This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3067379, IEEE Internet of Things Journal

Towards A Low-cost Software-Defined UHF RFID System for Distributed Parallel Sensing

Yanwen Wang, *Member, IEEE*, Jiannong Cao, *Fellow, IEEE*, Yuanqing Zheng, *Member, IEEE*

***Abstract*—This paper presents the design and implementation of a low-cost software-defined RFID system for distributed parallel sensing. We aim to implement essential sensing functionalities with low-cost commodity radio components and provide full access to physical layer raw data (*e.g.*, PHY samples of backscatter signals) to enable various RFID sensing applications at low implementation cost. The proposed solution is built in a distributed way where the functionalities of interrogating RFID tags and receiving their backscattered signals are separated into two modules, which naturally supports distributed parallel sensing. A set of innovative techniques are developed, *e.g.*, packet-in-packet communication, carrier frequency offset cancellation, self-interference cancellation, *etc*. to address a range of practical challenges including RFID command generation with cross technology communication, real-time correction of carrier frequency offset, *etc*. We present three case studies enabled by the proposed solution, which demonstrates its applicability and potential of boosting RFID sensing research by substantially cutting the implementation cost of software defined RFID sensing system.**

***Index Terms*—RFID, Software Defined Reader, Parallel Sensing, EPC standard communication**

I. Introduction

Radio Frequency IDentification (RFID) has been widely applied to general inventory applications in logistics and stock management. Recently, RFID has been studied intensively to enable a range of RF sensing applications including object tracking. [1–3], health care [4–8], and human computer interaction [9, 10]. To fulfil sensing tasks, several readers need to be deployed at different positions to interrogate a target tag and collect backscattered signals as illustrated in Fig.1. To avoid communication collisions at the tag, the four readers interrogate the tag in a round-robin manner. We notice that the interrogation time increases with the number of readers. Besides, in order to improve the sensing performance (*e.g.*, localization accuracy), each reader needs to read the tag multiple times so as to filter out noise.

Distributed parallel sensing offers an opportunity to fundamentally boost the sensing efficiency of RFID systems. As illustrated in Fig.2, a tag is interrogated by one transmitter and its backscattered signal can be received by multiple distributed receivers simultaneously. In contrast to the round-robin mechanism, the tag backscatters to multiple readers with one backscatter operation in one-time slot. As the backscatter signal propagates to the receivers over different paths in the air, the receivers obtain distinct channel measurements, each of which independently characterizes the channel between the tag to one of the receivers. In addition, distributed parallel sensing enables users to flexibly add more receivers at different locations and receive signals in the same time slot.

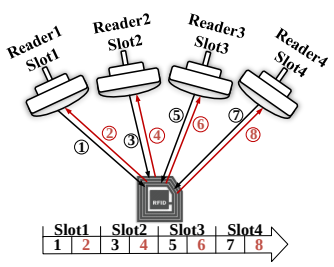


Fig. 1. Sequential interrogation.

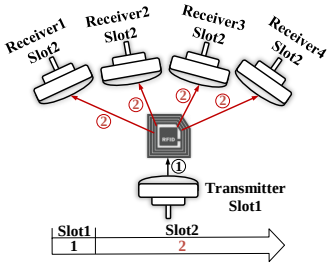


Fig. 2. Parallel interrogation.

Commercial readers available on the market (*e.g.*, Impinj R series, Alien ALR series, *etc*.), however, do not support such distributed parallel sensing due to various technical challenges including frequency synchronization between the transmitter and receivers. Instead, a commercial RFID reader typically adopts the mono-static architecture where a full-duplex transceiver is used to transmit continuous waves and meanwhile receive backscatter signal. When a reader (*e.g.*, Reader 1 in Fig.2) reads a tag in a time slot, other readers (*e.g.*, Reader 2-4) need to keep silent to avoid communication collisions. Previous work has discussed and implemented a RFID system in a bistatic configuration [11]. However, it still uses high cost commercial RFID readers to communicate with tags. Moreover, commercial readers do not provide access to raw physical layer samples. As such, physical layer RFID sensing systems resort to high-end software-defined radios such as USRP N210 to collect raw physical layer samples, which incurs high implementation cost.

The distributed architecture design, being effective to decouple functionalities and enable parallel sensing, comes at the cost of increased technical challenges. First, it is challenging to accurately measure backscatter channels between a tag to RFID receivers, because the transmitter and the receivers are separately operated in the distributed architecture. As a result, the carrier frequency offset (CFO) between the transmitter and the receivers are unavoidable and can drastically affect channel measurements. Existing CFO compensation methods can be applied to mitigate the problem, but residual errors will lead to accumulated errors in phase measurements.

Y. Wang is with College of Electrical and Information Engineering, Hunan University, Changsha 410082, China, and also with Department of Computing, The Hong Kong Polytechnic University, Hong Kong, China. E-mail: wangyw@hnu.edu.cn.

J. Cao and Y. Zheng are with Department of Computing, The Hong Kong Polytechnic University, Hong Kong. E-mail: jiannong.cao@polyu.edu.hk, csyqzheng@comp.polyu.edu.hk. (Y. Zheng is the corresponding author.)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3067379, IEEE Internet of Things Journal

Second, it is challenging to use low-cost general-purpose transmitters to send protocol-compatible RFID commends and continuous waves to RFID tags. Note that the low-cost general-purpose transmitters are not designed for RFID communication, meaning that the packet of general-purpose transmitters cannot be read by RFID tags.

We present a systematic design and implementation of our solution to address the above challenges with low-cost components applied in both transmitter and receiver, which substantially reduces the system implementation cost. First, we propose an online CFO cancellation technique to address carrier frequency mismatch between the transmitter and the receivers. The online CFO cancellation does not require any communication between the transmitter and any receivers. Unlike existing CFO compensation methods, our online CFO cancellation method can track CFO for each tag response and accurately measure physical layer information of backscatter signals (*e.g.*, phase). Second, to allow RFID tags to read the packets (carrying RFID commends) transmitted by a low-cost general-purpose transmitter, we propose a packet-in-packet technique that allows a transmitter to send protocol-compatible commands to RFID tags. We note no modification to RFID tags is needed for the tags to read the packets. Finally, we implement a software-defined UHF RFID system that enables distributed parallel sensing and provides full access to physical layer raw samples for various applications. In implementation, an RFID transmitter is implemented with a low-cost send-only module, which sends EPC C1G2 compatible commands and continuous radio waves to energize commodity passive RFID tags. RFID receivers are implemented with low-cost receive-only software-defined radios (SDRs), which receive the response of tags and collects raw physical layer (PHY) samples. As the system is software-defined, new functionalities can be flexibly added if necessary.

We conduct comprehensive evaluation and test the performance of our system. The experiment results show that the system can work with commodity RFID tags and accurately measure PHY information of the backscattered signal. We then present three case studies for RFID sensing applications applying our RFID solution.

In sum, our work makes the following contributions:

- We present a systematic design of a low-cost RFID system that enables parallel sensing. The proposed system exploits cost-effective and fully programmable commodity radio components, which are built in a distributed architecture, while is fully compatible with the standard EPC communication protocol.
- We develop a set of innovative techniques to address a series of practical challenges, *e.g.*, cross-technology communication, self-interference, carrier frequency offset, to extract low-level data and fulfil parallel sensing tasks.
- We implement a prototype of our low-cost system and conduct extensive experiments as well as three case studies to show the applicability of our design. The results of experiments and case studies show that our low-cost system can accurately measure the low-level data compared with the COTS RFID reader and can achieve parallel sensing in a flexible manner.

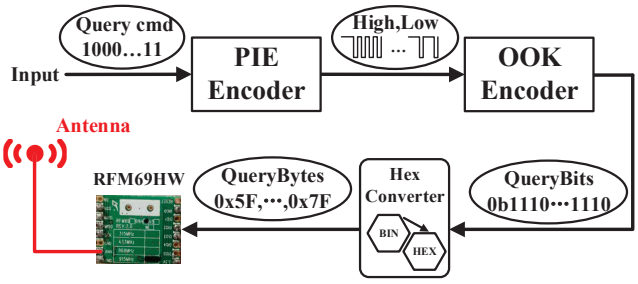


Fig. 3. Overview of transmitter design.

II. Background and Problem Statement

RFID Identification. The communication between a COTS reader and tags consists of multiple inventory rounds: (1) the reader initiates an inventory round by sending a Query command to tags. The Query command specifies several important communication parameters including the number of slots, data encoding scheme (FM0 or Miller) and backscatter link frequency. The Query command is encoded by the reader using PIE (pulse interval encoding) scheme. (2) Receiving the Query command, a tag selects a slot and responds a 16-bit random number (*i.e.*, RN16). (3) Receiving an RN16, the reader sends an acknowledgement (*i.e.*, ACK). (4) Finally, the tag responds its tag ID, which can be used to identify tag-labelled objects. COTS RFID reader and tag strictly follow the *de facto* EPC C1G2 RFID protocol.

RFID Sensing. Different from RFID identification whose purpose is to use EPC information (*i.e.*, tag ID) to identify objects, RFID sensing aims to fulfil various sensing tasks including localization, stock management, gesture sensing, health monitoring and *etc*. The intuition is that objects in the environment may impact the backscattered signals of the RFID system, which causes changes in the backscatter channel measurements (also known as low-level data, *e.g.*, phase, RSSI, and Doppler frequency shift). By extracting unique patterns from the corresponding channel measurements, one can infer different sensing tasks. COTS RFID systems (e.g., Impinj, Alien) are able to provide interface and output such channel measurements from the physical layer, which facilitates the implementation of a variety of applications.

Problem Statement This paper aims to design and implement a low-cost software-defined UHF RFID system that can enable distributed parallel RFID sensing by providing full access to raw physical layer samples of backscatter signals. We aim to reduce the implementation cost so that the system can be used to develop various RFID sensing applications at low implementation cost, thereby boosting education and research in the field of RFID sensing and their applications.

