tenstorrent

Atlantis SoC Data Sheet

Version 1.0, 2026-07-02

Table of Contents

Legal Notice

Important Use Notes

This document describes a preliminary version of the Atlantis SoC and is provided for evaluation and planning only. All specifications (features, parameters, performance numbers, pinout, package, boot modes, etc.) are subject to change without notice. This document must not be used as a final manufacturing, safety, or compliance specification. This document must only be used for the purposes of building or using a Tenstorrent product.

No Warranty

While every precaution has been taken in the preparation of this document, it may contain technical inaccuracies, omissions and typographical errors. Tenstorrent is under no obligation to update or otherwise correct this information. Tenstorrent makes no representations or warranties with respect to the accuracy or completeness of the contents of this document, and assumes no liability of any kind, including the implied warranties of noninfringement, merchantability or fitness for particular purposes, with respect to the operation or use of Tenstorrent products described herein.

No Implied License

No license, including implied or arising by estoppel, to any intellectual property rights is granted by this document. Terms and limitations applicable to the purchase or use of Tenstorrent products are as set forth in a signed agreement between the parties or in an applicable Tenstorrent Terms and Conditions of Sale.

Trademarks

A list of Tenstorrent's trademarks can be found at tenstorrent.com/trademarks. Other product names and links to external sites used in this publication are for identification purposes only and may be trademarks of their respective companies.

Third-Party Pointers

Links to third party sites are provided for convenience and unless explicitly stated, Tenstorrent is not responsible for the contents of such linked sites and no endorsement is implied.

Export

The technical data presented in this document may be subject to U.S. and international export, re-export, or transfer laws. Diversion contrary to U.S. and international law is strictly prohibited. The user and recipient of this document must comply with all the applicable regulations and laws which include restrictions on prohibited destinations or countries, end users, and product usage. If in doubt, contact the appropriate US Export Control Agency or Tenstorrent.

Patents

Tenstorrent's patents can be found on its webpage at tenstorrent.com/patents.

Revision History

Date	Version	Changes
June 08, 2026	1.0	First release.

Chapter 1. Overview

Atlantis is a high-performance RISC-V SoC integrating an **8-core Ascalon-X CPU cluster**, a high-performance GPU, and a dual-channel **LPDDR5/4x** memory subsystem. It is intended as a:

- **Reference Platform** for Tenstorrent's Ascalon-X CPU IP and related system IP, and
- **General-purpose RISC-V development SoC** suitable for High-Performance Computing (HPC), robotics, automotive, and Edge AI applications.

Atlantis combines modern RISC-V CPU and virtualization features, a production-class GPU and multimedia pipeline, rich PCIe and storage connectivity, and an integrated security and management subsystem around a multi-NoC fabric with a cache-coherent Ascalon-X CPU cluster.

An Atlantis Evaluation Board is the vehicle to deliver the complete hardware and software stack to developers. Atlantis supports the Ubuntu operating system, providing a familiar environment for platform evaluation, performance benchmarking, and hardware-software co-development.

The figure below shows the top-level Atlantis SoC block diagram, including the Ascalon-X CPU cluster, LPDDR subsystem, PCIe/HSIO, multimedia subsystem, RCPU subsystem, security subsystem, and the multiple NoCs that connect them.

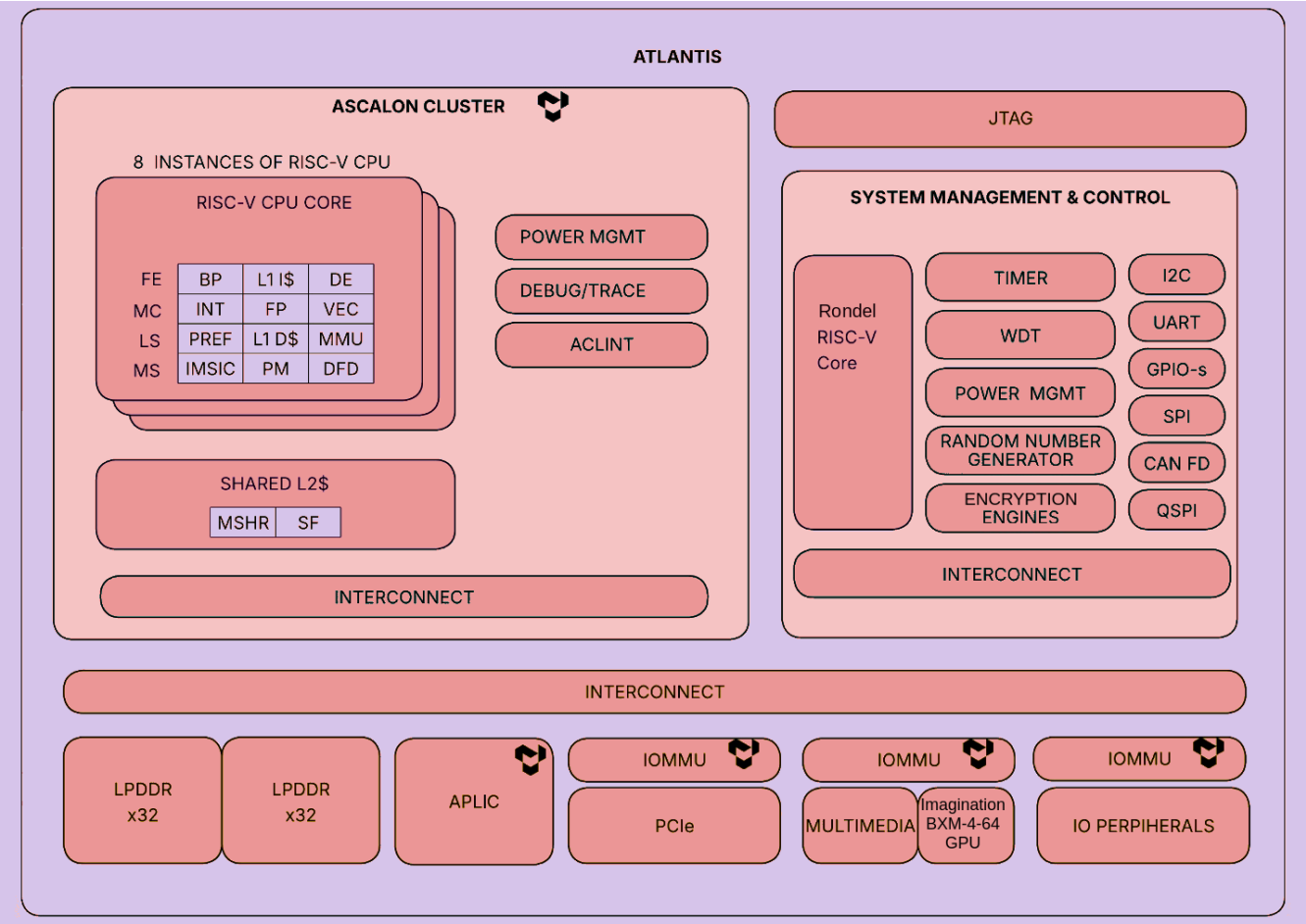

Figure 1. Atlantis SoC top-level block diagram

1.1. General Features

1.1.1. CPU and Compute

- 8 × **Ascalon-X** 64-bit RISC-V cores, out-of-order, high-performance
- Shared **12 MB L2 cache** for the Ascalon-X cluster
- Coherent CHI-E interface from CPU cluster into the SoC fabric
- RISC-V privilege levels and virtual memory support suitable for Linux and type-1 hypervisors (see Ascalon-X IP documentation for full ISA profile)

1.1.2. Memory

- Dual-channel **LPDDR5/4x**, 2 × 32-bit, up to **64 GB** total
- LPDDR5 up to **6400 MT/s**; LPDDR4/4x up to **4266 MT/s**
- Controllers at 800 MHz with interleaving and **in-line ECC**
- Support for multiple DRAM vendors and densities (configurable via controller registers and board design)

1.1.3. Graphics, Video, and Display

- High-Performance **Imagination BXM-4-64** GPU
- APIs: **OpenGL ES 3.2**, **OpenCL 3.0**, **Vulkan 1.3** (Linux)
- Multi-format **hardware video decode** for HEVC/H.264/VP9/MPEG2/others at up to 4K60 / 8K-class resolutions
- Display outputs: **HDMI 2.0** and **4-lane MIPI-DSI** TX, plus flexible composition and scaling pipeline

1.1.4. Connectivity and Storage

- **PCIe Gen4 x20** total lanes (configurable as 1×16 + 1×4 or 2×8 + 1×4)
- 2 × **USB 3.1 Gen2** + 2 × **USB 2.0**
- 2 × **Gigabit Ethernet MAC** (AVB/TSN-capable, IEEE 1588 PTP)
- 2 × **eMMC5.1 / SDIO 3.0** interfaces
- SPI / DSPI / QSPI for SPI NOR / SPI NAND boot and storage

1.1.5. Management and Real-Time

- 32-bit RISC-V Real-time CPU (RCPU) subsystem for boot, power, system management, and safety hooks
- Integrated timers, watchdogs, PVT monitors, and GPIO with rich low-speed I/O
- Dedicated **System Management NoC (SMN)** for configuration and control traffic

1.1.6. Security

- Crypto engines: AES-128/256, SHA-256/384, DES-64
- True Random Number Generator (TRNG)
- 16 KB OTP (eFuse) and secure-boot-capable security subsystem
- Firewalls and IOMMUs on major NoCs and master ports for isolation and virtualization

1.1.7. Process, Package, Power

- **12FFC**, 0.8 V core (nominal)
- FCBGA package, **~35x35 mm**, 8-layer substrate (preliminary)
- TDP (Thermal Design Power) target: **To be determined**. TDP will be quoted in a future revision once power characterization is complete.
- Junction temperature: −40 °C to +125 °C (target)
- Ambient: 0 °C to +85 °C (target)

Chapter 2. Specifications

2.1. CPU Subsystem (Ascalon-X Cluster)

The Atlantis SoC is built around a high-performance **8-core Ascalon-X RISC-V cluster** serving as the main application processor subsystem. Designed by Tenstorrent, the Ascalon-X core is a 64-bit, out-of-order, superscalar RISC-V processor optimized for high instructions-per-cycle (IPC), general-purpose computing, virtualization, and Linux-class operating systems. The Atlantis platform represents Tenstorrent's first silicon implementation of the Ascalon-X IP, showcasing its capabilities in a production SoC environment.

The Ascalon-X cluster integrates:

- **8 high-performance Ascalon-X cores** with private L1 caches
- **12 MB shared L2 cache** (last-level cache, LLC) for the cluster
- **CHI-E (Coherent Hub Interface - E variant) interface** for coherent interconnect
- **Advanced Interrupt and Timer subsystems** (APLIC, ACLINT) per RISC-V specifications
- **Debug and trace infrastructure** for development and production diagnostics

The cluster is the primary compute engine for OS, middleware, and application workloads, while the separate RCPU Section 2.2 handles low-level system management and real-time control.

