doi:10.20396/revpibic.v0i0.id

XXVI Congresso de Iniciação Científica Unicamp
17 a 19 de outubro Campinas | Brasil

Adaptation to hardware variability for energy efficient use in the Internet of Things

Patrick de C. T. R. Ferreira*, Dr. Lucas F. Wanner.

Abstract

Due to the evolution of silicon chips manufacturing processes at smaller and smaller nanoscales, the computing and memory substrates are increasingly changing, especially from an energy point of view. Considering projects with transistors composed of countable atoms quantities, small variations in the construction process can lead to energy costs of 10 times greater for the same processing. Software systems that identify hardware characteristics and take measures to adapt to the execution context amortize unnecessary energy expenditures. This economy is crucial for the execution continuity of embedded systems and/or applications used in the Internet of Things (IoT), where the energy resources are limited. One way to do this is by staggering tasks efficiently, postponing low-urgency services and running them all in only one period of CPU activity. The objective of this project was to increase the energy efficiency of IoT devices by creating a system that is capable of reorganizing and scheduling its tasks, as well as redistributing them in a network to be executed in nodes with characteristics of lower resource consumption.

Key words:

Energy Efficiency, Embedded Systems, Internet of Things.

Introduction

Current hardware devices are being produced with ever more miniaturized technologies. Transistors are mounted on circuits with countable quantities of atoms, and any variation in their quantity implies in a significant difference in their performance, both in speed and in terms of consumption.

Within systems mounted in networks where it is possible that there is a shift of processing load from one node to another, one can take advantage of this discrepancy by sending the data to be processed all to the most efficient node.

This research used a Brazilian system for microcontrollers called EPOS (Embedded Parallel Operating System), which allows the generalization of code for multiple platforms and the abstraction of hardware concepts.

Results and Discussion

The built system used 3 microcontrollers that verified the temperature and humidity of 3 different environments and processed some calculations regarding the correlation between these data. The microcontrollers then exchanged information on how much remaining battery charge each still had, in order to identify which one would be the most economical, and collected data were sent to this node through the internal radio modules of the boards themselves, no longer being processed by the source device, which only collects information from then on.

New features have been added to the EPOS system, such as new, more comprehensive and practical power management modes, allowing to put the CPU and all peripherals in different modes of consumption through only 1 command, the methods of recovering the suspension modes through interruptions were deprecated and were rehabilitated, in addition to providing a new generalist scheduling mode and more accessible to developers, including methods for migrating tasks in a collaborative way for networks running this system. The energy consumption verified for each microcontroller in full mode was, on average, 10 times higher than in sleep mode and 24 times higher than in deep sleep mode. Comparison data between the system duration without the processing load displacement and with the system exploiting the hardware variability indicated a saving of about 20% of the system life time extension before the network nodes started to exhaust your energy supply.

Conclusions

The extension of the embedded systems duration through the hardware variability exploration has shown to be a very expressive gain regarding one of the biggest weaknesses of the IoT embedded systems that is the energy demand difficult to be supplied by the current batteries. In places that are difficult to access, or even when you have a network with many devices, it becomes impracticable to replace all batteries manually in the short term, every week, for example. For most IoT systems, savings of 20% may be sufficient to determine whether portable recharging means such as sunlight or mini wind generators will be sufficient to make it self-sufficient or not.

Acknowledgement

I would like to thank my professor Lucas Wanner for the support and encouragement throughout this work and also to CNPq, for the financial support for this.