Battery-Free Smart Objects Based on RFID Backscattering

Gaia Maselli, Matteo Pietrogiacomi, Mauro Piva, and John A. Stankovic

Abstract

The Internet of Things era has witnessed an explosion of smart objects. As we move toward connecting the next billion wireless devices to the Internet, however, the use of batteries to power them will become unworkable, with significant repercussions on health and the environment if improperly disposed. Hence, the need for more eco-friendly technologies.

This article shows how radio-frequency identification technology enables the re-design of personal wireless computing devices in a battery-less manner, representing a major leap forward in moving beyond chargers, cords, and dying devices. Specifically, we study the development of various battery-free devices, and identify the types of devices that can be handled today and what is future work. We describe testbed experiments that clearly demonstrate the feasibility of the devices we built, presenting performance comparable to commercial battery-powered counterparts.

Introduction

The Internet of Things (IoT) era has witnessed an explosion of wireless devices — home, office, and personal devices — that have created an ever-increasing demand for batteries. Consumers dispose each year an embarrassing number of batteries (on the order of billions), which is highly dangerous for the environment if not properly disposed. Rechargeable batteries limit the problem only partially as they become unusable after some time of daily recharging. The need for more eco-friendly wireless devices is evident.

The question is: Is it possible to re-design smart objects so that they can work without batteries? The answer is backscattering. A breakthrough in wireless communications, backscattering allows powering of sensor devices and eliminates the need to have any inbuilt batteries at all. Several low-power devices can use radio frequency (RF) signals as a power source and use them to sense, compute, and transmit data via reflecting the RF signal. Two backscattering techniques — Ambient [1] and RF identification (RFID) [2] — are available; they enable data computation and transmission on battery-free devices.

With ambient backscattering, devices harvest power from signals available in the environment (e.g., TV, cellular, and Wi-Fi transmissions). The main advantage of ambient backscattering is the use of existing RF signals without requiring any additional emitting device. However, it presents several performance drawbacks. Current techniques for ambient backscattering achieve low data rate (below 1 kb/s). Thus, it can be employed in applications that need to transmit data only occasionally, for example, to exchange money between smart cards or detect misplaced objects in a grocery store, but cannot support real-time applications, which need continuous communication. The availability of signals is another limitation: Although TV towers broadcast signals 24 hours a day without interruption, signal ubiquity cannot be guaranteed, with negative effects on the transmission of data in real time. If the signal is weak, smart devices are not able to accumulate the energy necessary to operate. Moreover, signals weaken significantly in indoor environments, even in places where they are supposed to be ubiquitous (e.g., TV signals in metropolitan areas located at a distance greater than a few, 8 to 10, kilometers from the tower).

RFID backscattering powers tags up by harvesting power from signals emitted by an interrogator (i.e., RFID reader), and makes them communicate by backscattering the incident signal [2]. The traditional RFID technology involves a set of tags — devices without any power source — that absorb and reflect the high-power constant signal generated by a powered device, namely the reader, which interrogates them to get their unique ID. With the advent of IoT, new RFID-based devices have been developed, namely sensor-augmented RFID tags, which harvest energy from the reader and exploit it to run some low-power sensors and transmit sensed data. This new technology enables the development of battery-free smart devices like cell phones [3], cameras (WISPCam [4]), remotes [5], and video game controllers (JoyTag [6]).

In this article we investigate the design of battery-free smart objects (or smart devices) based on RFID technology that can be deployed in a smart home and make the following contributions:

- We have built a representative set of battery-free RFID-based device types (we use the UMich Moo platform[1]) to illustrate the solution including devices that are real-time (e.g., a video game controller, a microphone), periodic (e.g., a temperature sensor) and event-based (detecting presence, a fire detector). Specifically, we present the development of a video game controller, called SapyJoy, which is able to interact with several types of video games.
- We have identified the types of devices that can be handled today and what is future work, and have done extensive controlled experiments to evaluate the performance of different types of devices showing that multiple smart objects can be made battery-free and the challenges for their coexistence in the same smart environment.
- We have shown through experiments that our newly developed devices are very fast in communicating with the corresponding applications, performing even better than commercial benchmarks.

