APPLICATION AND SOFTWARE CAPABILITY OF RFID

An Inter-Qualifying Project Report submitted to the faculty of

WORCSTER POLYTECHNIC INSTITUTE

in partial fulfillment of the requirements for the

Degree of Bachelor of Science

by

Nixon Mathew

nixon@wpi.edu ________________

Ehsun Siddiqui

esiddiq@wpi.edu ________________

Date: May 21, 2007

Approved:

Prof. Sergey Makarov, Electrical and Computer Engineering, WPI Major Advisor

Dr. Javier Berrios, Vice President Engineering, Tego Inc., Co-Advisor

Acknowledgements

We would like to thank Professor Sergey Makarov for his never ending support and constant guidance. His advice was detrimental to the success of this project. Also, we would like to extend our thanks to Doctor Javier Berrios and Tego Inc. for their provision of the Mercury 4 reader and antenna. The equipment was necessary for the completion of this project. We would also like to specially thank Professor Susan Jarvis and David Holl, a graduate student at WPI for their extensive help with the coding.

Table of Contents

Table of Figure

Table of Tables

1. Abstract

The Radio Frequency Identification (RFID) technology is making a tremendous impact on the consumer market. RFIDs can be easily integrated into almost all the current technologies to produce a more convenient and efficient product. RFIDs can replace many of the conventional approach of tagging data with a much reliable system. The versatile use of RFIDs, their discrete size and their ability to operate with or without an internal power supply attracts manufactures, and engineers to incorporate this technology into their product.

2. Executive Summary

The main focus our Inter Qualifying Project is to research and analyze the impact of RFID in the current technology, biomedical field, retail market and the manufacturing industry. This paper reviews the history of the RFIDs and addresses the effect of it on the consumer market. The biomedical, manufacturing, and retail are some of the major industries which embrace the RFID technology and implement them for different purposes. Furthermore, this paper explores the use of RFID for individual purposes and the effect of RFID implantation. The principle behind the RFID technology is also discussed along with the methods to compensate for any disadvantage of RFIDs. In order to understand the economics of the RFID technology, a feasibility study along with any short comings of the technology is researched.

For a more hands on experience, a small RFID system has been set up in one of the MQP labs at Atwater Kent building. The system is assigned an IP address; thus it is a part of the WPI network and a connection can be established through Ethernet. The system also includes a web based user interfacial applet called Yagi MercuryOS 2.4 by ThingMangic Inc. to perform various functions with the tags. This applet is used to conduct a performance testing of the RFID system.

In addition to the performance testing, an external user interfacial C applet is created to communicate between the tags and the tag reader. The external applet performs similar operations as the Yagi Mercury OS 2.4 applet. The Yagi Mercury OS 2.4 system and the external applet are discussed in further detail in section 11 and 12 respectively.

3. History of RFID

Heinrich Rudolf Hertz unveils the path to RFID technology in 1887 when he successfully transmitted and received radio waves. Since Hertz's achievement, the generation and transmission of radio signals evolved tremendously. The genesis of the RFID technology soon embarked upon us when Harry Stockman published his paper on "Communication by Means of Reflected Power" in 1948. Unfortunately for Stockman, his vision did not become a reality for another thirty years. This was because many of the technologies such as transistors, microcontrollers and integrated circuits were not developed at that time [5].

In the 1960's the companies developed electronic article surveillance (EAS) equipment to prevent theft by using one bit tags. These equipments sense the presence of the tags and if any of the tags are absent from the specified location then the system will alert authorities. However, to have an efficient counter theft system, the tags must be cheap and discrete. Such specifications were an issue with the one bit tags. For more complex antitheft system multi bit tags were needed, but they were not viable because of its loaf size dimensions of the multi bit tags [5].

The RFID technology bloomed in the 1970s when extensive research was conducted by academic universities and science laboratories. Los Alamos Scientific Laboratory, North-Western University, and the Microwave Institute Foundation in Sweden were some of the foundations that pursued RFID technology. Many different companies also developed commercial products using RFID. The electronic identification system developed by Richard Klensh of RCA in 1975 started to be widely used in tracking applications such as animal tracking, vehicle tracking, factory automation etc [5].

The RFIDs were completely revolutionized when the low power Complementary Metal Oxide Semiconductor (CMOS) logic circuits were integrated into the RFID tags in the late 1970s and early 1980s. The tag technology had improved in efficiency and the tag size was significantly reduced. The tag memory was also improved when the fusible link diode was implemented to RFID tags. Nonvolatile EEPROM memory became the conventional memory for the tags since it permitted large scale manufacturing of identical tags. The tags could be easily segregated for different applications through programming. As a result further reduction of size and higher performance was obtained [5].

In 1990 E-Z pass was introduced to the market of Northeastern America. The E-Z pass implemented RFID tags to develop an electronic toll collecting system. The system utilized a single tag that was encoded with a single billing account per vehicle to access specific major high ways and bridges. The RFID tags excelled in tolling applications that it was employed in different applications such as parking lot access, gated community access and closed campus access. The growing RFID tag applications catalyzed the further advancement of the tags. The new tags that were developed enabled the customer to have either single or dual accounts. The RFID tags continued to become more discreet as the microwave Schottky diodes were fabricated to CMOS integrated circuit. Since the Schottky diodes share the same die as the tag circuitry, it increased functionality, reliability and efficiency. The fabrication of internal circuitry of the tags in one chip reduced the overall tag price [5].

The RFID technology continues to grow in the twenty first century. The tags can be now placed in sticky labels and can be easily attached to virtually any object where it can reflect the transmitted signal properly. The RFIDs are starting to become more mainstream and the future for RFIDs seems to be very promising. Different standards for the RFID applications are being regulated such as the Federal Communications Commission standards. Different privacy policies are being set and other delegation of legal aspects are in progress. The RFID technology research continues as engineers search for a more efficient and powerful antenna designs and better nonvolatile memory. The RFID technology continues to grow and it now spans numerous fields such as system engineering, software development, circuit theory, antenna theory, radio propagation, microwave techniques etc. As a result the consumer market is being incubated with RFID technology. Some of the effect of RFID technology on the consumer market is discussed in the next section [5].

4. RFID Technology

The RFID technology consists of mainly three components: base station, transponder with a strip antenna, and memory and processing unit. The operation frequency of the RFID varies from a few kHz to 5 GHz. However, the frequency regulated for the tags are 13.56 MHz, 400 MHz, 915 MHz, 2.45 GHz and 5.3 GHz. In the US, FCC15-247 approval is required with an ERIP of 36 dBm. FCC 15-247 is a Federal Communication Commission regulation, which confines the operational frequency of RFID. In the North America 902 MHz to 928 MHz is allocated for RFID systems [8].

4.1 RFID Tags

The section below introduces different tags technology, tag specifications, tag anatomy, and tag operation.

4.1.1 Different Tag Technology

There are three distinguished transponder tags currently available on the consumer market. They are passive tags, semi-passive tags and active tags. The passive tags contain no internal power supply. The passive tags use the power in the modulated signal transmitted from the base station to operate the tag circuitry and to transmit information. Therefore the ranges of these tags are very small, approximately five to six feet, but the lifespan of the passive tags ideally are unlimited. The only exception to this would be the failure of individual parts in the internal circuitry. The price of the tags is much cheaper than semi-active tags and active tags.

As discussed above the passive tags obtain the power in the transmitted signal to operate the internal circuitry and to respond to the reader. This is referred to as the backscattering of the RFID tags [6] [8]. Usually the tags have two stages of input impedances which it switches in between to modulate a backscattered signal, when enough induced voltage is provided to the RFID chip [6] [8]. The passive RFID tag chip anatomy, typical passive RFID tag specifications and the principle behind the backscattering signal are explained in section 4.1.2, 4.1.3 and 4.1.4 respectively.

In contrast to the passive tags, the semi-active tags do have an internal power supply. However it only powers the internal circuitry, and uses the passive tag methodology to transmit information. Since the tag circuitry is being powered internally, the response time of the tags are much faster than the passive tags. The lifespan of the tags are limited to the internal power supply, but it operates on much wider distance range. The active tags are completely powered by a battery. As a result it responds very fast and has longer range. The major tradeoff is that the circuit draws more power than the semi-passive tags and the passive tags, which results in a much shorter lifespan. Another disadvantage of the active transponder tags is that they are much bigger than the passive tags. They are also more expensive than both the semi-passive tags and the passive tags.

4.1.2 Tag Anatomy

The main components of the RFID transponders are the Application Specified Integrated Circuit (ASIC), and the tag antenna network. The tag antenna network is a carefully etched substrate in the printed circuit board. The most common substrates are polyamide, polyester, paper, FR4 etc. The design engineers use different substrates and antenna layout for optimizing the antenna range. The ASIC chips are surface mounted, wire bonded or attached through Flip-chip technology. The picture below shows a transponder passive tag with its ASIC and the tag antenna network [8].

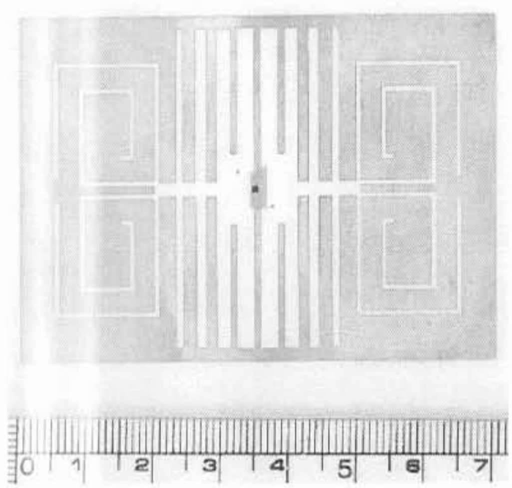


Figure 1: RFID Tag Transponder

The ASIC chip usually consists of the following components: EEPROM memory, RF front end, Analog Section, Digital Section, and Power storage device/battery. Conventionally the RFID tags use Electrically-Erasable-Programmable-Read-Only-Memory (EPROM) for storing data. The RF frond end establishes the connection between the ASIC chip and the RF antenna. The analog component synergize with the RF front end, EEPROM memory, and the power supply. The Digital component only responds to the RF Front End. The block diagram shown below illustrates the ASIC operation [8].

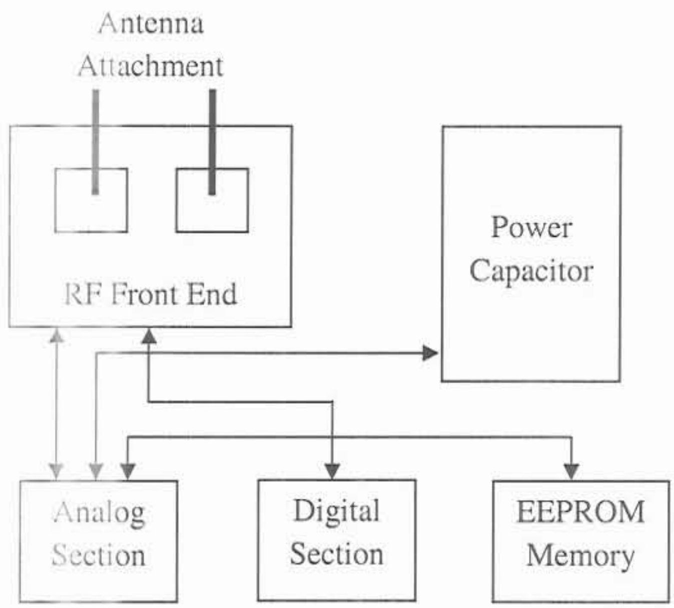


Figure 2: RFID tag Block Diagram

4.1.3 ASCI Chip Specifications

The typical ASCI chips are made on silicon. Some of the specifications of the chips are listed below [8].

