2024
DESIGN AND VERIFICATION™
DVCON
CONFERENCE AND EXHIBITION
UNITED STATES
SAN JOSE, CA, USA
MARCH 4-7, 2024

DV UVM based AMS co-simulation and verification methodology for mixed signal designs

Sandeep Sharma

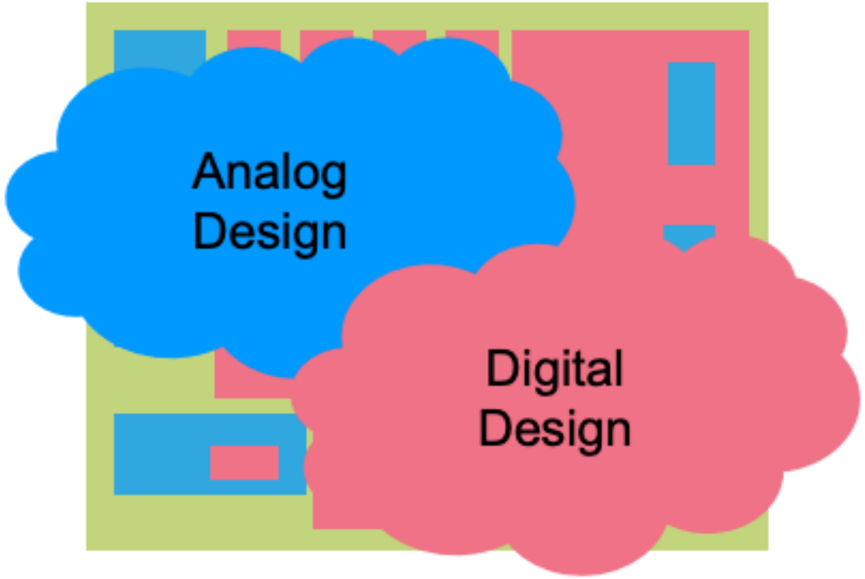
Meta reality labs

Agenda

- AMS verification challenges
- Legacy verification flow
- New MSV architecture
- Mixed signal co-simulation setup
- Verification improvements
- Future work
- Conclusion

Challenges in AMS Verification

- Handling the complexity of AMS designs
- Automation gap between analog and digital
- Prolonged analog/spice simulation run times
- Integration of analog and digital components
- Ensuring accuracy and performance
- No clear indication of verification completeness



Digital Verification Advancements

- Easier development and automation of digital design and verification methodologies
- Advanced simulation and formal verification tools enable thorough testing of complex designs
- AI and ML integration
- Hardware description language evolution and standardization

Limitations of Traditional SPICE Simulation

- Time-consuming nature of SPICE simulation
- Wide performance gap between spice and digital simulators
- Inefficient in handling complex SoC designs
- Lack of integration with digital verification tools

