Energy Evaluations for Wireless IPv6 Sensor Nodes

Cedric Chauvenet
Watteco Inc
La garde, France

Bernard Tourancheau
LIG UMR 5217
Grenoble University, France

Denis Genon Catalot
LCIS EA 3747
Grenoble University, France

ABSTRACT

Typical Wireless Sensor Networks (WSN) include some energy autonomous nodes and main powered router nodes. To meet a reasonable lifetime, i.e. years, the autonomous devices have to encompass strong energy constraints. Our target network architecture consists in such wireless autonomous nodes connected to a backbone based on PLC nodes. Our given building automation application scenario runs a 10 minutes periodic probes sampling which reports a data frame over the Internet using IPv6. For this purpose, our WSN nodes use open RPL/ContikiOS softwares and typical off the shelf electronic components.

In order to optimize the node energy consumption we propose a systemic analysis that includes all the software layers and hardware components. The energy distribution among the different components and the critical software parameters are weighted against the global energy consumption. Thanks to measurements and technical data, we propose a simple model that allows to identify pitfalls and determine optimized solutions. Following our established guidelines, we believe our future WSN monitoring platform will achieve more than 10 years lifetime with the target scenario.

Categories and Subject Descriptors

C.3 [**Computer Systems Organization**]: Special purpose and application-based systems; D.4.7 [**Software Engineering**]: Organization and Design—*measures, performance, energy*

Keywords

WSN; Energy optimization; Lifetime; Performance; Measurement

1. INTRODUCTION

WSN are called to new challenges given their energy constraints. To meet their power, price, size and deployment requirements, typical platforms fostered on a design based on a low power micro controller unit (MCU) and a low power RF transceiver. Adding a battery and probes to this communicating platform leads to a cheap autonomous WSN node. WSN's lifetime should be long enough, i.e. years, to provide a reasonable return on investment. Any costly maintenance like battery replacement annihilates the interest in WSN deployments. WSN devices will be sleeping most of the time and periodically wake up to perform their operations. This is controlled by their embedded operating software stack. Though carefully designed, these softwares should be carefully tuned with respect to the application requirements. In this work, we review the energy optimization process of a node that is compliant with IETF recommendations. The target application is home and building automation.

Section 2 states the art of the domain and section 3 presents the context of our work. In section 4 and 5 the power consumption study of several MCU and transceivers are presented. Section 6 shows the impact of the routing parameters. In section 7 the global leaf node energy spending is optimized. Section 8 and 9 presents software optimizations for the MCU and the transceiver. Section 10 shows the probes consumptions that will be added to the node.

2. RELATED WORK

Energy consumption is the most important criterion for the development of autonomous sensor network nodes. Especially when they target several decades or perpetually powered systems. As mentioned in [11], battery replacement is not an option for networks with thousands of physically embedded nodes and the paper list various and commonly used techniques to save power such as power-aware computing, energy-aware software or power management of radios.

Energy has been considered since the very beginning of WSN developments, from the first mote to concretize the idea of communicating sensors such as the WeC node designed in 1998 [9]. Then, some similar platforms have been designed, with more powerful MCU and the same radio chip until 2001, where a new generation of motes emerged with the Mica [5], designed in Berkeley in 2002. This mote was carefully designed to serve as a general purpose platform for WSN research. Work in [5] highlights the need for node sleeping most of the time with periodic wake up, and shows that a total improvement factor of 11 can be reached with this technique. From this date on, all the sensor network mote designs used this duty cycling techniques to save power. Leveraging on experimental feedback, Mica2 corrected many shortcomings: the boost converter was discarded, new components were chosen. But this new design

UBIMOB '13 Nancy, France

leads to higher power consumption. Continuing the evolution, MicaZ replaced the CC1000 radio with a CC2420, an IEEE 802.15.4 2.4GHz compatible radio with 250 kbps throughput capability, and evolved O-QPSK modulation. This transceiver embedded a part of the 802.15.4 standard, limiting the software processing needed to send and receive packets by DSSS signal treatment, thus saving energy.

