IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received December 15, 2020, accepted December 24, 2020, date of publication January 1, 2021, date of current version January 7, 2021.

Digital Object Identifier 10.1109/ACCESS.2020.3048673

Wake-Up Radio: An Enabler of Wireless Convergence

MARTÍ CERVIÀ CABALLÉ[1], ANNA CALVERAS AUGÉ[1], AND JOSEP PARADELLS ASPAS[1,2]
[1]Department of Network Engineering, Universitat Politècnica de Catalunya (UPC), 08034 Barcelona, Spain
[2]i2CAT Foundation, 08034 Barcelona, Spain

Corresponding author: Martí Cervià Caballé (marti.cervia@upc.edu)

This work was supported in part by European Regional Development Fund of the European Union and the Spanish Government, through the Agencia Estatal de Investigación, under Grants TEC2016-79988-P and PID2019-106808RA-I00, and in part by the Secretaria d'Universitats i Recerca del Departament d'Empresa i Coneixement de la Generalitat de Catalunya under Grant 2017 SGR 376.

ABSTRACT Nowadays, several wireless communication solutions targeted at low-power devices (e.g. sensors, actuators) compete for market dominance. As a consequence, the deployment of the Internet of Things is slowed by fragmentation. However, Cross-Technology Communication (CTC) has appeared as a solution capable of bridging the compatibility gap between non-interoperable devices. In this way, we propose using Wake-up Radio (WuR) to achieve high rate bidirectional CTC, introducing WuR assisted CTC (WuR-CTC). Although WuR was originally proposed to reduce radio power consumption, it provides a secondary communications channel that can be used for high-speed and bidirectional CTC. With WuR-CTC, we demonstrate that WuR can lead to interoperability between non-compatible wireless devices, therefore, providing a compelling reason for the harmonization of the nascent WuR specifications. This article presents a WuR-CTC solution implemented on an heterogeneous testbed with IEEE 802.11 and IEEE 802.15.4 devices. The resulting testbed achieves reliable CTC with an effective throughput of 26.7 kbps. Moreover, the addition of WuR to the testbed devices provides energy savings, and, in a clear advantage over traditional duty-cycled solutions, allows devices to asynchronously maintain communications.

INDEX TERMS WLAN, IEEE 802.11, WPAN, IEEE 802.15.4, IEEE 802.11ba wake-up radio, cross-technology communication, energy-efficient communication.

I. INTRODUCTION

The Internet of Things (IoT) has motivated the extension and development of a considerable number of Wireless Personal Area Network (WPAN) solutions that suit its set of requirements (e.g., low-power consumption, low-cost and increased range). However, these solutions define non-compatible physical layer implementations. Thus, preventing direct communications between heterogeneous devices. The lack of compatibility between existing wireless solutions is one prominent cause of the often cited IoT fragmentation problem.

Recently, Wireless Local Area Network (WLAN) solutions have entered the IoT market in force, capturing up to 48% of new developments [1]. The adoption rate of WLAN solutions is rising and new WLAN releases, such as IEEE 802.11ba [2], [3], will allow deployments of WLAN devices in more energy-constrained application domains. Nonetheless, despite these recent developments, fragmentation remains one of the main challenges for the IoT, which now includes both WLAN and WPAN solutions.

Gateway devices provide a simple way to mitigate this problem by performing protocol translation between non-compatible devices. This approach has led to the adoption of the gateway as the central element in many IoT networks [4]. However, the use of a gateway presents drawbacks. One of them is that signals must be retransmitted by the gateway after translation, thus decreasing the spectral efficiency of the network. Another disadvantage is the coexistence issue caused by the inclusion of different wireless solutions in the gateway. Additionally, the use of a gateway may cause reliability problems. For example, a gateway malfunction can cut connectivity between non-compatible devices as well as with external networks.

Cross Technology Communications (CTC) research concentrates on finding techniques that allow direct interaction

The associate editor coordinating the review of this manuscript and approving it for publication was Renato Ferrero.

between non-compatible devices. In this way, CTC research proposes interaction methods that, in most cases, were not foreseen by the wireless solution designers. Fig.1 shows the comparison between CTC and gateway use on an example network with IEEE 802.11 and IEEE 802.15.4 devices. Unfortunately, the use of those unforeseen methods comes with drawbacks (see Section II) that can make CTC low-throughput and, in most instances, unidirectional. For example, [5] defines a mechanism for CTC from an IEEE 802.11 device to one or more IEEE 802.15.4 devices, but not the reverse. In this article, we propose using Wake-up Radio (WuR) to provide high-throughput bidirectional CTC.

Unidirectional CTC is only applicable to those use cases supported by one-way communications, such as sending commands unreliably and data dissemination through broadcast. In contrast with bidirectional CTC, heterogeneous networks where all devices can interact with no connectivity limitations are possible. Additionally, in a clear advantage over unidirectional CTC, bidirectional CTC supports reliable communications with acknowledgment messages.

Bidirectional CTC enables use cases that apply to the current IoT landscape. One of them concerns direct interaction between end-user devices and a WPAN network. For example, a mobile IEEE 802.11 terminal could configure, actuate, and obtain sensor data from an IoT network composed exclusively of IEEE 802.15.4 devices. Furthermore, bidirectional CTC benefits heterogeneous network architectures that appear as a consequence of the rapid pace of innovation in the IoT space. Specifically, after the initial deployment of an IoT network, subsequent extensions may use other wireless solutions due to improvements in the state-of-the-art. With bidirectional CTC, the devices forming these heterogeneous networks can freely communicate, regardless of their respective wireless implementations. Thus, CTC helps to create heterogeneous networks with more robust network topologies than those depending on gateways. For example, bidirectional CTC could enable the expansion of an IEEE 802.15.4 network composed of specialized low power devices with higher throughput IEEE 802.11ba devices, which could go on to form the backbone of the network. Thus, improving the overall performance of the resulting network. Of course, this example can be generalized to other wireless solution combinations. Simultaneously, CTC improves the coexistence between the different wireless solutions present in the network. For example, by using CTC, an heterogeneous network can adapt to changes in the environment by coordinately changing the operating frequencies of all device radios.

WuR [6] appeared as an energy-saving technique. Devices that implement WuR save power by sleeping in a low-power state, with their main radios turned off. However, they maintain a low-power Wake-up Receiver (WuRx) waiting for any incoming Wake-up Signal (WuS). WuR-enabled devices also can include a Wake-up Transmitter (WuTx) that is capable of sending WuS. Once woken up by a WuS, the receiver device turns on its main radio and becomes ready to

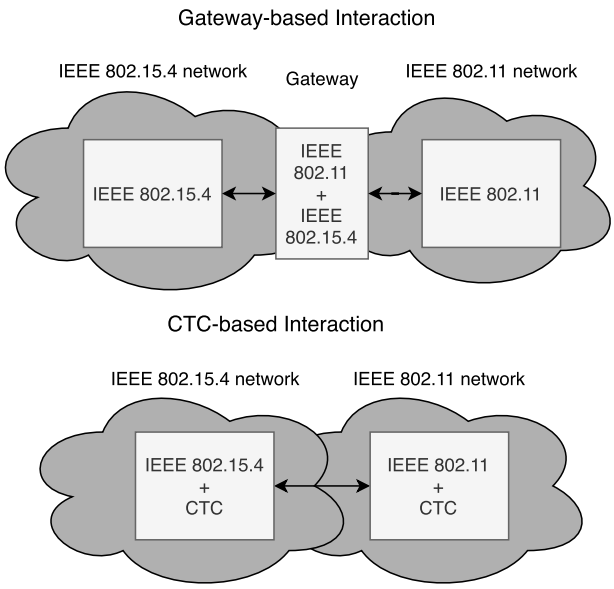

FIGURE 1. Examples of communication scenarios comparing gateway-based interaction with CTC-based interaction.

communicate with other devices. The capability to provide asynchronous interactions between sleepy devices enables WuR to support latency-sensitive communications with greater energy efficiency than traditional duty-cycling energy-saving schemes [7].

Besides, WuR hardware can send binary data. This capability, initially introduced to include addressing data in the WuS, can be repurposed for general-purpose communications. Hence, introducing a secondary communications channel that enables devices to interact, even those using non-compatible main radios. Therefore, we propose WuR assisted CTC (WuR-CTC) to enable non-compatible wireless devices to interoperate.

Instead of standardizing a single WuR solution compatible with all IoT devices, the current industry trend is to develop specific WuR solutions related to existing WPAN and WLAN solutions. The first example of this approach is IEEE 802.11ba [2], which defines a WuR solution exclusively aimed at IEEE 802.11 devices. These specific WuR solutions could become non-compatible with each other. Thus, squandering the opportunity to use WuR-CTC to reduce fragmentation in the IoT. Therefore, we want to highlight the harmonization opportunity that new WuR developments bring to the WPAN and WLAN environment. Fragmentation in the IoT landscape can be greatly reduced by WuR-CTC, but only if the actors in the field harmonize future WuR implementations.

The main contributions of this article are the following:

- Present the WuR-CTC concept to achieve heterogeneous high-rate bidirectional communication by implementing a power-saving mechanism. To the best of the author's knowledge, this proposal is entirely original.
- Specify a WuR-CTC solution featuring a physical and Medium Access Control (MAC) layer design for

resource-constrained devices that allows for reliable point-to-point communications using On-Off Keying (OOK) modulation.
- Design a flexible testbed for the evaluation of WuR-CTC between commodity IEEE 802.11 and IEEE 802.15.4 devices. Communications take place using the main radios of both device types as WuTx paired with prototype WuRx hardware constructed with widely available off-the-shelf components.

