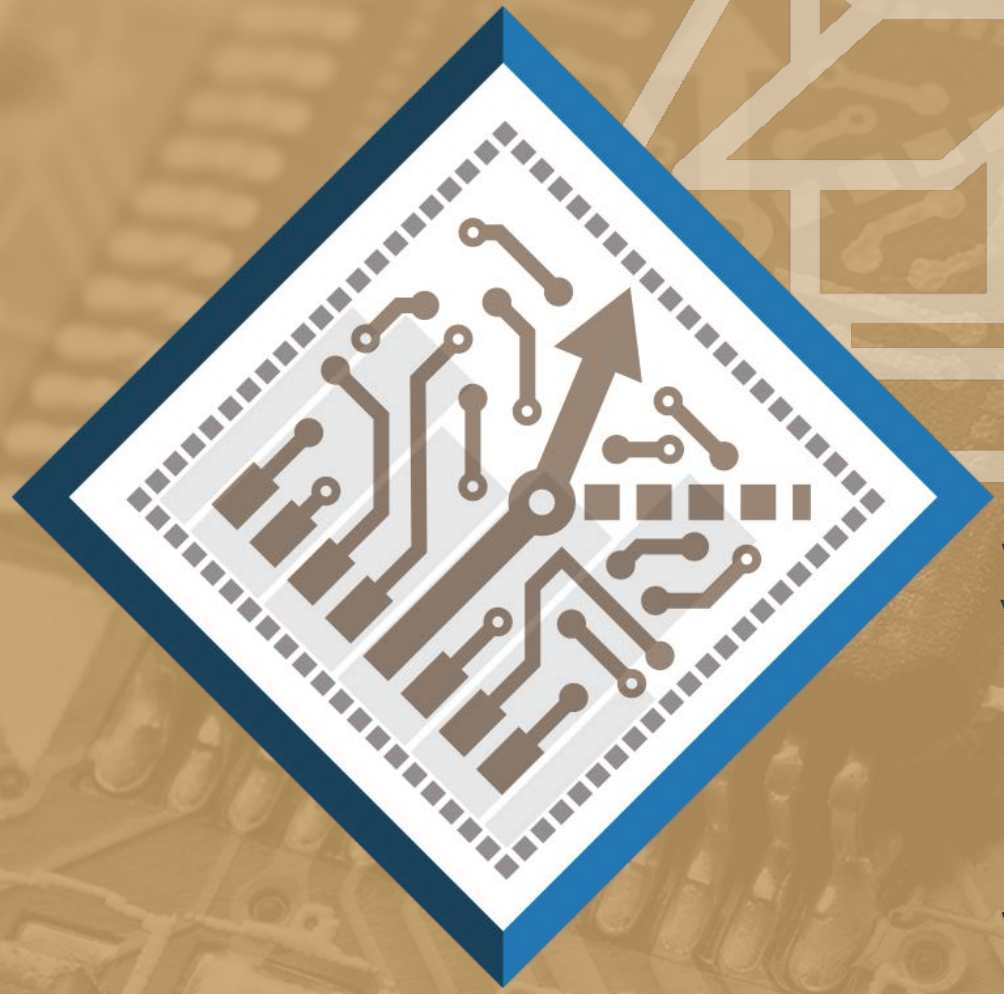
CLARK BARRETT

STANFORD UNIVERSITY



UPSCALE:

SCALING UP VERIFICATION FOR OPEN SOURCE HARDWARE

TEAM

Clark Barrett

CS Department,
Stanford University

Expertise in constraint solving and formal verification

Co-founder of Satisfiability Modulo Theories (SMT) research area

ACM distinguished scientist; Haifa Verification Conference Award; IBM Software Innovation award

Aarti Gupta

CS Department,
Princeton University

Expertise in formal verification, program analysis, decision procedures

Led industry research dept for 10 years (NEC Labs)

Fellow of ACM; three NEC technology commercialization awards

Mark Horowitz

CS/EE Departments
Stanford University

Expertise in analog and digital design

High-speed I/O in industry (founder of Rambus Inc)

Fellow of IEEE and ACM; Natl Academy of Engineering; American Academy of Arts and Science; Don Pederson IEEE Technical Field Award

Sharad Malik

EE Department
Princeton University

Expertise in digital design, propositional satisfiability (SAT)

Award-winning SAT solver (Chaff) widely used in research and industry

Fellow of IEEE and ACM; DAC most-cited paper; CAV award; ACM/IEEE Technical Impact Award in EDA

Subhasish Mitra

CS/EE Departments
Stanford University

Expertise in robust computing, design, validation, and test

X-Compact test compression widely used in industry

Fellow of IEEE and ACM; SRC Technical Excellence Award; Intel Achievement Award; ACM/IEEE Technical Impact Award in EDA

