On the Use of Radio Frequency Identification for Continuous Biomedical Monitoring

William M. Mongan Drexel University Philadelphia, Pennsylvania 19104 wmm24@drexel.edu

Shrenik Vora Drexel University Philadelphia, Pennsylvania 19104 shrenik.vora@drexel.edu Ilhaan Rasheed Drexel University Philadelphia, Pennsylvania 19104 ir54@drexel.edu

Kapil Dandekar Drexel University Philadelphia, Pennsylvania 19104 krd26@drexel.edu

Timothy Kurzweg Drexel University Philadelphia, Pennsylvania 19104 kurzweg@coe.drexel.edu

ABSTRACT

Radio Frequency Identification (RFID) technology is often deployed for inventory management scenarios. In inventory applications, a known or unknown number of RFID tags are queried in a discrete manner and for a single, short period of time, until each tag is recognized by the interrogator device. Passive RFID provides several benefits conducive to ubiquitous deployment, including RFID tags that are energized from the wireless RF interrogation signal itself that obviates the need for a battery or wired power, and antenna assemblies that can be integrated with the chip with only a small footprint. We have utilized these benefits to enable continuous biomedical sensing devices with minimal footprint and batteryless deployment. These devices are fabric-based smart garments with an embedded RFID tag and antenna assembly. However, traditional inventory-based RFID interrogation presents several challenges due to the RFID protocols and regulations that govern their use. In this paper, we discuss the considerations necessary to utilize RFID interrogation to enabling passive, continuous sensor monitoring, and the techniques we employed in developing software to do so.

IoTDI 2017, Pittsburgh, PA USA

DOI: http://dx.doi.org/10.1145/3054977.3055002

Drexel University Philadelphia, Pennsylvania 19104 khyati.u.ved@drexel.edu

Khyati Ved

Genevieve Dion Drexel University Philadelphia, Pennsylvania 19104 gd63@drexel.edu

Adam Fontecchio Drexel University Philadelphia, Pennsylvania 19104 af63@drexel.edu

CCS CONCEPTS

Computer systems organization →Sensor networks;
Hardware →Digital signal processing; •Software and its engineering →Software design engineering;

KEYWORDS

IoT sensor processing framework, Signal processing, Software architecture

ACM Reference format:

William M. Mongan, Ilhaan Rasheed, Khyati Ved, Shrenik Vora, Kapil Dandekar, Genevieve Dion, Timothy Kurzweg, and Adam Fontecchio. 2017. On the Use of Radio Frequency Identification for Continuous Biomedical Monitoring. In *Proceedings of The 2nd ACM/IEEE International Conference on Internet-of-Things Design and Implementation, Pittsburgh, PA USA, April* 2017 (IoTDI 2017), 6 pages.

DOI: http://dx.doi.org/10.1145/3054977.3055002

1 INTRODUCTION

RFID has been used in various applications in inventory management, access control and tracking for many years [Landt]. The technology provides several advantages such as low power consumption, low cost and small client form factor that make it attractive. Recently, there has been interest in using RFID for applications that require continuous monitoring, as opposed to discrete detections that contemporary systems employ. Achieving continuous monitoring with RFID will make it adoptable in a wider range of applications.

In order to extend the use of RFID beyond it's current applications, an architecture that supports the collection, transportation and storage of RFID data needs to be developed. RFID readers do not possess computation, storage or networking capabilities that would be required in continuous monitoring applications such as biosensing. Further, security is always a concern when it comes to medical data and contemporary RFID systems do not implement secure methods for data storage and transportation. This is a major hindrance in the adoption of RFID for wireless medical

Our research results are based upon work supported by the National Science Foundation Partnerships for Innovation: Building Innovation Capacity (PFI:BIC) subprogram under Grant No. 1430212. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. Research reported in this publication was supported by the National Institute of Biomedical Imaging and Bioengineering of the National Institutes of Health under award number U01EB023035. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

[@] 2017 Copyright held by the owner/author (s). Publication rights licensed to ACM. 978-1-4503-4966-6/17/04. . . \$15.00



Figure 1: System-level block diagram showing interactions between the smart garment, the RFID interrogator, the data collection module, as well as the processing modules

sensors. Most current RFID data processing techniques involve onoff or backscatter power measurements [Amendola et al.] and do not leverage additional measurements available in newer readers. Through wearable smart garments, we are able to continuously monitor biofeedback such as infant respiratory patterns, uterine contractions during labor and delivery, and motion artifacts to detect the potential onset of Deep Veneous Thrombosis. Our wireless, passive Bellyband smart garment device [Mongan et al.] uses a passive UHF RFID tag and knit antenna composed of conductive fabric to form a low-power wearable biosensor that is energized entirely from the wireless monitoring signal.

