arm

SOC Design and Architecture

Prasanth Pulla & Pratik Bhattacherjee

Jul-2023

Who are we- Bios

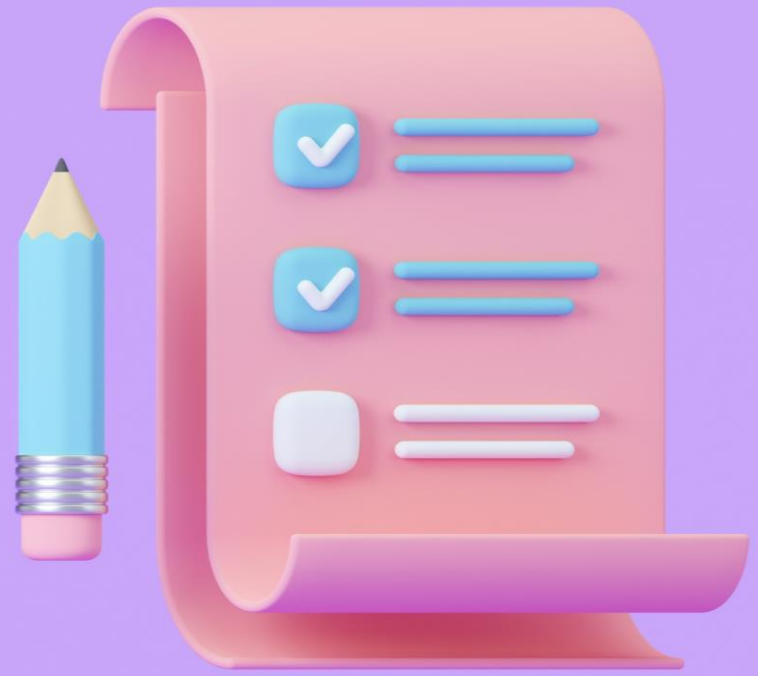
- Pratik Bhattacherjee a Principal Engineer in Arm. I have over 12 years of experience in Processor/CPU verification, my expertise lies in working with diverse Application (A-Class) and real-time (R-Class) processors. My background includes extensive work in CPU Architecture Compliance and System Compliance, where I have worked with various Arm IPs and third-party IPs.

- Prasanth Pulla is a Senior Principal CPU Validation Architect in the Architecture and Technology group at Arm. I have been part of the Silicon and Firmware industry for 20 years. I am the author of the Arm BSA (Base System Architecture) specification. I have contributed to multiple Arm System compliance projects which are open source and available on Github.

Agenda

Goal of this session: To understand the basic concepts in the hardware, and SoC design, enabling engineers to get started with optimized and efficient System development.

- Understand IPs involved in SoC Design.
- Design basics
  - Walk-through different blocks in a SoC
- Case Study: Security Camera Subsystem