Battery-Free Smart Devices

Many important types of sensors — temperature, humidity, light, accelerometers, pressure buttons, analog joysticks, and so on — can be integrated with Moo tags to devise battery-free devices. The main constraint is related to tag energy consumption. Sensors should not require more than 3 V, and each sensor's consumption should be less than 10 mW in order to allow continuous sensor activity. With more demanding sensors, up to 100 mW, the sensor has to exploit a sensor duty cycle in order to satisfy the energy constraints. As shown in [7], with deep duty cycle tuning, it is possible to power devices requiring up to 200 uA at 1.8 V with 10 Hz refresh rate. This implies that a Moo-based solution is effective for many smart devices.

In the following, we present the set of battery-free devices that we built by leveraging University of Michigan (UMich) Moo Computational RFID tags.

Digital Object Identifier: 10.1109/IOTM.0001.1900048

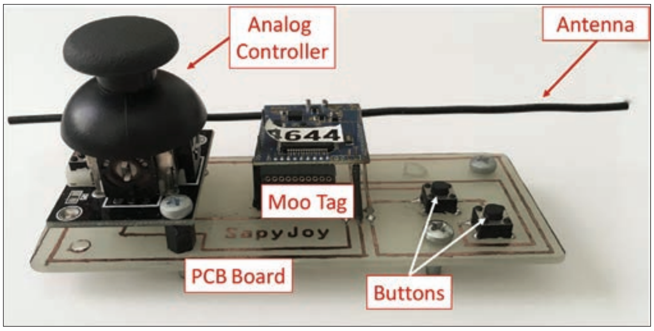

FIGURE 1. SapyJoy videogame controller: A PCB board connects the analog controller and the buttons to the Moo tag.

Specifically, we illustrate the breadth of solutions by building devices that are *periodic*, *event-based*, and *real-time* (performing burst sensing). Then, to further show applicability, we present other devices that could easily be developed, and finally discuss several devices that are already present in the literature (developed by others). Overall, this section demonstrates the wide variety of devices (with different rate requirements) that can be accommodated by our solution.

Newly Developed Devices

We built a representative set of Moo-based battery-free devices discussed in this subsection, including periodic, event-based, and real-time devices.

Light Switch: This is an event-based device, realized by mounting a button on the Moo Tag. When the user presses the button on the wireless and battery-less light switch, the system switches on an LED on an actuator. Depending on the application, it is possible to embed multiple buttons on the same Moo tag to control different lights deployed inside a smart building. The logical connection between the tag switch and the corresponding light is placed inside the server.

Remote for a Tea Kettle: This is an event-based remote able to switch on a kettle. It is realized by means of two Moo tags. The first one is equipped with a button and acts as a remote to activate the kettle. The second one acts as an actuator, and is connected to the kettle through a relay that is activated by a reader message.

Video Game Controller: This is a real-time device that is realized by mounting an analog joystick and two buttons on a Moo tag. The resulting wireless and battery-less video game controller is able to interact with several types of video games (e.g., adventure, action, puzzle, and role-playing games). Figure 1 shows our video game controller (called SapyJoy): a printed circuit board (PCB) connects the analog joystick and the two buttons to the Moo tag, which also has an accelerometer embedded, allowing for complex game experiences. Another version of the controller featuring only an accelerometer (no buttons and no analog joystick) was presented in [6].

Mouse: A platform analogous to SapyJoy can work as a wireless and battery-less mouse by interfacing its x and y axes with the pointer on the screen. We embedded the information regarding the analog controller inside packets transmitted by the tag and realized a virtual mouse driver able to decode this information and translate it into the pointer position.

Devices that Can Be Built

By studying the technical characteristics of different sensors and actuators, we identified the set of devices that can be developed easily. The following is a description of some of the devices that can be built by leveraging Moo tags. This increases the applicability of our solution for battery-less smart homes.