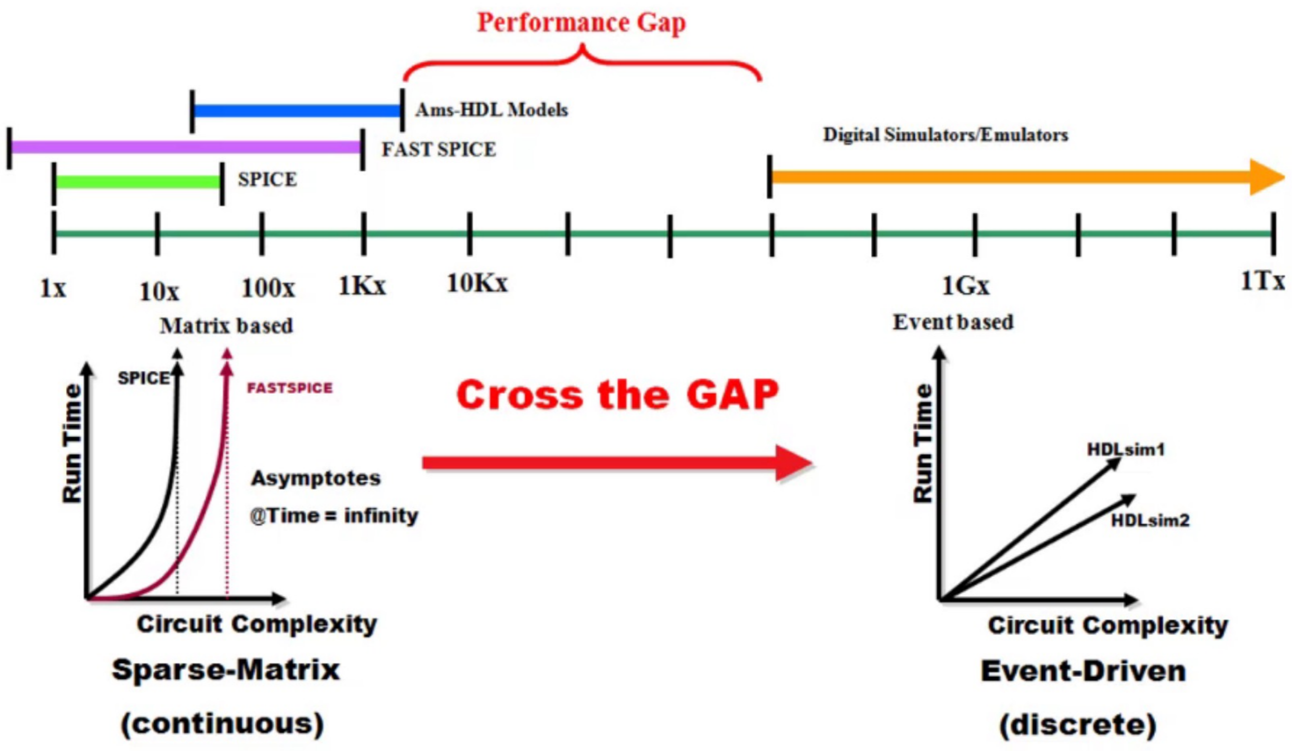
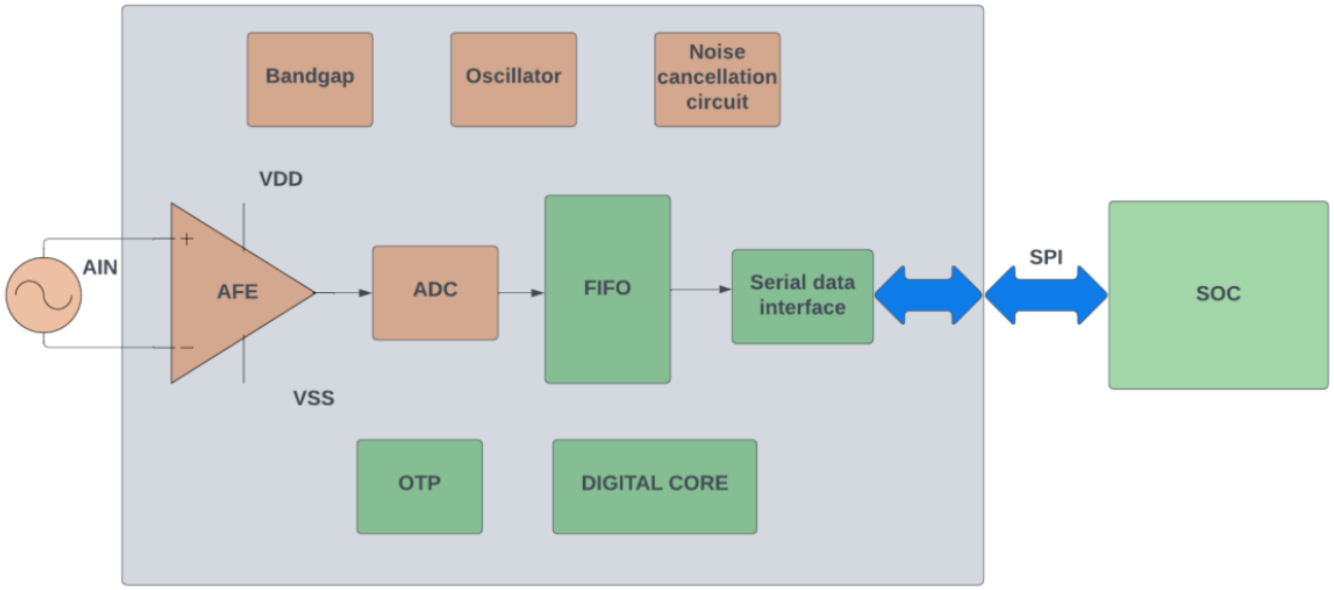