- Ultra low power operation
  - 5-10 micro-watts during read operation
  - 80-120 micro-watts during writing operation

- Total EEPROM memory of 128 bytes
- Reserved system memory of eight bytes
- Eight byte tag identifier
- Operation temperature ranges from -40°C to 175°C

4.1.4 Passive and Semi-passive Tag Operation

As discussed before, typically, most passive RFID technology systems are based on the principle of modulated backscattered signal detection. A modulated signal from the base station can be picked up by a transponder tag. The principle of modulated back scattered signal detection is illustrated in the figure below, where the base station will transmit a wave which is modulated by the "Load" block and is reflected back to the base station [8].

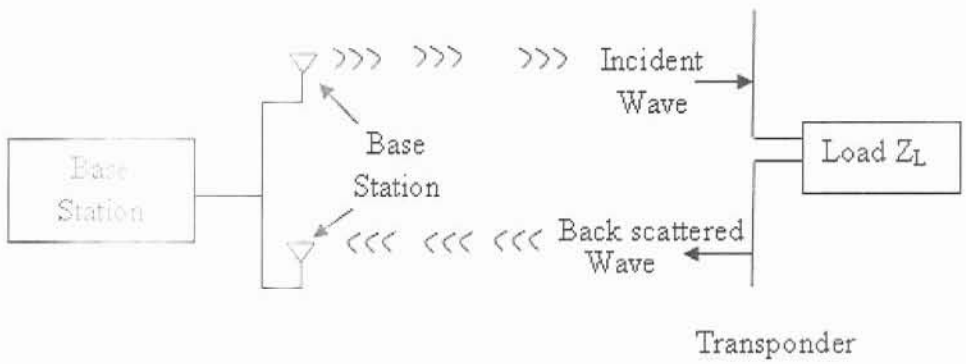


Figure 3: Back Scattered Signal Detection Diagram

When a transponder tag picks up a modulated signal, an induction voltage will appear at the input terminals of the transponder as a result of the RF fields from the base station [8]. The induced voltage is then detected by the RF chip that matches the impendence between the chip and the antenna. It is very important to have the antenna and the chip to have matched impedance because it affects tag performance such as read range, write range etc [6]. The induced voltage then charges a high capacitor providing the necessary bias for the processing circuitry. Therefore the capacitor needs to provide at least the minimum voltage required for the chip to be functional. This is depicted in the figure below [8].

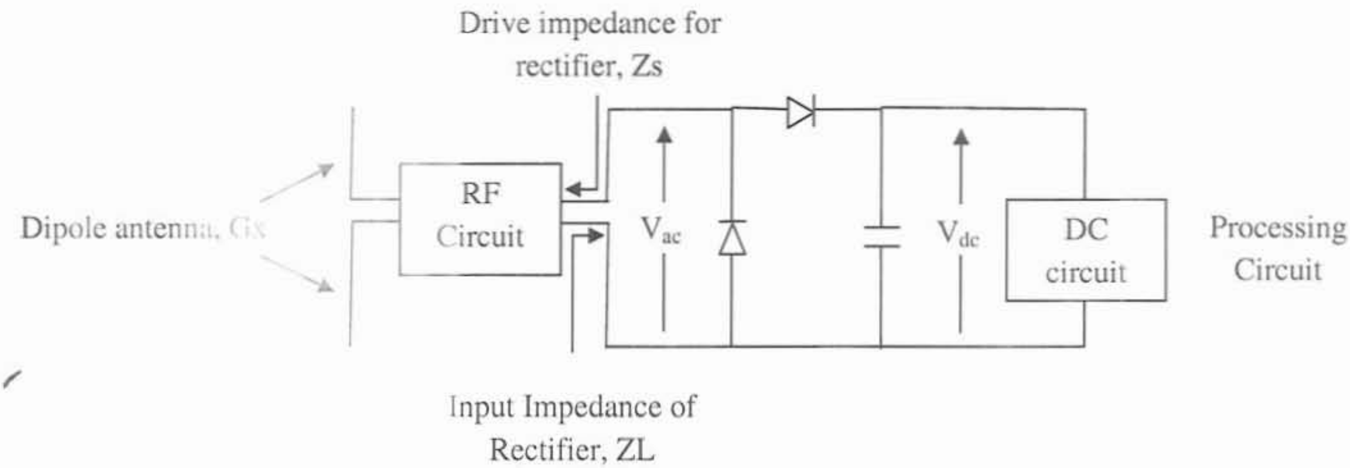


Figure 4: **Circuit Representation of the Transponder**

The transponder response due to the modulated signal is determined by the induced voltage. The transponder will respond if the induced voltage exceeds a threshold voltage that varies depending on the design of the antenna [8].

The processing unit takes in DC voltage and applies to a nonlinear load. The nonlinear load typical consists of two complex impedance stages. The impedances stages are significantly different, so a detectable backscattered signal can be produced. If the processing circuit responds to the modulated signal, it will run according to transponder protocol by transmitting a detectable back scatter signal to the base station [6]. The most important part of this process is that by changing the RF impendence, a successful the back scatter signal can be transmitted from the tag to the reader. This process can be used to either read from the tags or write to the tags [8]. The figure below shows the diagram of the processing circuit.

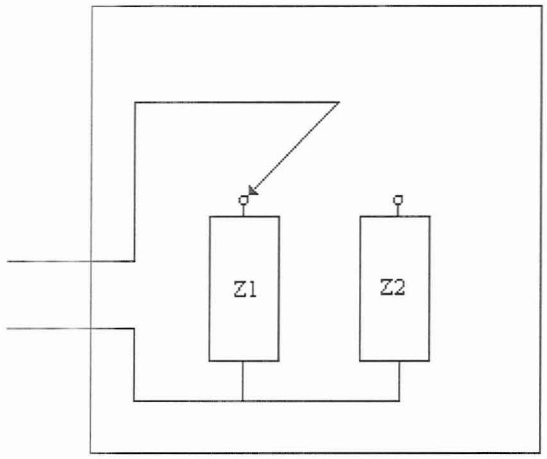


Figure 5: Processing Circuit Diagram

The Z1 and Z2 are the different impedance stages of the processing system. The RF impedance is varied by switching between Z1 and Z2 to produce the backscattering wave [6].

5. Applications of RFID

The RFID technology has a huge impact on the consumer market. It increases efficiency and convenience. The following sections provide a detailed explanation of this.

5.1 Retail Store

The RFID tags can be placed on each product to increase organization and expediency. Conventionally, retail stores uses barcodes to identify their product for either sales or inventory. Even though the barcode system is effective and is still used in stores, the RFID technology completely out grades the barcode system. The barcodes can only identify each product, whereas the RFID tags can store much more detailed data than just mere identification. The RFIDs are also compactable with external programs [3].

In contrast to the barcodes the RFID tags provides reading instantaneously and accurately, with minimal human interaction or involvement. The RFIDs eliminates the need of physically scanning each item, with automated and accurate readings. This removes the human error and lowers labor cost during check out or inventory. This functionality is a great use to the retail store if any of the packages or containers needed to be opened for individual scanning of the product inside. The RFID tags can also function effectively under difficult conditions such as dust or snow within the packaging materials [3].

Furthermore, the RFIDs can be implemented for data sharing between the store and the warehouse. Thus enabling the stores to monitor the stock's stored in the warehouse. Different applications driven by RFIDs can note the traffic of each item, as well as the remaining items on the shelves. This can be compared to the items in stock and further action can be taken to either alert proper officials, or automatically place an order to the warehouse of the manufacturer for restocking each item when the supply is low. The tags can also contain expiration dates on each product and warn the employees if the product is expired or about to expire [3].

The current delivering process of each store can also be simplified with RFID technology. The tags can be used for container-level, pallet-level, or case-level tracking. Utilizing this tracking hierarchy can reduce supply chain transit times and has a significant impact if the products are moved by multiple carriers in a single journey. Generally, each transaction of the products requires proof of delivery and other paper work to confirm the order to complete the delivery process. This can be automated using RFID technology with the case-level tracking; thus increasing efficiency by cutting down on labor cost and other plausible human error [3].

Another major advantage of replacing barcode with tags is that it can locate misplaced items. In most stores, customers often pick up items and leave them in the wrong places, which portrays a shortage or "out of stock" item in shelves. This hinders possible sale of the item and the store will appear disorganized. This will have a tremendous effect on the retail stores if the product is a high priced item such as electronics, DVDs or other apparels. If the RFID tags are incorporated in these products then they will appear in the system and it can be reshelf in the appropriate place [3].

Additionally the tags could be used to easily facilitate pricing changes and "streamline the return process" from customer to retailers. If the store prices tend to fluctuate often, due to different sales, expiration date, discontinuation of certain products or open items, the tags can be easily reprogrammed with the most updated prices. The return process of the products from the customer can be restructured by tagging the item if it is damaged, should be discounted or safe to put back to the shelves [3].

5.2 Manufacturing Market

In the manufacturing industry the RFID technology is growing rapidly. One of the main advantages of using RFID in the manufacturing industry is that it can be used to track different parts, programmed for different functionality, monitor operation status, and access past history. When manufacturing custom parts, the tags can be embedded with any raw statistical process control (SPC) data that is essential to the product. This will improve customer satisfaction and make the repair or replacement process of the malfunctioning parts much simpler. For example such applications can be targeted towards the automobile industry. The tags placed in the automobile can carry special part or assembly information. This can reduce labor time and provides reliable information [3].

Many manufacturing industries require timely update on the progress of each product assembly. The RFIDs can easily and effectively provide both the location and the product status instantaneously. As a result the sorting and pickup process of the products becomes a lot easier. The RFID can provide real time information about the product without any physical movement or additional labor [3].

6. Negative Impact or Short Coming of RFID

As the RFID technology continues to grow, more public concerns and stronger controversies of RFIDs also arise. The RFID technology has yet to ensure people's privacy and assure the safety of any personal or valuable information. The RFID technology enables companies to monitor their product even after it leaves the shelves of the retail stores. Companies can exploit this by tracking people's daily activities to find what stores they go frequently, what products they often buy etc. Also any third party individual with a RFID receiver and a RFID antenna can access information in the tags if they are not secured properly. Even if a tag is secured and is carrying important information such as credit card information and social security number it can potentially be infiltrated and the information can be accessed [3].

The more obvious short coming of RFIDs is that the initial cost of setting up a RFID system and maintaining the system can be expensive. Therefore many companies might not want to replace their older systems. The new RFID systems may require additional training for the employees. If such complications occur then the older systems will prevail and the RFID technology will be forced out of the market [3].