To facilitate wearable sensors via RFID, it is necessary to develop software to drive the interrogator for continuous polling of RFID tags, and to develop software that efficiently collects these potentially large and rapidly updating data sets in real-time. The software is loosely coupled via web services communicating over HTTP+SSL (HTTPS) sockets, and threaded to support simultaneous data capture, processing, and visualization. The high-level system modules to support RFID interrogation is shown in Figure 1, in which each module is provided with a common interface that can be implemented to support a variety of interrogators, databases, sensors, and processing modules.

The repurposing of RFID interrogation for continuous biofeedback monitoring and high-speed data collection present several challenges addressed in this effort. If the network traffic arrives too quickly, the time required to process it for storage can exceed the time between interrogation messages, resulting in a delay in packet processing that worsens with time and prohibits real-time performance. Additionally, because RFID interrogation is used for ubiquitous deployment and inventory of RFID tags in the local environment, Federal Communications Commission (FCC) regulations mandate that RFID interrogations "frequency hop" by interrogating for only a short time on any one frequency within the RFID frequency spectrum, before moving to the next frequency in that spectrum [U.S. Government Publishing Office]. Frequency hopping changes the RSSI measured by the interrogator due to interrogating the antenna using a light wave of a different frequency, producing variations in the biomedical readings.

In this paper, we have presented a software framework that is capable of collecting, transporting, storing and processing real-time medical data from wireless sensors in a reliable and secure manner. This framework is compatible with commercial off the shelf RFID readers and tags and allows for easy deployment. We also present how passive UHF RFID can be used for medical monitoring without modifications to its existing protocols. Furthermore, in addition to backscatter power measurements from RFID tags, we also use phase and doppler to classify signals from the Bellyband sensor.

The rest of this paper is organized as follows. In Section 2, we describe background including the selection of RFID as a sensor medium, its benefits, and challenges presented. In Section 3, we describe related work, we detail our approach in Section 4 and our results in Section 5; finally, we conclude in Section 6.

2 BACKGROUND

Although RFID was not designed with continuous monitoring applications in mind, we are able to configure the RFID interrogator such that continuous, consistent interrogation is feasible at rates that are sufficient for relatively low-frequency monitoring such as respiration and heart rate. Monitoring changes in RFID reflected signal enables battery-free monitoring that can be embedded directly into clothing in an unobtrusive way. As long as the subject is 50 cm from the interrogator, RF exposure levels remain below the peak Specific Absorbtion Rate (SAR) [Mongan et al.].

The Impinj Speedway R420 [Impinj] was selected as the RFID interrogator, which communicates with external software via the Low Level Reader Protocol (LLRP) [EPCglobal] protocol. A Python LLRP protocol implementation [Giometti] and wrapper library [Ransford] were adapted for the R420 and our biosensor application.

The Impinj uses the unlicensed UHF RFID frequency band utilizing frequencies between 902 MHz and 928 MHz, frequency hopping between 50 channels, each with a bandwidth of 500 kHz. The interrogator polls for RFID tags in the far field at each frequency for 200 ms, in accordance with FCC regulations [U.S. Government Publishing Office]. Using the RFID configuration specified in Section 4, we interrogate an RFID tag at a rate of 90 Hz, resulting in up to 18 tag interrogations per frequency channel. To mitigate the change in observed RSSI of the tag resulting from the change in transmitter frequency, we replace each measured RSSI with the difference of the measured RSSI and the mean RSSI observed for all tags within that channel during the 200 ms "channel burst."

