Course: Energy Efficiency in Embedded Systems – A System-Level Perspective for Computer Scientists

Ulf Kulau
dept. of Computer Engineering
TU Braunschweig
eMail: kulau@c3e.cs.tu-bs.de

***Abstract*—The improvement of the energy efficiency of embedded devices has always been an important research topic. Nowadays with more and more autonomous devices, e.g. IoT, this topic is yet popular as questions arise "how to power all these devices?" or "how to enable lifetimes of years or decades?".**

As theses challenges are not solely an issue of computer architects, this course specifically addresses computer scientists. While computer scientists would not be able to influence the basic architecture of an e.g. MCU, the knowledge on the basics, the architectures, mechanisms and software paradigms should sensitize students how to implement energy efficiency on system level.

***Index Terms*—Course, Computer Science, Energy Efficiency, Embedded Systems**

I. Introduction

Energy efficiency is a broad term and each branch of study has its own definition. For example electrical engineers might associate energy efficiency with a focus on energy conversion [1]–[3]. The topics in this area are mainly focused on the efficiency factor for power generation, energy transport or households (e.g. thermal insulation). In fact, not much that would attract students of computer science classes.

Of course, there is also a plethora of existing courses on energy efficiency in embedded systems that are slightly more related to our field of study [4], [5]. Each of them are indeed excellent courses, but they still provide only a narrow perspective on e.g. semiconductor technology or computer architecture [6], [7].

However, nowadays embedded systems are often ubiquitous and we have to broaden the topic as systems can not be considered as a single hardware device [8], [9]. For example ever growing Internet of Things (IoT) applications require deep knowledge on both, hard- and software level. In addition, the networked systems demand a great deal of students to establish energy efficiency on several levels. Many technology forecasts underline the importance of this topic and thus the educational need.

Hence, energy efficiency is a system issue. Low power has been the focus of hardware designers for decades and great advances have been made in low power designs, yet most computer scientists and, in particular, most software developers are blissfully unaware of the impact their decisions have on energy consumption:

> "If you want an ultimate low-power system, then you have to worry about energy usage at every level in the system design, and you have to get it right from top to bottom, because any level at which you get it wrong is going to lose you perhaps an order of magnitude in terms of power efficiency."[1]

II. Content and Structure of the Course

As mentioned above, this course addresses students of computer science classes. Although many of them attend a course related to computer engineering, the knowledge on basic electrical engineering, semiconductor technologies and computer architectures is limited. However, it is highly important to know the origin and background of power dissipation of the underlaying hardware to establish energy efficiency on system level [10]. Thus a broad spectrum from low level hardware fundamentals up to software mechanisms on user layer is covered.

Therefore the course introduces four levels that illustrate the form and content:

1) *Device Level:*
   Basics and the power dissipation of processing elements are discussed. Moreover, the characteristics of different transistors (dimension, technology, etc.) are analyzed.
2) *Circuit Level:*
   Here the impact of different circuit technologies is considered. For example the power dissipation of outdated technologies like the Register-Transistor-Logic is compared against CMOS. Of course, the power dissipation of CMOS is reviewed extensively.
3) *Module / MCU Level:*
   The first level where a user can interact with the hardware to achieve energy efficient applications. Mechanisms but also limitations of e.g. sleep states, frequency and voltage scaling are presented. Moreover, undervolting schemes are discussed and how they can help to increase the efficiency while ensuring a dependable execution.
4) *System Level:*
   General system design e.g. voltage conversion or the selection and dimension of devices are considered as well as use cases to establish energy efficiency on application

[1] ACM queue Interviews: Power Management, February 1, 2010, Volume 8, Issue 2. Conversation with Steve Furber. The designer of the ARM chip shares lessons on energy-efficient computing. https://queue.acm.org/detail.cfm?id=1716385

layer software. In addition the popular topics Energy Harvesting, Energy Neutral Computing and Transiently Powered Computing are part of this level.

