ACSIJ Advances in Computer Science: an International Journal, Vol. 2, Issue 4, No.5 , September 2013
ISSN : 2322-5157
www.ACSIJ.org

ACSIJ
WWW.ACSIJ.ORG

Power and Energy Aware Design of an Autonomous Wireless Sensor Node

Nicolas Ferry[1], Sylvain Ducloyer[1], Nathalie Julien[1] and Dominique Jutel[2]

[1] Lab-STICC Laboratory, University of South Brittany (UBS/UEB)
56325 Lorient, BP92116, France
Nicolas.Ferry@univ-ubs.fr, Sylvain.Ducloyer@univ-ubs.fr, Nathalie.Julien@univ-ubs.fr

[2] Eryma Security System
56607 Lanester, Parc Technellys – La Montagne du Salut, France
Dominique.Jutel@eryma.com

Abstract

The design of Wireless Sensor Networks is a challenge, requiring to correctly balancing between performance, time, cost and energy. But the main problem with rechargeable WSNs is to predict at design time which will be the total system autonomy. Moreover, it depends on the energy harvested from the environment, and we know that weather may be very unsettled. Thus, it is crucial to design and fine scale the entire power supply chain in order to produce a robust WSN. In this article, we propose an energy estimator able to handle environment like weather parameters to estimate the system autonomy. The key innovation comes from the capability to dynamically rebuild the models all along the project evolution with real measurements on the hardware and to include weather forecasts as dynamic parameters of the DPM policy. Finally, we have experiment various configurations and compared the hardware WSN against the simulator. The results have validated the relevance of the estimator for prospecting various energy problems. By experiment, the estimator has shown that most environmental energy was wasted due to the battery charging constraints. This will foresee the opportunities of energy gains, and the definition of newer extra power modes for the Dynamic Power Management. This work contributes to the domain for WSN design methodology, energy scavenging and energy management to optimize system autonomy.

Keywords: *Wireless Sensor Networks, Low Power, Energy Management, Energy Harvesting, Energy Estimator, Dynamic Power Management, Weather Forecasts.*

1. Introduction

The design of embedded systems is a challenge because it must be found a trade-off between functionalities, performance, cost and energy. Concerning energy part, Low Power Design technique enables to reduce the overall power consumption of a system. Low Power Design (LP) can be applied at several levels from physical gate low-level to system high-level. Common LP techniques include clock gating which shutdown the clock activity in a chip or a branch if it is not required. In the same way, block gating deactivates an entire component, providing significant power reduction. Dynamic Voltage and Frequency Scaling (DVFS) technology [25] can regulate the supply voltage and the frequency dynamically at runtime. This approach reduces the power consumption without the overhead of a complete shutdown. Additionally, main processors and Radio Frequency chips support Low Power modes which can change the power state depending on the activity. For example, Idle, Sleep, and Deep Sleep are common processor low power modes that can reduce the power consumption during idle times. Each mode has its own specific couple of power consumption and wakeup overhead time. We can notice that these power modes are always functional modes where performance is degraded from nominal mode. Turbo mode is a particular mode where extra performance is given from nominal mode at the cost of energy lost. It is traditionally a frequency increase, and thus is similar to the DVFS approach. The Dynamic Power Management (DPM) approach [15] controls dynamically at runtime the system power consumption. The DPM applies a power management policy that reacts and handles the power consumption by applying DVFS or gating techniques to adjust the performance/power consumption ratio depending on events and on task activity. All these techniques can be applied to design an embedded system.

Wireless Sensor Systems (WSN) are autonomous embedded systems, thus they have important energy constraints. For system using non-rechargeable accumulators, the energy capacity is known and setup at design to support the charge. But as WSNs are spread systems, most of them rely on the environment and use it as direct energy or rechargeable energy source. Depending on the system power supply, this required to use supercapacitor or battery for energy storage system. In

Copyright (c) 2013 Advances in Computer Science: an International Journal. All Rights Reserved.

this case, the balance between environment recharge and power consumption becomes a key factor for the node autonomy.
But the problem with rechargeable WSNs is that environment and thus weather can be highly variable. Thus, the provided harvested energy is changing and cannot be predicted in advance with good accuracy. The design of well-scaled WSNs requires from one side to determine the global system autonomy and from the other side to anticipate both the system power consumption and the environment potential energy. This is crucial to design the system with an appropriate battery capacity, to size the energy harvesters correctly, and to optimize the power supply chain accordingly.

In this paper, the proposed solution is to define and realize an energy estimator able to predict the WSN node autonomy. The simulator aims at validating DPM policy and hardware choices by estimating the impact on the entire power supply chain. The estimator relies on the modeling of two parts: the hardware configuration and the environment. Its approach is to simulate the environment by replaying business scenarios, DPM strategy, RF communications, and weather scenario and then to solve the hardware models to provide both the power consumption and the node autonomy for a given geo-localization. It helps to scale the battery system and the harvesters accordingly, and to finely tune the power management strategy of the DPM. One major advantage of this approach is to handle adaptive accuracy models depending on the required error, accuracy, time, cost, or choices made for the project enabling the production of industrial well-scaled WSNs [2].

In this approach, the system is modeled at functional level; this means that the system is split into several high-level components. Thus, most physical components have been directly modeled with real hardware measurements when available. Then, an energy estimator has been created to predict the node autonomy by simulating business scenarios. Actually, one can notice that few environmental WSNs take care about the final localization and its environmental energy profile. As environment is highly variable due to weather, the approach has been to take into account weather for the energy recharge. Also, including weather forecasts into the dynamic power management (DPM) enables prospective energy decisions. This work has focused on the balance between weather/environment recharges and power consumption as a key factor on the node autonomy. Experiments have been explored for various hardware configurations and software DPM with the simulator. These results have validated the dynamic accuracy view relevance for prospecting various energy problems on the CAPNET project.

The CAPNET (CAP/Sensors - NETworks) consortium brings together industrials and academics for the R&D of an industrial WSN. The project goal is to evaluate the gain of migrating from a existing wired solution to a full wireless one. Thus, the system is an autonomous wireless sensor network prototype using multiple energy harvesting capabilities with a predictive DPM. It requires special features like fast alarm system and pluggable heavy industrial sensors support. In this project, we must handle realistic business scenarios for a fire brigade intervention. Thus, the consortium has provided software simulator and hardware demonstrator.

The article is organized as follows: In Section 2, we will recall how energy is handled in Co-Design Methodology. In Section 3, we will present the CAPNET project and the hardware model characterization. In Section 4, we will detail the Power and Energy Estimator functional behavior. In Section 5, we will present results about the test strategy to validate the simulator against the demonstrator, after which we will conclude the paper.

2. Energy Evaluation in Co-Design Methodology

Low Power Design [11] advancement has recently moved onto a higher level with algorithmic dedicated implementation, and operating system advancement with power down modes, Dynamic Voltage and Frequency Scaling (DVFS), and power-aware scheduling. Thus, power reduction can be applied all along the co-design [12] by respecting a low power design methodology [13]. The emerging wireless sensor network domain exists for a few years; however an increasing number of WSNs have been referenced [4]. Technology integration improves and nodes tend to achieve lower size like Smartdusts, Picocube [5], and Hitachi RFID Tag. In this part, we focus on embedded system design methodology and more precisely method that can be applied to WSNs.

2.1 Design Methodology

Low Power Design Methodology describes for a typical embedded system an approach for its realization. Its consists of heterogeneous components such as General Programmable Processor (CPUs, DSPs, µControllers), memories, and hardware blocks such as FPGAs, ASICs, which are connected through communication buses to form the system architecture. Moreover, the system

Copyright (c) 2013 Advances in Computer Science: an International Journal. All Rights Reserved.

designer must develop the software which commands the system. Components communicate with exterior thanks to analog or numeric inputs/outputs ports I/O, Analog-to-Digital Converters (ADC), Digital-to-Analog Converters (DAC). They interact at a higher level with sensors, actuators, and energy harvesters.

The design of an embedded system requires the following of several design steps. One possible and common design flow is characterized to use four design steps. The system specification (Step 1), the conception step often called the Co-Synthesis step (Step 2), the hardware and software synthesis (Step 3), and the integration part (step 4). The approach basically starts from the new product idea.

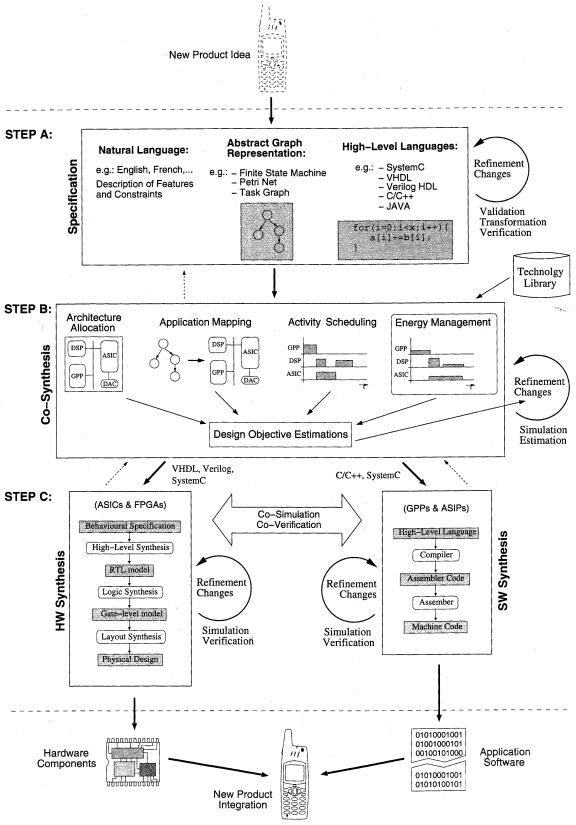


Fig. 1 Embedded System Co-Design Methodology. Source: [13]

Specification: Specifications consist to define, describe and specify the system. It consists mainly to answer the question: "WHAT the system does?" This could be done in natural language by spelling description of the features and constraints. But also by using conceptual specification models such as Finite State Machine (FSM), Petri Networks (Petri), Task/Dataflow Graph, or High-level languages like SystemC, Verilog, VHDL, C/C++, and Java to define a global view of the system.

Conception: The Co-Synthesis step consists to structure and architect the system. It consists mainly to answer the question: "HOW will work the system?" To achieve this goal, the architecture needs to address the fundamental problems: Architecture Allocation, Application Mapping, Activity Scheduling, and Energy Management.

Architecture Allocation enables to estimate the cost of several hardware solutions by determining advantages and drawbacks of different hardware solutions.

Application Mapping consists to assign the tasks or the functionalities, depending on the granularity reached, to a specific IP block. This Hardware/Software partitioning problem consists to determine if a task is mapped into a programmable unit or a hardware fixed unit. Consider that General-Purpose Processor (GPP) or Digital Signal Processing units (DSP) are more flexible and low cost; the counterpart is also less performance and less energy efficiency compared to FPGAs or ASICs system. Thus, care must be made on technology choices.

The *Activity Scheduling* is to order the execution of tasks and communications such that timing constraint is respected. The performance comes mainly from the application parallelism and the improvement of the critical path. This job can be done by using task graph and deadline constraints. Notice that various task mapping on the same allocated architecture can achieve different results, which could match or not a requested time constraint.

The *Energy Management* is necessary to estimate the energy requirements of the system. Energy Management can be enhanced by several ways and at different levels. First, one can include power management into the design methodology by defining a global Power Management Strategy (PMS) for the system. This could be done by defining various power modes, gating techniques, Dynamic Voltage Frequency Scaling, efficient algorithms, and low power components...

Traditional approaches were to search IDLE periods which consist of system downtimes. Each power mode transitions take time, thus it is necessary to enter special low power mode only when there is enough IDLE time. DVFS exploits voltage and/or frequency scaling to adjust performance to a requested energy profile. But using DVFS for reducing performance also delay the final task ending. The only way to use DVFS without performance penalty is to use slack time period. This time margin gain could be exploited for free to reduce the power consumption without performance losses.

It is important to notice that IDLE time mainly depends of hardware architecture, application mapping and activity scheduling. Finding the most suitable implementation

Copyright (c) 2013 Advances in Computer Science: an International Journal. All Rights Reserved.

requires iterative execution of the step 2 until matching constraints. This is done by applying the design Objective Estimations which is the most important phase to validate the conception. But whereas this phase is traditionally applied once at the end of the conception, in this paper it is proposed to use it all along the design evolve. Therefore, it is necessary to rely on an efficient energy estimator to determine if the solution matches the energy requirements.

HW&SW Synthesis: The Hardware and Software synthesis step brings the mixed hardware/software description of the system down to a physical implementation. This is achieved by using two separate concurrent synthesis processes: the hardware and the software synthesis. The main goal is the design of the individual hardware components and the software tasks running on programmable processors.

The *Hardware synthesis* is based on very large scale integration (VLSI). It is composed of several steps which depend of methodology and synthesis tools. But globally speaking, we can found at least these three steps: A high-level or behavioral specification is transformed into a structural description at Register-Transfer Level (RTL), by representing components with data path, arithmetic unit, control unit... The RTL description (in structural VHDL) is then translated into a gate level representation using a logic synthesis tool. Finally, the layout mask used for IC fabrication is generated from the gate level description through a layout synthesis tool. The gates are placed and interconnections are routed.