RESEARCH PROGRAM

POSH UPSCALE

TA – 1, L1

Verification Techniques

- **Symbolic QED**
- **A-QED**
- **ILA-based Verification**

Foundational Technologies

- Model Checking
- ILA Models
- Mixed-Signal Models

TA – 1, L3

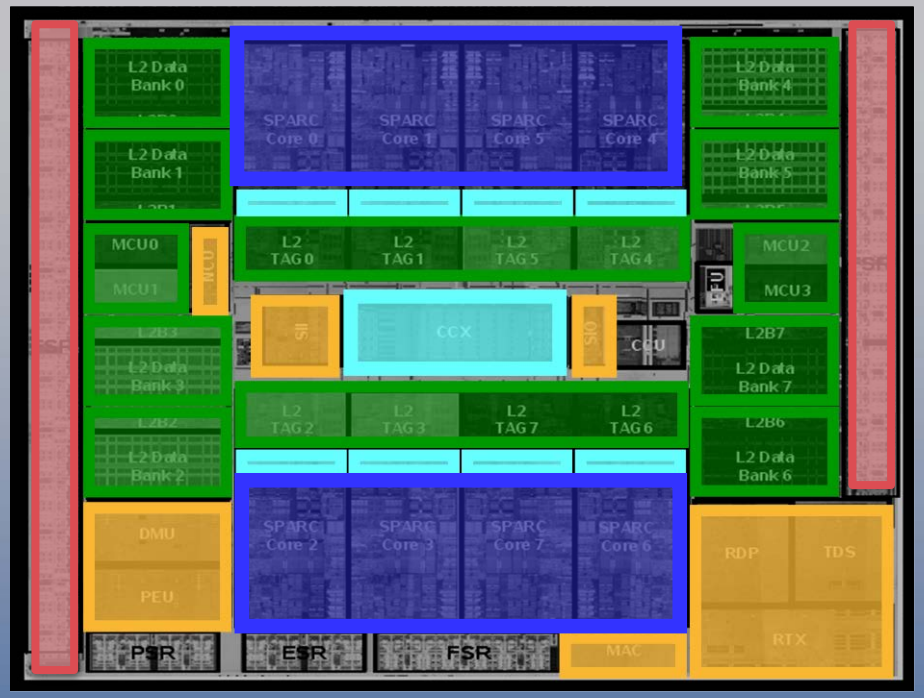
- **E-QED**
  For System Prototyping

TA – 2

- **Open Source High-speed Phy**

SOC VERIFICATION

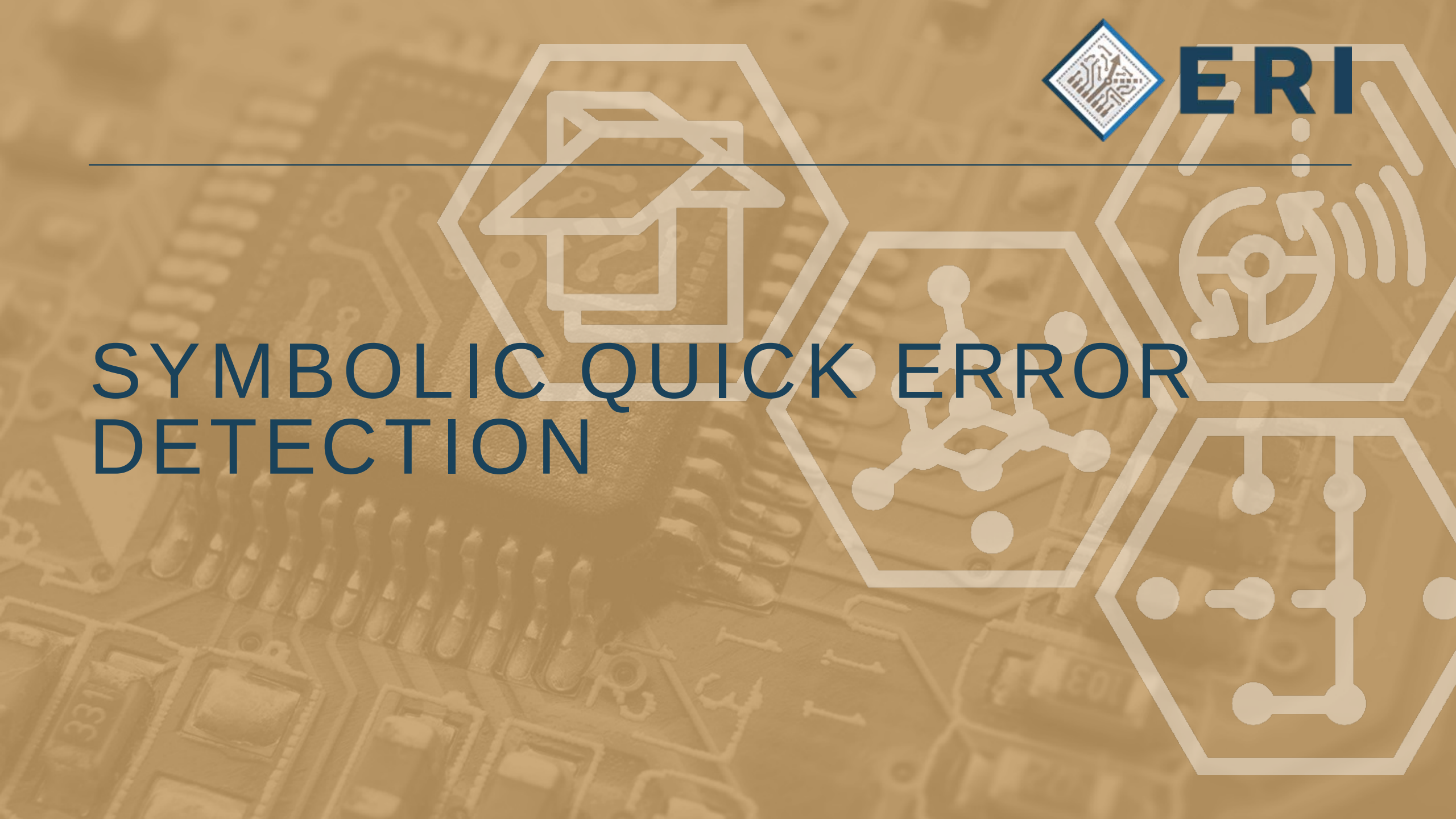
System on Chip



Processor cores

Uncore, accelerators

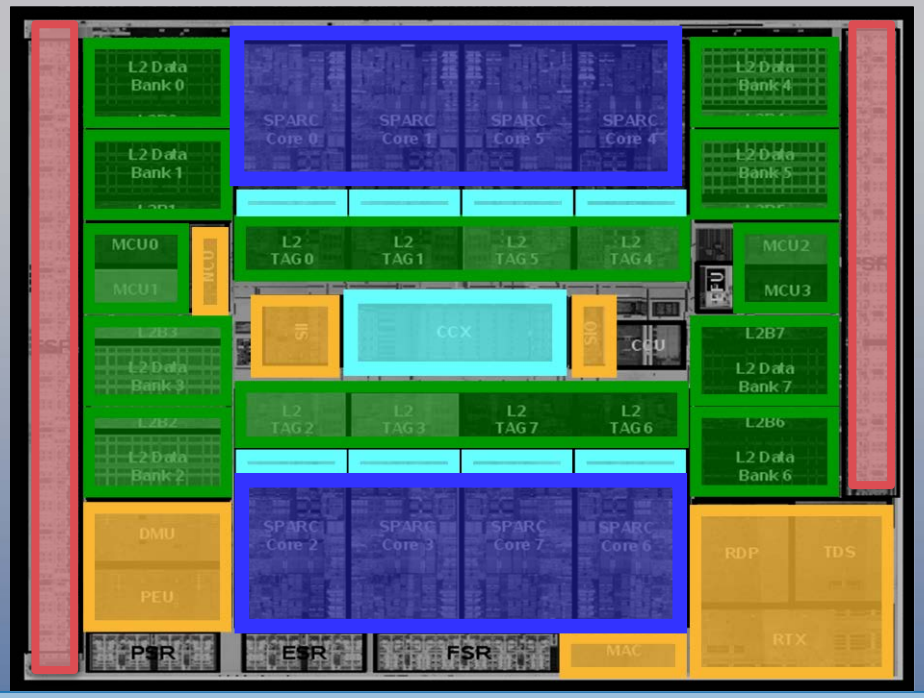
Analog / Mixed Signal



SYMBOLIC QUICK ERROR DETECTION

SOC VERIFICATION: SYMBOLIC QED

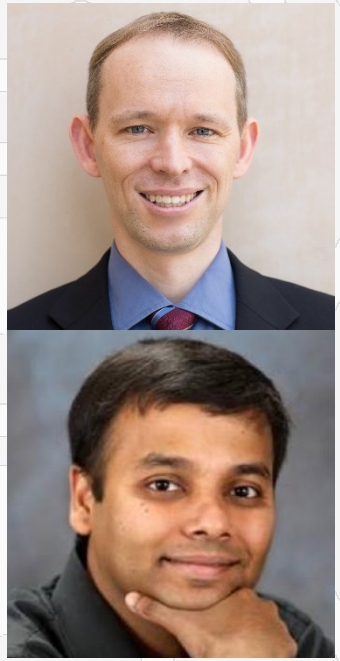
System **o**n **C**hip



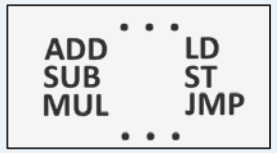
Processor cores

Uncore, accelerators

Analog / Mixed Signal



