

Passive EPC Class 1 Gen 2 UHF RFID Sensor Tag

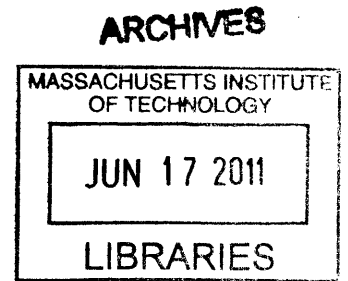
For Health Monitoring Applications

by

Haobo (Jack) Dong

B.S. Electrical and Computer Engineering

Brigham Young University (2009)



ARCHIVES

Submitted to the Department of Electrical Engineering and Computer Science

in Partial Fulfillment of the Requirements for the Degree of

Master of Science in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2011

© 2011 Massachusetts Institute of Technology. All rights reserved.

Signature of Author

Department of Electrical Engineering and Computer Science

May 18, 2011

Certified by

Joel L. Dawson

Associate Professor of Electrical Engineering

Thesis Supervisor

d

Accepted by

/ / U

Leslie A. Kolodziejski

Chair, Department Committee on Graduate Students

Passive EPC Class 1 Gen 2 UHF RFID Sensor Tag For Health Monitoring Applications

by

Haobo (Jack) Dong

Submitted to the Department of Electrical Engineering and Computer Science

on May 18, 2011 in Partial Fulfillment of the

Requirements for the Degree of

Master of Science in Electrical Engineering and Computer Science

ABSTRACT

Parkinson's disease (PD) is a chronic and degenerative condition that affects millions of Americans. Current approach of PD evaluation, diagnosis, and treatment is mainly qualitative using the Unified Parkinson's Disease Rating Scale (UPDRS) or the Hoehn and Yahr scale. Assessment of the efficacy of the drugs used is difficult and subjective. A long-term monitoring device that can collect movement data in assisting quantitative analysis proves to be useful and needed.

This thesis discusses a discrete prototype of a passive EPC Class 1 Gen 2 UHF RFID sensor tag which is a preliminary step in realizing such a monitoring device. The prototype is capable of collecting 8-bit sensor (temperature, inertial, etc) data and transmits it in real-time through a RFID backscatter link to an UHF reader. It is shown that the device can achieve a read distance up to 3 meters at 5 reads/s and a max data rate of about 640 Kbps.

Thesis Supervisor: Joel L. Dawson

Title: Associate Professor of Electrical Engineering

Dedication

To my wife Kara,
my daughter Emma,
and my mom Lucy.

Acknowledgements

I'd like to thank my research advisor Prof. Joel Dawson for the opportunity and privilege to work in his research group. I'm amazed by his devotion to and care for the success of his students. On several occasions, he has provided practical hands-on guidance to the experiments I was conducting. He is always optimistic and patient, allowing me to explore unknown spaces and learn for myself. I want to thank him for his vision in setting the direction of my project and for all the encouragements he has given me. Thank you very much, Joel.

I want to thank my colleagues in the Dawson, Charlie Sodini, and Harry Lee's groups. Especially, I'd like to thank Philip Godoy, Sungwon Chung, Kailiang Chen, William Sanchez, Tania Khana, Jack Chu, Sunghyuk Lee, and Harneet Khurana for their needed helps on many occasions. I'd like to thank Amanda Gaudreau, David He, Eric Winokur, Grant Anderson, Khoa Nguyen, Zhen Li for their associations and contributions in making the lab a social, fun, and memorable place. I want to thank Coleen Kinsella, our wonderful secretary, for being so efficient at ordering parts and materials, which made my work a lot easier.

I like to thank Alanson Sample and Daniel Yeager at the University of Washington for helpful and practical discussions on their WISP project.

I want to thank my wife Kara for her loving support, encouragement, and her work taking care of Emma so that I can focus. I want to thank Emma for being so good at night and so cute. Whenever I held her in my arms and she stared at me, all my worries went away. I want to thank my mom Lucy for all she did in the home supporting me in a loving way. Without the support of my family, I can hardly imagine I have the strength to get anywhere.

Contents

Chapter 1 Introduction	14
1.1 Motivation	14
1.2 Overview.....	14
1.3 RFID History.....	17
1.4 Tag Classifications.....	24
1.4.1 Passive, Semi-passive, and Active Tags.....	24
1.4.2 Inductive vs. Radiative Coupling.....	26
1.4.3 Protocols.....	29
1.5 Summary.....	30
Chapter 2 UHF Gen 2 Protocol	31
2.1 Gen 2 Data Link Layer	32
2.1.1 Select Operation	32
2.1.2 Inventory Operation.....	34
2.1.3 Access Operation.....	38
2.2 Tag Memory	40
2.2.1 EBV-8 Scheme.....	41
2.2.2 Reserved Memory	41
2.2.3 EPC Memory.....	42
2.2.4 TID Memory.....	43
2.2.5 User Memory	44
2.2.6 Reply to the ACK Command.....	44
2.3 Tag States.....	45
2.4 Gen 2 Physical Layer	48
2.4.1 Operating Frequency.....	48
2.4.2 Reader-to-Tag (R=>T) Communications	48
2.4.2.1 Modulation	48

2.4.2.2	Data Encoding.....	48
2.4.2.3	Preamble and Frame-Sync.....	49
2.4.3	Tag-to-Reader (T=>R) Communications.....	50
2.4.3.1	Modulation.....	50
2.4.3.2	Data Encoding.....	51
2.4.3.3	FM0 Baseband.....	51
2.4.3.4	Miller-Modulated Subcarrier (MMS).....	53
2.4.3.5	BLF and Data Rate.....	55
Chapter 3	Gen 2 FPGA Implementation.....	57
3.1	Controller.....	58
3.2	Receiver.....	59
3.3	Transmitter.....	61
3.3.1	Preamble.....	62
3.3.2	Encoder.....	64
3.3.3	CRC16.....	64
3.4	Data Sources.....	65
3.5	Simulation Results.....	66
3.5.1	Query and RN16.....	66
3.5.2	ACK and EPC.....	69
3.5.3	Req_RN and Handle.....	72
3.5.4	Read and Data.....	74
Chapter 4	Analog Frontend and Sensor Circuitry.....	76
4.1	Analog Frontend.....	76
4.1.1	Reader.....	76
4.1.2	Reader's Antenna.....	76
4.1.2.1	Bandwidth and Impedance.....	78
4.1.2.2	Gain, Directivity, Efficiency.....	79
4.1.2.3	Radiation Pattern.....	80
4.1.2.4	Polarization State.....	81

4.1.3	Links	81
4.1.4	Tag Antenna.....	83
4.1.5	Matching Network.....	84
4.1.6	Voltage Multiplier	87
4.1.7	Demodulator	88
4.1.8	Modulator	90
4.2	Sensor Circuitry.....	90
Chapter 5 Experimental Results		91
Chapter 6 Conclusion and Future Work.....		95
Appendix A.....		97
A.1	Friis Transmission Formula Derivation	97
A.2	Testing Equipment List.....	100

List of Figures

Figure 1-1: RFID system overview.....	15
Figure 1-2: Frequency band for RFID systems. Adapted from [6].....	26
Figure 1-3: Antenna near-field far-field regions. Adapted from [23].....	28
Figure 1-4: RFID standards across all common frequency bands. Adapted from [6].....	30
Figure 2-1: Reader communicates with tags implementing several states. Adapted [24]. .	32
Figure 2-2: Inventory and access of a single tag.....	35
Figure 2-3: Tag memory organization.....	40
Figure 2-4: EBV-8 examples.....	41
Figure 2-5: StoredPC structure.....	42
Figure 2-6: Tag state diagram.....	47
Figure 2-7: PIE symbol for data-0 and data-1.....	49
Figure 2-8: R=>T preamble and frame-sync.....	50
Figure 2-9: FM0 symbols and sequences.....	52
Figure 2-10: FM0 baseband generator state diagram.....	52
Figure 2-11: FM0 signaling end with a dummy 1.....	52
Figure 2-12: Miller symbols and state diagram.....	53
Figure 2-13: MMS sequence generation.....	54
Figure 2-14: Data encoding for a 2-bit data.....	55
Figure 3-1: Top level design.....	57
Figure 3-2: High low symbol detector.....	60
Figure 3-3: Packet state tracker.....	60
Figure 3-4: Response construction state machine.....	62
Figure 3-5: FM0 preambles.....	63

Figure 3-6: Miller preambles	63
Figure 3-7: CRC-16 generation	65
Figure 3-8: Tag receives <i>Query</i> and replies with RN16	67
Figure 3-9: Zoomed-in view of the <i>Query</i> command.....	68
Figure 3-10: Zoomed-in view of the <i>QueryRep</i> command.....	68
Figure 3-11: Zoomed-in view of the tag's reply (RN16)	69
Figure 3-12: Tag receives an <i>ACK</i> and replies with PC/EPC	70
Figure 3-13: Zoomed-in view of the <i>ACK</i> command	71
Figure 3-14: Zoomed-in view of the tag's reply (PC/EPC).....	71
Figure 3-15: Tag receives a <i>Req_RN</i> and replies with a new handle.....	72
Figure 3-16: Zoomed-in view of the <i>Req_RN</i> command	73
Figure 3-17: Zoomed-in view of the tag's reply (a new handle)	73
Figure 3-18: Tag receives a <i>Read</i> and replies with the data.....	74
Figure 3-19: Zoomed-in view of the <i>Read</i> command	75
Figure 3-20: Zoomed-in view of the tag's reply (data)	75
Figure 4-1: Laird Technologies circular polarized panel antenna.....	77
Figure 4-2: Panel antenna circuit model.....	79
Figure 4-3: Panel antenna radiation pattern	80
Figure 4-4: Tag antenna model	83
Figure 4-5: LC matching network	84
Figure 4-6: Tag analog frontend circuit model	85
Figure 4-7: Tag frontend equivalent circuit	85
Figure 4-8: Single-stage voltage doubler	87
Figure 4-9: Single stage voltage doubler simulation	88
Figure 4-10: Inputs to the comparator.....	89

Figure 4-11: Demodulator output	89
Figure 5-1: Measured ADC output vs. input voltage	91
Figure 5-2: Logic analyzer screen capture.....	92
Figure 5-3: Modulation output signal triggered on the TX enable	92
Figure 5-4: Data captured in the PC software	93
Figure 5-5: Measurement of the rectified voltage, read rate versus distance	94

List of Tables

Table 1-1: Tag Comparisons	24
Table 1-2: Inductive vs. Radiative Coupling Comparisons.....	29
Table 2-1: Select command and tag's response to Action	33
Table 2-2: <i>Query</i> , <i>QueryAdjust</i> , and <i>QueryRep</i> command structures and tag reply.....	37
Table 2-3: <i>ACK</i> and <i>NAK</i> command structures.....	38
Table 2-4: <i>Req_RN</i> command structure and tag reply	39
Table 2-5: <i>Read</i> command structure and tag reply	40
Table 2-6: Tag response to the <i>ACK</i> command	45
Table 2-7: T=>R data rate.....	54
Table 2-8: T=>R link frequencies	56
Table 3-1: Commands table	59
Table 4-1: Panel antenna specifications.....	78
Table 4-2: Polarization States.....	81

Acronyms

AFI: application family identifier

ANA: Article Number Association

ANSI: American National Standards Institute

CRC: cyclic redundancy check

CW: continuous wave

DOA: Department of Agriculture

DoA: Department of Agriculture

DoD: Department of Defense

DoE: Department of Energy

E: electric field

EAN: European Article Numbering, also known as GS1

EAS: electronic article surveillance

EIRP: effective isotropic radiated power

EM: electromagnetic

EPC: electronic product code

H: magnetic field

HF: high frequency

IC: integrated Circuits

IEC: International Electrotechnical Commission

Interrogator: reader

ISM: Industrial, Scientific, and Medical

ISO: International Organization for Standardization

Kbps: kilo bits per second

LASL: Los Alamos Scientific (now National) Laboratory

LF: low frequency

LFSR: linear feedback shift register

LSB: least significant bit

MSB: most significant bit

NSI: numbering system identifier

PCB: printed circuit board

PD: Parkinson's disease

PW: pulse width

RF: radio frequency

RFID: radio frequency identification

RNG: random number generator

S: power density

SNR: signal-to-noise ratio

Transponder: tag

UCC: Uniform Code Council, also known as GS1 US

UHF: ultra high frequency

UIN: user-specified input

UMI: user-memory indicator

UPDRS: Unified Parkinson Disease Rating Scale

VNA: Vector Network Analyzer

VSWR: voltage standing wave ratio

XI: XPC_W1 indicator

Chapter 1 Introduction

1.1 Motivation

In the past several decades, microelectronics has revolutionized the world of consumer electronics. Entering the 21st century, it is drastically changing the landscape of the biomedical field and has the potential to bring powerful medical electronics to the homes of individual consumers. Currently, research efforts are vibrant in the areas of monitoring, diagnostics and therapy delivery [1].

One significant problem doctors deal with is monitoring of tremor in patients with Parkinson's disease (PD). At present, PD has no cure. It is a chronic, degenerative condition that requires long-term monitoring of the disease progression and the efficacy of the drugs used for potential treatment. Currently, patients have to frequently visit the doctor's office to get a snap-shot evaluation. The assessments are often done qualitatively using the Unified Parkinson's Disease Rating Scale (UPDRS) or the Hoehn and Yahr scale, which classify severity on scores of 0 (none) to 4 (most severe). The responses are often subjective based on the doctors' observations. A more quantitative, objective method of evaluation is needed. One solution is to develop an electronic device that can provide long-term monitoring and collect movement data for assessment analysis.

1.2 Overview

This work is a preliminary, proof-of-concept step in developing such a monitoring device. It discusses a discrete prototype of a passive EPC Class 1 Gen 2 UHF RFID sensor tag.

Passive refers not only to the power source of the tag, which comes from the reader's RF field, but also to the means of wireless communication which is by backscatter modulation—the tag doesn't contain an active transmitter. EPC Class 1 Gen 2 is an air interface protocol widely adopted in the industry for communications in the UHF (860-960 MHz) band. The need to conform to such standard is that commercial UHF RFID readers can be used to test and verify the in-house-built sensor tag. This not only eliminates the need to develop an in-house custom reader, but also allows the developed tag to be deployed immediately. Combining sensors with traditional RFID technology creates a powerful platform for a wide array of novel applications. It extends beyond the basic function of identification to monitoring and perhaps computation.

The overall RFID system setup for this work is shown in the Figure 1-1.

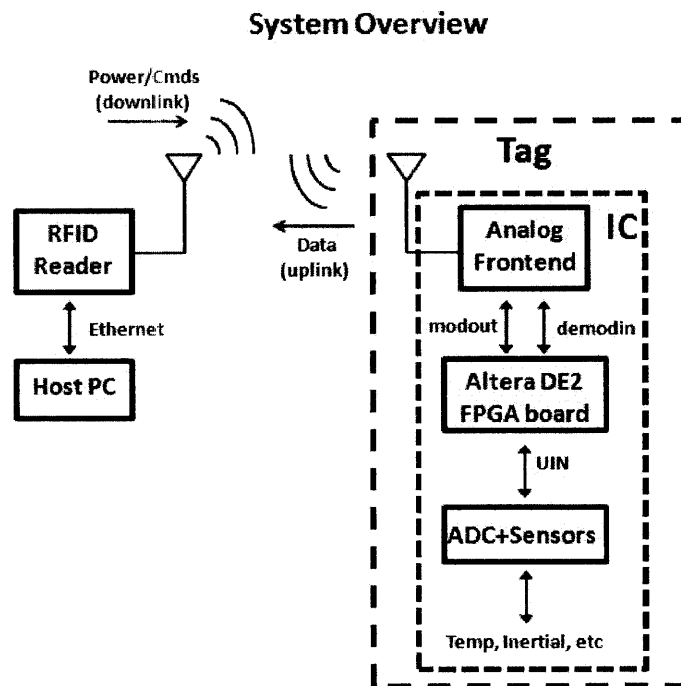


Figure 1-1: RFID system overview

Traditional radio frequency identification, or RFID, is a technology that uses RF waves to identify an object. Figure 1-1 shows that the system consists of a transponder (tag), an interrogator (reader), and a host computer. The tag is usually labeled to a physical object. It is composed of an antenna and a silicon microchip IC which contains an electronic product code (EPC) and implements the communication protocol. In this work, discrete components are used to make the system blocks of analog frontend, FPGA, ADC, and sensors. EPC is embedded in and the Gen 2 protocol is implemented on the FPGA. Sensors extend tag functionality beyond identification to monitoring. In the future, these blocks will be integrated into a single IC.

In the case of passive UHF, the tag is powered entirely by the RF waves of the reader which sends commands to interrogate the tag. To reply, the tag sends requested data back to the reader by modulating the reflection of the RF waves—a technique called backscatter modulation. Only passive and semi-passive tags use backscatter modulation. Active tags directly broadcast data to the reader. The communication direction of reader-to-tag is termed downlink or forward link while the tag-to-reader direction is called uplink or reverse link. Data received by the reader is sent to the host PC through an Ethernet network where application software processes the data.

The remainder of Chapter 1 will discuss RFID history and types. Chapter 2 will discuss the Gen 2 protocol. Chapter 3 will discuss how the protocol is implemented on the FPGA and shows simulation results. Chapter 4 will discuss the RFID reader, links, analog frontend, ADC, and sensors. Chapter 5 will discuss the experimental results. Finally, Chapter 6 concludes and discusses the needed future work.

1.3 RFID History

RFID is not a new technology. It has existed for over half of a century. However, the cost of ICs and lack of communication protocol standards prevent it from large-scale deployment until recently. With the advent of inexpensive, low-power modern ICs and globally uniform standards, RFID is being rediscovered and innovated in the 21st century.

RFID traces its origin back to the demonstration of radar by the Scottish scientist Sir Robert A. Watson-Watt in February 1935 [2]. Originally, the capitalized term RADAR was coined for radio detection and ranging. During World War II, it was often too late to respond when the enemy's airplanes could be seen visually at a distance. This was the motivation behind the invention and development of radar technology, which allowed early detection of incoming airplanes far beyond the visual range. Radar sends out radio waves to detect the location, distance, and speed of an airplane by the reflected waves bounced off that airplane. However, the problem was that it could not distinguish whether the incoming airplane was a friend or an enemy. In other words, radar could only detect but not identify an object. Luftwaffe, the German air force, solved this problem initially by using a simple maneuver. As they were returning to their base, they would roll their airplanes to change the strength of the backscattered radio waves to the ground station to identify themselves as the German planes. This was the first known instance of passively modulating backscattered radio waves for identification. The problem associated with this simple technique was that any airplane could be rolled and more capable ways of identifying radar targets were needed.

In the 1940s, Germany, Britain, and United States developed sophisticated, airborne, active Identity: Friend or Foe (IFF) transponder systems, such as the FuG-25, XAE, Mark I, and III for radar identification. When illuminated by the radio waves sent

from the ground radar station, the transmitter on the airplane would broadcast a coded signal back to positively identify the airplane as friendly. Modern RFID systems work on basically the same principle. An interrogator sends out a radio signal and it wakes up the transponder, which either backscatters the signal or broadcasts a signal back depending on whether it is passive or active.

The first known passive device of using modulated RF backscattering to transmit voice data is the eavesdropping device “The Thing”, also known as the Great Seal bug, invented by Léon Theremin in Russia before 1945 [3]. This ingenious device contained no active electronic components and was completely passive—it contained no transmitter or power supply. It became operational when illuminated by the incident radio waves of a certain frequency sent by a remote transmitter. Sounds in the ambience of the device caused the metallic diaphragm it contained to vibrate, changing the capacitive load seen by the antenna, which in turn modulated the backscattered radio waves. Although its purpose was not for identification but eavesdropping conversations, this device used the exactly same principle—backscatter modulation as the modern passive RFID systems do. It marked the very beginning and was the predecessor of RFID.

In October 1948, Harry Stockman published a landmark paper “Communication by means of reflected power” [4], an early work that explored the backscatter modulation communication. In this paper, Stockman conducted an experiment in which he used a voice-modulated corner reflector to transmit voice, by reflected power, to a speaker a hundred yards away. In conclusion, Stockman stated, “Evidently considerable research and development work has to be done before the remaining basic problems in reflected-power communication are solved, and before the field of useful applications is explored”. Stockman’s time was not ready and the technology needed more time to develop.

In the 1950s, passive backscattering was investigated with the goal of creating inexpensive wireless telephone systems. Notable developments include works of F. L. Vernon's paper "Application of the microwave homodyne" published in October 1952 [8], D.B. Harris's patent "Radio transmission systems with modulated passive responder" filed in August 1952 [9], and Joseph H. Vogelman's patent "Passive data transmission technique utilizing radar echoes" filed in May 1959 [13].

In the 60s, works such as Robert M. Richardson's patent "Remotely activated radio frequency powered devices" filed in September 1961 [10], Roger F. Harrington's paper "Theory of loaded scatterers" in April 1964 [11], and Jorgen P. Vinding's patent "Interrogator-responder identification system" filed in May 1965 [12] kept pushing the RFID development forward. Commercialization of RFID began in the late 60s. Companies such as Sensormatic, Checkpoint, and Knogo developed inexpensive, one-bit electronic article surveillance (EAS) tags for countering theft, which marked the beginning of RFID commercial deployment [5].

The 70s saw significant practical developments of RFID. In January 1973, Mario Cardullo and William Parks received a US patent "Transponder apparatus and system" [14] for a passive RFID tag with a rewritable internal memory—a feature that distinguished itself from any previous transponders. This invention was significant and it was the ancestor of modern RFID transponders. In August that same year, Charles Walton received a patent for his "Electronic identification and recognition systems" [15]. His invention utilized inductive coupling to identify a coded object and it found success with Schlage Lock Company to make electronic locks for building access control. Later, Walton was granted the first patent to use the acronym RFID in May 1983 [20]. In November 1973, William Arnold published a paper "The toll highway faces automation" [16] in which he identified areas of RFID applications in traffic management. To an extent,

this paper set the directions for many later application developments. Also in 1973, Raytheon developed the commercial “Raytag”, an electronic remote data readout system. In October 1975, Richard Klensch at RCA labs received a patent for his “Electronic identification system” [18]. His work explored higher microwave frequency (8-9 GHz) for RFID. In January 1977, Fred Sterzer at RCA received a patent for having developed an “Electronic license plate for motor vehicles” [19]. Some of these scientists later left the lab and started companies such as Indentronix and Amtech. In 1978, R. J. King published the landmark book—*Microwave Homodyne Systems*, which was one of the earliest discourses on the theory and practices used in RFID.

In the 70s, the U.S. government was interested in developing RFID. At the request of the Department of Energy (DoE), Los Alamos Scientific (now National) Laboratory (LASL) developed a system to track vehicles and nuclear materials. LASL also developed passive RFID tags to track cows and report their temperatures for the Department of Agriculture (DoA). In August 1975, Alfred Koelle, Steven Depp, and Robert Freyman of LASL published their work for the DoA in the paper “Short-range radiotelemetry for electronic identification using modulated backscatter” [17]. In the paper, they introduced the technique of antenna load modulation, which was switched at a subcarrier frequency for effective backscattering.

