Zero Power Energy-Aware Communication for Transiently-Powered Sensing Systems

Alessandro Torrisi
University of Trento, Italy
alessandro.torrisi@unitn.it

Davide Brunelli
University of Trento, Italy
davide.brunelli@unitn.it

Kasım Sinan Yıldırım
University of Trento, Italy
kasimsinan.yildirim@unitn.it

ABSTRACT

Battery-less wireless sensors powered directly by miniaturized energy harvesters can be appealing only if communication between nodes is realized without wasting energy. In devices that implement *intermittent computing*, efficient communications remain an open challenge. Transmitters should be aware of unavailable receivers to prevent packet losses due to power failures. Backscatter transmissions can be used to propagate the energy state almost for free in the surrounding. This paper presents a backscatter radio mechanism and a protocol that regulates the communication between nodes, guaranteeing packet transmissions only if sufficient energy is stored in the transmitter and the receiver. Simulation results demonstrate our approach's effectiveness and show the performance of this new type of *intermittent communication*.

CCS CONCEPTS

• **Networks → Network simulations**; • **Computer systems organization → Sensor networks**; • **Hardware → Power and energy**.

KEYWORDS

Transiently-powered Communication, Batteryless Sensors, Energy Awareness

1 INTRODUCTION

The emergence of batteryless platforms enabled a new application space from body implants and wearables [9] to the deployments in extreme locations, even in space [7]. Contemporary batteryless platforms, such as WISP [23], Flicker [8] and Camaroptera [16], harvest energy from several ambient sources (e.g., solar [16], radio-frequency (RF) [23], bacteria species [22]) and store the harvested energy into a capacitor. This capacitor stores a marginal amount of energy and powers the microcontroller, sensors, radio, and other peripherals. On the one hand, these peripherals drain the capacitor frequently. On the other hand, ambient energy sources are sporadic and provide power transiently. Therefore, batteryless platforms operate intermittently, which is composed of charge, sense/compute/send, and die cycles.

There is an ample body of research focusing on the intermittent computing on batteryless platforms, e.g., [2, 3, 5, 12, 17, 18, 26]. The main challenge of intermittent computing is to preserve the forward progress of computation and memory consistency despite frequent power failures. The reason is that power failures reset the volatile state of the device, e.g., registers and the contents of the memory. Therefore, the outputs of the past computation are lost. Without recovering these values, the computation cannot continue from where it left. Several software-based techniques are proposed to enable intermittent computing considering different requirements, such as reduced programmer burden, systems overhead, and timely execution.

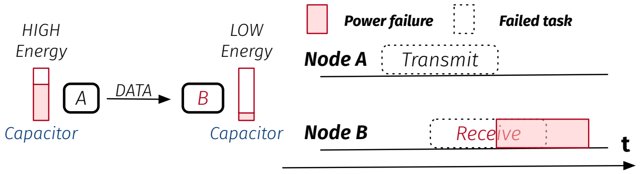


Figure 1: Two transiently-powered devices miss packets during communication due to unpredictable power failures. When a high-energy transmitter device transmits information to a low-energy receiver, there might be a packet loss due to a power failure. Therefore, the energy spent during transmission and reception might be wasted. A coordination mechanism is required to ensure packet delivery and eliminate the waste of energy.

Zero-power Communication. Due to the energy constraints of batteryless systems, radio transmission using active radios are costly. Backscatter communication, implemented by traditional RFID tags, enables almost zero-power communication by eliminating the energy-hungry hardware components of active radios, e.g., power-hungry mixers that generate carrier waves. Therefore, backscatter is a perfect choice for batteryless devices considering their marginal energy budgets. In traditional backscatter, tags transmit by modulating the reflections of RF signals generated by a dedicated reader, that requires several orders of magnitude less energy than transmission with active radios [27]. Recent work demonstrated bidirectional zero-power communication among batteryless devices, without the need for a dedicated reader [14].

Transiently-Powered Communication Problem. Despite the aforementioned progress achieved in zero-power communication, the intermittent operation of the transiently-powered nodes is overlooked in the literature. In particular, the prior work on zero-power communication assumed that the batteryless devices are powered continuously during data communication, e.g., via the RFID reader. However, transiently-powered batteryless devices operate intermittently, and they are not always available. Without coordination, the success of packet delivery among these devices is always random and subject to a significant amount of failures. Consider the scenario depicted in Figure 1, where a node with high energy wants to transmit data to a node with low energy. Since the receiver node does not have enough energy to receive the packet:

(1) the transmitted packet will not be received since the receiver will drain its storage system and will die due to power failure;

(2) due to the packet delivery failure, the energy spent on both sides, i.e., transmitter and receiver, will be wasted.