Image source: Cadence design systems

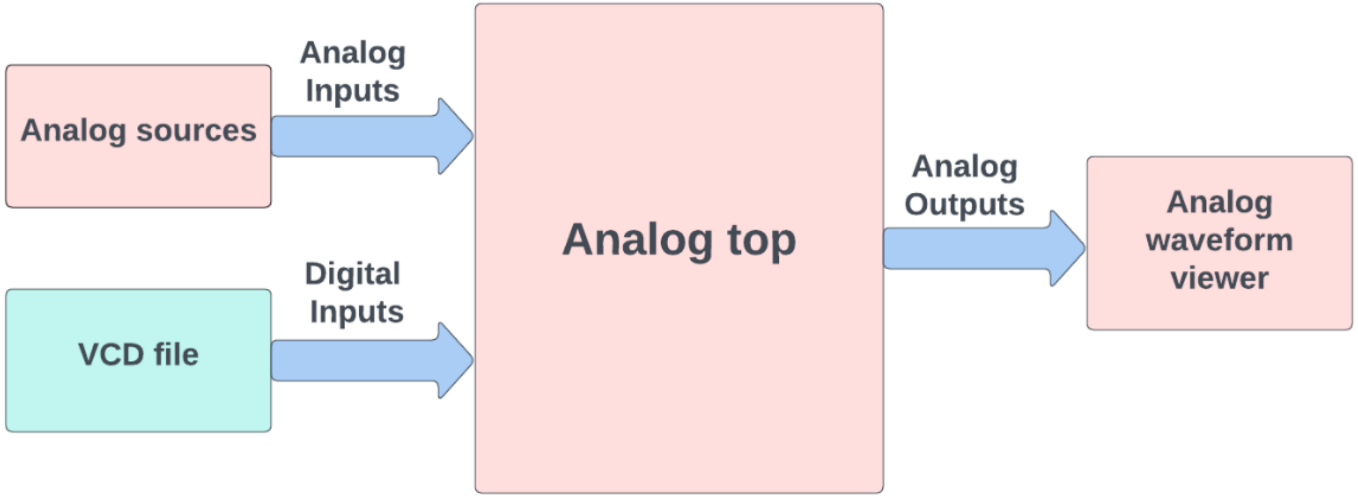
DUT and Design Environment

- Primary application: sampling real-world signals, process, and propagate to SoC
- Input analog signal amplified by analog front end (AFE) and then Sampled for digital conversion
- Digital data processed by digital ADC controller and loaded into FIFO
- Final digitized data output from subsystem accessed by SoC using SPI interface
- SoC acts as master device and provides clock to AMS subsystem



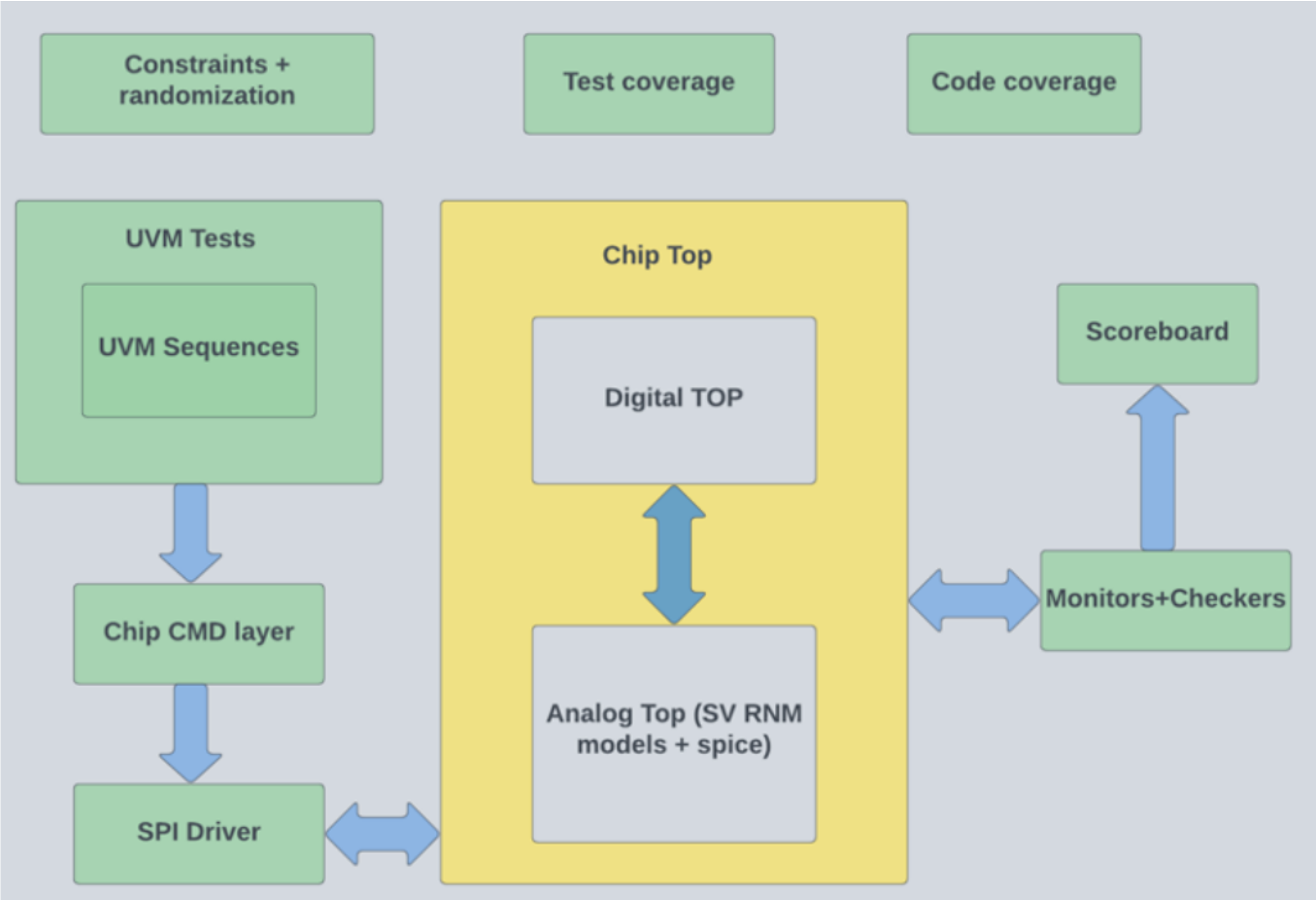
Legacy AMS Verification Environment Overview

