Research Article

A Sub-\(\mu\)A Ultrasonic Wake-Up Trigger with Addressing Capability for Wireless Sensor Nodes

Emanuele Lattanzi, Matteo Dromedari, Valerio Freschi, and Alessandro Bogliolo

Department of Basic Sciences and Foundations, University of Urbino, Piazza della Repubblica 13, 61029 Urbino, Italy

Correspondence should be addressed to Emanuele Lattanzi; emanuele.lattanzi@uniurb.it

Received 15 July 2013; Accepted 19 August 2013

Academic Editors: A. Bonastre and O. Sentieys

Copyright © 2013 Emanuele Lattanzi et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Wireless sensor nodes spend most of the time waiting either for sensed data or for packets to be routed to the sink. While on board, sensors can raise hardware interrupts to trigger the wake-up of the processor, incoming packets require the radio module to be turned on in order to be properly received and processed; thus, reducing the effectiveness of dynamic power management and exposing the node to unintended packets cause energy waste. The capability of triggering the wake-up of a node over the air would make it possible to keep the entire network asleep and to wake up the nodes along a path to the sink whenever there is a packet to transmit. This paper presents an ultrasonic wake-up trigger for ultra-low-power wireless sensor nodes developed as a plug-in module for VirtualSense motes. The module supports a simple out-of-band addressing scheme to enable the selective wake-up of a target node. In addition, it makes it possible to exploit the propagation speed of ultrasonic signals to perform distance measurements. The paper outlines the design choices, reports the results of extensive measurements, and discusses the additional degrees of freedom introduced by ultrasonic triggering in the power-state diagram of VirtualSense.

1. Introduction

The reduction of power consumption through the adoption of low-power design solutions and dynamic power management policies is mandatory in order to meet the needs of wireless sensor networks (WSNs). The average power consumption of on-board transceivers, at the time of writing, can be estimated to be around 20 mA (either in receiver or transmitter mode), which is commonly recognized as the main contributor to the energy drain of sensor nodes [1]. Since the radio module has to be powered on to enable the reception of incoming packets, strategies and techniques have been developed with the aim of limiting the impact of idle listening on power consumption. Current state-of-the-art solutions make use of pseudosynchronous rendez-vous schemes or purely asynchronous methods that exploit ad hoc hardware. The former, in all their variants, are protocols developed in order to allow efficient communication between duty-cycle-based receivers, which typically trade off latency for power consumption. The latter are techniques that exploit separate low-power wake-up receivers that are devoted to continuous monitoring of the communication channel in order to enable the main radio module to be switched off while idle to avoid energy waste.

A further possible approach to the problem entails the adoption of global clock synchronization. However, since achieving synchronization involves power-hungry mechanisms and protocol overhead, pure synchronized methods are not currently considered a viable solution in typical WSN scenarios [1, 2]. Asynchronous receivers are indeed considered a promising technique for sensor networks. As a matter of fact, there have been several efforts in designing efficient low-power asynchronous systems, some of which integrate addressing capabilities.

In this work, we present a novel addressable low-power wake-up receiver which exploits ultrasonic signals to selectively trigger sensor nodes. The contribution of the proposed approach is manifold. First, a sub-\(\mu\)A triggering system is...
proposed, which improves the state-of-the-art of wake-up receivers in terms of power consumption. Second, an open-hardware approach is adopted to provide to the scientific community a reusable platform made of low-cost off-the-shelf components. Third, accurate on-board internode distance measurements are obtained by exploiting the propagation speed of ultrasonic waves, thus providing valuable input data to many WSN algorithms, including localization and routing [3].

The rest of the paper is organized as follows: in Section 2, a description of background work concerning both pseudo-synchronous and asynchronous solutions is provided; in Section 3, the proposed approach is introduced, discussing design choices and implementation details; in Section 4, the results of experiments are reported in terms of energy consumption, performance, and distance measurement accuracy; in Section 5, the power state model of the mote is discussed; and in Section 6, the contribution of the work is summarized and conclusions are drawn.

