

Enabling Mobile Innovation with the Cortex™-A7 Processor

TechCon 2011. Santa Clara, CA

Brian Jeff, ATC-217,

October 2011

Abstract

ARM's newest processor, the Cortex-A7, is designed for the very efficient, low-cost main stream mobile handset market. In addition, because of a new ARM innovation, this power efficient processor will also be used in high-end superphones and tablets as a companion processor to the Cortex-A15 CPU as a complementary pair, in a new approach called big.LITTLE processing. This paper will discuss how the extremely power-efficient design will enable entry smartphone SoC designs as well as high end mobile products. This paper will describe in detail the design choices considered including choice of feature set and performance level, and how its simplified pipeline enables dramatically lower power consumption. This processor is ideal for not just the mobile but a slew of other embedded markets.

Contents

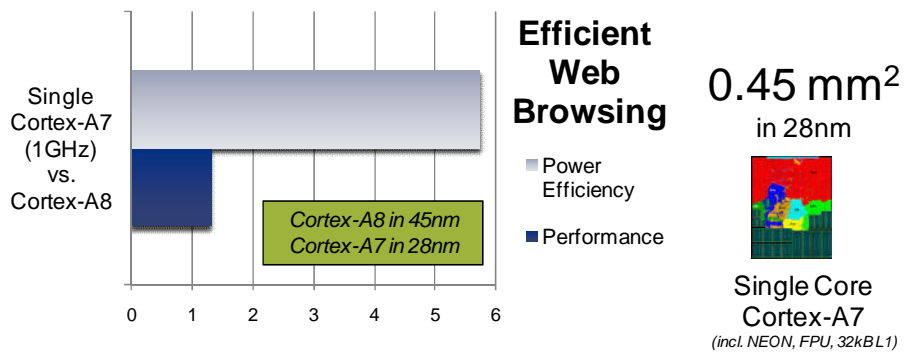
1	Cortex-A7 applications processor	2
	Cortex-A7 Microarchitecture	2
	Energy Efficiency Features of the Microarchitecture	4
	Memory System Tuned to Minimize memory latency	4
	Cortex-A7 performance.....	4
	Implementation	4
	Software Benchmarks.....	4
	Cortex-A7 Target Markets	5
	Cortex-A7 in Low-Cost Smartphones	6
	Market Requirements.....	6
	Power/Performance	6
	Power/Performance for Multicore Cortex-A7	6
	Area Diagrams.....	6
	Cortex-A7 in High-end Smartphones, Tablets, and Beyond	7
	Market requirements for high-end mobile	7
	big.LITTLE Processing	7
4	Conclusion.....	8
5	About the author	8

Cortex-A7 applications processor

The Cortex-A7 processor was designed primarily for power-efficiency and a small footprint. The design team based the pipeline on the extremely power efficient Cortex-A5 CPU, then added microarchitecture enhancements to increase performance and architectural enhancements to deliver full software compatibility with the Cortex-A15 CPU. These architectural enhancements include support for virtualization and 40-bit physical address space, and AMBA® 4 bus interfaces. Virtualization and large address space are unusual features for so small a CPU, but are critical to present a software view of the Cortex-A7 that is identical to the Cortex-A15 high-end CPU.