The *Software synthesis* part aims at transforming all tasks mapped on a programmable device from a high-level representation into a low-level machine code. In general this part is composed of two sub steps. First, the high-level representation is transformed into an intermediate object file by the compiler. This intermediate language format depends mainly from the target platform (or OS) and mostly defines program sections, library calls, memory rights and address relocation. Then depending the processor targeted, one or several intermediate files are linked together to map the final machine code.

Integration: In the integration part, the hardware mask layout of the physical devices is sent and produced by the factory. Test and verification are done to validate the IC stability and correctness. If test features have been introduced into the chip, the Build-In-Self-Test (BIST) can be used to check basic functionalities. The software code bitstream is loaded into the electronic card. Finally, validation tests perform scenarios behavior checks on the system as specified into the specification document. And the system is finally boxed and ready to be distributed.

2.2 Problem Description

Independently of the methodology used, the problem is the same: the design of Wireless Sensor Networks is a challenge because design is constantly evolving. It requires handling an issue by iterating a step when a problem appears or modifications are requested. But it is difficult to determine if a new decision will impact or not the system constraints and in particular its autonomy. Moreover, an issue does not impact design the same way if it appears at the beginning or at the end. At beginning, a structural issue does not impact much because the project has not yet requested heavy realization. But as far as design evolves, it becomes more and more difficult to include a structural change. Thus, it is crucial for project team to estimate that time, cost, technology and energy constraints match all along the project evolution.

To handle this drawback, the project leader needs to estimate that the system is kept in the project perimeter. Traditionally, this is done by verifying, that the constraints satisfy the requirements. If it is easy to verify simple constraints, it can become more problematic when constraints imbricate and rely on special behaviors. One problem is for battery powered system since batteries have special charge/discharge behaviors [26] over time. Another problem is weather which is highly variable and not determinist [32], because power harvesters directly rely on it. It is also true for external (scenario) events that must dynamically perturb/change the Dynamic Power Management behavior. And finally exploring all the space could explode in exponential time.

As described previously, objective estimations are necessary to validate a project design. For the energy point of view, it is important to adapt the hardware and the software co-design so that the energy constraints are verified. The use of a dedicated tool enables to validate these energetic constraints.

2.3 Tools overview

Several Power tools are available, in table I, for embedded systems and energy analysis but we underline the lack of a global view simulator, which is less accurate than dedicated tools but can integrate the high-level components and the weather environment at system level. Interesting works have also detailed recent techniques in battery modeling [16], DPM [15] to design a WSN [2]. As for tools, there are several power consumption platforms which analyze at specific power levels from gate-level to methodology-level. As can be seen in Table 1, there are good tools to predict the power consumption of

Copyright (c) 2013 Advances in Computer Science: an International Journal. All Rights Reserved.

microcontrollers or OS overhead of embedded systems. However, most of them focus on particular CPU/microcontrollers or platforms and do not consider WSN as a whole, therefore giving only a partial vision of power consumption that must be coupled with other sources. Energy tools and WSNs state-of-the-art has been detailed in [20]. A WSN power estimator is necessary to verify if some choices are relevant and contribute to system consistency.

Table 1: Power/Energy Estimation Tools

Power Tool	***Platform***	***Analysis Level***
SPICE	General-purpose circuit	Transistor
Simbed	Motorola M-CORE platform	Architectural
Skyeye	MIPS platform	Instruction
EMSIM	StrongARM platform	Instruction
JouleTrack	StrongARM, SH-4	Instruction
Softwatt	Mipsy, MXS platforms	System
powerTOSIM	TinyOS simulator on Mica2	System
SoftExplorer	CPU estimation tool	Funct.+Inst.
Interconnect	Interconnections tool	Architectural
CAT	Power estimation of RTOS	System

Although some components are well established and can clearly be measured to provide and specify the granularity model, some component models must be based on generic models. Consequently, the two approaches must be considered depending on component selection.

2.4 Solution Description

The goal of the simulator is to help at design to identify major power consumption hot-spots. The key concept here is to drive the initial system concept down to its realization.
To do so, the simulator builds a project target which is an abstract view of the system at a given time. The simulator can estimate both the node power consumption and the system total autonomy to validate if the proposed solution matches the objective. To give a good similarity, this approach is like progressive JPEG images which become more and more accurate along they are received and decoded. Here the same applies, except for some parts of the system that do not need to be detailed while others must be very sharp depending on cost, time, need, and energy constraints. Thus, the key innovation comes from the possibility to dynamically adjust the design concept, to evaluate its relevance.

We rely on the modeling approach which consists to mathematically and/or algorithmically describe the system behavior.

At node level, the model defines two layers: the logical part and the hardware part. The logical layer models the "environment" like business scenario, RF communications, weather, weather forecast and the DPM whereas the physical layer models the hardware architecture. Both are tightly coupled and are synchronized to work together.
We propose to start by defining the hardware Architecture allocation of a node which is a global view of the system architecture. This concern not only CPU and RAM units but also DC/DC converters, battery, energy harvesters, MPPT converters, and most significant components. To follow the design evolution, models need to be dynamically revised depending on the desired accuracy and the knowledge of the solution.
Thus, we apply an existing measurement methodology named F.L.P.A to model each principal elements of the WSN node architecture.

The Functional Level Power Analysis (F.L.P.A) methodology [21] can be adapted to any component and has already been successfully applied to extract processor, and FPGA power consumption models. This methodology is based on physical measurements in order to guarantee realistic values with good accuracy.

The F.L.P.A methodology has four main parts, which are given below:
1) A primary functional **analysis** describes the system as a grey box with high-level parameters having an effective impact on the power consumption.
2) The power **characterization** step explains the power consumption behavior (obtained by physical measurements) when each parameter varies independently.
3) Then the complete power **model** is created with a curve fitting or regression approach; it expresses the overall power consumption variations related to all the parameters.
4) Finally, the model accuracy is estimated by comparing real measurement set which gives the **absolute error**.

With this methodology, it can take less than a month to build a power model of a complex processor. At project start-up, all components or decisions have not been validated yet, thus we define a variable accuracy granularity depending on project advancement. Thus, model accuracy evolves during project maturation by evaluating the cost to develop a more accurate model. Two cases are used:

Copyright (c) 2013 Advances in Computer Science: an International Journal. All Rights Reserved.

- If the physical element is not available, we search for an existing model (datasheet, analytical law, etc.). Then, we build a generic model which is integrated into the simulator.
- If the physical element is available, we proceed with the F.L.P.A methodology to extract power measurement for a given accuracy.

Then for the logical layer, it is necessary to solve the software power management dynamically (i.e. the DPM) depending on tasks and external events. This has been addressed by defining business scenarios of typical applications, and decomposed into events. Weather has been included by extracting real weather station data for past and present, together with weather forecast website data for previsions. RF communication events (i.e. the frames) are modeled using a separate RF simulator developed by a partner laboratory. All these models will interact to the system, so that the DPM will react by controlling the hardware part. In counterpart, the hardware state also can affect the DPM response.

Thus, by realizing an estimator, one can determine by simulation the design objective. To illustrate the approach, we will present the Estimator developed for a special case project named CAPNET.

3. CAPNET PROJECT

CAPNET is a collaborative project including industrial and academic partners for designing an autonomous wireless sensor network prototype using multiple energy sources. Its objective is to study and implement a WSN simulator and a demonstrator that show the interest of the wireless approach over the wired one. The project is expected to reduce the cost by a factor of ten over the previous technology. Nodes must support a recharging feature by harvesting multiple environmental power sources. The specification includes special aspects such as an integrated alert system able to warn a supervisor in less than one second. The WSN uses a linear topology ranging from 100 meters to 2 kilometers. The nodes are spaced at intervals of 100 meters and support a bidirectional data transport protocol. If a node presents a dysfunction, the system is able to detect and reconfigure itself to automatically skip one (or up to two) failing nodes. Thus, each node can communicate by skipping two direct neighbours. The detection occurs when a timeout is triggered due to a defective node. Then, the node increases the amplifier range to reach the next node distance. Weather Forecasts (WFs) are transmitted to the nodes in order to switch the battery charger to the maximal available power source. WFs can also inform the supervisor of the remaining node autonomy for each DPM service. We have integrated the WF into the specific DPM called the WF-DPM.

3.1 Applications

The network topology is linear and forms a polygon around the secured area perimeter. This choice guarantees an optimized alert timing for all the nodes from the farthest to the nearest. A node is presented in Figure 2.

The system must handle various applications like fire services, frontier and perimeter surveillance, network monitoring (of water, gas, and petrol), environment observation (earthquake, water overflow, and meteorology), automatic meter reading, road network surveillance, industrial control command. Customers provide representative use case scenarios corresponding to each application. The representative scenario presented here has been established in cooperation with the fire brigade of Brest, France.



Fig. 2. CAPNET Physical Node

3.2 Scenario

Fire alert in a manufacturing plant: one fuel tanker has outflow and is threatening to boil over. A fire brigade must secure the plant by water cooling the tanker with fire hoses.

Copyright (c) 2013 Advances in Computer Science: an International Journal. All Rights Reserved.

It is very threatening due to the presence of heat, fire, or blaze which could cause a tanker boil over. To monitor and prevent the risks, a security area is delineated around the plant from 300 to 500 meters. During intervention, the WSN equipment is deployed all around to monitor and help with management decisions. All people entering the intervention area send an alert signal to the supervisor who can view snapshots or video of the scene. For this mission, nodes embed traditional meteorology sensors, but also special gas detectors, explosive detector, and for special nodes a thermal camera to track hot points. The system must be able to work continuously outside and in all types of weather.

3.3 Contributions

In this project, our work is to specify and realize the Power and Energy Estimator that will permit the playback of release scenarios. Our needs are to predict the remaining node autonomy with harvesting capability. But estimating the power peaks is also necessary for the battery model accuracy. The simulator results provide the total node autonomy and the different energy consumption for different groups of internal components. Moreover, by integrating weather forecasts into the DPM, we can predict the Ambient Potential Power Value (APPV) and then give the maximal time duration per service.

The APPV is the potential future energy prediction. Therefore, the simulator can estimate the power and energy consumption for each component according to their control attributes, the weather environment, the energy provision, the temperature degradation, and the power management strategy (PMS).

4. Hardware Models

CAPNET node showed in figure 2, is composed by waterproof box IP57 compliant integrating the electronic boards and the battery. It embeds a tubular flagstaff that supports bidirectional antennas, the solar panel and the wind harvester on the top. The board features three power harvesting channels, a microcontroller unit coupled with external memory, the connections interfaces, and support up to two RF chips. The electronic prototype board, in Figure 3, has been created including Low Power concepts. This effort is traduced by the use of Low Power chips. The board supports power gating on most of the power supply chain to reduce power consumption when unneeded. The node also supports modular pluggable devices. The RF part is done by an industrial partner; it uses special self-build Low Noise Amplifier (LNA), Power Amplifier (PA) and bipolar antenna [31] for RF sensibility extension.

The board embeds the power supply which consists on a battery Sealed Lead-Acid AS512 / 2Ah from Sonnenschein.
The battery is a 12Volts battery with a floating point recharge voltage of 13.8V with a 200mA maximum current.
The energy harvesting part acquires energy on three channels. The solar panel is connected through an MCV connector which is connected to one LTC3652. This module acts as a Maximum Power Point Tracking Converter; it is also a boost DC/DC converter with a high energy performance. The output is regulated at constant voltage of 13.5V output voltage. Energy is stored into a 3.5F supercapacitor.

The wind energy channel works in the same way. The wind generator is connected to a MCV connector. An LTC3652 converts the maximum power point and provides the output as a regulated 6.5V. The energy is stored into a 3.5F supercapacitor. The thermal harvester is connected to a MCV connector. But due to small voltage, the voltage is firstly increased at 0.5V by a voltage transformer. Then a special LTC3108 is used for MPPT of low energy sources. The output voltage is regulated at 5.2V and energy is stored into a 3.5F supercapacitor.

Each channel is protected by a Schottky diode to avoid return currents, and provides current to the battery charger. The battery charger used is a MAX669 which provides a 13.5V DC and 200mA max current to recharge the battery. The power supply stage provides the energy to system board equipment, the RF chip and the sensors. The component input voltage ranges from 3.3 Volts, 5 Volts to 20 Volts components.

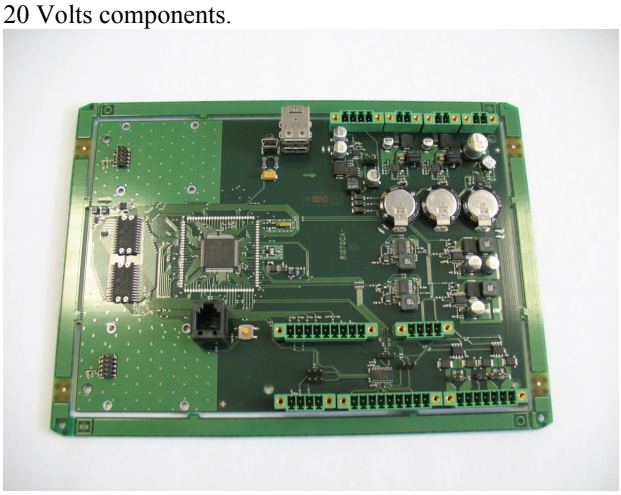


Fig. 3. The CAPNET Hardware Node Prototype

Copyright (c) 2013 Advances in Computer Science: an International Journal. All Rights Reserved.

