Smart Coulomb Counter for self-metering wireless sensor nodes consumption

Leonardo Steinfeld, Julián Oreggioni, Diego Andrés Bouvier, Carlos Andrés Fernández and Jorge Villaverde

***Abstract*—The design of a node of a wireless sensor network is a challenging task, as it is expected to be small, reliable and low cost, using hardware that is characterized by limited resources, including energy, computation power, memory and communication bandwidth. In order to save the valuable energy it would be instrumental to measure the energy consumption by every deployed node on the field. Prospective applications include, energy-aware routing and hardware diagnosis for detecting subsystems' faults or unusual power consumption. In this work we propose a measurement method and circuit, named Smart Coulomb Counter (SCC), that easily adds to a sensor node the capability of measuring its own energy consumption. This hardware-based approach consists in duty-cycling the measurement circuit together with the sensor node, thus lowering its power consumption. Experimental results demonstrates that SCC has a very low power consumption, ensuring an almost negligible impact in the battery lifetime. Other advantages of SCC include high dynamic range, high linear range, very low temperature drift, independence of the power supply, easy to integrate and low resource requirements in terms of memory and processor usage, as well as in terms of component count (area and cost). A fully-functional prototype based on the sensor node *TelosB* running ContikiOS was developed. The main features have been tested on the field and the advantages of our system were demonstrated in real conditions. Finally, a comparison between our solution and a software-based approach (Energest) is presented.**

***Index Terms*—energy measurement, charge counter, coulomb counter, energy efficiency assessment, low-power design, wireless sensor networks.**

I. Introduction

The Internet of Things (IoT) includes networked embedded systems which involve smart objects that embed computation and sensing in the physical world, enabling an unprecedented spectrum of applications in several fields of daily life, such as environmental monitoring, waste management, home automation, cattle management, elderly care, continuous care and medicine to name a few. Wireless sensor networks (WSNs) are a subclass of IoT, and usually comprises of nodes which sense, process the acquired data and communicates information wirelessly to other nodes. The design of a sensor node presents challenging requirements, as they are expected to be small, reliable and low cost, resulting in an extreme scarcity of resources, including energy, computation power, memory and communication bandwidth. Reduced energy consumption is one of the major goals in WSNs, since it determines the lifetime of sensor nodes when they are powered from batteries, or dictates strict requirements on the harvesting system when the nodes scavenges scarce energy from the environment.

The authors are with the Departamento de Electrónica, Facultad de Ingeniería, Universidad de la República, Montevideo, Uruguay.

The information of the node energy consumption is important not only for design-space exploration [1] and optimization [2] during the laboratory design stage, but also to enable on-field adaptive power management. In fact, real-time measurement of the energy consumption of sensor nodes on the field could have a major impact on developing wireless sensor networks. For instance, the energy-profile information, available readily to nodes, enables novel applications of interest, e.g. perform adaptive sampling [3], asses on the fly the energy efficiency of communication protocols [4], [5], predict the remaining battery charge for routing decisions [6], [7], [8] or perform hardware diagnosis for detecting subsystems faults or unusual power consumptions [9].

On-field energy measurement of sensor nodes presents many challenges. Firstly, the limited energy available prevents adopting most of previously proposed methods, since they were designed for laboratory operation. The energy consumption due to the measurement should be kept as low as possible to maintain the battery lifetime of the sensor node virtually unaltered (e.g. above 95% of the original duration without the measurement plug-in). Secondly, the measuring cost should not be significant in order to keep sensor nodes affordable. Thirdly, the measuring method should be non intrusive, so that large data manipulation or calculations should be avoided to obtain the measured data. Furthermore, the measurement has to be precise and reliable under harsh conditions. In particular, sensor nodes are exposed to a wide temperature variation, thus the measurement should not be affected by temperature changes. Finally, one of the most challenging requirements is to have a five decades dynamic range, as the current varies from a few microamperes (sleep mode) to tens of milliamperes (computing, transmitting or receiving) [10], [11].

On-field energy measurement of sensor nodes can be implemented following a hardware-based approach or a software-based one. The hardware-based approach provides a more accurate and reliable measure at an additional economic cost and increased complexity. The software-based approach is simple and has no additional cost, but it has three main drawbacks when it is adopted in real-life deployments. One disadvantage is that each sensor node should be individually calibrated. The second disadvantage is that even if each sensor is calibrated, the estimation error could be significant since the actual power level has deviations from the power measured in the laboratory due to temperature or battery voltage variation. Despite the fact that the accuracy can be improved by means of complex modeling (considering the individual calibration and temperature and/or battery voltage variation), the computation power needed would prevent its adoption for

in-node application, since it will increment dramatically the energy consumption. The third drawback is that any failure or behavior involving an increase of power consumption that was not properly modeled will not be reported by the software-based approach. Therefore, hardware-based approaches enable novel applications like hardware diagnosis that are not possible using a software approach.

It is worth emphasising that despite the sleep power consumption of a sensor node is several orders of magnitude lower than the active power, the sleep energy consumption represents more than 10% of node energy drain [12], [13]. Energy efficient MAC protocols keep the radio in reception or transmission mode below 1% [14], [15], and thus the node spends the majority of the time in sleep mode [16], [17]. Hence, the sleep energy contribution to the whole system must be considered in any energy measurement system.

This work presents three main contributions. Firstly, we propose a hardware-based approach, named Smart Coulomb Counter or SCC, to add to a sensor node the capability of measuring its own on-field energy consumption, featuring negligible power consumption, high dynamic range, easy integration and low resource requirements in terms of memory and processor usage, as well as in terms of component count (area and cost). Secondly, we have developed a fully-functional prototype based on the sensor node *TelosB*, running ContikiOS. The main features have been tested and the advantages of our system were demonstrated in real conditions. Finally, a comparison between our solution and a software-based approach (Energest [18]) is presented.

The remainder of this paper is organized as follows: Section II presents a brief overview of the state of the art of the problem that is being addressed, highlighting the main related work. Next, Section III introduces and describes our proposed SCC. Then Section IV presents an example of the application, where the SCC is integrated with a WSN node *TelosB*; and Section V presents the experimental results. Finally, Section VI contains concluding remarks and research directions.

II. Related Work

The power consumption of a sensor node depends on the states of the node components, such as microcontroller, radio, LEDs and so on. A sensor node power state can be defined as the compound state of the mentioned constitutive parts, in which the total power consumption is roughly constant. This fact has been exploited by software-based approaches, in order to estimate the energy consumption of sensor nodes, not only in operating systems e.g. Energest in ContikiOS [18] or Quanto in TinyOS [24], but also in other embedded devices and applications [25]. The energy consumption of a given state during a certain period of time can be estimated multiplying the power level by the elapsed time. The sensor node is characterized in laboratory, prior to the network deployment, to determine the power contribution of each component as a function of its operating mode. For example, sleep and active modes are usually considered for the microcontroller, while for the radio the usual model consists of three different states: idle, transmitting, and receiving. Finally, the node operating on the field measures and accumulates the elapsed time in each state. Then, the sensor node may periodically report the accumulate time values, as in Energest [18], so that the energy contribution of each component and the corresponding state is computed on a server or computer. This method could be extended so that the sensor node can estimate their own energy consumption during operation time. In this case the node must process the time data to compute the drained energy.

