CHAPTER 1
INTRODUCTION

1.1 Background of Study

Hospitals are the most complex of building types. Each hospital is comprised of a wide range of services and functional units. These include diagnostic and treatment functions such as clinical laboratories, imaging, emergency rooms and surgery; hospitality functions, such as food service, and housekeeping; and the fundamental inpatient care or bed-related functions. This diversity is reflected in the breadth and specificity of regulations, codes, and oversight that govern hospital construction and operations. Each of the wide-ranging and constantly evolving functions of a hospital, including highly complicated mechanical, electrical, and telecommunications systems, requires specialized knowledge and expertise. Despite its popularity and reputation, very often mistakes and accidents can easily occurs due to human errors. It is no secret that the hospital system is still primitive. The scope of this project is to improve hospital system using Radio Frequency Identification (RFID). RFID is a technology that uses radio frequency to uniquely identify people and items by assigning to them a personal ID that is identifiable and detectable remotely.

`1.2 Problem Statement

The findings of a study on wandering done by Meredith Rowe RN, PhD professor at University of Florida College of Nursing and Institute of Aging were reported. This study revealed that less than 4% of those who wander away are able to fund their way home without assistance. Law enforcement agencies were found to delay looking for adults thinking that they would return on their own. In 675 missing reported nationwide to Alzheimer's Association's safe Return over 13 months, patterns were identified. Finding included: 49 % were found between 1 and 5 miles from where last seen, 37% found less than one mile from home, 14% found more than 5 miles away, more ¼ found in residential yards, 22% found standing in middle of streets, 11.8% shopping centers, 3.7% standing on sidewalks, 3.4% along highways, 3.1% convenience stores, 3.1% in restaurants, 1.8% in remote natural areas, 1.6% in parking lots. 1.6% in food stores, 1% in banks, 0.8% along railroad trucks, 0.8% in senior care centers, and all found in remote areas were dead.

A Lancet May 1994 article explained "... wandering is a distinctive and difficult feature of dementia. The patient sets off to tour the house, the neighborhood, or the city, without apparent purpose. Unlike stray pets, they do not usually return home. Every geriatric specialist can tell horror stories of patients crossing highways and walking along railway tracks. Patients often come to grief in road traffic accidents; some continue to drive a car and run someone else over."

It is clear that the searching system currently in use has a lot of drawbacks. It fully depends on worker's integrity to have got the correct patient's information. Besides, they whenever there is an emergency in the hospital building (fire, natural disasters), this will lead to problem of time and work consuming to count patients in the hospital once they are being assembly at the assembly area. If a patient is discovered missing, this requires time and finding the particular victim without knowing the exact location of the person.

Analyzing the operational safety and economic challenges, these problems are:

- There is no proper monitoring system for Alzheimer patients and young patients where this often leads to their physical conditions could find their way back to their bed in hospital.
- Staff frustration with operational inefficiencies
- During emergency like fire or natural disasters, everyone is require to gather in a specific assembly area and conventional way of head counting is used to check and identify each patient in hospital.
- No efficient way of emergency search in the hospital.

1.3 Objectives

The objective of this project is to build an efficient and workable hospital system that is more reliable and more efficient that the one currently in use. Making use of RFID technology, it is possible to build an automated system that can considerably reduce the chance of improving the hospital managing system.

- To design a monitoring system for Alzheimer and helpless patients using RFID technology.
- To design a temperature measurement system using RFID technology.
- To design an emergency head counting system using RFID technology to increase the current system reliability and reduce processing time in searching victims.

1.4 Scope of Study

The scope of this project will be as follow:

- Conduct research on system components including hardware and software and identify their limitation. The hardware components more adapted to the system are RFID readers (short range and long range readers), RFID tags, standard field generator and motion sensor. This part of work also involves finding out how to configure the hardware (especially the field generator) so that it generates a field capable of covering the appropriate area.
- Create subsystem prototypes and conduct testing. This is done to ensure that all the features of the system can be built and operate as expected.