- Separate workflows for analog and digital verification
- Reduced productivity
- Impact on development time and efficiency
- Misalignment in development schedules
- Challenges in maintaining consistency across designs



Introduction to New MSV Architecture

- Address limitations with a new MSV architecture
- Digital on top verification strategy
- Leveraging advanced digital verification concepts:
  - UVM
  - SV assertions
  - Coverage analysis and closure
  - Constraint randomization
- Improved functional coverage at chip-level
- Enhanced analog verification through nightly regressions at full digital speeds using DMS simulation

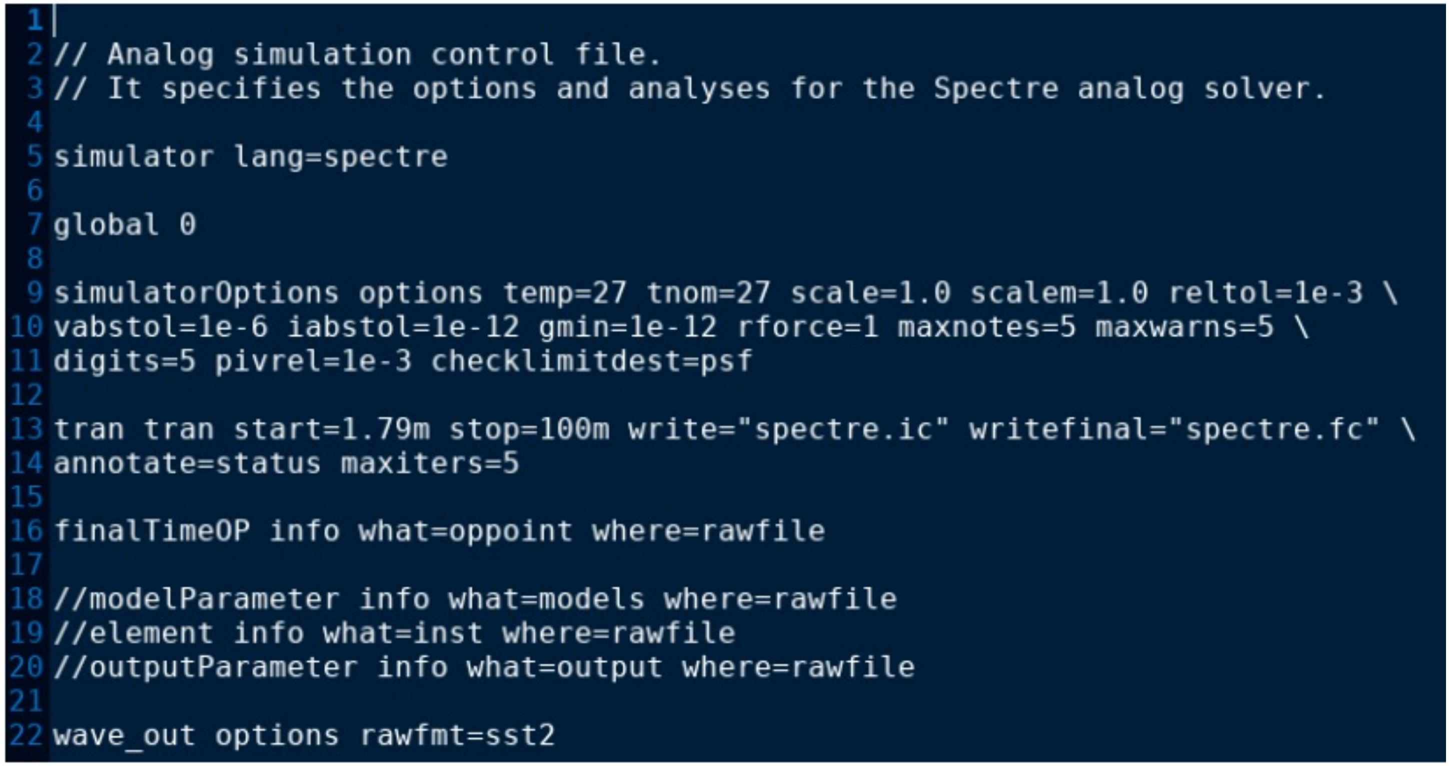


Mixed Signal Co-Simulation Setup

- Integration of analog spice simulator in DV UVM testbench environment
  - Analog control file
  - AMS control file
- Connect modules for mixed signal co-simulation
- SV and Spice netlist extraction from analog design environment
- Command line simulation script with appropriate simulation switches

