IMPLEMENTING UHF RFID READER ON SMARTPHONE PLATFORM FOR IOT SENSING

Douglas Lautner, Xiayu Hua, Scott DeBates and Shangping Ren

Department of Computer Science, Illinois Institute of Technology,
Chicago, IL 60616, USA
Device+ Research and Technology, Lenovo Motorola,
Chicago, IL 60654, USA

ABSTRACT

As a core component of the Internet of Things technology (IoT), Radio Frequency Identification (RFID) tagged items will add billions, perhaps trillions, of objects to the Internet. As a result, uses of Ultra High Frequency (UHF) RFID sensing become massive ranging from logistics, retail and healthcare to homes and even entire smart cities. Under this trend, mobile UHF RFID scanners also need to evolve. Consumers will interact with their surroundings via tagged RFID items taking full advantage of the advancing IoT. For mainstream consumer smartphones, unfortunately, UHF RFID connectivity has yet to be fully integrated. The major challenges are: 1) the compatibility of an RFID reader module to the host platform, 2) Radio Frequency (RF) signal coexistence interference between the RFID reader and other sensor/RF technologies, and 3) the unacceptable high current drain caused by RFID active scanning. In this paper, we present a design and implementation of a novel modular UHF RFID scanning subsystem, the UHF RFID reader module, on a Motorola Moto-Z smartphone. This module is fully integrated with an Android 7.0 Operating System (OS) and directly interconnects with the low-level smartphone hardware and software framework. With the new antenna design and the signal spectrum analysis, we guarantee the RF isolation of the Mod with the smartphone's other native wireless components and sensors. Our design and implementation also address the current drain issue and extends the battery life of Moto-Z smartphone up to 30.4 hours with IoT RFID scanning.

KEYWORDS

UHF RFID, Smartphone design, mobile system architecture, mobile sensing

This work is supported by NSF CAREER 381211; NSF CNS 381367; and NSF CPS 381396.

1. INTRODUCTION

The advance of the Internet of Things (IoT) technology introduces new application scenarios to smartphone designers and manufactures. For example, using smartphones to gather information from nearby IoT objects through RFID and providing surrounding status to the user is one of the potentials. Thanks to the massive production of UHF RFID tags (from 8.9 billion tags sold in

Natarajan Meghanathan et al. (Eds) : CCSEA, NCOM, AIFU, DKMP, EMSA, SIPRO, SEA - 2018
pp. 127– 142, 2018. © CS & IT-CSCP 2018 DOI : 10.5121/csit.2018.80311

2015 to 10.4 billion in 2016 [1]), its price becomes affordable [2], hence transforming common everyday items of a wide variety, such as jeans or brushes, into IoT objects [3]. Likewise, smartphone shipments have increased every year since 2006 with an estimated global market of 2.1 billion units shipped in 2021 [4]. With high-performance computing power and constant Internet connectivity, nowadays smartphones are already capable to associate mobile computing with the IoT [5]. Once equipped with a UHF RFID reader, a smartphone is able to interact with the surrounding objects tagged by UHF RFID, gather and process information from those tags locally or on the internet, then represent feedback to the user via its multimedia user interfaces.

However, to-date there has been no successful widespread deployment of a fully integrated UHF RFID reader on mainstream smartphones. Common UHF RFID reader solutions are either stand-alone devices or are deployed as add-on implementation on smartphones that are not fully integrated with the hardware and software of a smartphone. For example, a UHF RFID reader on microSD which can be inserted into a smartphone through external memory slot was developed in [6]. Another solution utilizes the proprietary audio port to connect a UHF RFID reader with a smartphone [7]. Current integrated solutions, such as Invengo XC-1003 mobile IoT device and Chafon CF-H802 run on older Android platforms, are more industrial and are not widely distributed [8], [9]. Moreover, since hardware and software of these devices are specially designed, they are expensive and their functionality are monotonous. Through the research and prototyping experience in our preliminary works, we have identified three major challenges that causes this absence of RFID reader on a smartphone:

1) No existing interface that enables the UHF RFID reader to establish a direct high-speed connection to the lower level smartphone hardware without significant modifications on the phones hardware framework.

2) Severe signal interferences amongst different RF sensors and communication modules while the RFID reader antenna is active.

3) Unbearable current drain to a consumer smartphone when the UHF RFID reader is up and running.

In this work, we address the three challenges and empower a commercial smartphone with UHF RFID reading capability. Specifically, we design and implement a fully functional consumer UHF RFID Reader Mod (URRM), as illustrated in Fig. 1, based on a mainstream Android-driven smartphone, Motorola Moto-Z, which has been ”sold around the world” [10]. By leveraging the advantage of the unique modularization features of Moto-Z, i.e. the support of third-party Mods [11], we are able to utilize the GreyBus protocol to connect our RFID reader module, i.e. the Impinj RS500 RFID module, to the phones lower-level hardware framework without changing the phones native hardware. Comparing to add-on implementations that connect through USB, audio jack or memory card ports [9,10], direct connection to lower-level hardware framework provides with high energy-efficiency, low communication overhead as well as wider applicable system support such as power management or chip-level control and scheduling. To guarantee URRMs RF performance and avoid interference with phones native wireless functionalities, we design an antenna for the URRM which isolates to the phones native wireless signal system. We further utilize the power-efficient sensor fusion core on the URRMs hardware infrastructure to manage the scanning and data processing tasks. We minimize the impact of the high current drain of RFID scanning to the phones battery life by optimizing the control and management system and coupling to an external power source. Our experiments prove that our prototype overcomes these three challenges and hence becomes practical and ready-to-manufacture. In addition, our design and implementation of the key hardware and software architecture is extensible and scalable for continued research. Other IoT sensors on different mobile platforms, such as WiFi or Bluetooth modules on NuttX-driven Rasperry Pi platforms, can also apply our solutions.