Event Detector: Embedding a smoke sensor on the Moo tag, it is possible to devise a fire alarm.[2] Another detectable event is detecting presence through a motion sensor.[3] In general, any ultra-low-power sensor able to detect an event can be exploited to build an event detector.

Remote for Appliances: Any appliance that can be actuated by a relay (coffee machines, shutters, doors, air fans, etc.) can be controlled by a battery-free remote through mounting a button on a Moo tag — the remote — and connecting the appliance to another Moo tag — the actuator — through a relay.

Infrared Remote Commander: Embedding an ultra-low-power infrared data association (IRDA) emitter[4] on the Moo tag we can create an IR remote controller for any appliance equipped with an IR interface, prolonging the lifetime of less recent and technological appliances. In this case the Moo Tag must be placed in front of the IR receiver on the controlled device. Even if the IRDA emitter consumption is quite high (170 uA for transmission), we expect to have some seconds between a command and the next one, enough to recharge the accumulator.

Environmental Sensors: Light, humidity, presence, and other sensors can be mounted on the Moo tag to allow environmental monitoring. The number of sensors that can be mounted on a Moo tag depends on the number of I/O ports that the microprocessor owns and the amount of energy available.

Already Developed Devices

There are a few battery-free devices that have already been developed by others.

Temperature Sensor: It is a device that periodically senses temperature and reports sampled data to the server to allow environmental monitoring.

Camera: As shown in [4], it is possible to implement an RFID tag with an embedded camera able to take pictures and transfer them with the power harvested by the RFID antenna.

Cordless Phone: As demonstrated in [3] it is possible to realize a simple phone, able to stream voice and audio from and to the reader. This device can be used as a phone, as a microphone, or a small sound diffusion system.

Information Display: By integrating an ultra-low-power electronic ink (e-ink) display on the moo tag, it is possible to realize a display for several types of information. A first solution is given in [9], where a number of wearable displays (shoes, t-shirt, etc.) have been realized using electromagnetic induction and e-ink displays. We believe that other devices can be deployed. For example, displaying the current time, it is possible to realize a battery free clock. The display can also be useful to show messages from authorized people outside the building. For example, in assisted living applications, remote relatives or caregivers can remind a person to take medication or perform some actions inside the home.

Monitoring Systems: The work in [8] shows how to create a series of sensors able to detect doors opening and monitor water usage of a drinking tap. Modifying the antenna circuit of tags, it is possible to open and close it in order to activate or deactivate the tag. The reader, depending on the tag status, can understand if a door is open or closed. For example, when the door is open, the tag is not activated, and when it closes, the tag is activated.

Experiments with Battery-Free Smart Devices

We now evaluate the performance of our battery-less smart devices, benchmarking their performance against those of commercial battery-powered devices.

Testbed

We implemented prototypes for two video game controllers, a mouse, a light switch, and a temperature sensor using the UMich Moo Computational RFID tag. To interact with our prototypes, we use a USRP RFID reader equipped with two RFID antennas, and a server that interconnects the RFID reader with smart home applications. The Moo tag receives the reader signal and uses it to harvest operating power using the RFID circuit. The harvested power runs onboard sensing, encoding

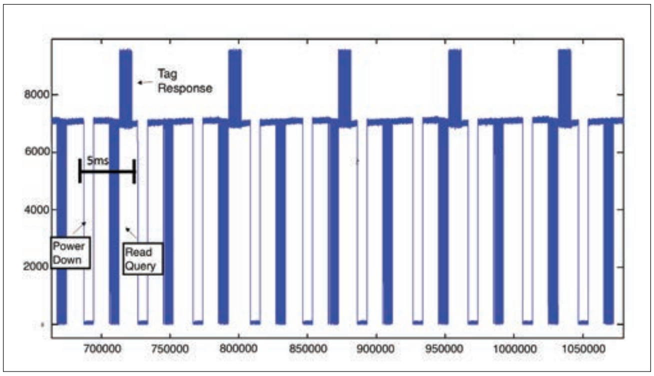

FIGURE 2. Matched filter for SapyJoy.