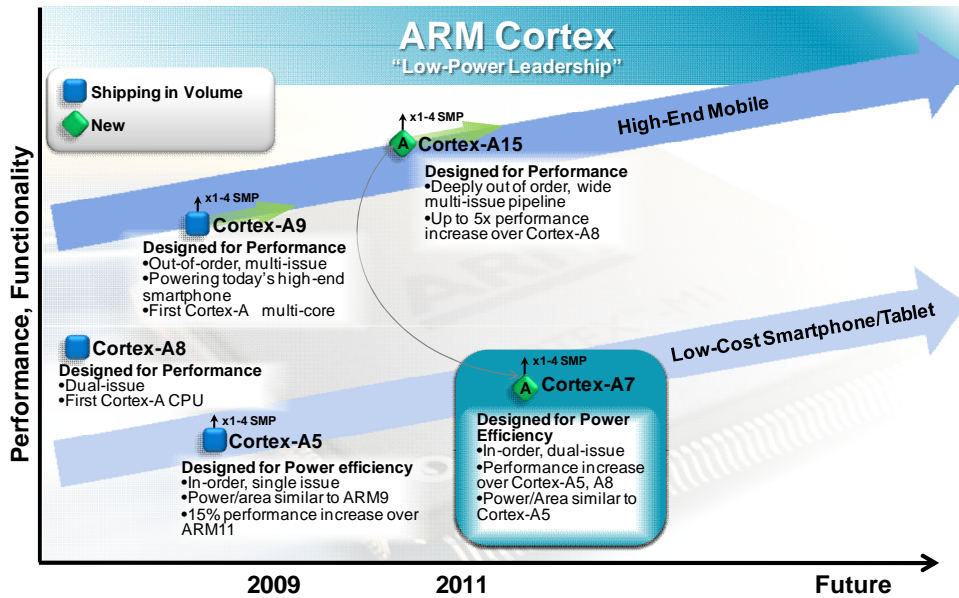
Like the Cortex-A5, Cortex-A9, and Cortex-A8 processors that came before it, the Cortex-A7 processor is a full ARM v7A CPU, with support for the Thumb®-2 instruction set, optional 32-bit/64-bit floating point acceleration and optional NEON™ 128-bit SIMD architectural blocks. The Cortex-A7 also includes support for TrustZone® to enable secure operating modes which are increasingly important in modern mobile OEM designs. To bring higher scalability, the Cortex-A7 is also configurable as a multicore processor, supporting 1-4 cores in a coherent cluster.

The Cortex-A7 is a simple in-order pipeline with significant but not complete dual-issue capability; however the careful choice of design features has enabled the performance of a single Cortex-A7 core to outperform the full dual-issue Cortex-A8 CPU on some important benchmark tests like web browsing, while consuming up to 60% less power.

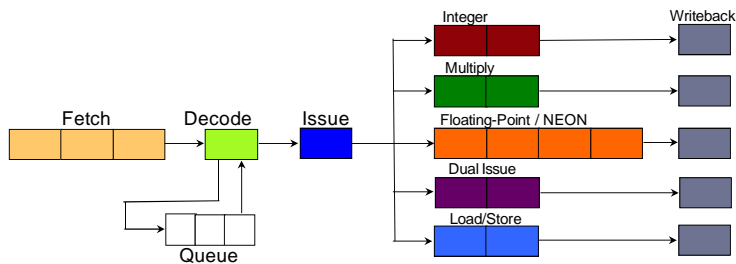


Cortex-A7 Microarchitecture

The roadmap below shows the legacy of Cortex-A class CPU designs, beginning with the Cortex-A8. In that design, ARM introduces the NEON SIMD architectural extension, and implemented a 2-way superscalar CPU that brought significant performance enhancements over the single-issue ARM11™. The Cortex-A9 extended the Cortex-A8 by bringing in MPCore capability for 1 to 4 CPU's with cache coherency managed efficiently by a snoop control unit. The Cortex-A9 also introduced performance enhancements inside the core that brought a 20-30% performance increase over Cortex-A8 for a single core.



Cortex-A7 makes use of a simple 8-stage in-order pipeline, extended to include dual-issue capability on a reduced range of data-processing and branch instructions. Increased dual-issuing coupled with other microarchitectural improvements allow the Cortex-A7 to reach very good levels of performance with very low power consumption.



Other performance enhancing features include an integrated L2 cache, which reduces latency to L2 memory and external memory. The integrated L2 cache simplifies OS support as it uses system mapped registers and can be managed using CP15 operations rather than the memory mapped registers needed for an external L2 cache. Integrating the L2 cache controller also reduces the amount of area consumed by an external controller and enables a tighter integration of the controller with internal bus structures.

The L2 cache controller itself was designed with low power in mind. The mechanism for looking up tags in the cache RAM includes consecutive tag followed by data lookup; similarly, the associativity is fixed at 8-way

to balance performance against lookup energy. External requests are triggered on an L2 miss, rather than on speculative requests, to reduce energy.

There are branch prediction improvements as well: the branch target instruction cache (BTIC) caches fetches after a direct branch and hides the branch shadow on tight loops.

