

ARM A55 Cortex

Austin Bae, Harrison
Ding
12/5/2018

Introduction

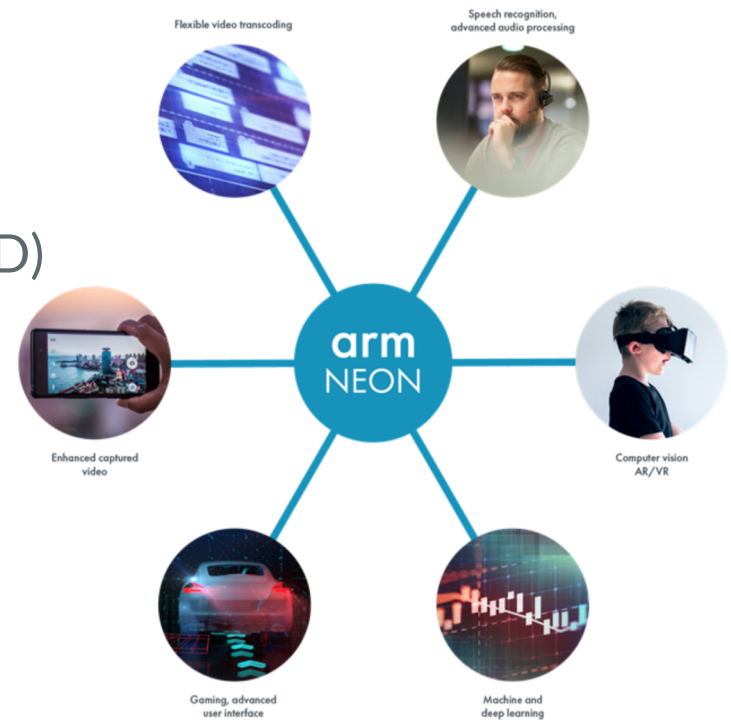
- Implements the ARM v8.2-A Instruction Set
 - Successor of ARM Cortex A53
 - 15% improved power efficiency
 - 18% improved performance
 - ARM stands for its 3 different profiles:
 - **Application Profile - Virtual Memory System Architecture**
 - Real-Time Profile - Protected Memory System Architecture
 - Microcontroller Profile - Programmer's model for low-latency interrupt processing
 - Great backwards compatibility through 2 different execution states
 - AArch64, AArch32 (compatibility with previous generations of ARM cortex)
 - DynamIQ technology Integration
 - Large focus on AI/Machine Learning
-

Microarchitecture Pipeline

- Dual-issue, 8-stage in-order pipeline
 - “Sweet Spot”
 - Branch Predictors
 - New conditional predictor uses Neural Net Algorithms
 - 0-cycle micro-predictors ahead of main predictor
 - Reduce Bubbles in the pipeline
 - Loop termination predictor to reduce penalty on loop exits
 - Separate indirect branch predictor that saves power
-

NEON Pipeline

- SIMD architecture extension
 - Audio/Video encoding/decoding
 - 2D/3D Graphics Rendering
 - AI (Machine Learning/Deep Learning/Computer Vision)
 - Signal Processing Algorithms
- NEON registers are considered as Vectors (SIMD)
- New operations added:
 - Dot Product/Cross Product (Vector Multiplication)
 - 16 int8/8 float16 operations per cycle
 - Made specifically for AI + Machine Learning
 - Affects 85% of Neural Net Algorithms
 - Fused Multiply-Add (FMA)
 - Very common sequential operation
 - Reduces latency by 50%