7. Possible Solutions

Many companies have proposed that the tags should be "killed" after leaving the stores. This will solve many of the public concerns of privacy. As for securing valuable information, the tags designers can develop complex tags that is very difficult for a third party to penetrate the security measures. Government involvement and strict privacy policies can also be regulated to reduce violation by a third party member. There will always be the risk of theft, but the best manner of securing the tags are overseeing all the possible cracks and programming the tags accordingly. There are many tags on the market currently, which has its own password enabled security. All the GEN2 tags have a 32 bit access password. As time progress any other short coming can be accounted for with better design or programming.

Even thought the initial cost of the RFIDs is a little expensive, their versatile functions may save the company more money on the long run. Also, competition between manufactures and new technology can decrease the cost of RFID technology in the near future.

8. Biomedical

In order to evaluate the effect of RFIDs on Biomedical field we used a prior IQP, which analyzed the current information access system and patient care in the hospitals. The report shows that RFID technology in the biomedical field can completely modernize the emergency room (ER) procedure and patient tracking. The following sections unfold the drawbacks of the hospital current system and provide a RFID solution, which can increase efficiency and convenience.

8.1 Typical Emergency Department Infrastructure

The medical emergency department that the prior IQP group studied was the UMass Memorial Health Center (UMMHC). The emergency department at UMMHC is four levels high, and about 200 by 400 feet. The staff at the Emergency Department (ED) includes 62 physicians and 14 nurses, with approximately 36 residents requiring care. The average salary of a physician is about $255,000, whereas the nurses' salary is about $125,000. The reason for such a high salary for nurses is due to the extensive amount of work required from them. On average they walk about 12 miles per shift. The nurses spend approximately 10 minutes entering data for a given patient. Physicians spend about half their time just trying to locate staff members and/or patients. Locating them is highly inefficient, since the emergency department at UMMHC receives about 75,000 patients a year, with about 300 patients on busy days, and 170 patients on slower days. Currently, UMMHC uses a non-real time barcode wristband system for patient identification. The system is called Ibex, and is a part of PICIS. It requires manual data entry for patient information. There is also another database program called Meditech which stores the patient's profile and medication. The systems are upgraded through the internet, and cost approximately $2,500 to $3,500 a year to maintain. The entire system requires a significant amount of work for data entry [7].

8.2 Typical Patient ER Procedure

This section provides a careful analysis of typical ER procedure. Many of the procedures consist of data entry or update of the patient. These procedures are manually executed and consume a lot time. Even though the current process is effective, it can be extensively simplified through RFID technology.

A typical ER procedure is initiated when a patient either enters the emergency department (ED) through the front door or through the dock via ambulance. The patient is then registered with the greeter or the registration staff and is sent to Triage, where the hospital categorizes the treatment plan. As soon as Triage is done, the patient's information is added into the Meditech database, and their name is placed with the Ibex tracking system. Next, the patient is sent to a room within the ED, or asked to wait in the waiting area, depending in the availability of rooms at the time [7].

Patients brought through ambulance have no need to go through the greeter or to go through the Triage process. They are first registered into Meditech and placed in the tracking system. Based on medical conditions, an ER patient may be moved from the ED to another hospital department through the course of treatment. A nurse would manually update the status of the patient's location in Ipex tracking system every time a move would occur. Figure 1 is taken from the work of the pervious IQP team, which shows the typical movement of an ER patient [7].

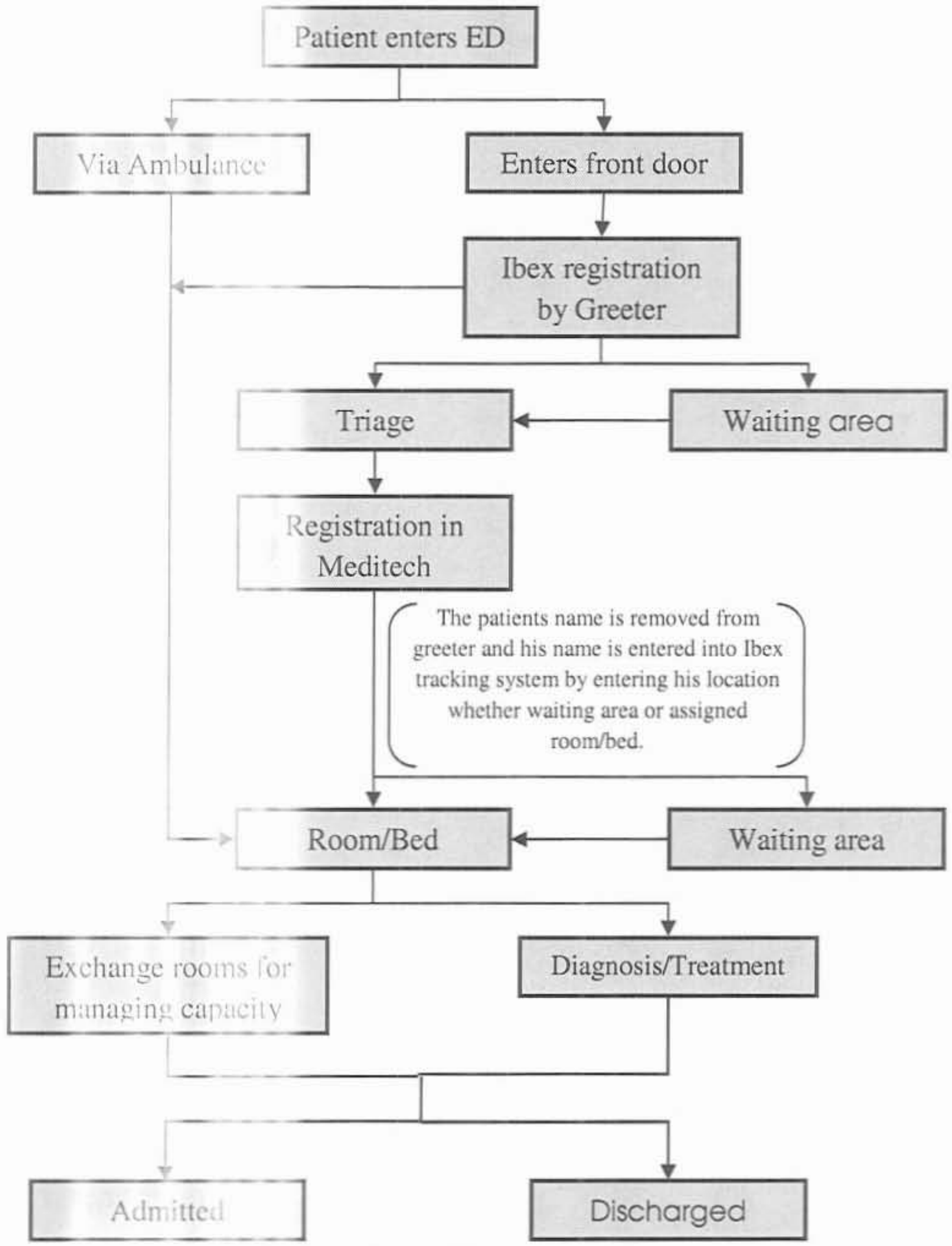


Figure 6: Typical Movement of ER Patient [7]

When a move occurs the patient enters the ED either by ambulance or by themselves from the front door. The patient entering from the front door is registered into Ibex greeter and sent to triage or waiting room depending upon the availability. The patient's treatment plan is prioritized in triage and assigned to a room or a hallway bed. Before they are sent to the room, their information is registered in Meditech and removed from the Ibex and then entered into tracking system. The patients entering by ambulance are directly taken to bed and registration for Meditech is done in their assigned room. All patients are often moved to different departments for diagnosis and treatment during their stay in ED. At the end of the process they are either admitted for after care and transported to the main hospital or are discharged. Hence for keeping track of patients their status is changed manually into Ibex whenever a movement occurs within ED [7].

During a patient's stay within the ED, the following is done to keep track of a patient's location and condition:

- The patient is registered into Ibex.
  - The patient's information, such as name, gender, and chief complain is entered into the Ibex tracking system.
  - And identification wristband is issued to the patient for tracking purposes.
- Patient is either asked to wait in the waiting area, or taken to Triage to determine the treatment.
  - Upon entering the Triage, the status of the patient is changed in the Ibex tracking system, and their name is taken out of the greeter screen. A patient's name in the greeter screen shows that the patient has not yet been seen at the Triage or by any medical staff.
  - Location changes are manually entered into Ibex.
- After being assessed at the Triage, the patient is registered into Meditech.
  - Patient's profile is entered into Meditech for record keeping.
  - Other medical information such as charts and medication are added to the profile through the course of treatment.
- Patient is asked to be seated in the waiting area, or taken to an ED room depending on room availability.
  - The patient's status is set to "wait" if there are no ED rooms available.
  - If an ED room is available, the status of the patient is changed to the location of that room.
  - The doctor and nurse assigned to the patient are entered into the patient's profile.

- Depending on the treatment required for the patient, the patient must be relocated to a different hospital department.
  - As a patient enters or leaves an assigned room, the location status in the Ibex tracking system is changed.
  - The status of the patient's diagnosis is also entered manually into the system.
- Patient is taken into operation room if surgery is required or other such treatment.
  - The patient's status is changed to the operating room when the location switch is made.
  - The status of treatment is again updated in the system.
- During busier times during the day, patients are moved from the ED rooms depending on availability and status.
  - The status of the patient leaving the room, and entering the room must be updated in the system manually.
- Follow on care may be required for the patient after treatment.
  - Follow-ups are arranged and entered into the Ibex system.
  - The patient, when discharged to a hospital bed outside the ED, is taken out of the Ibex tracking system, manually.
- The treatment may be complete, and the patient is stable.
  - The patient is discharged and the status within the Ibex system is changed to discharged and treated.
  - The status of the ED room that the patient was in is changed to vacant.

8.3 Problems in the Patient Tracking System

For the following reasons, it is absolutely vital to know the status and location of every patient:

- Medications that a patient needs must be delivered to the correct room and administered to the right patient.
- Any equipment necessary for the patient's treatment must be dispatched to the correct room.
- Family members or phone calls must be forwarded to the correct room.
- During the busier hours, it must be ensured that the patients are not misplaced.

All updates to the UMMHC tracking system are done manually with a combination of the Ibex tracking system and the Meditech database. This tracking approach severely stunts the efficiency of operation within the ED and places significant amount of work on the nurses. There is large room for human error in the data entry process, resulting in misplacement of patients, equipments, or medication administration. This can cause major delays in the operation of the department as a whole, since equipment may arrive at the wrong rooms, patient profiles can be misplaced, and medication misdirected [7].

The system inefficiencies may cause dissatisfaction among patients, and confusion among the staff. For example, if a patient sustains multiple injuries through a car crash, he/she may need to be seen by different specialists. For each of the specialists to diagnose and treat the patient, they would have to check the Ibex system for the status and location of the patient. Assuming a delay in the data entry into the system, the physician would not be able to access information on the patient. This would result in the delay of treatment for the patient. Also, if multiple physicians are involved in the treatment, this would require real-time updates [7].

Some of the vacant ED rooms may not show in the database if there is delay with data entry during busy periods. This would not allow the ED to be used to its full capacity. Some less likely problems are that a patient may wander from the waiting area, making him/her difficult to locate. Confusion and misidentification may also occur, to the point that the wrong medication may be administered to the incorrect patient [7].

