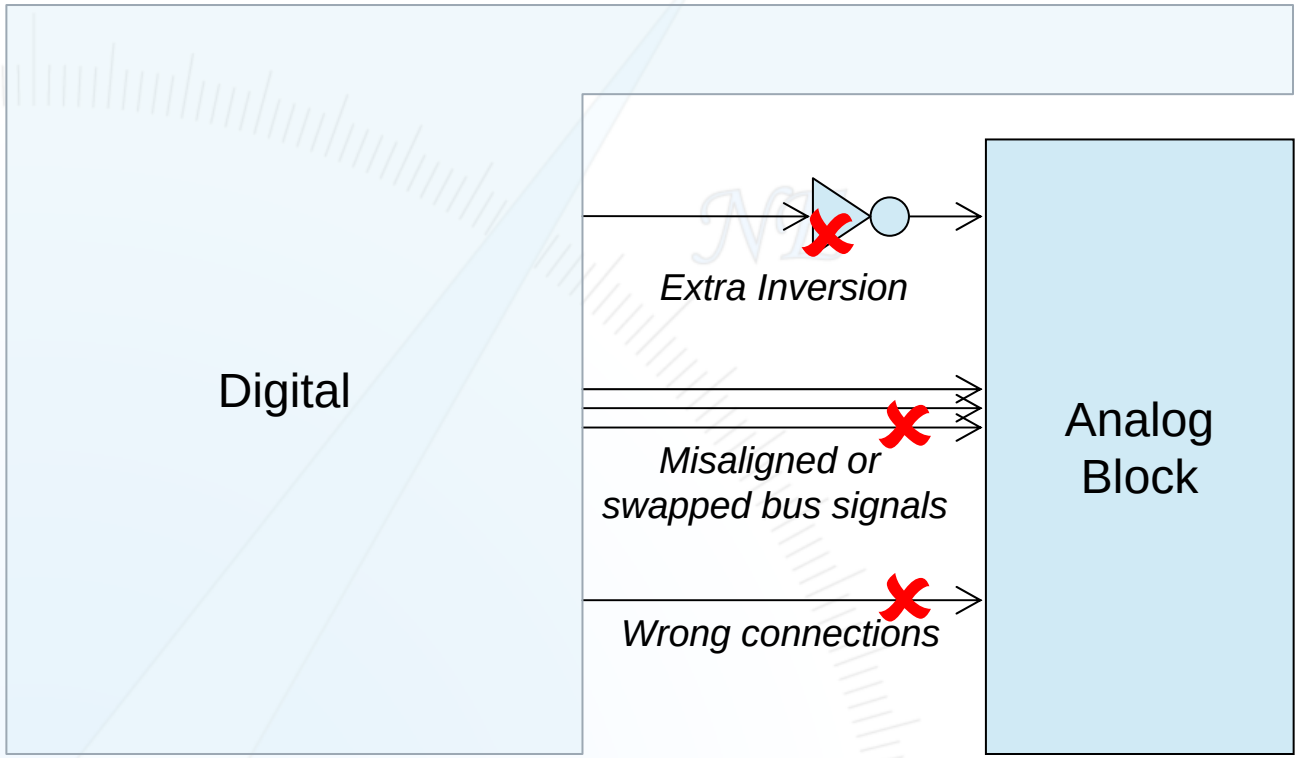
9/21/07

"Common" Fatal Errors: Chip Level



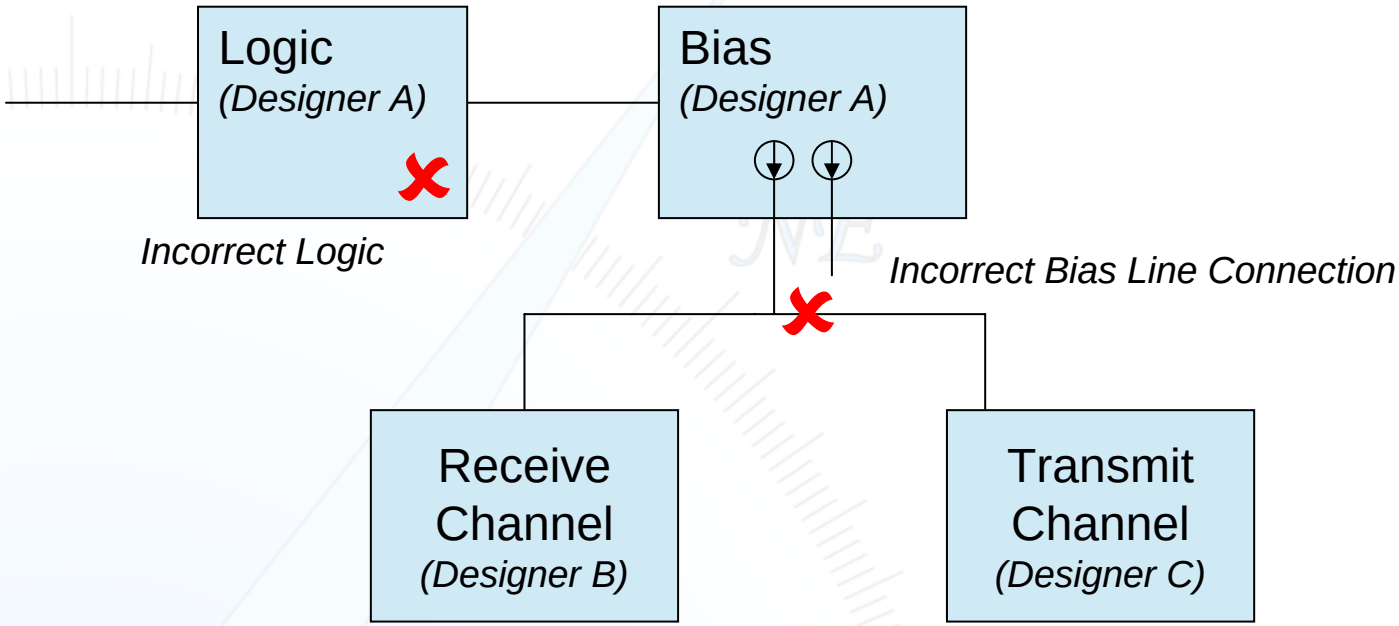