For successful communication, the stored energy on both sides of the channel should be sufficient to perform packet transmission and reception. A coordination mechanism is required to guarantee packet delivery and eliminate the waste of energy.

Contributions. In this paper, we aim at the design of a novel communication protocol for transiently-powered sensor devices. Transiently-powered communication requires a notion of coordination between the transmitter and receiver so that the transmitter device knows beforehand the receiver device's availability and starts transmitting its data. In different terms, we need to ensure both nodes simultaneously in a high energy level status before a packet transmission. To this end, we make the following contributions:

1. **Zero-Power Energy Status Circuit.** We present a novel circuit design, based on the backscatter circuitry presented in [14]. This circuit encodes the energy level of the storage capacitor by turning on a fixed frequency low-cost and ultra-low-power oscillator for a specific time. Thus, creating a burst with different duration (i.e., number of pulses) related to the energy status.
2. **Transiently-Powered Communication.** To the best of our knowledge, we introduce the first communication protocol, named TRAP (TRAnsiently-powered Protocol), for transiently-powered devices. TRAP relies on the energy status information, transmitted by the energy status circuit over the energy status channel. By using the energy state signal of a neighbor, a transmitter node can initiate data transmission on the data transmission channel, using a different backscatter radio dedicated to only data transmission.

We believe that our work proposes the first attempt to introduce the fundamental hardware support and the building block of future transiently-powered networking protocols.

2 RELATED WORK

The marginal energy budgets and intermittent operation of transiently-powered batteryless devices bring about several research challenges within the context of computation and communication.

2.1 Intermittent Computing

The intermittent operation of the transiently-powered devices prevents existing software designed for continuously-powered computers from being run correctly due to frequent power failures and loss of the computational state. In particular, power failures might hinder the forward progress of computation and lead to memory inconsistencies. The researchers proposed instrumenting existing programs with checkpoints to save the device state (e.g., registers, contents of the volatile memory) in non-volatile memory. Upon power failure, the device state can be restored from the latest checkpoint, and the computation can progress with consistent memory content [3, 12, 17]. Another approach is to rewrite existing programs using task-based programming models [5, 26] that offers an efficient alternative to checkpoints but require a non-trivial code transformation.

2.2 Zero-Power Communication

Backscatter communication enables almost zero-power wireless communication, and it is a perfect choice for energy-constrained batteryless devices. Most of the traditional backscatter networks [1, 24] allow one-way (unidirectional) communication. This means tags can only communicate directly with a dedicated master device (e.g., an RFID reader). With this approach, the decoding of the received weak backscattered signal, which requires complex digital signal processing techniques, is pushed to the RFID reader side. The consequence is the simplified design and reduced the energy requirements of the batteryless devices. The bidirectional communication among batteryless devices has also been demonstrated in [13, 14, 20, 21], also drastically reducing the complexity of the RF illuminator. This is enabled without the need for a dedicated RFID reader, by decoding the received signal using only low-power analog operations (e.g., using envelope detectors that require components like capacitors, diodes, operational amplifiers and comparators). There are also hybrid systems proposed [11, 19], i.e., systems that are equipped with an active and a backscatter radio. In all these prior work, the devices are continuously-powered during backscatter communication. Therefore, the researchers overlooked the intermittent operation of batteryless devices during data communication.

3 ENERGY STATUS CHANNEL

As indicated previously, the stored energy on both sides of the channel must be sufficient to perform the packet transmission and reception operations successfully. Otherwise, the packet transmission will be interrupted since either the transmitter or the receiver devices will die upon to a power failure. This will lead to i) a significant number of packet losses, and ii) a waste of precious harvested energy at the transmitter and receiver sides. Therefore, the transiently-powered devices must obtain the *energy state information* from their neighbors to decide either to trigger or give-up transmission.

Now, the question arises is how to transmit the energy status considering the marginal energy budgets of the transiently-powered devices. Due to its ultra-low-power operation, we consider backscatter communication to transmit and receive energy status information. Backscatter generates a modulated RF signal by the reflection of the incident RF power and allows zero-power communication for the end-nodes. In this case, the main challenge is how to modulate the RF signal to encode the energy state information without using power-hungry components and circuits. In this section, we present a dedicated circuitry, based on the backscatter transceiver presented in [14], that modulates the energy status via adjusting the on-air time of an ultra-low-power and low-frequency oscillator.

3.1 Energy Status Backscatter Circuitry

Figure 2 depicts the proposed circuitry, which is composed of a transmitter (TX) block that modulates the RF signal and a receiver (RX) block that demodulates this signal to extract the transmitted energy status information. The crucial component that facilitates the energy status modulation is the low-frequency and ultra-low-power oscillator.