For heavy gas sensor which requires 20 Volts, we have used one MAX618 Step-Up DC/DC Converter by sensor. It has a wide input voltage range to accept the battery voltage variation and can provide more than 500mA output current at 12V. It has a Low Power shutdown current of 3 μA and it is given with a high efficiency of 93%.

For 5Volts and 3.3Volts devices: Camera, MCU, RAM and RF modules, the system uses LM3100 Synchronous 1MHz Step-Down Voltage Regulator up to 1.5A output current. The input voltage range from 4.5V to 36V and the output voltage can drop down to 0.8V. Typical output current is, for the application, limited to 100mA by channel.

Data communications are decoupled from sensor devices by using Low Power MAX3241 (RS232), and ADM3485 (RS485). The MCU is a PIC24FJ256GB110 clocked at 8MHz from Microchip. This chip has been chosen for its low power modes, program memory size, and significant UART/SPI, and USB Host feature. The board integrates an external 64KB additional RAM M48T35AV enabling to buffer full JPEG images temporarily. These RAM integrates Ultra low power SRAM with a Real Time Clock and a little battery for memory saving during complete shutdown. Others MCP1252/1253 are used for local step-down conversion of USB port (for USB camera) and 5V external RAM.

For the simulator, the Hardware model is basically extracted from the main system hardware components. From this, the FLPA methodology is applied to each component. This helps the designer to define high-level parameters that characterize the power consumption and decrease the complexity by splitting the node into elements. Each model parameter is initialized in the configuration according to the respective datasheets and the simulation requirements.

4.1 Sensors

Each sensor is modeled at functional level because it permits to extract significant power parameters. Such parameters are specific voltage or current, but also power modes, power supply activation, and quality of service, which are modeled and controlled by the power management. However, sometimes real measurements on hardware could reveal datasheet distortions due to a mean function or different experimental conditions. Thus, datasheet models must be validated, if possible, in context before using them. For simulation performance consideration, it is also necessary to balance between model accuracy and computational complexity. Thus, models presented here rely on fast model resolving[1].

Oldham OLTC50 and OLTC80: The gas sensor from OLTC50 series are sensors with catalytic cell intended to flammable gas detection. They are specialized for measurement of flammable gas, toxic gas and are available with anti-detonation feature. Input voltage range is: 15V - 30V, its weight is 1.9Kg.

We measure the behaviour of the OLTC sensor located in an incubator at: -20, 0, +25, +35, +55 in °C. We vary the sensor input voltage from 15 to 30 Volts, and we record the intensity and power consumed by the device. We have obtained a series of I-V and P-V curves for different temperatures shown in Figure 4.

We notice that the power consumption evolution is similar for different temperatures. The consumption growth linearly when input voltage grows from 1024mW at 15V to 1157mW at 30V. This is a 12% increase at 20°C temperature.

There is lower power consumption gradually as temperature increase. The maximal increase is 6% when temperature ranges -20°C to +55°C. We can notice that the temperature growth causes a power consumption increase varying from 2.76% to 4.6% depending on the input voltage.

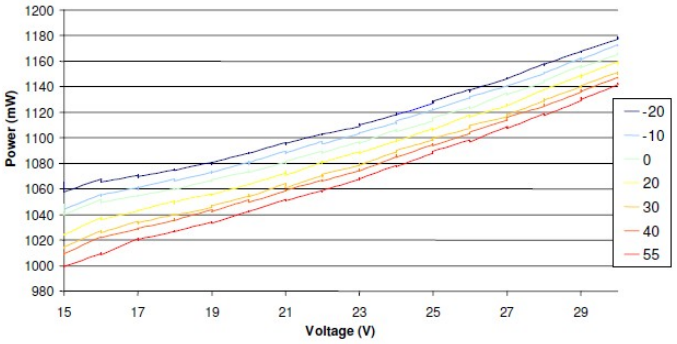


Fig. 4. Oldham OLTC sensor I-V characterization

[1] Notice that readers must take care when reading the error values, because values referred to sampled discrete signals which mean that error depends mainly from the mathematical law to match the discrete data. This is really different from measure-ments errors which rely on signal noise on an analogic data. Thus, we consider the problem and accept to include this phenomenon by smoothing to a perfect mathematical law, and sometime by using only one or few measures of the same law. This is because we work at coarse-grain high level representation, and we search fast model development at design stage.

Copyright (c) 2013 Advances in Computer Science: an International Journal. All Rights Reserved.

Thus, we can define the mathematical model by taking the maximum curve: the OLTC50 curve at +55°C.

$$I = 601.71U^{-0.815}$$
$$P = 9.175U + 860.53$$

with U: input voltage in V, and I in mA, P in mW.

(1)

The maximal increase associated to one input voltage is:

$$IncMax = -0.1142U + 6.1193$$ with *IncMax* in % **(2)**

The increase depending on the temperature is:

$$Increase = -IncMax.\frac{(T-55)}{20+55}$$ with *Increase* in % **(3)**

Final power consumption with temperature behavior integration:

$$P_{final} = (1 + \frac{Increase}{100}).I$$ with *Pfinal* in mA. **(4)**

Once the final model is built, the measured absolute error is **0.73%** at -20°C, and the maximum peak error is **2.63%** in the worst case. This gas sensor will be used at 20V in the final system, thus it will require a step-up boost converter after the battery.

SPI 107 Sensor: The SPI 107 sensor is an IR intrusion detector. The sensor sweeps a band to detect a person in a secure area up to 180 meters. This unit analyses the measurements and generates an alarm. Power supply is given for 12V +-25%. Functional temperature range is -40°C to +65°C.

For this power experimentation, we applied the FLPA methodology to measure and characterize the sensor. For the first step, we proceeded with a series of measurements by varying the sensor input voltage ranging into the battery voltage. These measurements give the device power consumption depending on the battery voltage.

This device has no special features, thus it is modeled by a linear equation:

$$y = 24.219x - 114.89 \quad \forall x \in [9,15[$$ in V. **(5)**

The quadratic error is **0.99%.**

S912 Sensor: The S912 sensor is an intrusion detector based on fence vibrations. While a person climbing a fence is detected, a signal is sent to the local unit. The unit analyses the measurements and generates an alert.

We measure the behavior of the S912 sensor located in an incubator at: -20, 0, +25, +35, +55 in °C. We vary the sensor input voltage from 10 to 30 Volts, and we record the intensity drained by the device. We have obtained a series of I-V measures presenting the I-V characterization curves for different temperatures shown in Figure 5.

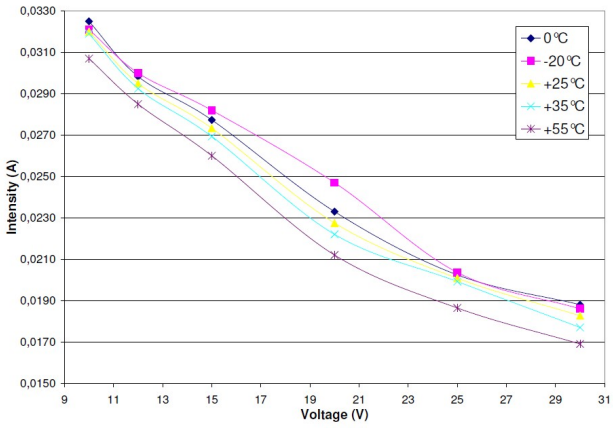


Fig. 5. S912 sensor I-V characterization

We notice that the power consumption increases as the input voltage grows, from 320mW at 10 Volts to 548mW at 30V.

This is an increase of 71.25% at 25°C. The power consumption has a 14% decrease when the temperature evolves from -20°C to +55°C. We can remark that the curve fits near a log function modeled by the equation (6):

$$y = -0.013\ln(x) + 0.0607$$ **(6)**

Moreover, we can remark that the increase of temperature causes a power consumption variation from 4% to 14% depending on the voltage. This increase can be modeled as a linear equation.

Once the final model is built, the measured absolute error is 1.9%. The maximum peak error is 4.7%. The sensor will probably be used at 10-15V like the battery nominal voltage. In this case, the consumption will range from 307 to 423mW and will depend directly on battery voltage. Here, a higher temperature gives lower consumption.

µCam Camera: The µCam device is a miniature color camera able to acquire a JPEG image through a serial or USB connection. Created for robotics and video surveillance, the camera is a 5 Volts RS-232 Devices which can send 0.75 fps video or compressed images (80x60 up to 640x480). The device has low power consumption.

Measurements have been taken at ambient temperature (15°C). We measure the intensity drained by the device at different phases by using the default software. We have

Copyright (c) 2013 Advances in Computer Science: an International Journal. All Rights Reserved.

obtained an intensity curve for various camera activities shown in Figure 6.

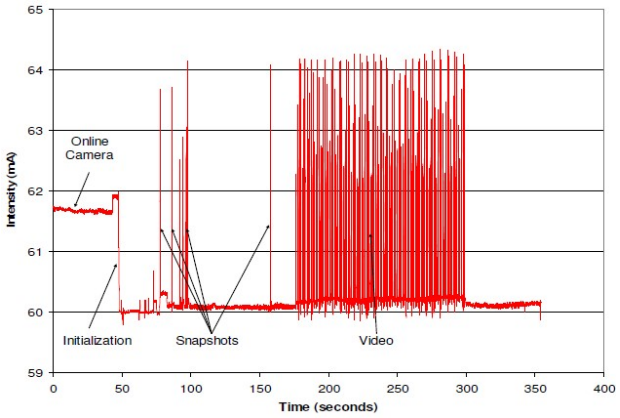


Fig. 6. µCam sensor

We notice that the power consumption increases while images are acquired and transmitted. There is no consumption difference when changing image resolution, or transfer speed.

The intensity consumption is 60mA in standby and 64mA for an image acquisition. Thus, it does not vary that much and can be assimilated as a constant model of 61mA. In this approximation, the error value is 1mA over estimated in µCam Camera Power standby mode and 3mA under estimated in activity. But we experiment a stability problem at 3.3 Volts with the TTL that forces us to fall back on a 5 Volts DC supply. With this voltage, the power consumption reaches 305mW.

4.2 Harvesters

Each harvester system is principally modeled by determining a standard I-V characterization (Intensity versus Voltage); therefore extracting the maximal P-V curves (Power versus Voltage) point law associated to it. Thus, a Maximum Power Point Tracking (MPPT) could provide the best power adjustment for the harvesting device. Harvester input parameters are supplied by the meteorology model.

Suntech 10W solar panel: The Suntech panel is a multi-crystalline silicon solar cell offering 10Watt maximal power. It uses a textured cell surface and bypass diode design to harvest maximum sunlight by preventing partial shadowing attenuation. The Suntech total efficiency is 8.8%.

Measurements have been taken on location with real weather contexts. For this, we use a Kimo SL200 Pyranometer to measure the solar intensity reference. The solar panel is oriented at 35° to the south direction. It is the optimal constant angle in the applicative region.

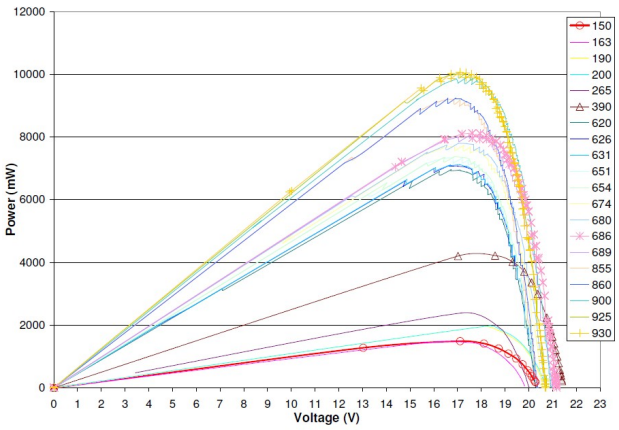


Fig. 7. Suntech Solar Panel Characterization

We use a programmable power supply to measure the intensity provided with different programmed ranges from 0 to Voc Volts. As one solar measurement takes up 10 seconds to record on the entire range, solar sunset could slightly vary, and thus introduce jitter. We have obtained an intensity-voltage (I-V) curve for different sunset irradiation powers ranging from 150 to 930W/m2. Each I-V curve is converted to a PV (Power-Voltage) curve where the maximum indicates the Maximum Power Point. Figure 7 shows the different P-V curves collected.

With the P-V curves, we reach 10Watt with 930W/m2 solar irradiation. We observed that the maximum power point propagates nearly 17Volts for this solar power range. It corresponds to the Vmp=17.20V and the Imp=0.58A given by the datasheet. Then, we modeled the Suntech maximum power point by a linear equation. Thus, we confirmed that the solar micro source harvested power range varies from 0 to 10W.

Rutland 504: The Rutland 504 is a wind generator producing 90W at 12V. Energy harvesting starts above 5 knots, the generator can typically produce 25W at 19 knots. We try to make measurement of the energy profile in real situation. But wind is not stable as sun irradiance can be. The wind generator start to oscillate depending on the wind direction and wind bursts. Thus, it's no easy to characterize both the regulator and the wind generator in this condition. Best way, would be to use a wind tunnel but it was out-of-the-scope of this project. Thus, we use the manufacturer datasheet curve.

Copyright (c) 2013 Advances in Computer Science: an International Journal. All Rights Reserved.