The rest of the paper is organized as follows. We discuss related work in Section II. Section III presents the design and implementation of the portable UHF RFID reader architecture. In Section IV we study the radio performance of the UHF RFID reader Mod with respect to signal isolation, sensing range and sensitivity. We conclude in Section V.

2. RELATED WORK

The fast growing IoT technology and its expanding scope of applications demand for a wide spectrum of both the sensor type and the sensing content [12]. In [13], researchers developed CARISMA (Context-Aware Reflective Middleware System for Mobile Applications), a software framework for IoT contextual sensing and awareness tasks on mobile systems. This framework is generic with respect to mobile operation systems and sensor types, hence it can be implemented on different smartphone platforms and work with different types of sensors. Similar middleware system, the SOCAM (Service Oriented Context-Aware Middleware), was also developed by researchers in [14]. To utilize the advantages of heterogeneous sensing sources in IoT scheme, the e-SENSE [15] mobile sensing system enables ambient intelligence using wireless multi-sensor networks for making IoT oriented context-rich information available to applications and services. e-SENSE combines body sensor networks (BSN), object sensor networks (OSN), and environment sensor networks (ESN) to capture context in the IoT paradigm. Later, Hydra3 [16] was developed as an IoT middleware that aims to integrate wireless devices and sensors into ambient intelligence systems. Hydra3 identifies context reasoning rule engine, context storage, context querying, and event/action management as the key components of a context-aware framework. In the IoT sensing scheme, RFID reader plays a key role. In [17], researchers illustrate eleven different categories of use cases that need to utilize RFID technology to interact with the IoT surroundings. In [18], researchers introduce a semi-passive, reconfigurable UHF RF identification (RFID) sensing tag operating as the generic sensing platform (GSP). The tag is highly configurable and can be dynamically switched between a Continuous data transmitting platform (Online mode) or a Data logging platform (Offline mode). Researchers in [19] develop a suite of web-based, user-level tools and applications designed to empower users by facilitating their understanding, management, and control of personal RFID data and privacy settings. Besides researches on RFID technology itself, research topics of implementing RFID based IoT systems on other application fields, such as massive manufacture workflow optimization [20], Geofencing item tracking [21] and industrial-level quality control [22], have drawn increasing attention not just from computer developers and researchers, but from all the industrial and science community. IoT is advancing fast, RFID and their combined technologies are also becoming ubiquitous and playing a critical role of identifying objects. However, smartphones, one of the most convenient and widely used portable computing devices, is still out of the picture of research on RFID based IoT systems. Although it matches all the requirements from both the hardware and the software perspective, severe RFID integration challenges remain.

3. SYSTEM DESIGN AND IMPLEMENTATION

The core concept of our system design is the URRM and its control and support components. In particular, our system architecture design contains five subsystems: the UHF RFID reader Mod (URRM) subsystem, the Mod control and management subsystem, the Mod support subsystem, the Battery subsystem and the Antenna subsystem. The following subsections discuss the five subsystems in details, followed by a discussion of the system workflow.

A. UHF RFID Reader Mod Subsystem

To embed a RFID reader module into the low-level of a native smartphone system, the major challenge is to establish a physical connection from the RFID module directly to the low-level smartphone system without significantly modifying the native system. We achieve this goal by designing the URRM for a Moto Z smartphone, both of which are showed in Fig. 1, respectively. In particular, we embedded the RFID reader module into the MotoMod platform [23] with a HAT (Hardware Attached on Top) adapter board [24]. Due to the modularization design of the Moto-Z mobile system, the URRM utilizes GreyBus [25] protocol to communicate the Moto Zs hardware layer through Serial Peripheral Interface (SPI) once the URRM is attached to a Moto Zs backboard. The picture and the brief architecture of the Moto-Z system and the URRM are depicted in Fig.1(a) and Fig.1(b), respectively.

(a) MotoMod and Moto Z products (b) MotoMod and Moto Z structure

Fig. 1. Moto Mods and Moto Z

GreyBus is developed by Google and Motorola in 2014, it supports hot plug and play communications between the GreyBus Master and GreyBus Slave. The Moto Zs Mod physical hardware interface contains 16 metal touch points which supports USB2.0, USB3.1, SPI, Mobility Display Port (MyDP), Inter-IC Sound (I2S) and other mainstream communication protocols. Up to 15 Mps of high speed data exchange with a dedicated transmit and receive Direct Memory Accesses (DMA) between Master and Slave devices. To drive up the reader module, we use Universal Synchronous/Asynchronous Receiver/Transmitter (UsART) to port the reader Mod to the HAT board. To improve the energy efficiency, we further design a two-layer architecture which utilizes both the MuC on the MotoMod, i.e., the Mod Control and Management Subsystem, and the Application Processor (AP) on the Moto-Z, i.e., the Mod Support subsystem, to control and manage the RFID reader module jointly. Moto-Z smartphone system is equipped with a

SnapDragon 820 CPU as its AP which has 2.2GHz main frequency and 512 KB + 1 MB L2 cache. It is designed to support the latest version of Android system, i.e., Android 8.0, and smartphone applications. However, the current drain is high (28.8mA in full performance and 4.8mA in idle state). For scan-related tasks such as conducting IoT sensing or processing the raw data, the AP is over-powerful and hence becomes very energy-inefficient. The MuC on the MotoMod, on the other hand, has higher energy-efficiency (4.3mA current drain in full performance and 1.1uA in idle state). This system is powered by a STM-32L4 based core that has 80MHz main frequency and 1MB flash memory. It's computing capacity is less powerful comparing to the AP, but it is sufficient to handle the scanning routings such as sending commands or parsing events to or from the RFID module. In our stress test, the MuC can process no less than 1200 RFID tags per second without depleting its computing power and buffer. In our design, we keep computing and control oriented tasks of UHF RFID sensing, such as the scanning routings or raw data process procedures, on the MCU and create the Mod Control and Management subsystem. For tasks that can only run on Moto Z, such as the scan application with Graphic User Interface (GUI), we shift them to the AP and create Mod support subsystem. Both subsystems are discussed in the following two subsections.