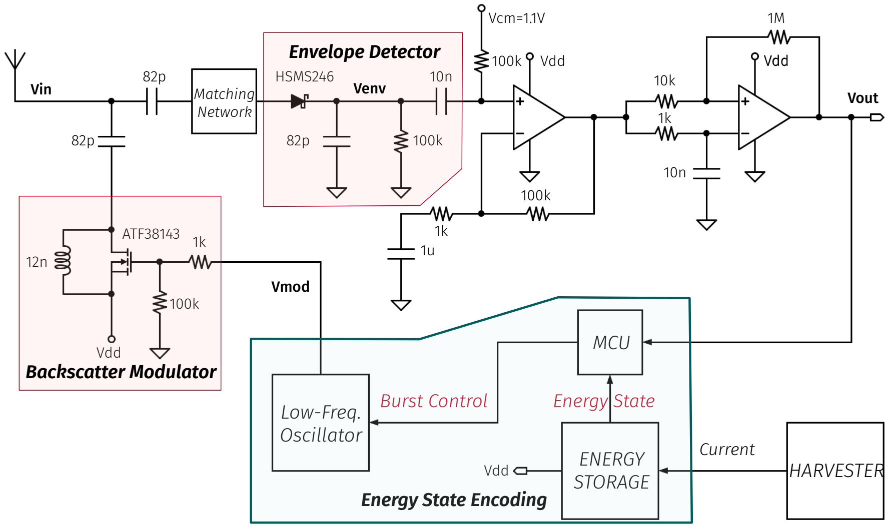


Figure 2: The main blocks of the backscatter transceiver presented in [14] and the proposed additions for the energy state information encoding and decoding. Based on the voltage level of the energy storage, the microcontroller adjusts the duty-cycle of the low-frequency oscillator to encode the energy status. The backscatter signal is modulated by the low-frequency oscillator, by controlling the (Vmod) signal, to send the encoded energy status information. On the receiver side, the input backscatter signal (Vin) is detected by the envelope detector, whose output signal (Venv) is filtered and amplified to form the output digital signal (Vout).

Energy Status Transmission. The backscatter transmitter mismatches the antenna by exploiting RF switches and different match impedances. A single MOSFET (or an analog RF switch, e.g., the ADG904 presented in [14]) can be used as a switch to support a simple ON-OFF keying modulation. When the switch is open, the antenna is matched. In this case, very little of the input signal is reflected by the antenna. When the switch is closed, the antenna is mismatched. In this case, the antenna reflects the input signal. The modulation, i.e., turning on and off the switch, is driven by the `Vmod` signal depicted in Figure 2. We propose to drive the modulation, i.e., the control of the `Vmod` signal, by a low-power and low-frequency oscillator. This oscillator can even be tuned at a specific frequency that identifies and differentiates multiple end-nodes. Specifically, the following steps are taken during energy status transmission:

(1) The microcontroller samples the voltage level of the energy storage (through an analog-to-digital converter ADC);
(2) Considering the measured energy level, the microcontroller selects a burst duration to encode the information regarding the energy status;
(3) Based on the duty cycle period, the microcontroller turns the oscillator on and off, which leads to a burst whose length is defined by the duty-cycle.

As an example, while the node is in a charging transient and/or the energy is too low to compute specific tasks (e.g., as receiving information from the neighboring nodes), the oscillator can be kept active only for a short time. Hence, the burst duration will be small (e.g., the burst has 10-20 pulses). On the contrary, if the available energy is enough to guarantee communication, the oscillator can be kept active for a longer period, in turn, the burst becomes longer (e.g., the burst has 100 pulses).

Energy Status Reception. The backscattered energy status information can be decoded on the receiver side when the end-node has enough energy to perform the necessary computation. The RX block presented in [14] can be used in our case without any modifications. This backscatter receiver is specially designed as a demodulator for simpler modulations such as ON-OFF keying with lower data-rates. This is enabled by exploiting an envelope detector to perform the frequency shift in the baseband by implementing a low-power and cheap circuit. The main actor is a biased Schottky diode envelope detector, which is finely matched with the RF input (`Vin`) and the antenna (see Figure 2) at the frequency of 868MHz. The remaining circuitry aims to optimize the voltage swing of the low frequency demodulated signal using a high pass filtering amplifier stage and a comparator for the final digital output `Vout`. Specifically, the receiver circuitry takes the following steps during energy status reception:

(1) The digital output `Vout` interrupts the microcontroller, that represents high-to-low and low-to-high bit transitions;
(2) The microcontroller accumulates the bits and forms the transmitted burst;
(3) Considering the duration of the burst, the microcontroller decodes the energy status and decides to transmit data, avoiding the problem of packet-loss due to power failures.
(4) Considering the frequency of the received and decoded burst, the MCU identifies the transmitter node.

