Approximating Outside the Processor

Phillip Stanley-Marbell Martin Rinard

MIT

psm@mit.edu, rinard@csail.mit.edu

Abstract

Energy usage in many contemporary computing systems is constrained by factors other than computation. The major components of energy use include transferring data on-chip in high-performance systems, activating and accessing sensor and display interfaces in wearable and mobile systems, and accessing and refreshing memories in all system types. The relative proportion of energy usage due to factors outside computation will only grow in the future. This is because many of the factors in question are functions of system-level interconnects and packaging, which do not benefit directly from semiconductor technology scaling.

There is an opportunity to improve the energy-efficiency of computing systems by exploiting their tolerance to inaccuracy, imprecision, and unreliability. But, seizing this opportunity requires *approximating outside the processor*, applying approximation to sensors and output devices, instead of only focusing on the operations in computations. This observation is particularly relevant for embedded, wearable, and mobile systems, which form the dominant (and growing) majority of the world's computing devices.

1. Introduction

Computation is not the dominant source of *instantaneous power dissipation* in many wearable and mobile systems. These systems are often organized around sensors, whose power dissipation when active can be larger than that of many of the embedded processors (microcontrollers) with which they are typically paired. The sensors are typically sampled whenever computation is active and as a result the fraction of overall *energy usage over time* attributable to computation, relative to sensors, is also often small.

Figure 1 shows the average power dissipation when active, for a collection of components. The components include an implementation of the lowest-power variant of the ARM architecture currently available (Cortex-M0+[1] [7]), several state-of-the-art sensors [1, 14, 15, 22, 23] and a Bluetooth Low-Energy (Bluetooth LE) radio[2] [21]. In addition to computation and sensors, many wearable systems have organic light-emitting diode (OLED) displays, whose power

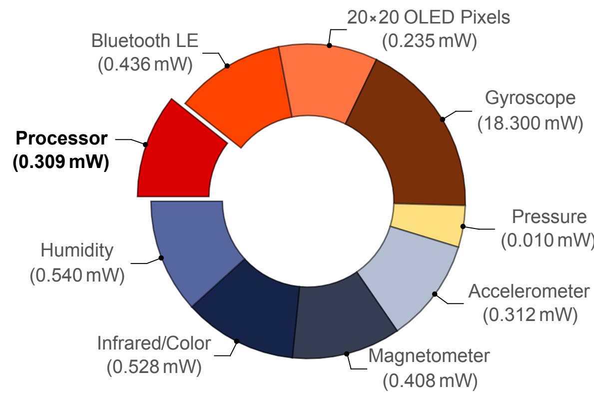

Figure 1. Sector plot of the power dissipation for several state-of-the-art system components typical of wearable and sensor-driven systems. The sectors are shown scaled logarithmically to simplify visualization of the large range of values: Clearly, not all systems will contain a gyroscope (which dominates the power breakdown in this collection of system components).

dissipation is proportional to the number of pixels which are lit and to their color (there is no backlight). We have therefore included, for reference, the power dissipation of a 20×20 pixel subset of an OLED display, based on measurements we performed on one such display panel [10].

From Figure 1, it is clear that the processor dissipates less power when active than almost all the other components. Since most wearable systems sample their sensors periodically, the energy usage over time is also likely to still be dominated by components other than the processor.

Despite these observations that only a small fraction of system power dissipation is attributable to computation, the majority of existing research aimed at exploiting tolerance of *inaccuracy* (distance from a ground truth), *imprecision* (repeatability over time), or *unreliability* (probability of outright failure) has focused on computation [2, 6, 11, 13]. The evidence suggests that the focus of these existing research efforts on computation will not yield the best system-level benefit in the important and growing class of wearable and sensor-driven systems.

However, are there tradeoffs between accuracy, precision, reliability, and power or energy usage in system components

[1] Running a `while(1)` loop from its on-chip SRAM at 2 MHz and 3.0 V.

[2] In *advertising/discoverable* mode.

other than processors? And, can these tradeoffs be exploited directly at the level of sensors and displays?

2. Approximate Inputs: Assumptions of Noise

The data that serve as input for many important computational problems in the real world, increasingly come from sensors of physical phenomena of various kinds. These sensors may range from accelerometers and gyroscopes in wearable and health-tracking systems, to continent-spanning radioastronomy telescopes.

