2013 AUG 13 TAIPEI Graser User Conference

Mixed-Signal Design for Analog-Centric & Digital-Centric Verification

Stacy Chen / Paine Chuang

13/Aug/2013

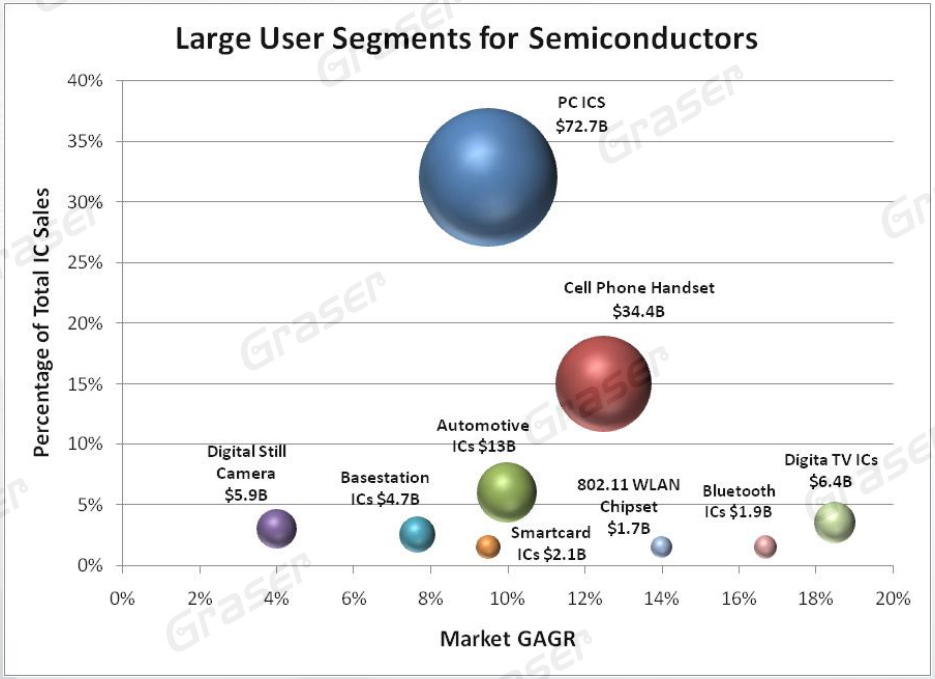
Topic

- Mixed-Signal Verification Challenges
- Overview of Mixed-Signal Solution
- Mixed-Signal Verification
  - Analog-Centric Verification
  - Digital-Centric Verification
- Summary

Mixed-Signal Verification Challenges

Market Trends

- High growth applications generating demand for digital and analog circuitry in turn fueling mixed-signal growth
- Over 80% of these designs are mixed-signal
  - Real-world data requires analog and digital processing
  - Operating at Ghz+ requires high speed interfaces
  - Integrated systems require analog/RF components mixed with digital
  - Low power requirements driving integration



Source: IC Insights

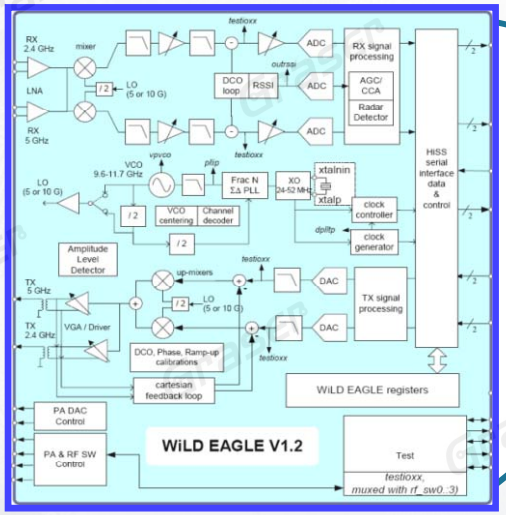
MS Verification Challenges

Mixed-signal SoC Complexity

How do I Test such a complex mixed signal SoC?

How do I verify the digital content in this SoC?

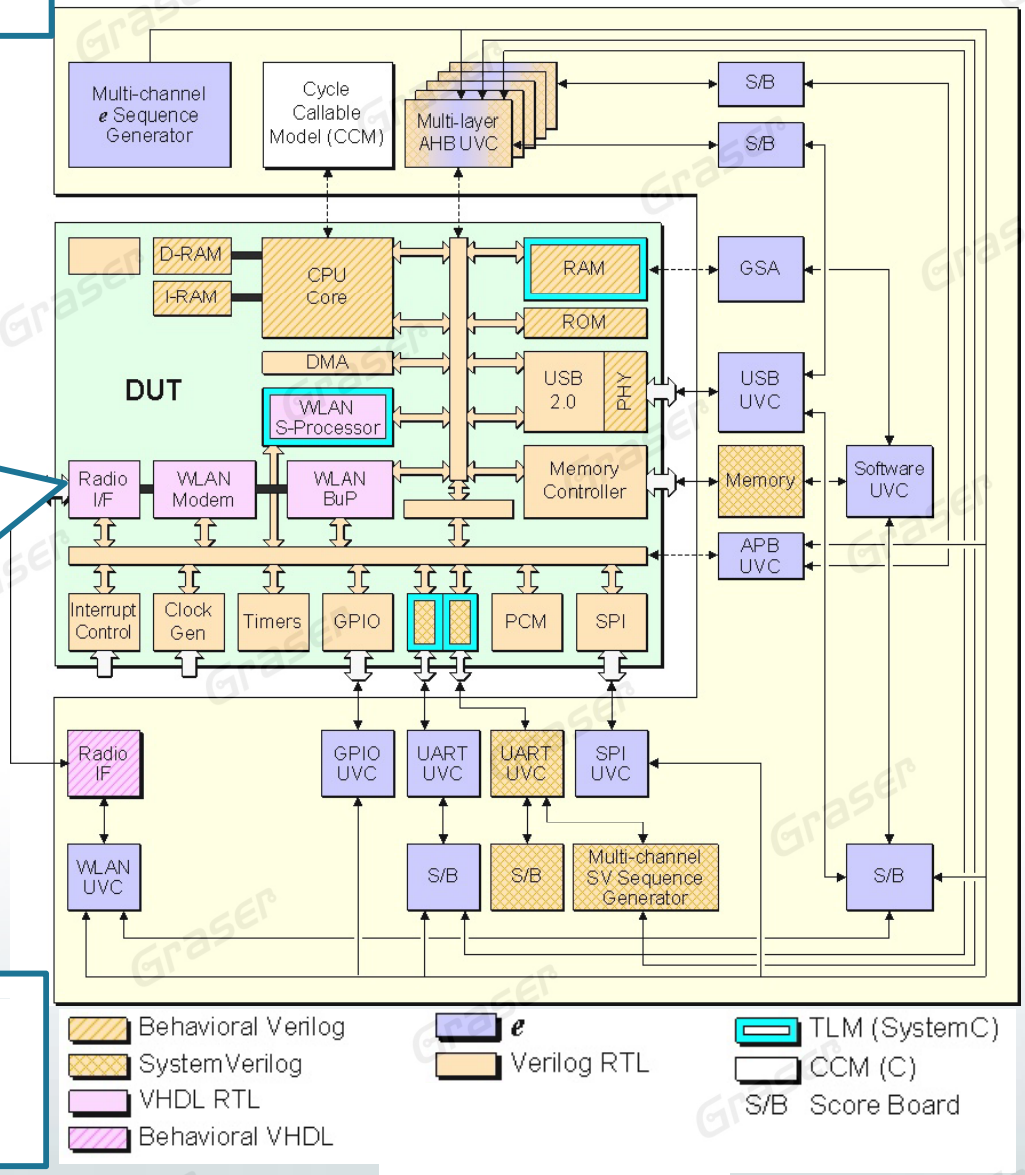
How do I verify the mixed-signal interconnects?



How do I balance tradeoffs between accuracy vs. performance simulations?

How do I know when my MS SoC is fully verified?

Digital mixed-signal



MS Verification Challenges

Why is SoC Level MS Verification so difficult?

- Mixed Signal simulation issues:
  - CoSim: Analog domain is continuous and Digital is event driven
  - MS/Analog simulation performance limitations
  - Speed vs. Accuracy tradeoffs
  - Analog coverage – need visibility into Analog space to gauge quality of verification
  - Pin connectivity errors
  - Analog and cross domain low power issues
  - Complex debugging

- MS Modeling issues:
  - Verification environment must support different levels of abstraction (Spice / Verilog-AMS / Real-Number Models)
  - Model Creation: fidelity of model – how much accuracy is really needed vs. how much accuracy is possible?
  - Model Validation: how to establish models equivalence at different levels of abstraction

MS Verification Challenges

The usual dilemma …

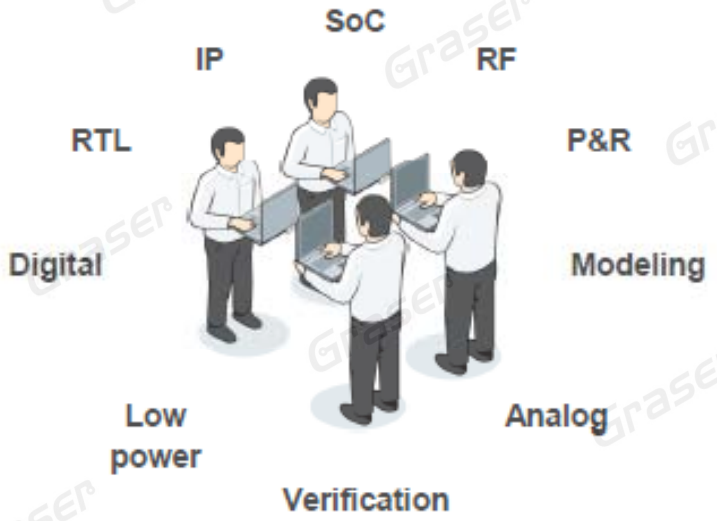
- I need my simulation results as soon as possible …
- I need maximum accuracy …
- I need a full chip simulation …

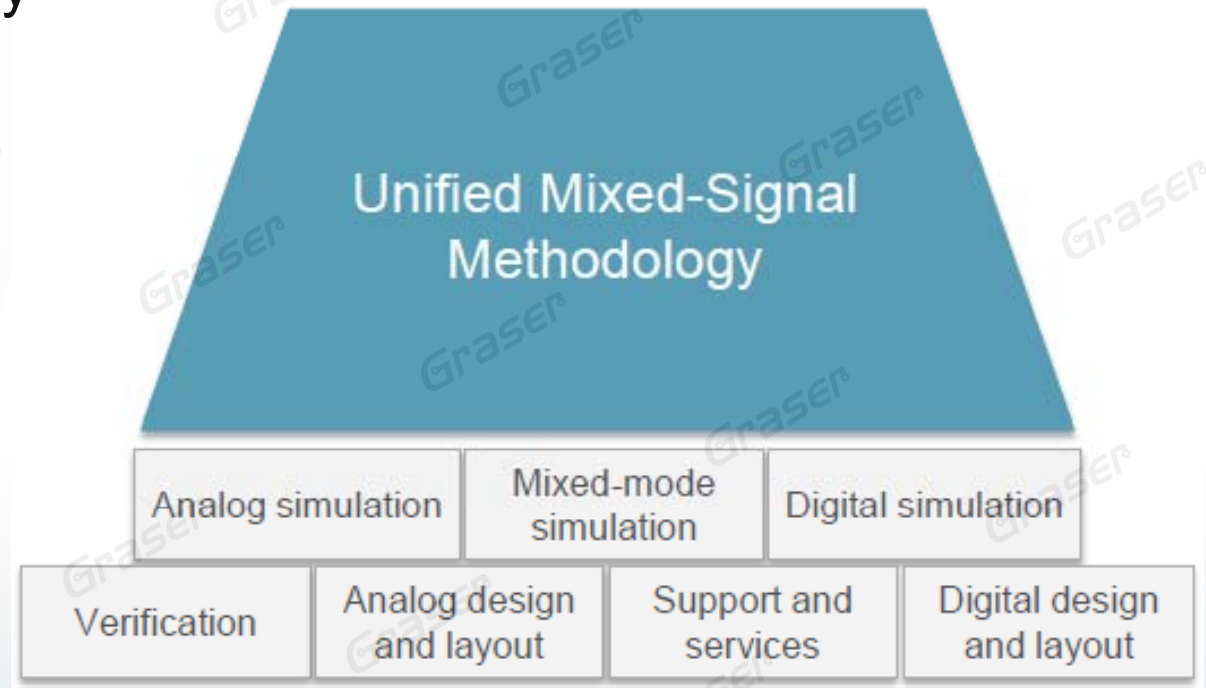
- Started a spice solver on full chip transistor level netlist
  - still waiting for the simulation results
- Finished a pure digital simulation
  - but don't see analog effects
- Ran a small block on spice
  - how do I know that it works correctly in the bigger context?