Device	Reaction time (ms)	CI
SapyJoy	92.92	[82.31–102.92]
Commercial controller	104.58	[96.31–112.85]
Commercial mouse	110.41	[103.09–117.74]

TABLE 1. Reaction time for different controllers.

of measurement data, cyclic redundancy check (CRC) error coding, and backscatter communication to wirelessly send data back to the reader. The communication protocol between the reader and the tags is based on the EPC Gen 2 Class 1 standard [12], which has been modified to acquire data from sensors and store them in the buffer that is traditionally used to maintain the tag ID. As only a few bits (e.g., 8 bits) are sufficient to represent the tag's ID, the remaining, typically 96 – 8 bits, can be used to send sensed data. We limited the data field to 1 byte for tag ID and 6 bytes for data samples (including 4 CRC bits). This number guarantees low packet error rate — confirmed by our experimental study — and enough space for data samples for all devices except the camera, which would require data fragmentation even in the case of longer payloads.

Metrics

We evaluated the performance of our prototypes by measuring the following metrics:

- **Reaction time** is the time since the generation of new sensor data to the corresponding action on the recipient application. This is an application layer metric. In the case of the joystick, it measures the time between an action on the joystick (e.g., a button press) and the corresponding event on the video game application. In the case of an environmental sensor, this metric measures the time between the generation of new sensor data and the corresponding reaction on the recipient actuator (e.g., a presence sensor activating a camera).
- **Packet delay** is the time from the generation of new sensor data to its reception by the reader.
- **Throughput** is the number of bits that the reader receives per unit of time.
- **Packet error rate** is the fraction of incorrect packets received by the reader over the total number of sent packets.

While it is possible to measure the last three metrics (i.e., packet delay, throughput, and packet error rate) at the reader side, reaction time requires a more complex procedure because of synchronization issues between sensors and actuators (e.g., the player's action and the corresponding game reaction). Besides the packet delay at the network layer, reaction time also includes the time it takes for the packet to proceed up the protocol stack at the recipient. For these reasons we use a digital camera to measure reaction time. The camera frames the sensor and the actuator at the same time so that we have a unique clock to record events (e.g., in the joystick case, the camera frames the controller and the screen to record button presses and corresponding actions on the screen). In this way we can also measure time for commercial devices, which is impossible at the software level.

Results on Single Devices

We now evaluate the feasibility of the devices we built.

Video Game Controller: The first battery-free device we evaluate is our video game controller, SapyJoy, which is compared to two commercial Bluetooth devices: a Logitech controller per console (cordless precision controller for Playstation3) and a Logitech wireless mouse (cordless optical mouse for notebooks). The three controllers were used to play with navigating video games, which have an update rate of 30 fps, as well as shooting video games, which have an update rate of 60 fps. The three controllers were all good, and we did not notice any difference in playing ability. To quantify this ability, and considering the difficulty in identifying a reaction to a user's action in a video game, we implemented a simple application that represents the joystick through arrows and buttons through circles. When the player moves the joystick, the corresponding arrow changes color on the screen (e.g., if the player moves the joystick ahead, the top arrow changes color). Analogously, when the player presses a button (i.e., the right one), the corresponding circle (i.e., the circle on the right of the screen) changes color (Fig. 3).

Table 1 shows the observed reaction time (with 5 percent confidence interval) for the three devices measured through a video camera framing at the same time as the controller and the screen (the update rate of the viedo camera is 60 fps). SapyJoy takes on average 92.92 ms to see the outcome of a button pressure on the video game, while the two commercial devices — controller and mouse — take 104 ms and 110 ms, respectively, to perform the same operation. These results show that SapyJoy is even faster than battery-powered devices.

