6lo-RFID: a Framework for Full Integration of Smart UHF RFID Tags into the Internet of Things

I. Farris, S. Pizzi, M. Merenda, A. Molinaro, R. Carotenuto, A. Iera
DIIES, University "Mediterranea" of Reggio Calabria
Via Graziella, Loc. Feo di Vito, 89100, Reggio Calabria, Italy
Email: [name.surname]@unirc.it

Abstract

Recent technological advancements have transformed ultra-high-frequency (UHF) passive Radio-Frequency Identification (RFID) tags into "smart" transponders enhanced with computational and sensing capabilities, thus legitimated to become fully-fledged components of the Internet of Things (IoT). Nonetheless, the RFID technology has not been addressed yet by the Internet Engineering Task Force (IETF) *6lo* working group among the various link layer technologies for running IPv6 over networks of resource-constrained nodes in the IoT. Research works in the scientific literature only consider *proxy-based* solutions for the RFID inclusion in the IoT and do not suggest any ultimate approach to enable interoperability at the network and application layers. This paper fills the gap by designing and testing a unified IPv6-based framework to access and manipulate RFID tags' resources by means of the standard Constrained Application Protocol (CoAP). To this purpose, an adaptation layer, named *6lo-RFID*, is specifically designed to run IPv6 over the RFID link technology. Key design aspects are addressed, such as 6lo-RFID frame format and exchange, IPv6 header compression, fragmentation, network addressing and communication on the air interface. The designed solution is also compliant with EPCglobal Class-1 Generation-2 standard for RFID. In addition, we manufactured a smart UHF RFID tag platform to demonstrate the full-IPv6 protocol stack and assess its performance for bidirectional real-time communications over the RFID radio interface.

Index Terms

Internet of Things, RFID technology, IPv6, 6lo, CoAP, Wireless Sensor Networks

I. Introduction

Advancements in the design and manufacturing of ultra-high-frequency (UHF) Radio-Frequency Identification (RFID) systems have fostered the evolution of passive tags from very simple identification transponders, e.g., used in the supply chain, to general-purpose battery-less "computing" devices, enhanced with "sensing" and "data-logging" capabilities [1]. Hereinafter, we will refer to these augmented tags as "smart" tags.

Smart tags have the potential to substitute or complement existing wireless sensor networks (WSNs) for their tiny size, low-power, and very cost-effective identification mechanisms. They may bring undisputed benefits into various IoT application scenarios [2], where networked tags with sensing and computing capabilities create smart environments, such as in enhanced supply chain, factory automation, smart-home, smart-city, and personal healthcare scenarios [3].

The next step in the evolution of smart RFID tags consists in enabling them to be accessible from and to communicate with *any* other networked devices in the Internet, in the view of their real integration into the future Internet of Things (IoT).

The Internet Engineering Task Force (IETF) is driving the standardization of IPv6-based protocols for the IoT. The IPv6 over Networks of Resource-constrained Nodes (6lo) working group (WG) [7] is currently enabling IPv6 connectivity for a variety of link layer technologies in constrained node networks with limited power, memory and processing resources, such as Bluetooth Low Energy (BLE) [5] and Near Field Communication (NFC). The 6lo standardization effort has not yet addressed UHF RFID among the link layer technologies of interest, albeit the potential of modern smart tags to be included as fully-fledged components in the IoT.

In this paper, we design an IPv6-based internetworking solution, which natively enables smart RFID tags *(i)* to directly communicate with *any* other devices connected to the Internet and *(ii)* to expose their resources via a REST (Representational state transfer) application programming interface relying on the standard Constrained Application Protocol (CoAP). In details, we specify *6lo-RFID*, an *IPv6 adaptation layer* customized for running on top of the RFID link layer similarly to what 6LoWPAN did for IEEE 802.15.4, along with the 6lo-RFID frame format, header compression, fragmentation, network addressing and communication. The designed architecture is *fully interoperable* with the EPCglobal Class-1 Generation-2 UHF RFID protocol (hereinafter referred to as Gen-2 protocol) [6].

In the proposed framework, the RFID reader operates as an “enhanced IPv6 edge router”; it forwards data to and retrieves data from the IPv6-enabled RFID tags. To this purpose, the organization of the RFID tag’s User Memory (UM) is specifically extended to enable temporary storage of an IPv6 packet. The proposed approach also enables smart RFID tags *to exchange data with one another*, thus opening up new application opportunities for RFID systems.

Another major contribution of this paper is the demonstrator built as a proof-of-concept to show the feasibility of implementing a CoAP-over-IPv6 protocol stack in an UHF smart RFID tag fabricated in our labs.

Section II reports a brief overview of the standardization activities for IoT within the IETF and the basics of the RFID Gen-2 protocol. Section III discusses the motivations behind this study and related research works. Section IV describes the proposed IPv6-based architecture designed for inclusion of the RFID technology into the IoT. The smart-tag platform developed for demonstration and performance assessment purposes along with early experimental results are included in Section V. Section VI and VII conclude the paper by summarizing results, discussing open challenges and providing an outlook on future works.

II. Reference standards

IETF protocol stack for IoT

In the last decade, several IETF WGs (notably, CoRE, 6LoWPAN, 6lo) have been active on standardizing an IPv6-based framework for the IoT, which retains IP stability, scalability and interoperability, while meeting the resource-constrained nature of IoT devices.

The Constrained RESTful Environments (CoRE) WG has defined CoAP [8], a lightweight application-layer protocol intended to run on constrained IP networks for accessing and manipulating simple devices' resources through a RESTful approach.

Running IPv6 on constrained nodes has to cope with characteristics such as small packet size, low bandwidth, and low power. For these reasons, the IPv6 over Low power WPAN (6LoWPAN) WG [4] has standardized an appropriate adaptation layer for sending IPv6 packets over IEEE 802.15.4 networks.