The first part of the course is structured as usual. A weekly lecture provides the theoretical information while a weekly exercise provides additional information and some more insights. For the second part of the course each student has to prepare a short mini-presentation on a relevant publication in the field of energy efficiency in embedded systems.

A. Lecture

The lecture is divided into nine sections of about 90min each. In the following a brief description of the specific content is provided.

1) 00 Organization and Topics: The objectives of the course, general information and the time schedule are discussed. Moreover a basic motivation to emphasize the relevance of energy efficiency as well as related literature is provided. The goal is to encourage the students to attend this course.

2) 01 Motivation and Power Dissipation: Initially, the motivation is focused on classical data centers and show the recent problem of waste heat. With regard to small embedded systems the waste heat is still the issue but rather in terms of the equivalent power dissipation. It is explained how the electrical energy by means of a flow of electrons through a semiconductor is converted to thermal energy [11], [12]. As the efficiency of this conversion is rather good, the question "what means energy efficiency in embedded systems" is discussed. Some metrics like performance per Watt are considered but also scrutinized. The lecture closes with the guideline of the course, that energy efficiency of embedded systems can be rather described as energy saving. Therefore, the entire system has to be considered, from hard- to software [4], [9], [13].

3) 02 Digital Circuits: As this course addresses students in computer science classes, some general basics on digital circuits are provided, e.g. how Boolean functions can be described by logic gates. The needed stitching components (mainly transistors) are described while bipolar transistors and Field Effect Transistors (FETs) are discussed in detail. Not only the general theory of operation but also the power dissipation are analyzed. The advantage of FETs against bipolar transistors is therefor stated out, where we consider both the active as well as the leakage power. Factors of influences, e.g. the dimension, are discussed and explained [11], [12].

Subsequently digital circuits based on transistors are presented and the advantage of FET based circuits (CMOS) against bipolar logic is highlighted [14]. Of course, a huge part of this lecture derives and examines the power dissipation of CMOS. While computer scientists usually are not able to design custom CMOS gates, the knowledge about the power dissipation is essential for further optimization (e.g. the impact of the clock rate or the voltage level). In sum this lecture should provide an overview how the energy efficiency is closely related to the used gate technology and how CMOS technology saved about 10 years of power dissipation compared to bipolar logic [15].

4) 03 Sleep Modes (Power- and Clock Gating): The introduction of sleep modes is motivated based on the knowledge of the power dissipation of CMOS [16]. It is obvious how duty cycling an MCU could save energy but the question arises how sleep states could be implemented. Local and global clock gating are described and pros and cons are discussed. Based on real MCU architectures the advantage of clock gating and and therefore freezing parts of an MCU is presented [17].

To reduce also leakage power, power gating is introduced. Again local and global power gating is considered. Here the issue of state retention and the overhead for waking up devices is discussed [18], [19]. Scheduling as well as online algorithms to weight off the overhead for power gating an MCU vs. energy savings due to deep sleep are part of this lecture [20].

In the end, students should know how sleep states could be implemented and, however, that sleeping might not always be beneficial (overhead).

5) 04 Frequency and Voltage Scaling: With regard to the active power dissipation of CMOS, frequency and voltage scaling are an effective way to increase the energy efficiency. However, as a down-scaled frequency enlarges the run-time of tasks, students learn that a sole frequency scaling is not sufficient. Hence, frequency and voltage scaling [21] is introduced (Dynamic Voltage Scaling (DVS)) [22]. In this connection the dependency of the propagation delay and the voltage level is essential [23].

As usual MCUs do have minimum operating voltages, also DVS might not be ideal for several scenarios. Hybrid approaches (DPM + DVS) [24] are introduced and their benefit is demonstrated. Nevertheless, yet again DVS techniques induce an overhead. Software overhead, as well as additional hardware components (voltage conversion) might prune the efficiency. Thus, also Passive Voltage Scaling (PVS) [25] techniques are introduced to show the variety of possible scaling approaches.

As DVS also affects the run-time of tasks, scheduling mechanisms for optimizing the energy efficiency are part of this lecture [26], [27]. Again, the students should get both, an idea of which techniques are available and a feeling of pros and cons when utilizing these mechanisms to save energy.

