big.LITTLE technology moves towards fully heterogeneous Global Task Scheduling

Brian Jeff
Sr. Product Manager, Mainstream Mobile and High Efficiency Applications Processors
Agenda

- big.LITTLE™ Overview
- big.LITTLE Software
- Measurement of Key Benefits
  - Peak performance and responsiveness in a given power budget
  - Maximum performance within thermal limits of SoC
  - Improved power-efficiency of processing system
- Building a big.LITTLE System
- Demo
What is big.LITTLE Processing?

Power and energy optimization technology
Combines performance- and efficiency-tuned processors
Transparent to application software, shipping today

“Highest Performance Threads”
“All but the highest performance threads”

Increased Performance
Cortex-A9 smartphone | big.LITTLE

Significant Power Saving
Cortex-A9 smartphone | big.LITTLE
(selected use cases)
Coherency allows two CPU clusters to appear as one SMP cluster to the OS

2013: Cortex-A15 + Cortex-A7 in symmetric topologies

2014: Cortex-A57 + Cortex-A53, and others in symmetric and asymmetric topologies
Agenda

- big.LITTLE™ Overview
- big.LITTLE Software
- Measurement of Key Benefits
  - Peak performance and responsiveness in a given power budget
  - Maximum performance within thermal limits of SoC
  - Improved power-efficiency of processing system
- Building a big.LITTLE System
- Demo
big.LITTLE Software Evolution

Cluster Migration

CPU Migration

Global Task Scheduling

big.LITTLE MP

Improving Performance and Efficiency

2012

H1 2013

H2 2013
Global Task Scheduling

1. System starts
   *Task fill up the system*

2. Demanding Tasks detected
   - Based on amount of time a thread is ready to run (*run queue residency*)
   *Moved to a big CPU*

3. Task no longer demanding
   *Moved to a LITTLE CPU*

4. Global load balancer consolidates the workload
Agenda

- big.LITTLE™ Overview
- big.LITTLE Software
- Measurement of Key Benefits
  - Peak performance and responsiveness in a given power budget
  - Maximum performance within thermal limits of SoC
  - Improved power-efficiency of processing system
- Building a big.LITTLE System
- Demo
big.LITTLE Operation

big.LITTLE delivers optimum performance and efficiency in all use cases

1 - Maximum Responsiveness for high-intensity workload
2 - Sustained maximum interactive performance
3 - Long-use low-intensity workload
Responsive Peak Performance

- Peak performance bursts above typical power envelope of the SoC
- Delivering enhanced user experience at a touch
Web Browsing + Audio uses big cores for bursts of performance

LITTLE cores run background tasks, audio, and browser tasks after the main screen render completes

Partner Platform, 4 A15 cores, 4 A7 cores
Bursts of Performance

- Moderately intensive game, it can run predominantly on Cortex-A7 processor
- Cortex-A15 used for short bursts of performance

Partner Platform, 4 A15 cores, 4 A7 cores
Maximum Sustained Performance

- Maximise performance within the sustainable power envelope
- Efficiency of big.LITTLE enables increased performance in SoC
Performance Demands Increasing

- Mobile device performance advancing at ever faster pace
Increased Performance for Threaded Software

Well threaded games can make use of big and LITTLE processors extensively.

Partner Platform, 4 A15 cores, 4 A7 cores
Maximizing Sustained Performance

- **big.LITTLE** enables maximum performance under the thermal envelope for sustained gaming

**Power**

- Castle Master
- Real Racing 3

- **big.LITTLE**, Rest of SoC
- **big.LITTLE**, CPU
- A15 Only, Rest of SoC
- A15 Only, CPU

**Partner Platform, 4 A15 cores, 4 A7 cores**
Maximizing Sustained Performance

Continuous play on advanced console-quality gaming challenges thermal limits.

