Wireless Sensor Networks for Environmental Monitoring powered by Microprocessors Heat Dissipation

Luca Rizzon, Maurizio Rossi, Roberto Passerone and Davide Brunelli
University of Trento, Italy
{name.surname}@unitn.it

ABSTRACT

We present an energy harvesting solution for a wireless sensor network for indoor environmental monitoring in data centers. The energy that supplies the nodes is harvested from the heat generated by the server microprocessors using Thermo Electric Generators (TEG), which convert a temperature gradient into electrical energy. We present a performance comparison between two commercial TEGs under different server processor load profiles. We focus our attention on server boards based on ARM CPUs (Arndale with ARM Cortex A15 and Pandaboard with ARM Cortex A9), supplying nodes equipped with gas sensors. From our results and simulations, we are able to demonstrate the possibility of powering a perpetual environmental monitoring WSN with a 0.0027% duty-cycle with the energy scavenged from computationally intensive embedded platform.

Categories and Subject Descriptors

C.4 [**Performance of Systems**]: Design studies; B.8.m [**Performance and Reliability**]: Miscellaneous

General Terms

Experimentation, Measurement, Performance

Keywords

Thermo-Electric Harvesting, TEG, Seebeck effect, Energy-Neutral Systems, ARM CPU, Web server, WSN

1. INTRODUCTION

Power management for embedded systems is a topic of great interest for engineers and members of the designer community. For those systems that are disconnected from the power grid, many different solutions have been investigated in recent years. Power supply from batteries has been integrated with a combination of other technologies, such as complementary or alternative power sources (solar cells [3], windmills [11], other forms of harvester, etc.). Even if, in recent years, the research activities have been prolific from the point of view of power management and software policies (e.g. [2,8]), significant attention is needed to ad-hoc hardware power supply solutions for wireless sensor network (WSN) devices.

In this paper, we evaluate the effectiveness of thermo electric harvesting in the context of embedded systems. Our goal is to demonstrate that it is possible to recover electric power from the thermal energy dissipated by data center's CPUs and use it to supply a microcontroller-based WSN for environmental monitoring in the data center room itself. The interest on this topic is given by the huge amount of energy absorbed by server farms that is dissipated as heat rather than being used and servers are always on providing a virtually perpetual energy source for the WSN.

We focus our attention on ARM CPUs, because nowadays they are considered the most promising technology for the realization of the future server farms [12], as those used by large IT companies for Web 2.0 and cloud computing services. Notwithstanding the limitations that affect them, embedded systems perform a significant number of operations, and execute complex applications. ARM-based devices offer lower energy requirements, reduced costs compared to multicore architectures, higher thermal dissipation and consequently lower management costs. Besides this, the low computational complexity required by web services routines makes ARM CPUs highly interesting for this novel application [12].

The most used complementary energy source for WSN is the sunlight. The power harvested by means of solar panels extends the lifetime of WSNs for environmental monitoring in outdoor scenarios [16], but also in indoor video surveillance systems as it has been demonstrated [6]. This solution suffers a dramatic decrease of performance in case of scarce direct illumination as in case of server rooms which it seeks to settle in dark and cold places. Other technologies for energy harvesting have been developed, but they provide even lower performance in indoor environment compared with the above [15].

We considered thermo-harvesting because cloud farms are plenty of wasted heat. A Thermo Electric Generator (TEG) is a device that converts a local temperature gradient (usually due to a heat flow from thermal source) into electrical energy, by exploiting the Seebeck effect [15]. A TEG is made by a junction of two dissimilar metal bars. When bars face different temperatures, electrons move between the hot side and the cool side. The electromotive force is proportional