6) 05 Undervolting (active): A recent topic to reduce the power dissipation of embedded systems is undervolting. In this lecture *active* undervolting is considered where parts of an embedded systems are operated on voltage levels below their specifications. This technique is part of more powerful CPUs which is discussed using the example of Critical Path Monitors (CPMs) [28]–[30]. As these CPMs are not available for small embedded MCUs, a case study of COTS parts is presented. It is legitimated why undervolting is feasible and how it can be established even for COTS parts [31].

On the one hand online testing methods as well as the impact of the environmental temperature is explained and the significant potential of energy savings are shown. On the other hand, the disadvantage of increased (static)overhead and the

risk of faulty execution are discussed. The implementation of a software-based supervised learning scheme to implement undervolting [32] on embedded devices is also part of this lecture. The suitability for daily use, in particular the reliability, is also shown by a field study.

7) 06 Undervolting (passive): While *active* undervolting focuses on the active state of an MCUs and devices, respectively, *passive* undervolting is utilized when components are idle. In this case a reliable state retention has to be guaranteed while the device is in sleep state and the voltage is set below the specification. The fundamentals for state retention of SRAM cells is examined and thus the usage of *passive* undervolting is legitimized.

An implementation to reduce the leakage of an MCU during sleep state [33] is presented but also the utilization for peripheral components [34]. For further optimization an optimal scheduling is shown. The theoretical consideration as well as a practical implementation illustrate how additional overhead can be weighted off against the energy savings due to *passive* undervolting.

8) 07 Energy Harvesting and TPC: Nowadays energy harvesting is again popular as it has become cost efficient compared to classical energy sources like batteries [35]. The general architecture for enabling energy harvesting is presented and the need for this technique is motivated by more and more long term applications, e.g. IoT sensing applications. Several sources of energy like solar, heat or vibration are discussed in detail while the need for efficient energy harvesting management chips is stated out. An efficient and effective voltage conversion and storage is essential and therefore discussed. For example the Maximum Power Point Tracking (MPPT) and charge pumps are part of this lecture.

Moreover, software snippets to utilize energy harvesting are inevitable [36]–[38]. In particular the principles, pros and cons of forecasting mechanisms are discussed. In this connection modeling of systems and harvesting processes are presented to estimate the energy budget of energy harvesting systems [39].

One step further, Transiently Power Computing (TPC) as part of energy harvesting systems, is part of this lecture [40]–[42]. Here the issue of state retention with NVRAM and the corresponding overhead taught. Optimization and real world systems demonstrate the relevance of this topic.

9) 08 Recent Research and Application Layer Optimization: Finally, each implementation to increase the energy efficiency of a system is closely related to the actual application. Each developer of a system should be aware of the mechanism that are utilized to lower the power dissipation. For example what happens if an undervolting scheme induces an error to a system? Such errors could propagate through a system and thus lower the efficiency or lead to even more power dissipation. The impact of such errors as well as methods to avoid such issues are discussed [43].

Other mechanisms to reduce the duty cycle of systems are techniques to adapt the sample rate. A practical example shows how an adaptive sample rate could be implemented based on the data stream [44].

Students should recognize that the acquired knowledge of the previous lecture helps to identify potential spots to optimize the energy efficiency of a system. Furthermore they should be aware of potential drawbacks like errors or additional overhead that might prune the energy efficiency.

10) Planned Future Extensions of the Lecture: The course recently took place for the first time in Summer Term 2018 and we received some constructive feedback how to revise and extend the lecture, respectively. While the current content discusses energy efficiency on hardware level, raises awareness and closes the skills gap towards the higher layers in the system stack, the lecture could be extended by more software related approaches to analyze, model and estimate energy consumption [45]–[47].

Yet again it is important to point out that software controls the hardware and that there is a need for computer scientists who understand the connection between low power hardware and an appropriate software design [48]. Moreover, frameworks [49], [50] and tools [51] to support efficient software development should be part of the lecture in near future.