The more recently formed 6lo WG [7] has replaced 6LoWPAN with the purpose of extending IPv6 over constrained node networks other than IEEE 802.15.4, by designing "IPv6-over-foo" adaptation layer specifications. Unfortunately, solutions designed for BLE, NFC, or other link-layer technologies, are unsuited to RFID Gen-2 systems, thus motivating this work.

UHF RFID EPCglobal Gen-2 Air Interface Protocol

The RFID air interface for UHF passive tags is regulated by the EPCglobal standardized Gen-2 protocol [6]. It defines modulations, encoding, medium access schemes, and a set of basic commands for tag-reader interaction. Three steps allow the reader to establish a link with a targeted tag: *(i) select* for choosing the transponders to be polled; *(ii) inventory* for completion of the identification of individual tags; and *(iii) access* to perform specific operations once the tag has been uniquely identified.

The *singulation* of a tag is required before a reader can access it and perform reading or writing operations on the tag's UM. The reading operation consists of a *Read* command allowing to read up to 255 16-bit words at a time, whereas the *Write* (operating on a single 16-bit word at a time) or *BlockWrite* (optional, handling up to 255 16-bit words in a single operation) commands are used to update the UM content.

III. Background and motivation

Introducing computing and sensing capabilities into passive UHF tags represents a tremendous breakthrough in the RFID technology and opens up new indoor [1] and outdoor [9] application scenarios. Compared to classic sensor nodes, sensing RFID tags benefit from a unique passive air interface used for both wireless power transfer and communication. Notwithstanding, long-range passive RFID systems still miss global networking interoperability, and this limits their large-scale adoption.

Evidence of remarkable efforts towards a networking solution for sensing RFID tags is found in the literature, but the related works are quite fragmented and *do not support internetworking at the IP layer*, which represents the de-facto standard approach to enable global communications for heterogeneous IoT systems. In [1], sensor data are embedded into an EPC-compliant ID of a programmable RFID sensor. An alternative strategy, designed in

[2], relies on a microcontroller to acquire samples from the embedded sensor and store them in the tag's UM for later retrieval by the reader. In [3] data logging is enabled in a Gen-2 compatible semipassive reconfigurable UHF RFID sensing platform by defining a management scheme for the tag's UM, where some bits are used to control the sensing functionality. All the above mentioned researches have just defined proprietary protocols to enable the exchange of RFID sensing data, and thus they do not provide the desired level of interoperability required by IoT. In [10] the authors propose a solution to enable generic-purpose data traffic exchange through RFID readers which act as relays, collecting and forwarding messages from/to smart RFID tags. However, in this case, the adoption of a specific frame format makes the internetworking with existing IP-based IoT systems complex.

Other works in the literature focus on RFID inclusion by leveraging IP stack, but their goals are only partially achieved and, typically, they lack flexibility. The IoT6 architecture in [11] integrates and makes the information stored in the EPCglobal platform transparently accessible in the IoT landscape. This approach only enables global access to data generated by RFID systems, but real-time communications with RFID tags are not allowed. On the other hand, to extend IPv6 to the RFID world, in [12] the *virtual endpoint* concept is introduced, i.e., a device that does not natively support IPv6 is enabled to act as a data source/destination thanks to a transparent proxy, which proactively manages its operations. We have also proposed a CoAP-compliant solution in [13], where RFID tags' resources are accessible via a reader acting as a CoAP proxy. The solution is backward compatible with legacy RFID systems, although it enforces the reader to interpret CoAP application requests and consequently manage tags' resources. However, flexibility and scalability are limited by vendor lock-in issues and by the need for reader's firmware updates whenever tags have new resources to offer. In [14] an ultra-low-power hybrid sensing network composed of 6LoWPAN nodes that integrate UHF RFID functionalities is designed for e-Health applications. This solution requires that hybrid RFID Gen2 tags are equipped with an additional IEEE 802.15.4 interface to exchange 6LoWPAN-compliant packets, thus increasing both device complexity and production cost.

Providing native IP networking capabilities based only on RFID radio interfaces is the next step to foster worldwide deployment of smart RFID tags in the IoT ecosystem. To this aim, we have designed an *IPv6 adaptation layer* customized to run on top of the RFID link layer and thus able to guarante full interoperability with existing IP-based systems.

IV. A UNIFIED IOT ARCHITECTURE TO ACCESS RFID RESOURCES

The unified architecture definition follows a holistic approach to enable full access to the resources offered by RFID sensing tags similarly to any other IoT node. *Network-layer interoperability* is achieved by relying on the IPv6 protocol, whereas a CoAP-based interface, designed to access the tag's resources, provides *application-layer interoperability*.

A full IETF-compliant CoAP-over-IPv6 protocol stack is implemented and properly customized to support IPv6 communication over UHF Gen-2 smart tags (upper part of Fig. 1 - *Smart Tags*). The availability of lightweight IP stack implementations, such as *uIP* (micro-IP) and *lwIP* (lightweight-IP), makes the solution feasible by meeting typical memory and computational constraints of smart tags.

In the proposed architecture, the *6lo-RFID layer* operates as an adaptation layer to enable transparent bidirectional data exchange of smart UHF RFID tags both locally, i.e., among tags under the coverage of the same reader, and with remote IPv6 Internet nodes, such as other constrained IoT devices, general end-user terminals, IoT-Cloud management systems. By following the 6LoWPAN terminology, the RFID reader operates as a 6LoWPAN Border Router (6LBR), i.e., an enhanced IPv6 edge router that forwards messages to and proactively retrieves data from RFID tags, which are equivalent to 6LoWPAN Nodes (6LNs).

The reader is in charge of retrieving and forwarding a 6lo-RFID frame by issuing *reading* and *writing* commands to the smart tag. 6lo-RFID frames are stored in the tag's UM, while access to the UM is regulated on a *master-slave* basis, as explained in the following.

