RISECREEK: From RISC-V Spec to 22FFL Silicon

Vinod Ganesan, Gopinathan Muthuswamy

SHAKTI Group | CSE Dept | RISE Lab | IIT Madras

Amudhan, Bharath, Alagu - HCL Tech

Konala Varma, Ang Boon Chong, Harish Villuri - Intel Corporation

SHAKTI

Outline of the talk

- Overview
- Core Micro-Architecture
- Soc Micro-Architecture
- Verification Framework
- Code Coverage
- FPGA Emulation
- RTL to FAB
- Power On Reset Spec
- Top level Configurations
- Design For Testability
- Physical Design process

C-Class Core

Overview

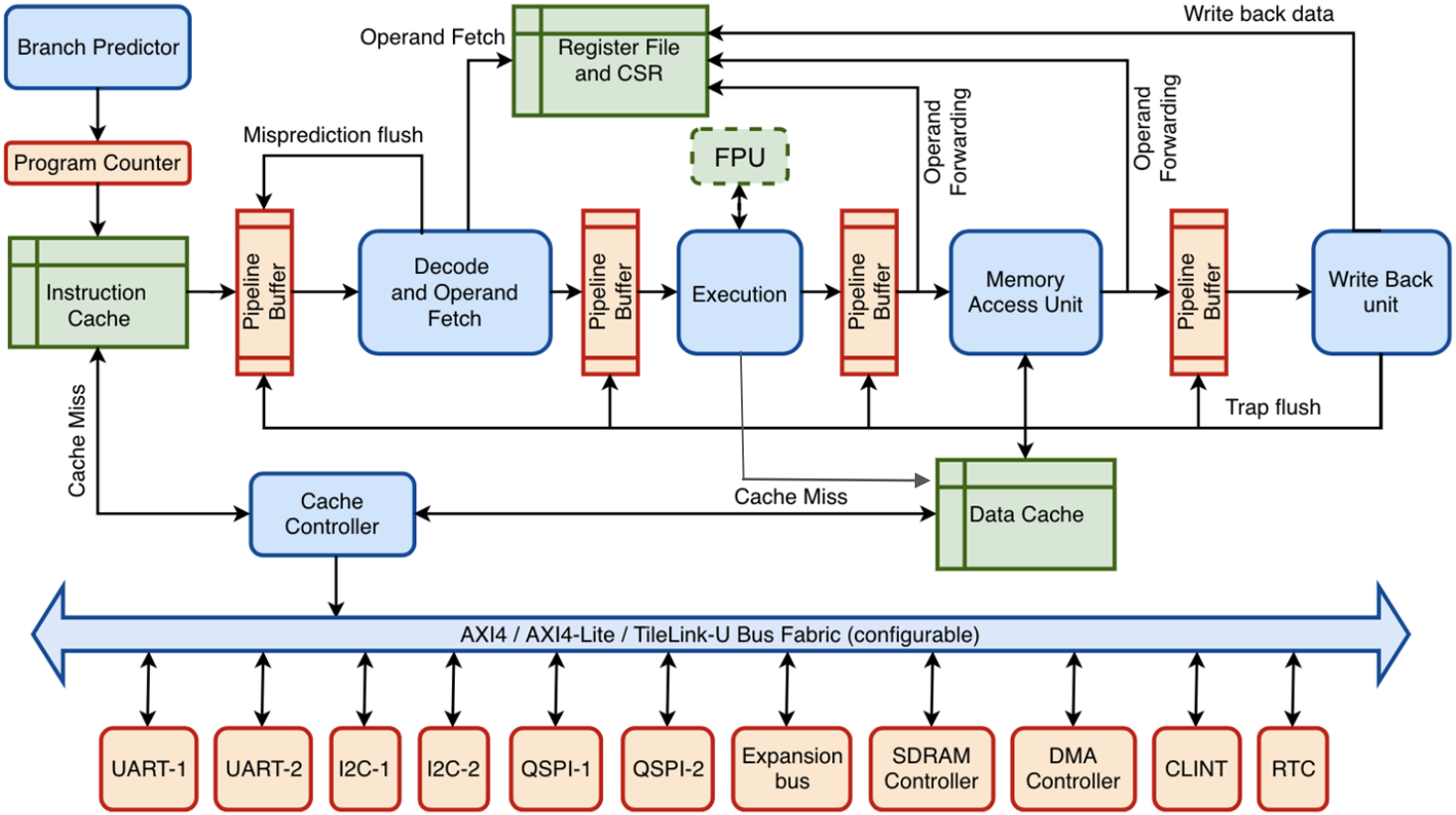
Overview

- An in-order 6-stage 64-bit microcontroller supporting the entire stable RISC-V ISA.
- Targets mid-range compute systems: 200-800MHz
- Supports RISC-V Linux, secure L4
- Variants for low-power and high-performance.
- Positioned against ARM's Cortex A35/A55
- Performance Stats:
  - Dhrystone: 1.67 DMIPS/MHz
  - CoreMark : 2.2