Mixed-signal errors difficult to catch

Interface Level



Typical analog block can have 150-500 I/O pins

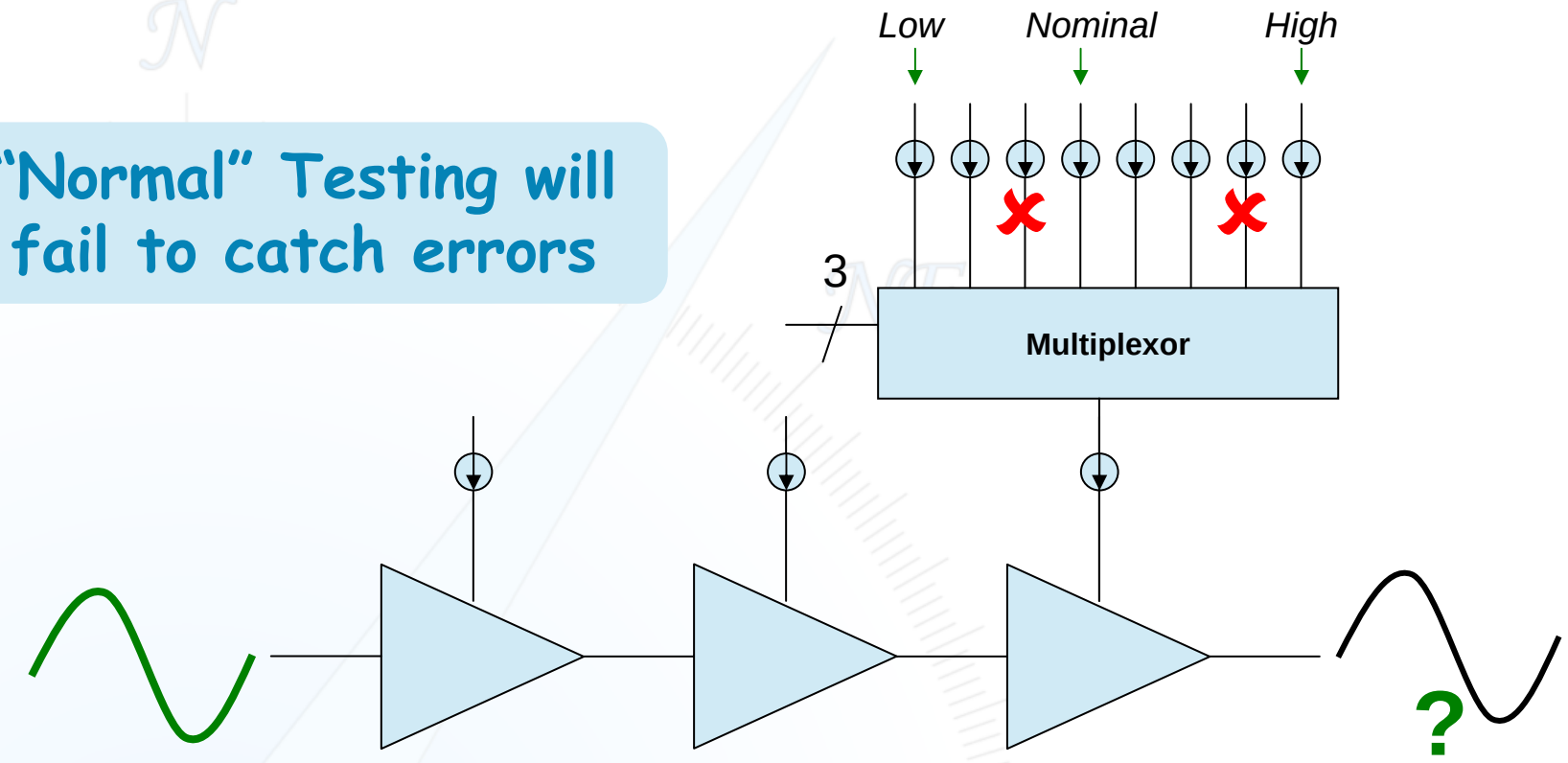