Our solution purposely relies on the standard Read and Write commands, so to reuse the existing RFID reader logic. It results feasible and convenient not only to RFID sensing tags, like in [1] wherein the MCU handles both the RFID protocol and the sensor data, but also to RFID sensing platforms wherein the MCU relies on an embedded UHF RFID chip featuring a Serial Peripheral Interface (SPI) to transfer sensor data [2]. In this way, only a firmware update is required to allow available smart RFID tag platforms to apply our solution.

For the sake of completeness, the lower part of Fig. 1 - *Legacy Tags* reports a proxy-based solution such as the one we proposed in [13] for integration of legacy (i.e., non smart) RFID tags, which only implement Gen-2 protocol. In this case, the reader implements a full IPv6 stack and acts as a transparent CoAP proxy on behalf of the tags under its coverage.

The implementation of the illustrated architecture on RFID readers and smart tags requires the detailed specification of frame and memory formats and tag-reader communication protocols. Before going into these details in the next sections, we highlight the challenges raised by Gen-2 RFID tags compared to IoT sensors, such as those based on IEEE 802.15.4:

- *No native support for unicast mode*. Gen-2 is designed for fast inventory of extremely simple tags, whose unique information is the associated EPC code. Our solution aims at introducing unicast/multicast communication into the RFID system to allow data exchange among smart tags and generic IPv6 nodes.
- *Reader-oriented communication*. The message flow on the air interface is centrally managed by the reader that must implement proper polling schemes to grant each tag a chance to send its own generated message.
- *No network-layer addressing*. EPC codes have not been created to enable routing functionalities over the Internet and to guarantee remote access to RFID tags. Specific mechanisms that map EPC codes onto IPv6 addresses are required.
- *Limited frame size*. The UM size of commercial tags, which will store the upper-layer packet, typically ranges from 16 to 1024 bytes. Our solution must implement efficient methods to carry upper-layer messages through Read and (Block)Write commands on the UM, and include fragmentation and reassembly procedures to convey large IPv6 packets (up to 1280 bytes).

A. UM and 6lo-RFID frame format

To enable the exchange of IPv6 packets generated by (or directed to) a given tag, we propose a new organization of the UM, illustrated in Fig. 2.

According to the EPCglobal Tag Data Standard, the first 8 bits of the UM contain the Data Storage Format Identifier (DSFID) field, which specifies the format for the remainder of the UM bank. This field includes the Data Format to indicate what data system predominates in the UM. Obviously, a new *data format code* must be associated to IPv6-based RFID communications; this allows for keeping compatibility with other RFID data standards. The following bits are reserved for future use (RFU) to manage protocol version, priority handling or queue congestion management.

As Read and Write commands operate on 16-bit word units, we propose to store the 6lo-RFID frame by starting from the second 16-bit word of the UM to avoid useless reading/writing of the DSFID field when sending/retrieving data packets. The following fields are included in the (2-4 bytes long) *6lo-RFID header*:

- *Token* (1 bit) is used to guarantee exclusivity in the writing operation and, thus, manage the concurrent access to the UM from the reader and the MCU. Its value 0 enables the reader operations, while value 1 allows the MCU to operate on the UM. To avoid deadlock, the reader resets the UM to start a new information exchange if the MCU does not clear the Token bit before a given timeout (set by considering the estimated time required by the MCU to insert a new message and/or the traffic pattern).
- *Reader Message - RMG* (1 bit): indicates if the reader has written a message into the tag. The tag subsequently clears this bit after reading the relevant reader message.
- *Tag Message - TMG* (1 bit): indicates if the tag has inserted a message in the UM. The reader clears this bit after reading the tag message.
- *Extended Length - EL* (1 bit): if the packet length is more than 255 bytes and the UM contains it, then this bit must be set to 1 to use an extended Length field of 16 bits.
- *Local addressing - LA* (1 bit): its value 1 indicates that a local addressing strategy is performed. In this case, the header includes source and destination addresses (more details are given in the following).
- *Reserved for future use - RFU* (3 bit).
- *Length* (8-16 bits): indicates the number of bytes that compose the 6lo-RFID packet (the field allows to exchange up to 65535 byte-packets).

A Frame Check Sequence (FCS) of 32 bits is added at the tail of the 6lo-RFID frame to detect tampering or inconsistencies due to errors or partial writing of the UM.

Considering the limited frame size, headers compression is highly advisable to efficiently exchange data. To save valuable bits in the memory, the IPv6 and UDP headers shall be compressed according to the standard encoding schemes defined in IETF RFC 6282 [4]. Since IPv6 defines a maximum transmission unit (MTU) of 1280 bytes, the 6lo-RFID layer must also implement fragmentation and reassembly functionality, as defined in RFC4944 [4], when the UM is not large enough to contain the whole IPv6 packet.

B. 6lo-RFID communication

Given the passive nature of RFID tags, we propose a Master-Slave approach to access the UM and manage reader-tag interaction, where the reader acts as the master and controls the data flows from/to all the tags in its coverage range. A centralized solution at the 6lo-RFID layer is the most suitable one to work on top of the Gen-2 protocol, since it can easily exploit the already defined Gen-2 commands and guarantee full compatibility with the underlying RFID layers. In particular, the reader can forward a message to a tag (downlink) and give the tag the opportunity to transfer its data in the opposite direction (uplink), as illustrated in Fig. 3, where the message exchange over the RFID air interface is shown to carry a single data transfer.