Specifications

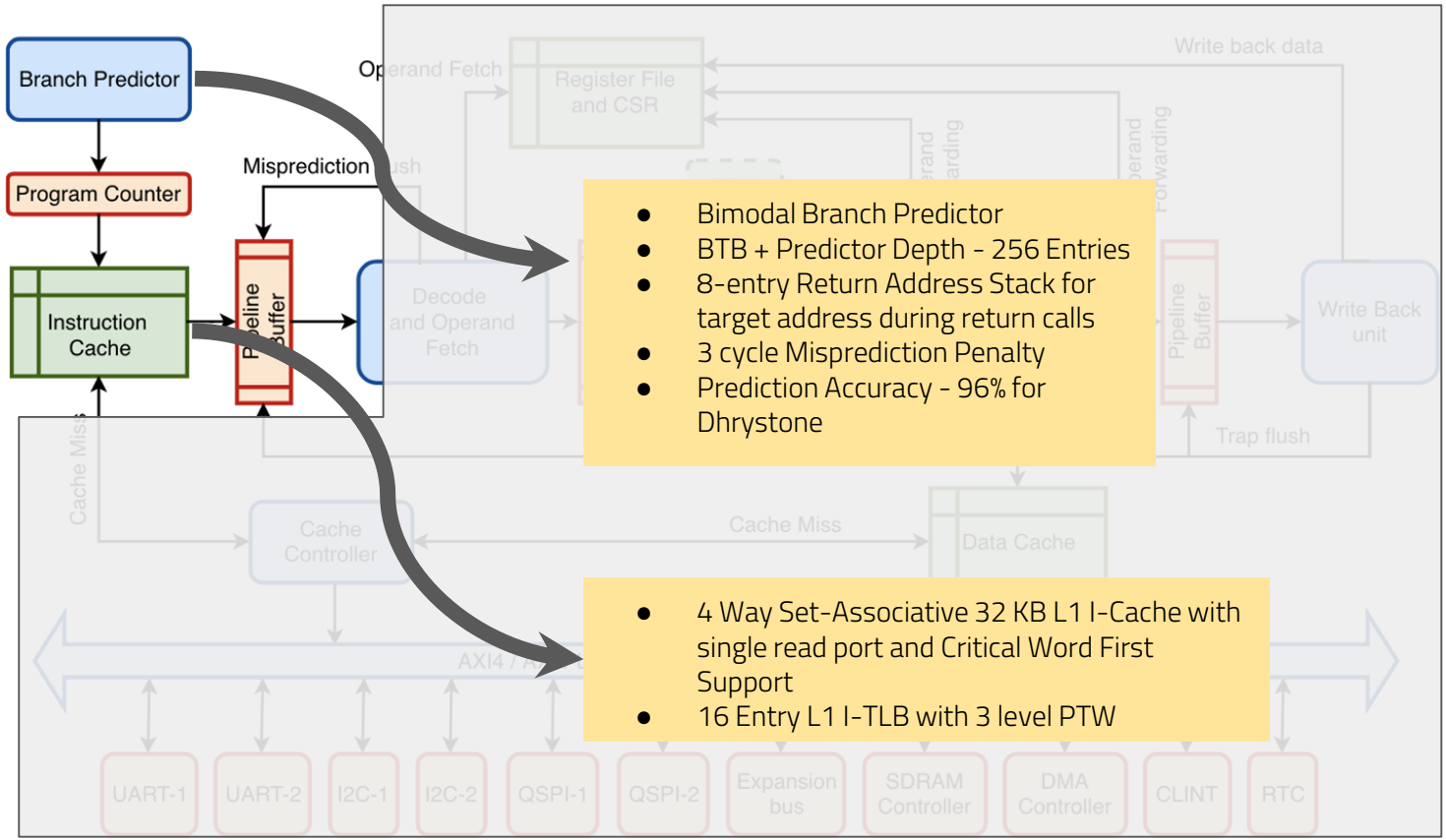
- Supports RISC-V ISA: RV64G.
- Compatible with latest privilege spec of RISC-V ISA and supports the sv39/48 virtualization scheme.
- Supports the OpenOCD based debug environment.
- Includes a High performance branch predictor with a Return-Address-Stack.
- Caches: 16-64KB non-blocking pipelined Instruction and Data caches. Optional L2

Core Micro-Architecture



SHAKTI

Fetch Stage

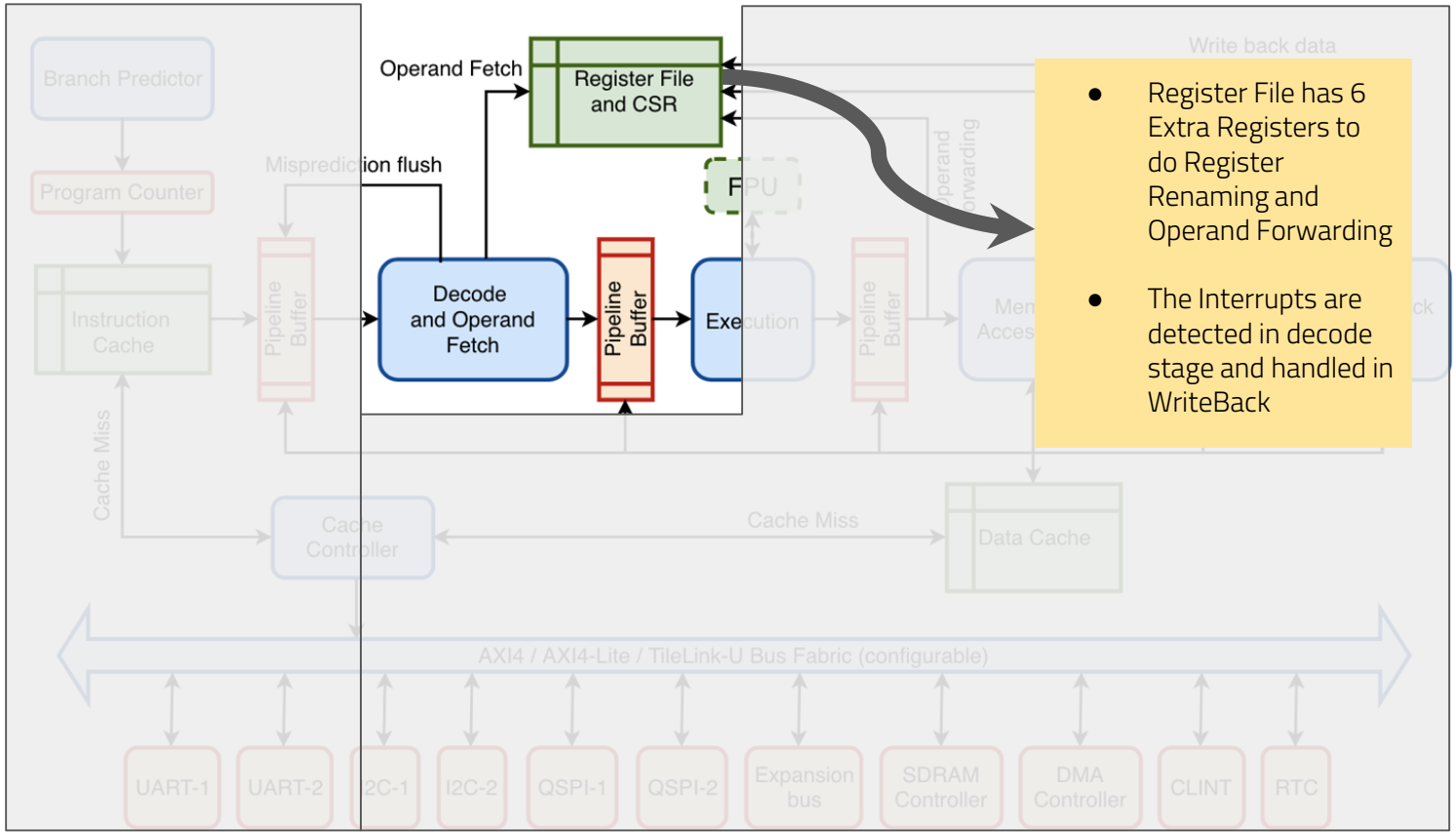


- Bimodal Branch Predictor
- BTB + Predictor Depth - 256 Entries
- 8-entry Return Address Stack for target address during return calls
- 3 cycle Misprediction Penalty
- Prediction Accuracy - 96% for Dhrystone

