SoC Processor Selection & CPU Cores

COE838/EE8221: Systems-on-Chip Design
http://www.ecb.torontomu.ca/~courses/coe838/

Dr. Gul N. Khan
http://www.ecb.torontomu.ca/~gnkhan
Electrical, Computer & Biomedical Engineering
Toronto Metropolitan University

Overview

- Programmable SoC Processors
- Selection of Processors for SoC
- SoCs and Soft Processors
- ARM Cortex A9 Specs and Applications.
- Cortex-A9 Micro-architecture.
- Neon Media Processing Engine.

Cortex A9 Data and some Material from Chapter 3 of the Text by M.J. Flynn

Cortex A9 based Processing System

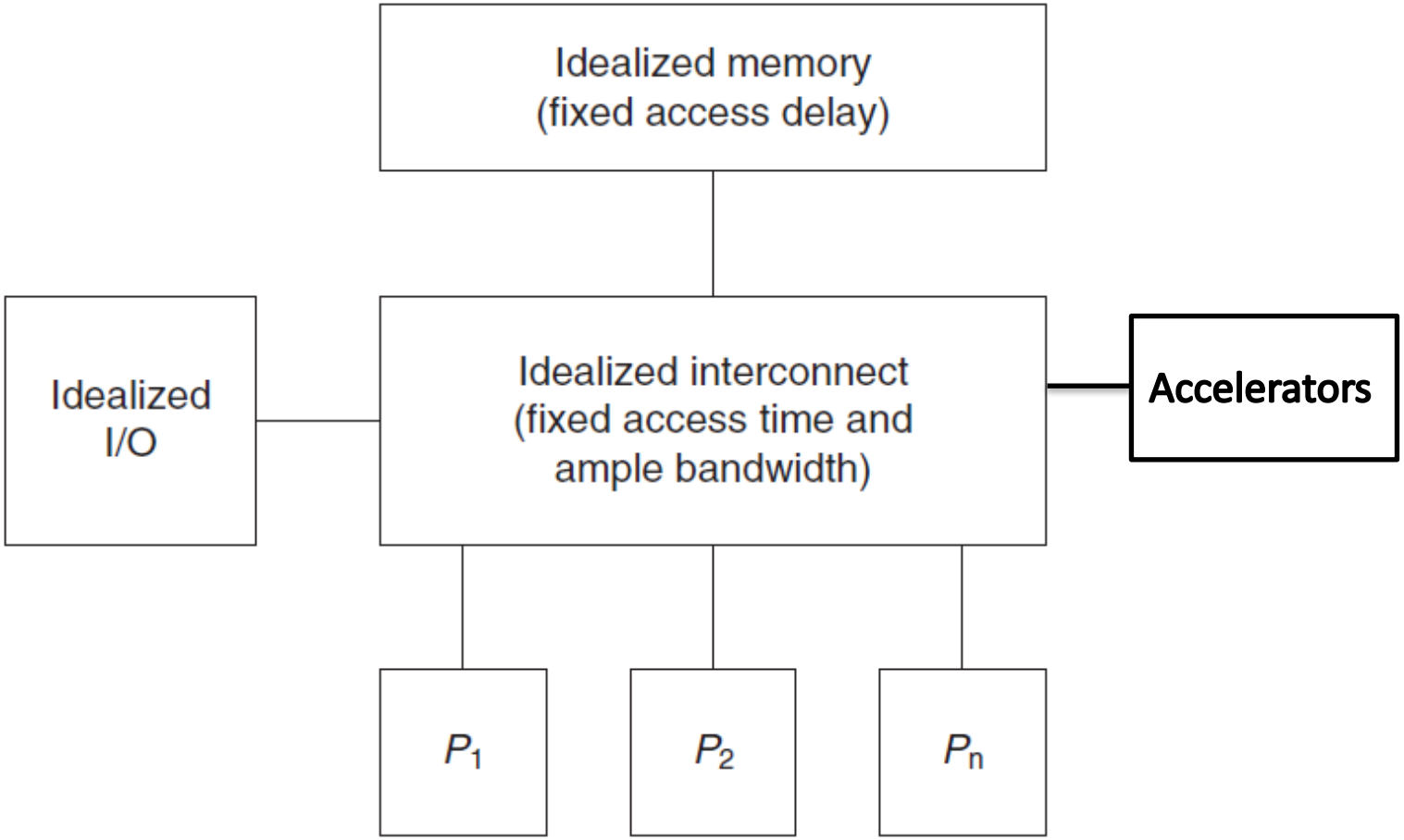
Xilinx and Intel-Altera Programmable SoCs have dual Cortex A9.

Hardened peripherals with Cortex-A9 CPU include Ethernet, I2C, USB, and CAN.

- Zynq 7000 SoC
- Cyclone-V and Arria-V (Stratix-V) platforms by Altera

SoC Model and Processors

Idealized (conservative performance) processor selected by function and real-time requirements

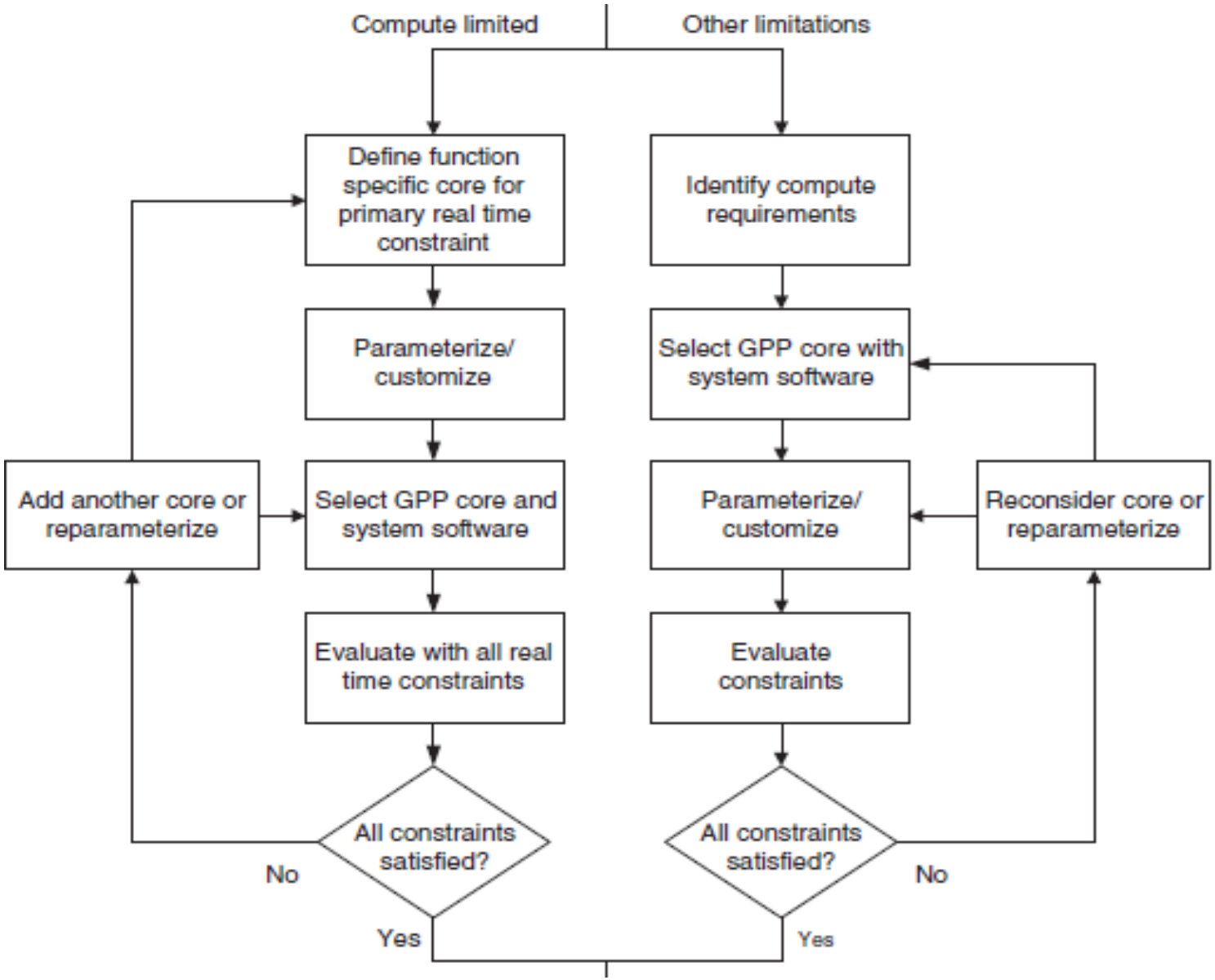


SoC Processor Selection

The process of CPU selection is limited as there can be a real time requirement that must be met by one of the processors or an accelerator:

- **This will become a primary consideration at an early stage in the initial SOC design phase.**
- **The processor selection and parameterization will produce the initial SoC design.**
- **The Initial SoC design appears to fully satisfy all the functional and performance requirements set out in the specifications.**
- **Such designs have higher chip area & power. However, there are some advantages of these systems.**
  - Cost reduction in terms of system-level integration
  - Design reuse
  - Creating an exact fit for a CPU/peripheral combination
  - Provide future expansion protection (against discontinuation)

Processor Core Selection



Soft Processors

OpenRISC and **Leon4** are free and open-source soft-core processors. Leon4 - implements a complete SPARC v8 (ISA).