The reader starts the tag singulation phase by specifying the EPC code of the desired tag in a *Select* command, so that only the tag owning the specified EPC can participate in the inventory phase, and tag-to-tag collisions are avoided. The algorithm for tag selection from the reader's polling list to perform singulation is out of the scope of this paper. Then, the *Write* and *Read* commands allow data transfer in downlink and uplink directions, respectively. If the reader has data to send, then it writes the 6lo-RFID frame into the UM and sets the fields in the header accordingly. Otherwise, it simply sets the Token bit to give the tag's MCU the chance to send its packet. Since the reader cannot be aware of when the tag has completed its operation, it periodically reads the 6lo-RFID header until the Token bit has been released by the tag. If the MCU has put a new packet in the UM, then the reader reads it; otherwise, it can either send a new message to the same tag or proceed to poll the next tag.

Multicasting transmissions from the reader can be managed through multiple unicast communications (same multicast packet replicated on each 6lo-RFID downlink). Differently, a tag willing to transmit an IPv6 multicast packet delivers the packet to the reader, which will forward it to the multiple destinations. *Tag-to-tag* data exchange is also allowed, but it can only happen via the reader.

C. Network addressing and Neighbor Discovery

RFID identifiers, i.e. EPC codes, are unique and potentially usable to address nodes in the Internet. Unfortunately, they require 96-bits address space and are not compliant with the 128-bit IPv6 addressing. Thus, an appropriate addressing scheme for 6lo-RFID communications must be designed.

We use a short-address assignment for *local* (i.e., within the reader coverage area) data exchange among smart tags. Each time the reader discovers a new tag, it allocates an *8-bit address* and creates a new entry in its Neighbor Cache; this way, the reader can manage a population of 255 tags.

End-to-end communication over the Internet is instead enabled by a mechanism that *maps EPC codes onto IPv6 addresses*. We assume that a hashing function (such as CRC-32) is used to map the EPC code onto the *IPv6 interface identifier* (IID), similarly to [15]. The approach is feasible for RFID tags with EPC codes of any length, and likely satisfies the requirements of consistency and uniqueness within a subnet.

Then, the *IPv6 link-local address* is built by appending the IID to the link-local unicast prefix, whereas the *global IPv6 addresses* can be formed by prepending a valid sub-network prefix provided by the reader acting as 6LBR. This information could be obtained by using a simplified version of the standard 6LoWPAN Neighbour Discovery (in RFC

6775 [4]), accounting for the star topology of RFID systems (no further complexity due to multi-hop). In particular, a 6lo-RFID tag must register its non-link-local addresses with the reader, by sending a Neighbour Solicitation message with the Address Registration Option, and then process the Neighbour Advertisement accordingly.

V. Platform Implementation

In this Section, we describe the experimental results achieved by exploiting a smart RFID platform specifically designed and built in our labs to the purpose of managing bidirectional 6lo-RFID compliant communications.

The fabricated smart tag platform illustrated in Fig. 4(a) includes an eXtreme Low Power (XLP) MCU and an RFID tag IC connected to a dipole antenna used for data transmission, designed to resonate at the frequency of interest. The system is powered by a 3 V coin battery, which supplies the power requested by the MCU and the tag to operate in Battery Assisted Passive (BAP) mode. The MCU provides the required energy to the digital circuitry of the RFID tag allowing the communication through the SPI bus and monitoring the radio activity. The platform is implemented by using off-the-shelf discrete components on a FR-4 substrate with a dielectric constant of ϵ_r=2.2 and a dielectric thickness of 0.8 mm.

The embedded Fujitsu FRAM MB97R803A/B UHF RFID chip is used within the platform to ensure data transmission according to the Gen-2 protocol. The tag IC also supports SPI communications for Read/Write operations by the MCU and is equipped with a FRAM memory, which allows faster writing times compared to classic EEPROM memory.

The MCU is an XLP 8-bit PIC16LF1503 from Microchip Technology, programmed with an energy-efficient firmware that supervises the whole operations of the platform. The implemented firmware enables the tag's memory access from the MCU by Read/Write operations, allowing to define access priorities and preventing concurrent access from a standard Gen-2 reader. To this aim, also an RF switch ADG904-R from Analog Devices Inc. has been added. This allows for enabling/disabling communication between the antenna and the tag and providing a priority access from the MCU to the tag UM when requested by the proposed approach. A dipole antenna is designed and patterned directly on the PCB.

It is worth underling that, although open-source documentation is publicly available to build smart tag platforms, we preferred to build our own platform since commercial smart RFID tags are not freely programmable and customizable. On the other hand, our networking solution can be easily applied to the majority of smart tags in literature by just updating the MCU firmware. The only requirement to guarantee the 6lo-RFID communication is the availability of mechanisms to define access priorities and prevent concurrent access of the UM by both reader, via the RFID radio interface, and the MCU, via the SPI interface.

Finally, in the current version of the 6lo-RFID tag platform, we have implemented the "Data Plane" functionalities, to enable packet exchange compliant with the IETF standards for constrained devices. Therefore, the packet headers follow the specifications of the 6LoWPAN, IPv6, UDP, and CoAP standards. In the next future, we are going to enhance the tag firmware by comprehensively implementing the Neighbour Discovery functionality.

A. Experimental Results

In the deployed testbed (in Fig. 4(b)), the smart tag is interrogated by a standard commercial RFID reader, ThingMagic Micro Embedded, equipped with a circularly polarized antenna. The reader operates in the 865-868 MHz band, its transmission power is 30 dBm, and the preconfigured Gen-2 protocol setting is used.

The "PC Host" and the "Reader" represent the *IPv6 6lo-Reader*, a unique logical entity from an architectural point of view. Due to the vendors' lock-in, commercial readers do not allow either for modifying the original firmware or for executing application code on top of it. This is why we implemented the logic of the IPv6 6lo-Reader in a PC directly connected to the reader.

We set the distance between reader antenna and tag to 30 cm. Beyond this distance, we observed an increasing number of erroneous Read/Write operations, which caused command retransmissions and severely increased latency. Clearly, the communication range can be further improved by introducing hardware enhancements, as discussed in the next section.