During the 90s, an important technological breakthrough took place. For the first time, microwave Schottky diodes were fabricated on a regular CMOS IC which allowed UHF RFID tags to contain only a single IC, a capability that was unavailable previously. In 1999, Klaus Finkenzeller published *The RFID Handbook* [21] which was one of earliest and authoritative texts on RFID.

The 80s and 90s were decades in which RFID systems were widely deployed in several industries, including the railroad, traffic management, livestock, supply chain logistics, access control, and etc [6].

In the US, the early adopter was the railroad industry which faced the problem of tracking thousands of railcars over thousands of miles of tracks. In the early 70s, the industry tried to use circular optical barcode reflectors to identify individual railcar. As the railcar passed by an optical base station, the station would transmit a beam of light which would be reflected back by the barcode reflector. However, the optical reflector didn't work well in the snowy, muddy, and rainy outdoor weather as its surface could easily be covered or damaged. It was this problem that triggered Mario Cardullo's invention of a passive RFID tag with a changeable internal memory filed in 1970. By the late 1980s, the rail industry established a standard (AAR S-918) for railcar identification based on backscatter transponders operating in the ISM band at 902-928 MHz. By 1994, over 3 million railcars in the US were equipped with compliant RFID transponders.

The traffic management industry needed vehicle identification and automated tolling which found RFID useful. In the early 80s, Philip Electronics developed an inductive system V-com or V-tag with antennas embedded in the roads to track buses travelling certain repetitive routes. In October 1987, the city Alesund in Norway launched the operation of world's first RFID electronic toll collection systems [7]. It was followed quickly by the Dallas North Turnpike in the US in 1989. Around that time, the Port Authority of New York and New Jersey began commercial operation of RFID for buses going through the Lincoln tunnel. The E-ZPass Interagency Group was created in 1991 to develop regionally compatible electronic tolling systems for the eastern states. In the same year, Oklahoma implemented open highway automated tolling system. Houston's combined toll collection and traffic management system was implemented in 1992. Title

21 standard was adopted in California and other western states. By 2001, the use of RFID for electronic toll collections had expanded in the US to 3,500 traffic lanes. Today, the wide use of RFID in vehicle identification and automated tolling is evidently seen in everyday life.

The livestock industry, mostly driven by the DoA in the US, found RFID useful. The paper Los Alamos published in 1975 was directed toward the identification of dairy cattle to address the DoA's needs. In 1991 and 1996, the ISO approved the ISO 11784 and 11785 standards (later updated to ISO 14223), respectively. Since then many countries have adopted the standards and RFID has been widely deployed in the livestock industry.

The supply chain logistics industry, mostly driven by the DoD and large retailer like Wal-Mart and Tesco, found RFID promising. The multi-mode standardized shipping containers, invented around 1956, revolutionized worldwide logistics as they can be transported on trains, trucks, ships, and airplanes. However, using paper forms to track these containers and their contents proved to be expensive and problematic. RFID effectively provided a solution for such a challenge. In the 1990s, the DoD used active tags from Savi Technologies (operated under the ANSI 371.2/ISO 18000-7 standard at 433 MHz) for over a million shipping containers travelling worldwide. In 2003, Wal-Mart mandated its top 100 suppliers to apply RFID tags to all shipments delivered to Wal-Mart by January 2005. The DoD issued similar mandate to their suppliers in 2005. Today, wide usage of RFID is readily seen in the supply chain industry.

RFID has also been useful to track people in addition to valuable assets, such as railcars, vehicles, and shipping containers. Personnel access control in corporate buildings required short-range, inductively coupled employee badges or access cards to enter the buildings. Healthcare facilities used RFID to track and monitor its patients. However,

using RFID to track people runs squarely into privacy issues and it has been a controversial topic of debate.

A significant milestone in the history of RFID was the launch of the MIT Auto-ID center in 1999. Researcher David Brock and Sanjay Sarma envisioned the concept of “Internet of Things”—low-cost tags that contained only a ubiquitous, globally unique EPC to identify every manufactured product. With the identified EPC, information about the object in a database can be accessed through the internet. Their work resulted in two first-generation air-interface Class 0 and Class 1 standards, the EPC numbering scheme, and a network to look up information about objects. In October 2003, the Auto-ID center licensed the technology to UCC (GS1 US) which created EPCglobal as a joint venture with the EAN international (GS1). The EPCglobal seeks to establish worldwide standard and adoption of the EPC technology and EPCglobal network, while the Auto-ID labs continued research in RFID. The Class 0 and Class 1 standards are important for the supply chain industry; but unfortunately, they were mutually incompatible. In 2004, the EPCglobal developed a new protocol Class 1 Generation 2 for passive UHF RFID. In 2006, it was ratified by the ISO as ISO 18000-6C. This Gen 2 protocol, of which reader will read more about in Chapter 2, is significant for the supply chain management industry. It paved the way for large-scale deployment of passive UHF RFID and may become the globally accepted standard.

The 21st century sees an explosion of RFID technology in both traditional and new application areas (such as healthcare, item management, etc) driven by inexpensive, low power modern ICs and globally accepted protocol standards. Novel developments such as integration of sensors for sensing applications, low-power microcontrollers for computation, etc will expand the forefront of the technology. The future of RFID is very promising. The realization of its full potential is only limited by our imagination.

1.4 Tag Classifications

Tags are often classified by power, frequency, and protocols. Paper [21] includes a comprehensive review of tag classifications, suggesting categories based on other characteristics such as data processing, programmability, etc.

1.4.1 Passive, Semi-passive, and Active Tags

Based on the source of power supply, tags are categorized into passive, semi-passive, and active. Table 1-1 shows a brief comparison of these three types.

Table 1-1: Tag Comparisons

Type	Power Source	Communication Method	Advantages	Disadvantages
Passive	RF field of the reader	Backscatter modulation	<ul style="list-style-type: none"> • Battery-free • No transmitter • Low power consumption • Low cost/maintenance 	<ul style="list-style-type: none"> • Short operating range (~2-10m) • Limited computational capability
Semi-passive	RF field of the reader & battery	Backscatter modulation	<ul style="list-style-type: none"> • Higher power budget allows complex designs • Long operating range • Long battery life (~ yrs) 	<ul style="list-style-type: none"> • Medium cost • Large size • Battery constrained • Maintenance
Active	Battery	Active radio transmission	<ul style="list-style-type: none"> • Allow advanced modulation schemes • Long operating range (1km) • Powerful processing capabilities 	<ul style="list-style-type: none"> • Power hungry, short battery lifespan (~ weeks, months) • Expensive • Bulky

Passive tags are batteryless devices. They don't have an on-board battery to power their circuitries, but rely completely on the power provided by the reader. Unlike conventional wireless sensor nodes, passive tags don't contain radio transmitters, which are power hungry; instead, they modulate the reflection of the RF wave emitted by the reader (using backscatter modulation) to transmit data. As a result, passive tags consume little power. They are very inexpensive to manufacture and virtually require no maintenance—they can work as long as their electronic components work. Due to these advantages, they are very popular in today's market. However, passive tags suffer relatively short operating range on the order of 2-10m. Their communication links are asymmetrical and forward-link limited due to the fact that tags have less sensitivity (-20 dBm) compared to the sensitive reader (-80 dBm). This is because tags are power constrained while the reader is not. Because of that, passive tags have limited computational capability compared to the other two types.

Semi-passive tags are battery-assisted devices. They contain an on-board battery to power their circuitries but like the passive tags, they use backscatter modulation for data transmission. Due to increased power budget provided by the battery, semi-passive tags allow more complex IC designs and RF functions. Compared with the active tags, they have relatively longer battery lifespan (on the order of years) because the data transmission is not powered by the battery. In addition, semi-passive tags can achieve read range in the 10s of meters. However, they are more expensive in terms of large size (battery area) and maintenance cost.

Active tags are completely powered by the battery for both their IC circuitries and data transmission. They are basically full-fledged radios with sophisticated transmitters and receivers. With complex RF functions, they can implement advanced modulation schemes which often provide superior noise robustness. Active tags can achieve read range

up to 1 km and support powerful microprocessors. However, due to power hungry radio and processing components, active tags suffer from short battery lifespan ranging from weeks to months. They are typically expensive, large in size, and involve complex circuitries.

1.4.2 Inductive vs. Radiative Coupling

Another common way tags are categorized is by their operating frequency. Figure 1-2 shows the common frequency bands for RFID—125/134 KHz falls in the low frequency (LF), 13.56 MHz in the high frequency (HF), 860-960 MHz in the ultra-high-frequency (UHF), and 2.4-2.45 GHz in the microwave band. It also shows their corresponding wavelength (λ) which is defined as:

$$\lambda = \frac{c}{f} \tag{1.1}$$

where c is the speed of light (3×10^8 m/s) and f is the frequency.

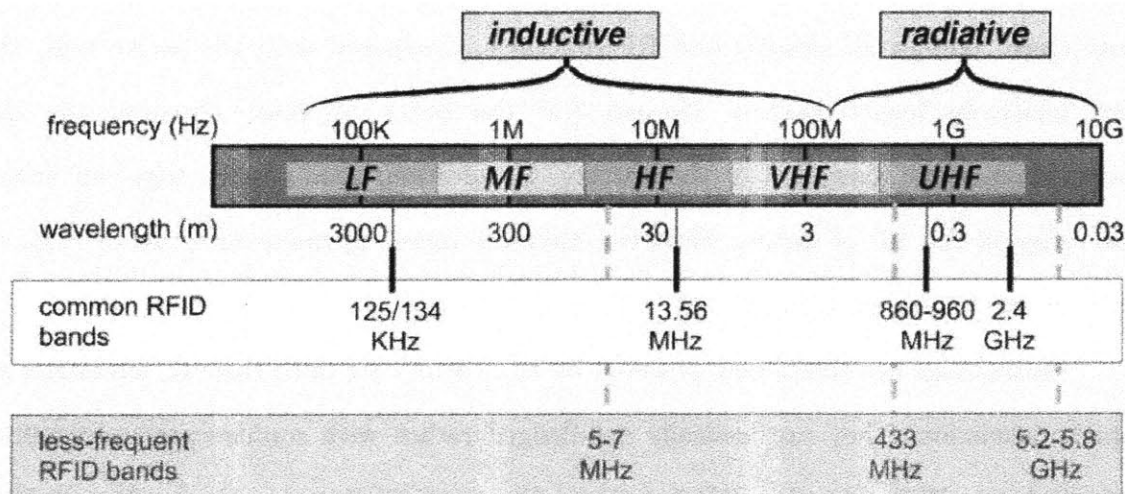


Figure 1-2: Frequency band for RFID systems. Adapted from [6]

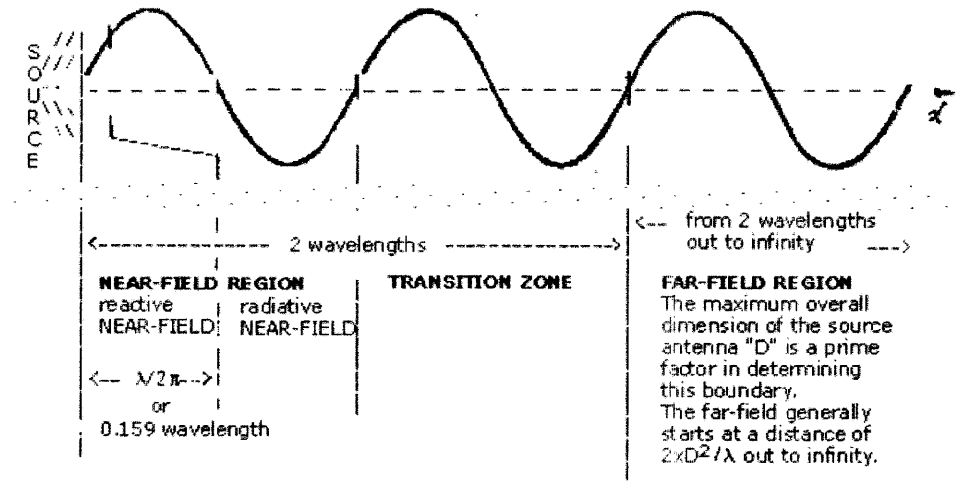
Tags operating in the different frequencies demonstrate different coupling behaviors. Generally, the LF and HF tags are inductively coupled using magnetic fields whereas the UHF and microwave tags are radiatively coupled using electromagnetic fields. This distinction is primarily determined by their wavelength (ranging from 12cm to 2400m) in comparison to their antenna size. Tags with wavelength much larger than their antenna size are inductively coupled and tags with wavelength comparable to their antenna size are radiatively coupled. The inductive coupling acts effectively as a magnetic transformer since the traveling time is such small fraction of the wave period. For the radiative coupling, however, the wave propagation takes several cycles to reach the tag and back.

Inductive tags operate in the near-field of the reader antenna where power is transferred by magnetic (H) fields whereas radiative tags operate in the far-field where power is transferred by EM waves. As shown in the Figure 1-3 , near-field is approximately one λ out from the antenna (some papers use $\lambda/2\pi$, the reactive near-field region) and far-field is approximately 2λ . In the near-field region, high magnetic energy is held back and stored (not radiated). The magnetic field strength falls rapidly as $1/d^3$. In this region, the relationship between the E and H fields is very complex and the power density S (W/m²) is difficult to predict. All polarization types (circular, elliptical, linear) may be present.

In the far-field region, there's no reactive energy stored and the magnetic field strength drops as $1/d$ (power density drops off as $1/d^2$). Only one polarization type is present in the region. The E, H, and S can be easily predicted with defined relationships

$$\mathbf{E} = \mathbf{H} \cdot \sqrt{\frac{\nu_0}{\epsilon_0}} \quad \text{and} \quad \mathbf{S} = \mathbf{E} \cdot \mathbf{H} \quad (1.2)$$

where free space permeability $\mu_0 = 4\pi \times 10^{-7} \text{ H/m}$, permittivity $\epsilon_0 = 8.85 \times 10^{-14} \text{ F/cm}$, and the characteristic impedance $z_0 = \sqrt{\frac{\mu_0}{\epsilon_0}} = 120\pi \Omega$.



Antenna field Regions for Typical Antennas

Figure 1-3: Antenna near-field far-field regions. Adapted from [23]

As a consequence of this rapid H field strength fall off, inductive tags can only be read within a short distance (~10s of cm) whereas radiative tags offer greater range, up to 10m. Inductive tags use antenna coils and typically have slow data rate, around 1 Kbps, whereas radiative tags often use dipole antenna and can achieve data rate of 100s of Kbps. However, the advantage of using inductive tags is that it works well with metallic objects and water whereas radiative tags don't. Radiative systems tend to have more multiple reader interference than the inductive counterparts. These comparisons are summarized in the Table 1-2.

Table 1-2: Inductive vs. Radiative Coupling Comparisons

Coupling Method	Frequency	Antenna Configuration	Advantages	Disadvantages
Inductive	LF, HF	Coils (many turns for LF, fewer for HF)	<ul style="list-style-type: none"> • Works well with metal and water • Suitable for short-range applications 	<ul style="list-style-type: none"> • Short read range (10s of cm) • Slow data rate (1kbps for LF, 10s of kbps for HF) • Expensive for antenna windings
Radiative	UHF, microwave	Dipole (~16cm for UHF, 6cm for microwave)	<ul style="list-style-type: none"> • Longer read range (~10m, limited by transmit power) • Fast data rate (100s Kbps) 	<ul style="list-style-type: none"> • Not work well with metal and water • Multiple reader interference

1.4.3 Protocols

Tags are also categorized by the protocol standards they use. As shown in the Figure 1-4, there exist many standards across all common bands for passive, semi-passive, and active tags. Most of these standards are incompatible to each other. Some of these are proprietary and exist for a particular application while others exist simply because of historical reasons. Since 2006, for passive tags in the 860-960 MHz, EPC Class 1 Gen 2 (ISO-18000-6C) has become the most common as it improves upon EPC Class 0 (ISO 18000-6A) and Class 1 Gen 1 (ISO 18000-6A).

Tag type:	Frequency					
	125/134 kHz	5-7 MHz	13.56 MHz	303/433 MHz	860-960 MHz	2.45 GHz
Passive	ISO 11784/5, 14223 ISO18000-2 HiTag	ISO10536 iPico DF/iPX	MIFARE ISO14443 Tag-IT ISO15693 ISO18000-3 TIRIS Icode		ISO18000-6A,B,C EPC class 0 EPC class 1 Intellitag Title 21 AAR S918 Ucode	ISO18000-4 Intellitag μ -chip
Semipassive					AAR S918 Title 21 EZPass Intelleflex Maxim	ISO18000-4 Alien BAP
Active				ANSI 371.2 ISO18000-7 RFCCode		ISO18000-4 ANSI 371.1

Figure 1-4: RFID standards across all common frequency bands. Adapted from [6].

1.5 Summary

It is apparent to see there are tradeoffs with passive, semi-passive, and active tags across different frequency bands with different standards. Passive tags are low cost, suited for labeling with inexpensive objects. Semi-passive tags are more expensive while active tags are most expensive, suited for expensive, important items. Low frequency offers good working compatibility with water and metal objects but suffers from short read distance and low data rate. Higher frequency offers longer read range, higher data rate, smaller antenna size, but suffers from working with metal and water objects.

For health care applications where sensors are used for monitoring, it is sometimes advantageous to use semi-passive or even active tags. But when minimum cost and batteryless power are required, passive tags are more suited. In fact, hybrid tags where low-power backscatter modulation technique is utilized and ultracapacitors are used to store energy to power the sensors and internal circuitries can offer the most advantages.

Chapter 2 UHF Gen 2 Protocol

Protocol is an agreement between the reader and tag on how information is exchanged. For passive RFID systems, careful attention needs to be paid to minimize the demands on limited power and computational capability of the tag. In the early days, different vendors used different protocols which made cross-vendor readers and tags incompatible—tags from one vendor couldn't talk to a reader made by another vendor and vice versa. This created a practical problem as RFID systems were widely deployed in the supply chain of a world market. From 1999 to 2003, the MIT AutoID Center played an important role in standardizing the various protocols. Early standards included Class 0, Class 1 Gen 1, ISO 18000-6A, -6B, etc. Since 2006, the EPC Class 1 Generation 2, also known as ISO 18000-6C, has become the globally accepted standard for passive UHF RFID systems.

To implement Gen 2 on the FPGA, a thorough understanding of its intricacies is needed. This chapter focuses on the most important and relevant aspects of the protocol. A top-down approach is used where the link layer will be discussed before the physical layer. The discussions are not intended to be comprehensive. Readers are encouraged to explore the specification details in [24]. However, it is proven to be difficult to navigate through such an elaborate document and gain an understanding how it works. Therefore, materials are organized in a systematic and logical manner so that they can most efficiently help readers understand.

2.1 Gen 2 Data Link Layer

The reader manages communication with a tag using three basic operations: *Select*, *Inventory*, and *Access*. Figure 2-1 shows a reader using these operations to interact with a tag implementing several logical states.

The reader uses the *Select* operation to select a tag population for inventory and access. Through *Inventory*, a unique tag can be identified by its EPC. *Access* allows the reader to read from or write to an individual tag's memory.

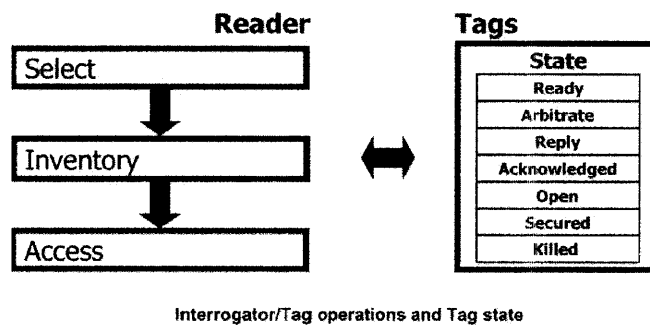


Figure 2-1: Reader communicates with tags implementing several states. Adapted [24].

2.1.1 Select Operation

The *Select* operation consists of only one command: *Select*, which is used to select a subset of a tag population by using a mask. In this work, there is only one tag the reader communicates with. Therefore, it is optional to implement. However, it will be briefly described for completeness.

The structure of the *Select* command is shown in the Table 2-1. The Target parameter specifies which of the five flags—inventoried flags in each of the four sessions and the select flag (SL) —it will target to modify. RFU is reserved for future usage.

Action is a 3-bit field specifying eight possible actions to be taken by the tag depending on whether the tag's memory data matches the Mask bits (a variable bit sequence up to 256 bits). The MemBank, Pointer, and Length specify which bank of the tag's memory (there're four which will be discussed later), the exact location within the bank, and the bit length of the data to be matched to the Mask.

These parameters basically allow user to select the tags whose memories contain certain data. For example, users can select the tags whose EPCs start with the hex AA. EBV is an extensible bit vector which extends the size of the reference to the memory address. This will be discussed in the tag memory section. The 1-bit Truncate indicates whether the backscattered reply will be truncated to include only an abbreviated EPC. The CRC-16 is a 16-bit cyclic-redundancy check that appends at the end of the command to ensure validity.

Table 2-1: Select command and tag's response to Action

Select command

	Command	Target	Action	MemBank	Pointer	Length	Mask	Truncate	CRC-16
# of bits	4	3	3	2	EBV	8	Variable	1	16
description	1010	000: inventoried (S0) 001: inventoried (S1) 010: inventoried (S2) 011: inventoried (S3) 100: SL 101: RFU 110: RFU 111: RFU	See Table 6.20	00: RFU 01: EPC 10: TID 11: User	Starting Mask address	Mask length (bits)	Mask value	0: Disable truncation 1: Enable truncation	

Tag response to Action parameter

Action	Matching	Non-Matching
000	assert SL or inventoried → A	deassert SL or inventoried → B
001	assert SL or inventoried → A	do nothing
010	do nothing	deassert SL or inventoried → B
011	negate SL or (A → B, B → A)	do nothing
100	deassert SL or inventoried → B	assert SL or inventoried → A
101	deassert SL or inventoried → B	do nothing
110	do nothing	assert SL or inventoried → A
111	do nothing	negate SL or (A → B, B → A)

As shown in the Action Table, there are eight possible tag actions. For example, if in the *Select* command the Action is 000, and the content of the memory location of the tag specified by the MemBank, Pointer, and Length matches the Mask specified by the user, a matching occurs. Depending on the Target value, the tag will either assert SL flag or flip the inventoried flag to A. If a non-matching occurs, the tag will deassert SL or flip the inventoried flag to B.

The reader can issue multiple *Select* commands to perform a complex Boolean logic, such as union (OR), intersection (AND), XOR, to include or exclude certain tags. It simply selects a sub population and the tags don't reply.

2.1.2 Inventory Operation

After the *Select* operation, the *Inventory* operation is initiated. This operation basically uniquely identifies a single tag (singulation) among the selected population and establishes subsequent access to it.