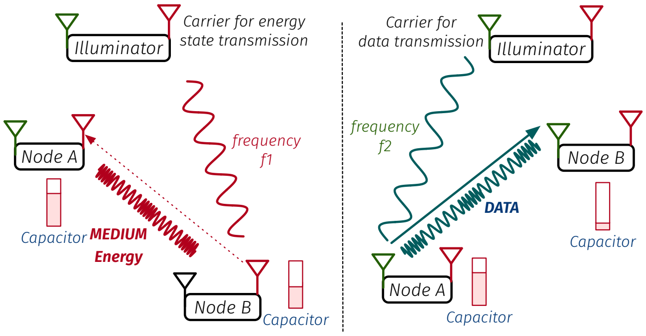


Figure 3: TRAP network architecture. Transiently-powered devices are equipped with two backscatter radios operating at different carrier frequencies. The energy status is backscattered using the backscatter circuitry described in Section 3. The second backscatter circuitry operates at a different frequency and used only for data transmission.

4 INTERMITTENT COMMUNICATION FOR TRANSIENTLY-POWERED DEVICES

In this section, we present a protocol design that enables, for the first time, intermittent communication among transiently-powered devices. Our protocol, named TRAP (TRAnsiently-powered Protocol), relies on the energy status information provided by the energy status channel. Based on this information, TRAP ensures that transmitter and receiver devices have sufficient energy to pursue communication and prevent packet losses due to power failures. Before delving into the details, we would like to present the necessary hardware requirements for the TRAP protocol.

4.1 Network Architecture and Hardware Requirements

TRAP requires an illuminator that emits *continuously* the necessary carrier waves for the backscatter devices. In some scenarios, there might not be a need for a dedicated illuminator. The ambient signals [13] (e.g., TV signals or Wi-Fi signals) can also be exploited. It is worth mentioning that the objective of the illuminator is not charging the devices. However, the devices can use an RF energy harvester to receive also energy from the illuminator. We assume that the devices are assumed to operate intermittently, which means their operation is composed of frequent charge/discharge cycles and power failures.

Each transiently-powered node in our system is equipped with two radios and antennas—see Figure 3:

1. The first transceiver is the energy status backscatter circuitry described in Section 3. This transceiver is used to transmit and receive energy status information.
2. The main radio can be an active radio or a backscatter radio that will be used for only data transmission. Within this study, we consider a backscatter radio that operates at a different frequency than the energy status backscatter circuitry.

The separation of the energy status channel from the data transmission channel prevents the packet collisions and enables simultaneous data and energy status transmission among the devices. The receiver hardware presented in [14] can be specifically tuned to different frequency with a very narrow band-pass response.

TRAP requires periodic timers for each node. Since the values of the timer registers in microcontrollers are lost upon each power failure, transiently-powered nodes lose their notion of time. Therefore, nodes are also equipped with zero-power persistent timers [6, 10] to keep track of time when the device is off. These timers use a capacitor as an hourglass: when the device is on, the capacitor is charged to a specific voltage; when the power is lost, the capacitor slowly discharges. The elapsed time during the power failure is estimated by measuring the voltage decay across the capacitor.

4.2 TRAP Protocol Definition

TRAP relies on the communication over two channels: (i) each transiently-powered device transmits its energy status over the energy status channel using the backscatter radio described in Section 3. Considering the energy status information of the neighbors, devices start data transmission over another channel using their main radio.

4.2.1 Energy Status Transmission. In TRAP, each node transmits its energy availability at a *fixed period* that is the same for all the nodes. The periodical data transmission has been employed by many protocols in wireless sensor networks, as in [4, 15, 25]. However, to enable a periodic transmission on a transiently-powered device, the persistent timing circuitry needs to be used. To minimize the packet collisions, nodes add a random offset to the fixed period, which is also employed in [4, 15]. Due to the random offset value, the energy status transmission of the neighboring nodes will drift apart quickly in case of a collision. Specifically, the following steps are taken by each node:

1. Add a random value to the fixed period to generate the actual transmission period.
2. Set a timer operating at this period using the microcontroller's volatile timers.
3. Upon the timer fires, use the energy status backscatter channel to encode the energy status, which as a burst. Set the timer again. The frequency and the length of ON-OFF keying characterize the burst. The length of the burst indicates the energy status information, and the frequency of the burst identifies the node. A short burst indicates a low energy status, a longer burst indicates a higher energy level.
4. Upon recovery from a power failure, calculate the off-time using the persistent timer circuitry. Subtract it from the period of transmission and set the microcontroller's timer using this value.