Memory Hierarchy

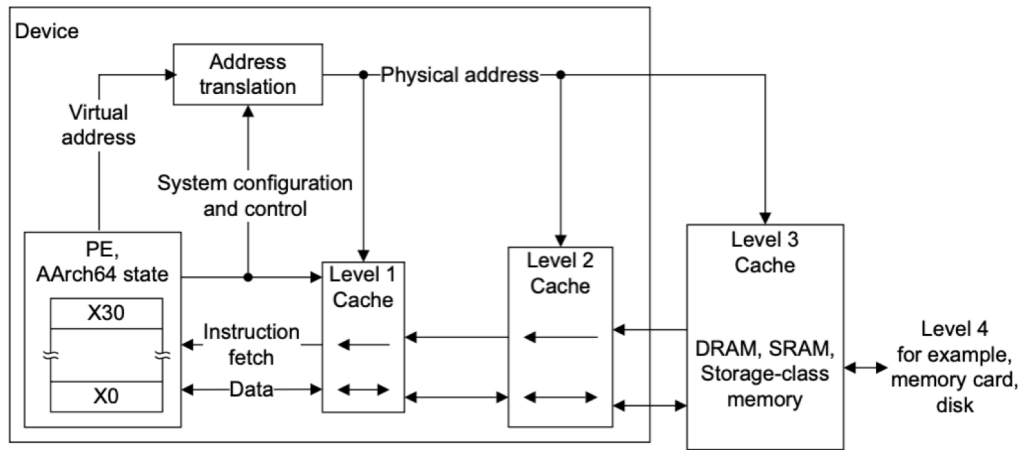
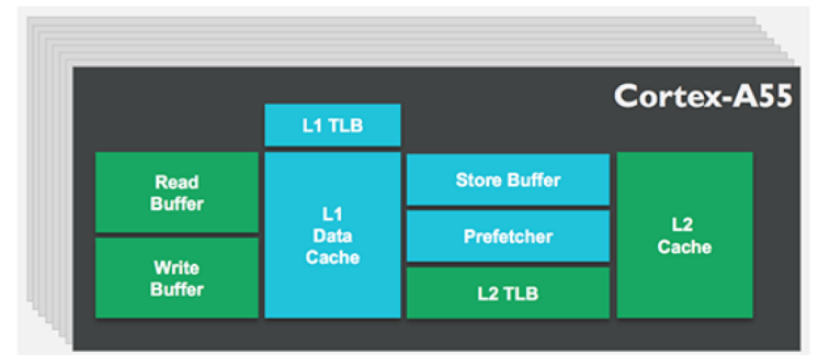
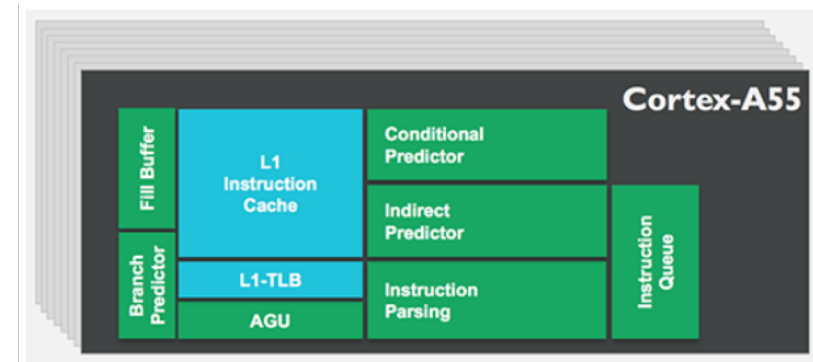


Figure B2-1 Multiple levels of cache in a memory hierarchy

- Includes L1 (Separate Instruction + Data Cache) and L2 on chip, and shared L3 cache
- All caches are 4-way associative
- Much better performance than A53 due to higher bandwidth

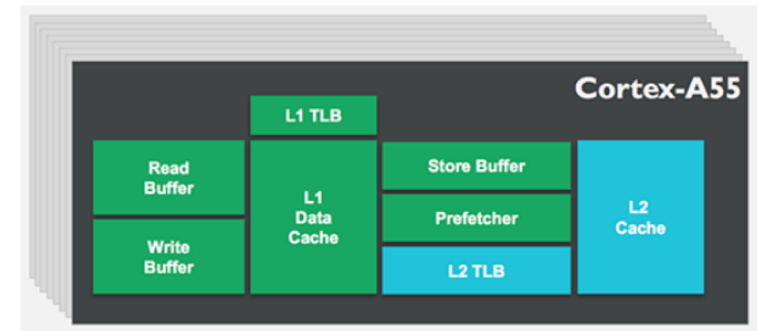
L1 Cache

- Instruction Cache
 - Configurable cache memory of 16KB, 32KB, or 64KB
 - VIPT (Virtually Indexed, Physically Tagged)
 - 15-entry TLB that supports different page sizes
- Data Cache
 - Higher Bandwidth upon prefetch, and can prefetch directly from L3 cache
 - Can detect more complex cache miss patterns
 - VIPT, but PIPT support as well (from A53)
 - 16-entry TLB (previously 10)
 - Larger store buffer with higher bandwidth