- The scope of study will comprise first, **communication system**. This is to understand more on radio frequency, and the interference around. Secondly, **microelectronics** is involved in order to study on the circuit and configurations on the RFID readers, field generators and tags. Lastly, there will be some **programming** works. This is needed to build the database of the system.

CHAPTER 2
LITERATURE REVIEW

Basically, most of the technical information about FRID is brought from few well written books entitled RFID Handbook – Applications Technology Security and Privacy by Shey Ahson and Mohammad Ilyas (Mar 2008) , RFID Techology and Application by Stephen B.Miles, Samjay E.Sarma and John R.Williams (JUN 2008), and RFID – a Guide to Radio Frequency Identification by John Wiley & sons (2007)

Those books are intended for students and engineers who could be confronted with RFID technology for the first time. The book's basic chapters describe the functionality and the physical and IT-related principles of RFID technology. Another good resource for this thesis was from RFID organizations and manufacturers. Few websites were used as resources. These websites are owned and managed by people and companies with huge RFID expertise and qualifications. Activewave, Microchip and RFID journal . These sites consist of useful collection of RFID news, articles, case studies, vendors, products, etc.

2.1 System Communication

Typical communication procedure between and transponder and the reader can be highlighted and refer to figure 2.1 as follows

Handshake:

1. The reader sends a command to start communication tag in the interrogator field and also to power it (active tags).
2. Once the tag has received signal and command from the reader, it replies with its ID for acknowledgment.
3. The reader now knows which tag is in the field and sends a command to the identified tag for instructions either for processing (read or write) or Sleep.

Data exchange:

4. If the tag (transponder) receives processing and reading commands, it transmits a specified block data and waits for the next command
5. If the tag receives processing and writing commands along with block data, it writes the block data into the specified memory block, and transmits the written block data for verification.

Termination:

6. After the processing, the interrogator sends an End command to send the tag into the Sleep ("silent") mode.
7. If the device receives an End command after processing, it sends an acknowledgement and stays in Sleep mode. During the Sleep mode, the device remains in non-modulating (detuned) condition as long as it remains in the power-up.

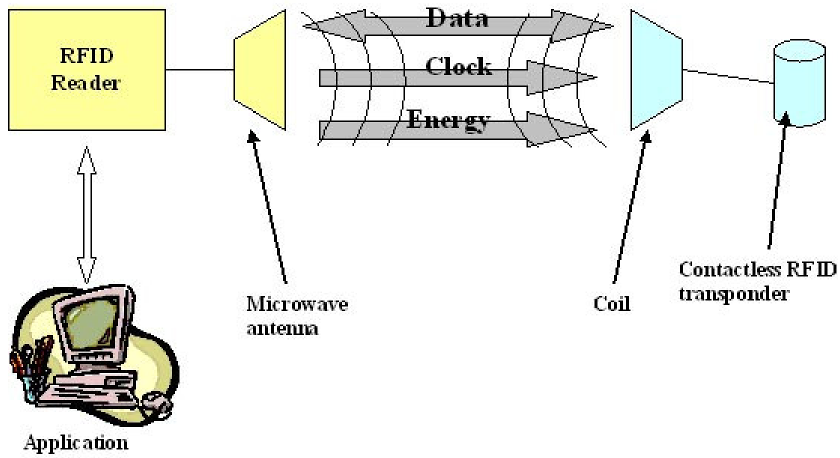


Figure 2.1 – RFID System components

2.2 Basic Theory

Basic RFID operations

The hardware components of the RFID system include readers with built-in field generators. The field generator creates a field. Whenever a person holding a tag enters the field, the tag that is normally asleep wakes up and transmits its information to the reader. This information includes the tag's ID, the strength of the signal transmitted from the reader to the tag, and some additional information out of the scope of this project. Besides the reader, its built-in field generator and the tag as basic components, sometimes additional components are needed: a standard field generator is normally used to boost the field range when the built-in field generator cannot cover up a specific area of work. Sometimes, standard field generators can be associated with motion sensors. In this case, the motion sensor activates the field generator to send a broadcast signal. Whenever a tag is within the field's range, it wakes up and sends its information to the reader.