- 4 Way Set-Associative 32 KB L1 I-Cache with single read port and Critical Word First Support
- 16 Entry L1 I-TLB with 3 level PTW

SHAKTI

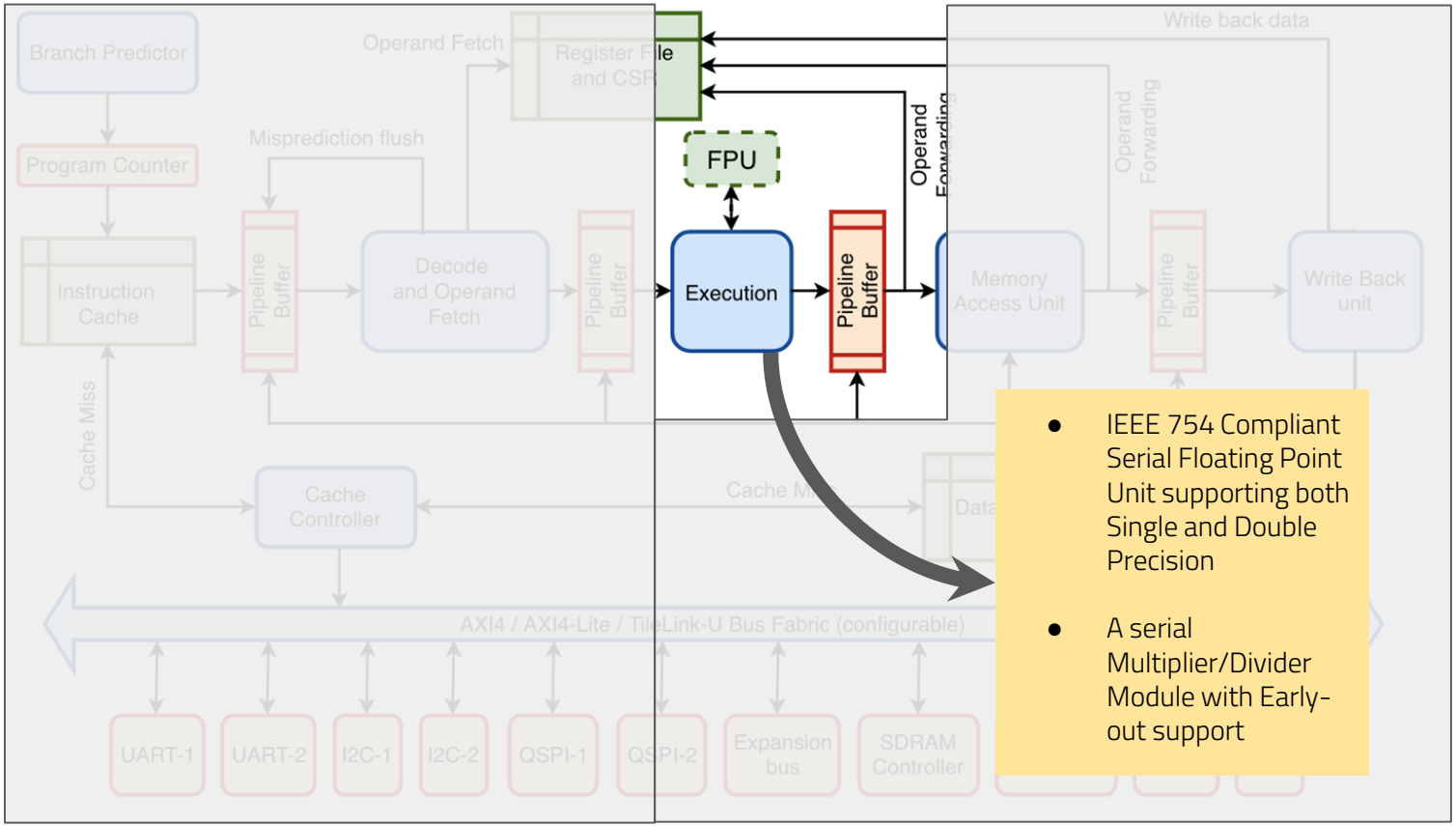
Decode & Operand Fetch



- Register File has 6 Extra Registers to do Register Renaming and Operand Forwarding
- The Interrupts are detected in decode stage and handled in WriteBack