The remainder of the article is structured as follows: Section II presents the CTC concept in detail and its state of the art; Section III explains the communication scenario envisioned, as well as the characteristics of the desired WuR-CTC solution; Section IV details the WuR solution adopted and its implementation in the testbed; Section V presents the experimental results obtained from the testbed operation; finally, Section VI concludes the study and points out next research issues.

II. CROSS-TECHNOLOGY COMMUNICATIONS

Two approaches have been previously evaluated in the CTC literature to provide direct communications between non-compatible devices: first, repurposing Carrier Sense Multiple Access (CSMA) mechanisms and, second, signal emulation. In this section, we present the state-of-the-art of each CTC approach, as well as its advantages and disadvantages. Finally, we discuss WuR-CTC in comparison with the aforementioned approaches.

A. CSMA-BASED CTC APPROACH

This approach repurposes the coexistence mechanisms that WPAN and WLAN protocols implement for its use with CTC. The most widely deployed WLAN and WPAN technologies that have appeared in the last decades operate on the 2.4 GHz ISM band [8]–[10] using CSMA with Collision Avoidance (CSMA/CA) as their MAC approach. CSMA-enabled stations avoid collisions by sensing the channel before starting a transmission. Thus, CSMA equipped stations can detect transmission bursts initiated by non-physically compatible devices. Moreover, they can also determine the duration of an incoming transmission burst by measuring its length. Therefore, CSMA/CA-based CTC encodes binary data using signal features that can be received by any CSMA-enabled station, such as the frame length and the inter-frame interval. This approach to CTC was first proposed in [5], which presents a proof-of-concept implementation supporting communications between IEEE 802.11 based transmitters and IEEE 802.15.4 receivers. Reference [5] uses several IEEE 802.11 frame lengths as codewords to build a dictionary for CTC, reaching a throughput of 1.6 kbps in a best-case scenario. Later, the idea of using frames to encode information was extended to code data with the interval between frames [11].

However, low spectral efficiency and throughput limit the usability of CSMA-based CTC. To maximize spectral efficiency, authors in [12] proposed piggybacking CTC communications on existing data fluxes. On that contribution, binary data is modulated by changes in the transmission intervals of frames from pre-existing TCP/UDP fluxes sent by an IEEE 802.11 station. Although [12] still presents a relatively low bitrate, it reuses frames from already present data streams with minimal disruption. Nonetheless, CSMA-based CTC uses entire frames as symbols, only encoding a few bits of information in each of them. As a result, CSMA-based CTC throughput is low when compared to the underlying radio solution. Moreover, the symbols used by CTC (frame lengths, or inter-frame intervals) can be spontaneously generated by non-CTC network activity, causing false detection events. The interference caused by those needs to be mitigated either by using codification or by performing retransmissions, further reducing CSMA-based CTC throughput. In a realistic scenario with the presence of interference, the maximum throughput achieved by CSMA-based CTC is close to 3.1 kbps from BLE to IEEE 802.11 [13].

B. EMULATION-BASED CTC APPROACH

The emulation-based CTC approach can overcome the low throughput associated with CSMA-based CTC. This approach tries to reproduce the signaling defined by a wireless solution with a transmitter implementing another wireless solution. This approach appeared in the literature with WEBee [14], which defines a procedure to emulate IEEE 802.15.4 signals with an Orthogonal Frequency Division Modulation (OFDM) transmitter following the IEEE 802.11 standard. WEBee provides a unidirectional link between a transmitting IEEE 802.11 station and one or more receiving IEEE 802.15.4 stations at IEEE 802.15.4 full data rate, 250 kbps. Nevertheless, inaccuracies on the emulation of IEEE 802.15.4 cause degraded range and a frame error rate close to 40%, which the authors mitigate with a repetition code. Subsequent proposals present solutions that reduce the high frame error rate of WEBee using features introduced in IEEE 802.11ah to improve emulation quality, as well as by introducing channel codification [15], [16]. CTC by emulation has been also introduced to other technologies, including Bluetooth to IEEE 802.15.4 [17] and Bluetooth/IEEE 802.15.4 to LoRa [18]. However, in most implementations emulation is not able to provide a bidirectional link, thus, limiting its usefulness in most communication scenarios. Nonetheless, [19] introduced support for reliable communications by using confirmation messages. In this proposal, the authors use emulation-based CTC to send high rate data streams from IEEE 802.11 to IEEE 802.15.4 using WEBee [14] with low-rate confirmation messages sent from the IEEE 802.15.4 nodes. These confirmation frames are received by the IEEE 802.11 nodes using raw access to the channel samples, a feature which is not accessible in commodity IEEE 802.11 devices. Still, the lack of symmetric bidirectional connectivity relegates emulation-based CTC to one-way communications or data dissemination via broadcast and multicast.

C. WuR-CTC APPROACH

WuR-CTC-enabled devices feature a compatible receiver: the WuRx. As a consequence, in contrast to emulation-based CTC, WuR-CTC supports symmetric bidirectional communications. Moreover, WuR-CTC can provide higher throughput than CSMA-based CTC. However, WuR-CTC requires the addition of at least one hardware element to the device, the WuRx. Despite this, the inclusion of a WuRx facilitates low-power operation, which is a requirement in most IoT application domains.

III. WuR-CTC SCENARIO

The scope of the communication scenario proposed in this article is chosen to demonstrate the utility of WuR-CTC in multiple application domains. In this way, the scenario considered includes:

- Symmetric and reliable point-to-point communications.
- Coexistence with other standards operating at the same frequency band.
- Support for IP-based communications on top of the proposed WuR-CTC solution (e.g., an IP stack with a 6Lo adaptation layer [20]).
- Time-sensitive communications requiring low-latency device interactions.

Moreover, the solution needs to apply to devices that are representative of the current trends in IoT hardware. In this way, prospective WuR-CTC devices are:

- Heterogeneous, operating at the 2.4 GHz ISM band which is supported by the physical layers of most WPAN and WLAN standards.
- Low-power and with a low activity ratio.
- Limited in computational capability, i.e., microcontroller hardware.

Derived from the scenario described above and the device profile we have identified the following requirements for the WuR-CTC solution.

- To reduce the power consumption of the device, the WuRx must be implemented by separate low-power hardware.
- To ensure compatibility with low power WuRx the WuS needs to be coded with OOK, which is used throughout most of the WuR implementations in the literature [6].
- To achieve bidirectional and reliable communications in a point-to-point scenario, the solution needs to provide an Automatic Repeat Request (ARQ) mechanism and two-way addressing.
- A CSMA-CA implementation is required to coexist with other WPAN and WLAN standards operating on the same unlicensed frequency band.
- To provide compatibility with IP communications, the solution needs to provide a payload size that is sufficient to achieve low fragmentation overhead with a prospective 6Lo implementation. Payload size should be close to those defined by other WPAN solutions with existing 6Lo implementations such as IEEE 802.15.4.

IV. IMPLEMENTATION

This section presents a solution for the evaluation of WuR-CTC, as well as its implementation on a testbed. The resulting WuR-CTC solution and the corresponding testbed implementation aim to provide a flexible low-cost platform to evaluate WuR-CTC using commodity hardware. For this purpose, the implementation uses, when possible, existing off-the-shelf components. Moreover, the developed software has been made open-source to facilitate the reproduction and extension of the results presented in this article [21]–[24].

The following subsections detail the wireless devices selected for the implementation of the WuR-CTC testbed, the protocol layers developed for the WuR-CTC solution, and the implementation details of the WuTx and WuRx.

A. WIRELESS DEVICES USED IN THE WUR-CTC TESTBED

This article presents the WuR-CTC concept as standard agnostic. In this way, the WuR-CTC implementation presented here is designed to showcase communications between different WPAN and WLAN solutions by the means of a compatible WuR implementation. However, with no loss of generality, we propose a testbed using devices implementing two wireless communication standards: IEEE 802.11 and IEEE 802.15.4.

Both IEEE 802.15.4 and IEEE 802.11 standards are massively deployed for industrial and domestic use cases, however, they are not compatible. Their physical layers (PHY) are defined with different modulations, channel assignments, physical framing, bandwidth, and transmission rates. Despite these differences, both can operate at the 2.4 GHz ISM frequency band. Commodity devices implementing both of these standards are available for development. Moreover, it is possible to generate OOK signaling at a high symbol rate (250 kBd) with the main radios of devices implementing both standards.

A pseudo-OOK modulation can be generated with IEEE 802.11g/a transmitters by using the technique presented in [25]. Although IEEE 802.11ba also defines a WuTx implementation, the specification is still to be ratified and, at the time of writing of this document, there are no commercially available IEEE 802.11ba embedded devices. Using the legacy compatible IEEE 802.11 WuTx implementation described in [25] enables the evaluation of WuR-CTC with a testbed based on commercially available commodity IEEE 802.11 hardware. However, the chosen IEEE 802.11-based WuTx uses a WuS with a bandwidth of 20 MHz, higher than the 4 MHz MC-OOK defined by the IEEE 802.11ba specification. As a consequence, the WuRx power consumption of our solution will be higher than a prospective IEEE 802.11ba compliant implementation. Nonetheless, our solution has the advantage of being legacy-compatible.