A time-based energy estimation is easy to implement and does not require extra hardware to be added to the sensor nodes. It had been previously adopted for energy efficiency assessment and comparison of communication protocols (e.g. [26]). However, this method has several drawbacks when it is adopted in a real-life deployment. One drawback is that each sensor node should be individually characterized for an accurate energy estimation, because the power drain has differences from node to node, up to 10%, as shown by Hurni et al. [27]. Another disadvantage is that even if each sensor was calibrated, the estimation error could be significant. The actual power level has deviations from the measured power at laboratory due to temperature or battery voltage variation. In case that the estimation is performed in a computer, the accuracy could be improved by means of complex modeling taking into account the individual calibration and the temperature and battery voltage variation (also reported by the nodes). However, the computation power needed for adjusting the estimation would prevent its adoption for in-node application, since it will increment the energy consumption of the node.

Direct measurement of energy consumption may overcome the aforementioned disadvantages of time-based or software energy estimation. The measurement method itself determines the magnitude that is directly or indirectly obtained. On the one hand, if the current is measured the power can be calculated multiplying by the supply voltage. Since the battery voltage decreases slowly remaining stable for long periods, the battery voltage can be sampled (with almost no extra hardware) at very low rates and with negligible energy. Then, the energy can be directly computed by integrating the computed power values. On the other hand, if a charge value is obtained from the measurement, the energy is computed multiplying by the voltage supply, while the current can be obtained by differentiating (computing the incremental ratio). Keeping these relations in mind, different methods for energy measurement are presented next.

Energy Bucket [19] is a charge meter for laboratory sensor node energy characterization implemented using a pair of switching capacitors. While one capacitor is feeding the device, the other is charging. In this case, the consumed charge is the number of times that the capacitor switches multiplied by the charge. This method would be easy to adapt for on-field measurement. The node microcontroller must only control the capacitor switches and keep track of the number of times it switches them. However this method has some inherent drawbacks. First, due to the charge-discharge nature, a ripple is introduced in the voltage supply. Second, the switching frequency increases for higher current drain, increasing the microcontroller load and the overall node consumption.

TABLE I
COMPARISON OF WSN NODE CHARGE MEASUREMENT HARDWARE-BASED APPROACHES.

Method	Technique	Cost / Area	Dynamic range	Self-meter / processing load	Low power	On-field	Observations
Energy Bucket [19]	switched cap.	Low	High	Yes / High for high currents	Yes (*)	Yes (*)	Adds ripple
Powerbench [20]	shunt + ADC	High	Low (3 dec.)	No	No	No (high power and cost)	(***)
Aveksha [21]	shunt + 2 amp + 2 ADC	Med.	High (5 dec.)	No	No	No (high power and med. cost)	(***)
MicroPower [22]	shunt + VFC	Med.	High (5 dec.)	No	No	No (high power and med. cost)	(***)
DC/DC [23]	DC-DC switching freq	Low	High (5 dec.)	Yes / Very Low	No (**)	Yes	Requires a DC-DC converter
SCC (this work)	smart sleep state	Low	High (5 dec.)	Yes / Very Low	Yes	Yes	-

(*) Easy to adapt for on-field measurement.
(**) Fails for low duty-cycle and sleep consumption.
(***) Require Extra board and additional power supply.

A shunt resistor is usually adopted as a current sensing element situated between the power supply and the source of power drain. The current consumption is computed as the voltage drop through the resistor multiplied by the resistor value. This method is adopted by Haratcherev et al. [20] in Powerbench for acquiring the power consumption of the nodes in a wireless sensor network deployed in laboratory. An interface board is attached to each node to acquire the voltage drop by an A/D converter (the voltage signal is previously amplified). The output data is collected by an embedded Linux platform (Linksys NSLU2). The current and time resolution obtained were $30\mu A$ and $200\mu s$ respectively. Similarly, Aveksha [21] which is a debug board for event logging and tracing, also performs power monitoring based on a shunt resistor. It uses two instrumentation amplifiers in parallel with different gain to cover the wide dynamic range of current consumption of the sensor nodes. However, it presents high power consumption, since the ADC is continuously working, and poor temporal resolution that prevents capturing power variation of interest.

Alternatively, the voltage drop across the shunt resistor can feed a voltage controlled oscillator operating as a voltage to frequency converter (VFC), constituting a current-to-frequency converter or a current-to-pulse converter. This method is adopted by Jiang et al. [22] in which the consumption data is acquired by the node itself. This approach has a poor temporal resolution if the current is too small. Additionally, because of its relatively high power consumption, the measurement board needs a separated power supply to avoid restricting the sensor autonomy. Moreover, the cost of this solution is such that it prevents its adoption in every network node.

Finally, Dutta et al. [23] presents an interesting method for measuring the node consumption on the field. If the sensor board already has switching DC-DC voltage regulator, the energy meter is "for free" (as states in the work title). This work has three limitations. First, the switching regulator efficiency drops for small currents, this means that for low duty-cycles operation together with very low sleep current consumption, the regulator power consumption contribution is not negligible. The authors has suggested to turn off the regulator off during sleep periods to save power, however this also means to turn off the measurement circuit, and to lose an important fraction of the consumption information. Third, the measurement methods requires calibration and is sensitive to fluctuations in the environmental conditions.

Table I compares the reviewed methods and shows that none of them fulfills all the requirements needed for on-field self-energy metering. Therefore, SCC can be summarized as an innovative proposal that successfully provides a method for energy measurement that can be generally adopted for on-field operation. The initial results based on simulations were presented in [28] and some preliminary experimental results in [29].

III. SYSTEM DESCRIPTION

Considering the challenges, constrains and requirements that were presented in the previous sections, the SCC proposes a method based on a CPC (current-to-pulse converter) for measuring the charge consumption of the sensor node. The first stage of the CPC converts the sensor node current drained into a voltage drop across a shunt resistor. The value of the shunt resistor is limited by the maximum permissible voltage drop, resulting from the maximum current consumption, the minimum device operating voltage and the battery voltage drift. A low resistance selected to fulfill this criteria, in presence of very low current values (i.e. sleep current) will produce a voltage drop that is below the circuit noise level, preventing its measurement [30]. Our novel approach successfully measured the current consumption in the full range of operation without compromising the remaining requirements.

The method has essentially three main characteristics. Firstly, the measurement circuit is duty-cycled together with the sensor node. Secondly, the active current of the sensor node is directly measured using the aforementioned CPC. Finally, the charge consumed during the sleep mode is accounted upon waking-up by means of the same CPC used to measure the active current. The total sleep charge is obtained measuring the recovering charge of a "buffer capacitor" that fed the node in the sleep state. Fig. 1 presents the block diagram of the SCC and how it is integrated with the sensor node.

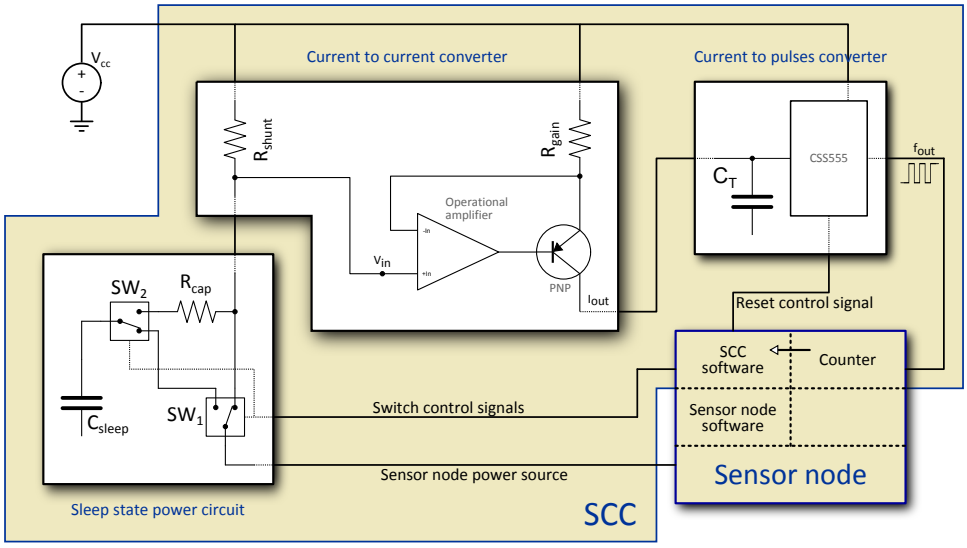

Fig. 1. Block diagram of the SCC showing the integration with the sensor node.