SHAKTI

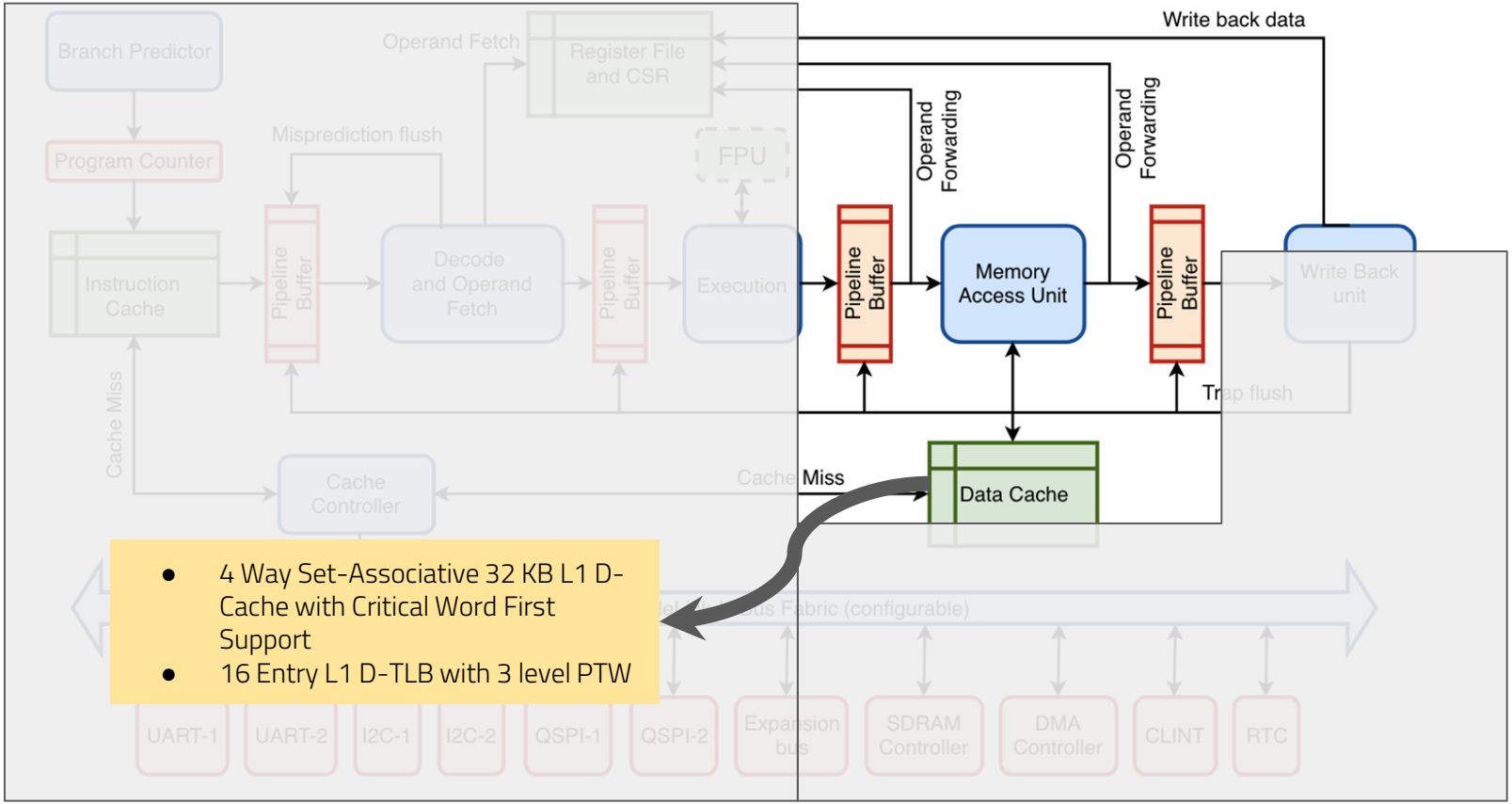
Execute Stage



- IEEE 754 Compliant Serial Floating Point Unit supporting both Single and Double Precision
- A serial Multiplier/Divider Module with Early-out support

SHAKTI

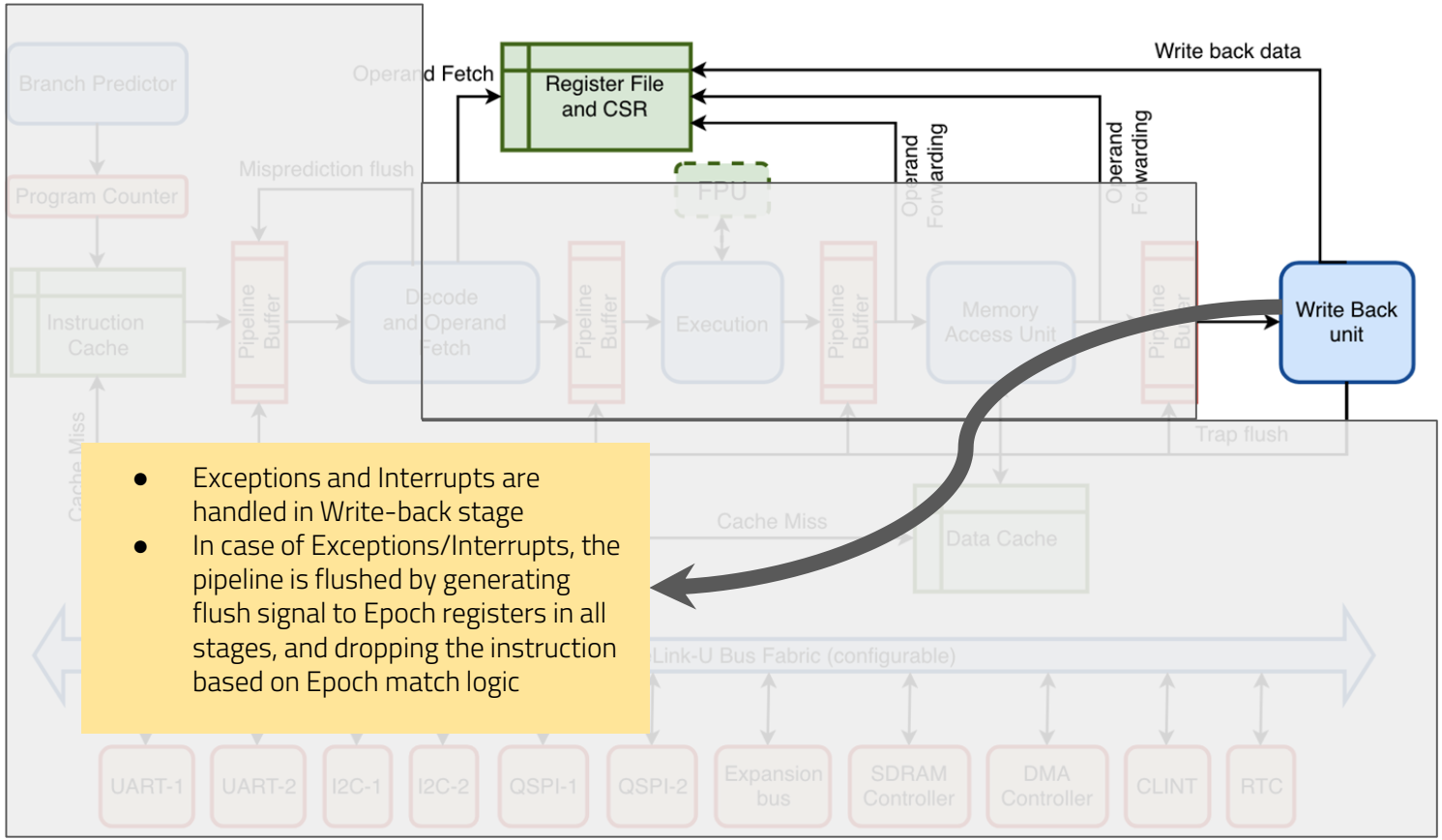
Memory Stage



- 4 Way Set-Associative 32 KB L1 D-Cache with Critical Word First Support
- 16 Entry L1 D-TLB with 3 level PTW