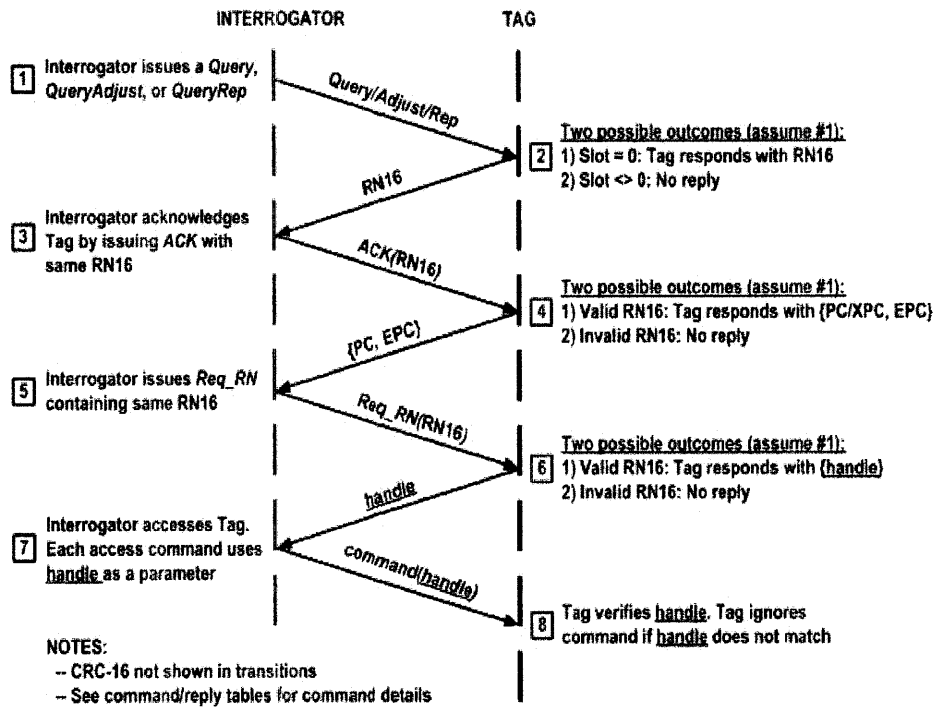
Figure 2-2 illustrates a basic session of tag inventory and access. For now we will focus on the general operations. Details of the commands will be discussed later.

Actually before even jumping into the general operations, we will first look at the slotted Aloha (or the Q-protocol) Gen 2 uses to singulate a tag, which differs from the Gen 1 binary tree variants. This is how it works:

- The reader specifies the number of slots in a round by a Q number
- Each tag randomly rolls die and selects a slot
- If a tag has chosen the first slot in the round, it replies with a 16-bit random number (RN16) while other tags remain silent
- Upon receiving this RN, if the reader can acknowledge it, the tag sends its EPC

- With a proper handle, the reader can maintain an access session with the tag and is able to read from and write to the tag's memory

Example inventory and access of a single Tag



Example of Tag inventory and access

Figure 2-2: Inventory and access of a single tag

With this understanding, let's look at Figure 2-2 again. Beginning at step 1, the reader issues a *Query* command which contains the Q parameter that sets the number of slots. The advantage of slotted Aloha is that this Q number can be dynamically adjusted by the user in the reader software so that in the case there is known numbers in the read zone, it can be specified so that the response time can be improved. Upon receiving *Query* at step 2, tag randomly picks a value between 0 and $2^Q - 1$ inclusively and loads it into its

slot counter. If the value is not zero, the tag doesn't reply and waits for the reader to issue either a *QueryAdjust* which adjusts the Q or a *QueryRep* which decrements the tag's slot counter until it reaches zero. When this happens, the tag responds by backscattering a RN16. After receiving this RN16 at step 3, the reader issues an *ACK* which contains this same RN16. If the tag receives an invalid RN16 at step 4, it ignores the command. Otherwise, it replies by backscattering a Protocol Control (PC) and EPC which will be discussed later in the tag memory section. At step 5, the *Inventory* operation is complete. The reader may choose to terminate by issuing a *Query*, *QueryAdjust*, *QueryRep* or access the tag memory by issuing a *Req_RN* command.

Now, let's look at the details of these commands and tag's reply to them. The *Inventory* operation consists of *Query*, *QueryAdjust*, *QueryRep*, *ACK*, and *NAK* commands.

Query initiates a new inventory round. It contains parameters, as shown in the Table 2-2, that specify the tag-to-reader (T=>R) backscatter link frequency (BLF), encoding format (thus the data rate), flag status, session, and the Q, which sets the number of slots (2^Q) for the round.

QueryAdjust command adjusts (increment, decrement, or no change) the Q without changing any other parameter whereas *QueryRep* instructs the tag to decrement its slot counter. The tag will not reply if it has not previously received a *Query*, or the command doesn't match the session number specified by the previous *Query*. Details of both commands are shown in the Table 2-2.

Tag's reply to all three commands is the RN16.

Table 2-2: *Query*, *QueryAdjust*, and *QueryRep* command structures and tag reply

Query command

	Command	DR	M	TRExt	Sel	Session	Target	Q	CRC-5
# of bits	4	1	2	1	2	2	1	4	5
description	1000	0: DR=8 1: DR=64/3	00: M=1 01: M=2 10: M=4 11: M=8	0: No pilot tone 1: Use pilot tone	00: All 01: All 10: ~SL 11: SL	00: S0 01: S1 10: S2 11: S3	0: A 1: B	0-15	

QueryAdjust command

	Command	Session	UpDn
# of bits	4	2	3
description	1001	00: S0 01: S1 10: S2 11: S3	110: Q = Q + 1 000: No change to Q 011: Q = Q - 1

QueryRep command

	Command	Session
# of bits	2	2
description	00	00: S0 01: S1 10: S2 11: S3

Tag reply to *Query*, *QueryAdjust*, *QueryRep*

	Response
# of bits	16
description	RN16

The *ACK* command acknowledges the tag with the RN16 it previously sends to the reader. The tag replies with PC/EPC up to 528 bits. The *NAK* doesn't acknowledge the tag. Any tag receives this command simply change its state (or in some cases doesn't change state) and will not reply. The structures of both commands and tag reply are shown in the Table 2-3.

Table 2-3: *ACK* and *NAK* command structures

***ACK* command**

	Command	RN
# of bits	2	16
description	01	Echoed RN16 or <u>handle</u>

Tag reply to a successful *ACK* command

	Response
# of bits	21 to 528
description	See Table 2-6

***NAK* command**

	Command
# of bits	8
description	11000000

2.1.3 Access Operation

After the *Inventory* operation has completed in step 5, the reader may choose to access the tag's memory by issuing a *Req_RN*. This command, as shown in the Table 2-4, requests the tag to generate a new RN16, or the handle. At step 6 the tag replies with a new handle after it receives the correct RN16 sent by the *Requ_RN*. Notice this reply differs from that of *Query*, *QueryAdj*, or *QueryRep* by the CRC16 check. If the received RN16 is incorrect, the tag simply ignores the command.

Table 2-4: *Req_RN* command structure and tag reply

***Req_RN* command**

	Command	RN	CRC-16
# of bits	8	16	16
description	11000001	Prior RN16 or <u>handle</u>	

Tag reply to a *Req_RN* command

	RN	CRC-16
# of bits	16	16
description	<u>handle</u> or new RN16	

At step 7 after receiving the new handle, the reader can issue commands to access the tag's memory. These access commands include *Read*, *Write*, *Kill*, *Lock*, *Access*, *BlockWrite*, *BlockErase*, and *BlockPermalock*. The only relevant command for this thesis is *Read* and there's no point to kill or lock my only tag.

The *Read* command is shown in the Table 2-5. Along with the handle and the CRC-16 check, *Read* contains the MemBank, WordPtr, and WordCount parameters. The MemBank specifies which of the four banks to be accessed. WordPtr points at the starting word address of the memory bank. WordCount specifies the number of the 16-bit words to be read. The tag replies with a 1-bit header, the accessed memory words, the handle, and the CRC-16 check.

Table 2-5: *Read* command structure and tag reply

Read command

	Command	MemBank	WordPtr	WordCount	RN	CRC-16
# of bits	8	2	EBV	8	16	16
description	11000010	00: Reserved 01: EPC 10: TID 11: User	Starting address pointer	Number of words to read	handle	

Tag reply to a successful *Read* command

	Header	Memory Words	RN	CRC-16
# of bits	1	Variable	16	16
description	0	Data	handle	

2.2 Tag Memory

As shown in the Figure 2-3, the tag memory is organized into four banks, each of which contains zero or more 16-bit memory words. They are Reserved, EPC, Tag ID (TID), and User memory banks at binary 00, 01, 10, and 11 locations, respectively.

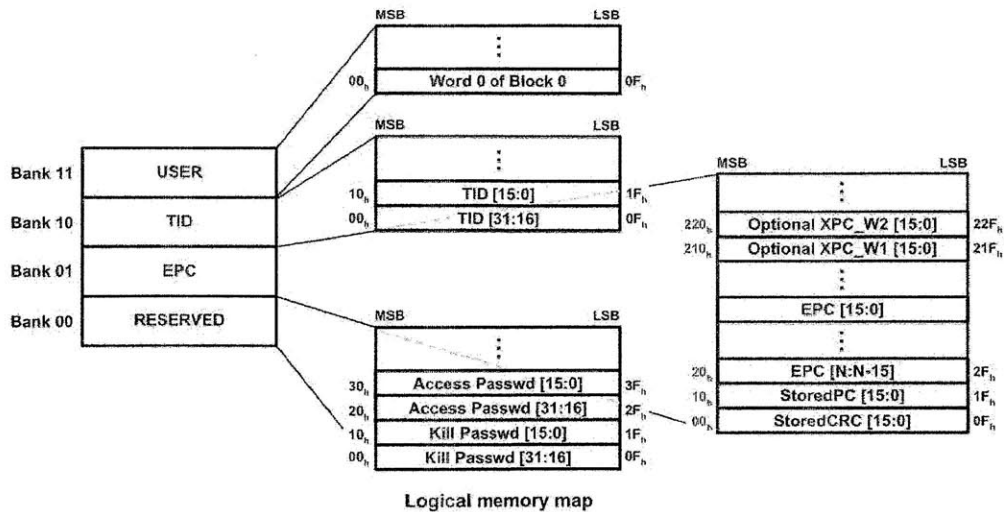


Figure 2-3: Tag memory organization

2.2.1 EBV-8 Scheme

Notice that the *Select* and *Read* commands contain parameters—Pointer in the *Select* and WordPtr in the *Read*, which use the extensible bit vector (EBV). The EBV is a flexible format used to select a particular location in the memory bank. It is a data structure with an extensible data range that allows the memory to be of arbitrary size. As indicated by the dots in the Figure 2-3, memory banks are not constrained to the size shown.

Gen 2 uses the 8-bit EBV scheme where the first bit is the extension bit and the rest seven bits are the data. For example, as shown in the Figure 2-4, since the extension bit is 0, the first three rows are non-extended and the data bits contain the complete address. When the extension bit is 1, the next three rows are extended. The last two rows are calculated as $2^{14} - 1 = 16383$ and $2^{14} = 16384$.

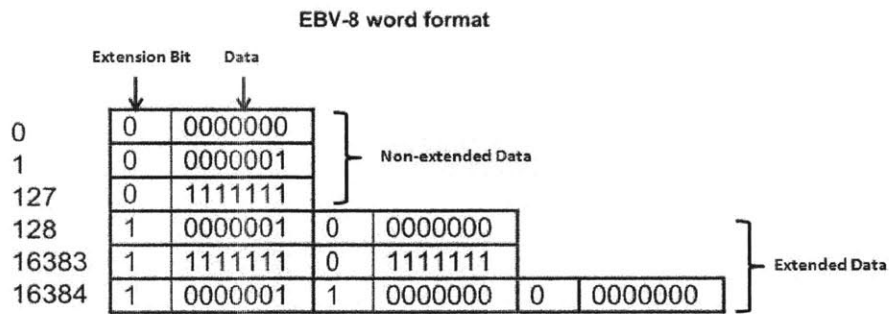


Figure 2-4: EBV-8 examples

2.2.2 Reserved Memory

Refer to Figure 2-3. The Reserved memory bank holds the 32-bit Kill and Access passwords, stored from 00_h to $1F_h$ and 20_h to $3F_h$, respectively. This bank is useful when

data encryption is implemented on the tag. If passwords are not implemented, by default they are all set to zeros.

2.2.3 EPC Memory

The EPC memory bank is where the EPC which uniquely identifies a product resides. It holds the StoredCRC, StoredPC, EPC, and the two optional XPC words at the locations shown in the Figure 2-3. The StoredCRC is the CRC-16 value that the tag calculates over the StoredPC, EPC and stores in the EPC memory at power-up. It's used to protect the backscattered data. Tag that supports XPC should implement PacketCRC which is calculated dynamically over the backscattered PC word, optional XPC word, and EPC.

The structure of the 16-bit StoredPC is shown in the Figure 2-5. It primarily determines the bit length of the EPC. Therefore, for a standard 96-bit EPC, the default StoredPC value is hex 3000_h as shown in the figure.

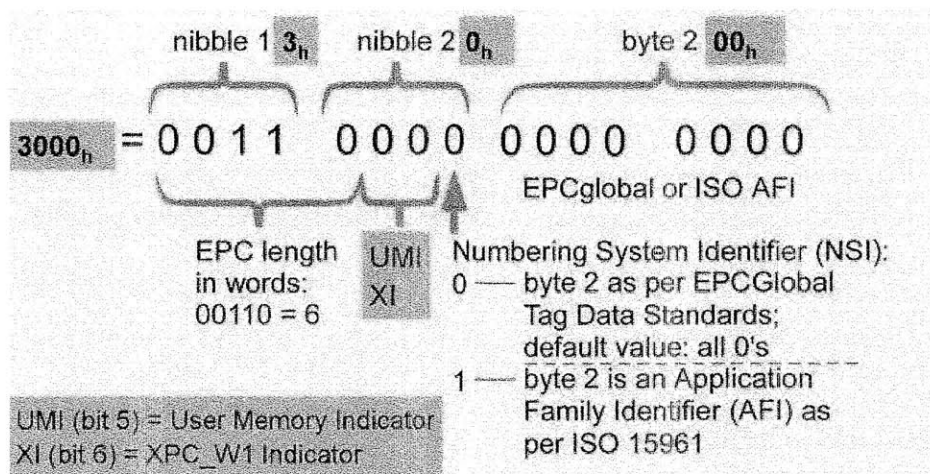


Figure 2-5: StoredPC structure

The StoredPC comprises the EPC length field from locations 10_h to 14_h , a user-memory indicator (UMI) at 15_h , an XPC_W1 indicator (XI) at 16_h , and a numbering system identifier (NSI) from 17_h to $1F_h$. Tag that supports XPC should implement a PacketPC which is different from the StoredPC in its EPC length field.

The maximum 5-bit EPC length is 11111_2 or 31 words, which allows 496-bit EPC. In PacketPC, the max EPC length is 11101_2 or 29 words which allow 464-bit EPC. The difference of 2 words is used for the two optional XPC words.

The UMI bit 15_h indicates whether the User memory bank is used or contains information. The XI bit 16_h indicates whether the optional XPC_W1 is implemented. Implementing XPC_W1 is important only when there is a need to recommitment tags. In the case where Kill command is not used, there's no need to implement this optional word. The NSI MSB at 17_h indicates whether it's an EPCglobal application. If the bit is 0, the following 8 bits is defined per the EPCglobal Tag Data Standards [25]. Otherwise, it is a non-EPCglobal application and the 8 bits contain application family identifier (AFI) defined in the ISO/IEC 15961.

The EPC structure is defined in the EPCglobal Tag Data Standards [25]. Interested readers should explore the details in the specification document.

2.2.4 TID Memory

The Tag ID memory bank contains information about the tag's manufacturer and the tag itself instead of the object it attaches to. Specifically, the first 8-bit location 00_h to 07_h contains one of the two ISO/IEC 15963 class identifiers, either $E0_h$ or $E2_h$, assigned by the tag manufacturer.

If the identifier is $E0_h$, the second 8-bit location 08_h to $0F_h$ contains an 8-bit manufacturer ID. The following 48-bit location 10_h to $3F_h$ contains the tag serial number. Together, they make a unique 64-bit Tag ID.

If the identifier is $E2_h$, the next 12-bit location 08_h to 13_h contains the tag's mask designer ID. The following 12-bit 14_h to $1F_h$ contains a vendor-defined tag model number. Usage of locations above $1F_h$ is defined in [25].

It's important to note the difference between TID and EPC memory banks--that TID contains information about the tag itself whereas EPC identifies the product the tag is attached to. They are distinct and different.

2.2.5 User Memory

The user memory bank provides opportunity to store user specific data. In the case of sensor integrated tags, this is the location to store all sampled data. The memory size can be arbitrarily large, only constrained by the physical size limitation.

2.2.6 Reply to the ACK Command

Refer to step 4 in the Figure 2-2 again. After receiving the ACK command with the verified RN16, if the tag doesn't implement the optional XPC word which is often the case, the tag will only send PC, the full EPC, and CRC-16. Otherwise, the tag response is based on the Table 2-6. In the case of a single tag where XPC is not implemented, only the first row of the table is relevant.

Table 2-6: Tag response to the ACK command

Tag data and CRC-16 backscattered in response to an *ACK* command

XI	XEB	Truncation	Tag Backscatter			
			PC	XPC	EPC	CRC-16
0	0	Deasserted	StoredPC	None	Full	StoredCRC or PacketCRC
0	0	Asserted	00000 ₂	None	Truncated	StoredCRC
0	1	Deasserted	Invalid ¹			
0	1	Asserted	Invalid ¹			
1	0	Deasserted	PacketPC	XPC_W1	Full	PacketCRC
1	0	Asserted	00000 ₂	None	Truncated	StoredCRC
1	1	Deasserted	PacketPC	Both XPC_W1 and XPC_W2	Full	PacketCRC
1	1	Asserted	00000 ₂	None	Truncated	StoredCRC

Note 1: XI is the bitwise logical OR of the 16 bits of XPC_W1, and XEB is the MSB (bit F₁) of XPC_W1, so if XEB=1 then XI=1.

2.3 Tag States

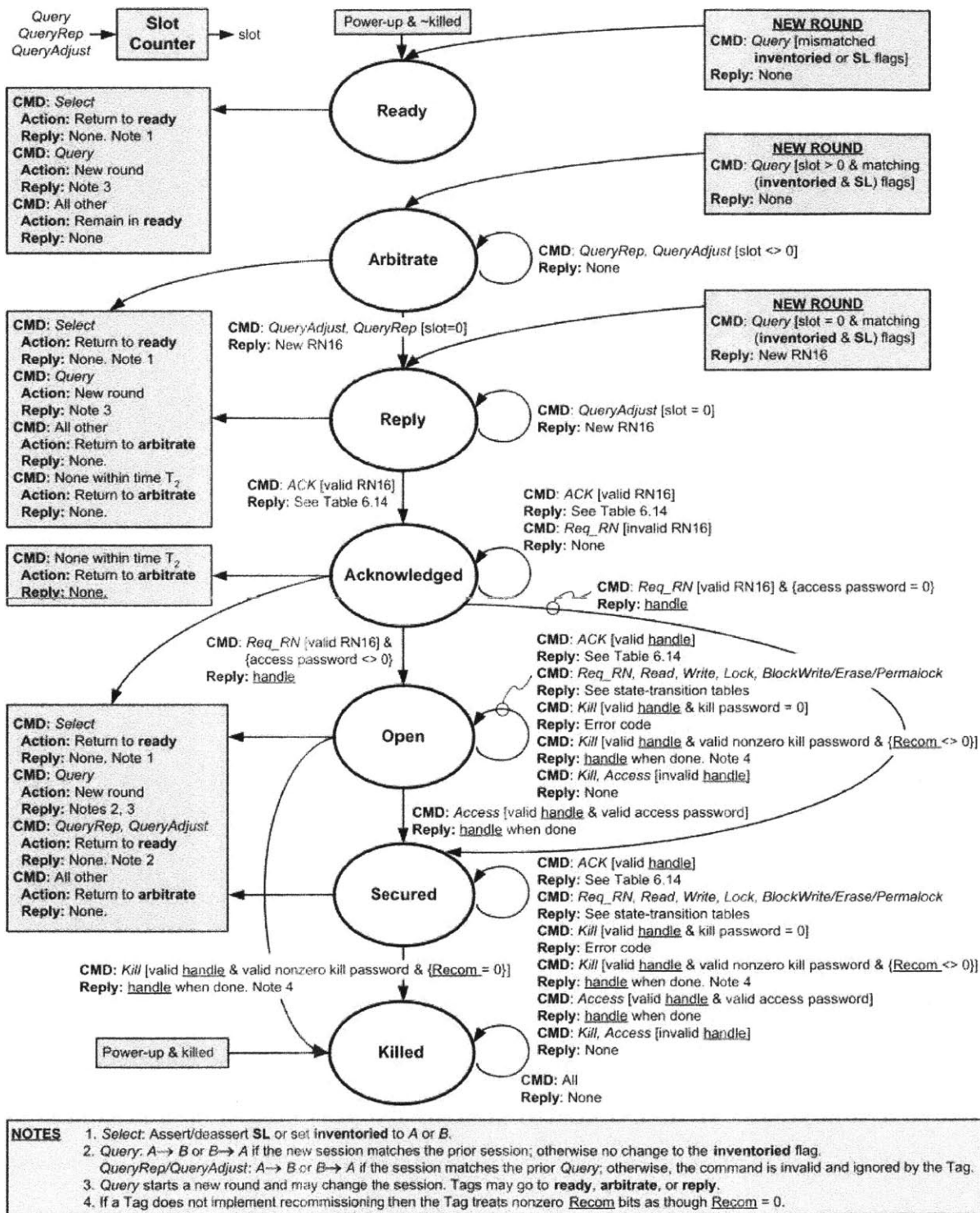
While communicating with the reader, the tag implements several logical states. Figure 2-6 shows the tag state diagram. At the first glance, the most straightforward path of execution is from Ready to Arbitrate to Reply to Acknowledged to Open/Secured and finally to Killed. In this work of a single sensor tag, the Killed state is not implemented.

Upon entering an energizing RF field and being powered up, the tag enters the Ready state. It stays there until it receives a *Query* command which contains Target (Inventoried flag) and Sel parameters that match the tag's flag values. The matched tag draws a random number in the 0 to 2^Q-1 range from its pseudo random number generator (RNG). It loads this RN into its slot counter and transition to Reply if it's zero or Arbitrate if it's nonzero. Tag in the Arbitrate will remain there until it receives a *QueryRep* that decrements its slot counter to zero. When this happens, it sends a RN16 to the reader while transitioning to Reply. In the Reply state, if the tag receives an *ACK* with the valid RN16, it transitions to the Acknowledged state while sending PC and EPC

to the interrogator according to the Table 2-6. If it doesn't receive *ACK* within a certain timing constraint T_2 , or the RN16 is incorrect, or receive other commands except *Select* and *Query*, the tag will return to the Arbitrate state.