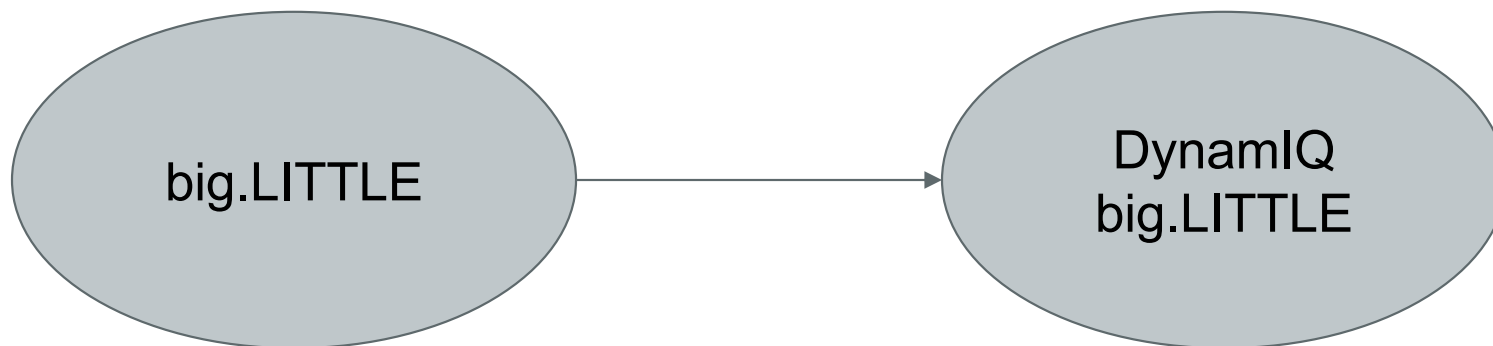


L2 and L3 Cache

- L2 Cache
 - Private to the core compared to shared L2 Cache in A53
 - Allows it to operate at core speed (variable)
 - 50% lower latency than off-chip L2s
 - Uses PIPT (Physically-Indexed, Physically-Tagged)
 - Simpler to implement
 - Waiting for TLB okay since L2 access naturally incurs higher latency than L1
 - 1024-entry TLB (increased size)
 - Smaller (4-way) associativity
- L3 Cache
 - Optional shared L3 cache off-chip



Multicore and Thread-Level Parallelism



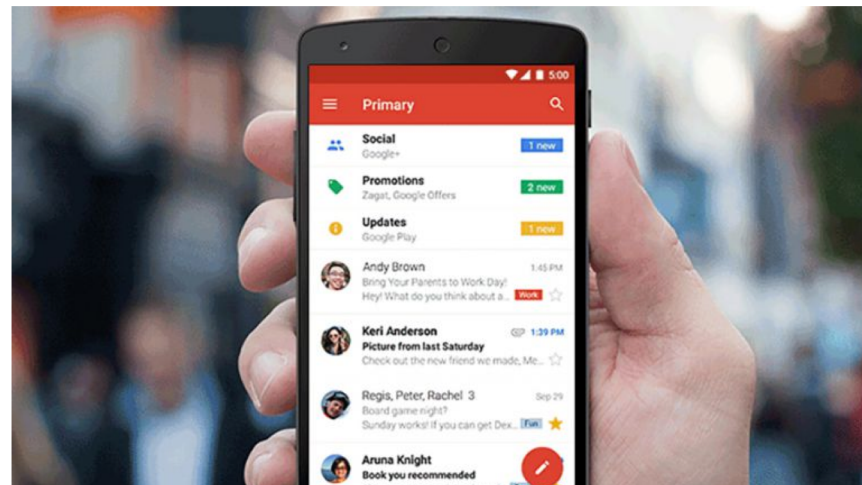
Basics of big.LITTLE

- Heterogenous processing architecture
 - LITTLE processor designed for power efficiency
 - big processor designed for maximum computing performance
- Dynamically allocates tasks to a big or LITTLE
- big and LITTLE cpus must be architecturally identical
 - Same instructions, support same extensions (e.g. virtualization and large physical addressing)

	1st Generation: Armv7 (32-bit, 40-bit physical)	2nd Generation: Armv8 (32-bit/64-bit)
High-performance CPU (big)	<u>Cortex-A15, Cortex-A17</u>	<u>Cortex-A57, Cortex-A72</u>
High-efficiency CPU (LITTLE)	<u>Cortex-A7</u>	<u>Cortex-A53</u>