B. Exercise

In parallel to the lectures there are six exercises. Each exercise picks up the topics of the lectures, provide additional information and theoretical tasks to train the subject matter. For each exercise an exercise sheet is handed out one week before the actual exercises. Thus, students have the opportunity to prepare own solutions beforehand. Apart from that, all solutions are discussed during the exercise by using slides and the blackboard. The blackboard might be an outdated instrument but when solving mathematical problems it is still the tool of choice.

1) Exercise 01: Beside the general discussion of lecture 00 and 01, two theoretical tasks are part if the first exercise. To understand the source of power dissipation in electronics, one task is to calculate the bulk resistance of a semiconductor. Subsequently, the power dissipation in terms of waste heat has to be calculated as a function of an exemplary clock signal. In a second task we provide a typical current profile of a system. The students have to determine the peak power, the average power as well as the energy over a period of time. Based on their calculations, the students have to select a battery that would be able to power such a system for a lifetime of 1 year.

2) Exercise 02: Exercise 02 is mainly focused on the power dissipation of CMOS. Thus, the power dissipation for different voltage levels, clock rates and activity factors have to be calculated as well as evaluated.

3) Exercise 03: Using the example of simple logic gates, the activity factor is discussed more detailed. Methods to lower the activity are explained and some examples are calculated to show how e.g. restructuring can help to reduce the utilization of gates.

The second part of this exercise considers the sleep states and how they can be established in practice. In particular power- and clock gating are analyzed by comparing their effectiveness and overhead using the example of a fictive

system. Students have to calculate the energy for different modes, e.g. no optimization, utilization of sleep modes and system-level power gating. The analyzes are very fine-grained and, e.g., also includes the power dissipation of I^2C pull-up resistors.

4) Exercise 04: To understand the connection between the voltage level and the clock rate of synchronous circuits, the propagation delay, the critical path and, thus, the maximum clock rate are calculated by using an example. These basics are important for DVS and its utilization. While DVS can save a significant amount of energy, it will also lead to scheduling issues on system level. Different scheduling algorithms are discussed and executed to solve exemplary DVS problems.

5) Exercise 05: In this exercise a fictive system should be optimized by using undervolting schemes. Active times of the system should be optimized by *active* undervolting while sleep periods should utilize *passive* undervolting. The calculation of the optimized system should be evaluated against the usual system. In sum the optimized system is far more efficient but the exercises concludes with a discussion about the pros and cons of the potential unreliable undervolting.

6) Exercise 06: Exercise 06 contains several tasks regarding energy harvesting where a preliminary calculation reveals the economical benefit for long-term applications. An exemplary system composed of a thermal energy harvester and a MCU is introduced and theoretically simulated. Therefore a system model including MPPT and harvesting as a function of the temperature gradient is derived. Moreover, a simple forecasting algorithm is discussed and calculated exemplarily.

C. Mini-Presentations

Mini-presentations are a good method to create a more interactive course by enforcing discussions on the lecture's topics among the students. Therefore a paper of a recent conference is assigned to each student. Of course, the scope of the selected conferences and papers is closely related to the topic "energy efficiency in embedded systems".

The students have to prepare a short presentation where the content, the concepts and the connection to the course are highlighted. After the presentation (about 10-15min) a critical examination together with all students is desired.

Although the preparation of the mini-presentation is a slightly increased amount of work for students, we received only positive feedback. In addition, the mini-presentations offer the opportunity to rejuvenate the course as always recent research results are integrated naturally.

III. Conclusion

This short paper presented a course on energy efficiency in embedded systems. Although there is a plethora of existing courses targeting this topic, the requirements for students of computer science classes are somehow different. Many courses solely focus on one specific area of interest, e.g. computer architecture. However, a conventional computer scientist will never be able to exert influence of the architecture of e.g. MCUs but will be able to implement mechanisms on system level to utilize existing power management techniques.

Thus, this course provides an overview beginning from device level (electrical fundamentals) up to optimization techniques on system level (software techniques). Beside classical lectures and exercises, students are encouraged to prepare a mini-presentation about a recent topic related to energy efficiency in embedded systems.