MicroPelt: CAPNET node embeds a DC/DC converter for harvesting thermal energy. We have evaluated the interest of thermal energy for this application. A thermopile is made of several serial interconnected thermocouples. The temperature difference between both sides is converted into electricity based on the Seebeck effect.

In the project, the solar panel is clearly an interesting thermal source because it partially converts solar radiations into heat. The basic idea is thus, to gather thermopiles at the rear of the solar panel. This will not perturb the direct solar input requirement and help solar panel cooling.
The temperature sensor has required special attention because of the thermal probe inertia. Firstly, we measure the response time of the thermal probe in order to set its incertitude.

A metal objet with an important mass is used to calibrate the probe. The reference object has been applied at different temperatures ranging from 5 to 70°C using an incubator. These measures can be approximated by the equation below which shows the thermal probe minimal time to acquire a correct temperature value:

$$y = -0.006x^2 + 0.7701x + 3.8067 \quad \textbf{(7)}$$

From these results, with x the response time in seconds, and y the temperature in °C, we will use the thermal probe between 20 to 45°C. This is equivalent to a response time of 25 seconds. Above this time, it corresponds to user manipulation or weather variation.

The equipment used: Two thermopiles: GM250-127-28-35. Thermal heat sinks (6+2): 1.65°C/W. Three solar panels (in parallel): Suntech 10W. One power supply from 0 to 24 VDC. Resistances 24 Ohm/3W, 2.4 Ohm/3W and 4.7 Ohm/3W. One thermal probe with a multimeter and two voltmeters. Special attention has been taken about heat insulation by managing a separator between each solar panel.

Measurement takes place with clear sky and low wind speed to prevent fast cooling of thermopiles. It consists to measure the electric power provided by a reference solar panel (SP Ref), by a solar panel with two thermopiles and two heat sinks (SP+TH+HS), and a solar panel with 6 heat sinks (SP+HS). The temperature measures are the mean value of five repeated measures which take response time into account.

Table 2: Temperature and Energy Analyse for Micropelt

Temperatures	*SP+TP+HS*	*SP Ref.*	*SP+HS*
Front face	28.4	29.2	27
Rear face	30.9	31.5	29
Heat sink	26.2	N/A	27.8
Ambient	25	N/A	25
Power Gain	6.7247W	6.7W	6.784W

Results in table 2 shows that the thermopiles, despite their thermal resistance, improve the solar panel cooling. The heat sink improves the solar panel energy provided an increase of 84mW which is 0.5% /°C which matches nearly the theory. In conclusion, this evaluation clearly shows, in term of cost, that energy gain provided by thermopiles and head sinks cannot compete against an increase of solar panel surface. Thus, the thermopiles must be reserved for very low power consumption WSNs and where high thermal energy difference is available. Consequently, thermal energy has not been selected for CAPNET nodes.

4.3 Microprocessor/RAM

Microprocessors often offer low power modes or a Dynamic Frequency and Voltage Scaling (DVFS) ability. Microprocessor model is obtained by using FLPA measurement methodology on real chips programmed to play the DPM.

The CPU model defines multiple power levels (Normal, Idle, Sleep, Doze modes) and switches between power modes depending on the DPM activity. If a more complex CPU model is required, we can rely on a dedicated external tool like CAT [19]. The DPM has the ability to shutdown the RAM when entering low-power sleep mode. An enhancement could be to include OS task scheduling activity for applying DVFS.

The power consumption of CMOS can be used to characterize a processor depending on voltage and frequency expressed by:

$$P = P_{stat} + \alpha.C.Vdd^2.F \quad \textbf{(8)}$$

For the project, the PIC24F will run at 8MHz frequency and with a 3.3V power supply. Thus, power is directly measured by experimentation for the different PIC power modes.

Copyright (c) 2013 Advances in Computer Science: an International Journal. All Rights Reserved.

4.4 Radio Modules

The radio device is usually the most power-hungry component. For the project, we rely on two MRF24J40 2.45GHz Transceivers from Microchip which support Miwi, Zigbee, and 802.15.4 protocols. One module uses a V-polarized antenna for alarm sensitive and control signals, and one other H-polarized for image transmission. The chip has also an AES 128Bits cipher and supports low power modes. The module has been completed by dedicated Low Noise Amplificator (LNA) and Power Amplificators (PA) to extend sensibility.
Radio is modeled using different power states and activity (RX, TX, sleep, etc.) in temperature. Power modes and timing are integrated in the model and are controlled by the DPM. For the measurement part, the MRF24J40 transceiver is located in an incubator at: -20, 0, +25, +35, and 55 in ° C. The Rx and Tx power mode consumption are modeled depending on the temperature. Thus, we can establish the power consumption by using second level equation:

$$P_{RX} = -0.0006.x^2 + 0.104.x + 118.12 \quad \text{with x in °C.} \quad \textbf{(9)}$$

$$P_{TX} = -0.0009.x^2 + 0.1264.x + 174.96 \quad \text{x in °C.} \quad \textbf{(10)}$$

4.5 Power Supply/Power Dispatcher

The goal of the power supply unit is to store energy harvested from the environment into the battery. The power dispatcher splits system power supply into several voltages for each component specification. Both use mainly DC/DC converters to step-down or step-up the voltage to provide regulated voltage. Converter efficiency is variable and low leakage current can appear. It depends on the manufacturer, the temperature, the input voltage and the output current.

LM3100 DC/DC Step-Down: The LM3100 is a BUCK 1.5A CMS regulator. It provides an output voltage between 3.3V and 5V from the 12V battery voltage. The global efficiency depends on voltage and intensity. We notice, here again, the efficiency increase as input voltage growth. There is a low efficiency variation when modifying the output voltage.

$$\mu = 79.(1 - \exp(-\frac{Ps}{55})) \quad \text{with Ps in} \quad \textbf{(11)}$$

We can compute linearly the maximal increase for the voltage:

$$Inc_{Max} = 1.206.U - 12.28 \quad \text{with U in V.} \quad \textbf{(12)}$$

Finally, the final efficiency is computed by applying the increase at the input voltage.

$$\mu_{final} = \mu.(1 + \frac{Inc_{Max}}{100})) \quad \text{with } \mu \text{ in \%.} \quad \textbf{(13)}$$

MAX618 DC/DC Step-Up: The MAX618 is a DC/DC Step-Up PWM CMS Converter providing for the project the 20V voltage for heavy sensors. We can model the converter as:

$$\mu = 90.(1 - \exp(\frac{-Ps}{150})) \quad \text{with } \mu \text{ in \%.} \quad \textbf{(14)}$$

Figure 8 shows the converter µ-curves for different input voltages.

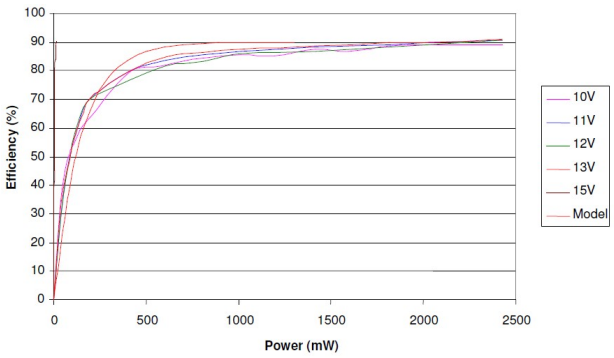


Fig. 8. MAX618 Step-Up Converter

MAX669: The battery charger MAX669 is a PWM Step-Up CMS chip providing an output voltage varying between 1.8V to 28V. The chip accepts a variable input voltage range, we experiment 14.4V and 13.8V which are the battery voltage recharge and the battery floating voltage.

These measurements show the efficiency increase for two voltages. Indeed, the maximum efficiency increases from 71.7% at 4V to 82% at 10V. This is a 14.37% increase at 20°.

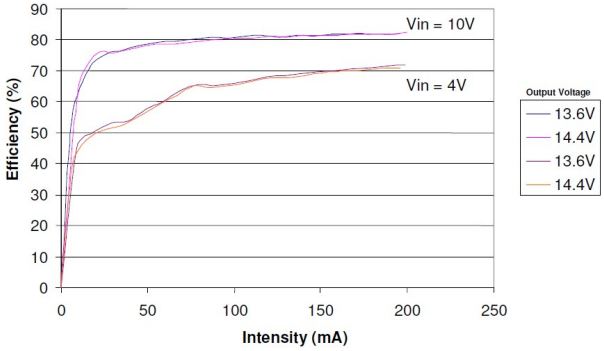


Fig. 9. MAX669 Step-Up Converter with battery charger

Copyright (c) 2013 Advances in Computer Science: an International Journal. All Rights Reserved.

We can notice the low efficiency variation produced when the output voltage is modified. By studying the efficiency evolution, we can determine the efficiency by a mathematical law:

$$\mu = 72.(1 - \exp(\frac{-Pe}{350})) \quad \text{with Pe in} \qquad \textbf{(15)}$$

We can compute linearly the maximal increase for the voltage:

$$IncMax = 2.73.U - 9.27 \quad \text{with U in V.} \qquad \textbf{(16)}$$

Finally, we compute the final efficiency by applying the increase at the input voltage.

$$\mu_{final} = \mu.\frac{IncMax}{100} + \mu \quad \text{with } \mu \text{ in \%.} \qquad \textbf{(17)}$$

4.6 Energy Storage System

One of the most important models is that of the energy storage system. The battery behavior is by far the most complex [26]. Most battery models predict the discharge time with a constant current, but the system uses environment harvesting technologies and can recharge itself while discharging.

Therefore, we use a battery model which handles both charge and discharge variable currents. This one is similar to the model developed by [27] but we use different curve fitting mathematical equations. As a battery presents poor reactivity to pulses, supercapacitor [10] shows good alternatives to assist main battery systems for harvesters [6], [9].

AS512/2 Battery: The AS512/2 cell is a sealed lead-calcium battery of 12 Volts for a 2Ah capacity. The battery uses lead-calcium grids with gas recombination and offers good recovery from deep discharge.
For this model, we use a similar method to [27] to model a battery with simultaneous charge and discharge. By using the F.L.P.A methodology, we define the high-level parameters from an electrical point of view. We use parameters defined in electricity: the electromotive force, the capacity, the impedance, the efficiency, and the self-discharge as the base of the model criteria. The electronic model is defined by (18) as:

$$V_{out} = V_{emf} - R_{\text{int}} * I \qquad \textbf{(18)}$$

Where *Vout* is the battery tension (V), *Vemf* the battery electromotive force (V), *Rint* the battery internal resistance
(Ohms), and *I* the provided current (A).

To take the measurements, we used a battery bench. The bench is composed of a power supply which provides the recharge current to the reference battery. The battery is placed in an incubator to measure behavior as a function of temperature. The bench integrates an electronic load which varies according to a measurement scenario.

To experiment a scenario, we must calibrate the power supply to deliver a constant recharge current. In this bench, a constant recharge current of 200mA is applied while a variable discharge current is applied with a repeated pattern. The battery voltage is measured over time until the threshold tension limit has been reached.

We calculate the Rint and the Vemf as a function of capacity. To do this, we take a couple of points on the data curve at each state transition. For example, one point on a peak period and one on another peak (with a nearest delta). As we have a couple of points, we can solve (18) by the double equation system (19).

$$\left\{ \begin{array}{l} V_{out1} = V_{emf} - R_{\text{int}} * I_1 \\ V_{out2} = V_{emf} - R_{\text{int}} * I_2 \end{array} \right\} \qquad \textbf{(19)}$$

By solving (19), we determine the evolution of the internal resistance and the electromotive force depending on the evolution of the battery capacity.

Then, we model the electromotive force and internal resistance parameters. We notice that Vemf can be modeled as a linear equation (20).

$$V_{emf} = -a.q + b \quad \text{with q the capacity (A.h).} \qquad \textbf{(20)}$$

The internal resistance has an exponential aspect modeled by equation (21).

$$R_{\text{int}} = a.e^{(b.q)} + c \quad \text{with q the capacity (A.h).} \qquad \textbf{(21)}$$

Coefficients a, b in equation 20, and a, b, c in equation 21 are determined by regression approach. They are tightly coupled with the battery and the technology chosen. Finally, the battery model is evaluated and compared to the real measurements in Figure 10. The absolute maximum measurement error is **8%**.

Copyright (c) 2013 Advances in Computer Science: an International Journal. All Rights Reserved.

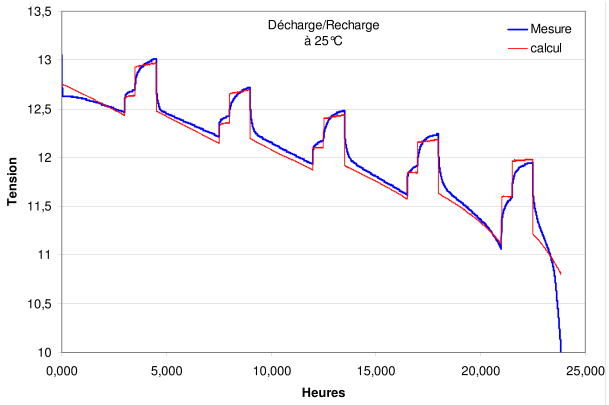


Fig. 10. AS512 battery model results

As the battery model handles simultaneous charge and discharge, it can be used to model a self-recharging WSN. Thus, it can predict the battery autonomy of a WSN node according to its activity and environment.
Finally, by repeating the process for all other components, we can build a power map for the node equipments. If components or circuit design change, models are rebuilt to reconstruct the power consumption map. Table 3 shows an example of the power consumption map for the required equipment of the CAPNET project.