A. Current-to-pulse converter

The CPC is composed of two blocks: one is a current-to-current converter and the other is a current-to-pulse (or current-to-frequency) converter that uses a classic 555 chip, as shown in Fig. 2.

The output of the first stage is:

$$I_{out} = \frac{R_{shunt}}{R_{gain}} I \tag{1}$$

The 555 timing circuit has a resistive divider between V_{cc} and ground formed by R_1, R_2 and R_3 (where $R_1 = R_2 = R_3$). In order to eliminate the dependence between $f(I)$ and V_{cc}, the voltage at the inverting input of $Comp_1$ is fixed by means of voltage reference chip. Therefore the threshold voltages are V_{REF} at the inverting input of $Comp_1$ and $V_{REF}/2$ the non-inverting input of $Comp_2$.

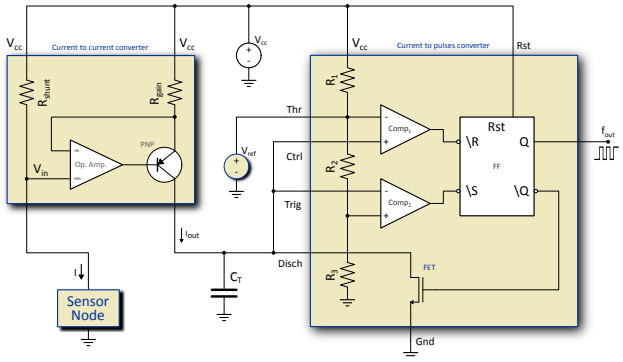

Fig. 2. SCC Block Diagram with 555 internal structure.

The second stage takes the current I_{out} and repeatedly charges and discharges a capacitor C_T from ground to the threshold voltage V_{REF}. The output of the last block responds to this charge/discharge behavior generating a square wave signal (f_{out}) ready to be used by the sensor node.

The frequency of f_{out} expressed as a function of the consumed current I, named $f(I)$, is presented in (2).

$$f(I) = \frac{R_{shunt}}{V_{REF}.R_{gain}.C_T} I \tag{2}$$

where I is the current drained from the batteries by the sensor node, and R_{gain}, R_{shunt}, C_T and V_{REF} are the components previously described.

The square wave signal (f_{out}) is fed to a general purpose timer/counter module in the micro-controller of the sensor node, where each count represents a fixed amount of charge consumed by the sensor node (a *charge quantum* Q). Considering that the charge quantum Q is I/f we obtain the following expression:

$$Q = \frac{V_{REF}.R_{gain}.C_T}{R_{shunt}} \tag{3}$$

B. SCC operation

SCC uses two switches to implement the measurement method: i) direct measurement of the node active charge, and ii) measurement of the node sleep charge by measuring the recovering charge of the buffer capacitor. Switch SW_1 connects the device to the batteries (V_{CC}) or to the buffer capacitor C_{sleep}, and switch SW_2 connects the C_{sleep} capacitor to the device or to the batteries (see Figs. 3 and 4). The resistor R_{cap} limits the charging current of C_{sleep}, so the current is within the CPC input range.

During the node active state the buffer capacitor C_{sleep} is charged. Before the node is set in sleep state, SCC is also set in standby by lowering the reset signal of the 555, to reduce its power consumption. At the time that the node goes to sleep mode, the buffer capacitor C_{sleep} is fully charged. The switches states in sleep mode are shown in Fig. 3.

When the active state is reached, the sensor node power supply is switched from C_{sleep} to the batteries (V_{CC}), and C_{sleep} is switched in order to be recharged and be ready for the next cycle, as shown in Fig. 4. In this scenario, the charge

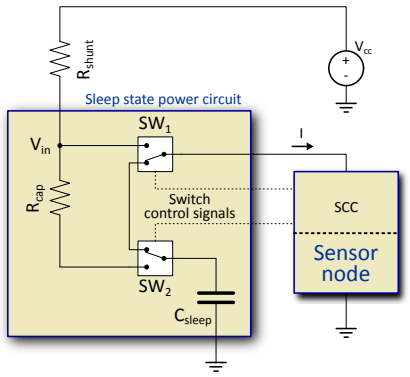

Fig. 3. Sensor node with the SSC included. Sleep State operation.

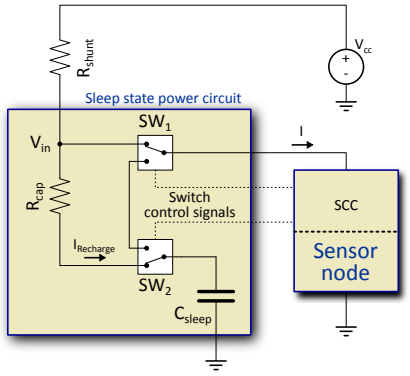

Fig. 4. Sensor node with the SSC included. Measuring the energy consumed in the Sleep State.

consumed during the sleep state is measured when recharging C_{sleep}.

The measuring current range with the proposed configuration is reduced from five decades (from micro to tens of milliamperes) to only two decades (milliamperes to tens of milliamperes). This is achieved measuring the sleep state energy consumption in the wake-from-sleep event, during the C_{sleep} charge.

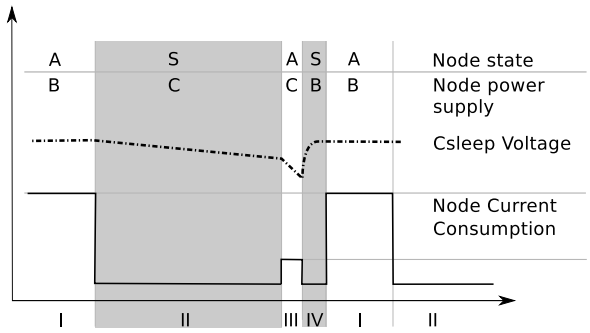

Fig. 5. Timing diagram of the measuring cycle. Active state is denoted by A, Sleep state by S. When the device is powered from the batteries it is marked as B and when it is powered from the buffer capacitor it is denoted as C. The axes scales are exaggerated for illustrative purposes.

The time line diagram in Fig. 5 shows the sensor node states and the measuring phases. During the interval I, the node and SCC are in active state (denoted by A) and powered by batteries (denoted by B). The current consumption is measured directly. During interval II, the node and SCC are both in the lowest power consumption state (sleep state, denoted by S). Note that the power is taken from the buffer capacitor C_{sleep} (denoted by C) and that there is no current through R_{shunt} during this interval. When the node wakes up, the sleep consumption is measured during interval III and IV. At the beginning of the interval III, the node controls the switches so that the node is powered from batteries, and C_{sleep} is connected to V_{CC} to be recharged. The current through R_{shunt} is now the sum of the currents of the supply to the node and the recharging current of C_{sleep}, and thus the total measured charge corresponds to the actual consumption of the device in interval II, III and IV. The node returns to the sleep state again at the beginning of interval IV to minimize the microcontroller contribution during the measurement period. The charge of the sleep state plus a very small consumption increment is measured. This extra consumption is negligible, since the intervals III and IV are relatively short.