A major step has been reached with the design of the Telos platform [10] released in 2004. It enables experimentations with minimal power consumption, a maximum 10 KBit RAM memory for managing increasing stack fonctionnalities and increased software and hardware robustness. Telos is based on a MSP430 MCU with a CC2420 radio. The MSP430 offered significantly lower consumption, reducing the total active power down to 41 mW, with the same transmitting power consumption as MicaZ. According to [10], the power consumption of the MSP430 is 20 times lower than the MicaZ in sleep mode, and 4 times lower in active mode.

However even if hardware greatly improved over time, software requirements changed, especially for the communication stack. The IETF proposed the adoption of IPv6 with an adaptation layer called 6LoWPAN [8] and a routing proposal named RPL [2]. This software complexity impacts the key parameters governing power requirement. In [7], the authors evaluate the performance of RPL and 6LoWPAN using the TinyOS stack. They showed that RPL parameters' values and the number of downwards stored routes affect the power consumption. Also they proposed an optimization for fragmented packets where only the header part is decompressed for routing decisions. Authors of [19] proposed a power consumption model that deeply describes the wireless communication. Hence, this work shows that the number of hops should be reduced to the minimum in order to saved power.

We address the power consumption issues by conducting real hardware measurements and analysis of the software impact on consumption. As suggested in [11], we split the system into several parts and conduct an analysis for each of them in the first part of the paper. However, the system should also be considered as a whole in order to insure a good integration between hardware and software components. This is the aim of section second part of our paper where we optimize the software behavior against energy.

3. PLATFORM SETTINGS

This paper relates our research in order to define an operational and efficient WSN platform using PLC and RF communication transceivers. This allows to build energy efficient hybrid architectures, as described in [12]. These architectures gathers autonomous battery powered RF nodes, main powered PLC or RF nodes and PLC-RF routers. At the network level, the nodes are seamlessly interconnected using the Internet IPv6 protocols. Such an architecture restricts the need for routing capabilities to mains powered devices. This avoids battery operated devices to drain their battery when acting as a router.

In order to operate the nodes for this study, we used ContikiOS and tools [4] which provide an open source micro operating system for constrained devices.

The the RF transceivers follows the IEEE 802.15.4 standard [6] which was designed for low power, short range and low throughput networks Initially released in 2003, this standard offers a maximum throughput of 250 kbps. New revisions of this standard added other modulations scheme and a sub-GHz RF band for less attenuation transmission with lower throughput, 20k-50Kb/s, to achieve longer range and more robust communication.

The target MAC layer relies on the CSMA/CA mechanisms required by the 802.15.4 standard. Also, as in most of WSN protocol stacks, a radio duty cycling (RDC) protocol is added in order to save a significant amount of power. We use an adapted version of contikiMAC RDC [3] without the preamble sampling protocol. This protocol induces periodic radio wake up to sense the radio activity at a fairly high frequency, 4 Hz by default. This protocol may sends packet copy until the recipient acknowledge it. Notice that if the traffic required by the application is lower than 4 Hz, the periodic wake up will induces non necessary overhead. Our target temperature monitoring application typically requires one packet per node every 10 minutes send to the sink.

The IPv6 for Low power Wireless PAN, 6LoWPAN, is the adaptation layer specified by IETF (RFC 6282) for IEEE 802.15.4 type links. It offers header compression to save bytes and allows frame fragmentation to resolve MTU issues.

The Routing Protocol for Low power and Lossy Networks, RPL, is the first routing protocol recommendation of IETF for LR-WPAN (RFC 6550). RPL is a proactive distance vector protocol that creates Destination Oriented Directed Acyclic Graphs (DODAG). Notice that node energy can be used in the metric governing the routing topology construction.

On top of this IPv6 stack, our target application uses the IPSO application framework [14] profiles designed primarily for smart energy applications. These profiles relies on the Constrained Application Protocol, CoAP [15], that enables sensing nodes to be interrogated through RESTfull primitives.

4. MCU POWER CONSUMPTION

In a node, the MCU wakes up periodically to check if an event is detected by the probe or the RF transceiver, and executes the corresponding software tasks. Otherwise, it stays in sleep mode. Table 1 compares the current drawn for 3 MCUs in different states. The results shows the great gap between active and low power modes. For instance, the MSP430f5438A running at 20 MHz can theoreticaly achieve a 5 years lifetime only if it stays in sleep mode more than 99,6% of the time. The MCU frequency impacts its energy

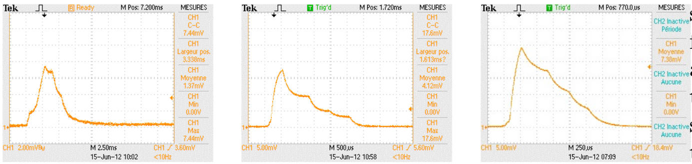
Table 1: Micro controllers drawn current under 3V at different modes and frequencies. Experimental measurements are in bold.