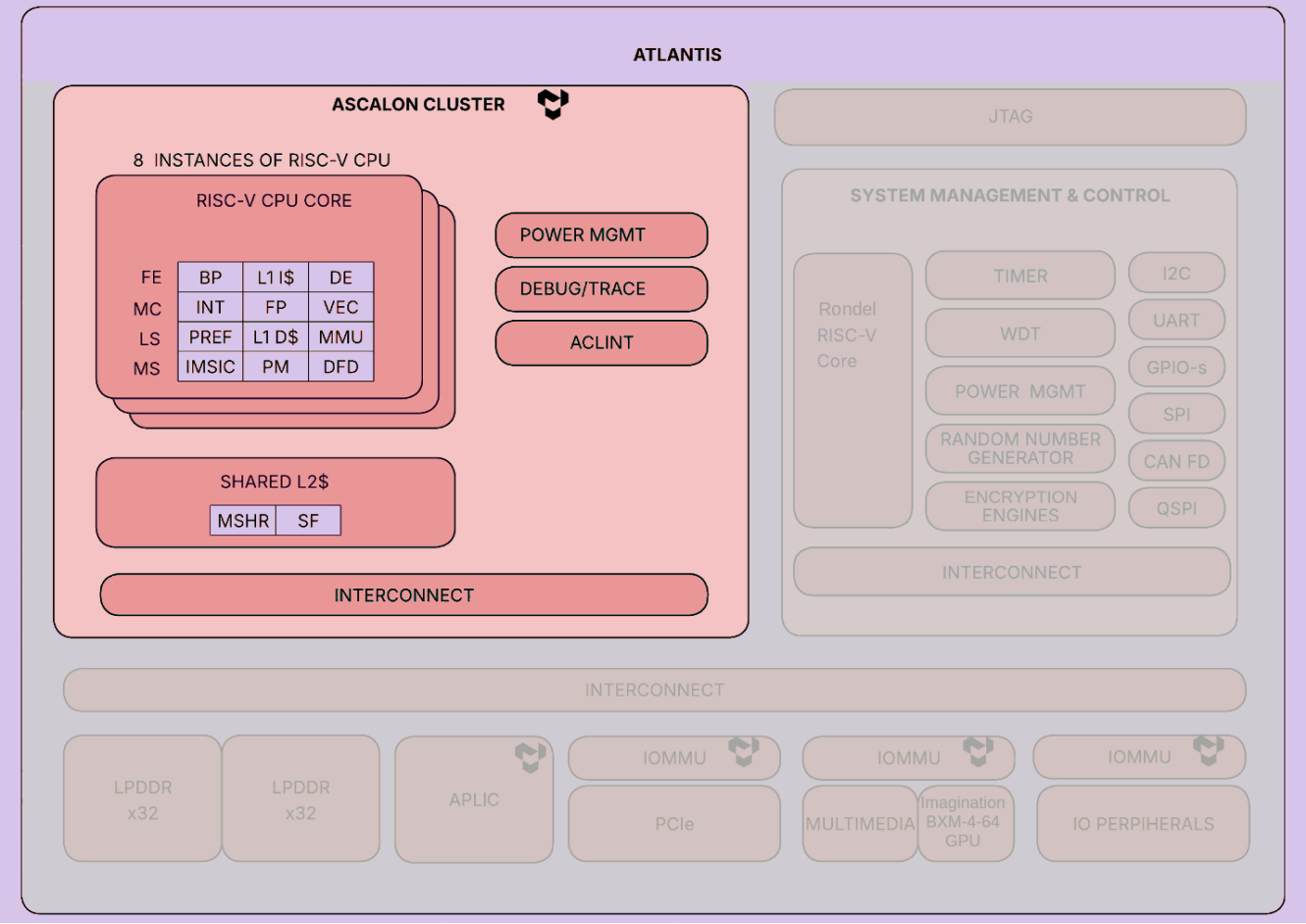

Figure 2. Atlantis CPU Cluster Block Diagram

2.1.1. Ascalon-X Core Architecture

The Ascalon-X core is a 64-bit, out-of-order, superscalar RISC-V application processor with **8-wide instruction decode**, engineered for high instructions-per-cycle (IPC) across general-purpose, server, AI, and edge workloads. Ascalon-X is **RVA23 application-profile compliant**, ensuring out-of-the-box compatibility with standard RISC-V operating systems, toolchains, and software ecosystems. The architecture scales across the Ascalon-X family from wide, high-performance implementations down to compact, power-efficient configurations; Atlantis integrates the high-performance variant.

Instruction Set Architecture (ISA)

- 64-bit RISC-V, RVA23 application-profile **compliant**
- **RV64GC** base: integer, multiply/divide, atomics, single- and double-precision floating-point, and compressed instructions
- **Vector (RVV 1.0)** with dual 256-bit vector datapaths for SIMD, DSP, and AI/ML kernels
- **Hypervisor (H) extension** for hardware-assisted virtualization
- Bit-manipulation and scalar **cryptography** extensions per the RVA23 profile
- Cache-management and additional RVA23 extensions for modern OS and runtime support

Privilege Levels and Virtualization

- All three standard RISC-V privilege levels: **Machine (M)**, **Supervisor (S)**, and **User (U)**
- **Hypervisor (H) extension** with **Virtualized Supervisor (VS)** and **Virtualized User (VU) modes**
- Hardware-assisted virtualization suitable for type-1 and type-2 hypervisors and container runtimes
- Architectural **security and side-channel mitigations** and **RAS** (reliability, availability, serviceability) features

Virtual Memory and Address Translation

- Per-core **memory management unit (MMU)** with hardware page-table walking
- Standard RISC-V paged virtual memory with **two-stage (guest/host) translation** under the hypervisor extension
- **Multi-level TLB hierarchy** with large-page support to reduce address-translation overhead

Microarchitecture

- **8-wide**, out-of-order, superscalar execution with speculation and in-order retirement for precise exceptions
- Deep out-of-order instruction window with **register renaming** to maximize extracted instruction-level parallelism
- Rich execution resources: **six integer ALUs** (two with integrated branch resolution), **three load/store pipelines**, **two floating-point datapaths**, and **two 256-bit vector datapaths**
- **Advanced branch prediction** (TAGE-class direction and target prediction with a return-address stack) and **hardware prefetchers**
- High-bandwidth instruction fetch (**32 bytes per cycle**) feeding the wide decode front-end
- Out-of-order load/store unit with **store-to-load forwarding**

2.1.2. Cache Hierarchy

L1 Caches (Per Core)

- L1-I : 64KB, 8-way set associative
- L1-D : 256KB, 4-way set associative
- Non-blocking caches support multiple outstanding misses

L2 Cache (Shared, Cluster-Level)

- **Size: 12 MB** shared across 8 cores (last-level cache, LLC)
- **Associativity**: 16-way+ set-associative, 64B line
- **Coherence**: **MOESI** protocol with cache-to-cache transfers, coherence directory/snoop filter
- **Features**: Optional way partitioning/QoS, LRU/pseudo-LRU replacement, shared L2 prefetcher

2.1.3. Cache Management

- Hardware-managed coherence (MESI L1, MOESI L2)
- Zicbom (invalidate/clean/flush), Zicboz (block zero), Zicbop (prefetch hints)
- Optional cache locking for real-time determinism

2.1.4. Coherency and Interconnect

Interconnect

- **CHI-E protocol**
  - ARM AMBA 5 CHI (Coherent Hub Interface) variant optimized for embedded SoCs
  - Credit-based flow control and split-transaction protocol
  - Supports snoops, cache maintenance operations, and distributed virtual memory (DVM) operations
- **MMCAB / SCB (CHI-to-AXI Bridge)**
  - Converts CHI-E transactions from Ascalon-X cluster to AXI for SoC fabric
  - Bridges coherent domain (Ascalon-X cluster) to non-coherent or IO-coherent peripherals
- **NOCC (Core NoC)**
  - High-bandwidth, low-latency interconnect for CPU-to-peripheral traffic
  - The SoC NoC is AXI-based (implementation-dependent)
  - Provides access to
    - **NOCM (Memory NoC)** for DRAM (LPDDR5/4x)
    - **HSIO subsystem** (USB, PCIe, Ethernet)
    - **Multimedia subsystem** (GPU, VPU, display)
    - **Peripherals** (I2C, SPI, UART, GPIO)
- **IO-coherent access**
  - Designated AXI ports support IO coherence for DMA masters (e.g., PCIe, GMAC, GPU)
  - DMA masters can snoop CPU caches via CHI-E or cache coherent interconnect
  - Reduces software overhead for buffer synchronization (no explicit cache flushes needed)

2.1.5. Memory Ordering and Synchronization

- **Memory consistency model**: RISC-V Weak Memory Ordering (RVWMO)
  - Relaxed memory model for high-performance
  - Explicit fence instructions (FENCE, FENCE.I, SFENCE.VMA) for ordering guarantees
- **Atomic operations** (A extension)
  - Load-Reserved / Store-Conditional (LR/SC) for lock-free algorithms
  - Atomic Memory Operations (AMO): atomic add, swap, AND, OR, XOR, min, max
  - Support for acquire/release semantics (.aq / .rl suffixes)
- **Barriers and synchronization**
  - **FENCE** – Memory ordering barrier (I/O, read, write ordering)
  - **FENCE.I** – Instruction fetch fence (synchronize instruction cache after code modification)
  - **SFENCE.VMA** – TLB invalidation and page table synchronization

2.1.6. Interrupts and Timers

Advanced Interrupt Architecture (AIA)

- **APLIC (Advanced Platform-Level Interrupt Controller)**
  - Cluster-level interrupt controller for external interrupts
  - Supports **512 interrupt** sources (total for cluster)
  - **Message-Signaled Interrupts (MSI)**
    - **M-mode MSI**: 512 interrupts
    - **S-mode MSI**: 512 interrupts
    - **VS-mode MSI**: 63 interrupt identifiers per guest file (for virtualization)
  - **Wire interrupts**: Traditional level/edge-triggered interrupts
  - Interrupt priority, masking, and targeting (per-core interrupt routing)
- **ACLINT (Advanced Core-Local Interruptor)**
  - Per-core local interrupts and timers
  - **Machine-mode timer interrupt (MTI)**: 64-bit timer with compare register
  - **Software interrupts (MSI)**: Inter-processor interrupts (IPI) for core-to-core signaling
  - Supervisor-mode timer interrupt (STI) support
- **IMSIC (Incoming MSI Controller)** (optional, implementation-dependent)
  - Per-core MSI interrupt files for high-performance interrupt delivery
  - Reduces interrupt latency for virtualized environments
- **NMI (Non-Maskable Interrupt)**
  - **NMI switch block** routes critical external interrupts to core NMI pins
  - Use cases: safety-critical events, platform errors (ECC errors, thermal shutdowns)
  - NMI bypasses standard interrupt masking for guaranteed delivery

Timers

- **Per-core timers** (via ACLINT)
  - **64-bit machine-mode timer (M-timer)**: Real-time counter with compare register
  - **Supervisor-mode timer (S-timer)**: Virtual timer for OS use
  - Timer interrupt on match (periodic or one-shot)
- **Timer frequency**: Derived from system clock **(100MHz)**. (See Section 2.7.2).
- **Integration with broader SoC interrupt hierarchy**
  - RCPU interrupts routed via SMN (System Management NoC).
  - Peripheral interrupts aggregated and forwarded to APLIC

2.1.7. Performance Monitoring

Performance Monitoring Unit (PMU)

- **RISC-V PMU (Zihpm extension)** compliant
- **Hardware performance counters (HPCs)**
  - **Basic counters** (RISC-V standard):
    - **CYCLE** - Cycles elapsed (64-bit)
    - **TIME** - Wall-clock time (64-bit)
    - **INSTRET** - Instructions retired (64-bit)
  - **Event counters** (implementation-specific, 4-8 programmable counters):
    - L1 I-cache hits/misses

        - L1 D-cache hits/misses, load/store operations
        - L2 cache hits/misses
        - Branch predictions, mispredictions
        - TLB hits/misses
        - Instruction fetch stalls, decode stalls, execution stalls
        - Vector instruction counts (if RVV enabled)
        - Floating-point operation counts
- **Event selection and filtering**:
    - Programmable event selectors per counter
    - Filtering by privilege level (M/S/U mode)
    - Overflow interrupts for event counting and sampling-based profiling
- **Profiling and tracing integration**:
    - PMU data exported via debug/trace infrastructure
    - Integration with Linux perf, RISC-V trace tools

2.1.8. Debug and Trace

RISC-V Debug Specification Compliance for Ascalon-X

- **RISC-V Debug Spec** 1.0
- **JTAG-accessible debug** via SoC debug fabric
    - External JTAG TAP (Test Access Port) for debug probe connection
    - Debug Module (DM) provides access to all cores in cluster
- **Debug features**:
    - **Halting debug**: Halt individual cores or all cores (cluster-wide halt)
    - **Single-step execution**: Step through instructions for debugging
    - **Breakpoints and watchpoints**:
        - Hardware breakpoints (instruction address match) – 4-8 per core
        - Hardware watchpoints (data address match for load/store) – 2-4 per core
    - **Abstract commands**: Read/write registers, memory access while core is halted
    - **System bus access**: Debug module can access system memory independently of cores
- **Debug triggers**: Hardware breakpoints and watchpoints are supported. Full Sdtrig compliance is not claimed for the current version of Ascalon-X.