So far we received positive feedback from the students (educational evaluation in Summer Term 2018: 1.4 with 1="very good" to 5="very poor") and it is planned to extend the lecture in future.

Acknowledgment

I like to thank the the anonymous reviewers for their helpful comments. Their hints and references will help to improve the course in near future.

References

[1] F. Kreith and R. E. West, *CRC handbook of energy efficiency*. CRC Press, 1996.

[2] A. V. Dimitrov, *Introduction to Energy Technologies for Efficient Power Generation*. CRC Press, 2017.

[3] J.-C. Sabonnadière, *Low emission power generation technologies and energy management*. John Wiley & Sons, 2013.

[4] J. Rabaey, *Low power design essentials*. Springer Science & Business Media, 2009.

[5] S. Kaxiras and M. Martonosi, "Computer architecture techniques for power-efficiency," *Synthesis Lectures on Computer Architecture*, vol. 3, no. 1, pp. 1–207, 2008.

[6] J. S. Yuan and J. Di, "Teaching low-power electronic design in electrical and computer engineering," *IEEE transactions on education*, vol. 48, no. 1, pp. 169–182, 2005.

[7] M. Winzker, "Addressing low-power electronics in a digital system and fpga design course," in *Global Engineering Education Conference (EDUCON), 2014 IEEE*. IEEE, 2014, pp. 69–73.

[8] Y. C. Lee and A. Y. Zomaya, "Energy efficient resource allocation in large scale distributed systems," in *Ninth International Symposium on Distributed Computing and Applications to Business Engineering and Science (DCABES), 2010*. IEEE, 2010, pp. 580–583.

[9] H. Karl and A. Willig, *Protocols and architectures for wireless sensor networks*. John Wiley & Sons, 2007.

[10] M. T. Schmitz, B. M. Al-Hashimi, and P. Eles, *System-level design techniques for energy-efficient embedded systems*. Springer Science & Business Media, 2004.

[11] U. Tietze, C. Schenk, and E. Gamm, *Electronic circuits: handbook for design and application*. Springer, 2015.

[12] N. H. Weste, D. Harris, and A. Banerjee, *Cmos vlsi design*. Pearson India, 2010.

[13] N. Storey, *Electrical & Electronic Systems*. Pearson Education, 2004.

[14] R. C. Jaeger and T. N. Blalock, *Microelectronic circuit design*. McGraw-Hill New York, 1997, vol. 97.

[15] R. Chu, R. Simons, and G. Chrysler, "Experimental investigation of an enhanced thermosyphon heat loop for cooling of a high performance electronics module," in *Semiconductor Thermal Measurement and Management Symposium, 1999. Fifteenth Annual IEEE*. IEEE, 1999, pp. 1–9.

[16] J. M. Rabaey, A. P. Chandrakasan, and B. Nikolic, *Digital integrated circuits*. Prentice hall Englewood Cliffs, 2002, vol. 2.

[17] N. Ickes, Y. Sinangil, F. Pappalardo, E. Guidetti, and A. P. Chandrakasan, "A 10 pj/cycle ultra-low-voltage 32-bit microprocessor system-on-chip," in *ESSCIRC (ESSCIRC), 2011 Proceedings of the*. IEEE, 2011, pp. 159–162.

[18] D. Flynn, R. Aitken, A. Gibbons, and K. Shi, *Low power methodology manual: for system-on-chip design*. Springer Science & Business Media, 2007.

[19] K. Agarwal, K. Nowka, H. Deogun, and D. Sylvester, "Power gating with multiple sleep modes," in *Proceedings of the 7th International Symposium on Quality Electronic Design*. IEEE Computer Society, 2006, pp. 633–637.

[20] M. S. Manasse, "Ski rental problem," in *Encyclopedia of Algorithms*. Springer, 2008, pp. 1–99.