ENSSys'13 November 14 2013, Rome, Italy

http://dx.doi.org/10.1145/2534208.2534216

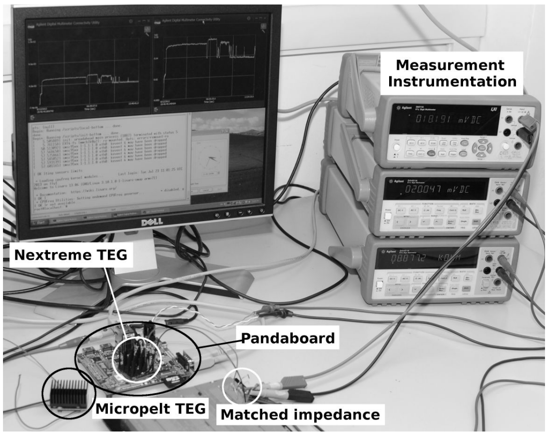

Figure 1: The setup for Pandaboard with Nextreme TEG measurement.

to the thermal difference that exists between the two metal layers. The efficiency of the conversion also depends on the composition of the two conductive bars.

Various studies in the literature have evaluated the energy available using thermo electric harvesting in other but similar settings, such as server racks or desktop computers [5,17]. Unlike these, we focus on energy harvested from embedded platforms that require much lower power and produce lower thermal gradients, which are therefore preferable in dense server room scenarios. The energy generated in our set up is comparable to that obtainable in server racks [17]. Higher energy can be obtained in particular specific situations [5]. However, our case study uses realistic benchmark applications which give us an estimate of the power generated by the harvester for different CPU load conditions.

The work is organized as follows. Section 2 introduces the experimental setup used to perform the characterization of the TEGs considered. Section 3 and Section 4 present the performance of the TEGs in terms of energy harvested with respect to the different configurations used. The lifetime simulation of a WSN for environmental monitoring is presented in Section 5. Finally Section 6 concludes the paper.

2. EXPERIMENTAL SETUP

We carried out a series of experiments to determine the amount of electric energy that can be harnessed from the thermal energy dissipated by a microprocessor. In this paper, we present the results obtained using two different commercial TEGs mounted on the top of the CPU package of two embedded computing platforms, for a total of four possible configurations. The two TEGs that we used in our experiments are:

- *Nextreme* eTEG HV56 Thermoelectric Power Generator without its output power regulator [9];
- *Micropelt* TE-CORE7 TGP-751 ThermoHarvesting Power Module with a 33 mm heat sink [7].

We do not use cooling fans in this work because the goal is to assess the energy harvesting capability using natural convection. It is reasonable to expect that forced convection using cooling fans will increase the power converted by the TEG due to the higher thermal gradient between the package and the region on the top. The two embedded systems used in our setup are:

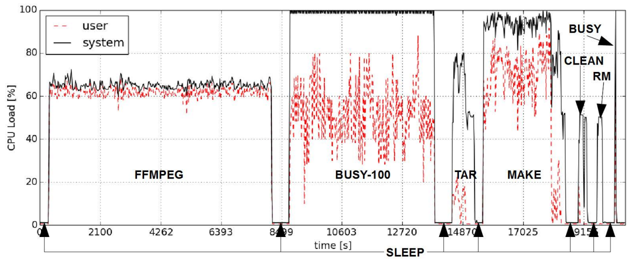

Figure 2: The CPU load generated by the benchmark application.

- a Pandaboard with a 1 GHz ARM Cortex A9 CPU and 1 GB of low power DDR2 RAM [10];
- a Samsung Arndale equipped with an Exynos 1.7 GHz dual-core ARM Cortex A15 processor with 2GB 800 MHz DDR3 RAM [1].

Both systems run Linux kernel version 3.10.

Experiments were conducted to determine the range of energy values that can be generated by the TEGs according to different CPU loads. In our experiments, we placed the thermal generator on top of the CPU package and measured the generated current while the board runs a benchmark application (see Fig. 1). We placed thermal gel between the CPU and the harvester surface to guarantee a good thermal contact between the two. While the hot-side of the TEG is adherent to the top of the CPU package, a heat sink is placed on the cool-side, to facilitate the heat dissipation and to keep a thermal gradient between the TEG surfaces. The two TEGs used in this work have different aspect ratio, heat spreader and sinks. The contact surface of the *Nextreme* is 32 mm×32 mm square, greater then the CPU packages involved in the experiments; while the *Micropelt* footprint is circular, with a diameter of 10 mm (comparable to the CPU packages). Both the TEGs come with their default factory heat sinks. *Micropelt* heat sink has a straight fin arrangement, while the one on the *Nextreme* has pin-fins. These differences can be crucial in the comparison.

