2024 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION UNITED STATES

SAN JOSE, CA, USA
MARCH 4-7, 2024

Enabling True System-Level Mixed-Signal Emulation

Nimay Shah[1], Paul Wright[1], Pranav Dhayagude[1]
Raj Mitra[2], Adam Sherer[2]

[1]Analog Devices, Inc. [2]Cadence Design Systems, Inc.

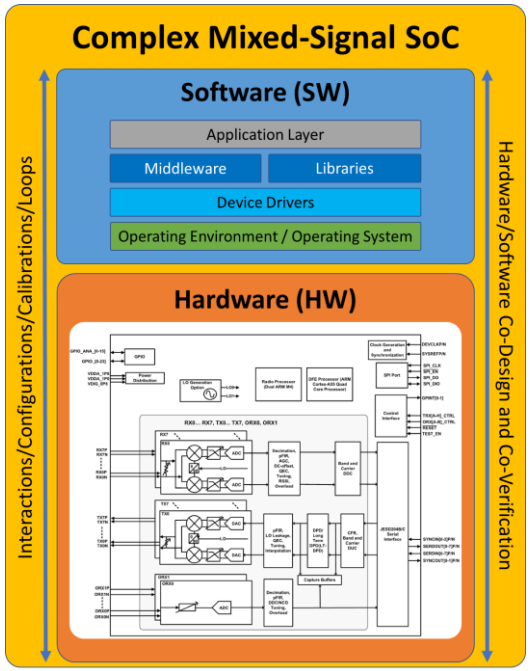
ANALOG DEVICES

cadence

accellera SYSTEMS INITIATIVE

Motivation

- Modern day SoCs = Intricate HW + Complex SW stack
  - Analog increasingly controlled/parameterized by digital
  - Pre-Silicon HW+SW co-design and co-verification is a must
- Simulation runtimes are prohibitive for HW+SW DV
  - Milliseconds/seconds of realtime takes days/weeks in simulation
  - Emulation commonly deployed to address this problem
  - Emulation of highly mixed-signal systems is challenging
- AMS blocks modeled as Real Number Models (RNMs)
  - Commonly black-boxed or recoded for Emulation
  - Recoding or adopting domain specific languages → effort+++



How can *True Mixed-Signal Emulation* be enabled?

Technical Challenges and Objectives

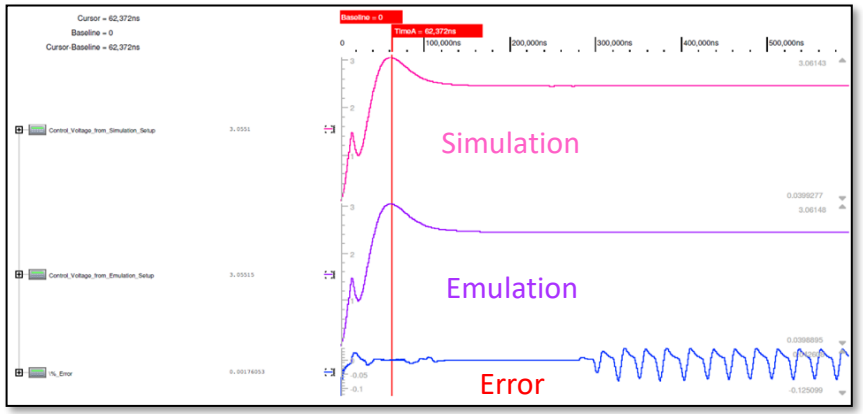
- Enable Digital Mixed-Signal (DMS) Emulation via synthesizable RNMs
  - Seamlessly retarget RNMs built for simulation to emulation platform
- Enhance compiler to support:
  - Real datatypes, User-Defined Nettypes (UDNs), Complex floating-point operations
  - # delays (procedural and continuous assignment)
  - Datatype coercion
- Provide coding guidelines to overcome emulation limitations
- Prove approach on real-life, complex mixed-signal SoC

***Emulation Ecosystem Innovation* for DMS Synthesis**

Results – Charge Pump PLL (CP-PLL)

- Vehicle to test enhanced compiler
- Filter is a key mixed-signal block
  - Implemented as $H(s)$ xfer function
- Feedback modeling, # delays
- $wrealsum$ for CP current modeling
- Datatype coercion required
- Convert dynamic arrays to static
- PLL locks successfully in emulation

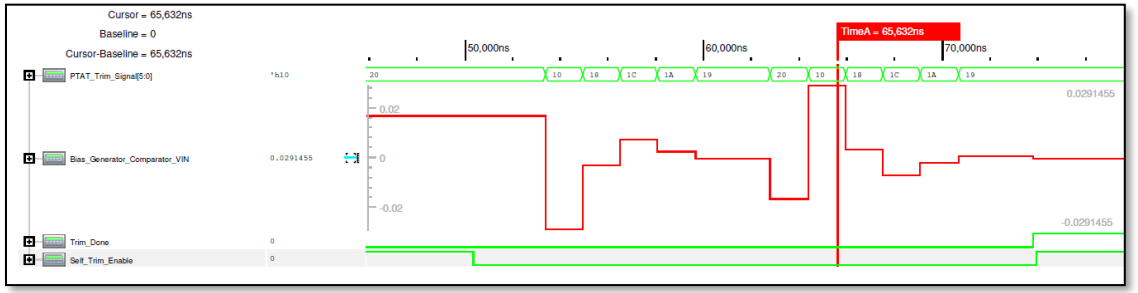
```
// Library - pll, Cell - filter, View - schematic_behav

module filter (
output wrealsum  out,
input  wrealsum  in );

wrealsum i0_0;

ccvs  #( .a_sv('{1, 44e-6}), .b_sv('{0, 22.22, 146.08e-6, 193.6e-12}),
     .dim(1), .k("1e9"), .s2z_fs("1G"), .source_type("s-domain"), .AP(2), .BP(4))
   e0 (.NIN(nin), .NOUT(i0_0), .PIN(in), .POUT(out));
gn_vref  #( .voltage("0"), .conn(0))  i0 ( .O(i0_0));

endmodule
```



Error < 0.04% vs. simulation!

Results – 4T4R Transceiver SoC

- 4T4R transceiver with ARM cores
- Current emulation approach – (1)
  - Black-box all AMS, Sim-XL with AVIPs
- White-box bias generator RNM
  - Integrated into system-level emulation env
- Self-trim via reg write by ARM core
- 33% throughput impact vs. (1)



320x speedup vs. simulation!

Conclusions

Synthesize RNMs Directly, no Black-Boxing or Recoding required!

Successfully Emulate Complex, Mixed-Signal HW+SW Systems!

Comprehensive Pre-Silicon HW Verification and SW Validation!