Overview of Mixed-Signal Solution

Mixed-Signal Solution

- Benefits
  - Design quality, area
  - Productivity, TAT
  - Fewer iterations/re-spins
- Unified design methodology
  - Addresses design challenges
  - Scalable for complexity
- Foundation
  - Products and technology
  - Support and services
  - IP and ecosystem

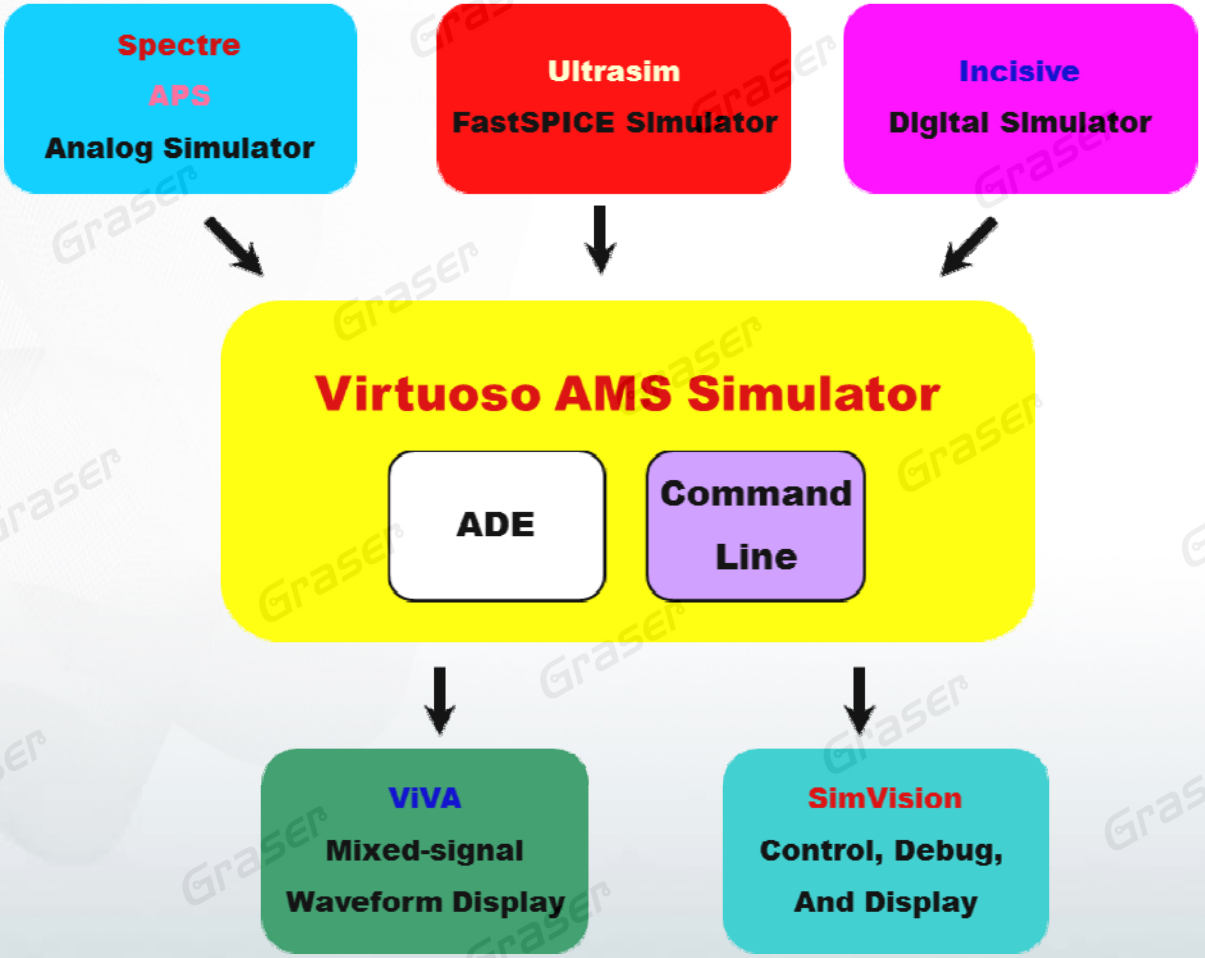




Mixed-Signal Solution

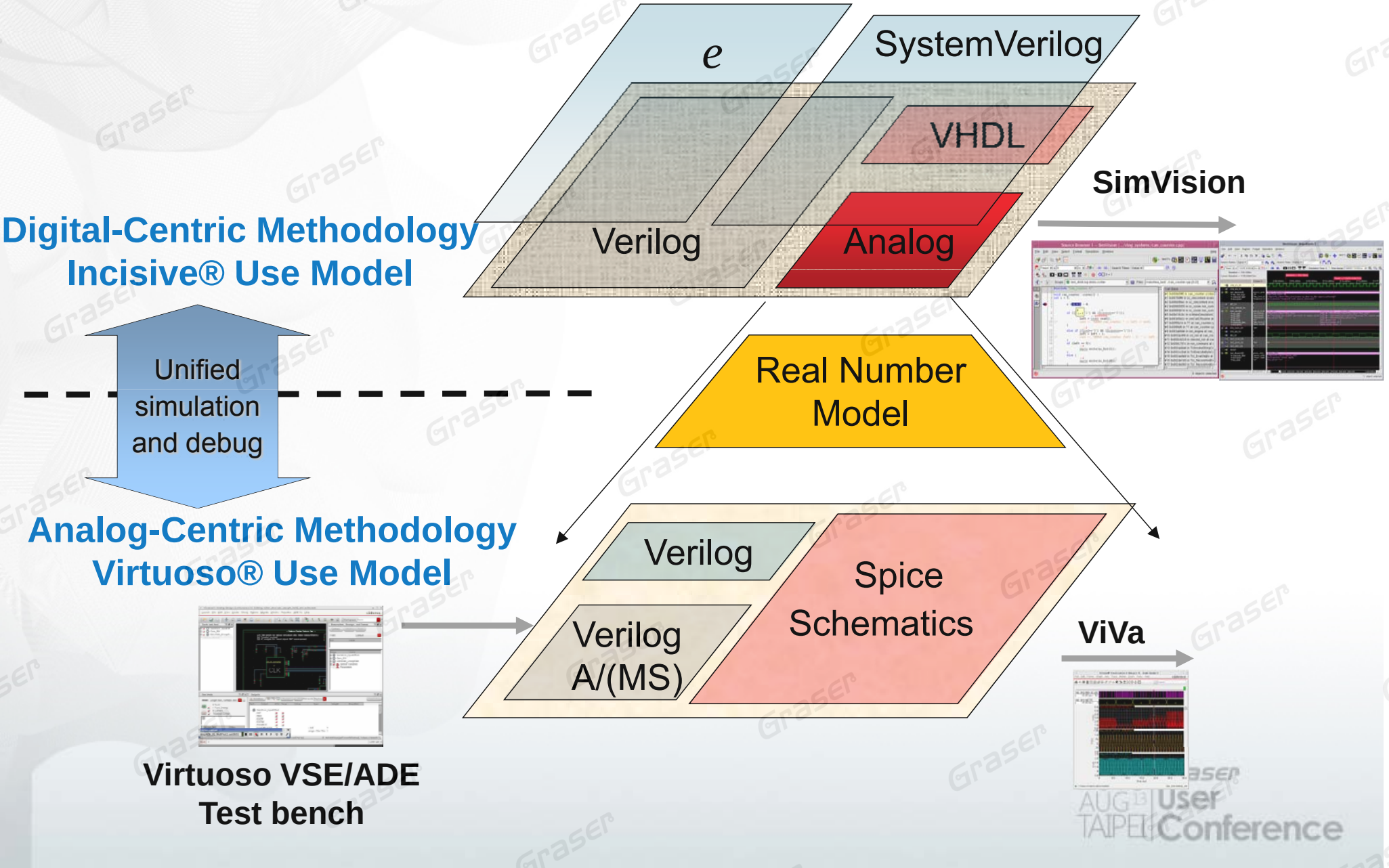
Virtuoso AMS Designer Verification

- **Virtuoso AMS Designer** is a single executable mixed-signal simulator based on the proven technology of *Virtuoso Spectre, Virtuoso Accelerated Parallel Simulator, UltraSim* Full-Chip Simulator, and the Incisive digital simulation capabilities.