Block Internal



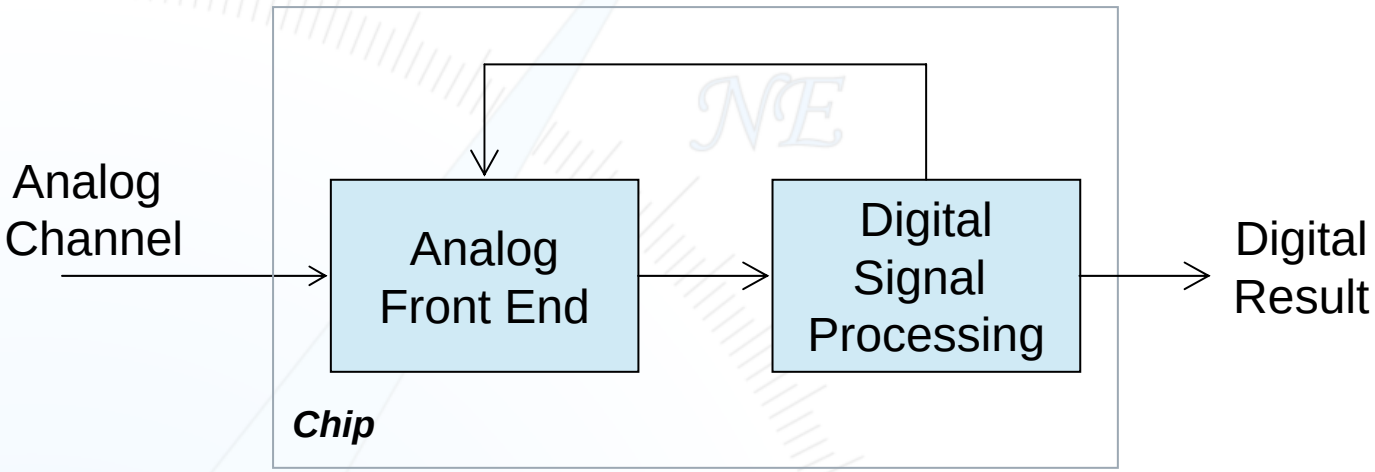
Fatal errors stalls development with economic consequences