The legacy-compatible IEEE 802.11-based WuTx implementation [25] requires the transmitter PHY to use a predictable scrambler sequence (a condition that occurs in several IEEE 802.11 devices [14], [25]). For this purpose, we

Synchronization Sequence	Frame Delimiter	WuR-CTC PSDU
10 symbols	2 symbols	Variable

FIGURE 2. WuR-CTC PPDU structure.

have selected the ESP-32 (a microcontroller-based System on a Chip (SoC) that includes an IEEE 802.11 transceiver [26]) for the testbed implementation, which fulfills this last requirement. Additionally, the ESP-32 is a low-cost wireless device that fits the profile of low-power and resource-constrained devices defined in Section III.

The generation of OOK is not possible with all standard-compliant IEEE 802.15.4 devices. However, several devices implementing IEEE 802.15.4 include reconfigurable radio hardware that can generate OOK signals while maintaining its operation as a standard-compliant IEEE 802.15.4 device. One of these is the EFR-32MG12 [27], which can transition its main radio between IEEE 802.15.4 and OOK in less than 100 μs. Therefore, this device can operate as a part of an IEEE 802.15.4 network, reconfigure its radio to transmit with OOK, send a frame and, finally, reconfigure its radio back to IEEE 802.15.4. For this reason, we have selected the EFR-32MG12 as the IEEE 802.15.4 device to be used in this testbed.

B. WuR-CTC PHYSICAL LAYER SPECIFICATION

The PHY specification presented in this work considers using the main radio as WuTx and a separate dedicated low-power receiver as WuRx. The advantage of this approach is that it reduces implementation cost, as the only required addition is the WuRx.

The WuR-CTC PHY uses a single transmission rate of 250 kbps with OOK symbols that is supported both in the legacy IEEE 802.11 WuTx [25], and the IEEE 802.15.4-compatible WuTx [28].

The WuR-CTC Physical Protocol Data Unit (PPDU) includes a PHY header that is added before the WuR-CTC Physical Service Data Unit (PSDU) to aid in synchronization and allow the WuRx to retrieve the WuR-CTC PSDU. This header has two components: a preamble sequence and a frame delimiter. The structure of the WuR-CTC PPDU is shown in Fig.2.

The preamble sequence consists of 10 symbols with a "10" pattern (1). Its transitions from high to low can be used to synchronize the WuRx with the symbol period. The preamble also maintains a constant average amplitude that the WuRx uses to calibrate its OOK symbol detection level.

$$\{1, 0, 1, 0, 1, 0, 1, 0, 1, 0\} \tag{1}$$

After it, a frame delimiter field composed of two ON symbols (2) is appended.

$$\{1, 1\} \tag{2}$$

The WuR-CTC solution presented in this work is designed to operate with commodity transmitters. These are not fully reconfigurable since they already implement other wireless solutions. Therefore, the WuR-CTC PHY needs an encapsulation mechanism to embed the WuR-CTC PPDU inside the frame structure followed by the radio operating as WuTx. This encapsulation mechanism must allow any WuRx to retrieve the WuR-CTC PPDU regardless of which WuTx has sent it.

For the testbed, we defined two encapsulation formats: one for IEEE 802.11-based WuTx [25] and another for IEEE 802.15.4-compatible WuTx based on a reconfigurable radio [27].

On an IEEE 802.11-based WuTx the WuR-CTC PPDU is encapsulated in the MAC Service Data Unit (MSDU) of a standard IEEE 802.11g frame. This differs from the mechanism introduced in [25], which describes the encapsulation of the WuR-CTC PPDU directly in the IEEE 802.11g PSDU. This change provides wider compatibility to the method described in [25] since most IEEE 802.11 implementations allow for raw byte access to the MSDU from software. As stated before, WuR-CTC PPDU is coded with OOK symbols at one bit per symbol, for a data rate of 250 kbps. Each of the WuR-CTC symbols is also a standard-compliant IEEE 802.11g OFDM symbol, therefore, the bandwidth used by the IEEE 802.11 WuR-CTC PPDU is 20 MHz [25]. The encapsulation of the WuR-CTC PPDU inside an IEEE 802.11g OFDM frame is represented in Fig.3a. The IEEE 802.11 encapsulation uses the ERP-OFDM PLCP format, which enables frames sent by the IEEE 802.11-based WuTx to coexist with other non-WuR-CTC IEEE 802.11g OFDM stations. Nonetheless, the WuR-CTC IEEE 802.11 encapsulation is not compatible with the backward compatible DSSS-OFDM preambles.

The IEEE 802.15.4-compatible WuTx must configure its reconfigurable radio to send ideal square OOK symbols at 250 kbps. The bandwidth of this OOK signal is slightly lower than 1 MHz at a 20 dB attenuation. To encapsulate the WuR-CTC PPDU, an OOK preamble including 18 OOK symbols with a "10" pattern is prepended. This preamble establishes the OOK symbol detection threshold on the WuRx before the start of the WuR-CTC PPDU. The encapsulation of the WuR-CTC PPDU inside a frame sent by a reconfigurable radio is represented in Fig.3b.

The maximum length that can be carried in the WuR-CTC PSDU is limited by the encapsulation format used. IEEE 802.11-based encapsulation provides the most limiting scenario. As mentioned before, WuR-CTC uses the OFDM symbols of the IEEE 802.11g MSDU as OOK symbols. However, the maximum number of OFDM symbols that the IEEE 802.11g MSDU can carry is limited by its maximum size, which is 2304 in octets of IEEE 802.11g-level data. These octets are mapped in a number of OFDM symbols that depends on the transmission rate used by the IEEE 802.11g radio. Lower transmission rates yield a higher number of available OFDM symbols for encoding WuR-CTC data since

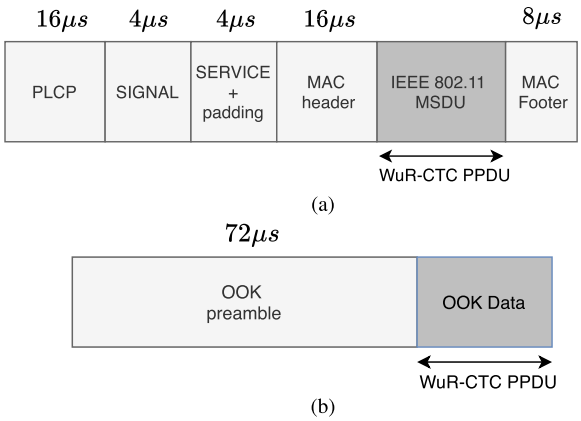

FIGURE 3. WuR-CTC PPDU encapsulation format for each WuTx on the testbed. (a) IEEE 802.11g WuTx. IEEE 802.11 fields length calculated with a 6 Mbps data rate. (b) IEEE 802.15.4 WuTx.

TABLE 1. Maximum WuR-CTC PSDU Size.

Throughput (Mbps)	OFDM symbol loading (bits)	WuR PSDU size (bytes)[a]
6	24	94
9	36	62
12	48	46
18	72	30
24	96	22
36	144	14
48	192	10
54	216	9

[a] Obtained by rounding the number of bytes to the lowest integer.

the number of WLAN level bytes encoded per OFDM symbol is lower. Higher transmission rates yield a lower maximum WuR-CTC PSDU size because the number of OFDM symbols required to encode the 2304 byte maximum IEEE 802.11 MSDU length is lower.

The maximum WuR-CTC PSDU size ($WuR\ CTC_{PSDU}$) can be calculated as a function of the IEEE 802.11 maximum PSDU ($MSDU_{max}$) size in octets, the bits per OFDM symbol ($ODFM_{bits}$), and the overhead bits introduced by the WuR-CTC PPDU headers ($PPDU_{header}$), which corresponds to 12 bits (3).

$$WuR\ CTC_{PSDU} = \frac{ODFM_{bits}}{8 \cdot MSDU_{max}} - PPDU_{header} \tag{3}$$

Table 1 lists the results of (3) for each transmission rate supported by IEEE 802.11g, as well as the bit loading on each OFDM symbol. To maximize the WuR-CTC PSDU size, the testbed uses the 6 Mbps transmission rate, which yields a maximum PSDU size of 94 bytes.

C. WuR-CTC LINK LAYER SPECIFICATION

The WuR-CTC link layer, which defines both MAC and Logical Link Control (LLC) sublayers, provides three services: transmission medium sharing, addressing, and reliability.

TABLE 2. WuR CSMA/CA Parameters for the reconfigurable radio.

CSMA/CA parameter	Value	Unit
Slot Time	20	μs
Minimum Backoff Exponent	4	Slots
Maximum Backoff Exponent[a]	8	Slots
Retries	7	n/a
Detection Threshold[b]	-82	dBm
CCA Interval	14	μs

[a] Reduced from the 10 defined by the IEEE 802.11-2003 spec [29] to 8 due to hardware limitations on the EFR-32MG12 reconfigurable radio [27].
[b] The minimum allowed by EFR-32MG12 reconfigurable radio [27].

In most wireless regulatory domains, devices using the 2.4 GHz band are required to implement strategies to share the medium with other stations. The regulations require either the use of a variant of the CSMA MAC mechanism or to severely limit the radio duty-cycle of the station. WuR-CTC uses CSMA/CA for coexistence, in compliance with most unlicensed band use regulations. However, the implementation of this mechanism differs depending on the technology the device WuTx is based on:

1) On an IEEE 802.11 based WuTx the WuS is encapsulated in the MSDU of an IEEE 802.11g compliant frame, which is sent with a standard complying transmitter. Therefore, the CSMA/CA coexistence mechanisms defined by IEEE 802.11 apply.
2) On the IEEE 802.15.4 compatible WuTx, the frame is sent using the CSMA/CA method provided by the reconfigurable radio of the device [27]. The configurable parameters of the CSMA/CA implementation (see Table 2) are tuned to be as close as possible to those defined by IEEE 802.11 for mixed b/g networks [29] to optimize coexistence with IEEE 802.11 stations.