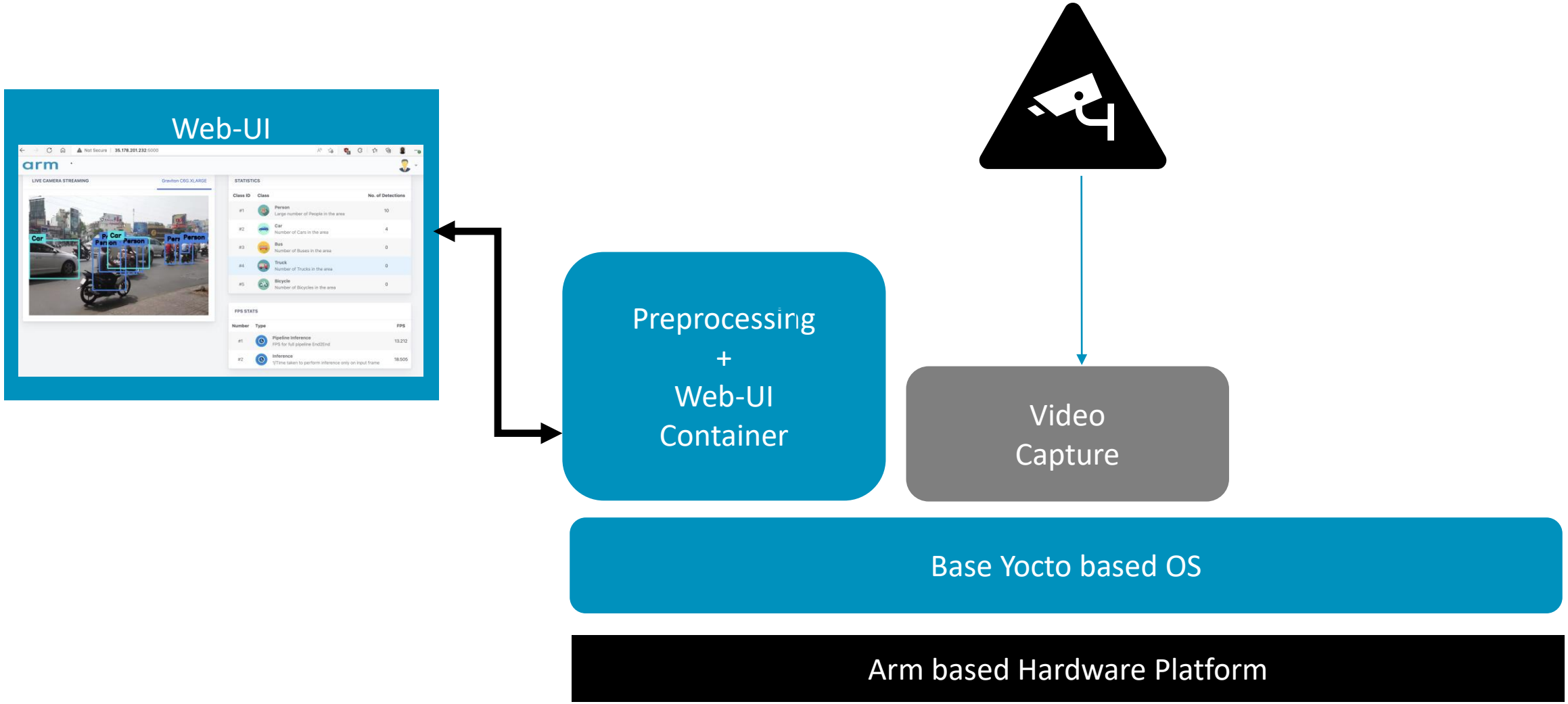


SOC Design Considerations

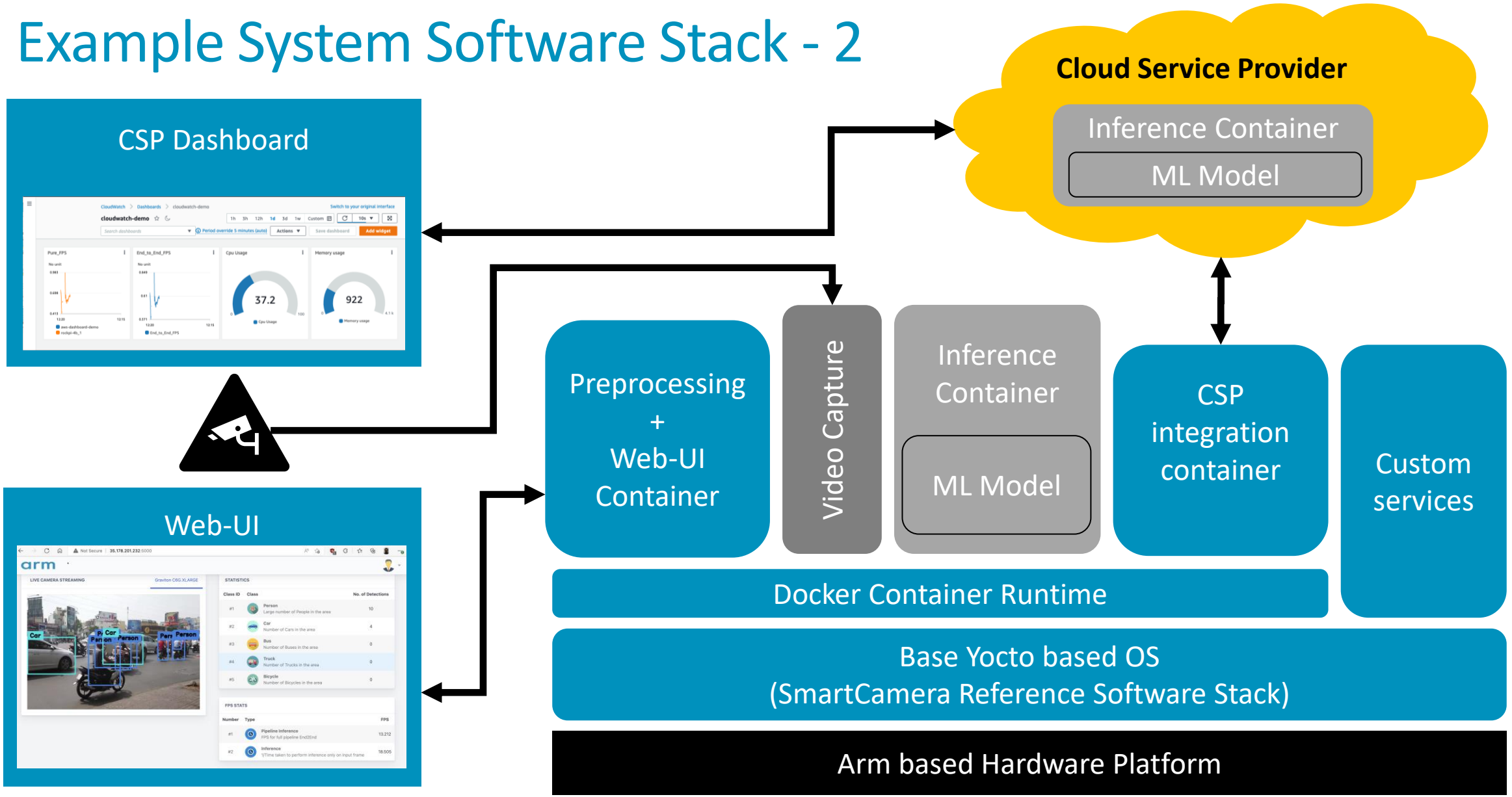
- Functionality

- Power
- Performance
- Security
- Reliability
- Maintainability
- Cost

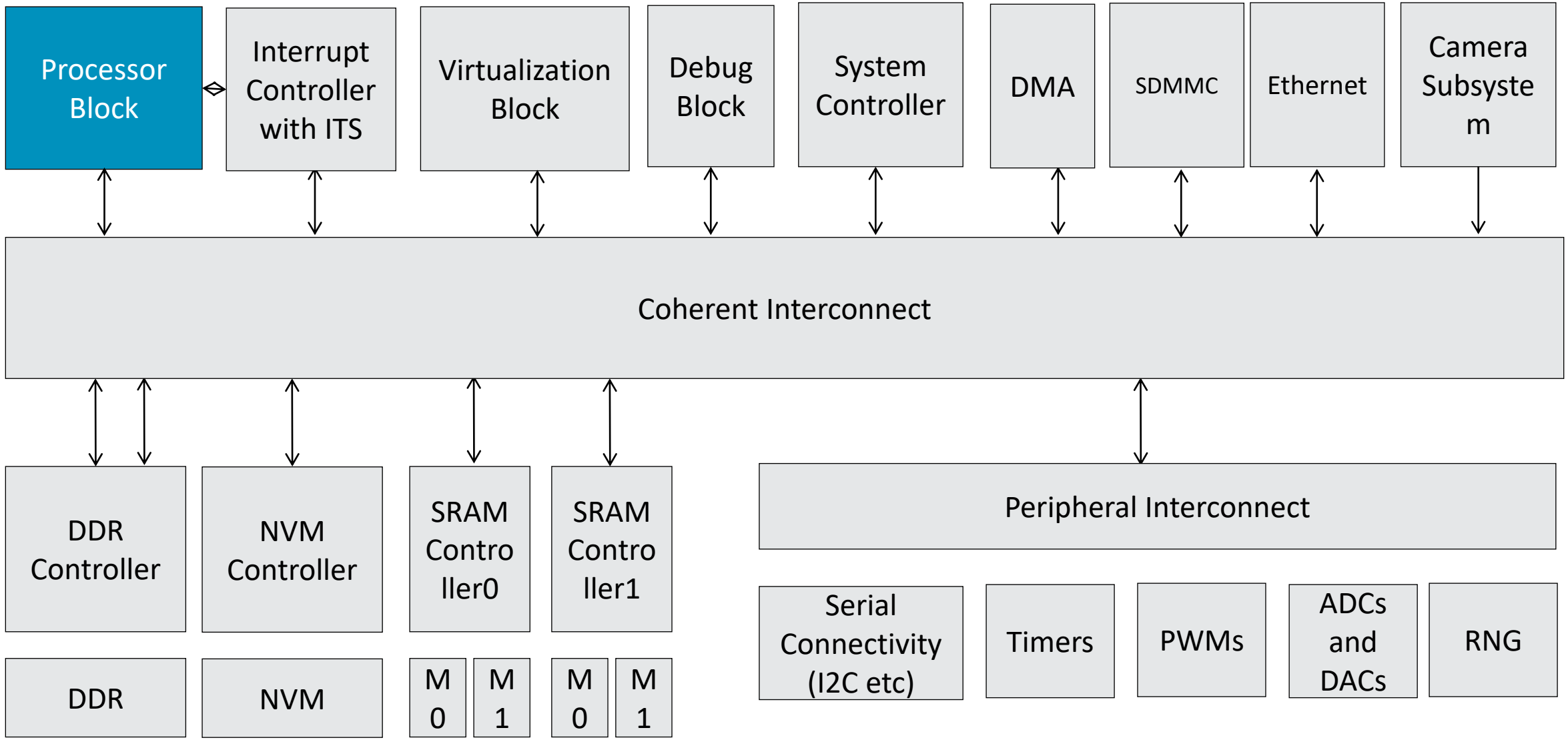
Example System Software Stack - 1



Example System Software Stack - 2



SOC - Block Diagram



Arm processor families

- Cortex-A series (application)
  - High performance processors capable of full OS support
  - Applications include smartphones, digital TVs, and smart books.
- Cortex-R series (real-time)