In the Acknowledged state, the tag can transition to any other state except the Killed state. If the tag receives *Req_RN* with the valid RN16 and the access password is zero as in the default case, it transitions to the Secured state. This is often the case when password is not implemented on the tag. If the access password is coded and is nonzero, the tag transitions to the Open state. For either case, the tag replies with the new RN16 handle during the transitioning.

Depending on the received command, tag in the Secured state can transition to any state except the Acknowledged and Open states. Tag in the Open state can transition to any state except the Acknowledged state. Tag in the Open and Secured state can be killed by the Kill command. It's important to note that killing a tag is irreversible. In the case of a single tag communication, there's no point to kill the only tag.



Tag state diagram

Figure 2-6: Tag state diagram

2.4 Gen 2 Physical Layer

2.4.1 Operating Frequency

UHF RFID operates in the 860-960 MHz depending on the regions. In the US under the FCC regulations, RFID uses the Industrial, Scientific, and Medical (ISM) band of 902-928 MHz with 915 MHz center frequency. In Europe regulated by ETSI, RFID uses 865-868 MHz band.

The US industry has converged on 915 MHz tags for commercial deployment due to a balance between tag cost (increase with higher frequency), size (smaller with increased frequency), and operating distance (increase with higher frequency).

2.4.2 Reader-to-Tag (R=>T) Communications

2.4.2.1 Modulation

The reader communicates with the tag by modulating an RF carrier using amplitude shift keying (ASK). The three modulation schemes used in the Gen 2 RFID systems are double-sideband (DSB), single-sideband (SSB), or phase-reversal (PR).

2.4.2.2 Data Encoding

The R=>T link uses the pulse-interval data encoding (PIE) as shown in the Figure 2-7Figure 2-1. PIE is advantageous over simple on-off keying (OOK) as it ensures power is delivered for each symbol. Data-0 is encoded by a high interval—transmitted continuous wave (CW) followed by a low pulse width (PW)—attenuated CW. Data-1 is

encoded by a longer high interval followed by the same length PW. Tari (ranging from 6.25 to 25us) is used as the reference time and is the duration of data-0. Common values are 6.25us, 12.5us, and 25us corresponding to the R=>T data rates of 160, 80, and 40 Kbps. PW is normally a half Tari but can vary from 0.265 to 0.525 Tari.

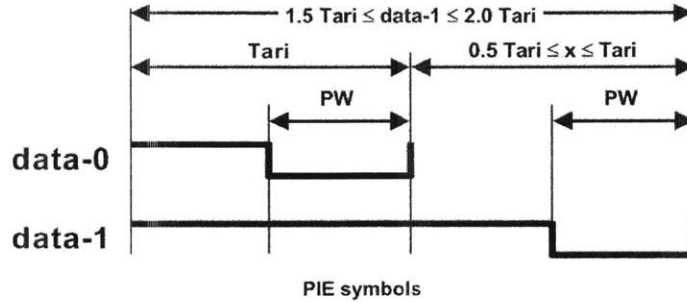


Figure 2-7: PIE symbol for data-0 and data-1

2.4.2.3 Preamble and Frame-Sync

Figure 2-8 shows the preamble and frame-sync symbols. The two symbols are identical except the addition of the T=>R calibration (TRcal) in the preamble. Preamble precedes *Query* command which starts the inventory round while frame-sync precedes all other commands.

The preamble consists of a fixed-length delimiter, data-0, R=>T calibration (RTcal) symbol, and the TRcal symbol. The RTcal, whose length is the sum of the data-0 and data-1, determines the pivot which is RTcal/2. Any subsequent symbol shorter than pivot is interpreted as data-0 and any longer is data-1.

The reader uses TRcal in the preamble to specify the tag's backscatter link frequency (BLF). BLF is calculated as:

$$BLF = DR / TRcal \quad (2.1)$$

where DR is the divide ratio specified in the *Query*. We will discuss more about the BLF in the tag-to-reader communication section.

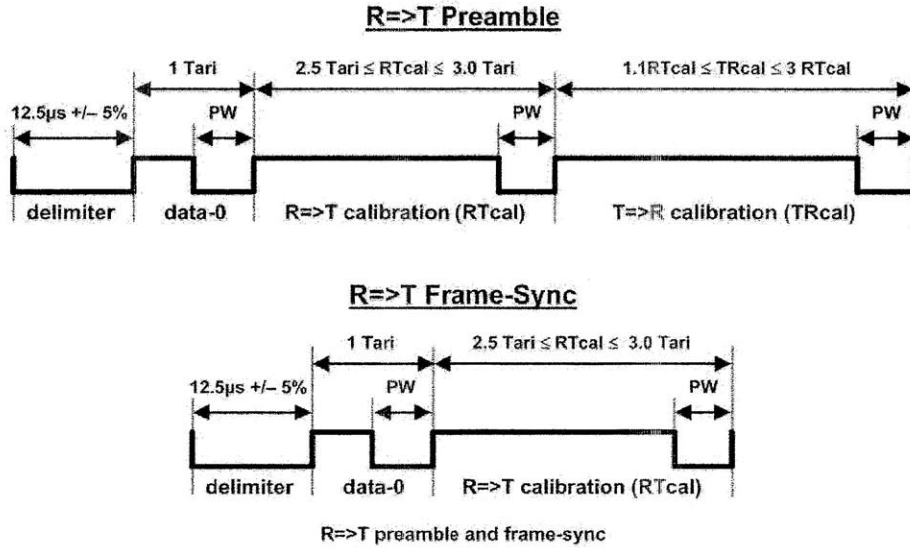


Figure 2-8: R=>T preamble and frame-sync

2.4.3 Tag-to-Reader (T=>R) Communications

The tag uses backscatter modulation to communicate with the reader. It switches the antenna load impedance, thus the reflection coefficient, between two states according to the data being sent. Data-1 causes complete power reflection while data-0 causes power absorption.

2.4.3.1 Modulation

Backscattered data can be modulated using either amplitude-shift keying (ASK) or phase-shift keying (PSK). The choice is usually made by the tag designer.

2.4.3.2 Data Encoding

The backscattered data can be encoded in two formats—FM0 baseband or Miller-modulated subcarrier. The reader selects the encoding format and data rate through the *Query* command that initiates the inventory round. Parameter M (1, 2, 4, or 8) defines the encoding and DR (either 64/3 or 8) defines the data rate, as seen in the equation 2.1.

2.4.3.3 FM0 Baseband

The FM0 symbols are shown in the Figure 2-9. Data-0, as in S2, is encoded with a high and low, changing the phase (state) at the mid symbol. S3 is the inverse of S2 and it also encodes data-0. Data-1 is encoded with a high which is the entire length of data-0. S4 is the inverse of S1 and it encodes data-1.

FM0 has to invert the baseband phase at every symbol boundary, as shown in the Figure 2-9. Otherwise, it is a violation. For example, S2 can't be followed by S3 because phase isn't changed at the symbol boundary. Likewise, S1 can't be followed by another S1. Figure 2-10 shows the state diagram for generating FM0 baseband. Notice there's no transition between S2 and S3, and no condition allows remaining at S1. As shown in the Figure 2-11, FM0 signaling always ends with a “dummy” data-1 at the end of the transmission in the low state. The preamble sequence for FM0 is 1010V1. The violation indicates to the reader that it's a preamble for data encoded in the FM0 baseband.

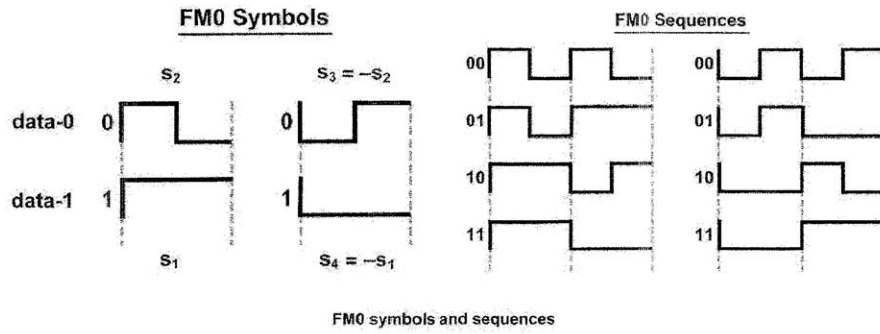


Figure 2-9: FM0 symbols and sequences

FM0 Generator State Diagram

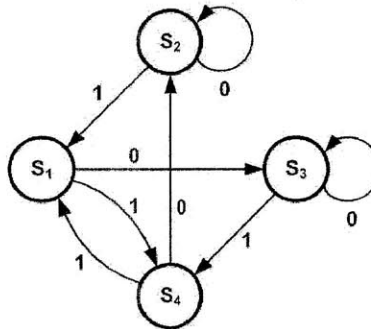
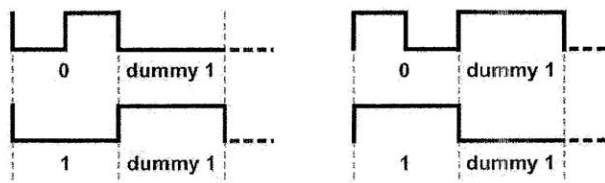


Figure 2-10: FM0 baseband generator state diagram

FM0 End-of-Signaling



Terminating FM0 transmissions

Figure 2-11: FM0 signaling end with a dummy 1

2.4.3.4 Miller-Modulated Subcarrier (MMS)

Figure 2-12 shows the baseband Miller symbols, which are defined opposite from the FM0 symbols—data-1 inverts phase in the mid symbol and data-0 doesn't. It also shows the state diagram for generating Miller baseband sequence. It is important to note that baseband Miller inverts its phase only when two consecutive 0s are in sequence. In other words, phase is not inverted at symbol boundaries in sequences such as two 1s, or between a 1 and a 0.

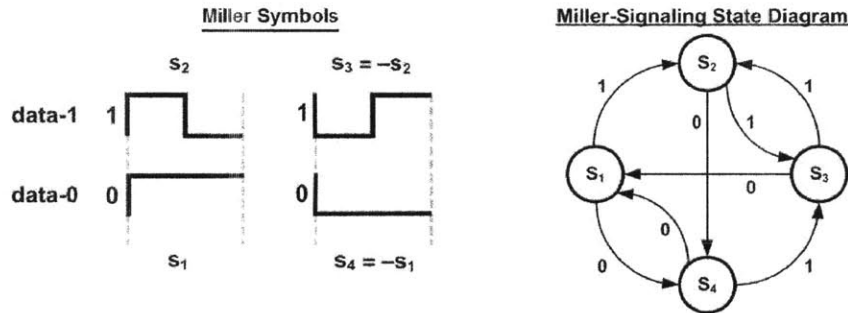


Figure 2-12: Miller symbols and state diagram

Figure 2-13 shows an example of how a MMS sequence is generated from the baseband. Notice that the phase is not inverted at the 0, 1 symbol boundaries. To generate the MMS sequence with $M=2$, the baseband is multiplied by a square wave containing two cycles within a symbol. T_{pri} is the $T \Rightarrow R$ link period, or $1/BLF$. Note that M doesn't change T_{pri} or the BLF ; it simply indicates the number of cycles within every baseband symbol. As shown in the Table 2-7, the data rate of the backscattered MMS sequence is BLF/M . In the case of $M=2$, the data rate is $BLF/2$.

Figure 2-14 shows the encoding for a 2-bit data using FM0 and MMS. Notice with increased M ($M=1$ for FM0), data rate decreases by half while maintaining the same BLF .

There's a strict tradeoff between data rate and noise. With reduced data rate, SNR improves and tag experiences less interference. Similar to FM0, MMS ends with a "dummy" data-1 at the end of transmission in the low state.

MMS Sequence Generation with M=2

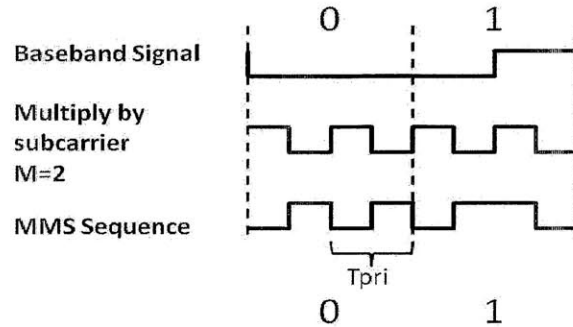


Figure 2-13: MMS sequence generation

Table 2-7: T=>R data rate

Tag-to-Interrogator data rates

M: Number of subcarrier cycles per symbol	Modulation type	Data rate (kbps)
1	FM0 baseband	BLF
2	Miller subcarrier	BLF/2
4	Miller subcarrier	BLF/4
8	Miller subcarrier	BLF/8

Miller Subcarrier Sequences

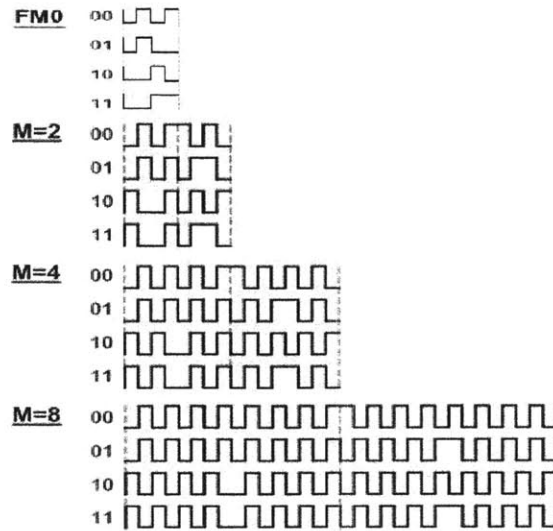


Figure 2-14: Data encoding for a 2-bit data

2.4.3.5 BLF and Data Rate

Recall equation $BLF = DR / TR_{cal}$. Table 2-8 shows how these parameters are related. DR is either 64/3 or 8. Since TR_{cal} can vary from 17.2 to 225us, BLF can range from 40 to 640 KHz. Therefore, data rate (BLF/M) for the T=>R communication can range from 5 to 640 Kbps.

Table 2-8: T=>R link frequencies

Tag-to-Interrogator link frequencies

DR: Divide Ratio	TRcal ¹ (μs +/- 1%)	BLF: Link Frequency (kHz)
64/3	33.3	640
	33.3 < TRcal < 66.7	320 < BLF < 640
	66.7	320
	66.7 < TRcal < 83.3	256 < BLF < 320
	83.3	256
	83.3 < TRcal ≤ 133.3	160 ≤ BLF < 256
	133.3 < TRcal ≤ 200	107 ≤ BLF < 160
	200 < TRcal ≤ 225	95 ≤ BLF < 107
8	17.2 ≤ TRcal < 25	320 < BLF ≤ 465
	25	320
	25 < TRcal < 31.25	256 < BLF < 320
	31.25	256
	31.25 < TRcal < 50	160 < BLF < 256
	50	160
	50 < TRcal ≤ 75	107 ≤ BLF < 160
	75 < TRcal ≤ 200	40 ≤ BLF < 107

Chapter 3 Gen 2 FPGA Implementation

In this work, the Gen 2 protocol is implemented using Verilog on an Altera DE2 FPGA board.

Figure 3-1 shows the top-level design. It consists of mainly four modules—controller, receiver, transmitter, and data sources. The on-board 50 MHz oscillator source is divided down to 5 MHz to clock the design. Users can interface with the design through custom inputs (i.e. sensor data) and controls (enable, etc). The receiver receives the demodulated input (PIE coded) from the analog frontend and the transmitter outputs the FM0/Miller-encoded data as modulation output to the MOSFET transistor that controls the backscattering. The controller coordinates the effort to receive and send data, sourcing appropriate data as response to specific command.

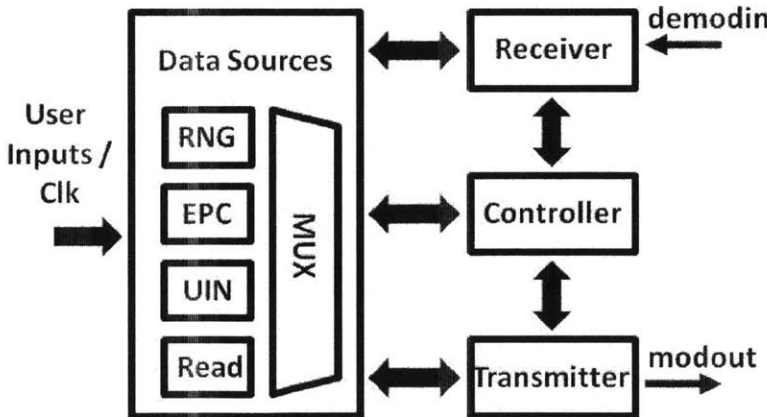


Figure 3-1: Top level design

3.1 Controller

The controller module is the brain of the digital logic that implements the Gen 2 protocol. It decides what response to make upon receiving a complete data packet from the reader. Specifically, the controller manages two tag states—the “receive” and “transmit” states. If the tag has finished sending data, the controller switches from the transmit to the receive state. In the receive state, if the data packet is not completely received, it waits until it is. When this happens, the controller manages the response to the nine commands that are potentially received from the reader. They are the nine commands in the Table 3-1—*QueryRep*, *ACK*, *Query*, *QueryAdjust*, *Select*, *NAK*, *Req_RN*, *Read*, and *Write*.

For *QueryRep*, *Query*, and *QueryAdjust*, the controller implements the slotted Aloha protocol. It draws a RN and loads it to the slot counter. When the slot counter is not zero, it keeps decrementing until it reaches zero at which point the tag transitions to the transmit state. For the *ACK*, the controller checks to see whether the RN16 matches the current handle and moves to transmit when it does. User has the option to choose sending user-specified data or the tag’s EPC. *Select* and *Write* are optional for the purpose of this work and they are not implemented—they just resets the receiver. For *NAK*, no reply is required and it simply resets the receiver. For *Req_RN*, the controller stores the current handle and also sends it to the reader as response. For *Read*, the controller sets up the data source to be read.

Table 3-1: Commands table

Command	Code	Length (bits)	Mandatory?	Protection
<i>QueryRep</i>	00	4	Yes	Unique command length
<i>ACK</i>	01	18	Yes	Unique command length
<i>Query</i>	1000	22	Yes	Unique command length and a CRC-5
<i>QueryAdjust</i>	1001	9	Yes	Unique command length
<i>Select</i>	1010	> 44	Yes	CRC-16
<i>Reserved for future use</i>	1011	–	–	–
<i>NAK</i>	11000000	8	Yes	Unique command length
<i>Req_RN</i>	11000001	40	Yes	CRC-16
<i>Read</i>	11000010	> 57	Yes	CRC-16
<i>Write</i>	11000011	> 58	Yes	CRC-16

3.2 Receiver

The receiver module consists of two functional stages. The first stage receives the entire packet (i.e. preamble and *query*) and converts the PIE symbols to bits on a bit clock. The second stage extracts the command and parameter information from the received bits.

The first stage is made up of two state machines—a high low symbol detector and a packet state tracker. The symbol detector is shown in the Figure 3-2. On reset, it enters the “wait demod low” state. As long as the demodulation signal is a high bit, it stays in that state. Once the signal becomes low, it advances to the “wait demod high” state. It doesn’t advance until it gets the next high bit. When this happens, it starts to evaluate the detected bit in the second state machine as shown in the Figure 3-3.

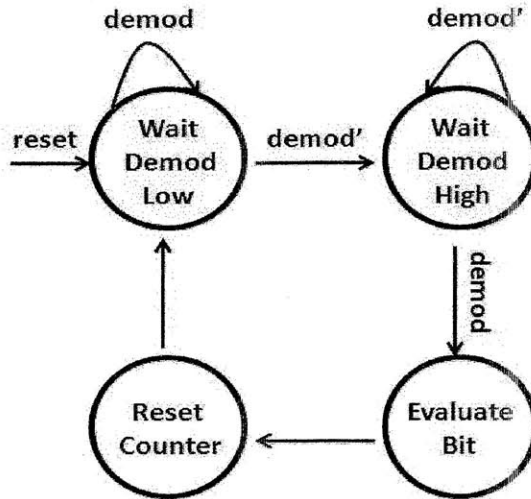


Figure 3-2: High low symbol detector

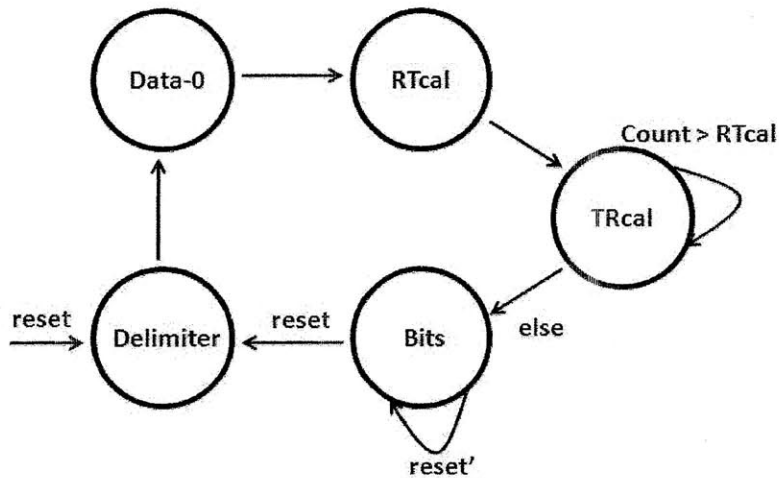


Figure 3-3: Packet state tracker

The packet state tracker tracks the state of the packet which includes a preamble preceding a *Query* or a frame-sync followed by all other commands. Refer to the Figure 2-8, the preamble consists of a delimiter, data-0, RTcal, and TRcal symbols while frame-sync is the same except not having the TRcal. The tracker tracks these states as it receives each bit. To differentiate preamble and frame-sync, the tracker uses a 10-bit

counter to measure the length of RTcal and TRcal. When the bit received after RTcal is longer than RTcal, it is a TRcal which will be stored for calculating BLF. Otherwise, it is a frame-sync and the bit received after RTcal is a command bit. In the “Bits” state, it compares the length of each bit with the pivot which is RTcal/2—bit 1 when longer and bit 0 when shorter. As each bit is received, it outputs a high or low pulse to indicate so. In the “Reset Counter”, when the state tracker is in the “Bits” state, bit clock pulse is generated to synchronize the received bits.

The second stage of the receiver extracts the command and parameters information from the received bits and outputting “complete” signals. The parser implements a counter that counts each bit clock period and determines the command based on the Table 3-1. For example, if 4 bits have been received and they are 1000, it is *Query*. After *Query* is received, it outputs a “complete” signal to indicate command received. After the command, the parser extracts parameter information such as DR, M, TRext, and Q. After 22 bits are received, as shown in the Table 3-1, the *Query* packet is complete. For *QueryAdj*, the 3 updn bits are extracted and stored. For *ACK* and *Req_RN*, extracted handle bits are compared against the current tag handle. When match or non-match takes place, flags are raised.