Traditional Analysis Fails

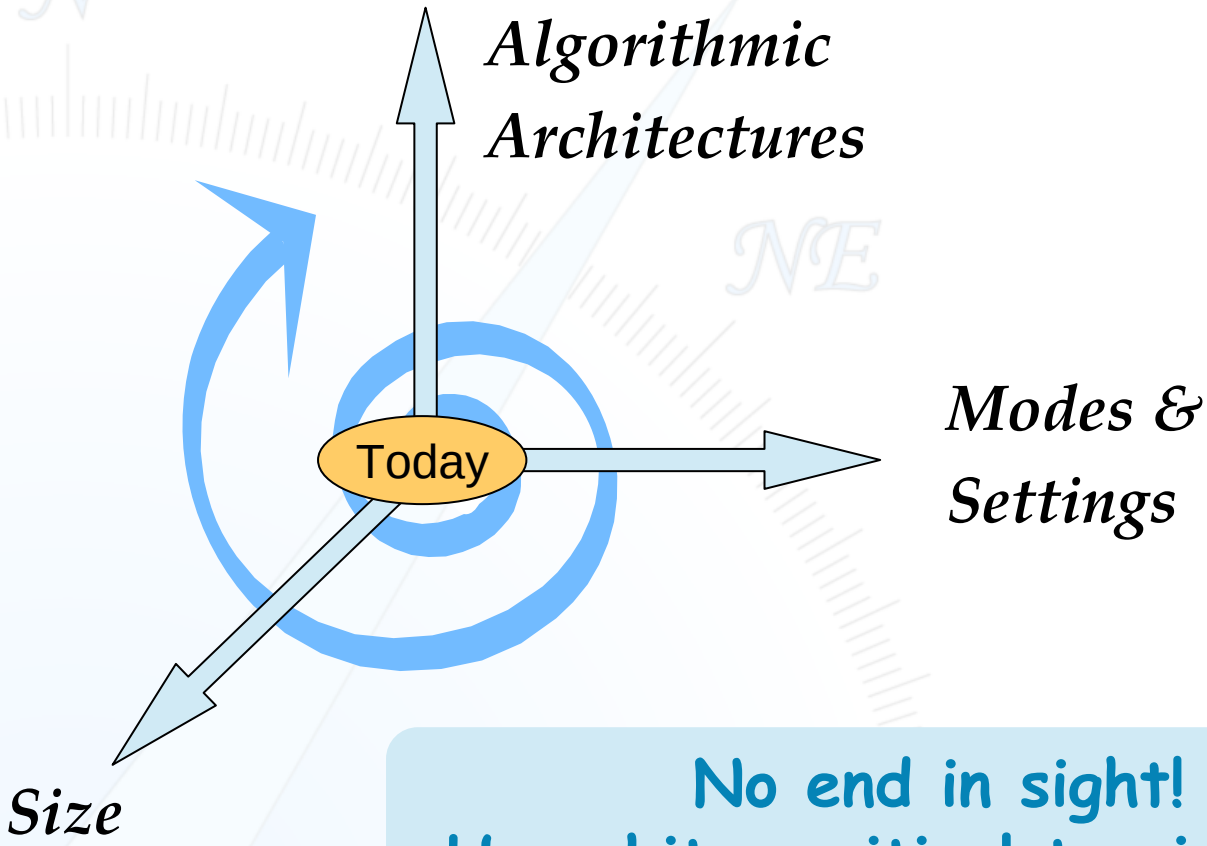
"Normal" Testing will fail to catch errors



System and Block Performance



Complexity Explosion



No end in sight!
Have hit a critical turning point

Circuit Simulation "Gap"

- Goal of overnight simulations
- Chip Level
  - 1 week for 2 clock cycles
  - 4 modes: Power-up, power down, sleep, wake
  - 1,000 clock cycles / mode tested
  - 4 x 1,000/2 x 7 x 2 = **28,000x** in simulation speed required
- Interface Level- RTL + analog
  - 30 x 8 bit busses (240 pins)
  - 1 day for analog block / mode

  ➔ 2^8 * 30 * 2 = **15,360x**

10,000x+ required

Gap ➔ Methodology

"It's the Methodology…"

Analog / Mixed-Signal Verification solves this today!