  - High performance and reliability for real-time applications
  - Applications include automotive braking systems and powertrains.
- Cortex-M series (microcontroller)
  - Cost-sensitive solutions for deterministic microcontroller applications
  - Applications include microcontrollers and smart sensors.

Processor Choice Based on Application

Application : Video Doorbell
Requirement: Low-Medium resolution, Web Interface

Processor Choice: M Class

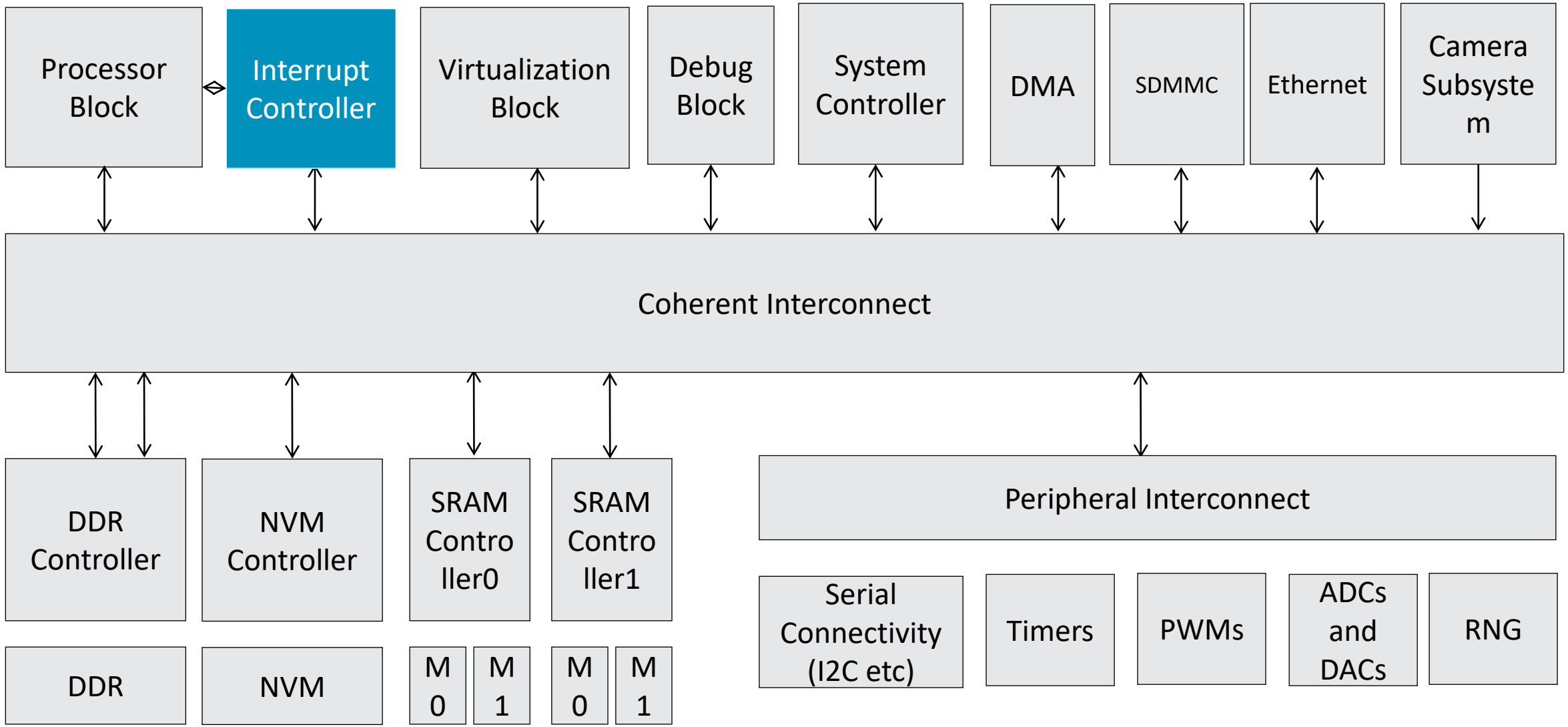
Application : Automotive Breaking System
Requirement: Medium-High resolution, Deterministic & quick response time.