A Laerdal SimBaby [Laerdal] was utilized as an experimental testbed for our software and sensors. The SimBaby is an actuated mannequin that is driven by proprietary software to simulate various behaviors, including respiratory activity and anomalies. For experimental purposes, it is programmed to either breathe at a predefined rate for a certain period of time (*i.e.*, 30 respirations per minute for 1 minute), and then change to a different rate for the same predefined period, and to cease respiratory activity before repeating.

Uterine activity monitoring is simulated using a mannequin wearing the Bellyband that is mechanically actuated using an air pump. Here, uterine activity is simulated and measured simultaneously by the RFID-based Bellyband and a tocodynamometer pressure sensor that serves as the present "gold standard" for fetal monitoring. A side-by-side comparison of the mannequin monitored by the Bellyband and tocodynamometer is shown in Figure 2. On the Use of Radio Frequency Identification for Continuous Biomedical Monitoring



Figure 2: A tocodynamometer pressure sensor is placed above a Bellyband on a "pregnant mannequin" and monitored using our software (the RFID data plot is to the left, and the tocodynamometer data plot is to the right)

3 RELATED WORK

The system in [Volk et al.] uses HF RFID (13.56 MHz) with implanted sensors to monitor biofeedback in animals. The system is able to collect ECG, arterial blood pressure and body temperature wirelessly from the implanted device. The authors developed a software framework similar to ours that utilizes a client-server topology to extend capabilities of their RFID reader. While the client server architecture does provide their RFID based system much needed features, it does not incorporate security in the transmission and storage of collected data. Further, HF RFID has a much shorter read range [Weis] and a lower read rate that makes it impractical for human biosensing applications. Our software has been designed and optimized to handle the higher date rates of UHF RFID and provide real time data collection.

NightCare is a passive RFID based sleep monitoring system that uses RFID tags to detect motion, accidental falls, long-term absence and interactions with nearby objects [Amendola et al.]. Power readings from tags placed in different parts of a patient's vicinity such as beds, tables, floors are used to infer patient movement and absence. NightCare is capable of using this data to generate reports on sleep patterns that can be used to diagnose sleep related medical conditions such as sleep apnea and bedsores. Thus, the system does not read medical signals such as respiration or heart rate.

Other wireless biomedical sensors have been proposed that do not use RFID. Vital-Radio is a system that uses reflected wireless signals to monitor respiration and heart rate in real-time. The system transmits low-power wireless signals and measures the time taken for these signals to reflect back to the device[Adib et al.]. Movements due to respiration and heartbeats cause variations in the reflection time of these signals and is used to determine respiration and heart rate. This system completely alleviates the need for a device to be placed on a patient. However, the system cannot guarantee privacy for all individuals within wireless range and requires a 1.5 meter separation between patients.

The unique design of our Bellyband sensor enables us to determine biofeedback such as respiration and ambient motion using only a single RFID tag. We have also designed a scalable software framework that is capable of securely reading, transporting and storing RFID data. This allows the framework to be used in human medical applications, where privacy and security would be a huge concern. The framework also consists of data processing tools that are used to extract information such as respiration and contraction rate from raw RFID data. The flexible nature of the framework allows seamless integration of other RFID based biomedical devices, such as a heart rate monitor. Finally, our data processing algorithms utilize RFID data other than backscatter power such as doppler and phase, allowing us to improve accuracy and extract more features from the Bellyband sensor.

4 APPROACH

To collect data from RFID-based sensors, it is necessary to appropriately control the interrogator to obtain data at a desired rate and for the desired number of sensors, interrogators, and antennas. A modular software framework was developed to both interrogate sensor devices and to store that streaming data for live and post-processing.

4.1 Software Framework for Continuous RFID Monitoring

The interrogation, server, and signal processing software are deployed to a network of Raspberry Pi 3 computers and a computer with the Laerdal SimBaby simulation software [Laerdal] used to drive the air compressor to enable respiratory motion on the baby mannequin. The Raspberry Pi running the interrogation software communicates with and drives the Impinj R420 [Impinj] over a wired ethernet connection to the interrogator. Although the Sim-Baby is physically tethered to the simulation computer and air compressor, and the Raspberry Pi is physically tethered to the RFID interrogator, all communication between the RFID interrogator and the RFID tag(s) are wireless and either passive, or energized using power harvested from the wireless interrogation signal.

