Performance, Energy Characterizations and Architectural Implications of an Emerging Mobile Platform Benchmark Suite – MobileBench

Dhinakaran Pandiyan, Shin-Ying Lee, and Carole-Jean Wu

Arizona State University

September 24, 2013
IEEE International Symposium on Workload Characterization
Smartphones today...

- 2GB RAM, 8 cores, full HD (1920 x 1080) AMOLED display, hardware accelerators
- User experience is a primary selling point
- Space and weight constraints limit battery capacity
- Need to reduce power consumption wherever possible
- Can consume power even as high as 3 Watts during regular use

Image of a Qualcomm Snapdragon 800 [http://www.qualcomm.com/snapdragon/processors/800]
Problem statement

- Given multiple, heterogeneous SoC components, what is the major energy consumer on smartphones?

- What is the performance impact of various microarchitecture components given the application usage behavior?
Presentation Outline

- Motivation
- Introduction to an emerging smartphone benchmark suite - MobileBench
- Real system and simulation infrastructure setup
- Characterization of energy consumption
- Performance characterization of microarchitecture features
- Summary
- Conclusion
MobileBench
Benchmark suite of Android applications

- **General Web Browsing (GWB)**
  Offline, automated benchmark

- **Realistic General Web browsing (R-GWB)**
  Represents realistic browsing behavior (scroll-up, horizontal scroll and random delays)

- **Educational web browsing (EWB)**
  Models reading documents along with browsing

- **Video playback**
  1280×720 HD playback for 30 minutes

- **Photo Viewing**
  High resolution (up to 4912*3264) images

---
Real System and Simulation infrastructure setup

- Real device measurements
  - Samsung galaxy S3 I9300
  - Instrumented Android framework for collecting energy numbers
  - Validate with the Watts Up energy meter

- Full-system simulation
  - ARMv7
  - Android ICS operating system
  - Modeling an ARM Cortex A9
    - L1 I cache – 64KB, 4-way set associative, private
    - L1 D cache – 64KB, 8-way set associative, private
    - L2 unified cache – 1 MB, 16-way set associative, private, inclusive
    - MSHR – holding 32 outstanding misses
Hardware component energy characterization

- Developed EnergyUsageCollector by instrumenting the Android framework
  - *power-profile.xml* provides current consumption values for various components
  - Validation using a Watt meter
  - Energy = Voltage * power_profile current constant * $\Delta t$

![Bar chart showing current consumption for various components]
Software component energy characterization

Energy Profile Breakdown (%)

- GWB
- RealisticGWB
- EWB-Blackboard
- PhotoView
- VideoPlayback
- Average

- other
- main application
- libraries
- kernel
- wifi
- screen
Software component energy characterization

Application cores show significant power consumption
Presentation Outline

- Motivation
- Introduction to an emerging smartphone benchmark suite - MobileBench
- Real system and simulation infrastructure setup
- Characterization of energy consumption
- Performance characterization of microarchitecture features
- Summary
- Conclusion
Performance Characterization

• Instruction distribution
  • 62.08% integer, 0.75% floating point, 8.94% branch, 26.33% memory and 1.89%

• Use MobileBench to study micro-architectural features
  • How sophisticated should the branch predictor be for smart phone applications?
  • Is a large TLB needed to accelerate virtual-to-physical address translation?
  • What would be the appropriate cache hierarchy for a smartphone?
  • How can we improve the cache utilization?
  • Is a hardware prefetcher needed on smartphones?
  • Is the memory B/W a performance bottleneck?
MobileBench apps show distinct TLB requirements

- GWB and R-GWB benefit the most from a bigger ITLB – larger instruction address space
- Video playback benefits the most from a bigger DTLB – larger data working set
L2 misses are a significant portion of memory access time

- Low L2 utilization - 41% L2 miss access time
- Viewing high resolution photos needs large BW
- Available BW - 12800 MB/s on Exynos 4 Quad, 8528 MB/s on A6
MobileBench apps have an active working set larger than the L2 cache.
MobileBench exhibits heterogeneous cache reuse pattern

- Only 20% of memory accesses have reuse distance less than the L2 cache associativity
- MobileBench exhibits streaming and mixed access patterns
- Fast-changing access patterns

L2 cache needs better cache management!
L2 cache utilization can be improved with advanced cache management

- SHiP* and DRRIP** improve cache utilization by handling streaming and mixed access patterns
- EWB shows the biggest improvement

![Graph showing L2 Cache Miss Rate Normalized to LRU]


MobileBench is prefetching-friendly

- A simple stride prefetcher improves performance across all apps.
Tournament predictor outperforms local predictor for EWB and R-GWB

- More complicated browsing behavior in EWB and R-GWB

![Branch Predictor Accuracy Graph](image)

![IPC Normalized to Local Br. Predictor](image)
Summary

- Using a sophisticated branch predictor can improve branch prediction accuracy but this does not translate into observable performance gain.
- Large TLB can improve performance by an avg. 14%.
- L2 cache miss rate improves by 4.2% on average when an exclusive cache hierarchy is used.
- Effective cache management policies can improve L2 cache utilization as much as 29.3%.
- Using a stride prefetcher can increase performance of MobileBench by 14% on average.
- B/W requirements are well under the capacity.
Conclusion

• We provide MobileBench, a collection of smartphone applications that represents applications commonly run on mobile platforms

• We present detailed performance and energy characterizations for MobileBench

• With the architectural insights provided in the paper, we hope to inspire innovative designs that consume less power and offer high performance.
Performance, Energy Characterizations and Architectural Implications of an Emerging Mobile Platform Benchmark Suite – MobileBench

Thank you!

https://lab.engineering.asu.edu/mobilebench/