- OpenSPARC- SPARC T1 core supports 1-4 threads on FPGAs.

	Nios II(fast)	MicroBlaze	OpenRISC	Leon4
Open source	No	No	Yes	Yes
Hardware FPU	Yes	Yes	No	Yes
Bus standard	Avalon	CoreConnect	WISHBONE	AMBA
Integer div. unit	Yes	Yes	No	Yes
Custom coprocessor/insts.	Yes	Yes	Yes	Yes
Maximum freq on FPGA (MHz)	290	200	47	125
Max-MIPS on/FPGA	340	280	47	210
Resources	1800 LE	1650 slices	2900 slices	4000 slices
Area estimate	1500 A	800 A	1400 A	1900 A

Soft Processors

OpenRISC: A free and open - source soft - core processor.

Leon4: Another free and open - source soft - core processor that implements a complete SPARC v8 **(ISA)**

OpenSPARC: SPARC T1 core supports 1-4 threads on FPGAs

- These Processors are based on a 32-bit Reduced Instruction Set Computer (RISC) architecture **(OpenSPARC has 64 bits)**
- Single-issue 5-stage pipelines and have configurable data-instruction caches
- Support for GNU (GCC) compiler tool chain.
- They also feature bus architectures suitable for adding extra processing units as slaves or masters.
- Some even go further and allow the addition of custom instructions and/or coprocessors.

More Soft Processors

CPU core	Architecture	Bits	License	Pipeline depth	Cycles per instruction[1]	MMU[2]	MUL[3]	FPU[4]	Area (LEs[5])
S1 Core	SPARC-v9	64	Open-source (GPL)	6	1	+	+	+	37000 - 60000
LEON3	SPARC-v8	32	Open-source (GPL)	7	1	+	+	+	3500
LEON2	SPARC-v8	32	Open-source (LGPL)	5	1	+	+	ext	5000
OpenRISC 1200	OpenRISC 1000	32	Open-source (LGPL)	5	1	+	+	-	6000
MicroBlaze	MicroBlaze	32	Proprietary	3, 5	1	opt	opt	opt	1324
aeMB	MicroBlaze	32	Open-source (LGPL)	3	1	-	opt	-	2536
OpenFire	MicroBlaze	32	Open-source (MIT)	3	1	-	opt	-	1928
LatticeMico32	LatticeMico32	32	Open-source	6	1	-	opt	-	1984
Cortex-M1	ARMv6	32	Proprietary	3	1	-	+	-	2600
DSPuva16	DSPuva16	16	Open-source	no	4	-	+	-	510

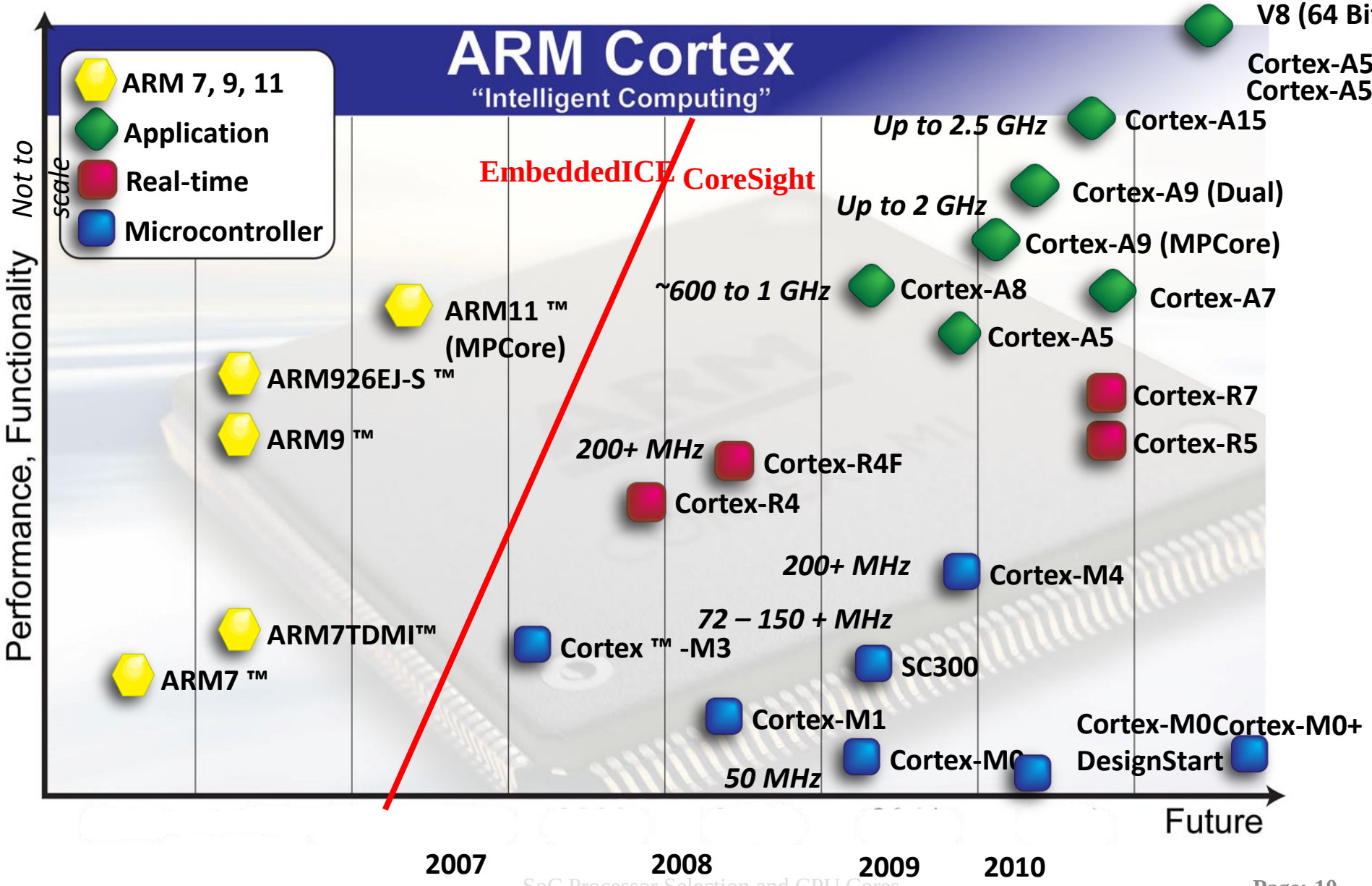
SoC Processor: ARM Cortex A9

Implementation size, performance, and very low power consumption are the key attributes of ARM architecture.

- ARM CPUs are RISC type with Uniform register file
- **Load/store architecture**
- **Simple addressing**

- The ARM Cortex-A9 processor is the high-performance choice from a family of low power, cost-sensitive CPU devices.
- The Cortex-A9 micro-architecture is delivered either as a Cortex-A9 single core processor or a scalable multi-core processor *i.e.*, Cortex-A9 MPCore processor.

ARM Public Processor Roadmap



Cortex-A9 Usage

- **Apple A5 (iPhone 4S, iPad 2, iPad mini)**

  A 32-bit system-on-chip by Apple manufactured by Samsung.

  A5 consists of a dual-core ARM Cortex-A9 CPU with NEON SIMD accelerator and a dual core GPU.

