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.

2024 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION UNITED STATES

SAN JOSE, CA, USA

MARCH 4-7, 2024

INTRODUCTION

- Modern day SoCs = Complex Software stack + Intricate Hardware
- Pre-Silicon HW+SW co-design and co-verification is a must
- Pure simulation runtimes are prohibitive for HW+SW co-verification
- Emulation commonly deployed to address this problem
- Emulation of highly mixed-signal silicon systems is challenging
- Analog/Mixed-Signal modeled as Real Number Models (RNMs)
- Commonly black-boxed or recoded for Emulation [1], [2], [3]
- Black-boxing imposes severe limitations on what can be verified
- Recoding or adopting domain specific languages means extra effort

How can True Mixed-Signal Emulation be enabled?

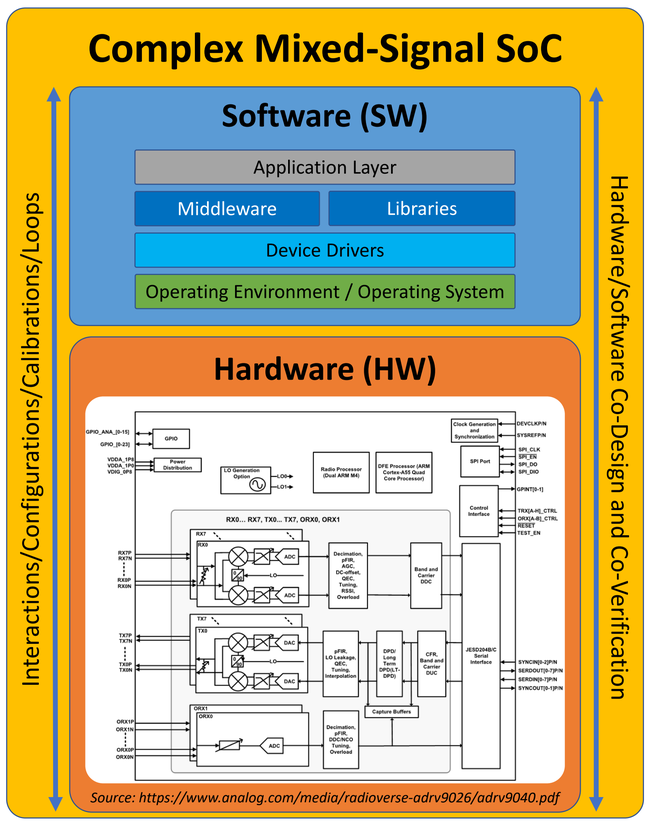

Figure 1. Typical Modern SoC – SW Stack + Highly Mixed-Signal HW

TECHNICAL CHALLENGES & 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)
  - # delays (procedural and continuous assignment)
  - Datatype coercion
  - Complex floating-point operations
- Provide coding guidelines to overcome emulation limitations
- Minimize emulation area overhead and throughput impact
- 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 and DMS support
- Filter implemented as complex $H(s)$ transfer function with poles and zeros via bilinear transform
- Feedback modeling, # delays in PFD block
- $wrealsum$ UDN for charge pump current modeling
- Datatype coercion supported, enabling netlist reuse
- Convert dynamic objects to static – coefficient arrays
- Required additional compiler arguments and options
- PLL locks successfully in emulation
- VCO Control Voltage, V_c, plotted in fig 4.

Error < 0.04% vs. simulation

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

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

wrealsum i0_O;

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_O), .PIN(in), .POUT(out));
gn_vref  #( .voltage("0"), .conn(0))  i0 ( .O(i0_O));

endmodule
```

Figure 2. Code snippet from the PLL showing the loop filter (filter) SV-RNM

```
module pll (  );

wire vss, i4_out, vc;  /*don't need to be declared wrealsum as coercion is supported*/
bit ckin ;
wire logic ckdiv ;
wire logic ckout ;
wire up, i0_OB, down ;

pfd i1 ( .down(down), .up(up), .ckdiv(ckdiv), .ckin(ckin));
vco i2 ( .out(ckout), .vc(vc));
filter i3 ( .out(vc), .in(i4_out));
cpump i4 ( .out(i4_out), .down(down), .up(up));
divby32 i5 ( .ckout(ckdiv), .ckin(ckout));
lclock i0 ( .ckin(ckin), .i0_OB(i0_OB));
gn_vref  #( .voltage("0"), .conn(0))  i7 ( .O(vss));