SYMBOLIC QED



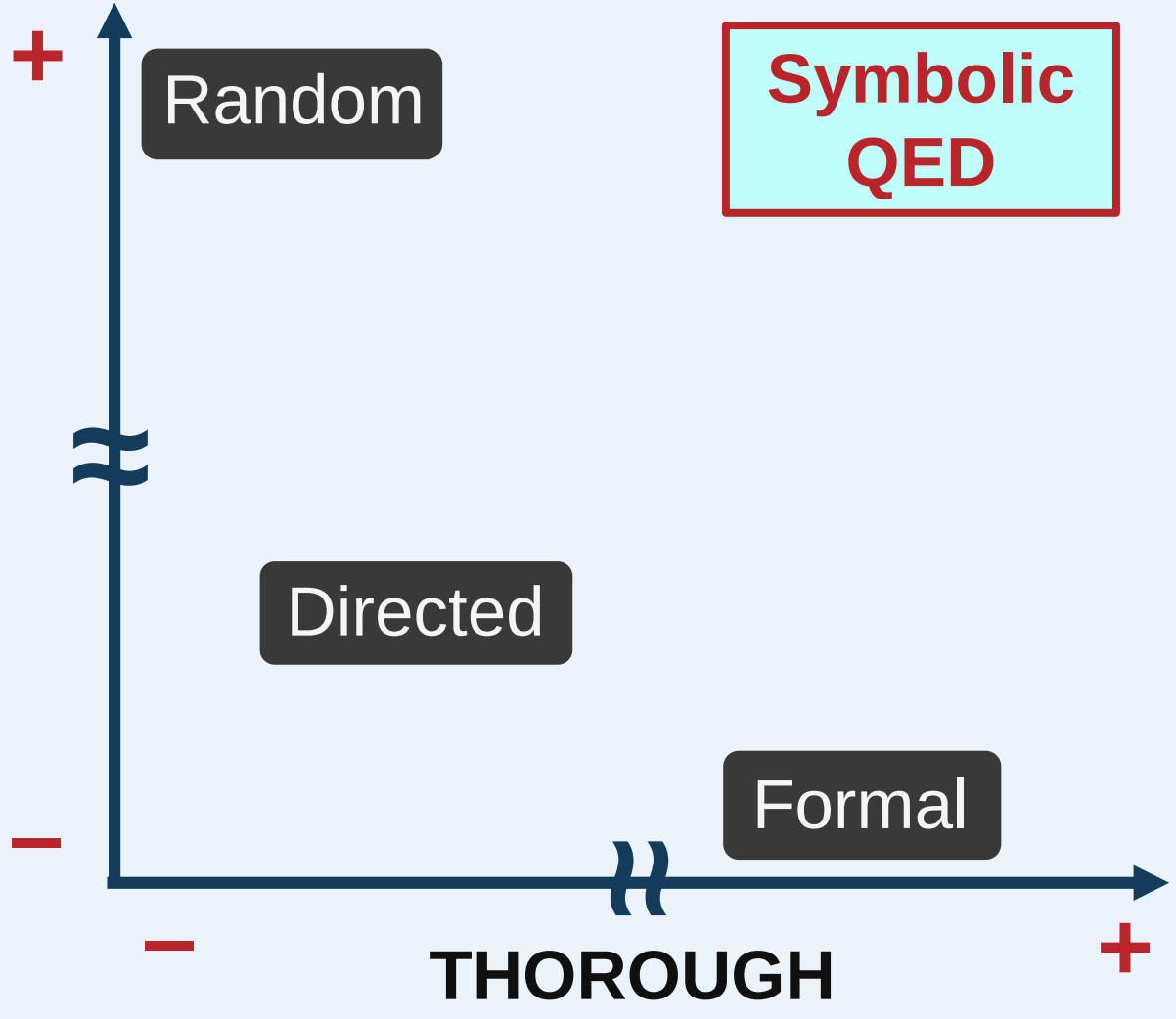
Design
+
Instruction Set Architecture

Symbolic QED

Uses model checking

Detected bugs

Processors, accelerators,
Billion-transistor chips

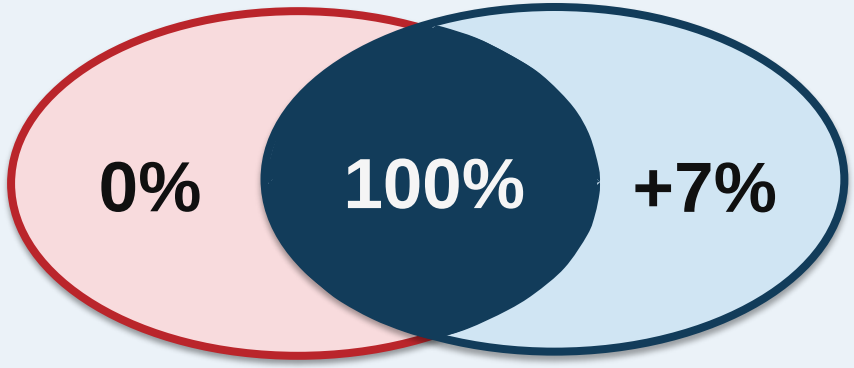


SYMBOLIC QED: DRASTIC BENEFITS

INFINEON case study: 16 automotive IP versions verified over 5 years

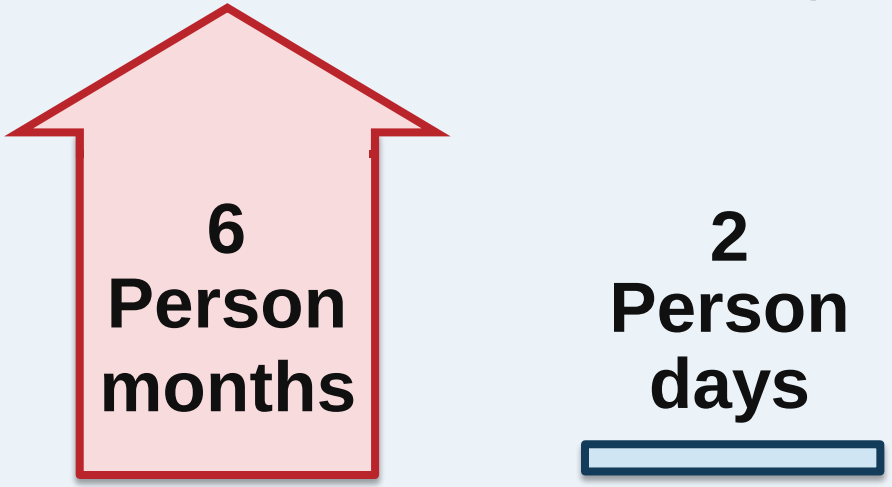
1. Extremely Thorough

Detected bugs & spec. errors



Industry Flow **Symbolic QED**

2. 60X Productivity

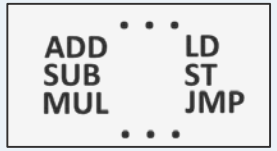


Industry Flow **Symbolic QED**

Hardware security applications

1. Derive new attacks (beyond Spectre, Meltdown)
2. Detect Trojans

SYMBOLIC QED



Design
+
Instruction Set Architecture

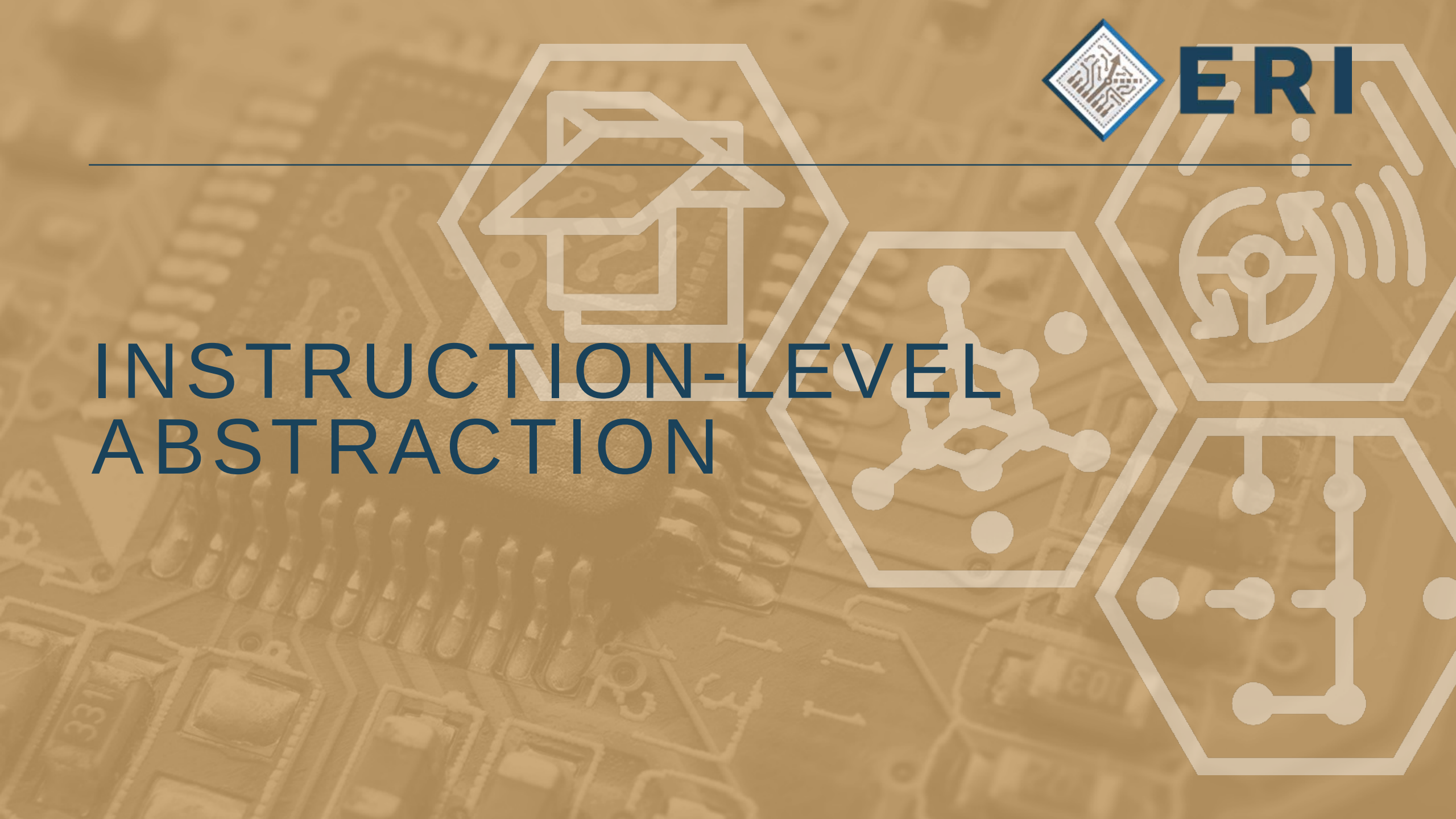