III. System Design and Implementation

The proposed distributed sensing system decouples the functionality of an RFID reader into two functional com-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3067379, IEEE Internet of Things Journal

TABLE I
QUERY COMMAND

	Cmd	DR	M	TRext	Sel	Sess	T	Q	CRC
# of bits	4	1	2	1	2	2	1	4	5
Data	1000	0:DR=8 1:DR=$\frac{64}{3}$	00:M=1 01:M=2 10:M=3 11:M=4	0:No PT 1:Use PT	00:All 01:All 10:∼SL 11:SL	00:S0 01:S1 10:S2 11:S3	0:A 1:B	0-15	CRC

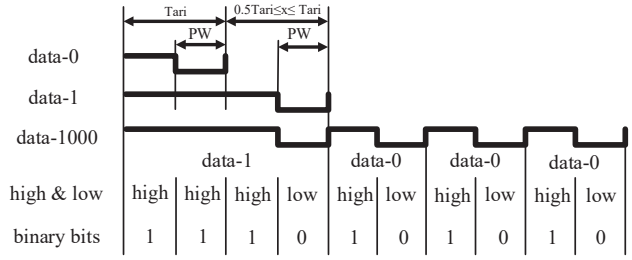


Fig. 4. Data encoding process.

ponents, *i.e.*, RFID transmitter (§III-A) and RFID receiver (§III-B).

A. RFID transmitter design

In order to use billions of commodity RFID tags deployed in the field, the RFID transmitter should be fully compatible with the RFID protocol. We start with an overview of RFID transmitter design, as illustrated in Fig.3. To be compatible with RFID protocol, the Query command defined in C1G1 Gen2 protocol is firstly encoded to the high and low values using Pulse-interval encoding (PIE) scheme. The OOK encoder then translates the high and low values to the binary QueryBits. In the following, we will describe the implementation in detail.

1) A reader-to-tag command: To initiate the communication and collect backscatter signal from a tag, the reader needs to send a protocol-compatible command to the tag. Without loss of generality, we will focus on how to initiate the backscatter communication with a Query command. The Query command defined in EPC C1G2 protocol is shown in Table I. A Query command consists of 22 bits in total, which starts with a 4-bit command code of `1000`, followed by 1-bit `DR` specifying the divide ratio for tag-to-reader link frequency, and 2-bit `M` for modulation format (*i.e.*, FM0 or Miller). `DR` determines the Backscatter Link Frequency (*BLF*) of the tag, where $BLF = \frac{DR}{TRcal}$. `TRext` specifies whether to prepend a tag-to-reader preamble with a pilot tone. The command also includes a 4-bit `Q`, which specifies the number of time slots of 2^{Q}. A 5-bit `CRC` is appended to ensure the command integrity. Receiving such a Query command, a commodity RFID tag should randomly select one slot out of 2^{Q} slots and respond a 16-bit RN16 with the specified modulation scheme and the link frequency.

2) Encode a reader-to-tag command: A command needs to be encoded according to the RFID protocol before transmission. In EPC C1G2 protocol, the commands (*e.g.*, 22-bit Query command) are encoded using PIE scheme. For example, the data-0 symbol (encoding bit `0` of a command) is represented with a short high value followed by a short low value, while the data-1 symbol (encoding bit `1`) is represented with a long high value followed by a short low value as shown in Fig.4. In practice, the high values correspond to emitted continuous waves, whereas the low values correspond to attenuated continuous waves, respectively.

The duration of data-0 (represented as Tari in Fig.4) ranges from 6.25μs to 25μs and the Pulse Width (PW) is approximately $0.265 \sim 0.525$ times of Tari. The first 4 bits of the Query command (*i.e.*, `1000`) are encoded into one data-1 symbol followed by three data-0 symbols as illustrated in Fig.4, which can be represented by 10 bits `1110101010`. The 10 bits (representing high and low values) will be sent using an OOK module to send the command.

3) Transmit a packet using on-off keying transmitter: In order to send an encoded command to a tag, a transmitter needs to generate high and low values according to the encoded command. Many commodity-off-the-shelf (COTS) transmitter modules support on-off keying (OOK) modulation, which can be used to generate the high and low values.

Many low-cost OOK modules (*e.g.*, RFM69HW [12] as shown in Fig.5) support OOK modulation and can operate over a wide range of frequency bands including UHF ISM band (*e.g.*, 915 MHz). The transmitter module provides interfaces to configure most of the major RF communication parameters (*e.g.*, carrier frequency, transmission power, and link frequency).

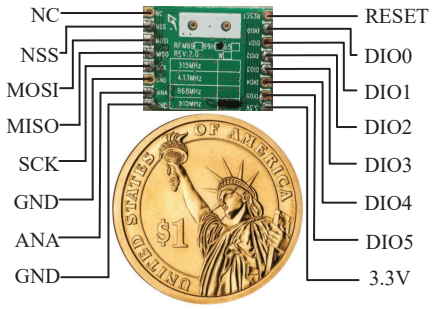


Fig. 5. OOK transmitter module: HopeRF RFM69HW.

Arduino	RFM69HW
GND	GND (ground in)
3V3	3.3V (3.3V in)
Interrupt 0 pin D2	DIO0 (interrupt request out)
SS pin D10	NSS (chip select in)
SCK pin D13	SCK (SPI clock in)
MOSI pin D11	MOSI (SPI Data in)
MISO pin D12	MISO (SPI Data in)

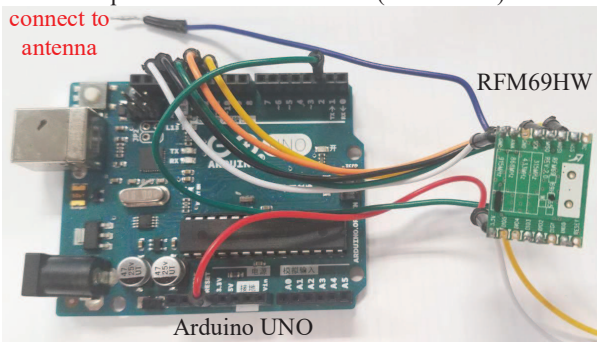


Fig. 6. Connect RFM69HW transmitter module to Arduino UNO board.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3067379, IEEE Internet of Things Journal

TABLE II
THE QUERY COMMAND IN OUR WORK AND CORRESPONDING QUERYBYTES SENT OUT BY THE TRANSMITTER.

<table>
<tr><td></td><td colspan="7">reader-to-tag preamble</td><td colspan="29">Query</td></tr>
<tr><td></td><td>D</td><td>0</td><td colspan="2">RTcal</td><td colspan="3">TRcal</td><td colspan="5">Command</td><td>DR</td><td colspan="3">M</td><td>TRext</td><td colspan="3">Sel</td><td colspan="2">Session</td><td colspan="2">Target</td><td colspan="5">Q</td><td colspan="7">CRC</td></tr>
<tr><td>Query cmd</td><td>/</td><td>/</td><td colspan="2">/</td><td colspan="3">/</td><td>1</td><td colspan="2">0</td><td>0</td><td>0</td><td>0</td><td colspan="2">0</td><td>0</td><td>1</td><td colspan="2">0</td><td>0</td><td>0</td><td>0</td><td colspan="2">0</td><td>0</td><td>0</td><td>0</td><td colspan="2">0</td><td>1</td><td>0</td><td colspan="2">0</td><td>1</td><td colspan="2">1</td></tr>
<tr><td>QueryBits</td><td>0</td><td>10</td><td>11111</td><td>0</td><td>1111111</td><td>11111111</td><td>110</td><td>1110</td><td>1</td><td>0</td><td>10</td><td>10</td><td>10</td><td>1</td><td>0</td><td>10</td><td>1110</td><td>1</td><td>0</td><td>10</td><td>10</td><td>10</td><td>1</td><td>0</td><td>10</td><td>10</td><td>10</td><td>1</td><td>0</td><td>1110</td><td>10</td><td>1</td><td>0</td><td>1110</td><td>111</td><td>0[1]</td></tr>
<tr><td>QueryBytes</td><td colspan="3">0x5F</td><td colspan="2">0x7F</td><td>0xFF</td><td colspan="3">0xDD</td><td colspan="5">0x55</td><td colspan="4">0x5D</td><td colspan="5">0x55</td><td colspan="5">0x55</td><td colspan="4">0x75</td><td colspan="3">0x77</td><td>0x7F</td></tr>
</table>

[1]The last bit `0` should combine with seven `1`, resulting in QueryBytes of 0x7F.

To enable flexible programming, we connect the IC pins of the transmitter module to an Arduino Uno board as illustrated in Fig.6. We program the transmitter module using the RadioHead library v1.84 [13]. The library provides flexible interfaces to configure key parameters. We configure the carrier frequency to 915 MHz, specify the transmission power to 20 dBm, and send a packet with the payload of QueryBytes. We note that the RFM69HW communication module can be connected to both Arduino and Raspberry Pi for emitting Query command. We choose Arduino for the following reasons:

- The Arduino is cheaper than Raspberry Pi due to its simpler hardware infrastructure. Arduino is a microcontroller, which is a part of the computer, while Raspberry Pi is a mini-computer with its own operating system. Therefore, Arduino is more suitable for low-cost system design in our work.
- It is very simple to interface sensors and other radio components to Arduino, while Raspberry Pi requires the installation of libraries and software for interfacing electronic components.
- Arduino can be powered up using a battery pack, which is more suitable for fulfilling parallel sensing tasks, while Raspberry Pi is difficult to power using a battery pack.
- Arduino board is a just plug-and-play device, which enables running program when it is connected to the power and simply stop running if it is disconnected. However, Raspberry Pi could result in a risk of file corruption and software problems if it is not properly shut down.