`ifdef IXCOM_COMPILE
IXCclkgen #(500) ckg(mclk);  // 1ns CAKE master clock
`endif

endmodule
```

Figure 3. Code snippet showing top-level PLL netlist

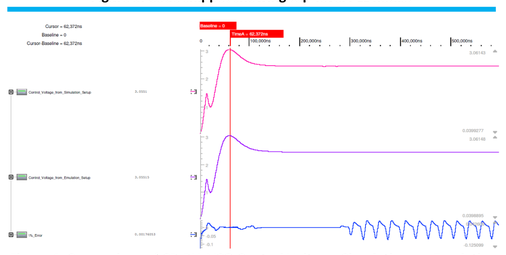

Figure 4. Comparison of CP-PLL VCO Control Voltage, Simulation vs. Emulation

RESULTS – 4T4R ORAN TRANSCEIVER

Figure 5. Transceiver bias generator, Simulation vs. Emulation

Figure 6. Simulation results for the bias generator self-trim routine

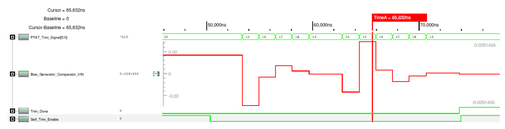

Figure 7. Emulation results for the bias generator self-trim routine

- 4T4R transceiver with application class ARM cores
- Current emulation approach - Black-box all AMS, Simulation Acceleration mode with AVIPs [4]
- White-box bias generator block (hierarchical)
- RNMs from ADI library, Extracted LUT, Hand-coded SV
- Validated model on all platforms vs. transistor circuit
- Integrated model into system-level emulation env
- Self-trim initiated via register write by ARM core
- Results align between simulation and emulation
- 33% throughput impact – within tolerance

320x speedup vs. simulation!

CONCLUSIONS

- **Innovations in compiler, emulation hardware, coding guidelines enable true DMS synthesis**
- Synthesize RNMs to floating-point fidelity with DMS features without recoding
- CP-PLL effectively pipe-cleaned new solution, demonstrates synthesized RNM fidelity
- Transceiver SoC tested out new approach in production on real-life SoC
- 320x speed-up on bias generator use-case with throughput impact well within tolerance
- Enhanced ADI behavioral modeling library targeted at both simulation and emulation
- Limitations: CM insertion, dynamic objects, Verilog gate primitives
- No new methodologies or tools - all fits within current, well-established solutions

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!

FUTURE WORK

- Continue to enhance ADI behavioral modeling library
- Extend compiler support and automation for more RNM constructs and DMS techniques
- Apply DMS emulation on more complex use-cases
- Enhance solutions to minimize throughput impact

REFERENCES

[1] F. A. Nothaft et. al., "Pragma-based floating-to-fixed point conversion for the emulation of analog behavioral models," in 2014 IEEE/ACM International Conference on Computer-Aided Design (ICCAD), San Jose, CA, USA, 2014.

[2] S. Herbst, G. Rutsch, W. Ecker and M. Horowitz, "An Open-Source Framework for FPGA Emulation of Analog/Mixed-Signal Integrated Circuit Designs," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, July 2022.

[3] R. Kolker, G. Best and P. Len Orlando III, "Mixed-Signal Emulation Digital Twin for Defense Applications," in CadenceLive, Burlington, MA, USA, Sept. 2022.

[4] "Accelerated VIP," Cadence Design Systems, Inc., [Online]. Available: https://www.cadence.com/en_US/home/tools/system-design-and-verification/verification-ip/accelerated-vip.html.

Contact: nimay.shah@analog.com, asherer@cadence.com

The authors would like to acknowledge the contributions of various people on teams at Analog Devices, Inc. and Cadence Design Systems, Inc. that made this possible.

© Accellera Systems Initiative