Processor Choice: R Class

Application : AI workload with Image processing
Requirement: High resolution, Web Interface, OS boot support etc.

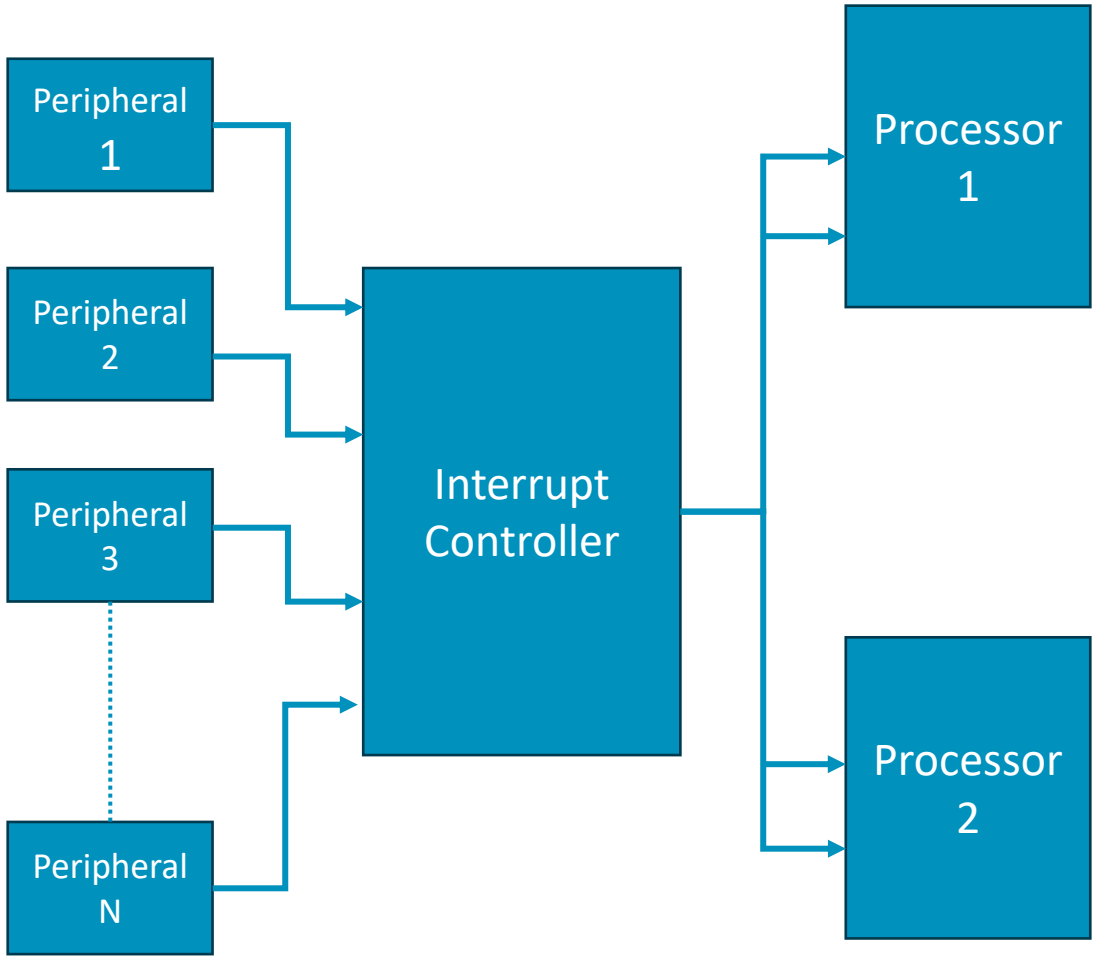
Processor Choice: A Class (+) Optionally R class as well.

