The Cortex-A15 Verification Story

University of Florida

Dan Millett

October 2012

ARM Introduction

- IP licensing company
  - R&D outsourcing, supplying all major semiconductor companies
  - Processor “brain” in the chip
- Started in 1990
  - Based in Cambridge, UK
  - Listed on London and NASDAQ
  - $8 Bn market cap (4x in two years)
- Now 1900 people
  - Mainly R&D engineers
- $600m revenue, 40% operating profit
- Partnership business model
  - ~6 Bn shipments in 2010

ARM started in a barn

Now 30 offices in 15 countries

How many ARM's Do You Have?

Mobile phones

~100%
market share

Smartphones

3x 100%
market share

Mobile Computers

5x 100%
market share

Digital TVs

30%
market share

Disk Drives

~70%
market share

PC Peripherals

30%
market share

Cars

5x 40%
market share

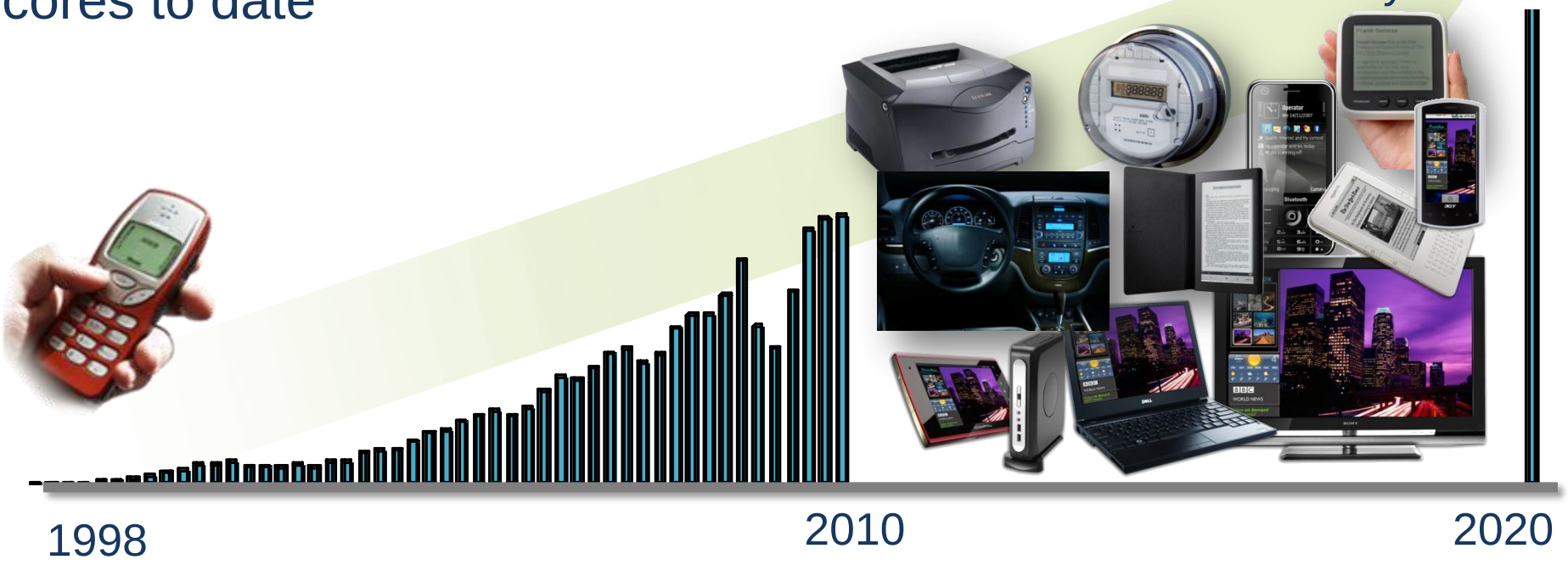
Microcontrollers

10%
market share

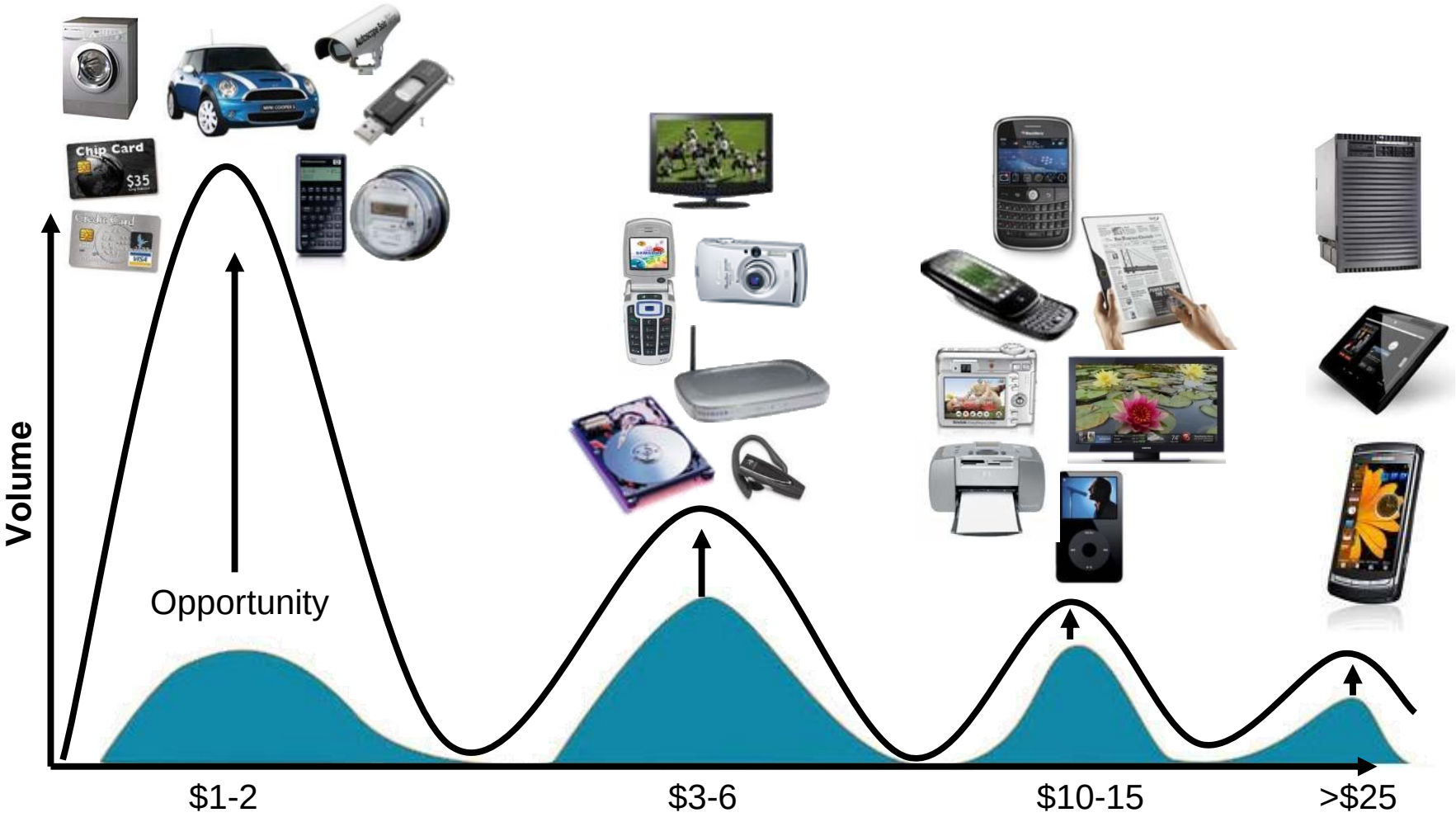
Huge Opportunity For ARM Technology

20+
billion
cores to date

100+
billion
cores accumulated
after next 10 yrs



ARM's Opportunity at all Price Points



ARM Connected Community – 700+

ARM Austin

- Austin site opened in 1999
- Currently 250 engrs
- Growing 10%+ per year

- Top right of CPU roadmap
- Interconnect fabric
- Verification tools
- R&D
- Sales, AEs, Support…

Austin is the center of the CPU world

- CPU Teams
  1. ARM — High end of Mobile
  2. Qualcomm — DSPs
  3. Intel — Atom
  4. Freescale — PowerPC and more
  5. IBM — Servers
  6. Oracle — Servers
  7. Centaur — Low cost X86
  8. Broadcom — Networking processors
  9. AMD — Multiple CPUS
  10. Samsung — ARM CPUs
  11. Apple — shh… it's Apple

- SoC teams
  1. Calxeda — ARM servers
  2. Nvidia
  3. TI
  4. Cirrus
  5. Plus another 5-10

WHAT IS CORTEX-A15?