  A5 has 122.2 mm^2 die area, employ 45nm technology

- **NVIDIA Tegra 2 has A9 &**

  Geforce GPU, MP3/MPEG

  Motorola Xoom and Droid X2

http://en.wikipedia.org/wiki/ARM_Cortex-A9_MPCore#Implementations
http://en.wikipedia.org/wiki/Iphone_4s

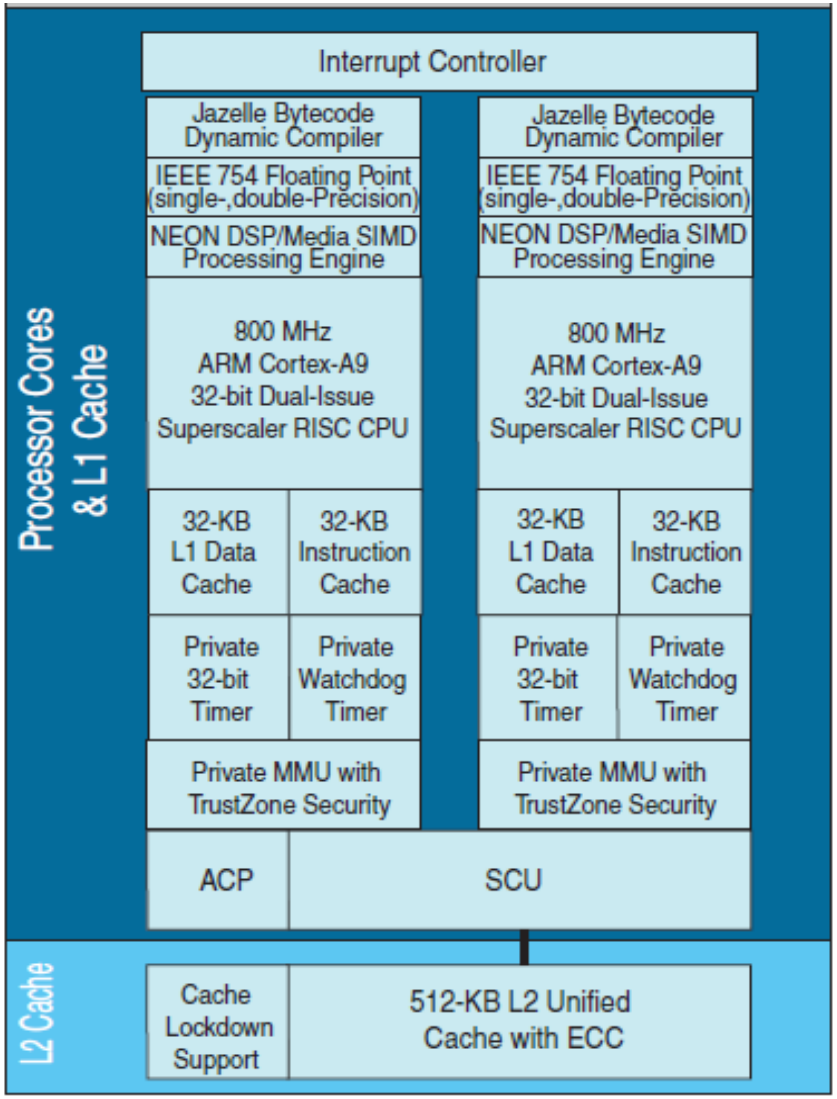
Cortex-A9 Usage

Play Station Vita

Up to 2GHz Quad-core ARM Cortex-A9 MPCore

http://en.wikipedia.org/wiki/ARM_Cortex-A9_MPCore#Implementations
http://en.wikipedia.org/wiki/PlayStation_Vita

Dual Core A9 CPU for the SoC



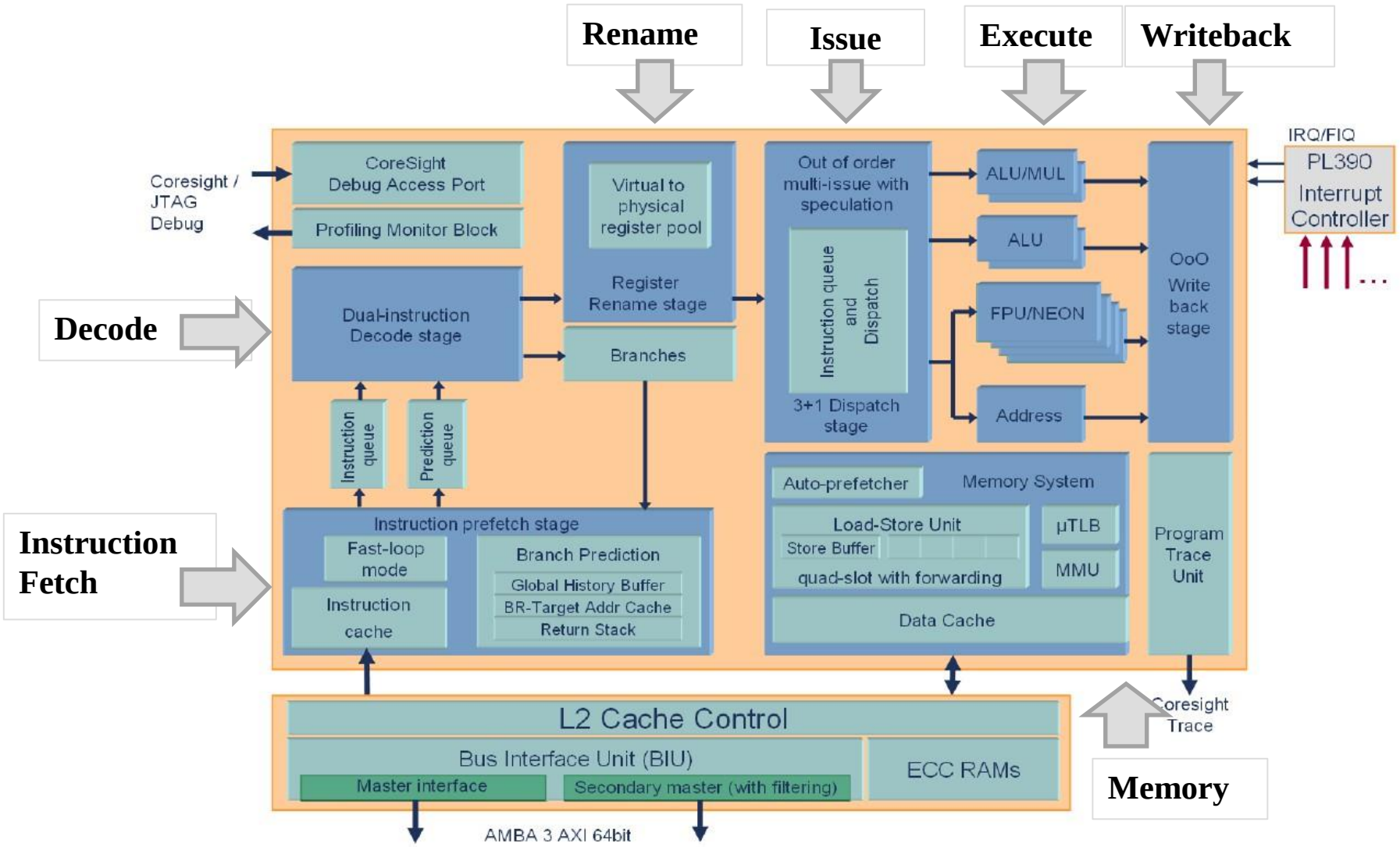
DE1-SoC MPU subsystem includes:

- Two Cortex-A9 processors
- L2 cache and memory subsystem
- SCU: Snoop Control Unit
- Accelerator Coherency Port (ACP)
- Debug functions

Cortex-A9 Micro-architecture

- Variable length, out of order, superscalar pipeline
  - Two instructions are fetched in one cycle
  - Issue up to 4 instructions per cycle into:
    - Primary data processing pipeline
    - Secondary data processing pipeline
    - Load-store pipeline
    - Compute engine (FPU/NEON) pipeline
- Speculative execution
  - Supporting virtual renaming of physical registers and removing pipelines stalls due to data dependencies.