Mixed-Signal Solution

Virtuoso AMS Designer Verification



Mixed-Signal Solution

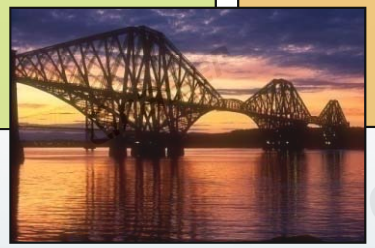
Virtuoso AMS Designer Verification

Analog Centric Verification

- Target: AMS IP creation
- Analog Design Engineer
- Analog (and analog-centric MS) Methodology
  - ❑ Schematic driven, GUI based
  - ❑ Transistor level simulation
    - ❑ Functional
    - ❑ DC, AC, Transient
  - ❑ Corner Analysis
  - ❑ Monte Carlo Analysis
  - ❑ Performance
    - ❑ AC, RF, Noise

Digital Centric Verification

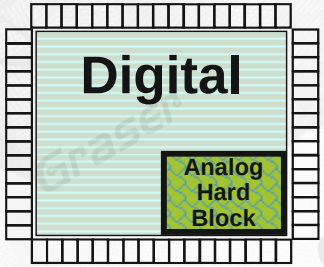
- Target: MS SoC Verification
- Verification Engineer
- Digital MDV Methodology
  - ❑ Command-line driven
  - ❑ Directed random testbenches
  - ❑ Assertion-based Verification
  - ❑ Metrics and Coverage Driven
  - ❑ Low Power – CPF
  - ❑ Verification Management
  - ❑ HW/SW Verification



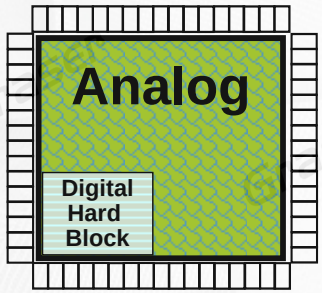
Comprehensive Verification Methodology bridging the gap

Mixed-Signal Solution

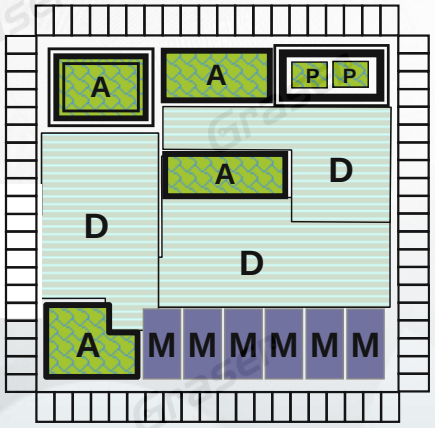
Virtuoso AMS Designer Verification



- Digital-centric Mixed-signal Verification User
  - Mainly application specific digital blocks designed with standard cell methodology
  - Analog blocks support specific function protocol Integration through hard analog IP import



- Analog-centric Mixed-signal Verification User
  - Dominant analog, custom-digital and RF blocks developed using custom methodology
  - Digital blocks for control, calibration & connectivity
  - Integration through hard digital IP import

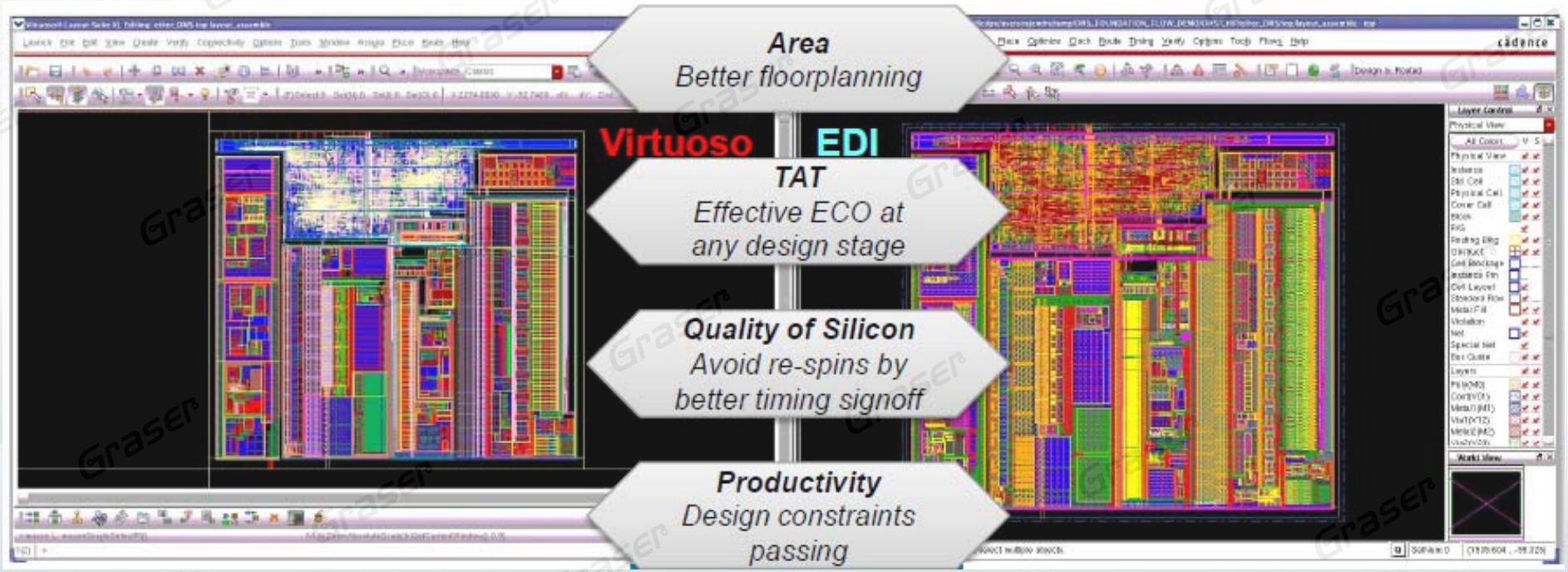


- Mixed-Signal SoC Verification User
  - Full-chip SoC verification
  - Iterative verification done by digital group
  - High volume digital-centric nightly regressions tests
  - Concurrent analog & digital block design flows
  - Soft IP import and integration done by digital group

Mixed-Signal Solution

Unified Environment for MS Implementation

- Unified Design DataBase _OpenAccess
  - No Data translation for Productivity
    - Floorplan
    - Placement
    - Routing
- Unified Library and Technology setup
  - Techfile
  - Standard cell library



Analog-Centric Verification

AMS Designer Verification in Virtuoso

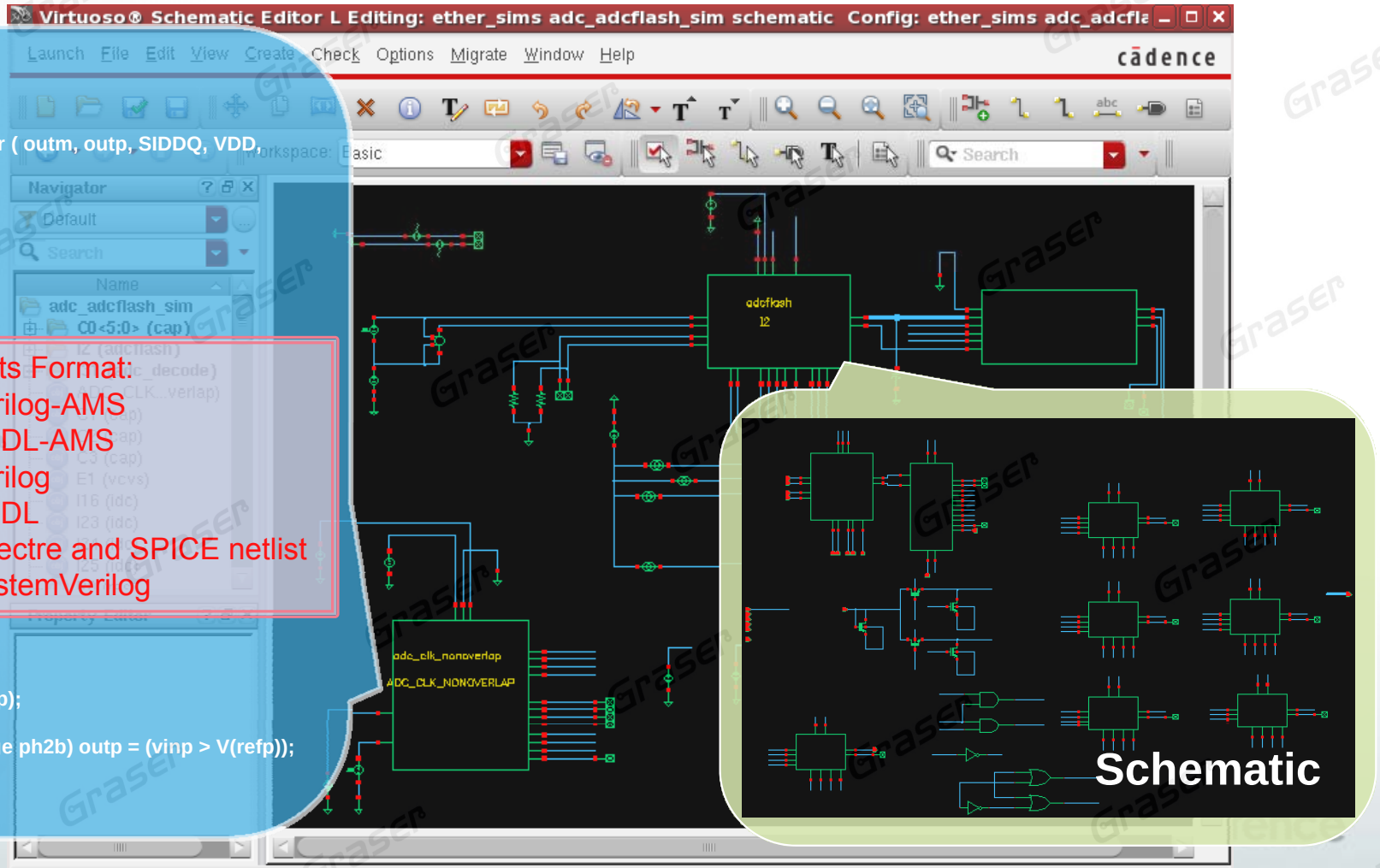
AMS Designer Verification in Virtuoso

Virtuoso Schematic Editor

- Hierarchical schematic entry and hierarchy editor supports
- Analog, digital, mixed-signal, and RF symbol libraries are supported