3.3 Transmitter

The transmitter module is responsible for generating the response packet based on the received command. It encompasses several sub-modules, such as the Preamble, Encoder, CRC16, etc, that handle the preamble generation, FM0 or Miller encoding, and the generation of the 16-bit CRC.

The top level module of the transmitter accomplishes managing the packet construction through the state machine shown in the Figure 3-4. Upon reset, it enters the preamble state and waits for the preamble construction to finish. When this is done, it moves on to the data state where it waits for the data done signal. The data are muxed and sourced from the RNG, EPC, UIN, or Read module determined by the received command. Also depending on the received command, the controller decides whether the packet needs a CRC16 appended to the end of the response to ensure validity.

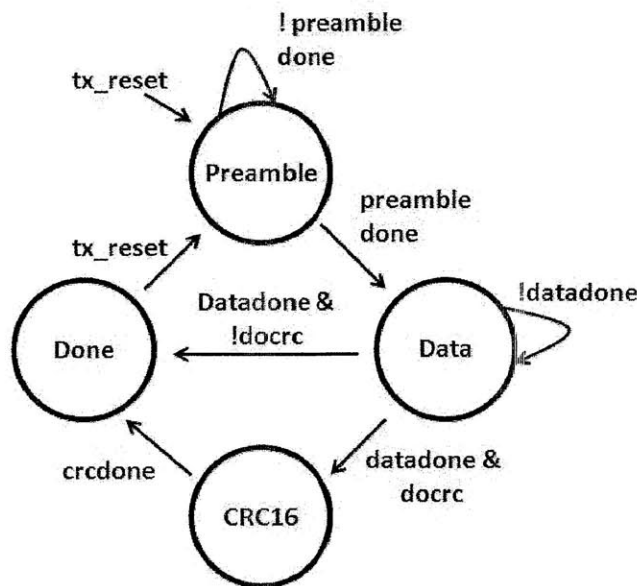


Figure 3-4: Response construction state machine

3.3.1 Preamble

The preamble module generates the preamble that precedes the tag responses, namely the RN16, EPC/UIN, handle, and the read data. Based on the encoding specified in the *Query*, there are different preambles for FM0 and Miller, as shown respectively in the Figure 3-5 and Figure 3-6.

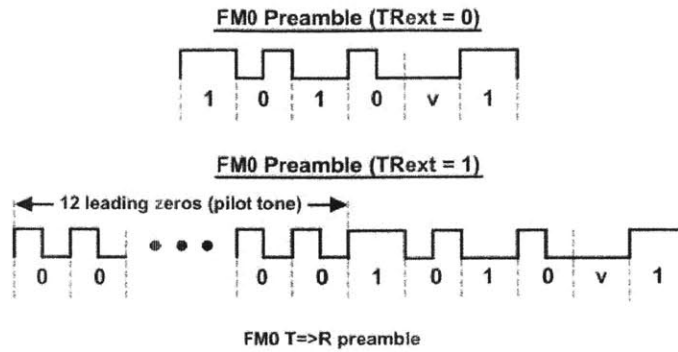


Figure 3-5: FM0 preambles

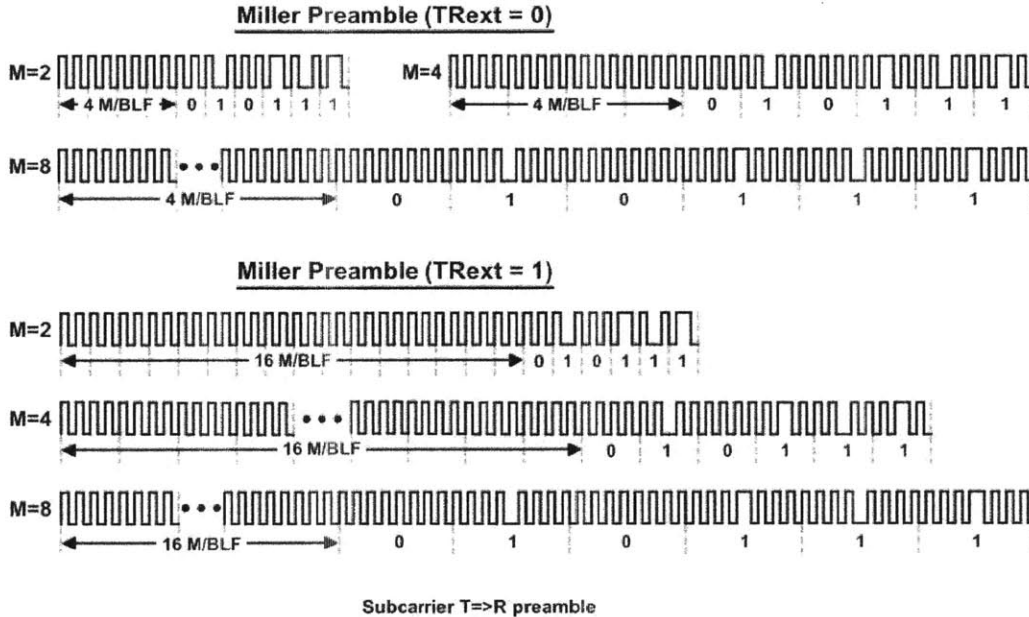


Figure 3-6: Miller preambles

In the case of FM0, the preamble is 1010V1. The violation is a signaling symbol to differentiate preamble from regular data. If the TRext bit is 1, it precedes with 12 leading zeros. In the Miller case, the preamble is 010111 with 4 leading zeros if TRext is 0. If TRext bit is 1, it precedes with 16 leading zeros. The module uses a bit counter to track counts for the leading zeros.

3.3.2 Encoder

The Encoder module encodes the response in either the FM0 or Miller format as discussed in the sections 2.4.3.3 and 2.4.3.4. This is accomplished through phase inversion at the appropriate timing. As explained, the FM0 differs from Miller in that its symbols are opposite from Miller and phase inverts at every boundary whereas in the Miller case phase only inverts when there are two consecutive zeros. The output bit is clocked at twice the BLF in order to flip phase at the mid symbol. It's generated through XOR logic of the phase invert and the clock that runs at BLF. After all the bits are encoded, a “datadone” signal is sent to the controller which moves the state to done. At that point, the transmitter stops and resets. The output signal “modout” of the encoder connects to the gate of the MOSFET in the modulator that controls the data backscattering.

3.3.3 CRC16

The CRC16 module calculates a 16-bit CRC based on the polynomial $x^{16}+x^{12}+x^5+1$. This is implemented as shown in the Figure 3-7. The 16 registers are preloaded with FFFF_h. The incoming data is the bits that need to be transmitted (EPC, UIN, etc). The LSB is the XOR of the last bit and the data bit. The fifth bit is the XOR of data bit, last bit, and the fourth bit. Same for the twelfth bit. After clock in all the data bits, Q[15:0] holds 1s-complement of the CRC16 and to get CRC16 all bits are inverted.

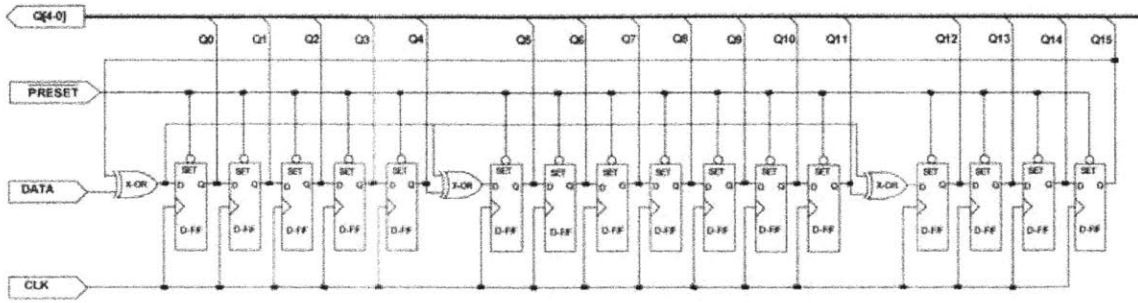


Figure 3-7: CRC-16 generation

3.4 Data Sources

Several data sources are muxed connecting to the transmitter one at a time depending on the received command and needed response. These modules are the pseudo RNG, EPC, UIN, and Read.

The pseudo RNG module leverages the generation methodology of the CRC16 and generates RNs based on the same algorithm as shown in the Figure 3-7. It takes the LSB of the receiver counter for the incoming demodulator bit as the input and passes it through a linear feedback shift register (LFSR) to generate a 16-bit handle. The RNG source is fed to the transmitter in response to *Query*, *QueryAdj*, *QueryRep*, and *Req_RN*, as shown in the Figure 2-2.

The EPC module assigns a 96-bit EPC preceded by a 16-bit PC word 3000_h (see Figure 2-5) and clocks out one bit at a time MSB first. It's fed to the transmitter for transmission in response to an *ACK* when the user chooses to disable the user inputs.

Similar to the EPC module, but instead of sending the EPC, the UIN module sends twelve user-specified, 8-bit data to the reader. Like an EPC, it's preceded by a PC word and clocks out one bit at a time. It's connected in response to *ACK* when the user chooses to input real-time sensor data.

The Read module allows interface with an external source like a microcontroller (μC) or a serial ADC. The response consists of a zero header bit, a 16-bit data, followed by a 16-bit handle.

3.5 Simulation Results

Reader communications (inventory and access) with a single tag, as shown in the Figure 2-2, are simulated and verified in the ModelSim.

3.5.1 Query and RN16

Simulation of steps 1 and 2, in which the reader issues *Query/QueryRep* commands and tag replies with a RN16, is shown in the Figure 3-8. The tag receives the demodulated baseband of the reader signal through “demodin”. At around 230us, “rx_cmd” indicates *Query* (coded 4) is received. After the entire *Query* packet is received at around 410us, the slot counter is loaded with 3. This random value is drawn from between 0 and 2^Q-1 (Q specified in the *Query* is 2) and determined by the Q LSBs of the RN. Since the value in the slot counter is not zero, the reader issues several *QueryRep* (coded 1) to decrement until the slot counter reaches zero. When this happens at around 1030us, the tag starts to reply with the RN16 encoded in FM0. In between commands, reader sends continuous wave to ensure power delivery.

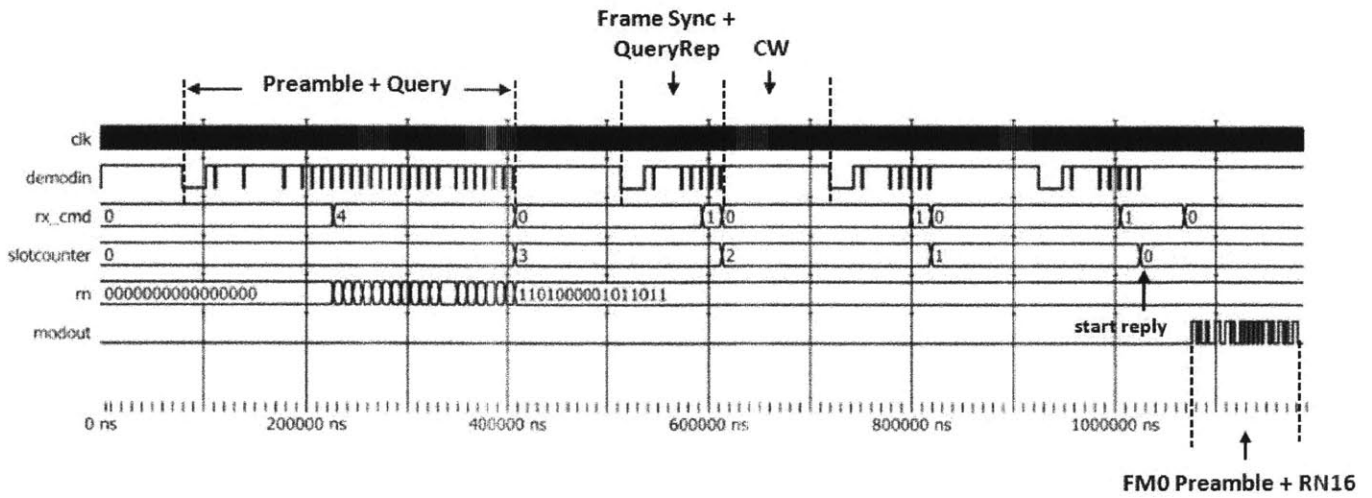


Figure 3-8: Tag receives *Query* and replies with RN16

The zoomed-in view of the *Query* command is shown in the Figure 3-9. The command is preceded by a preamble that defines the session. The preamble consists of a delimiter, data-0 (or Tari), RTcal, and TRcal (see Figure 2-8). *Query* consists of the command, dr, m, TRext, sel, session, target, Q, and CRC5 (see Table 2-2). It is shown that the received Q is 2. “rx_cmd” is loaded with 4 which encode *Query* after the command code is received.

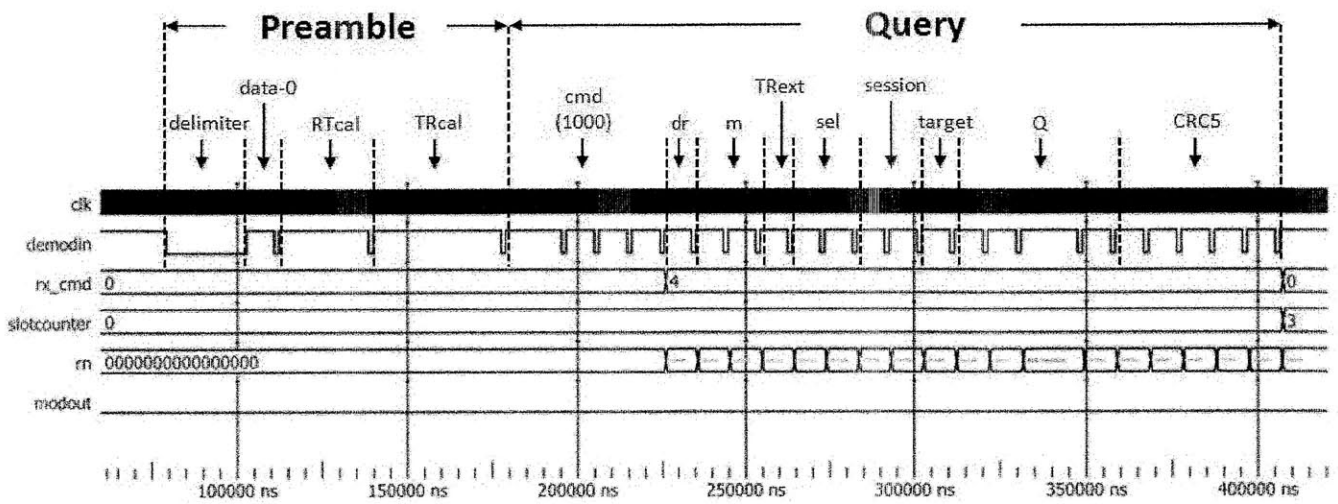


Figure 3-9: Zoomed-in view of the *Query* command

The zoomed-in view of the *QueryRep* is shown in the Figure 3-10. It's preceded by the frame-sync. The *QueryRep* is fairly simple. It includes a 2-bit command and session.

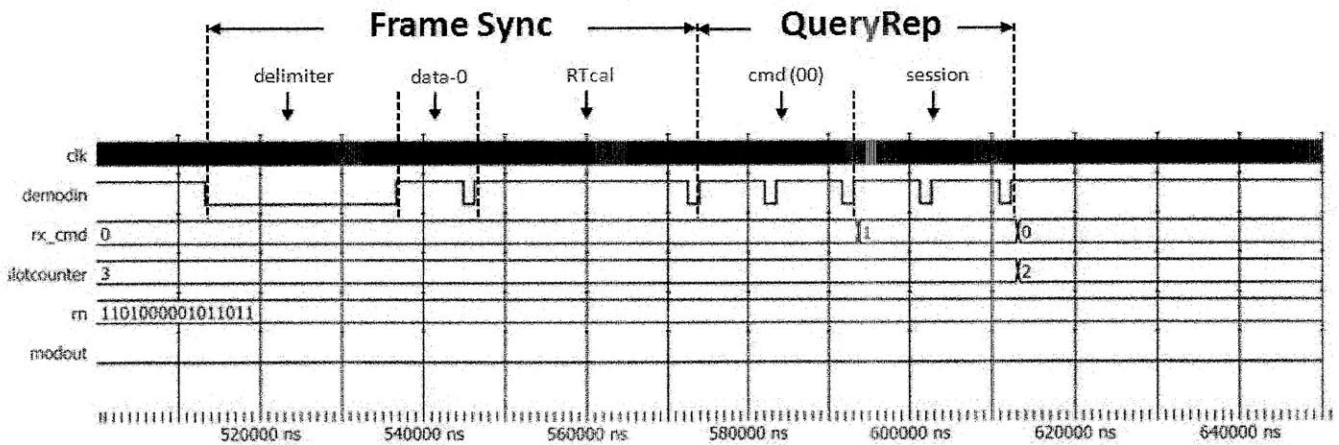


Figure 3-10: Zoomed-in view of the *QueryRep* command

Figure 3-11 shows the zoomed-in view of the tag's reply. The RN16 is encoded in FM0 as specified by the *m* in *Query*. It's preceded by the FM0 preamble which is 1010V1. The transmitted RN16 is seen to match that of the tag.

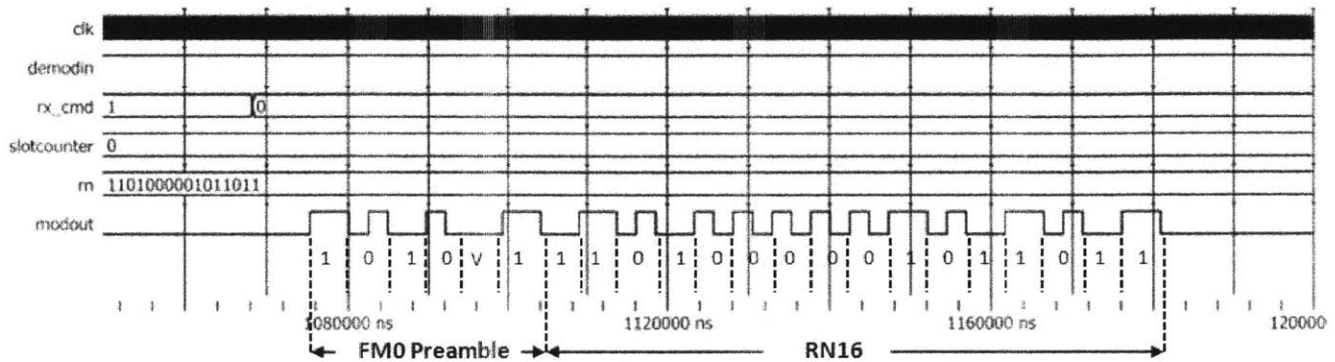


Figure 3-11: Zoomed-in view of the tag's reply (RN16)

3.5.2 ACK and EPC

Simulation of steps 3 and 4 in which the reader issues an *ACK* command and tag replies with PC/EPC is shown in the Figure 3-12. The PIE symbols from the reader are received at the “demodin”. The tag decoded the received command (*ACK* is encoded as 2) at around 860us. It checks the received RN16 against that of the tag. After verifying they match, the tag starts to reply with PC and EPC at around 1.03ms. The CRC16 is calculated based on the transmitted bits and is attached to the end of the reply.

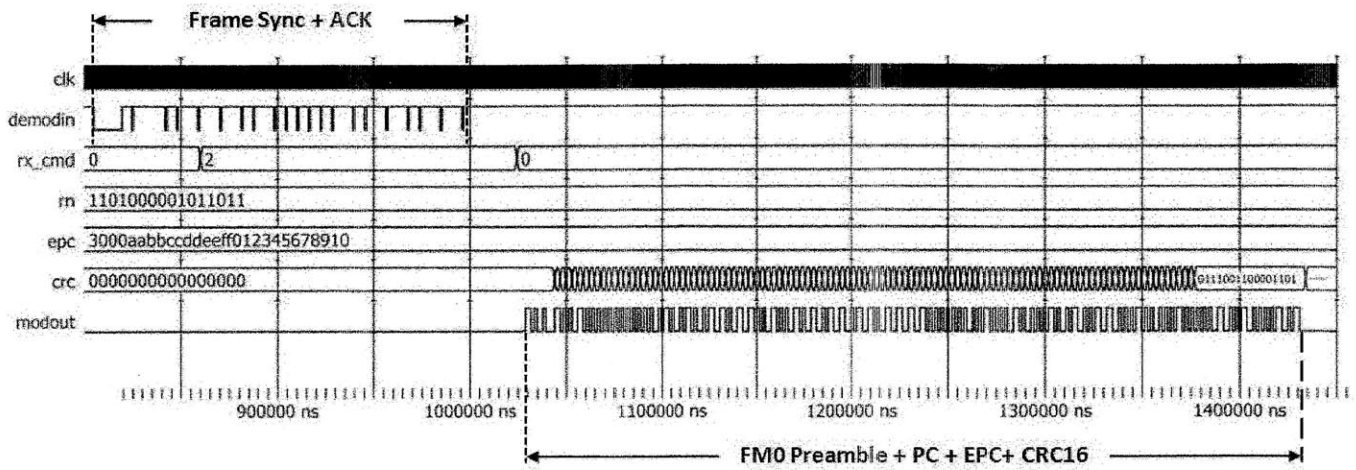


Figure 3-12: Tag receives an *ACK* and replies with PC/EPC

A zoomed-in view of the *ACK* command is shown in the Figure 3-13. The command is preceded by the frame sync. It includes a 2-bit command and the RN16 that the tag sends previously as reply to the *Query*. It is shown that the received RN16 matches with the tag's handle.

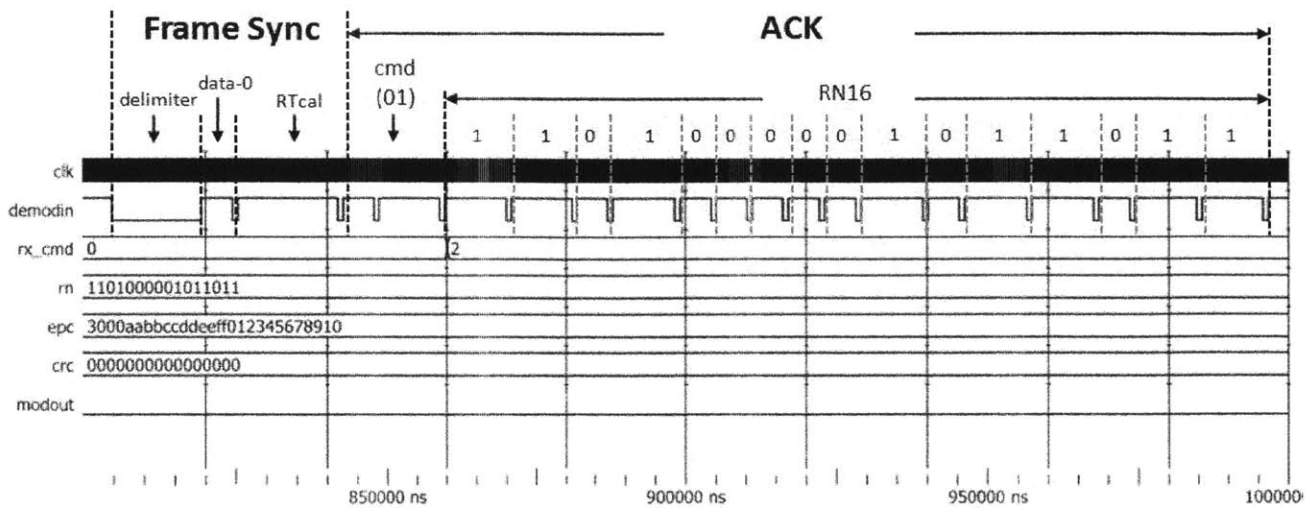


Figure 3-13: Zoomed-in view of the ACK command

Figure 3-14 shows the zoomed-in view of the tag's reply, which includes a FM0 preamble, PC, EPC, and CRC16. It is shown that the transmitted EPC is correctly encoded in FM0 and matches with that of the tag. The CRC16 is calculated and ready before the last EPC bit and is attached to the reply.