Integration of Analog SPICE Simulator in DV UVM Testbench Environment

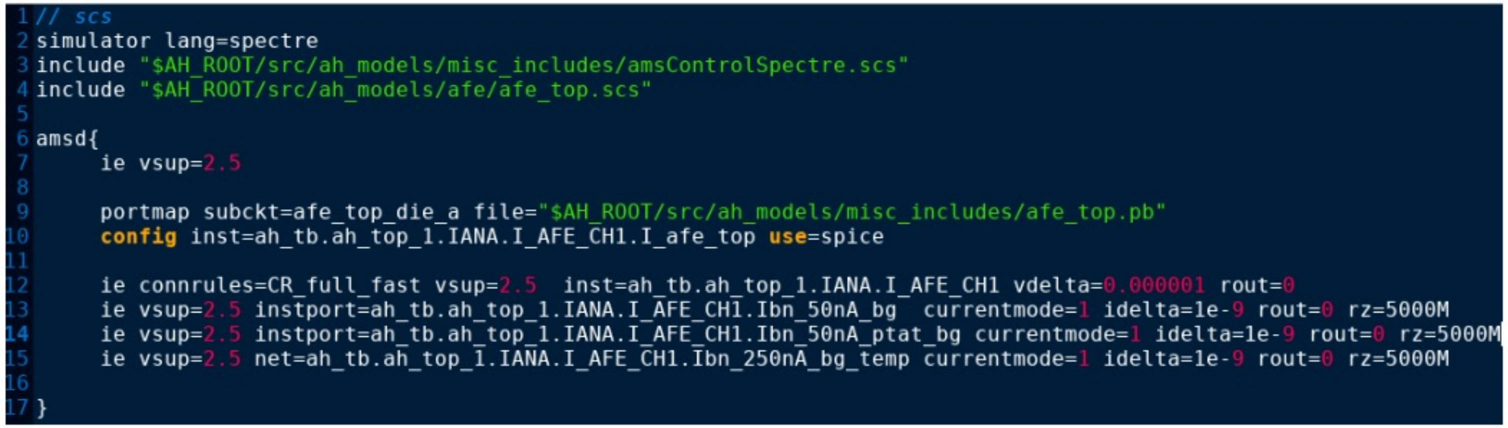
- Initially, DV UVM environment only used for digital mixed-signal (DMS) configuration
- Analog simulation engine is used during co-simulation with SPICE
- Analog control file needed to describe analog simulator options



```

// Analog simulation control file.
// It specifies the options and analyses for the Spectre analog solver.

simulator lang=spectre

global 0

simulatorOptions options temp=27 tnom=27 scale=1.0 scalem=1.0 reltol=1e-3 \
vabstol=1e-6 iabstol=1e-12 gmin=1e-12 rforce=1 maxnotes=5 maxwarns=5 \
digits=5 pivrel=1e-3 checklimitdest=psf

tran tran start=1.79m stop=100m write="spectre.ic" writefinal="spectre.fc" \
annotate=status maxiters=5

finalTimeOP info what=oppoint where=rawfile

//modelParameter info what=models where=rawfile
//element info what=inst where=rawfile
//outputParameter info what=output where=rawfile

wave_out options rawfmt=sst2
```

Integration of Analog SPICE Simulator in DV UVM Testbench Environment

- AMS control file sets up AMS configurations
- Centralizes controls for MS simulations
- Typical contents:
  - Inclusion of analog objects
  - Design configuration (spice vs SV vs verilog-A, etc.)
  - Analog and digital connectivity
  - Supply specification
  - Connect module setup



```
// scs
simulator lang=spectre
include "$AH_ROOT/src/ah_models/misc_includes/amsControlSpectre.scs"
include "$AH_ROOT/src/ah_models/afe/afe_top.scs"

amsd{
      ie vsup=2.5

      portmap subckt=afe_top_die_a file="$AH_ROOT/src/ah_models/misc_includes/afe_top.pb"
      config inst=ah_tb.ah_top_1.IANA.I_AFE_CH1.I_afe_top use=spice

      ie connrules=CR_full_fast vsup=2.5  inst=ah_tb.ah_top_1.IANA.I_AFE_CH1 vdelta=0.000001 rout=0
      ie vsup=2.5 instport=ah_tb.ah_top_1.IANA.I_AFE_CH1.Ibn_50nA_bg  currentmode=1 idelta=1e-9 rout=0 rz=5000M
      ie vsup=2.5 instport=ah_tb.ah_top_1.IANA.I_AFE_CH1.Ibn_50nA_ptat_bg currentmode=1 idelta=1e-9 rout=0 rz=5000M
      ie vsup=2.5 net=ah_tb.ah_top_1.IANA.I_AFE_CH1.Ibn_250nA_bg_temp currentmode=1 idelta=1e-9 rout=0 rz=5000M

}
```