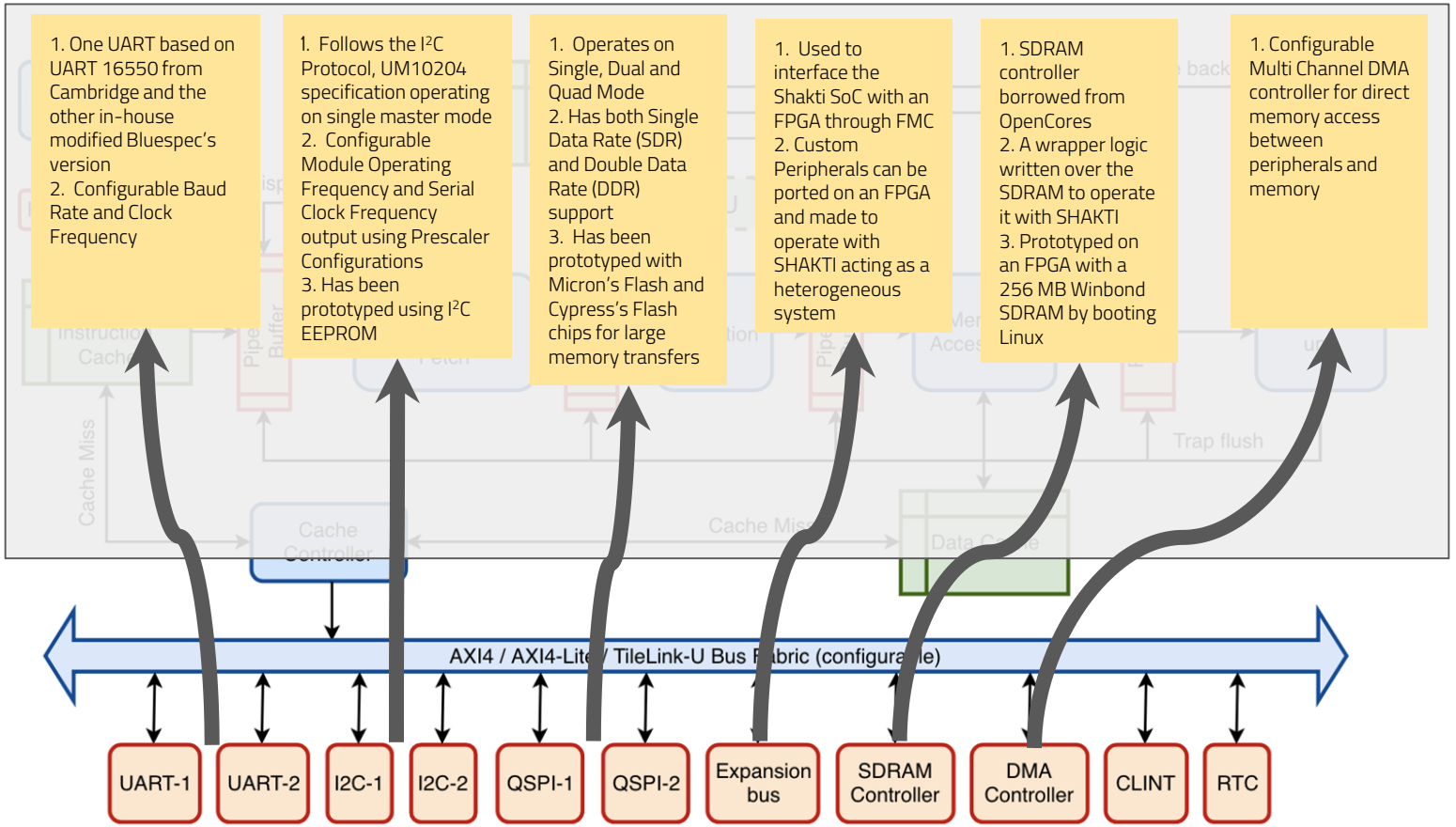
SHAKTI

Write-Back Stage



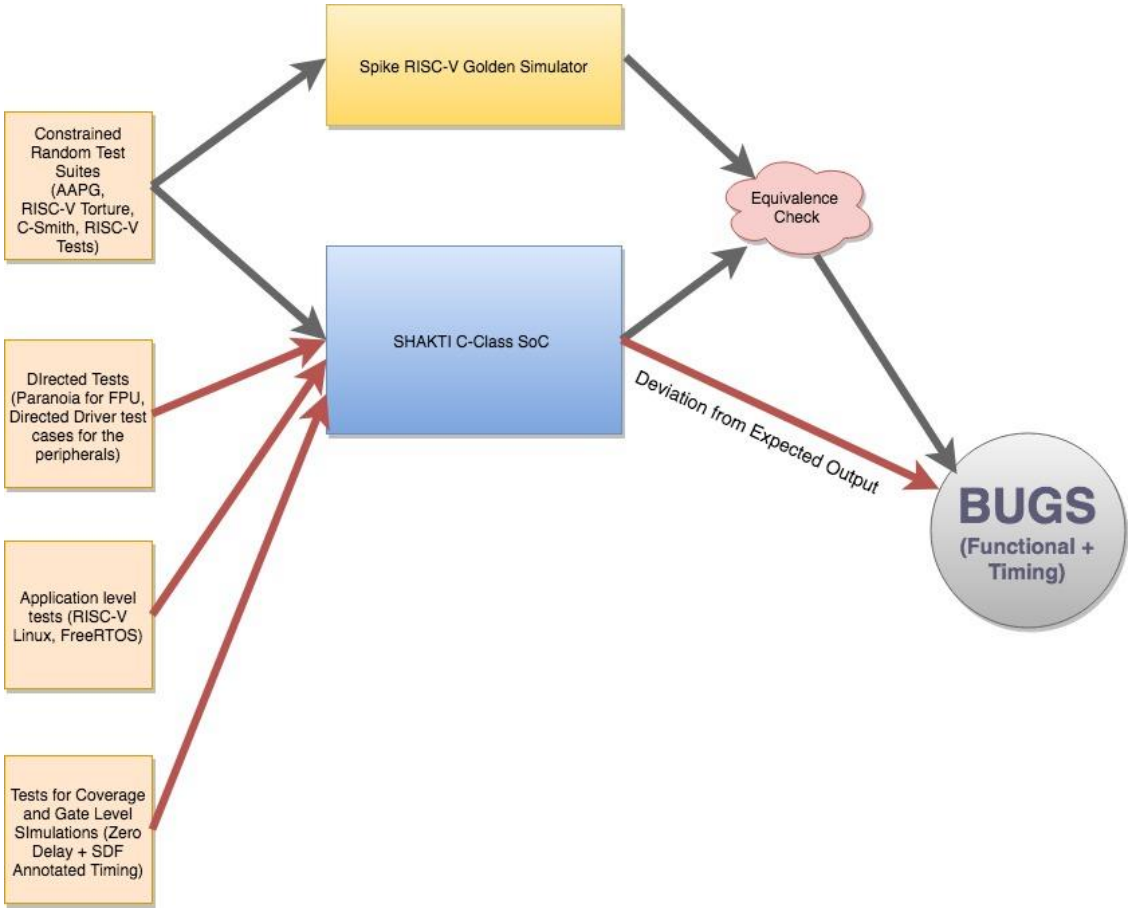
- Exceptions and Interrupts are handled in Write-back stage
- In case of Exceptions/Interrupts, the pipeline is flushed by generating flush signal to Epoch registers in all stages, and dropping the instruction based on Epoch match logic