B. Mod Control and Management Subsystem

The MuC and its baseline software framework [26] are the core components of the MotoMod system and provide Mod developers a POSIX and ANSI standards develop environment. The software framework is built upon a real-time operating system, the NuttX Real-time operation system (NuttX) [27], to provide basic I/O control, task scheduler, power management and drivers for specific circuits running on the MotoMod. In particular, the slave-end GreyBus protocol is implemented in the NuttX driver layer and is abstracted as a virtual file. All the data transmissions between Moto Z and MotoMod are implemented through reading or writing this virtual file. The system architecture is illustrated in Fig.1(b).

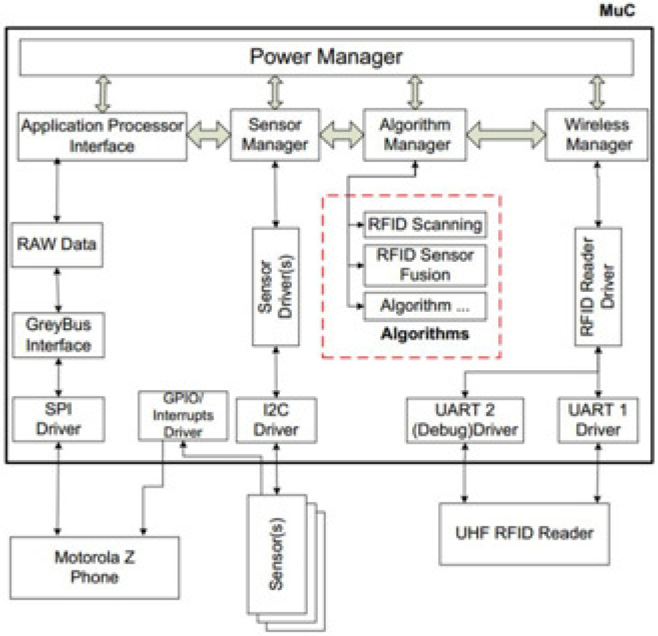


Fig. 2. Sensor Fusion Architecture

To control the RFID reader module, we deploy the sensor fusion architecture modified from in our previous works [28], [29], which is illustrated in Fig.2, on the baseline software framework. The core components of the sensor fusion architecture are: Power Manager, Sensor Manager, Algorithm Manager, Application Processor Interface and Wireless Manager. We specify their functions separately.

Power Manager: responsible for managing the tasks power mode votes and configuring the MuC's power mode based on the votes. The power manager also controls and monitors the status of the Mods battery.

Sensor Manager: responsible for configuring the sensors and managing the raw sensor data obtained from the sensors.

Algorithm Manager: responsible for configuring, scheduling, and providing the raw sensor data for all data processing algorithms.

Application Processor Interface: responsible for all communications between the sensor fusion core and the application processor. Its tasks include managing features turned on/off by the application processor, and interrupting applications when sensor data is available.

Wireless Manager: responsible for configuring the UHF RFID reader IC, and write/reading data of surrounding UHF RFID tags.

The design of the Mod Control and Management subsystem has two advantages. First, the architecture is extensible which means not only the RFID reader, but other similar wireless technologies, such as Bluetooth, WiFi, and NFC, can be deployed on the Mod and can work jointly following the same approach. Second, all the drivers and implementation details are encapsulated on the MuC and are transparent to the smartphone, which means even a URRM with different configurations, like with a different RFID reader module, is attached on, systems on the smartphone are not affected.

C. Mod Support Subsystem

On the smartphone side, we design a vertical system framework running on Android system which communicates through from the top (Application) layer of the Android system to the bottom (GreyBus protocol over the SPI transport) layer of the smartphone system. All functionalities of the whole framework are encapsulated and embedded with an API library, the Motlib, and are exposed to the Androids application layer. Our application, as well as any other third-party applications, utilizes the framework by simply including the Motlib library.

The architecture of the vertical system, including the Mod support subsystem and Mod control and management subsystem, are illustrated in Fig.3. For the Mod support subsystem,each component in this framework from top layer to bottom layer is discussed as follows:

RFID Application an Android application with a GUI that allows users to interact with the URRM by sending commands and displaying the reader Mod status and scan results. This application utilizes functions in the Motlib library to interact with the Mod Manager layer.

Mod Manager: The Mod Manager is a component of the Hardware Abstract Layer (HAL) in the NuttX system. It encapsulates the implementation details of hardware under NuttX and maps system calls from the Application Layer to the device driver functionalities. In our case, the URRM is registered to the Mod Manager with its all functionalities. Plus, Mod Manager can directly update the firmware of the MuC in URRM.