4) Cross technology packet-in-packet communication: Allow COTS tags to receive reader-to-tag command: We notice that commercial RFID readers cannot be applied in our distributed design since commercial readers do not provide access to raw physical layer samples and incur high deployment cost. A transceiver module can receive a packet and retrieve the payload transmitted by other modules. In reception, the reverse operation can be performed by the transceiver module, including detection of the preamble, synchronization to Sync word, optional AES decryption, CRC detection, and DC-free decoding. However, we note that the intended receiver of a transmitted packet is not a transceiver module but *a passive COTS tag*. As we expect no modification to COTS tags, the transmitted packet should be received and interpreted as a protocol-compatible command (*e.g.*, a Query command).

We program the transmitter such that the generated radio waves in the air should be exactly the same as those generated by a commodity reader. The Query command and the corresponding QueryBytes sent out by the transmitter are shown in Table II. In the table, the reader-to-tag preamble and the Query command are encoded with QueryBits representing high and low values, *i.e.*, 1 for high value corresponding to the on state in OOK, and 0 for low value corresponding to the off state in OOK. The QueryBytes (*i.e.*, from the first `0x5F` to the last `0x7F`) are the payload bytes that we can send to the transmitter module using the `send()` function.

In order to provide sufficient power for tags to wake up and detect the preamble (corresponding to `0x5F`, `0x7F`, `0xFF`, and `0b110`) and receive the Query command, the transmitter prepends multiple `0xFF` before the Query command packet by sending high values to generate continuous waves (CW). The transmitter module also appends multiple `0xFF` while waiting for tag's response of RN16 after sending the last `0x7F` of the Query command.

We configure the bit rate of transmitter such that the duration of high and low values follow the EPC C1G2 timing requirement (*e.g.*, 12.5μs delimiter, Tari, RTcal, *etc.*), as illustrated in Fig.7. In addition, we disable DC-free encoding of data of the transmitter module, so that the encoded reader-to-tag command can be faithfully transmitted using on-off keying; Otherwise, the default DC-free encoding scheme (*e.g.*, Manchester or whitening) of the transmitter module could encode the encoded payload and make the double-encoded command incomprehensible to COTS tags. We also avoid the automatic packet handling of the module by disabling preamble, Sync word, AES, and CRC by configuring relevant registers [12], as shown in Fig.8.

B. RFID receiver design

1) A receive-only SDR as an RFID receiver: In order to extract PHY information and measure low-level data, we use an SDR to build an RFID receiver. Many COTS SDRs (*e.g.*, USRP, WARP, SoRa, HackRF, RTL-SDR, NESDR Mini 2+, *etc.*) can be used to collect the PHY information. One advantage of the proposed distributed architecture is that the RFID receiver does not need to transmit any radio signals but only need to receive PHY samples. As such, many low-cost SDR receivers (*e.g.*, RTL-SDR, NESDR Mini 2+, *etc.*) can be used to receive backscatter signals from tags. More importantly, multiple receivers can be depolyed to simultaneously receive the backscatter signals without any collision. In our implementation, we choose a RTL-SDR dongle with RTL2832 ADC chip (approximately 20 USD) to build an RFID receiver. We use the same circular antennas for the transmitter and the receiver. We note that our design is not limited to a particular type of SDR and many high-end SDRs can potentially deliver a better performance.

Fig.9(a) plots the PHY samples measured with the receive-only RTL-SDR dongle. We see that CW was sent to power up the tags followed by the Query command. During the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3067379, IEEE Internet of Things Journal

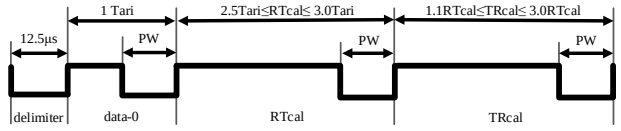


Fig. 7. Reader-to-tag preamble.

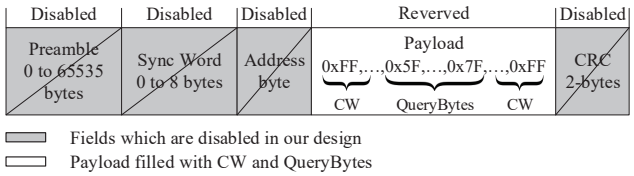


Fig. 8. Packet-in-packet technique in our design.

backscatter communication, the amplitude changed since the tag switches between the reflect state and the absorb state. The CW was continued during the tag's response of RN16 to provide power to the tag. Our observation is that 1) the Query command was successfully transmitted using the transmitter; and 2) the tag responded an RN16, which validates the successful transmission of the Query command. However, the PHY samples exhibit dramatic fluctuation compared with those measured with USRP N210 [14]. To test whether the fluctuation is because of the relatively poor performance of the SDR (RTL-SDR dongle), we use a high-end SDR (USRP N210) to measure the PHY samples without changing the settings of either the transmitter or the tag. The experiment results with the high-end SDR measures similar fluctuations in PHY samples, indicating that the fluctuation of PHY samples is not due to the performance of the receive-only SDR. As a matter of fact, the performance of the SDR turns out to be sufficient for our purpose of RFID sensing.

2) CFO between transmitter and receiver: In this subsection, we will investigate the root cause of the fluctuation in the measured PHY samples. Different from COTS RFID reader, our design uses a distributed architecture, which decouples the functionality of a single reader into two parts: a transmitter (*e.g.*, RFM69HW module) and a receiver (*e.g.*, RTL-SDR dongle). These devices are manufactured by different vendors and equipped with different oscillators. As a result, the clock frequency of the transmitter and the receiver may not be strictly synchronized in practice. For example, the local oscillator signal of the receiver in down-conversion and the transmitted carrier signal may not be with the same frequency, leading to the CFO between the transmitter and the receiver.

The received signal $y(t)$ is the superposition of the carrier wave and the tag's backscattered signal, which is down-converted to the baseband signal as follows:

$$S(t) = \overbrace{[A_{cw}e^{-j(2\pi f_{c1}t+\alpha)} + x(t)A_{tag}e^{-j(2\pi f_{c1}t+\Phi)}]}^{\mathbf{y(t)}} \times e^{-j2\pi f_{c2}t} \quad (1)$$

where A_{cw} and A_{tag} are the amplitude of the carrier wave and the backscattered signal of the tag, respectively. $x(t)$ is the binary values, indicating either absorb or reflect states. α and Φ are the phase of the carrier wave and the backscattered signal, respectively.

After passing a bandpass filter, if there exists CFO between transmitter and receiver (*i.e.*, $f_{c1} - f_{c2} = \Delta f \neq 0$), the down-converted complex baseband signal becomes:

$$S(t) = \underbrace{A_{cw}e^{-j(2\pi\Delta f t+\alpha)}}_{\textbf{Self-interference}} + \underbrace{x(t)A_{tag}e^{-j(2\pi\Delta f t+\Phi)}}_{\textbf{Backscattered signal}} \quad (2)$$

where Δf is the unknown CFO between the transmitter and receiver. The received signal is a function of t and the phase changes at the constant rate $2\pi\Delta f$. Since $\Delta f \neq 0$ due to the CFO, the phase of $S(t)$ will change over time. In addition, the first term is known as the self-interference, which should be removed since it is irrelevant to the backscattered signal.

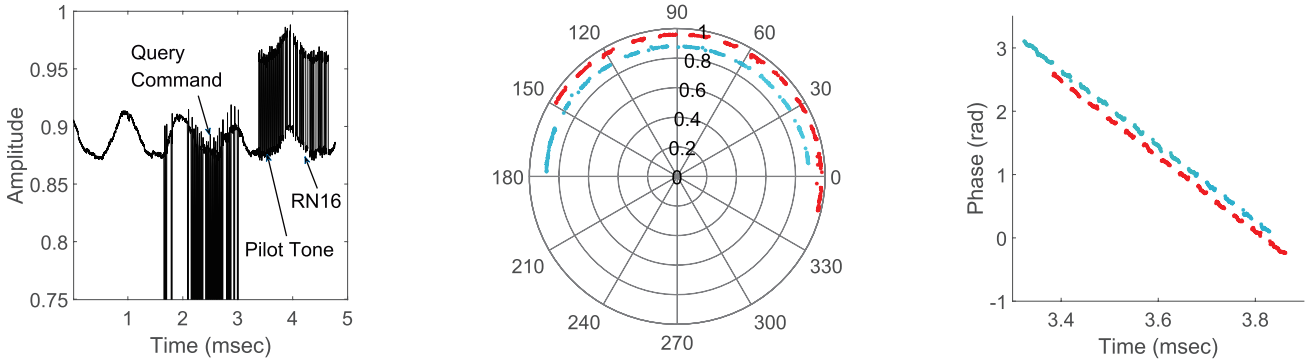
We examine how CFO affects the signal phase in PHY in Fig.9. We observe a sinusoidal-like profile throughout the Query and RN16 response process (*i.e.*, starting from the continuous wave) in Fig.9(a). We focus on the PHY samples of the RN16 pilot tone and separate the samples into two states using a threshold on the signal amplitude. Due to CFO, the PHY samples rotate and exhibit a circular pattern as shown in Fig.9(b). The red and blue dots represent the reflect state and the absorb state, respectively. The radius of a PHY sample indicates the signal amplitude, while the angle indicates the signal phase in Fig.9(b). In order to know how such a circular pattern was formed in the I/Q constellation, we plot the phase of PHY samples in the time domain. In Fig.9(c), we observe that the phases linearly decrease for both states with the same decreasing rate. The results indicate that the phases of PHY samples rotated clockwise in the I/Q constellation diagram. During the rotation process, the phase of absorb state (blue dots) decreased from around $180°$ to $0°$, while the reflect state (red dots) also decreased and rotated clockwise for $180°$.