As a consequence of delays introduced with WuRx interfacing, WuR-CTC implementations send ACK frames with a higher delay than the Short Interframe Space (SIFS) interval defined by the IEEE 802.11 standard. Therefore, the WuR-CTC ACKs do not gain priority over other frames sent by coexisting IEEE 802.11 stations and WuR-CTC ACKs need to contend with IEEE 802.11 traffic. This coexistence issue with IEEE 802.11 stations is present in other wireless solutions such as Bluetooth [10] and IEEE 802.15.4 [9], which also use inter-frame spaces for their confirmation messages that are higher than the SIFS defined by IEEE 802.11 releases.

With the IEEE 802.11-based WuTx, the IEEE 802.11 Network Allocation Vector (NAV) can be used to reserve the channel with enough time to protect the whole transaction, including the expected WuR-CTC ACK. For example, protection could be achieved by using a CTS-to-self frame before starting a WuR-CTC exchange. Nonetheless, this mechanism cannot protect transactions initiated by IEEE 802.15.4 WuTx.

10 bit	10 bit	3 bit	1 bit	8 bit	0 - 89 bytes	8 bit
Receiver Address	Sender Address	Type	Seq. Num.	Length	MPSU	CRC-8

FIGURE 4. WuR-CTC MPDU.

These do not include an IEEE 802.11 compatible radio capable of transmitting the frame fields required to set the NAV of IEEE 802.11 stations.

To provide LLC functionality, WuR-CTC implements a stop-and-wait ARQ mechanism that supports ACK piggybacking and uses 10-bit unicast addresses. The LLC protocol also defines the control messages necessary to awaken and sleep other stations. To support this, the WuR-CTC MAC Protocol Data Unit (MPDU) includes several header fields shown in Fig.4.

The included fields are:

- Receiver Address: A 10-bit address that identifies the destination WuR-CTC node.
- Sender Address: A 10-bit address that identifies the source WuR-CTC node.
- Type: A 3 bit field with the following flags, ordered from the most significant bit to less significant bit, in Big Endian:
  1) Data flag.
  2) ACK flag.
  3) WuS flag.

  Using these flags, the protocol defines 6 types of frames:
  - "100": **DATA frame**. Includes data.
  - "010": **ACK frame**. Acknowledges a previously transmitted frame.
  - "001": **WAKE frame**. Awakens/sleeps other stations.
  - "110": **DATA+ACK frame**. Includes data and acknowledges a previous frame.
  - "101": **WuS+DATA frame**. Includes data for a sleepy station. This frame type does not keep the station awoken after reception.

  The two remaining frame types are not used in the testbed and remain reserved for future use.
- Sequence Number: A 1 bit sequence number. The **ACK frame** sequence number must match the sequence number of the acknowledged frame.
- Length: An 8 bit length indicator. Contains the WuR-CTC MSDU length in bytes. Although, currently, the maximum MSDU length supported is only 89, the length of this field is set to 8 bit due to implementation constraints on the EFR32MG12 reconfigurable radio [27].
- MSDU: The payload, which can range from 0 to 89 bytes. The maximum length corresponds to the maximum allowable WuR-CTC PSDU obtained from Table 1 minus 5 bytes, which account for the length of the LLC header and footer.
- CRC-8: A 8 bit length CRC field. It is calculated on the rest of the MPDU, including headers and payload, and appended at the end of the payload. The CRC-8 polynomial used is (4):

$$p(x) = x^7 + 1 \tag{4}$$

All frame types previously defined, except **ACK frames**, must be acknowledged by the receiver. No other frame can be sent before receiving the corresponding **ACK frame** or the expiration of the reception timeout. Moreover, all frame types must feature all the fields defined for the WuR-CTC MPDU as shown in 4. It is important to notice that, in contrast to other wireless protocols, WuR-CTC **ACK frames** include source and destination addresses, as well as a sequence flag. Thus, WuR-CTC frames can be acknowledged unambiguously without strict timing constraints. This is not the case in other protocols, which omit the sender's address from the ACK frame. As a result, these protocols need to guarantee strict timing constraints to ensure that the acknowledgment frame is sent before any other frame. The SIFS used before an acknowledgment frame in IEEE 802.11 is an example of such a constraint.

For **WAKE frames**, payload data length must be 1 byte, and its value encodes the intent of the frame:

- A payload of **0xFF** is a **WAKE frame**, which indicates that the device should be woken up.
- A payload of 0×00 is a **SLEEP frame**, which indicates that the device needs to be returned to sleep.

Sending a **SLEEP frame** is recommended. However, a previously woken device will return to sleep automatically after a timeout passes.

A typical data exchange between two sleepy WuR-CTC devices consists of a **WAKE frame** to awake the receiver device, one or more **DATA frames**, and, finally, a **SLEEP frame** to indicate the receiver that it can return to sleep. Each one of the aforementioned messages must be acknowledged with an **ACK frame**. This type of exchange is shown in Fig.5 for devices with assigned WuR addresses 0×001 and 0×002 on a particular case featuring a request/response interaction.

However, this is not the minimal exchange bearing data. A station can send data to another with a single **WuS+DATA frame**, without neither a prior **WAKE frame** nor a **SLEEP frame**. This exchange conveys information but does not force the receiver device to remain awake for any given time, as it is the case with an interaction that starts with a **WAKE frame**.

D. WuTx IMPLEMENTATION

The WuTx for the IEEE 802.11g is based on previous work [25]. That article defined a software implementation capable of generating OOK signals using an IEEE 802.11g legacy PHY. The implementation of the OOK WuTx in ESP-32 uses an IEEE 802.11g transmission rate of 6 Mbps to encapsulate the WuR-CTC signal. As can be seen in Table 1, this rate allows the maximum possible PSDU length for IEEE 802.11g WuR-CTC WuTx.

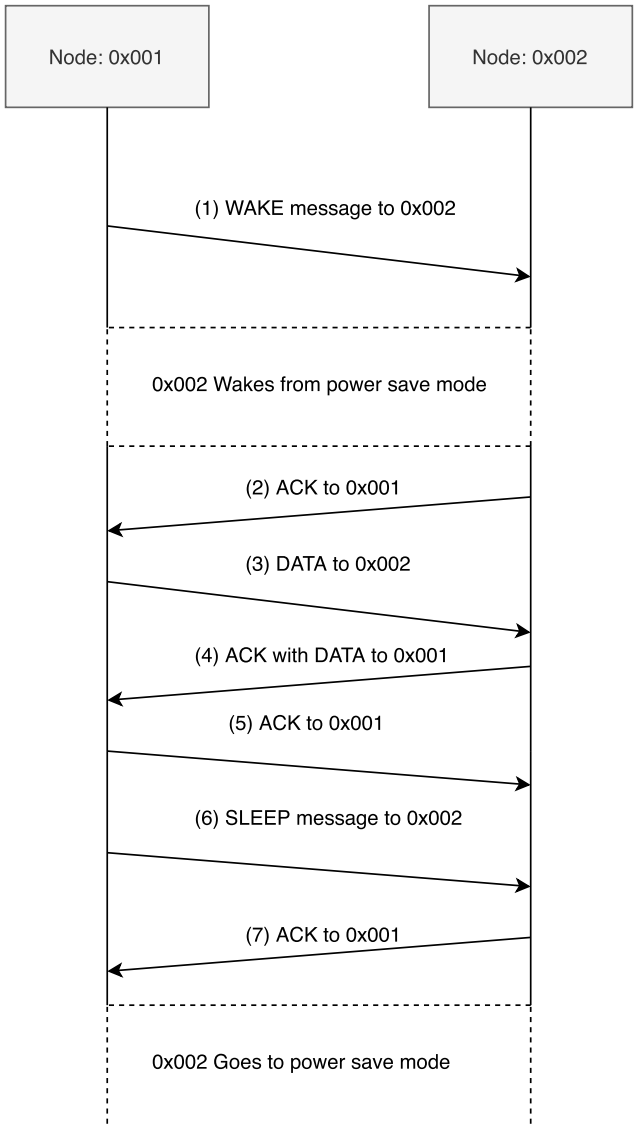

FIGURE 5. Request/response WuR-CTC communications between two sleepy nodes.

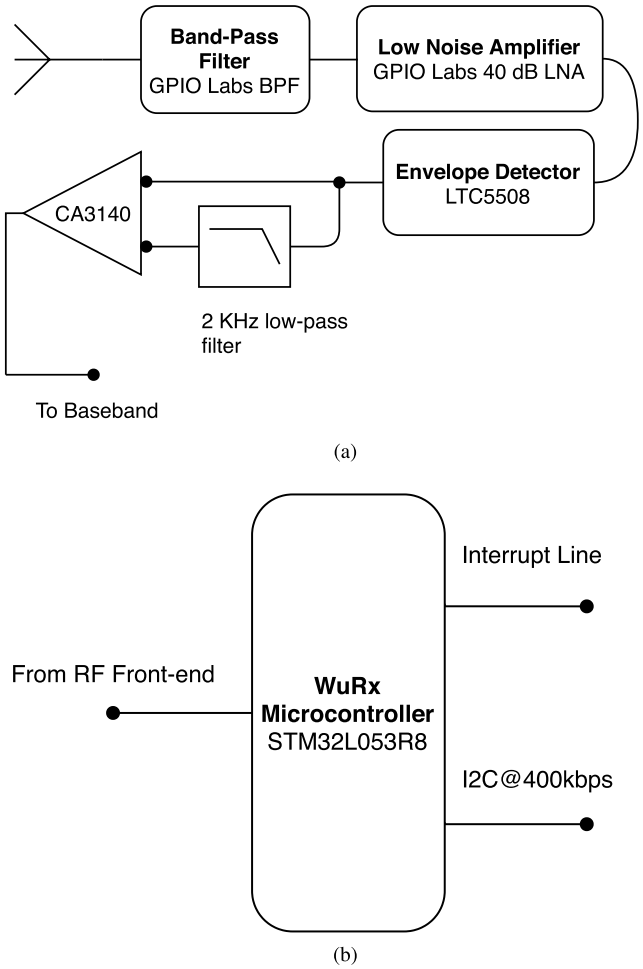

FIGURE 6. Block diagrams of the two WuRx components. (a) Structure of the RF front-end. (b) Structure of the baseband.