Symbolic QED

Uses model checking

Detected bugs

DEMO

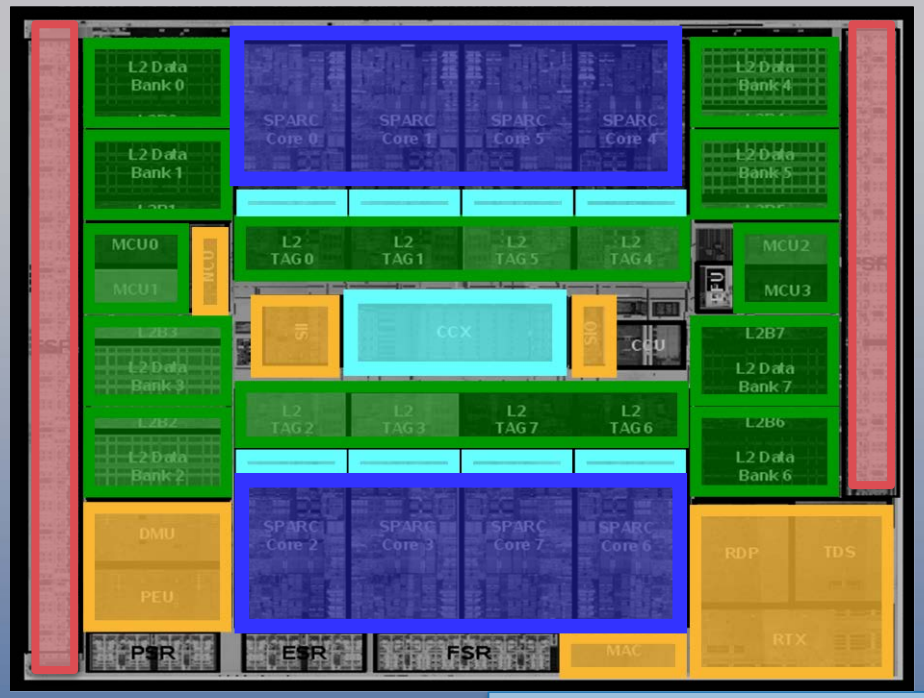
https://github.com/upscale-project/generic-sqed-der



INSTRUCTION-LEVEL ABSTRACTION

SOC VERIFICATION: INSTRUCTION LEVEL ABSTRACTION

System on Chip



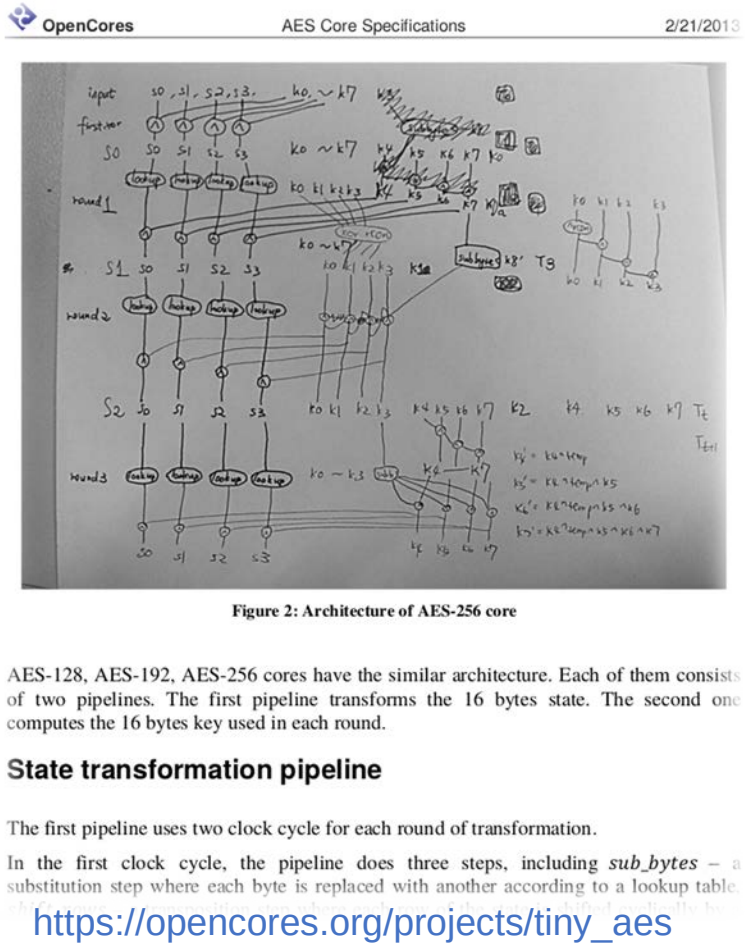
Processor cores

Uncore, accelerators

Analog / Mixed Signal

Open Source Hardware: Need for Formal Specification

AES Specification Document from OpenCores.org



OpenCores AES Core Specifications 2/21/2013

Figure 2: Architecture of AES-256 core

AES-128, AES-192, AES-256 cores have the similar architecture. Each of them consists of two pipelines. The first pipeline transforms the 16 bytes state. The second one computes the 16 bytes key used in each round.