Cortex-A15: Next Generation Leadership

Target Markets

- High-end wireless and smartphone platforms
- tablet, large-screen mobile and beyond
- Consumer electronics and auto-infotainment
- Hand-held and console gaming
- Networking, server, enterprise applications

Cortex-A class multi-processor

- 40bit physical addressing (1TB)
- Full hardware virtualization
- AMBA 4 system coherency
- ECC and parity protection for all SRAMs

Advanced power management

- Fine-grain pipeline shutdown
- Aggressive L2 power reduction capability
- Fast state save and restore

Significant performance advancement

- Improved single-thread and MP performance

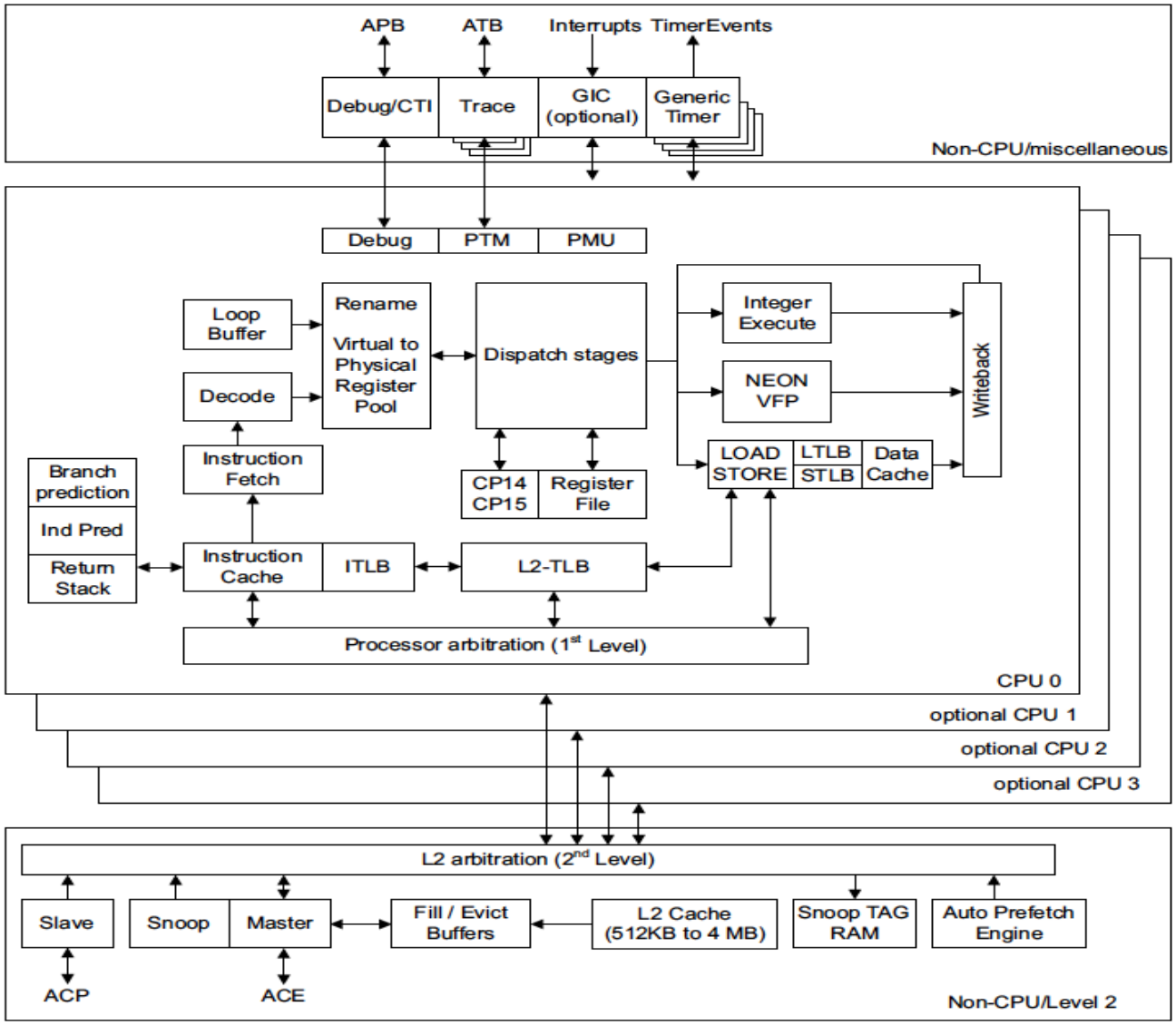
Targets 1.5 GHz in 32/28 nm LP process
Targets 2.5 GHz in 32/28 nm G/HP process

Cortex-A15 MPCore Block Diagram

- **Global Interrupt Control, Trace and Debug handled across all cores**

- **1-4 Processors per cluster**
- **Each processor has full Out-of-Order (OoO) pipeline.**

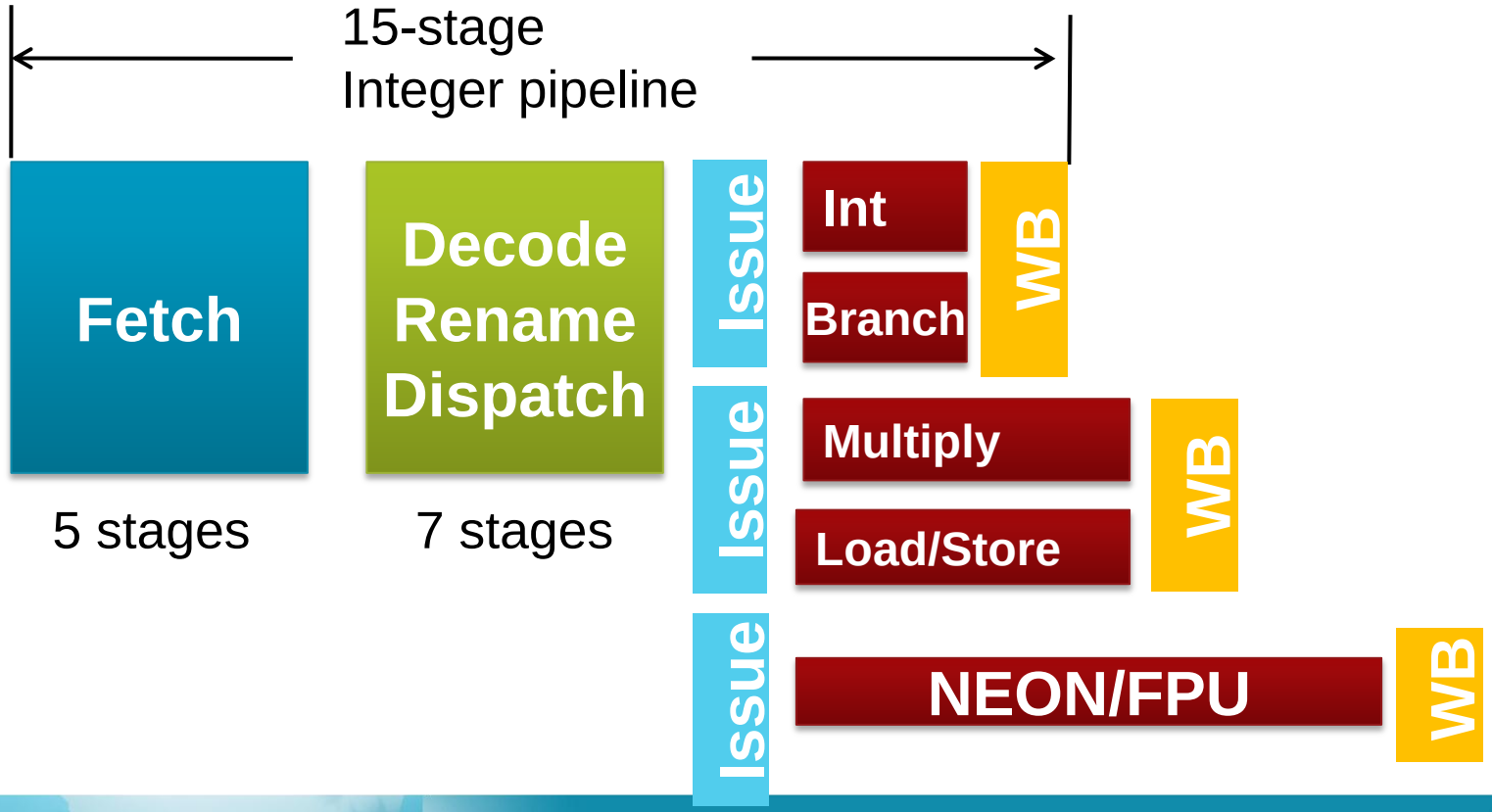
- **Integrated Level 2 cache**



Cortex-A15 Pipeline Overview

15-Stage Integer Pipeline

- 4 extra cycles for multiply, load/store
- 2-10 extra cycles for complex media instructions