3) CFO cancellation: Due to CFO, the PHY samples do not concentrate in two states, but exhibit circular patterns in the I/Q constellation. The low-level data cannot be extracted if we do not handle the CFO properly, One possible CFO cancellation method is to first measure the exact CFO by calculating the slope of the signal phase and then calibrate the frequencies of the transmitter and the receiver. We name such a calibration method as *offline calibration*. However, in practice, the CFO is affected by many factors (*e.g.*, power supply, temperature, *etc.*), which may vary over time. As a result, it is hard to calibrate the frequencies of transmitter and receiver before the transmission and the reception.

To address this problem, we propose an *online calibration* method to mitigate the impact of CFO. Our key observation is that the CFO would not change much during a short period from the transmission of CW to the end of tag's response. In other words, Δf should remain relatively stable during one interrogation, which is around $5ms$ in practice.

The key idea of our CFO cancellation method is to cancel the impact of CFO during RN16 transmission with the same CFO during the CW measurement. In particular, we divide the PHY samples measured during RN16 by the PHY samples measured during CW (indicated in Fig.9(a)) to cancel the impact of CFO. We denote CW as $A_{cw}e^{j(2\pi\Delta f t+\alpha')}$, where α' is the unknown initial signal phase. After the CFO cancellation,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3067379, IEEE Internet of Things Journal



(a) The received physical samples. (b) I/Q constellation of the RN16 pilot tone. (c) Phase of the RN16 pilot tone

Fig. 9. The impact of CFO.

we have

$$S'(t) = \frac{S(t)}{CW} = \frac{A_{cw}e^{-j(2\pi\Delta ft+\alpha)} + x(t)A_{tag}e^{-j(2\pi\Delta ft+\Phi)}}{A_{cw}e^{-j(2\pi\Delta ft+\alpha')}} = \underbrace{e^{-j(\alpha-\alpha')}}_{\textbf{Self-interference}} + x(t)\frac{A_{tag}}{A_{cw}}e^{-j(\Phi-\alpha')} \tag{3}$$

By doing so, the CFO term $2\pi\Delta ft$ in $S(t)$ can be canceled and $S'(t)$ is not affected by the CFO.

We plot $S'(t)$ in the In-phase and Quadrature (I/Q) constellation as shown in Fig.10(a). The PHY samples of $S'(t)$ are clearly divided into two clusters corresponding to the two tag states. More importantly, as the CFO is removed, the two clusters are relatively concentrated instead of exhibiting circular patterns. We measure the phase of $S'(t)$ according to different states and the phases of both two states keep constant over time, as illustrated in Fig.10(b).

4) Low-level data extraction: Eq.(3) can be illustrated in an In-phase and Quadrature (I/Q) constellation in Fig.10(c). The self-interference (also known as carrier wave) is denoted as $\overrightarrow{OC}$ with phase α and the backscattered signal is denoted as $\overrightarrow{CA}$ with phase Φ, respectively. When $x(t) = 0$, the received baseband signal is $\overrightarrow{OC}$. When $x(t) = 1$, the received baseband signal becomes $\overrightarrow{OA} = \overrightarrow{OC} + \overrightarrow{CA}$. The phase of backscattered signal Φ can be measured from $\overrightarrow{CA}$, which can be obtained by subtracting the carrier wave $\overrightarrow{OC}$ from $\overrightarrow{OA}$.

Although the CFO is successfully removed, we still have two undesired terms (1) self-interference which affects the received backscattered signal and (2) an unknown parameter α', which denotes the unknown initial signal phase and may vary depending on the CW measurements. We observe that the first and second term in Eq.(3) both contain the same unknown parameter α'. Note that the self-interference term $e^{-j(\alpha-\alpha')}$ in Eq.(3) is a constant value, which can be estimated by averaging the signal samples in absorb state in $S'(t)$. Therefore, the self-interference and unknown parameter α' can be removed as follows:

$$S_{tag} = \frac{\overbrace{S'(t) - e^{-j(\alpha-\alpha')}}^{\textbf{Self-interference Cancellation}}}{e^{-j(\alpha-\alpha')}} = \frac{x(t)\frac{A_{tag}}{A_{cw}}e^{-j(\Phi-\alpha')}}{e^{-j(\alpha-\alpha')}} = x(t)\frac{A_{tag}}{A_{cw}}e^{-j(\Phi-\alpha)} \tag{4}$$

From Eq.(4), we notice that 1) the self-interference is canceled; 2) the unknown phase offset α' is removed such that we can randomly segment the continuous CW to remove the CFO and 3) the measured phase turns out to be $\Phi - \alpha$ rather than Φ. However, α is a constant phase offset representing the initial phase of the carrier wave. Therefore, the measured phase $\Phi - \alpha$ and the phase of backcattered signal Φ share the same trend. Note that the constant phase offset α can be further removed if we use relative phase values. We name S_{tag} as *the tag signal* and the corresponding phase $\theta = \Phi - \alpha$ as *the phase of tag* (*i.e.*, angle θ in Fig.10(c)).

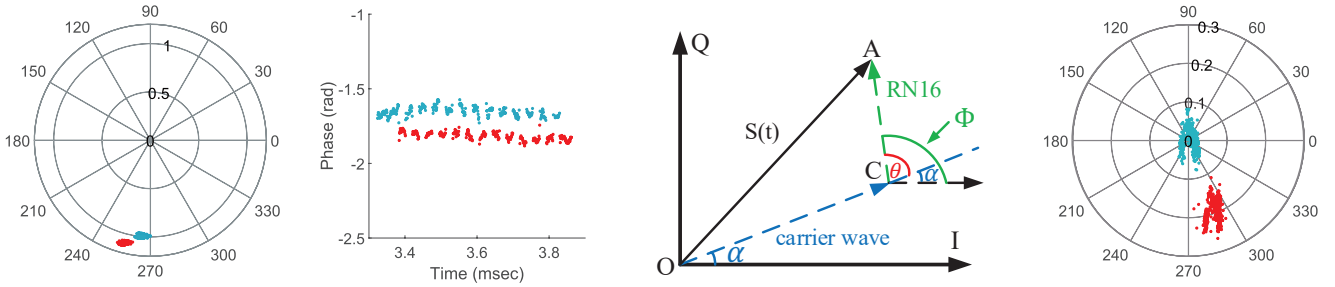
The I/Q constellation diagram of *the tag signal* S_{tag} is shown in Fig.10(d). The samples are gathered in two clusters. The PHY samples in absorb state assemble near origin point with attenuated amplitude and phase, which reveals that the self-interference is successfully canceled. Therefore, the phase θ can be directly extracted from S_{tag} by measuring the phase of the centers of the two clusters, while the amplitude of the backscattered signal can be measured with the distance between two cluster centres.

IV. Evaluation

A. Experiment setting

The validity of measured phase values significantly affect the performance of sensing tasks. In this section, we evaluate the validity of the extracted phase values. The transmitter is composed of an RFM69HW chip connected to an Arduino Uno Rev3 microcontroller board. The RFM69HW chip is programmed using the RadioHead library for Arduino. We transmit the Query command by programming the Arduino board to periodically send the QueryBytes (Table II) at 915MHz. The RFM69HW chip works at its High-Power mode which can provide the maximum transmission power of +20dBm. To get higher transmission power and longer communication range, we add an external power amplifier. In particular, we use an RF signal power amplifier of 2W (approximate 15 USD) to ensure the tag can still be activated at some distances to the transmitter. An RTL-SDR dongle with RTL2832 ADC chip is used as the receiver. We use the RTL-SDR based on GnuRadio to receive and demodulate the backscattered signal with sampling rate 1M/s and output the demodulated signal as complex values. Fig.11 illustrates the experiment setup.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3067379, IEEE Internet of Things Journal



(a) CFO removal. (b) Phase of the pilot tone without CFO. (c) IQ plot of the backscattered signal. (d) I/Q constellation of *the tag signal*.

Fig. 10. Measuring *the tag signal*.

B. Evaluation of phase

Experiment (1): We move six RFID tags towards the transmitter for $1m$ at a constant speed. The measured phase depends on the propagation distance, as well as the hardware imperfection factors. In fact, the phase of the backscattered signal can be formulated as:

$$\Phi = mod(\frac{2\pi d}{\lambda} + \theta_{tx} + \theta_{rx} + \theta_{tag}, 2\pi) \quad (5)$$

where λ is the signal wavelength, and θ_{tx}, θ_{rx} and θ_{tag} denote the extra phase offsets caused by the hardware of transmitter, receiver and tag, respectively. Note that different hardware (*e.g.*, Impinj RFID, USRP N210) incurs different phase offsets. As a result, we cannot regard the phase value extracted by the commodity RFID system as ground truth. However, as the phase changes 2π when the propagation distance increases one wavelength, the trend of the signal phase during tag movement should change linearly and periodically from 0 to 2π.

Fig.12 shows the extracted *phase of tag* for all six tags. The phase values linearly change from 0 to 2π for three times during tag movement with the wavelength of $32.8cm$, which approximately matches the moving distance of $1m$. We extend the experiment with ten different RFID tags from different manufacturers and observe similar trends.

From Fig.12, we notice that although the absolute phase values may differ due to different hardware, their relative phase values (*i.e.*, phase difference) should remain the same even if different hardware is used. This is because the phase offset caused by the same hardware is canceled when measuring the relative phase values [15, 16]. More importantly, the phase offset of carrier wave (*i.e.*, α) is removed as well by adopting relative phase values. Therefore, to further validate the extracted phase values, we conduct another experiment that uses the relative phase values.

Experiment (2): We evaluate the measured relative phase values. In this experiment, we place the antenna of a commodity Impinj RFID reader at the transmitter position (*i.e.*, upper blue circle). An RFID tag is then placed at position $p1$ to $p4$ (*i.e.*, red circles) separated by $0.1m$. The Impinj Reader measures phase values of the backscattered signal for 5 minutes at each position. The measured signal phase captures the round-trip propagation distance between the antenna and the tag since the commodity RFID system supports a full-duplex transceiver with the same antenna. The phase differences for each position pair are then calculated, which are regarded as the ground truth.