Trace Infrastructure

- **RISC-V N-Trace (Processor Trace)**
    - Hardware trace encoder generates compressed instruction and data trace packets
    - Trace packets exported via dedicated trace port or embedded trace buffer (ETB)
    - Off-chip trace capture or on-chip trace memory (implementation-dependent)
- **Trace features**:
    - **Instruction trace**: Branch outcomes, jump targets, executed instructions
    - **Data trace**: Load/store addresses and data values
    - **Cycle-accurate or timestamp-based trace** for performance analysis
- **Integration with SoC DFD (Design-For-Debug)**:
    - **Scan dump**: Capture core state (register file, pipeline state) for failure analysis
    - **MBIST (Memory Built-In Self-Test)**: On-chip memory testing (cache, register file)
    - **Trace capture buffers**: Store trace data on-chip for post-mortem debugging

Debug Access and Security

- **Secure debug**: Debug access controlled by security policies (OTP fuses, firmware)
  - Debug locked in production devices or customer-controlled unlock
- **Non-invasive debug (trace without halting) vs. invasive debug (halt, step, breakpoint)**
- **Multi-core debug**: Debug all 8 cores simultaneously or individually

2.1.9. Power Management

Dynamic Frequency Scaling (DFS)

- Per-cluster DFS - Frequency adjusted dynamically based on workload
  - Frequency range: Target maximum frequency is **1.38 GHz**. Nominal State (**1.38 GHz**) and Low Power State (500 MHz).
- DFS control: orchestrated by the in-cluster Cluster Power Management Logic (CPL) firmware under supervision of the RCPU system-management firmware
  - Thermal and power feedback from PVT controller (See Section 2.7.3).

Clock Gating

- **Dynamic clock gating**: Idle functional units clock-gated to reduce dynamic power
  - Per-core, per-pipeline-stage, per-execution-unit granularity
- **WFI (Wait-For-Interrupt)** instruction: Core enters low-power idle state until interrupt arrives
  - Clock-gated, reduced power consumption
  - Fast wake (< 1 µs typical)

Power States

Ascalon-X supports C2, C3, and C4 runtime core idle states, and CLC3 and CLC4 cluster-level idle states.

- **C0** (Active): Normal execution.
- **C2**: Shallow core idle state with core clocks gated.
- **C3**: Deeper core idle state with core clocks gated.
- **C4**: Deep core idle state with core clocks gated and L1 caches flushed.
- **CLC3**: Cluster idle state with enabled cores in C3 or deeper, cluster clock/PLL off, and shared cache retained.
- **CLC4**: Deep cluster idle state with enabled cores in C4, shared cache flushed, cluster interfaces drained/disabled, and cluster clock/PLL off.

2.1.10. Physical Implementation

- **Process technology**: TSMC 12FFC (12 nm FinFET Compact)
- **Core count**: 8 cores per cluster (single cluster in Atlantis).

2.1.11. Software and Ecosystem Support

Operating Systems

- **Linux kernel**: Mainline RISC-V support (kernel 5.x, 6.x)
  - SMP (Symmetric Multi-Processing) for all 8 cores
  - KVM hypervisor support (with H-extension)
- **FreeBSD, OpenBSD**: RISC-V ports available
- **Real-time OS (RTOS)**: FreeRTOS, Zephyr

Toolchain and Compilers

- **GCC**: RISC-V GCC 12.x+ with RV64GC, RVV, H-extension support
- **LLVM/Clang**: RISC-V backend with vector and bitmanip extensions
- **Binutils**: Assembler, linker, objdump for RISC-V
- **GDB**: RISC-V debugging support

Virtualization and Containers

- **KVM (Kernel-based Virtual Machine)**: Type-2 hypervisor on Linux
- **Xen**: Type-1 hypervisor (RISC-V port in progress)
- **Docker, LXC**: Container runtimes on RISC-V Linux

2.2. Real-Time / Management CPU (RCPU Subsystem)

Atlantis includes a 32-bit **RISC-V core** (RCPU) used for boot, low-level control, and ongoing system management. The RCPU subsystem provides a microcontroller-class execution environment with tightly coupled memories, its own memory view, and a dedicated address-translation mechanism to reach the full 64-bit system address space. It controls clocks, resets, power states, PVT monitoring, and low-speed peripherals through the **System Management NoC (SMN)**.

The core is a high-performance 32-bit RISC-V embedded processor featuring a **6-stage, dual-issue superscalar pipeline**, with tightly-coupled memories (TCMs), advanced interrupt controller (CLIC), and comprehensive debug/trace capabilities. It is designed for real-time, system management, and safety-critical applications requiring deterministic execution and low interrupt latency.

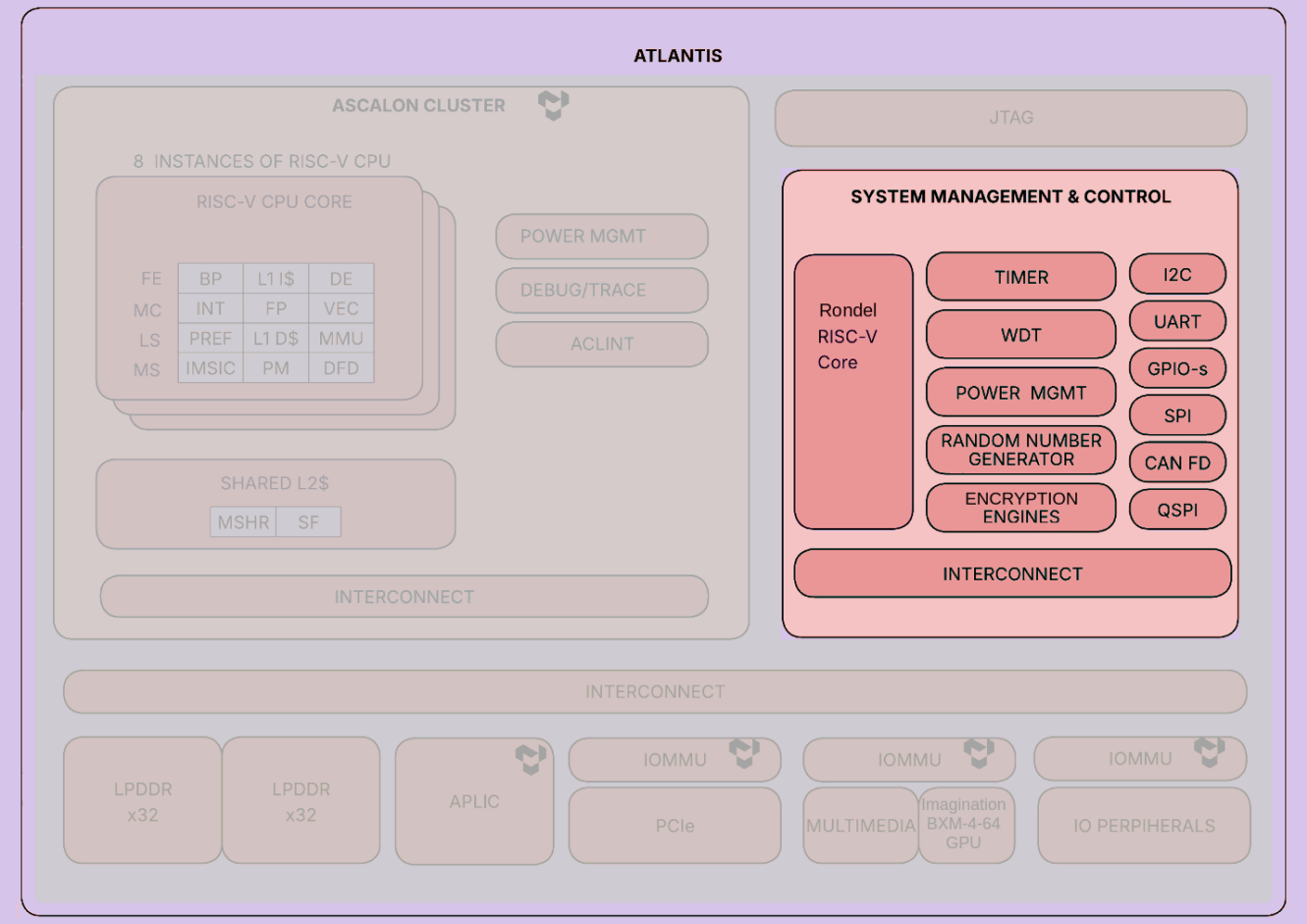

Figure 3. Atlantis RCPU Block Diagram

2.2.1. Core Features

ISA and Extensions

- Baseline RV32IM
- **ISA**: RV32IMAFDC_B_Zcb_Zicbom (Zicsr, Zifencei included)
- **Privilege modes**: Machine (M-mode), User (U-mode)
- **Physical Memory Protection (PMP)**: 16 configurable regions
- **Physical Memory Attributes (PMA)**: Cacheable/non-cacheable, idempotent/non-idempotent attributes

Microarchitecture

- 6-stage, dual-issue (2-wide) superscalar pipeline
- **Mostly in-order execution** (out-of-order division)
- **Branch prediction**: BHT, BTB, RAS
- **Execution units**: Integer ALU, multiply/divide, FPU (F/D), DSP (PEXT, optional), load-store unit
- **Performance**: 5.5 CoreMark/MHz

2.2.2. Memory Hierarchy

Tightly-Coupled Memories (TCMs)

- **ITCM**: **256 KB**, 0-wait-state, ECC disabled
- **DTCM**: **256 KB**, 0-wait-state, ECC disabled
- **DMA access**: External masters can access TCMs via AXI/AHB slave interface

Caches

- **I-cache**: **32 KB**, 4-way set-associative, 32-byte line, pseudo-LRU, ECC disabled
- **D-cache**: **32 KB**, 4-way set-associative, 32-byte line, pseudo-LRU, write-back, ECC disabled
- **Coherency**: Software-managed (CMO instructions for cache flush/invalidate), no hardware coherency with Ascalon-X

2.2.3. Low-Speed Peripherals and System I/O

Serial and Control I/O

Atlantis provides extensive serial and control I/O connectivity for embedded system integration, including UART, I²C, SPI, and CAN-FD interfaces for communication with external peripherals, sensors, and subsystems.

UART (Universal Asynchronous Receiver/Transmitter)

- **Instances**: 6 UART controllers
- **Standards**: 16550-compatible
- **Features**: 256-byte TX/RX FIFOs, hardware flow control (RTS/CTS), DMA support, programmable baud rates (300-3.686 Mbps)
- **Use cases**: Serial console, GPS, Bluetooth/Wi-Fi modules, industrial sensors

I²C (Inter-Integrated Circuit)

- **Instances**: 8 I²C controllers
- **Standards**: I²C v3.0 compliant, SMBus compatible
- **Modes**: Master and slave, standard (100 kHz), fast (400 kHz), fast-plus (1 MHz), high-speed (3.4 MHz)
- **Features**: 7-bit and 10-bit addressing, multi-master arbitration, clock stretching, DMA support
- **Use cases**: PMICs, sensors, EEPROMs, touchscreens, audio CODECs

SPI (Serial Peripheral Interface)

- **Instances**: 6 SPI controllers
- **Features**: Master and slave modes, full-duplex, up to 50 MHz clock, programmable frame size (4-32 bits)
- **Modes**: SPI modes 0-3 (CPOL/CPHA configurable)
- **Advanced**: DMA support, multi-slave chip select, FIFO buffers
- **Use cases**: Flash memory, displays, ADCs/DACs, sensors

CAN-FD (Controller Area Network with Flexible Data-Rate)

- **Instances**: 2 CAN-FD controllers
- **Standards**: CAN-FD (ISO 11898-1:2015), CAN 2.0 Part B, ISO 16845-1:2016 conformance
- **Identifiers**: 11-bit standard, 29-bit extended
- **Data**: 0-64 bytes (CAN-FD), 0-8 bytes (CAN 2.0)
- **Bit rates**: Arbitration up to 1 Mbps, data phase up to 5 Mbps (CAN-FD)
- **Features**: 128 message buffers, receive FIFO, ID filtering, time synchronization
- **Use cases**: Automotive, industrial control

Timers, Watchdogs, and PVT

Atlantis integrates comprehensive timing, watchdog, and process/voltage/temperature (PVT) monitoring capabilities for system management, real-time control, and reliability.