Note that the sleep consumption measurement introduces an extra latency to the effective wake-up transition, which must be tolerated or handled properly.

C. Non-idealities

1) 555 timing circuit delay: The period of the signal f_{out} is determined by the sum of the three components: the charge and discharge time of C_T, and the delay time t_d^* that has the 555 timing circuit between its inputs and outputs (see Fig. 2). Note that the discharge time of C_T is negligible due to the low resistance of the transistor. Finally, considering the delay time t_d^* , and following the same procedure done for (2), it is obtained:

$$f(I) = \frac{1}{\frac{V_{REF}.R_{gain}.C_T}{I.R_{shunt}} + t_d^*} \tag{4}$$

where I is the current drained from the batteries, and R_{gain}, R_{shunt}, C_T and V_{REF} are the components previously described. The transfer function $f(I)$ is slightly non-linear because of the effect of t_d^*.

Considering that the charge quantum Q is I/f and $f < 1/t_d^*$ we obtain the following expression:

$$Q(f) = \frac{V_{REF}.R_{gain}.C_T}{R_{shunt}} \frac{1}{1 - f.t_d^*} \tag{5}$$

R_{gain}, R_{shunt}, C_T and V_{REF} should be sized in order to meet $\frac{V_{REF}.R_{gain}.C_T}{I.R_{shunt}} \gg t_d^*$. This can be easily achieved by taking into account the low values of I involved in the application. In this case, the linear approximations of (2) and (3) remains valid.

2) Operational amplifier offset voltage and bias current: The offset voltage V_{off} and bias current I_b of the operational amplifier will affect the linearity of the circuit:

$$I_{out} = \frac{R_{shunt}}{R_{gain}} I \pm \left| I_b \left(\frac{R_{shunt}}{R_{gain}} - 1 \right) - \frac{V_{off}}{R_{gain}} \right| \tag{6}$$

Considering that $R_{gain} \gg R_{shunt}$:

$$I_{out} = \frac{R_{shunt}}{R_{gain}} I \pm |I_b + \frac{V_{off}}{R_{gain}}| \quad (7)$$

and then,

$$f(I) = \frac{1}{\left[\frac{V_{REF} R_{gain} C_T}{R_{shunt} I \pm |I_b R_{gain} + V_{off}|}\right] + t_d^*} \quad (8)$$

To overcome this problem it is necessary to select an operational amplifier with low voltage offset and low bias current.

3) Uncertainty components: The uncertainty in Q due to the uncertainty of the components values, named δQ, is derived from (3) following the guidelines established in [31]:

$$\begin{aligned} \delta Q \quad = \quad & \frac{V_{REF}.R_{gain}.C_T}{R_{shunt}^2}.\delta R_{shunt} + \frac{V_{REF}.C_T}{R_{shunt}}.\delta R_{gain} \\ & + \frac{V_{REF}.R_{gain}}{R_{shunt}}.\delta C_T + \frac{R_{gain}.C_T}{R_{shunt}}.\delta V_{REF} \end{aligned} \quad (9)$$

It is important to know δQ not only to characterize the measure instrument, but also to evaluate the need of calibration. A reasonable low δQ using off-the-shelf components could avoid a calibration process. This would represent a great advantage, since it could be widely adopted for a precise charge measurement without the tuning cost.

D. Design considerations

In Table II we summarize the effect, that changing the main components of the SCC, has on its main features. These effects are given by (6), (8) and (9), and considered for the components selection in the next section.

TABLE II
EFFECTS OF VARIATION OF COMPONENT PARAMETERS

Parameter	Linearity	SCC Consumption	Freq.	Uncertainty
R_{gain} ↗	↗	↘	↘	↗
R_{shunt} ↗	↘	–	↗	↘
C_T ↗	↗	–	↘	↗
V_{off} ↗	↗	↗	↘	↗
I_b ↗	↘	↗	↗	↗

It is remarkable to note that the resistance of the switches SW_1 and SW_2, as well as R_{cap}, are not involved in the computation of $f(I)$ nor $Q(I)$, so its values (as well as its non-idealities) do not affect the precision of the measurement. This is because the measurement, for both active and sleep modes, are taken across R_{shunt} (see Fig. 1). These components only affect the recharge time of C_{sleep}, which is desirable, but not critical, to be kept at a minimum.

Charge losses of actual capacitors (due to temperature change, or leakage currents) are small because the temperature varies slowly, and the capacitor self-discharge is negligible compared to one sleep cycle.

IV. APPLICATION EXAMPLE

In order to validate our proposal, the SCC was integrated into the WSN node *TelosB* [10] (see Fig. 6). A PCB was fabricated and a software module that runs over ContikiOS [32] was developed.

TABLE III
TelosB CURRENT CONSUMPTION IN DIFFERENT STATES [10].

State	Current
Sleep	$10\mu A$ to $750\mu A$
Active w/ radio off	1.8 mA
Active w/ radio RX	21.8 mA
Active w/ radio TX (0dBm)	19.5 mA

A. Component selection

The *TelosB* sensor node uses a radio that complies with the IEEE 802.15.4 standard. The Media Access Control (MAC) protocol is based on Carrier Sense Multiple Access (CSMA), which implies that the communication channel must be checked for activity prior to transmission. This event, in which the radio is turned on for a very short period of time, is called CCA (Clear Channel Assessment). In the default configuration of ContikiOS, the CCA lasts $125\mu s$ and is performed two consecutive times, eight times per second.

The CCA is the shortest event of interest to be measured by the system. We defined that the CCA energy had to be at least ten times the charge measurement resolution. The charge consumed during a CCA was estimated in $5.8\mu C$, thus the maximum charge quantum is $Q_{max} = 0.58\mu C$. The estimation was made using a digital oscilloscope, by approximating the area below the current-time graph.

The measured sleep current of the *TelosB* using ContikiOS in the default configuration was $I_{sleep} = 750\mu A$ and the maximum sleep time $t_{sleep} = 125ms$ (corresponding to the system tick). A voltage drop in C_{sleep} was defined to be $1V$. Finally, the minimum value for C_{sleep} is calculated as:

$$C_{sleep}^{min} = \frac{I_{sleep} \cdot t_{sleep}}{\Delta V_{C_{sleep}}} \quad (10)$$

Table III shows the nominal current consumption of a *TelosB* mote in different states. Based on these values, we defined that the maximum current to be measured accurately is $30mA$. Currents from $1.8mA$ to full scale are measured directly by the CPC (node and SCC in active state) and currents below $1mA$ are measured using the buffer capacitor (node and SCC in sleep state).

The values for the rest of the components are selected based on the following considerations. R_{cap} limits the recharging current for C_{sleep}, which in this case should be kept below $30mA$. R_{shunt} should be kept small to minimize the voltage drop in the power supply. For higher values of R_{gain}, the power consumption of SCC is decreased. Finally, the R_{shunt} and R_{gain} ratio together with C_T determine the operation frequency, and the current consumption of the SCC. The output frequency, and thus the value of Q, is a compromise between resolution and linearity.

The selected values for the main pasive components of SCC are presented in Table IV. The active components selected were:

- switches: *ADG884* (ultra low resistance, low power);
- operational amplifier: *OPA333* (low offset, low power);
- PNP transistor: *MMBT3906*;

TABLE IV
SCC COMPONENTS VALUES.