Reaction time includes the packet delay at the network layer, plus the time to deliver the packet from the reader to the server, plus the time to produce the game commands corresponding to the actions performed by the user and send them to the video game application. Thus, if we measure only the packet delay at the network layer, SapyJoy takes on average only 4.79 ms to deliver sensed data to the reader (note that we cannot measure this metric for the commercial devices because they are not programmable).

Analyzing the matched filter for our SapyJoy, we observed that although the packet delay is below 5 ms, to achieve the best performance —avoid any reader-tag collision due to any possible delay from the tag — the reader can issue a new query every 6 ms (Fig. 2). By querying tags at this interval of time, the throughput at the reader is 6.6 kb/s (including sensor data and protocol control bits), with less than 1 percent packet error rate.

Light Switch and Mouse: Now we evaluate our battery-free light switch and mouse. We again use a video camera framing at the same time as the sensor and actuator. In the case of the light switch, the sensor is the tag equipped with a pressure button, while the actuator is a tag with an LED onboard. In the case of the mouse, we use the same platform as for the joystick. The mouse communicates with an application showing cursor movements and button pressures through a circle that moves on the screen and changes color when a button is pressed (Fig. 3).

Table 2 shows the reaction for the two devices. The light switch takes only 62.91 ms to collect data from the pressed button, send it to the actuator, and switch on the LED. Although we do not have benchmarks to compare, we believe that this time would satisfy any stringent application requirements.

Reaction time increases to 92.92 ms in the case of the mouse, because as for the video game, data has to reach the final application on the server, taking some time to ascend the protocol stack. However, even in this case the system is very reactive, with the user perceiving real-time communication.

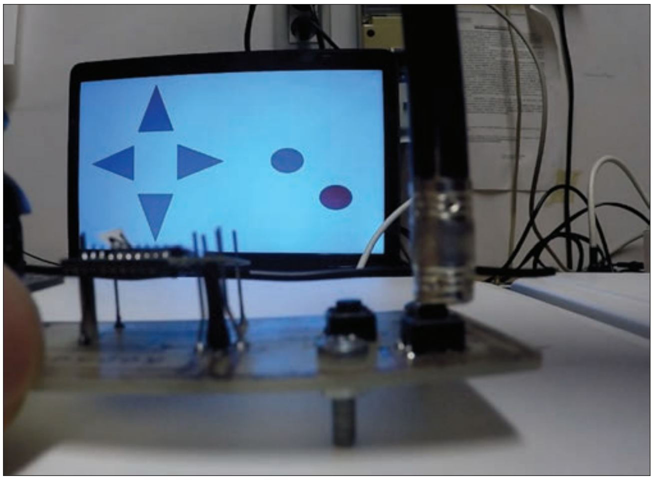

FIGURE 3. Button pressure on the battery-free mouse and corresponding action on the screen.

Device	Reaction time (ms)	CI
Light switch	62.91	[67.41–73.41]
Mouse	92.92	[82.91–102.92]

TABLE 2. Reaction time for battery-free light switch and mouse.

Results with Multiple Devices

We now present experimental results that evaluate multiple battery-free devices simultaneously so that we can evaluate their interoperability. We run experiments with three devices working at the same time: two environmental sensors — temperature and presence — and a video game controller (our SapyJoy). The devices are queried (and hence transmit sensed data) following a time-division multiple access approach, which provides different time slots to different devices in a cyclically repetitive frame structure. The first difference with respect to experimentation with a single device is reaction time. If a device is queried at each slot, the reaction time is clearly shorter with respect to the case in which it is queried once every multiple slots. The outcome of our experimentation is that the reaction time increases significantly (i.e., 200 ms) with respect to when it works alone (i.e., 92.92 ms). This delay would certainly increase if the number of transmitting devices increases, making interoperability a challenge as the joystick may experience delays that are too long.

Lesson Learned

Our experimentation highlights two big challenges for the design and deployment of battery free environments, like smart homes, in which there are many sensors and smart devices, such as surveillance cameras, smoke, presence, temperature, light sensors, smart meters, and many others.