4.2.2 Data Transmission. In TRAP, data transmission is performed by considering the energy status information received from the neighboring nodes. If a receiver node has the necessary energy to perform computation and data transmission, it first decodes the received energy status information. Based on the energy status of the neighbor, the node immediately starts data transmission. Therefore, the energy status signal received from the backscatter circuitry serves as a synchronization point for the transmitter device to start transmission over the main backscatter radio. Specifically,

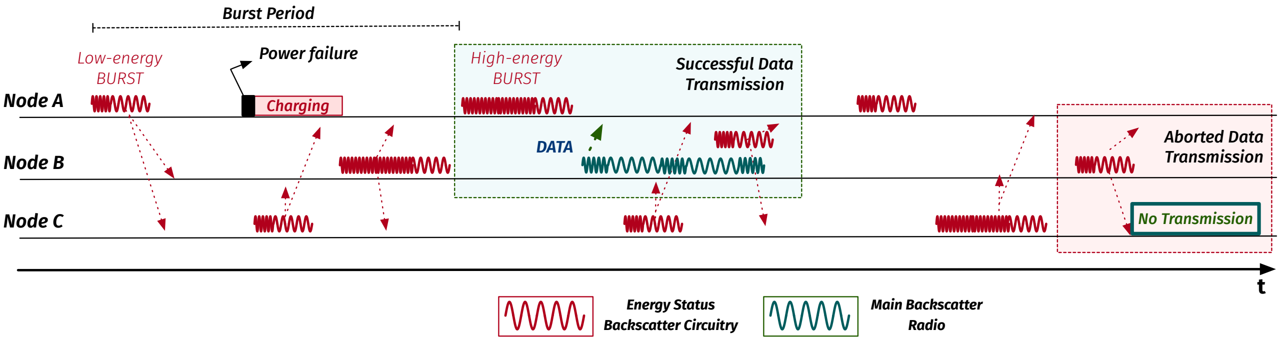


Figure 4: A communication scenario that shows how the energy status information, transmitted from a different channel, can be exploited to communicate successfully or to abort communication to save precious harvested energy.

a transmitter node takes the following steps to communicate with its neighbors:

1. First, the node checks if it has sufficient energy to perform computation and transmission. Note that this check is already performed periodically to transmit its energy status over the energy status channel.
2. If the transmitter node has sufficient energy to perform data transmission, it starts listening to the energy status channel via the backscatter circuitry described in Section 3.
3. When the node starts receiving individual bits from `Vout` depicted in Figure 2, it decodes the received packet by counting the pulses within the energy status burst as well as measuring the duration of the burst in order to determine the energy status information and the frequency that identifies the neighboring node.
4. The transmitter node initiates the backscatter communication over its main radio if it has data to be delivered to the neighboring node and the neighboring node has enough energy to receive the packet.

4.2.3 Carrier Sense. In TRAP, two nodes can simultaneously decide to transmit a packet upon receiving a high-energy burst from a common neighbor. This situation might lead to a collision on the receiver side, and can be prevented by employing a random back-off and carrier sensing approach. Before deciding on the transmission, nodes might generate a random back-off and check for any bit reception over their main backscatter radio. At the end of the back-off period, if no bit is received, the node can initiate the transmission.

4.3 A Communication Scenario

Figure 4 presents a communication scenario among the nodes that employ the TRAP protocol. In this scenario, initially the nodes A and C have low energy, and node B has high energy. As can be noticed from this figure, the nodes transmit their energy status periodically via their energy status backscatter circuitry. Even though nodes might die due to power failures (as Node A dies in the figure), they charge and they can preserve the periodicity of the energy status transmission using their persistent timers upon recovery from power failures.

Successful Data Transmission. Initially, Node B has enough energy for data transmission. Therefore, it waits for a high-energy burst from its neighbors. In the meantime, Node B transmits its energy status with a high-energy burst via its energy status backscatter circuit. Node A initially has not enough energy and transmits a "low-energy" burst. After a power failure, Node A harvests sufficient energy, and its energy status changes to a high energy level. Then, it starts transmitting a high-energy burst. After Node B detects this high-energy burst, it immediately replies by sending data via the main backscatter radio. With this communication scheme, the energy availability of both sides is ensured, and there will be no packet losses due to power failures.