Timers

- **General-purpose timers**: 8 instances, 32-bit or 64-bit counters, configurable prescalers, up/down counting
- **RISC-V timers**: Per-core mtime/mtimecmp (Machine Timer), ACLINT (Advanced CLINT) for inter-core timer interrupts
- **RTC (Real-Time Clock)**: Battery-backed or crystal oscillator, calendar functions, alarm/wakeup
- **Use cases**: Task scheduling, timeout detection, periodic interrupts, timestamping

Watchdog Timers

- **Instances**: 4 watchdog timers
- **Features**: 32-bit down-counter, configurable timeout, reset or interrupt on expiration, write-protected control
- **Modes**: Standard watchdog, window watchdog (early/late timeout detection)
- **Use cases**: System hang detection, safety-critical monitoring, automatic recovery

PVT Sensors (Process, Voltage, Temperature)

- **Temperature sensors**: Multiple on-die sensors, ±5°C accuracy typical, -40°C to 125°C range
- **Voltage monitors**: Core and I/O rail monitoring, under/over-voltage detection
- **Thermal management**: RCPU reads sensors, implements shutdown policies
- **Use cases**: Thermal monitoring and emergency shutdown, reliability monitoring

PWM (Pulse Width Modulation)

- **Instances**: 4 PWM controllers
- **Features**: Configurable frequency/duty cycle, multiple channels per instance
- **Use cases**: Fan control, LED dimming, motor control, audio generation

2.2.4. System Integration

Bus Interfaces

- **AXI5 master** (64-bit): Connects to System Management NoC (SMN) for DRAM and peripheral access
- **AHB5 master** (32-bit): Low-latency peripheral register access (GPIO, I²C, SPI, UART, timers, PVT sensors)
- **AXI/AHB slave**: TCM access by external masters (firmware download, shared buffers, debug)
- **Address Translation Unit (ATU)**: Extends 32-bit RCPU address space to 64-bit system address space

Interrupt and Exception Handling

- **CLIC (Core-Local Interrupt Controller)**: RISC-V CLIC v0.8 compliant
  - **128 external interrupt sources**
  - 1–16 priority levels: **16**
  - Level/edge-triggered, vectored/non-vectored modes, hardware context save
- **CLINT**: Legacy interrupt controller for backward compatibility
- **PMP exceptions**: Access-violation handling across the 16 PMP regions

Debug and Trace

- **RCPU**: **RISC-V Debug v0.13.2** compliant, JTAG interfaces, progbuf and SBA modes
- **Hardware breakpoints/watchpoints**: Instruction and data address matching
- **N-Trace**: Program flow tracing with compression, instruction and optional data trace
- **IDE**: Development environment with integrated debugger and trace analyzer

Performance Monitoring

- **Standard counters**: mcycle, minstret
- **Programmable counters**: 4 typical (mhpmevent3-6), monitor cache miss, branch misprediction, stalls, ITCM/DTCM access (mhpmevent3h-mhpmevent6h).

2.2.5. System Management

The RCPU serves as the **system management and boot controller** for Atlantis.

Boot Sequence

- RCPU executes first from Boot ROM (FSBL, SoC initialization)
- Initializes clocks, PLLs, LPDDR5/4x (PHY calibration, training)
- Configures security (OTP keys, crypto verification, IOMMUs, firewalls)
- Releases Ascalon-X cluster after boot validation

Ongoing Management

- **PVT monitoring**: Thermal and voltage status, emergency shutdown
- **Power control**: clock-gating control and subsystem clock/reset sequencing
- **Watchdog**: Timeout handling, reset control
- **Peripheral initialization**: GPIO, I²C, SPI, UART, CAN, board-level devices
- **Error monitoring**: Security subsystem, IOMMU, firewall event handling
- **Inter-processor communication**: Mailbox and shared memory for RCPU ↔ Ascalon-X communication

2.2.6. Physical Implementation

- **Process**: 12FFC (12 nm FinFET Compact)
- **Frequency**: Up to 600 MHz, 0.8 V

2.3. GPU and Multimedia Subsystem (MM_SS)

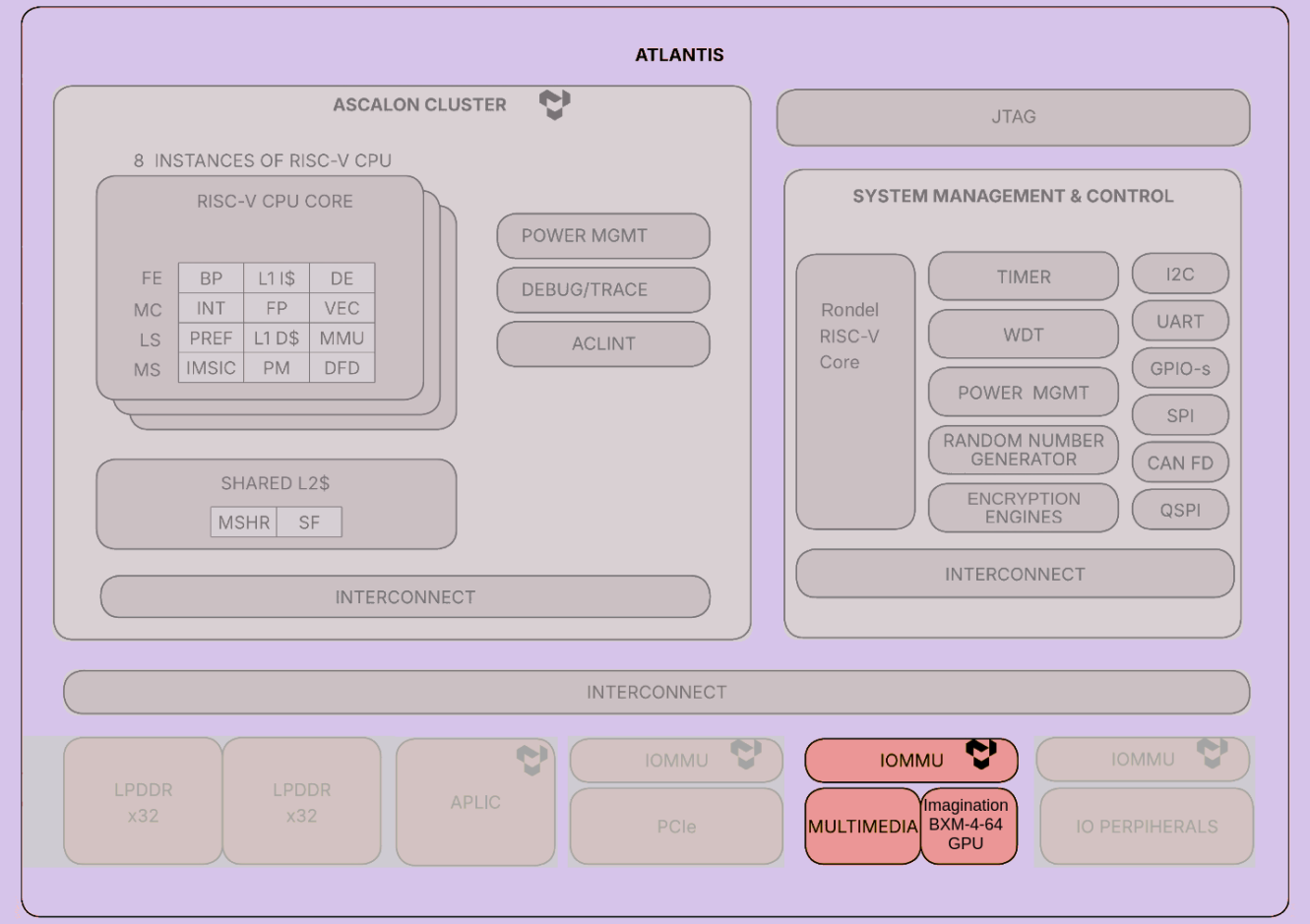

Figure 4. Atlantis GPU and Multimedia Subsystem Block Diagram

2.3.1. GPU (Imagination BXM-4-64)

The Atlantis multimedia subsystem integrates a high-performance Imagination BXM-4-64 GPU core to provide modern 3D graphics, GPU compute, and UI acceleration capabilities. This GPU architecture is built around multi-threaded Unified Shading Clusters (USCs) which feature an ALU architecture with high SIMD efficiency, and support tile-based deferred rendering (TBDR) with concurrent processing of multiple tiles. The GPU is connected to the LPDDR subsystem through the MM-NoC and NOCM, providing unified memory access for CPU and GPU workloads. It targets use cases such as UI rendering, visualization, moderate -complexity 3D content, and GPGPU-style workloads under Linux.

Graphics and Compute API Support

- Fully compliant with major graphics and compute APIs:
  - **OpenGL ES 3.2**
  - **EGL 1.4**
  - **Vulkan 1.3**
  - **OpenCL 1.2 EP, 2.1 EP, and 3.0 EP** (Embedded Profile)
  - **Android NN HAL** support for neural network acceleration

Architecture and Rendering Pipeline

- **Tile-Based Deferred Rendering (TBDR)** architecture for 3D graphics workloads
  - Concurrent processing of multiple tiles for improved efficiency
  - Maximum **6 tiles in flight** per Image Synthesis Processor (ISP)
  - On-chip tile floating point depth buffer
  - 8-bit stencil with on-chip tile stencil buffer

    - **32 parallel depth/stencil tests** per clock
- **Programmable high-quality image anti-aliasing**
    - Maximum **4× multi-sampling** (MSAA)
    - Programmable AA quality settings
- **Fine-grain triangle culling** for improved rendering efficiency
    - Early hidden object removal
    - Vertex compression
    - Tile acceleration

Unified Shading Cluster (USC) Engine

- **Multi-threaded USC** incorporating:
    - Pixel shader (fragment processing)
    - Vertex shader (geometry processing)
    - GPGPU compute shader functionality
- **USC ALU Architecture**:
    - **4 ALU pipelines** with **16 parallel instances** per clock
    - High SIMD efficiency design
    - USC F16 Sum-of-Products Multiply-Add (SOPMAD) ALU
    - Scalar and vector SIMD execution model
    - Variable length instruction set encoding
- **Per-USC Caches**:
    - Local data cache
    - Texture cache
    - Instruction cache
- **Full support for OpenCL atomic operations**

Memory Architecture

- **Fully virtualized memory addressing**
    - Up to **64 GB address space**
    - Unified Memory Architecture (UMA) with CPU
- **System Level Cache (SLC)**
    - Customer-configurable size
    - Shared between GPU modules for bandwidth reduction
- **Specialized Texture Cache Unit (TCU)**
    - Dedicated texture fetch and filtering unit
    - Compressed texture decoding support
- **Advanced DMA-driven operation** for minimum host CPU interaction

Data Compression Technologies

- **PVRGC (PowerVR Geometry Compression)**
    - Lossless geometry compression performed during geometry processing phase
- **TFBC (Texture and Framebuffer Compression)**
    - Lossless and/or visually lossless low-area image compression
    - Multiple levels of lossy compression: 75%, 50%, or 25% of original data size
    - Block size: 64 × 32-bit pixels (8×8 or 16×4 shape)
    - Throughput: up to 4 pixels × 4 channels × 8 bits per clock

    - Supports 1, 2, 3, or 4 channels of U8, U16, U32, F16, or F32
    - Per-plane YUV planar (2 or 3 plane) video compression
- **Compressed Texture Format Support**:
    - ASTC LDR (Adaptive Scalable Texture Compression)
    - ETC (Ericsson Texture Compression)
    - TFBC for non-compressed and YUV textures

Texturing Capabilities

- **Texture Resolution**:
    - Maximum texture size: **8K × 8K** (8192 × 8192)
    - YUV planar support
    - 10-bit sRGB and YUV format support
- **Filtering**:
    - Point, bilinear, and trilinear filtering
    - Anisotropic filtering
    - Corner filtering for cube environment mapped textures
- **Texture Operations**:
    - Load from source instruction support
    - Texture writes through Texture Processing Unit

Rendering Features

- **Resolution Support**:
    - Frame buffer max size: **8K × 8K** (8192 × 8192)
- **Render to Buffers**:
    - Multiple on-chip render targets (MRT)
    - Twiddled format support
    - Lossless and/or visually lossless framebuffer compression
- **Programmable Geometry Shader Support**
- **Direct Geometry Stream Out (Transform Feedback)**
- **Fixed-Function Rasterization**:
    - **1 fixed-function rasterization pipeline**
    - Deferred pixel shading

Compute Features

- **1, 2, and 3-dimensional compute primitives**
    - Block DMA to/from USC Common Store (for local data)
    - Per-task input data DMA (to USC Unified Store)
- **Conditional execution and execution fences**
- **Compute workload can be overlapped with any other workload**
- **Round to nearest even** arithmetic mode

Virtualization and Multi-Context

- **Imagination BXM-4-64 GPU Virtualization Support (Full Virtualization)**:
    - Up to **8 virtual GPU** instances
    - Each guest OS runs full graphics driver stack independently
    - Concurrent command submission from multiple OSs

- **Hyperlane Technology:**
  - **8 hyperlanes** available
  - Separate IRQs per OS ID (OSID)
  - Multi-lane isolation for security and performance
- **Context Switching (Low-Latency Scheduling):**
  - Fine-grained task switching and workload balancing
  - Geometry phase: stops at end of Index List Block
  - Fragment phase: stops at end of tile
  - Compute: stops at end of workgroup (max 512 work-items)

Security

- **DRM (Digital Rights Management) security support**
  - Protected content rendering paths
  - Secure memory access

Firmware Processor

- **Dedicated processor for Imagination BXM-4-64 GPU firmware execution:**
  - Single-threaded firmware processor
  - **2 KB instruction cache**
  - **2 KB data cache**
  - Separate power island for firmware processor
  - Handles GPU core management tasks

Performance Monitoring

- **On-chip performance, power, and statistics registers** for system monitoring and profiling

Power Management

- GPU operates in a dedicated MM_SS clock domain (single SoC voltage domain)
- Fine-grained clock gating for power efficiency
- Clock-gating controls exposed to firmware/OS

2.3.2. Video Decode (VPU)

Atlantis includes a dedicated **hardware video decode engine (VPU)** capable of offloading compressed video playback. This VPU is a single-core multi-format video decoder IP solution offering low power consumption and negligible load on the host CPU. The VPU supports multiple modern codecs at up to 8K30 and is intended for use in media playback, HMI, and streaming scenarios, freeing the Ascalon-X cluster and GPU for application logic and rendering. Reference Frame Compression (RFC) provides significant savings on system bus bandwidth.