8.4 Tracking with RFID

Tracking with RFID is a very useful application in respect to patients, staffs, new born babies, medical equipment and surgical tools. The following section provides a more detail explanation of this.

8.4.1 Patients

To ensure patient safety, patient identification and real-time location is vital. At times, there may be a need for urgent medical attention. The ED needs to have optimized patient flow for quality and efficiency. A suboptimal patient flow includes delays, overcrowding, and financial losses. Safety is also an issue when considering lost or misplaced patients, especially patients who are children, patients with Alzheimer's, or the elderly pose a higher risk.

According to the Healthcare Advisory Board, hospitals lose 4-5 % of their revenue due to the lack of proper patient management. Tagging patients with RFID can significantly decrease revenue losses, as well as aid in better patient identification at times of surgery and administration of medicine. The RFID approach can significantly reduce medical errors, considering that 44,000 to 98,000 patients die each year in the United States due to medical errors [7].

8.4.2 Staff

Staff tracking is important to the work flow of the ED at the bigger picture. In the current system in place at UMMHC, nurses and other important staff are hard to locate which increases cost and damages work flow. Using RFID technology authorized officials can get real time update of the staffs and eliminate most of these problems [7].

8.4.3 Newborn Babies

A significant number of newborn babies are stolen from hospitals each year. In addition, babies are sometimes matched with the wrong parents due to logging errors. Using RFID tags to track babies will not only track the location, but also the identity the newborn. This will reduce most human errors and negligence [7].

8.4.4 Medical Equipment

Many hospitals struggle with the management of their equipment, such as infusion pumps, EKGs, ventilators, portable x-ray machines, gurneys and wheelchairs. Nurses that should be caring for patients are obligated to find the proper equipment instead. This results in unsatisfied patients. To counter this, hospitals then purchase additional equipment to ensure that the quality of care is not affected. Hospitals report that approximately 15% - 20% of their assets are missing. This is about $3.9 billion each year. With the use of RFID tagging, thefts or misplacement of medical equipment can be prevented and promotes proper allocation of the equipments [7].

8.4.5 Surgical Tools

Surgeons sometimes fear that after an operation, a tool may be left inside the patient's body. Using a small RFID tag on each tool would eliminate the possibility of such an occurrence. This would allow the surgeon to concentrate entirely on the operation itself, rather than possible mishaps [7].

9. RFID Implants

The RFID implants are paving the road for a "key-free world." In order to get a better understanding of the RFID implants and the effect of RFID implants on people, this section covers the experience of Amal Graafrstra and his girl friend Jennifer Tomblin getting RFID implants to simplify their lifestyle. Graafrstra and Tomblin are ordinary individuals who pursued the RFID implantation life style. Graafrstra and Tomblin are among the fewest people who have RFID implants. Graafstra felt that his big ring of keys was too overbearing and the keys could be stolen and forged at any hardware store. Therefore he was compelled to implant RFIDs to eliminate the use of keys. The section below explains Graafstra's getting "chipped" process [4].

9.1 Graafstra's Experience in Choosing the Right Tag

There is variety of tags available on the market. In order to get a RFID implant, the individual must choose a tag that suits their needs. Some of the important questions that the user needs to consider are that what purpose do the RFIDs serve them? Where is the tag being implanted? In Graafstra's case he was just looking for a secure easy access solution. Thus he researched many different tags that are safe to be implanted in the human body. Graafstra consider of using one of the earliest implantable tags, which are manufactured by VerChip Corp. However, the VeriChip tags had many draw backs which Graafstra found unsuitable for his life style. The tag readers are very expensive and the procedure must be approved by the US Food and Drug Administration. Above all, the tag must be placed in the upper arm, which makes it a huge inconvenience for him to wave the tag over the reader. Therefore he searched for an alternative implantable tag [4].

Graafstra turned this attention towards the animal tracking tags. Even though they are safe to use, the tags contain an anti-removable coating and the tags readers were not much cheaper than the VeriChip tags. Moreover, the tags were not intended for any thing other than animal tracking, so it was difficult to manipulate them for Graafstra's purposes [4].

Finally, Graafstra found the EM4102 chip. The EM4102 chip is similar to the animal tracking chip, but without the anti-removable coating. The tag reader was also very inexpensive, which ranged from $30 to $50. The tags contained in a small glass packing were similar to the VeriChip tags and the animal tracking tags. The only concerns were that the tags were not intended for human implantations, so it was not sterile and the glass could break inside Graafstra's body. However, when he called the company he found out that the tags were safe to use, but they are not designed for the communication standards for animals. This was not a huge concern for Graafstra. He also tested the durability of the tags by smashing them with a hammer. He found that the tags can handle some abuse and the stronger blow required to break the glass casing of the RFID would injure his hands regardless of the RFID implantation [4]. Therefore he made arrangements to implants the tags in hands, which is explained in the section below.

9.2 RFID Implantation Procedure

RFID implantation procedure is very simple. Graafstra met with a cosmetic surgeon and filled out appropriate wavier for surgery. The surgeon made a small insertion beneath the skin and planted the tag after sterilizing with a liquid disinfectant. There are single use dispenser kits that inject 2 by 12 millimeter tags into your body. With the aid of a family doctor, a cosmetic surgeon or even a body piercing specialist, this procedure can be completed within five minutes [4].

9.3 RFID Implantation Advantages

RFID implantation integrates all the desired functions of the RFID technology into a chip at your disposal. All the important applicable data can be carried on you, thus making life easy and convenient for the user. Graafstra designed an easy entry system into his house just by waving his hand. He also implemented this with his computer for auto login. He extended this technology to his motor cycle and auto mobile for entry and auto start. Several others have automated many of mindless duties with RFID implants such as controlling the lights, and storing driving license information. Others used it for medical purposed such alerting doctors of medical conditions. This is extremely useful if the patient is admitted unconscious and the implant contains information that identifies the person as diabetic patient, and contains information of medication allergies and blood type. The tags are also easily removable if it doesn't have an anti-removing coating on them. Therefore upgrades and disposal of the tags are very simple [4].

9.4 Securing Access

Breaking in and entry was a concern for Graafstra, since there are devices that can decrypt a signal transmitted by the tags. The foreign device can store the decrypted signal and play back the analog signal to get access to most systems. Thus, Graafstra implanted a second chip in his hands with 40 bit security inscription to reduce some of his concerns. The new implanted tag has protected data storage area and has random key assigned every time the tag is read. This keeps the third party guessing of what the next access key is. This makes any forgery or replication of access key very difficult, costly and time consuming [4].

9.5 Feasibility Study

There are wide range of tags and readers on the market. The tags are usually under couple of dollars, but the readers' ranges from $30 to $2000, depending on the system. We are mainly focused on readers that operate on 915 MHz frequency. The table below looks at some common RFID competitors in the 915 MHz range:

Table 1: Comparing Different RFID readers (1)(2)(9)(10)(11)

Reader	Tags Supported	Features	Price
Mercury 4	EPC Class 0, 0+ EPC Class 1 Gen 1 EPC Class 1 Gen 2 ISO 18000-6B	High Speed DSP. Compatible with most tags. Low Cost.	$ 995
Mercury 5	EPC Class 0, 0+ EPC Class 1 Gen 1 EPC Class 1 Gen 2 EPC Class 1 Gen3 ISO 18000-6B ISO 18000-6C Variants of Gen 2	High Speed DSP Much less susceptive to interference. High Performance. DevKit included. Compatible with almost all tags.	$1,995
Alien ALR 9780	EPC Class 1 Gen 1 EPC Class 1 Gen 2	DevKit Included.	$1,899
Alien ALR 9800	EPC Class 0, 0+ EPC Class 1 Gen 1 EPC Class 1 Gen 2	High Performance. Less Interference. Real-time health status and revision information. Dev Kit Included.	$2,195
Symbol XR480	EPC Class 1 Gen 1 EPC Class 1 Gen 2	Easy integration with third party software. Supports European ETSI EN 302 208 standard.	$1,749

As shown on table 1, the Mercury 4 RFID reader would be the best choice, due to its extensive ability to read many different tag protocols and it's extremely low cost relative to other readers with similar features. However, the Mercury 5 reader would also be an excellent investment due to its increased performance and it's ability to read future tag protocols that are being developed [1][2][9][10][11].

10. Field Testing and Tag Performance

To test the performance of our reader, we used the MQP laboratory of the Atwater Kent building. The picture below shows the RFID system setup in the MQP lab. The setup consists of 915 MHz Mercury4 RFID reader, 915 MHz Sensomatic RFID antenna, and a tripod with sticky passive tags placed on them.

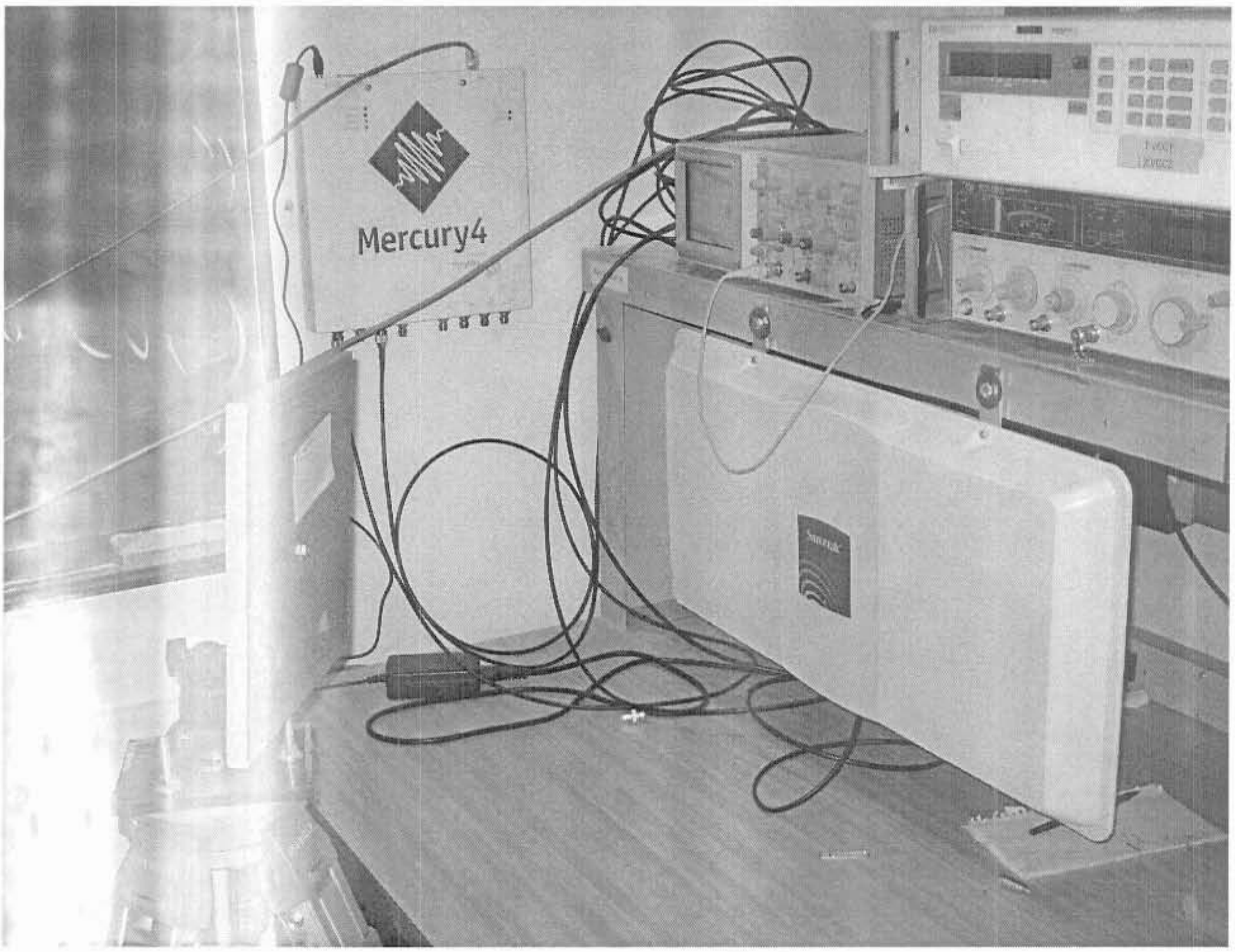


Figure 7: RFID System Setup

Although the room is not very environmentally friendly due to the amount of metal, the test bed was more or less equivalent for all our tests. We tested from a range of 1 foot to 10 feet. The tag used was a UPM Rafsec, model PD70_18_6 A1. The tests were run for 60 seconds each, and the number of times the tag was read was recorded.