Component name	value
R_{cap}	1Ω
R_{shunt}	1Ω
R_{gain}	$1.5k\Omega$
C_{sleep}	$100\mu F$
C_T	$100pF$

- voltage reference: *LT6656* (high accuracy, low power, low dropout);
- 555 timing circuit: *CSS555C* (with internal trimmed capacitor $C_T = 100pF$ with $\delta C_T = 1\%$).

B. Hardware modifications

In order to integrate the SCC, the sensor node has to fulfill a minimum set of requirements. Firstly, a counter to receive the output of the CPC (preferably implemented in hardware) must be available. Secondly, it is necessary to use three digital I/O lines: one for enabling/disabling the CPC, and the rest for controlling the recharging and feeding switches independently.

Because *Telosb* is an off-the-shelf product, not all the microcontroller pins are available for a direct connection with SCC. For example, the microcontroller pin connected as an external clock source to the internal hardware counter (the Timer/Counter not used by ContikiOS) is used by the external flash memory. In order to maintain the flash functionality, this signal was re-routed to an unused input/output pin of the microcontroller.

All the SCC signals (including control signals and power supply) were routed to the *Telosb* expansion connector. In this way the SCC was implemented as a daughter board for the *TelosB*. Fig. 6 shows the actual board with a *TelosB* node underneath. The area of this prototype was not optimized and is approximately $15cm^2$ ($30mm$ by $50mm$). The emphasis of the actual design was in mechanical restrictions and assembly (to plug SCC on the *TelosB*) and ease of testing. An area-optimized design could have an estimated area of $4cm^2$.

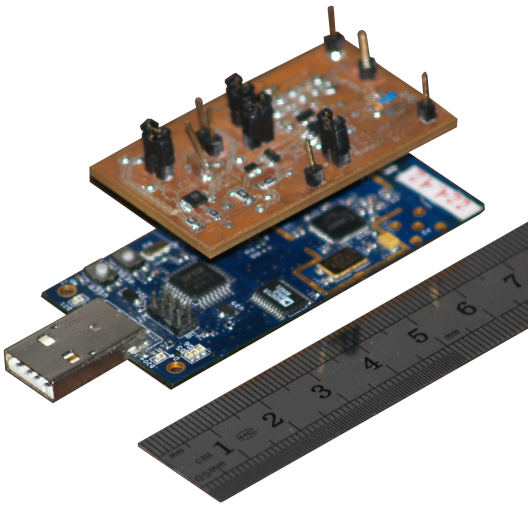

Fig. 6. TelosB with SCC daughter board integrated.

C. Software modifications

In order to include the SCC to a sensor node, a new software module was created and integrated to the ContikiOS to control and use the measurement circuit. Additionally, a few modifications were made to the original ContikiOS code.

The initialization of the module includes the hardware configuration: I/O ports to control SCC operation, and the Timer/Counter configured as a counter with external clock source. The SCC output is connected to this external clock source. The charge counter is two words wide (2 x 16 bits). The counter register corresponds to the low word of the charge counter, while the interrupt service routine associated to the Timer/Counter overflow interrupt increments the high word.

A state machine controls the SCC operation determining the SCC control outputs (to turn on/off the switches and output the 555 reset signal). The sleep and wake up routines of ContikiOS were modified in order to take into account the presence of SCC (essentially to recharge C_{sleep}).

In order to not increase the wake-up latency, and do not affect the timing of the operating system nor the application, the node wakes-up earlier to measure the sleep consumption. In this case, the duration of intervals II, III and IV in Fig. 5 corresponds to the original node sleep time. This is possible because the ContikiOS clock system implementation is based on an "interrupt tick" that periodically wakes up the microcontroller node.

Finally, the hardware modification stated in Section IV-B required a reconfiguration of the ports definitions in ContikiOS in order to make the pins available.

D. Energest

ContikiOS comes with a built-in functionality, named Energest [18], for the estimation of the power consumption. This time-based estimation, described in Section II, is implemented as a collection of macros and functions that allows to record the time that the node spends in a predefined state.

The conversion from the accumulated time to the energy consumption is processed in a computer (not in the nodes).

In order to make the comparison easier between SCC and Energest, the SCC interface was selected equal to the Energest interface. On the one hand, Energest measures elapsed times, by counting pulses of fixed duration (system clock), later the number of these pulses will be multiplied by the corresponding current level to compute the consumed charge. On the other, SCC counts pulses of variable duration, where each pulse corresponds to a fixed charge, so the number of pulses directly represents the total charge consumed without needing further computation.

V. RESULTS

In this section we present the results of the experiments performed to validate the proposed solution.

A. Transfer characteristics

The characterization of SCC was made measuring the output of the current-frequency converter. Firstly, measurements were

performed at normal temperature condition (25^oC) and at nominal power supply ($3.3V$). Later, these measurements were repeated at various other temperatures. The SCC operation varying the power supply from $2.4V$ and $3.3V$ is ensured by design, as it was explained in Section III.

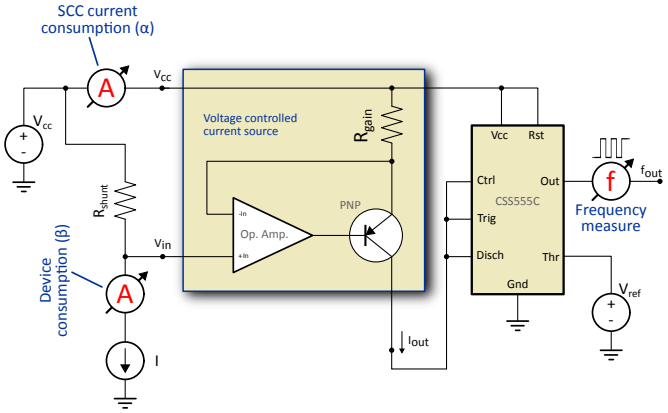

Fig. 7. Measurement setup circuit.

Fig. 7 depicts the measurement setup. The current at points α (SCC consumption) and β (sensor node or device consumption) were measured with two Fluke 45 multimeters, and the output frequency was registered with a Tektronix TDS220 oscilloscope. The voltage supply, V_{CC}, was obtained from a Tektronix PS280 regulated power supply and the temperature was varied by means of a custom temperature-controlled oven.

The energy consumption of a typical sensor node in active state was emulated using a standard current source I_{Node} (a Wilson current source). The current was varied from $0.2mA$ to $100mA$, extending the design range for an expanded characterization.

Fig. 8 shows the relationship between the output frequency of the current-frequency converter and the instantaneous current I_{Node} for the full range of source currents. It can be seen that the transfer characteristics of the SCC remains constant with the temperature variation.

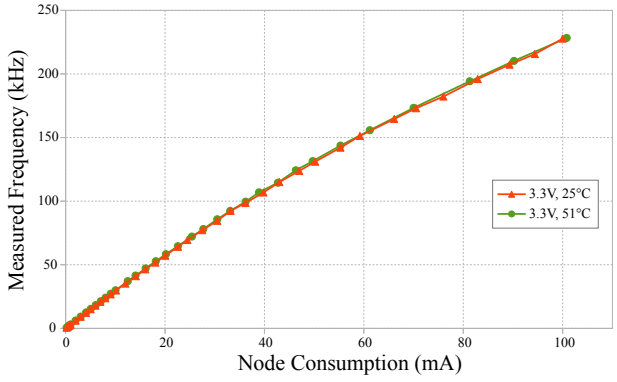

Fig. 8. Output frequency vs. sensor node current consumption, varying the temperature.