Codec Support (Decode)

The VPU supports the following codec standards for decoding:

- **HEVC (H.265)**
  - Main Profile and Main10 Profile
  - Up to 10-bit depth
  - Picture sizes: 144 × 144 to 8192 × 8192 (step size: 8 pixels)
  - Maximum frame rate: **7680 × 4320 @ 30 fps** (8K30), single core
  - Bitrate: HEVC Level 5.1 High Tier
- **H.264/AVC**

    - Baseline, Main, High, and High10 Progressive Profiles
    - Picture sizes: 144 × 144 to 8192 × 8192 (step size: 16 pixels)
    - Maximum frame rate: **7680 × 4320 @ 30 fps** (8K30), single core
    - Bitrate: H.264 HP Level 5.1
- **H.264 MVC (Multi-view Video Coding)** support
- **H.264 SVC (Scalable Video Coding)** support
- **VP9**
    - Profile 0 (8-bit)
    - Profile 2 (10-bit depth, not 12-bit)
- **VP8**
- **AVS Part 2** (Jizhun)
- **AVS Part 16** (AVS+, Guangdian)
- **AVS2 Part 2** (GY/T 299.1)
    - Main Picture, Main, and Main10 bit profiles
- **VC-1**
    - Simple Profile (SP)
    - Main Profile (MP)
    - Advanced Profile (AP)
- **MPEG-4**
    - Simple Profile (SP)
    - Advanced Simple Profile (ASP)
- **MPEG-2 Main Profile** (MP)
- **H.263 Profile 0**
- **RealVideo 8, 9, and 10**
- **JPEG**
    - Baseline sequential
    - YUV420 and YUV422

Input Stream Formats

- Byte stream format
- NAL unit stream format
- Frame-by-frame decoding
- Packet-by-packet decoding
- 256 bytes by 256 bytes decoding

Output Picture Formats

- **YCbCr 4:2:0 semi-planar** 4×4 tiled
- **YCbCr400 monochrome** 4×4 tiled (H.264)
- **DEC400 compatible compressed formats**
    - Significantly reduces memory bandwidth
    - Compatible with VO and GPU for direct consumption

Performance and Resolution

- **Maximum picture size:** 8192 × 8192 pixels
- **Maximum frame rate:** 7680 × 4320 @ 30 fps (8K30) for HEVC and H.264, single core

Error Handling

- **Stream error detection**
- **Error concealment:** Continue decoding from next frame
- Interrupts to CPU for error conditions

Reference Frame Compression (RFC)

- Provides significant savings on system bus bandwidth
- Reduces DRAM traffic for reference frame storage and retrieval
- Transparent to software decoder stack

Data Path and Integration

- Compressed bitstreams are read from LPDDR via MM-NoC
- Decoded frames are written back to LPDDR in:
  - YCbCr 4:2:0 semi-planar 4×4 tiled format
  - DEC400 compressed formats (bandwidth optimized)
- **Output picture buffer stride:** Configurable and supported
- VO (Video Out controller) and GPU can consume decoded frames directly for composition and post-processing

Control and Firmware

- Control interface exposed through MM_SS register space
- Interrupts to CPU for:
  - Job completion
  - Buffer underflow/overflow
  - Error conditions
- Firmware/software stack manages:
  - Bitstream parsing
  - Job queueing and buffer management
  - Error recovery and fallback behavior
  - Multi-core scheduling (when applicable)

Power and Clocking

- VPU clocks sourced from MM_SS clock tree
- Clock gating and power-domain isolation allow disabling VPU when not in use to save power
- Low power consumption design for battery-powered applications

2.3.3. Display, Composition, and Output (VO / HDMI / MIPI-DSI)

Atlantis integrates a flexible **display pipeline** consisting of a Video Out controller (VO), HDMI 2.0 transmitter, and 4-lane MIPI-DSI transmitter. Together, they provide composition, scaling, color conversion, and output to a wide range of panels and display devices, enabling both evaluation board usage and embedded HMI products. The display subsystem supports simultaneous output to multiple displays and can handle resolutions up to 4K.

Video Out Controller (VO)

Display Plane Architecture

- **Up to 4 display planes** for flexible composition:
  - **Background layer** (base layer)
  - **Layer 1** (video layer)

    - **Layer 0, Layer 2, Layer 3** (OSD layers)
- **Per-layer capabilities**:
    - Configurable position (start pixel and start line)
    - Independent cropping
    - Alpha blending control

Supported Input Formats

- **Video Layers (Layer 0, 1, 2)**:
    - **2-plane YUV420** data format
    - **2-plane YUV422** data format
    - Configurable start pixel and start line positioning
- **OSD Layers (Layer 0, Layer 2, Layer 3)**:
    - **RGB 24-bit** with separate alpha channel
    - **RGB 16-bit (565)** with separate alpha channel
    - **ARGB 32-bit** (8 bits per channel)
    - **ARGB4444** (4 bits per channel)
    - **ARGB1555** (5 bits RGB, 1 bit alpha)
    - **Monochrome 8-bit** grayscale
    - Supports 1 or 2 plane ARGB formats
    - TFBC lossy and lossless decompression (Supported **only** on Layer 0)

Display Mixing and Blending

- **OSD layer blending**:
    - Configurable global alpha blending
    - Pixel-by-pixel alpha blending
- **Video layer blending**:
    - Configurable global alpha blending only

Color Processing

- **Configurable color space conversion**:
    - RGB ↔ YUV conversion
    - Support for BT.601 and BT.709 standards
    - RGB (4:4:4) to/from YCbCr (4:4:4 or 4:2:2)
- **Gamma correction** on RGB data before output

Resolution Support

- All common picture formats including:
    - **4K (3840 × 2160)**
    - **QXGA (2048 × 1536)**
    - **UXGA (1600 × 1200)**
    - **SXGA (1280 × 1024)**
    - **1080p (1920 × 1080)** progressive
    - **1080i (1920 × 1080)** interlaced
    - **720p (1280 × 720)**
    - **576i, 480i** (standard definition)

Interrupt Support

- **Up to 3 configurable interrupts**

- Support for configurable line position to generate interrupt
- Frame completion and error interrupts

Memory Interface

- **AXI interface** for high-bandwidth frame buffer access from DRAM
- **AHB interface** for register configuration
- Efficient DMA operation to minimize CPU involvement

HDMI 2.0 Transmitter

Video Standards and Formats

- **HDMI 2.0b compliant** output for external monitors/TVs
- **All CEA-861-E video formats**:
  - **720p / 1080i @ 120 Hz**
  - Up to **1080p @ 60 Hz**
  - Up to **4K @ 60 Hz** (3840 × 2160 @ 60 Hz)
- **HDMI 1.4b video formats**:
  - All CEA-861-E formats up to **1080p @ 120 Hz**
  - **HDMI 1.4b 4K × 2K** video formats

Input Video Formats

- **RGB 4:4:4** (Single Data Rate)
- **YCbCr 4:4:4** (Single Data Rate)
- **YCbCr 4:2:2** (Single Data Rate)
- **YCbCr 4:2:0** (Single Data Rate)

Color Space Conversion

- **Built-in CSC (Color Space Converter)**:
  - RGB (4:4:4) to/from YCbCr (4:4:4 or 4:2:2)
  - Configurable conversion matrices

Pixel Clock Range

- **13.5 MHz to 594 MHz** pixel clock support
- Covers all standard video timings from SD to 4K

Color Depth

- **Output up to 48-bit** color mode (16 bits per component)
- Full HD resolution support with deep color

MIPI-DSI Transmitter

Standards Compliance

- **MIPI Alliance Specification for Display Serial Interface (DSI) Version 1.2** (16 June 2014)
- **MIPI Alliance Specification for Display Command Set (DCS) Version 1.3** (5 October 2015)
- **MIPI Alliance Standard for Display Pixel Interface v2.00 (DPI-2)** (15 September 2005)
- **MIPI Alliance Specification for Stereoscopic Display Formats (SDF) v1.0** (22 November 2011)
- **MIPI Alliance Specification for D-PHY v1.2** (01 August 2014)
- **AMBA 2.0 (APB)** and **AMBA 4.0 (AXI)** compliant

Physical Interface

- **4-lane MIPI D-PHY** configuration
- **Lane speed:** Up to **2.5 Gbps per lane**
- **Total bandwidth:** Up to 10 Gbps aggregate (4 lanes × 2.5 Gbps)

Operating Modes

- **Video mode:** High-speed continuous video transmission
- **Command mode:** Packet-based command transmission
- **Low-power mode:** Commands sent in low-power during video mode
- **High-speed mode:** Commands sent in high-speed during video mode

D-PHY Features

- **Continuous clock mode** (clock lane always active)
- **Non-continuous clock mode** (clock gated during LP state)
- **Ultra Low-Power mode** with PLL disabled for power savings
- **Bidirectional communication** via data lane 0
- **Escape mode support** through data lane 0

Pixel Clock Range

- **5.6 MHz to 264 MHz** pixel clock
- Up to **4K UHD @ 30 Hz** (4-lane D-PHY, up to 2.5 Gbps/lane)

Command and Control

- Supports all commands defined in MIPI DCS specification
- Transmission of all Command mode packets through APB interface
- **PHY Protocol Interface (PPI)** for D-PHY communication

Reliability Features

- **ECC (Error Correction Code)** capabilities
- **Checksum** verification for data integrity

Display Pipeline Integration

Data Flow

- VO fetches rendered frames from GPU and/or decoded video from VPU via LPDDR
- Multi-layer composition and blending performed in VO
- Color space conversion and gamma correction applied
- Composite frame output to:
  - **HDMI PHY** for external displays, or
  - **MIPI-DSI PHY** for embedded panels, or
  - Both simultaneously (mirrored/clone modes, subject to bandwidth)

System Integration

- Configuration via **MM_SS register space**
- Accessible from:
  - **RCPU** for low-level board initialization and boot splash
  - **Ascalon-X cores** for OS display drivers and runtime control
- Interrupt routing to CPU for event handling (frame completion, errors, line triggers)

Timing Generation

- **Programmable display timings**:
  - Resolution, refresh rate, sync parameters
  - Horizontal and vertical blanking intervals
  - Sync polarities
- Integration with HDMI and DSI timing requirements
- Support for standard VESA and CEA timings