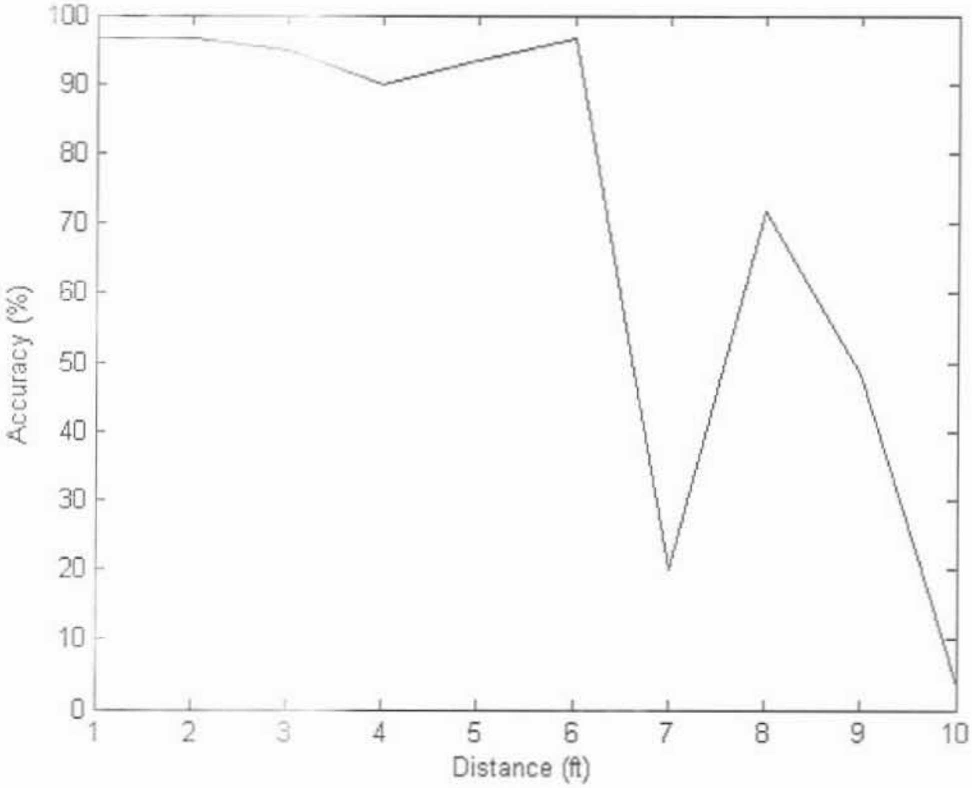


Figure 8: Tag Performance Graph

As can be seen from the graph, the readings for the graph were 90% - 97% accurate from one to six feet. There was a small anomaly at seven feet. The number of reads per minute dropped significantly, but at eight feet the readings rose again. A plausible reason for this is that there is significant tag impedance mismatch between the chip impedance and the input power. This usually results in "dead spots" within the tag operational range [6]. Given a cleaner and friendlier environment, the reading for the RFID might have been smoother and more accurate. At about eight feet, the readings dropped rapidly and at ten feet the readings were only about three percent accurate, or two reads per minute.

11. Yagi Mercury4 OS

Yagi Mercury4 OS is a web applet provided by Tego Inc. This applet is used to perform various tag functions. The most common operations are query and write. The following sections explain the query operations, provides the data description of tags and the writing procedures.

11.1 Query and Data Description

Tesla MercuryOS 2.3 is an online software that is used to read, write and implement 915 MHz RFID tags. The software supports RQL programming language for the above purposes. The tags that we analyzed are Alien Technology ECP1 model ALL-9350-02, Matrics RX3000 EPC0 and UPM Rafsec GEN2 tags. A sample query of the tags is shown below.

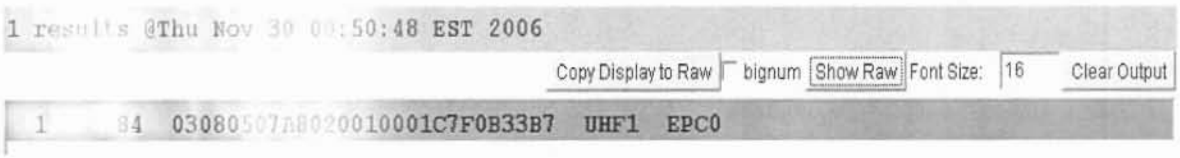


Figure 9: Query data

The first term '1' represents the item number. Since there was only one tag read by reader the applet outputs only one. If there are multiple tags present then each tag would be consecutively numbered. The second term "84" represents the number of time that the tag was counted. The third term shows the hex digits that are embedded in the tag, in which the last four digits are the cyclical redundancy check (CRC) packet coding. However in EPC1 tags, the CRC packet is the first four digits of the data. The fourth and the fifth terms are antenna and protocol respectively.

The recorded data can be further analyzed by selecting the "Show Raw" button, which is shown below.

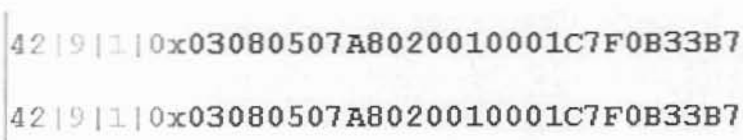


Figure 10: Raw data

The terms are divided by the "|" symbol. Each of the first terms corresponds to the number of time the tag was read by the applet for each query which is 42 counts. Since there are two queries made the total number of counts are 84, which equal the second term of figure 10. The second term is the protocol for the tags. The reader shows that '9' is Matrics tag's protocol and '1' is the Alien technology tag's protocol. These protocols vary according to different companies, but stay the same for the same models. Thus the three Alien technology tags have the same protocols. Furthermore, the third term signifies the antenna, so it will remain 1 for different tags because we are only using one antenna. The following term displays the data in hex digits which are denoted by 0x. As explained before the last four digits are CRC packets.

11.2 Writing Procedures

The writing procedures only comply with tags that are designed to be writable. Most of the GEN2 tags and unlocked EPC class 1 tags are rewritable. In the Yagi applet, the writing procedure is done by selecting the appropriate tag choice in the writing applet and pressing the "Make Update Button", which will result in the following code appearing in the GUI console.

UPDATE tag_id SET id=0x WHERE protocol_id='EPC1' AND antenna_id=1;

The data entry we wished to be added must be inserted after "id=0x" where the 0x indicates a hexadecimal value.

Then the applet requires you to press the "Submit Query" button. Submitting the Query results in the following message displayed in the bottom console:

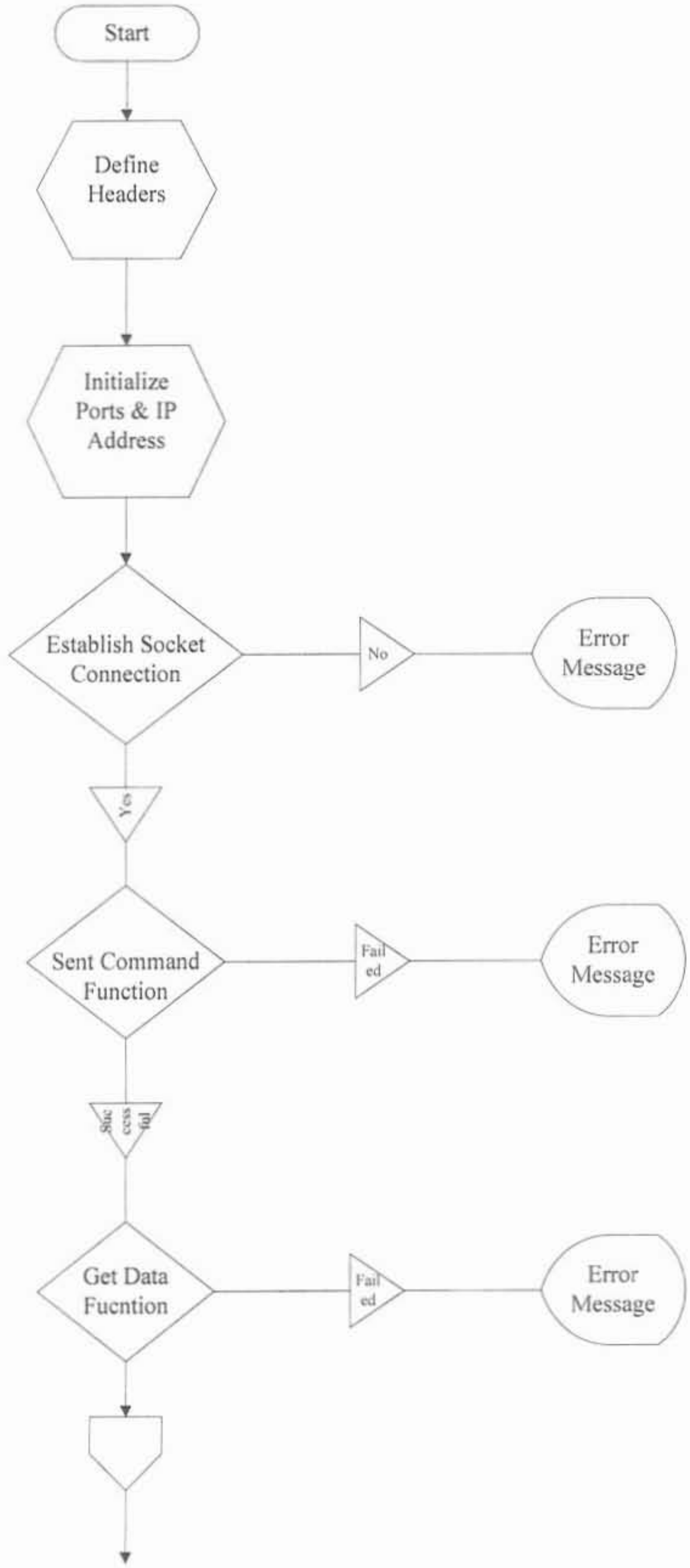
UPDATE tag_id SET id=0x121212121212121212121212 WHERE protocol_id='EPC1' AND antenna_id=1;