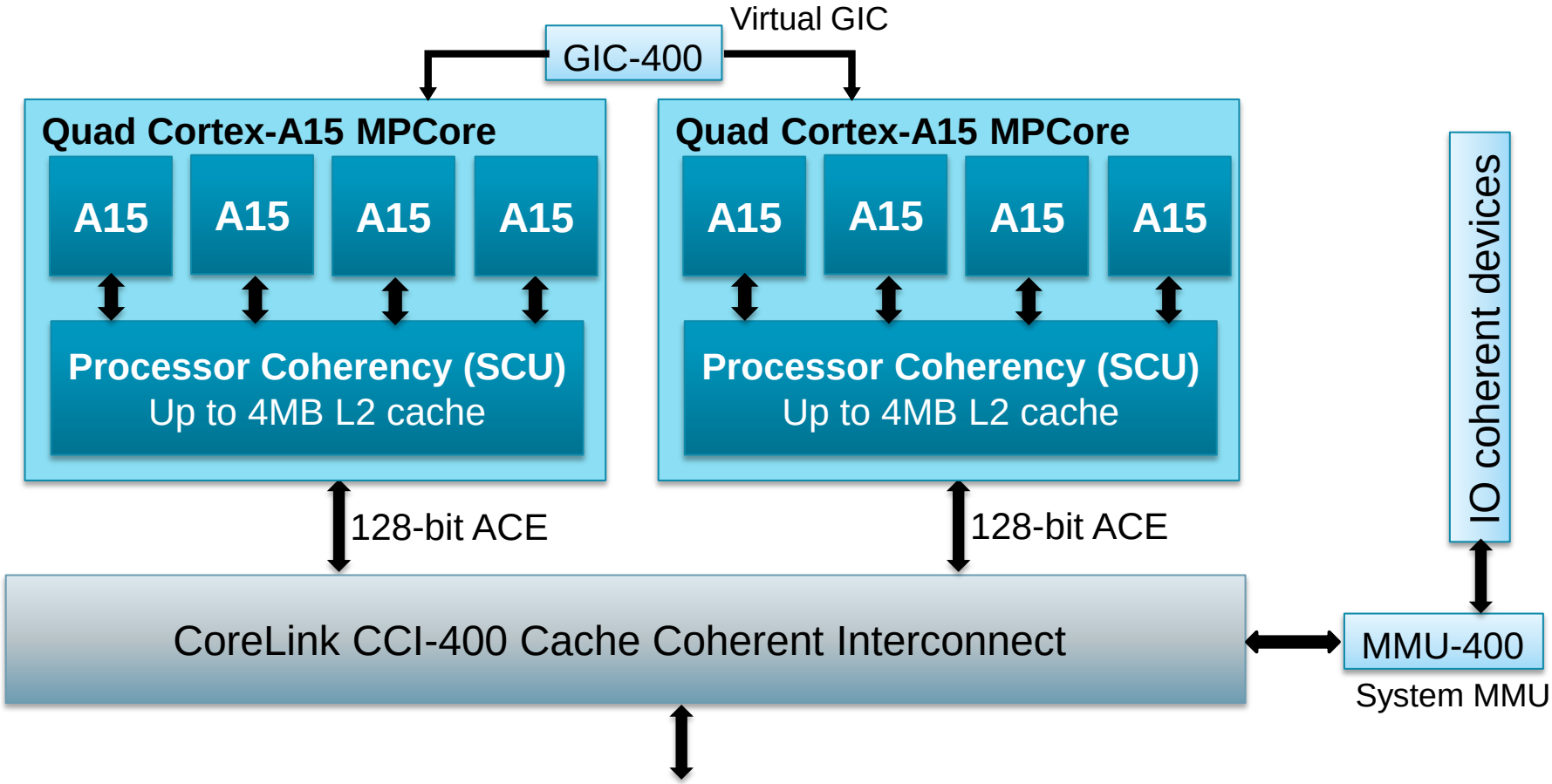


Configuration Challenge

System feature	Cortex-A15
Number of CPUs	1-4
L1 cache size	Fixed at 32 KB
L2 cache controller	Included
L2 cache size	512KB, 1MB, 2MB, 4 MB
L2 tag RAM register slice	0, 1
L2 data RAM register slice	0, 1, 2
L2 arbitration register slice	0, 1
Error protection	None, L2 cache only, L1 and L2 cache
Interrupt controller	Optional
Number of SPIs	0-224 in steps of 32
Power management	Optional clamp/power-gate control pins
Floating point / NEON	None, VFP Only, VFP and NEON
Trace	PTM (integrated, required)

Cortex-A15 System Scalability

- Processor-to-processor coherency and I/O coherency
- Memory and synchronization barriers
- Virtualization support with distributed virtual memory signaling



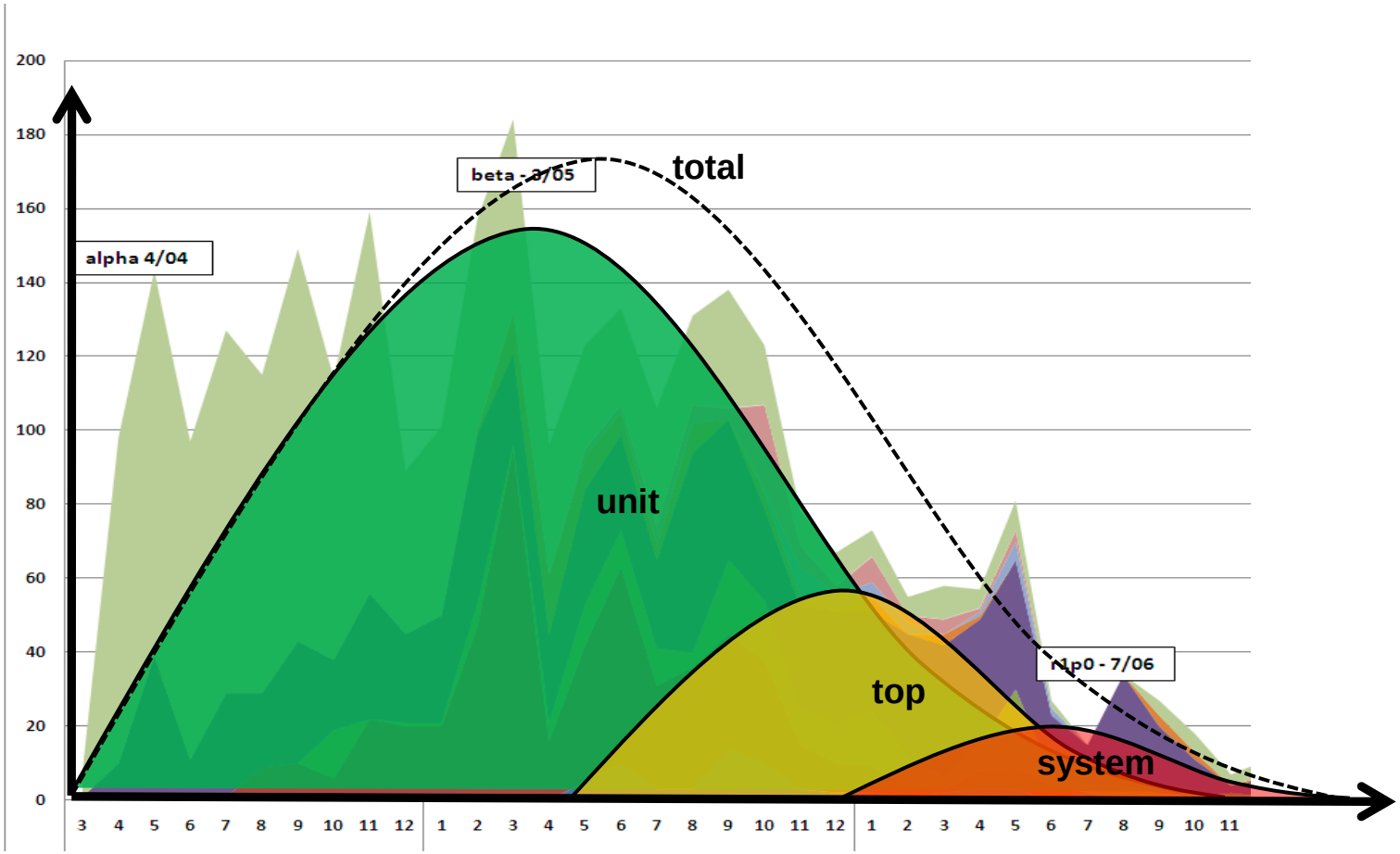
VERIFICATION METHODOLOGIES

ARM CPU Verification Strategies

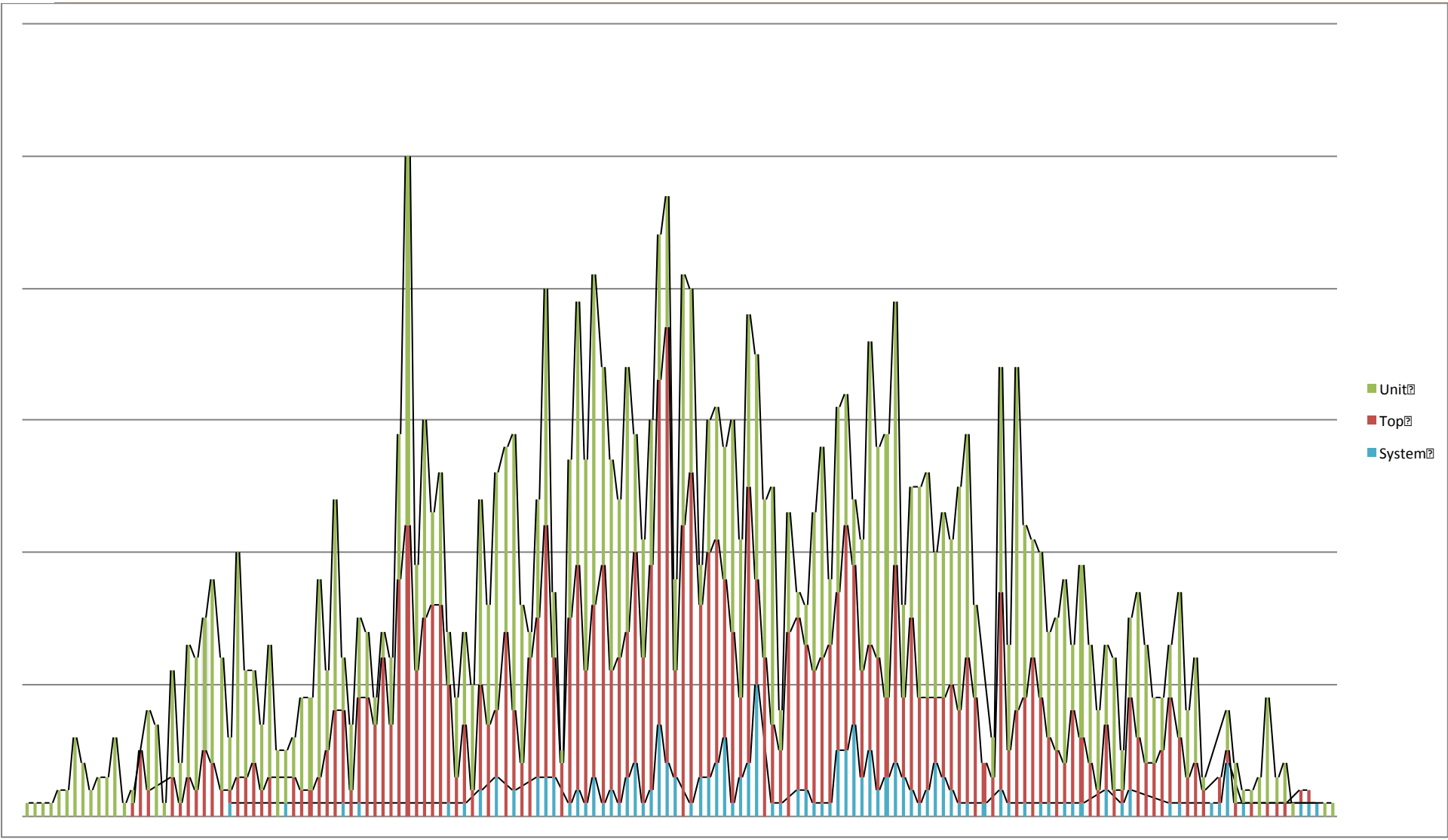
- Design practices – “correct by construction”
- Test planning
- Multiple and varied verification methods emphasizing:
  - Unit level
  - Top level - RIS (random instruction sequences)
  - System level - stress testing
- Coverage
- Soaking / Bug Hunting

