2025 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION EUROPE

MUNICH, GERMANY
OCTOBER 14-15, 2025

Implementing Functional Coverage for Analog IPs in Mixed-Signal Verification Environments

Akib Mohammad Azam Zaidi Ana-Maria Radu

ams OSRAM Group

accellera SYSTEMS INITIATIVE

Why Analog Coverage matters

Motivation

- Growing complexity of mixed-signal systems
- Digital verification uses MDV successfully
- Analog verification lacks structured coverage
- Prior work:
  - RNM + UVM flows exist
  - Some require schematic instrumentation or rely on abstraction

Challenges

- Continuous nature of analog signals
  - Dynamic behavior
  - Limited observability at schematic level
- RNM helps, but schematic-level coverage is still underdeveloped
- No scalable, unified approach for RNM + schematic coverage

Proposed Solution

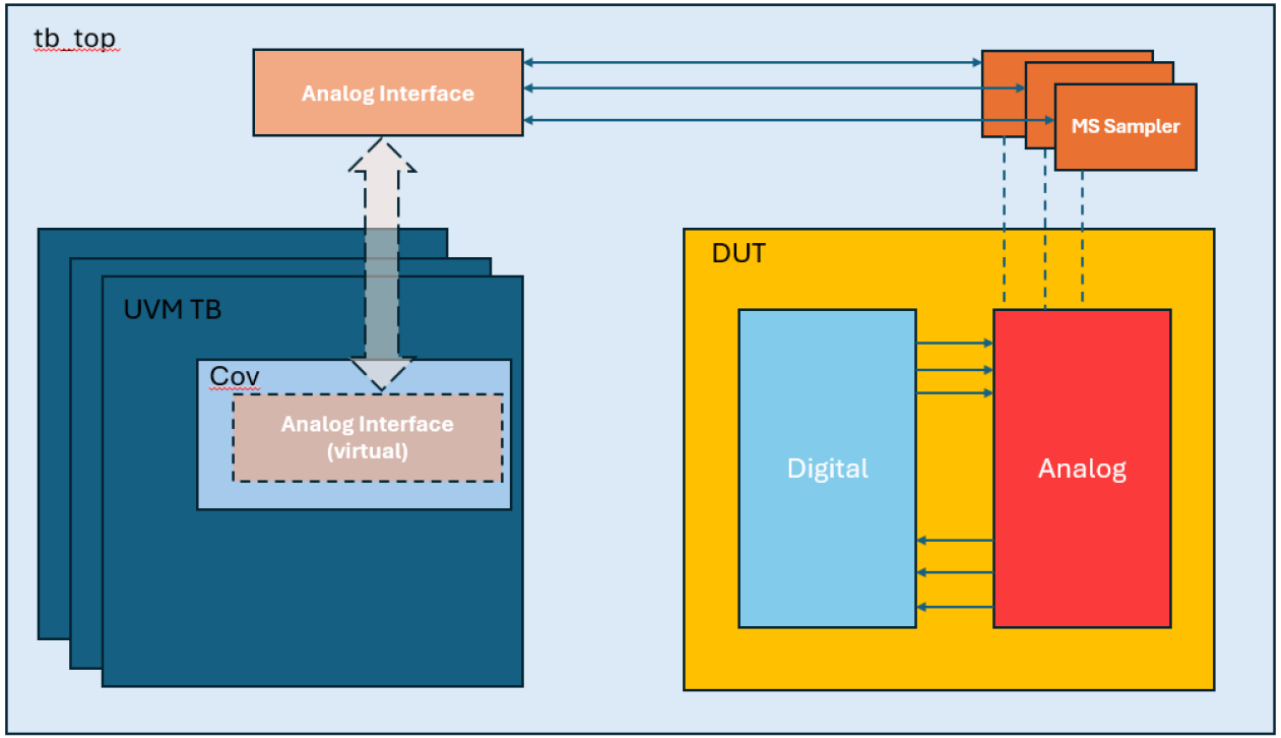
- UVM-based top-level testbench for mixed-signal verification
- Cross-compatible coverage for:
  - **Real Number Models (RNM)**
  - **Analog schematics**
- Convert analog signals → discrete real values for monitoring
- Define SystemVerilog covergroups from IP specifications
- Scalable approach for AMS & DMS environments