We developed benchmark applications with heterogeneous tasks, to test the conversion process under different conditions, as follows:

video encoding using `ffmpeg`[1] with four threads, which converts a two hour long movie;

multithread application that performs millions of algebraic and trigonometric computations of floating point numbers using a user defined number of threads;

kernel operations in this task a Linux kernel is uncompressed, compiled and then cleaned, then the folder is removed.

The multithread application is executed twice to test different length in time. In the first call, 100 concurrent threads

[1] ffmpeg.org

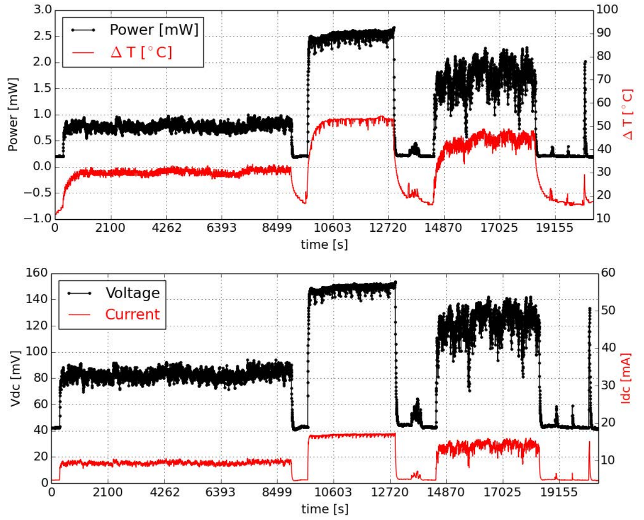

Figure 3: Power harvested with Arndale board and *Nextreme* TEG.

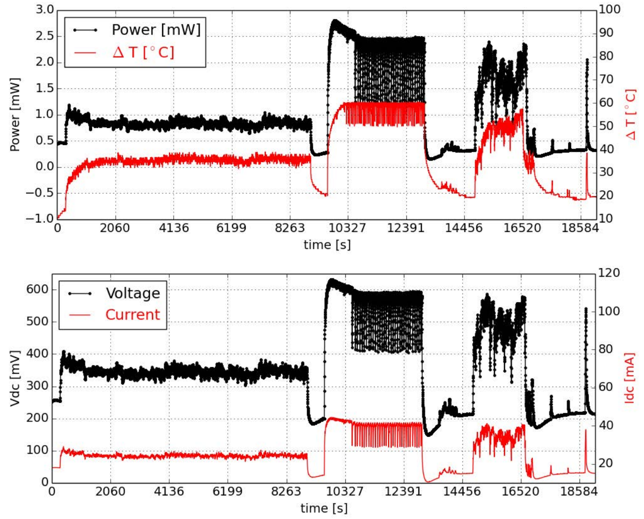

Figure 4: Power harvested with Arndale board and *Micropelt* TEG.

are launched a hundred times; in the second call, after the end of the kernel operations, ten threads are executed ten times. The purpose is to fill the cache of the processor and execute millions of context switch and operations. In between each of the aforementioned tasks, we let the device sleep for ten minutes.

While running the application, the software collects the values of CPU load and temperature for the Arndale Board, by reading the values of the internal sensors. Regarding the Pandaboard, the temperature has been monitored using a NTC thermistor placed on the hot surface of the TEG, because the Linux kernel we used lacks the module for internal sensors reading. Room temperature has been monitored using a NTC thermistor. During the entire experiment, the TEG output voltage and current have been measured over a matched load with a 2 s period using multimeters[2] controlled via PC software.