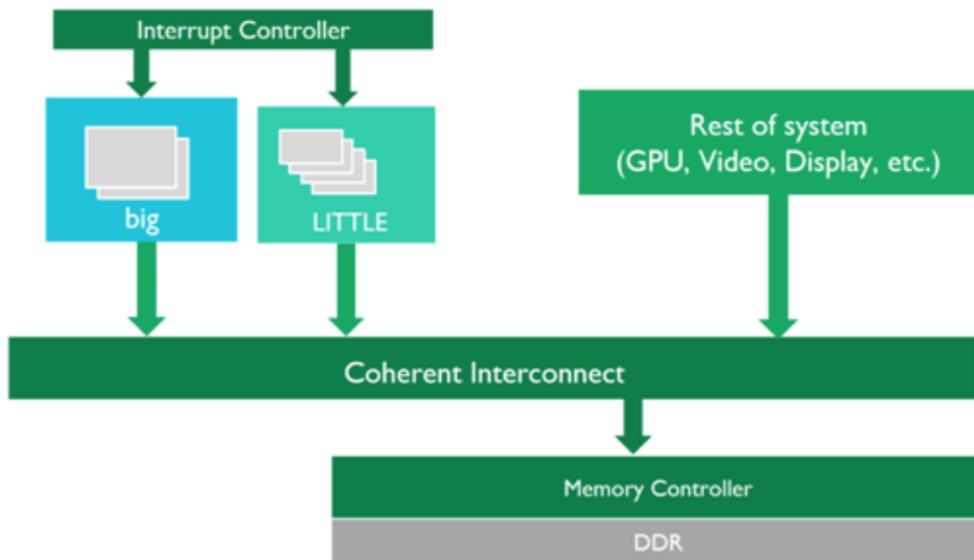
Basics of big.LITTLE (cont)

- Why we need it
 - Mobile gaming and web browsing vs. Texting and emailing
 - Highly varying computing requirements over the same system
- High peak performance + maximum energy efficiency
- Cores are allocated to clusters
 - Each cluster must contain the same type of cores
 - Maximum number of cores per cluster = 4
 - Nintendo Switch uses 4 Cortex A57 (big) and 4 Cortex A53 (LITTLE)