SOC - Block Diagram



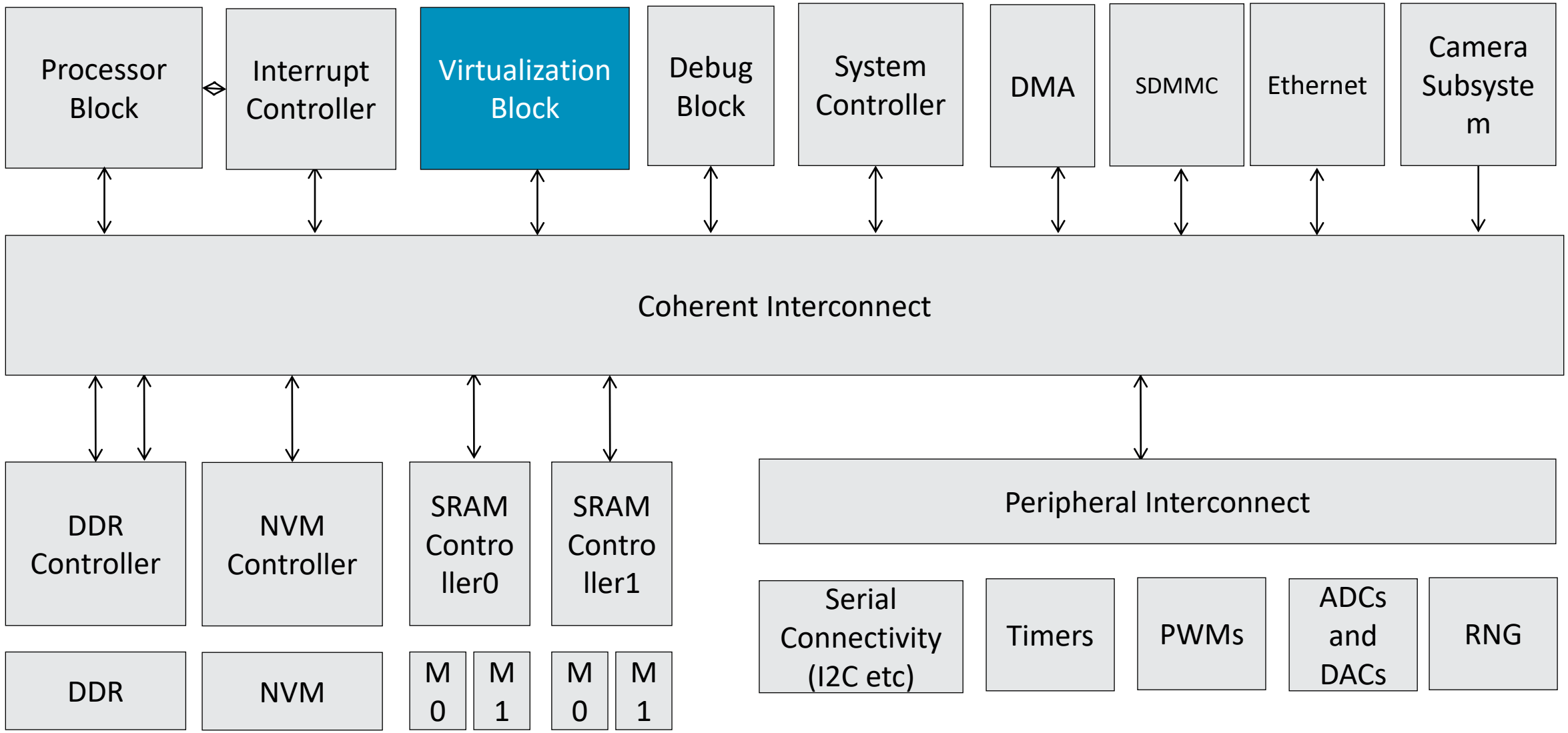
Interrupt Controller

- In a laptop/Mobile like system there are several Peripherals.
- Each peripheral(Keyboard, NIC) may generate thousands of interrupts.
- These interrupts must be distributed in limited number of processor (e.g., 16).
- Processor has limited number of pins to accept interrupts.
- Interrupt controller helps to distribute interrupts between multiple processors, and it also provide mechanism to report thousands of interrupts using limited number of processor input pins.

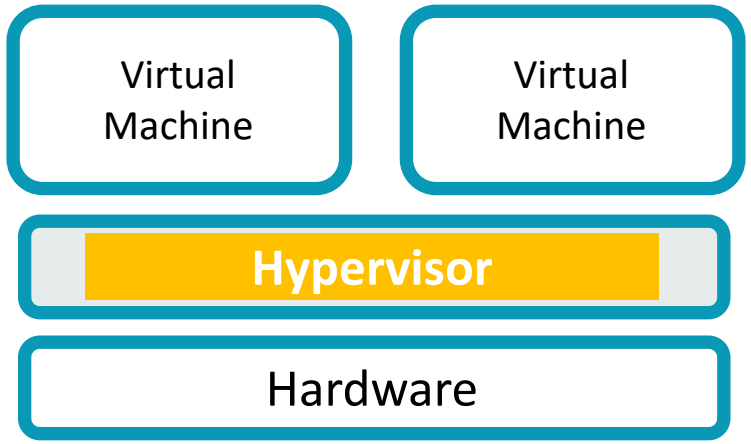


arm

SOC - Block Diagram



What is virtualization?

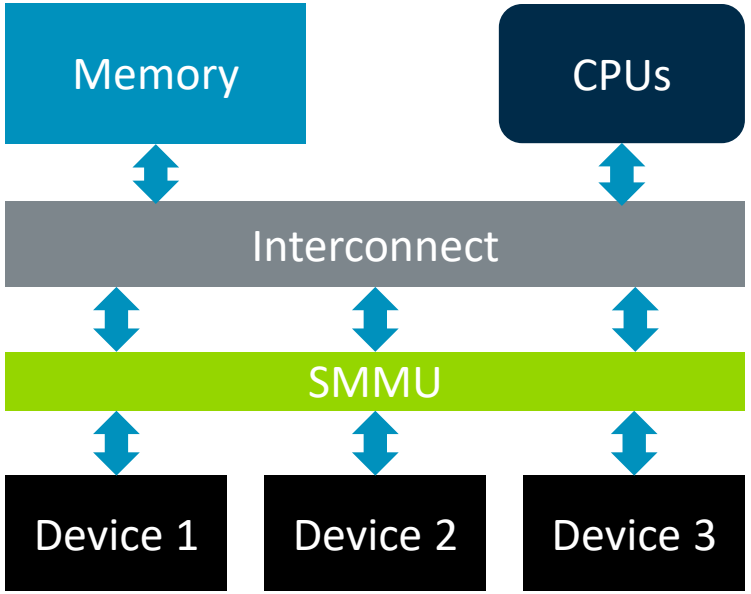


- Virtualization is the ability to create virtual machines that act like *real* machines
  - These virtual machines can run their own OS, often referred to as a Guest OS
- A Hypervisor or Virtual Machine Manager controls allocation of physical resources and execution time
- The level of virtualization/abstraction can vary based on the use case
- Memory Management Unit (MMU)