There are several improvements in memory system performance. The Load-Store path has been increased to 64-bits from the 32-bit path in the Cortex-A5. The external bus structure has been upgraded to 128-bit AMBA4 to improve bandwidth and introduce support for coherency extension beyond the 1-4 SMP cluster using AMBA 4 ACE.

Energy Efficiency Features of the Microarchitecture

There are several features of the L1 Memory system which reduce the power consumption of the CPU or the system. The merging Store-buffer after the write stage reduces data cache lookups. The 2-way set associative instruction cache trades off the slightly improved hit rate of a 4-way set associative cache for the reduced power on each lookup.

Memory System Tuned to Minimize memory latency

There are several performance optimizing features in the memory system. The address generation unit is shifted one stage back in the pipeline to enable a single cycle load-use penalty. The design team increased TLB size to 256 entries, up from 128 entries for the Cortex-A5 and Cortex-A9; this reduces page walks saving power and significantly improves performance for large workloads like web browsing with large data sets that span a large number of pages. Also, page table entries can be cached in L1, improving the speed of page table walks on TLB misses. The bus interface unit has support for multiple outstanding read and write transactions. Finally, the physically indexed caches enable efficient OS Context switching.

Cortex-A7 performance

Implementation

The Cortex-A7 has been designed to enable high speed implementation. In the latest process nodes, the Cortex-A7 has been tested to 1GHz speeds for typical silicon at a conservative measurement corner and with conservative design margins. The implementation trials used 12-Track libraries, fast cache instance RAMs, and only nominal-Vt cells. A performance optimized implementation takes up just 0.45mm² for a single core, configured with FPU & NEON and 32K L1 caches, and consumes static and dynamic power similar to the highly efficient Cortex-A5.

Target implementations of Cortex-A7 in production SoCs are expected to be in 28nm. Top end frequencies above 1GHz are possible in that node with the use of LVt cells or voltage overdrive.

Software Benchmarks

The performance of Cortex-A7 on a range of benchmarks is 15%~20% higher than Cortex-A5. It trails behind Cortex-A8 slightly on integer workloads where data and code are L1 cache resident, but is faster at floating point math and can also outperform the larger Cortex-A8 CPU on typical modern workloads that have

complicated branch and TLB behavior, due to the memory system optimizations that were included in the Cortex-A7 design.

Large workloads like web browsing or compute-intensive Apps do stress the memory system of a processor, and on these types of workloads the Cortex-A7 performance can be more than 20% faster than Cortex-A5 and can actually outperform the Cortex-A8 at an equivalent clock rate. The Cortex-A8, with its full dual-issue superscalar design, outperforms Cortex-A7 as expected on integer benchmarks that have low cache miss rates and TLB miss rates, but on complex workloads the memory system improvements in Cortex-A7 have enabled the simpler processor to outperform the more complex superscalar Cortex-A8.

Cortex-A7 Target Markets

The target markets for the Cortex-A7 processor include two distinct segments of the mobile phone market. At the high end, smart phones and tablets are increasingly demanding ever high levels of performance with the same or longer battery life. The Cortex-A7 can be combined with the Cortex-A15 processor in a coherent pairing of the larger and smaller core, dynamically migrating the context to the smaller or bigger core based on instantaneous performance requirements. This approach is a new innovation from ARM called big.LITTLE processing, which will be introduced briefly later in this paper, and described in much more detail in separate papers specifically addressing that topic. The big.LITTLE processing enables the peak performance of the Cortex-A15, with an average power consumption profile closer to that of the small and very power efficient Cortex-A7.



The second segment of the mobile market where Cortex-A7 has a unique value is low-cost mass market smartphones. A dual or quad-core Cortex-A7 design can be implemented in a very low-cost SoC, bringing performance comparable to the Cortex-A8 or Cortex-A9 processor that power today's mainstream and high end smartphones and tablets. The small area and power of the Cortex-A7 will enable the designs of 2013 to offer the performance of today's mainstream and high end mobile devices in entry level devices.

Other markets can also take advantage of the Cortex-A7 processor, including I/O processors in enterprise applications, where the processor mainly watches over high speed data traffic which doesn't impose high per-thread performance requirements.