[21] T. Kuroda, K. Suzuki, S. Mita, T. Fujita, F. Yamane, F. Sano, A. Chiba, Y. Watanabe, K. Matsuda, T. Maeda *et al.*, "Variable supply-voltage scheme for low-power high-speed CMOS digital design," *IEEE Journal of Solid-State Circuits*, vol. 33, no. 3, pp. 454–462, 1998.

[22] J. Pouwelse, K. Langendoen, and H. Sips, "Dynamic voltage scaling on a low-power microprocessor," in *Proceedings of the 7th annual international conference on Mobile computing and networking*. ACM, 2001, pp. 251–259.

[23] M. Balch, *Complete digital design: a comprehensive guide to digital electronics and computer system architecture*. McGraw-Hill, 2003.

[24] U. Kulau, F. Büsching, and Lars, "A Node's Life: Increasing WSN Lifetime by Dynamic Voltage Scaling," in *Distributed Computing in Sensor Systems (DCOSS), 2013 IEEE*, May 2013, pp. 241–248.

[25] Y. Cho, Y. Kim, and N. Chang, "Pvs: passive voltage scaling for wireless sensor networks," in *Proceedings of the 2007 international symposium on Low power electronics and design*. ACM, 2007, pp. 135–140.

[26] C. Schurgers, V. Raghunathan, and M. B. Srivastava, "Power management for energy-aware communication systems," *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 2, no. 3, pp. 431–447, 2003.

[27] P. Pillai and K. G. Shin, "Real-time dynamic voltage scaling for low-power embedded operating systems," in *ACM SIGOPS Operating Systems Review*, vol. 35, no. 5. ACM, 2001, pp. 89–102.

[28] A. Drake, R. Senger, H. Deogun, G. Carpenter, S. Ghiasi, T. Nguyen, N. James, M. Floyd, and V. Pokala, "A distributed critical-path timing monitor for a 65nm high-performance microprocessor," in *Solid-State Circuits Conference, 2007. ISSCC 2007. Digest of Technical Papers. IEEE International*. IEEE, 2007, pp. 398–399.

[29] M. Ware, K. Rajamani, M. Floyd, B. Brock, J. C. Rubio, F. Rawson, and J. B. Carter, "Architecting for power management: the ibm® power7 approach," in *High Performance Computer Architecture (HPCA), 2010 IEEE 16th International Symposium on*. IEEE, 2010, pp. 1–11.

[30] S. Das, C. Tokunaga, S. Pant, W.-H. Ma, S. Kalaiselvan, K. Lai, D. M. Bull, and D. T. Blaauw, "RazorII: In Situ Error Detection and Correction for PVT and SER Tolerance," *IEEE Journal of Solid-State Circuits*, vol. 44, no. 1, pp. 32–48, 2009.

[31] U. Kulau, F. Büsching, and L. Wolf, "Undervolting in wsns: Theory and practice," *IEEE Internet of Things Journal*, vol. 2, no. 3, pp. 190–198, 2015.

[32] ——, "IdealVolting: Reliable Undervolting on Wireless Sensor Nodes," *ACM Transactions on Sensor Networks (TOSN)*, vol. 12, no. 2, p. 11, 2016.

[33] H. Jayakumar, A. Raha, and V. Raghunathan, "Hypnos: an ultra-low power sleep mode with sram data retention for embedded microcontrollers," in *Proceedings of the 2014 International Conference on Hardware/Software Codesign and System Synthesis*. ACM, 2014, p. 11.

[34] S. Friedrichs, U. Kulau, and L. C. Wolf, "Energy-Efficient Voltage Scheduling of Peripheral Components on Wireless Sensor Nodes," in *ICC'14 - Workshop on Energy Efficiency in Wireless Networks & Wireless Networks for Energy Efficiency (E2Nets)*, June 2014.

[35] S. Priya and D. J. Inman, *Energy harvesting technologies*. Springer, 2009, vol. 21.

[36] J. R. Piorno, C. Bergonzini, D. Atienza, and T. S. Rosing, "Prediction and management in energy harvested wireless sensor nodes," in *Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology, 2009. Wireless VITAE 2009. 1st International Conference on*. IEEE, 2009, pp. 6–10.