Mode	MHz	MSP430 f1611 (mA)	MSP430 f5438A (mA)	SIM3C1xx (mA)
Active	4	2 [17]	**1.51**	
Active	8	4 [17]	1.84 [18]	
Active	16		**5.21**	
Active	20		6.37 [18]	7.8 [13]
Active	80			22 [13]
Low P	0.032	0.002 [17]	0.0021 [18]	0.0008 [13]
Sleep		0.002 [17]	0.0012 [18]	0.000145 [13]

consumption. In order to quantify this, we measured the voltage on a 10,1 Ω load added between a regulated DC power source of 3 V and the target architecture. In the experiments, this voltage reflects the current drawn by the MCU plus the ATRF212 transceiver in its standby mode which is 0.2 μA in [**?**]. We ran the complete software stack over different frequencies without any RF communication nor connected probe. Figure 1 shows the corresponding oscilloscope traces, averaged over 128 wake up samples to include the variety of wake up profiles. We calculate the energy, E by integrating the power over time during the period of visualization following equation 1.

$$E(J) = \frac{3}{10.1} \times \int_0^{t_{startup}} V_{average} \quad (1)$$

Table 2 summarized the corresponding computed energy.



3.9 MHz 8 MHz 16 MHz

Figure 1: Current profile of MSP430 average wake up at different operation frequency

Table 2: Characteristics and energy cost of MSP430 wake up at different operation frequency

Frequency of the MSP (MHz)	3.9	8	16
Rise Up Time (ms)	2.5	0.5	0.25
Maximum current (mA)	0.74	1.74	2.38
Drop Time (ms)	7.5	2.5	1.5
Time of Observation (ms)	25	5	2.5
Average Voltage (V)	1.37	4.12	7.83
Energy (μJ)	**10.2**	**6.12**	**5.48**

As expected, the maximum current drawn increases with the MSP frequency while the rise up and drop time decrease, although none of these happens in a linear fashion. But we notice a higher gap between 3.9 and 8 MHz than between 8 and 16 MHz for these three parameters. Regarding the maximum current, there is a 1 mA gap between 3.9 and 8 MHz, and a 0.64 mA gap only between 8 MHz and 16 MHz. Notice that our measurements differ from the data sheet maximum current consumption. This can be explained for 2 reasons. Firstly, they are averages over 128 samples. And secondly, the wake up duration is not long enough to reach the maximum current consumption. Similarly, the rise up and drop time are roughly divided by a factor of 2 when jumping from 8 to 16 MHz but there is a respective factor of 4 to 5 when moving from 4 to 8 MHz. The poor performances of the MCU, while down clocked at 4 MHz, may explain this behavior. Taking into account these results, the SIMC3C1 seems promising for our future platform.

5. TRANSCEIVER POWER CONSUMPTION

With our architecture, the leaf nodes cannot route messages. An important parameter is thus the transmission range to ensure that the given leaf will reach a main powered router. Available transceivers for WSN follow the IEEE 802.15.4 standard which proposes several frequency bands. We focus on 2 of them : 868 MHz and 2.4 Ghz. In order to quantify the theoretical advantage of each transceiver for our application requirements, we computed the power needed to communicate within a range d. The power received at a given distance d in free space is given by the Friis formula that states the distance where the power received match the sensibility of the transceiver.

$$P_{received} = 22dB + 20 * log(d/\lambda) \quad (2)$$

Table 3 presents the results. The ATRF230 has a low instantaneous power consumption and a high throughput, confirmed by its "Tx + Rx" energy number. However, its coverage range is small compare to the others. The ATRF212 and the CC1120 transceivers have fairly similar "Tx + Rx" consumptions, because the higher power consumption of the latter in balanced by its higher throughput capability. Though, the 2.4GHz band offers a shorter range than the two others, it is more subject to external disturbances and the maximum power of transmission authorized in this band is lower. As a result, for a leaf in our architecture, a router may be out of range in 2.4 GHz while connected in 868 MHz.