Cortex-A9 Micro-architecture

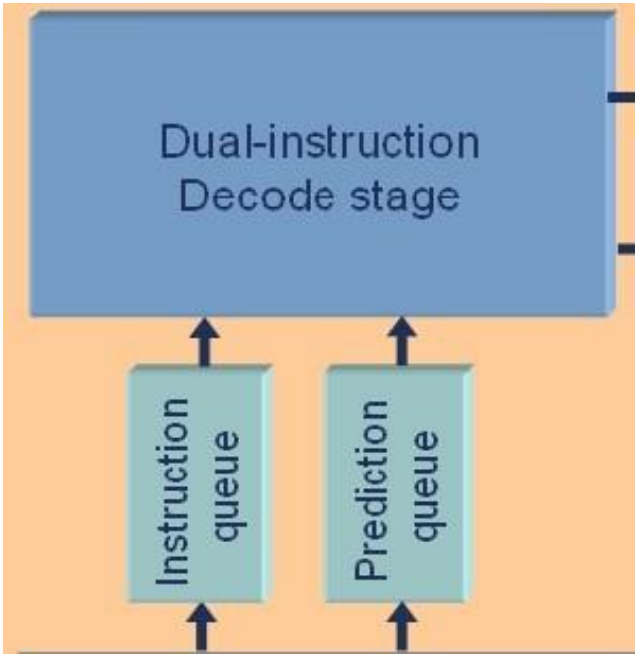


http://infocenter.arm.com/help/topic/com.arm.doc.100511_0401_10_en/arm_cortexa9_trm_100511_0401_10_en.pdf

Instruction Fetch

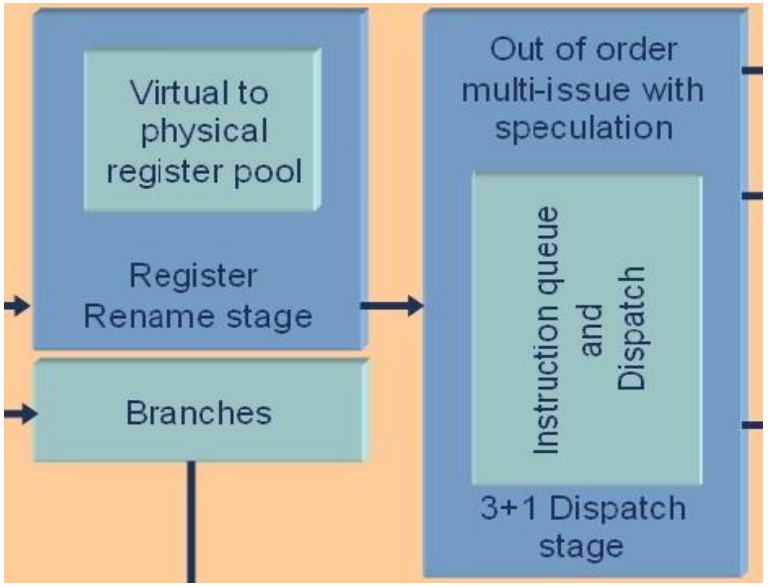
- **Instruction cache size:** 16KB, 32KB, or 64KB
- **Superscalar pipeline:** fetching two instructions at once
- **Branch Prediction:**
  - Global History Buffer: 1K ~ 16K entries
  - Branch-Target Address Cache: 512 ~ 4K entries
  - Return stack of 4 x 32 bits
- **Fast-loop mode:** instruction loop that are smaller than 64 bytes often complete without additional instruction cache accesses

Instruction Decode



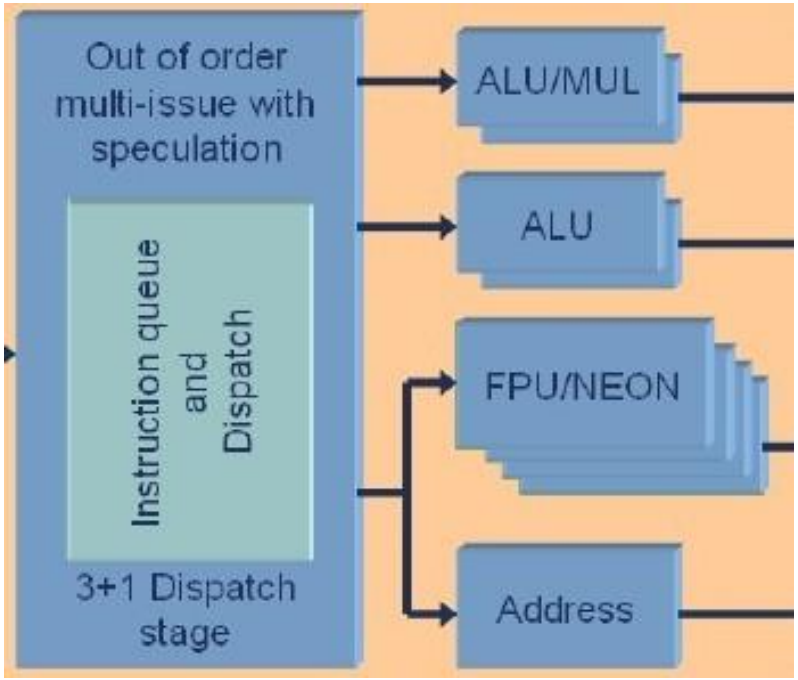
- **Super Scalar Decoder**
  - Capable of decoding two full instructions per cycle

Rename



- **Register Renaming**
  - Resolving data dependencies and unroll small loops by hardware

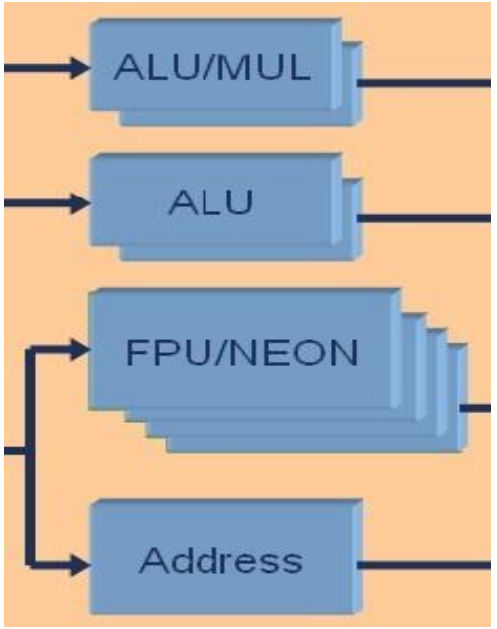
Issue



- **Issue can be fed a maximum of 2 instructions/cycle**
- **Issue can dispatch up to 4 instructions per cycle**
- **Out of order selection of instructions from the queue**

Execute Stage

- Variable length Executing Stage (1 ~ 3 cycles)
- Most Instructions finish in 1 cycle
- Instruction which folds shifts and rotates can take upto 3 cycles