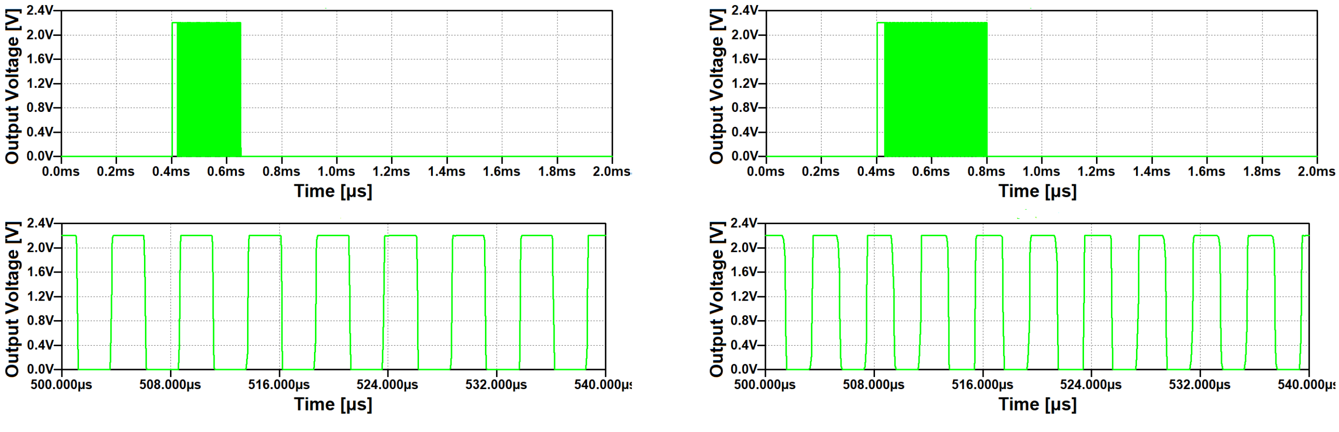
Aborted Communication. Later in this scenario, the energy availability of Node C increases, and it starts to transmit a high-energy burst. This node has data to be delivered to Node B. Therefore, it waits to receive an energy status information from Node B. Unfortunately, Node C receives the low-energy burst from Node B and aborts the transmission of data to save energy. If Node C transmitted the packet, Node B would die and not be able to receive the transmitted packet. Node C can use the saved energy for some useful computation operations.

In summary, when there is not any notion of coordination, there would be a lot of failed transmissions, and in turn, energy waste. In the presented scenario, TRAP prevents failed packet transmissions, and in turn, the waste of energy.

5 RESULTS

As we propose, the nodes use the backscatter channel encoding the energy status information with specific duration bursts. As indicated previously, the ON-OFF keying commutation frequency of the burst identifies a specific node. Moreover, the duration of the pulse represents the energy status information (i.e., in the number of pulses). A short burst with less than 20 pulses identifies a low energy status. On the other hand, a long burst with up to 100 pulses indicates a higher energy level. Therefore, a long pulse indicates that there is available energy on the receiver side to communicate using the main radio.

To validate our proposal, we carried out some simulations, using LTSpice, on the electronic circuit depicted in Figure 2. We simulated the receiving circuit with two different backscatter signals as input,



(a) (b)

Figure 5: Simulation results on the circuit presented in figure (circuit). Figure 5a reports the comparator digital output voltage with an applied input signal of a 200kHz ON-OFF keying burst with a duration of $250\mu s$. Figure 5b reports the comparator digital output voltage with an applied input signal of a 250kHz ON-OFF keying burst with a duration of $400\mu s$. Figures in the second row present a more detailed plot of the corresponding pulses presented in the first row. The frequency differences can be noticed from these figures, where the number of pulses in the same interval window is different. Similarly, the MCU can differentiate the received signal from different nodes.

showing the comparator output voltage (`Vout` in Figure 2). The first applied input signal is a $200kHz$ burst with a duration of $250\mu s$ (i.e., 50 pulses reported in Figure 5a). The second signal is a $250kHz$ burst with a duration of $400\mu s$ (i.e., 100 pulses reported in Figure 5b). The MCU can easily calculate the frequency of the burst by counting the bit transitions, and the duration of the burst using its internal volatile timers. Taking a closer look to the Figure 5a) and Figure 5b), a different number of pulses can be counted by looking at a fixed time window of $40\mu s$, thus identifying different frequencies. The simulations confirm that the proposed circuit can be used to decode the information regarding the frequency and burst duration from the backscattered energy status signals.

As discussed, we propose different ON-OFF keying frequencies for the burst to differentiate multiple nodes. Such frequencies must be in the operating band of the receiver circuit, which is limited roughly between 10kHz and 500kHz. Moreover, the capability of the MCU to record the burst duration and number of pulses is typically is high enough to ensure a wide frequency band to choose. Finally, the burst repetition period can be as long as the energy information update rate of the node. Thus, the system can scale up pretty easily.