[37] Z. Jiang, X. Jin, and Y. Zhang, "A weather-condition prediction algorithm for solar-powered wireless sensor nodes," in *Wireless Communications Networking and Mobile Computing (WiCOM), 2010 6th International Conference on*. IEEE, 2010, pp. 1–4.

[38] U. Kulau, D. Bräckelmann, F. Büsching, S. Schildt, and L. Wolf, "Reaperadaptive micro-source energy-harvester for wireless sensor nodes," in *Local Computer Networks Workshops (LCN Workshops), 2017 IEEE 42nd Conference on*. IEEE, 2017, pp. 1–8.

[39] C. Renner, S. Unterschütz, V. Turau, and K. Römer, "Perpetual data collection with energy-harvesting sensor networks," *ACM Transactions on Sensor Networks (TOSN)*, vol. 11, no. 1, p. 12, 2014.

[40] M. H. Alizai, J. Beutel, J. A. Bitsch, O. Landsiedel, L. Mottola, P. Pawełczak, K. Wehrle, and K. S. Yıldırım, "Doctoral school on transiently powered computing," *IDEA League*, 2017.

[41] D. Balsamo, A. S. Weddell, A. Das, A. R. Arreola, D. Brunelli, B. M. Al-Hashimi, G. V. Merrett, and L. Benini, "Hibernus++: a self-calibrating and adaptive system for transiently-powered embedded devices," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 35, no. 12, pp. 1968–1980, 2016.

[42] J. Hester, L. Sitanayah, and J. Sorber, "Tragedy of the coulombs: Federating energy storage for tiny, intermittently-powered sensors," in *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*. ACM, 2015, pp. 5–16.

[43] U. Kulau, S. Müller, S. Schildt, A. Martens, F. Büsching, and L. Wolf, "Energy Efficiency Impact of Transient Node Failures when using RPL," in *Proceedings of the 18th IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks*, ser. WoWMoM 2017, 2017.

[44] U. Kulau, J. van Balen, S. Schildt, F. Büsching, and L. Wolf, "Dynamic sample rate adaptation for long-term iot sensing applications," in *Internet of Things (WF-IoT), 2016 IEEE 3rd World Forum on*. IEEE, 2016, pp. 271–276.

[45] P. Wägemann, T. Distler, T. Hönig, H. Janker, R. Kapitza, and W. Schröder-Preikschat, "Worst-case energy consumption analysis for energy-constrained embedded systems," in *Real-Time Systems (ECRTS), 2015 27th Euromicro Conference on*. IEEE, 2015, pp. 105–114.

[46] K. Eder, J. P. Gallagher, P. López-García, H. Muller, Z. Banković, K. Georgiou, R. Haemmerlé, M. V. Hermenegildo, B. Kafle, S. Kerrison *et al.*, "Entra: Whole-systems energy transparency," *Microprocessors and Microsystems*, vol. 47, pp. 278–286, 2016.

[47] J. Morse, S. Kerrison, and K. Eder, "On the limitations of analyzing worst-case dynamic energy of processing," *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 17, no. 3, p. 59, 2018.

[48] G. Pinto and F. Castor, "Energy efficiency: a new concern for application software developers," *Communications of the ACM*, vol. 60, no. 12, pp. 68–75, 2017.

[49] H. Field, G. Anderson, and K. Eder, "Eacof: A framework for providing energy transparency to enable energy-aware software development," in *Proceedings of the 29th Annual ACM Symposium on Applied Computing*. ACM, 2014, pp. 1194–1199.

[50] X. Li and J. P. Gallagher, "A source-level energy optimization framework for mobile applications," in *Source Code Analysis and Manipulation (SCAM), 2016 IEEE 16th International Working Conference on*. IEEE, 2016, pp. 31–40.

[51] S. Abdulsalam, D. Lakomski, Q. Gu, T. Jin, and Z. Zong, "Program energy efficiency: The impact of language, compiler and implementation choices," in *Green Computing Conference (IGCC), 2014 International*. IEEE, 2014, pp. 1–6.