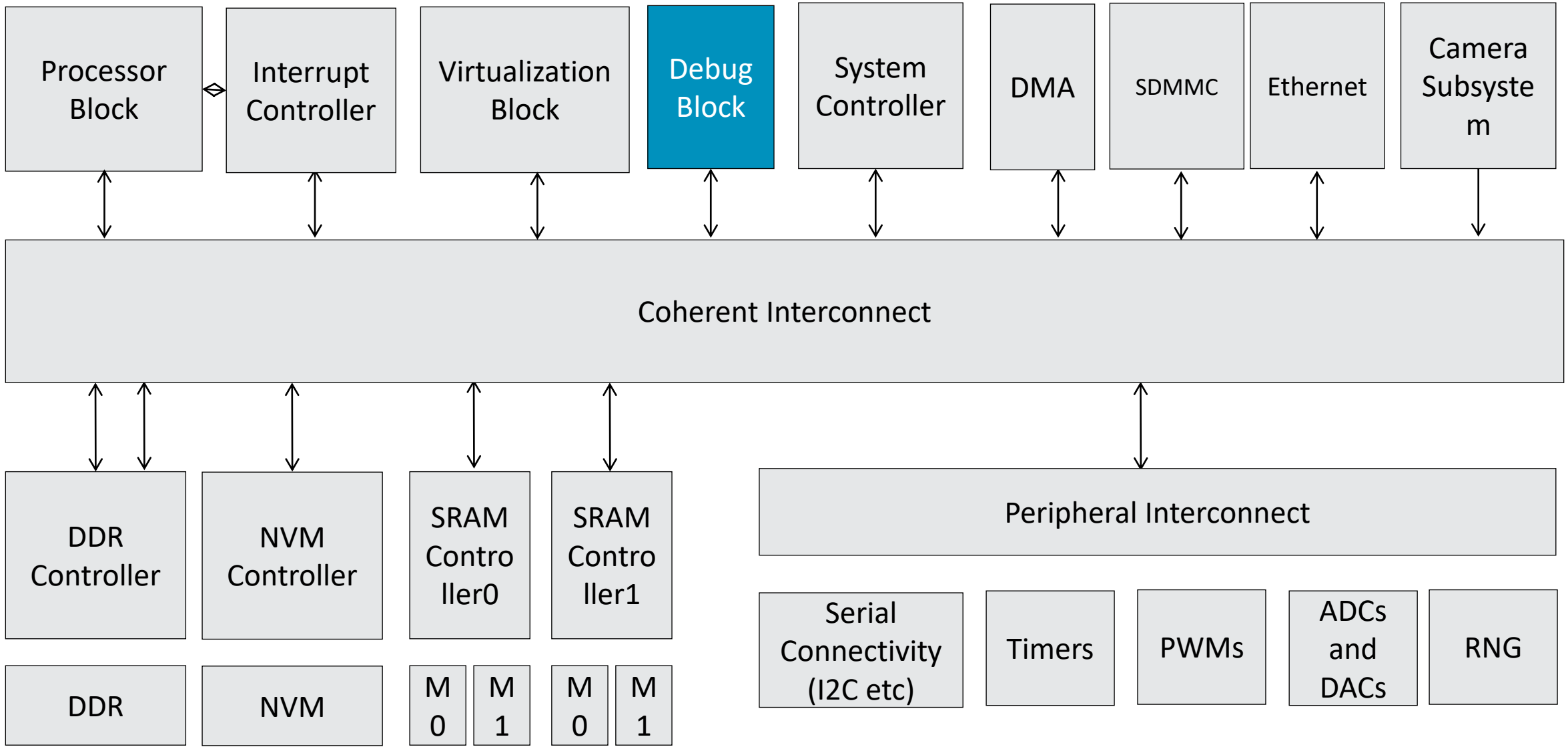
IO Virtualisation / System MMU /IOMMU

Why do we need one?

1. The SMMU can protect memory regions from devices
   - This helps provide extra security to the system.
2. Also, the SMMU can translate device addresses into physical memory addresses,
   - Very useful for devices only capable of 32-bit DMA
     - This allows these devices to function within a system without *bounce buffers*
   - On systems with insufficient contiguous physical memory, SMMUs provide contiguous virtual memory maps.