Head-count using RFID

It is possible to turn the basic use of RFID into a head-count system simulating an actual searching system. In this case, each patient should hold a tag. The tag has the size of a small box attached to hand with strip. That makes it easy to carry. The tag is configured in a way that its ID is the same as the ID of its holder. The challenge would be to create a program that would be able to retrieve the tag ID, retrieved from database the information corresponding to the ID, and set up the status of the patients in the hospital. (NOTE: this system apply to workers in hospital as well)

2.3 Operation Principle

RFID communication basically involves a two ways radio frequency communication process between transceiver (reader) (refer to appendix 1) and transponder (tags) using wireless air interface. The principles of RFID operation can be classified into two forms which are reactive/inductive coupling and coupling by propagation of electromagnetic waves [Monahan 2006].

2.3.1 Induction Coupling and Load Modeling

Inductive Coupling is the transfer of energy from one circuit to another through a shared magnetic field. An electrical current passing through the coil of a primary conductor creates a magnetic field that induces an electrical current in the coil of a secondary conductor exposed to the magnetic field. The coupling between the reader and tags is inductive for low and high frequencies (typically 13.56 Mhz) [Delia A.C.]. In inductive coupling, predominantly data in the communication of tags and readers is carried by magnetic field causing coupling to occurs (between primary and secondary coils) in an air-cored transformer. Electric current is induced in the tag's antenna, which is used to power the integrated circuit and obtain the ID. The data transfer operation between transponder and reader is operated using load modulation. According to *RFID Design Principle by Harvey Lehpamer*, in load modulation, the carrier signal is modulated by switching impedance from a matched condition to an unmatched condition to alter the reflection coefficient. Inductive coupling is usually applied in near field communication system.

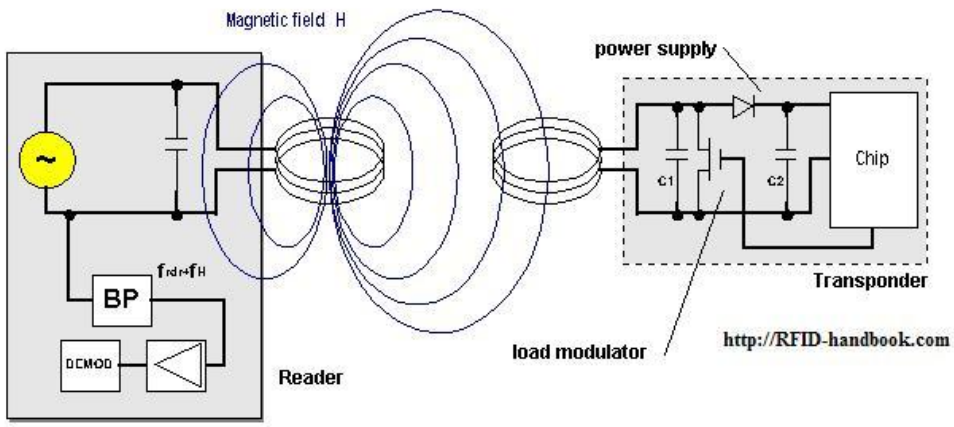


Figure 2.2: Load modulation circuitry

2.3.2 Propagation Coupling and Backscatter

Propagation coupling involves ultra-high frequency (UHF). Using electromagnetic field to read and interrogate tags, these system field components dissociates from their source in the reader and propagate into free space[M. Buettner 2008]. Backscatter modulation is a form of communication method used by transponder to transmit data to transceiver. In other terms, backscattering is also defined as reflected power. This is because, the incident wave radiates by transceiver will be absorbed (part of it), then the rest will be reradiated as a backscatter wave. Propagation modulation and backscatter are usually used in far field communication system.