Bug Discovery Timeline - Theoretical

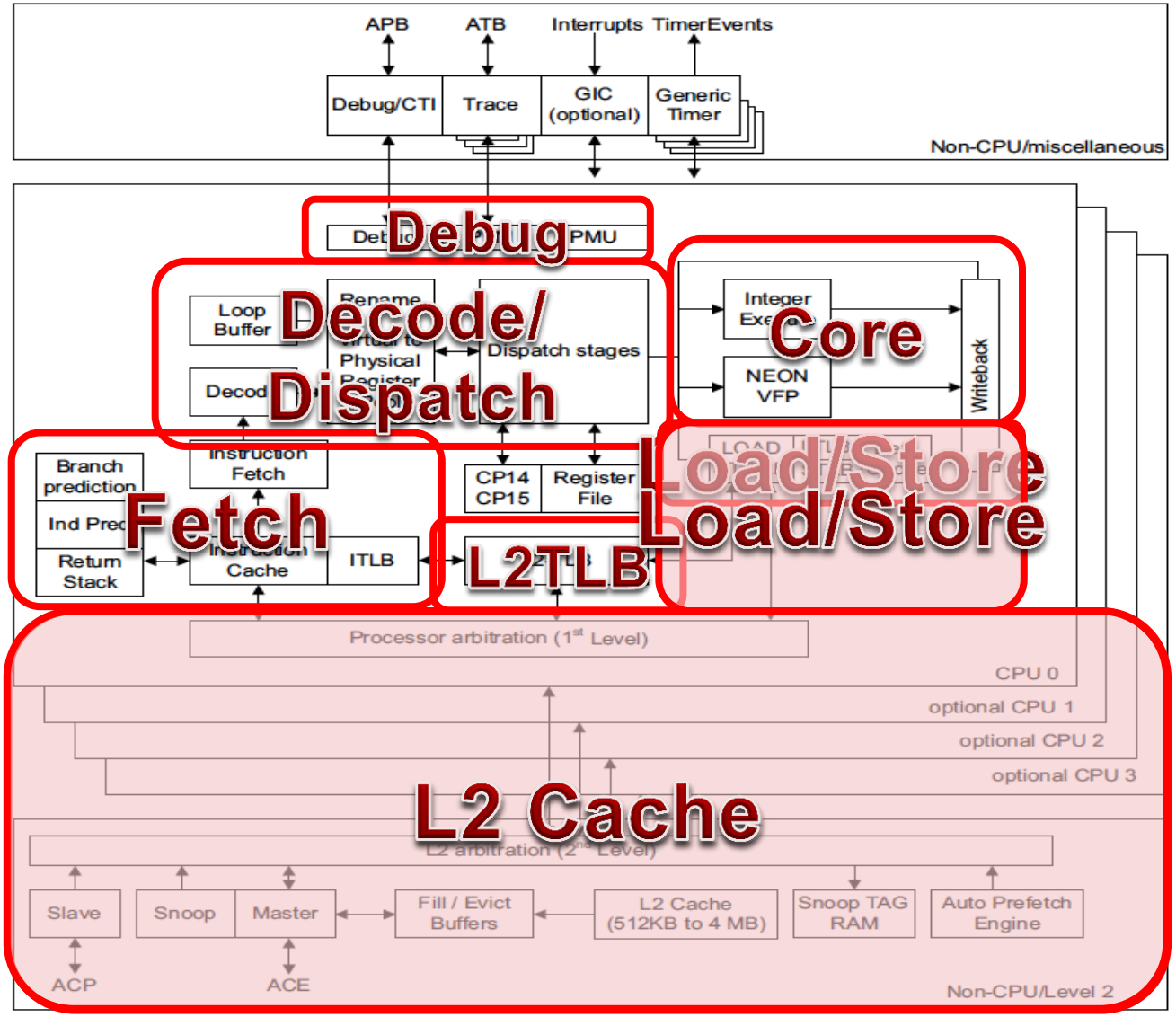
- Where are bugs discovered?



Where Are Bugs Found - Actual



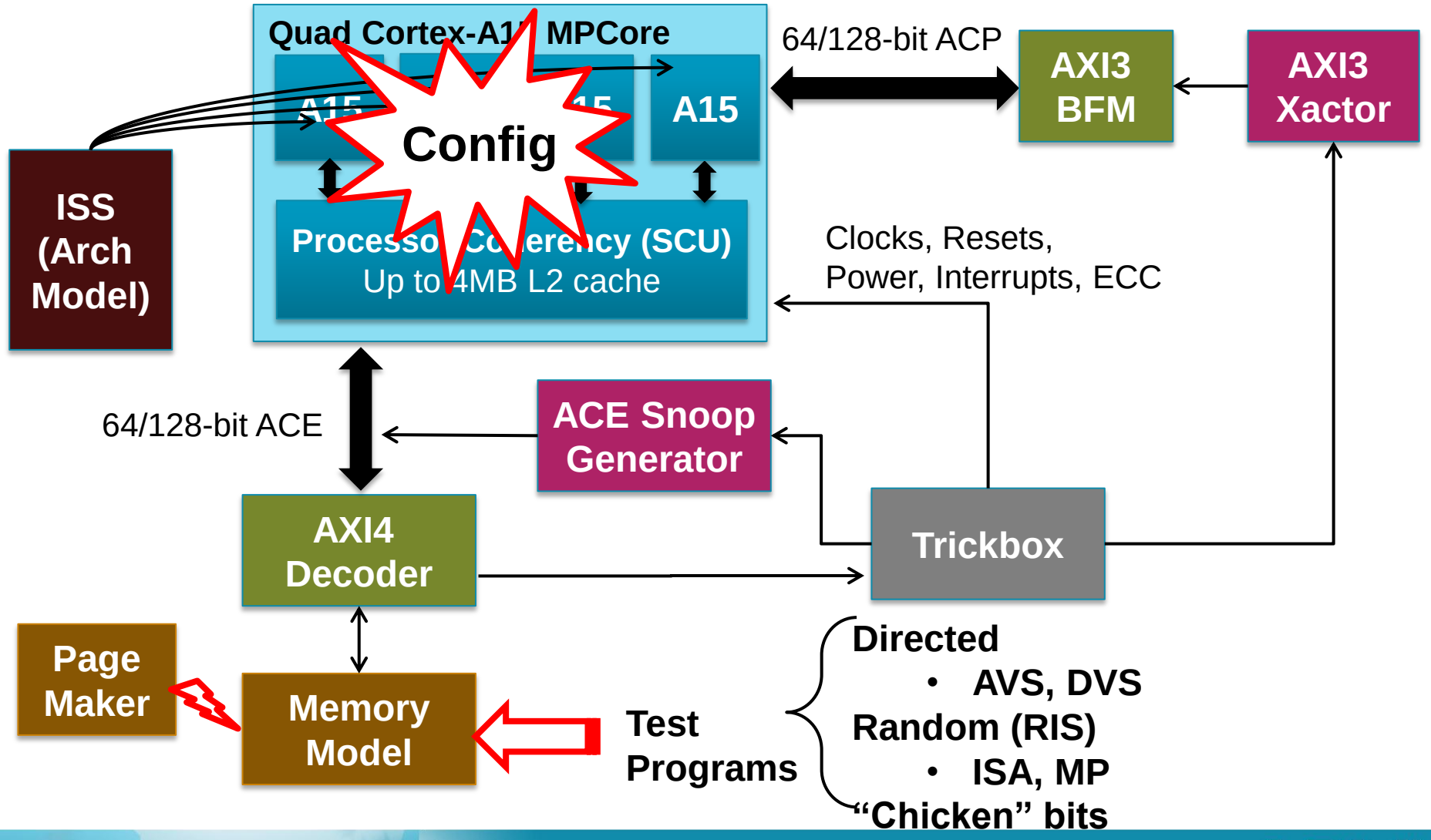
Cortex-A15 Unit Level Testbenches



Unit Level Simulation

- Simulation is the corner-stone verification method
- Coverage driven, constrained random SystemVerilog
  - Assertions for interfaces, white box internals
  - Higher level checkers
  - Code and functional coverage drives stimulus completeness
- Well-defined and testplan-linked functional coverage
- Multi-unit testbenches are used where appropriate
- Simulator performance and compute cluster
- Debug visualization and automation