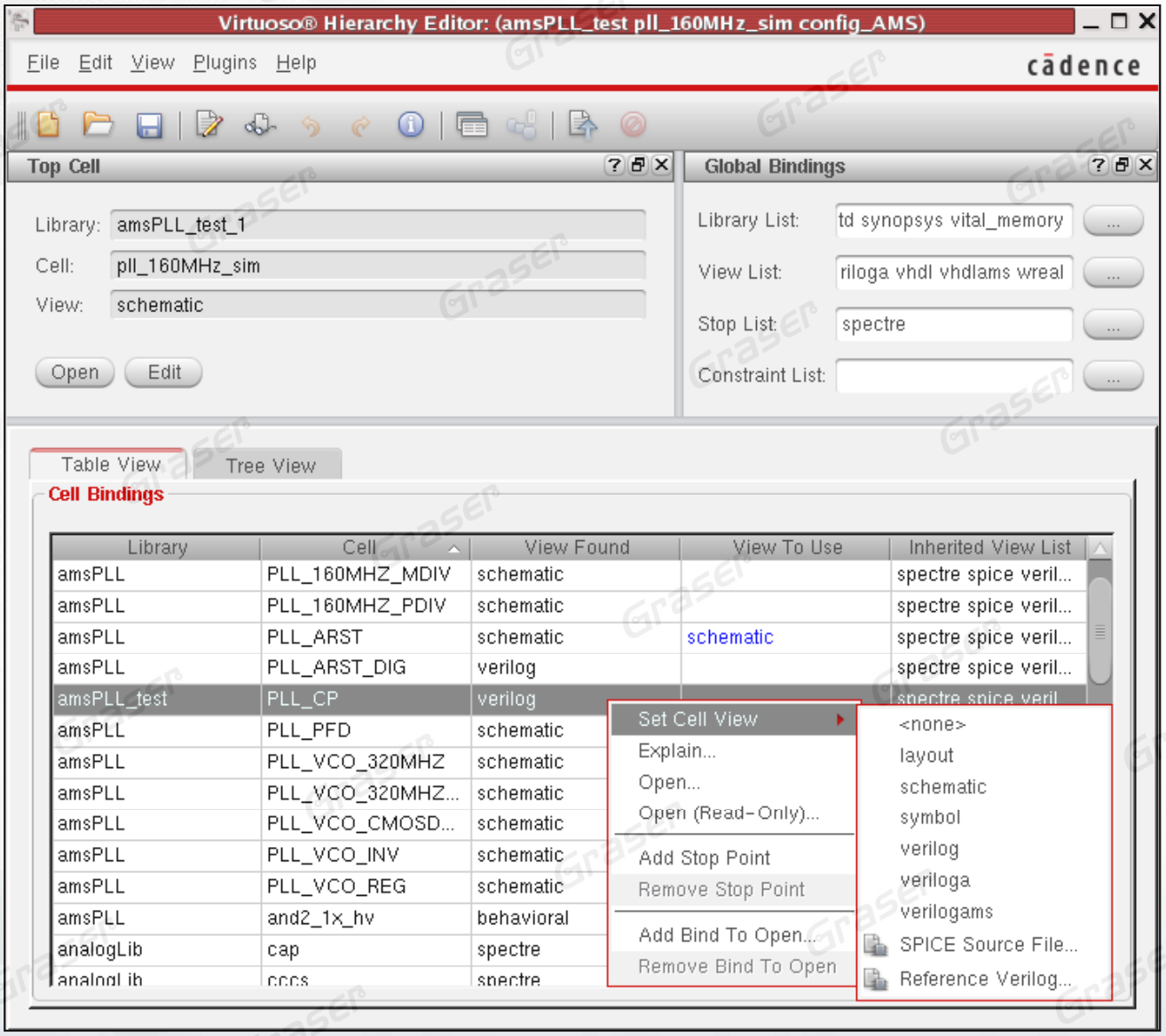
```
`include "constants.vams"
`include "disciplines.vams"

module adcflash_comparator_actr ( outm, outp, SIDDQ, VDD,
VSS, inn, inp, ph1,
   ph2b, refn, refp, vbias );

output  outp, outm;
input   inp, inn;
input   ph1, ph2b;
input   refp, refn;
input   VDD, VSS, vbias;
input   SIDDQ;

logic       outp, outm;
logic       ph1, ph2b;
logic       SIDDQ;
electrical    inp, inn;
electrical    refp, refn;
electrical    VDD, VSS;
electrical    vbias;

real vinp, vinm;
reg  outp;
wire outm = ~outp;

analog  vinp = (SIDDQ) ? 0.0: V(inp);
initial outp = 1'b0;
always @(posedge ph1 or posedge ph2b) outp = (vinp > V(refp));

endmodule
```

Inputs Format:
- Verilog-AMS
- VHDL-AMS
- Verilog
- VHDL
- Spectre and SPICE netlist
- SystemVerilog



AMS Designer Verification in Virtuoso

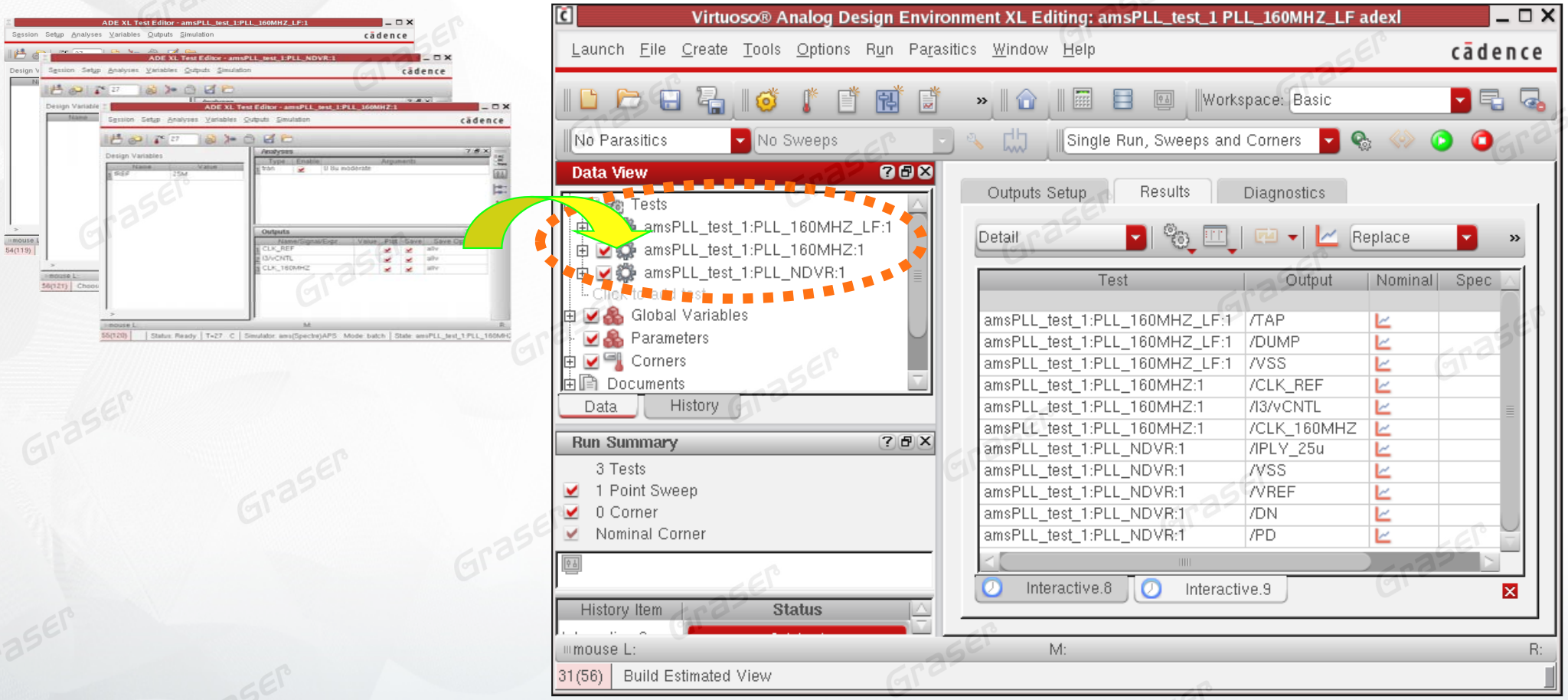
CoSim of digital configuration input and analog behavior



AMS Designer Verification in Virtuoso

AMS Designer Multiple Tests

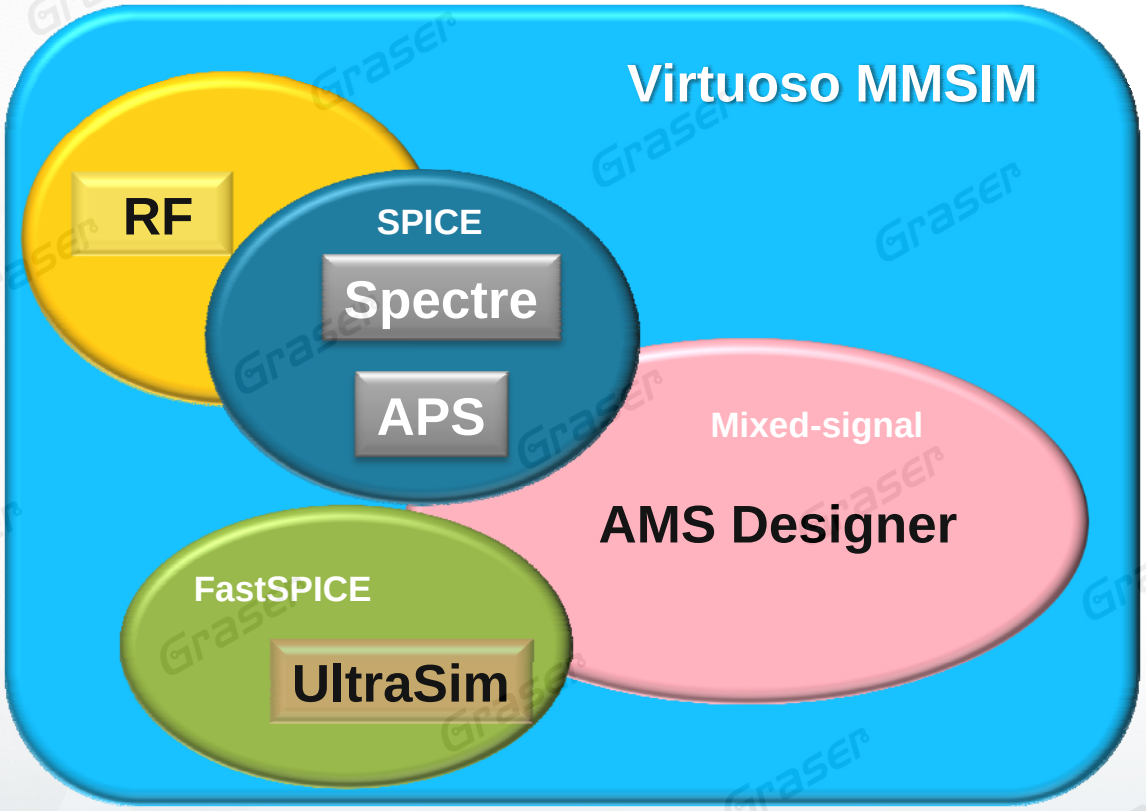
- Can set simulations running in parallel or series in ADE