3. EXPERIMENTAL RESULTS

The TEG output depends on the temperature gradient that is directly influenced by the CPU activity. Fig. 2 shows the CPU load of an entire benchmark run for both user and operating system.

The values in figure are taken from the Arndale, due to space reasons we omit a plot for the Pandaboard, for which - however - we obtain similar curves. The user load line represents the amount of resources allocated for the application itself, while the system load includes, in addition to the user load, also the operations handled by the OS, therefore it represents the effective load of the processors that directly influence heating. In the same figure, labels indicate the name of the processes running as listed in the previous section. For the last three tasks (`clean, rm, busy`) the user load is very low, because those tasks are completely handled by the operating system.

In Fig. 3 - 6 we show the values for voltage and current at the TEG output, along with the thermal gradient as well as the resulting power computed as the product between voltage and current. The benchmark execution time for the Pandaboard (about 11 hours) is almost twice the time required by the Arndale (approximatively 6 h 30 m).

`ffmpeg` generates a lot of memory accesses, contrary to the multithread application (`busy-100`) which performs many mathematical operations, therefore the latter requires higher performance and produces more heat. In fact, we measured a 20° C difference on the hot surface for the two experiments run with the Arndale (cfr. Fig. 3-top and 4-top). This translates in more than twice the harnessed power from the CPU heat, while the values obtained with the Pandaboard are lower, since its core reaches lower temperatures.

Multithread operations appear to be the most CPU hungry. The shorter multithread task is not able to generate a considerable temperature gradient. Brief tasks generate bursts in the CPU usage that cause a rapid growth of the temperature on the hot surface that are visible in the graph that represents the instantaneous power. But those burst are captured differently by the two TEGs.

Operations on the Linux kernel give rise to a non-uniform CPU usage. Tasks like `tar, clean` and `rm` are handled by the OS, therefore they do not require operations on RAM and as a consequence they slightly warm the CPU. On the contrary, `busy`, that requires a hundred million mathematical operations and takes little time (about 30 s on Arndale and 55 s on Pandaboard), is able to cause a thermal growth of about 20° C.

4. DATA ANALYSIS

For each of the experiments, the average power generated per each task is shown in Fig. 7, while in Fig. 8 we plot the total energy gathered with the benchmark application. The overall performance of the TEG provided by *Micropelt* are better. However, it can be noticed that during the execution of tasks that generate lower temperature but run for longer time, the *Nextreme* TEG provides slightly higher output values. TEGs performance are influenced by the execution time of the task. Let us consider the `busywork-100` application and its simpler version `busywork`. When they

[2] Agilent 34401A and 34411A Digital Multimeters.

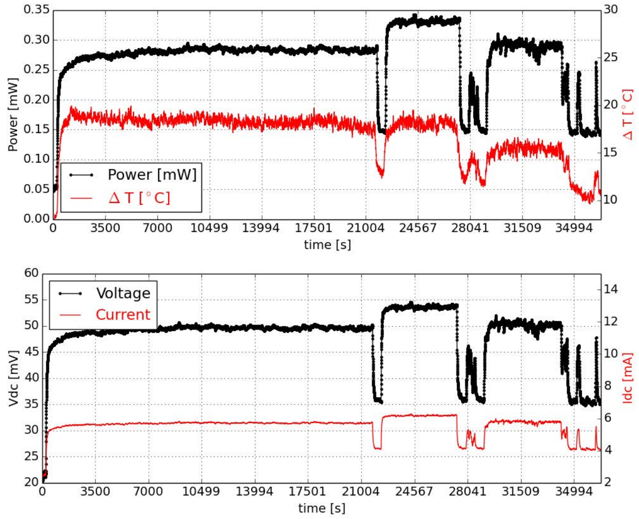

Figure 5: Power harvested with Pandaboard and *Nextreme* TEG.