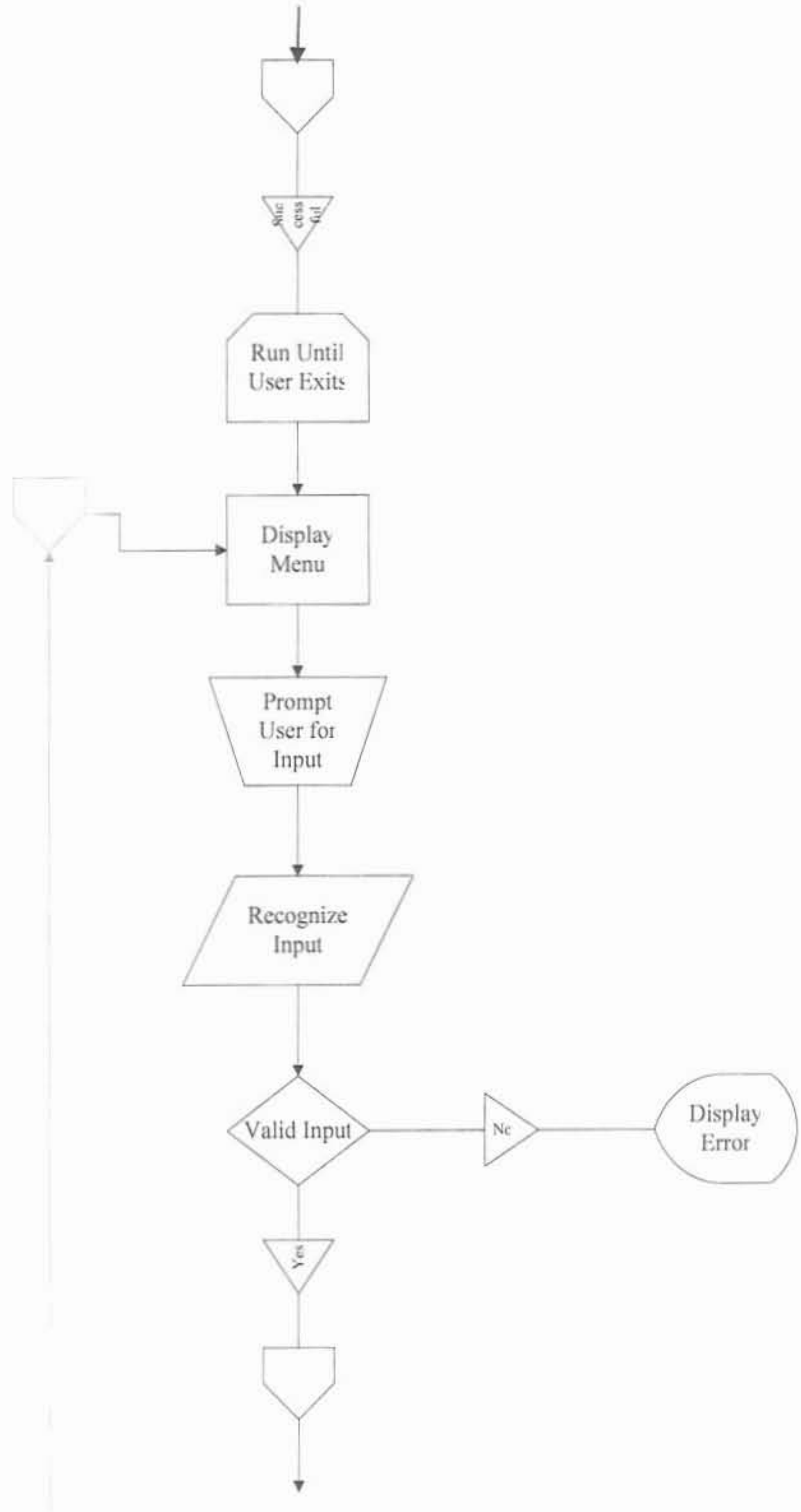
0x0AE2121212121212121212121212

The 0x0AE2121212121212121212121212 is the tag id and the first four digits are the cyclic redundancy check packet.

12. External Applet Application

The external applet application is created in Visual C++ environment. This application is very similar to the Yagi Mercury4 OS and it can be executed from any computer. It is completely independent of the Mercury4 header files and DLL files that were provided for us. The block diagram shown in Figure 11 illustrates our coding approach.


Start
Define Headers
Initialize Ports & IP Address
Establish Socket Connection
No
Error Message
Yes
Sent Command Function
Failed
Error Message
Successful
Get Data Fucntion
Failed
Error Message


Suc
cess
ful
Run Until
User Exits
Display
Menu
Prompt
User for
Input
Recognize
Input
Valid Input
No
Display
Error
Yes

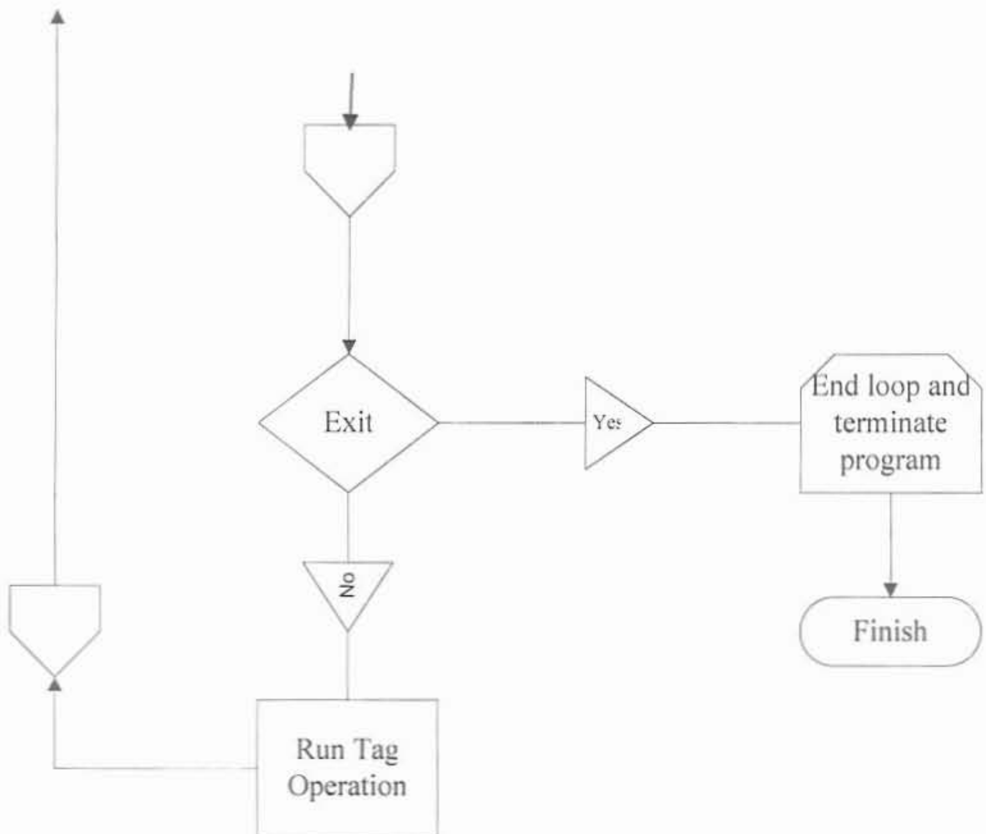


Figure 11: Code Block Diagram

As shown in block diagram the first step is to initialize all the required headers for the program. Different functions that we implemented specify which headers to utilize. The next step is to establish a socket connection between the reader and the external application by specifying the port and the IP address. If the connection fails then an error message is displayed to alert the user. Furthermore, two different functions are used to communicate with the reader. The send command and get data command are the main functions that we utilize for either sending or receiving data. Reader Query Language coding can be sent to the reader with the send command to perform different tag operations. The get data function will return any data from the reader to the external program. Some of the tag operations that we implemented are read tag id, write tag id, kill tag and exit. These are shown in the user interface below.

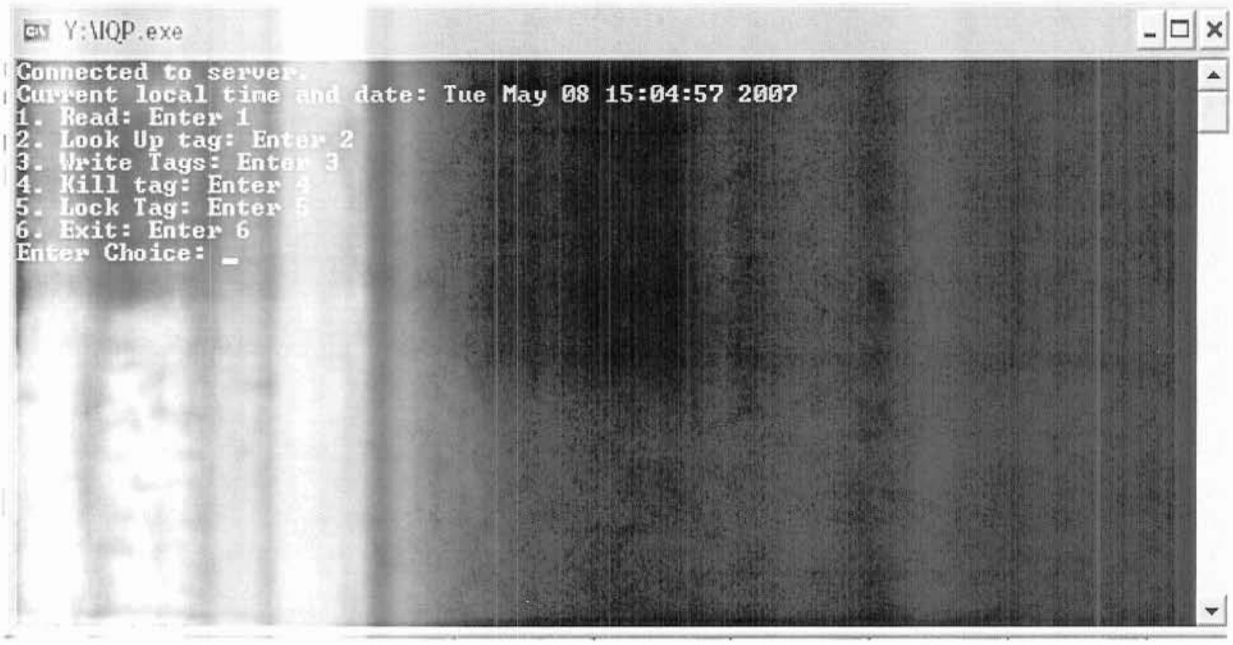


Figure 12: User Interface Menu

The Read function simply displays the tag id, antenna id, tag type, and CRC coding. We w[illegible] able to read the [illegible] class1 and GEN2 tag ids. The output of the Read function is a[illegible] to Figure 10 which is the Yagi query interface. The figure below shows the output of o[illegible]de.