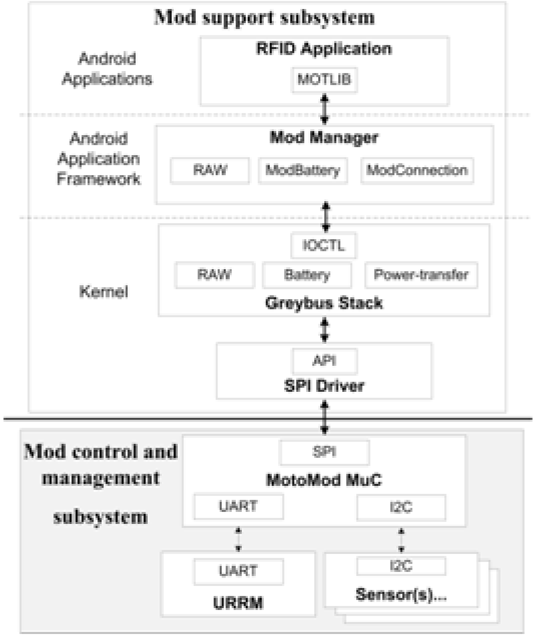


Fig. 3. Architecture of URRM and Mod Support Subsystems

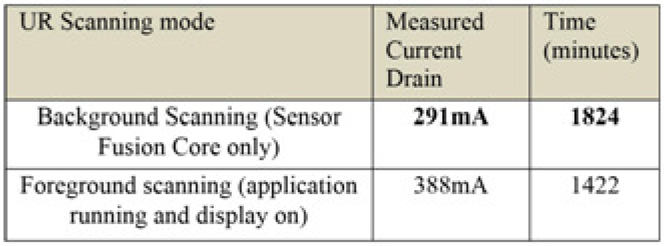
GreyBus Stack (master end): GreyBus is running on the Kernel layer of the Android system. GreyBus defines multiple class protocols that are used to communicate and provide function to specific hardware. In our design, we enable three protocols: RAW, Battery and Power Transfer Protocol (PTP) which enable us to use our own designed communication package format (RAW), enable URRM to manage its own battery (Battery) and enable Moto Z can provide power to URRM (PTP).

SPI Driver: SPI Driver is running in the Android kernel. It controls the SPI Bus which is the physical communication interface between the MotoMod and Moto-Z mobile device for GreyBus communications to the MuC.

This design has two advantages. First, it allows Android applications to flash the MuC's firmware, either for driver updates or for upgrades of functionality, algorithms or even for the baseline system. This feature brings more flexibility and convenience to the application developer since without this function, such update or upgrade procedures can only be done manually by opening up the Mods enclosure and flash new firmware through J-tag or CC-debugger that is attached to the MuC. Second, the power management function in the Mod Support subsystem allows us to embed the external power source in the URRM and enable URRM to gain power supply from either the external power source or the Moto Z. The detailed implementation of the external power source is discussed in the next subsection.

D. Battery Subsystem

In our design, the RFID Mods MuC is responsible for controlling the power management. We use a MAX17050X chipset for monitoring the battery when in-use and charging. To maximize device power when RFID scanning is active, we couple a regenerative 3.7 volt battery with a capacity of 1500 mAH in the RFID Mod with the Moto Z through the MuC controller. Total energy capacity available is 5000 mAH when combined with Moto Z's native battery capacity of 3500mAH. Utilizing the MuC's power management component, our configuration uses the Mods battery until the battery level reaches 5%, at that point the MuC will switch in the Moto Zs battery for continuous use ensuring no degradation in performance.



UR Scanning mode	Measured Current Drain	Time (minutes)
Background Scanning (Sensor Fusion Core only)	**291mA**	**1824**
Foreground scanning (application running and display on)	388mA	1422

Fig. 4. Battery Life Evaluation

The method we use to measure the battery consumption is a NI USB-6251 DAQ to monitor current consumption at the battery contacts of the Moto Z. Once the DAQ is connected, we place the phone into an airplane mode to disable all wireless processing, and measure the current consumption with the UHF RFID application launched but not on. This provides us the idle screen current drain baseline measurement. We then enable the UHF RFID reader Mod. The configuration of the scanning is 10sec on and 10sec idle. The results in Fig.4 show that our design is capable of scanning for an entire day in the background. As we continue to develop and refine existing algorithms we anticipate further reductions in battery consumption.

E. Antenna Subsystem

When active, the reader Mod actively project energy to the RFID receiver, such as RFID tags, and capture their echoes as responses. Therefore, the performance of the reader as well as the interference amongst other wireless communication modules could be severe and becomes one of the main blockage for the deployment of RFID readers on smartphone platform. We provide our antenna design here and will evaluate the antennas performance in the next section.

We design the antenna to operate at 902MHz-928MHz, and an ambient distance of one meter for small passive tags, and three meters for active, or larger UHF RFID tags. We use Rogers 3010 material [30] to reduce the effects of increasing the Z height in our design. The antenna dimensions are 80x52x1.2mm. The feed impedance of the antenna was 50 Ohms. The direction of the antenna's radiation is perpendicular to the Moto Z's plane, and the polarization is across the Moto Z's antennas. The isolation between the UHF RFID antenna and the Moto Z's communication frequencies is larger than 30dB. The radiation efficiency of our design from 902MHz-928MHz was 5 - 10%.

F. System Workflow

Once the UHF RFID reader Mod is attached to the back of the Moto Z, the Mod transmits its manifest file from the MuC to the Motorola Mod Manager. The Motorola Mod Manager then parses the manifest file and determines what type of Mod is connected and its communication capabilities to expose for applications use. After the reader Mod is registered, applications can use Motorola Mod Manager to send RFID commands over the RAW protocol to the Mod. Once the reader Mod receives the RAW packet it will parse the data and determine which Manager (Sensor, Algorithm, Wireless) gets the payload. The Manager will then determine which driver or algorithm to send the payload. The driver or algorithm then parses the payload to perform the requested action. Upon a completion of the requested action a returned status (success or failure) is returned to the application.