During a continuous bitstream transmission, the power consumption of the backscatter transmitter in [14] is reported as 0.7mW. Based on this value, for a burst duration of $400\mu s$ (i.e., 100 pulses) and the burst repetition period of 1s, the overall power consumption of the backscatter modulation in TRAP is expected to be 0.28μW. We need to perform measurements on a real hardware implementation to obtain the actual power consumption, which we leave as future work.

In summary, our results show that the proposed intermittent communication protocol can overcome packet losses and energy waste by enabling highly-stable communication between nodes in real-world scenarios.

6 CONCLUSION AND FUTURE WORK

We presented TRAP, the first "intermitted communication" protocol for transiently-powered batteryless devices. TRAP uses a backscatter radio circuit to propagate each node's energy state's information. The energy status information is encoded by the use of specific frequency and duration burst, not only identifying the energy level, but also the transmitting node. Therefore, the protocol can regulate the packet transmission in transient computing devices and guarantee zero energy-wasting, because transmitters can always be updated about the receiving nodes' availability. Even in the scenario of tag-to-illuminator transmission TRAP can be used avoiding packet collisions. Simulation results demonstrate that addressing can be implemented, and the approach can be scalable for a network of sensors. TRAP is the first attempt to introduce fundamental hardware support and the building block of the future transiently-powered networks. We leave the real hardware implementation and test-bed deployment of TRAP as future work.

REFERENCES

[1] Panos N Alevizos, Konstantinos Tountas, and Aggelos Bletsas. 2018. Multistatic scatter radio sensor networks for extended coverage. *IEEE Transactions on Wireless Communications* 17, 7 (2018), 4522–4535.

[2] Domenico Balsamo, Anup Das, Alex S. Weddell, Davide Brunelli, Bashir M. Al-Hashimi, Geoff V. Merrett, and Luca Benini. 2016. Graceful Performance Modulation for Power-Neutral Transient Computing Systems. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 35, 5 (2016), 738–749. https://doi.org/10.1109/TCAD.2016.2527713

[3] Domenico Balsamo, Alex S Weddell, Anup Das, Alberto Rodriguez Arreola, Davide Brunelli, Bashir M Al-Hashimi, Geoff V Merrett, and Luca Benini. 2016. Hibernus++: a self-calibrating and adaptive system for transiently-powered embedded devices. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 35, 12 (2016), 1968–1980.

[4] Nicolas Burri, Pascal Von Rickenbach, and Rogert Wattenhofer. 2007. Dozer: ultra-low power data gathering in sensor networks. In *Proceedings of the 6th international conference on Information processing in sensor networks*. 450–459.

[5] Alexei Colin and Brandon Lucia. 2016. Chain: tasks and channels for reliable intermittent programs. In *Proceedings of the 2016 ACM SIGPLAN International Conference on Object-Oriented Programming, Systems, Languages, and Applications*. 514–530.

[6] Jasper de Winkel, Carlo Delle Donne, Kasim Sinan Yildirim, Przemysław Pawełczak, and Josiah Hester. 2020. Reliable Timekeeping for Intermittent Computing. In *Proceedings of the Twenty-Fifth International Conference on Architectural Support for Programming Languages and Operating Systems*. 53–67.

[7] Bradley Denby and Brandon Lucia. 2020. Orbital Edge Computing: Nanosatellite Constellations as a New Class of Computer System. In *Proceedings of the Twenty-Fifth International Conference on Architectural Support for Programming Languages and Operating Systems*. 939–954.

[8] Josiah Hester and Jacob Sorber. 2017. Flicker: Rapid prototyping for the batteryless internet-of-things. In *Proceedings of the 15th ACM Conference on Embedded Network Sensor Systems*. 1–13.

[9] Josiah Hester and Jacob Sorber. 2017. The future of sensing is batteryless, intermittent, and awesome. In *Proceedings of the 15th ACM Conference on Embedded Network Sensor Systems*. 1–6.

[10] Josiah Hester, Nicole Tobias, Amir Rahmati, Lanny Sitanayah, Daniel Holcomb, Kevin Fu, Wayne P Burleson, and Jacob Sorber. 2016. Persistent clocks for batteryless sensing devices. *ACM Transactions on Embedded Computing Systems (TECS)* 15, 4 (2016), 77.

[11] Ivar in't Veen, Qingzhi Liu, Przemyslaw Pawelczak, Aaron Parks, and Joshua R Smith. 2016. BLISP: Enhancing backscatter radio with active radio for computational RFIDs. In *2016 IEEE International Conference on RFID (RFID)*. IEEE, 1–4.