  - Unified infrastructure
  - Reusable components

  - Non-Intrusive Approach
  - Consistent Metrics Across Configurations

Methodology Overview

- Identify key analog signals and nodes
- Convert analog signals → real values
- Route the signals to SV interface
- Define covergroups based on IP specs
- Merge coverage data from DMS & AMS simulations



Signal Extraction

- Identify critical analog signals and nodes
- Use proprietary sampling code (**MS-Sampler)** to extract signals from DUT boundaries
- Convert analog signals → real values for monitoring
- Route real-valued signals to SystemVerilog interface
- Broadcast interface to UVM components via *uvm_config_db*
- Supports both **DMS** and **AMS** simulation modes

```
BEGIN
  INPUT : STRING PATH, LOGIC_HI, LOGIC_LO
  OUTPUT : DISCRETE REAL SIGNAL, LOGIC SIGNAL

  READ PORT FROM STRING PATH
  IF ANALOG PORT
    IF CURRENT_MODE
      SAMPLE CURRENT @ ANALOG STEP
      CONV TO DISCRETE REAL → OUTPUT REAL SIGNAL
    ELSE
      IF LOGIC MODE
        SAMPLE VOLTAGE @ ANALOG STEP
        CONV TO LOGIC USING LOGIC_HI, LOGIC_LO → OUTPUT LOGIC SIGNAL
      ELSE
        SAMPLE VOLTAGE @ ANALOG STEP
        CONV TO DISCRETE REAL → OUTPUT REAL SIGNAL
  ELSE
    IF LOGIC MODE
      SEND DISCRETE FROM PATH → OUTPUT LOGIC SIGNAL
    ELSE
      SEND DISCRETE FROM PATH → OUTPUT REAL SIGNAL
END
```

Coverage Plan

- Analyze IP spec & align with designers
- Gather key parameters: voltage, current, timing

- Define SV package for real-type attributes
- Create coverage plan: items, behaviors, spec relevance

Block	Coverage Item	Functional Behavior	Specification Relevance
LDO_A	cp_LDO_A_sel	Selection bits exercised	Selection Settings
	cross_LDO_A_sel_x_SUPPLY	Impact of output voltage selection	Output response match selection setting
POR1	cross_SUPPLY_VOLTAGE_x_POR_TRIGGER	Expected POR1 triggers for different supply voltage region	Validate voltage trip window sensitivity
	cp_RAMP_TIME	Fast, Nominal and slow ramping of input supply	Capture ramp sensitivity
	cross_RAMP_DIRECTION_RAMP_TIME	Fast and slow ramping in both direction	Power Sequencing
UVA	cross_SUPPLY_VOLTAGE_x_UV_A_OUT_TRIGGER	UVA supply threshold match with output response	Expected UV condition trigger
	cross_RAMP_TIME_x_GLITCH_DETECTION	Rejection and Glitch Exercise	Corner Case Stimulus