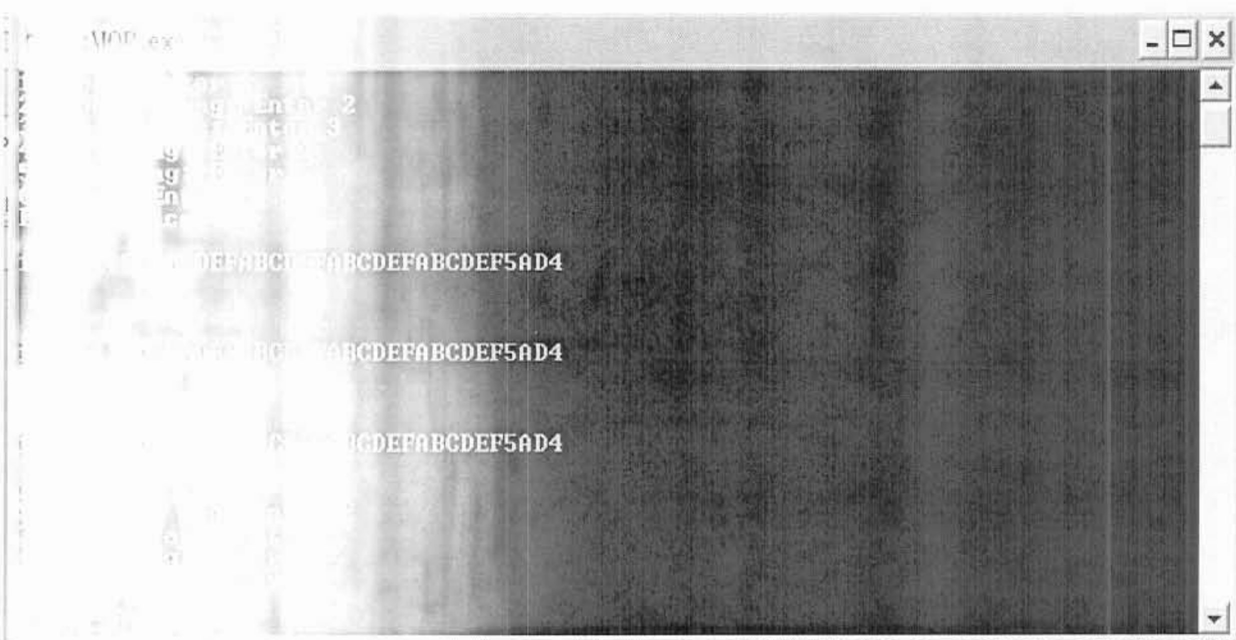


Figure 13: Read User Interface

The write function requires some parameters to be inputted from the user. The user needs to specify the tag type and tag id. When using the write function, the tag must be writable and within one foot of range. We were able to successfully write to EPC class1 and the GEN2 tags. The figure below illustrates the operation of the write function for EPC class1 tags.

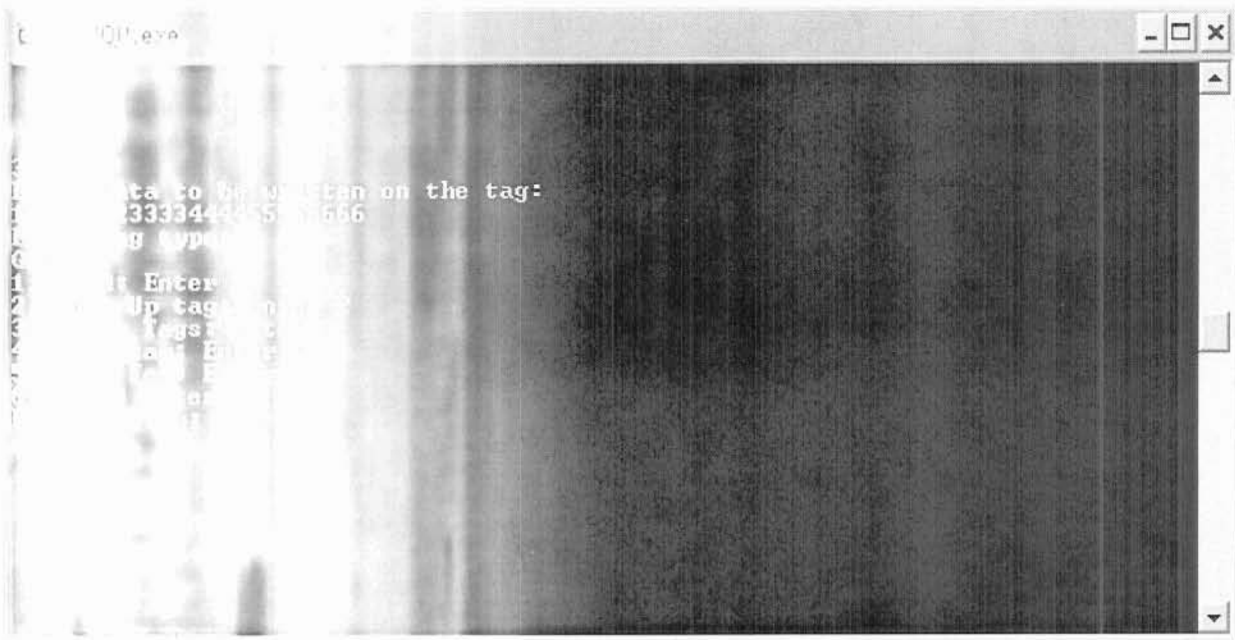


Figure 14: Write User Interface

The Kill tag function will permanently destroys the tags, thus the tags will be no longer operational. This function requires a password input from the user to authenticate them to kill the tags. The user also needs to specify the tag id. We were successfully able to kill two EPC class 1 tags, but the GEN2 tags were not responding to either the external applet or the Yagi web applet. Similar to the kill tag function, the lock tag will temporally make the specified tag nonfunctional. This function also requires a password to either lock or unlock tags. Neither the Yagi applet nor our external program was able to assign a password for to the tags. Therefore we concluded that there is a bug in the Mercury4 firmware because both applet outputs the same generic error. Once the firmware is updated this will not be an issue. Another useful function is the Look Up tag function, which will look either GEN2 or EPC class one tags at the users' discretion. The figure below illustrates the Look Up tag function.

Figure 15: Look Up User Interface

Finally, the exit function will exit out of the continuous loop and terminates the program. If the user inputs any invalid choices or tag types the program will alert the user that an invalid entry is entered and then the program will return main menu.

Our code is easily updatable. Different functions can be implemented into the old code by making test cases. The code is designed to run forever, unless the user chooses to exit. The most updated version of our code for our external applet is shown in Appendix A.

11 Recommendations

For future work on our project, our program should incorporate a Graphical User Interface (GUI) for ease of operability. Also, it should have a logging feature where it would log all tags read and be able to search through the log for a given ID. This feature should also incorporate a filtering option that can show the user ID's by date, time, location, ID number, protocol, and other such characteristics. It would also help to have a better testing environment for the reader since the environment we tested in was very unfriendly to the reader. Performance testing in dusty and cold area can also be conducted to understand the tag performance under severe conditions. This will helps us get a better understating of how the tag will perform in a retail store and manufacturing industry.

Further development of other RFID features are also desired, including the further development of Lock/Unlock, and Password Set commands. This could be extended to the GEN2 tags and the ISO tags, so a versatile number of tags could be used. A timing feature for the reader should also be developed, which would allow the user to enter a time period in which the reader would continue searching and logging RFID tags of certain protocols. The GEN2 tags also have memory banks which the user can utilize. The user can write, read, or lock the memory banks. This would be a great feature to develop in the future.

14. Conclusion

Our IQP project consists of two parts, which are researching and conducting a study of the effect and the social implication of RFID on the consumer market, and executing RFID tag implementation through both Mercury4 Yagi OS and an external C application.

We started our social implication of RFID technology by reviewing the history of the development of RFIDs. Our research illustrates that the RFID technology has grown a great deal since it was developed, but it has a long way to go before the technology is exhausted to its full potential. Throughout the years the tags are becoming more discrete and efficient and the different operation capabilities of the tags are growing. This indicates that RFID tags will continue to greatly influence the consumer market.

In order to understand the RFID technology, we looked the different types of tags and tag designs. Our research indicates that for low range applications the passive tags are ideal since they are cheap, discrete and has a very large life span. For longer range applications active tags could be used, but the life span is limited to the life of the battery. If needed, the semi active tags could be used for everything in between, which will boost the batter life, but shortens the tag range. Both the passive tags and the semi active tags uses the backscattered wave method to transmit data back to the base by using a nonlinear impedance load that will switch between two states to produce the backscattered wave.

Furthermore, we conducted a market research that reimburses the fact that the RFID technology is growing tremendously in the consumer market. The retailers, the manufactures, and the hospitals are considering incorporating RFID technology into their product. Their main concern is the initial price of setting up the RFID technology. However it is being implemented in some stores such as Wal-Mart. Some branches of the General Motor Corp. are also taking advantage of the RFID technology by integrating them into their car parts to simplify the assembly process. Many other companies are utilizing RFIDs for simple applications such as easy access entry. This is because RFID increases the ease of operability in various environments, lowers costs and expenses in the long term, and decreases the amount of human errors. As time passes the RFID technology will continue to be cheaper because of competition and it will be widely used in the consumer market.

As a part of our research a feasibility study was also conducted. Our feasibility study shows that even though the 915 MHz readers are pretty expensive, they can perform many tags functions, support multiple antennas and tags. If the barcode system was to be replaced by such a system, and then only one reader and four antennas are required for running four registers. Such properties boost the efficiency and applicability of the RFID technology.

Our field and performance testing showed the great aptitude of the reader given the extremely unfriendly environment for RFID. We experienced an anomaly at 7 feet and a 3% accuracy drop at 1 foot, due to impedance mismatching. Again, a more RF friendly environment would create much more favorable results.

Our code is a very good example, which illustrates that anyone can implement RFID technology to their product, work or lifestyle. Just like Graafstra's easy access doors application, any one can simply create an external application that will suit their needs. The application and the tags can be reused or upgraded easily without many expenses to the user. This makes the RFID technology very flexible, and can be applied in many different fields.

Through doing this project, we agree that RFID is indeed an excellent solution for many applications throughout the world, and especially in the consumer market and the biomedical field.

References

[1] "ALIEN ALR 9780 915-MHZ Reader KIT CLASS-1 4-PORTS - Trust RFID." TrustRFID. 2000. TrustRFID Inc. 7 May 2007 http://store.trustrfid.com/alien-alr-9780-915mhz-reader-kit-class1-4ports-p-3243.html

[2] "ALIEN ALR 9800 RFID Reader - Trust RFID." TrustRFID. 2000. TrustRFID Inc. 7 May 2007 http://www.store.trustrfid.com/alien-alr-9800-rfid-reader-p-3245.html

[3] Chopra, Sunil; Sodhi, ManMohan S. "In Search of RFID's Sweet Spot". Wall Street Journal March 3, 2007: R10.