[12] Vito Kortbeek, Kasim Sinan Yildirim, Abu Bakar, Jacob Sorber, Josiah Hester, and Przemysław Pawełczak. 2020. Time-sensitive Intermittent Computing Meets Legacy Software. In *Proceedings of the Twenty-Fifth International Conference on Architectural Support for Programming Languages and Operating Systems*. 85–99.

[13] Vincent Liu, Aaron Parks, Vamsi Talla, Shyamnath Gollakota, David Wetherall, and Joshua R Smith. 2013. Ambient backscatter: Wireless communication out of thin air. *ACM SIGCOMM Computer Communication Review* 43, 4 (2013), 39–50.

[14] Amjad Yousef Majid, Michel Jansen, Guillermo Ortas Delgado, Kasim Sinan Ytidtnm, and Przemystaw Pawetczak. 2019. Multi-hop backscatter tag-to-tag networks. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*. IEEE, 721–729.

[15] Miklós Maróti, Branislav Kusy, Gyula Simon, and Ákos Lédeczi. 2004. The flooding time synchronization protocol. In *Proceedings of the 2nd international conference on Embedded networked sensor systems*. 39–49.

[16] Matteo Nardello, Harsh Desai, Davide Brunelli, and Brandon Lucia. 2019. Camaroptera: A batteryless long-range remote visual sensing system. In *Proceedings of the 7th International Workshop on Energy Harvesting & Energy-Neutral Sensing Systems*. 8–14.

[17] Benjamin Ransford, Jacob Sorber, and Kevin Fu. 2011. Mementos: System support for long-running computation on RFID-scale devices. In *Proceedings of the sixteenth international conference on Architectural support for programming languages and operating systems*. 159–170.

[18] Alberto Rodriguez Arreola, Domenico Balsamo, Anup K. Das, Alex S. Weddell, Davide Brunelli, Bashir M. Al-Hashimi, and Geoff V. Merrett. 2015. Approaches to Transient Computing for Energy Harvesting Systems: A Quantitative Evaluation. In *Proceedings of the 3rd International Workshop on Energy Harvesting & Energy Neutral Sensing Systems* (Seoul, South Korea) *(ENSsys '15)*. Association for Computing Machinery, New York, NY, USA, 3–8. https://doi.org/10.1145/2820645.2820652

[19] Mohammad Rostami, Jeremy Gummeson, Ali Kiaghadi, and Deepak Ganesan. 2018. Polymorphic radios: A new design paradigm for ultra-low power communication. In *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication*. 446–460.

[20] Jihoon Ryoo, Jinghui Jian, Akshay Athalye, Samir R Das, and Milutin Stanaćević. 2018. Design and Evaluation of "BTTN": A Backscattering Tag-to-Tag Network. *IEEE Internet of Things Journal* 5, 4 (2018), 2844–2855.

[21] Jihoon Ryoo, Yasha Karimi, Akshay Athalye, Milutin Stanaćević, Samir R Das, and Petar Djurić. 2018. Barnet: Towards activity recognition using passive backscattering tag-to-tag network. In *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services*. 414–427.

[22] Davide Sartori and Davide Brunelli. 2016. A smart sensor for precision agriculture powered by microbial fuel cells. In *2016 IEEE Sensors Applications Symposium (SAS)*. 1–6. https://doi.org/10.1109/SAS.2016.7479815

[23] Joshua R Smith. 2013. *Wirelessly powered sensor networks and computational RFID*. Springer Science & Business Media.

[24] Vamsi Talla, Mehrdad Hessar, Bryce Kellogg, Ali Najafi, Joshua R Smith, and Shyamnath Gollakota. 2017. Lora backscatter: Enabling the vision of ubiquitous connectivity. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3 (2017), 1–24.

[25] Kasım Sinan Yıldırım, Ruggero Carli, and Luca Schenato. 2017. Adaptive proportional–integral clock synchronization in wireless sensor networks. *IEEE Transactions on Control Systems Technology* 26, 2 (2017), 610–623.

[26] Kasım Sinan Yıldırım, Amjad Yousef Majid, Dimitris Patoukas, Koen Schaper, Przemyslaw Pawelczak, and Josiah Hester. 2018. Ink: Reactive kernel for tiny batteryless sensors. In *Proceedings of the 16th ACM Conference on Embedded Networked Sensor Systems*. 41–53.

[27] Pengyu Zhang, Mohammad Rostami, Pan Hu, and Deepak Ganesan. 2016. Enabling practical backscatter communication for on-body sensors. In *Proceedings of the 2016 ACM SIGCOMM Conference*. 370–383.