4. URRM SIGNAL FREQUENCY ISOLATION ANALYSIS

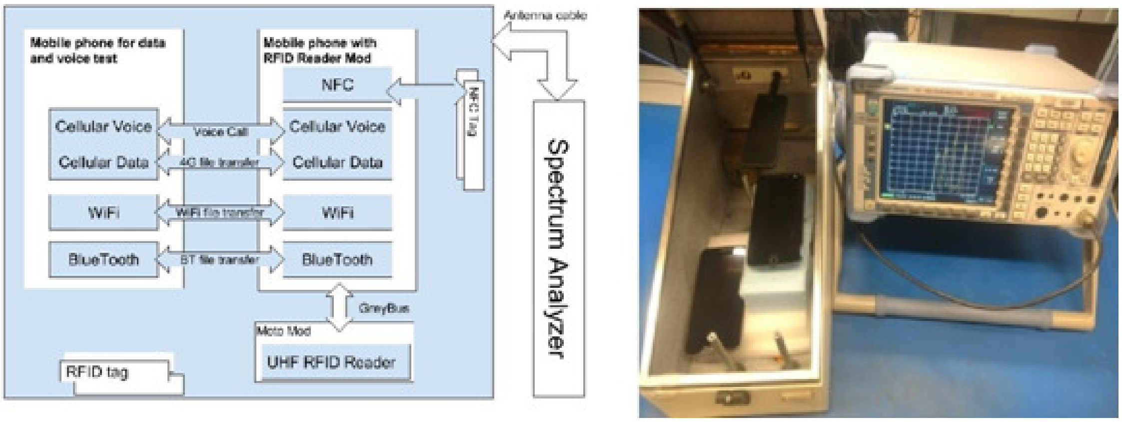
As a part of the smartphone wireless system, the reader Mod works in a very close range with other native sensors and wireless modules. Therefore, the radio signal interference becomes a key feature which determines if the reader Mod is practical for commercial use or is just a prototype. Moreover, the radio performance of the reader Mod is another key feature as a sensor that needs to be evaluated, too short sensor range or too low sensitivity may significantly reduce the applicable range of the reader Mod in real-world scenarios. In the following subsections, we experimentally evaluate these two key features.

A. RF Signal Interference Validation

In Section I, we discuss about the antenna design in the hardware level that isolates the RFID signal from other sensors. In specific, as antennas of different wireless components are tightly coupled in a small enclosure, we need to confirm that our new RFID reader has no 1) Radiated spurious emissions [31] and 2) Inter-modulation distortion [32], [33]) against the components in the original system. In this section, we experimentally proof the signal isolation effectiveness of our design from the radio spectrum perspective.

Our experiment testbed is setup as follow. We use a RF shielded box (Will Technology model SH-120) to provide a signal-free environment and use two developer version Moto Z smartphones to perform data transmission via different wireless modules. One smartphone equips with the RFID reader Mod. A spectrum analyzer (Rohde & Schwarz model FSP13) is connected to the RF shield box to monitor and record the RF signals in the box from frequency 13Mhz to 6Ghz. Another antenna is wired through the shield and is implanted into the enclosure to provide cellular signal. The experiment testbed design is shown in Fig.5(b) and its setup illustrated in Fig.5(b).

To perform the interference test on a wireless module or sensor, we take the following steps. First, we place RFID tags into the RF shielded enclosure. Then, we engage the RF modules on both smartphones and start transferring data from one smartphone to another or to the implanted antenna. The data volume is 10G, hence its transmission provides us enough time to place both phones into the RF shield box and complete the setup. In the third step, we engage the reader Mod and keep it scanning the RFID tags. We then close the shield box and record the spectrums with and without the reader Mod. Following sections illustrates the test results for each wireless module/sensor, respectively. We label the start, peak and end of the spectrum with green dot and specific frequency and decibel values.



(a) Tested design (b) Tested setup

Fig. 5. Testbed for Frequency Isolation Analysis

B. Radio Spectrum of RFID Scanning Signal

In Fig.6 we show our design that is active at frequency band ranging from 864 MHz to 944MHz with energy peak at 912MHz. The strength at the peak is 15 dB at distance zero meters.

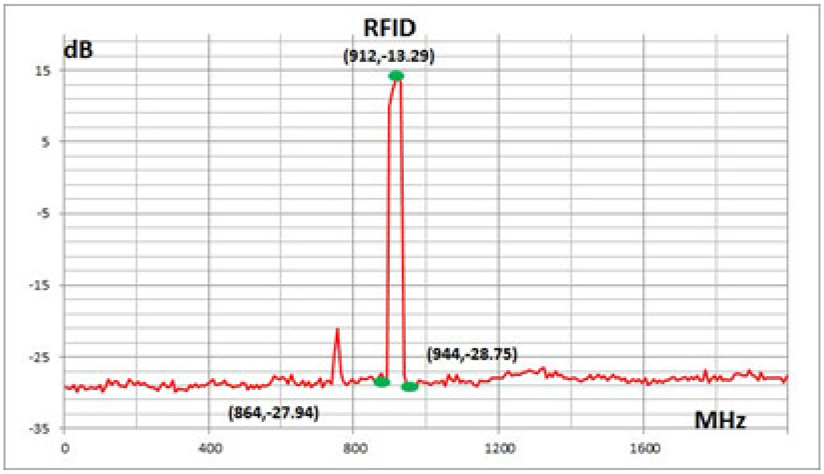
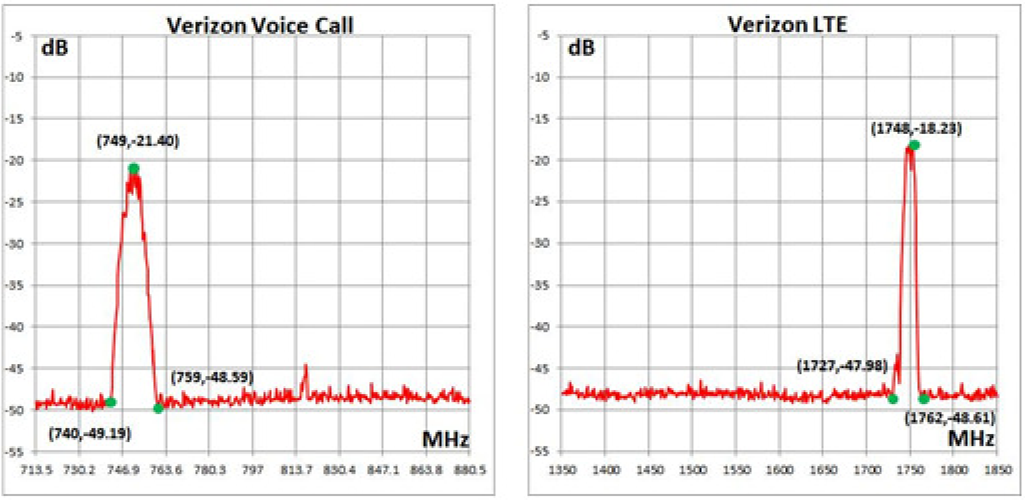