Note that the proposed RFID system uses a distributed architecture with separated antennas of the transmitter and the receiver. To equalize a round-trip scenario, we symmetrically deploy the antenna of the transmitter and receiver (*i.e.*, upper and lower blue circles as illustrated in Fig.13(a)). We repeat the PHY samples collection at 4 different positions (*i.e.*, $p1$ to $p4$) and calculate the residual deviation of phase difference compared to the ground truth.

Fig.13(b) shows the CDF of the residual deviation of phase difference between the distributed RFID system and the ground truth, where p_{ij} denotes the phase difference between position pi and pj. The maximum residual deviation of phase difference for all position pairs is less than 0.25 and only approximate 5% percent of residual deviation exceeds 0.15, which represents a distance error of less than $7.8mm$. The results of Experiment (1) and (2) indicate that the proposed RFID system is capable of delivering similar phase measurement accuracy while incurring approximately 2% of a high-end commodity RFID system.

C. Evaluation of responding rate

A stable and consistent respond rate reflects the ability of an RFID reader successfully interrogating the RFID tag and the tag correctly responding RN16, which greatly affects the resolution of the channel measurements, and hence the performance of the sensing applications [4, 17]. In our low-cost system, the respond rate is measured as the number of successfully responded RN16 of a tag per second, which indicates the number of output channel measurements per second. We configure the Arduino such that approximately 40 Query commands are sent to the tag per second. We evaluate the following factors affecting the reading rate: distances and angles between tag and antenna and the duration of CW.

Distances and angles: We evaluate the impact of different distances and angles between the transmitter and the tag on the respond rate. The transmitter gain is configured to its maximum value of $+20$dBm. We place the tag at different positions in the $180°$ area in front of the transmitter antenna, as illustrated in Fig.14(a). The blue dot represents the location of the transmitter antenna and the same tag is placed at positions ranging from $0m$ to $2.5m$ and in different angles ranging from $0°$ to $180°$ (*i.e.*, red dots). The receiver antenna is placed $5m$ away from the transmitter antenna in the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3067379, IEEE Internet of Things Journal

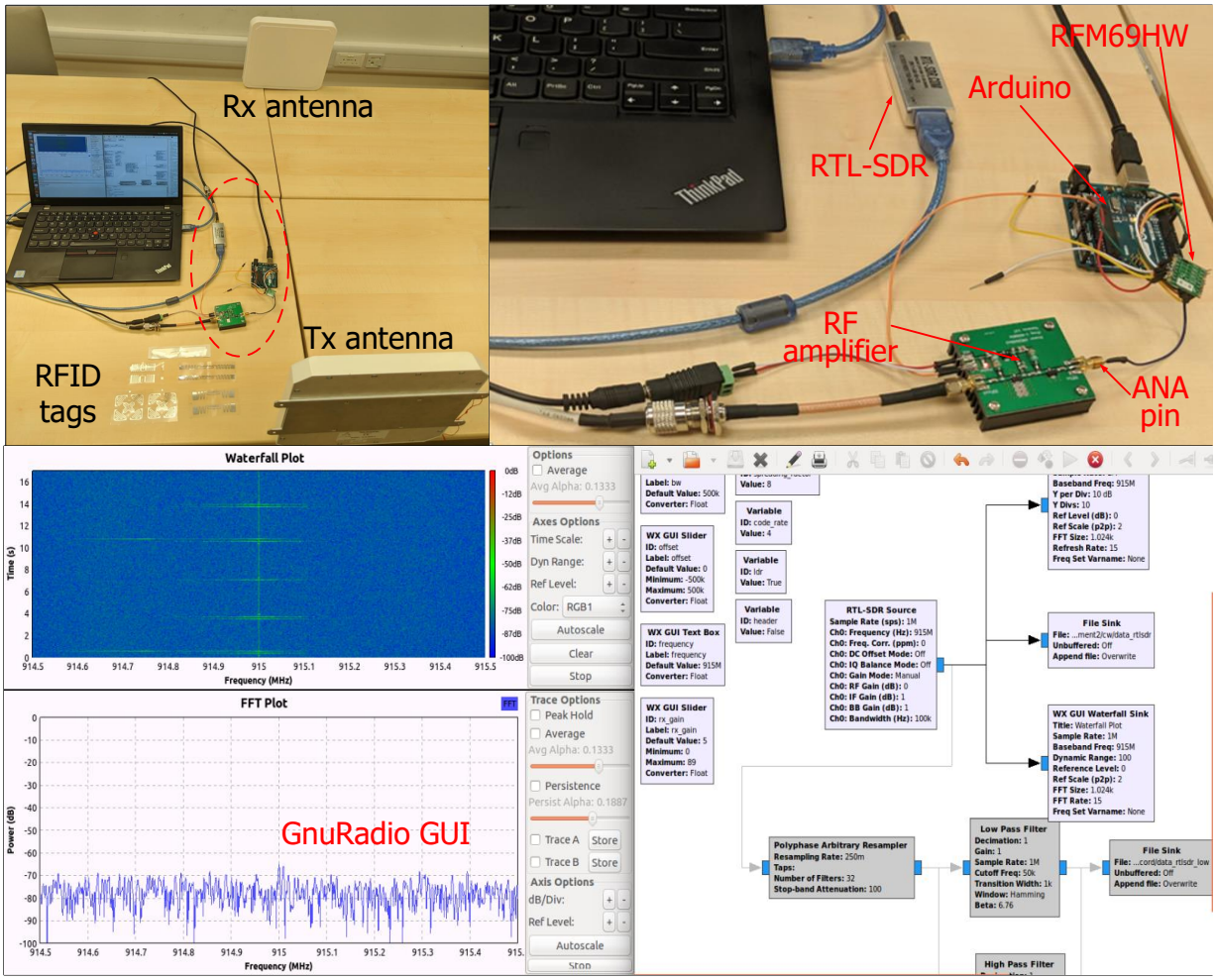
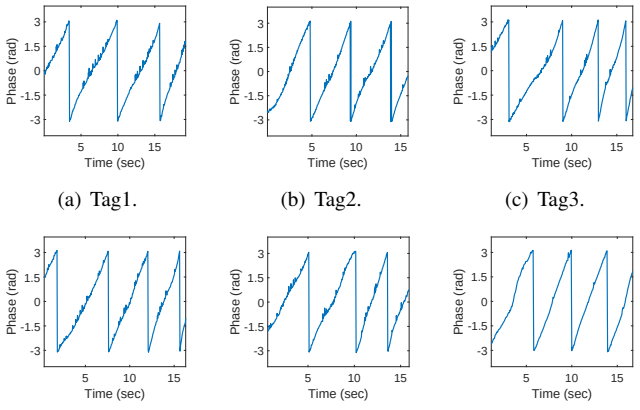


Fig. 11. The experiment setup.



(a) Tag1. (b) Tag2. (c) Tag3.

(d) Tag4. (e) Tag5. (f) Tag6.

Fig. 12. The extracted phase values when the tags move for $1m$.

direction of 90° (not shown in Fig.14(a)). At each position, we collect the PHY samples for 2min.

In Fig.14(a), we mark the average reading rate at each position. Note that the response rate can be nearly 100% in our experiments, meaning that the tag can almost always decode the Query command and respond accordingly if the tag is in the effective sensing area. The effective sensing area can be enlarged by increasing the transmission power, *e.g.*, using a low-cost power amplifier.

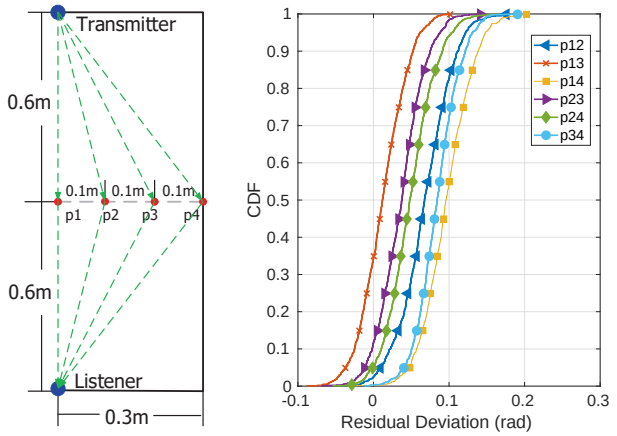
The relative distance and angle between the tag and receiver's antenna may influence the respond rate as well since the proposed system uses a distributed architecture. In the second experiment, we guarantee the activation of the tag by fixing the distance between transmitter's antenna and tag to $1m$ in the direction of 90°. The receiver's antenna is placed in the direction of 90° as well while at different distances to tag ranging from $1m$ to $4m$ in $1m$ step. The receiver gain is set to its maximum value to maximize the sensing range. At each position, we rotate the receiver's antenna for 360° in 45° step and collect the PHY samples for 2min.

Fig.14(b) shows the average respond rate. The receiver can receive tag response with a consistent respond rate of approximately 40 reads/sec across all directions. Once the tag is activated, the signal will be backscattered using the tag's omnidirectional antenna, which enables more flexible deployment of the receiver. In addition, the sensing area of the receiver can be easily enlarged by setting corresponding receiver gain.

V. Case Study

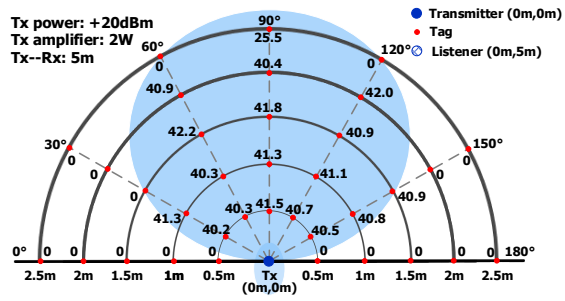
We conduct three case studies with the developed RFID system: 1) Respiration monitoring; 2) Object localization; and 3) Cardinality estimation. We envision more applications that

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3067379, IEEE Internet of Things Journal

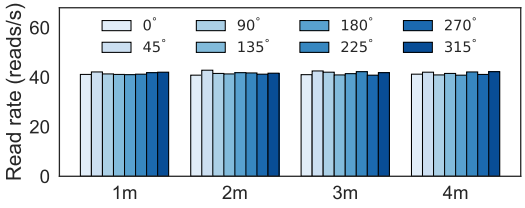


(a) Experiment layout. (b) CDF of residual deviation of phase difference.