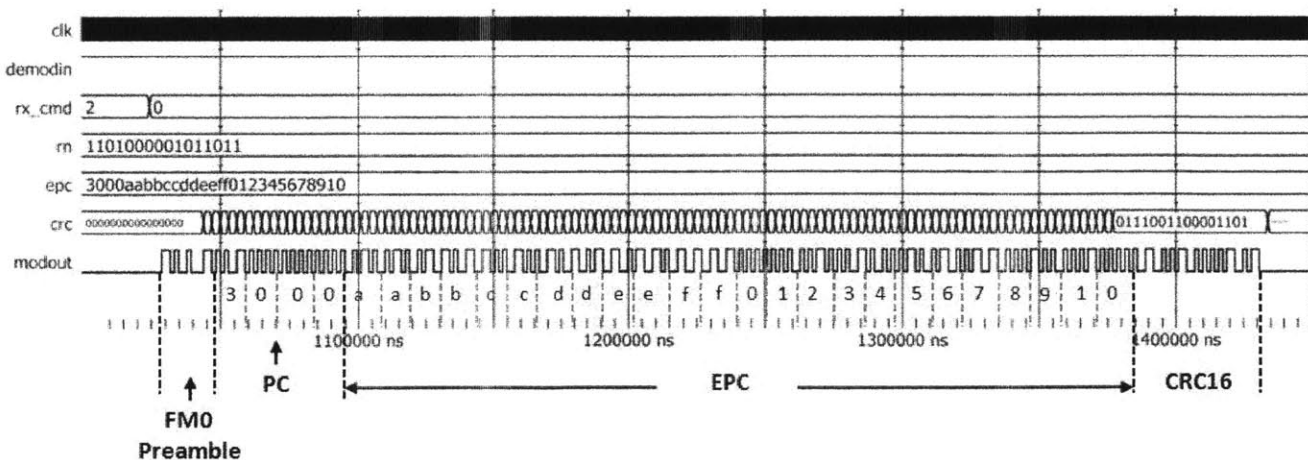


Figure 3-14: Zoomed-in view of the tag's reply (PC/EPC)

3.5.3 Req_RN and Handle

Simulation of steps 5 and 6 in which the reader issues a *Req_RN* command and tag replies with a new handle is shown in the Figure 3-15. The PIE symbols from the reader are received at “demodin”. At around 1.6ms, the tag decoded the received command (*Req_RN* is encoded as 64). At around 1.73ms, the tag starts to generate a new RN16 and at around 1.83ms the new handle (f05c) is ready. At around 1.86ms, the tag starts to transmit the new handle. The CRC16 is calculated at around 1.93ms and is attached to the end of the reply.

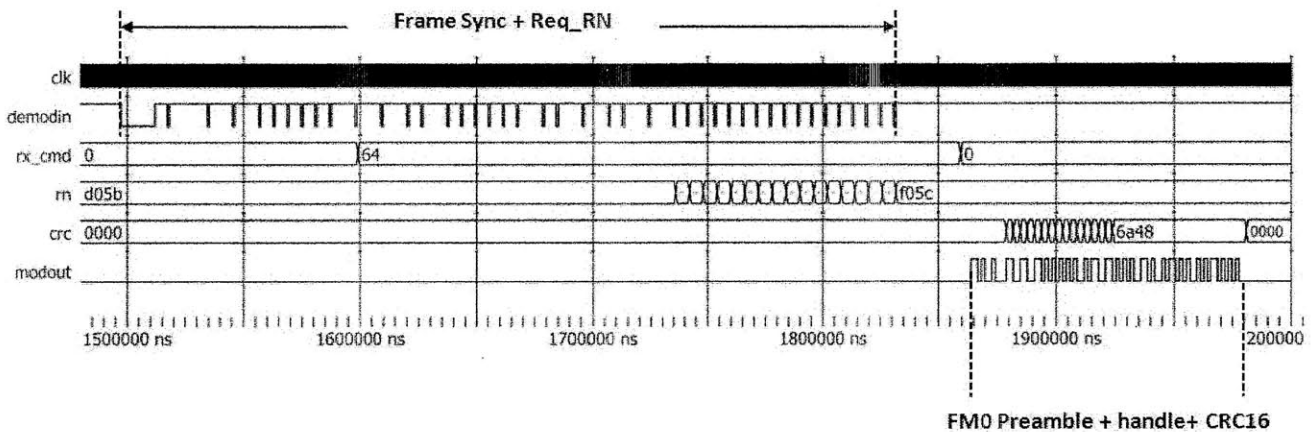


Figure 3-15: Tag receives a *Req_RN* and replies with a new handle

A zoomed-in view of the *Req_RN* command is shown in the Figure 3-16. The command is preceded by frame sync. It contains an 8-bit command, RN16 that the tag sends previously, and CRC16. It is shown that the received RN16 matches with that of the tag. The CRC16 by default is zeros and matches with the tag's CRC16.

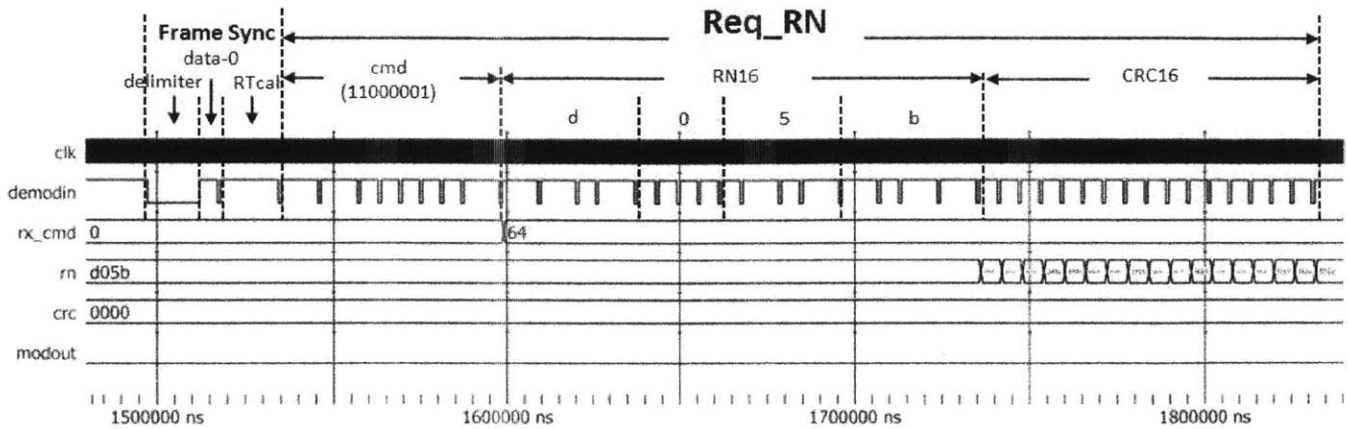


Figure 3-16: Zoomed-in view of the *Req_RN* command

Figure 3-17 shows a zoomed-in view of the tag's reply. The reply contains the FM0 preamble, the new handle which reader requested, and the CRC16. It is shown that the transmitted new handle and CRC16 match with those of the tag's.

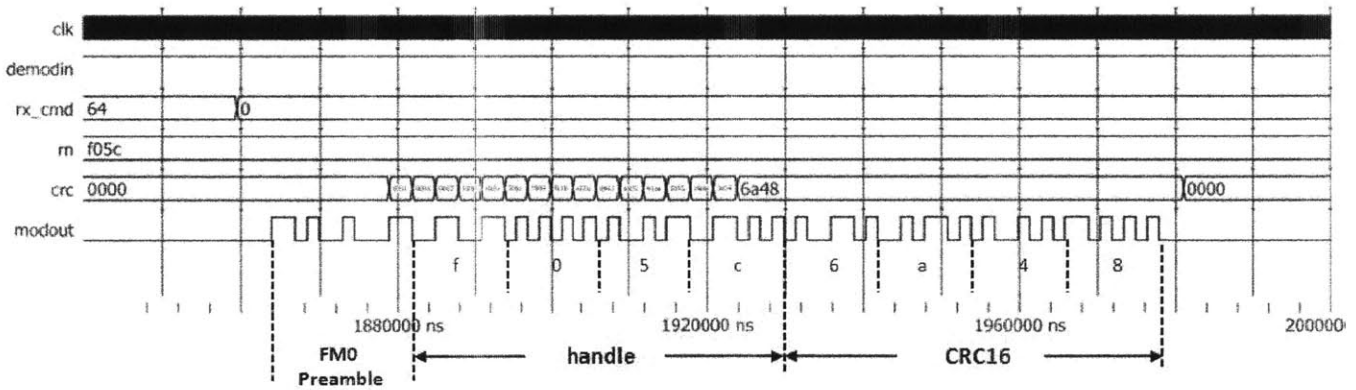


Figure 3-17: Zoomed-in view of the tag's reply (a new handle)

3.5.4 Read and Data

Simulation of steps 7 and 8 in which the reader issues a *Read* command and the tag replies with the data is shown in the Figure 3-18. The PIE symbols from the reader are seen at “demodin”. At around 2.14ms, the tag decoded the received command (*Read* is encoded as 128). The tag’s RN16 is seen to be f05c. At around 2.54ms, the tag starts to transmit the encoded data. The CRC16 calculation is completed at around 2.64ms and is attached to the end of the reply.

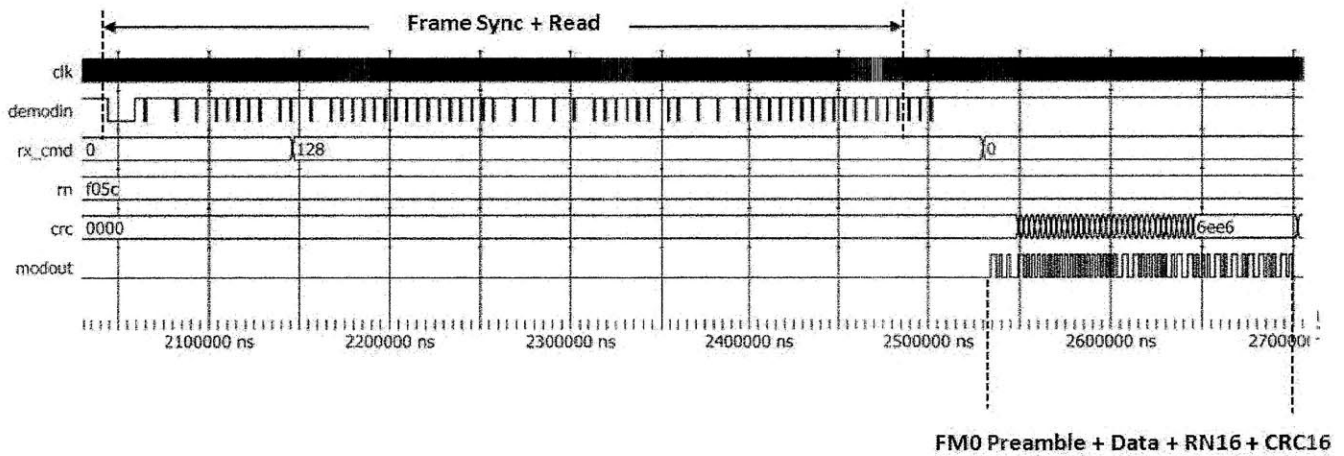


Figure 3-18: Tag receives a *Read* and replies with the data

A zoomed-in view of the *Read* command is shown in the Figure 3-19. The command is preceded by frame sync. It includes an 8-bit command code, the memory bank pointer, starting address pointer, word count, RN16, and CRC16. It can be seen that the received handle matches with the tag’s RN16.

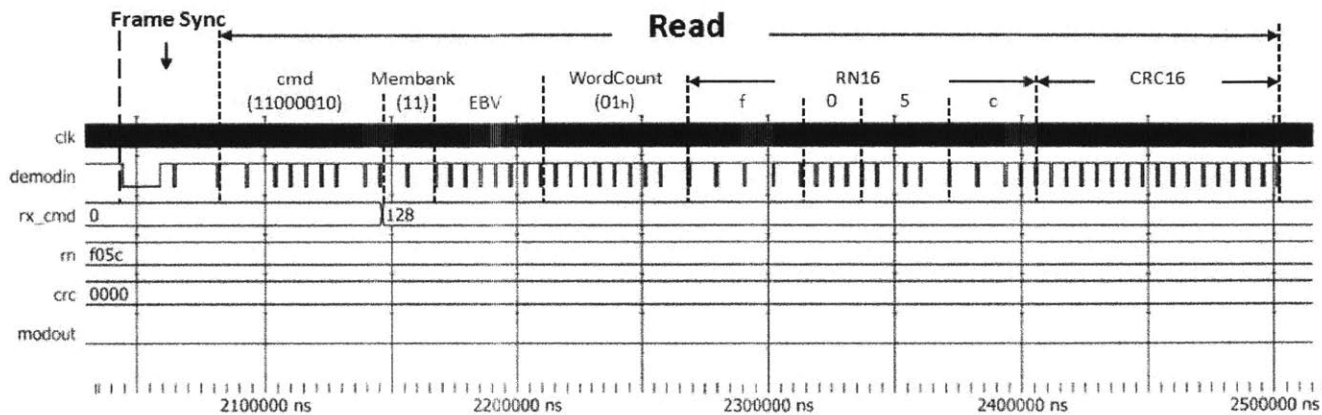


Figure 3-19: Zoomed-in view of the *Read* command

Figure 3-20 shows a zoomed-in view of the tag's reply. The reply contains the FM0 preamble, a zero header bit, a 16-bit data word (zeros by default), RN16, and CRC16. It can be seen that the transmitted RN16 and CRC16 match with those of the tag's.

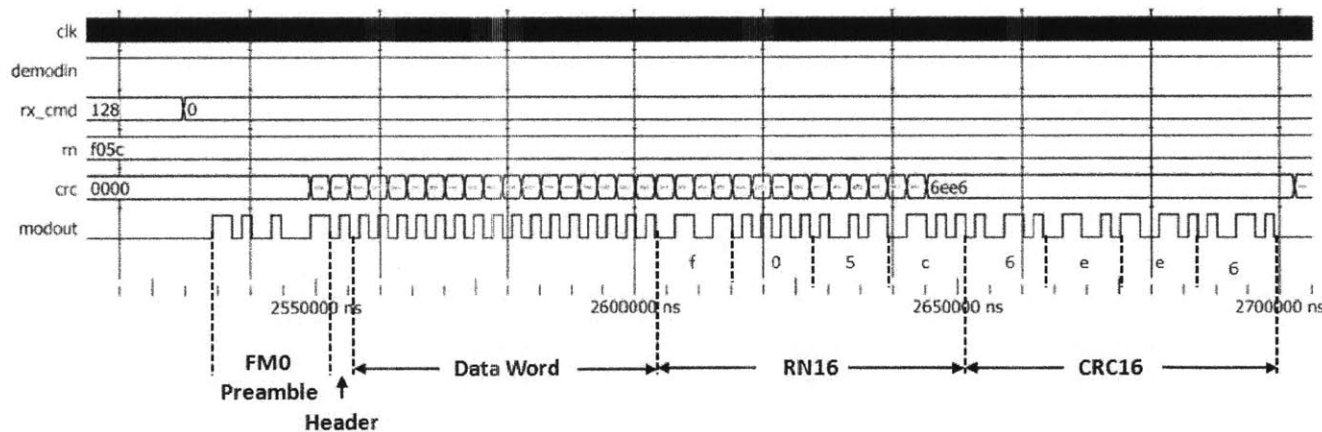


Figure 3-20: Zoomed-in view of the tag's reply (data)

Chapter 4 Analog Frontend and Sensor Circuitry

4.1 Analog Frontend

The analog frontend circuitry of the tag receives the radiated EM wave from the reader's antenna and converts it to a digital signal for the FPGA. This section discusses the reader, reader's antenna, and tag's analog frontend, which includes a dipole antenna, L-section matching network, voltage multiplier, demodulator, and modulator.

4.1.1 Reader

This work used the commercial Impinj Speedway Revolution reader (R420). Per FCC regulation, the maximum power the reader is allowed to radiate is 36 dBm EIRP (4 W). By default, the Speedway reader transmits 30 dBm (1 W) but can be set to radiate 32.5 dBm max (1.78 W) by the Multireader software. The specifications claim it can achieve max receive sensitivity of -82 dBm (6.3 pW) but in reality the software only allows the max sensitivity to be -70 dBm (100 pW). The reader has a worst 10 dbm return loss.

4.1.2 Reader's Antenna

The reader uses the Larid Technologies' circular polarized panel antenna. Directional panel antenna is preferred over omni-directional antenna because it improves read range and gives user control of the read zone.

As shown in the Figure 4-1, the antenna is constructed with two circular plates separated by about 1.5cm. The smaller plate has a diameter of 16cm while the larger one has about 19cm. They sit on a 26cm square ground plate. The splitter structure ensures 90° phase delay to produce the circularly polarized wave.

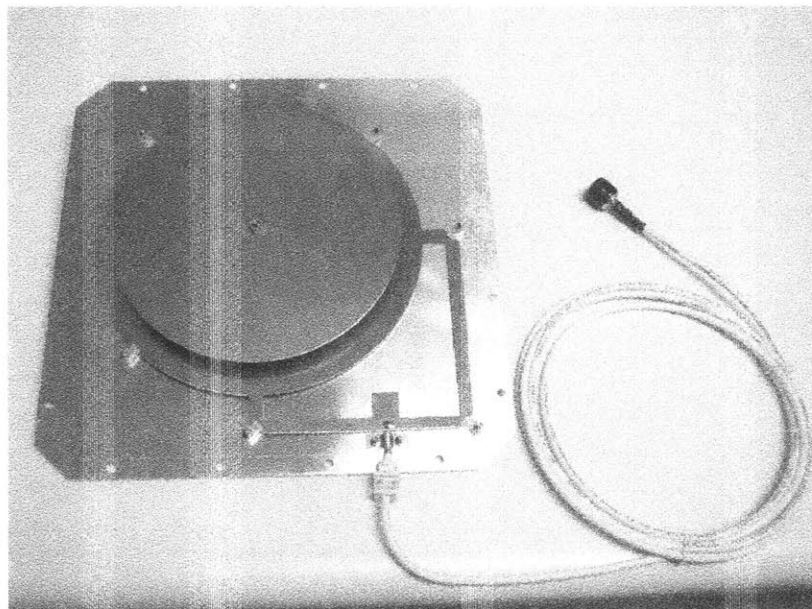


Figure 4-1: Laird Technologies circular polarized panel antenna

The specifications of the panel antenna are shown in the Table 4-1.

Table 4-1: Panel antenna specifications

SPECIFICATIONS	
Antenna Part Number	S9028PLC / S9028PCR
Frequency Range	902 - 928 MHz
Gain	9 dBic
Maxium VSWR	1.3:1
3 dB Beamwidth - Azimuth	70°
Front to Back Ratio	20 dB
Polarization	Circular Right or Left
Maxium Input Power	10 Watts
Input Impedence	50 Ohms
Axial Ratio	1dB
Weight (Kg)	1.75 lbs (1.13)
Mechanical Size	10.2" x 10.2" x 1.32"
Antenna Connection	Coax Pigtail, Rev TNC Male (others available)
Radome	High Strength PC
Mount Style	Threaded Stud
Temperature Operational	-25°C to +70°C
Lightning Protection	DC Grounded
Environmental Rating	IP 54

4.1.2.1 Bandwidth and Impedance

The frequency range of 902-928 MHz refers to the bandwidth over which the antenna is matched to a 50 Ω coaxial cable so that most of the power sent to the antenna is radiated and very little is reflected (this is evident by the max VSWR of 1.3:1, or -20dBm). The bandwidth (BW) can be used to estimate the effective quality factor Q_{eff} as:

$$Q_{\text{eff}} = \frac{f_0}{\text{BW}} \quad (4.1)$$

where f_0 is the resonant frequency at 915MHz. Q_0 is calculated to be 35.2.

The finite value is due to the loss in the antenna inductance and the radiation resistance that models the transmitted power. Since the typical radiation resistance for a panel antenna is about 150Ω , capacitance can be estimated to be about 41pF using:

$$C_{\text{ant}} = \frac{1}{2\pi R_{\text{ant}} BW} \quad (4.2)$$

Therefore, the antenna inductance can be calculated to be around 0.74nH using:

$$w_0 = \frac{1}{\sqrt{L_{\text{ant}} C_{\text{ant}}}} \quad (4.3)$$

The panel antenna can be modeled as shown in the Figure 4-2. The input impedance needs to be properly matched to provide 50Ω .

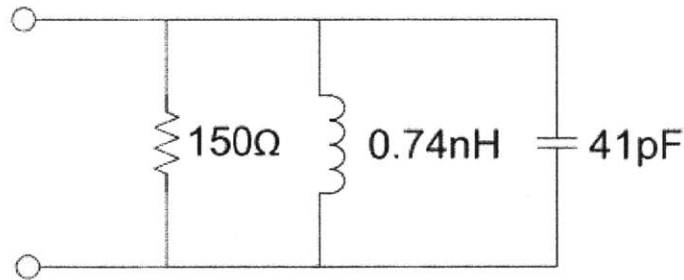


Figure 4-2: Panel antenna circuit model

4.1.2.2 Gain, Directivity, Efficiency

The gain of the antenna is closely related to the concepts of directivity and efficiency. Directivity describes how narrowly focused an antenna radiates in a certain direction as compared to an isotropic radiator which emits equally in all directions. Efficiency is the ratio of power radiated over the total supplied power to the antenna. Gain is the product of these two parameters. It is the factor by which the power density S

is greater than that of an isotropic antenna at the same transmission power. Mathematically, gain is expressed as:

$$G = \xi \cdot D \quad (4.4)$$

where $\xi = \frac{P_{\text{rad}}}{P_t}$ and $D = \frac{S_{\text{max}}}{S_{\text{av}}}$, $S_{\text{av}} = \frac{P_{\text{rad}}}{4\pi R^2}$.