Fig. 6. UHF RFID Spectrum

1) Cellular Signal Interference: Testing for Cellular Interference is performed by connecting the Moto-Z with our UHF RFID reader Mod, and a live Verizon SIM. In both cases, with or without engaging the RFID reader, the radio spectrum of the Cellular voice call signal remains the same, as illustrates in Fig.7(a), which indicates that there is no interference between RFID reader signal and the Cellular signal. For the Cellular data streaming, there is also no interference. The signal spectrum of data streaming with and without RFID reader

scanning both follow the pattern depicted in Fig.7(b), all datasets streamed where received without retransmission.



(a) Celluar Voice (b) Celluar LTE

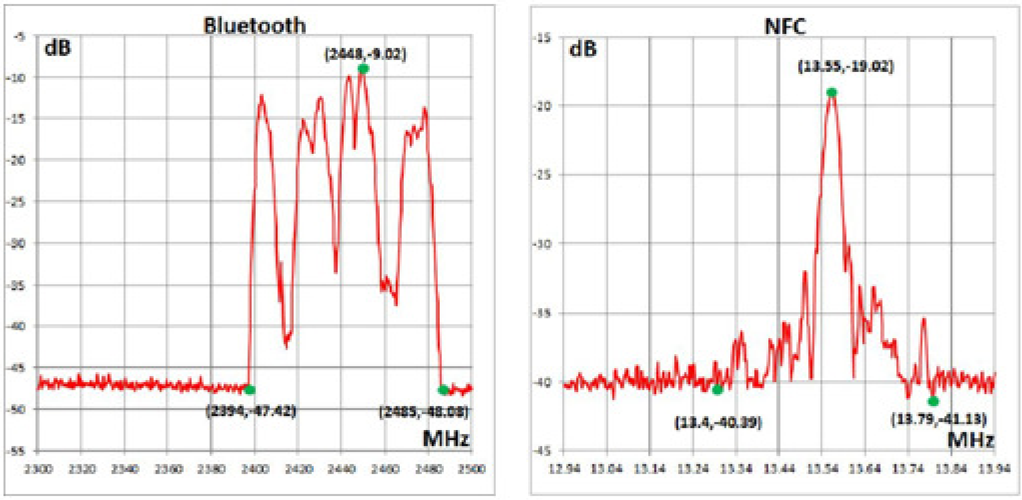
Fig. 7. Frequency analysis I

2) Bluetooth Signal Interference: To test Bluetooth interference, we capture the signal spectrums of all four stage of BT communications, i.e., BT scanning, advertising, paring and data transmission. The accumulated peak spectrums of all stages of BT communication, both with and without RFID scanning activity, are followed the same pattern depicted in Fig.8(a). The test results in Fig. 8 show no signal interference with Bluetooth, and we do not find transmission errors with the dataset received on the PC side, which indicate there is no interference between BT and RFID reader Mod.

3) Near Field Communication (NFC) Signal Interference : Testing for NFC interference is performed by connecting the Moto Z with Mod, and using a NXP NFC tag. While reading the tags memory, RFID background scanning is running. The test results in Fig.8(b) show no signal interference with NFC, and neither errors when reading or writing to the NFC tag.

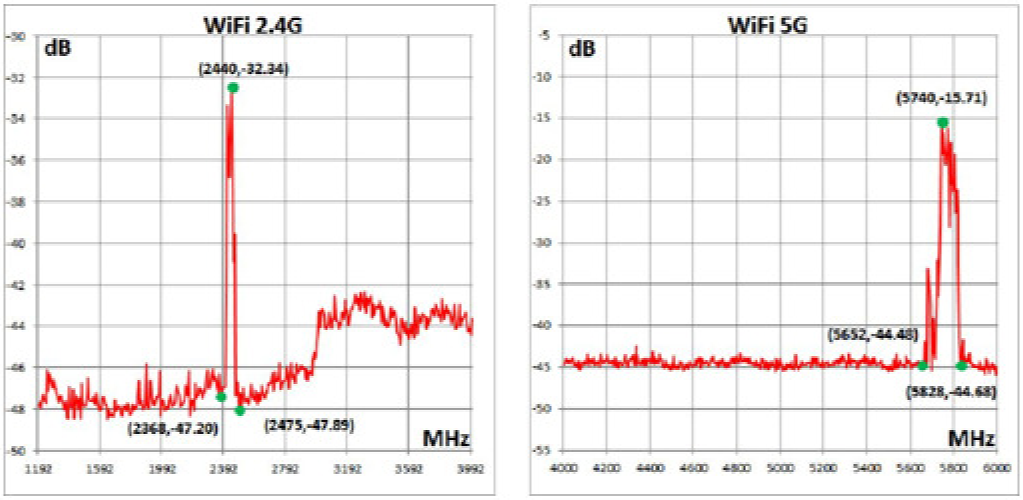
4) WiFi Signal Interference: For WiFi test, we measure the two modes of WiFi communication: in 2.4G channel and in 5G channel. Both test are performed by set one smartphone as a hotspot and use another one to connect to the hotspot. Experimental data shows that for both modes, WiFi communication has no interference to the RFID reader Mod. The specific spectrum data are illustrated in Fig.9(a) and Fig.9(b) for 2.4G mode and 5G mode, respectively.

