Power Profiling on Mobile Platforms

Andrea de Oliveira[1,3], Victor Sun[2,3]

Supervised by Onur Sahin[3], Dr. Ata Turk[3], Prof. Ayse Coskun[3]

Crossroads School for Arts and Sciences, 1714 21st St, Santa Monica, CA 90404[1];
Monta Vista High School, 21840 McClellan Rd, Cupertino, CA 95014[2];
Boston University, ECE Department, 8 St Mary's St, Boston, MA 02215[3]

BOSTON UNIVERSITY

PEACLAB

Abstract

High-performance mobile devices often operate at a high energy cost, and, therefore, have limited battery lives. Power profiling is the first step towards increasing energy efficiency. Through collecting temperature, FPS (frames per second), CPU utilization, and CPU frequency data, we can identify sources of energy inefficiency and gain insights into the potential power/performance tradeoffs. Such performance and power profiling would allow for designing energy efficient dynamic power management techniques, which could adjust CPU frequencies or even shut off certain CPU cores altogether to find which settings maximize energy efficiency without sacrificing user performance (in FPS). We ran our tests on two platforms: the ODROID-XU3 board, which includes ARM's big.LITTLE heterogeneous processor technology, and the Inforce 6410 Single Board Computer, which uses a high-performance homogeneous multi-core CPU. Heterogeneous CPU architectures are gaining popularity in mobile devices (i.e., the Samsung Exynos 5 and Qualcomm Snapdragon 810) for their ability to balance both performance and energy efficiency, while homogeneous CPU architectures are currently implemented in most contemporary mobile devices (i.e., Intel's i5 and i7 processors). Our tests required familiarity with the Linux operating system, the android development environment, Android Studio, and bash scripting. To display data, we built an Android application that gives the user the option of viewing temperature, CPU frequency, CPU utilization, or FPS data, runs bash scripts to collect the data, and displays the data. The application provides the user with a central platform through which he or she can view various performance and energy related metrics (frequency, temperature etc.).

Introduction

*Performance often takes precedence over energy efficiency, leaving most of today's mobile devices with limited battery lives.

*At an architectural level, developers have addressed this by incorporating heterogeneous CPU architectures into their mobile devices.

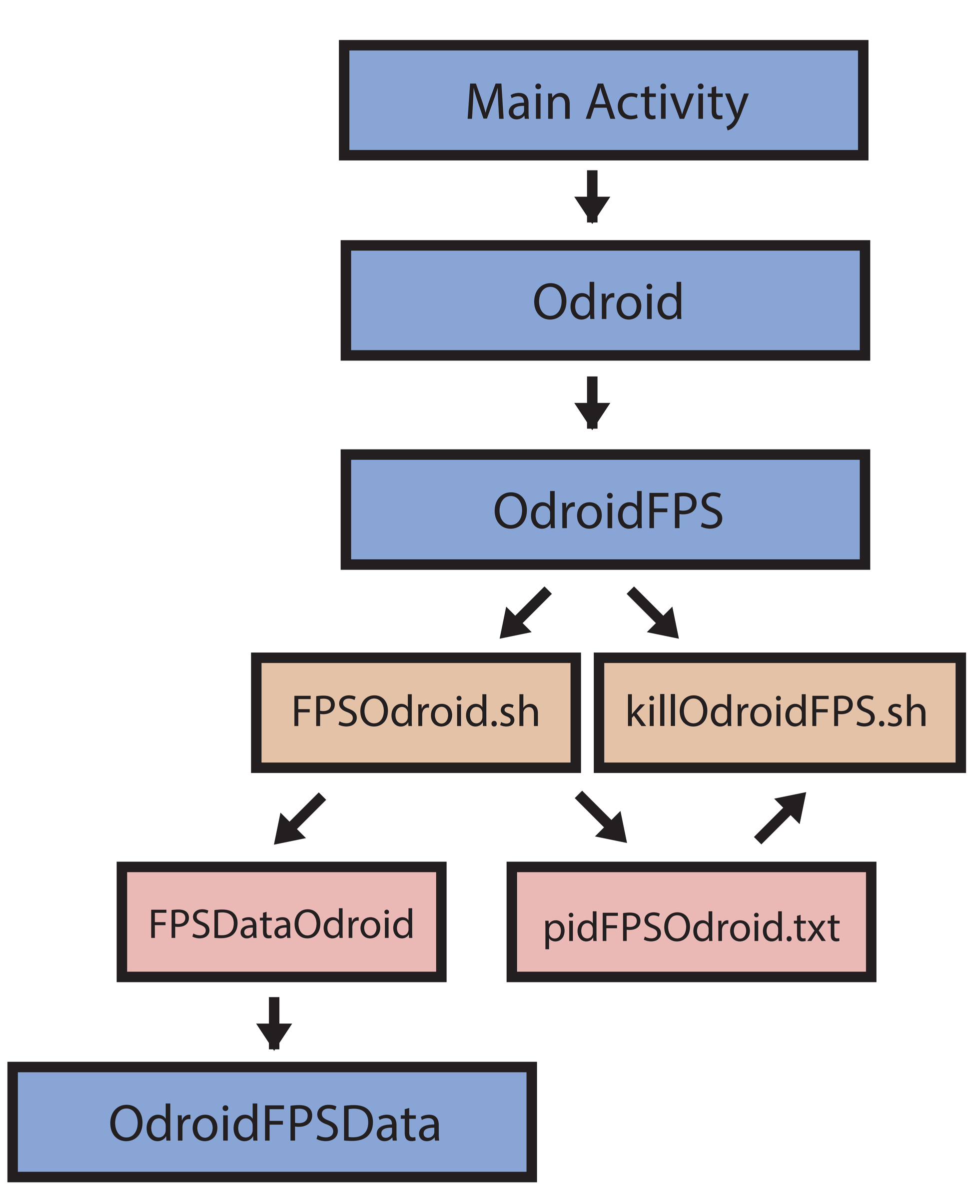
-Heterogeneous CPU architectures comprise high-performance cores, energy-efficient cores, and scheduling mechanisms to yield the optimal power/performance tradeoff.