Table 3 also shows that the maximum reachable distance over 868 MHz is 27 times greater than over 2.4 GHz using the transceivers and regulation considered. The energy cost per bit in much lower in 2.4 GHz for distances shorter than the 1.5km range. Also, the channel occupation is longer in 868 MHz and this may become an issue in dense environments. With our architecture and target application, we decided to use the 868 MHz band and thus the SI4461 transceiver seems a good candidate. Beside energy considerations, this band may increase the path diversity while being able to connect to several routers ad this allows a leaf mode to select backup routes.

Of course, additional parameters could be added to this model and in particular, the gain and the type of antennas that could greatly impact the transmission range.

6. ROUTING CONTROL IMPACT

The literature exhibits the high power cost of RF transmissions [], and many mechanisms have been investigated to reduce this energy budget. Some of them rely on frame size reduction such as compression [1] or data aggregation []. A complementary solution is to limit the number of control packets sent. For instance, RPL uses the trickle algorithm to this end. Several other parameters may influence the transceiver consumption, for instance throughput and packet delivery rate have direct impact on the energy consumption, as they define the time of each transmissions and the average number of retransmission.

In our target home or building automation use case, leaf nodes running over battery are periodically reporting values sensed by the probe, and downward traffic is restricted to node configuration and is not periodic. This results in highly asymmetric traffic from sensors to the border router.

The RPL routing control messages messages are described in table 4. The leaf nodes mainly interact at initialization

Table 3: Characteristics of radio transceivers [] and corresponding energy for different transceivers.

Module	Rate (kbps)	Freq (MHz)	Sensib (dBm)	Current Tx (mA)	Current Rx (mA)	Energy Tx+Rx (μJ/bit)	Max Power (dBm)	Max Range (km)	Power Recv (dBm)	Energy at Max Range (μJ/bit)	Energy at 1.5 km (μJ/bit(dBm))
ATRF212	20	868	-110	25	9.2	5.13	10	25	-109.19	10.26	2.96 (-15)
CC1120	50	868	-110	45	22	4.02	14	40	-109.27	4.02	2.52 (-15)
SI4464	40	868	-110	37	13	3.75	14	40	-109.27	3.75	2.33 (10)
SI4461	40	868	-110	33	13	3.45	14	40	-109.27	3.45	2.33(10)
ATRF230	250	2400	-101	16.5	15.5	0.38	3	1.5	-100.58	0.38	0.38(3)

in the proactive RPL strategy.

Table 4: RPL messages periodicity.

Type	Origin	Message Periodicity	Comments
NS	Router	Best parent 60s; Other p. 120s ; Other neighbor 360s	No NS to leaf
NS	Leaf	Best p. : 3 to 600s	Trickle timer
NA	Router	Same as NS	
NA	Leaf	None	No NS to leaf
DIO	Router	Imin 1s ; Imax 1050s	Trickle
DIO	Leaf	Not regular	Once attached inform neighbors about rank
DATA	Router Leaf	600s	Application dependent
DAO	Router	DTSN increment 360s	For each DIO with incremented DTSN and if parent switching
DAO	Leaf	Lower period between DIS and DTSN increment	Each time DIO with incremented DTSN & if parent switching
DAO ACK	Router Leaf	Same as DAO	When correctly received
DIS	Router	Not regular	Only when node needs infos to be attached to DAG
DIS	Leaf	360 s	To request DIO from parent and update DAG infos
ACK	Router, Leaf	Same as unicast	For each unicast frame sent

Afterward, the leaf node control traffic is independent from the topology. For multicast packets, a leaf mode gets back to sleep just after the sending whereas it waits for the acknowledgment for unicast packets. There is one exception for multicast DIS packets, where we forced the leaf node's transceiver in reception mode during a given duration, by default two seconds. This ensures that the node gets all DIO from all its potential parents. The energy cost needed to keep the transceiver active during this period is balanced by the efficiency of the parent selection during the lifetime of the network. Moreover, if we limit the parent selection only to the first DIO received, there is a risk that the parent access goes through a lossy link, or has a high RPL rank. All this resulting in suboptimal path or transmission retries and finally higher power consumption.