2.4. Memory and Storage Subsystem

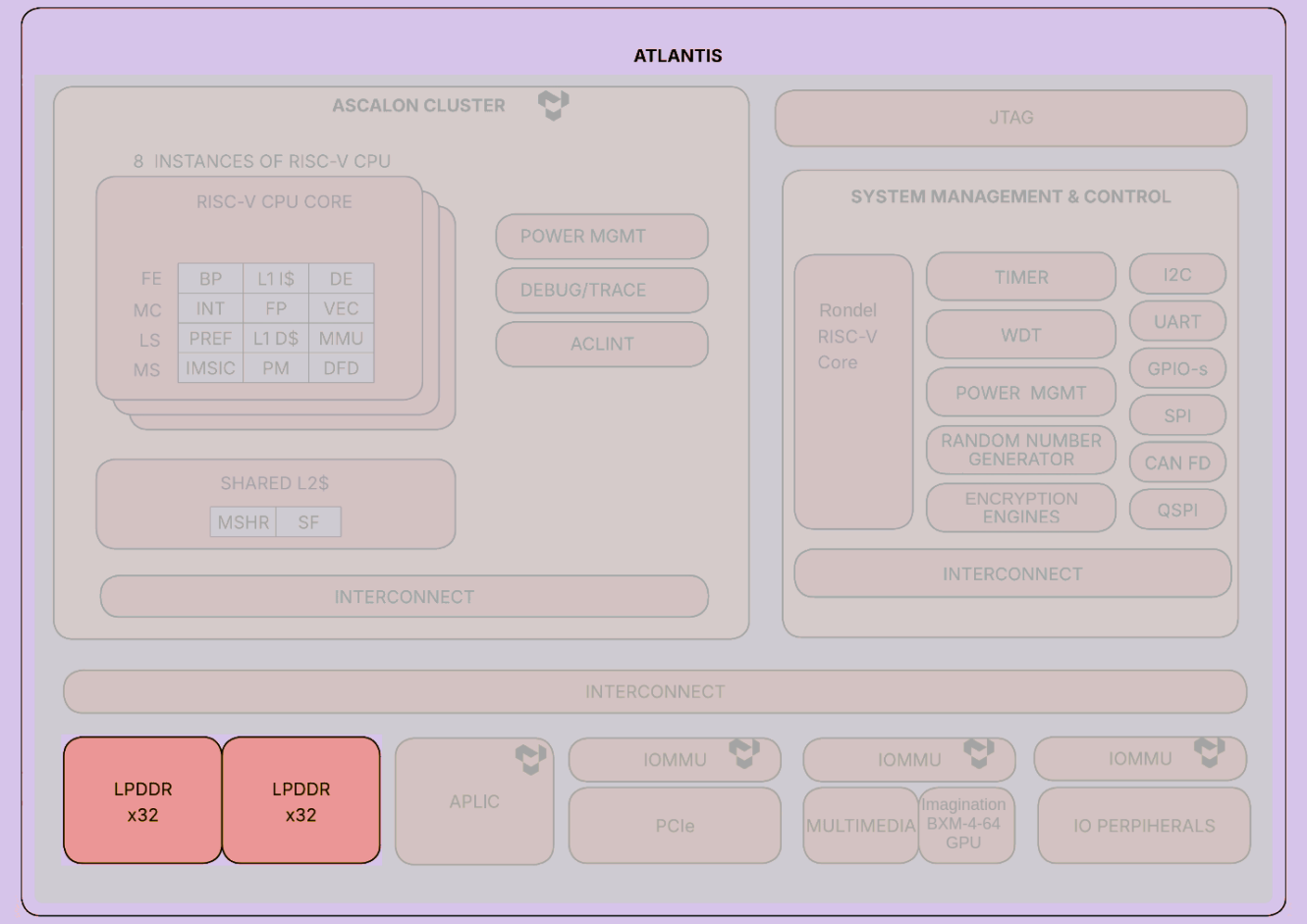

Figure 5. Atlantis Memory and Storage Subsystem Block Diagram

2.4.1. DRAM (LPDDR5/4x)

The Atlantis DRAM subsystem provides high-bandwidth, low-power memory using dual-channel LPDDR5/4x. It is composed of two 32-bit channels, each with a dedicated **LPDDR5/4/4X memory controller** and **LPDDR5/4x PHY**, connected to the SoC fabric via the Memory NoC (NOCM).

Memory Configuration

- **Supported types**: LPDDR5 up to 6400 MT/s, LPDDR4/4x up to 4266 MT/s
- **Channel organization**: 2 independent 32-bit channels (64-bit total), 1-2 ranks per channel
- **Capacity**: Up to 64 GB total
- **Controller frequency**: Up to 800 MHz, scalable 1:2/1:4 (LPDDR4) or 1:1:2/1:1:4 (LPDDR5) ratio

Controller Features

- DFI compliance, 16-64 read/write buffers (CAM), dynamic scheduling, out-of-order command execution
- **QoS**: High/variable/low priority read classes, normal/variable write classes, port throttling
- **Command latency**: 3-5 cycles typical

- **Features**: Write combine, ECC (inline SECDED, address/command parity), refresh management

Performance

- **LPDDR5-6400**: 25.6 GB/s per channel, **51.2 GB/s total**
- **LPDDR4x-4266**: 17.06 GB/s per channel, **34.12 GB/s total**

PHY Features

- DWC LPDDR5/4x PHY, DLL-based architecture, write/read leveling, ZQ calibration
- Training: Read DQS gate, read/write DQ/DM bit deskew, VREF training, CA training (LPDDR5)
- Power modes: Self-refresh, power-down, deep power-down, clock stop

2.4.2. On-Chip Memory Map

Atlantis uses a **unified 64-bit physical address space**. To balance simplicity for firmware and flexibility for SoC integration, the design distinguishes between:

- a **general memory map** (full 64-bit "system view"), and
- an **RCPU view** mapped via SMN and ATU (32-bit "management view").

System view (64-bit)

- DRAM regions:
  - Lower DRAM window mapped near the bottom of the physical address space (e.g., up to 2 GB for boot / OS)
  - Large extended DRAM region (tens of GB) above 4 GB for full capacity utilization
- Register and configuration regions:
  - Distinct windows for:
    - SOC_SS (PRCM, PVT, timers, watchdogs, LSIO)
    - CPU_SS (APLIC, ACLINT, CPU subsystem configuration)
    - LPDDR_SS (controllers and PHY utility)
    - PCIe_SS (controllers, ECAM and MMIO windows)
    - HSIO_SS (USB, GMAC, eMMC/SDIO, QSPI, etc.)
    - MM_SS (GPU, VPU, VO, HDMI, DSI)
    - SEC_SS (SPACC, TRNG, OTP, security control)
  - Each mapped as a distinct bus segment with well-defined base/size regions

RCPU view (32-bit)

- Memory map constrained to 0–4 GB, with:
  - ITCM, DTCM, scratchpad, ROM starting at low addresses
  - Fixed windows for SOC_SS, LSIO, SEC_SS, CPU_SS, and subsystem configuration
  - ATU-backed windows into DRAM and extended address ranges above 4 GB

Register conventions

- Many blocks share a common convention for:
  - Base address + offset per IP block
  - Region sizes that align with powers of two (e.g., 64 MB, 256 MB windows) for clean decoding
- Detailed base/size tables are provided in internal programming manuals and are referenced by OS/firmware bring-up code

2.4.3. Non-Volatile Storage and Boot Sources

Atlantis supports both **eMMC/SD** and **SPI NOR/NAND** as non-volatile storage and potential boot sources. These interfaces are grouped into HSIO and QSPI subsystems and are accessible to both RCPU and Ascalon-X. The SoC integrates **two SD/eMMC Host Controllers** with dedicated **SD/eMMC PHY macros**, providing high-performance, standards-compliant storage connectivity for embedded applications.

eMMC/SD/SDIO Host Controllers (×2)

Controller Features

- **Two instances** of SD/eMMC Host Controller + PHY
- **Compliance**: JEDEC eMMC 5.1, SD 6.0, SDIO 4.10, SDHCI specification
- **eMMC modes**: Legacy (26 MB/s), HS-SDR (52 MB/s), HS-DDR (104 MB/s), HS200 (200 MB/s), **HS400 (400 MB/s)**
- **SD modes**: Default/High Speed, UHS-I SDR12/25/50, **SDR104 (104 MB/s)**, DDR50
- **Bus width**: 1/4/8-bit configurable, voltage support 1.2V/1.8V/3.3V
- **Command Queuing Engine (CQE)**: Up to 32 queued commands, priority-based task reordering
- **DMA**: ADMA2/ADMA3 scatter-gather, 32-bit and 64-bit AXI master interface
- **PHY features**: DLL and delay lines for HS400/HS200/SDR104 tuning, hardware-driven tuning
- **Advanced features**: Optional in-line encryption, power/clock gating, CRC validation
- **Performance**: 400 MB/s per controller (HS400), 104 MB/s (SDR104), 808 MB/s total (both active)

Typical Use Cases

- **eMMC**: Primary boot storage, OS and application storage, user data storage
- **SD card**: Removable storage, data logging, firmware updates
- **SDIO**: Wi-Fi, Bluetooth, GPS modules (single controller configured as SDIO host)

Quad-SPI (QSPI) Controller

Controller Features

- **Data modes**: Single/Dual/Quad-SPI (1/2/4-bit data width)
- **Clock frequency**: Up to 102 MHz (13.25 MHz minimum)
- **Flash support**: SPI NOR (1–256 MB) and SPI NAND (128 MB–1 GB), 1.8V/3.3V signaling
- **Operating modes**:
  - **XIP (Execute-In-Place)**: Memory-mapped access, CPU directly fetches instructions/data from flash
  - **Page mode**: Software-controlled read/write/erase via APB registers
- **SPI NOR commands**: Standard READ, Fast READ, Quad I/O Fast READ, PAGE PROGRAM, SECTOR/CHIP ERASE
- **SPI NAND features**: On-die ECC (1/4-bit per 512 bytes), software bad block management, 2/4 KB pages
- **Boot support**: Primary boot source, XIP or copy to ITCM, secure boot verification
- **Performance**: ~40–45 MB/s read (XIP, Quad, 102 MHz), 200–500 KB/s write (device-dependent)

Boot Sources and Boot Flow

Boot Sources

- **Supported sources**: QSPI NOR/NAND, eMMC, SD card, USB DFU, UART download
- **Selection**: Boot strap pins (2–4 GPIO) or OTP/eFuse configuration, with fallback sequence
- **RCPU Boot ROM** executes first, selects source, loads/verifies FSBL, hands off to bootloader

Boot Flow

1. Power-on reset → RCPU executes from Boot ROM
2. Sample boot strap pins, initialize clocks/PLLs, initialize boot source interface
3. Load bootloader from source into ITCM/DRAM
4. Secure boot verification (optional): Verify signature (RSA-2048/ECDSA-P256) using OTP public key hash
5. Jump to bootloader entry point

Boot Performance

- **QSPI NOR (XIP)**: ~10–50 ms (zero-copy)
- **eMMC (HS400)**: ~50–100 ms (1 MB @ 400 MB/s, ~2.5 ms copy + init/verify)
- **SD card (SDR104)**: ~100–200 ms (1 MB @ 104 MB/s, ~10 ms copy + slower init)