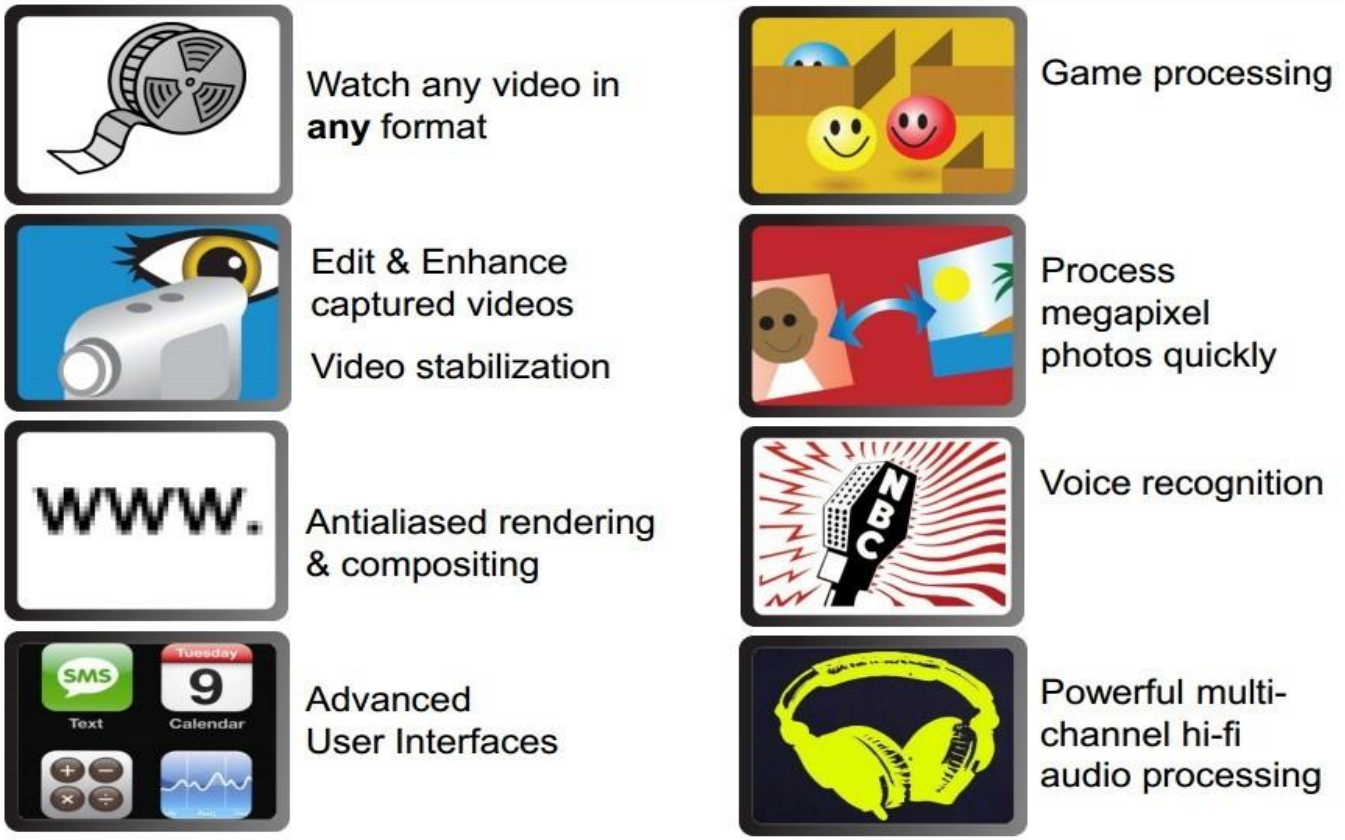


- ADD r0, r1, r2 (1 cycle)
- ADD r0, r1, r2 LSL #2 (2 cycle)
  Corresponds to a = b + (c << 2);
- ADD r0, r1, r2 LSL r3 (3 cycle)
  Corresponds to a = b + (c << d);

NEON MPE

- **NEON Media Processing Engine**

NEON technology supports instructions targeted primarily at audio, video, 3D graphics, image and speech processing.



http://infocenter.arm.com/help/topic/com.arm.doc.ddi0409i/DDI0409I_cortex_a9_neon_mpe_r4p1_trm.pdf

Cortex-A9 NEON

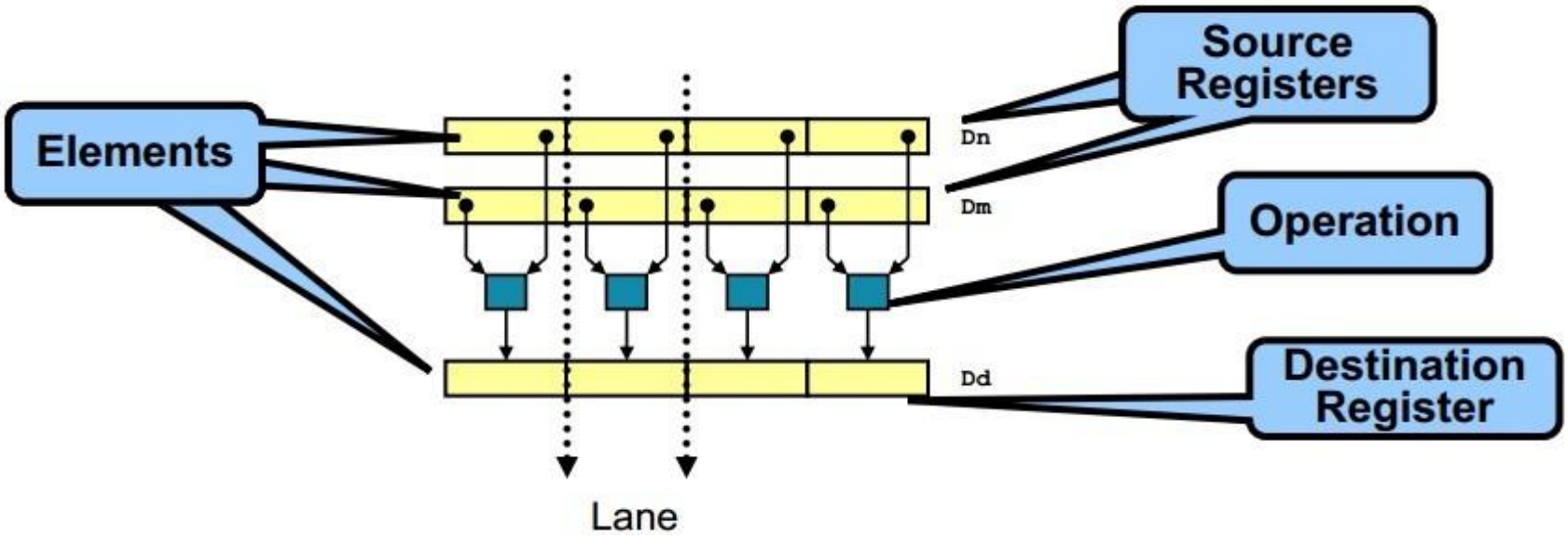
The Cortex-A9 NEON MPE (Media Processing Engine) features:

- SIMD and scalar single-precision floating-point computation
- Scalar double-precision floating-point computation
- SIMD and scalar half-precision floating-point conversion
- 8, 16, 32, and 64-bit signed and unsigned integer SIMD computation
- 8 or 16-bit polynomial computation for single-bit coefficients
- Large, shared register file, addressable as:
    * thirty-two 32-bit S (single) registers
    * thirty-two 64-bit D (double) registers
    * sixteen 128-bit Q (quad) registers.

NEON

What is NEON?

- NEON is a wide SIMD data processing architecture
  - 32 registers, 64 bit wide or 16 registers, 128 bit wide
- NEON instructions perform “Packed SIMD” processing
  - Registers can be considered as “vector” of same data type
  - Instructions perform the same operation in all lanes

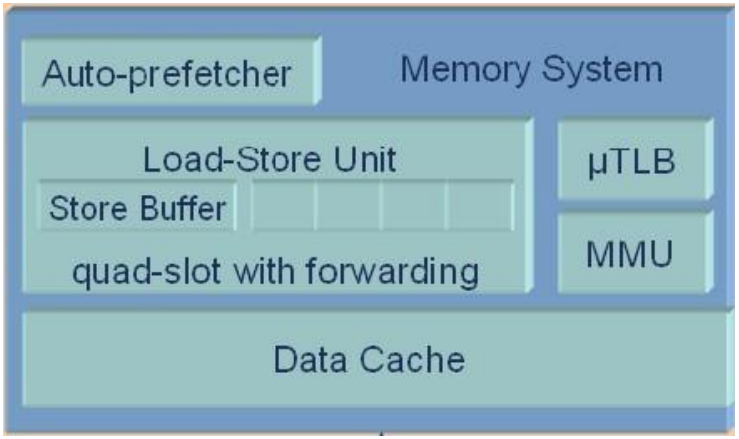


NEON MPE

- **NEON MPE (Media Processing Engine) supports vector computations on:**
  - half-precision (16bit), single-precision (32bit), double-precision (64bit) floating-point numbers
  - 8, 16, 32 and 64 bit signed and unsigned integers

- **Supported Operations Include:**
  - addition, subtraction, multiplication
  - maximum or minimum of a vector of operands
  - Inverse square-root approximation ($y = x^{-1/2}$)
  - And many more

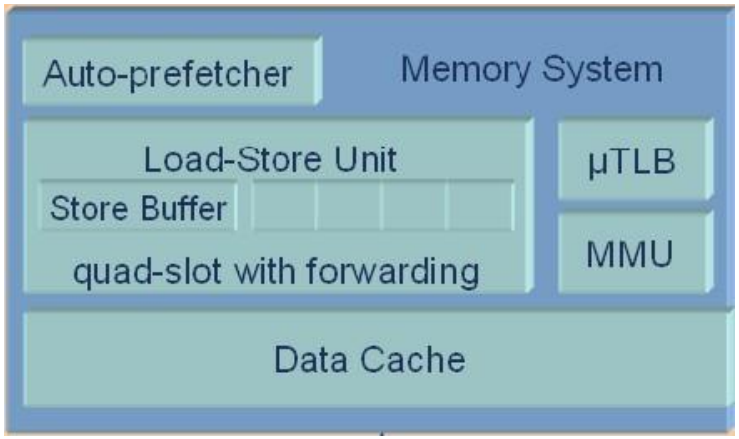
Memory



- **Dependent load-store instructions forwarded for resolution within memory system**
- **2-level TLB (*Translation Lookaside Buffer*) structure**
  - micro TLB
    - 32 entries on data side and 32 or 64 entries on instruction side
    - to reduce power consumed in translation and protection look-ups
  - main TLB