2. Background

The issue of avoiding early battery depletion in WSNs is typically addressed either by means of specific networking protocols or through the development of ad hoc hardware solutions. Since radio channel monitoring is recognized as one of the main causes of energy consumption, while typical applications have low bandwidth requirements, powering down the transceivers during long and frequent inactivity periods represents a valuable option. This, in turn, prompts the need for strategies aimed at guaranteeing both the transmitter and the receiver to be simultaneously active in order to communicate.

2.1. Rendez-Vous Schemes. Rendez-vous schemes are specific communication protocols designed to ensure that both the transmitter and the receiver are active and ready to operate into a duty-cycle timing framework. Transceivers are switched on and off periodically and signaling packets (usually called beacons) are exchanged to make transmitters/receivers aware of the willingness/availability to communicate of their counterparts. In particular, rendez-vous policies where communication is started by the transmitter are called TICER (Transmitter-Initiated CycEd Receiver), while schemes where the communication is started by the receiver are known as RICER (Receiver-Initiated CycEd Receiver) [1, 2]. Needless to say, the parameters controlling the duty-cycle on-off pattern have opposite effects on power consumption and latency/throughput. TICER and RICER schemes can be substantially considered equivalent in terms of performance, except for the case of high-level fading, for which RICER has demonstrated to outperform TICER [2]. Recently, a new version of cycled receivers, namely, wake-on receiver, has been developed. Nodes equipped with wake-on capable radio modules are able to autonomously execute cycled channel listening without the need for any intervention of the microcontroller unit (MCU) [4, 5]. In this way, the MCU wakes up only when a packet has to be received by the node.

Since the MCU does not need to account for the radio channel sensing process, this results into further energy saving w.r.t. standard cycled receivers [4].

2.2. Wake-Up Receivers. Asynchronous strategies depend on ad hoc separate receivers in charge of waking up the MCU. Needless to say, the design of low-power separate devices is crucial for energy efficiency. Reducing the power consumption of these circuits is an active research area and several advancements have been achieved since the first proposed approaches [6–16]. Different types of solutions are usually compared in terms of power consumption, maximum reachable distance, operating frequency, latency, and sensitivity. Another (prominent) feature is also represented by the possibility of selective wake-up by means of out-of-band addressing [6, 8, 11, 14, 16].

2.3. Comparison. Asynchronous methods are considered advantageous in terms of power saving with respect to cycled receivers in many application scenarios [2]. Recently, simulations conducted on a wide range of wake-up receivers to evaluate lifetime prolongation of motes showed a substantial benefit coming from the use of asynchronous wake-up schemes compared with a pseudo-asynchronous wake-on solution [1]. In particular, the analysis undertaken compared different types of wake-up receivers [7, 8, 13, 15, 16], with a standard cycled wake-on receiver. Under common operating conditions, the results report percentage prolongations of node’s lifetime around 150% for most of the tested wake-up receivers. In summary, the analysis of scientific literature highlights the usefulness of wake-up receivers as solutions for mitigating the impact of radio listening on power consumption.

A common feature shared by all the techniques described so far is that they make use of radio signal propagation (with frequencies ranging from 433 MHz to 2.4 GHz) to establish communication. While ultrasounds have been proposed as an enabling technology mainly for localization tasks [17, 18], their use for the purpose of triggering sensor nodes has not been proposed so far. The only exception is a very recent work by Yadav et al. which presents preliminary thoughts and simulation results on the feasibility of ultrasonic-based wake-up receivers [19]. This paper goes beyond the results of Yadav et al. by presenting a working prototype made of off-the-shelf components, with an overall power consumption lower than that estimated for ad hoc solutions.

3. Design and Implementation

For the sake of generality, we assume each mote to be equipped with an ultrasonic module which accommodates both a wake-up receiver and a transmitter. The design of the ultrasonic module was driven by three main requirements: sub-μA standby consumption, sub-second wake-up time, and high availability of hardware components. The key component is a piezoelectric ultrasonic transducer with a center frequency of 40 kHz and a bandwidth of 2 kHz [20]. The baseline operating mode is based on carrier wave detection which causes the wake-up of all the receiving nodes in range. Selective wake-up
is optionally implemented on top of this simple mechanism by applying on-off-key (OOK) modulation to the ultrasonic carrier to encode the 8-bit address of the target receiver.