Fig. 9 shows the charge quantum as a function of the frequency, which remains almost constant. The very slight difference is caused by the non-linear effects analyzed previously and described by (4) and (5). If a more linear transference is desired, i.e. a more constant quantum, it could be achieved increasing R_{gain} at the cost of more power consumption.

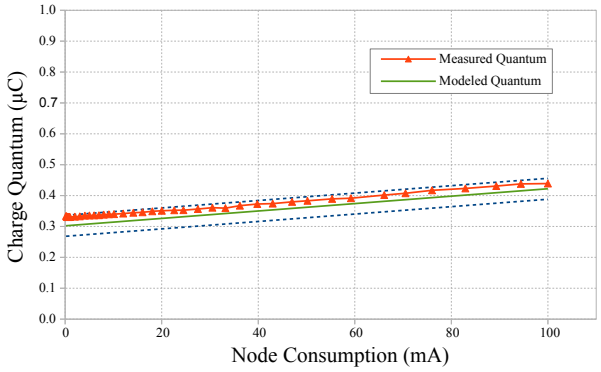

Fig. 9. Charge quantum prediction: model (continuous green) and its uncertainty (dashed blue), and measured (red with triangle marks).

The SCC current consumption is shown in Fig. 10 as a function of the sensor node current consumption. It is apparent that this current is negligible compared to the one consumed by the sensor node (less than one thousand times), and that it is independent of the temperature variation. This ensures that the use of SCC does not significantly affect the battery lifetime.

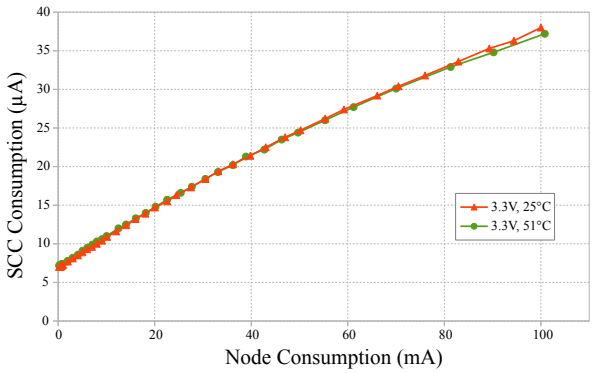

Fig. 10. Current consumption of SCC, varying the temperature.

B. SCC without calibration

The uncertainty in Q (δQ) due to the uncertainty of the components values was presented in (9).

Using resistors within the E24 series ($\delta R = 5\%$) results in $\delta Q = 0.034\mu C$. This represents an uncertainty of 9.7% in the quantum value. This uncertainty is depicted in Fig. 9 (dashed line) and compared with the measured quantum.

If a more precise quantum is required, resistors from the E96 series ($\delta R = 1\%$) can be used without compromising the cost. In this case $\delta Q = 0.009\mu C$ with an uncertainty of 2.7%. Furthermore, with 0.01% resistors results in $\delta Q = 0.009\mu C$, which implies an uncertainty of 0.9%.

This analysis shows that the measurement method achieve good accuracy without calibration, by decreasing the inaccuracy in just two of their components.

TABLE V
COMPARISON BETWEEN SCC AND A TRUE RMS AMP-METER IN A *TelosB*.

Method	Current (mA)
SCC	1.01 ± 0.01
True RMS amp-meter	1.04 ± 0.04

C. SCC validation in a real sensor node

As it was described in Section IV, the SCC was integrated in the WSN node *TelosB* in order to validate our solution. First, we test the overall accuracy of SCC. It was conducted by measuring the current over a minute of operation using a TrueRMS amp-meter (UNI-T E60) and the SCC. During this experiment, the main scenarios where a WSN node operates are covered: different states of microcontroller (active and sleep mode) and radio (idle, transmitting, and receiving). The average current was registered and compared. The result of this comparison is shown in Table V.

This first comparison validates the measurement method as a whole system, taking into account not just the consumption during the active state of the sensor node, but also the consumption during the sleep state.

D. SCC versus an estimation method

In order to highlight the benefits of SCC over Energest, a very simple "proof of concept" was implemented introducing a power level deviation from the laboratory calibration in the WSN node *TelosB*. For ease, this deviation was injected by turning on a LED to add a step of power consumption not considered by Energest. In a real-life deployment, the deviation source may result from temperature variations or a sensor malfunction.

A software based on the Collect-view example from the standard distribution of ContikiOS was developed, including both modules simultaneously (Energest and SCC). This software represents a very common application of data collection, in which sensors data are acquired periodically by each node and sent through the network in a hop-by-hop basis to a sink node.

In this experiment, the node sent the values acquired by its sensors (temperature and so on) and the data registered by Energest and SCC. The Collect-view parameters "report interval" and "report randomness" were set to two seconds. Consequently, the effective report period ranges from (almost) zero to four seconds. The LED was toggled between *ON state* and *OFF state* every n data packets. The configured toggle rate was $n = 13$, which is nominally 26 seconds.

Fig. 11 illustrates the charge consumption of the sensor node: Energest (green filled circle) and SCC (red filled triangle) between packets.

Fig. 12 highlights the difference between the SCC measurement and Energest estimation (*SCC* minus *Energest*), presented in Fig. 11, and the states of the LED. The observed charge peaks in the figure corresponds to "beacons" sent by the node (broadcast messages sent repeatedly). Note that, since the effective report interval is not deterministic, the measured charge for a constant current could differ.

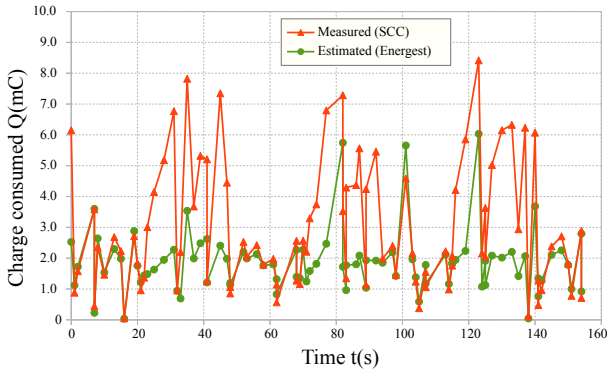

Fig. 11. Comparison between SCC and Energest, turning on/off a LED in a *TelosB*.

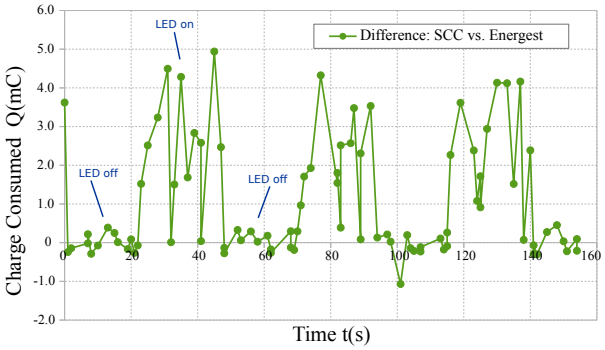

Fig. 12. Difference between SCC (measured) and Energest (estimated) power consumption.

Table VI shows the charge consumption of the sensor node calculated in a computer from the data registered in real-time by both Energest and SCC modules. Energest and SCC compute the charge consumption as was described in Section IV-D. Table VI shows for each period (ON and OFF) the elapsed time and the average current. The average currents is computed as the charge consumed during the period divided by the elapsed time. In the OFF state, the values registered by SCC and Energest are very similar, with a low relative error (1.4%). This difference is a result of Energest using nominal current values for the energy estimation. In the ON state, the consumption values differ greatly (more than 85% of relative error) and corresponds to the LED current consumption, approximately 1250μA. Table VI shows this behavior repeated throughout the entire experiment.