The first challenge concerns interoperability of devices. Although results clearly show the feasibility of battery-free RFID-based smart objects, whose performance is comparable to that of the battery-powered counterparts, their coexistence cannot be taken for granted. When multiple devices operate simultaneously, the reaction time increases significantly with respect to the case of a device working alone. In addition, an equal assignment of channel resources would not satisfy devices' needs. Multi-kind multiple battery-free devices, operating simultaneously, have widely varying communication requirements in terms of data transmission, ON/OFF activity, and deadlines. To pick an example, a joystick may sense no changes for hours (while it is OFF), and then start sensing new data (while used for playing) at very different rates (from a few milliseconds to one or more seconds), depending on the game type and player activity. Thus, a communication protocol for battery-free devices should schedule channel access such that devices requirements are satisfied and data is delivered in time. A first solution in this direction is given in [13].

The second big challenge regards operational limits of RFID technology: communication range is a major obstacle for the real-world implementation of this low-cost technology. The transmission power of our reader is P_t = 0.5 W, and the communication range between the antennas and the tags is below 1 m. With this technology, it is possible to realize smart devices such as the joystick of the light switch, but not a video camera, which requires real-time streaming. Increasing the power of the reader (e.g., up to P_t = 1 W) would allow a longer transmission range (up to 3 m) between the reader's antennas and the tags, but it would not satisfy real-time frequencies. The need for technological improvement is clear. A first attempt toward more efficient devices in terms of bit rate, distance, and energy is given in [14], where the RFID device is powered not only by RF harvesting but also by a small solar panel (3 cm × 3 cm), reaching a transmission range of 21 ft and a maximum bit rate of 21.7 kb/s. This trend is confirmed in [15], where the use of photovoltaics increases the transmission range by providing additional power to the RFID tag integrated circuit.

Conclusions

The last decade has witnessed an explosion of wireless devices that have created an ever-increasing demand for batteries. In this article, we demonstrate that RFID technology is a key enabler for realizing many battery-less smart devices, performing real-time, periodic, and event-based sensing. Most of these devices are doable now — we realized light switches, remotes for a tea kettle, video game controllers, and a mouse, and studied how to realize event detectors, IF remote commanders, and remotes for general appliances — while others are more difficult to realize (e.g., video cameras). Results clearly show the feasibility of our approach, but also highlight the need for new communication protocols that can distinguish between fewer and more demanding devices.

We believe that our work is useful for practical use of RFID technology in the development of wireless and battery-less devices, and motivates further work with the goal of investigating new techniques to support more demanding devices, such as video cameras, and more powerful technology, which achieves longer transmission distance.

References

[1] S. Gollakota *et al.*, "The Emergence of RF-Powered Computing," *IEEE Computer*, vol. 47, no. 1, Jan. 2014.

[2] R. Want, "Enabling Ubiquitous Sensing with RFID," *IEEE Computer*, vol. 37, no. 4, 2004, pp. 84–86.

[3] V. Talla *et al.*, "Battery-Free Cellphone," *Proc. ACM Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1.2, 2017.

[4] S. Naderiparizi *et al.*, "Wispcam: A Battery-Free RFID Camera," *Proc. IEEE Int'l. Conf. RFID*, 2015.

[5] A. J. S. Boaventura and N. B. Carvalho, "A Batteryless RFID Remote Control System," *IEEE Trans. Microwave Theory and Techniques*, vol. 61, no. 7, July 2013.

[6] G. Maselli *et al.*, "JoyTag: A Battery-Less Videogame Controller Exploiting RFID Backscattering: Demo," *Proc. ACM MOBICOM*, 2016.

[7] A. P. Sample *et al.*, "Design of A Passively-Powered, Programmable Sensing Platform for UHF RFID Systems." *IEEE Int'l. Conf. RFID*, 2007.

[8] T. Zhang *et al.*, "BitID: Easily Add Battery-Free Wireless Sensors to Everyday Objects," *IEEE Int'l. Conf. Smart Computing*, 2017.