Covergroup - example

```
covergroup cg_POR_A
   option.per_instance = 1;
   cp_SUPPLY_VOLTAGE : coverpoint dut_monitor_ams_vif.POR_A_SUPPLY  {
      bins LOW       = {[0.0                  : POR_A_LOW]          };
      bins THRESHOLD = {[POR_A_THRESH_LL      : POR_A_THRESH_UL]    };
      bins NOMINAL   = {[POR_A_THRESH_NOM_LL : POR_A_THRESH_NOM_UL]}; }
   // Coverage for how fast voltage ramps up/down
   cp_RAMP_TIME : coverpoint dut_monitor_ams_vif.ramp_time {
      bins SLOW   = {[POR_A_SLOW_RAMP_LL : POR_A_SLOW_RAMP_UL] };
      bins MID    = {[POR_A_NOM_RAMP_LL  : POR_A_NOM_RAMP_UL]  };
      bins FAST   = {[POR_A_FAST_RAMP_LL : POR_A_FAST_RAMP_UL] };}
   // Voltage ramping direction up/down
   cp_RAMP_DIRECTION: coverpoint dut_monitor_ams_vif.ramp_dir {
      bins rise = {1};
      bins fall = {-1};}
   // Whether POR trip output asserted
   cp_POR_TRIGGER :coverpoint dut_monitor_ams_vif.POR_A_OUT {
      bins yes = {1};
      bins no  = {0};}
  cross_RAMP_TIME_x_POR_TRIGGER: cross cp_RAMP_TIME, cp_POR_TRIGGER;

  cross_SUPPLY_VOLTAGE_x_POR_TRIGGER: cross cp_SUPPLY_VOLTAGE, cp_POR_TRIGGER {
    ignore_bins ignore_0 = binsof(cp_SUPPLY_VOLTAGE.LOW) && binsof(cp_POR_TRIGGER.no);
    ignore_bins ignore_1 = binsof(cp_SUPPLY_VOLTAGE.NOMINAL) && binsof(cp_POR_TRIGGER.yes);}

   cross_RAMP_DIRECTION_RAMP_TIME: cross cp_RAMP_DIRECTION, cp_RAMP_TIME;
endgroup: cg_POR_A
```

Cover items with bins as planned:

- Supply voltage
- Ramp time
- Ramp directions
- Trigger
- Crosses between the above

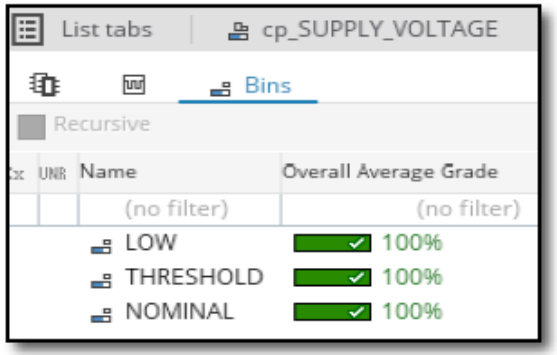
ignore_bins exclude irrelevant combinations to focus on spec-relevant cases;
e.g.: POR will always be triggered when supply reaches LOW

Simulation Flow

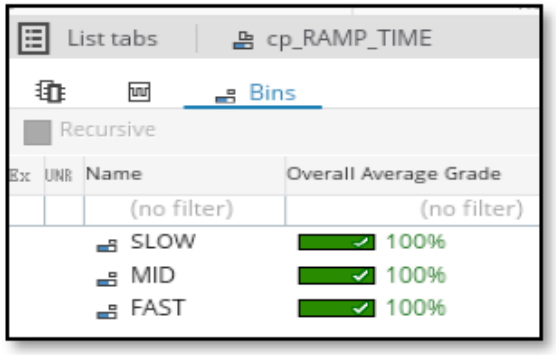
1. DMS runs – all analog coverage enabled
   - RNM for speed
   - Ideal for early-stage stimulus validation – aim 100% functional coverage
2. AMS runs – selective analog coverage enabled
   - Schematic level simulation for accuracy – same testcases as on DMS run
   - Coverage sampling only enabled for selected schematic IP
   - Avoid mixing RNM and schematic data
3. Merge of all coverage database across configuration mix
   - Reuse coverage models across RNM and schematic setups
   - Use spec-based parameters for corner-aware AMS runs
   - Merge results for consistent comparison and observability
   - Avoid mixing RNM and schematic data for same IP

```
task ams_fcoverage::run_phase(uvm_phase phase);
   super.run_phase(phase);
   fork
      `ifdef AMS_SIM
      // Invoke Sample Task for Covergroups
      `ifdef LDO_A_SCHEM
        cg_LDO_A_OUT_sample();
      `endif
      `ifdef POR_A_SCHEM
        cg_POR_A_sample();
      `endif
      `ifdef UV_A_SCHEM
        cg_UV_A_sample();
      `endif
      `else
        cg_LDO_A_OUT_sample();
        cg_POR_A_sample();
        cg_UV_A_sample();
      `endif
      join_none
endtask: run_phase
```