To summarize, this section has shown that the estimation methods fail in scenarios where a *new hardware* is added without a full characterization of the new power consumption operation modes, or when the power profile varies from nominal or measured values due to real-life uncontrolled environmental conditions.

VI. CONCLUSIONS

The main contribution of this paper is to have proposed a method and a circuit (named SCC) that allows a WSN sensor node to measure in real-time its own energy consumption on the field, with a high dynamic range, easy integration,

TABLE VI
COMPARISON BETWEEN SCC AND ENERGEST, TURNING ON/OFF A LED IN A *TelosB*.

Packet Number n	Time (ms)	SCC (μA)	Energest (μA)	Difference (μA)	Relative Error (%)	State
$2-14$	22.383	989.6	1003.5	-13.9	1.4	OFF
$15-27$	24.508	2338.9	1028.5	1310.4	127.4	ON
$28-40$	23.039	953.3	939.8	13.5	1.4	OFF
$41-53$	21.984	2421.7	1186.8	1234.9	104.1	ON
$54-66$	22.672	1051.9	1100.5	-48.7	4.4	OFF
$67-79$	25.297	2338.7	1100.9	1237.8	112.4	ON
$80-92$	23.070	965.7	952.1	13.6	1.4	OFF
$93-105$	24.367	2475.8	1335.0	1140.8	85.5	ON
$106-118$	25.438	1023.5	1055.2	-31.8	3.0	OFF
$119-131$	22.148	2462.4	1150.8	1311.6	114.0	ON

TABLE VII
MAIN CHARACTERISTICS OF SCC.

Parameter	Min	Typ	Max	Units
Current Consumption	6.6	-	18.4	μA
Power Supply	2.2	-	5.0	V
Frequency	3.0	-	84.6	kHz
Linearity	-	-	2.6	$\%FS$
Resolution	-	0.35	-	μC
Precision	-	1%	-	-

low resource requirements in terms of energy, memory and processor usage, as well as in terms of component count (area and cost). This was achieved with very good results in terms of linearity, temperature stability, independence of the battery voltage drop over time, and without significantly affecting the battery lifetime. Table VII summarizes the main characteristics of SCC.

This contribution has been made possible by the novel method proposed to measure the sleep consumption. The method proposes to use a buffer capacitor as a *sleep consumption memory* enabling to measure a range of five decades using the same measurement principle and therefore the same circuit.

Moreover, we developed a fully-functional prototype based on the sensor node *TelosB* running ContikiOS. The main features have been tested on the field and the advantages of our system were shown under different temperature and energy consumption conditions.

Furthermore, a comparison between our solution and a time-based energy estimation (Energest) was conducted, illustrating scenarios where the estimation methods fail (e.g. consumption variations due to temperature or battery voltage changes or when a *new hardware* is added) and the SCC is able to provide a reliable measurement.

The actual power level has deviations from the laboratory measurement due to temperature or battery voltage variation. In case that the estimation is performed in a computer, the accuracy could be improved by means of complex modeling for considering the individual calibration, temperature and voltage variation (also reported by the nodes).

The proposed method can be used as-is in other embedded systems. However, the design imposes some restrictions. The sleep time must be bounded (imposed by energy stored in the C_{sleep} sleep capacitor). If the platform sleeps for longer periods than the considered in the current design, it should be necessary to increase the value of C_{sleep}. Furthermore, the other platforms need to have similar power consumption levels. If not, R_{gain} and R_{shunt} need to be adjusted. The SCC can be used in a wide range of systems, as long as it has a counter capable of carrying the count of the pulses that the SCC generates, and can handle the signals that control the switches.

Future work could include the development of an SCC for measurement of power consumption in FPGA's. Another area of interest is the implementation of the SCC in an integrated circuit. Finally, long term deployments should be conducted in order to exploit the advantages of energy measurement on every node of the WSN.

ACKNOWLEDGMENT

The authors would like to thank Marion Hoffer for proof-reading the initial version of this manuscript. This work was partially funded by CSIC-UDELAR (Comisión Sectorial de Investigación Científica, Universidad de la República, Uruguay).

REFERENCES

[1] M. Hempstead, M. J. Lyons, D. Brooks, and G.-Y. Wei, "Survey of hardware systems for wireless sensor networks," *Journal of Low Power Electronics*, vol. 4, no. 1, pp. 11–20, 2008.

[2] Y. Huang, W. Yu, and A. Garcia-Ortiz, "Accurate energy-aware workload distribution for wireless sensor networks using a detailed communication energy cost model," *Journal of Low Power Electronics*, vol. 10, no. 2, pp. 183–193, 2014.

[3] Y. Hamouda and C. Phillips, "Adaptive sampling for energy-efficient collaborative multi-target tracking in wireless sensor networks," *Wireless Sensor Systems, IET*, vol. 1, no. 1, pp. 15–25, March 2011.

[4] C. Haas and J. Wilke, "Energy Evaluations in Wireless Sensor Networks: A Reality Check," in *Proceedings of the 14th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, ser. MSWiM '11. New York, NY, USA: ACM, 2011, pp. 27–30. [Online]. Available: http://dx.doi.org/10.1145/2068897.2068905

[5] R. Yan, H. Sun, and Y. Qian, "Energy-Aware Sensor Node Design With Its Application in Wireless Sensor Networks," *Instrumentation and Measurement, IEEE Transactions on*, vol. 62, no. 5, pp. 1183–1191, May 2013. [Online]. Available: http://dx.doi.org/10.1109/tim.2013.2245181

[6] A. Junior, R. Sofia, and A. Costa, "Energy-awareness in multihop routing," in *Wireless Days (WD), 2012 IFIP*. IEEE, Nov. 2012, pp. 1–6. [Online]. Available: http://dx.doi.org/10.1109/wd.2012.6402881

[7] S. Zairi, B. Zouari, E. Niel, and E. Dumitrescu, "Nodes self-scheduling approach for maximising wireless sensor network lifetime based on remaining energy," *Wireless Sensor Systems, IET*, vol. 2, no. 1, pp. 52–62, March 2012.

[8] C. Menegazzo and L. Pessoa Albini, "Unadvertised energy saving method for static and homogeneous wireless sensor networks," *Wireless Sensor Systems, IET*, vol. 4, no. 3, pp. 105–111, September 2014.