The WuTx for IEEE 802.15.4 devices is based on the reconfigurable radio incorporated in the EFR-32 Mighty Gecko (EFR-32MG12) [28], which supports sending OOK symbols and obtain a data rate of 250 kbps. Moreover, it can do so while also acting as a standard-compliant IEEE 802.15.4 radio. In this testbed, the EFR-32MG12 is running a full IEEE 802.15.4 based protocol stack (the Thread stack [30]) concurrently with the WuR-CTC solution proposed in this article.

E. WuRx IMPLEMENTATION

The WuRx is based on the implementation presented in [31]. It is divided into two parts: the RF Front-End that amplifies, demodulates, and normalizes the incoming signal; and the baseband, which processes incoming signals into a binary stream and parses them according to the protocol. The block structure of the WuRx is presented in Fig.6

The RF front-end developed for this testbed is composed of non-low-power, off-the-shelf RF components. Its block structure is shown in Fig. 6a. A monolithic WuRx implementation including the front-end and the baseband using an integrated circuit would be ideal for the reduction in power consumption [32], [33]. Nevertheless, the off-the-shelf approach we have used allows us to provide a testbed for the evaluation of WuR-CTC that is both low-cost and flexible. The main characteristics of the current non-low-power RF front-end are summarized in Table 3. All components, except the LNA, which is USB powered, are powered from the STM32L0R8 3.3V voltage regulator output.

The RF front-end uses one band-pass filter that covers the complete 2.4 GHz ISM band instead of a single IEEE 802.11 channel filter covering 20 MHz. As a result, the RF front-end output is derived from the overlap of all channels operating in the 2.4 GHz band. This design increments the interference received by the WuRx versus a single-channel filter. Nonetheless, the WuRx design remains fit to demonstrate the viability of WuR-CTC. Not selecting a single-channel filter increments the robustness of the results presented, as these include a pessimistic scenario

TABLE 3. Components of the WuRx.

Component	Parameters	Values
LNA	Gain[1]	31 dB
	Noise Factor	2.8 dB
	Supply Voltage	5 V
	Supply Current	200 mA
BFP	Center Frequency	2.45 GHz
	Bandwidth[1]	150 MHz
	Insertion Loss	2 dB
LTC5508	Active Current	550 μA
	Standby Current	2 μA
CA3140	Supply Current[1]	2 mA
STM32L053R8	System Clock	16 MHz
	Active Current	2.82 mA
	Standby Current	410 nA

[1] Obtained through component characterization.

concerning received interference. Moreover, using this type of filter makes the testbed design more accessible to other researchers since, at the moment of writing of this document, single-channel IEEE 802.11 filters are not available off-the-shelf on major component providers. The consequences of the increased interference, both for frame error rate and false wake-ups are studied and discussed in depth in Section V.

The filter is followed by an LNA and an envelope detector to demodulate the incoming signal down to baseband. The envelope detector has a low-pass filter behavior that can be tuned according to the load at its output. The input signals considered are both the 20 MHz bandwidth pseudo-OOK used by the IEEE 802.11 WuTx, and the 1 MHz bandwidth OOK used by the IEEE 802.15.4 compatible WuTx. Therefore, the envelope detector bandwidth cannot be matched specifically to any of the two signals without compromising the reception of the other. Instead, the detector output bandwidth was tuned to 2 MHz, a compromise in reception performance for the two input signal types. Although 2 MHz is lower than the 10 MHz baseband bandwidth of the IEEE 802.11 WuTx signal, in agreement with the WuRx simulation results in [25], we found the receiver to exhibit better performance with lower bandwidth. Moreover, a 2 MHz receiver bandwidth also favors the reception of OOK signals sent by the IEEE 802.15.4 compatible WuTx by matching more closely their 1 MHz bandwidth. Nevertheless, such a design is sub-optimal to a matched filter implementation, such as those that can be implemented in IEEE 802.11ba receivers, which only need to be matched to the 4 MHz bandwidth of the MC-OOK symbols.

To translate the OOK pulses into a binary waveform, the front-end uses a comparator, which matches the instantaneous signal level with a reference. In the prototype, this role is fulfilled by a CA3140 operational amplifier. The reference signal for the comparator is obtained via a passive RC first-order low-pass filter with a cutoff frequency of 2 kHz.

The baseband implementation is shown in Fig. 6b. It uses an STM-32L053R8 as the WuRx microcontroller instead of the PIC-8 used in [31]. With an instruction clock of 16 MHz, this more powerful microcontroller is capable of processing the more complex WuR-CTC protocol. However, its standby power consumption is higher (410 nA with the required peripherals active, compared to the 10 nA of the PIC-8 implementation). Nonetheless, it is still a sub-μA current consumption, fit for low-power operation with a suitable low-power RF front-end. In addition, the active current consumption of the STM-32L053R8 is lower than with the PIC-8 (2.82 mA on the STM-32L053R8 vs 3 mA on the PIC-8).

Analogously to [31], the WuRx microcontroller is woken up from standby mode with a rising edge from the comparator output, i.e., at the start of an incoming frame on the band of interest. The synchronization procedure is not based on an optimal correlation mechanism [34] since such a procedure cannot be implemented on the current WuRx microcontroller. As in the previous work [31], WuRx synchronization is achieved by edge detection. After waking up, the WuRx microcontroller waits for a transition to a low level on the comparator output. If the WuRx does not find any transition to 0 after 40 samples sampled at 1.25 Msps, it discards the event as a false wake-up. After detecting the transition, the WuRx starts sampling preamble symbols at 250 ksps. At this point, if the WuRx finds more than 3 consecutive "0" symbols it discards the frame as a false wake-up. Otherwise, it keeps sampling the preamble until it encounters the frame delimiter. After the delimiter, the WuRx starts to match the address. If it fails to do so, it returns to sleep immediately.

Once a valid address is detected, the rest of the WuR-CTC frame is received and its CRC-8 checksum is calculated. If it matches the checksum received from the frame, the WuRx microcontroller saves the frame and generates a pulse on the interrupt line to wake up the device connected to the WuRx. Otherwise, the frame is silently discarded. Once woken-up with the interrupt line, the connected device will retrieve the frame from the WuRx. The communications between the host device and the WuRx microcontroller occur through an I2C bus at 400 kbps.

To allow high transmission rates while keeping the power consumption of the baseband low, the WuRx microcontroller uses two clock sources. These are switched according to its state:

1) An internal low-power RC oscillator with a clock accuracy of 0.25%. With this clock source, the WuRx can receive a frame of up to 20 bytes without losing synchronization. The internal RC is used for receiving **WuS** and **WuS+DATA frames** when in low-power mode.
2) An external oscillator circuit that incorporates a crystal resonator rated to 30 ppm accuracy. It has a power consumption of up to 100μW. However, with this more

precise clock source, the WuRx can receive **DATA frames** with the maximum length allowed by the WuR-CTC protocol. This oscillator is activated once the device is woken up to activity with a **WAKE frame**. It is deactivated after receiving a **SLEEP frame**.

F. SOFTWARE IMPLEMENTATION OF THE WuR-CTC SOLUTION

Since the only entity capable of interacting with the WuTx is the host device connected to the WuRx, most of the WuR-CTC protocol implementation is found there. Said protocol implementation is programmed in C and strives to be portable, with a low resource footprint. Currently, it supports both the ESP-32 and EFR-32MG12 platform libraries. The implementation can be configured to be OS aware and run efficiently in a separate task or thread. Moreover, it can be executed in an event loop in platforms where no OS is present. To aid in the reproduction of the results and extension of the WuR-CTC solution presented in this article we have published the source code of the WuRx firmware [21], the cross-platform WuR-CTC protocol implementation [24], and the demonstration application code for both the ESP-32 [22] and EFR-32MG12 [23].

V. RESULTS

This section presents the results obtained from the WuR-CTC testbed evaluation. Some of them are obtained directly from the testbed operation while others are obtained through the emulation of network traffic. The results obtained from the testbed include throughput, frame error rate, and latency. These are used to compare the WuR-CTC testbed to other state-of-the-art CTC systems. Although the RF front-end used in this testbed is not low-power, the microcontroller-based baseband is designed to operate in low-power applications. The results obtained from emulation allow us to assess the viability of the baseband of our off-the-shelf WuRx solution in a low-power scenario.

The following subsections detail the setup and results obtained from the WuRx testbed and through the emulation of network traffic.

A. TESTBED RESULTS

The testbed includes a WuRx equipped IEEE 802.11 device, the ESP-32, and a WuRx equipped IEEE 802.15.4 device, the EFR-32MG12. Since the objective of the tests is not to profile the RF front-end of the WuRx, the layout of the devices is set up to not introduce relevant propagation losses. The two WuR enabled devices are separated by 1 meter and placed in an indoor residential environment. A diagram with the testbed layout is shown in Fig.7.