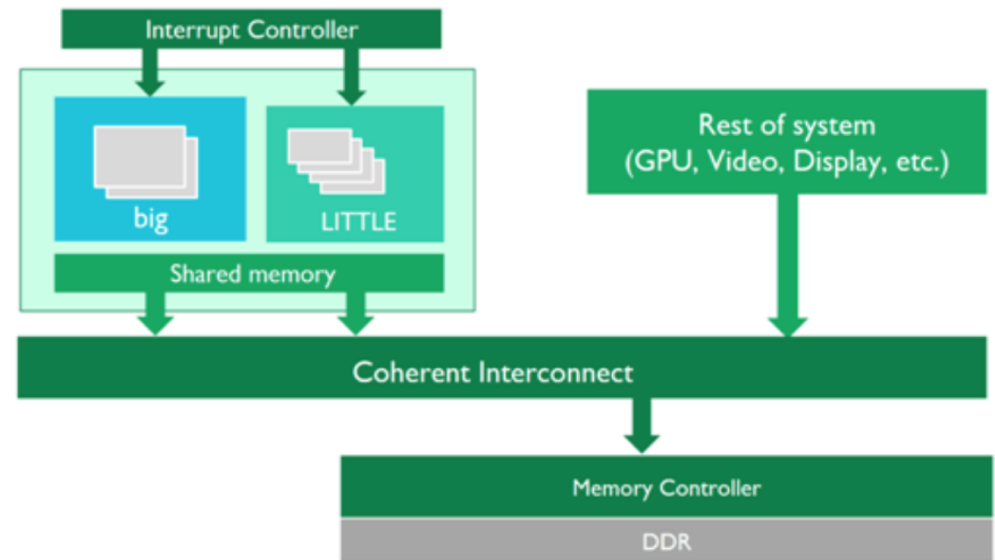


Introducing DynamIQ

big.LITTLE system



DynamIQ big.LITTLE system



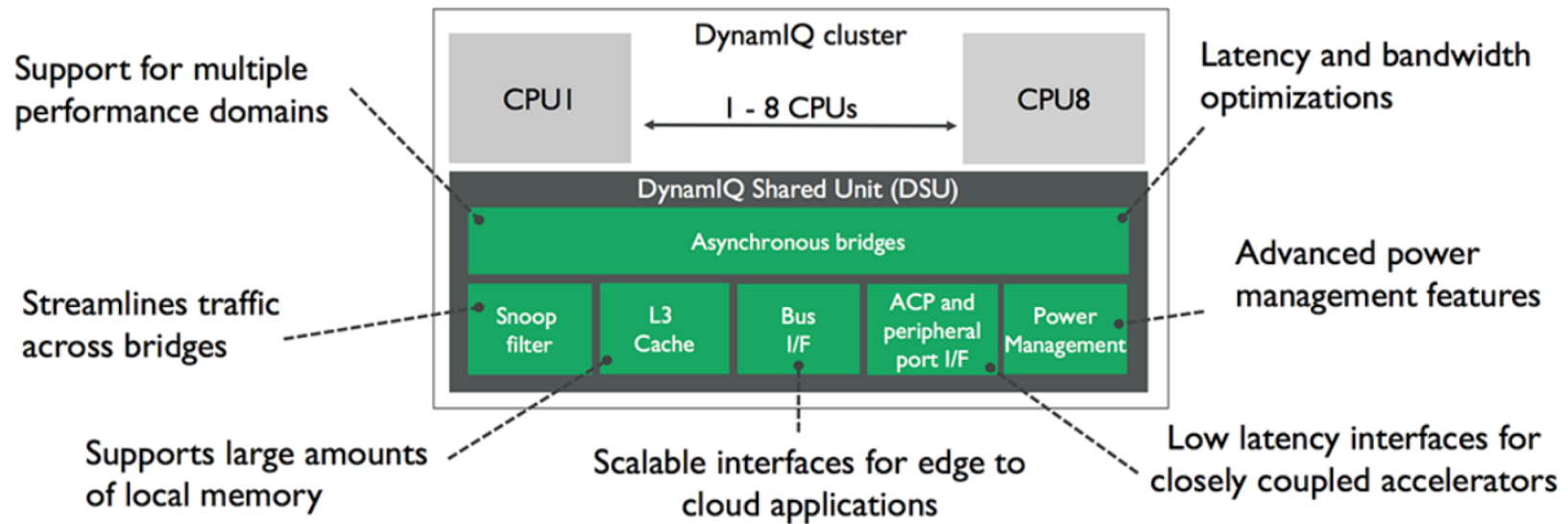
big.LITTLE

- Cluster containing up to 4 cores
- Each core in the cluster must be the same (e.g. all LITTLEs or all bigs)
- No L3 Cache
- Shared L2 cache

DynamIQ big.LITTLE

- Cluster containing up to 8 cores
- Any combination of LITTLEs and bigs through asynchronous bridging
 - 1 big + 7 LITTLEs or 2 bigs + 6 LITTLEs
- Pseudo-exclusive L3 cache
- Cache stashing
- Improved Power Management
- Private L2 cache
- Requires v8.2 ARM Architecture

DynamiQ Shared Unit (DSU)



DynamiQ Shared Unit (DSU)

- Asynchronous bridges
 - Technology behind running different processors in the same cluster
 - Each DynamiQ cluster is divided into domains based on Voltage/Frequency
 - Each domain contains an asynchronous bridge linked to the DSU
 - Enables support for different cores within each cluster
 - Sharing data within clusters is easier
 - Reduces latency between migrating threads from a big to a LITTLE and vice versa
 - Cache Stashing
 - Allows a specialized accelerator (such as a GPU) to read/write data directly into the L3 or even L2 cache
-

DynamIQ Shared Unit (cont.)

- Pseudo-exclusive L3 Cache
 - An optional cache that exists external to the CPU
 - 16-way set associative cache
 - Most likely reason why L2 cache is now private
 - Most of L3 cache data does not contain data in the L2 or L1 cache
 - Power Management
 - Portions of L3 cache can be turned off
 - Reduces leakage of power since L3 is optional
 - DSU performs all cache and coherency management through hardware rather than relying on software
 - Saves several steps in changing CPU power states
-

Works Cited

*All Images are from 2017 ARM Presentation for Cortex A55

“ARM Architecture Reference Manual.” *ARM v8*, ARM Holdings, 2018, static.docs.arm.com/ddi0487/da/DDI0487D_a_armv8_arm.pdf.

Arm Ltd. “Technologies | Big.LITTLE – Arm Developer.” *ARM Developer*, ARM Holdings, 2018, developer.arm.com/technologies/big-little.

Arm Ltd. “Technologies | DynamIQ – Arm Developer.” *ARM Developer*, ARM Holdings, developer.arm.com/technologies/dynamiq.

Humrick, Matt. “Exploring DynamIQ and ARM's New CPUs: Cortex-A75, Cortex-A55.” *RSS*, AnandTech, 29 May 2017, www.anandtech.com/show/11441/dynamiq-and-arms-new-cpus-cortex-a75-a55/4.

Triggs, Robert. “A Closer Look at ARM's New Cortex-A75 and Cortex-A55 CPUs.” *Android Authority*, Android Authority, 14 Aug. 2018, www.androidauthority.com/arm-cortex-a75-cortex-a55-breakdown-770380/.