2.5. High-Speed I/O and Networking (PCIe_SS, HSIO_SS)

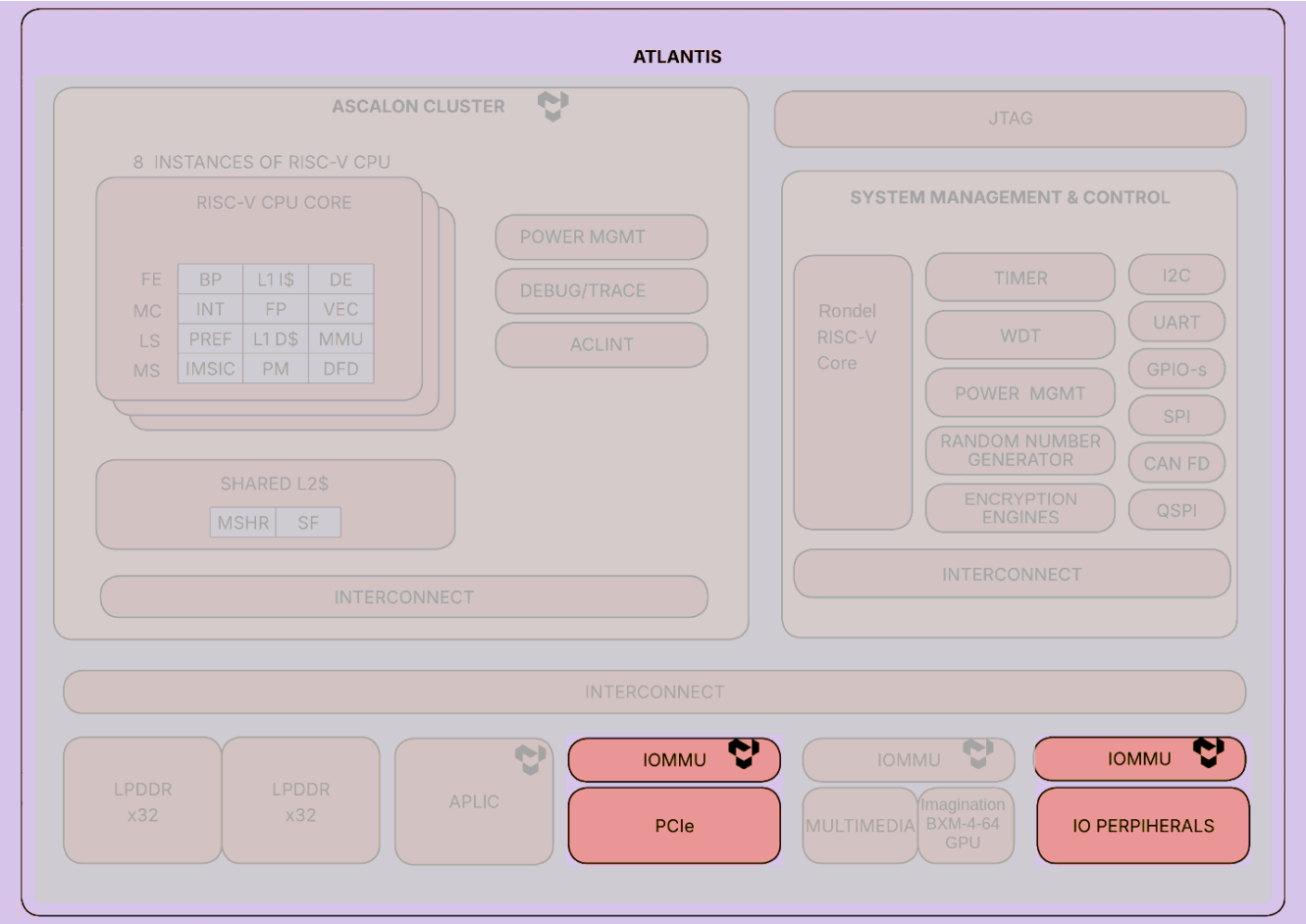

Figure 6. Atlantis Networking Block Diagram

2.5.1. PCI Express (PCIe_SS)

Atlantis integrates a comprehensive **PCIe Gen4 subsystem** providing up to **20 lanes** over three **PCIe controllers**. The controllers comply with **PCI Express Base Specification Revision 4.0** and support data rates from Gen1 (2.5 GT/s) through Gen4 (16 GT/s).

Controllers

- **3 PCIe controllers**: total lanes (configurable as 1×16 + 1×4 or 2×8 + 1×4) [controllers sum to 28 lanes]
  - **Controller 0**: 1×16 or 1×8 or 1×4 (16 lanes)
  - **Controller 1**: 1×8 or 1x4 or 1x2 (8 lanes)

    - **Controller 2**: 1×4 or 1x2 or 1x1 (4 lanes)
- **Modes**: Root Complex (RC), Endpoint (EP), configurable per controller
- **Bifurcation**: Lane assignment configurable via strapping pins or firmware

Performance

- **Gen4 (16 GT/s)**: ~1.97 GB/s per lane (128b/130b encoding)
- **Total bandwidth**: Up to 39.4 GB/s (20 lanes × 1.97 GB/s)
- **Latencies**: TLP processing <100 ns, completion <500 ns typical

Advanced Features

- ACS (Access Control Services), ATS (Address Translation Services), PRI (Page Request Interface)
- SR-IOV (Single Root I/O Virtualization), VF up to 256 per PF
- ASPM L0s/L1/L1.1/L1.2, clock gating, PHY power states
- MSI/MSI-X interrupts, AER (Advanced Error Reporting), ECRC

2.5.2. USB (HSIO_SS)

Atlantis integrates comprehensive USB connectivity with **USB 3.1 Gen2** (SuperSpeed+, 10 Gbps) and **USB 2.0** (High-Speed, 480 Mbps) support. The subsystem provides **four USB controllers — 2 × USB 3.1 Gen2 and 2 × USB 2.0** — each supporting host, device, and On-The-Go (OTG) operation.

Controllers

- **2 USB 3.1 Gen2 controllers**:
    - USB 3.1 Gen2 (10 Gbps), Gen1 (5 Gbps), backward compatible with USB 2.0 HS/FS/LS
    - Modes: Host, Device, OTG (On-The-Go) with HNP/SRP
    - Dual-role capable (host <→ device switching)
- **2 × USB 2.0 controllers**:
    - High-Speed (480 Mbps), Full-Speed, Low-Speed
    - Modes: Host, Device, OTG (On-The-Go) with HNP/SRP
    - Dual-role capable (host <→ device switching)

Performance

- **USB 3.1 Gen2**: 10 Gbps (1.25 GB/s theoretical), ~1 GB/s effective
- **USB 2.0 HS**: 480 Mbps (60 MB/s theoretical), ~40 MB/s effective

Advanced Features

- xHCI (host mode), scatter-gather DMA, USB 3.1 hubs
- Power management: U0/U1/U2/U3 link states, LPM (Link Power Management)
- OTG: HNP (Host Negotiation Protocol), SRP (Session Request Protocol), VBUS control

PHY

- 2x USB 3.1 PHY (SuperSpeed+) + 2x USB 2.0 PHY (High-Speed)
- Supports 8b10b encoding (USB 3.x), LFPS, equalization, spread-spectrum clocking

2.5.3. Ethernet (GMAC)

Atlantis integrates a **Gigabit Ethernet MAC (GMAC)** providing standards-compliant 10/100/1000 Mbps Ethernet connectivity via RGMII interface.

MAC Controller

- **Standards**: IEEE 802.3-2018 (Ethernet), IEEE 802.3az (EEE)
- **Speeds**: 10 Mbps, 100 Mbps, 1000 Mbps (Gigabit)
- **Modes**: Full-duplex and half-duplex, auto-negotiation
- **Interface**: RGMII (Reduced Gigabit Media Independent Interface) to external PHY

Performance and Features

- **Throughput**: Up to 1 Gbps (125 MB/s) Gigabit Ethernet
- **DMA**: Multi-channel scatter-gather DMA, programmable burst lengths, separate TX/RX DMA engines
- **Offloads**: Checksum offload (IPv4/IPv6, TCP/UDP/ICMP), TCP segmentation, VLAN tagging
- **QoS**: IEEE 802.1Qav (credit-based shaper), 802.1Qbv (time-aware shaper), weighted round-robin
- **Power**: IEEE 802.3az EEE (Energy Efficient Ethernet), low-power idle

Timestamping and PTP

- IEEE 1588-2008 PTPv2 (Precision Time Protocol)
- Hardware timestamping, <8 ns accuracy typical

2.6. Security and Manageability

2.6.1. Crypto Engine (SPACC)

The Atlantis security subsystem integrates a **Security Protocol Accelerator and Crypto Co-processor (SPACC)** providing hardware-accelerated cryptographic operations for symmetric encryption, asymmetric encryption, hashing, and authentication. The crypto engine is designed to offload computationally intensive security operations from the CPU, enabling secure boot, secure storage, network protocol offload (IPsec, TLS/SSL), key management, and trusted execution services.

The SPACC is accessible to both RCPU (for secure boot and system management) and Ascalon-X cores (for application-level cryptography) via memory-mapped registers and DMA-based data transfer.

Algorithms and Features

- **Symmetric encryption**: AES-128/256 (CBC, CTR, GCM, CCM), DES/3DES (legacy)
- **Hash**: SHA-256/384, HMAC
- **Asymmetric crypto**: RSA-1024/2048/3072/4096 (signature/encryption), ECDSA/ECDH (P-256/384/521), SM2 (optional)
- **Key management**: HKDF, PBKDF2, AES key wrapping, OTP key storage integration
- **Architecture**: 64-bit AXI master (DMA scatter-gather), APB/AHB register interface, OTP key ladder
- **Performance @ 800 MHz**: AES 1–2 Gbps, AES-GCM 800 Mbps–1.5 Gbps, SHA-256 500 MB/s–1 GB/s, RSA-2048 verify ~20–50 ms, ECDSA-P256 verify ~5–20 ms
- **Use cases**: Secure boot (RSA/ECDSA verification), inline storage encryption (AES-XTS), IPsec/TLS offload
- **Software support**: Linux Crypto API, OpenSSL/BoringSSL integration, bare-metal MMIO

2.6.2. True Random Number Generator (TRNG) and OTP/eFuse

Atlantis integrates a **True Random Number Generator (TRNG)** and a **16 KB One-Time Programmable (OTP) memory array** (also known as eFuse) to provide hardware root of trust, secure random number generation, secure boot and lifecycle management, and device authentication capabilities.

The TRNG and OTP are critical security primitives accessible to RCPU (for boot and system management) and Ascalon-X cores (for application-level security) with appropriate access controls.

TRNG Features

- **Entropy source**: Hardware physical noise (ring oscillator, metastability, or analog noise)
- **Health monitoring**: Continuous tests (RCT, APT per NIST SP 800-90B, AIS-31), post-processing with SHA-256/AES-CTR whitening
- **Output**: 32-bit or 256-bit random words, 10–100 Kbps raw entropy, 1–10 Mbps post-processed
- **Use cases**: Key generation (AES, RSA, ECDSA), IV/nonce generation, ASLR seeds, DRBG seeding
- **Software interface**: MMIO registers, Linux /dev/hwrng
- **Compliance**: NIST SP 800-90B, SP 800-22, AIS-31 PTG.2/PTG.3, FIPS 140-2/140-3

OTP/eFuse Features

- **Capacity**: 16 KB (131,072 bits), banked organization with access control
- **Typical content**:
  - Device ID (UDID 128/256-bit), manufacturing data, calibration values
  - Secure boot: Public key hash (SHA-256), boot mode locks, rollback protection counter
  - Lifecycle: JTAG/trace disable, secure debug enable, lifecycle state (dev/production/RMA/EOL)
  - Crypto keys: AES-256/SM4 master keys, key encryption keys (KEKs)
- **Programming**: One-time electrical fusing (irreversible), MMIO interface with bank locking
- **Security**: Automatic load at reset, shadow registers, optional ECC, redundant storage
- **Access control**: RCPU full access, Ascalon-X privileged read, secure/non-secure partitioning
- **Software support**: RCPU firmware (provisioning), Linux NVMEM framework, TEE APIs

2.6.3. IOMMUs and Firewalls

To enable **virtualization**, **strong compartmentalization**, and **secure multi-tenant operation**, Atlantis deploys multiple **I/O Memory Management Units (IOMMUs)** and **address-range firewalls** at the edges of major NoC interconnects and subsystems. These security mechanisms provide address translation for I/O masters, access control and filtering, protection against malicious or buggy I/O devices, and support for hypervisors and virtual machines.

IOMMUs and firewalls are configured by privileged software (hypervisor, secure OS, or RCPU firmware) and enforced in hardware, providing defense-in-depth against software and hardware vulnerabilities.