To measure accurately the throughput and frame error rate, each device performs a transmission of 6.4 kbytes spread over 100 frames, each containing 64 bytes of randomly obtained payload on its WuR-CTC MSDU. In addition to the 100 frames bearing data, and their corresponding **ACK frames**, the total transmission time accounts for the **WAKE** and **SLEEP** frames required to interact with the receiving station. This procedure is repeated 25 times for each of the devices in the testbed to average the effect produced by bursts of interference. The throughput results include the effect of frame errors and retransmissions.

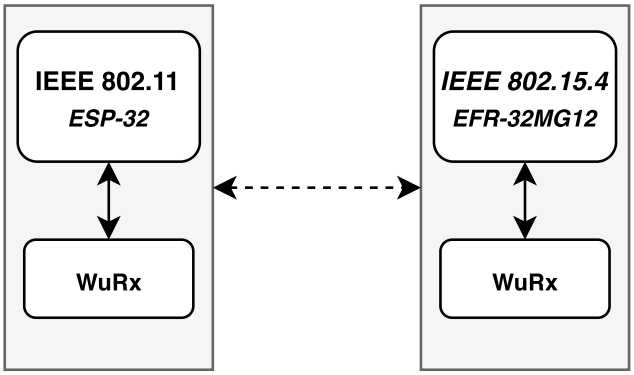

FIGURE 7. Diagram of testbed experimental setup.

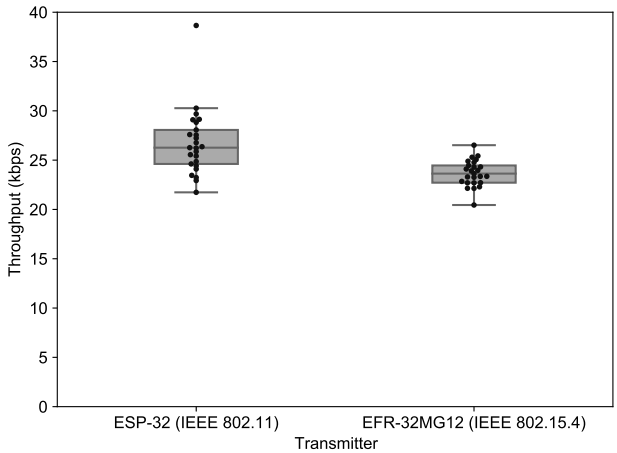

FIGURE 8. Throughput obtained from testbed operation. Results are shown for each of the WuTx.

The throughput results of the test, classified by WuTx device technology, are shown in Fig.8. Fig.9 shows the frame error rate from test sessions. These figures show the data with a box and whisker plot visualization with median and interquartile ranges, with the results from each of the test realization overlayed.

The mean throughput is 26.737 kbps for the ESP-32 (IEEE 802.11) and 23.647 kbps for EFR-32MG12 (IEEE 802.15.4). These differ by roughly 3 kbps. This reduced difference is caused by device-specific frame processing times, including differences in the speed and latency of I2C communications with the WuRx microcontroller.

The WuR-CTC devices achieved a 7-fold throughput improvement over the CSMA-based CTC state of the art (3.1 kbps in [13]). The mean frame error rate observed was 13% and 14.48%, for ESP-32 and EFR-32MG12, respectively. A relatively high frame error rate is expected due to the nature of the experimental setup. The WuRx receives interference over the span of the 2.4 GHz ISM band while the WuTx

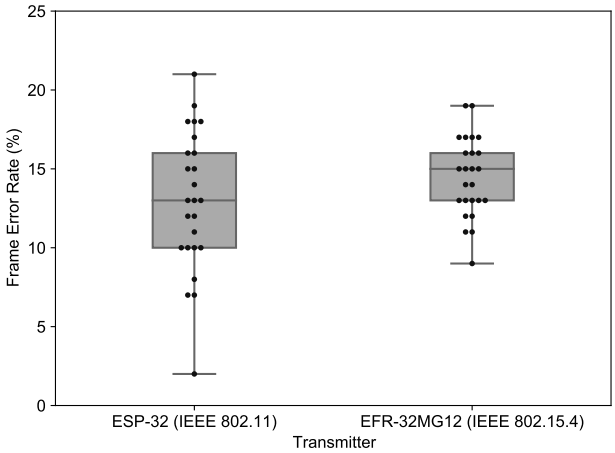

FIGURE 9. Frame error rate obtained from testbed operation. Results are shown for each of the WuTx.

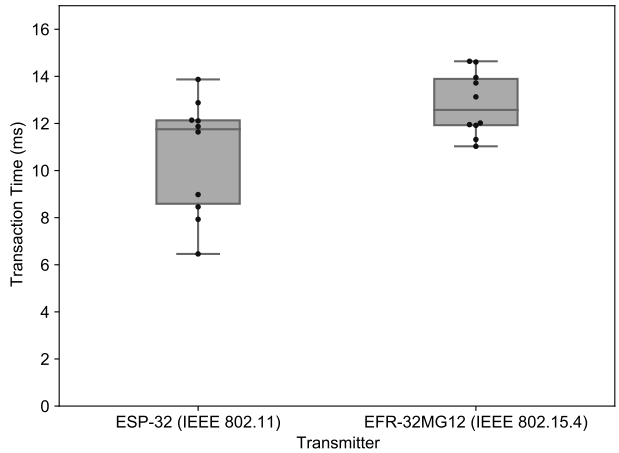

FIGURE 10. Transmission time obtained from testbed operation. Results are shown for each of the WuTx.

only uses CSMA/CA on the channel where the main radio operates. Therefore, the testbed is vulnerable to collisions that will affect the WuRx but that neither the WuTx nor interfering stations can detect. In addition, the current WuRx implementation is more prone to synchronization errors than a WuRx based on correlation. However, the frame error rate measured is a three-fold improvement over emulation-based CTC between IEEE 802.15.4 and IEEE 802.11 [14] (40% in [14]). As a result, the throughput obtained with the WuR-CTC testbed, which includes the loss of throughput due to retransmissions caused by frame errors, more than doubles the maximum throughput found on the literature for emulation-based CTC with reliability (10.67 kbps in [19], also evaluated at 1 meter distance).

The transaction time is measured using a **WAKE+DATA frame** with a 1-byte payload. This is the shortest transaction bearing data that is viable to perform on a sleepy node. The transaction time takes into account the time required to complete the whole transaction, including the generation and transmission of the **WuS+DATA frame** and the reception of the **ACK frame** generated by the receiver device. The test is performed 10 times to reduce the variability in delays. These varying delays are caused by jitter in the host device event processing time and OS introduced delays. The transaction time results obtained are shown in Fig.10 with the same box plot format as the two preceding figures. When the ESP-32 acts as WuTx the transaction is completed in a mean of 10.63 ms. When the EFR-32MG12 acts as the transmitter, the transaction takes a mean of 12.83 ms.

Additionally, we measured the probability of a missed wake-up transaction to characterize the performance of the WuR mechanism implemented for the WuR-CTC testbed. This probability was evaluated using 500 wake-up transactions initiated by each of the WuTx. A transaction was counted as valid only if both, the **WAKE frame** and its respective **ACK frame**, were correctly received. Each wake-up transaction was spaced at least 100 ms to minimize bias introduced by bursts of interference. All **WAKE frames** were received with the WuRx in the low-power state. With the ESP-32 WuTx, the missed wake-up transaction probability is 3.8%, while with the EFR-32MG12 WuTx it is 3.2%. Both rates are lower than the 13,74% frame error rate calculated for the 64 byte data frames evaluated for throughput. Nonetheless, the effect of a missed wake-up transaction is subsequently mitigated by retransmissions, which will occur after the receiving station fails to acknowledge the **WAKE frame** before a 20 millisecond timeout.

B. NETWORK LOAD EMULATION RESULTS

Although the prototype WuRx presented in this testbed has a non-low-power RF front-end, its baseband is designed to operate under restrictive power constraints. The WuRx microcontroller, which implements the WuR-CTC baseband, awakens to decode incoming frames and returns to sleep after processing them. Unfortunately, the WuRx microcontroller needs to wake-up to discriminate WuR-CTC frames from non-WuR-CTC frames. Therefore, non-WuR-CTC network activity increases WuRx power consumption. For this purpose, we measured the length of the WuRx microcontroller wake-ups caused by non-WuR-CTC network frames. Results of the measurements are shown in (5) show the dependence between the length of the non-WuR-CTC frame received and the WuRx wake-up length, according to the preamble sampling mechanism explained in Section IV-E. This mechanism separates (5) into three regions. On the first, the non-WuR-CTC frame ends before the WuRx wakes up and is discarded after sampling three "0" symbols. On the second, the WuRx finds a transition to "0" in the expected range, but, in the same way as before, discards the frame after reading three "0" symbols. On the last, a transition to "0" and the comparator output remains high, causing the WuRx to discard the frame. Fig.11, shows graphically how the WuRx reacts to non-WuR frames according to (5).

$$f(Frm_{len}) = \begin{cases} 64\mu s & : Frm_{len} < 42\mu s \\ Frm_{len} + 22\mu s & : 42 \leq Frm_{len} < 74\mu s \\ 78\mu s & : Frm_{len} \geq 74\mu s \end{cases} \quad (5)$$

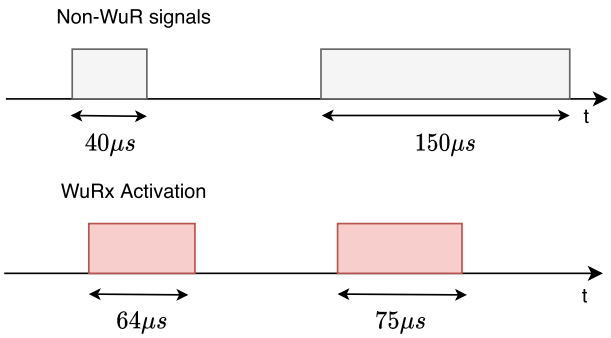

FIGURE 11. Example of non-WuR signals and the corresponding WuRx activation length according to (5).