5. Logical/Environment Models

The logical level defines the Node environment as several external models: Scenario Events, RF Events, Meteorology, and Weather Forecast models. The DPM model describes the energetic strategy developed on the node.

5.1 Scenarios

In Table 4, scenarios are composed of rule-based events which control the entire sequence. Event signals define the start time, the duration, the periodicity for periodic events, and the number of repetitions. This produces a signal that characterizes an event context like a sensor notification or other particular behavior (fire explosion, secure area intrusion, etc.). Events are exploited by the DPM which verifies the rules at each discrete time and consequently performs the corresponding actions. Basically, the DPM impacts the devices and updates their model values on the fly.

Table 4: Scenario Model Example

Name	Nature	Type	Start	Period	Dur.	Rep.
Gas Measures	Measures	Periodic	2	10min	2s	500x
Intrusion Niv1	Alarm 1	Periodic	1	300min	10s	32x
Intrusion Niv2	Alarm 2	Periodic	3	2hours	4s	10x

...

B. RF Events

The RF model describes the communication protocol as a set of RF events. RF Events define the incoming and outcoming packets exchanged by the node. The RF model is provided by our partners working on the RF modules and the RF simulation. Thus, scenario described in the both simulators must be the same, else the results could not be guaranteed.

Table 3: Node Power Consumption Maps

Type	Component/Chip	Mode 1	Mode 2	Mode 3
CPU	PIC24F256 @3.3Volts	Full: 16 mA 53 mW	Idle: 2.5 μA 8.5 μW	Sleep: 100 nA 330 nW
SRAM (x2)	A623308A @5Volts	Op R/W: 35 mA 175 mW	StandBy: 10 μA 50 μW	OFF: 0 mA 0 mW
RF1 AES128	MRF24J40 @3.3Volts	TX: 130 mA 430 mW	RX: 25 mA 82.5 mW	Sleep: 2 μA 6.6 μW
RF2 AES128	MRF24J40 @3.3Volts	TX: 130 mA 430 mW	RX: 25 mA 82.5 mW	Sleep: 2 μA 6.6 μW
Sensor1	SPI107 @10-13Volts	Operating: 300 mW	/ /	/ /
Sensor2	S912 @10-13Volts	Operating: 600 mW	/ /	/ /
Video	µCam @5Volts	Capture: 64 mA 320 mW	StandBy: 61 mA 305 mW	/ / /
Gas1	OLCT 50 @20Volts	(18-30V) 1080 mW	/ /	/ /
Gas2	OLCT 80 @20Volts	(12-28V) 900 mW	/ /	/ /
MPPT	LT3652 η=90-95% (in=4.95-32V)	Operating: 2.5 mA Variable	StandBy: 85 μA Variable	Idle: 15 μA Variable
DC/DC (Down)	LM3100 η=85-92% @(10.5-13V) Power stages:	Operating: 0.7 mA Variable (1-3x) 3.3V	StandBy: 17 μA Variable (1x) 5V	/ / / (0-1x) 9V
DC/DC (Up)	MAX618 η=93% @(10.5-13V) Power stages:	Operating: 2.5 mA Variable (0-2x) 18V	Shutdown: 3 μA Variable	/ / /
MS1 Solar	SunTech 10W PMax	1000W/m^2 10W	500W/m^2 5W	200W/m^2 2W
MS2 Wind	Rutland 504 PMax	75 Km/h 60W	20 Km/h 4.3W	15 Km/h 1.5W
MS3 Thermal	MPG604	dT=70°C 80 mW	/ /	/ /
CELL Battery	AS512/2 (10.5-13V)	12V 2Ah	/ /	/ /

Table 5 presents the RF matrix model. The problem here comes from radio time-slot resolution which is very short (in milliseconds) compared to the energy simulator (in seconds).

Copyright (c) 2013 Advances in Computer Science: an International Journal. All Rights Reserved.

To solve this problem, all the RF are simulated at the finest granularity. Typical scenarios have, in general, lot of repetitions or blank areas which can be exploited to reduce the RF computation time. Then the energy is computed by estimating over one second the percentage of time spent in each modes (RX, TX, Sleep or OFF) from the real RF data. Then, the hardware model will estimate the real energy waste with the equivalent power model and temperature attributes.

Table 5: RF Model Example

Mode	Time T_{x-1}	Time T_x	Time T_{x+1}	...
Radio1 OFF	100%	80%	60%	80%
Radio1 Sleep	0	0%	0%	10%
Radio1 Rx	0	20%	25%	0%
Radio1 Tx	0	0%	15%	10%
Radio 2 ...	...	...	...	...

5.2 Meteorology

Since energy is scavenged from the environment, the node model must include a model for each power source type. These environmental sources can be provided in multiple ways including solar, wind, thermal, radio, vibrations, mechanical constraints, etc. Currently, CAPNET focuses on solar, wind, and thermal power sources. Meteorology context is constructed with the Meteo-builder integrated with the simulator.

The context can be created manually, scanned from a website, or extracted from a meteorology file recorded day after day. The Table 6 presents a model example for the city of Lorient, France.

Table 6: Meteorology Model Example

$Date/Time$	T	W_{dir}	W_{spd}	R	H	P	S
30/06/2010 11:00	19	262	10	0	74	1022	229
30/06/2010 14:00	22	251	10	0	66	1021	228
30/06/2010 17:00	22	241	15	0	64	1020	275
30/06/2010 20:00	21	241	15	0	67	1019	45
Date/time$_i$	...	...	...	...	...	...	...

5.3 Weather Forecasts

Meteorology is the science which studies atmospheric phenomena mainly in the first 31 km of the atmosphere. The meteorologist Lorenz discovered the chaos theory [32] showing that the atmosphere cannot be entirely determinist. Small variations in the initial states can result in huge consequences in the final results: this phenomenon is known as the "Butterfly Effect". Therefore, the final interpretation is left to human interpretation to formulate the weather forecasts.

A new predictive approach consists in evaluating a set of different runs. Each run slightly modifies an initial parameter corresponding to a well-chosen particular error/perturbation. Therefore, the standard deviation can be exploited to give a trusted indicator for the predictions. Multiple Meteorology centers have developed their own models. We could cite: WRF, GFS, ECMWF, UKMO, JMA, ARPEGE, and AROME... Among these different models, the Global Forecast System (GFS) is an American numerical weather forecast model. This model divides the Earth into a mid-meshing grid of squares of 30 km and the atmosphere into 64 altitude layers. A model computation is called "a run" and predicts up to 16 days forward. The main advantage of this model is the output which is freely available and in real-time after generation. We reuse information given by the Meteociel website [22] which digests the GFS information in a vector. Each piece of interesting information is collected to supply the simulated harvested external energy potential. However, solar information is only given as a variable icon representation. To translate this information, we rely on another website to convert solar potential into a sunset irradiance in W/m2.

The European JRC - Photovoltaic Geographical Information System (PVGIS) is a website [23] for photovoltaic technology. For Europe and Africa, the system can give a photovoltaic estimation of the solar potential during an entire typical day of a specific month. PVGIS gives three sunset indicators: IR global clear-sky sunset, IR global real-sky sunset, IR diffuse real-sky sunset as a time function. IR global clear-sky is the maximal sunset irradiance expected with no nebulosity.

IR global real-sky is the standard sunset irradiance for the average cloud-covered sky in a month. IR diffuse real-sky is the expected sunset irradiance by diffuse luminosity which can be translated as the minimum irradiance that can be harvested even by a heavily clouded sky. All these information must be merged to provide the weather coefficients.

Since meteorology data are now widely available through multiple sources, they can be extracted directly from Internet websites. Weather coefficients can be combined together to produce an energy function prediction over time. In our Power Estimator, the scenario builder works as follows for the meteorology part. It can connect to specific meteorology Internet websites, from which the weather data is downloaded in HTML string syntax. The

Copyright (c) 2013 Advances in Computer Science: an International Journal. All Rights Reserved.

next step consists in parsing and extracting the weather data from the page. Of course, this module is generic and its implementation depends on the target website. Weather Data are presented in a vector structure where each column represents a weather component (Sunset irradiance, Wind speed, Temperature...) and the row increments in the time dimension. As most websites do not give the sunset irradiance, we must combine various website data. Depending on your world position (Latitude, Longitude) on the globe, and the month in which the simulation must run, PVGIS gives sunset measurements: IR clear-sky sunset, IR med sunset, IR diffuse sunset. These values are extracted for each month once at start up. The time granularity is given every 15 minutes.

Weather data and PVGIS data are interpolated according to a Lagrange Law, and are finally combined into a two dimensional vector of the simulation resolution. The result is a data file that can be exploited in the simulator to reproduce the weather parameters. However, we tested the Runge interpolation error that forced us to fallback to a Hermite cubic interpolation. Since energy is scavenged from the environment, the node model must include a model for each power source type. Currently, CAPNET focuses only on solar, wind, and thermal power sources. The meteorology context is constructed with the Weather-Builder integrated into the simulator. The context can be created manually, scanned from a website, or extracted from a meteorology file recorded day after day. Then, when the DPM replays a scenario, the weather state is updated and provides the weather data for the simulation. Finally, the simulator converts them into power depending on the actual hardware. However, we must keep in mind that despite WF models continuously evolve and offer better accuracy; WFs suffer from interpretation errors and/or inaccuracies. The WF theory limit is announced to a maximal of a two-week forward window. Currently, most WF models can predict 1-day up to 3-days forward with relatively good accuracy but a complete week with more uncertainty.

5.4 DPM

Dynamic Power Management (DPM) policy controls the different power states enabled for the system. This model is not directly the microprocessor program, but rather a subset of the global power states view. The DPM has been modeled as a Finite State Machine (FSM) composed of a set of states Si and transitions Ti. Each state represents a particular system power consumption state Pti, and has a variable duration Tti. The DPM commands the behavior of the system, entering special power modes or monitoring power supply. A sequence is a set of one-to-many states linked together and form a path to produce a service.

The WF-DPM model introduces the capacity to exploit the meteorological WF coefficients. WFs can be used for various applications like power switch policies to determine which power harvesting source will probably have the highest energy availability. For CAPNET application, WFs is computed on the supervisor which transfers only the indicator to stay or not in waiting states. In CAPNET, the intrusion detection service is defined by the DPM shown in figure 11. The system uses two sensors for detection except when there is not enough energy.

Copyright (c) 2013 Advances in Computer Science: an International Journal. All Rights Reserved.

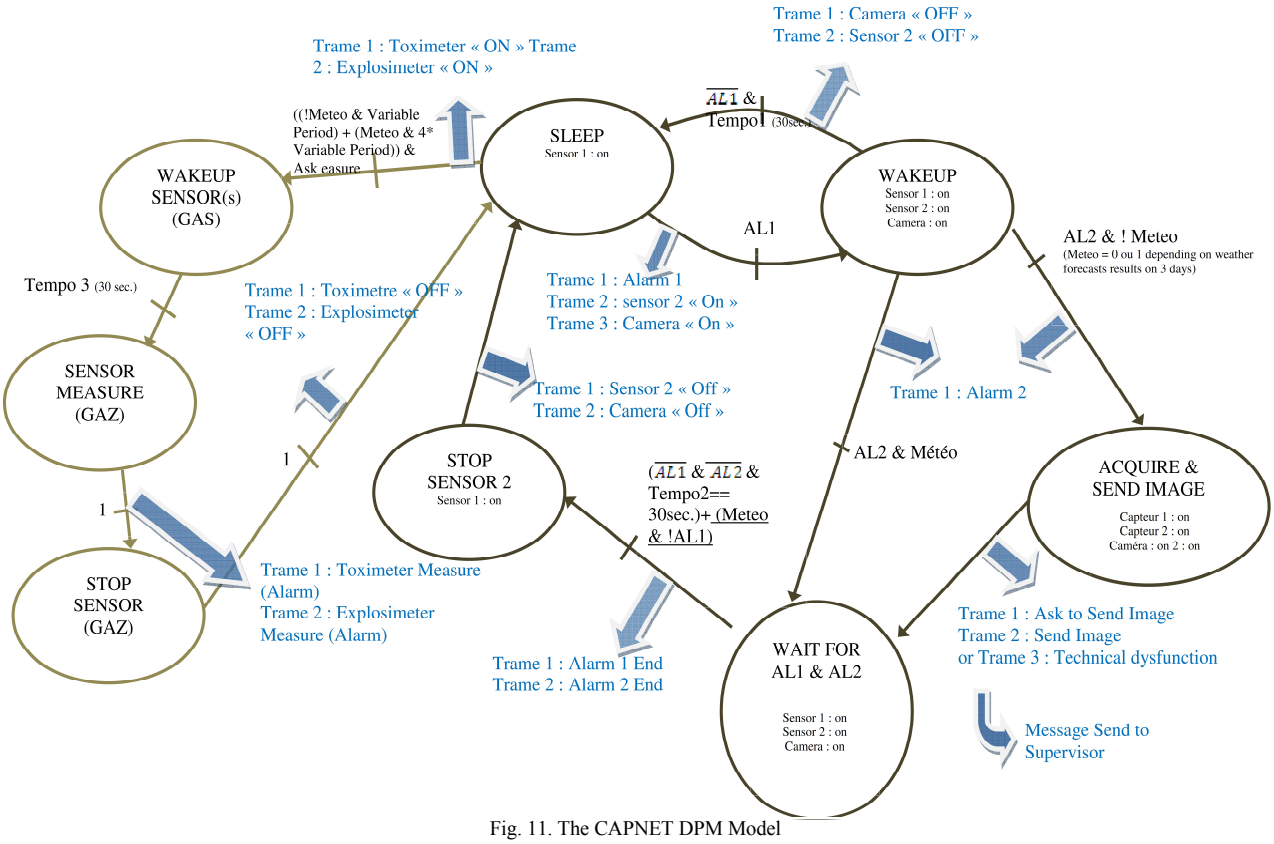


Fig. 11. The CAPNET DPM Model

The node directly sends an alarm and enters an IDLE waiting state. When the sensor 1 detection flag is raised, Sensor 2 is activated to verify the intrusion. If both are raised, the node sends an alarm to the supervisor. With enough energy, the system sends a video for a variable duration. Or it sends a snapshot in high or low resolution depending on the energy level. Then, the DPM waits for the two sensor releases before stopping sensor 2 and returning to Sleep State.