[9] M. M. H. Khan, H. K. Le, M. LeMay, P. Moinzadeh, L. Wang, Y. Yang, D. K. Noh, T. Abdelzaher, C. A. Gunter, J. Han, and X. Jin, "Diagnostic powertracing for sensor node failure analysis," in *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks*, ser. IPSN '10. New York, NY, USA: ACM, 2010, pp. 117–128. [Online]. Available: http://dx.doi.org/10.1145/1791212.1791227

[10] J. Polastre, R. Szewczyk, and D. Culler, "Telos: enabling ultra-low power wireless research," in *Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium on*. IEEE, Apr. 2005, pp. 364–369. [Online]. Available: http://dx.doi.org/10.1109/ipsn.2005.1440950

[11] A. Prayati, C. Antonopoulos, T. Stoyanova, C. Koulamas, and G. Papadopoulos, "A modeling approach on the TelosB WSN platform power consumption," *Journal of Systems and Software*, vol. 83, no. 8, pp. 1355–1363, Aug. 2010. [Online]. Available: http://dx.doi.org/10.1016/j.jss.2010.01.015

[12] L. Steinfeld, M. Ritt, F. Silveira, and L. Carro, "Optimum design of a banked memory with power management for wireless sensor networks," *Wireless Networks*, vol. 21, pp. 81–94, Jul. 2014. [Online]. Available: http://dx.doi.org/10.1007/s11276-014-0763-5

[13] C. Chauvenet, B. Tourancheau, and D. Genon Catalot, "Energy Evaluations for Wireless IPv6 Sensor Nodes," in *SENSORCOMM 2013, The Seventh International Conference on Sensor Technologies and Applications*, 2013, pp. 97–103. [Online]. Available: http://www.thinkmind.org

[14] M. Kovatsch, S. Duquennoy, and A. Dunkels, "A Low-Power CoAP for Contiki," in *Mobile Adhoc and Sensor Systems (MASS), 2011 IEEE 8th International Conference on*, ser. MASS '11. Washington, DC, USA: IEEE, Oct. 2011, pp. 855–860. [Online]. Available: http://dx.doi.org/10.1109/mass.2011.100

[15] A. Dunkels, L. Mottola, N. Tsiftes, F. Österlind, J. Eriksson, and N. Finne, "The announcement layer: beacon coordination for the sensornet stack," in *Proceedings of the 8th European conference on Wireless sensor networks*, ser. EWSN'11. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 211–226. [Online]. Available: http://portal.acm.org/citation.cfm?id=1966270

[16] M. Ghamari, B. M. Heravi, U. Roedig, B. Honary, and C. A. Pickering, "Improving transmission reliability of low-power medium access control protocols using average diversity combining," *Wireless Sensor Systems, IET*, vol. 2, no. 4, pp. 377–384, Dec. 2012. [Online]. Available: http://dx.doi.org/10.1049/iet-wss.2012.0029

[17] M. R. Akhavan, S. Choobkar, A. Aijaz, and A.-H. Aghvami, "Adaptive preamble-sampling techniques for receiver-based medium access control protocols in lossy wireless sensor networks," *IET Wireless Sensor Systems*, vol. 5, pp. 31–41, July 2014. [Online]. Available: http://digital-library.theiet.org/content/journals/10.1049/iet-wss.2013.0117

[18] A. Dunkels, F. Österlind, N. Tsiftes, and Z. He, "Software-based On-line Energy Estimation for Sensor Nodes," in *Proceedings of the Fourth Workshop on Embedded Networked Sensors (Emnets IV)*, Cork, Ireland, Jun. 2007. [Online]. Available: http://www.sics.se/~adam/dunkels07softwarebased.pdf

[19] J. Andersen and M. T. Hansen, "Energy Bucket: A Tool for Power Profiling and Debugging of Sensor Nodes," in *Sensor Technologies and Applications, 2009. SENSORCOMM '09. Third International Conference on*, vol. 0. Los Alamitos, CA, USA: IEEE, Jun. 2009, pp. 132–138. [Online]. Available: http://dx.doi.org/10.1109/sensorcomm.2009.29

[20] I. Haratcherev, G. Halkes, T. Parker, O. Visser, and K. Langendoen, "PowerBench: A Scalable Testbed Infrastructure for Benchmarking Power Consumption," in *Int. Workshop on Sensor Network Engineering (IWSNE)*, 2008.

[21] M. Tancreti, M. S. Hossain, S. Bagchi, and V. Raghunathan, "Aveksha: A Hardware-software Approach for Non-intrusive Tracing and Profiling of Wireless Embedded Systems," in *Proceedings of the 9th ACM Conference on Embedded Networked Sensor Systems*, ser. SenSys '11. New York, NY, USA: ACM, 2011, pp. 288–301. [Online]. Available: http://dx.doi.org/10.1145/2070942.2070972

[22] X. Jiang, P. Dutta, D. Culler, and I. Stoica, "Micro Power Meter for Energy Monitoring of Wireless Sensor Networks at Scale," in *Proceedings of the 6th International Conference on Information Processing in Sensor Networks*, ser. IPSN '07. New York, NY, USA: ACM, 2007, pp. 186–195. [Online]. Available: http://dx.doi.org/10.1145/1236360.1236386

[23] P. Dutta, M. Feldmeier, J. Paradiso, and D. Culler, "Energy Metering for Free: Augmenting Switching Regulators for Real-Time Monitoring," in *Information Processing in Sensor Networks, 2008. IPSN '08. International Conference on*. IEEE, Apr. 2008, pp. 283–294. [Online]. Available: http://dx.doi.org/10.1109/ipsn.2008.58

[24] R. Fonseca, P. Dutta, P. Levis, and I. Stoica, "Quanto: tracking energy in networked embedded systems," in *Proceedings of the 8th USENIX conference on Operating systems design and implementation*, ser. OSDI'08. Berkeley, CA, USA: USENIX Association, 2008, pp. 323–338. [Online]. Available: http://dl.acm.org/citation.cfm?id=1855741.1855764

[25] S. Dhouib, E. Senn, J.-P. Diguet, D. Blouin, and J. Laurent, "Energy and power consumption estimation for embedded applications and operating systems," *Journal of Low Power Electronics*, vol. 5, no. 4, pp. 416–428, 2009.

[26] J. W. Hui and D. E. Culler, "IP is dead, long live IP for wireless sensor networks," in *Proceedings of the 6th ACM conference on Embedded network sensor systems*, ser. SenSys '08. New York, NY, USA: ACM, 2008, pp. 15–28. [Online]. Available: http://dx.doi.org/10.1145/1460412.1460415

[27] P. Hurni, B. Nyffenegger, T. Braun, and A. Hergenroeder, "On the accuracy of software-based energy estimation techniques," in *Proceedings of the 8th European conference on Wireless sensor networks*, ser. EWSN'11. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 49–64. [Online]. Available: http://portal.acm.org/citation.cfm?id=1966257

[28] J. Villaverde, D. A. Bouvier, C. Fernández, L. Steinfeld, and J. Oreggioni, "Low-Power Self-Energy meter for wireless sensors network," in *IEEE DCoSS 2013 Posters (IEEE DCoSS 2013 Posters)*, Cambridge, USA, May 2013, pp. 315–317.

[29] J. Villaverde, L. Steinfeld, J. Oreggioni, D. Bouvier, and C. Fernandez, "Self-energy meter in duty-cycle battery operated sensor nodes," in *Instrumentation and Measurement Technology Conference (I2MTC) Proceedings, 2014 IEEE International*, May 2014, pp. 1595–1599. [Online]. Available: http://dx.doi.org/10.1109/I2MTC.2014.6861015

[30] A. Hergenroder and J. Furthmuller, "On energy measurement methods in wireless networks," in *Communications (ICC), 2012 IEEE International Conference on*. IEEE, Jun. 2012, pp. 6268–6272. [Online]. Available: http://dx.doi.org/10.1109/icc.2012.6364942

[31] Joint Committee for Guides in Metrology (JCGM), *Evaluation of measurement data - Guide to the expression of uncertainty in measurement*, 2008, JCGM 100:2008. [Online]. Available: http://www.bipm.org/en/publications/guides/gum.html

[32] A. Dunkels, B. Grönvall, and T. Voigt, "Contiki - A Lightweight and Flexible Operating System for Tiny Networked Sensors," *Local Computer Networks, Annual IEEE Conference on*, vol. 0, pp. 455–462, 2004. [Online]. Available: http://dx.doi.org/10.1109/LCN.2004.38