Because there may be transient or persistent noise in sensor data, the computational problems which consume them, and hence the algorithms which embody these compute problems, can often operate on data of varying accuracy, precision, or reliability. Sensors however typically require different amounts of time and energy resources to generate data of different degrees of fidelity. When hardware and system software permit, tolerance of imprecision, inaccuracy, and unreliability can be exploited: The tolerance can be harnessed to achieve sensor activation and sensor data acquisition which uses less energy, which is faster, or which is cheaper to build.

Lax [18] is one example of a system that builds on these insights. Lax improves the energy-efficiency of sensor-driven systems by controlling the power supplies of sensors such as gyroscopes, so that they provide inaccurate, imprecise, or unreliable data, but consume significantly less power. Because of the empirically-observed variation in the type, frequency, and severity of sensor data errors with supply voltage, Lax uses descriptions of the amount of error that can be tolerated by applications to determine how much energy to save. These descriptions of tolerable error are provided in Lax's domain-specific language, but could in principle also be inferred by a compiler.

In addition to circuit techniques to reduce the power dissipation of sensor integrated circuit *operation* [18], it is possible to develop value encodings which reduce the power dissipated in *moving data* [17] between sensing and computing devices, at the cost of controlled data infidelity [19].

3. Approximate Outputs: Limits of Perception

Analogous to the pervasiveness of transient and persistent noise in real-world data sources, when the results of computation are consumed by the human aural or visual system, variations in accuracy, precision, or reliability may not always be perceptible. Such variations can be exploited directly, in the generation of audio or display of results, for lower-energy, faster, or cheaper output devices (e.g., displays). For example, for displays, a few research efforts have investigated exploiting the variability in human sensitivity across the color spectrum. This phenomenon has been exploited to reduce power dissipation in OLED displays [4, 5, 8, 9, 12, 20, 24] as well as in those traditional LCDs that have coarse-grained controllable backlighting [3]. Even when the results are consumed by non-human entities such as control systems, some amount of tolerance to imprecision, inaccuracy, and unreliability may still exist.

The interfaces for surfacing perceptual signals, such as displays and audio, contribute an increasing fraction of system energy usage in wearable and mobile systems. Because the phenomena underlying their operation (e.g., photon generation, mechanical displacement) are less amenable to improvements in transistor properties than computation is, their relative importance will likely grow in the future.

4. Theoretical Bounds Inform Practical Implementations

Computing systems can exploit tolerance to inaccuracy, imprecision, and unreliability in computational problems, sensor activation and access, I/O, and display interfaces, to the benefit of performance, energy, and cost. But, in tandem with developing algorithmic, systems software, and hardware techniques to exploit tolerance of imprecision, inaccuracy, and unreliability, it is prudent to explore upper bounds on the potential benefits thereof.

Bounds are important for several reasons. First, upper bounds on the achievable benefits to either performance or energy serve as a reference for the maximum time or energy overhead that a practical technique can incur if it is to attain a net improvement. Second, bounds often yield new or deeper insights into the problem under investigation and are thus interesting in their own right. For example, the investigation of encoding efficiency of digital number representations for approximate communication [16] has recently led to the development of practical approximate value encoders [19].

We must study bounds on the potential benefits of approximation applied to input data, output data destined for perception, communicated and stored data, computational steps in algorithms, and computational problems. For example, analogous to the Shannon efficiency bounds for forward-error-correcting codes that guarantee the correction of a fixed number of errors or erasures, what are the upper bounds on overheads for representing values in programs and microarchitectures when we are willing to tolerate inaccuracy, imprecision, or unreliability?

5. Summary

We should exploit tolerance to imprecision, inaccuracy, and unreliability in the subsystems which dominate performance and energy usage. For the growing majority of the world's computing devices, these energy-dominant subsystems are sensors, displays, on-chip and board-level interconnects, and volatile and non-volatile memories. The fraction of energy usage relative to computation attributable to these subsystems will likely only grow in the future, as advances in semiconductor technology benefit digital logic more than they benefit packaging and interconnects. We should think outside the box and approximate outside the processor.

References

[1] Bosch Sensortec. *BMX055 Small, Versatile 9-axis Sensor Module*, Data Sheet, November 2014.

[2] M. Carbin, S. Misailovic, and M. C. Rinard. Verifying quantitative reliability for programs that execute on unreliable hardware. In *Proceedings of the 2013 ACM SIGPLAN International Conference on Object Oriented Programming Systems Languages & Applications*, OOPSLA '13, pages 33–52, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-2374-1.

[3] J. Chuang, D. Weiskopf, and T. Möller. Energy aware color sets. *Computer Graphics Forum*, 28(2):203–211, 2009. ISSN 1467-8659.

[4] M. Dong and L. Zhong. Chameleon: A color-adaptive web browser for mobile oled displays. In *Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services*, MobiSys '11, pages 85–98, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0643-0.