As illustrated in the five experiments above, our UHF RFID reader Mod has no interference on mainstream smartphone connectivity systems, which means our design is able to coexist with the native wireless sensors and communication modules.



(a) Bluetooth (b) NFC

Fig. 8. Frequency analysis II



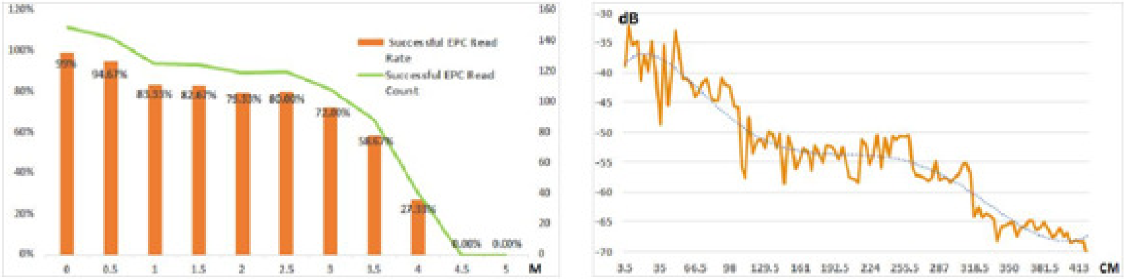
(a) WiFi in (b) WiFi in 5GHz

Fig 9. 2.4GHz Frequency analysis III

C. Sensor Performance Evaluation

As a IoT sensor, the sensing range and the signal attenuation are two key parameters which determine the quality of service of IoT sensing. In the following, we design two experiments to evaluate these two features, respectively. For these two experiments, the environmental temperature is 18.3 Celsius, the target RFID tag is SMARTRAC belt RFID paper tag [34] and the experiment lab is covered with radio absorb material to eradicate the radio reflection from wall faces. Battery is charged to 70%. Each value is the average value of 50 readings.

Our first experiment determines the sensing range of the UHF RFID reader Mod. From range 0 meter to 5 meters, we use our Mod to send 150 scanning signals at each distance and count the successful Electronic Product Code (EPC) reads from RFID tag. Fig.10(a) illustrates the Successful EPC read count/rate of a RFID tag at different distances. As the picture shows, the signal picking up ability reduces when the distance increases but still very reliable before 3.5 meters. From 3.5 meters to 4.5 meters, it degrades dramatically and tags with range longer than 4.5 meter cannot be recognized. We determine the sensing range as 4 meters where the expected retry time is four, which is acceptable to most IoT sensing scenarios.



(a) EPC Read Evaluation Result (b) Attenuation Evaluation

Fig. 10. Evaluation Result

Our second experiment evaluates the signal sensitivity and attenuation of our reader Mod. In this experiment, we measure the RSSI (Received Signal Strength Indicator) provide from the reader IC from range 0 Meter to 4.2 Meter. The result is depicted in Fig.10(b) where the solid line is the actual value and the dot line is the trendline. With the result from the previous experiment, the attenuation distribution shows that the reader Mod has stable radio performance when signal strength is higher than -65 dB (3.5 meters) and signals that are weaker than -70dB cannot be detected. One interesting finding is that, in average, the attenuation increases with the grows of the distance, but it also shows dramatic decrease in some distance pints. In our case, there are three: 0.97 Meter, 3.18 Meter and 4.21 Meter. We are not able to explain this phenomenon for now, but we will further study it in our future work.

5. CONCLUSION

In this paper, we design and implement a UHF RFID reader Mod with direct connection to a mainstream smartphone, Moto-Z, without changing the hardware of the phone. The experimental data proves that our design of the UHF RFID reader Mod can coexist with other smartphone sensors and communication components with no conflict. Also, the battery life with active RFID scanning is extended significantly due to the control system and the external power source design.

Our research and architecture has been tested for commercial deployment and we have high-lighted the key hardware and software architecture components which are extensible and scalable for continued research.

Through our work, mobile devices will take advantage of UHF RFID capabilities by evolving into sensory data accumulators passing raw and fused sensor states to cloud infrastructure for further compute-intensive processing. In our future work, we will optimize the algorithms running on the sensor fusion core to further reduce the power consumption to the acceptable range enabling us to remove the external power source. Next we will introduce RFID as a Sensor (RaaS), providing even wider sensing spectrum and more comprehensive environmental information. Coupled with a highly optimized Sensor Fusion Core (SFC) and Bluetooth Low Energy (BLE) these sensors jointly provide a real-time, always-on and comprehensive sensory system optimized for a mobile devices limited battery life. Optimized wireless monitoring algorithms will be developed minimizing power consumption while guaranteeing an applications Quality of Service (QoS) requirements. We will also further study the signal sensativitiy and attenuation behaviors when the reader Mod is activated.

REFERENCES

[1] IDTechEx, R. Das, and P. Harrop, RFID forecasts, players and opportunities 2011-2021. IDTechEx, 2011.

[2] L. Yang, A. Rida, R. Vyas, and M. M. Tentzeris, "Rfid tag and rf structures on a paper substrate using inkjet-printing technology," IEEE Transactions on Microwave Theory and Techniques, vol. 55, no. 12, pp. 2894–2901, 2007.

[3] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," Computer networks, vol. 54, no. 15, pp. 2787–2805, 2010.

[4] B. Intelligenceo, "The global smartphone report: The forces behind the global deceleration in smartphone sales," http://www.businessinsider.com/, 2016.

[5] A. Kamilaris and A. Pitsillides, "Mobile phone computing and the internet of things: A survey," IEEE Internet of Things Journal, vol. 3, no. 6, pp. 885–898, 2016.