State transformation pipeline

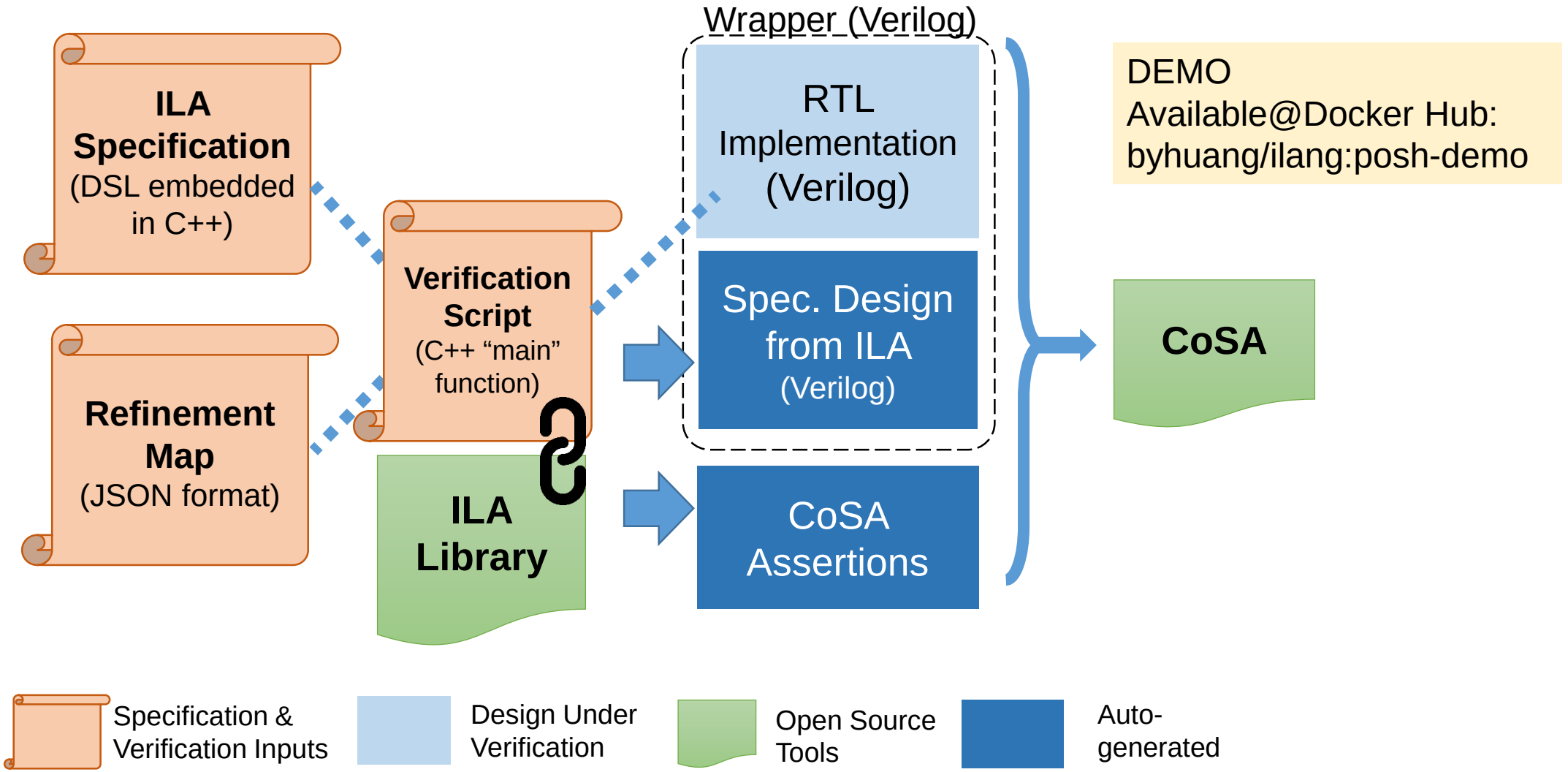
The first pipeline uses two clock cycle for each round of transformation.

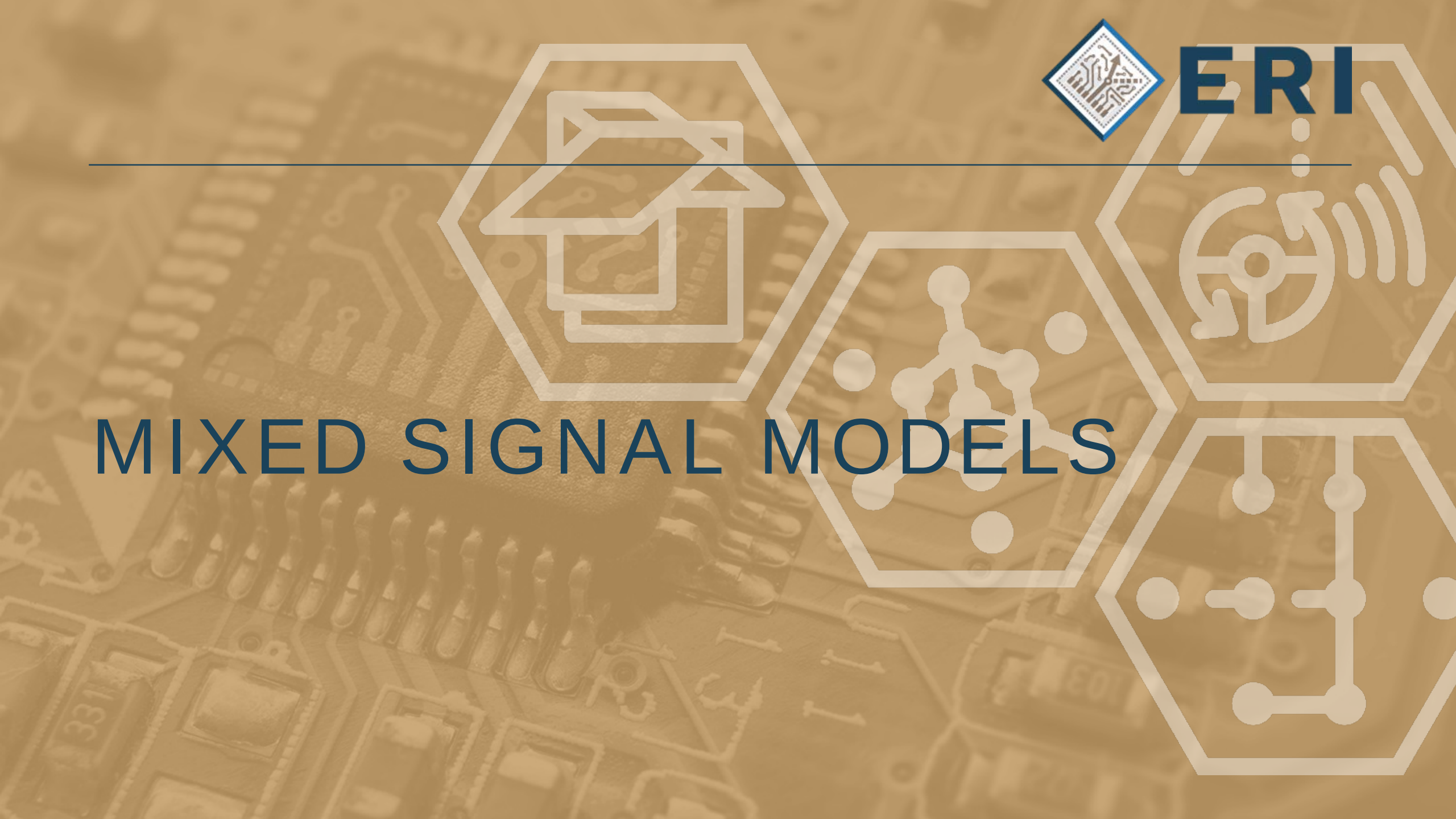
In the first clock cycle, the pipeline does three steps, including *sub_bytes* – a substitution step where each byte is replaced with another according to a lookup table.