SoC Micro-architecture



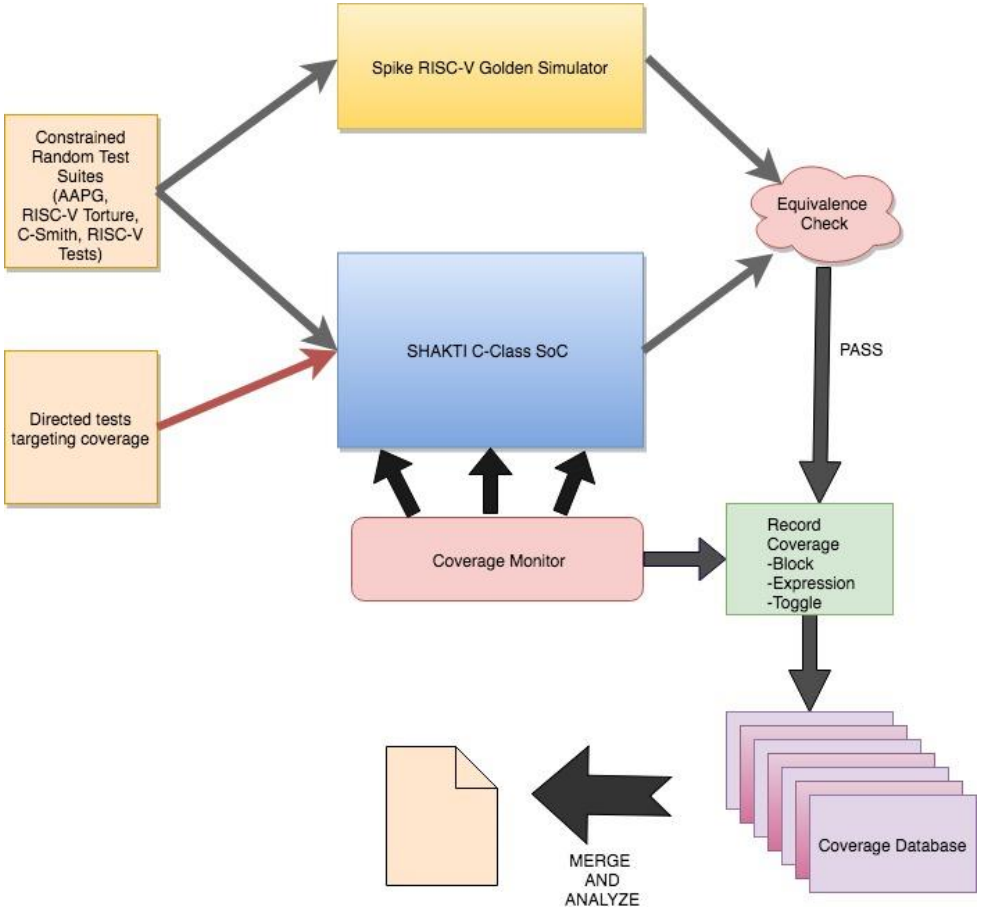
SHAKTI

Verification Framework



SHAKTI

Code Coverage

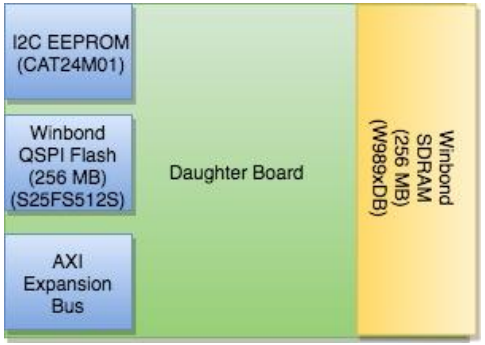


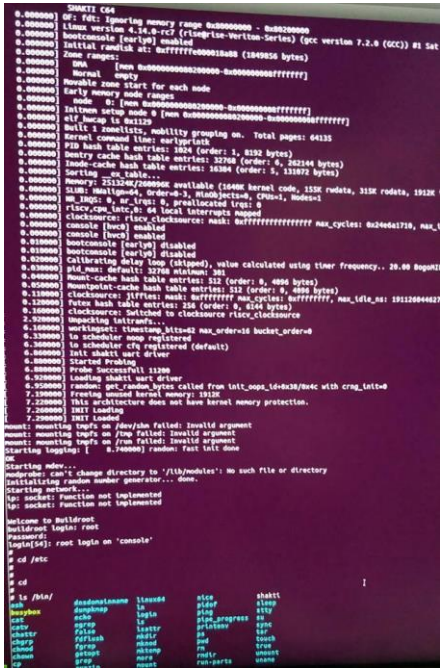
Issues in coverage analysis using HLS:

- Any HLS to Verilog Conversion involves generation of redundant logic and extra signals which gets optimized away by the backend synthesis tool
- These redundant extra signals generated by the synthesizer is hard to cover and hence attaining 100% coverage is not possible

FPGA Emulation

FMC Connector





SHAKTI C-Class on Xilinx FPGA

(Artix-7, Virtex UltraScale, Kintex-7)