Other applications that can benefit from Cortex-A7 include offload processors for handling tasks like mobile audio, and SoCs for televisions, Set-top boxes, and residential gateways. Low cost general microprocessor applications can also take advantage of the low power Cortex-A7, including smart power meters, storage devices, and digital cameras to name a few.

Cortex-A7 in Low-Cost Smartphones

Market Requirements

Low-cost OEM handsets require several device features to bring a competitive offering to market, including support for the full Android™ OS with all ARMv7 optimizations at low cost, access to full range of apps in Google Marketplace, full web browser, Adobe® Flash® Player 10.x support, and full HTML 5 support. On the graphics side, low-cost handsets will require a 3D user interface UI accelerated with OpenGL® ES 2.0. All of this will need to be delivered at a price point that will enable sub\$100 unsubsidized pricing.

The market opportunity for these low-cost handsets is large and growing, as users in emerging markets will use smartphone as a primary connectivity device, bypassing the PC. The opportunity could run as high as a billion smartphones in 2016.

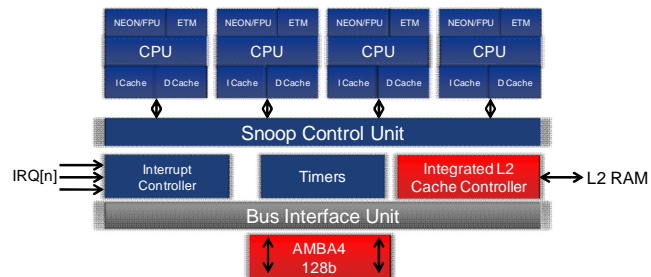
The performance requirements for low-cost handsets have followed a trend whereby the performance of mainstream and high end phones in a given year will become standard in low-cost phones 2 years later. This trend is expected to continue, which would mean that 2013 entry level smartphones will be expected to achieve the performance of 2011 mainstream smartphones, which today are based on Cortex-A8 and Cortex-A9 application processors.

Power/Performance

The Cortex-A7 can deliver similar performance in 28nm as current high-end smartphone SoCs, and significantly better performance than Cortex-A8 in the earlier 40 and 45nm geometries. While it is possible to implement Cortex-A9 or Cortex-A8 in 28nm, for example, it will be more likely for partners to implement the latest cores like Cortex-A7 and Cortex-A15 in 28nm.

Power/Performance of Multicore Cortex-A7 SoCs

The Cortex-A7 improves on the MP model in Cortex-A9 and Cortex-A5 based on learning from 3 generations of multicore designs at ARM. In particular, the Cortex-A7 incorporates bandwidth optimizations such as 128-bit wide data read buses, 256-bit wide data write buses and 256-bit wide data snoop buses. The external interface to the SoC is also revised to the 128-bit AMBA4 master port, which helps multicore performance by increasing the bandwidth delivered to the coherent cores in the SMP cluster.

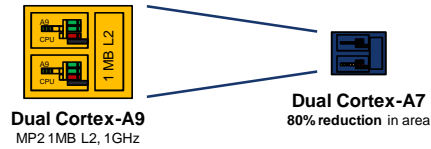


Area Diagrams

In addition to reducing power, Cortex-A7 significantly reduces die area for typical SoC implementations. A comparable dual-core implementation of Cortex-A7 in 28nm will take up just 20% of the die area of a Cortex-

A9 dual-core implementation in 40nm. The area savings can translate into lower cost SoCs targeting markets like mainstream and entry level smartphones.

A quad-core implementation for Cortex-A7 can potentially bring higher performance than current dual-core Cortex-A9 SoCs for software with moderate levels of parallelism, and in 28nm the area of the quad core Cortex-A7 can still come in much lower than current dual-core Cortex-A9 implementations. This area savings can translate into significant cost savings that enable quad-core CPUs in mainstream and entry level mobile and consumer products.