Fig. 13. Validation of backscattered signal phase.



(a) Impact of distance and angle between transmitter and tag on respond rate.



(b) Impact of distance and angle between receiver and tag on respond rate.

Fig. 14. Evaluation of read rate.

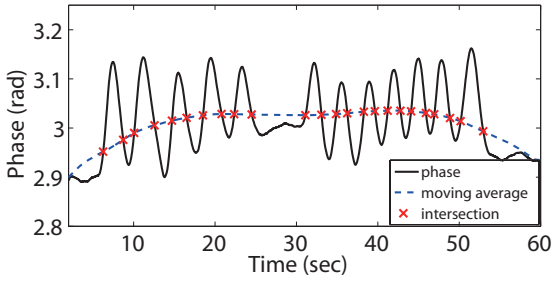


Fig. 15. Extract respiration rate.

are not limited to these use cases.

Respiration Monitoring: We explore the possibility of using the proposed low-cost RFID system to monitor human respiration by using extracted phase measurement. We attach an RFID tag to a user, track the movement of the tag, and thereby monitor respiration. We ask a volunteer to wear a tag on his chest and sit $1m$ away facing the two antennas. The volunteer firstly holds his breath for a few seconds and breathe normally for a while and repeats two times. The intuition is that human chest moves forward and backward periodically during respiration, which causes changes in the phase of the RF signal. The tag respond rate is set to 40Hz and we collect the PHY samples for around 1min. During collection, the volunteer breathes normally in his comfortable manner.

The black line in Fig.15 shows the measured phase values during the experiment. When the volunteer is breathing, the signal phase exhibits a clear periodical pattern due to the chest movement, while the signal phase remains flat when the volunteer holds his breath. The result in Fig.15 shows the potential application of our proposed low-cost RFID system in respiration monitoring.

Object Localization: One of the promising applications of commodity RFID systems is to locate RFID-labelled objects. We use the proposed distributed RFID system to localize an RFID tag in a monitoring area, as illustrated in Fig.16(a). We adjust a well-known RFID localization algorithm RF Hologram [15, 18] to localize an RFID tag. We note that in our application, the transmitter and the receiver are not collocated, which requires adjustment to the RF Hologram algorithm accordingly.

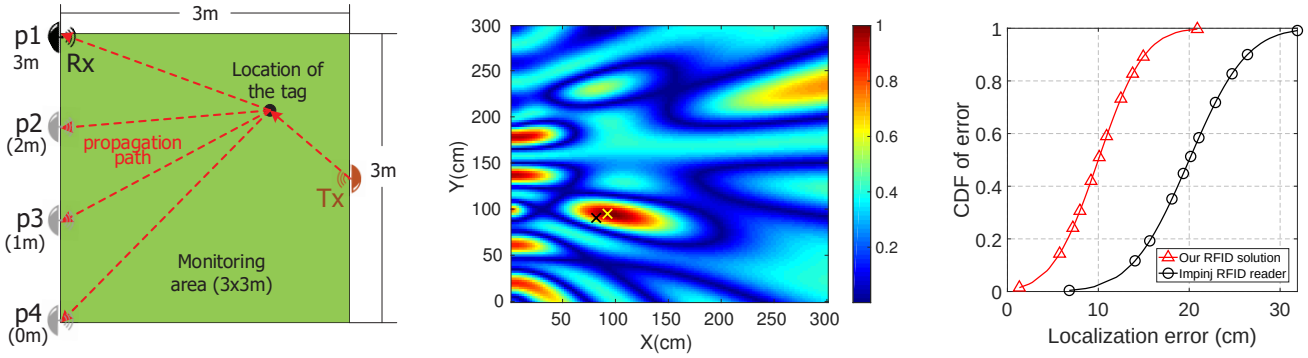
In specific, the antennas of transmitter (Tx) and receiver (Rx) are deployed at each side of the square area as in Fig.16(a), which covers an area of $3m \times 3m$. The monitoring area is divided into small grids of $1cm \times 1cm$ and an RFID tag is located in one of the grids. To generate an RF hologram, in previous work, COTS RFID readers are needed to measure phase values at different localization (*e.g.*, $p1$ to $p4$) and each reader is activated to interrogate the tag in a sequential manner to avoid collision. In our experiment, we concurrently place four receive-only receivers at four positions separated by $1m$ (*e.g.*, $p1$ to $p4$) and collect PHY samples.

For each grid, we can calculate the theoretical phase of the signal transmitted from the transmitter, backscattered at the grid and received by each receiver (*i.e.*, the red dotted line in Fig.16(a)). Therefore, the theoretical phase differences of each grid for each receiver antenna pair can be calculated (*e.g.*, in our experiment, 4 positions generate 6 antenna pairs). The pixel value of each grid in the hologram is calculated as:

$$p_{i,j} = |\sum_{k=1}^{K} e^{j(\theta_{i,j,k} - \theta_{m,k})}|/K \quad (6)$$

where i and j are the index of row and column, K denotes the number of antenna pairs, $\theta_{i,j,k}$ and $\theta_{m,k}$ denote the theoretical phase difference of grid $G_{i,j}$ and measured phase difference for antenna pair k, respectively. The term $e^{j\theta}$ represents the signal with unit amplitude and phase θ. The basic idea of the RF hologram is that if the tag is located in the grid G, the pixel value of grid G is significantly larger than the pixel

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3067379, IEEE Internet of Things Journal



(a) The case study of localizing an RFID tag. (b) Rf hologram of location 1. (c) Localization error.

Fig. 16. RF hologram of Localization.

values of other *none-tag* grids. This is because when grid $G_{i,j}$ is the target location, the term $(\theta_{i,j,k} - \theta_{m,k})$ approaches 0 for all k and $p_{i,j}$ reaches its maximum intensity equals to 1. Otherwise, $p_{i,j}$ will be close to 0.

Fig.16(b) shows an example of the hologram generated by the proposed RFID solution. The estimated location (yellow marker) is very close to the ground truth (black marker) with an error of less than $10cm$.

To compare the localization performance with the COTS RFID system, we use an Impinj R420 reader to sequentially collect the low-level data from $p1$ to $p4$ to generate RF hologram. While using the proposed RFID solution, we concurrently deploying four SDR receivers from $p1$ to $p4$ and one transmitter at Tx. We randomly place RFID tags at 10 different positions in the monitoring area. For each position, we record the ground truth and measured 2000 phase readings for Impinj R420 reader and 8000 phase readings for our RFID system. We average the phase difference every 20 measurements for COTS RFID reader and 80 for the distributed system to remove the noise and calculate the localization error by measuring the Euclidean distance between the estimated location and the ground truth. We measure 100 localization errors at each location.

Fig.16(c) shows the CDF of the localization error for all 10 different locations using COTS RFID reader (black curve) and our RFID solution (red curve), respectively. The proposed distributed RFID solution significantly outperforms the COTS RFID system. Over 80% localization errors are below 12cm when using the proposed RFID solution, while reaches 23cm for Impinj reader. Our RFID solution supports parallel sensing, which achieves higher network throughput (*i.e.*, $4\times$) than round-robin manner. Therefore, by collecting and averaging more phase reads within a certain period, the Gaussian noises in data can be more effectively removed and the quality of the data to generate hologram is improved as well.

Cardinality Estimation: Cardinality estimation aims to estimate the total number of tags from the responses of tags (*e.g.*, presence of RN16) *without collecting tag IDs*. Suppose the tag population is N. Given an accuracy requirement of (ϵ, δ)-approximation (*e.g.*, $\epsilon = 1\%$ and $\delta = 1\%$), we expect an estimate $\widehat{N}$ that satisfies $\Pr\{|N - \widehat{N}| \leq \epsilon N\} \geq 1 - \delta$.

We use our RFID transmitter to send Query command and use RFID listener to tell whether there is any RN16 response from tags. We adopt one of many prior cardinality estimation protocols (*i.e.*, ZOE estimator [19]). In ZOE, each tag responds to reader's request with a probability of P and keeps silent with $1 - P$. Intuitively, the more tags there are, the more likely that the reader receives responses. Specifically, $\Pr(idle) = (1 - P)^N$, where $\Pr(idle)$ represents the probability of observing no response (*i.e.*, idle) from N tags. Let X be a random variable which takes 1 with probability $\Pr(idle)$ and 0 with probability $1 - \Pr(idle)$. According to the law of large numbers, we can estimate $\widehat{N} = \ln \bar{X} / \ln(1 - P)$, where $\bar{X}$ is the average of m independent measurements (*i.e.*, $\bar{X} = \frac{1}{m}\sum_{i=1}^{m} X_i$) and X_i denotes the i^{th} measurement.

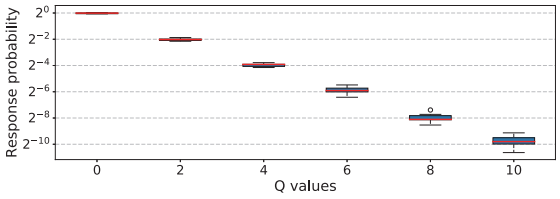
In our implementation, we specify the number of slots (*i.e.*, 2^{Q}) by adjusting Q in the Query command. Receiving such a command, each tag randomly selects one out of the 2^{Q} slots and responds RN16. As such, we can set the response probability of each tag in each time slot as $P = 1/2^{\mathrm{Q}}$. In each interrogation, *we only examine if there is any RN16 response in the first time slot. We avoid and skip the remaining* $2^{\mathrm{Q}} - 1$ *time slots, which takes no communication or execution time in practice*. We repeat the interrogation for m rounds and measure $\bar{X}$ and $\widehat{N}$.