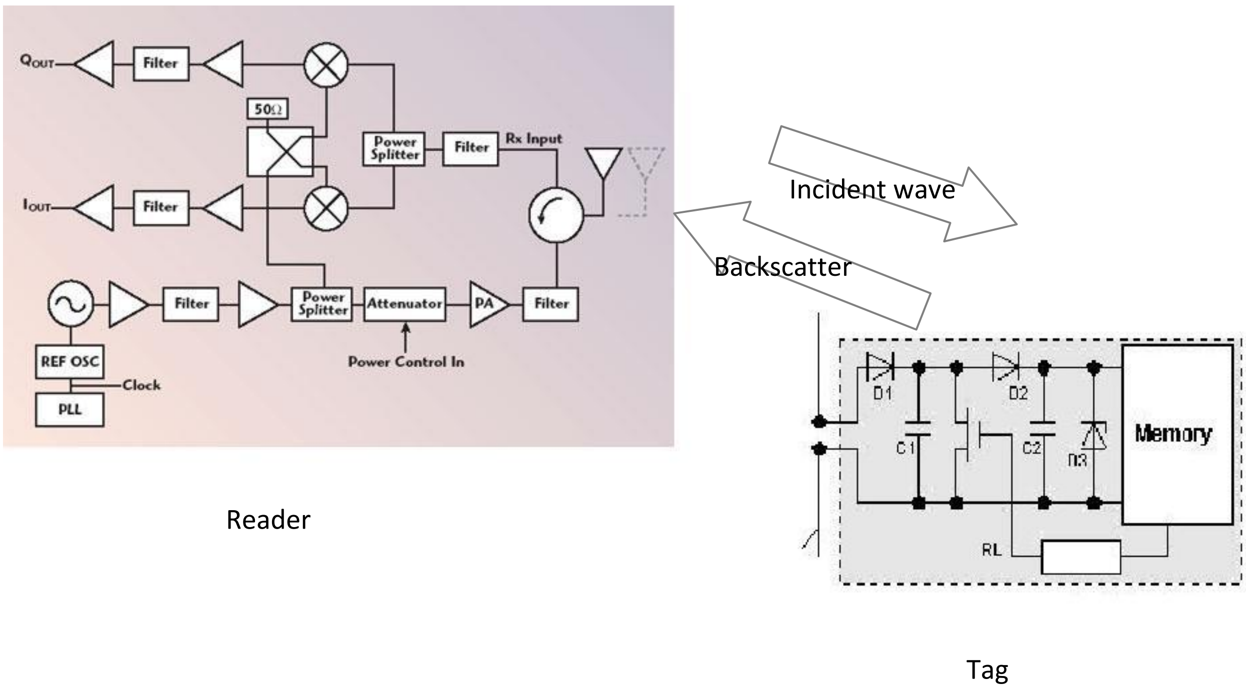


Figure 2.3: Backscatter modulation circuitry

2.4 Passive and Active RFID

RFID technology is divided into two main categories which are active RFID and passive RFID. Whilst the real difference between both of them is that active RFID tags have an onboard battery which passive one doesn't have, there are many more differences in term of its capability and design which will be discussed in discussion.

2.4.1 Passive RFID

Passive tags only depend on the power emitted by the reader for data processing and data transmission. This kind of transponders must have an RF front end, an analog and digital circuit (depends on the data processing techniques). Because of its limited power supply, it has a limited transmission and limited range of detection, which are only a few meters. Similarly, passive tags can only contain a small amount of data, mostly ID's. Since the data processing abilities are limited with memory block of few kilobytes, passive tags are very hard to be programmed. The advantage of passive tag is that it is very cheap and has low maintenance. It is usually applied in environment which movement of the tagged assets is highly consistent, for example in medical, supply chain and inventory tracking.