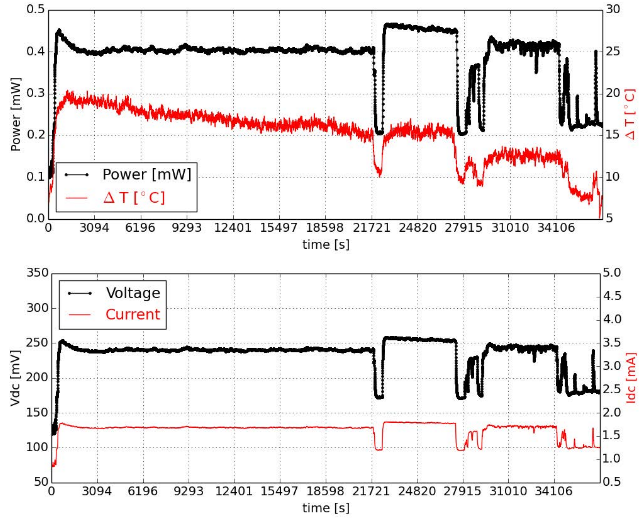

Figure 6: Power harvested with Pandaboard and *Micropelt* TEG.

run on the Pandaboard, we observe that the *Micropelt* performance are higher with respect to the competitor. On the Arndale, it can be seen that the *Nextreme* offers higher performance when the longer routine is executed. Therefore, we can infer that the *Nextreme* TEG produces higher output, but is less responsive in time with respect to the *Micropelt*.

By looking at the values for the instantaneous power obtained with the *Micropelt* (Fig. 4-top and 6-top), we can see that at the beginning of each task the power undergoes a steep rise and then it gradually comes down to stability, particularly evident in the case of task `busywork-100`. *Nextreme* TEG performs better when exposed to higher thermal gradients and longer execution time. This characteristic may be due to the adoption of a better heat sink, or to the use of metals that have better thermal characteristics. *Micropelt* TEG performs better for lower thermal gradients and longer execution times, and also for higher temperature rise in short time. Therefore, we can infer it is more reactive, with a better response to transients. The presence of a higher number of p-n junctions may be the reason for this characteristic. Moreover, the use of a heat sink with lower thermal exchange may expose the harvester to a lower thermal gradient.

We decided to use the *Micropelt* TEG as power source for further experiment of the environmental monitoring application because it demonstrates better performance with short bursts of computation, scenario that fits the kind of tasks required usually by a web server.

5. BATTERYLESS WSN APPLICATION

Embedded system can be used to detect potentially hazardous situations or monitoring environmental conditions inside the server room. Environmental monitoring can be performed using a wireless sensor network of gas sensing nodes (Wireless Gas Sensor Networks or WGSN), made by a small embedded system that collects information from standard sensors (temperature, humidity and light intensity) and processes the information provided by a gas sensor. Gas sensors employed in small electronic systems, like smoke detectors,

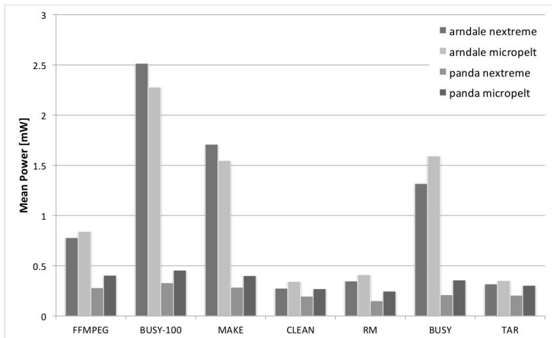

Figure 7: Mean Power per Algorithm.

are very cheap but they require a huge amount of power to work in comparison with the whole node [4, 13]. For this reason, they are mostly used in continuously powered applications. Conversely, battery-powered WGSNs require non-traditional sampling and processing strategies to achieve a reasonable maintenance-free lifetime, for example the ones proposed in [13, 14].