Figure 1 shows the schematics of the ultrasonic module. It is worth noticing that, for the sake of simplicity, transmitter and receiver do not share any analog components. Nevertheless, the overall circuit is small enough to be accommodated on a single VirtualSense [21] layer.

3.1. Ultrasonic Wake-Up Receiver. According to the nominal value of transmitting sound pressure level and receiving sensitivity of the ultrasonic transducers adopted [20], the signal level is around 100 mV at a receiver placed at 1 m from the transmitter. Hence, the signal needs to be amplified in order to extend the communication range and improve the detection capability of the ultrasonic module. A passive AC/AC transformer with a turn ratio of 1:10, expressly developed for ultrasonic communication (viz., K4000001 [22]), is used to this purpose in Figure 1 achieving a suitable amplification without adding to the power consumption of the module. The amplified signal is then analogically compared with an adjustable reference threshold by means of a MCP6541 ultra-low-power comparator with a quiescent current of about 600 nA at 2 V [23].

Thanks to the narrow bandwidth of the receiver-transformer chain, the noise level measured at the output of the transformer in the absence of an ultrasonic signal is in the order of 1 μVrms. This allows us to use sub-mV reference signal levels to increase the sensitivity of the receiver without impairing reliability. In the circuit of Figure 1, the reference level is provided by a resistive divider properly sized to minimize static power consumption. In particular, the resistive divider is made of a 9.4 MΩ resistor (R1) plus a variable resistor in the range of 0–20 KΩ (T1), drawing a maximum static current of about 200 nA at 2 V. In the experimental setup, the threshold was set to 0.7 mV, making it possible to detect ultrasonic pulses at a distance of up to 10 m.

At the limiting distance, in spite of the sensitivity of the receiver, the dynamic range of the signal is lower than the 3 mV hysteresis of the ultra-low-power comparator. This prevents the comparator to reset the output signal at the end of a detected input pulse. In principle, this is not an issue for a simple wake-up receiver, the main goal of which is to detect the rising edge of the triggering signal. However, this would make it impossible to decode an OOK-modulated signal possibly used to implement a selective wake-up mechanism. In fact, the hysteresis would prevent the ultrasonic module to distinguish the subsequent pulses used to encode the symbols.

In order to overcome such a limitation, a PIC12LF1840 sub-μA microcontroller [24] has been mounted on the ultrasonic layer to reset the comparator right after the detection of a pulse. Referring to Figure 1, the power supply of the comparator (pin 7) is directly provided by an I/O port of the microcontroller (pin 3), so that the reset can be easily achieved by means of shut down. The PIC12LF1840 is particularly suited for ultra-low-power applications since it draws only 20 nA in standby state and about 3 μA while running at 32 kHz.

In the proposed configuration, the microcontroller is normally in standby mode with the output pin 3 asserted in order to provide power supply to the comparator. As soon as a pulse is detected, the output of the comparator works as an interrupt signal for the microcontroller (pin 4), which causes it to switch to the running state and to reset the comparator.
As long as there is a detectable ultrasonic signal, the comparator keeps raising its output signal and the microcontroller keeps reacting by switching it off. A timeout is used by the microcontroller to detect the end of an ultrasonic pulse. At this point, the behavior of the microcontroller depends on the protocol adopted, as detailed in Section 3.3: in case of a simple wake-up trigger, the interrupt of the VirtualSense mote (INT) is directly raised at the end of the pulse; in case of an addressable wake-up, the entire address is received and compared in order to decide whether or not to wake-up the mote. The address assigned to the node is provided to the microcontroller of the ultrasonic module by the VirtualSense MCU by means of pins SDA and SCL.

The overall current drawn by the receiver in waiting mode is the sum of the three active components: the comparator (600 nA), the resistive divider (200 nA), and the embedded controller (20 nA), resulting in an overall standby power of about 1640 nW at 2 V.