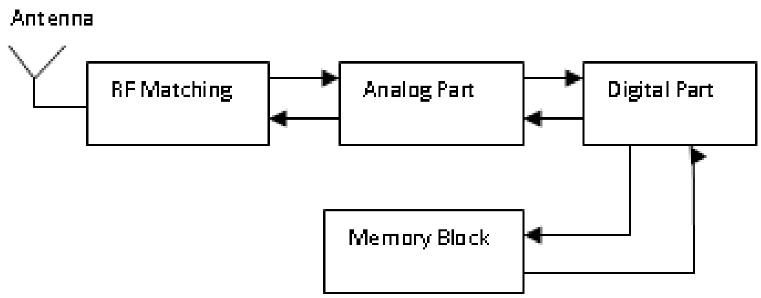


Figure 2.4: Block Diagram of Passive RFID Tag

2.4.1.1 Passive Application

In today's world, passive RFID has been used widely in industries. The applications of passive RFID are so wide and it is used more often that active RFID. Some of the applications for passive RFID are:

1. Passports
   - The first E-passport was issued by Malaysia in 1998. In addition to information also contained on the visual data page of the passport, Malaysian e-passports record the travel history (time, date, and place) of entries and exits from the country.
   - Standards for RFID passports are determined by the International Civil Aviation Organization (ICAO)
   - Fingerprints and biometrics data are also included in the passports in some countries.

2. Product tracking
   - As a replacement for barcode tags by Canadian Cattle Identification Agency to identify a bovine's herd of origin and this is used for tracing when a packing plant condemns a carcass.
   - Used in library book or bookstore tracking, jewellery tracking, pallet tracking, building access control, airline baggage tracking, and apparel and pharmaceutical items tracking. High-frequency tags are widely used in identification badges, replacing earlier magnetic stripe cards.

3. Inventory System
   - Provides an accurate knowledge of the current inventory. In an academic study performed at Wal-Mart, RFID reduced Out-of-Stocks by 30 percent for products selling between 0.1 and 15 units a day.
   - In 2004, Boeing integrated the use of RFID technology to help reduce maintenance and inventory costs and was able to save $29000 USD in labour alone after six months of integration.

2.4.2 Active RFID

Active tags have an onboard power supply (refer to appendix 2), which is used to amplify signal from the reader and then transmitting data back to the reader. Because of this, it has longer reading range (up to 100 feet) and doesn't need the RF carrier signal to energize the data processing section [Yanna Hao 2007]. Active tags can hold more data and have the ability to store data from the power consumptions and longer life, sleep mode is introduced. The tag that doesn't go through interrogation mode will stay in sleep mode, thus reserving power. Active tags came in bigger size than the passive one and are more expensive. The main advantage is that active tags can be programmed, thus can be used on various items repetitively. It is usually applied to tag variable movement assets, more sophisticated security and sensing.