We focus on the performance of the 6lo-RFID framework when two CoAP methods are used to execute different operations on the tag memory:

- GET: to retrieve data from the resource identified by the requested Uniform Resource Identifier (URI);
- PUT: to create or update the resource identified by the requested URI with the enclosed representation.

The analysis of these methods allows for characterizing the performance in both uplink (GET) and downlink (PUT) directions. Indeed, with GET the resource (i.e., the packet payload) is transferred from the tag to the reader, whereas with PUT the value of the resource to update/change is sent by the reader to the tag.

We assume that the CoAP communication endpoints are the reader and the smart 6lo-RFID tag platform, since we are interested in characterizing the 6lo-RFID link performance. Obviously, in case of a remote CoAP client, the end-to-end performance should include the Internet connection between the CoAP client and the RFID reader. Compression of IPv6 and UDP headers (i.e., using 6LoWPAN Context Identifier to completely elide source and destination addresses) is assumed to minimize the packet overhead.

The performance is measured in terms of *latency* in the CoAP communication between reader and tag by evaluating the time required to receive the desired response from a target device over the investigated 6lo-RFID link. Indeed, this metric accounts for the elapsed time from the instant when the reader begins to transmit the CoAP command (either GET or PUT) to the instant when it retrieves the CoAP response. Fig. 5 shows the experimental latency of both GET and PUT methods for different packet payload size. An increase in the message size implies an increment in the latency in both cases. Higher delays are also measured for the PUT method, since higher time is required by the reader to write the payload of the CoAP request into the UM over the RFID radio interface, compared to the time needed by the MCU for writing operations over the SPI interface. Furthermore, according to the used ThingMagic reader setup, we remark that the reading operation of the tag memory is based on the Gen-2 Read command. This latter allows to read up to 64 16-bit words at a time. Differently, the writing operation relies on the Gen-2 Write command, which operates on a single 16-bit word at a time. The different performance

in downlink and uplink directions of the envisaged 6lo-RFID link represents a peculiar feature of the investigated system.

From the latency results, it is easy to derive the values of the maximum achievable goodput, defined as the *useful* data rate [bit/s], i.e., excluding any control overhead information such as the 6lo-RFID and the upper-layer headers. The goodput is computed as the ratio between the amount of payload transmitted in a time slot and the duration of the data exchange between reader and tag. For packets with 1536 bits of payload, the resulting goodput for the PUT and GET methods is respectively around 1.58 and 3.36 kbps.

As a final experiment, we evaluate the power consumption of the proposed 6lo-RFID framework. Fig. 6 shows the mean power consumption of the smart tag platform when considering different request time intervals for the GET command. Obviously, higher consumption corresponds to shorter intervals and longer payload. A zero interval is the condition used to evaluate the latency in Fig. 5.

We underline that a feature of our smart RFID tag platform is the ability to wake MCU up when a memory writing operation is recognized on the RFID interface. This allows for monitoring radio activity with an extremely low power consumption.

To sum up, in sensing and actuation applications, the IoT devices typically need to transmit data periodically or when queried by other clients. The net payload for such scenarios is typically small, ranging from 10 to 200 bytes per packet, and the latency requirements are often moderate, in the range of a few seconds. These devices need to be energy efficient and low cost. Our performance evaluation has highlighted that the proposed 6lo-RFID tag is able to perform full CoAP-based interactions in less than a second for different payload sizes feasible for a broad range of IoT use cases. Furthermore, the power consumption is extremely low, accounting that packets are exchanged over the RFID radio interface.

VI. Open challenges

Relying on the UHF RFID radio interface for Internet connectivity opens up several interesting areas of research:

- *Hardware*: Enhanced *energy harvesting* techniques can improve the efficiency of RF power transfer and increase the communication range with the ultimate goal to develop a fully passive, IPv6-compliant, and energy-efficient RFID tag platform.
- *Networking*: Enabling IPv6 connectivity over smart UHF tags raises other challenges beyond packet transfer, such as mobility, network management, and polling schemes optimized to different application traffics. Indeed, future studies still needs to be conducted to identify the maximum number of tags which can be served while guaranteeing the desired Quality of Service. To this aim, peculiar features of the RFID communication, such as the persistence of the frame stored in the tag memory, may be exploited to improve the overall network goodput. Multi-reader scenarios are challenging, as they require appropriate reader-to-reader anti-collision protocol to avoid interference at the network level.
- *Security*: To guarantee the safe deployment of IPv6-based RFID tags, security issues must be carefully addressed. Although security mechanisms such as Datagram Transport Layer Security (DTLS) are currently

investigated by the IETF WGs to guarantee end-to-end secure communications for IoT devices, further efforts are required to protect the link between reader and tags. In this regard, the last version of the Gen-2 standard includes support for cryptographic authentication of tags and readers; however, additional lightweight techniques for data confidentiality and integrity are recommended.

- *Semantic interoperability*: In the last decade, RFID communities have defined specific ontologies for their core applications. To enable a unified access to IoT services across manifold domains, RFID schemes should be appropriately integrated in the promising semantic Web of Things.

VII. Conclusions

In this paper, an IPv6-based framework has been designed and tested to transparently integrate RFID systems into the IoT domain and enable IPv6 communication and networking over UHF RFID smart tags. An adaptation layer is proposed, in accordance with the IETF protocol stack for IoT. The solution acts as an overlay above the link-layer technology and is fully compliant with the standard EPCglobal Gen-2 protocol. The experimental results demonstrated the feasibility to achieve response times and data rates fitting sensing applications, and testified to an extremely low power consumption.

References

[1] A. P. Sample, D. J. Yeager, P. S. Powledge, A. Mamishev, and J. R. Smith, "Design of an RFID-based battery-free programmable sensing platform," IEEE Trans. on Instrumentation and Measurement, 57:11, pp. 2608 2615, 2008.