Video duration depends on a maximal timing and the result of the energy prediction function which is described below.

We define an energy function based on the battery State Of Charge (SOC) and the WFs. With WFs, we have prediction data for each ambient micro source: the sunset irradiance potential in W/m2, the wind speed potential in Km/h, the thermal temperature potential in °C. Due to specific models built for each micro source, we can determine the equivalent power produced. This gives us a power representation in Watt and, for a one-second resolution, the energy in Joules. Each micro source power potential is cumulated to provide the Potential Ambient Energy Value (PAEV), which is a total amount of external energy expected in the near future. We can quantify the harvested energy by expressing the PAEV as:

$$PAEV = \sum_{i=1}^{n} E_i . \mathrm{Res} \quad \text{with } Ei\text{: energy potential inW and } Res\text{: the resolution in s.} \quad \textbf{(22)}$$

Where i is the index in seconds in the WF forward moving windows, n its length size, Ei the cumulated energy of each harvester and Res is the resolution in seconds. Thus, if we define the battery SOC in Joules, we can express the battery charging duration as:

$$SOC + \sum_{i=1}^{n} E_i . \mathrm{Res} \leq SOC_{full} \quad \textbf{(23)}$$

The DPM has several states, each having its own level of power consumption. A sequence is a set of states realized to perform a typical task. If we know the duration of each state, we can calculate the Required Energy value for a sequence. We can express the total Required Energy (RE) for a sequence as:

$$RE = \sum_{j=1}^{nbT} Pt_j . Tt_j \quad \text{with } Pt\text{: task power in W, and } Tt\text{: task duration in s.} \quad \textbf{(24)}$$

Copyright (c) 2013 Advances in Computer Science: an International Journal. All Rights Reserved.

Thus, we can express if the battery will be discharged by a service, by solving:

$$SOC - \sum_{j=1}^{nbT} Pt_j . Tt_j \leq SOC_{low} \quad \textbf{(25)}$$

And finally, if we want to compute the remaining service duration Sd, we express it relatively to the battery State Of Charge (SOC) in Joules:

$$Coef = \frac{SOC}{\sum_{j=1}^{nbT} Pt_j . Tt_j} \quad \text{with } Pt\text{: task power in W, and } Tt\text{: task duration in s.} \quad \textbf{(26)}$$

Thus, if we can approximate a maximal sequence duration as:

$$Sd = Coef . \sum_{i=1}^{n} Tti \quad \text{with } Sd\text{: service duration in s, and } Tt\text{: task duration in s.} \quad \textbf{(27)}$$

Finally, the next axiom must be always true for the prediction window:

$$\forall i, SOC + PAEV - RE \geq SOC_{low} \quad \textbf{(28)}$$

The energy function results in two cases: if true the sequence has potentially enough energy to be executed, otherwise the sequence will probably fail due to a lack of energy early before reaching the maximal prevision window limit.

As batteries have special behaviors [26], modeling battery charging with only a sum/minus operator is an approximation. In fact, we have built a battery model that supports both discharging and charging in order to handle these behaviors. Such a model has been developed based on previous work done by [27]. The model enables us to estimate efficiently the battery capacity to handle service.

From this point, we can explore multiple solutions to adapt the quality of service. The supervisor can take control of the node whenever he wants, forcing it to use a special service. With these report indicators, he is alerted of the maximum service duration and can take suitable decisions.
Now that the Node model context has been presented, we will detail the Power and Energy Estimator itself which exploits the models built in this section.

6. The Power and Energy Estimator

The goal of the simulator is to help at design stage to identify major power consumption hot-spots. As the project constantly evolves, models are dynamically revised with the FLPA process depending on the required accuracy. Thus, the simulator can estimate both the node power consumption and predict the system autonomy. The simulator accuracy mainly depends on model specific errors.

On the other hand, simulating a DPM permits one to adjust the strategy and fine tune DVFS and DPM opportunities. Thus, designers can evaluate the relevance of the developed DPM policy. Finally, we can give a nearly realistic estimation of the global system viability for a configuration.

Figure 13 presents the Power and Energy Estimator (PEE). In the main part, the screen describes the node major hardware components, specific to studied WSN. The right part includes the controls of the system parameters in manual mode. Controls are updated in real-time in automatic mode, when playing a scenario. The bottom region displays the energy storage system evolutions during the simulation.

The Power and Energy Estimator supports the next features: the user is able to create, load and save scenarios in an easy way. Scenarios can be replayed in order to study the power consumption behavior depending on the selected parameters.

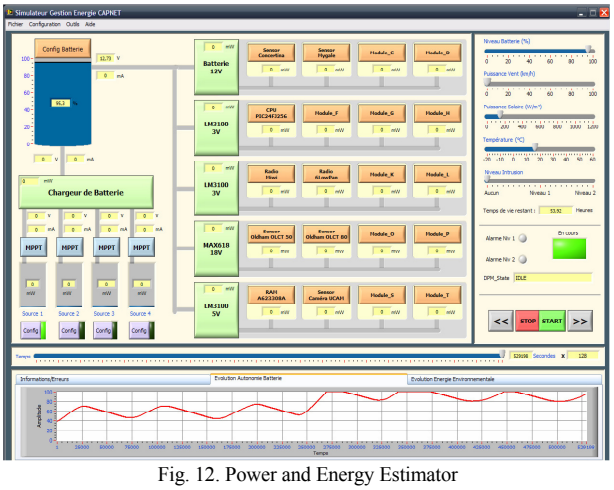


Fig. 12. Power and Energy Estimator

The hardware configuration is fully generic: the user defines the node architecture and specifies the power supply chain: harvesters, MPPT chips if any, battery

Copyright (c) 2013 Advances in Computer Science: an International Journal. All Rights Reserved.

charger, the battery, and the DC/DC converters. Then, he can plug the different equipments per power stage. Parameters like battery type, wind speed, CPU type, etc. are modified through user interface forms. Scenarios record events which chronologically update the different parameters (measures, alerts, external events...). These updates are reflected directly in simulation by the node energy consumption. The estimator is able to monitor and display different component consumptions (sensors, RF, microcontroller...).

The total amount of power consumption and the battery autonomy is summarized. Information is updated in real-time as the scenario is played. The dataset is automatically saved and translated to produce graphs. The power estimator will evaluate the real node power consumption and give a prediction of the complete discharge time depending on the power management policy. The simulator records all errors and important actions in a log file for later analysis. And a help file with tutorials and examples will be provided to help novice users.

6.1 Simulator concept

The simulator acts as a solver by relying on discrete time slots to estimate the power consumption of several system components. The process is to read environmental models first, the scenario and check all rules to determine the impacting events. The highest priority event is forwarded to the Dynamic Power Management. The weather model is read and environment values are dispatched to the corresponding harvester. Sun irradiance parameter is dispatched to the solar panel, wind speed and direction are forwarded to the wind charger, and temperature to the thermopile and the overall system temperature. The radio RF activity is linked to the associated hardware radio component model.

The DPM is started for the current time slot; it is responsible to hold all the current state parameters. The DPM interprets the current state and process the event which can activate a new energy state and the associated processes (sensor awake, mode change...) which modify the hardware parameters. After that, the hardware level process is started to solve the power consumption models. The power consumption is evaluated by using the electric laws. Thus, the tension (U), the intensity (I), the power (P), and the value (V) are evaluated for each component. We define the quadruplet U; I; P; V as the minimal common results for all the devices. A tree hierarchic structure describes the power architecture from the furthest away components (sensors, harvesters) to the closest (the battery). This is to respect the propagate results from input to outputs. Thus, the battery level can be dynamically evaluated by solving its model which requires:

$$I_{ch\arg e} - I_{disch\arg e}$$

Each model is estimated one by one by executing its model algorithm, the input parameters have been updated by the DPM, at the previous iteration. The output parameters are filled for the next iteration. The process is repeated each iteration.

Another simulator like IDEA1 [30] used an interesting optimization approach, which consist to compute only power state when they changed, this reduces overall time computation. But in practice, as we handle environmental variation, the power state variation is updated at each time slot. Thus, we do not rely on this approach to optimize computational speed. In counterpart, most external parameters are known. For scenario, weather, and RF communications with the exception of weather forecast, we can pre-calculate most models input data before starting. This increase the overall simulator performance. Globally, the simulator needs 15 minutes, on a Core2 2.5GHz PC, to simulate one node with all equipments during 7 days.

6.2 Simulator implementation

The simulator has been developed with Labview from National Instruments. It combines a graphical language with massively parallel programming and object paradigm support.

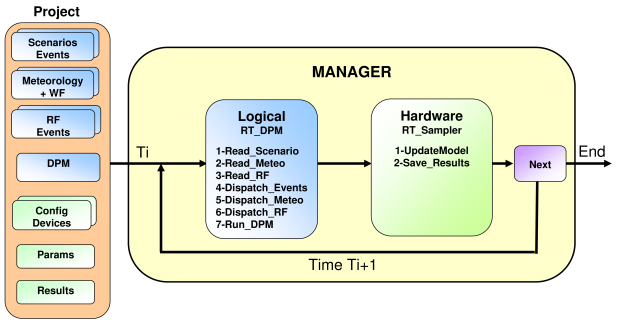


Fig. 13. Simulator core engine

The simulator is based on the global node model presented in Figure 13. Thus, the simulator engine is composed of the two layers. First, the power management policy layer reacts and describes DPM activities. Second, the physical layer includes all the hardware component models.

Copyright (c) 2013 Advances in Computer Science: an International Journal. All Rights Reserved.

The DPM layer models the PMS by using a FSM defined by a state-transition schema. The DPM follows the scenario which defines the interaction with the internal and external events. The policy decides the behavior to adopt, and it updates the device states/parameters accordingly to the hardware configuration.

The Hardware layer is based on node components and their characteristics. Each component or peripheral is called a Device. Devices are classified according to their types: DC/DC Converters, Batteries, Micro-sources, CPU, Radio, Sensors as specified by the hardware model. Each device is modeled by a class specialization that defines the device behaviors in terms of power consumption. A Configuration is a set of channels on which devices are mapped. Each device class embeds the properties, the parameters, and the methods to evaluate and update its model.

The power simulator is managed by the "Manager" which commands the entire simulation evolution. The manager controls playing/recording of scenarios by using a discrete time representation which divides time into time-slice quanta. The manager task is to apply the simulator process which synchronizes both layers [1]. A more complete process description has been published in [20].

6.1 Simulator Results

The result files contain data over time which can be directly analyzed to present the results. Results are grouped into sections: Sensors, Micro-sources, MPPT, DC/DC, CPU/RAM, and Radio components. For each component, the simulator provides the intensity, the voltage, the power consumption, and a component specific value (sensor value, battery SOC...) curves. Thus, the maximum power peaks can be used to study the impact on the battery for high discharge sequences. The total harvested energy and the total consumed energy is also determined by integrating the power over time. The energy directly relies on the node autonomy which are presented in Watt-hours for section results.

[1] Reader must notice that the simulator accuracy (over time) is dependent of the time slot granularity. and the simulator results accuracy over value depends on model standard deviation, but also the error combining is not the sum of all the model errors. We does not explore much on error combining but we search most to know if the simulator results converge or not which is described in experimental tests section. Error granularity (i.e. near 10%) and time accuracy (1s) can also be chosen in accordance on simulation performance.

7. Experiment Hardware Prototype versus Software Simulator

We have built the Power and Energy Estimator (PEE) to simulate the whole node power and energy consumption. We want now to compare the simulator against the real demonstrator through experimentations.

7.1 Test Strategy

The test strategy developed here has been to verify the power consumption of the root node for the nominal scenario "Boil over - Firemen". Thus, an initial test campaign has been launched to test, explore and optimize the CAPNET system. Tests have been launched sequentially, and corrections were applied from feedback acquire on each test.

In test 1, the system embeds the A512 battery, the solar panel (10W), and the wind harvester. We obtain strange results that clearly showed we are far of the energy recharge expected. Thus, we restart the test again in same condition. In the test 2, the wind generator has been identified to not provide all the power.