3.2. Ultrasonic Transmitter. The 40 kHz carrier used to drive the ultrasonic transmitter (denoted by 400 PT [20]) is generated starting from a 40 kHz square waveform made available by the microcontroller (viz., the PIC12LF1840 introduced in Section 3.1) on I/O pin 7. The square wave is then used to drive a transistor-transistor-logic (TTL, [25, 26]) stage which provides the supply current required to drive the transmitter through a passive filter and a 1:10 transformer, as shown in Figure 1.

It is worth noticing that OOK modulation is directly performed by the microcontroller which generates the 40 kHz square wave. This is done according to the protocol described in the next section.

The standby power consumption of the ultrasonic transmitter is mainly due to the standby current of the microcontroller (about 20 nA), which was already accounted for in the power budget of the receiver as detailed in previous subsection. In fact, the microcontroller is shared by the transmitter and the receiver, so that its standby current has to be accounted for only once to compute the overall power consumption.

On the contrary, the power consumption of the circuit in transmit mode is given by the active state of the PIC12LF1840 and by the current drawn by the TTL stage, leading to an overall consumption of 37 mW at 2 V during the transmission of an ultrasonic pulse.

3.3. Ultrasonic Triggering Protocol. The ultrasonic module presented so far supports two triggering modes: broadcast wake-up and unicast wake-up.

The broadcast wake-up protocol entails a single ultrasonic burst of 50 ms to be transmitted by the triggering node and detected by the ultrasonic modules of all the nodes in range. The receiving nodes are woken up by their triggering circuitry as soon as the end of the burst is detected, that is, when a 25 ms timeout has elapsed without receiving any ultrasonic signal.

The unicast wake-up protocol entails the transmission of a preamble, which is nothing but the 50 ms burst used for broadcast wake-up, followed by the 25 ms timeout and by the address of the destination node, encoded by means of OOK modulation. In our experimental setup, the transmission of each symbol takes 50 ms as shown by the timing diagram of Figure 2, where CMP OUT refers to the output of the comparator (node B in Figure 1), CMP EN refers to the power supply of the comparator as provided by the microcontroller (node C in Figure 1), while INT represents the interrupt signal generated by the receiver to wake up the MCU of the VirtualSense mote.

If the nodes are configured to work in unicast wake-up mode, the first burst is regarded as a preamble used to denote the beginning of the transmission of the address without requiring any rendez-vous. As soon as the end of the preamble is detected, the comparator is switched off for 25 ms and then powered to receive the first bit of the address in the second half of the symbol period. The reset of the comparator is repeated after each bit until the overall address is received. Once all the 8 bits are received and sampled by the microcontroller, the received address is compared with the local one and the INT signal is raised if they match.

With 8-bit addresses, unicast wake-up takes almost 0.5 s. Although this can be regarded as a sizeable latency, using a symbol time of 50 ms allows us to keep the operating frequency of the microcontroller at 32 kHz, with a sizeable benefit in terms of power consumption. On the other hand, in case of applications subject to tight real-time constraints, the symbol time could be shortened by raising the operating frequency of the microcontroller, thus trading power consumption for reactivity.

It is also worth mentioning that the address transmission protocol is not optimized. For instance, the reception of the address could be stopped as soon as the first mismatching bit is detected, thus saving overhearing power. Also, address reuse policies could be explored in order to allow the reduction of the address length. Protocol optimization, however, goes beyond the scope of this work.

4. Experimental Results

This section reports and discusses the results of extensive experiments and measurements performed on VirtualSense motes equipped with a prototype of the ultrasonic wake-up layer shown in Figure 3.

In particular, Section 4.1 outlines the experimental setup, Section 4.2 reports the results of power and performance measurements, while Section 4.3 discusses the suitability of the ultrasonic module to perform pairwise distance measurements.

4.1. Measurement Setup. Digital waveforms were sampled by means of a National Instruments NI-DAQmx PCI-6251 16-channels data acquisition board connected to a BNC-2120 shielded connector block [27, 28], while current waveforms were collected by means of a National Instruments PXI-4071 digital multimeter with a sensitivity down to 1 pA [29].