Thus, instrumenting a server room with an environmental monitoring network may pose powering issues due to the presence of chemical sensing devices. We evaluated the feasibility of replacing the batteries of the wireless nodes with the energy collected by the thermal harvester. In our simulation we considered both Arndale and Pandaboard acting as web-servers with the *Micropelt* TEG harvesting energy to supply the WGSN. To simulate a generic server load, we considered a random sequence of short CPU tasks interleaved by idle states. The tasks chosen were: `tar, clean, rm` and `busy`, because these generate variable CPU load and they have different length (from few seconds up to some minutes).

The WGSN is based on the W24TH wireless node[3], a state-of-the art electronic board, designed to minimize the

[3]Produced by Wispes s.r.l. http://www.wispes.com

Table 1: Execution Time expressed in seconds

	FFMPEG	BUSY-100	MAKE	CLEAN	RM	BUSY	TAR
Arndale Nextreme	8731	3285	3921	48	4	31	361
Arndale Micropelt	8705	3436	2097	47	2	33	588
Pandaboard Nextreme	21507	4981	5373	86	4	53	548
Pandaboard Micropelt	21560	4973	5660	31	8	58	578

Table 2: Mean Impedance Matched Power in μW

	FFMPEG	BUSY-100	MAKE	CLEAN	RM	BUSY	TAR	IDLE
Arndale Nextreme	779	2515	1709	275	347	1318	318	198
Arndale Micropelt	841	2278	1547	342	409.7	1593	352	320
Pandaboard Nextreme	279.5	329.1	284.7	196	150	210	205.7	141
Pandaboard Micropelt	404	455	399.9	270	246	357	304	218

Table 3: Energy Recovered in J

	FFMPEG	BUSY-100	MAKE	CLEAN	RM	BUSY	TAR
Arndale Nextreme	6.802	8.266	6.717	0.013	0.001	0.041	0.114
Arndale Micropelt	7.321	7.818	3.245	0.016	0.0008	0.052	0.207
Pandaboard Nextreme	6.012	1.639	1.529	0.017	0.0006	0.011	0.113
Pandaboard Micropelt	8.716	2.264	2.263	0.008	0.002	0.021	0.176

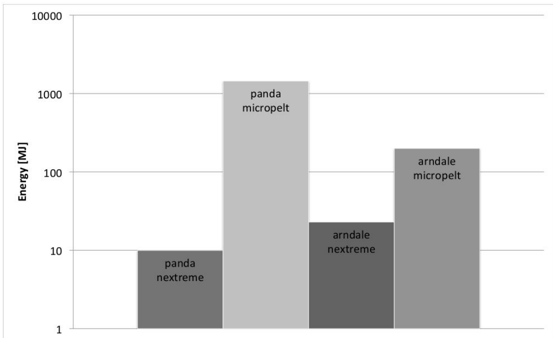

Figure 8: Total energy harvested per experiment.

energy wasted during the idle state [13]. The embedded board features an enhanced 32-bit RISC processor and a 2.4 GHz IEEE 802.15.4 compliant transceiver (current requirements are 15 mA for TX and 18 mA for RX @ 3V) in a single microcontroller[4] (MCU). The board also embeds sensors for measuring temperature, relative humidity, light and a dock for the gas sensor. We use a MICS-5121 gas sensors, a device targeted to volatile organic compound measurement (whose power consumption is 76 mW). We developed and measured a monitoring firmware that collects data for temperature, relative humidity and gas concentration using the low-power sampling and processing scheme developed by [13]. The network stack used to communicate is the standard IEEE 802.15.4 provided by the MCU manufacturer. In this configuration, the firmware takes almost 5 s to execute with a mean power consumption of 88.2 mW (29.4 mA @ 3V, evaluated with the same instrumentation as above). Each mote collects environmental parameters, then sends those to the network coordinator, and switches

[4]The MCU is JN5148 from NXP.