Thus, we launched test 3 with the battery and the wind harvester only, then after 1 minute the solar panel has been added too. These experiments have thrown several problems on the power supply chain. It also underlines battery lack in term of the wished autonomy of 7 days. Thus, industrials request to enhance the power stage to be able to keep a 2Ah battery for size constraint, whereas maintaining up to 7 days of work around. Thus, we have readjusted and optimized the power supply chain; and validated once again the new system.

Once technical tests were passed, we start the validation part. The first validation experiment aims at establishing the typical node power consumption without external energy. This test will give us the detailed power consumption map by devices.

The Second validation step is the same but with harvesting feature enabled. This test gives the first clue of the global node autonomy in real situation. And thus, we can validate if power recharge is well-scaled or not for the localization.

These experiments showed that simulated power consumption results and recharge rate results match demonstrator. Finally, we validate the simulator global energy balance to verify its coherency. Thus now, let us see in detail the preparation of the experiments.

Copyright (c) 2013 Advances in Computer Science: an International Journal. All Rights Reserved.

7.2 Prepare

These tests have been together evaluated on software simulator and the hardware final nodes prototypes. The power estimator simulator used the release 1.4. The simulator set-up consists to define the entire models: the project is in automatic mode with 7 days duration. It used the file TrameRF.rf given by the RF simulator developed by IETR. The DPM used is the one defined to handle the Boil over scenario, with the exception of sensor frequency dynamic changes (x4) in low Battery Power mode. Weather has been extracted from the nearest weather station (from meteorology database) with a refresh period of 30 minutes. The business scenario used is the Boil over, described previously, with perimeters surveillance and toxic sensors with 7 days duration. The scenario events hypotheses are: Simulated Sensors 1, then Sensor 2 activation (confirmation sensor) for periodic intrusion starting at T=10, Period=4320, Duration=14 seconds, 139 Repeats. Toxic gas sensor 1 is activated each 20 minutes for 30 seconds, then gas Sensor 2 measurement is activated each 20 minutes for 30 seconds, 503 Repeats. An image grab is requested every hour with a 10 second image capture, repeated 167 times. The hardware configuration has been described in section 3, and will be used for this experiment. When hardware test were finished, the battery was discharged entirely in order to measure the exact state of charge.

Once the simulator has been prepared, a simulation can be started. This preparation is repeated for each test run.

7.3 Experiments

Test n°1 campaign takes place from 04 to 07 November 2011 in Lorient, France. The node embedded the 10W solar panel with an Rutland 504 wind generator. After running the test, the real battery duration has been measured at 3 days whereas the simulated autonomy has predicted to run 7 days without drop down under 92% of battery state of charge. Thus, this is an important gap between the real demonstrator and the simulator.

By studying the demonstrator result, we notice that the solar panel clamps easily at 16.2 Volts which corresponds to the voltage limit fixed (16.5V) before the solar LT3652. Returns to zero peaks are probably due to clouded sky which stalled episodically the solar panel power. The wind seems to have few impacts on the recharge whereas it is highly expected to outperform the solar harvested energy. Thus, this particular behavior drives us to continue the tests measurements.

Test 1 has also enlightened that the demonstrator was using
a charge on sensor 1 similar to a SPI 108 sensor rather than the SPI 107 expected. Thus, we had updated the power consumption map of different components:
- SPI 107 is measured at 16mA@12V
- Radio1 is measured at 12mA@12V and 18°C
- Le S912 has been re-evaluated at 58mA@12V
- µCam is 24mA@12V at 18°C
- OLTC50-80 have correct values after readjusting the comparison at 18°C

This enable us to adjust the real sensor charge at the nearest for the next tests.
Test 2 has been measured from 10 to 17 November 2011 at Lorient, France. The node embeds the 10W solar panel and the Rutland wind generator. After changing for a true SPI 107, not a SPI 108 gave for a 107, the battery autonomy has been measured for 7 days duration with the demonstrator.
And the battery autonomy has been estimated for 7 days for the simulator.

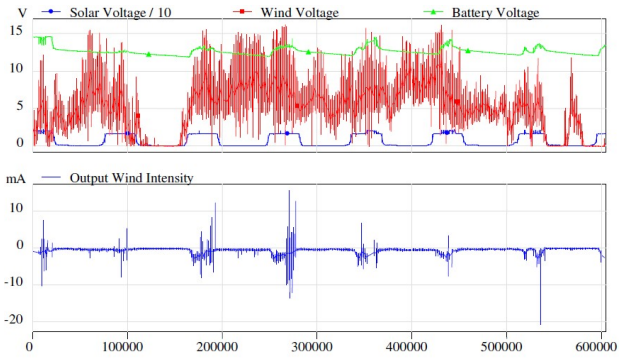


Fig. 14. Test2 Result Curves

Figure 14, we can see that the battery never drop down below 10V and matches the objective of keeping the system alive during 7 days. But it's true that the battery curve looks more smoothed on the simulator. Here again, a strange behavior, the wind does not seems to have an impact on the recharge part (i). Therefore, we decide to launch a dedicated test with the wind generator only, then with solar activation later after a delay.

Test 3 takes place from the 28 to 29 November 2011 at Lorient, France. At start, the node embeds only the Rutland harvester, then system is shutdown during the solar panel connection and enabled again. This cause measurement disconnection during this step. For this test, the battery is discharged and starts at 8 Volts which is a deep discharge state. Figure 15 shows the wind intensity, the solar voltage and the battery voltage curves.

Copyright (c) 2013 Advances in Computer Science: an International Journal. All Rights Reserved.

In the first part where there is only the wind generator, we can see that it provides at best only 2W in the best cases with a wind burst speed over 30 Km/h (ii). The energy profile of this energy is very variable; this is a succession of peaks with large empty areas. In this case, the recharge is almost variable and difficult but at final, it recharges slowly the battery at 25% state of charge in 2 hours despite the heavy sensors charge. In the second part, the solar panel is added. The power curve is here more intensive due to the weather but it is always limited at 2 Watts (ii). The solar panel voltage is nearly constant except when cloudy sky drops the efficiency but this almost has not very impact on a well charged battery. At final, the solar recharge offered a more stable and the fastest recharge rate for the node.

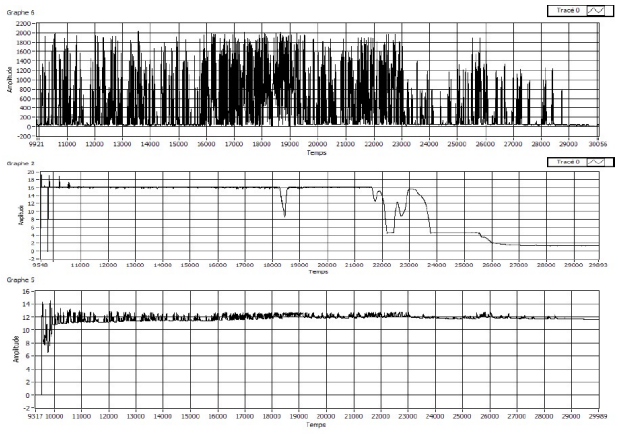


Fig. 15. Test3 Result Curves

In all the tests, we notice that once discharged, the battery never reach again its maximal voltage at the floating voltage 13.8V. The battery maximum recharge limit reaches 12 Volts (iii). This behavior has been identified from the battery charger which uses 2 phases constant charging. Most advanced charger offering 3 steps charging with boost feature, where a complete recharge can be expected.

Remarks (i), (ii) and (iii) drive us to optimize and enhance the power supply chain. Firstly, supercapacitor price/interest could be discussed in regard of the battery capacity and peaks power charge which must be handled. Secondly, the power supply chain has been optimized to reduce the number of stages. Indeed, the LT3652 components have special features that enable to directly charge the battery. Thus, the MAX669 battery charger has been suppressed; this will skipped one power stage and its 20%-50% power losses. Regarding the Wind harvester, the MPPT threshold regulator used within the board force us to set it with a threshold superior or equal to the battery floating voltage for recharge. Tests showed this is not a pertinent solution to acquire the wind kinetic energy at low speed between 10-30 Km/h. Thus, the final product will embed an enhanced regulator offering an optimal MPPT research in real-time. This one regulates the output voltage so as to maintain the battery floating voltage even when the wind harvester voltage is inferior.
When the new power supply chain has been implemented for the ten nodes, the global validation tests have been launched for power consumption and power recharge measurements.

The first validation Test: Node global power consumption, this test aims to establish the power consumption map for the nominal scenario.
Table 7 shows the energy and power consumption results independently from recharge and group the result by categories. The average node power consumption for nominal scenario is 586mW. In this WSN application, the first surprise comes from the PIC which has low impact on the consumption with 6.689%. We have also noticed that the RAM should have a low 2.52% consumption due to shutdown mode. The DC/DC converters have a high efficiency ratio and their impact depends more on the connected devices. But their cumulated power consumption is high and represents 22.20%. The Radio1 power consumption is also not negligible: it affects 20.43% of the consumption due to permanent Rx-listening.

Table 7: Simulation Results

Component	Min	Max	Avg	Energy	Ratio
SPI107	193.622	224.008	223.703	123073	38.14%
Concertina	0	458.518	4.769	2623	0.813%
MicroCam	0	305	4.127	2270	0.703%
OLCT 50	0	1070	27.59	15174	4.70%
OLCT 80	0	874.085	22.294	12261	3.80%
PIC24F256	38.4	50.4	39.233	21578	6.69%
M48T35AV	0	207.26	14.788	8133	2.52%
Radio 1	118.821	176.771	119.771	65874	20.42%
Radio 2	0	176.765	0.041065	22.586	0.007%
DCDCs	0	/	130.19	71605	22.20%
Others	0	0	0	0	/
Total Sensors	-		282.483	155366	48.16%
Total DC/DC	-		130.19	71605	22.20%
Total Radios	-		119.812	65897	20.43%
Total CPU/RAM	-		54.021	29712	9.209%
Total Cons.			586.506	322578	100%

In the test 2: Global battery recharge feature Test - table 8, during a period with low sunset and average wind power, a node has been tested for its battery recovery. This test evaluates mainly the harvesters and the power supply chain to verify the node recharge capability. At the end,

Copyright (c) 2013 Advances in Computer Science: an International Journal. All Rights Reserved.

the battery has been partially recharged during a test of 17500 seconds (5 hours).

Table 8: Recharging Feature Comparison between Simulation and Demonstrator

Feature	Demonstrator Results	Simulator v1.4 Results
Consumption during 5 h	2.7 Wh (9720 J)	2.3 Wh (8269 J)
Energy from wind gen.	10.7 Wh	9.36 Wh (33690 J)
Energy from solar	n/m	20.79 Wh (74834 J)
Real Energy harvested	n/m	11.51 Wh (41428 J)
Battery SOC at start	0 Wh (0%)	0 Wh (0%)
Battery SOC at the end	**9.6 Wh (40%)**	**9.20 Wh (38.38%)**

In this test, simulator and demonstrator show very nearest results for the global battery recharge time. But we must take care about weather model because curve are really smoothest compared to real one. Thus, results are better as long as weather assume small variations in average. At final for the project, simulation results have been considered significant for the need of the WSN recharge modeling.
Last test presented in this section is the simulator energy balance in Table 9.

Table 9: Energetic Balance

Results	Energy (Joules)	Energy (Wh)
Battery 2Ah	86400 Joules	24 Wh
Harvested energy after the charger	322421 Joules	89.562Wh
Harvested energy for the battery	310870 Joules	86.353Wh
Harvested energy for direct use	11151 Joules	3.209Wh
Total Energy Consumption	322578 Joules	89.605 Wh

Thus, the balance between the harvested energy and the total energy consumption is almost neutral, despite a few deltas on the harvested energy. This phenomena has not been explained but can be due to a propagate accumulation error effect.

7.4 Discussion

This part wants to address the differences between the demonstrator (real hardware) and the simulator. By examining the results of Table 3 and Table 7, we can extract some conclusions. When we study the system equipment, we can see that the battery system has a capacity of 2Ah. The highest consuming element (i) is the professional Sensor 1 with 223.7mW average in Sleep, besides all sensors cumulated reach 48% of the total consumption (72.58% in Test1). This is explained by analysis of the DPM whose Sensor 1 runtime is permanently on. This limitation cannot be addressed easily because the monitoring requires constant listening and depends more on manufacturer technology. Currently, some manufacturers do not provide low power equipment; however it would be of interest to encourage most of them to follow this direction.

Some experience returns are commented below. First, the wind generator is far behind its theory 60W energy profile. It was at best equivalent to a 2W generator with a local weather. The simulator model is also over-dimensioned because it is based on the manufacturer datasheet; this is due to the lack of a wind tunnel for experiments. When studying the real wind profile, wind appears very variable and fluctuant: it represents a noisy signal including large zero areas due to wind speed limit before starting. And finally, only the average wind speed is considered in the simulator even if we handle the minimal speed limit (behind 10 Km/h) before harvester starts.

Another special point is the weather data accuracy model. In fact, professional weather station as a periodic refresh delay (30 minutes for the test station). With wind variations with boost wind speed, this highly smooth the signal compared to real wind signal. Again the solar irradiation is estimation and keeps an average depending of the sky local cloudiness. So, it would be easy to conclude that it will be preferable to use a local weather model based on real sensors [17].