Top Level Testbench



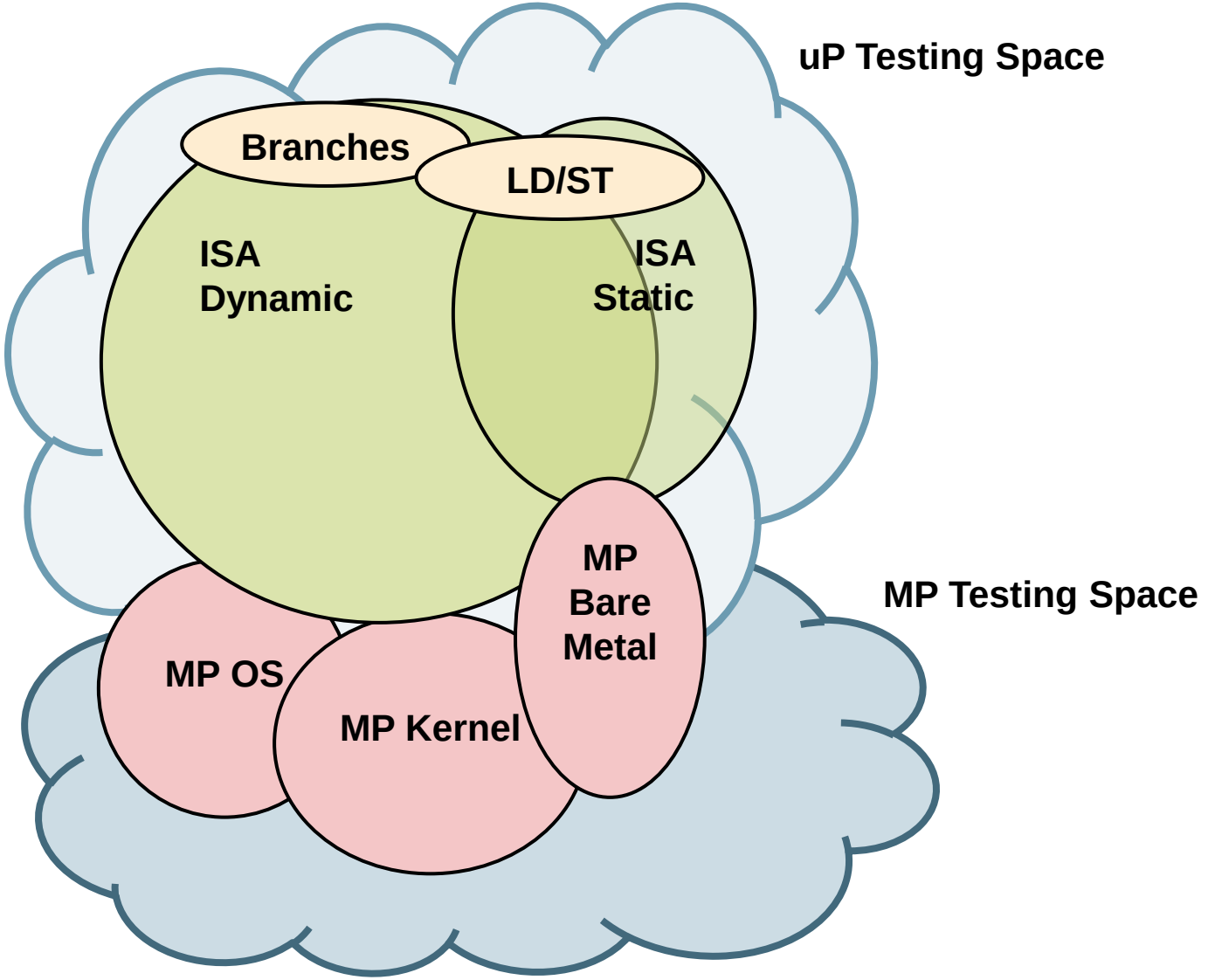
Top Level Simulation

- Uses a CPU Top-Level Testbench
  - Simple memory, simple trickbox, Arch reference model integration
- Tests are binary executable programs
- Exercise various Cortex-A15 configurations
- Directed tests
  - AVS is architecture compliance suite every ARM CPU must pass
  - DVS is a suite of directed tests for this ARM implementation
- Random tests (RIS = Random Instruction Sequences)
  - ISA
  - MP/coherency
- Irritators: interrupts, ECC, page tables, “chicken bits”

RIS (Random Instruction Sequences)

- Track record of hitting un-planned scenarios
- Multiple RIS engines have been developed over >12 years and applied to all CPUs
  - 3 mainstream ISA based engines
  - 3 MP targeted engines
  - Plus 5 additional engines to target load/stores, VFP, M and R class cores
- Engines being enhanced to scale in H/W platforms
  - To achieve much higher throughputs (>10^{15} cycles)

RIS Generator Testing Space

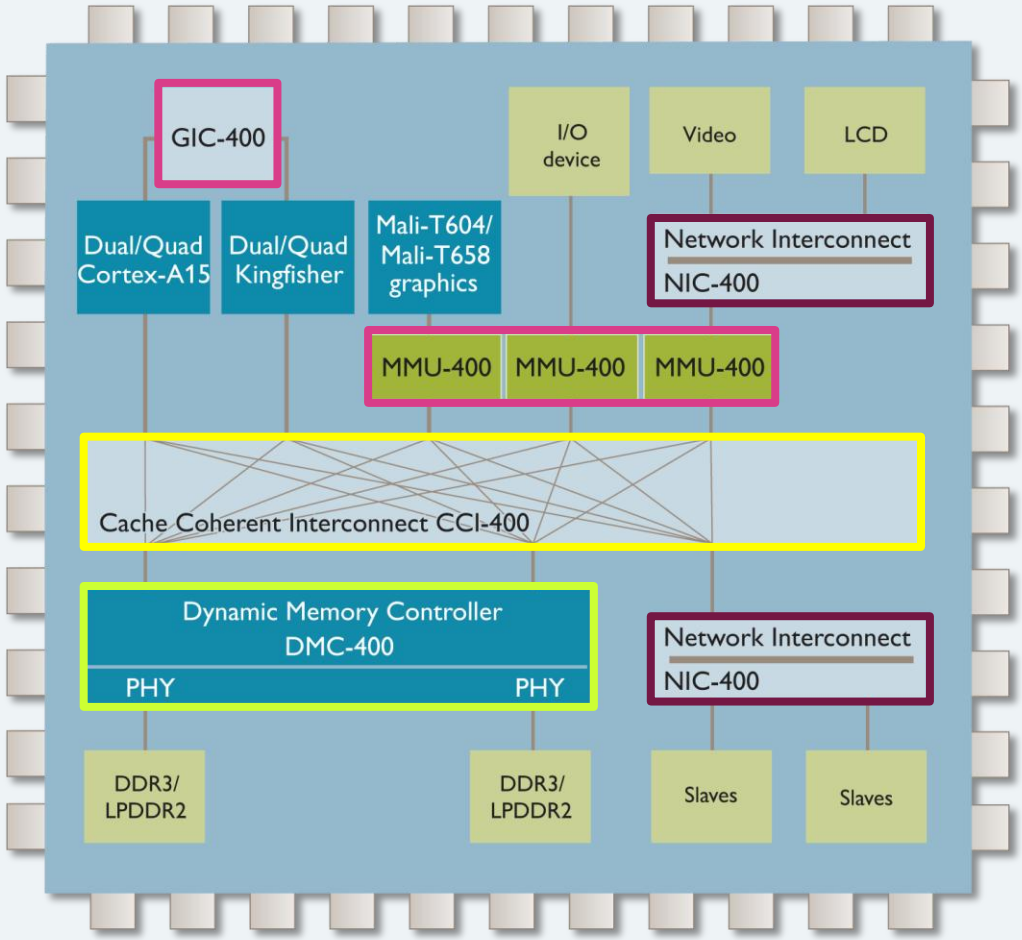


System Level Validation

- Objective to perform “in-system” validation of ARM IP
  - Extended validation of IP in system context
  - Find IP product bugs from real-world testing
- Platforms
  - Emulation
    - SystemBench = *configurable* platform for running SV tests
  - FPGA
    - High throughput to enable deep soaking of the design
- Test Content
  - Bare-metal
  - OS-based apps, stress tests

System Validation Platform Example

CoreLink™ 400 Series
Intelligent System IP by ARM



Coherency

- CCI-400
  - Full cache coherency
  - I/O coherency
  - Prioritization and utilization

Virtualization

- MMU-400
  - OS level virtualization
- GIC-400
  - Virtual interrupts
  - Multicore support

External Memory Subsystem

- DMC-400
  - DDR utilization
  - PHY integration

Rest of SoC Interconnect

- NIC-400
  - Routing efficiency

System-Level: Validation Strategies

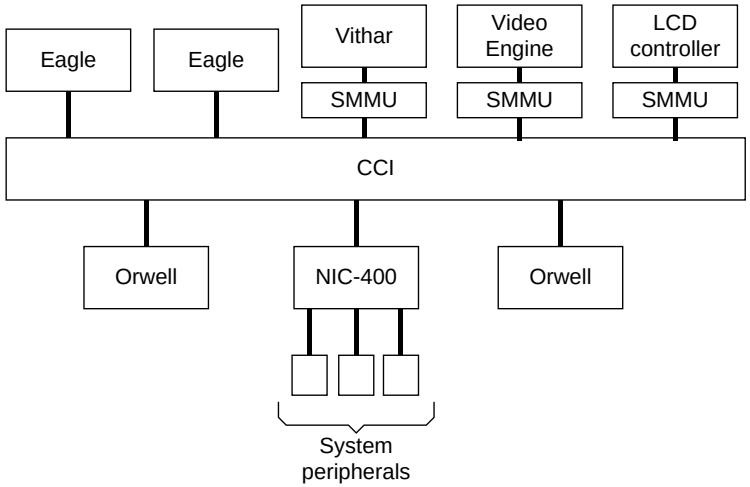
TEST CONFIGURATIONS

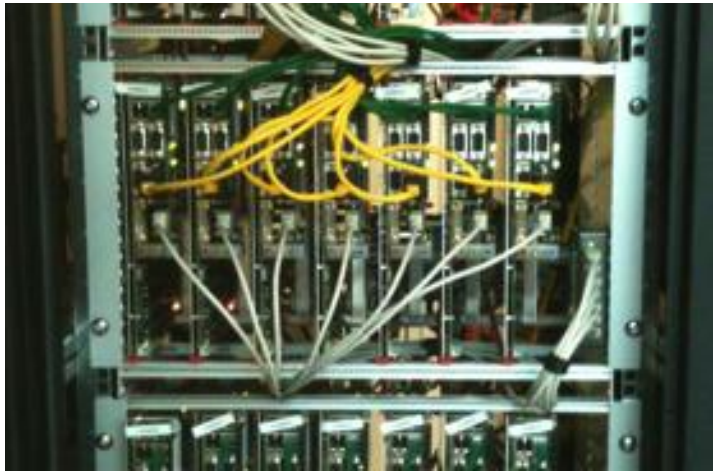
- IP component build configs
  - Multi-core, Neon/VFP engine, cache sizes, interconnect configs, etc
- Systembench topologies
  - Multi-cluster, ACP, DMC, SMC, DMA, etc
- Runtime initialisation
  - Memory regions, performance modes, etc)

TEST PAYLOADS

- OS and Application compatibility testing
  - Linux, Windows, Android, LTP, benchmarks
  - Hypervisor, TrustZone
- MACK (simplified OS for validation) based stress testing
  - MPRIS – pthead based tests for MP
  - ‘C’ Stress testing library (including coherency tests and targeted stress)
- Bare metal directed/random tests
- RIS
- Runtime traffic irritators (DMA, GPU, VIP)