https://opencores.org/projects/tiny_aes

- Informal, ambiguous, possibly incomplete
- Limits implementation verification, limits adoptability
- Instruction-Set Architecture (ISA) works for processors – beyond processors?
- Instruction-Level Abstraction (ILA)
  - Generalizing ISAs to beyond processors

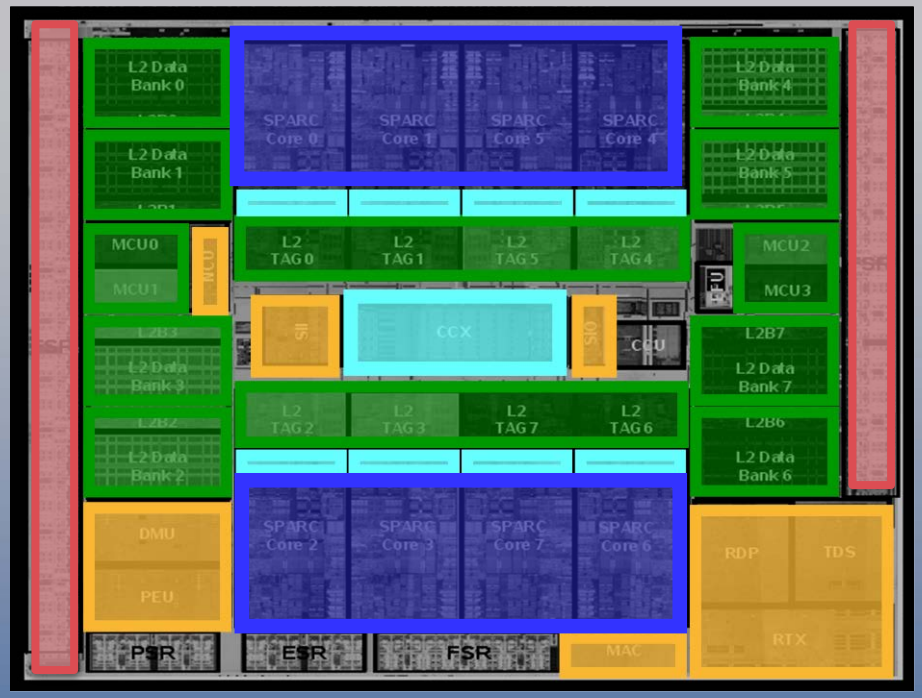
Open Source Tool Flow




ERI