This approach is interesting but it is not recommended since a local model will never be able to predict perturbations as forecasters can do with satellites. In counterpart, global view models are unable to catch locality effects if they are under resolution. Thus, compromise could be to exploit WFs meteorology and minimize delta errors with local parameters based on real measurements. In every case, due to the Lorenz effect [32], it is difficult to obtain relevant predictions over time at fine granularity. Finally, for a system with harvesters and a battery power system, we can postulate that the total energy balance of a system is a constant which can be synthesized as:

$$Bat_{SOC+1} = Bat_{SOC} + NRJ_{harvested} - NRJ_{consumed} + \varepsilon \quad \textbf{(29)}$$

Nevertheless, we cannot summarize the power consumption model only by this equation because batteries have special behaviors [26]. Especially, batteries have charge and discharge constraints inherent to their technology. Some accumulators must be charged with a maximum C/10 rate (lead-acid battery, sealed battery) or 1C rate (Li-ion, Li-Po) but in practice Lithium battery are charged in the step between 0.2C and 1C range. This maximum recharge current limits the power supply chain in most cases if harvesters can provide more power. In this case, we have noticed that the total energy harvested cannot be totally stored or used. Thus, this mean that lot

Copyright (c) 2013 Advances in Computer Science: an International Journal. All Rights Reserved.

of power is wasted and unexploited by the system. This is particularly true when battery reaches its full state of charge level. Here, the system consumes at most its nominal mode power consumption and wastes nearly all environmental energy.

To handle these energy opportunities, due to battery charging constraint and full charge battery, several solutions can be explored. First, a design stage helping the battery by joining supercapacitors can provide an energy buffer to keep energy otherwise wasted. This is an already known and used solution in the CAPNET project. Drawback for this solution comes from the capacitor recharge duration which can be low in comparison on a sunset exposure. Secondly, the system can use the dynamic feature of the DPM to adjust its operation to enter "extra"-positive power modes. These new modes can be used to achieve better QoS, better performance, better RF transmission quality and distance, etc. Drawback implies a system able to monitor the harvested energy, the power consumption, and the battery state of charge at runtime. Unfortunately the monitoring will add components and complexity, but can offer newer DPM modes.

As described, the design of Wireless Sensor Networks is a challenge because the design constantly evolves. Thus, to handle the change, we reevaluate the system with the simulator. This one has enabled to estimate and experiment most energetic decisions relevance. And finally, the simulator and the prototype has helped to identify major power hotspots:
(1) detection of heavy sensors consumption
(2) micro-sources energy profiling (for deployment)
(3) harvesters scaling properly
(4) power supply chain optimizations
(5) validation of the battery total autonomy

In conclusion, the simulator has given relevant results for the system test and validation, and deployment considerations. It features the fine tuning of component power consumption providing the average power consumption, and underline problems concerning major elements. It will be interesting to incorporate real-time measurement points for intensity and voltage weather and hardware monitoring. This idea has not been retained for the system specification, but it really appeared that the direct measurements are key values for real hardware monitoring. This will enabled to debug more deeply into the system and handle DPM with dynamically acquired weather data. Moreover, this work has enlightened new research ways to explore extra capacity recharge to handle positive power modes, not only degraded ones. Finally, the CAPNET nodes are under production by our partner industrial which is bringing CAPNET concept into industrial process.

8. Conclusion

In this article, we have emphasized the benefits of estimating design evolutions in a low power design methodology. Thus, we have recalled the need to use Low Power for industrial embedded systems and WSNs production. This is done by using an appropriate design methodology which takes care of overhead all along the process. We explain how to transform the initial concept idea into the final product by using a progressive abstract representation. This abstract representation can be modeled by a tool such as the Power and Energy Estimator developed for the CAPNET project. It validates if the specified requirements and constraints of the concept will matched on the implemented features. The key outcome of our proposition is to handle project evolutions by rebuilding the abstract representation and its accuracy as far as the project progress. Avoiding the risk of divergence by evaluating structural modifications at start rather than at the end. Thus, for the CAPNET project, we present the design and the building of an autonomous embedded system. This is done by using the FLPA methodology on hardware components, and by developing dedicated logical models for environments, RF and DPM. Then, we present model characterization results and errors for various component of a WSNs node. We have also described how to build the Power and Energy Estimator as a time-slot model solver, enabling the reproduction of business scenarios to validate the design. The estimator error has reached 2% error with the demonstrator at final with the last models.

We made experiments by comparing the final WSN demonstrator composed with ten nodes against the abstract simulated results. The key verified elements where: the simulator ability to handle recharging WSNs to predict node autonomy in a given scenario, the capacity to identify power bottlenecks such heavy sensor energy consumption, the capacity to calibrate the total autonomy of the battery and to scale the harvesters accordingly, and at final, the experimental result campaign confirmed us in the convergence of this approach which is by far the most valuable. The results have shown that the environmental energy is rarely fully exploited. There are lot of wasted energy due to battery constraints, which could offered energy gain opportunities to exploit advanced extra power modes.

Copyright (c) 2013 Advances in Computer Science: an International Journal. All Rights Reserved.

The CAPNET project has been finished at time by using this approach. Now, the prototype is being implemented as industrial product by industrial partner for firemen delivery.
The simulator is freely available for research purpose and has been copyright protected for commercial exploitation.
Thanks to this experience, we develop new ideas to enhance an embedded system power stage alimentation to increase its efficiency. Future works will be to explore deeper electronic embedded system for person home aids. One main difficulty will be to adapt the embedded system dynamically in accordance of the human problematic.

Acknowledgments

We would like to thank the CAPNET project partners*: Fire Service partners: SDIS29 (Service Départemental d'Incendie et de Secours du Finistère)*. Industrial partners: *ERYMA Security Systems (Leader), DeltaDore, AtlanticRF*. Academics partners: *I.E.T.R Rennes, Lab-STICC Lorient.* Financial pole: "*Pôle Image et Réseaux*", "*Conseil général du Morbihan*", "*Région Bretagne*", and "*Ministère Français de l'Economie, des Finances et de l'Industrie*". Spell-checker: Nathalie Julien.

References

[1] Idris, M.Y.I., E.M. Tamil, N.M. Noor, Z. Razak and K.W. Fong, "Parking guidance system utilizing wireless sensor network and ultrasonic sensor", Information Technology Journal 8 (2): 138-146, (2009).
[2] Pai H. Chou, Chulsung Park, "Energy-Efficient Platform Designs for Real-World Wireless Sensing Applications", In Proceedings of the 2005 IEEE/ACM International conference on Computer-aided design, Irvine,
CA, (2005).
[3] F.J. Rincon, M. Paselli, J. Recas, Q. Zhao, M. Sanchez-Elez, D. Atienza, J. Penders, G. De Micheli, "OS-Based Sensor Node Platform and Energy Estimation Model for Health-Care Wireless Sensor Networks", Proceedings of Design, Automation and Test in Europe, Munich, Germany,
(2008).
[4] http://en.wikipedia.org/wiki/List_of_wireless_sensor_-nodes, Wikipedia, List of wireless sensor nodes, last visited 6/09/2012
[5] Yuen-Hui Chee, Mike Koplow, Michael Mark, Nathan Pletcher, Mike Seeman, Fred Burghardt, Dan Steingart, Jan Rabaey, Paul Wright, Seth Sanders, "PicoCube: A 1cm3 Sensor Node Powered by Harvested Energy", Proceedings of the 45th annual Design Automation Conference, California, (2008).
[6] Xiaofan Jiang, Joseph Polastre, and David Culler, "Perpetual Environmentally Powered Sensor Networks", Prometheus Project, Proceedings of the 4th international symposium on Information processing in sensor networks (2005).
[7] Vijay Raghunathan, Aman Kansal, Jason Hsu, Jonathan Friedman, and Mani Srivastava, "Design Considerations for Solar Energy Harvesting Wireless", Proceedings of the 4th international symposium on Information processing in sensor networks, (2005).
[8] Min Chen, Gabriel A. Rincón-Mora, "Single Inductor, Multiple Input, Multiple Output (SIMIMO) Power Mixer-Charger-Supply System", Proceedings of the International Symposium on Low power electronics and design (2007).
[9] Chulsung Park, Pai H. Chou, "AmbiMax: Autonomous Energy Harvesting Platform for Multi-Supply Wireless Sensor Nodes", AmbiMax Project, Proceedings of the international IEEE SECON, (2006).
[10] Farhan I. Simjee, Pai H. Chou, "Everlast: Longlife Supercapacitoroperated wireless Sensor Node", Proceedings of the international symposium on Low power electronics and design (2006).
[11] Sandy Irani, Sandeep Shuklay, Rajesh Gupta, "Algorithms for Power Savings", Proceedings of the fourteenth annual ACM-SIAM symposium on Discrete algorithms, Baltimore, Maryland, p37-46, (2003).
[12] Massoud Pedram, Jan Rabaey, "Power-Aware Design Methodologies", Kluwer Academic Publishers, OSBN, 1-4020-7152-3, (2002).
[13] Marcus T. Schmitz, Bashir M. Al-Hashimi, Petru Eles, "System-Level Design Techniques for Energy-Efficient Embedded Systems" Kluwer Academic Publisher, ISBN 1402077505, 2003 [14] R.J.M. Vullers, R. van Schaijk, I. Doms, C. Van Hoof, R. Mertens, "Micropower energy harvesting", Solid-State Elec. 53, p.684-693, (2009).
[15] Luca Benini, Alessandro Bogliolo, Giovanni De Micheli, Fellow, "A Survey of Design Techniques for System-Level Dynamic Power Management", IEEE Transactions on Very Large Scale Integration (VSLI) Systems, Vol. 8, No. 3, (2000).
[16] Ravishankar Rao, Sarma Vrudhula, Daler N. Rakhmatov, "Battery modeling for energy aware system design", Computer, vol. 36, no. 12, pp.77-87 (2003).
[17] Yong Yang, Lu Su, Yan Gao, Tarek F. Abdelzaher, "SolarCode: Utilizing Erasure Codes for Reliable Data Delivery in Solar-powered Wireless Sensor Networks" Proceedings IEEE INFOCOM, San Diego, California, USA, 2010.
[18] Eric Senn, Johann Laurent, Nathalie Julien, Eric Martin, "SoftExplorer: Estimation, Characterization, and Optimization of the Power and Energy Consumption at the Algorithmic Level." InternationalWorkshop on Power

Copyright (c) 2013 Advances in Computer Science: an International Journal. All Rights Reserved.

and Timing Modeling Optimization and Simulation (PATMOS), p342-351, (2004).
[19] Saadia Dhouib, Eric Senn, Jean-Philippe Diguet, Johann Laurent, Dominique Blouin, “Model Driven High-level Power Estimation of Embedded Operating Systems Communication Services”, IEEE 2009 International Conferences on Embedded Software and Systems, Hangzhu, Chine, (2009).
[20] Nicolas Ferry, Sylvain Ducloyer, Nathalie Julien, Dominique Jutel, "Power/energy estimator for designing WSN nodes with ambient energy harvesting feature" EURASIP Journal on Embedded Systems, Hindawi Publishing Corp, Volume 2011
[21] Nathalie Julien, Johann Laurent, Eric Senn, Eric Martin, "Power consumption modeling and characterization of the TI C6201", IEEE Micro Volume 23, Issue 5. p40-49, (2003).
[22] Meteociel, http://www.meteociel.fr, last visited 15/02/2013
[23] The European JRC - Photovoltaic Geographical Information System, http://sunbird.jrc.it/pvgis/, last visited 15/02/2013
[24] Meteo-France, http://france.meteofrance.com/, last visited 15/02/2013
[25] Sandy Irani, Sandeep Shukla, Rajesh Gupta, ”Online Strategies for Dynamic Power Management in Systems with Multiple Power-Saving States”, ACM Transactions on Embedded Computing Systems, Vol. 2, No. 3, p 325-346, (2003).
[26] Luca Benini, Alberto Macii, Enrico Macii, Massimo Poncino, “Discharge Current Steering for Battery Lifetime Optimization”, Proceedings of the Int. symposium on Low power electronics and design, (2002).
[27] O. Erdinc, B. Vural, M. Uzunoglu, "A dynamic lithium-ion battery model considering the effects of temperature and capacity fading", International Conference on Clean Electrical Power, (2009).
[28] Lijun Gao, Shengyi Liu, Roger A. Dougal, "Dynamic Lithium-Ion Battery Model for System Simulation", IEEE Transactions on Components and Packaing Technologies, Vol. 25, No. 3, (2002).
[29] Nicolas Ferry, Sylvain Ducloyer, Nathalie Julien, Dominique Jutel, "Fast electrical battery model builder for embedded systems", Conference on Faible Tension Faible Consommation (FTFC), 2011 DOI: 10.1109 / FTFC.2011.5948915, p47-50, 2011.
[30] Wan Du, Fabien Mieyeville, David Navarro, Ian O’Connor, "IDEA1: A validated SystemC-based system-level design and simulation environment for wireless sensor networks." EURASIP J. Wireless Comm. And Networking, 2011
[31] jean-Marie Floch, Jean Michel Denouval, Yvan kokar, "Design of Dual Beam Printed Dipole Antenna", Proceeding of the 5th European Conference on Antennas and Propagation (EUCAP), 2011.
[32] Edward N. Lorenz, "The Essence of Chaos", Book, University of Washington Press, 227 pages, 1995

Copyright (c) 2013 Advances in Computer Science: an International Journal. All Rights Reserved.