Connect Module Setup for Mixed Signal Co-Simulation

- CM's can be configured within an AMS control file
- Key considerations while configuring CM's:
  - Analog sampling speed
  - Input/output impedance parameters (rin, rout)
  - Nominal supply voltage value (vsup)
- Incorrect settings can lead to simulation slowdowns or incorrect results
- Careful attention to these parameters is crucial for optimal performance and accuracy

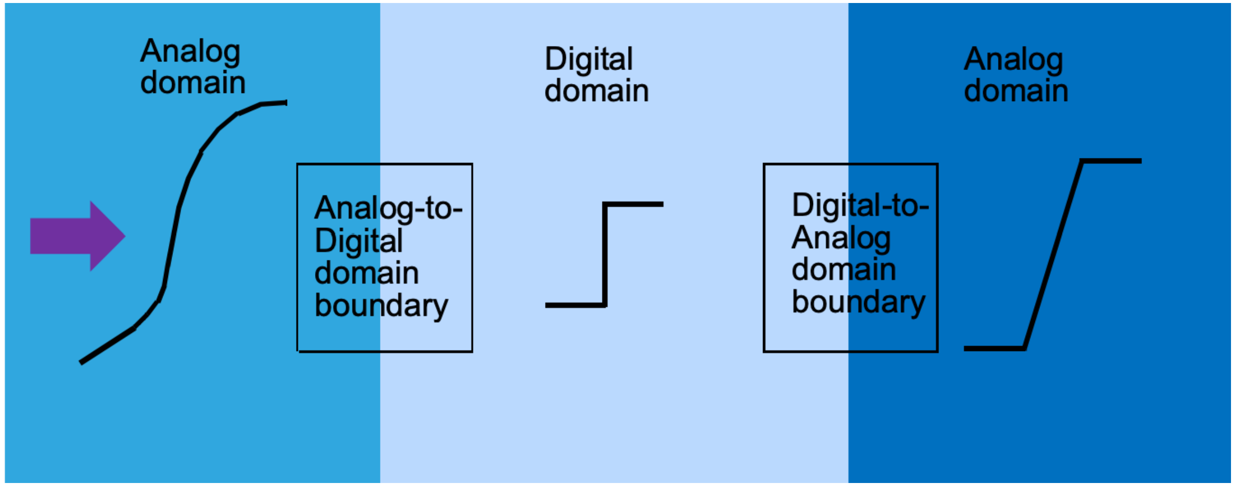

Analog domain
Analog-to-Digital domain boundary
Digital domain
Digital-to-Analog domain boundary
Analog domain

Image source: Cadence design systems

Advantages of New AMS co-simulation Environment

- Capability to run nightly regressions at full digital speeds (DMS simulations)
- Increase in analog functional coverage at chip level
- Capability to run nightly regressions for co-simulation tests with spice
- Use of SV constraint randomization for DMS and AMS co-simulation
- Use of SV assertions to check digital and spice signals in design hierarchy
- Use of advanced digital tools to analyze and debug simulation