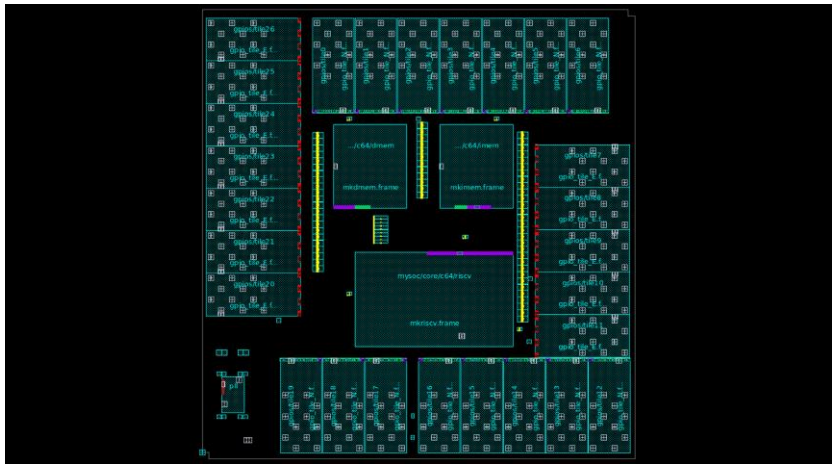
SHAKTI

RTL to FAB - Partnering with Intel Corporation and HCL

- Fabricated on Intel 22nm FinFET Low-power (FFL) with die size of 4mm x 4mm
- Flip-chip package hitting a frequency of 320 MHz, operating at 0.7Vc core
- IO dominated design (324 Signal IO's)

DESIGN PARTITION

- FPU
- Core
- Instruction Memory
- Data Memory
- Peripheral Controllers

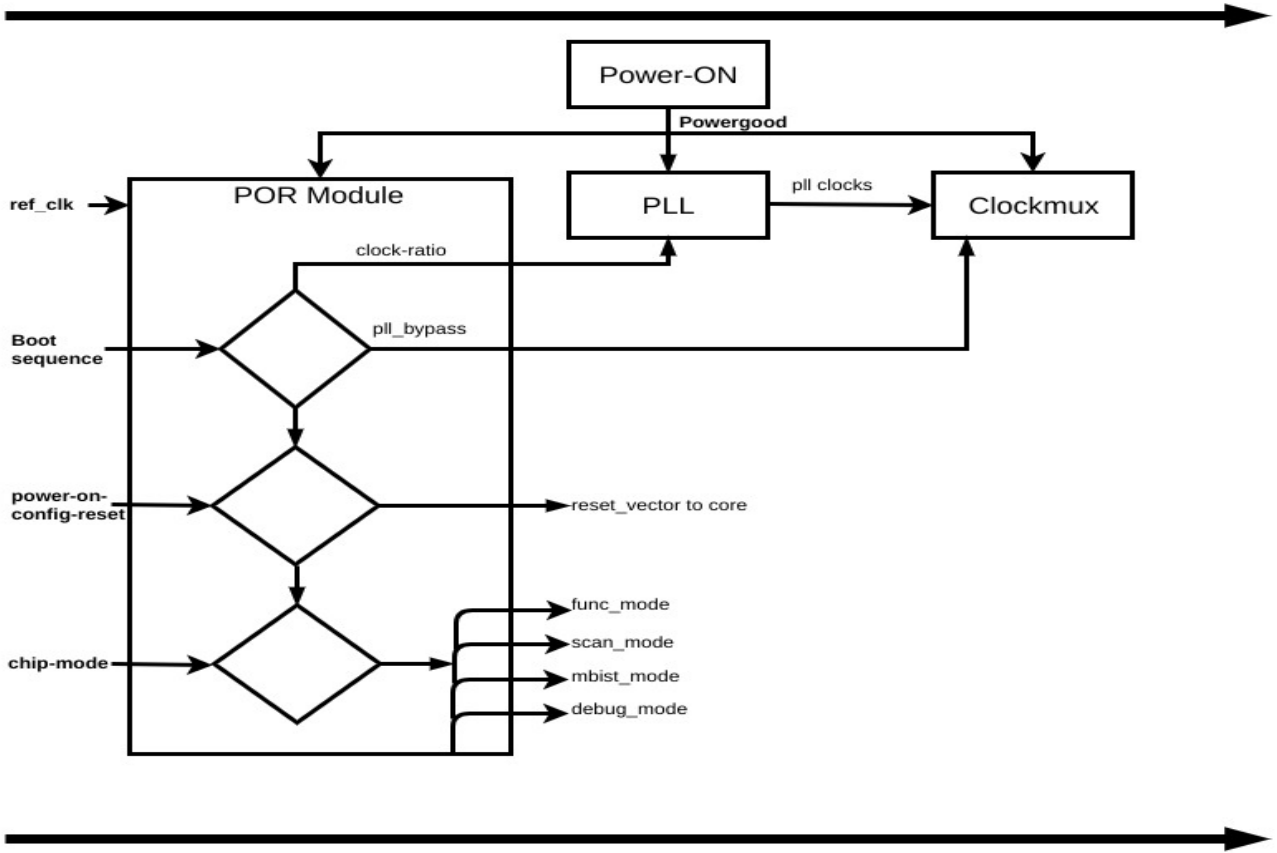


Top Level Floor Plan

Power On Reset Spec

- Describes about the behaviour of chip during power up, reset sequence and mode selection for the chip.
- Rise Creek has a power good pin asserted externally through Power Management IC (PMIC) on board to indicate that all the desired voltage was achieved.
- This pin is also used to reset some of the logics which needs to be active before system reset.
- Once the power good has reached, the chip mode is selected using the following chip mode signals:
  - Scan mode
  - Functional mode
  - MBIST mode
  - Debug dump mode
- Boot_sequence -> Select between clock_ratio and pll_bypass
- Power_on_config_reset -> Select reset vector value

Power On Reset Spec



SHAKTI

Top level features

Clock Mux Logic:

- This logic is implemented to avoid glitches between the selection of clocks. Glitch may be caused due to immediate switching of the output from Current Clock source to the Next Clock source, when the SELECT value changes.
- Definite cells were provided by Intel Custom foundry for performing Integrated clock mux and clock gating.
- Logic to mux between PLL reference clock and PLL core clock is implemented to ensure the chip function in PLL bypass mode and PLL clock mode.

Divide PLL Lock:

- The locked PLL output is divided by 8 and sent out from the chip for clock observability.