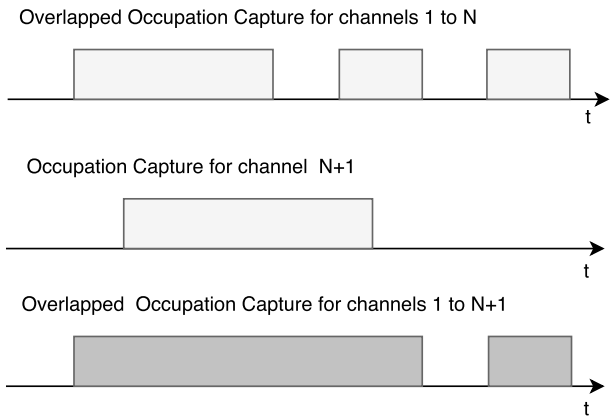

FIGURE 12. An iteration of the channel overlapping process.

To help quantify the activity of the WuRx microcontroller, we define the WuRx activity rate, which is the time fraction that the WuRx microcontroller spends on a high power state. The lower the WuRx activity rate is, the longer the battery life of a WuR-CTC device will be. This figure resembles the duty-cycle used in traditional synchronous power-saving mechanisms. To assess the viability of the baseband, we measure the WuRx activity rate under two different emulated scenarios of non-WuR interference. The first scenario models a residential scenario. A location with a low IEEE 802.11 network activity coming from a limited number of devices. The second scenario models a crowded location with a public WLAN and, consequently, features high network activity coming from a considerable number of devices.

As presented in Section IV-E and Table 3, the WuRx uses a band-pass filter covering the entire 2.4 GHz ISM band. The filter output is then fed to an envelope detector. As a consequence, the signal that reaches the WuRx microcontroller contains a binary version of the IEEE 802.11 network activity aggregated from all the 2.4 GHz band. This signal reaches a high level when a frame is being transmitted in one or more channels and a low level when no transmission is occurring in any of them. To the author's best knowledge, there is no public dataset describing such a signal generated by IEEE 802.11 network activity at the time of writing of this document.

As a solution, we use one or more single-channel captures of IEEE 802.11 frames to generate an overlapped occupation capture, which emulates the signal received by the WuRx microcontroller. This capture is derived from overlapping IEEE 802.11 frame sequences at the same time axis. These sequences are obtained from one or more IEEE 802.11 network captures. The overlapping process is the following: First, an occupation capture is derived from each source capture. The capture is generated by registering the intervals when a frame transmission occupies the channel. Second, the occupation captures are overlapped in an iterative process, as shown in Fig.12 to produce the final overlapped occupation capture; Last of all, the length of the intervals of the overlapped occupation capture is used to calculate the activity rate of the WuRx, according to (5).

Using IEEE 802.11 frames provides a more pessimistic scenario for the emulation of the WuRx activity rate than using other standards operating on the 2.4 GHz ISM band. IEEE 802.11 uses shorter frames than other prominent technologies operating in the same band (IEEE 802.15.4 [9] or Bluetooth [10]). This leads to a higher proportion of WuRx wake-up time for each frame and an increased WuRx activity rate. Moreover, the method used to overlap the occupation captures overestimates the time that the channel will be seen as occupied by the WuRx since it assumes that all frame interference events generate constructive interference that can be detected by the WuRx and trigger a wake-up event.

For the low traffic residential scenario, the overlapped occupation capture is generated by using as source data eleven captures registered ad-hoc. These captures were taken with a commodity IEEE 802.11bgn WLAN card on a Linux PC in a single residential location from all IEEE 802.11 channels where activity was detected in the study period. The use of captures from different channels reflects the effect of different channel traffic profiles on the aggregated network activity. To construct the overlapped occupation capture for the high network activity scenario, we used as source data a capture obtained from the VWave dataset [35] scenario "Pioneer". This capture features traffic obtained from a public WLAN network with a high number of devices. However, it does only cover a single channel. Therefore, in this case, the original "Pioneer" capture is separated into eleven equal length consecutive segments, which are used to generate the overlapped occupation capture.

We use the overall occupancy rate of the channel, defined as the rate of time that the channel remains occupied, to relate the network activity with the WuRx activity. As can be observed in Fig.13, the final activity rate for the WuRx is lower than the occupancy rate of the channel in all scenarios. In the high traffic scenario, the WuRx activity rate is 2.31%, while the channel occupancy rate is 19.33%. In the low traffic scenario, the activity rate is only 0.24%, while the channel occupancy rate is 1.17%. The ratio of WuRx activity compared to the overall channel occupancy rate is 0.21 in the low traffic scenario and 0.12 in the high traffic scenario.

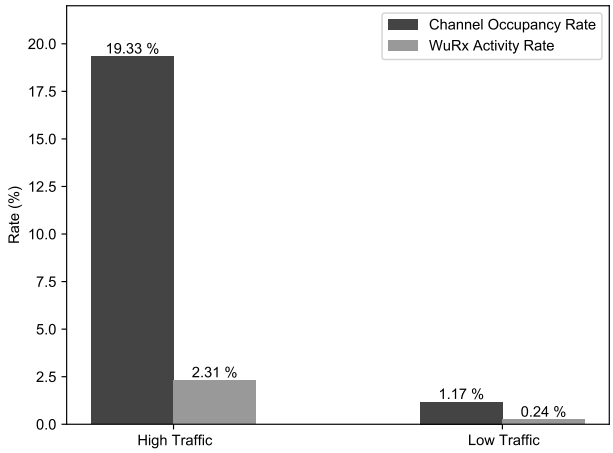

FIGURE 13. Channel occupancy rate and WuRx activity rate for both scenarios.

This reflects the influence of the distribution of frame lengths received.

With these activity rates, the WuRx baseband consumes an average of 65.54 μA and 7.18 μA in the high and low traffic scenarios, respectively. These results support continuing the development of an off-the-shelf RF front-end to use with the current WuRx baseband. This combination would provide an accessible solution to evaluate WuR-CTC in its main application field: energy-restricted wireless networks.

Scenarios with even higher sustained traffic and, therefore, higher WuRx activity rate than the high traffic scenario emulated here are plausible. However, these scenarios take place in locations with a high device density. To support the high density of devices, such locations usually need to have a reliable power source available. As a result, these scenarios fall outside of the application scope of low-power technologies.

VI. CONCLUSION AND FUTURE WORK

In this article, we have proposed WuR-CTC, a solution that helps to reduce IoT fragmentation by providing direct, reliable, and bidirectional communication between non-compatible devices. With WuR-CTC, we have demonstrated that WuR is not only a power-saving mechanism but a contribution to the CTC landscape. WuR-CTC defines a high-throughput CTC solution that can be added to existing WPAN and WLAN devices. To demonstrate the viability of WuR-CTC, we have developed a WuR-CTC solution and implemented it on a testbed. The devices on the testbed were capable of reliable and fully bidirectional CTC. Moreover, the testbed devices have shown improvements in throughput over the current CTC state of the art. Finally, we have shown that the off-the-shelf approach developed in the testbed can operate in power restricted environments, even those with high network activity. Although the WuR-CTC implementation presented here might not be optimal, it serves in demonstrating the viability of the WuR-CTC concept. In future work, WuR-CTC could be implemented using more performant WuR hardware solutions, such as those being developed for IEEE 802.11ba. Regardless of this, the results obtained with the WuR-CTC testbed highlight what can be achieved by harmonizing the nascent WuR standards: enabling bidirectional communications between heterogeneous devices in a transparent, and low-power manner.

In future work, we will concentrate on adding a low-power off-the-shelf RF front-end to the WuRx testbed presented in this work and evaluate its operation in a power restricted environment. In parallel, we will work on extending the WuR-CTC paradigm to more WLAN and WPAN solutions and in developing and evaluating IP-based communications with WuR-CTC using a 6Lo adaptation layer.

REFERENCES

[1] *Eclipse IoT Developer Survey*. Accessed: Jun. 30, 2020. [Online]. Available: https://outreach.eclipse.foundation/download-the-eclipse-iot-developer-survey-results

[2] *IEEE Draft Standard for Information Technology–Telecommunications and Information Exchange Between Systems Local and Metropolitan Area Networks–Specific Requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 3: Wake-Up Radio Operation*, IEEE Standard P802.11ba/D7.0, Sep. 2020, pp. 1–189.

[3] D. Bankov, E. Khorov, A. Lyakhov, and E. Stepanova, "IEEE 802.11ba—Extremely low power Wi-Fi for massive Internet of Things—Challenges, open issues, performance evaluation," in *Proc. IEEE Int. Black Sea Conf. Commun. Netw. (BlackSeaCom)*, Jun. 2019, pp. 1–5.

[4] T. Zachariah, N. Klugman, B. Campbell, J. Adkins, N. Jackson, and P. Dutta, "The Internet of Things has a gateway problem," in *Proc. 16th Int. Workshop Mobile Comput. Syst. Appl.*, Feb. 2015, pp. 27–32.