The panel antenna has a circular gain of 9dBic with respect to an isotropic antenna. Since the circular polarization delivers power half of the time, it has a 3dB loss. Therefore, in terms of dBi, the gain is 6dBi.

4.1.2.3 Radiation Pattern

The radiation pattern of the panel antenna is shown in the Figure 4-3. The 3dB beam width is shown to be around 70°. The back leakage radiation is minimal with a front-to-back ratio of 20dB.

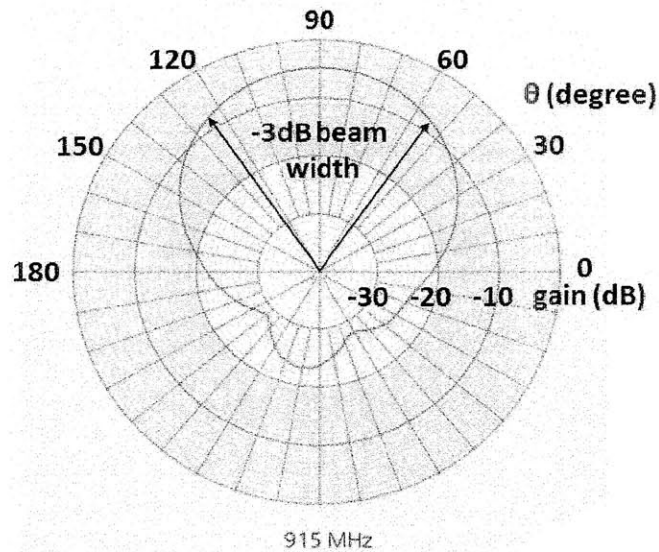






Figure 4-3: Panel antenna radiation pattern

4.1.2.4 Polarization State

Polarization refers to the shape drawn by the tip of the electric field in the plane orthogonal to the wave propagation. Simply imagine you are looking at the wave that is travelling at you in the z direction, the trace you see the electric field tip draws in the x-y plane is the polarization of the wave.

Two common polarization states—circular and linear are shown in the Table 4-2. For reader antenna, circular is preferred because it ensures half of the power radiated is delivered regardless of the orientation of tag’s antenna. In order to receive power from the linear polarization, the tag’s antenna cannot be orthogonal to the linear axis.

Table 4-2: Polarization States

Polarization States	
Left Circular	
Right Circular	
Vertical Linear	
Horizontal Linear	

4.1.3 Links

The reader tag communication links are asymmetric. For passive tags, it is forward-link limited whereas for semi-passive it is reverse link limited. Passive tag in this

work has turn-on sensitivity of about -10 dBm, which limits the read range. The stationary reader, however, due to complex design and ample power, achieves sensitivity of -70 dBm. The reader radiates 36 dBm. The allowable path loss is about 46 dBm in order for the tag to receive enough power to turn on. Adding another 46 dBm loss from the return path, it is about -56 dBm when the signal arrives at the reader. For semi-passive tags, turn-on power is not required since battery powers the internal circuitry. Therefore, the read range is only limited by the reserve link, which allows 106 dBm losses.

The path loss is characterized by the Friis transmission formula, which is derived in the Appendix A.1. From the equation A.1.1, the Friis formula in dBm is expressed as

$$P_{RX} = P_{TX} + G_{TX} + G_{RX} - L_{FS} \quad (4.5)$$

The derivation doesn't consider non-idealities such as encoding, polarization, transmission line, connector, fading, multipath losses, etc. The complete transmission transfer function should be

$$P_{RX} = P_{TX} + G_{TX} + G_{RX} - L_{FS} - L_{en} - L_p - L_{mis} \quad (4.6)$$

where L_{en} = encoding loss, L_p = polarization loss, and L_{mis} = miscellaneous loss.

In the worst case, the PIE encoding has a loss about 3 dB when PW is at its max (52.5% of Tari) and data are all zeros. The FM0 and Miller also incur about 3 dB losses. Circular polarization has a 3 dB loss. Assuming the miscellaneous loss (line, connector, fading, multipath, etc) is about 1 dB, we arrive at the tag's received power of -10dBm at 3m (30dBm+6dBi+2.2dBi-41.2dB-3dB-3dB-1dB), which is about the read range limit.

4.1.4 Tag Antenna

Since at this prototyping stage size is not constrained, for simplicity the tag uses a half-wave ($\lambda/2$) dipole antenna made of copper lines printed on the FR4 PCB. The length of the dipole is about 16cm and width about 1mm. Its gain is about 2.15 dBi (1.64) and has 73Ω input impedance at 915 MHz resonance. The antenna size is a serious limiting factor of the tag size and in the future it should be reduced by using meandered or fractal designs.

As shown in the Figure 4-4, the reader EM wave induces an open-circuit AC voltage V_{oc} across the tag antenna's terminals. The antenna impedance is modeled as

$$Z_A = R_A + jX_A \quad (4.7)$$

where $R_A = R_{rad} + R_{loss}$ and $X_A = j(\omega L - \frac{1}{\omega C})$. The loss resistance R_{loss} is usually much smaller than the radiation resistance R_{rad} and it can be ignored. The reactance is approximately to zero at resonance when $\omega_0 = \frac{1}{\sqrt{LC}}$.

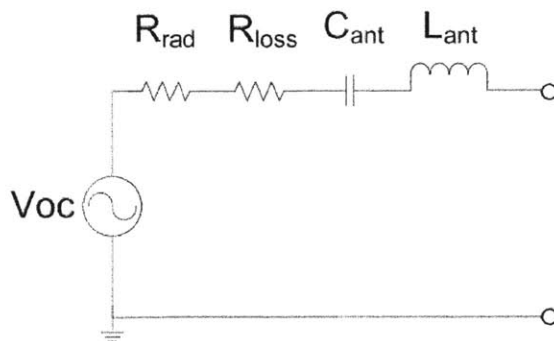


Figure 4-4: Tag antenna model

4.1.5 Matching Network

A simple LC matching network, as shown in the Figure 4-5, is used to transform the tag antenna impedance ($Z_A = 73 \Omega$) to conjugate match ($Z_A = Z_L^*$, or $R_{\text{rad}} = R_L$ and $X_A = -X_L$) the input impedance of the voltage multiplier, which is about $Z_L = 8 - j100 \Omega$. Conjugate matching is necessary to ensure the max power is delivered and the Friis transmission formula, which is based on the assumption of impedance matching, is valid. Instead of using discrete components, the network can be incorporated into the antenna design. However, this approach is more involved. The Berkeley impedance matching network designer calculator can be used to determine the initial LC values but ultimately a Vector Network Analyzer (VNA) has to be used to experimentally determine these values since many parasitic factors cannot be considered in the theoretical calculation.

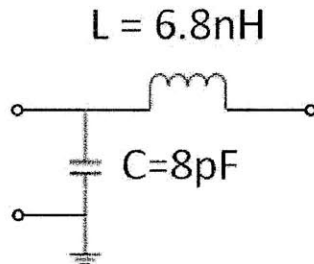


Figure 4-5: LC matching network

Figure 4-6 shows the complete circuit model for the tag analog frontend. After the conjugate matching, the circuit is reduced to a voltage source connected to $2R_{\text{rad}}$ since $R_L = R_{\text{rad}}$. This is shown in the Figure 4-7. The average power delivered to the load is:

$$\begin{aligned}
 P_{\text{rec}} &= \frac{1}{2\pi} \int_0^{2\pi} \frac{\left(\frac{1}{2} V_{\text{OC}} \cos \omega t\right)^2}{R_L} d\omega t \\
 &= \frac{V_{\text{OC}}^2}{8R_{\text{rad}}}
 \end{aligned}
 \tag{4.8}$$

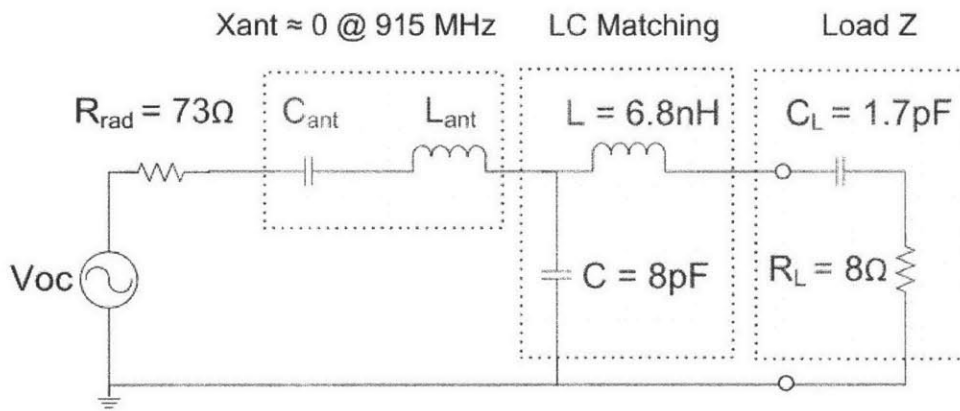


Figure 4-6: Tag analog frontend circuit model

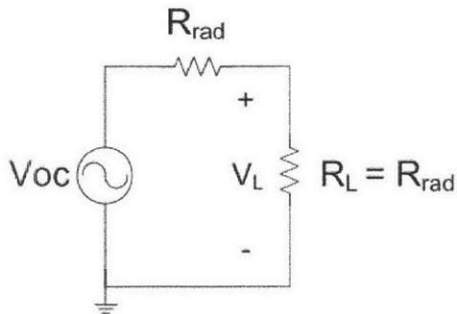


Figure 4-7: Tag frontend equivalent circuit

Rearranging terms, the amplitude of V_{OC} is:

$$V_{\text{OC}} = 2\sqrt{2R_{\text{rad}}P_{\text{rec}}}
 \tag{4.9}$$

Since the Friis formula shows (see A.1.1 in the Appendix)

$$P_{\text{rec}} = P_{\text{EIRP}} G_r \left(\frac{\lambda}{4\pi d}\right)^2 \quad (4.10)$$

Plug 4.10 into 4.9, we get:

$$V_{\text{OC}} = 2\sqrt{2R_{\text{rad}}P_{\text{EIRP}}G_r} \frac{\lambda}{4\pi d} \quad (4.11)$$

The voltage across the load, or the input impedance of the voltage multiplier, is therefore:

$$V_L = \frac{1}{2}V_{\text{OC}} = \sqrt{2R_{\text{rad}}P_{\text{EIRP}}G_r} \frac{\lambda}{4\pi d} \quad (4.12)$$

If we rearrange terms, we have:

$$d = \frac{\lambda}{4\pi V_L} \sqrt{2R_{\text{rad}}P_{\text{EIRP}}G_r} \quad (4.13)$$

These results are valid in the far field. Given a certain operating frequency, equation 4.13 says to increase range, we have to increase R_{rad} , P_{EIRP} , G_r or decrease the required turn-on voltage V_L . Since P_{EIRP} is fixed by regulation to 36 dBm, we either have to design better tag antenna or better voltage multiplier. This result makes sense.

Although equation 4.13 provides good insight into design, it's not accurate when predicting read range because the equation 4.10 doesn't take into account the various losses. For example, given $P_{\text{rec}} = 100 \text{ uW}$ (-10 dBm), $R_{\text{rad}} = 73 \text{ } \Omega$ and $G_r = 1.64$, equations 4.9-4.13 show $V_{\text{OC}} = 242\text{mV}$, $V_L = 121\text{mV}$, and $d = 6.7\text{m}$. In section 4.1.3, we arrive at read range of about 3m when considering the losses. The V_{OC} and V_L estimates are generally correct. We need basically about 121 mV to turn on the tag.

4.1.6 Voltage Multiplier

The voltage multiplier employs five stages of voltage doublers to rectify the input RF AC voltage to about 5V DC. Figure 4-8 shows a single-stage voltage doubler which consists of a diode clamp and a peak detector. The diodes used are zero-bias RF Schottky diodes optimized for RFID applications.

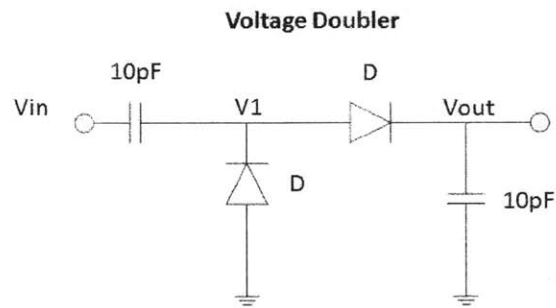


Figure 4-8: Single-stage voltage doubler

Figure 4-9 shows the simulation for the single stage voltage doubler. In the negative half cycle, the diode conducts which charges the capacitor. In the positive half cycle, charges are added on to the previous accumulated charges to reach a higher voltage. $V1$ doubles the input voltage and V_{out} detects the peaks to produce a constant DC voltage.

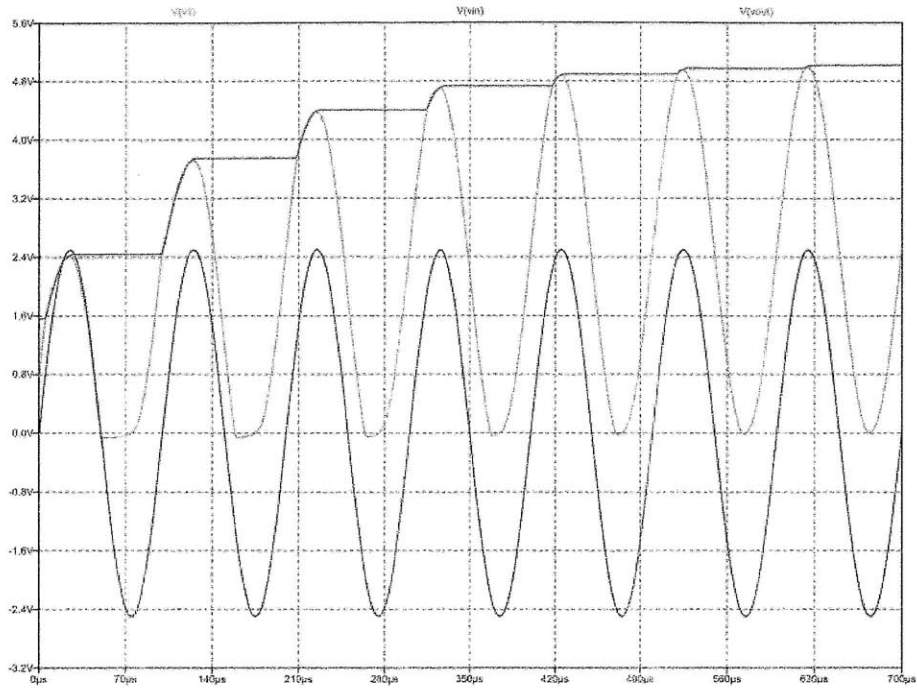


Figure 4-9: Single stage voltage doubler simulation

4.1.7 Demodulator

The demodulator uses a low-power, high speed comparator to compare the baseband of the RF input and its average value. The baseband is produced by an envelope detector and the average reference value is created by another envelope detector that operates on the baseband. Figure 4-10 shows the two inputs to the comparator. V_{in} is the baseband of the RF input and V_{ref} is the average reference value. Whenever V_{in} is greater than V_{ref} , the comparator pulls the output to its positive rail and produces a one. Otherwise, it produces a zero. This result is shown in the Figure 4-11. Note that the comparator requires a 35µs power up time.

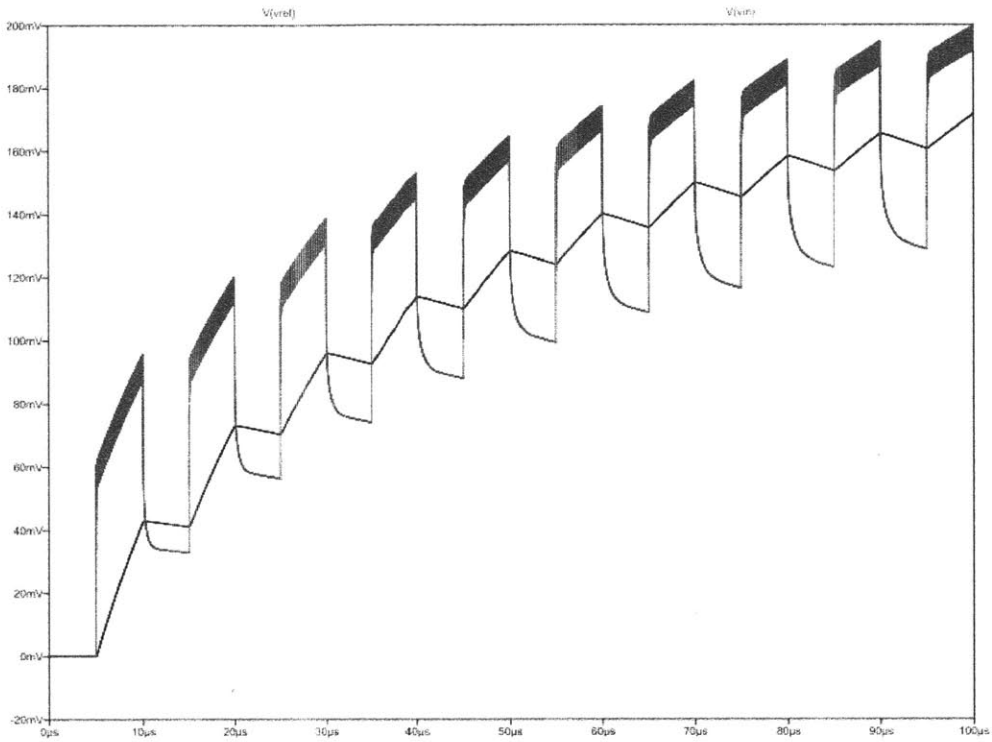


Figure 4-10: Inputs to the comparator

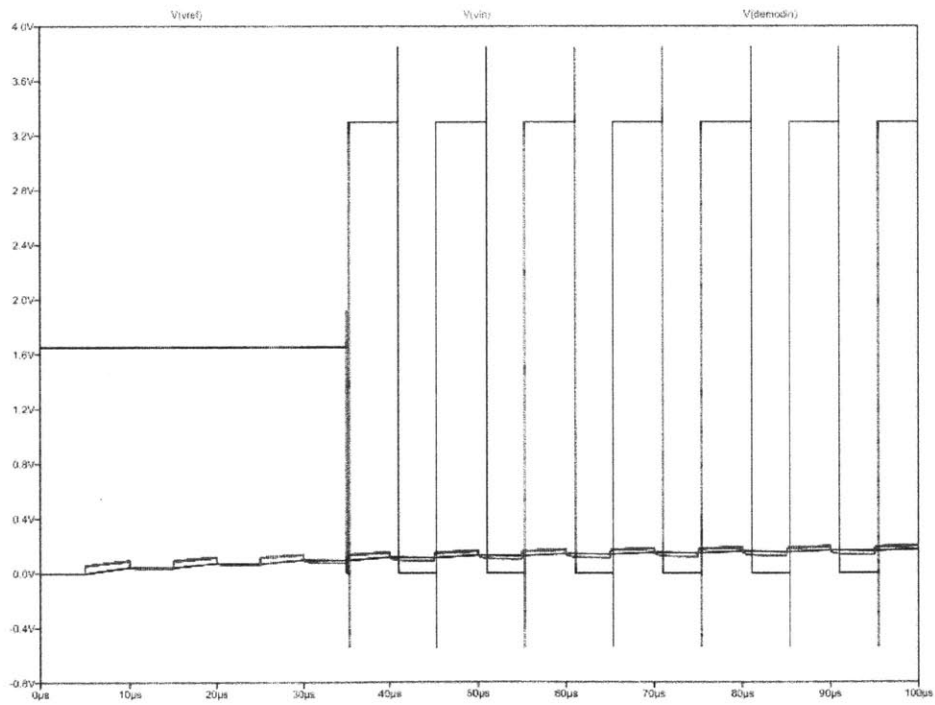


Figure 4-11: Demodulator output

4.1.8 Modulator

The modulator uses a single MOSEFT transistor to modulate the load impedance of the antenna by switching between the matched and shorted states. When the switch is off, the load matches with the antenna radiation resistance and half of the power received at the antenna delivered to the load while the other half gets backscattered to the reader. When the switch is on, no voltage is seen across the load and virtually all the power is backscattered. Switching between these two states create an ASK modulation as seen by the reader. The MOSFET switch can handle at least 640 KHz of switching. Although PSK (modulating capacitive load component) can also be used, it provides little advantage as it may provide better power delivery to the load at the expenses of lower backscattered power and more complex implementation.

4.2 Sensor Circuitry

Sensors are transducers that transform natural phenomena (such as temperature, acceleration, light, pressure, etc) to an electrical voltage. In this work, a discrete temperature sensor (LM60) and potentiometer are used. The low-power temperature sensor can achieve a maximum drain current of 110 uA at 25°C and can measure range of -40°C to 125°C (corresponding to 174 mV to 1205 mV) with a temperature coefficient of 6.25mV/°C. It presents an offset of 424 mV at 0°C with nonlinearity of less than 1°C. Its accuracy is about 2°C across the entire range. The pot represents an inertial sensor and provides the needed control for the collected data.

The output voltage of the sensor is fed into an 8-bit SAR ADC for digitalizing. It's discretely implemented and powered off from the stable FPGA 5V Vdd. The 8-bit parallel outputs are fed through UIN to the FPGA.

Chapter 5 Experimental Results

The goal of the experiment is to demonstrate the capability to transmit 8-bit sensor data over the passive UHF link to the commercial RFID reader, and explore transmission performance.

Figure 5-1 shows the measured ADC output across the input voltage range. A potentiometer was used to vary the voltage in 0.5 V step. The figure shows the ADC output has a slight gain error but overall very linear. Figure 5-2 shows the logic analyzer capture of the ADC output value at 2.5 V, which is the output hex data 7F. This verifies the measured value in the Figure 5-1. Figure 5-3 shows the modulation output signal encoded in FM0 triggered on the transmission enable. It appears that data is transmitted at a rate of around 640 Kbps. Finally, the data is captured by the PC software as seen in the Figure 5-4.