We studied our scenario's traffic under RPL in order to precisely determine the communication activity. Table 5 shows the size and the number of packets exchanges over a 24h period running the target monitoring application. The majority of the traffic is concentrated in NS/NA exchanges and data reporting. The overhead induced by the routing control messages is very low. NS/NA exchanges are required by our active data request mechanism and enable a periodical check of the bidirectional connection with the selected parent. They also allows to update the expected transmission count, ETX, metric accordingly to their exchanges success or fails, and MAC retries.

Table 5: Size and number of messages exchanged by a leaf node over a 24h period.

Mesg type	Size (B)	Sent	Recv	Comments
NS	72	151	0	
NA	72	0	151	
DIO	111	0	24	ETX metric, DODAG conf, prefix Info
DATA	93	144	0	UDP 10 bytes payload
DAO	66	24	0	target, transit Info
DAO ACK	40	0	24	requested for each DAO
DIS	33	24	0	
ACK	5	199	344	
Mesg		543	543	
Size (B)		28 892	17 128	

7. LEAF NODE POWER CONSUMPTION

Unlike generic power consumption models like in [20], we built our model according to our application. In our use case, a leaf node executes periodic tasks : sleeping, waking up, running, sensing, transmitting and receiving. We computed the energy according to our scenario with the software stack for the WSN different hardware components. The sleep mode current is integrated over the time of the simulation minus the active period of time. Throughputs are the same as in table 3, and the number of packet exchanges is deducted from the message periodicity presented in Table

4. We then computed the total cost over a day according to a wake up frequency of 4 Hz. We consider a raw platform without any probe and probes impact is presented independently in section 10. To complete the energy model of the radio, we add the energy spent when the transceiver stays in RX mode, waiting for acknowledgments. Notice that our simple model considers that all packets are successfully received and that there is no retry at the application or MAC layer. Table 1 gives the results for our reference platform with MSP430 at 16MHz and ATRF212 against the best up-to-date components.

We first investigate the power consumption of the low dropout voltage regulator, LDO, computed from its yield characteristics, with an input of 1.5V provided by a single AAA battery and the desired output voltage of 3.3V.

Table 6: Energy repartition for different leaf nodes.

	MSP430+ATRF212				SIM3C1+SI4461	
	With LDO		Without		Without	
	(J)	%	(J)	%	(J)	%
LDO	18.38	82	-	-	-	-
Radio Tx	0.59	3	0.59	14	0.57	34
Radio Rx	0.20	1	0.20	5	0.14	8
CCA - Back-off - Wait for Ack	0.03	<1	0.03	1	0.04	2
MCU Wake Up	2.09	9	2.09	51	0.76	45
MCU + RF Sleep	1.22	5	1.22	29	0.19	11
Total	22.51	100	4.13	100	1.71	100
Expect Life-time (Yrs)	1.31		7.16		17.24	

Table 6 shows that the LDO consumes 82% of the overall energy which is unacceptable. Relatively, the current consumed by the mote is very low, between 10 and 20 μA, and this fall in the range where the LDO efficiency is the worst (between 0,5 and 0,6), leading to a huge energy waste. To achieve an energy efficient design, this element must be removed. A battery with the required voltage, typically 3V, should be selected and connected directly to the mote. Moreover, as stated in [5, 10], the battery voltage cutoff should also correspond to the node components cutoff. The voltage regulation is not mandatory, as all components used on the mote can work over a voltage range between 2.1 and 3 V, matching the battery voltage range during its lifetime. Though, attention should be paid to the behavior of probe precision and clock drift against the voltage depletion. In our reference platform, we selected a 3V battery with a capacity of 1000 mAh, a self discharge current below 1 % per year, and a dropout voltage of 2.0 V. We took into account the 1% self discharge of the battery by subtracting 1% of the remaining energy at the beginning of each year.

The MSP430 is supposed to requires a minimum voltage of 2.2V to run at 16 MHz. We observed that it was able to run correctly with a voltage supply as low as 1.8 V meaning that the entire energy from the battery will be used.

After removing the LDO on our reference platform, the major part of the power consumption is due to MSP430 wake up (51%). The sleep mode of the RF transceiver and the MSP430 represents the second bigger power consumption (29%). This is an interesting result, because most of the literature about energy consumption in WSN considers the RF transceiver as the bigger energy consumer []. This focus is correct when looking at instantaneous power consumptions. However, when integrating over a long time, our results shows that the transceiver is not the main consumer if the periodic traffic is low like in our scenario.