[2] D. De Donno, L. Catarinucci, and L. Tarricone, "A battery-assisted sensor-enhanced RFID tag enabling heterogeneous wireless sensor networks," IEEE Sensors Journal, 14:4, pp. 1048 1055, 2014.

[3] M.S. Khan, M.S. Islam, and D. Hai, "Design of a Reconfigurable RFID Sensing Tag as a Generic Sensing Platform Toward the Future Internet of Things," IEEE Internet of Things Journal, 1:4, pp. 300 310, 2014.

[4] IETF 6LoWPAN Working Group, https://tools.ietf.org/wg/6lowpan.

[5] J. Nieminen, et al., "Networking Solutions for Connecting Bluetooth Low Energy Enabled Machines to the Internet of Things," IEEE Network, 28:6, pp. 83 90, 2014.

[6] Specification for RFID Air Interface, EPC Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860 MHz - 960 MHz, Version 2.0.1, April 2015.

[7] IETF IPv6 over Networks of Resource-Constrained Nodes (6lo) Working Group, https://tools.ietf.org/wg/6lo (accessed on June 2016).

[8] Z. Shelby, K. Hartke, C. Bormann, and B. Frank, "The Constrained Application Protocol (CoAP)," RFC 7252, June 2014.

[9] I. Farris, L. Militano, A. Iera, A. Molinaro, and S. C. Spinella, "Tag-based cooperative data gathering and energy recharging in wide area RFID sensor networks," Ad Hoc Networks, 36:1, pp. 214228, 2015.

[10] A. Molina-Markham, S. S. Clark, B. Ransford, and K. Fu, "Bat: Backscatter anything-to-tag communication," in "Wirelessly Powered Sensor Networks and Computational RFID," Springer, 2013.

[11] A. J. Jara, P. Lopez, D. Fernandez, J. F. Castillo, M. A. Zamora, and A. F. Skarmeta, "Mobile digcovery: discovering and interacting with the world through the internet of things," Personal and ubiquitous computing, 18:2, pp. 323-338, 2014.

[12] A. J. Jara, P. Moreno-Sanchez, A. F. G. Skarmeta, S. Varakliotis, and P. Kirstein, "IPv6 Addressing Proxy: Mapping Native Addressing from Legacy Technologies and Devices to the Internet of Things (IPv6)," Sensors, 13:5, pp. 6687-6712, 2013.

[13] I. Farris, A. Iera, A. Molinaro, and S. Pizzi, "A CoAP-compliant solution for efficient inclusion of RFID in the Internet of Things," 2014 IEEE Global Communications Conference, Austin, TX, 2014, pp. 2795-2800.

[14] L. Catarinucci et al., "An IoT-Aware Architecture for Smart Healthcare Systems," in IEEE Internet of Things Journal, vol. 2, no. 6, pp. 515-526, Dec. 2015.

[15] G. Rizzo, et al., "IPv6 mapping to non-IP protocols," draft-rizzo-6lo-6legacy-03, March 2015.

Ivan Farris is currently a Ph.D. student in Information Technology Engineering at the University of Reggio Calabria, Italy. He received a B.Sc. degree in Telecommunications Engineering and a M.Sc. degree in Computer and Telecommunications Systems Engineering from the University of Reggio Calabria in 2011 and 2013, respectively. His research interests include Internet of Things, Edge computing, and network softwarization.

Sara Pizzi is a Research Associate in Telecommunications at University Mediterranea of Reggio Calabria, Italy. From the same University she received the 1st (2002) and 2nd (2005) level Laurea Degree, both cum laude, in Telecommunication Engineering and the Ph.D. degree (2009) in Computer, Biomedical and Telecommunication Engineering. Her current research interests focus on QoS provisioning and resource allocation in wireless networks, multicast support in broadband wireless networks, Internet of Things, and RFID systems.

Massimo Merenda received the BS, MS, and PhD in electronic engineering from Mediterranea University of Reggio Calabria, Italy, in 2002, 2005 and 2009, respectively. From 2003 to 2005 he was a fellow at IMM-CNR, Naples, Italy. Since April 2011 he has been a Temporary Research Associate in the DIIES department of Mediterranea University of Reggio Calabria. His research involved the design and development of Application Specific Integrated Circuits (ASIC), silicon sensors, embedded systems and intelligent radiofrequency identifiers (Smart-RFID).

Antonella Molinaro graduated in Computer Engineering at University of Calabria (1991), received a Master degree in Information Technology from CEFRIEL/Politecnico di Milano (1992) and a Ph.D. degree from University of Calabria (1996). She is an Associate Professor of Telecommunications at University Mediterranea of Reggio Calabria, Italy since 2005. Before, she was with University of Messina and University of Calabria as an Assistant Professor, with Polytechnic of Milano as a research fellow, and with Siemens, Munich, as a CEC Fellow in the RACE II program. Her current research activity focuses on wireless networking, vehicular networks, and information-centric networking.

Riccardo Carotenuto was born in Rome, Italy. He received the Dr. Sc. degree in Electronic Engineering from the University of Rome La Sapienza, Rome, Italy. He served as a postgraduate fellow at the Department of Electronic Engineering of the University of Rome La Sapienza, working on complex and partially known dynamical system. In 1997, he earned the Ph.D. degree from the University of Rome La Sapienza, and joined the Acustoelectronics Laboratory (ACULAB), Department of Electronic Engineering, of the University Roma Tre, working on ultrasonic micromotors, on resolution enhancement of echographic imaging systems, and on theory and technology of capacitive micromachined ultrasound transducers. Since 2002, he has joined the Department DIMET (now DIIES), University Mediterranea, Reggio Calabria, Italy, as Associate Professor. His main interests include indoor localization, ultrasound imaging, ultrasound actuators, time series prediction, nonlinear systems identification and control, neural networks theory and applications. Dr. Riccardo Carotenuto is author of more than 80 papers published on International Journals and Conferences Proceedings.