[6] S. Park, "A development of uhf rfid device for mobile iot service," Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON), 2016.

[7] A. M. Requist and C. A. Requist, "Method for connecting a portable rfid reader to a handheld computing device via an audio connection," Mar. 14 2017, uS Patent 9,594,932.

[8] http://www.invengo.com/, "Invengo users manual," 2014.

[9] http://chafon.com/, "Chafon users manual," 2014.

[10] R. Triggs, "Lenovos sold 1 million moto z handsets globally," http://www.androidauthority.com/1-million-moto-z-sales-727170/, 2016.

[11] M. Mobility, https://developer.motorola.com/, 2016.

[12] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context aware computing for the internet of things: A survey," IEEE Communications Surveys & Tutorials, vol. 16, no. 1, pp. 414–454, 2014.

[13] L. Capra, W. Emmerich, and C. Mascolo, "Carisma: Context-aware reflective middleware system for mobile applications," IEEE Transactions on software engineering, vol. 29, no. 10, pp. 929–945, 2003.

[14] T. Gu, H. K. Pung, and D. Q. Zhang, "A service-oriented middleware for building context-aware services," Journal of Network and computer applications, vol. 28, no. 1, pp. 1–18, 2005.

[15] A. Gluhak and W. Schott, "A wsn system architecture to capture context information for beyond 3g communication systems," in Intelligent Sensors, Sensor Networks and Information, 2007. ISSNIP 2007. 3rd International Conference on. IEEE, 2007, pp. 49–54.

[16] A. Badii, M. Crouch, and C. Lallah, “A context-awareness framework for intelligent networked embedded systems,” in Advances in Human-Oriented and Personalized Mechanisms, Technologies and Services (CENTRIC), 2010 Third International Conference on. IEEE, 2010, pp. 105–110.

[17] X. Jia, Q. Feng, T. Fan, and Q. Lei, “Rfid technology and its applications in internet of things (iot),” in Consumer Electronics, Communications and Networks (CECNet), 2012 2nd International Conference on. IEEE, 2012, pp. 1282–1285.

[18] M. S. Khan, M. S. Islam, and H. Deng, “Design of a reconfigurable rfid sensing tag as a generic sensing platform toward the future internet of things,” IEEE Internet of things journal, vol. 1, no. 4, pp. 300–310, 2014.

[19] E. Welbourne, L. Battle, G. Cole, K. Gould, K. Rector, S. Raymer, M. Balazinska, and G. Borriello, “Building the internet of things using rfid: the rfid ecosystem experience,” IEEE Internet computing, vol. 13, no. 3, 2009.

[20] M. Liukkonen and T.-N. Tsai, “Toward decentralized intelligence in manufacturing: recent trends in automatic identification of things,” The International Journal of Advanced Manufacturing Technology, vol. 87, no. 9-12, pp. 2509–2531, 2016.

[21] K. Ding, P. Jiang, P. Sun, and C. Wang, “Rfid-enabled physical object tracking in process flow based on an enhanced graphical deduction modeling method,” IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2017.

[22] F. Bibi, C. Guillaume, N. Gontard, and B. Sorli, “A review: Rfid technology having sensing aptitudes for food industry and their contribution to tracking and monitoring of food products,” Trends in Food Science & Technology, 2017.

[23] M. M. LLC, “Bring your moto mod from concept to market,” in https://enterprise.indiegogo.com/motomods. Motorola Mobility, 2016.

[24] M. Mobility, “Mdk user guide: Hat adapter board,” https://developer.motorola.com/build/mdk-user-guide/hat-adapter-board, 2016.

[25] A. ray, “Greybus protocol an application layer for unipro,” Project Ara Developers Conference, http://www.modularphonesforum.com/news/greybus-protocol-an-application-layer-for-unipro-491/, 2015.

[26] M. Mobility, “Moto mods system architecture,” https://developer.motorola.com/explore/system-architecture, 2016.

[27] G. Nutt, “Nuttx operating system user s manual,” 2014.

[28] D. Lautner, X. Hua, S. Debates, M. Song, J. Shah, and S. Ren, “Baas (bluetooth-as-a-sensor): conception, design and implementation on mobile platforms,” in Proceedings of the Symposium on Applied Computing. ACM, 2017, pp. 550– 556.

[29] D. Lautner, S. DeBates, J. Shah, M. Song, and S. Ren, “Sensor-based low power management for mobile platforms,” in High Performance Computing and Communications (HPCC), 2015 IEEE 7th International Symposium on Cyberspace Safety and Security (CSS), 2015 IEEE 12th International Conferen on Embedded Software and Systems (ICESS), 2015 IEEE 17th International Conference on. IEEE, 2015, pp. 934–940.

[30] J. C. Rautio, “Measurement of uniaxial anisotropy in rogers ro3010 substrate material,” in Microwaves, Communications, Antennas and Electronics Systems, 2009. COMCAS 2009. IEEE International Conference on. IEEE, 2009, pp. 1–4.

[31] Q. Yu, “Radiated spurious emissions measurement by substitution method,” in Electromagnetic Compatibility (EMC), 2010 IEEE International Symposium on. IEEE, 2010, pp. 159–164.

[32] C. H. Z. X. S. Linfang, “Inter-modulation distortion analysis for single-drive mach-zehnder modulator [j],” Acta Optica Sinica, vol. 11, p. 004, 2006.

[33] J. McRory and R. Johnston, “An rf amplifier for low inter-modulation distortion,” in Microwave Symposium Digest, 1994., IEEE MTT-S International. IEEE, 1994, pp. 1741–1744.

[34] K. Ding, P. Jiang, P. Sun, and C. Wang, “Smartrac belt rfid paper tag,” https://www.smartrac-group.com/belt.html, 2017.