During the experiments, the ultrasonic module was powered at 2 V by a NGMO2 Rohde & Schwarz dual-channel
power supply serially connected with the PXI-4071 configured as an amperometer [30] to capture the supply current drawn by the circuit. Current waveforms were sampled at 100 kHz.

The experimental setup involved two VirtualSense motes equipped with the ultrasonic module: the device under measurement, working either as a transmitter or as a receiver, and a second device playing the complementary role in each communication experiment.

4.2. Power-Performance Characterization. Power and performance measurements were performed by running ad hoc synthetic benchmarks on the two VirtualSense motes described in the experimental setup. The mote acting as a transmitter was programmed to issue an ultrasonic wake-up signal every 2 seconds, while the one acting as a receiver was waiting in an inactive state for the INT signal raised by its ultrasonic wake-up receiver. Each wake-up was acknowledged by the receiving mote by sending an echo radio packet back to the transmitter. In order to test unicast wake-up, the receiving mote was flashed with a unique 8-bit address (namely, “10101010”).

Figure 4 shows the signal waveforms sampled in three nodes of the wake-up receiver in case of unicast wake-up: (a) the output of the ultrasonic transducer (node A in Figure 1), which is a modulated sinusoidal waveform at 40 kHz with an amplitude of about 100 mV; (b) the output of the comparator (node B), which is a modulated square waveform at the same frequency with an amplitude compatible with the TTL signals, ranging from 0 to 2 V; and (c) the I/O port of the microcontroller used to power the comparator (node C). The experiment was conducted by placing the two motes at a distance of 1 m and by powering the ultrasonic modules at 2 V.

Figure 4(a) clearly shows the 50 ms preamble, followed by a silent period and by the 8 bits of the address (bursts correspond to the 1’s, while 0’s are simply encoded by the lack of signal according to the OOK modulation). Figure 4(b) shows the output of the comparator, reshaped by the microcontroller through the enable signal shown in Figure 4(c). In particular, the microcontroller issues periodic resets during the preamble (causing the narrow glitches in the first part of the curve) and provides the 25 ms sampling windows for the bits of the address.

Figure 5 shows the measured current consumption of the ultrasonic module acting as a wake-up receiver in case of unicast wake-up. The baseline supply current is of 874 nA, thus meeting the sub-μA design goal of the triggering module. Whenever an over-threshold signal is detected, the supply current jumps to about 7 μA needed by the comparator to generate the TTL output signal, and by the microcontroller to execute the protocol. It is worth noticing that the contribution of the comparator to the overall power consumption is almost
negligible during the reception of a 0. In real world scenarios, this could be taken into account for energy-aware address assignment. The overall measured energy spent to receive a unicast wake-up is of $3.75\mu J$. In case of a broadcast wake-up, the curves reduce to the preamble, and the overall energy reduces to $0.75\mu J$.

The current drawn by the ultrasonic transmitter to issue the unicast wake-up to a target mote assigned with address “10101010” is reported in Figure 6. The five bursts correspond to the transmission of the preamble followed by the four 1’s of the address. Within each burst, the current waveform varies between 0 and 65 mA at 40 kHz, while narrow glitches are measured at the beginning and at the end of each burst. The overall energy measured with 2 V power supply is of 16.25 mJ. It is worth noticing that no energy is spent to transmit a 0. As already observed for the receiver, this feature of the OOK modulation could be exploited to conceive energy-aware encoding and address assignment policies.

The overall energy consumed by the ultrasonic layer for transmitting or receiving wake-up signals is summarized in Table I for both unicast and broadcast wake-ups.