Top level configuration setting

Phase Locked Loop (PLL) and IO pads:

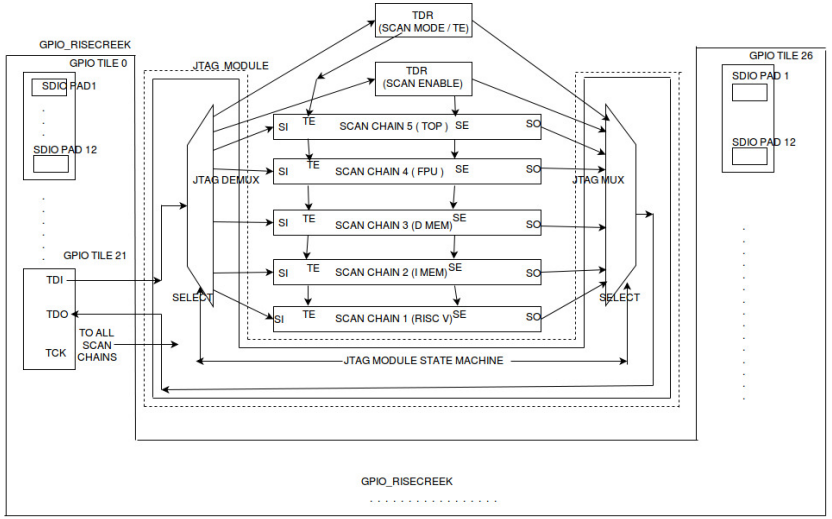
- Intel Custom foundry provided PLL IP for internal clock generation and SDIO pad library cells for IO ports.
- Enabling PLL on power up, frequency setting control bits, voltage supplies, enabling Low drop output (LDO), In Die Variation (IDV) interface etc, were set by configuring the PLL IP, specific to Intel.
- Configuration of SDIO pads were done in accordance with the design behaviour as input or output, enabling pull-up/pull-down, drive strength etc, specific to Intel
- Single block of GPIO constitute 12 SDIO cells, grouping to 27 GPIO's for accommodating 324 IO pins.

Design for Testability

Rise creek has the following DFX modes:

Scan Mode:

- Implementation of DFT was done using the debug JTAG interface.
- Five scan chains were implemented for FPU, RISCV, IMEM, DMEM and TOP level modules
- Scan mode signals test_enable/ test_mode, scan_enable are implemented as user defined registers in JTAG space and are programmed using JTAG sequences
- Boundary scan is implemented under IEEE 1149.1



SHAKTI

Design for Testability

MBIST mode:

- Implemented test logic for SRAM blocks used in the design.
- Tool Used - Mentor Tessent

Debug Dump mode :

- When the core hangs during the functional mode, the state of the chip is dumped through JTAG onto the host controller.
- This can be further used to diagnose the cause.

Physical Design Process

Following were the major steps and challenges involved in constructing the Rise creek chip during the backend process:

a) Bump map was created with the given area by importing the IO pin list and the block coordinates. This step involved the creation of Etch-Ring(ER), Edge Damage Monitor (EDM) ring and pattern recognition system for the chip. These are specific to Intel foundry.

b) The synthesized scan inserted netlist and constraints for all the blocks were processed for an automated placement and route (APR) using synopsys IC-Compiler-II.

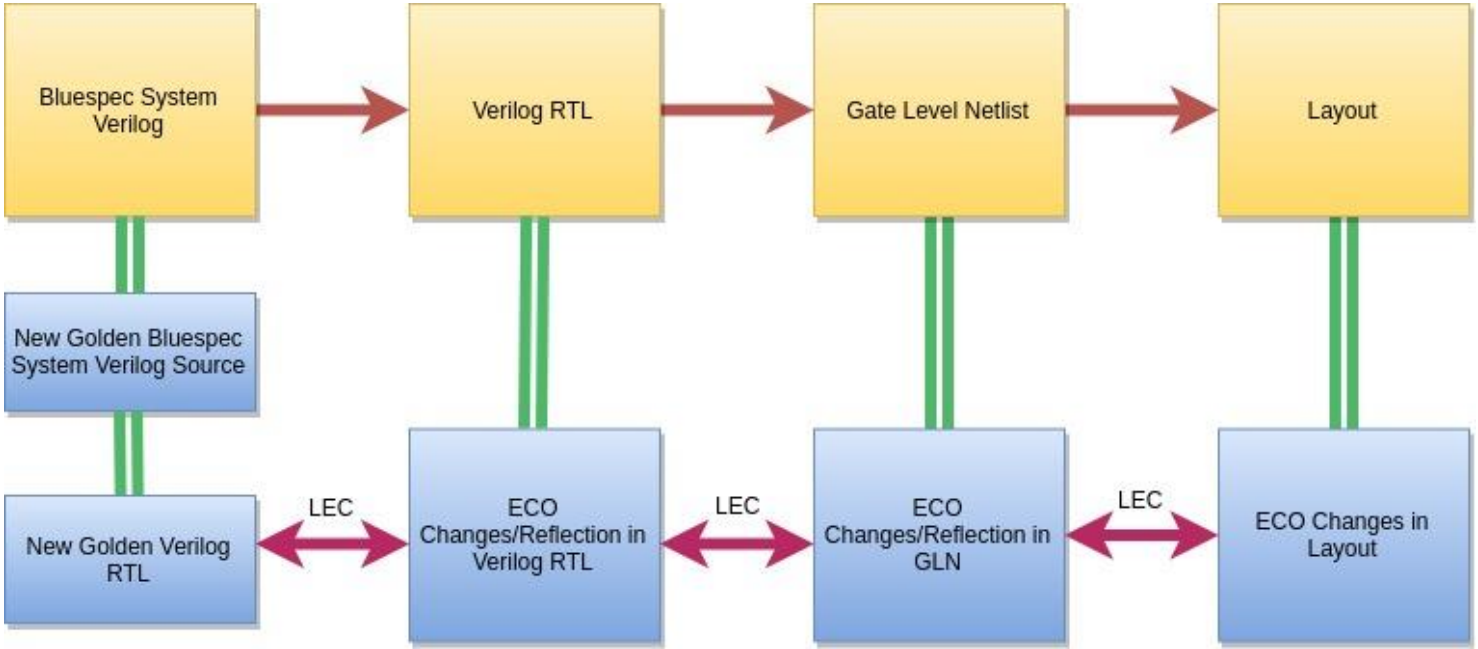
c) During Clock tree synthesis, placement legality and global route congestion was checked. Parasitic extraction is done using cadence QRC tools and spef file is generated. Static timing analysis using synopsys primetime tool for sign off the setup and hold timing requirement.

Physical Design Process

d) Analysis of leakage and dynamic power for each blocks was measured using cadence Voltus
e) Layout vs schematic (LVS), Design Rule Check (DRC) and Electrical Rule Check (ERC) is verified using Mentor Calibre. This process will ensure majorly for any shorts, metal-to-metal spacing, electro-static discharge and floating inputs.
f) After every iteration, the netlist was forced for Logical Equivalence checking using cadence LEC.

Handling ECO changes:

How do we implement ECO changes so that BSV is coherent.



Thank You

Questions?