Antonio Iera received the degree in computer engineering from the University of Calabria, Arcavacata, Italy, and the Masters Diploma degree in information technology from CEFRIEL/Politecnico di Milano, Italy, and the Ph.D. degree from the University of Calabria, in 1991, 1992, and 1996, respectively. Since 1997, he has been with the University of Reggio Calabria, Reggio Calabria, Italy, and currently holds the position of Full Professor of Telecommunications and the Director of the Laboratory for Advanced Research in Telecommunication Systems. His research interests include next generation mobile and wireless systems, RFID systems, and Internet of Things.

List of Figures

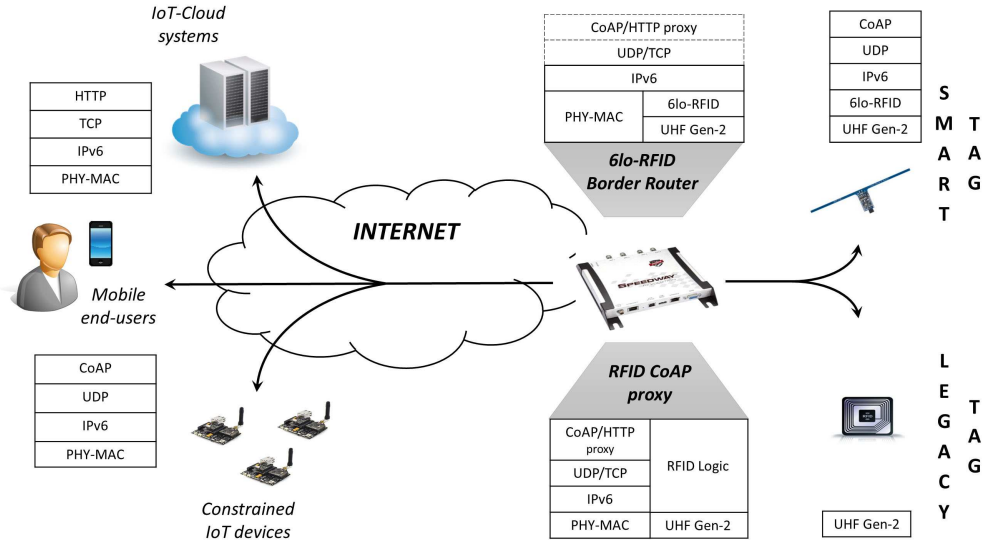


Fig. 1. A comprehensive IPv6-based architecture for RFID integration in IoT.

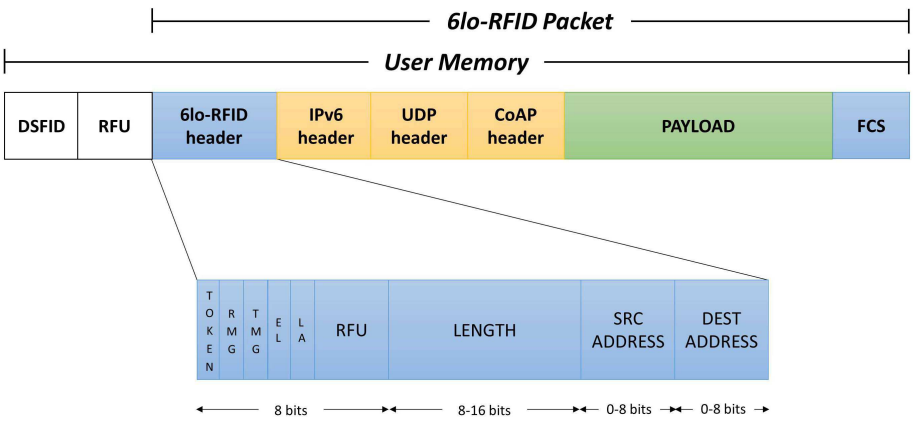


Fig. 2. 6lo-RFID packet format.