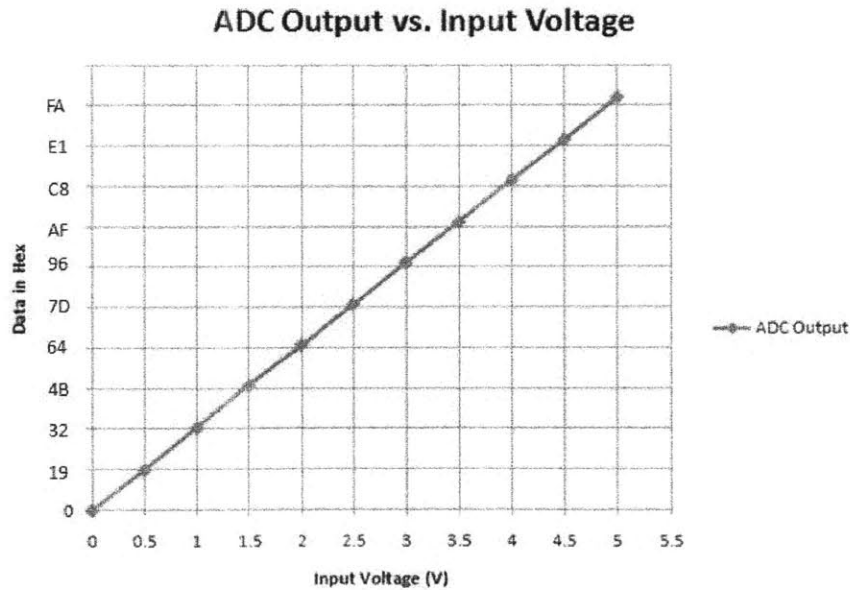


Figure 5-1: Measured ADC output vs. input voltage

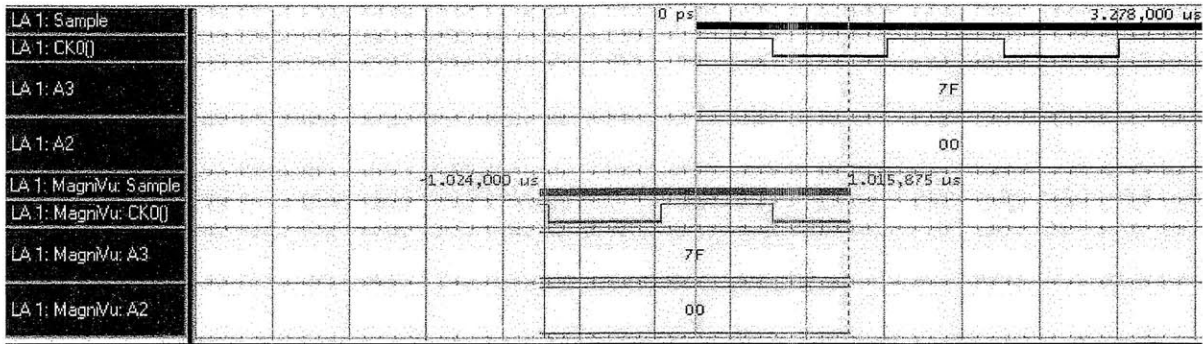


Figure 5-2: Logic analyzer screen capture

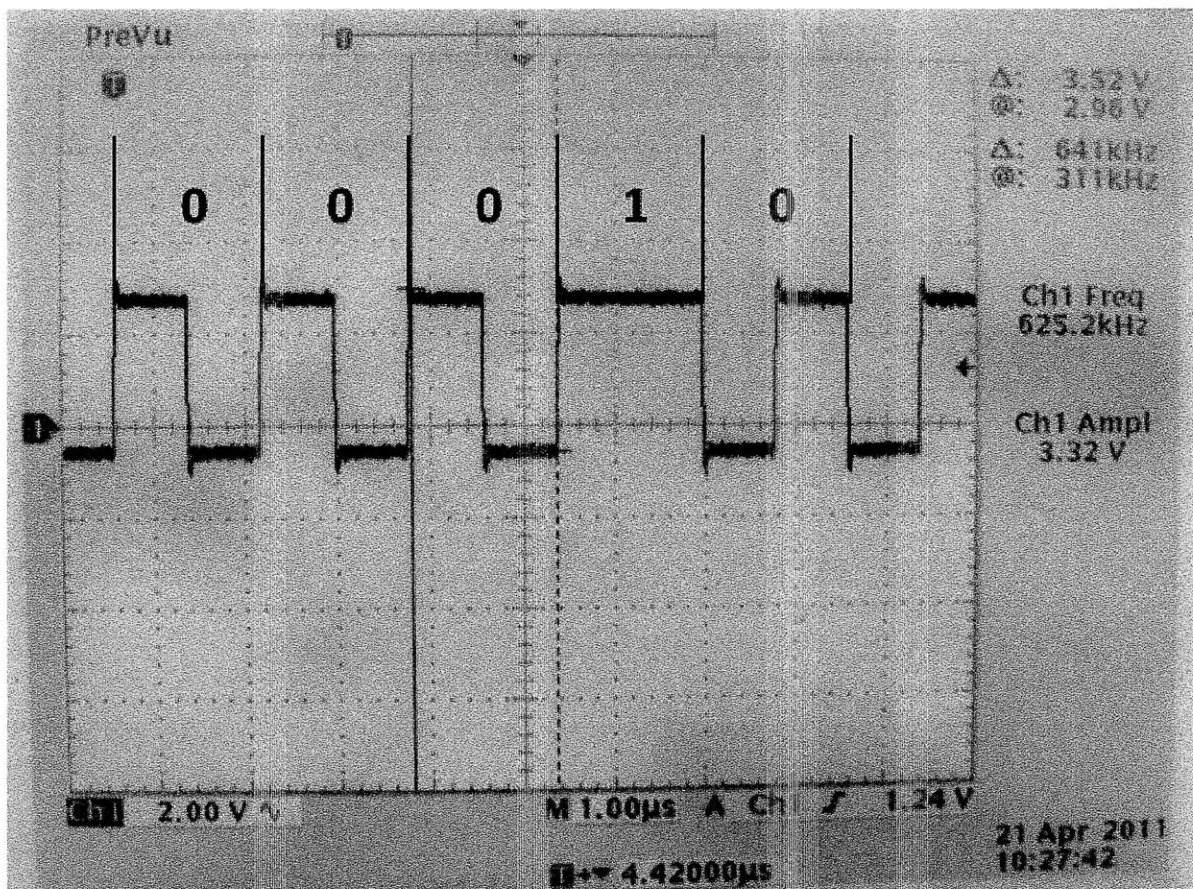


Figure 5-3: Modulation output signal triggered on the TX enable

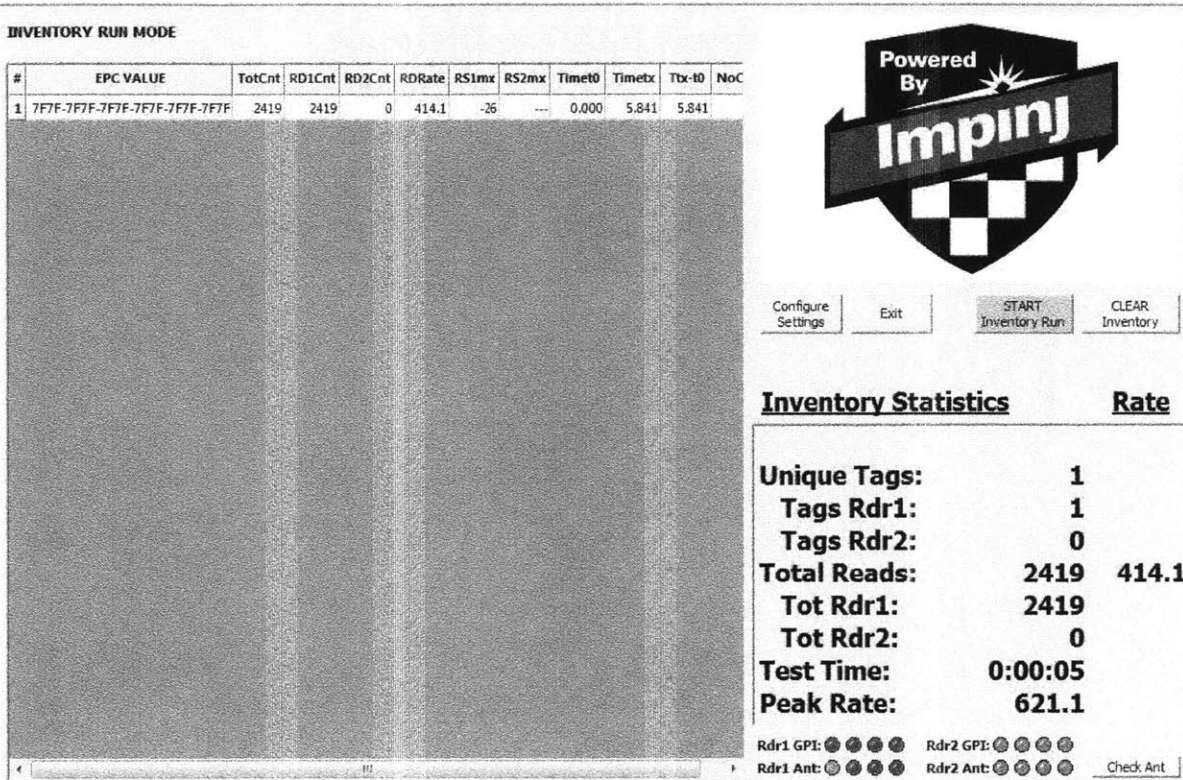


Figure 5-4: Data captured in the PC software

Surprisingly, the measured read distance is incredibly close to the theoretical link analysis of 3 m when the reader outputs 30 dBm. Figure 5-5 shows the measured rectified voltage, read rate versus distance. As the reader antenna farther away from the sensor tag, the rectified voltage drops from about 5.4 V to 1.8 V which is the minimum turn-on voltage. Read rate drops from about 420 reads/s to about 5 reads/s. It is apparent that there's a tradeoff between distance and read rate. At close distance, received voltage is far above the turn-on threshold and the circuit operates faster than the minimum case where only 1.8 V is received. The strength of the backscattering is stronger and more frequent than the case of far distance. This is evidently shown in the Figure 5-5. This read rate vs distance property can be used to approximate the distance of the object when the location of the object is not exactly known.

Voltage, Rate vs. Distance

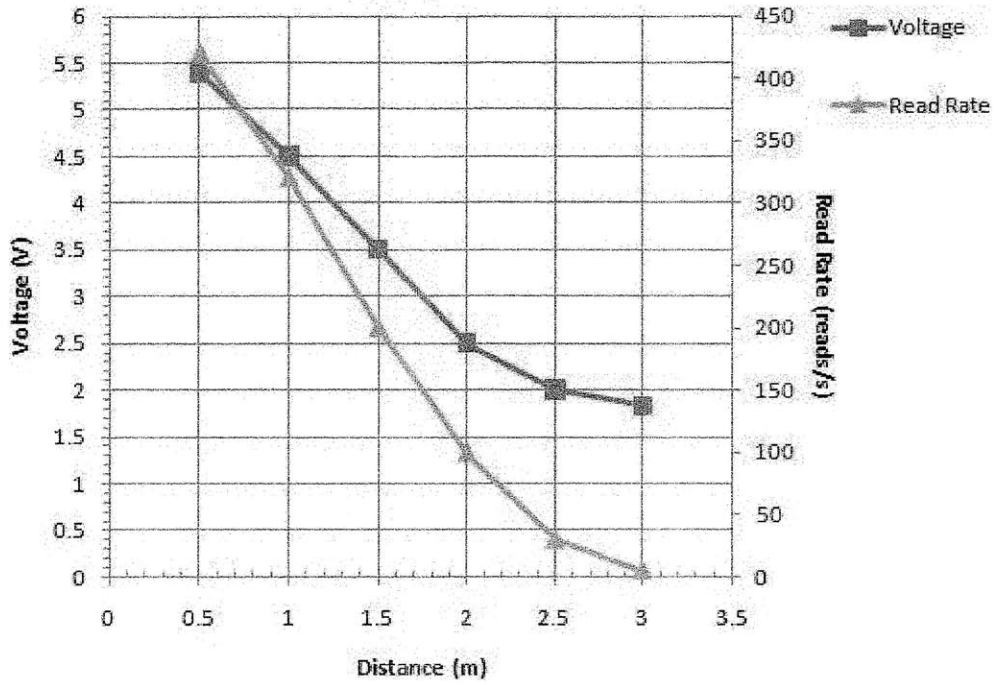


Figure 5-5: Measurement of the rectified voltage, read rate versus distance

Chapter 6 Conclusion and Future Work

This work constitutes a preliminary step towards realizing a long-term monitoring sensor device for patients with Parkinson's disease. It discusses the Gen 2 protocol, its implementation on the FPGA, the analog frontend circuitry, and the sensor circuitry. Experimental results of the prototype show that 8-bit sensor data can be transmitted in the 96-bit EPC format up to the max distance of about 3 m at 5 reads/s. Faster read rate can be achieved when the distance is reduced.

Future developmental work should focus on the following:

- Currently, Gen 2 protocol is implemented on the FPGA board. In the future, it needs to be synthesized and integrated into an ASIC chip.
- The sensor in this work collects data and sends them in real-time to the reader. In the future, memory needs to be integrated in the chip to store sensor data.
- This work uses a dipole antenna which is large in size. In the future, innovative meandered or fractal antenna designs need to be explored for miniaturization.
- Currently, data is read through EPC after the tag is acknowledged. In the future, this can be done through the *Read* command. However, this approach incurs more overhead and is only justified when multiple devices are in the read zone at the same time.
- In the future, ultracapacitors can be integrated to realize a somewhat semi-passive tag. This approach will realize wireless charging and allows sensor to run when the tag is not being powered. The tag power-on will not be limited by sensitivity or the read

distance. This will increase the read range which will only be limited by the reader sensitivity.

- It's also interesting to explore integrating low-power CPLD or FPGA directly on board to realize sophisticated programmability.
- More work needs to be done on the application software which interprets and processes sensor data. This is a practical concern as it needs to produce useful information for the system to be beneficial to the doctors.

Appendix A

A.1 Friis Transmission Formula Derivation

A time-varying AC current ($i(t) = I_0 \cos \omega t$) flowing on the reader's antenna generates an EM wave that radiates into free space. Depending on the type of the antenna used, the antenna radiation pattern differs. For theoretical analysis, it is assumed that the antenna is isotropic and lossless. Under the isotropic condition, EM waves are radiated equally in all directions and the total radiated power is uniformly distributed. We can calculate the power density S incident upon the receiving antenna at a distance d . This is simply the total radiated power P_{rad} over the surface of a sphere with radius d or

$$S_{\text{iso}} = \frac{P_{\text{rad}}}{4\pi d^2} \text{ (W/m}^2\text{)} \quad (\text{A.1.1})$$

Under the lossless condition, the radiation efficiency defined as:

$$\xi = \frac{P_{\text{rad}}}{P_t} \quad (\text{A.1.2})$$

is equal to 1, or $P_{\text{rad}} = P_t$. The power supplied to the transmitting antenna is completely radiated into space and no power is lost in heat. Therefore, equation A.1.1 becomes

$$S_{\text{iso}} = \frac{P_{\text{rad}}}{4\pi d^2} = \frac{\xi_{\text{iso}} P_t}{4\pi d^2} = \frac{P_t}{4\pi d^2} \text{ (W/m}^2\text{)} \quad (\text{A.1.3})$$

In reality, an isotropic lossless antenna is only hypothetical and doesn't exist. A real antenna has a gain G , defined as

$$G = \frac{S}{S_{\text{iso}}} \text{ (dBi)} \quad (\text{A.1.4})$$

It indicates the factor by which the power density S is greater than that of an isotropic antenna at the same radiated or transmission power. For example, an isotropic antenna has $G_{\text{iso}} = 0 \text{ dBi}$ or 1 , and a $\lambda/2$ dipole has $G_t = 2.15 \text{ dBi}$ or 1.64 ($G_t = 10 \log 1.64 \approx 2.15 \text{ dBi}$, the subscript t stands for transmitter). The gain can be expressed as

$$G = \xi \cdot D \text{ (dBi)} \quad (\text{A.1.5})$$

the product of the radiation efficiency and the directivity of the antenna. Therefore, the gain accounts for the ohmic losses in the antenna material and when the antenna is lossless ($\xi = 1$), $G=D$.

In practice, the intercepted power density incident upon the receiving antenna is

$$S_{\text{int}} = G_t S_{\text{iso}} = \frac{G_t P_t}{4\pi d^2} = \frac{\xi_t D_t P_t}{4\pi d^2} \text{ (W/m}^2\text{)} \quad (\text{A.1.6})$$

The term $G_t P_t$ is called $P_{\text{EIRP}}(\text{W})$, the effective isotropic radiated power. It is being expressed in dBm as

$$P_{\text{EIRP}}(\text{dBm}) = P_t(\text{dBm}) + G_t(\text{dBi}) \quad (\text{A.1.7})$$

With the FCC restricting the P_{EIRP} to 36 dBm (4 W), it becomes a tradeoff between antenna gain and transmitting power.

The intercepted power incident upon the receiving antenna is

$$P_{\text{int}} = S_{\text{int}} \cdot A_r \quad (\text{A.1.8})$$

where A_r is the tag's effective aperture. It can be shown that

$$A_r = \frac{\lambda^2}{4\pi} D_r \quad (\text{A.1.9})$$

for any matched antenna. Therefore,

$$P_{\text{int}} = S_{\text{int}} \cdot A_r = P_t G_t D_r \left(\frac{\lambda}{4\pi d}\right)^2 \quad (\text{A.1.10})$$

Assuming that the receiving antenna has an efficiency of ξ_r , the received power is therefore

$$\begin{aligned}
P_{\text{rec}} &= \xi_r P_{\text{int}} \\
&= \xi_r P_t G_t D_r \left(\frac{\lambda}{4\pi d}\right)^2 \\
&= P_t G_t G_r \left(\frac{\lambda}{4\pi d}\right)^2 \\
&= P_{\text{EIRP}} G_r \left(\frac{\lambda}{4\pi d}\right)^2
\end{aligned} \tag{A.1.11}$$

This is the Friis transmission formula, or the power transfer function which relates the received power at the receiving antenna to the supply power of the transmitting antenna. Conceptually, the

$$\text{Received Power (dBm)} = \text{Transmitted Power (dBm)} + \text{Gains (dBi)} - \text{Losses (dB)}$$

Expressed in the logarithmic dBm, the Friis formula is

$$P_{\text{RX}} = P_{\text{TX}} + G_{\text{TX}} + G_{\text{RX}} - L_{\text{FS}} \tag{A.1.12}$$

where

P_{RX} = received power (dBm)

P_{TX} = transmitted power (dBm)

G_{TX} = reader antenna gain (dBi)

G_{RX} = tag antenna gain (dBi)

L_{FS} = free space path loss (dB). Refer to A.1.11, $L_{\text{FS}} = 20 \log \frac{4\pi d}{\lambda}$.

If we rearrange the terms, we can obtain the forward link read range as

$$d = \frac{\lambda}{4\pi} \sqrt{\frac{P_{\text{EIRP}} G_r}{P_{\text{rec}}}} \tag{A.1.13}$$

A.2 Testing Equipment List

Equipments	Manufacturer/Model
Digital Oscilloscope	Tektronix TDS 3032B
Logic Analyzer	Tektronix TLA 5202
Function Generator	Tektronix AFG 3102 (1GS/s, 100MHz)
Digital Multi-meter	Agilent 34401A
Power Supply	Agilent E3630A
Signal Generator	Agilent E4430B (ESG-D series, 250KHz -1GHz)
UHF RFID Reader	Impinj Speedway R420
UHF RFID Circular Polarized Panel Antenna	Laird Technologies (S9028PCL/PCR)
FPGA Board	Altera DE2

Bibliography

- [1] T. Akinwande and C. Sodini , “MTL in Medical”, a presentation to the MTL faculties, 2005.
- [2] “The History of RFID Technology”, RFID Journal, 2005
- [3] Wikipedia. Internet: http://en.wikipedia.org/wiki/Thing_%28listening_device%29, Mar. 03, 2011 [Mar. 07, 2011].
- [4] Stockman, H., "Communication by Means of Reflected Power," Proceedings of the IRE, vol.36, no.10, pp. 1196- 1204, Oct. 1948
- [5] Landt, J., "The history of RFID," Potentials, IEEE, vol.24, no.4, pp. 8- 11, Oct.-Nov. 2005
- [6] D. M. Dobkin, The RF in RFID: Passive UHF RFID in practice, UK: Elsevier Inc., 2007.
- [7] Chawla, V., Dong Sam Ha, "An overview of passive RFID," Communications Magazine, IEEE, vol.45, no.9, pp.11-17, September 2007
- [8] Vernon, F., Jr., "Application of the microwave homodyne," Antennas and Propagation, Transactions of the IRE Professional Group, vol.4, no.1, pp. 110- 116, Dec 1952
- [9] Harris, D.B, “Radio transmission systems with modulated passive responder,” U.S. Patent 2,927,321, Mar. 01, 1960
- [10] Richardson, R.M, “Remotely activated radio frequency powered devices,” U.S. Patent 3,098,971, Jul. 23, 1963
- [11] Harrington, Roger F., "Theory of loaded scatterers," Electrical Engineers, Proceedings of the Institution of, vol.111, no.4, pp.617-623, April 1964
- [12] Vinding, J.P, “Interrogator-responder identification system”, U.S. Patent 3,299,424, Jan. 17, 1967
- [13] Vogelman, J.H, “Passive data transmission technique utilizing radar echoes,” U.S. Patent 3,391,404, Jul. 02, 1968
- [14] Cardullo, M. W; Parks, W.L, “Transponder apparatus and system,” U.S. Patent 3,713,148, Jan. 23, 1973

- [15] Walton, C.A, "Electronic identification & recognition systems," U.S. Patent 3,752,960, Aug. 14, 1973
- [16] Arnold, W.F., "The toll highway faces automation," *Electronics*, vol. 46, p. 74, Nov. 8, 1973
- [17] Koelle, A.R.; Depp, S.W.; Freyman, R.W., "Short-range radio-telemetry for electronic identification, using modulated RF backscatter," *Proceedings of the IEEE*, vol.63, no.8, pp. 1260- 1261, Aug. 1975
- [18] Klensch, R.J., "Electronic identification system," U.S. Patent 3,914762, Oct. 21, 1975
- [19] Sterzer, F., "Electronic license plate for motor vehicles," U.S. Patent 4,001,822, Jan. 04, 1977
- [20] Walton, C.A, "Portable radio frequency emitting identifier," U.S. Patent 4,384,288, May. 17, 1983
- [21] K. Finkelzeller, *The RFID Handbook*, 2nd ed., John Wiley & Sons, 2003.
- [22] S. Preradovic, N. Karmakar and I. Balbin, "RFID Transponders," *Microwave Magazine, IEEE*, vol. 9, pp. 90-103, 2008.
- [23] OSHA Cincinnati Laboratory, Field service memo: electromagnetic radiation and how it affects your instruments, US Department of Labor, 1990.
- [24] EPCglobal. Class 1 Generation 2 UHF Air Interface Protocol Standard, 2009.
- [25] EPCglobal. EPC Tag Data Standard (TDS), 2009.
- [26] D. Yeager, Fan Zhang, A. Zarrasvand, N. T. George, T. Daniel and B. P. Otis, "A 9uA, Addressable Gen2 Sensor Tag for Biosignal Acquisition," *Solid-State Circuits, IEEE Journal of*, vol. 45, pp. 2198-2209, 2010.
- [27] A. P. Sample, D. J. Yeager, P. S. Powledge, A. V. Mamishev and J. R. Smith, "Design of an RFID-Based Battery-Free Programmable Sensing Platform," *Instrumentation and Measurement, IEEE Transactions on*, vol. 57, pp. 2608-2615, 2008.