4.1.1 Data Storage Layer and HTTP Web Service Interface. In case a particular interrogator or sensor requires a certain framework or imposes operating system and programming language constraints, RESTful [Fielding] HTTP web services were used as the communication mechanism between the layers, so that another language, platform, or system would be compatible with the framework. In order to insert time-series data into the database storage or to query the database, it is only necessary to open a network socket and issue an HTTP request to a documented URL. The database includes a table for storing encrypted time-series data as well as a table for the time-series data as well as a table serving as an audit log that automatically logs access to the database in compliance with Health Insurance Portability and Accountability Act (HIPAA) regulations pertaining to accessing potentially protected health information (PHI). In addition, the data is encrypted as it is entered into the database, as described in Section 4.3.

When the web server is started, it instantiates the desired database backend implementation that will be used to service these HTTP requests to insert data from the interrogator(s) and retrieve data for the data processor(s). Additionally, a secure database implementation is provided that interfaces with Research Electronic Data Capture (REDCap) [Harris et al.], a research data backend conducive to storing clinical data from human trials. REDCap is intended primarily for manual data entry of individual data records. This is not compatible with a sensor passing data at a rate of dozens per second, so a web service API interface is provided by REDCap for this purpose and utilized by our web server. Essentially, the web server becomes a web service client when data is received, reformatting the request in a manner compatible with REDCap and passing the request to its server.

4.1.2 Sensor Interrogation Layer. To generate time-series sensor data, one or more interrogators are deployed for monitoring and data collection. The Impinj Speedway R420 [Impinj] was selected as the primary interrogator for our applications. Like the server backend modular design, each client interrogator implements a common interface for data collection that communicates with the web service. A Python client driver library called LLRPyc [Giometti] formats communications into the LLRP binary data format, and a Python wrapper module called sllurp [Ransford] provides a network socket interface to the underlying LLRP library. The Impinj R420 and R1000 were implemented within this framework, and the Intermec IP30 portable Bluetooth interrogator was programmed to communicate over a Bluetooth connection according to its protocol specifications [Intermec].

These LLRP libraries were extended to support Impinj custom reports, so that additional data to the RSSI can be obtained from each interrogation. Prior to configuring the Reader Operations Specification (ROSpec) as part of the LLRP specification [EPCglobal], a custom parameter is sent to the interrogator to enable the Doppler and Phase fields to be reported with each interrogation [Impini, Impinj]. Because LLRP is a synchronous protocol that sends requests to the interrogator and receives an acknowledgement response from the interrogator before moving on to the next state, it is necessary to enable the Impini custom extensions prior to issuing the ROSpec message to enable the ROSpec on the interrogator. The networking layer was implemented to be asynchronous for realtime performance, so it was necessary to add a deferred callback after enabling the extensions, which was invoked upon receipt of the message response. This had the effect of injecting the custom extensions configuration messages into the LLRP handshake and configuration finite state machine [EPCglobal].

4.2 Configuring and Driving the RFID Interrogator for Continuous Monitoring

Typical singulation protocols focus on inventory of dynamic tag populations; that is, tags that may dynamically enter and leave the interrogation field. For continuous monitoring of RFID tags, it is preferrable to continually interrogate a known quantity of tags at a constant rate. However, because RFID is intended for dynamic tag populations, a collision avoidance algorithm known as EPCglobal Class-1 Gen-2 is employed by the RFID interrogator as a singulation algorithm that ensures that only one RFID tag is targeted at any given time. Since the number of RFID tags in the population is generally known in our continuous monitoring application, the RFID reader is configured to a tag population that grows exponentially with the anticipated number of actual tags in the population, in order to reduce the probability of a collision in any time slot. Because an empty time slot is less expensive than a collision [Bu et al.], a more predictable consistent read rate is achieved by over-estimating the tag population in this way. The reader setting configured was "MaxMiller" [Impinj] mode with "M4" encoding, which reduces the tag read rate by a factor of 4, but this encoding is more robust to noise and interference [Buettner and Wetherall], which again enables a more predictable read rate.

Through the use of the cProfile Python profiler, it was determined that the sllurp rawDataReceived function, which implements this parsing functionality, required at times 3 seconds or more to complete. However, at 90 Hz, data must be processed in under 11 ms in order to maintain real-time performance. Instead of parsing the message as the data is received in the rawDataReceived method of the sllurp library, the data is instead placed into a queue as soon as it is received, and the library returns to polling the network card for additional data (which is subsequently appended to the queue).