In the experiment, we make sure that the COTS tags are within the communication range and respond to the Query command, meaning that we do not consider the problem of non-responding tags in this experiment.

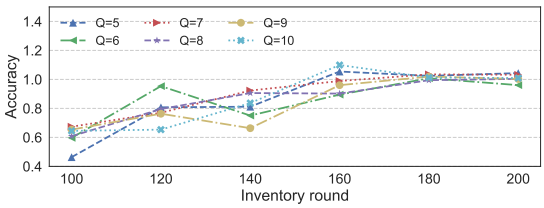
One assumption is that we should be able to set the response probability of each tag $P = 1/2^{\mathrm{Q}}$ by adjusting Q. To validate such an assumption, we send varied $\mathrm{Q}, 0 \leq \mathrm{Q} \leq 10$. We interrogate the tag for $m = 10000$ rounds and use the RFID listener to measures $\bar{X}$ in the first time slot when interrogating one COTS tag. We repeat this experiment 10 times for each Q. Fig.17(a) shows the measured response probability for varied Q. The response probability is consistent with the theoretical probability $P = 1/2^{\mathrm{Q}}$.

We place $N = 30$ COTS tags in front of the RFID transmitter and estimate the number of tags. We use the accuracy metric $Accuracy = \bar{N}/N$, where $\bar{N}$ and N denote the estimate and the actual number of tags, respectively. In the experiment, we estimate with different number of measurements and different Q in Fig.17(b). By interrogating 200 rounds, the accuracy is very close to 1 regardless of the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3067379, IEEE Internet of Things Journal



(a) Evaluation of response probability



(b) Estimation accuracy for different Q values.

Fig. 17. Evaluation of cardinality.

values of Q.

VI. Related Work

SDR UHF Reader for RFID identification. The distributed architecture, as well as SDR UHF reader for RFID identification, has been proposed in previous works [11, 20, 21, 25–27]. However, such works either only decode the reader's query messages [25] and read tag IDs [26] for RFID identification or do not support parallel sensing [20, 21, 27]. There are other systems that exploit SDR and purpose-built sensors for sensing tasks [22–24]. Universal Software Radio Peripheral (USRP) has been used to extract channel state information (CSI) in Wi-Fi, which can be applied to infer the type of human activities in a contact-free manner [22]. However, high-end USRP incurs a relatively higher cost (e.g., $>$ 2000USD), which is hard for large-scale deployment. Electrodermal Activity (EDA) sensors can be connected to the Arduino board and monitor the skin hydration level of humans by applying bio-electrical impedance analysis (BIA) technology [23, 24]. However, such purpose-built sensors are not fully programmable, which only fulfil particular applications and do not support parallel sensing tasks. Table III compares our proposed system with the most relevant state-of-the-art works. Our work differs from previous works in that our solution is low-cost, universal, fully programmable, and aims to support a variety of novel sensing applications by accurately extracting PHY information in the distributed architecture and enables parallel sensing to avoid communication collision.

Cross technology communication. TiFi [28] enables commercial WiFi receivers to identify UHF RFID tags by leveraging the harmonic backscattering. However, WiFi receivers cannot read tags by themselves but require the help of RFID readers in the identification process. FreeBee [29] enables cross-technology (*e.g.*, WiFi, ZigBee, Bluetooth) by shifting the timing of periodic beacon frames without incurring extra traffic. WEBee [30] enables high-speed cross-technology communication by emulating the desired signals of a low-speed radio (*e.g.*, ZigBee) with a high-speed wireless radio (*e.g.*, WiFi OFDM). Our work uses a COTS OOK module to transmit protocol-compatible RFID signals and energize tags.

RFID PHY. High-end COTS RFID readers (*e.g.*, Impinj R420, Alien ALR 9900+) provide interfaces for low-level data for application development. We have witnessed many innovative applications based on the COTS RFID systems (*e.g.*, object tracking [1–3, 18, 31], activity recognition [17, 32, 33], health care [4–7], vibration measurement [34], surface shape monitoring [35]). The RFID PHY, however, is not open to system developers or protocol designers. Open source projects [14] implement the RFID PHY based on the GNU Radio toolbox [36] for software-defined radios. Cross layer design and optimization have been proposed to improve the performance of RFID systems (*e.g.*, parallel decoding and identification [37–41], tag identification [8, 42–45], missing tag detection [46–48], moving tag detection [49], *etc.*).

RFID localization and tracking. Leveraging low-level data provided by COTS RFID systems, RFID localization and tracking have been advanced in recent years. Tagoram [18] uses COTS RFID systems for object localization and tracking. Tagoram proposes differential augmented hologram using the phase values collected from COTS RFID readers and achieves centimeter-level accuracy and high precision. Tadar [16] tracks moving objects without attaching tags to the objects even through a wall. As Tadar uses COTS RFID systems, the deployment cost is much lower than other see-through-wall technologies. OmniTrack [50] explicitly considers the impact tag orientation and proposes an orientation-aware phase model to address the practical challenge of orientation variation in RFID localization and tracking. Prior work [11] proposes a distributed architecture where one transmitter coexists with multiple listeners for distributed RFID sensing. The prior work uses one COTS RFID reader to energize tags and uses multiple USRP SDRs to collect PHY information for RFID sensing. Unlike prior work, we build a distributed RFID transmitter using a COTS send-only module and collect PHY information using a receive-only SDR. We build a backscatter model for the distributed architecture, where the transmitter and the listener are not collocated. Based on the model, we propose localization and tracking schemes.

Acknowledgement

This work is supported in part by Hong Kong GRF under grant PolyU 152165/19E, the Fundamental Research Funds for the Central Universities 531118010612, Hong Kong RGC Research Impact Fund (RIF) with project number R5034-18 and Hong Kong RGC Theme-based Research Scheme (TRS) with project number T41-603/20-R. Yuanqing Zheng is the corresponding author.

VII. Conclusion

In this paper, we propose the design and implementation of a software defined UHF RFID system that enables distributed parallel RFID sensing and provides full access to raw physical layer samples of backscatter signals. As such, our solution can be used to develop various innovative RFID applications. To this end, we overcome a series of technical challenges

TABLE III
COMPARISON OF EXISTING STATE-OF-THE-ART TECHNOLOGIES.

	Technology	Hardware	Cost	SDR	Parallel Sensing	Application
Our System	RFID	RF69+Arduino	Low-cost	Yes	Support	Sensing Tasks
Edward A. Keehr [20]	RFID	FPGA	Low-cost	Yes	Nonsupport	Identification
Pavel V. Nikitin et al. [21]	RFID	MSP430+TH7203x	Low-cost	Yes	Nonsupport	Identification
William Taylor et al. [22]	Wi-Fi	USRP	High-cost	Yes	Nonsupport	Human Activity Recognition
Sidrah Liaqat et al. [23] Ming-Zher Poh et al. [24]	BIA	EDA Sensor+Arduino	Low-cost	No	Nonsupport	Skin Hydration Level

and implement the system and extract PHY information using low-cost commodity components directly available on the market. We apply conduct three case studies of sensing applications. We note that the objective is not to replace the conventional COTS readers which implement the full RFID protocol stacks, while our RFID solution aims to implement the essential function of interrogating tags, measuring their backscattered signals and fulfilling various sensing tasks. In the future, we plan to develop the software defined RFID system by supporting more functions such as extraction of backscatter signals of multiple tags, parallel decoding and identification, *etc*.

REFERENCES

[1] J. Lai, C. Luo, J. Wu, J. Li, J. Wang, J.Chen, G. Feng, and H. Song, "Tagsort: Accurate relative localization exploring rfid phase spectrum matching for internet of things," *IEEE Internet of Things Journal*, vol. 7, no. 1, pp. 389–399, 2020.

[2] S. Zhang, C. Yang, D. Jiang, X. Kui, S. Guo, A. Y. Zomaya, and J. Wang, "Nothing blocks me: Precise and real-time los/nlos path recognition in rfid systems," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 5814–5824, 2019.

[3] J. Li, G. Feng, W. Wei, C. Luo, L. Cheng, H. Wang, H. Song, and Z. Ming, "Psotrack: A rfid-based system for random moving objects tracking in unconstrained indoor environment," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4632–4641, 2018.

[4] Y. Wang and Y. Zheng, "Tagbreathe: Monitor breathing with commodity rfid systems," *IEEE Transactions on Mobile Computing*, vol. 19, no. 4, pp. 969–981, 2020.

[5] D. He and S. Zeadally, "An analysis of rfid authentication schemes for internet of things in healthcare environment using elliptic curve cryptography," *IEEE Internet of Things Journal*, vol. 2, no. 1, pp. 72–83, 2015.

[6] S. Amendola, R. Lodato, S. Manzari, C. Occhiuzzi, and G. Marrocco, "Rfid technology for iot-based personal healthcare in smart spaces," *IEEE Internet of Things Journal*, vol. 1, no. 2, pp. 144–152, 2014.

[7] Y. Zhang, S. Chen, Y. Zhou, Y. Fang, and C. Qian, "Monitoring bodily oscillation with rfid tags," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 3840–3854, 2019.

[8] P. Zhang, J. Liu, Y. Shen, H. Li, and X. Jiang, "Lightweight tag-based phy-layer authentication for iot devices in smart cities," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 3977–3990, 2020.

[9] J. Wang, D. Vasisht, and D. Katabi, "Rf-idraw: Virtual touch screen in the air using rf signals," in *Proceedings of the 2014 ACM Conference on SIGCOMM*, SIGCOMM '14, (New York, NY, USA), p. 235–246, Association for Computing Machinery, 2014.