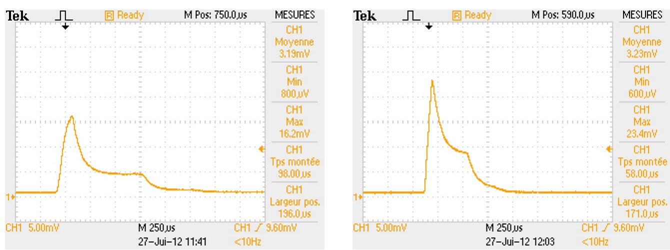
The expected lifetime is computed with a 1000 mAh battery. It reaches 7.16 years after removing the LDO, which matches the lifetime targeted by these nodes for smart energy applications. When injecting the characteristics of the ATRF230 and the CC1120 transceiver in the model, we found that the CC1120 and the ATRF212 design can achieve similar lifetime, around 7.1 years, and that their energy repartition are very closed. The lifetime of the platform embedding the ATRF230 transceiver achieve a 9 years lifetime thanks to its lower power consumption. Notice that the great difference of energy between these transceivers is balanced by the minor part of the radio in the overall power consumption. The new ARM architecture seems very promising because it reduces the leakage current and reduces the wake up cost due to its high frequency.

8. MCU SOFTWARE OPTIMISATION

Results from table 6 shows that the most important benefit consists in reducing the power consumption of the MCU in its wake up and sleep mode. While the sleep mode consumption depends on the hardware, we can reduce the cost of each wake up and decrease the wake up frequency.

During each wake up, ContikiOS runs several periodic processes governed by timers that are not mandatory in our application scenario. For instance, we disabled several timers related to routing maintenance, because a leaf node do not have routing ability. We also increased several timers, such as neighbor checking, neighbor unreachability detection (NUD), route lifetime checking and I/O checking, because our application do not require fine grain timing constraints. We eventually adapted some RPL timers to the leaf mode of operation, such as the DIO and DIS timer management, and timers that governs the sleeping mode.

In our building automation scenario, in order to keep an acceptable reactivity, a wake up frequency of 1 Hz is a good tradeoff in practice. We also activate the fast wake up function of the MSP that decrease the wake up time from 150 μs to only 5 μs.



8 MHz   16 MHz

Figure 2: Current average profile of MSP430 wake up at different frequency

We computed the average energy spent in each wake up after these optimizations from the new measurements plotted in table 8. This is respectively 2.3 and 2.6 times better than before these timers' optimizations and the corresponding expected lifetime moved from 7.16 years to 12.55 years.

The biggest part of the energy consumption is then due to the sleep mode of the components now representing 52% of the overall energy budget while the MCU wake up moved is only 13%.

When measuring experimentally the power consumption of the MSP430 in sleep mode, we found 4.3 μA. This is far above the expected value of 1.2 μA mentioned in the data sheet when using LPM3 mode because we were using the internal oscillator (REFO). According to the data sheet this oscillator consume 3 μA. The intend of this oscillator is to provide a precise clock at 32,768 KHz. In our case, we don't need such a precision. We eventually use the internal low power oscillator (VLO) and the power consumption matched the expected value of 1,2 μA under 3 Volts. With this configuration the average current consumption of the platform drops to 5.9 μA, giving a computed expected lifetime of 20.23 years.

9. RADIO SOFTWARE OPTIMIZATION

At this point, our study tells us that the next optimization should focus on radio transmissions that now represents 39 % of the total energy budget. First of all, we determined the impact of lowering the transmission power according to the transmission distance. This is presented in Table 3 where we adjusted the 868MHz transceiver's power in order to compare with the maximum range of the 2.4GHz transceiver. The resulting energy gain is small and the 2.4GHz transceiver stays 2 order of magnitude better. Also, with a transceiver at maxim gain, the transmission reaches the wider possible range with the best PDR. The only drawback concerns the interferences but with our seldom communication application and a good MAC layer, this is not really an issue.