### Table I: Energy spent by the ultrasonic module for transmitting and receiving broadcast/unicast wake-ups.

<table>
<thead>
<tr>
<th></th>
<th>Energy consumption</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Unicast</td>
</tr>
<tr>
<td>Transmitter</td>
<td>16.25 mJ</td>
</tr>
<tr>
<td>Receiver</td>
<td>3.75 $\mu J$</td>
</tr>
</tbody>
</table>

4.3. Distance Measurement. The propagation speed of sound in air ($s_{\text{sound}}$) can be exploited to estimate the distance between a transmitter and a receiver by measuring the time of flight of an ultrasonic signal. The accuracy of the estimated...
The distance \( \tilde{d} \) depends on the resolution of the time measurement \( t \) and on the variance of the overhead introduced by the measurement system \( t_{\text{overhead}} \).

The measurement protocol adopted to measure the distance between two VirtualSense motes (hereafter denoted by T and R) consists of (i) T and R turn on their MCU, radio module, and ultrasonic module; (ii) T resets a timer and transmits an ultrasonic pulse; (iii) R detects the ultrasonic pulse and transmits an echo radio packet to acknowledge the reception; (iv) T receives the radio packet, stops the counter, and computes the corresponding distance as

\[
\tilde{d} = (t - t_{\text{overhead}}) s_{\text{sound}}.
\]  

(1)

It is worth noticing that the time measurement is performed by T, while R is used only to send back an echo response across the radio channel. Hence, the time overhead includes the software running at both ends, the ultrasound detection time, and the time of flight of the radio packet. In principle, this last contribution depends on the distance between the two motes, but it can be neglected in (1) since the propagation speed of the radio packet is about 6 orders of magnitude faster than that of sound. In the distance range of the VirtualSense ultrasonic modules, the time of flight of a radio packet is shorter than the clock period of the MCU.

The accuracy of the distance estimates provided by the VirtualSense ultrasonic modules was tested by clocking the MCUs at 24 MHz, by using echo packets of only 1 byte, and by setting the maximum resolution for the timer, namely, 250 \( \mu \)s. Measurements were performed by positioning the motes at 100 different distances randomly chosen with uniform distribution in the range between 0.3 m and 10.0 m. The motes were accurately positioned and the measurements were repeated five times in each point. Half of the measures were then used to characterize the model of (1), while the remaining ones were used to evaluate out-of-sample accuracy.

Figure 7 shows the relation between the nominal distance and the measured time for the 50 points used for characterization. Error bars denote the standard deviation computed over the 5 measures performed in each point. The dashed curve represents the linear regression (with \( R^2 \) close to 1) the equation of which (also reported in the figure) provides the parameters of (1), namely, \( t_{\text{overhead}} = 4.8015 \) and \( s_{\text{sound}} = 1/3.0107 \).

The out-of-sample errors made by (1) are plotted in Figure 8 as a function of the actual distance. Error bars denote the standard deviation. The average absolute error is of about 0.06 m, which is below the theoretical resolution of the proposed methodology which is of about 0.09 m for a time resolution of 250 \( \mu \)s.

5. State Model

The simplest functional model which allows us to point out the distinguishing features of VirtualSense and the additional design options provided by the ultrasonic wake-up mechanism is composed of two states: \textit{Wait}, in which the mote is inactive and waits for incoming events, and \textit{Process}, in which the mote is active and performs some task. In general, a sensor node can react to three types of events: incoming packets, timed wake-ups, and sensing events. Since the ultrasonic trigger does not affect the capability of VirtualSense to react to wake-up and sensing events, the only outgoing edges of the Wait state represented in Figure 9 are the ones triggered by incoming packets. In particular, the overhearing of an unintended packet is represented as a self-loop of the Wait state, since it does not cause the execution of any processing task, while the reception of an intended packet causes a transition from Wait to Process. The transmission of a packet is modelled as a self-loop of the Process state, since it requires the mote to be active and it does not necessarily imply the mote to become inactive right after the transmission.
Transitions from Process to Wait are simply triggered by End-of-Process (EoP) events.

In order to make the state model of Figure 9 expressive enough to perform power and performance estimations, the average power consumption needs to be associated with each state, while transition time and energy need to be associated with each edge. The actual values of power, energy, and transition time depend on the way in which the power states of VirtualSense and the ultrasonic wake-up mechanisms are exploited to implement the target application. In particular, while Process requires the microcontroller unit to be active, Wait allows the mote to exploit any low-power inactive state, possibly combined with out-of-band ultrasonic triggering and addressing solutions, in order to find the most suitable tradeoff between power consumption and reaction time.