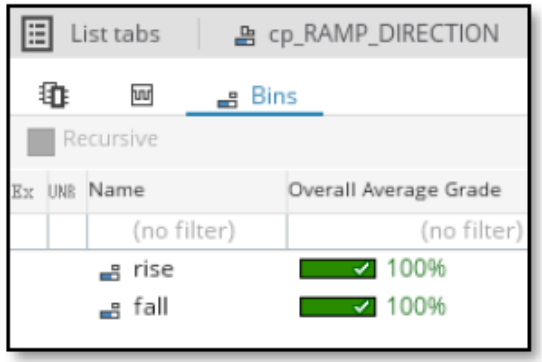
Results – Coverage Insights



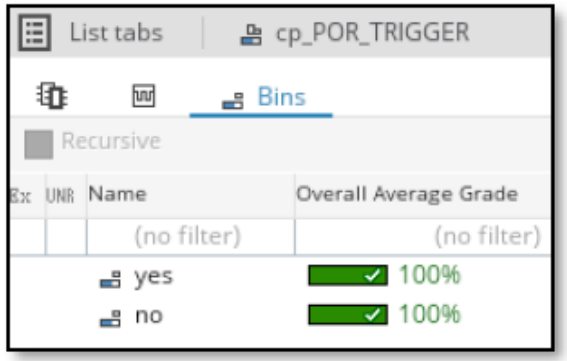
a



b



c



d

Full activation of Individual coverpoint ***cg_POR_A***

a) supply voltage
b) ramp time
c) ramp direction
d) POR trigger status

Results – Coverage Insights

List tabs | AxB cross_RAMP_TIME_x_POR_TRIGGER

Bins

Recursive

Exc	UNR	Name	cp_RAMP_TIME	cp_POR_TRIGGER	Overall Average Grade
		(no filter)	(no filter)	(no filter)	(no filter)
		SLOW,yes	SLOW	yes	100%
		SLOW,no	SLOW	no	100%
		MID,yes	MID	yes	100%
		MID,no	MID	no	100%
		FAST,yes	FAST	yes	100%
		FAST,no	FAST	no	100%

a

List tabs | AxB cross_SUPPLY_VOLTAGE_x_POR_TRIGGER

Bins

Recursive

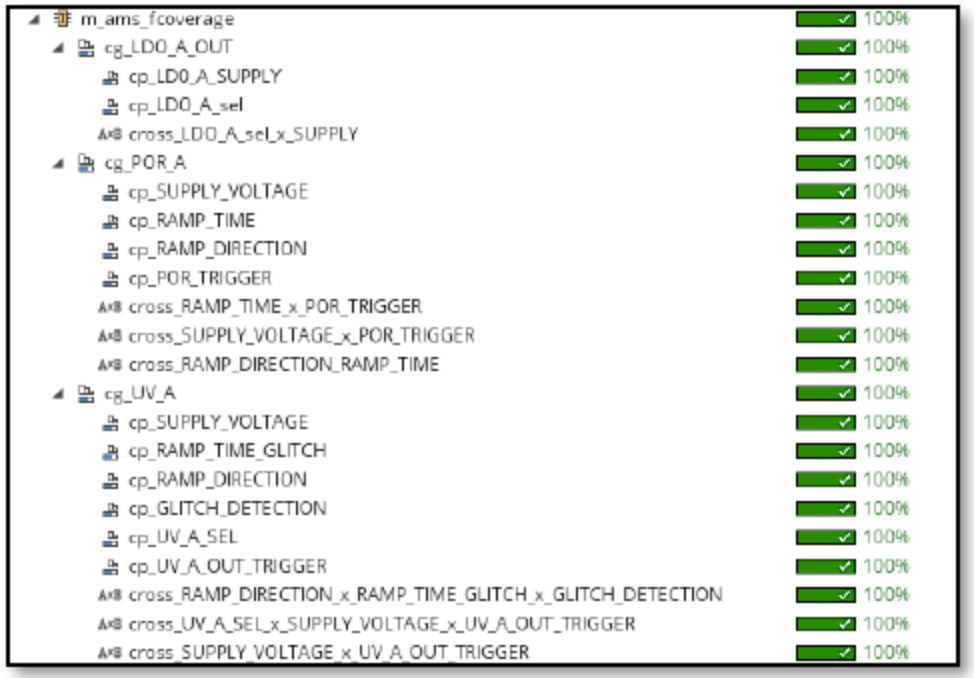
Exc	UNR	Name	cp_SUPPLY_VOLTAGE	cp_POR_TRIGGER	Overall Average Grade
		(no filter)	(no filter)	(no filter)	(no filter)
		LOW,yes	LOW	yes	100%
		THRESHOLD,yes	THRESHOLD	yes	100%
		THRESHOLD,no	THRESHOLD	no	100%
		NOMINAL,no	NOMINAL	no	100%