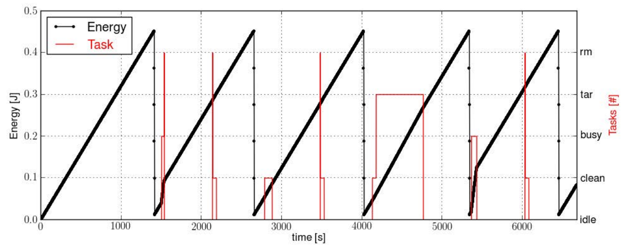

Figure 9: Duty-Cycle of a WSN node powered by Arndale with Micropelt.

off. In this implementation, the network configuration is stored in persistent memory thus a node can be completely switched-off.

An important characteristic of WSN monitoring applications is the duty-cycle that can be achieved: a trade-off between reliability, security and lifetime. To evaluate the performance of the W24TH mote in a batteryless configuration, we develop an energy budget simulator that takes as input the power harvested by the *Micropelt* TEG during the execution of different tasks, and simulates the WSN node operation. The simulator has been written in Python, it builds the sequence of tasks by randomly picking from the previous list, then computes the energy budget with 1 s steps by calculating the difference between incoming and outgoing power (using the values listed in Tab. 2). We considered ideal all the components to collect, store and connect the WGSN node to the supply. In this configuration, any WGSN node drains a total of 441 mJ to complete its duty so in the simulation a node is switched on only when the budget is greater than 450 mJ, the threshold value. We also considered the network already established by a coordinator node which is powered by the mains (as usual in WSN). The following results demonstrate that is possible to completely

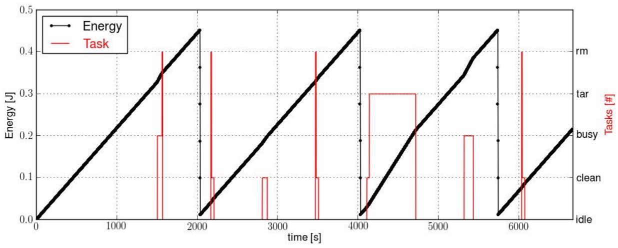

Figure 10: Duty-Cycle of a WSN node powered by the *Micropelt* on the top of the CPU package of the Pandaboard.

sustain the WGSN by means of the thermal harvester previously chosen. Fig. 9 and Fig. 10 show the simulations of the Energy Budget evolution in time for each of the boards used as *heater*. It can be noticed that the idle state by itself can sustain the WGSN measurement phase with a duty-cycle of about 0,0035% (5 s over a 23 min period) with the Arndale, and about 0,0023% (5 s over a 35 min period) with the Pandaboard. In the best case, it was possible to achieve 0,0046% (5 s over a 18 min period) for the first case and 0,0029% (5 s over a 28 min period) for the latter. These results show that with both of the considered boards and the Micropelt TEG, it is possible to sustain a distributed environmental monitoring system based on WSN and chemoresistive gas sensors with almost 0,0027% duty-cycle (two measurement per hour).

6. CONCLUSIONS

In this work, we presented a performance comparison between two TEGs applied on the top of of the package of two different CPUs for embedded devices. Our objective has been to evaluate which TEG is suitable as power supply for a wireless sensor network. Because of the recent trends regarding WSNs and the advantages offered from the adoption of ARM processors in big data centers, we focused our attention on those devices. Experimental results show that the *Micropelt* TEG offers better performance with both platforms. The principal advantage of the *Micropelt* is the higher responsivity to thermal bursts and higher heat dissipation at the cold side of the TEG, thus it can exploit the heat generated during tasks executions of the microprocessors, even if the CPU load is low. When CPUs are used at 100%, the performance of both harvesters are comparable.

In our case study, we evaluated the possibility of powering an environmental WSN that monitors the presence of chemical substances. From the simulations, we achieve a reasonable amount of power just harvesting the thermal power provided by a CPU in Idle state. In particular, the experimental setup allows the nodes of a WSN to sample environmental parameters and to transmit them to a sink node every 30 min.

7. ACKNOWLEDGMENTS

The work presented in this paper was supported by the project *GreenDataNet*, funded by the EU 7th Framework Programme (grant n.609000).