VirtualSense provides 4 families of power states [21]: Active, which executes a task on top of the Java compatible runtime environment provided by the Darjeeling virtual machine; Standby, where the CPU is switched off while both the clock and the memory are powered; Sleep, where both the CPU and the clock are switched off, but an external RTC is used to provide timing information and to grant to the mote self-wake-up capabilities; and Hibernation, where main memory is turned off and the heap of the Java VM needs to be stored in flash memory in order to provide state retention capabilities. In addition, memory-less hibernation (ML-Hibernation) can be used whenever the application does not require state retention, thus saving the energy otherwise required to store the heap.

In the baseline configuration, the Contiki MAC protocol [31] exploits the Standby mode of the MCU and the LPM1 mode of the radio module for duty cycling. All other configurations are obtained as a combination of the inactive states of the MCU (Standby, Sleep, Hibernation, and ML-Hibernation), the frame filtering option [21] of the radio module (FF), the ultrasonic triggering mechanism (US), and the ultrasonic addressing protocol introduced in this paper (USa). In all the configurations, but the baseline, the radio module is turned off whenever idle instead of being put in LPM1.

Table 2 reports the parameters of the state diagram of Figure 9 for all the relevant configurations of the VirtualSense mote. The first part of the name of the configuration denotes the inactive mode of the MCU exploited in the Wait state, while suffixes FF, US, and USa are used to denote the adoption of the corresponding solutions.

Since Table 2 is used to show the dependence of the parameters from the configuration of VirtualSense, those which are not significantly affected by the configuration are not reported in the table for the sake of simplicity. In particular, the average power consumption of the mote in Process state is always around 13.44 mW, while the time and energy spent to go from Process to Wait in case of EoP are negligible (below 1 ms and 1 µJ, resp.) for all the configurations but Hibernation, for which they take value 78.8 ms and 1.24 mJ, respectively, to save the heap of the VM. The time required to receive an unintended packet is not reported in the table in spite of its dependence on the configuration, since it does not entail any state transition.

The bar graph of Figure 10 shows, for each configuration of VirtualSense, the tradeoff between the power associated with the Wait state and the time required to receive an incoming packet. The competitive advantage of ultrasonic triggering is apparent in that it allows us to reduce the average power consumption of the mote about three orders of magnitude without impairing reactiveness to incoming events. It is worth mentioning that the configuration of VirtualSense motes can be selected at runtime by means of a specific API so that all the power-performance tradeoffs are made directly available to the Java programmer. The selective wake-up mechanism provided by ultrasonic addressing adds to the time required to send/receive a packet, but it significantly reduces the energy lost for receiving unintended packets, as shown in Table 2.

6. Conclusions

Keeping the routing nodes active to listen for incoming packets to be possibly forwarded or routed to the sink is one of the most energy-consuming tasks in wireless sensor networks, which limits the lifetime of battery-operated nodes and makes it hard to rely on unpredictable renewable energy sources. Remote triggering provides an attractive alternative by allowing the transmitting nodes to wake-up their neighbors acting either as routers or destination nodes. As long as the routers can react to the triggering events in time to receive the packets, they can take advantage of idle periods by staying in inactive low-power states. This enables the full exploitation of the degrees of freedom available for dynamic power management, and it reduces the energy usually wasted by overhearing unintended packets. On the other hand, the triggering circuitry has a power overhead that has to be carefully taken into account in order not to impair the benefits.

Existing approaches to remote triggering are mainly based on radio signals, exploiting either the same channel used for communication or a dedicated one. This paper has
explored the possibility of using out-of-band ultrasonic triggering, as an alternative to radio triggering which also provides the possibility of performing accurate pairwise distance measurements.

The ultrasonic module, composed of the transmitter and the wake-up receiver, has been implemented as a layer of VirtualSense motes and fully characterized in terms of power and performance by means of real-world measurements. Experimental results show that the overall module consumes less than $1\,\mu A$, when powered at 2 V, without losing the capability of reacting to ultrasonic triggering events.