-Homogeneous CPU architectures, consisting of identical cores, are currently implemented in most of today's mobile devices.

*At an operating system level, power profiling is the starting point in maximizing energy efficiency. Through power profiling, we can better understand and maximize efficiency on both more traditional and state-of-the-art CPU architectures by developing energy efficient dynamic power management techniques.

*In this work, we built an Android application that allows users to monitor performance/power related metrics using a lightweight user-level app.

Methodology



*Software and Tools:
-Java: For coding application behavior
-XML: For designing application layout
-Android Studio: Compiling Java and XML files along with drawable resources

*Devices supported
-Odroid-XU3 board (Figure 1): ARM's big.LITTLE heterogeneous processor technology
-Inforce 6410 Single Board Computer (Figure 2): high-performance homogeneous multi-core CPU

*Profiling Data:
-Time: /proc/uptime
-FPS: Performance metric measured through SurfaceFlinger
-CPU Utilization: Read through system files
-Temperature:
-Odroid: temp file under TMU directory
-Inforce: temp file under each of 13 thermal zones
-CPU Frequency: scaling_cur_freq files under the cpufreq subdirectory of each of 4 CPUs

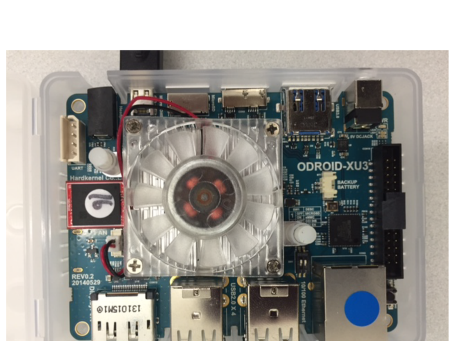


Figure 1. The ODROID-XU3 Board

Figure 2. The Inforce 6410 Board

Android Application Developement

Figure 3. Android Application components

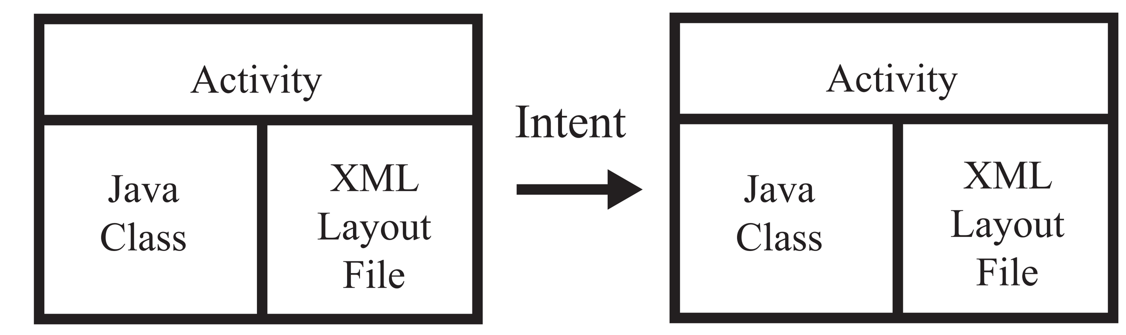


Figure 4. Intents allow for transfer between activities, comprised of XML and Java files