MIXED SIGNAL MODELS

SOC VERIFICATION: MIXED SIGNAL MODELS

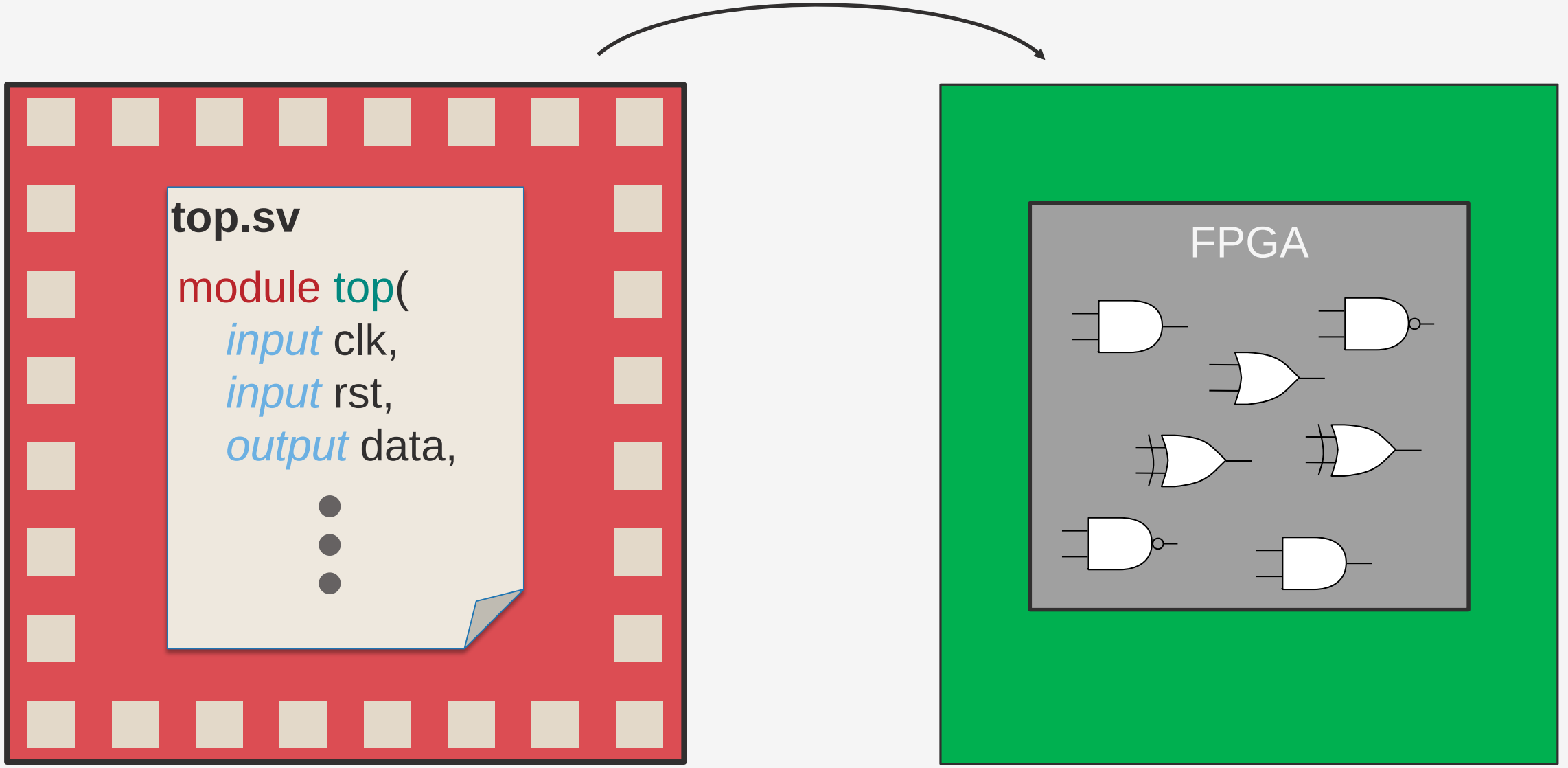
System **o**n **C**hip

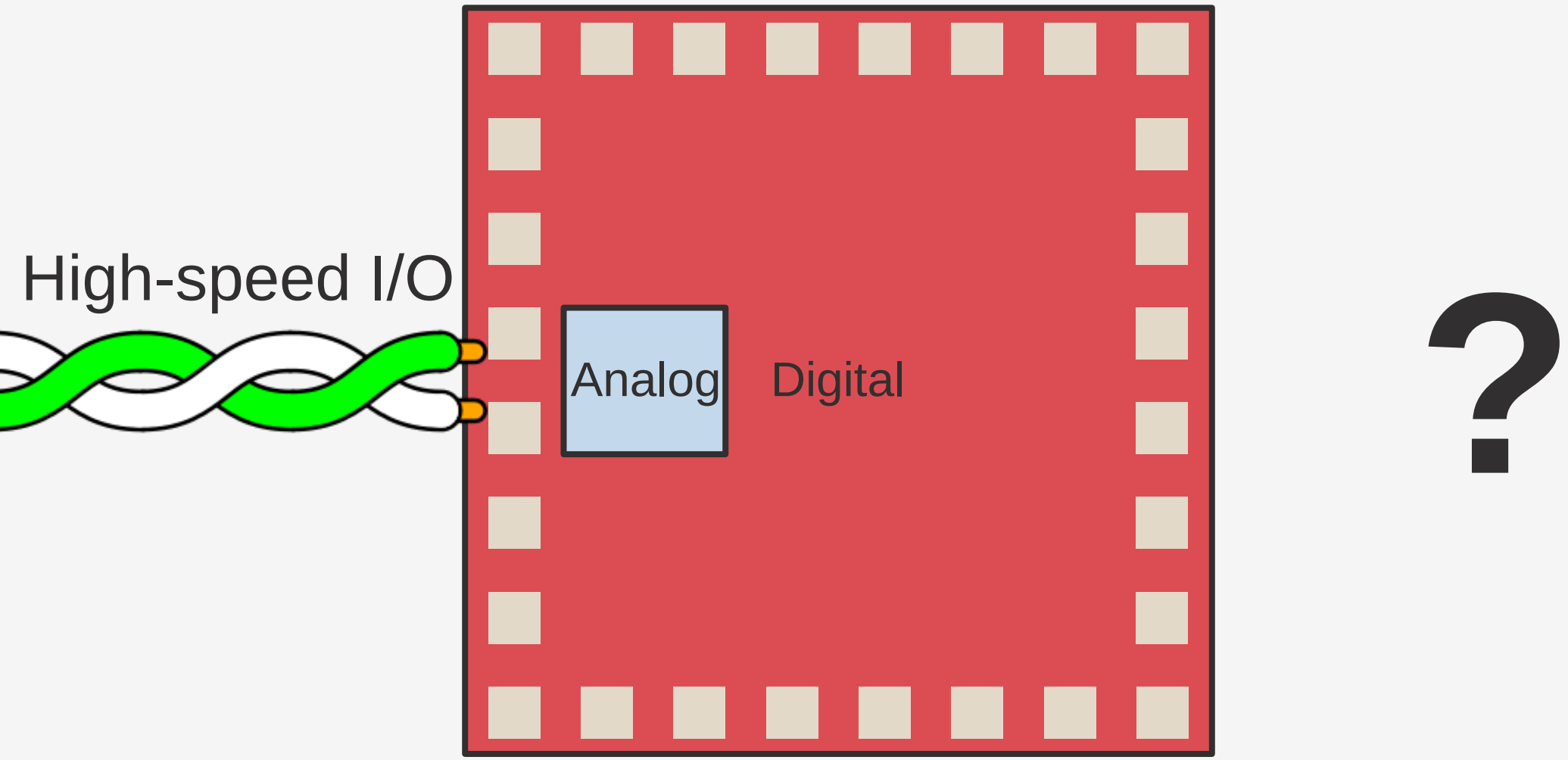


Processor cores

Uncore, accelerators

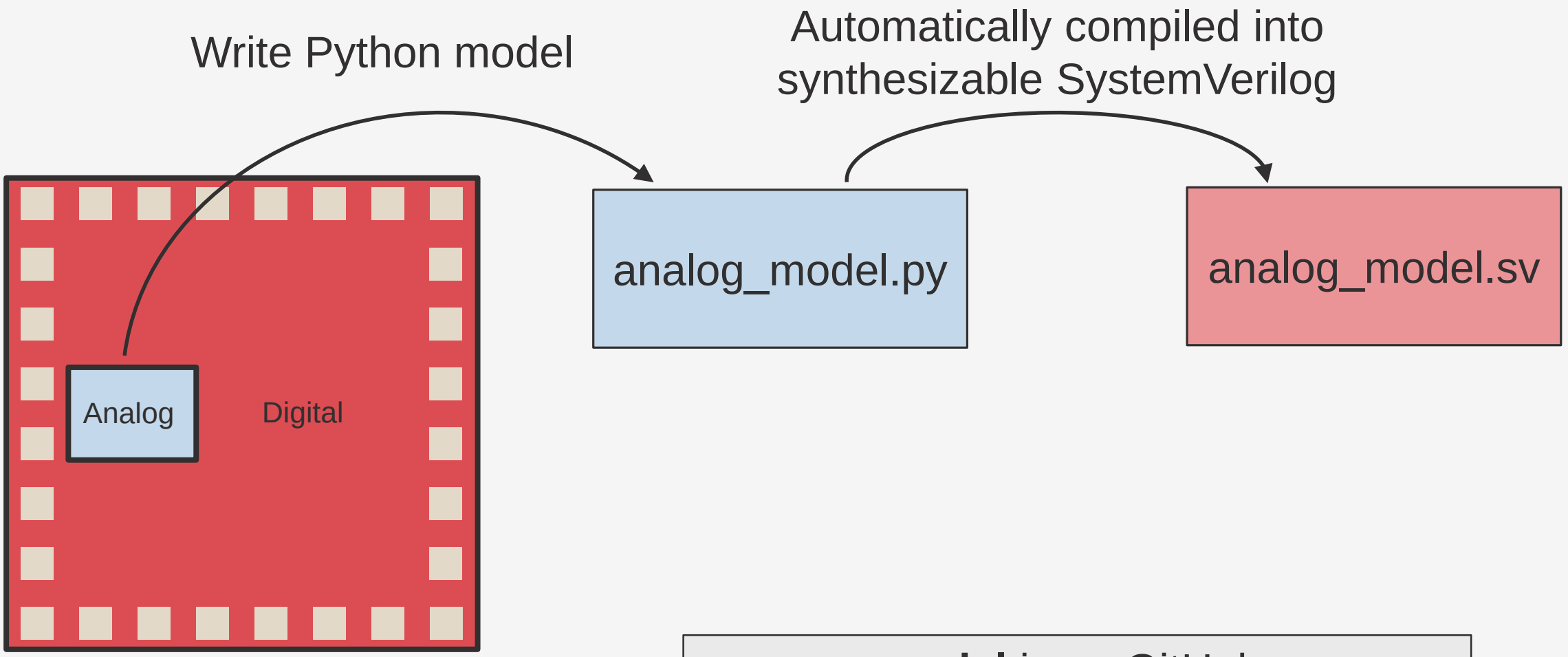
Analog / Mixed Signal


top.sv
module top(
input clk,
input rst,
output data,
FPGA


High-speed I/O
Analog
Digital

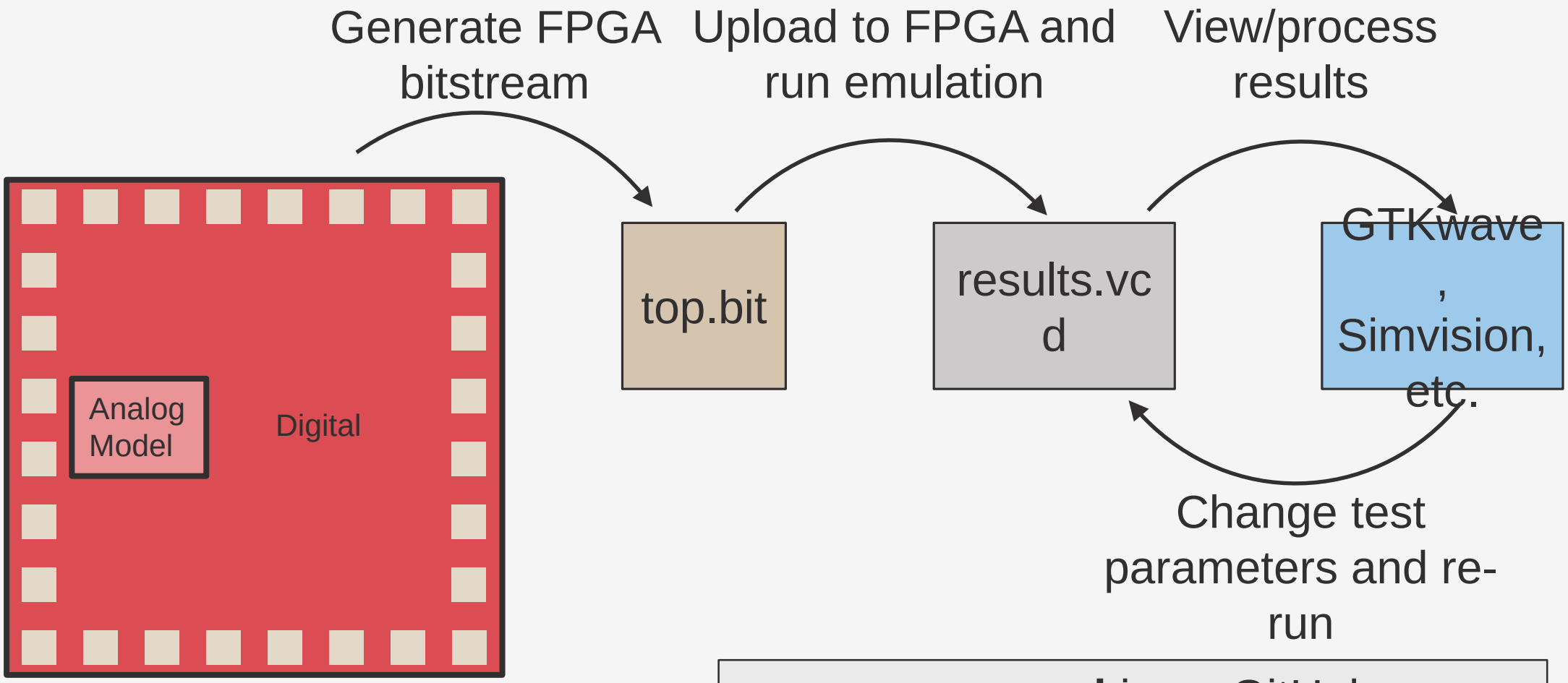