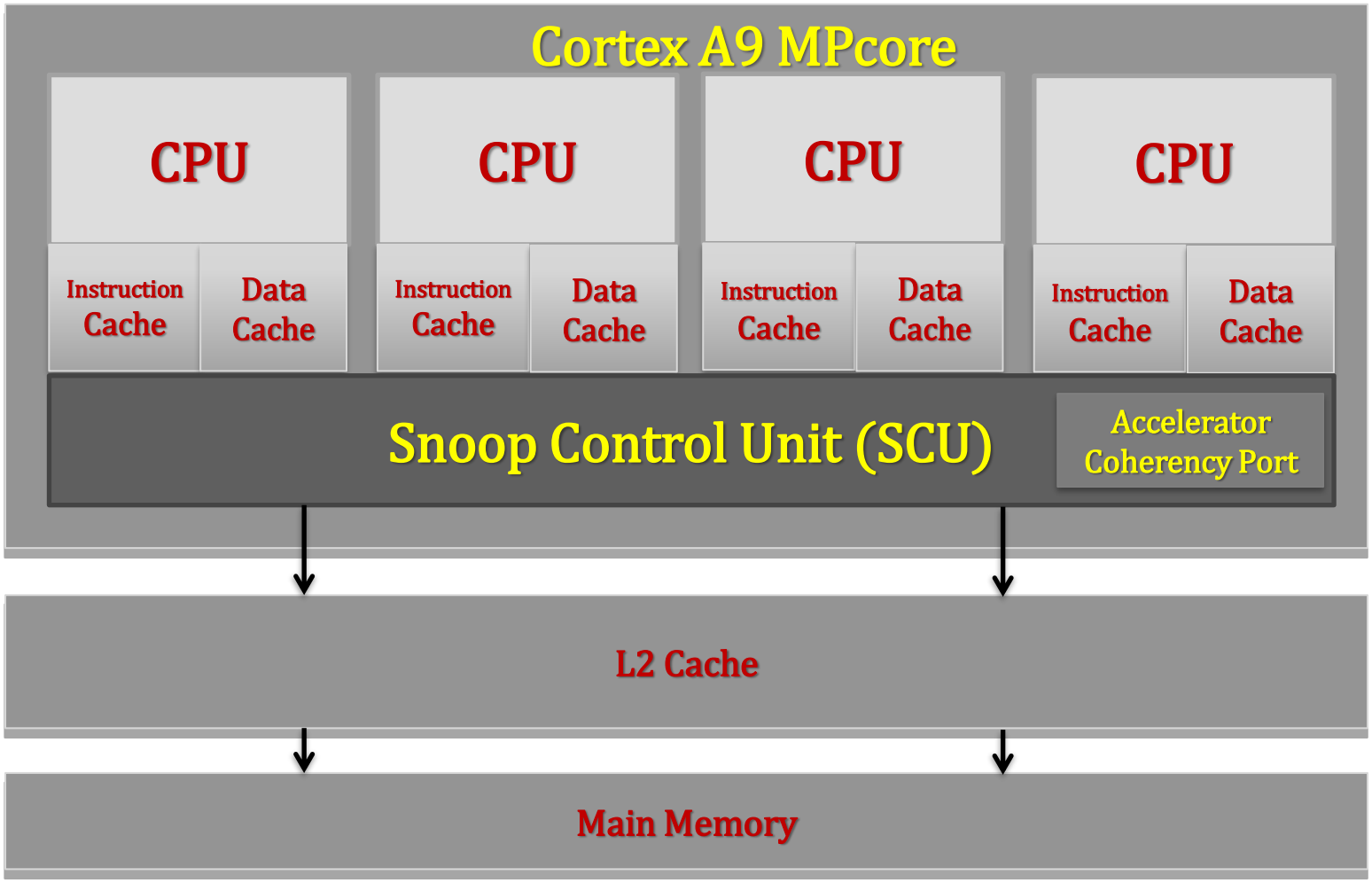
http://infocenter.arm.com/help/topic/com.arm.doc.ddi0388i/DDI0388I_cortex_a9_r4p1_trm.pdf

Memory (2)



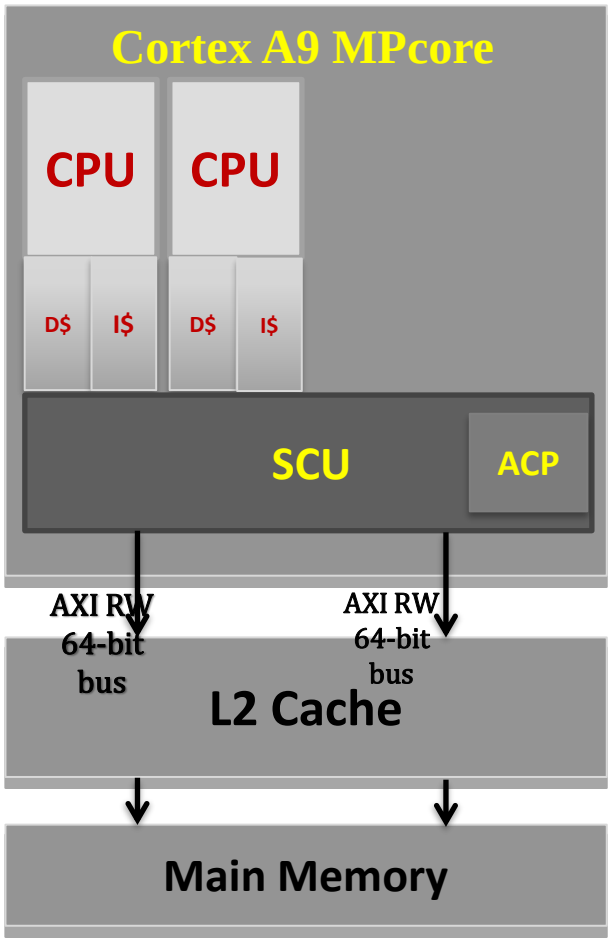
- **Data pre-fetcher**
  - monitor cache line requests by processor and cache misses to determine how much data to pre-fetch
  - can pre-fetch up to 8 independent data streams
  - pre-fetch and allocate data in the L1 data cache, as long as it keeps hitting in the pre-fetched cache line
  - When stop prefetching?

Memory Hierarchy



http://infocenter.arm.com/help/topic/com.arm.doc.ddi0407i/DDI0407I_cortex_a9_mpcore_r4p1_trm.pdf

L1 Caches

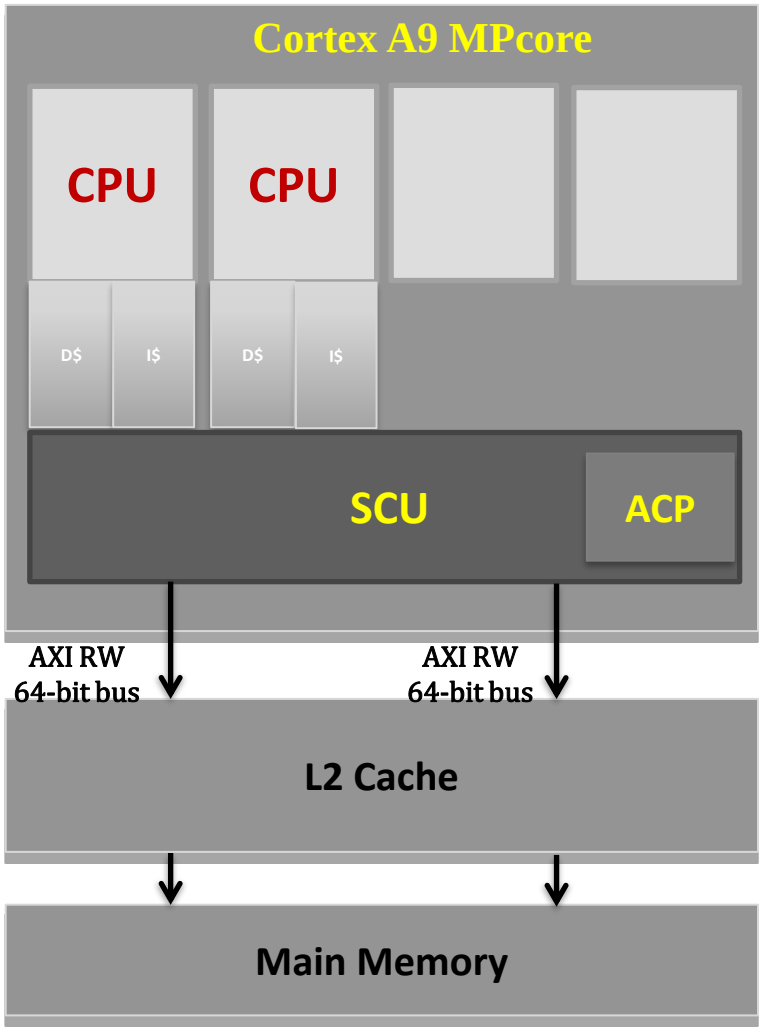


- Non-unified
  - 32 bytes line length
  - can be disabled independently
- 16, 32 or 64KB
- 4 - way associative
- support for Security Extensions
- **Instruction cache:**