4.3 Data Privacy Considerations

Data in transit is encrypted via the HTTPS SSL key used by the interrogation layer and web service layer. Data at rest is achieved by encrypting the data on the database using a user-specified password as an Advanced Encryption Standard (AES) [Miller et al.] key, or by the REDCap backend. One challenge presented by symmetric-key encryption strategy such as AES is that the same plaintext input will be encrypted to the same plaintext output. RFID interrogation of a mostly stationary RFID tag will result in encrypted values that are largely the same, until the RFID tag is moved. To help address this, AES encryption is run in "counter" (CTR) mode, in which the key is augmented with an increasing number so that subsequent duplicate plaintext inputs are encrypted effectively with different keys, resulting in different ciphertext results. Rotating the key in CTR mode is contrasted with Cipher Block Chaining (CBC) mode in which the data itself is encrypted in sequential, dependent blocks, and the former approach was chosen to facilitate random access decryption and to avoid chaining together several blocks of data that contain the same or similar time-series data points (i.e., several identical blocks that contain a respiration at approximately the same point in time). We chose the interrogator relative timestamp, which is a monotonically increasing counter value that we can easily retrieve for symmetric decryption.

4.4 Experimental Setup

This **Detector** signal processing module polls the web service at a pre-defined interval (*i.e.*, every 750 ms), and performs augmentation on the data suitable for use by signal processing modules. This augmentation includes interpolating and resampling the data, converting any values from two's complement or an integer representation of a decimal value into a more reasonable format for processing rather than a format that was more convenient for network transmission, and computing higher-order features from the raw data such as RFID tag velocity. These higher-order features include a more accurate computation of the phase angle using the Doppler measurement from the RFID, which has been found to be a more precise measurement than the one provided by the Impinj if successive interrogations are obtained from the same tag, antenna, and frequency channel [Han et al.]. The RSSI difference from the mean of the "channel burst" is computed at this stage to mitigate the effect of channel hopping. At each interval, data is requested from the web service, and passed to one or more **Processor** modules. Each subclass performs processing, statistical inference, or machine learning algorithms as appropriate to the application, and returns aggregate values for reporting, graphical plotting, and emergency alerting by the **Detector** class (for example, an alert might be raised if the respiration rate falls below a threshold such as 15 respirations per minute). The data are resampled to ensure a constant sampling interval, and data lost due to interference are interpolated.

The **Detector** can run in live real-time mode, in which data is polled from the web service at a predefined time interval, or in "simulated real-time" mode, in which the first second of data is retrieved before pausing for 1 second and repeating with the next subsequent second of data. This is useful when a database of data has been saved so that experiments can be validated for replicability.

4.4.1 Infant Respiratory Monitoring. For a respiration monitoring application, a Laerdal SimBaby is programmed to simulate respiratory motion that stretches the Bellyband at a rate of 30 complete respirations (inhale plus exhale) per minute. An experimental setup, in which the interrogator is placed approximately 1 meter from the baby, is used to collect data using our framework. By performing statistical analytics on Short Time Fourier Transform (STFT) windows of the data collected, we are able to make fine determinations about whether respiratory motion is occuring at a given point in time; thus, the duration and rate of respiratory motion can be estimated. The STFT and rate estimation analysis is performed within a real-time **Processor** module that collects time-series RFID data from the interrogator web service layer.

4.4.2 Uterine Monitoring. For a uterine monitoring application, an air balloon inside a "pregnant mannequin" is actuated by an air pump, and measured as fluctuations in RSSI by a Processor module. Unlike the respiratory application which detects properties of fine repetitive repetition, uterine monitoring seeks "anomolous" coarse motions that are more spread out over time. For this application, the Processor utilized a Support Vector Machine for classification of Bellyband stretching motion. However, it is infeasible to expect to train the Support Vector Machine (SVM) [Cortes and Vapnik] classifier on uterine contractions by inducing a contraction on-demand, so a One-Class SVM was used instead that trains on non-uterine motion data and classifies Bellyband "stretching" motions as anomalies. An example side-by-side plot of mannequin actuations monitored by the RFID Bellyband and a traditional tocodynamometer fetal monitor device is shown in Figure 2. An interrogator module was developed for purposes of collecting comparison data from the Philips 50XM tocodynamometer; as this device neither uses RFID nor communicates over ethernet, it was designed to extract data from the 50XM serial RS-232 port according to its protocol specifications [Philips]. Its transducer pressure value was collected and stored in the interrogation data structure in lieu of an RSSI value.