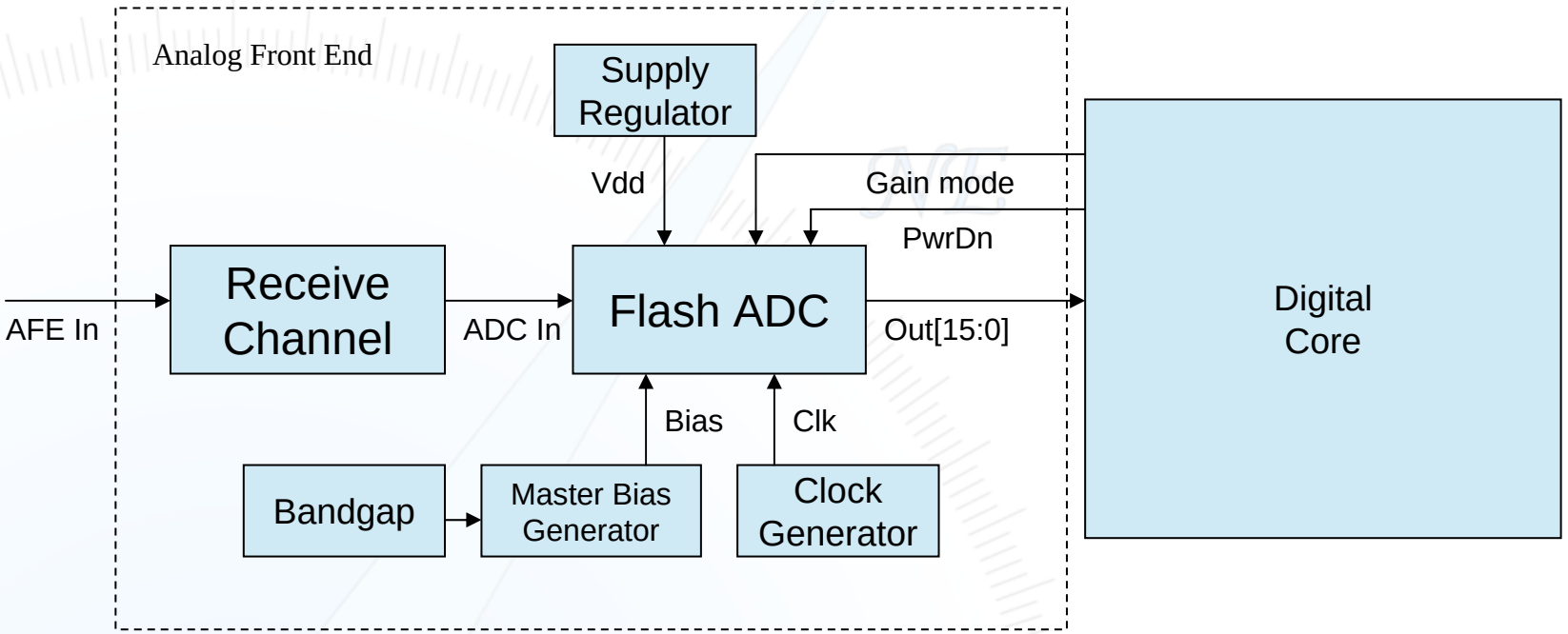
- Apply best known analog simulation techniques
- Apply/extend digital verification methodology
- Apply today's EDA tools
- Create analog verification engineer
- *Catch functional errors plus aid system and block performance validation… on a nightly basis*

Quick Start Guide

1. Analog verification & design lead begin together
2. Initial top-level schematic kicks off verification
3. Create verification, modeling, simulation plan
4. Write verification testbench & models using initial specifications
5. Exercise top-level schematic & models with testbench & regression scripts **Refine & Debug**
6. As block designers implement, run regression tests as blocks are completed **Refine & Debug**

More at http://www.designers-guide.com/

Example



http://www.designers-guide.com/newsletters/
(September, 2007)

Specifications

Name	Pin	Direction / Type	Description
Input	in	Input / Electrical	Analog Input
Power Down Control	pwrdn	Input / Logic	0: Normal Operation (500uA current consumption) 1: Off (1uA current consumption)
Gain Mode Control	gain_mode	Input / Logic	0 : 1.0x gain at input (0V <= analog input < 1V) 1 : 2.0x gain at input (0V <= analog input < 0.5V)
Clock	clk	Input / Logic	Output becomes available following the positive edge of the clock
Current Bias	bias	Input / Electrical	15 uA Input impedance: 20k Ohms
Supply Voltage	vdd	Input / Electrical	1.8V
Output	out[15:0]	Output / Logic	Thermometer coded output