[5] M. Dong, Y.-S. K. Choi, and L. Zhong. Power-saving color transformation of mobile graphical user interfaces on oled-based displays. In *Proceedings of the 2009 ACM/IEEE International Symposium on Low Power Electronics and Design*, ISLPED '09, pages 339–342, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-684-7.

[6] H. Esmaeilzadeh, A. Sampson, L. Ceze, and D. Burger. Architecture support for disciplined approximate programming. In *Proceedings of the Seventeenth International Conference on Architectural Support for Programming Languages and Operating Systems*, ASPLOS XVII, pages 301–312, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-0759-8.

[7] Freescale Semiconductor. *Kinetis KL03 32 KB Flash 48 MHz Cortex-M0+ Based Microcontroller*, Data Sheet, August 2014.

[8] T. Harter, S. Vroegindeweij, E. Geelhoed, M. Manahan, and P. Ranganathan. Energy-aware user interfaces: An evaluation of user acceptance. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '04, pages 199–206, New York, NY, USA, 2004. ACM. ISBN 1-58113-702-8.

[9] D. Li, A. H. Tran, and W. G. J. Halfond. Making web applications more energy efficient for oled smartphones. In *Proceedings of the 36th International Conference on Software Engineering*, ICSE 2014, pages 527–538, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2756-5.

[10] One Stop Displays. *OSD2828GDEDF11 OEL Display Panel*, Data Sheet, 2006.

[11] A. Sampson, W. Dietl, E. Fortuna, D. Gnanapragasam, L. Ceze, and D. Grossman. Enerj: Approximate data types for safe and general low-power computation. In *Proceedings of the 32Nd ACM SIGPLAN Conference on Programming Language Design and Implementation*, PLDI '11, pages 164–174, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0663-8.

[12] D. Shin, Y. Kim, N. Chang, and M. Pedram. Dynamic voltage scaling of oled displays. In *Proceedings of the 48th Design Automation Conference*, DAC '11, pages 53–58, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0636-2.

[13] S. Sidiroglou-Douskos, S. Misailovic, H. Hoffmann, and M. Rinard. Managing performance vs. accuracy trade-offs with loop perforation. In *Proceedings of the 19th ACM SIGSOFT Symposium and the 13th European Conference on Foundations of Software Engineering*, ESEC/FSE '11, pages 124–134, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0443-6.

[14] ST Microelectronics. *L3G4200D MEMS Motion Sensor: Ultra-stable Three-axis Digital Output Gyroscope*, Data Sheet, December 2010.

[15] ST Microelectronics. *LPS25H MEMS Pressure Sensor: 260–1260 hPa Absolute Digital Output Barometer*, Data Sheet, January 2014.

[16] P. Stanley-Marbell. Encoding efficiency of digital number representations under deviation constraints. In *IEEE Information Theory Workshop, ITW'09*, pages 203–207, Oct 2009.

[17] P. Stanley-Marbell and D. Marculescu. A programming model and language implementation for concurrent failure-prone hardware. In *Programming Models for Ubiquitous Parallelism, PMUP'06*, 2006.

[18] P. Stanley-Marbell and M. Rinard. Lax: Driver interfaces for approximate sensor device access. In *15th Workshop on Hot Topics in Operating Systems (HotOS XV)*, Kartause Ittingen, Switzerland, May 2015. USENIX Association.

[19] P. Stanley-Marbell and M. Rinard. Value-deviation-bounded serial data encoding for energy-efficient approximate communication. Technical Report MIT-CSAIL-TR-2015-022, MIT Computer Science and Artificial Intelligence Laboratory (CSAIL), June 2015.

[20] K. W. Tan, T. Okoshi, A. Misra, and R. K. Balan. Focus: A usable & effective approach to oled display power management. In *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '13, pages 573–582, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-1770-2.

[21] Texas Instruments. *CC256x Bluetooth® and Dual-Mode Controller*, Data Sheet, January 2014.

[22] Texas Instruments. *HDC1000 Low Power, High Accuracy Digital Humidity Sensor with Temperature Sensor*, Data Sheet, Nov 2014.

[23] Texas Instruments. *TMP006/B Infrared Thermopile Sensor in Chip-Scale Package*, Data Sheet, November 2014.

[24] M. Zhao, Y. Chen, X. Chen, and C. J. Xue. Online oled dynamic voltage scaling for video streaming applications on mobile devices. *SIGBED Rev.*, 10(2):18–18, July 2013. ISSN 1551-3688.