We would rather focus on the limitation of the number of messages. When parsing the radio activity presented in table 5, it appears that 49.7 % of them are NS/NA, 13,6 % are RPL messages, and the remaining 36,7 % are data reports. Reducing the number of data messages is not in the scope of our study. We study how to reduce RPL and NS/NA messages. Initially, NS/NA exchange happened every 10 minutes between a leaf mote and its parent. We augmented this interval, at the price of a greater latency for downward message transmissions, and more sparse connectivity checking. Though, notice that if we have a data reporting every 10 minutes, the link is regularly checked during these transmissions, and we may remove NS/NA exchanges, with some modifications of the neighbor discovery mechanism implemented in ContikiOS. In order to reuse our downward frame exchange mechanism, the data pending flag would also need to be put in the data frame's acknowledgments.Thus, data messages would also update the ETX metric and check the upward link, limiting the overhead to RPL messages only. Such optimization of the NS/NA exchanges decreases the average current consumed by the node to 4 μA, leading to an expected lifetime of 28.5 years.

10. PROBES POWER CONSUMPTION

Embedded probes need some current for their functioning. Table 7 presents the power consumption for classical probes, running under 3V, used in WSN building monitoring.

Table 7: Probe power consumption.

Type	Vendor	Current (μA) active	Current (μA) standby	Duration (s)	Energy /yr (J)
Temp	TI TMP112	7	0.5	0.035	47
Temp Humid	Sensirion SHT21	300	0.15	0.114(HP) 0.015(LP)	20 15
PIR	PANASONIC EKMB1103112	1,9	1	2	95
Door Open	Meder KSK-1A66-1015 + M4 magnet	0.0006		NA	<1
Light	TAOS TSL2561T	240	3,2	0.25	312
CO2	AlphaSense IRC-A1	> 20000		455	>>1000

Embedding a CO2 probe, such as the one in Table 7, on a battery powered node is not wise since it depletes a 1000 mAh battery in only two days ! On the contrary the door/window opening probe is consuming very few energy, because it uses an ILS bulb, that do not require any power to operate. For other probes, there is great difference between the active and the standby consumption, which means that we should set up duty cycling to be efficient. The PIR probe cannot be optimized, because it automatically wakes it up when it detects something, and go back to sleep after 2 seconds. With the luminosity probe, the MCU needs to trigger a sensing window and retrieve the value acquired. As a result, the MCU can stay in sleep mode during the sensing, and retrieve the luminosity value during the next wake up. The window size can be optimized as well as the periodicity of the measures. In our scenario, because luminosity can vary quickly, the sensing window period lasts 250 ms and is triggered every 10 s while the MCU retrieves the value during its next wake up. The temperature and temperature/humidity probes need some processing from the MCU in active mode when reading these values, increasing substantially the overall cost of the probe reading. However, the periodicity of sensing can be large, because humidity and temperature have a low dynamic. For the SHT21, the time needed to do the measurements depends on the precision. For a full precision, 14 bits temperature, it needs a maximum time of 85 ms and for 12 bits humidity value, 29 ms. Overall, the CPU needs to be active during 114 ms.This time drops to 15 ms for 11 bit precision temperature and 8 bit precision humidity. The TMP112 probe has a default precision of 12 bit, and an extended mode that can be activated to measure temperatures greater than +128 C that far exceed our requirements. According to [16], the maximum conversion time for a 12 bit temperature value is 35 ms.

Table 8 summarizes the power consumption of the probes depending on their configuration. We computed their relative part in the total energy budget, considering a platform with the latest improvements implemented and presented in preceding Sections. This shows that embedding a probe on a mote can impact significantly the average power consump-

Table 8: Power consumption of various probe according to their configuration

Probe	Precision / Period	Power (μW)	Lifetime in node (yrs)	Part of energy (%)
Node	-	16.92	20.23	-
Temperature	12 bit / 10 s	56	4.69	77
Temperature	12 bit / 60 s	10,62	12.43	39
Temperature Humidity	Full / 10 s	186	1.64	90
Temperature Humidity	Full / 60 s	31.8	6.43	60
Temperature Humidity	Min / 10 s	25.2	7.34	54
Temperature Humidity	Min / 60 s	4,6	13,18	18
PIR	/ 60 s	3.18	17.03	16
Door Window Opening	/ 1 for 10 μs	0.0018	20.23	< 1
Luminosity	/ 10s for 250ms	26.9	7.80	61
Luminosity	/ 6 s for 250ms	12.5	11.61	43

tion and that great attention should be paid to its configuration. A greater precision can substantially impact the lifetime of a node. For instance, using the full precision mode of the SHT21 reduces by an order of magnitude the expected lifetime of the node, as compared to the minimum precision mode.