Want to self check everything that's on this spec sheet

Regression Test*

Written by analog verification engineer

```
module flash_adc_regression ( );
 electrical vdd, in, bias;
 reg gain_mode, pwrdn, clk;
 wire [15:0] out;
 real in_value;
 flash_adc adc (.out (out), .in (in), …);
 analog begin
  V(vdd) <+ 1.8;
  I(bias) <+ 15u;
  V(in) <+ transition(in_value, …);
 end
 always #(1.0/50e6/2.0/`ticks) clk = ~clk;

<tasks>

 initial begin
   pwrdn = 'b0;
   clk = 'b0;
   gain_mode = 'b0;

   tests_adc ();
 end
endmodule
```

Testbench (DUT-transistors or model, input signals)

Circuit Setup

Run regression tests

```
task tests_adc;
   reg [15:0] expected;
   begin

     gain_mode = 'b0;
     for (i = 0; i < 16; i = i+1) begin
      in_value = (i+0.5)/16.0 ;
      for (j = 0; j < 16; j = j + 1)
        expected[j] = (j <= i) ? 'b1 : 'b0;
      @(negedge clk)
       if (out != expected)
         // $display error
     end

     gain_mode = 'b1;
     // repeat tests

     pwrdn = 'b1;
     // repeat tests
  endtask
```

Set mode

Test codes

Set other modes and repeat test

Lines omitted for illustration purposes (won't work as shown)

Model

Written by analog verification engineer

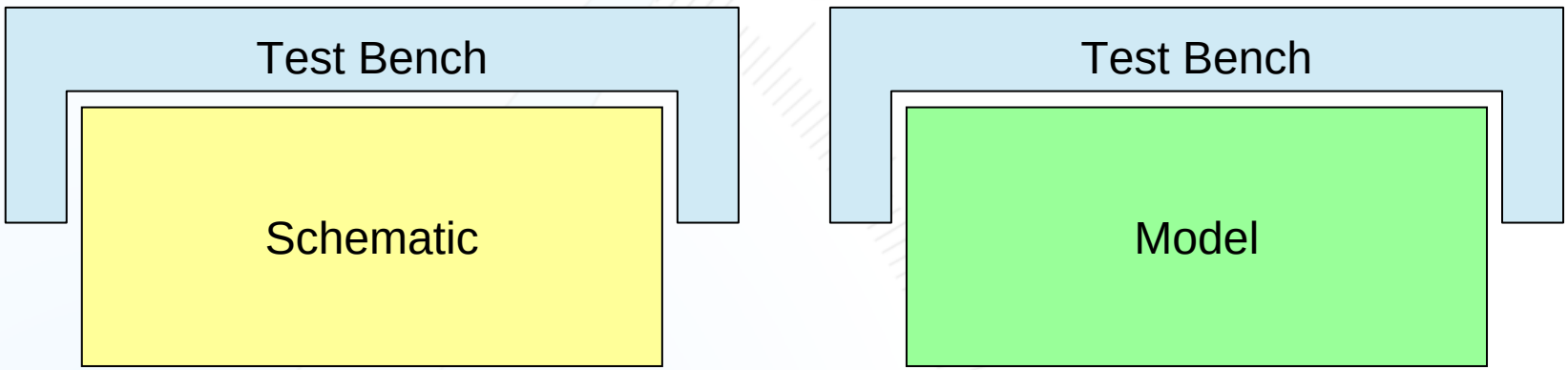
```
`include "constants.vams"
`include "disciplines.vams"

module flash_adc ( out, in, clk, bias, pwrdn, gain_mode, vdd );
    input in, clk, bias, pwrdn, gain_mode, vdd;
    output [15:0] out;
    electrical in, bias, vdd;
    integer i, level;
    real gain;
    reg pwrFault, biasFault;
    reg [15:0] d;
    always @(posedge clk) begin
        pwrFault = (V(vdd) > 1.9) || (V(vdd) < 1.7);
        biasFault = (I(vdd,bias) > 16u) || (I(vdd,bias) < 14u);
        gain = gain_mode == 'b1 ? 2.0 : 1.0;
        level = 16*V(in)*gain+0.5;  // convert input to an integer
        for (i=0; i<16; i=i+1)
            d[i] = (i < level);
    end
    assign out = (pwrdn || pwrFault || biasFault) ? 16'bx : d;
    analog begin
        V(vdd,bias) <+ pwrdn ? 0 : 0.5 + 20k*I(vdd,bias);
        I(vdd) <+ pwrdn ? 1u : 500u;
    end
endmodule
```