AMS Designer Verification in Virtuoso

Virtuoso AMS Designer

- Mixed Signal Design Verification
- High performance and accuracy



AMS Verification in Virtuoso

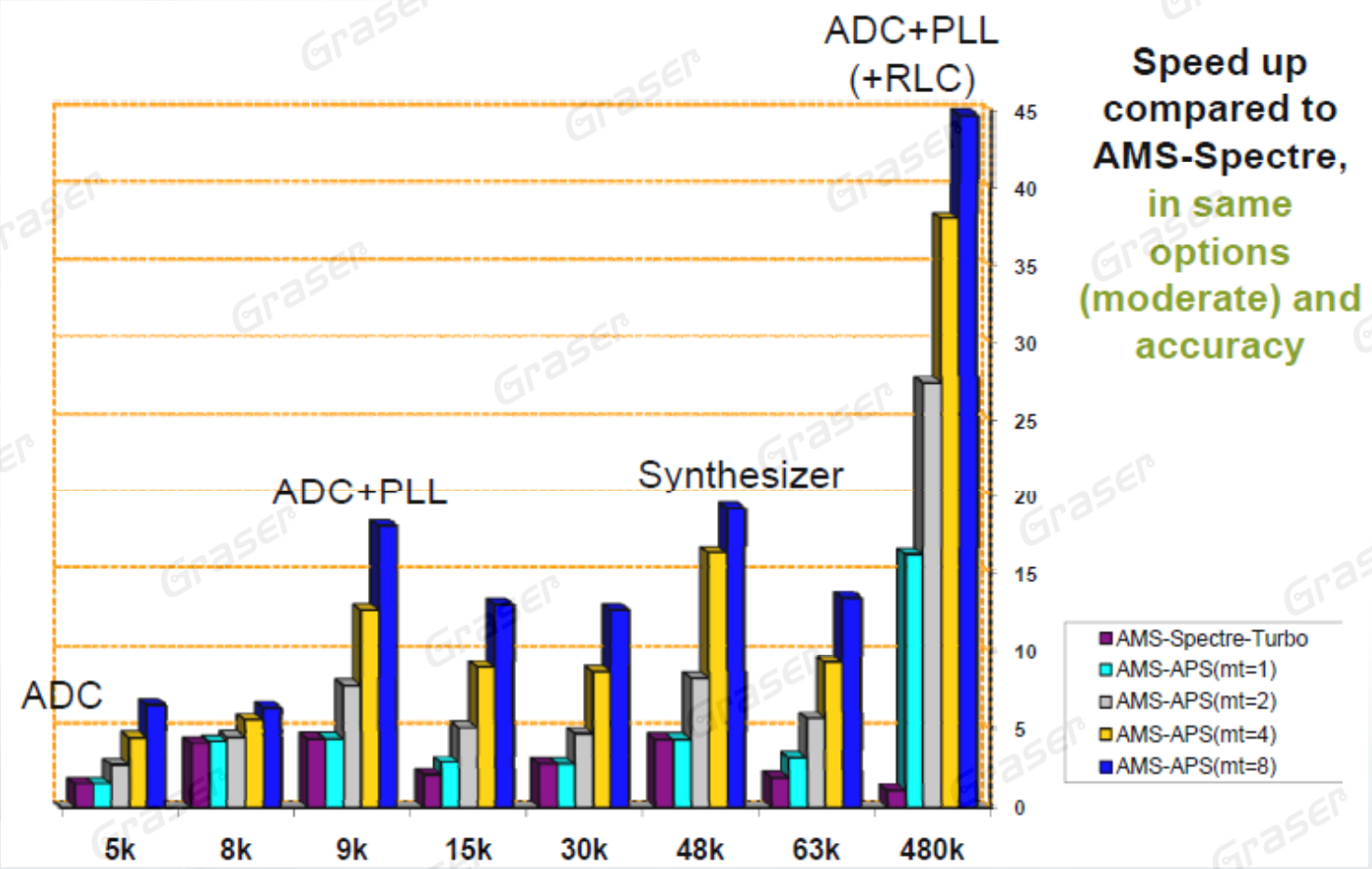
AMS Designer-APS Key Features

- APS Integration with AMS Designer
- Full Spectre accuracy
- Fully compatible with Spectre Solver
- Improved capacity compared to AMS-Spectre
- Maximum simulation performance with multi-threading support on multi-core/multi-CPU systems
- Simplified use model for command line (irun/ncsim) and ADE usage

AMS Verification in Virtuoso

Virtuoso AMS Designer-APS

- Performance and Capacity improvement
  - More performance gain with larger device count

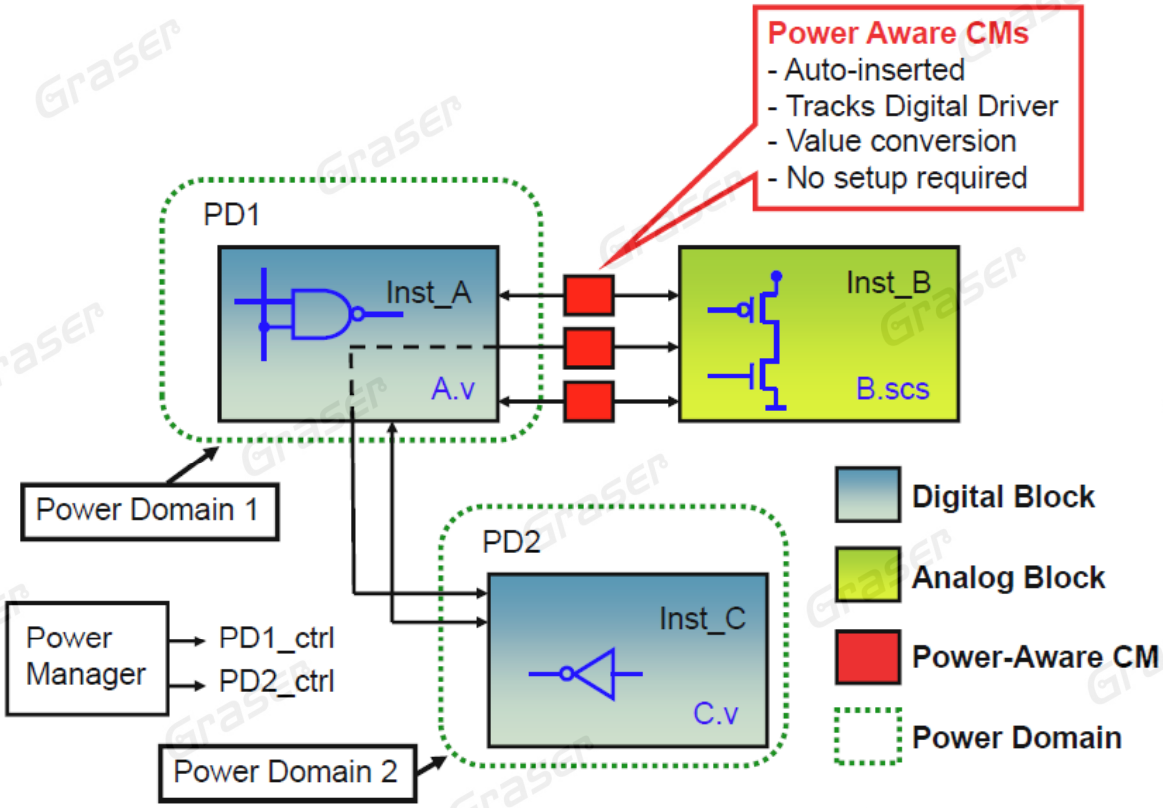


Speed up compared to AMS-Spectre, in same options (moderate) and accuracy

AMS Verification in Virtuoso

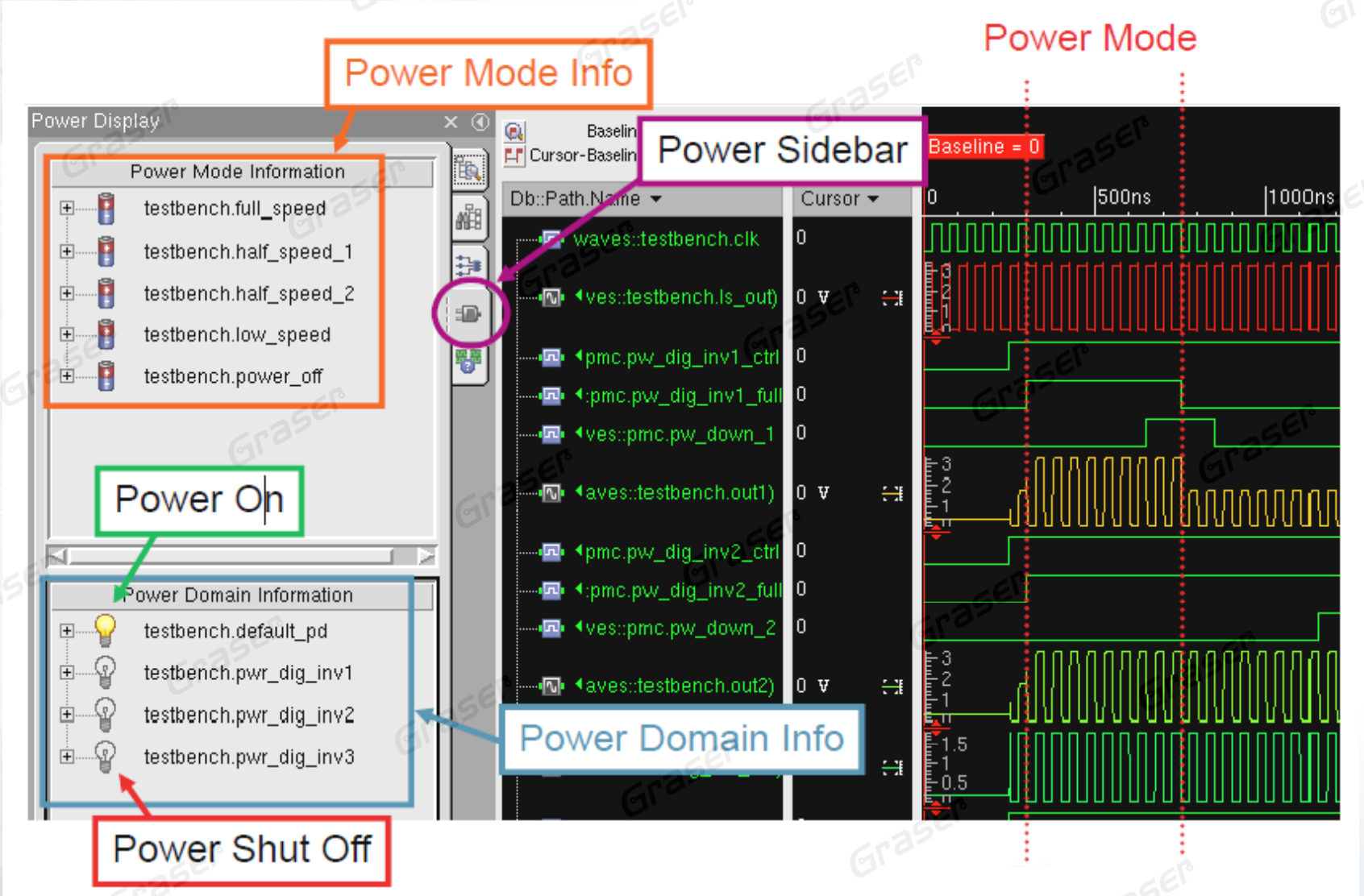
Mixed-Signal Low Power support

- Support for multiple-voltage power domains
  - Power Smart connect module carries the effect of power shut-off, power active state conditions, power modes and transitions onto analog blocks
- Simulator identifies the CPF influence on analog blocks
  - Automatically Inserts “Power-Smart” Connect modules
  - Carries the effect onto analog blocks



AMS Designer Verification in Virtuoso

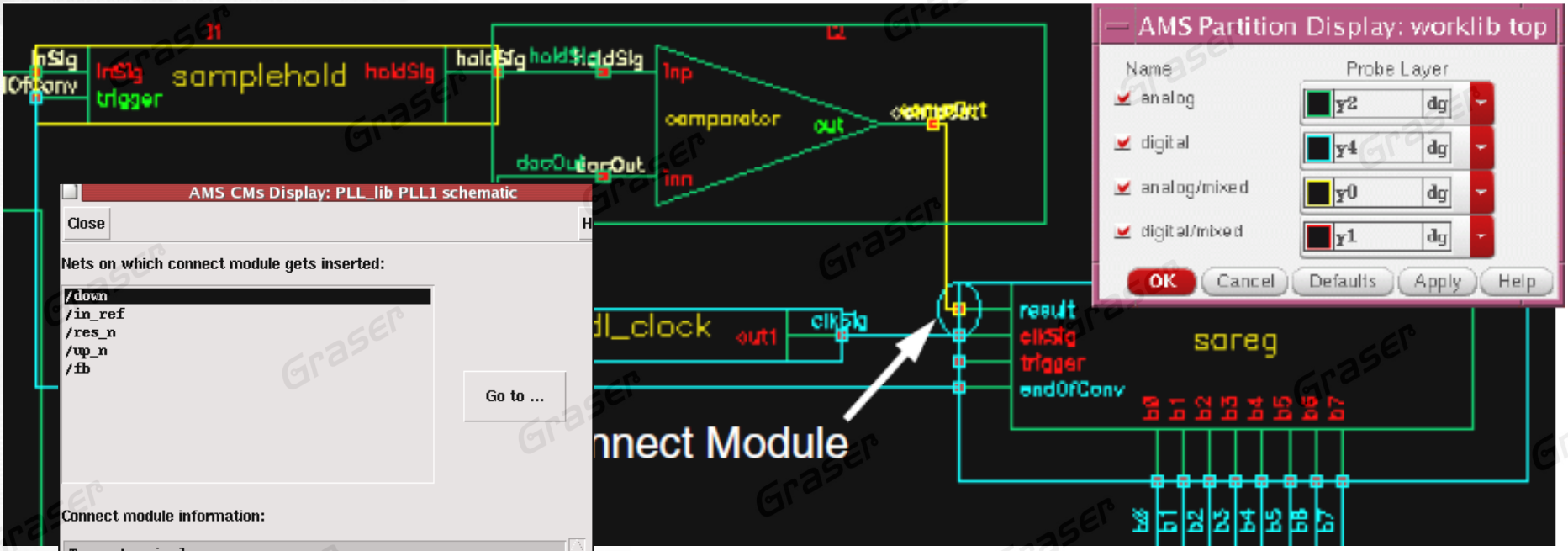
Low Power Info in SimVision



AMS Designer Verification in Virtuoso

Display domains and connect modules

- Show Info on the schematic after simulation



AMS CMs Display: PLL_lib PLL1 schematic

Close

Nets on which connect module gets inserted:

/down
/in_ref
/res_n
/up_n
/fb

Go to ...