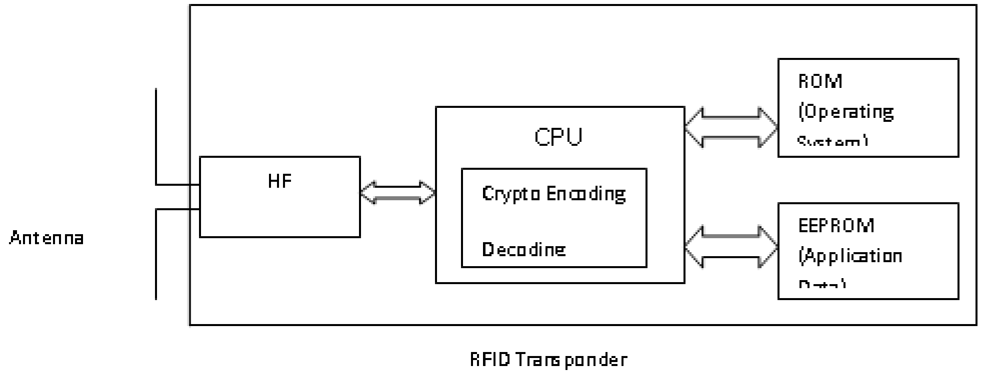


Figure 2.5: Block Diagram of Active RFID Tag

Sleep mode is introduced in order to optimize the energy usage for active RFID. According to Bjorn Nilsson in *Towards Energy Efficient Protocol for Active RFID*, in the optimal protocol there would be no energy loss for the tag to detect an RFID-reader and wake up from sleep-mode. The only energy that this optimized tag uses is when transmitting information to the RFID-reader and receiving a confirmation that the RFID reader has successfully retrieved the tag information. After the received acknowledge message the tag enters deep sleep-mode. The tag stays in this mode for a predetermined time, specified in the acknowledge message from the RFID-reader. The flow chart below shows the typical state operation for a tag executing the optimal protocol. The power consumption in deep-sleep- and sleep-mode for all described protocols is much less than in the wake-modes where the tag is receiving and transmitting. Of great importance is then, of course, the duty cycle between wake- and sleep-mode.

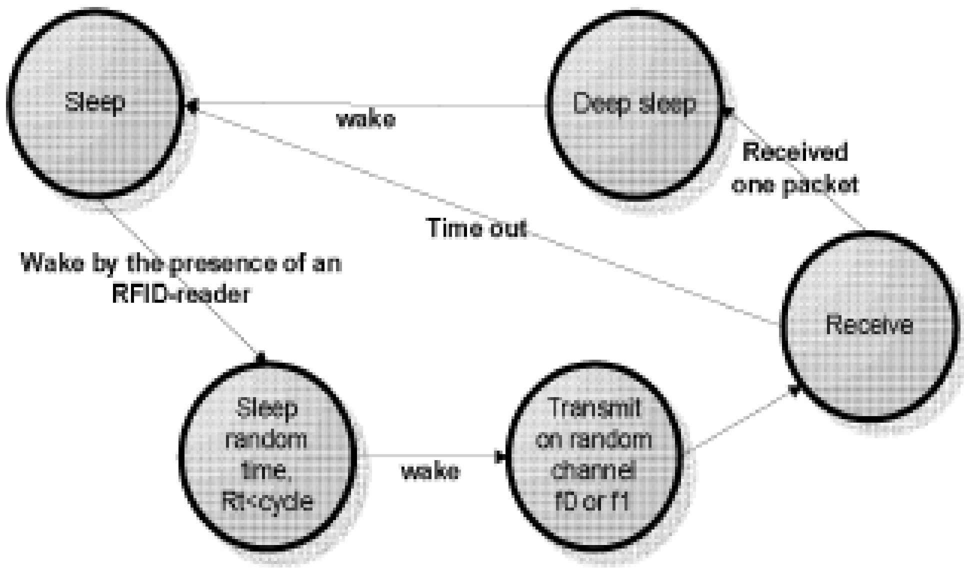


Figure 2.6: State Diagram of a Tag Executing the Optimal Protocol

2.4.2.1 Active Application

Although active RFID are more costly than the passive one, it has its own unique applications in industries which passive RFID can't handle. Some of active RFID applications are:

1. Asset tracking and management
   - Mobile computers, with integrated RFID readers, can deliver a complete set of tools that eliminate paperwork, give positive proof of identification and efficient searching system.
   - Web based management tools allow organizations to monitor their assets and make management decisions from anywhere in the world, giving real time information of their valuable assets.