System level: Emulation/FPGA Farm



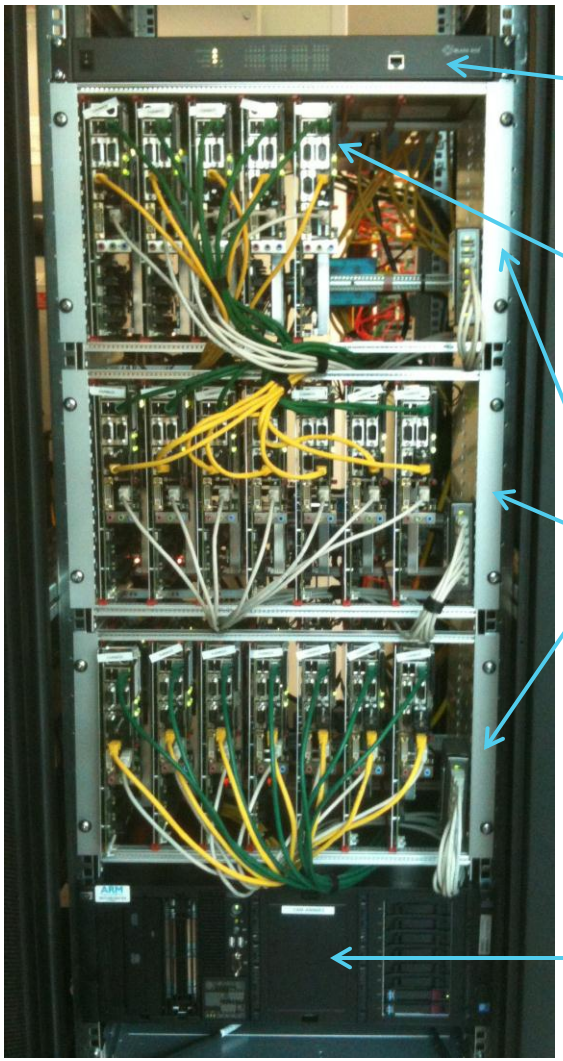


- Configurable “System-Level” Testbench
- Emulation achieves ~1MHz
- Effective debug visualisation
- More suitable to longer tests (OS boots, benchmarks, longer RIS sequences)

- Limited fixed configurations
- FPGA achieves 10-40MHz
- Poor debug visualisation
- Targeting RIS testing and stress testing

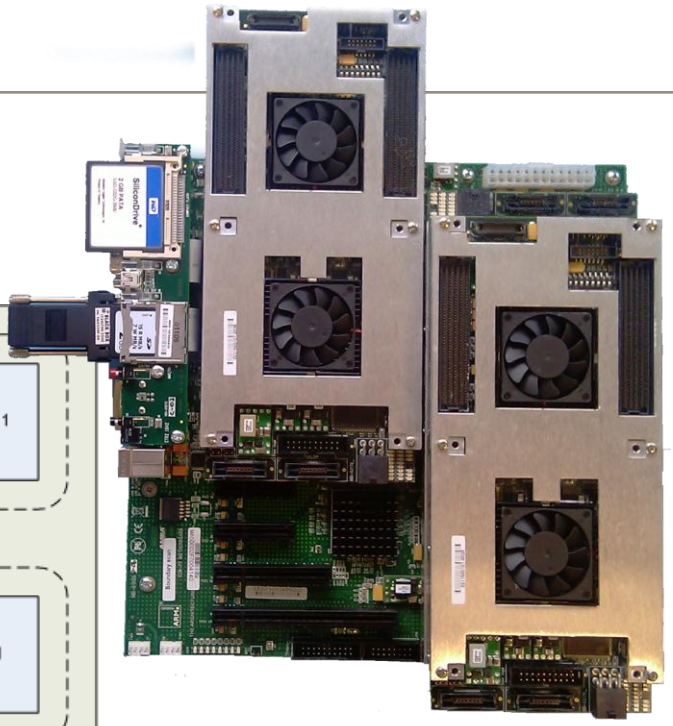
FPGA Farm

- 21 FPGA platforms per rack
  - V2F-2XV6
    - LX760 & LX550T
    - 4GB DDR2 SODIMM
    - JTAG and Trace
  - V2M-P1 motherboard
    - NOR Flash bootloader
    - Basic peripherals
      - UART for SW debug
      - Ethernet for network boot
      - Video/audio
      - SD/CF for local storage
- Cluster Control
  - Redhat Linux box
  - UART concentrator for debug
  - RVI for software debug
  - FPGA and SW image download



Dual Cluster Cortex-A15

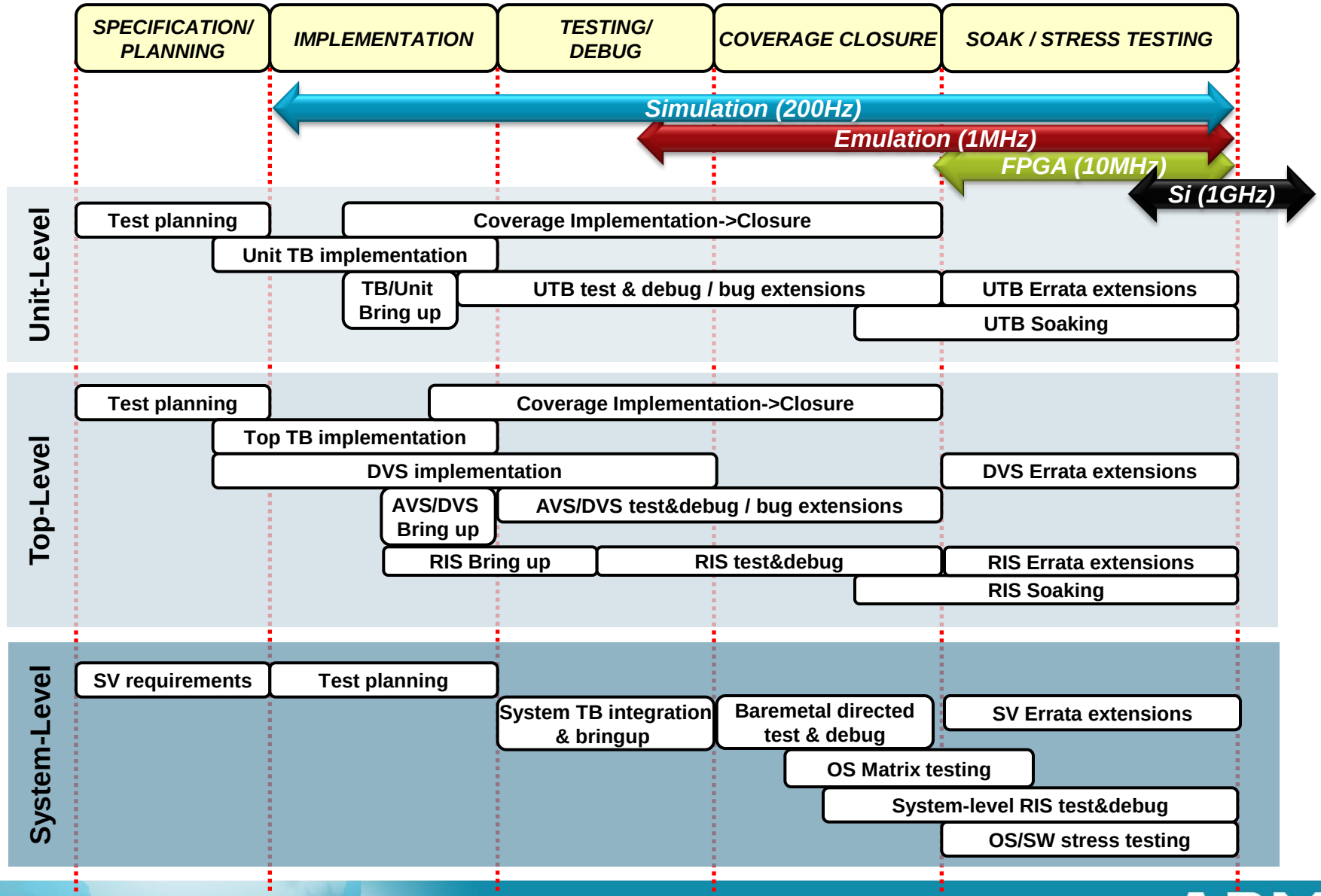
- Solution per VE Platform
- Use three V2F-2XV6 boards
  - 3x LX760
  - 3x LX550T
- Processor Support
  - Dual Cluster A15
  - A15 Neon & A7
- Performance
  - 10MHz system speed
  - 2-4GB memory space



Formal Property Verification

- ACE proof kit
  - Complete set of bus protocol properties
- Low level assertions
  - Prove assertions on LS unit interfaces
- High level properties
  - L2 ECC proof
  - L2 arbitration register slice