[10] R. Li, T. Song, N. Capurso, J. Yu, J. Couture, and X. Cheng, "Iot applications on secure smart shopping system," *IEEE Internet of Things Journal*, vol. 4, no. 6, pp. 1945–1954, 2017.

[11] D. De Donno, F. Ricciato, L. Catarinucci, A. Coluccia, and L. Tarricone, "Challenge: Towards distributed rfid sensing with software-defined radio," in *ACM MobiCom*, 2010.

[12] H. Electronic, "Rfm69hw ism transceiver module v1.3." http://www.hoperf.com/upload/rf/RFM69HW-V1.3.pdf, 2018.

[13] RadioHead, "Radiohead packet radio library for embedded microprocessors." https://www.airspayce.com/mikem/arduino/RadioHead/, 2018.

[14] N. Kargas, F. Mavromatis, and A. Bletsas, "Fully-coherent reader with commodity sdr for gen2 fm0 and computational rfid," *IEEE Wireless Communications Letters*, vol. 4, no. 6, pp. 617–620, 2015.

[15] T. Wei and X. Zhang, "Gyro in the air: Tracking 3d orientation of batteryless internet-of-things," in *ACM MobiCom*, 2016.

[16] L. Yang, Q. Lin, X. Li, T. Liu, and Y. Liu, "See through walls with cots rfid system!," in *ACM MobiCom*, 2015.

[17] Y. Wang and Y. Zheng, "Modeling rfid signal reflection for contact-free activity recognition," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 2, pp. 193:1–193:22, Dec. 2018.

[18] L. Yang, Y. Chen, X.-Y. Li, C. Xiao, M. Li, and Y. Liu, "Tagoram: Real-time tracking of mobile rfid tags to high precision using cots devices," in *ACM MobiCom*, 2014.

[19] Y. Zheng and M. Li, "Towards more efficient cardinality estimation for large-scale rfid systems," *IEEE/ACM Trans. Netw.*, vol. 22, pp. 1886–1896, Dec. 2014.

[20] E. A. Keehr, "A low-cost software-defined uhf rfid reader with active transmit leakage cancellation," in *2018 IEEE International Conference on RFID (RFID)*, pp. 1–8, IEEE, 2018.

[21] P. V. Nikitin, S. Ramamurthy, and R. Martinez, "Simple low cost uhf rfid reader," in *2013 IEEE International Conference on RFID (RFID)*, pp. 126–127, IEEE, 2013.

[22] W. Taylor, S. A. Shah, K. Dashtipour, A. Zahid, Q. H. Abbasi, and M. A. Imran, "An intelligent non-invasive real-time human activity recognition system for next-generation healthcare," *Sensors*, vol. 20, no. 9, 2020.

[23] S. Liaqat, K. Dashtipour, K. Arshad, and N. Ramzan, "Non invasive skin hydration level detection using machine learning," *Electronics*, vol. 9, no. 7, 2020.

[24] M. Poh, N. C. Swenson, and R. W. Picard, "A wearable sensor for unobtrusive, long-term assessment of electrodermal activity," *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 5, pp. 1243–1252, 2010.

[25] M. Buettner and D. Wetherall, "A "gen 2" rfid monitor based on the usrp," *SIGCOMM Comput. Commun. Rev.*, vol. 40, pp. 41–47, June 2010.

[26] M. Buettner and D. Wetherall, "A flexible software radio transceiver for uhf rfid experimentation," *UW CSE Technical Report*, 2009.

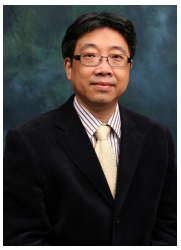
[27] E. A. Keehr, "A low-cost, high-speed, high-resolution, adaptively tunable microwave network for an sdr uhf rfid reader reflected power canceller," in *2018 IEEE International Conference on RFID (RFID)*, pp. 1–8, IEEE, 2018.

[28] Z. An, Q. Lin, and L. Yang, "Cross-frequency communication: Near-field identification of uhf rfids with wifi!," in *ACM MobiCom*, 2018.

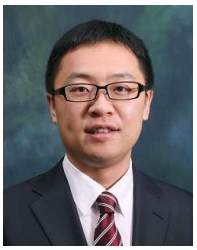
[29] S. M. Kim and T. He, "Freebee: Cross-technology communi-

cation via free side-channel," in *ACM MobiCom*, 2015.
[30] Z. Li and T. He, "Webee: Physical-layer cross-technology communication via emulation," in *ACM MobiCom*, 2017.
[31] J. Wang, D. Vasisht, and D. Katabi, "Rf-idraw: Virtual touch screen in the air using rf signals," in *ACM SIGCOMM*, 2014.
[32] X. Fan, W. Gong, and J. Liu, "Tagfree activity identification with rfids," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 2, pp. 7:1–7:23, Mar. 2018.
[33] L. Feng, Z. Li, C. Liu, X. Chen, X. Yin, and D. Fang, "Sitr: Sitting posture recognition using rf signals," *IEEE Internet of Things Journal*, pp. 1–1, 2020.
[34] L. Yang, Y. Li, Q. Lin, X.-Y. Li, and Y. Liu, "Making sense of mechanical vibration period with sub-millisecond accuracy using backscatter signals," in *ACM MobiCom*, 2016.
[35] H. Jin, J. Wang, Z. Yang, S. Kumar, and J. Hong, "Wish: Towards a wireless shape-aware world using passive rfids," in *ACM MobiSys*, 2018.
[36] E. Blossom, "Gnu radio: Tools for exploring the radio frequency spectrum," *Linux J.*, vol. 2004, pp. 4–, June 2004.
[37] J. Wang, H. Hassanieh, D. Katabi, and P. Indyk, "Efficient and reliable low-power backscatter networks," in *ACM SIGCOMM*, 2012.
[38] J. Ou, M. Li, and Y. Zheng, "Come and be served: Parallel decoding for cots rfid tags," *IEEE/ACM Trans. Netw.*, vol. 25, pp. 1569–1581, June 2017.
[39] P. Hu, P. Zhang, and D. Ganesan, "Laissez-faire: Fully asymmetric backscatter communication," in *ACM SIGCOMM*, 2015.
[40] M. Jin, Y. He, X. Meng, Y. Zheng, D. Fang, and X. Chen, "Fliptracer: Practical parallel decoding for backscatter communication," in *ACM MobiCom*, 2017.
[41] M. Jin, Y. He, X. Meng, D. Fang, and X. Chen, "Parallel backscatter in the wild: When burstiness and randomness play with you," in *ACM MobiCom*, 2018.
[42] D. Zanetti, B. Danev, and S. Capkun, "Physical-layer identification of uhf rfid tags," in *ACM MobiCom*, 2010.
[43] Y. Hou and Y. Zheng, "Phy-tree: Physical layer tree-based rfid identification," *IEEE/ACM Trans. Netw.*, vol. 26, pp. 711–723, Apr. 2018.
[44] L. Wu, P. Sun, Z. Wang, Y. Yang, and Z. Wang, "Toward efficient compressed-sensing-based rfid identification: A sparsity-controlled approach," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 7714–7724, 2020.
[45] Q. Lin, L. Yang, and Y. Guo, "Proactive batch authentication: Fishing counterfeit rfid tags in muddy waters," *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 568–579, 2019.
[46] T. Li, S. Chen, and Y. Ling, "Identifying the missing tags in a large rfid system," in *ACM MobiHoc*, 2010.
[47] Y. Zheng and M. Li, "P-mti: Physical-layer missing tag identification via compressive sensing," *IEEE/ACM Trans. Netw.*, vol. 23, pp. 1356–1366, Aug. 2015.
[48] H. Chen, G. Xue, and Z. Wang, "Efficient and reliable missing tag identification for large-scale rfid systems with unknown tags," *IEEE Internet of Things Journal*, vol. 4, no. 3, pp. 736–748, 2017.
[49] C. Wang, L. Xie, W. Wang, T. Xue, and S. Lu, "Moving tag detection via physical layer analysis for large-scale rfid systems," in *IEEE INFOCOM*, 2016.
[50] C. Jiang, Y. He, X. Zheng, and Y. Liu, "Orientation-aware rfid tracking with centimeter-level accuracy," in *ACM/IEEE IPSN*, 2018.

Yanwen Wang received the B.S. degree in electronic engineering from Hunan University, Changsha, China and the M.S. degree in electrical engineering from the Missouri University of Science and Technology, MO, USA, in 2010 and 2013, respectively, and the Ph.D. degree from the Department of Computing, Hong Kong Polytechnic University, Hong Kong, China, in 2020. He is currently an associate professor in College of Electrical and Information Engineering, Hunan University, China. His research interest includes mobile and network computing, RFID systems and wireless sensing. He is a member of IEEE and ACM.



Jiannong Cao received his M.Sc. and Ph.D. degrees in computer science from Washington State University. He is currently a Chair Professor with the Department of Computing at Hong Kong Polytechnic University. He is also the director of the Internet and Mobile Computing Lab in the department and the director of the University's Research Facility in Big Data Analytics. His research interests include parallel and distributed computing, wireless networking and mobile computing, big data and machine learning, and cloud and edge computing. He has co-authored 5 books, co-edited 9 books, and published over 500 papers in major international journals and conference proceedings. He is a fellow of IEEE.



Yuanqing Zheng Yuanqing Zheng received the B.S. degree in Electrical Engineering and the M.E. degree in Communication and Information System from Beijing Normal University, Beijing, China, in 2007 and 2010 respectively. He received the PhD degree in School of Computer Engineering from Nanyang Technological University in 2014. He is currently an Associate Professor with the Department of Computing in Hong Kong Polytechnic University. His research interest includes Mobile and Network Computing, Acoustic and Wireless Sensing, and Internet of Things (IoT). He is a member of IEEE and ACM.