  VIPT - **Virtually Indexed, Physically Tagged**
- **Data cache:**

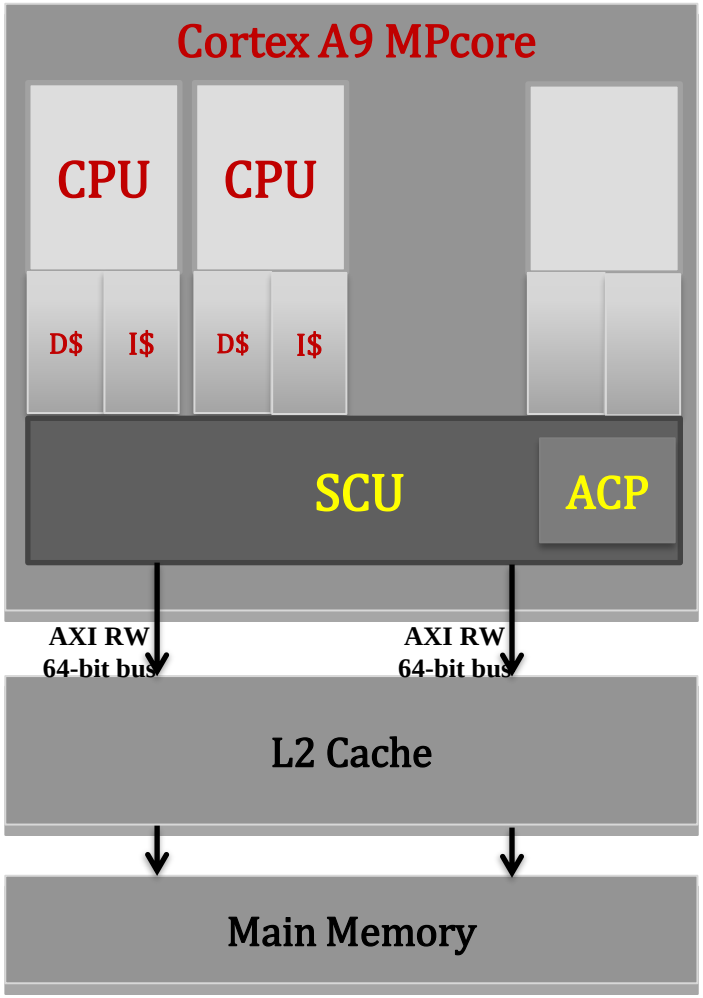
  PIPT - **Physically Indexed, Physically Tagged**
  - Reduce number of caches flushes and refills and save energy

L2 cache



- Shared, Unified
- Off-chip
- 128KB to 8MB
- 4 to 16-way associative

SCU: Snoop Control Unit



SCU connects 1-4 Cortex-A9 processor cores to memory system through AXI interfaces.

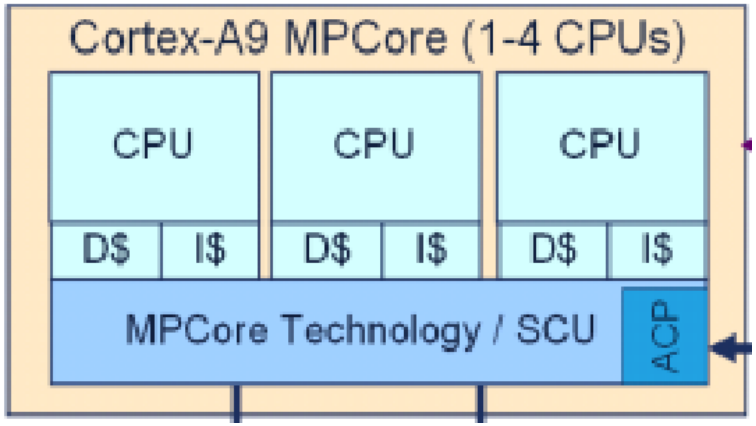
The SCU is an integral part of the cache memory systems and its functions are:

- Maintain data cache coherency between the Cortex-A9 processors
- Initiate L2 AXI memory accesses
- Arbitrate between Cortex-A9 processor cores requesting L2 accesses
- Manage ACP accesses.

The Cortex-A9 SCU does not support hardware management of coherency of the instruction cache.

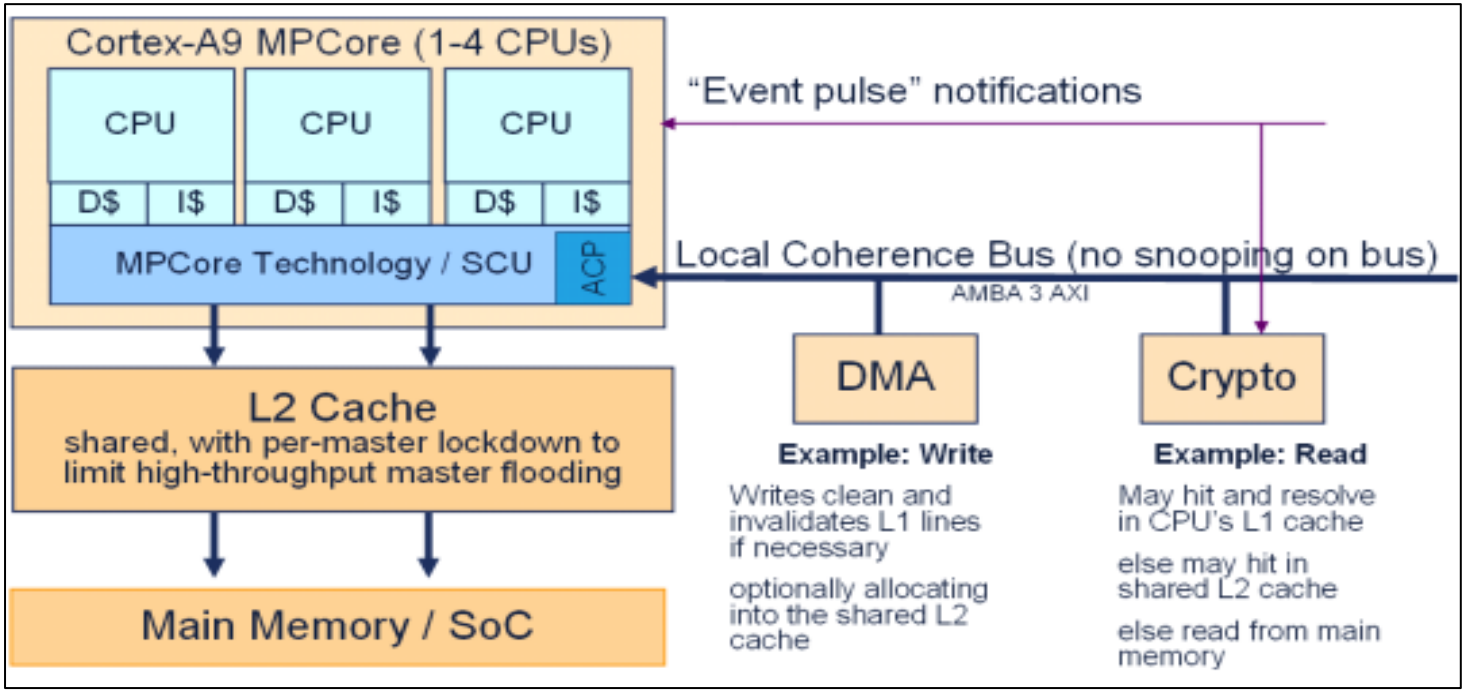
http://infocenter.arm.com/help/topic/com.arm.doc.ddi0407i/DDI0407I_cortex_a9_mpcore_r4p1_trm.pdf