?

MSDSL



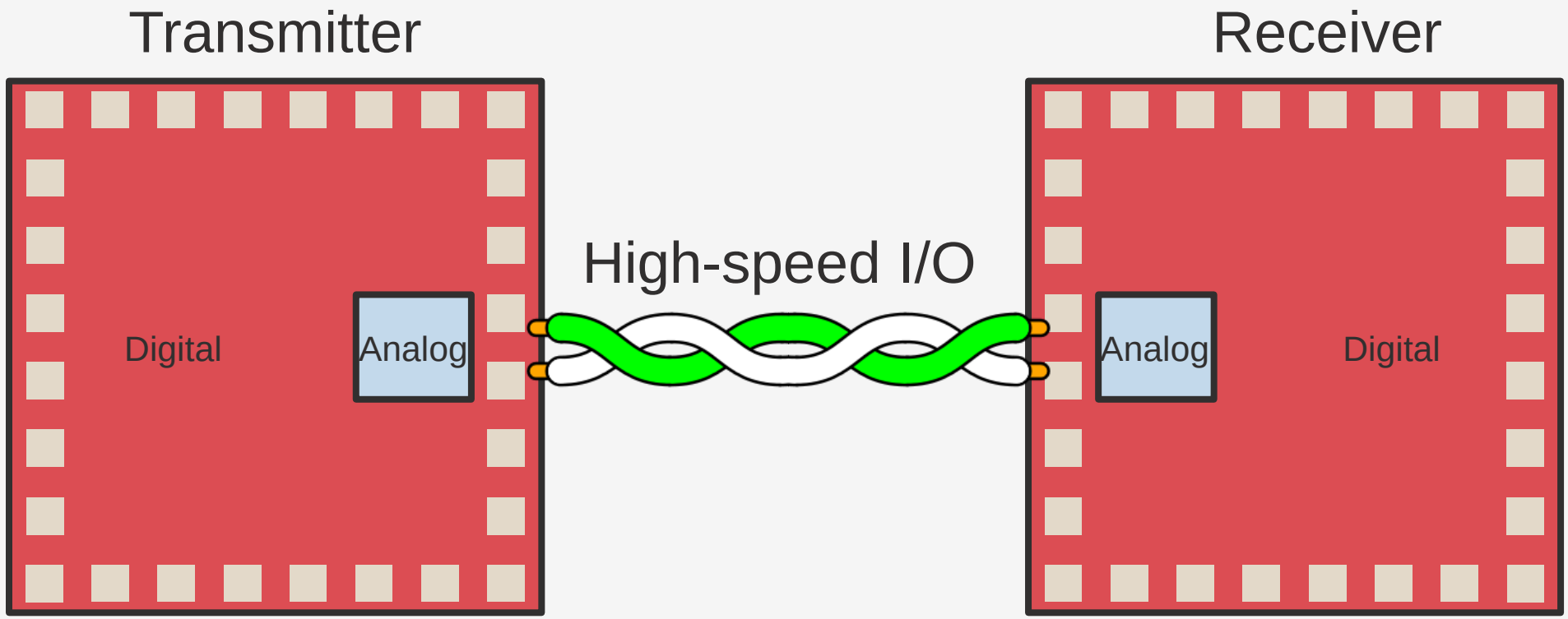
msdsl is on GitHub:
https://github.com/sgherbst/anasymod

ANASYMOD

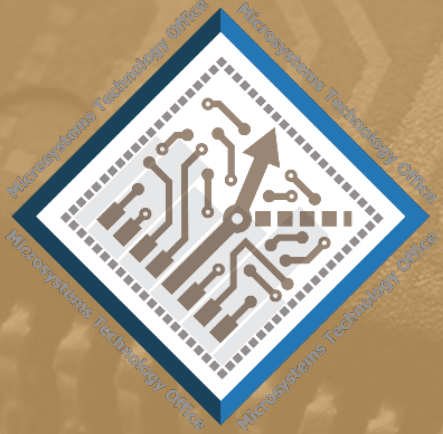


anasymod is on GitHub:
https://github.com/sgherbst/anasymod

DEMO AVAILABLE



https://github.com/sgherbst/hslink-emu



ERI ELECTRONICS RESURGENCE INITIATIVE

SUMMIT

2019 | Detroit, MI | **July 15 - 17**