4.4.3 *Heart Rate Monitoring.* RFID heart rate monitoring utilizes an active RFID circuit that is powered using energy harvested wirelessly from the RF interrogation signal. This is done to amplify the ECG signal from the heart. This application does not utilize the RSSI property of the RFID signal directly to estimate the heart rate as in the previous applications; instead, the RFID tag and antenna

connection is broken when a heart beat is detected. Therefore, the period between RFID interrogation responses is utilized to detect heart beats and, thus, the heart rate. This approach has the benefit of enabling heart monitoring on the same RFID chip and antenna as that used for respiratory monitoring, so that a single RFID tag assembly is feasible as a wearable smart garment device [Vora et al.]. Further, a single tag RFID system facilitates a higher RFID interrogation rate, since the interrogator need not oscillate between multiple tags in the system, querying each tag before returning to any one tag for a second interrogation. However, the RFID connection cannot be broken for a period shorter than the longest RFID interrogation period; otherwise, ordinary delays between interrogations could be misinterpreted as a heart beat. It is desirable to maintain a constant RFID interrogation period and a high interrogation rate in order to enable a short RFID communications break when a heart beat is detected. Although the RFID interrogation protocol does not guarantee a constant interrogation period, our threaded producer-consumer approach to interrogation data processing and consistent RFID tag environment help to ensure a fast and constant RFID interrogation period to the extent possible.

Data collection was conducted in a student office furnished with standard items like computers, desks, chairs etc. An RFID reader and an RFID tag were placed at varying distances from each other between 1 and 3 meters. To simulate the effect of additional tags in the environment, two more tags were added, one after the other, at the same aforementioned distances from the reader antenna. In the presence of additional tags, time between tag read data was calculated for the same tag and not for the additional ones. Data was collected at each setting for a period of three minutes. The timestamp for each tag read was recorded and the difference between successive timestamps was calculated to determine the time between tag reads. A faster read setup was also tested using only the flexible Monza tag at 1 meter distance from the antenna. The reader setting chosen was "Dense Reader Dual Target" [Impinj] mode.

5 RESULTS

5.1 Infant Respiratory Monitoring and Uterine Monitoring

Using our framework, we have developed real-time statistical processors and machine learning modules to estimate respiratory and uterine state from passive RFID biosensors [Mongan et al.]. Using classification, stretching activity from the SimBaby or Mannequin was detected with 70% accuracy using RSSI data alone, filtered with a Gaussean filter in lieu of channel normalization. This improved to 94% with the implementation of multisensor fusion of stretching duration, channel normalization, and and statistical analysis of the STFT [Mongan et al.].

5.2 Continuous RFID Datastream for Infant Heart Monitoring

When the time between tag reads exceeds a certain threshold, a heart beat is detected since the RFID tag connection is broken for a short period for each heat beat. Otherwise, the time between tag reads is very regular and for the most part, tag reads happen in between 27 and 30 milliseconds. There is an additional group of tag read times around 34 milliseconds and this delay can be attributed to tags that were missed on the first attempt. The smaller groups following these two relatively larger groups of tag read times happen mostly due to the delay caused by periodic frequency hopping. Almost all tag reads happen in under 40 milliseconds. The time between tag reads was found to be consistent at each tag distance tested. By modifying the LLRP library to be asynchronous and configuring the RFID interrogator to "Dense Reader Mode," we raised our read rate from 30Hz (an interrogation period of 33ms) to 90Hz, which allowed us to lower the RFID outage time from 100ms and thus enabled simultaneous strain gauge and heart monitoring from a single RFID tag. The time between tag reads is nearly three times lesser than the previous setting chosen; this setting can be used for faster biosignal measurements with greater resolution.