Verification Methodology Summary



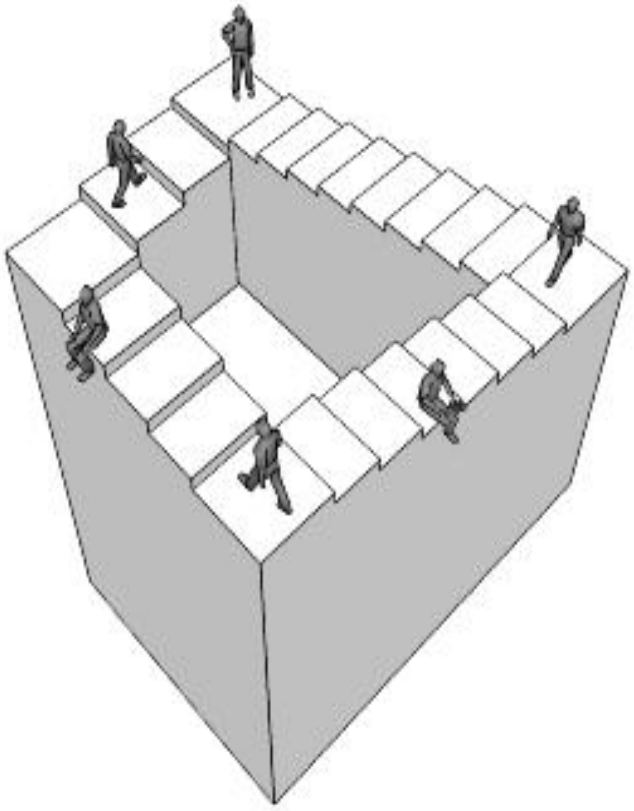
LESSONS LEARNED

Planning

- Take a step back now and then…
- Make sure to plan for the unplanned

Functional Coverage

- Don’t start too early
- Focus on the places where the bugs are



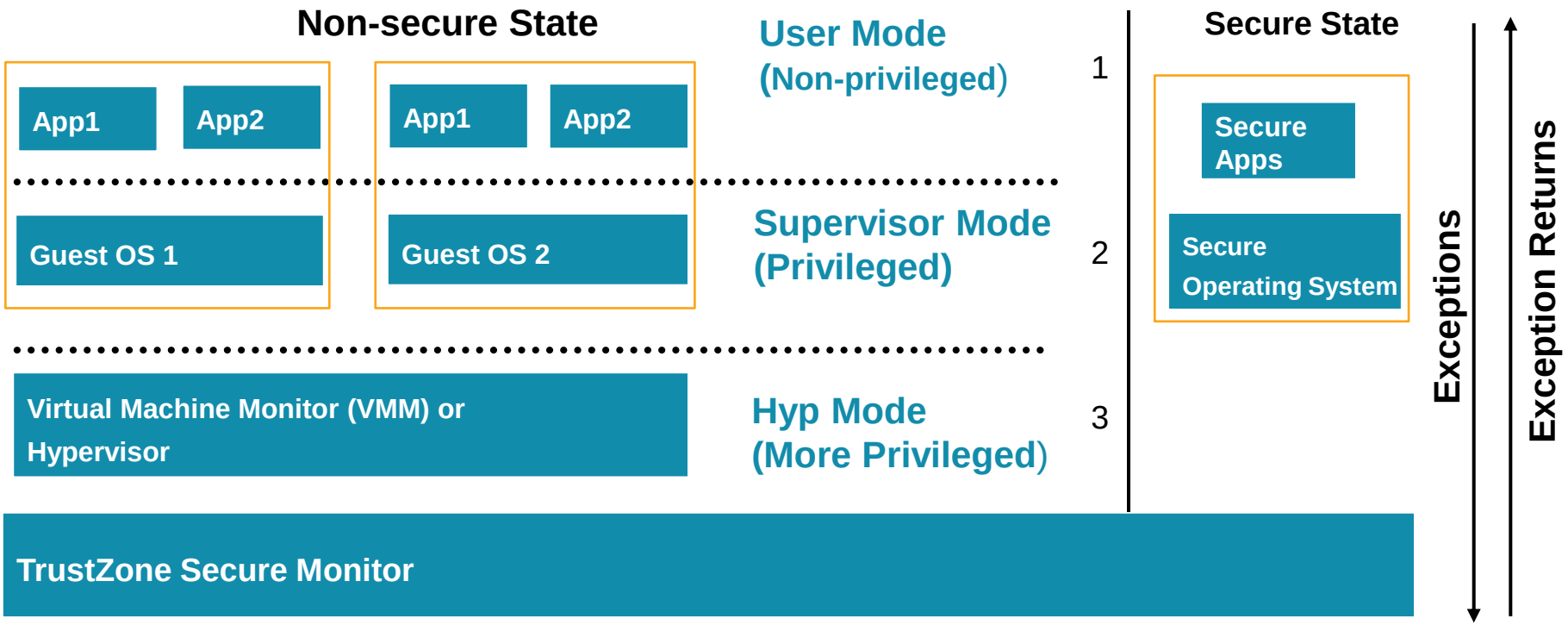
CHALLENGES

Configurability

System feature	Cortex-A15
Number of CPUs	1-4
Interrupt controller	Optional
Number of SPIs	0-224 in steps of 32
Power management	Optional clamp/power-gate control pins
Floating point / NEON	None, VFP Only, VFP and NEON
Error protection	None, L2 cache only, L1 and L2 cache
L2 cache size	512KB, 1MB, 2MB, 4 MB
L2 tag and data slices	00, 01, 02, 11, 12
L2 arb slice	Present or not

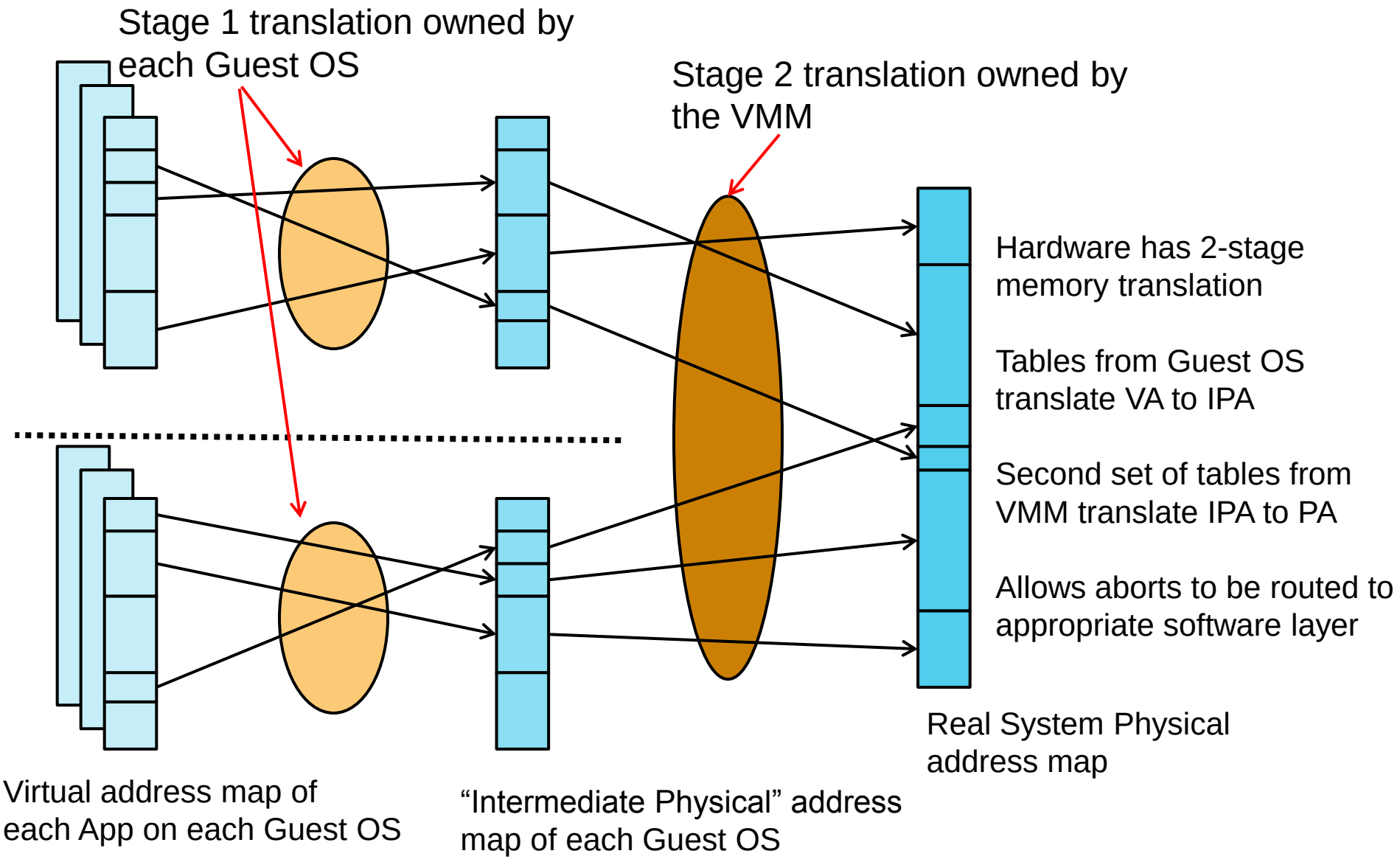
- 4*9*2*3*3*4*5*2 = 25920 total configurations ☹
- Exhaustive crossing of slices, ECC/no-ECC, number of CPUs at unit, top, and system level
- Focused directed testing of less intrusive configuration choices, then pairwise crossing in random testing

Virtualization: A Third Layer of Privilege



- Guest OS same privilege structure as before
  - Can run the same instructions
- New Hyp mode has higher privilege
- VMM controls wide range of OS accesses to hardware