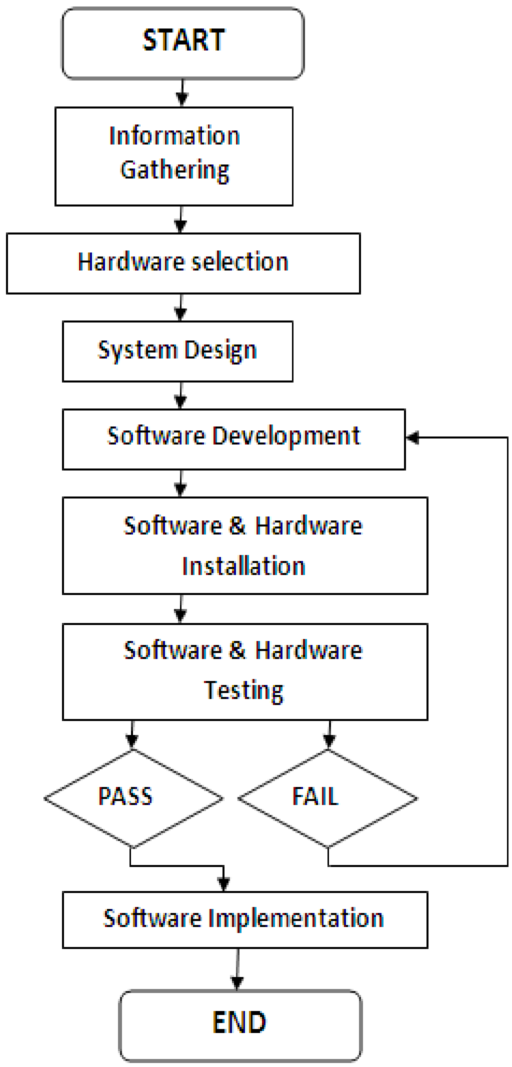
2. Headcount system

- An emergency headcount system to track and identify missing individuals during emergencies in plants. Applied in PASB, Kerteh.
- Improve old headcount system in many ways, thus reducing casualties and lost of lives.

CHAPTER 3

METHODOLOGY

3.1 Implementation Procedure



- First of all, all beneficial information and data are collected from sources in order to help understand more on RFID and its implementation.
- Several alternatives hardware is gathered and by using Delphi Decision Matrix, a final hardware will be selected.
- Designs for patients/staffs/inventories are designed.
- System software is then developed to match application. Hardware weaknesses are identified and improvement will be made.
- Software and hardware are then installed around the hospitals, based on the design created.
- Repetitive testing will be made on both hardware and software for each system designed. The most optimized system will then be selected to be implemented.
- The system that is working well is then implemented in real hospital, as a pilot project.

Figure 3.1: Implementation Procedure

3.2 Hardware Selection and Evaluation

As discussed earlier, the main objective of this project is to produce a system that is capable of detect tags and getting data from tags. The system must be non-time consuming and able to detect the location and time of any tag. Below are four hardware recommendations for this system, both with slight difference in their operation. Each hardware will be evaluated repetitively so that the system desired will be achieved.

3.2.1 Barcodes

The problem with the bar code is that the maximum throughput in any bar code system is one: that is, you can scan only one object at a time. In addition, because a limited amount of data is stored in a small form factor, the bar code doesn't have enough room for a unique serial number, expiration date, or other pertinent information. Lastly, the bar code reader has to be able to "see" the bar code marking to read it. For example, if a bar coded item is wrapped, packaged in a container, kept under a sheet or cover, or has somehow gotten dirty, dusty, or marked, the bar code can't be read.

Because of these limitations, most bar code innovations in the past few years have focused on data-capture and data-transmission devices to make bar codes more useful and to help them keep up with faster computing power and better network connectivity. This section explains the different bar code Systems available — the old linear bar code, the stacked bar code, and matrix symbols — and clarifies how they stack up.

Linear bar codes are the most widely used Auto-ID system. They can be found on everything from cans of soda to rental cars. They are formed by printing a series of alternating dark and light (usually white) bars of varying width. These patterns have very specific meanings and representations. The other component of an Auto-ID solution, the reader or scanner, is of course a key part of a linear bar code system. Many types of scanners can read linear bar codes. Fixed-location scanners can be used to read linear bar codes without significant operator intervention if there is a method to ensure that the label faces the scanner. In terms of the criteria discussed earlier, linear bar codes offer the following:

Modification of data: After a bar code is printed, it's done. You can't change the orientation of the markings after the symbol has been printed or etched.

Security of data: Linear bar codes are widely adopted, and the standards are well known; however, they are not encrypted for security.