- Include header files
- I/O declarations
- Internal variables
- Assertions for power and bias
- Functional model
- Generate the output
- Model the input impedance
- Model power consumption

Model Verification

Apply the same tests to both



Run mixed-level simulations if transistor level simulations are too slow

Observed Result

- Nightly full check of entire transistor design

Specifications ✓ ≡ schematic ✓ ≡ Verilog model

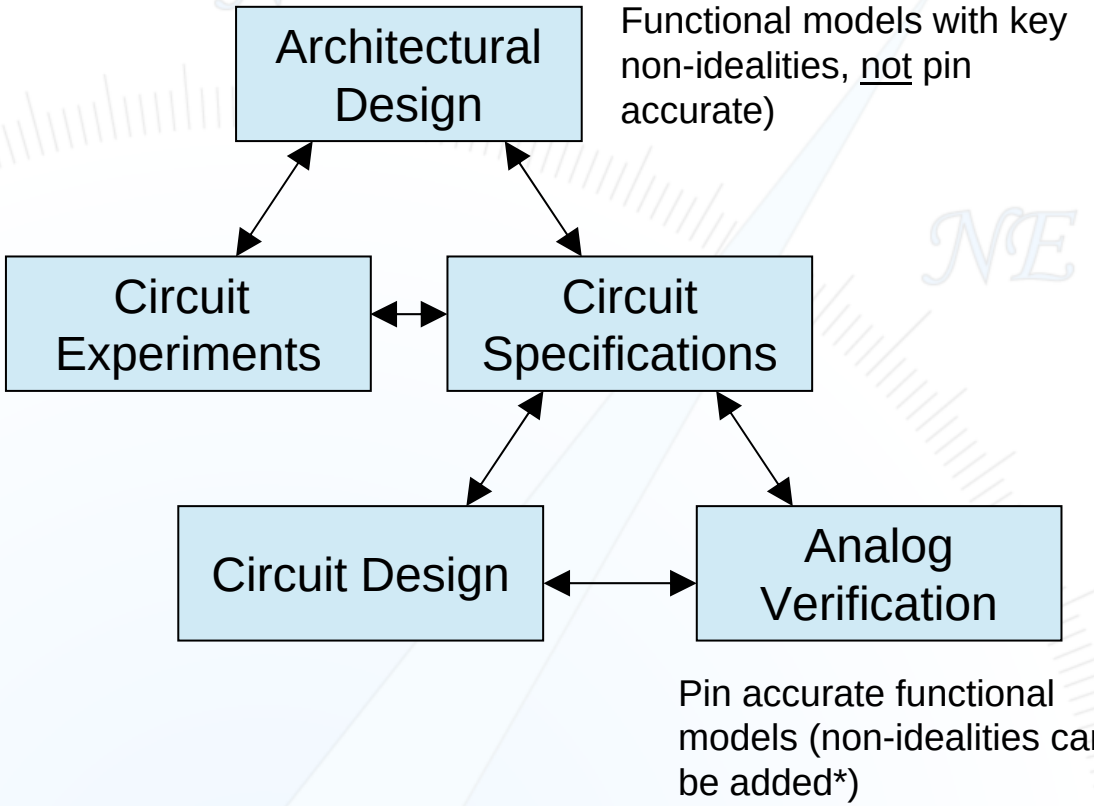
- Validated fully functional Verilog model for the system designer
- Improved system & block performance analyses
- More predictable and repeatable designs
- Control of risks enables greater innovation
- Aids reuse, test development
- Reduces hiring challenge

Performance

- Typical concerns:
  - System to Implementation gap
  - Simulations too slow to measure performance
  - Unknown interactions (e.g. substrate, supplies)
- Analog Verification provides scaffolding
- Verification plan captures how to address
- Realistic goal with methodology:
  - Functional correctness on first silicon, and performance on 2nd.