Nightly DMS and Cosimulation Regressions

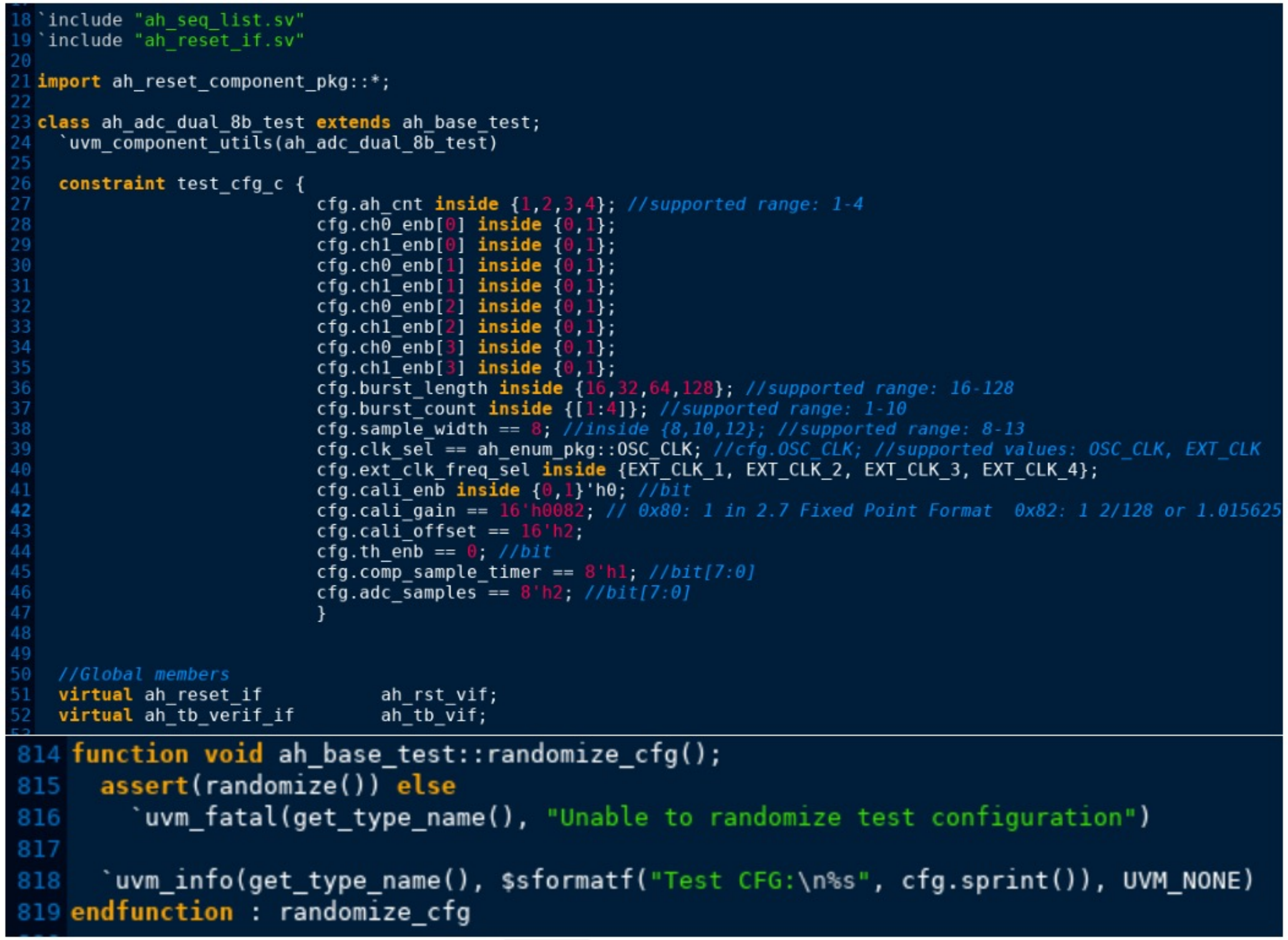
- Use of SV-RNM models enhance chip-level verification
- Accelerated verification cycle through digital-speed DMS regressions
- SV-RNM models are key enablers for advancing automated verification methodologies at chip-level
- SPICE Netlist integration enables nightly regressions for co-sim tests
- Robust verification and analog performance evaluation of important design blocks at chip level
- Use of Fast Spice simulators for reduced simulation time

Increased Analog Functional Coverage at Chip Level

- DMS model integration in DV-UVM eliminated simulation speed bottlenecks.
- Increased execution of functional scenarios led to substantial coverage improvement.
- More robust and thorough analog verification due to use of DV-UVM framework

SV Constraint Randomization for Co-Simulation

- Use of constraint randomization to avoid excessive verification in both DMS and AMS tests
- Nightly chip-level tests are conducted
- In just one week, we can verify seven different configurations with a single AMS spice test.