CPU + GPU Utilization

- **Cortex-A15**
  - Castle Master: 76%
  - Real Racing 3: 75%
- **Mali T628**

Power distribution:
- **CPU**
  - Castle Master: 100%
  - Real Racing 3: 0%
- **Rest of SoC**
Delivering Better User Experiences

big.LITTLE reduces sustained SoC power below the thermal limit

- Lower power for potentially longer playing time
- More power budget for GPU and visuals

CPU + GPU Utilization

ARM

Mali T628

Cortex-A15 44%

Cortex-A17 41%

Cortex-A17 43%

Cortex-A17 42%

Castle Master

Real Racing 3

thermal limit
Increased Performance for Threaded Software

All big and LITTLE cores running simultaneously
Capacity Advantage for workloads with >4 threads

Performance Increase vs. Cortex-A15 Standalone

CF-Bench  AndEBench Native  Antutu v4  Geekbench  Quadrant

<=4 threads: No slowdown

Partner Platform, 4 A15 cores, 4 A7 cores
Most Efficient Processing

- Maximise energy efficiency with LITTLE processor for typical workloads
- Extended device lifetime with the most efficient A-Class processor
Casual Games reside entirely on Cortex®-A7 processors, except for drawing of initial screens and App launch.
Low Intensity Use Cases

-76% -76% -73% -42% -73%

Significant power reduction at CPU level

Partner Platform, 4 A15 cores, 4 A7 cores
Low Intensity Use Cases

Savings are still significant at SoC level

Partner Platform, 4 A15 cores, 4 A7 cores
Agenda

- big.LITTLE™ Overview
- big.LITTLE Software
- Measurement of Key Benefits
  - Peak performance and responsiveness in a given power budget
  - Maximum performance within thermal limits of SoC
  - Improved power-efficiency of processing system
- Building a big.LITTLE System
- Demo
Integrating an Efficient System

CoreLink™ CCI-400
Cache Coherent Interconnect

- Display and Video Sub-system
- GIC-400 Interrupt Control
- Cortex-A15
  - L2 cache
- Cortex-A7
  - L2 Cache
- Mali- T628 GPU
  - Shader
  - Shader
  - Shader
- ADB-400
- MMU-400
- DMC
- DDR/LPDDR
- To Peripheral Interconnect

- CPU Cluster and IO Coherent GPU
- ACE Coherency enables big.LITTLE and GPU compute
- Efficient Voltage scaling for power management
- Common memory view for all SoC components
- Path to memory with Trustzone hardware security
Global Task Scheduling

- ARM implementation: “big.LITTLE MP”
  - Mobile optimised
  - Hosted openly in the Linaro Stable Kernel (LSK) and as a kernel patch set

Power Scheduler:
- Longer term effort to put global task task scheduling in Linux upstream kernel
- Still under development
  - Follow the discussion: https://lkml.org/lkml/2013/7/9/314

CPU Migration

- Linaro “In-Kernel Switcher”
  - Available Now: http://www.linaro.org/linaro-blog/2013/05/02/the-linaro-iks-code-now-publicly-available
Scalable Technologies for SoCs

ARM big.LITTLE
Processor Technology

Mali-T678
Mali-T628
Mali-T624
Mali-T622
Mali-V500
Mali-450
Mali-400
Mali-300

ARM Mali
Visual Computing

ARM Cortex
Processor Technology

Cortex-A57
Cortex-A53
Cortex-A15
Cortex-A9
Cortex-A7
Cortex-A5
Cortex-R7
Cortex-R5
Cortex-R4
Cortex-M4
Cortex-M3
Cortex-M0

ARM CoreLink
Processor System IP

Coherent System IP

ARM Artisan
Physical IP

Physical IP
big.LITTLE SoCs

- Initial big.LITTLE SoCs now in silicon
- 10+ licensees in various stages of development
- Rapid advancement in software and system optimization
Demo

ARM® big.LITTLE™
Processor Technology

ARM® M8L™
Visual Computing

ARM® CORTEX®
Processor Technology
big

LITTLE
Summary

big.LITTLE delivering on key benefits
- Significantly higher peak performance within a tighter power budget
- Improved SoC performance under thermal constraints
- Power savings across a range of workloads and use scenarios
- Performance and efficiency increase on threaded workloads

big.LITTLE technology is shipping in products today
- Wider range of differentiated solutions from silicon partners

Devices transitioning to more advanced big.LITTLE
- Demonstrated additional benefit from global task scheduling
- Global task scheduling available for production platforms
Brian Jeff
Sr. Product Manager, Mainstream Mobile and High Efficiency Applications Processors
brian.jeff@arm.com