Power and performance measurements have also been used to characterize a power state model which shows the effectiveness of both unicast and broadcast wake-up to reduce the idle power consumption of VirtualSense motes of up to three orders of magnitudes without significantly increasing the reaction time.

Extensive experiments have been also performed and reported to evaluate the suitability of the ultrasonic module to compute pairwise distances from the measured time of flight. Experimental results report an average absolute error of about 6 cm for distances ranging from 30 cm to 10 m.

Directional ultrasonic transducers were used for running the experiments reported in this paper. The directionality of the transducers installed on the motes results in the spatial selectivity of the wake-up signals they can issue, independently of the addressing mechanisms adopted. Such selectivity further reduces the power consumption by preventing the nodes which are outside the beam angle from receiving unintended wake-ups. On the other hand, directionality impairs flexibility and adaptability in that it requires the adoption of static routing schemes to be decided at deployment time. The approach proposed in this paper can be easily extended to support dynamic/multipath routing and adaptation by installing omnidirectional transducers in place of the directional ones. Selectivity can still be provided by adopting the unicast wake-up protocol.

### Table 2: Parameters of the state diagram of Figure 9 for all the relevant configurations of VirtualSense. All the parameters were measured on a mote powered at 3 V, with the MCU clocked at 16 MHz and the ultrasonic module powered at 2 V.

<table>
<thead>
<tr>
<th>Configuration</th>
<th>Wait [µW]</th>
<th>Unintended [µJ]</th>
<th>Receive [ms]</th>
<th>Transmit [µJ]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>3,451.5</td>
<td>1,502.3</td>
<td>73.4</td>
<td>1,559.44</td>
</tr>
<tr>
<td>Standby</td>
<td>747.5</td>
<td>1,502.3</td>
<td>73.4</td>
<td>1,559.44</td>
</tr>
<tr>
<td>Standby FF</td>
<td>747.5</td>
<td>521.4</td>
<td>73.4</td>
<td>1,559.44</td>
</tr>
<tr>
<td>Sleep</td>
<td>734.4</td>
<td>1,502.3</td>
<td>73.4</td>
<td>1,559.44</td>
</tr>
<tr>
<td>Sleep FF</td>
<td>734.7</td>
<td>521.4</td>
<td>73.4</td>
<td>1,559.44</td>
</tr>
<tr>
<td>Sleep US</td>
<td>3.678</td>
<td>1,503.2</td>
<td>28.2</td>
<td>1,560.31</td>
</tr>
<tr>
<td>Sleep US FF</td>
<td>3.678</td>
<td>522.3</td>
<td>28.2</td>
<td>1,560.31</td>
</tr>
<tr>
<td>Sleep USA</td>
<td>3.678</td>
<td>3.87</td>
<td>428.2</td>
<td>1,563.31</td>
</tr>
<tr>
<td>Hibernation US</td>
<td>2.718</td>
<td>5,987.5</td>
<td>564.8</td>
<td>6,044.6</td>
</tr>
<tr>
<td>Hibernation US FF</td>
<td>2.718</td>
<td>5,006.5</td>
<td>564.8</td>
<td>6,044.6</td>
</tr>
<tr>
<td>Hibernation USA</td>
<td>2.718</td>
<td>3.87</td>
<td>964.8</td>
<td>6,047.6</td>
</tr>
<tr>
<td>ML-Hibernation US</td>
<td>2.718</td>
<td>1,547.0</td>
<td>31.49</td>
<td>1,604.1</td>
</tr>
<tr>
<td>ML-Hibernation US FF</td>
<td>2.718</td>
<td>566.1</td>
<td>31.49</td>
<td>1,604.1</td>
</tr>
<tr>
<td>ML-Hibernation USA</td>
<td>2.718</td>
<td>3.87</td>
<td>431.49</td>
<td>1,607.1</td>
</tr>
</tbody>
</table>

### Conflict of Interests

The authors declare no conflict of interests.

### References