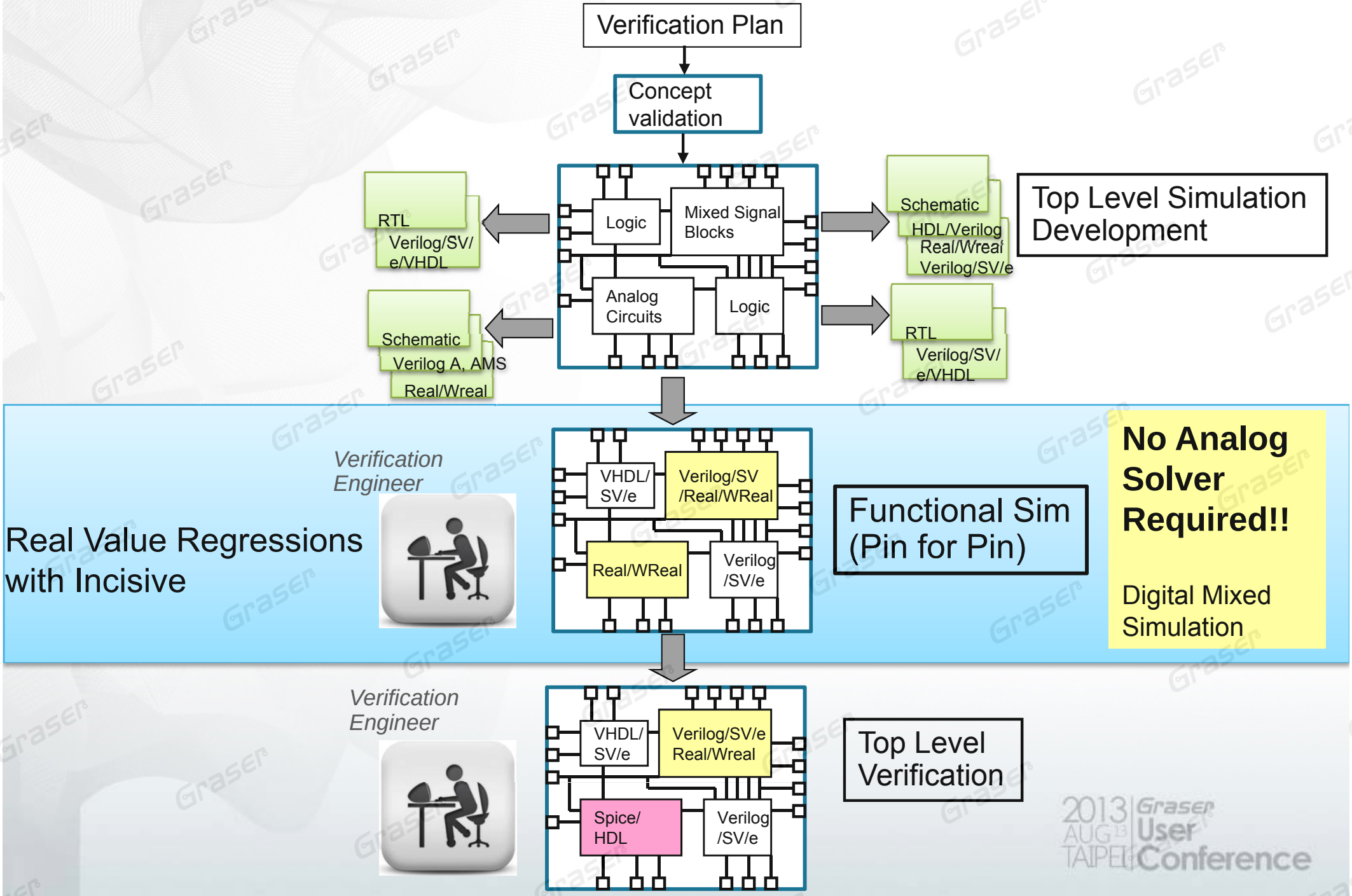
Connect module information:

```
Type: terminal
    Object Name: /I4/down
    Explanation: Discipline = logic
                 Connect module = myconnectLib.L2E:module
                 Rule = connect L2E merged
                 Parameters:
                   vsup = 5.000000000000000
                   vthi = 3.500000000000000
                   vtlo = 1.500000000000000
                   tr = 0.000000001000000
                   tf = 0.000000001000000
                   tx = 0.000000001000000
                   tz = 0.000000001000000
                   rlo = 200
                   rhi = 200
                   rx = 40
                   rz = 10000000.000000000000000
                 Terminal attributes:

Type: net
    Object Name: /down
    Explanation: Shortest path = down
                 Discipline = electrical
```

Digital-centric Verification

AMS Designer Verification in Incisive

Digital Centric Mixed Signal Use Model



Mixed Signal Packaging Proposal for Digital Centric Model

Real Value “IP Usage” Use Model

- Create a New mixed signal market segment for Cadence
  - Digital Mixed Signal (DMS)
- Provide a Real Value Modeling capability for digital centric use model
- Extremely high performance behavioral Mixed Signal verification
  - BigD/SmallA full-chip verification
  - Enable high volume digital-centric nightly regressions tests
- Allows customers to perform SoC Top-level mixed signal verification using only digital simulators

What's Real Number Modeling?

- Model analog blocks operation as signal flow models
- Digital only simulation → remains high simulation performance
- Use the event solver for fast calculation and avoid convergence issues
  - **wreal** ports in Verilog-AMS or
    - Note: wreal is defined only in the verilog-ams standard, however, only the digital kernel is used for simulation
  - **real** in VHDL or
  - **real** in SystemVerilog

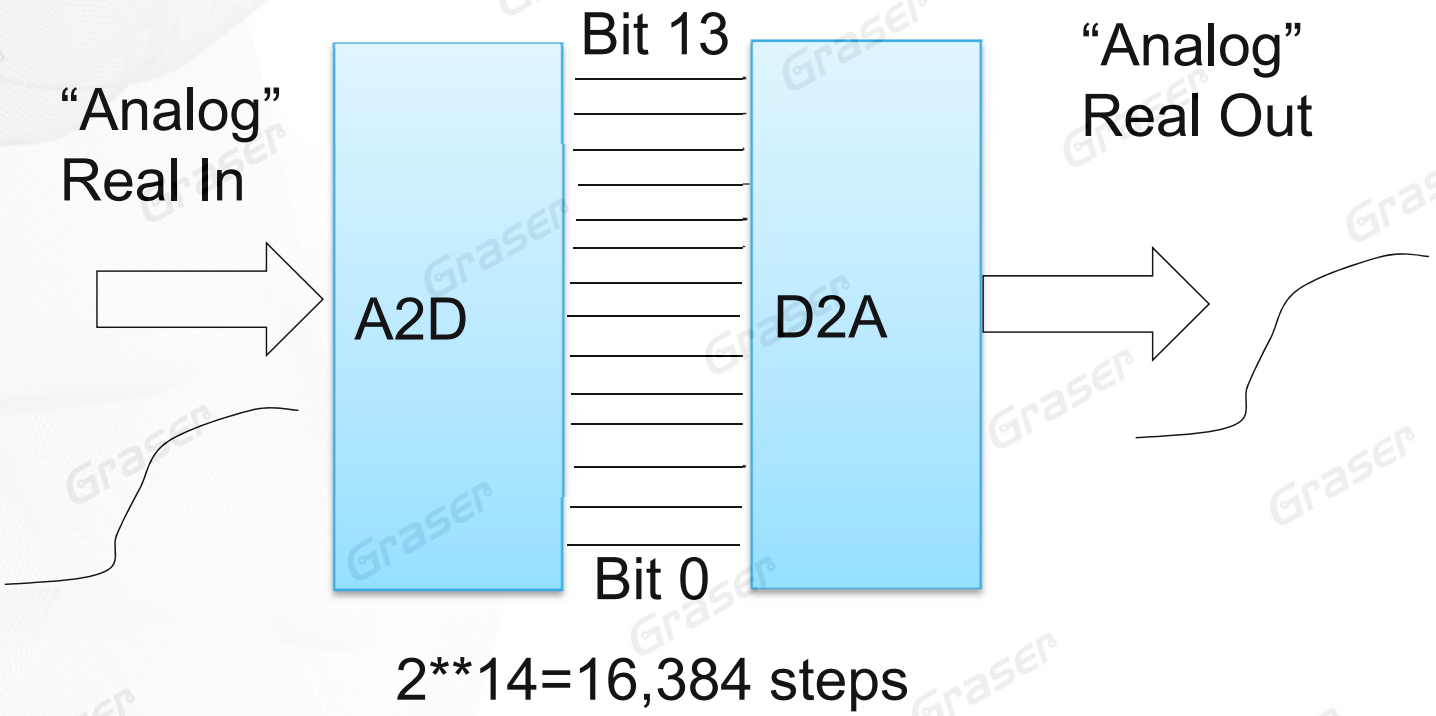
Real Modeling Concept

3 simulation paradigms in Mixed Signal:

1. Event simulation
   - Digital blocks
2. Analog solver: solve differential equations
   - Accurate analog simulation – based on circuit theory; ohms law, kirchoff's law
   - E.g. SPICE, transistor level
3. Signal-flow simulation
   - Analog and mixed-signal simulation based on behavioral models