Virtual Memory in Two Stages

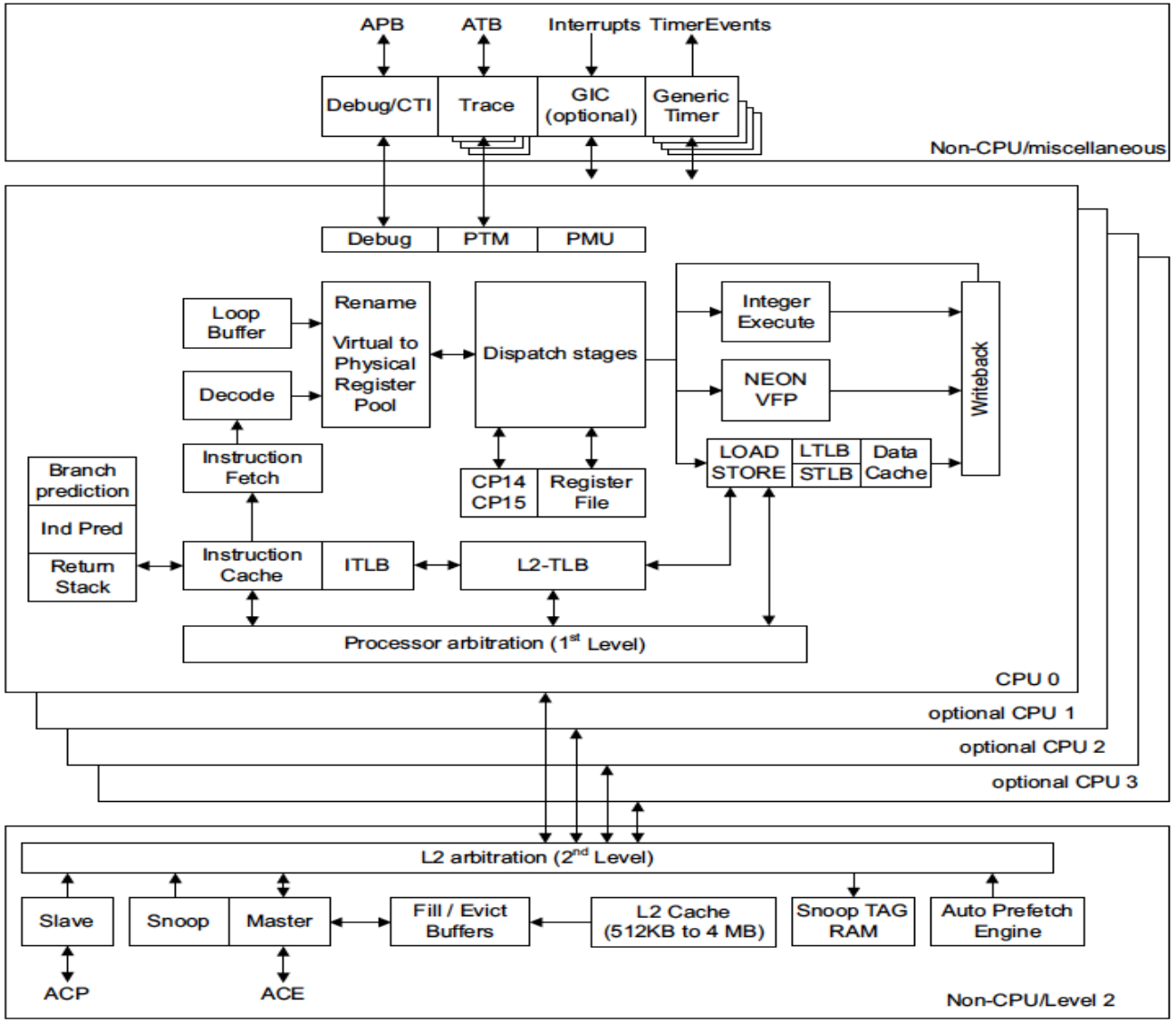


Virtualization - Testing

- Constrained random testing of instruction/event traps at core level
- PageMaker – constrained random generation of LPAE/v7 pages
- Unit level : exhaustive testing of tbw logic in L2TLB/TBW
- PageMaker reused in memory system testbenches and top level testbench
- Independently developed Virtualization AVS
- “Real” hypervisor at system level, running real and rogue OSes/apps

Out of Order Execution

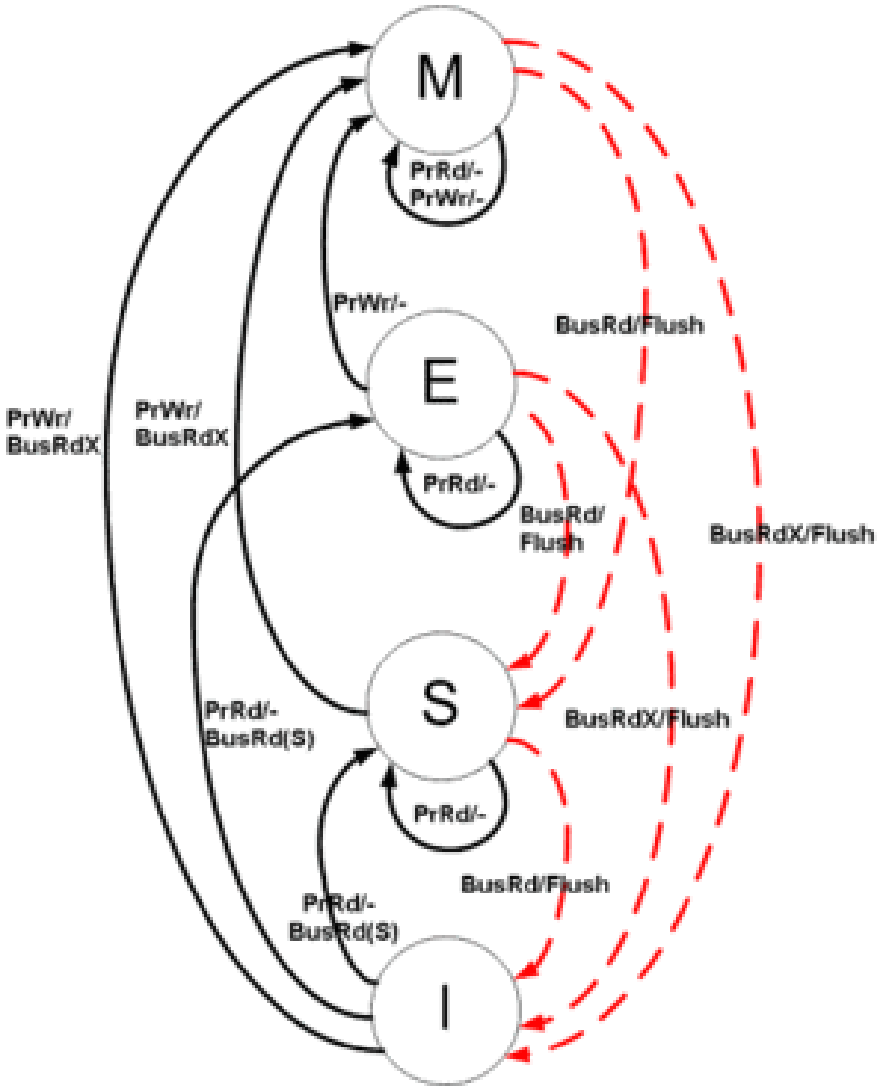
OoO in Cortex-A15



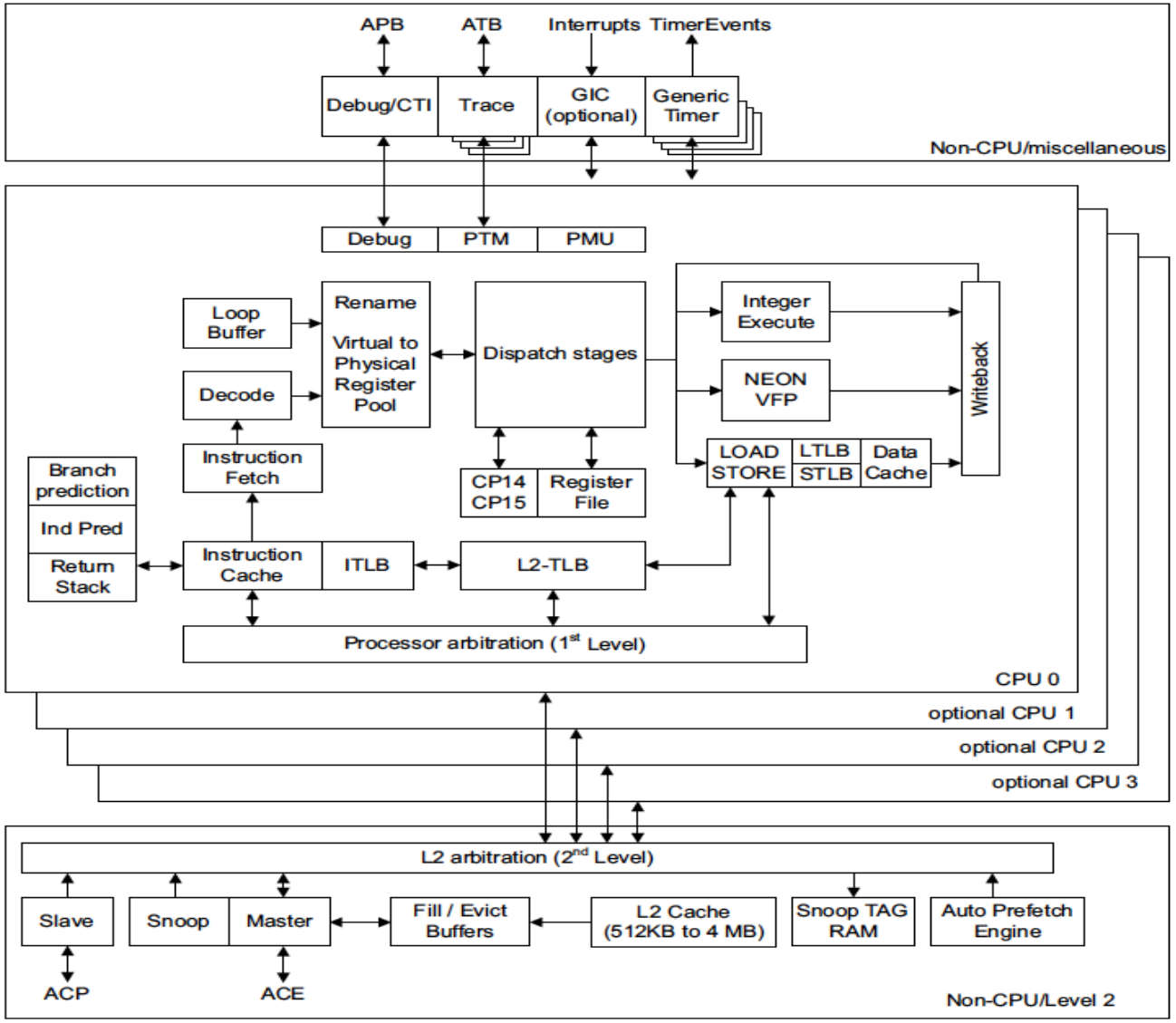
OoO - Testing

- Unit Level : detailed testbench models/checking
- Exhaustive fcov on retire/flush/rebuild scenarios
- Independent architectural checking vs. ISS model, AVS

Hardware Coherence



Hardware Coherence in A15



Hardware Coherence - Testing

- ACE functional coverage and protocol checkers
- Unit level : Detailed white-box modeling/checking
- Multi-unit : LS/L2
  - Focused hazard/starvation scenario testing
  - Global ordering data consistency checker
- Top level : RIS tests
  - False-sharing
  - Non-deterministic sharing
- System level : True-sharing, order-sensitive testing

ARM®