b

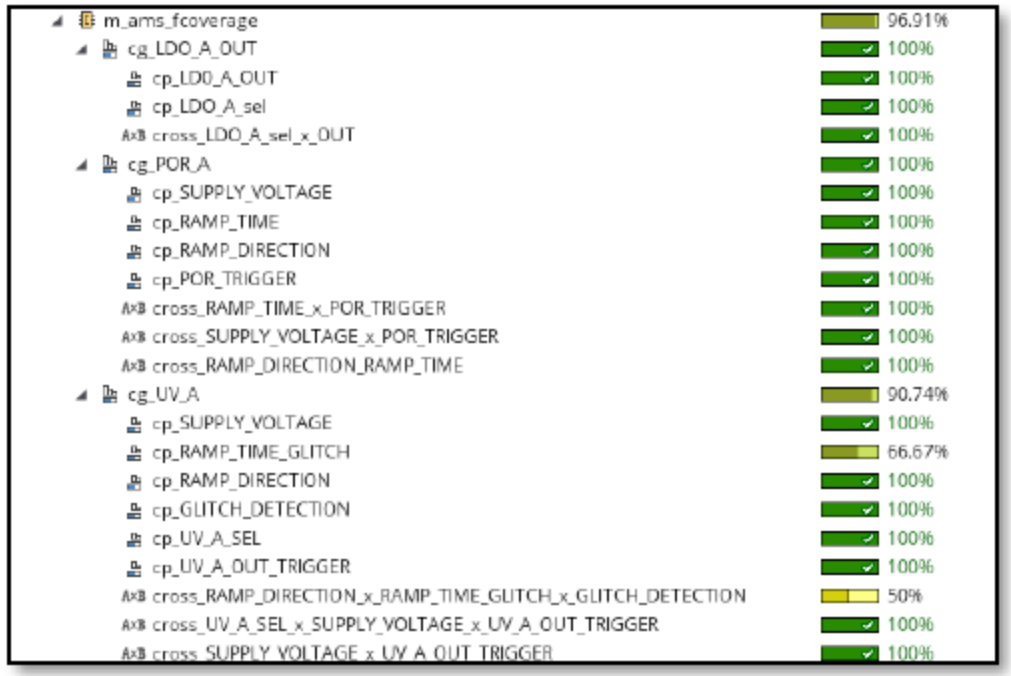
Cross-Coverage Results for POR Analog IP

a) ramp time x por trigger
b) supply voltage x por trigger

Results - Coverage Comparison



a



b

a) DMS Simulation – 100% coverage across all bins

b) AMS Simulation – partial coverage due to schematic-level effects

Conclusions

- Adapted digital coverage strategies for mixed-signal environments
- Unified MS-Sampler enables reusable access to analog signals
- Consistent coverage across DMS and AMS platforms AMS captures real-world effects
- Validates both functional intent and schematic integrity during pre-silicon verification

Questions

Thank you!