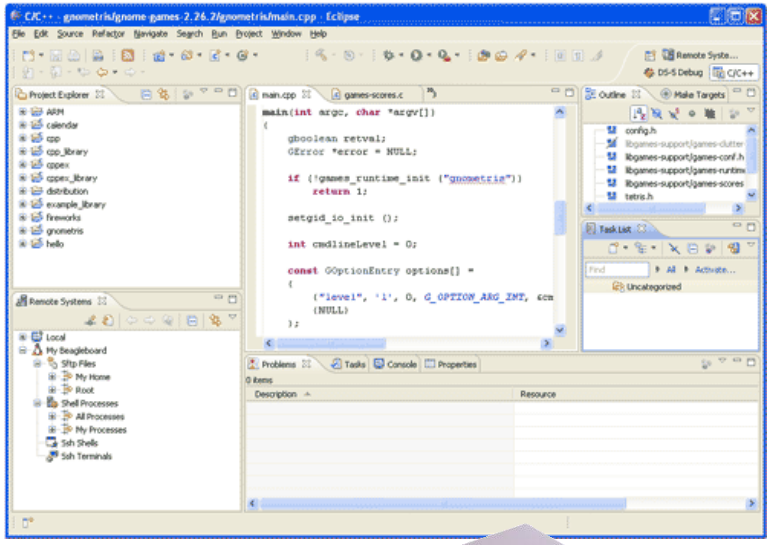


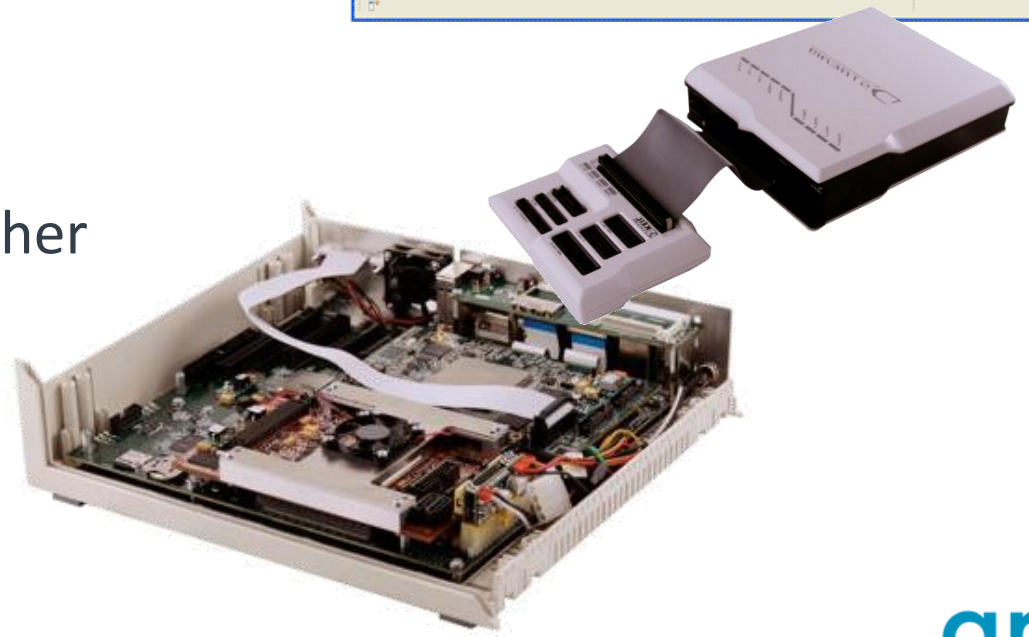
SOC - Block Diagram



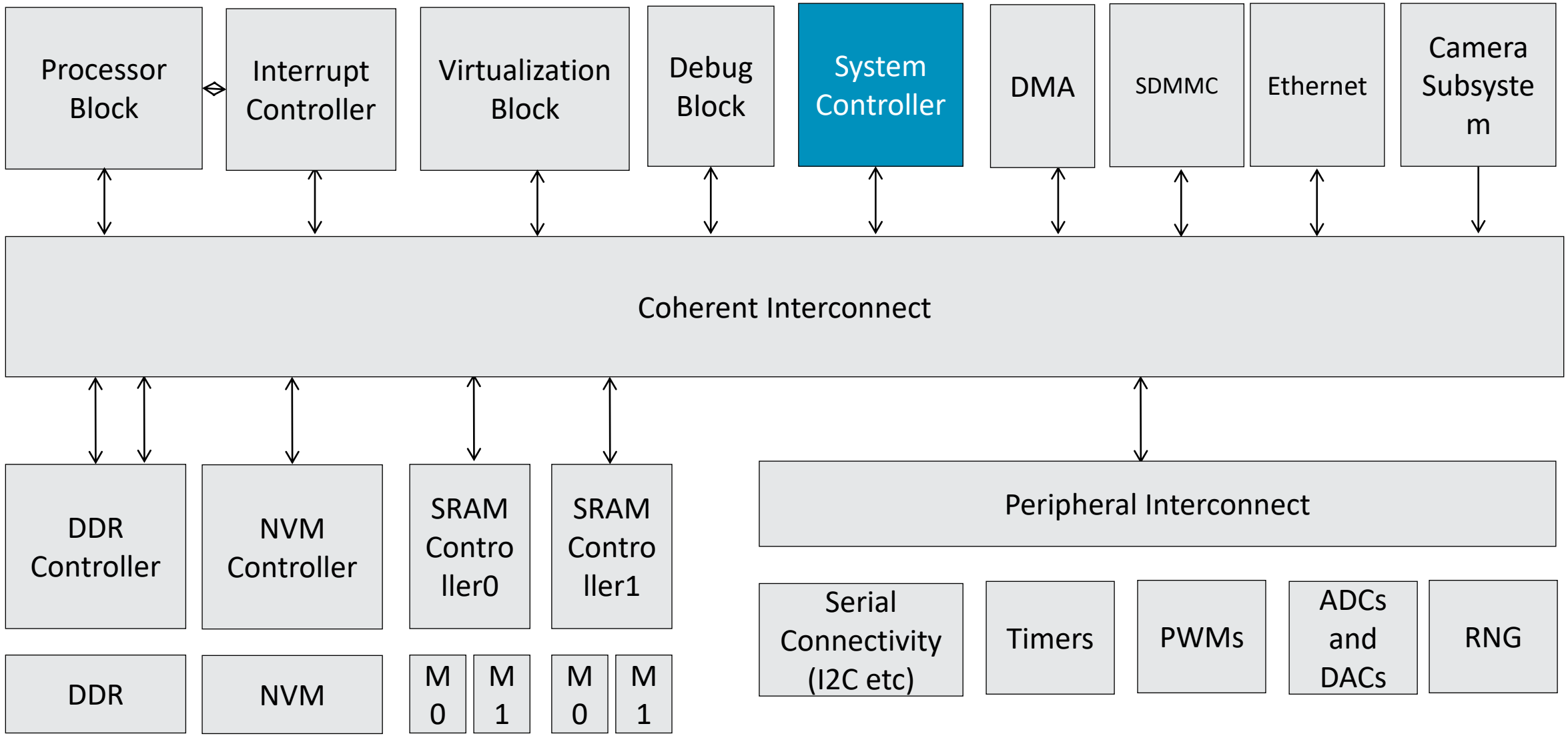
Debug Unit

- As of 2021 Linux kernel has 28 million lines of code
- Debugging failure in large code base is quite challenging
- Hardware Debug unit helps to pause, evaluate, inject instruction(in some cases) and restart
- Often used for board bring-up, driver/kernel development or isolating h/w issues
- Enables features not always available with other strategies – example: debug on reset