In a data set of over 60,000 tag reads, fewer than 100 tag reads took between 50 and 90 milliseconds. These delayed reads mainly happened at greater distances from the reader and in the presence of multiple tags; both factors can cause missed tag reads. Even though the delayed reads are few in number, there could be situations where the delays are increased. Hence, the choice of an outage duration should take into account these delayed reads. In summary, the time between tag reads can be in the vicinity of 6 milliseconds to about 40 milliseconds depending on the RFID reader setting. The tag read time can increase due to various issues that cause read misses but the aforementioned numbers are typical. The RFID outage duration should be significantly greater than the typical tag read time.

6 CONCLUSION AND FUTURE WORK

In this effort, we have enabled continuous sensor monitoring using wireless RFID tags that are passive or energized using wireless power harvesting. RFID as a physical layer and media access control (MAC) vehicle was re-purposed from inventory management to continuous monitoring by configuring LLRP-based interrogators and software libraries to transmit multiple interrogation reports for each RFID tag, at a consistent rate within the channel burst. Networking, storage, and processing layers were implemented to support this RFID time-series data, all with performance constraints that enabled real-time interrogation performance for at least 90 RFID interrogations per second. The data storage was implemented to support human clinical trials by securing data both in transit and at rest. As a result, several wearable RFID biosensing applications were enabled as a result of this work, including infant respiratory monitoring, uterine monitoring, and heart rate monitoring.

In the future, we plan an empirical study of RFID configurations for various RFID tag populations, in order to better bound expected interrogation performance that can be expected. Additionally, we plan to study the effects of environmental interference on RFID interrogation, with a goal of bounding a minimal read rate for a certain threshold of data processing accuracy. Finally, we plan to expand this work to support mobile interrogators with a goal of enabling wireless, mobile interrogation devices that can be carried about the body.

REFERENCES

 Fadel Adib, Hongzi Mao, Zachary Kabelac, Dina Katabi, and Robert C Miller. 2015. Smart Homes that Monitor Breathing and Heart Rate. Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems - CHI '15 (2015), 837–846. DOI: http://dx.doi.org/10.1145/2702123.2702200