[5] K. Chebrolu and A. Dhekne, "Esense: Communication through energy sensing," in *Proc. 15th Annu. Int. Conf. Mobile Comput. Netw. (MobiCom)*, 2009, pp. 85–96.

[6] R. Piyare, A. L. Murphy, C. Kiraly, P. Tosato, and D. Brunelli, "Ultra low power wake-up radios: A hardware and networking survey," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2117–2157, 4th Quart., 2017.

[7] J. Oller, I. Demirkol, J. Casademont, J. Paradells, G. U. Gamm, and L. Reindl, "Has time come to switch from duty-cycled MAC protocols to wake-up radio for wireless sensor networks?" *IEEE/ACM Trans. Netw.*, vol. 24, no. 2, pp. 674–687, Apr. 2016.

[8] *IEEE Standard for Information Technology—Telecommunications and Information Exchange Between Systems Local and Metropolitan Area Networks—Specific Requirements—Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE Standard 802.11-2016 (Revision IEEE Std 802.11-2012), Dec. 2016, pp. 1–3534.

[9] *IEEE Standard for Low-Rate Wireless Networks*, IEEE Standard 802.15.4-2015 (Revision IEEE Std 802.15.4-2011), Apr. 2016, pp. 1–709.

[10] Bluetooth SIG. (2019). *The Bluetooth Core Specification*. Accessed: Jul. 1, 2020. [Online]. Available: https://www.bluetooth.com/specifications/bluetooth-core-specification/

[11] S. M. Kim and T. He, "FreeBee: Cross-technology communication via free side-channel," in *Proc. 21st Annu. Int. Conf. Mobile Comput. Netw.*, Sep. 2015, pp. 317–330.

[12] W. Jiang, Z. Yin, S. M. Kim, and T. He, "Transparent cross-technology communication over data traffic," in *Proc. IEEE INFOCOM-IEEE Conf. Comput. Commun.*, May 2017, pp. 1–9.

[13] Z. Chi, Y. Li, H. Sun, Y. Yao, and T. Zhu, "Concurrent cross-technology communication among heterogeneous IoT devices," *IEEE/ACM Trans. Netw.*, vol. 27, no. 3, pp. 932–947, Jun. 2019.

[14] Z. Li and T. He, "WEBee: Physical-layer cross-technology communication via emulation," in *Proc. 23rd Annu. Int. Conf. Mobile Comput. Netw.*, Oct. 2017, pp. 2–14.

[15] Z. Li and T. He, "LongBee: Enabling long-range cross-technology communication," in *Proc. IEEE INFOCOM-IEEE Conf. Comput. Commun.*, Apr. 2018, pp. 162–170.

[16] Y. Chen, Z. Li, and T. He, "TwinBee: Reliable physical-layer cross-technology communication with symbol-level coding," in *Proc. IEEE INFOCOM-IEEE Conf. Comput. Commun.*, Apr. 2018, pp. 153–161.

[17] W. Jiang, Z. Yin, R. Liu, Z. Li, S. M. Kim, and T. He, "Bluebee: A 10,000 x faster cross-technology communication via phy emulation," in *Proc. 15th ACM Conf. Embedded Netw. Sensor Syst.*, Nov. 2017, pp. 1–13.

[18] Z. Li and Y. Chen, "Achieving universal low-power wide-area networks on existing wireless devices," in *Proc. IEEE 27th Int. Conf. Netw. Protocols (ICNP)*, Oct. 2019, pp. 1–11.

[19] S. Wang, Z. Yin, S. Wang, Y. Chen, Z. Li, S. M. Kim, and T. He, "Networking support for bidirectional cross-technology communication," *IEEE Trans. Mobile Comput.*, vol. 20, no. 1, pp. 204–216, Jan. 2021.

[20] C. Gomez, J. Paradells, C. Bormann, and J. Crowcroft, "From 6LoWPAN to 6Lo: Expanding the universe of IPv6-supported technologies for the Internet of Things," *IEEE Commun. Mag.*, vol. 55, no. 12, pp. 148–155, Dec. 2017.

[21] M. Cervià-Caballé. (2020). *WuR-CTC WuRx Firmware for the STM32L0R8*. [Online]. Available: https://github.com/marticervia/WuR-CTC-WuRx

[22] M. Cervià-Caballé. (2020). *WuR-CTC Demo Application for the ESP-32*. [Online]. Available: https://github.com/marticervia/WuR-CTC-Demo-ESP32

[23] M. Cervià-Caballé. (2020). *WuR-CTC Demo Application for the EFR32MG12*. [Online]. Available: https://github.com/marticervia/WuR-CTC-Demo-EFR32MG12

[24] M. Cervià-Caballé. (2020). *Multiplatform WuR-CTC Implementation*. [Online]. Available: https://github.com/marticervia/WuR-CTC-Library

[25] M. C. Caballe, A. C. Auge, E. Lopez-Aguilera, E. Garcia-Villegas, I. Demirkol, and J. P. Aspas, "An alternative to IEEE 802.11ba: Wake-up radio with legacy IEEE 802.11 transmitters," *IEEE Access*, vol. 7, pp. 48068–48086, 2019.

[26] Espressif. (2020). *ESP-32 Datasheet*. Accessed: Jul. 1, 2020. [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf

[27] (2018). *AN971: EFR32 Radio Configurator Guide 0.8*. Accessed: Jul. 1, 2020. [Online]. Available: https://www.silabs.com/documents/public/application-notes/an971-efr32-radio-configurator-guide.pdf

[28] (2018). *Silicon labs EFR32MG1X Datasheet*. Accessed: Jul. 1, 2020. [Online]. Available: https://www.silabs.com/documents/public/data-sheets/EFR32MG1-SF-DataSheet.pdf

[29] *IEEE Standard for Information Technology—Local and Metropolitan Area Networks—Specific Requirements–Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Further Higher Data Rate Extension in the 2.4 GHz Band*, IEEE Standard 802.11g-2003 (Amendment to IEEE Std 802.11, 1999 Edn. (Reaff 2003) as amended by IEEE Stds 802.11a-1999, 802.11b-1999, 802.11b-1999/Cor 1-2001, 802.11d-2001), Jun. 2003, pp. 1–104.

[30] T. Group. *Thread White Paper*. Accessed: Jul. 1, 2020. [Online]. Available: https://www.threadgroup.org/Portals/0/documents/support/ThreadOverview_633_2.pdf

[31] E. Lopez-Aguilera, M. Hussein, M. Cervia, J. Paradells, and A. Calveras, "Design and implementation of a wake-up radio receiver for fast 250 Kb/s bit rate," *IEEE Wireless Commun. Lett.*, vol. 8, no. 6, pp. 1537–1540, Dec. 2019.

[32] E. Alpman, A. Khairi, R. Dorrance, M. Park, V. S. Somayazulu, J. R. Foerster, A. Ravi, J. Paramesh, and S. Pellerano, "802.11g/n compliant fully integrated wake-up receiver with −72-dBm sensitivity in 14-nm FinFET CMOS," *IEEE J. Solid-State Circuits*, vol. 53, no. 5, pp. 1411–1422, May 2018.

[33] J. Im, J. Breiholz, S. Li, B. Calhoun, and D. D. Wenzloff, "A fully integrated 0.2 V 802.11ba wake-up receiver with −91.5 dBm sensitivity," in *Proc. IEEE Radio Freq. Integr. Circuits Symp. (RFIC)*, Aug. 2020, pp. 339–342.

[34] D.-J. Deng, S.-Y. Lien, C.-C. Lin, M. Gan, and H.-C. Chen, "IEEE 802.11ba wake-up radio: Performance evaluation and practical designs," *IEEE Access*, vol. 8, pp. 141547–141557, 2020.

[35] C. Phillips and S. Singh. (Jul. 2009). *CRAWDAD Dataset Pdx/Vwave (V. 2009-07-04)*. [Online]. Available: https://crawdad.org/pdx/vwave/20090704

MARTÍ CERVIÀ CABALLÉ was born in Manresa, Barcelona, Spain, in 1992. He received the B.S. degree in telecommunications engineering and the M.Sc. degree in telecommunication engineering from the Universitat Politècnica de Catalunya (UPC), in 2015 and 2017, respectively, where he is currently pursuing the Ph.D. degree in network engineering. His research interests include wireless sensor networks, digital communications, and low-power transceiver design.

ANNA CALVERAS AUGÉ was born in Barcelona, Spain, in 1969. She received the Ph.D. degree in telecommunications engineering from the Universitat Politècnica de Catalunya, Spain, in 2000. She is currently an Associate Professor with the Wireless Networks Group (WNG), Department of Network Engineering, Universitat Politècnica de Catalunya. She has been involved in several national and international research or technology transfer projects. She has published in international and national conferences and journals. Her research interests and expertise areas comprise the design, evaluation, and optimization of communications protocols and architectures for cellular, wireless multihop networks, ad hoc networks, wireless sensor networks, the Internet of Things, and application domains, such as smart cities, building automation, and emergency environments.

JOSEP PARADELLS ASPAS is currently the Director of the i2CAT Foundation and a Professor with the Universitat Politècnica de Catalunya. He is also the Co-Director of the master program in the Internet of Things (IoT) offered by Fundació UPC. He has been working in the area of the Internet of Things, in particular with wireless interfaces and protocols. He has participated in national and European public-funded research projects and collaborated with several Spanish telecommunications companies. In 2012, he was recognized as the best academic trajectory.