SOC - Block Diagram



System Controller (SCP/MCP) Features

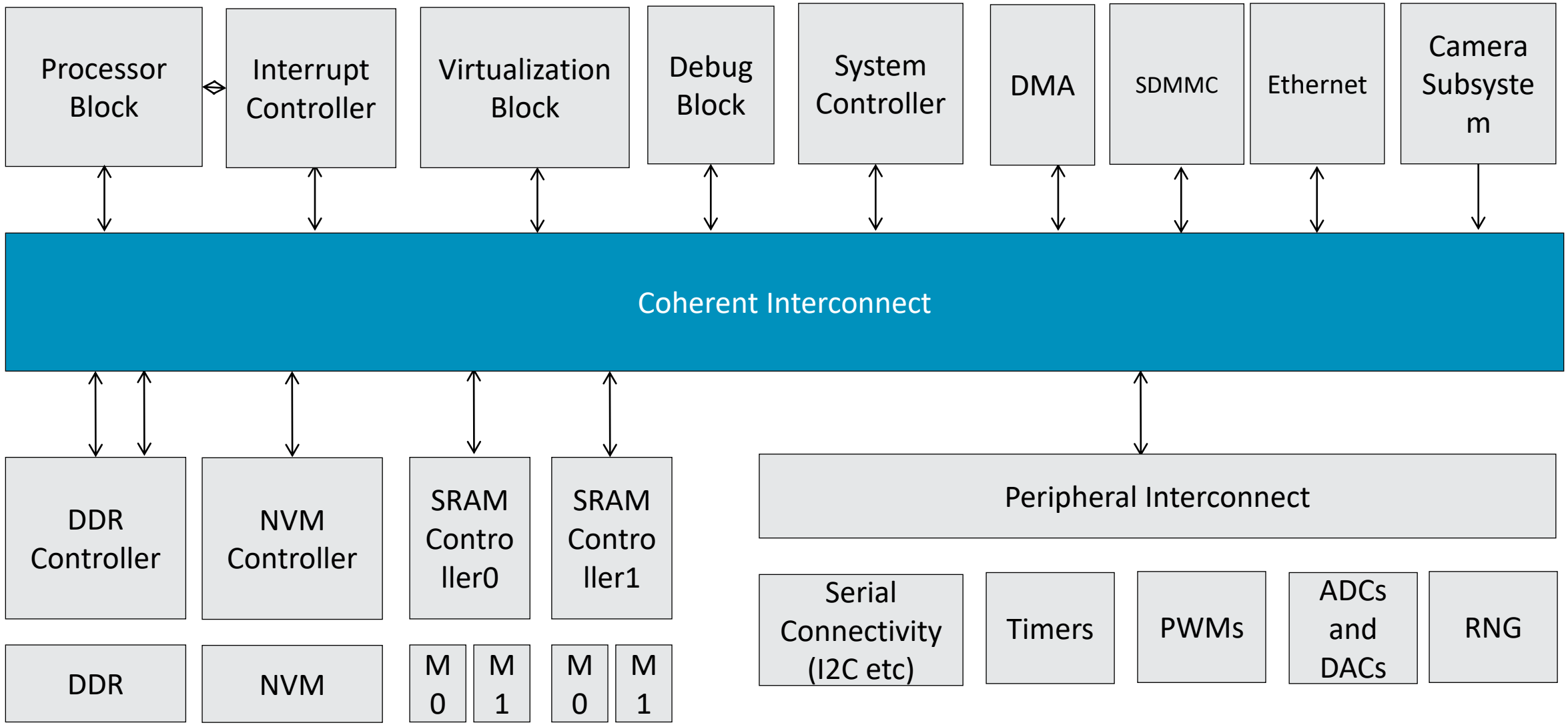
System Control Processor

- Cortex-M7 or similar micro-controller based system
- Manages the overall power, clock and reset of Compute Subsystem
- Manages thermal sensor control across SOC. Monitors sensors, power and thermal
- Start boot from its own private Boot ROM before the Application Cortex-'A' processor

Manageability Control Processor

- Cortex-M7 or similar micro-controller-based system
- Responsible for establishing connection with external BMC (Baseboard Manageability controller)
- Responsible for event logging
- Communicate alerts

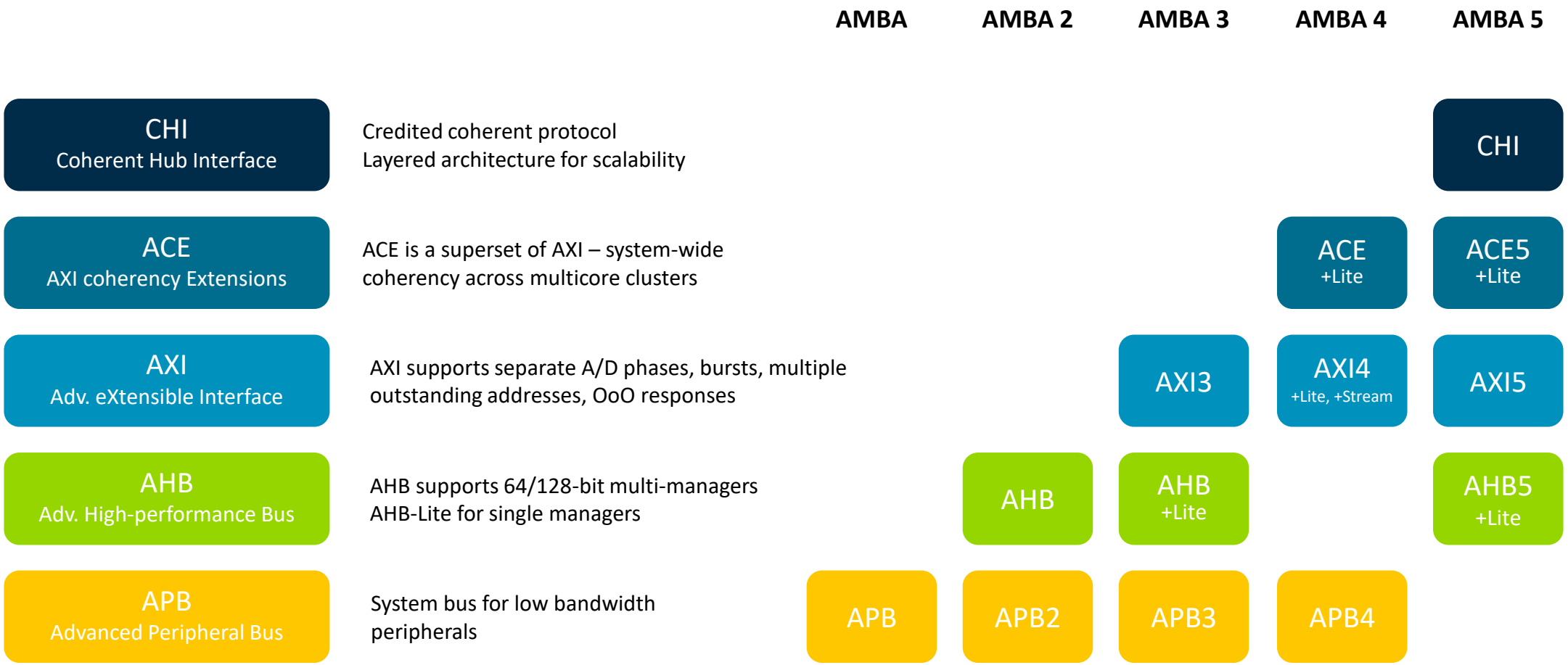
SOC - Block Diagram



Coherent Interconnect

- Efficiently communicate between multiple components of the SoC.
  - In a system there are multiple data storage locations (Cache, RAM etc). Data can be present in multiple locations simultaneously.
  - Coherent interconnect helps to fetch the latest data from all the components participating in the coherency.
- Support all features of the connected devices.
- Architecture helps to maintain backward compatibility

AMBA



arm

Thank You
Danke
Gracias
Grazie
谢谢
ありがとう
Asante
Merci
감사합니다
धन्यवाद
Kiitos
شكرًا
ধন্যবাদ
תודה