- [2] S Amendola, R Lodato, and S Manzari. 2014. RFID Technology for IoT-based Personal Healthcare in Smart Spaces. *Internet of Things* 1, 2 (2014), 144–152. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6780609
- [3] Kai Bu, Qingjun Xiao, and Bin Xiao. 2011. Efficient Monitoring of Dynamic Tag Populations in RFID Systems. Embedded and Ubiquitous Computing, IEEE/IFIP International Conference on 00 (2011), 106–113. DOI:http: //dx.doi.org/doi.ieeecomputersociety.org/10.1109/EUC.2011.30
- [4] Michael Buettner and David Wetherall. 2008. An empirical study of UHF RFID performance. In MOBICOM (2008-09-22), J. J. Garcia-Luna-Aceves, Raghupathy Sivakumar, and Peter Steenkiste (Eds.). ACM, 223–234.
- [5] Corinna Cortes and Vladimir Vapnik. 1995. Support-Vector Networks. Mach. Learn. 20, 3 (9 1995), 273–297. DOI: http://dx.doi.org/10.1023/A:1022627411411
- [6] EPCglobal. 2010. Low Level Reader Protocol (LLRP) Ratified Standard, Version 1.1. (October 2010). http://www.gs1.org/gsmp/kc/epcglobal/llrp/llrp_1_ 1-standard-20101013.pdf
- [7] Roy Thomas Fielding. 2000. Architectural Styles and the Design of Networkbased Software Architectures. Ph.D. Dissertation. University of California, Irvine. AAI9980887.
- [8] Rodolfo Giometti. 2016. LLRPyC LLRP Python Protocol Client Implementation. (2016). http://wiki.enneenne.com/index.php/LLRPyC
- [9] Jinsong Han, Han Ding, Chen Qian, Dan Ma, Wei Xi, Zhi Wang, Zhiping Jiang, and Longfei Shangguan. 2014. CBID: A customer behavior identification system using passive tags. Proceedings - International Conference on Network Protocols, ICNP (2014), 47–58. DOI: http://dx.doi.org/10.1109/ICNP.2014.26
- [10] Paul A. Harris, Robert Taylor, Robert Thielke, Jonathon Payne, Nathaniel Gonzalez, and Jose G. Conde. 2009. Research Electronic Data Capture (REDCap)-A Metadata-driven Methodology and Workflow Process for Providing Translational Research Informatics Support. *J. of Biomedical Informatics* 42, 2 (April 2009), 377–381. DOI: http://dx.doi.org/10.1016/j.jbi.2008.08.010
- Impinj. 2016. Impinj LTK Programmers Guide. (2016). https: //support.impinj.com/hc/en-us/article_attachments/205361988/Impinj_ LTK_Programmers_Guide.pdf
- [12] Impinj. 2016. Impinj Speedway Revolution. (2016). http://www.impinj.com/ products/readers/speedway-revolution/
- [13] İmpinj. 2016. Octane LLRP User Guide. (2016). https://support.impinj.com/hc/ en-us/article_attachments/205374367/Impinj_Octane_LLRP.pdf
- [14] Impinj. 2016. Optimizing Tag Throughput Using Reader-Mode. (2016). https://support.impinj.com/hc/en-us/articles/ 202756368-Optimizing-Tag-Throughput-Using-ReaderMode
- Intermec. 2016. IP30 Handheld RFID Reader User Guide. (2016). http://epsfiles. intermec.com/eps_files/eps_man/937-000.pdf
- [16] Laerdal. 2016. Laerdal SimBaby and Linkbox. (2016). http://www.laerdal.com/ us/item/245-02001
- [17] J. Landt. 2005. The history of RFID. IEEE Potentials 24, 4 (Oct 2005), 8–11. DOI: http://dx.doi.org/10.1109/MP.2005.1549751
- [18] Frederic P. Miller, Agnes F. Vandome, and John McBrewster. 2009. Advanced Encryption Standard. Alpha Press.
- [19] W. Mongan, E. Anday, G. Dion, A. Fontecchio, K. Joyce, T. Kurzweg, Y. Liu, O. Montgomery, I. Rasheed, C. Sahin, S. Vora, and K. Dandekar. 2016. A Multi-Disciplinary Framework for Continuous Biomedical Monitoring Using Low-Power Passive RFID-Based Wireless Wearable Sensors. In 2016 IEEE International Conference on Smart Computing (SMARTCOMP). IEEE, 1–6.
- [20] W. Mongan, K. Dandekar, G. Dion, T. Kurzweg, and A. Fontecchio. 2015. Statistical Analytics of Wearable Passive RFID-based Biomedical Textile Monitors for Real-time State Classification. In 2015 IEEE Signal Processing in Medicine and Biology Symposium (SPMB). 1–2. DOI: http://dx.doi.org/10.1109/SPMB.2015.7405465
- [21] Philips. 2016. Series 50 Fetal Monitors, Design Interface Protocol Specifications: Programmer's Guide. (2016). http://www. frankshospitalworkshop.com/equipment/documents/ecg/service_manuals/ Philips_Series_50_-_Programmers_guide.pdf
- [22] Ben Ransford. 2016. sllurp LLRP Python Protocol Client Implementation. (2016). https://github.com/ransford/sllurp
- [23] U.S. Government Publishing Office. 2016. Electronic Code of Federal Regulations, Title 47, Chapter I, Subchapter A, Part 15.247. (2016). http://www.ecfr.gov/ cgi-bin/text-idx?node=pt47.1.15&rgn=div5#se47.1.15_1247
- [24] T. Volk, S. Gorbey, M. Bhattacharyya, W. Gruenwald, B. Lemmer, L. M. Reindl, T. Stieglitz, and D. Jansen. 2015. RFID Technology for Continuous Monitoring of Physiological Signals in Small Animals. *IEEE Transactions on Biomedical Engineering* 62, 2 (Feb 2015), 618–626. DOI: http://dx.doi.org/10.1109/TBME.2014. 2361856
- [25] Shrenik Vora, William Mongan, Kapil Dandekar, Adam Fontecchio, and Timothy Kurzweg. 2016. Wireless Heart and Respiration Monitoring for Infants using Passive RFID Tags. In Proceedings of BHI-2016 International Conference on Biomedical and Health Informatics.
- [26] Stephen A Weis. 2007. RFID (radio frequency identification): Principles and applications. System 2, 3 (2007).