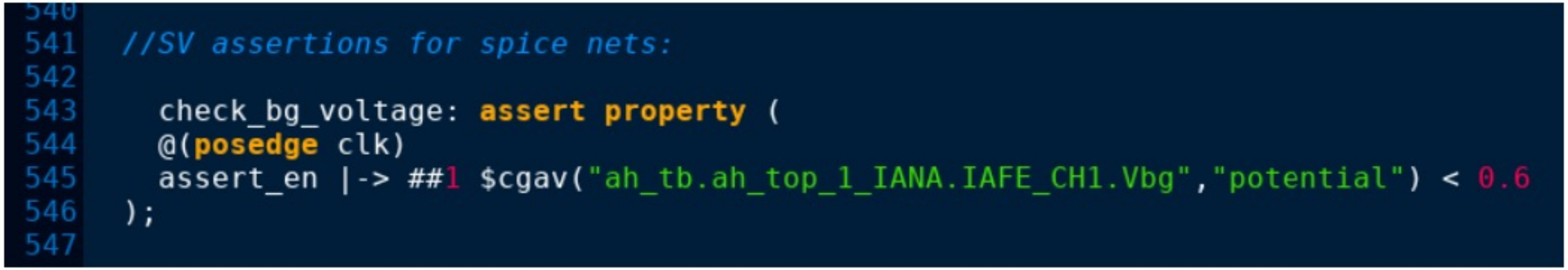


```
`include "ah_seq_list.sv"
`include "ah_reset_if.sv"

import ah_reset_component_pkg::*;

class ah_adc_dual_8b_test extends ah_base_test;
  `uvm_component_utils(ah_adc_dual_8b_test)

  constraint test_cfg_c {
                          cfg.ah_cnt inside {1,2,3,4}; //supported range: 1-4
                          cfg.ch0_enb[0] inside {0,1};
                          cfg.ch1_enb[0] inside {0,1};
                          cfg.ch0_enb[1] inside {0,1};
                          cfg.ch1_enb[1] inside {0,1};
                          cfg.ch0_enb[2] inside {0,1};
                          cfg.ch1_enb[2] inside {0,1};
                          cfg.ch0_enb[3] inside {0,1};
                          cfg.ch1_enb[3] inside {0,1};
                          cfg.burst_length inside {16,32,64,128}; //supported range: 16-128
                          cfg.burst_count inside {[1:4]}; //supported range: 1-10
                          cfg.sample_width == 8; //inside {8,10,12}; //supported range: 8-13
                          cfg.clk_sel == ah_enum_pkg::OSC_CLK; //cfg.OSC_CLK; //supported values: OSC_CLK, EXT_CLK
                          cfg.ext_clk_freq_sel inside {EXT_CLK_1, EXT_CLK_2, EXT_CLK_3, EXT_CLK_4};
                          cfg.cali_enb inside {0,1}'h0; //bit
                          cfg.cali_gain == 16'h0082; // 0x80: 1 in 2.7 Fixed Point Format  0x82: 1 2/128 or 1.015625
                          cfg.cali_offset == 16'h2;
                          cfg.th_enb == 0; //bit
                          cfg.comp_sample_timer == 8'h1; //bit[7:0]
                          cfg.adc_samples == 8'h2; //bit[7:0]
                          }


  //Global members
  virtual ah_reset_if          ah_rst_vif;
  virtual ah_tb_verif_if       ah_tb_vif;
```

```
function void ah_base_test::randomize_cfg();
  assert(randomize()) else
    `uvm_fatal(get_type_name(), "Unable to randomize test configuration")

  `uvm_info(get_type_name(), $sformatf("Test CFG:\n%s", cfg.sprint()), UVM_NONE)
endfunction : randomize_cfg
```

Use of SV Assertions to Check Digital and SPICE Signals in Design Hierarchy

- DV-based UVM testbench enables writing SV assertions to verify key digital and analog nets in AMS and DMS simulations
- SV assertion use for analog nets allows automatic checking, verifying correct device behavior for all possible stimulus variations
- Below snapshot demonstrates SVA on electrical net to check node voltage does not fall below 0.6V



```
//SV assertions for spice nets:

  check_bg_voltage: assert property (
  @(posedge clk)
  assert_en |-> ##1 $cgav("ah_tb.ah_top_1_IANA.IAFE_CH1.Vbg","potential") < 0.6
);
```

Future Work

- Explore SV EEnet Modeling for Analog
- For complex designs, AMS co-simulation flow may not be feasible due to simulation time bottlenecks
- Advanced methodologies like SV EEnet modeling can be leveraged in such scenarios
- SV model capable of modeling electrical quantities (voltage, current, frequency, phase) on a single net using custom user-defined nettypes (UDN)
- Closer to schematic design in terms of electrical accuracy
- SV RNM suitable for transferring single values between blocks, but EEnet required for interactions between modules, such as:
  - Voltage divider relationships with finite resistance
  - Current sources and resistive loads
  - Power supplies responding to changes in load current
  - Capacitive loading effects in certain analog designs

Future Work

- Adopting UVM-AMS Standard for Mixed Signal Verification
    - DV UVM testbenches lack an easy way to generate analog source signals for analog nodes
    - UVM-AMS methodology under development for improved verification of analog components at system level
    - Unified approach makes UVM more mixed-signal aware, enhancing verification of analog and mixed-signal components/subsystems
    - Objective of UVM-AMS standard is to standardize a method for driving and monitoring mixed-signal nets within UVM, including stimulus, scoreboarding, and analysis

Conclusion

- Legacy verification flow and gaps discussed
- Command-line-based MSV methodology introduced
- Integration of AMS sub-system testbench into SoC environment
- Incorporation of advanced features at SoC level
- Adaptable environment for multiple projects
- Potential widespread adoption of unified mixed-signal methodology
- More analog designers running mixed-signal verification from command line
- Testing spice blocks at chip level with advanced techniques
- Reuse of advanced testbenches created by DV team
- Highlighted verification improvements following new MSV architecture adoption

Acknowledgement

- I extend my heartfelt gratitude to the Meta design and DV engineers for their invaluable support and expertise in crafting this innovative AMS verification methodology. Special thanks to:
  - Justin Schauer
  - Joe Dao
  - Kamran Haqqani
  - Hisham Qureshi
  - Atif Malik
  - Chifa Dammak
  - Nirali Patel
- Your contributions have been instrumental in achieving our goals. Thank you for your dedication and assistance throughout this endeavor.

Questions?