Prof. Dr. Antônio Augusto Fröhlich
Federal University of Santa Catarina
Software/Hardware Integration Laboratory
PO Box 476, 88040-900 Florianópolis, SC, Brazil
Phone: +55 48 3721-9516
E-Mail: guto@lisha.ufsc.br
WWW: http://lisha.ufsc.br/∼guto

January 5, 2011

International Journal of Distributed Sensor Networks
Editor
Prof. Cem Ersoy

Dear Prof. Ersoy,

A revised version of the manuscript "A Comprehensive Approach to Power Management in Embedded Systems", initially submitted to the International Journal of Distributed Sensor Networks on September 2010, has just been uploaded through Hindawi manuscript system. Bellow you can find replies to each individual reviewer and also to your considerations.

Reviewer 1

Remarks

This paper does not suit IJDSN. The methods and tools can be applied to any embedded system. Therefore I strongly encourage the authors to submit the paper to a journal on embedded systems like Journal of Embedded Systems.

Reply

Certainly wireless sensors are not the only kind of embedded system operated on batteries that could benefit from the power management strategy proposed in the paper. However, from my point of view the fact that an article published on the IJDSN addresses a problem that reaches out not only for wireless sensor networks but for a broader range of battery-operate devices has the positive potential to attract new readers. Indeed, the sensor/actuator system used as a didactic example throughout the paper already establishes a bound to WSN. The strong emphasis on energy consumption by I/O devices, particularly by radio transceivers and analog/digital converters, would also probably be taken by readers as WSN-focused matter. Nonetheless, two new WSN case studies in the realm of EPOSMote have been added to Section V: the C-MAC protocol (Section V.D) and the AD-ZRP routing protocol (Section V.E).

Reviewer 2

Remarks

The paper presents an approach for the design of embedded systems with both, real-time and energy constraints. The approach comprises a design-time component as well as a runtime autonomic management component. The focus of the integrated approach is on the usability of power management features of peripheral components under the precondition of real-time requirements. A special tool modifies the function calls to accesses to peripheral devices at compile-time to control the power managment features of the corresponding device, especially to enable the device if necessary. Switching to power save mode must be performed explicitly by the developer (with his knowledge of future usage) and/or in combination with a modified software scheduler and an autonomic power manager. The modified scheduler is aware of real-time constraints as well as the battery utilisation and controls the power management accordingly. The approach is demonstarted and evaluated by a single example that is explained clearly. The results show a significant saving of energy of the standard configuration and also show some alternative configurations with additional threads resulting in a higher enrgy consumption and/or an impossible schedule because of the real-time constraints. In Section V.B. Example Application, first paragraph, the authors state that the EPOS tools deliver a "run-time library that realizes the required interfaces and hardware description that could be matched by virtually any hardware platform". What does this mean? Does the single run-time library support any kind of hardware device? Or does it mean that it is possible to generate a specialized run-time libraries for each device? Please clarify this. A further related work that should be cited is the work from Kluge et.al: "Optimisation of Energy Consumption of Soft Real-Time Applications by Workload Prediction", Florian Kluge, Sascha Uhrig, Jörg Mische, Benjamin Satzger, Theo Ungerer,13th IEEE International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing Workshops (First IEEE Workshop on Self-Organizing Real-Time Systems, SORT 2010), Carmona, Spain. They presented a technique that automatically detects the schedule within an embedded system and controls the DVS of the processor automatically. The proposed technique focuses on soft real-time systems only. Due to some typos, proof-reading should be a good idea. Especially, Figure, Section, Table should be written with capital letters.

Reply

Thank you for your review. Your question about how EPOS tools deliver a tailored run-time system is very pertinent. We are currently half the way in the transition to a new tool set that relies on evolutionary algorithms to optimize the configuration and shall be able to point out the most suitable platforms for a given application. This led me to the confusing sentence you identified. It has been promptly adjusted to reflect the current tool-chain, in which a two-phase process is performed. The tools are first speculatively deployed in order to produce an initial specification. This specification can either be taken or manually modified. A second invocation produces the useful run-time system. The process has been described in earlier publications, including [31]. The suggested related work was cited on the paragraph discussing DVS and the capitalization of text element names was done.

Reviewer 4

Remarks

This paper proposed a power management scheme that was designed and implemented from user level API to the infrastructure for embedded systems. The author has been studied power management issues through the EPOS (Embedded Parallel Operating system) project and published several papers related to this paper. The efficient power management is one of important issue in embedded system and it is well structured. However I will give some comments and some problems. - One of main contributions is to adapt the heuristic knowledge. How to present the knowhow of designer? Does it depend on only Petry Nets? - Semantic power modes of the proposed power management API are ambiguous to reader and user. It is possible that user selects wrong API. How to solve this problem? - This paper suggested that the proposed power management scheme considers real-time features. However, the author just mentioned some strategies. What components are designed and implemented in proposed infrastructure?

Reply

Thank you for your review. Capturing designers' knowledge is certainly a big open question for Software Engineering and many other disciplines. In the paper, I do not propose means, but strengthen that such knowledge should not be ignored as happens with operating systems designed for end users. In the first paragraph of Section II, traditional UML diagrams are pointed as possible means. On the diagrams throughout the text, power management knowledge is expressed through explicit invocations of Power() methods and also by tagging operations with remark notes. Indeed, the Petry Nets used to model the Operation Mode Transition Networks described in Section III.B formally capture the operating system designer knowledge about how mode transitions affect distinct components, but this is only one aspect of the system. Deploying it broadly would be unpractical. The paragraph that introduces the semantic power modes in Section II.B has been rewritten to include examples that make their role easier to understand. The question about the implementation of power management infrastructure components that preserve the real-time characteristics of the run-time system in which they are to be deployed is very relevant, but unfortunately cannot be answered in this paper. Our long-term work with static metaprogramming and aspect-oriented programming has shown that, even if all algorithms within a component are deterministic and can have their WCET estimated, the adaptation of components performed by the associated compilers and tools while generating the final library or kernel, as long as the interference caused by one component on others, can compromise the system as a whole. For now, this verification is empirical in EPOS.

Editor

Remarks

Although two of the referees found the paper more useful, one of the reviewers commented that "The methods and tools can be applied to any embedded system." Therefore, the paper is more suitable for a journal on embedded systems like Journal of Embedded Systems. I tend to agree with that reviewer. For that reason, I will request that you include more examples specific to sensor networks showing how your proposed ideas will be used and will be effective in practice. Some of the indicators showing that the paper is written generically for embedded systems can be found on Page 4 in the parts mentioning the file system. It sound like the envisioned platform is much more complex than a simple sensor node. Moreover, only a few references are related to WSN and majority of them are related to generic embedded systems. If that is the case, I will also recommend that you submit your work to another journal related to embedded systems.

One additional problem not mentioned by the reviewers is the fact that there is only one simple experiment reported in Section V.B, and many conclusions are made based on this demonstrative example. I believe that more experiments are needed in order to be conclusive.

Reply

Thank you for your constructive suggestions. Since I received your e-mail, I have been intensively working on two new experiments aiming at further supporting my conclusions. The C-MAC protocol (Section V.D) has been conditioned in an experiment that corroborates the proposed autonomous power manager, while the AD-ZRP routing protocol (Section V.E) was extended to be energy-aware and further confirmed all the proposed mechanisms.

Sincerely,

Fröhlich

Antônio Augusto Fröhlich