Results

Main Activity

Target Board Selected

Odroid Activity

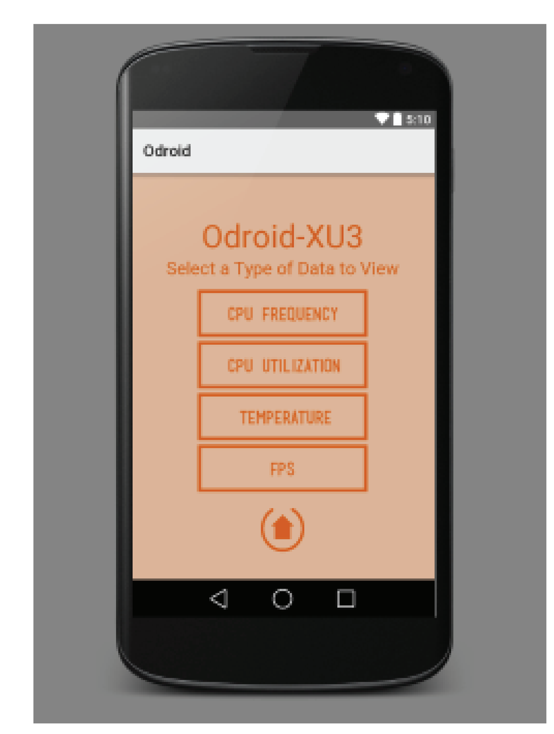
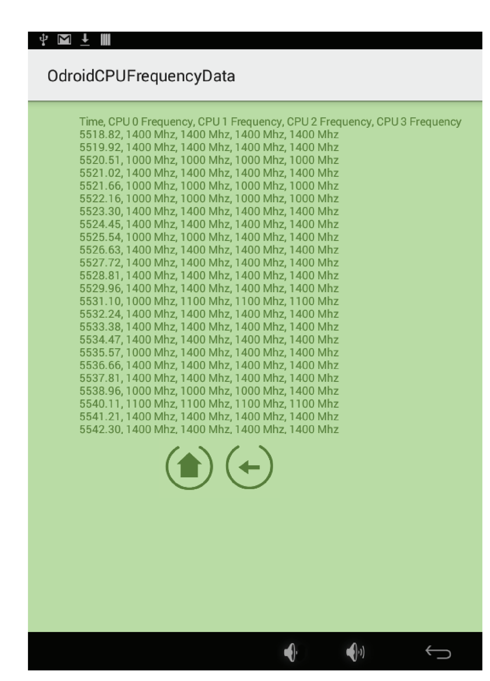
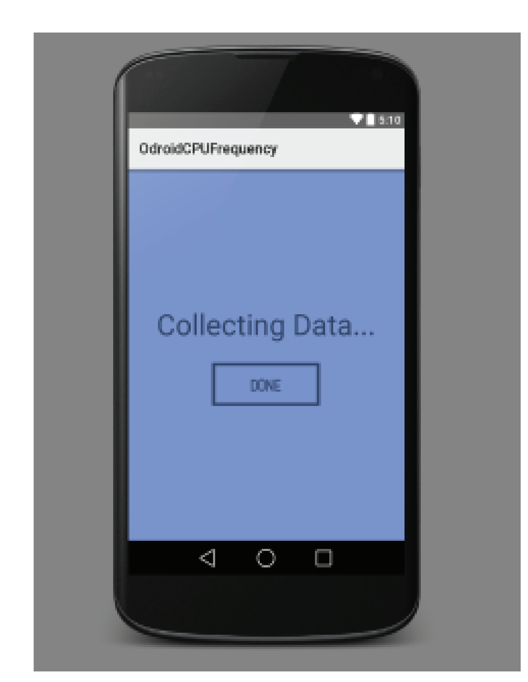


Figure 5. XML Layout of the Main Activity

Figure 6. XML Layout of the Odroid Activity



Launch an application to profile



Data Type Selected

Figure 8. OdroidCPUFreqencyData on the Odroid board

Figure 7. XML Layout of the OdroidCPUFrequency Activity

Conclusion

By providing easy access to FPS, temperature, CPU frequency, and CPU utilization data, our power profiling application will, ideally, facilitate data collection, the next stage in optimizing energy efficiency. With our app, researchers or application developers can run tests on a set of like applications to identify a base-line frequency (lowering frequency lowers temperature which lowers power consumption) that keeps the apps running above the generally accepter lower performance bound of 30 FPS. They can then identify which actions (i.e., starting an application) consume high amounts of energy and take these actions into consideration when developing power management mechanisms. The ultimate goal is to encourage more inclusive power management that will prompt the hardware to consume less energy.

References

[1] *Application Fundamentals*. (n.d.). Retrieved August 5, 2015, from http://developer.android.com/guide/components/fundamentals.html

[2] *big.LITTLE Technology: The Future of Mobile*. (2013). Retrieved August 3, 2015, from http://www.arm.com/files/pdf/big_LITTLE_Technology_the_Futue_of_Mobile.pdf

[3] *Example on how to access sensors on XU+E* [Msg 1]. mdrjr. Message posted to http://forum.odroid.com/viewtopic.php?f=61&t=2060 (2013, February 22).