Cortex-A7 in High-end Smartphones, Tablets, and Beyond

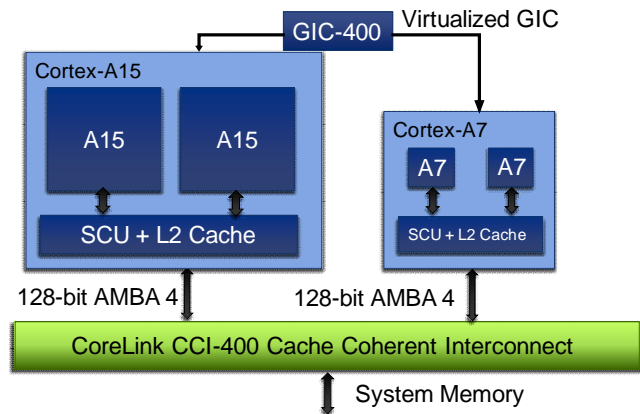
Market requirements for high-end mobile

High-end smartphones require high performance applications processors and graphics processors, but instantaneous performance requirements are highly elastic. During web browsing, for example, peak performance is required when pages are first rendered, but much lower levels of processor performance are required when reading or scrolling down a page. Similarly, applications have varying levels of performance requirements, typically requiring very high performance during launch, and low to moderate levels of required performance during at least some portion of runtime. For voice calls, the level of performance required by the applications processor is quite low, even on a high-end smartphone.

Given the wide range of required performance, it would be ideal if the phone could use a very power efficient CPU some of the time, and migrate the context to a high performance CPU at other times. ARM has been researching this idea for several years, and has specifically designed the Cortex-A7 CPU not only to ideally fit all but the high-end performance requirements of a high-end smartphone, but also to be able to connect tightly with the larger and higher performance Cortex-A15 CPU in a coherent system. When connected together through AMBA Coherency Extension (ACE) interface a Cortex-A15 CPU cluster can be connected with a cluster of Cortex-A7 CPUs in a processor complex with a single memory map, hardware managed cache coherency, and the ability to run workloads on the large CPU cluster or small CPU cluster depending on instantaneous performance requirements. This concept created by ARM is called big.LITTLE processing.

big.LITTLE Processing

Big.LITTLE refers to the coherent combination of High Performance and Power Efficient ARM CPUs



Copyright © 2011 ARM Limited. All rights reserved.

The ARM logo is a registered trademark of ARM Ltd. All other trademarks are the property of their respective owners and are acknowledged

A platform that contains both Cortex-A15 (big) and Cortex-A7 (LITTLE) can execute across a wider performance range with better energy efficiency than a single processor. Hardware coherency between Cortex-A15 and Cortex-A7 enables distinct big.LITTLE use models, either migrating context between the big and little clusters, or OS aware thread allocation to the appropriately sized CPU or CPUs. The CCI-400 cache coherent interconnect enables an extremely fast context migration between the big and little CPU clusters. Finally, software views the big and LITTLECPU clusters identically, and transitions are managed automatically by OS power management or directly by the OS. The Net result of big.LITTLE power management is a platform with the peak performance of the Cortex-A15, and average power consumption closer to the Cortex-A7. This enables significantly higher performance at lower power than today's high-end smartphones. The concept of big.LITTLE processing is only briefly introduced here; a more complete description of the hardware, software, and system implementation of big.LITTLE processing is covered in other TechCon presentations.

Conclusion

The Cortex-A7 CPU enables the performance of 2011 mainstream smartphones in entry-level smartphones and tablets of 2013, through enhancements to the microarchitecture, memory interface improvements, and innovative power efficient processor design. Large volume low-cost smartphones will take advantage of the Cortex-A7 CPU's low power and efficient performance.

In addition, the Cortex-A7 CPU enables big.LITTLE processing, a breakthrough innovation from ARM, delivering the peak performance of Cortex-A15 within a low average power budget driven by the Cortex-A7. Big.LITTLE processing will enable high-end smartphones tablets in 2013 with lower power consumption than high-end smartphones of today.

Finally, full architectural compatibility with the high end Cortex-A15 in the small Cortex-A7 power and area footprint will enable applications we haven't thought of yet.

About the author

Brian Jeff joined ARM in 2009 and currently is a CPU Product Manager with responsibilities including the Cortex-A5 and Cortex-A7 processors as well as next generation Cortex-A class CPU cores. Previous roles within ARM include CPU benchmarking and marketing. Prior to joining ARM, Brian held product management, engineering, and technical sales roles at Texas Instruments and Freescale Semiconductor. He holds a BSEE from Virginia Tech and an MBA from the University of Texas at Austin.