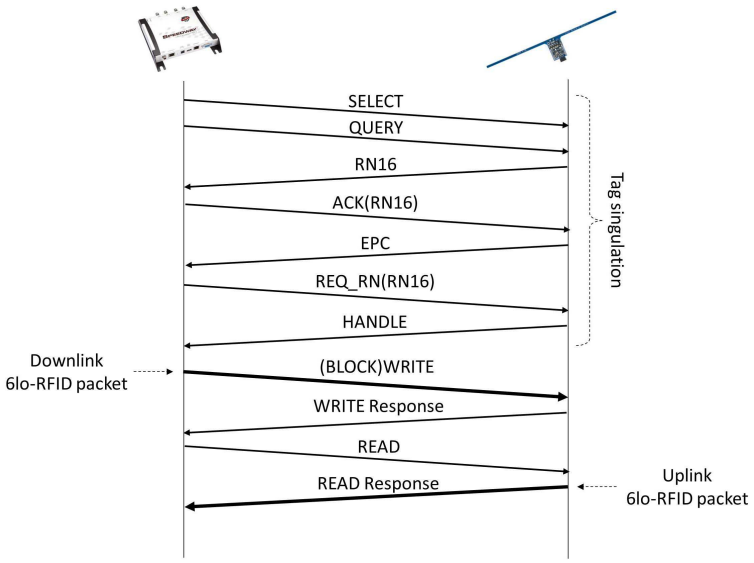
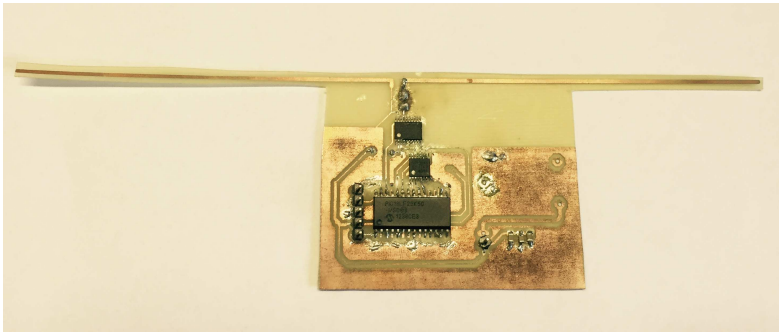
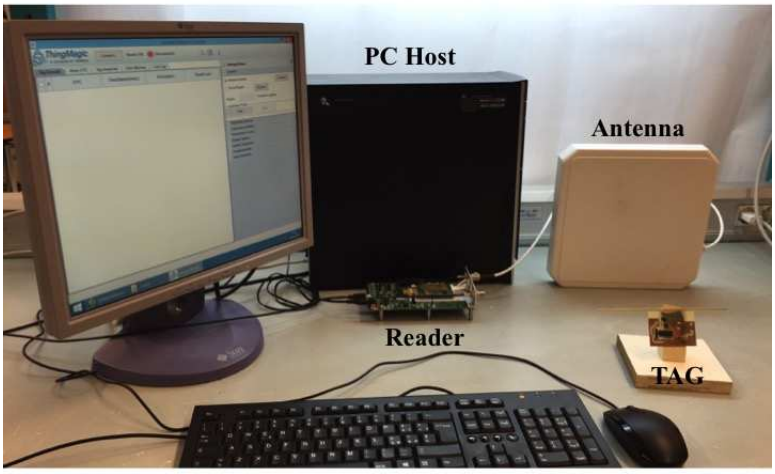


Fig. 3. Gen-2 messages exchange for 6lo-RFID communication.



(a)



(b)

Fig. 4. A snapshot of (a) the tag, and (b) the testbed.

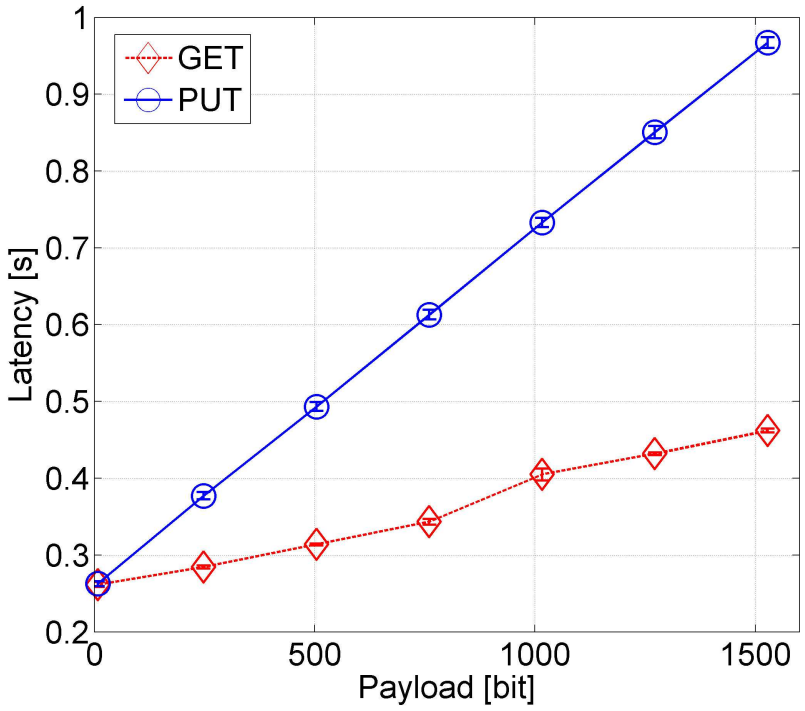


Fig. 5. 6lo-RFID latency for both GET and PUT methods.

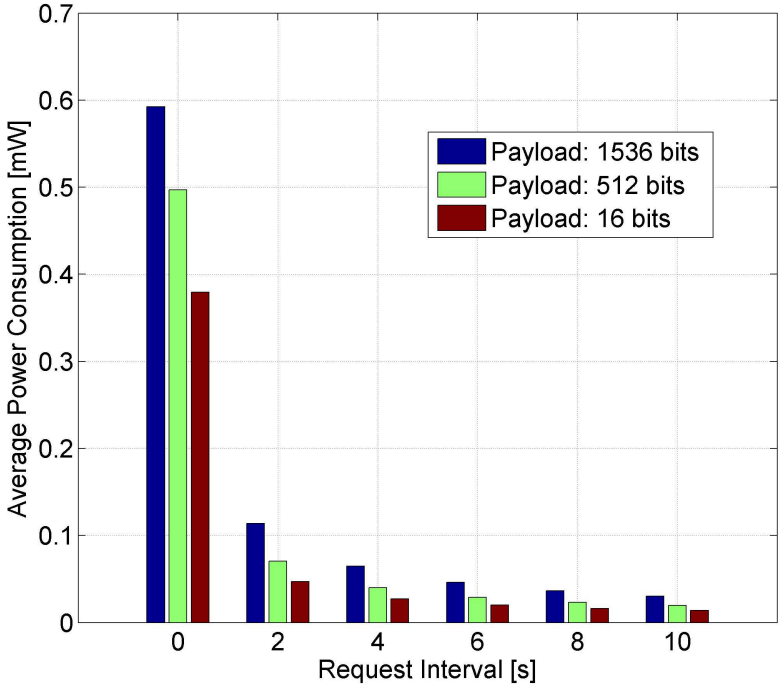


Fig. 6. Average power consumption of the proposed 6lo-RFID tag platform for the GET method at different request interval times.