[9] C. Dierk, N. M. J. Pearce, and E. Paulos. "Alterwear: Battery-Free Wearable Displays for Opportunistic Interactions." *Proc. 2018 ACM CHI Conf. Human Factors in Computing Systems*, 2018.

[10] D. Miorandi *et al.*, "Internet of Things: Vision, Applications and Research Challenges," *Ad Hoc Networks*, 2012.

[11] M. Roberti, "RFID Needs to Be Part of the Building," *IEEE RFID J.*, 2018.

[12] EPCglobal Inc. "EPC™ Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860MHz–960MHz"; http://www.epcglobalinc.org/standards/uhfc1g2, 2006.

[13] G. Maselli, M. Piva, and J. A. Stankovic, "Adaptive Communication for Battery-Free Devices in Smart Homes," *IEEE Internet of Things J.*, vol. 6, no. 4, Aug. 2019.

[14] P. Zhang and D. Ganesan, "Enabling Bit-by-Bit Backscatter Communication in Severe Energy Harvesting Environments," *Proc. 11th USENIX NSDI*, 2014.
[15] S. N. R. Kantateddy *et al.*, "Long Range Battery-Less PV-Powered RFID Tag Sensors," *IEEE Internet of Things J.*, 2019.

Biographies

Gaia Maselli [M] (maselli@di.uniroma1.it) is an assistant professor at the Department of Computer Science at Sapienza University of Rome, Italy. She holds a Ph.D. in computer science from the University of Pisa, Italy. Her current research interests concern design and implementation aspects of mobile networks and wireless communications systems, with particular focus on backscattering networks for the Internet of Things. Other interests include design and performance evaluation of networking protocols for RFID systems. In the past, she has contributed to research on cross-layer design for ad hoc networks. She serves as member of the TPC of several international conferences, is associate editor of Elsevier Computer Communications journal, and serves as reviewer for several journals, such as IEEE TMC, IEEE TPDS, IEEE TWC, IEEE TON. She participated to many European Community research projects such as: CHIRON, eDIANA, SENDORA, SENSEI, E-sense, WiseNts, MobileMAN.

Matteo Pietrogiacomi received the bachelor degree in computer science from the Sapienza University of Rome, Italy, in october 2017. In his bachelor thesis he realised a battery free controller for video games.

Mauro Piva [M] (mauro.piva@uniroma1.it) received the laurea degree with the highest honors in computer science from the Sapienza University of Rome, Italy, in 2017. He is currently working toward the PhD degree at the Department of Computer Science at the same University. His current research interests include backscattering communication, reinforcement learning for networking, IoT networks, and Software Defined Networking.

John A. Stankovic (stankovic@cs.virginia.edu) is the BP America Professor in the CS Department at the University of Virginia and Director of the Link Lab. He is a Fellow of IEEE and ACM. He has been awarded an Honorary Doctorate from the University of York for his work on real-time systems. He won the IEEE Real-Time Systems Technical Committee's Award for Outstanding Technical Contributions and Leadership. He also received the IEEE Technical Committee on Distributed Processing Distinguished Achievement Award. He has seven Best Paper awards, and an h-index of 115 and over 57,000 citations. In 2015 he was awarded the Univ. of Virginia Distinguished Scientist Award, and in 2010 the School of Engineering's Distinguished Faculty Award. He also received a Distinguished Faculty Award from the University of Massachusetts. He has given more than 40 Keynote talks at conferences. He was the founder and co-editor-in-chief for the Real-Time Systems Journal.

Footnotes

[1] The name Moo comes from the fact that it is the beefiest embedded platform in its class with the most code space and RAM for the least energy. The device also resembles a longhorn steer. https://spqr.eecs.umich.edu/moo/
[2] https://goo.gl/jPLfva
[3] http://www.ti.com/lit/ug/tiducu5/tiducu5.pdf
[4] http://bit.ly/2HiipVj