Real Value Model Example

14bits ADC + 14bits DAC complete transfer



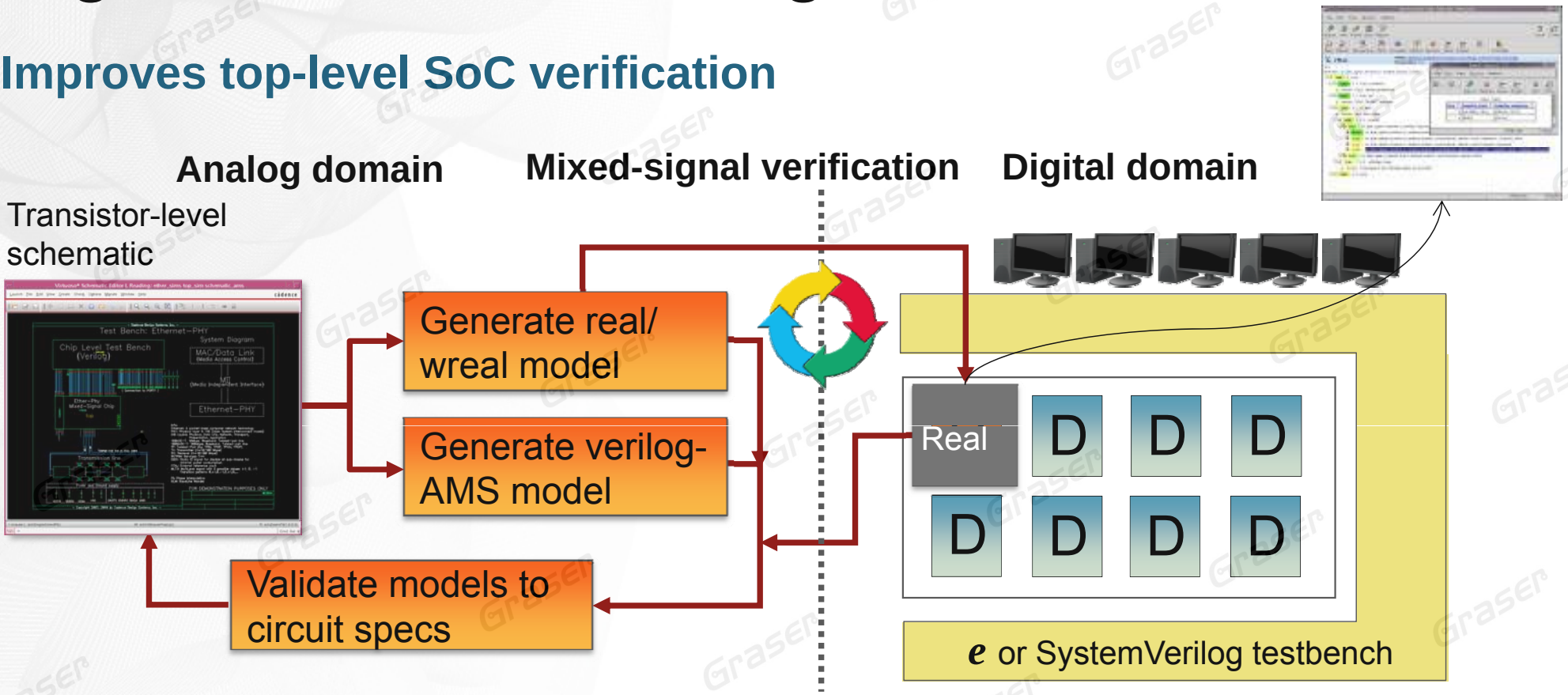
- HDL-A using “real” model: 3 seconds!
- Full transistor level simulation could take days …

Improve Analog/MS Simulation Performance

- **Transistor level ideal circuit (1.0 x)**
- Fast spice simulation (5-20 x)
- Analog behavioral modeling (5-100 x)
- Real number modeling (50-500 x)
- Pure digital model (500-10K x)

Digital-Centric Mixed-Signal Verification

Improves top-level SoC verification

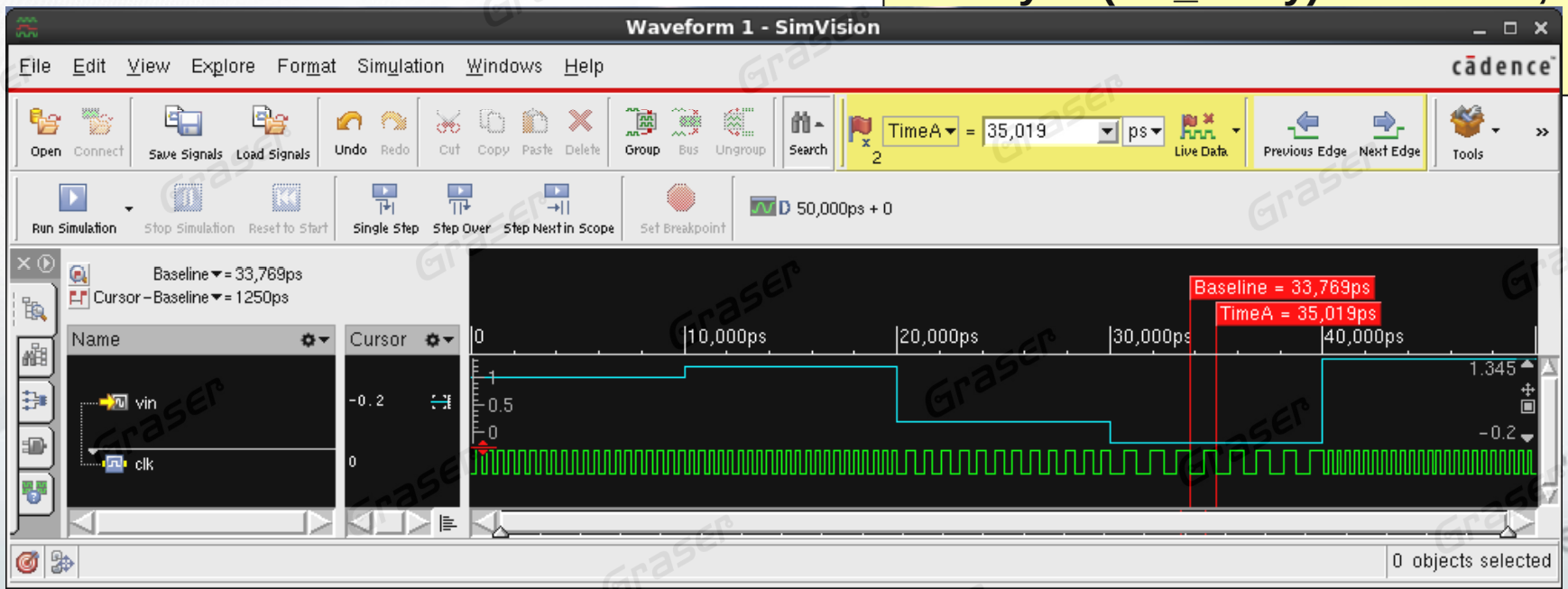


- High-performance, real-number modeling for mixed-signal verification
- Run full-chip verification regression suites at digital speeds
- Enabled by Incisive Digital Mixed-Signal (DMS) Option
- Cadence extensions to Verilog-AMS “wreal” feature donated to Accellera

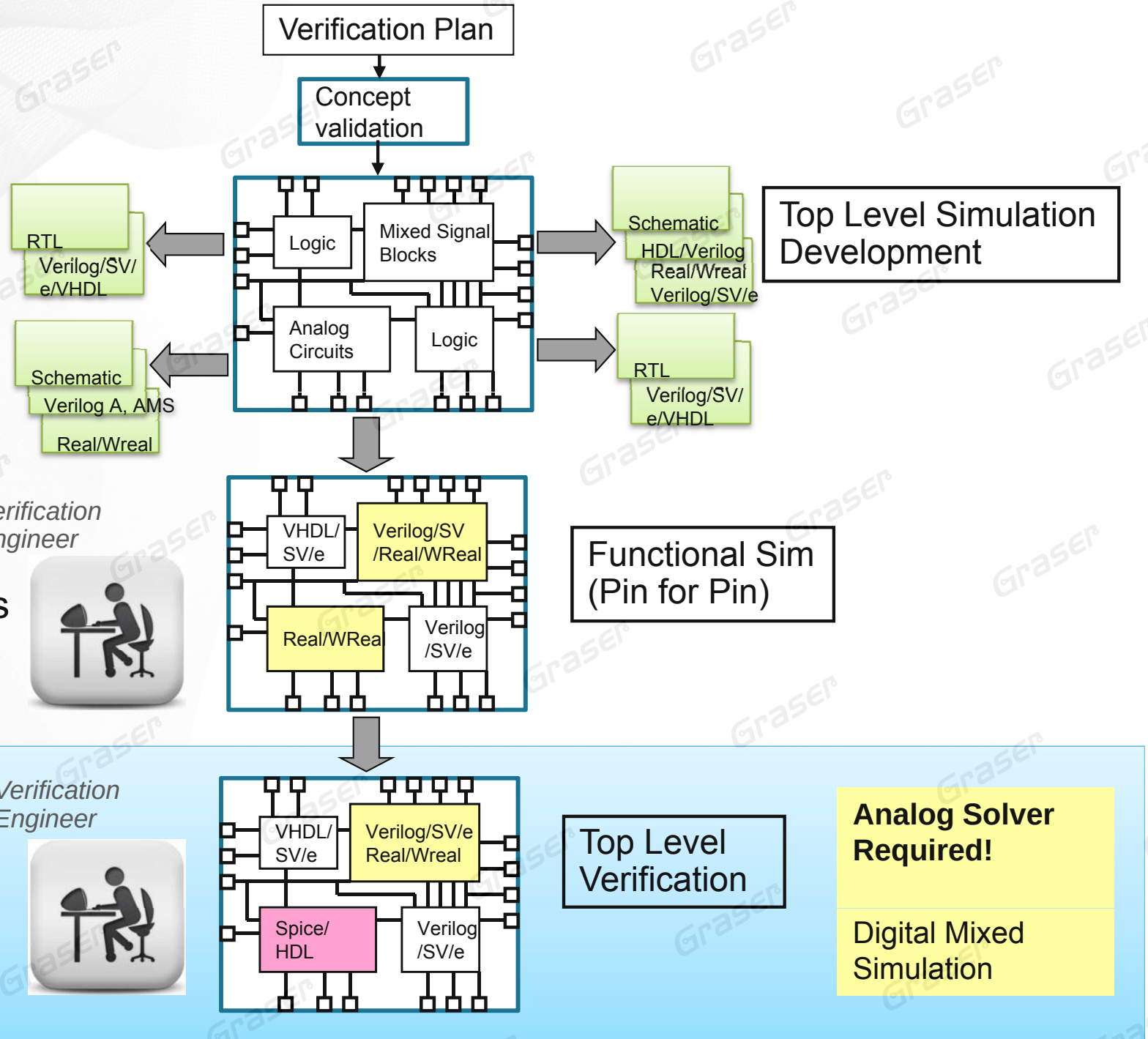
The Wreal Datatype in Verilog-AMS

- Wreal datatype declares a **real net** that has a real-valued connection to other modules
- A wreal net is discrete in time (event based, see the @(vin))
- It is continuous in value real