8. REFERENCES

[1] Arndale Board [Technical Reference], 2013. *http://www.arndaleboard.org/wiki/index.php/WiKi.*

[2] C. Caione, D. Brunelli, and L. Benini. Distributed compressive sampling for lifetime optimization in dense wireless sensor networks. *Industrial Informatics, IEEE Transactions on*, 8(1):30–40, 2012.

[3] D. Dondi, A. Bertacchini, L. Larcher, P. Pavan, D. Brunelli, and L. Benini. A solar energy harvesting circuit for low power applications. In *Sustainable Energy Technologies, 2008. ICSET 2008. IEEE International Conference on*, pages 945–949, 2008.

[4] V. Jelicic, M. Magno, D. Brunelli, G. Paci, and L. Benini. Context-adaptive multimodal wireless sensor network for energy-efficient gas monitoring. *Sensors Journal, IEEE*, 13(1):328–338, 2013.

[5] T. T.-T. Lai, W.-J. Chen, Y.-H. Hsieh, K.-H. Li, Y.-Y. Su, P. Huang, and H.-H. Chu. Why blow away heat? Harvest server's heat using thermoelectric generators. In *17th Int. Conference on Architectural Support for Programming Languages and Operating Systems, Poster Session (ASPLOS)*, London, March 2012.

[6] M. Magno, F. Tombari, D. Brunelli, L. Di Stefano, and L. Benini. Multimodal video analysis on self-powered resource-limited wireless smart camera. *Emerging and Selected Topics in Circuits and Systems, IEEE Journal on*, 3(2):223–235, 2013.

[7] Micropelt. *TE-CORE6 | TE-CORE7 ThermoHarvesting Power Module, 2013*. [Datasheet].

[8] C. Moser, D. Brunelli, L. Thiele, and L. Benini. Real-time scheduling with regenerative energy. In *Real-Time Systems, 2006. 18th Euromicro Conference on*, ECRTS '06, pages 261–270, Washington, DC, USA, 2006.

[9] Nextreme Thermal Solutions. *Thermomobility Wireless Power Generator (WPG-1)*. [Datasheet].

[10] pandaboard.org. *OMAP4 PandaBoard System Reference Manual*. [Datasheet], Rev. 0.4, 2010.

[11] D. Porcarelli, D. Brunelli, M. Magno, and L. Benini. A multi-harvester architecture with hybrid storage devices and smart capabilities for low power systems. In *Power Electronics, Electrical Drives, Automation and Motion (SPEEDAM), 2012 International Symposium on*, pages 946–951, 2012.

[12] Rick Merritt. Dell, IBM give thumbs up to ARM Servers. In *EE Times*, 2010.

[13] M. Rossi and D. Brunelli. Ultra low power wireless gas sensor network for environmental monitoring applications. In *Environmental Energy and Structural Monitoring Systems (EESMS), IEEE Workshop on*, pages 75–81, 2012.

[14] A. Somov, A. Baranov, A. Savkin, M. Ivanov, L. Calliari, R. Passerone, E. Karpov, and A. Suchkov. Energy-aware gas sensing using wireless sensor networks. In *Wireless Sensor Networks, 9th European Conference, EWSN 2012*, volume 7158 of *Lecture Notes in Computer Science*, pages 245–260. Springer Berlin / Heidelberg, 2012.

[15] R. Vullers, R. van Schaijk, I. Doms, C. V. Hoof, and R. Mertens. Micropower energy harvesting. *Solid-State Electronics*, 53(7):684 – 693, 2009.

[16] D. Wang, D. P. Agrawal, W. Toruksa, C. Chaiwatpongsakorn, M. Lu, and T. C. Keener. Monitoring ambient air quality with carbon monoxide wireless network. *Communication of the ACM*, 53(5):138–141, 2010.

[17] Y. Zhou, S. Paul, and S. Bhunia. Harvesting wasted heat in a microprocessor using thermoelectric generators: Modeling, analysis and measurement. In *Design, Automation and Test in Europe*, pages 98–103. DATE, March 2009.