Not all performance issues can be found, but the scaffolding provided works surprisingly well

Performance



- *Bridge to "Matlab"*- Use pin-accurate analog models at architectural level with digital RTL
- *Speed up simulations*- Use pin-accurate models for mixed-level simulation during circuit design.
- *Analyze hard to predict effects*- Introduce effects to ideal functional models to understand possible effects

**But think through why you're doing this before following through*

False Beliefs

Only a theory	We've caught errors designers' missed on large mixed-signal designs. *not true*
No one doing this	Start-ups have adopted this! *not true*
Designers will resist	Designers find that they benefit! *not true*
Designers make no errors	VPs, design leads, project managers know better! *not true*
Requires more effort on the designer's part	Verification engineer actually reduces workload of overworked designers *not true*
Adds risk to the design	Verification shadows design project- little time required from designers and verification engineers have read-only access to design *not true*
Too much cost	Overall cost lower, starting requires ~20% *not true*
Simulators will be faster	10,000x? Not likely *not true*
Top-down design methodology required	Any design with a design lead and blocks being designed by different designers is sufficiently "top-down" *not true*
Verification can only be practiced by experts	• Not "rocket science", application of best practices • Template based modeling, testbench development • Does require some analog background and training *not true*

Stepping Stone to Top-Down Design

- Little disruption to existing design process
- Immediately gives most "top-down" benefits
  - Catch errors earlier
  - Enable re-use / capture design intent
  - Draft system model before design is done
    - Earlier test vector development
    - Earlier system level verification
  - Smooths out design process
  - Design process made more systematic and repeatable
- Provides foundation for next design to be more "top-down"

Verification gives benefits design groups seek from "top down design"

Analog Verification Will Happen

- When complexity reaches a certain point, a "verification", "QA", "regression" type methodology always gets instituted
  - Digital Design (10 years ago)
  - Software Development
  - …
- Cost of failure too high

3-5 Years From Now

- Analog verification will be "normal"
  - Managing complexity, providing predictable and sustainable development, manufacturing and deployment cycles
  - Analog verification outsourcing
- Analog EDA industry will move beyond simulators & design environments to provide support for verification
  - Verification environments, languages, etc.
- Better simulators
  - Fast & robust SPICE simulators use multi-CPUs
  - Improved analyses- RF applied to mixed-signal, FFTs with better dynamic range, etc.

Analog Verification Newsletter

- To receive our monthly newsletter on analog verification, please give me your business card or give me your name, affiliation and e-mail address.

Designer's Guide Consulting

Analog, Mixed-Signal & RF Verification

http://www.designers-guide.com/

References

- H. Chang & K. Kundert. "The Rise of Analog Verification." *The Proceedings of the IEEE, March 2007.*
- K. Kundert & H. Chang. "Verification of Complex Analog Integrated Circuits." *The Proceedings of the Custom Integrated Circuits Conference, 2006.*
- K. Kundert & O. Zinke. *The Designer's Guide to Verilog-AMS*. 2004.
- K. Kundert. "Principles of top-down mixed-signal design." *www.designers-guide.org/Design*.
- A. Meyer. *Principles of Functional Verification*. 2003.

Bio

Dr. Henry Chang is an expert in analog/mixed-signal design methodologies, next generation tools and flows, and standards for analog & mixed-signal IP in system-on-a-chip design.

Henry is currently at Designer's Guide Consulting which he co-founded in 2005. From 1995 to 2005, Henry worked at Cadence Design Systems, Inc. in research and development, methodology services, product marketing, corporate strategy, and in the office of the Chief Technology Officer. He has also worked at Micro Linear and GE Lighting. He is the author of three books: Winning the SoC Revolution: Experiences in Real Design in 2003, Surviving the SoC Revolution: A Guide to Platform Based Design in 1999, and A Top-Down, Constraint-Driven Design Methodology for Analog Integrated Circuits in 1997. He is on the steering committee of the IEEE Custom Integrated Circuits Conference. He holds 11 US patents and has authored many technical papers.

Henry received his Ph.D. and M.S. in Electrical Engineering from the University of California at Berkeley in 1994 and 1992 respectively. He received his Sc.B. degree in Electrical Engineering from Brown University in 1989.