11. CONCLUSION AND FUTURE WORK

In this paper, we conducted an extensive power consumption study for the design of a wireless sensor network platform. We identified the different elements of the node, and assessed their relative energy consumption based on real power consumption measurements when possible and data sheet numbers. We discussed the power optimization of the MCU, the radio transceiver, the battery and a range of probes. Moreover, we discussed software optimizations, related to our application scenario, in ContikiOS and the RPL network stack.

We pointed out the key parameters that govern the energy consumption. We implemented all the energy improvements for the selected components in order to design a WSN node ensuring more than 10 years lifetime with a data reporting interval of 10 minutes. The results encompasses our aims and several probes can be added within the power budget.

We plan to further power our nodes with an energy harvesting system such as a solar PV panel and a super capacitor. In the home or building automation applications such an autonomous sensing node is or next target design. Given the average power consumption of our actual nodes that can be driven down to 17 μW, we are confident in the realization of such a design.

12. REFERENCES

[1] C. Bornmann. 6lowpan generic compression of headers and header-like payloads. IETF draft, 2011.

[2] A. Brandt, J. Buron, and G. Porcu. Home automation routing requirements in low-power and lossy networks. RFC 5826, April 2010.

[3] A. Dunkels. The ContikiMAC Radio Duty Cycling Protocol. Technical Report T2011:13, Swedish Institute of Computer Science, Dec. 2011.

[4] A. Dunkels, B. Gronvall, and T. Voigt. Contiki-a lightweight and flexible operating system for tiny networked sensors. In *29th Annual IEEE International Conference on Local Computer Networks*, pages 455–462. IEEE, 2004.

[5] J. Hill and D. Culler. Mica: A wireless platform for deeply embedded networks. *Micro, IEEE*, 22(6):12–24, 2002.

[6] IEEE. Ieee standard for local and metropolitan area networks–part 15.4: Low-rate wireless personal area networks (lr-wpans), 2011.

[7] J. Ko, S. Dawson-Haggerty, O. Gnawali, D. Culler, and A. Terzis. Evaluating the performance of rpl and 6lowpan in tinyos. In *IPSN*. ACM, 2011.

[8] N. Kushalnagar, G. Montenegro, C. Schumacher, and A. Danfoss. Ipv6 over low-power wireless personal area networks (6lowpans): Overview, assumptions, problem statement, and goals. RFC 4919, August 2007.

[9] J. McLurkin. *Algorithms for distributed sensor networks*. PhD thesis, Department of Electrical Engineering and Computer Sciences, University of California, 1999.

[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*, pages 364–369. Ieee, 2005.

[11] V. Raghunathan, C. Schurgers, S. Park, and M. Srivastava. Energy-aware wireless microsensor networks. *Signal Processing Magazine, IEEE*, 19(2):40–50, 2002.

[12] L. B. Saad, C. Chauvenet, and B. Tourancheau. Ipv6 (internet protocol version 6) heterogeneous networking infrastructure for energy efficient building. *Energy*, 44(1):447 – 457, 2012. Integration and Energy System Engineering, European Symposium on Computer-Aided Process Engineering 2011.

[13] Scilabs. Cortexm3 datasheet, 2012.

[14] Z. Shelby and C. Chauvenet. The ipso application framework, August 2012.

[15] Z. Shelby, K. Hartke, C. Bornmann, and B. Franck. Constrained application protocol (coap). IETF draft, 2012.

[16] Texas Instruments. Tmp112 datasheet, 2009.

[17] Texas Instruments. Msp4301611 datasheet, 2010.

[18] Texas Instruments. Msp430f5438a datasheet, 2010.

[19] Q. Wang, M. Hempstead, and W. Yang. A realistic power consumption model for wireless sensor network devices. In *SECON*. IEEE, 2006.

[20] Q. Wang and W. Yang. Energy consumption model for power management in wireless sensor networks. In *Sensor, Mesh and Ad Hoc Communications and*

Networks, 2007. SECON'07. 4th Annual IEEE Communications Society Conference on, pages 142–151. IEEE, 2007.