[4] Graafstra, Amal. "How Radio-Frequency Identification and I got personal". IEEE Spectrum March 2007: 18-23.

[5] Landt, Jeremy. "The history of RFID." IEEE. April 2007 http://ieeexplore.ieee.org/iel5/45/33027/01549751.pdf?tp=&arnumber=1549751&isnumber=33027

[6] Nikitin, Pavel V., and K.V. Rao. "Theory and Measurement of Backscattering from RFID Tags." 06 12 2006 212-217. 5/12/07 http://www.ee.washington.edu/faculty/nikitin_pavel/papers/APmag_2006.pdf

[7] Ng, Frankie, Sweta Patel, and Amid Zand. Patient Tracking in Emergency Department Using RFID. Worcester: WPI, 2006.

[8] Rao, K.V.S. "AN OVERVIEW OF BACK SCATTERED RADIO FREQUENCY." Intermec Technologies Corp. April 2007 http://ieeexplore.ieee.org/iel5/6678/18000/00833700.pdf

[9] "Symbol XR440 RFID Reader - Trust RFID." TrustRFID. 2000. TrustRFID Inc. 7 May 2007 http://www.store.trustrfid.com/symbol-xr440-rfid-reader-p-4268.html

[10] "ThingMagic Mercury4 RFID Reader - Trust RFID." TrustRFID. 2000. TrustRFID Inc. 7 May 2007 http://www.store.trustrfid.com/thingmagic-mercury4-rfid-reader-p-4271.html

[11] "ThingMagic Mercury5 Reader - Trust RFID." TrustRFID. 2000. TrustRFID Inc. 7 May 2007 http://store.trustrfid.com/thingmagic-mercury5-reader-p-4269.html

Appendix A: External Applet Code

```
// IQP.cpp : Defines the entry point for the console application.
//

#include <iostream>
#include <winsock2.h>
#include <stdio.h>
#include <string.h>
#include <tchar.h>
#include <time.h>
#include <fstream>

using namespace std;  //So we have cout, cerr, cin....

//send data to the Mercury4
void send_cmd(SOCKET the_sock, const char *the_cmd){
	//initialize parameters
	int num_bytes = int(strlen(the_cmd));
	int result = send(the_sock, the_cmd, num_bytes, 0);
	if (result != num_bytes){
	//error: send failed
	fprintf(stderr, "send failed\n");
	exit(3);  //exit
	}
}


//get data from Mercury4
int get_data(SOCKET the_sock, char *recv_buffer, int buffer_size){
	int num_bytes = recv(the_sock, recv_buffer, buffer_size, 0);
	switch(num_bytes){
		case 0:
			// connection closed
			exit(1);  // or throw something. this would be the C++ way
to "gracefully" report things
			break;
		case SOCKET_ERROR:
			// call WSAGetLastError to get a friendly message.
			exit(1);  // or throw something else (this would be the C++
way to "gracefully" report things
			break;
	}
	return num_bytes;
}

//main function
int main(int argc, char **argv)
try{

	//intialize ip address and port
	const char host_ip_str[] = "130.215.18.39";
	const int port = 8080;
```

Appendix A: External Applet Code Continued

```
      fstream filestr;
      filestr.open ("test.txt", fstream::in | fstream::out | fstream::app);
    filestr.close();

  // Initialize Winsock
  WSADATA wsaData;
  if (WSAStartup(MAKEWORD(2,2), &wsaData) != NO_ERROR)
  {
    printf("Error at WSAStartup()\n");
    return 1;
  }

  // Create a SOCKET for connecting to server
  SOCKET ConnectSocket = socket(AF_INET, SOCK_STREAM, IPPROTO_TCP);
  if (ConnectSocket == INVALID_SOCKET) {
    printf("Error at socket(): %ld\n", WSAGetLastError());
    WSACleanup();
    return 1;
  }

  // The sockaddr_in structure specifies the address family,
  // IP address, and port of the server to be connected to.
  sockaddr_in clientService;
  clientService.sin_family = AF_INET;
  clientService.sin_addr.s_addr = inet_addr( host_ip_str );
  clientService.sin_port = htons( port );

  // Connect to server.
  if ( connect( ConnectSocket, (SOCKADDR*) &clientService,
sizeof(clientService) ) == SOCKET_ERROR) {
    printf( "Failed to connect.\n" );
    WSACleanup();
    return 1;
  }

  printf("Connected to server.\n");

  //Initialize variables for the tag operations

  char data[24]; // input tag data
  data[0]; //set it to the first address
  char type[4]; //tag type
  type[0]; //set it to the first address
  int choice; //menu input
  char c; //continue? variable

  //Initialize variable for logging time
  time_t rawtime; //time
  struct tm * timeinfo;
  time ( &rawtime );
  timeinfo = localtime ( &rawtime );
```

```
  //output time
  printf ( "Current local time and date: %s", asctime (timeinfo) );

//create a while loop to loop until the user exits
  while(d!='n')
  {

 //menu
  printf("1. Read: Enter 1 \n");
  printf("2. Look Up tag: Enter 2\n");
  printf("3. Write Tags: Enter 3\n");
  printf("4. Kill tag: Enter 4\n");
  printf("5. Lock Tag: Enter 5\n");
  printf("6. Exit: Enter 6\n");
  printf("Enter Choice: ");
  scanf("%d", &choice);

 //initialize variable for the return value of the send command
  int bytes;
  char recv_buf[65536];
  char tag;
  char tag_end;
  unsigned tag_len;


 //initilize string for the send command and to append more string to it
  string str_msg_write = "UPDATE tag_id SET id=0x";
  string str_msg_look= "SELECT id FROM tag_id WHERE protocol_id='";
  string str_msg_kill= "UPDATE tag_id SET killed=1, id =0x";
  string str_msg_password_EPC1 = "UPDATE tag_id SET password=0x88, id=0x";
  string str_msg_password_GEN2 = "UPDATE tag_id SET password=0x33445566,
id

 //  alize tag type and password
  char password_EPC1[2];
  char password_GEN2[6];
  password_GEN2[0];
  c   EPC1[]= "EPC1";
  c   GEN2[] = "GEN2";
  c   password1[] = "88";
  c   password2[] = "33445566";


//  ting different cases
  switch (choice)
  {

 /
  c   :
        send cmd(ConnectSocket,"SELECT read_count,protocol_id,antenna_id,id
FR      _id WHERE (protocol_id='GEN2' OR protocol_id='EPC0' OR
pr     ol_id='EPC1') set time_out=1000;\n");
         recv buf[0];//for paranoia
         tag = recv_buf;
        tag_end = strchr(recv_buf, '\n');
         g_len=tag_end-tag;
```

```
        nbytes = get_data(ConnectSocket, recv_buf, 65536-1);//65536-1 because
we want to guarantee the last byte==0 (NULL)
        recv_buf[nbytes]=0; //make sure the string is null-terminated, so now
the string is C-style
        printf("\n%.*s", nbytes, recv_buf);
      break;

//look for tags
  case 2:
     //prompt user for tag type
        printf("Enter Protocol ID type: ");
      // get tag type
        scanf("%s", type);
     //append type
        str_msg_look.append(type);
     //append RQL code
        str_msg_look.append("' AND antenna_id=1;\n");
      //send command to Mercury4
        send_cmd(ConnectSocket, str_msg_look.c_str());
      //output data from mercury4
        recv_buf[0];//for paranoia
        tag = recv_buf;
      tag_end = strchr(recv_buf, '\n');
      tag_len=tag_end-tag;
        nbytes = get_data(ConnectSocket, recv_buf, 65536-1);//65536-1 because
we want to guarantee the last byte==0 (NULL)
        recv_buf[nbytes]=0; //make sure the string is null-terminated, so now
the string is C-style
        printf("\n%.*s", nbytes, recv_buf);
      break;

//write
  case 3:
     //prompt user for tag data
        printf("Enter Data to be written on the tag: \n");
      //get tag data
        scanf("%s", data);
     //prompt user for tag type
        printf("Enter tag type: \n");
      //get tag type
        scanf("%s", type);
      //append RQL commands
        str_msg_write.append(data);
        str_msg_write.append(" WHERE protocol_id='");
        str_msg_write.append(type);
        str_msg_write.append("' AND antenna_id=1;");
      //send RQL commands to mercury4
        send_cmd(ConnectSocket, str_msg_write.c_str());

      break;

 // [illegible]
  case 4:
      //prompt user for tags to be killed
        printf("Enter tag id to be Killed: \n");
      //get [illegible]
        scanf("%s", &data);
```

```
      //prompt user for tag type
        printf("Enter tag type: ");
      //get tag type
        scanf("%s", type);

    //compare if the user inputed EPC1 tag
        if(strcmp(type, EPC1) == 0)
        {
            //append RQL password code
              str_msg_password_EPC1.append(data);
              str_msg_password_EPC1.append(" WHERE protocol_id='EPC1' AND
antenna_id=1;");
            //prompt for password
              printf("\nEnter Password: \n");
            //get password
              scanf("%s", &password_EPC1);
            //check if the password is correct
              if(strcmp(password_EPC1, password1) == 0)
              {
                  //send Mercury4 the password
                    send_cmd(ConnectSocket, str_msg_password_EPC1.c_str());
                  //append kill command
                    str_msg_kill.append(data);
                    str_msg_kill.append(", password=0x88, WHERE
protocol_id='EPC1' AND antenna_id=1;");
                  //send kill command to mercury4
                    send_cmd(ConnectSocket, str_msg_kill.c_str());
              }

            //display that the user inputted the wrong password
              else
              {
                //output message
                  printf("\nInvalid Password\n");
              }
        }

      //compare if the tag is a gen2
        if(strcmp(type, GEN2) == 0)
        {
            //append password commands
              str_msg_password_GEN2.append(data);
              str_msg_password_GEN2.append(", WHERE protocol_id='GEN2' AND
antenna_id='1' AND type=0;");
            //prompt user for password
              printf("\nEnter Password: \n");
            //get password
              scanf("%s", &password_GEN2);

            //check if the inputted password is correct
              if(strcmp(password_GEN2, password2) == 0)
              {
                  //send password to mercury4
                    send_cmd(ConnectSocket, str_msg_password_GEN2.c_str());
                  //append kill code
                    str_msg_kill.append(data);
```

```
                    str_msg_kill.append(", password=0x33445566, WHERE
protocol_id='GEN2' AND antenna_id=1;");
                    //send kill code to mercury4
                    send_cmd(ConnectSocket, str_msg_kill.c_str());
              }

            //display invalid password to the user
              else
              {
               //output message
                  printf("\nInvalid Password\n");
              }

        }

      //if the wrong type then output invalid type
        else
        {
            //output message
              printf("\nInvalid tag type. Please Enter EPC1 or GEN2\n");

        }
  break;

  //lock tag


 //exit
  case 6:
        d = 'n';
  break;

 //defaul message
  default: printf("\nInvalid Entry\n");
  break;
  }
 }


//close connection
      //int closesocket(SOCKET the_sock);


      //memchr
 // WSACleanup();

  return 0;


}

//catch error
catch(const exception &e){
      cerr << e.what();
      return 9;
}
catch(...){
```

```
        fprintf(stderr, "Caught unknown exception.\n");
        return 10;
}
```