Snoop Control Unit



- **SCU functions :**
  - maintain data cache coherency
  - initiate L2 memory accesses
  - arbitrate between processors' simultaneous request for L2 accesses
  - manages accesses from ACP
- **does not support instruction cache coherency**

ACP: Accelerator Coherency Port



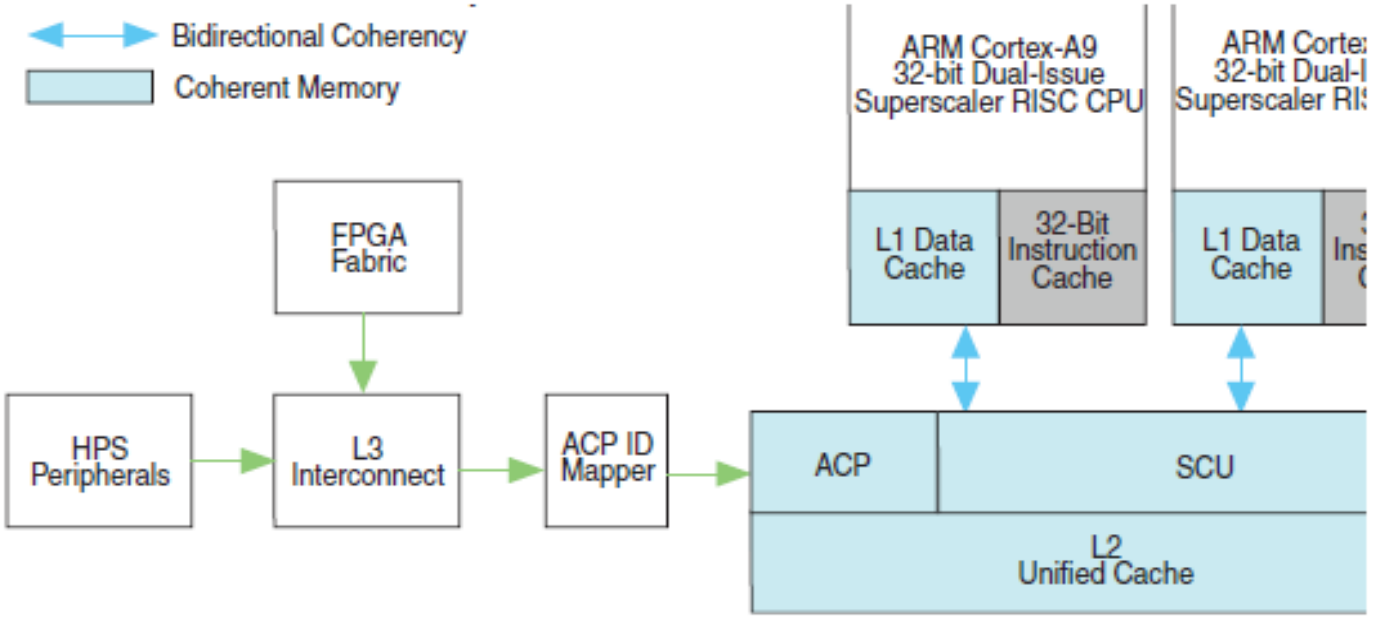
- **Optional AXI 64-bit slave port**
- **It allows to connect to non-cached system mastering peripherals and accelerators.**

  e.g. DMA engine or cryptographic accelerator
- **SCU enforces memory coherency**

Accelerator Coherency Port (ACP)

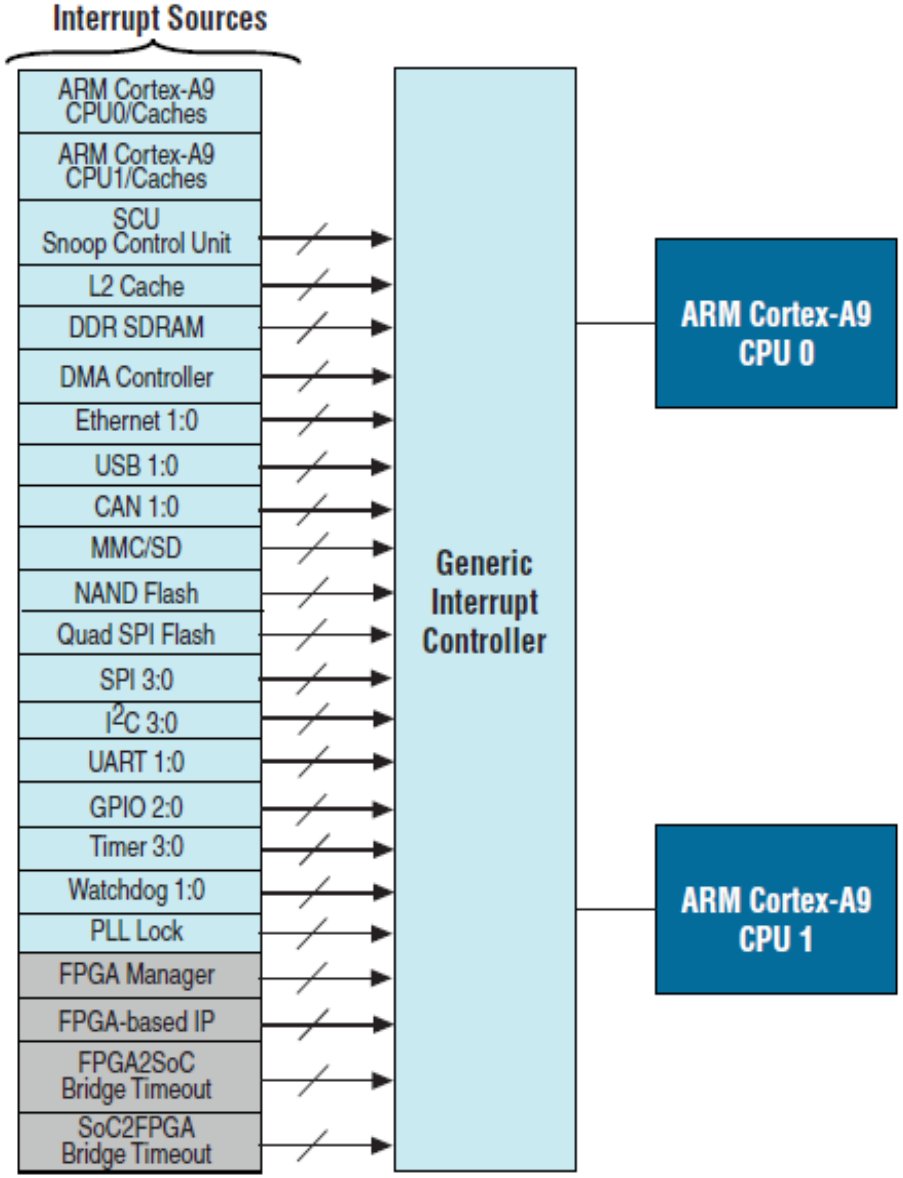
- ACP allows level-3 **(L3)** interconnect masters e.g., ethernet media access controller **(EMAC),** DMA and FPGA-to-HPS bridge to share data coherently with CPU.
- Read accesses to coherent memory regions **(With the ACP)** always return the most current data, whether in L1, L2 cache, or the main memory.
- Similarly, write operations to coherent memory regions cause the SCU to forward coherent data to memory system.
- The ACP ID mapper is located between the L3 interconnect and the ACP. The ARM ACP port is designed to support up to 8 unique transactions concurrently, where FPGA fabric may have any number of masters requesting coherent transactions.

Coherent Memory, SCU and ACP



SCU maintains bidirectional coherency among the L1 data caches ensuring both CPUs access the most recent data. When a CPU writes to coherent memory location, SCU ensures that the data is coherent ***i.e. updated/tagged/invalidated.*** SCU also monitors read operations from a coherent memory location. If the required data is already stored in the L1 caches, the data is returned directly to the requesting CPU. If the data is not in the L1 cache, the L2 cache checks its contents before the data is finally retrieved from the main memory.

GIC: Generalized Interrupt Control



The Interrupt Controller is a single functional unit, which is located in a Cortex-A9 MPCore.

- **GIC provides its control to the programmer.**
- **Handle interrupts in a centralized manner**

GIC: Generalized Interrupt Control

Which core services interrupts?

- GIC is responsible for centralizing all the interrupt sources before dispatching them to each individual Cortex-A9 processor core.
- There is one interrupt interface per Cortex-A9 processor core.
- The Interrupt Controller is memory-mapped.
- The Cortex-A9 processors access it by using a private interface through the SCU.