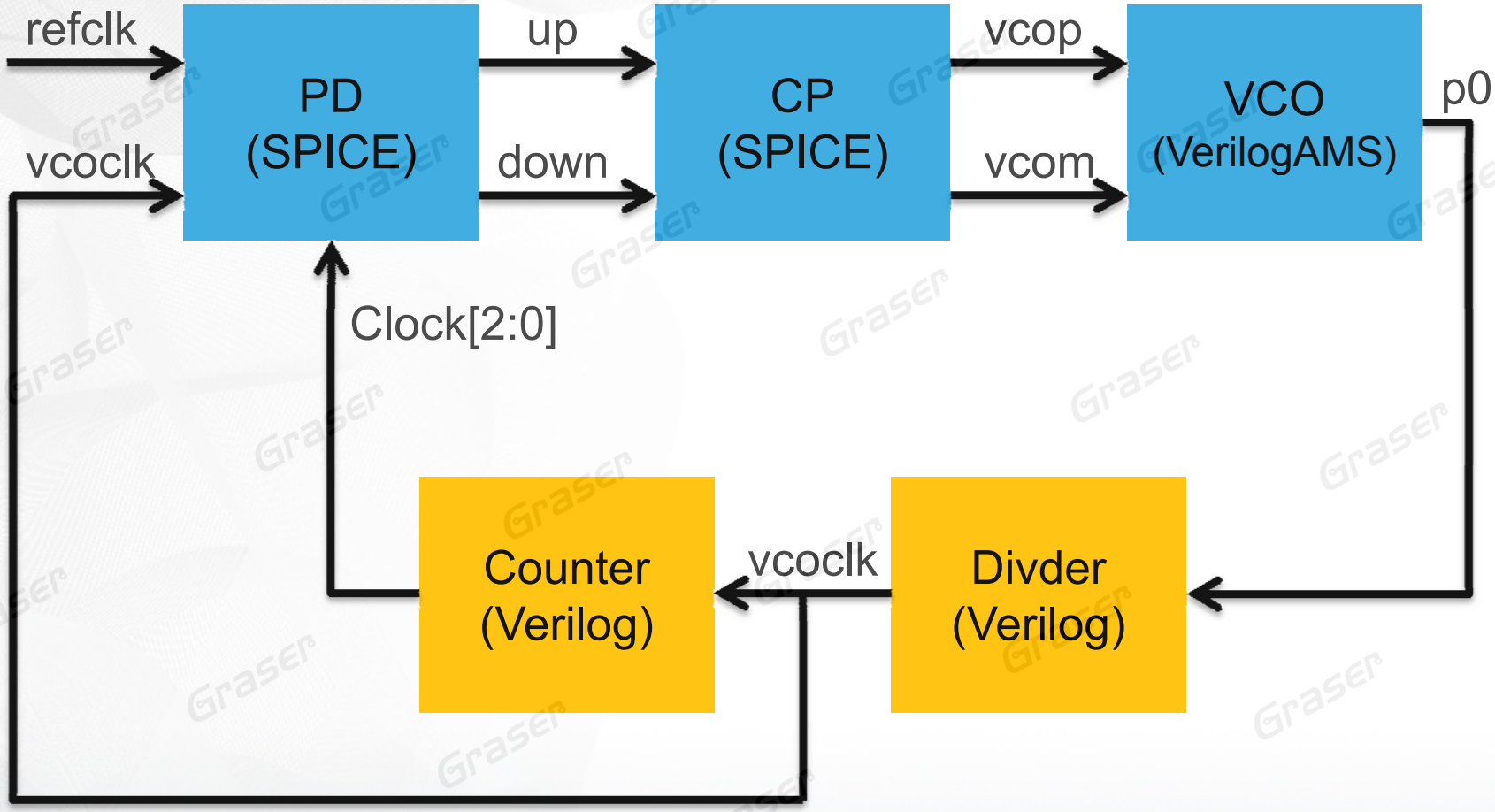
```
module vco(vin, clk);
  input vin; wreal vin;
  output clk;
  reg clk;
  real freq,clk_delay;
  always @(vin) begin
     freq = center_freq + vco_gain*vin;
     clk_delay = 1.0/(2*freq);
  end
  always #(clk_delay)  clk = ~clk;
```



Digital-Centric Mixed-Signal Use Model



PLL Example with SPICE, Verilog-AMS and Verilog Languages



File Preparation for Signal-Step Verification Flow

AMS Control File, Analog Control File and Digital Probing TCL File

Analog configure

```
**********************************
simulator lang=spice lookup=spectre
**********************************

.tran 1ns 40ns method=gear2
```

```
.probe v(*)
*.probe v(testbench.pll_top.vcom) v(testbench.pll_top.vcop)
*.probe v(testbench.pll_top.upm) v(testbench.pll_top.upp)
*.probe v(testbench.pll_top.downm) v(testbench.pll_top.downp)
```

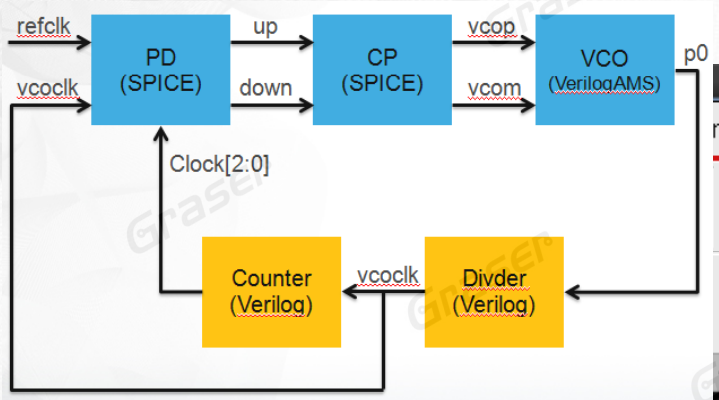
AMS configure

```
include "./models/resistor.scs" section=res
include "./models/diode.scs" section=dio
include "./models/pmos1.scs" section=nom
include "./models/nmos1.scs" section=nom
include "./source/analog/PLL.sp"
include "./acf.scs"
amsd {
        portmap subckt=pll_top busdelim="_"
        config cell=pll_top use=spice
        ie vsup=2.0
}
```

SimVision Script

```
database -open waves -into waves.shm -default
probe -create -database waves -all -depth all
probe -create -database waves testbench.refclk
probe -create -database waves testbench.clk_p0_1x
probe -create -database waves testbench.clk_p0_4x
probe -create -database waves testbench.p0
```

Mixed-Signal Flow for Digital-Centric Verification with Spice File



AMS configure

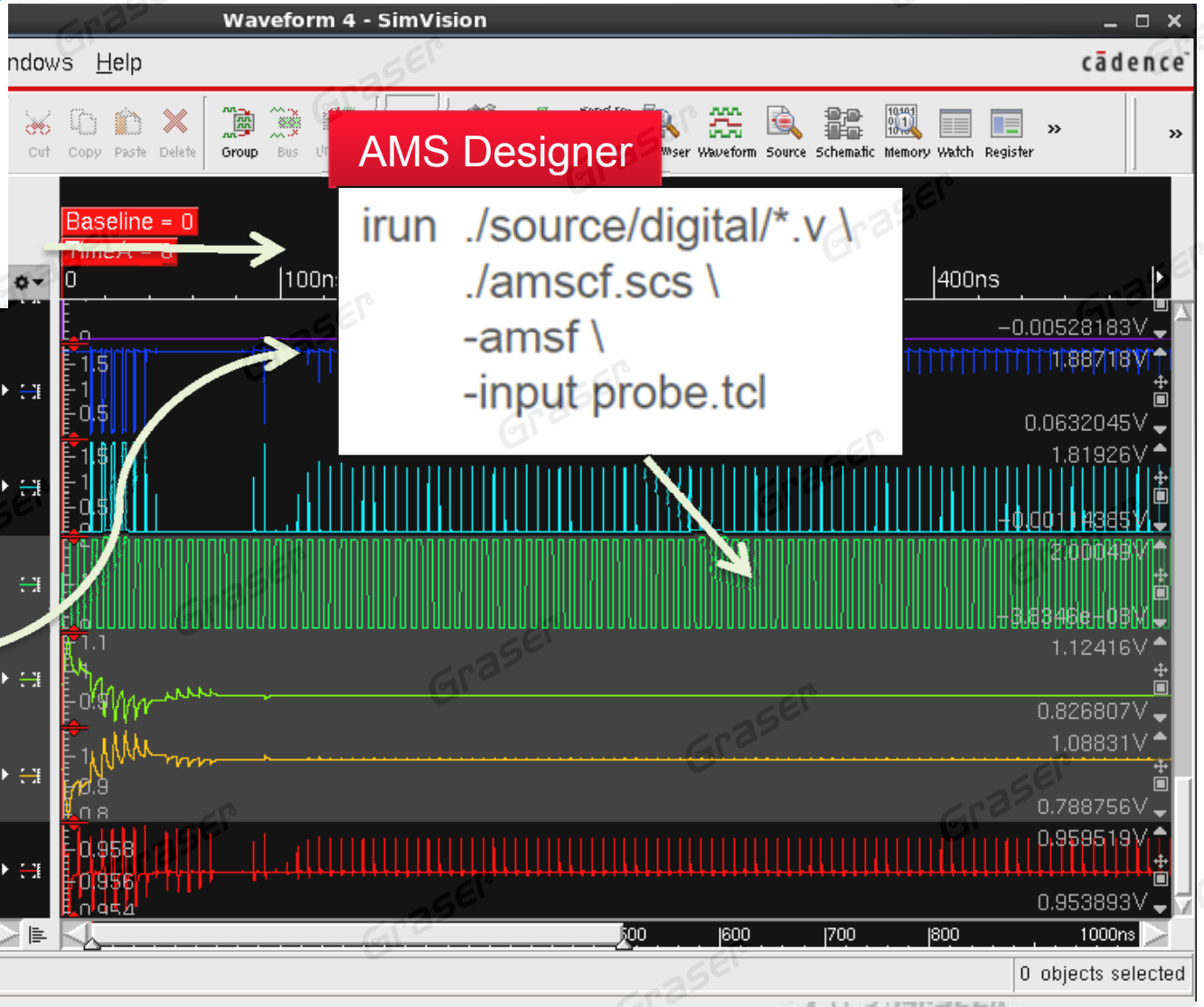
```
include "./models/resistor.scs" section=res
include "./models/diode.scs" section=dio
include "./models/pmos1.scs" section=nom
include "./models/nmos1.scs" section=nom
include "./source/analog/PLL.sp"
include "./acf.scs"
amsd {
        portmap subckt=pll_top busdelim="_"
        config cell=pll_top use=spice
        ie vsup=2.0
}
```

Analog configure

```
**********************************
simulator lang=spice lookup=spectre
**********************************

.tran 1ns 40ns method=gear2

.probe v(*)
*.probe v(testbench.pll_top.vcom) v(testbench.pll_top.vcop)
*.probe v(testbench.pll_top.upm) v(testbench.pll_top.upp)
*.probe v(testbench.pll_top.downm) v(testbench.pll_top.downp)
```

SimVision Script

```
database -open waves -into waves.shm -default
probe -create -database waves -all -depth all
probe -create -database waves testbench.refclk
probe -create -database waves testbench.clk_p0_1x
probe -create -database waves testbench.clk_p0_4x
probe -create -database waves testbench.p0
```



Summary

- Virtuoso AMS Designer is the integration of Virtuoso MMSIM and IES Simulators
- Two Facets for AMS Verification
  - Virtuoso GUI integration
  - Incisive batch mode
- Auto-inserted Connect Module Library
- AMS-APS Analog Solver Multi-threading Support and retaining SPICE accuracy
- Addressing Low Power Requirements
- Reduces re-spins
  - Leverages high-performance, real-number modeling
  - Performs SoC top-level mixed-signal verification
  - Finds and fixes errors much earlier in the design cycle by performing full-chip functional verification
- Boosts productivity
  - Eliminates convergence issues with digital-speed performance
  - Easily and accurately ports models between Virtuoso® and Incisive® environments
  - Achieves top-level verification