IOMMU and Firewall Features

IOMMUs

- **Single IOMMU** for the entire PCIe subsystem (not one per controller)
  - **PCIe Gen4 x20** total lanes (configurable as 1×16 + 1×4 or 2×8 + 1×4)
- **Functions**: IOVA→PA address translation (multi-level page tables, 4KB/2MB/1GB pages), per-device/per-domain R/W/X permissions, fault isolation and logging
- **Device identification**: Stream ID (SID) / Device ID (PCIe BDF, USB device address, GPU context ID)
- **TLB**: **128 entries**, global/per-domain/per-page invalidation
- **Faults**: Translation fault, permission fault, address size fault, external abort
- **Use cases**: VM device passthrough (SR-IOV), DMA attack prevention, GPU virtualization (vGPU)

Firewalls

- **Instances**: SMN Firewall (System Management NoC), NOCM Firewall (main memory interconnect)
- **Configuration**: 16 programmable address ranges per firewall, R/W permissions per master ID
- **Enforcement**: Secure/Non-Secure partitioning, blocking or logging violations
- **Master IDs**: Ascalon-X cluster, RCPU, GPU, USB, PCIe, Ethernet, eMMC/SD
- **Use cases**: RCPU-only regions, secure boot memory protection, peripheral access control

Software Support

- **Linux IOMMU API**: iommu_map/unmap/attach_device, VFIO for userspace passthrough
- **RCPU firmware**: Configures and locks firewalls during boot
- **Debug**: Fault logging registers (Stream ID, IOVA, address, fault type), interrupt-driven handling

Performance Impact

- IOMMU translation adds ~50-200 ns latency per DMA transaction (TLB hit: ~50 ns, TLB miss: ~200 ns)
- Negligible throughput impact for large transfers (DMA burst amortizes translation cost)

2.6.4. Debug and Design-for-Debug (DFD)

Atlantis provides comprehensive debug capabilities for development, bring-up, and field diagnostics. Debug features include JTAG access for Ascalon-X and RCPU cores, system-level debug infrastructure, instruction trace, and secure debug authentication for production devices.

Debug Features

JTAG Debug

- **Ascalon-X cluster**: RISC-V Debug v1.0 compliant, 8-core JTAG daisy-chain, halt/resume/single-step, register/memory access, hardware breakpoints/watchpoints
- **RCPU**: RISC-V Debug v0.13.2 compliant, JTAG interfaces, progbuf and SBA modes
- **System boundary scan**: IEEE 1149.1 JTAG for board-level interconnect test

Secure Debug

- **Authentication**: OTP-controlled (JTAG_DISABLE, SECURE_DEBUG_ENABLE fuses)
- **Challenge-response**: ECDSA/RSA-based authentication for production debug access
- **Lifecycle states**: Development (full access), Production (auth required), RMA (limited), EOL (disabled)

Trace

- **N-Trace**: RISC-V N-Trace for Ascalon-X and RCPU, instruction/branch trace, optional data trace, timestamp correlation
- **Trace buffer**: On-chip circular buffer (size **to be determined**) or external trace port (TPIU)
- **Trace compression**: Reduces bandwidth, typical 10:1 compression ratio

Design-for-Debug (DFD)

- **Clock/reset observation**: Dedicated pins for clock/reset monitoring
- **Power state indicators**: GPIO outputs for power domain status
- **Error injection**: Test modes for fault injection (ECC, parity, protocol errors)

2.7. Power, Clocks and Reset

2.7.1. Process Technology and Power Domains

Atlantis is fabricated in **12FFC** (12 nm FinFET Compact) and implements a **single core voltage (power) domain** with **multiple independent clock domains** (one PLL per subsystem) plus an **always-on (AON)** domain. Power is managed through clock gating of idle subsystems and dynamic frequency scaling (DFS) on the Ascalon-X CPU cluster, under RCPU/CPL firmware control.

Process Technology

- **Process:** 12FFC (12 nm FinFET Compact), 1P11M metal stack
- **Core voltage:** VDD 0.8 V nominal (−7%/+10%)

- **I/O voltages: 1.8 V** and **3.3 V** LVCMOS; LPDDR5/4x **VDDQ 0.5 V / 0.3 V**; 1.2 V eMMC PHY option
- **Transistor type:** FinFET for improved power efficiency and performance scalability

Clock and Power Domains

- **Single core power (voltage) domain (VDD)** plus an **always-on (AON)** domain (RTC, PVT sensors, wake logic, crystal, Boot ROM); separate I/O and LPDDR supply rails
- **Independent clock domains** (one PLL per subsystem), each at a fixed configured operating frequency:

Clock domain (PLL)	Max frequency
Ascalon-X CPU cluster (CPU PLL)	1.38 GHz
RCPU (RCPU PLL)	600 MHz
DDR controller (DDR PLL)	800 MHz
NoC fabric (NOC PLL)	800 MHz
Multimedia (MM PLL0 / PLL1)	—
High-speed peripherals (HS Peri PLL)	600 MHz

Power Management

Clock Gating

- Idle subsystems are **clock-gated by default**; RCPU firmware releases clock and reset gates (`BUS_CG` / `XX_BLK_CG` / `XX_BLK_RST` registers) to activate a subsystem and its IPs
- Fine-grained clock gating within active subsystems for dynamic-power reduction

DFS (Dynamic Frequency Scaling)

- CPU-cluster DFS: Frequency scaled over **0.5–1.38 GHz** based on workload. Nominal State (**1.38 GHz**) and Low Power State (**500 MHz**)
- Orchestrated by the in-cluster Cluster Power Management Logic (CPL) firmware under RCPU supervision; the OS requests performance levels through the management-firmware interface (it does not directly program the PLL or voltage rail)

CPU Low-Power States

- Intra-cluster core/cluster low-power states (C2 / C3 / C4) as described in Section 2.1.9.3.

2.7.2. Clocking

Atlantis uses on-chip oscillators and a PLL hierarchy to generate clocks for CPU, NoCs, DRAM, HSIO, MM, and RCPU subsystems. Clocks are distributed via well-defined domains with clock gating to reduce dynamic power.

- **Reference sources**: 24 MHz on-chip oscillator, 100MHz reference clock from external clock pin
- **PLLs**: Dedicated PLLs for NoC/CPU, DRAM, HSIO, MM, and auxiliary domains
- **Clock frequencies**: NoC clocks up to ~800 MHz, derived clocks for RCPU/SMN, Ascalon-X, DRAM controllers, PCIe PHY
- **Clock domains**: CPU_SS/NOCC, RCPU/SMN, NOCM/LPDDR, HSIO (PCIe/USB/GMAC/eMMC), MM (GPU/VPU/VO/HDMI/DSI)
- **Power saving**: Fine-grained clock gating at subsystem and IP level, firmware/OS controlled gating registers

2.7.3. Power Management and Thermal

Atlantis provides SoC-level power management with a Thermal Design Power (TDP) target **to be determined** and industrial-class temperature range (-40°C to +125°C junction). The PVT controller, RCPU firmware, and OS policies cooperate to maintain thermal and power boundaries.

> *TDP will be quoted in a future revision once power characterization is complete.*

PVT Monitoring

- Multiple on-chip thermal sensors in high-activity regions
- Supply monitors for core rails
- Process monitors for wafer and part characterization

Thermal Specifications (Draft)

- **TDP**: **To be determined**. TDP will be quoted in a future revision once power characterization is complete.
- **Junction temperature**: -40°C to +125°C
- **Ambient temperature**: 0°C to +85°C (depending on system cooling)

Power Sequencing

- Carefully ordered initialization: Rails/domains → PLLs/clocks → NoCs/DRAM controllers → CPU clusters/subsystems
- RCPU controls power-on sequence and domain release ordering

2.7.4. Reset Architecture

Atlantis uses a hierarchical reset scheme covering power-on reset, external reset, and software/firmware-initiated resets at global and subsystem levels.

Global Resets

- **Power-on reset (POR)**: Asserted on initial power ramp or core rail instability
- **External reset pin**: Board-level reset control
- **Global software reset**: Triggered via control registers

Local/Subsystem Resets

- Per-subsystem resets (PCIe, HSIO, MM, LSIO)
- Watchdog-initiated resets (CPU/RCPU or full device)
- Reset isolation and synchronization per clock/power domain

Reset Release Ordering

1. Apply power and de-assert the Power-On Reset (POR).
2. Wait for the oscillators to lock.
3. Boot the RCPU using the oscillator clock.
4. Configure the subsystem PLLs and switch the RCPU clock source to the PLL clock.
5. Release the NOCC reset.
6. De-assert resets for peripherals related to boot storage (e.g., UART/SDIO).
7. Release the DDR subsystem reset.
8. Release resets for remaining IP blocks.
9. Execute DDR training.
10. Release the Ascalon-X core reset.

2.8. Package and Environmental

2.8.1. Package

Atlantis uses an FCBGA package sized to carry the high pin-count required for LPDDR, PCIe, HSIO, and the core, I/O, and memory supply rails, while remaining practical for evaluation boards and customer designs. The package design balances thermal

performance, signal integrity, and manufacturability.

Package Type and Dimensions

- Package type: **FCBGA (Flip-Chip Ball Grid Array)**
- Package dimensions: **35mm × 35mm (nominal)**
- Process technology: **12FFC (12nm FinFET Compact)**
- Substrate: **8-layer**
- Package height: **1.0-1.4 mm (typical)** — Standard FCBGA profile

Ball Grid and Pin Count

- Total ball count: **~1000-1200 (estimated)**
  - Signal pins: ~590
  - Power and ground: ~400-600 (distributed across the core, I/O, and memory supply rails)
- Ball pitch: **To be determined**
- Ball diameter: **To be determined**
- Ball grid arrangement: **To be determined**

Package Materials

- Solder balls: Lead-free (SAC305: Sn96.5/Ag3.0/Cu0.5)
- Die attach: Flip-chip underfill
- Mold compound: Standard epoxy

Signal Routing

- High-speed interfaces (LPDDR, PCIe, USB, Ethernet) strategically placed to:
  - Minimize crosstalk and routing complexity
  - Optimize signal integrity
  - Reduce electromagnetic interference (EMI)
- Adequate ground and power pin distribution for:
  - Signal return paths
  - Power delivery integrity
  - Thermal dissipation

Die Floor Plan

Floor plan was selected for optimal package integration:

- PCIe subsystem: Bottom region
- LPDDR subsystem: Top region
- Ascalon-X CPU cluster: Central (8 cores + 12MB L2 cache)
- Uniform bump distribution for substrate routing feasibility

Package Outline Drawing (POD)

Status: **To be determined** — Detailed mechanical drawings pending from packaging team. Final POD will include:

- Precise package dimensions and tolerances
- Ball position map with coordinates
- Package marking layout
- Recommended PCB land pattern

- Keep-out zones and clearances

2.8.2. Operating Conditions and Environmental

Temperature Specifications

- Junction temperature (Tj): **−40 °C to +125 °C** (operating range)
- Ambient temperature (Ta): **0 °C to +85 °C** (with adequate cooling)

Thermal Design

- TDP (Thermal Design Power): **To be determined**.
- Cooling solution (**heatsink/airflow**) and recommended TIM conductivity: **To be determined**, to be specified once TDP is characterized.

> *TDP will be quoted in a future revision once power characterization is complete.*

Environmental and Compliance

- Moisture Sensitivity Level (MSL): **To be determined**
- RoHS compliant (lead-free package)
- REACH compliant

Chapter 3. System Architecture Summary

3.1. NoC and Subsystem Topology

Atlantis implements multiple **Networks-on-Chip (NoCs)** to decouple system-management traffic from high-bandwidth data traffic and to localize quality-of-service and security mechanisms. The main networks are:

- **SMN** (System Management NoC)
- **NOCC** (Core NoC)
- **NOCM** (Memory NoC)

Additional local NoCs are present in PCIe_SS, HSIO_SS, and MM_SS.

3.1.1. SMN (System Management NoC)

- 64-bit control-plane network for:
  - RCPU and Ascalon-X CSR accesses
  - Access to SOC_SS, SEC_SS, LSIO, subsystem configuration blocks
- Provides:
  - **QoS generators** for prioritized configuration writes/reads
  - Firewalls on master ports to restrict configuration from untrusted initiators

3.1.2. NOCC (Core NoC)

- High-bandwidth, low-latency interconnect for:
  - Ascalon-X CPU cluster (via MMCAB)
  - High-bandwidth DMA engines (PCIe, HSIO, MM)
  - Other initiators that need heavy DRAM bandwidth
- 512-bit datapath to minimize cycles per cache line transfer
- Includes:
  - Internal QoS configuration (e.g., per-master priorities)
  - Firewalls and IOMMU interfaces

3.1.3. NOCM (Memory NoC)

- Sits between NOCC/other master ports and LPDDR controllers
- Responsible for:
  - Mapping and routing to DRAM channels
  - Applying QoS controls at the DRAM level (e.g., via AXQOS signals into DDR controllers)
  - Enforcing DRAM protection policies as configured via firewalls/IOMMUs

3.1.4. Subsystem NoCs

- **PCIe-NoC**, **HSIO-NoC**, **MM-NoC**:
  - Local interconnects internal to their subsystems
  - Provide address decoding, QoS, and group-level firewalling
  - Aggregate DMA and register accesses before entering NOCC/NOCM