Amount of data: Linear bar codes can have up to 30 characters of data.

Life span: Life span is fairly low because they are usually printed. However, if they are etched, they can last a very long time.

Reading distance: Linear bar codes require line of sight to be read and have a range of a few feet.

Number that can be read at a time: Only one item can be scanned at a time.

Potential interference: Linear bar codes become unreadable when vertical damage occurs. Such damage occurs when a black bar is completely eliminated or altered or when a white bar is filled in. In the event of vertical damage to the symbol, there is typically no possibility of recovering the data. Only one bar code symbology (93*i*) contains erasure and error correction capabilities. The symbol also becomes unreadable if obscured by dirt or other contaminants or when severely abraded. In addition to the bar code being susceptible to dirt and dust, the readers also cease to function if dirt, dust, or other foreign objects obstruct the lens.

Another type of bar code is the *stacked bar code* (also called a *2-D bar code*). From a technology perspective, a stacked bar code comprises multiple rows of very short linear bar codes, arranged in a specific manner to ensure correct decoding. Although several stacked bar code symbologies are available, only one is commonly used: PDF 417. The stacked bar code is very similar to the linear bar code, with the exception of the following key differences:

Security of data: Because they lack the vertical redundancy of linear bar codes, stacked bar code symbologies employ a specification called *Reed-Solomon erasure and error correction,* which allows part of the tag to be destroyed while retaining all the original information. Data compaction schemes as well as encryption help to increase data capacity and enhance data security. Because it is a line-of-sight technology that carries more data than a simple linear bar code, additional security concerns exist. For instance, a PDF bar code can be photocopied, scanned or faxed and subsequently read, making counterfeiting and theft very simple — a continuing problem with the bar codes used on tickets for sporting events.

Amount of data: The stacked bar code is the only bar code on which a significant amount of storage can be added right to the tag. Stacked bar code symbols can contain more data than linear bar code symbols — up to a full kilobyte.

Potential interference: Although they are more tolerant of localized damage than linear bar codes, significant amounts of obscuring material or abrasion can still render them unreadable in spite of their errorcorrection capabilities

Matrix symbols are yet a third type of bar code. They're composed of discrete modules (typically round or square) arranged in a grid pattern. In the United States, the most widely known examples of a matrix symbol are the codes that the U.S. Postal Service prints on letters and postcards in order to sort the mail. Matrix symbols share many characteristics with the linear bar code, but they do have some unique traits that make they better suited for specific applications:

Security and amount of data: In these areas, matrix symbols have the same capabilities as stacked bar code symbologies and are roughly equivalent in data capacity and error correction.

3.2.2 Contact Memory Buttons

Contact memory buttons have also been around for nearly a generation. They are a specific type of Auto-ID that requires a wand to make physical contact with a button tag to read the data on the tag. Each button tag is about the size of a quarter. Given the limited adoption of contact memory button technology, comparatively little investment and innovation is occurring in this arena. Because contact memory will never be a widespread Auto-ID solution, a key concern surrounding this technology is that the three major contact memory button solutions in use today are proprietary systems. If those solutions are discontinued, finding a replacement may prove difficult. But as you can see from some of the key attributes, contact memory does have some distinct advantages.

Modification of data: Contact memory buttons can be written to and read many times. They are robust because they can withstand vibration and harsh environments and still can be read.

Security of data: Contact memory buttons can have their data encrypted.

Amount of data: Data storage can be up to 8MB.

Life span: The physical contact required for communication with the reader limits the usable life of that reader.

Reading distance: Because the tag reader has to come in physical contact with the button tag, the reading distance is essentially zero.

Number that can be read at a time: You can read these only one at a time.

Potential interference: The physical contact required also limits the efficiency with which the contact memory button can be read.

3.2.3 RFID

An RFID solution uses a radio frequency (RF) signal to broadcast the data captured and maintained in an RFID chip. An RFID system is composed of three components: a programmable transponder or tag, a reader (with an antenna), and a host.

An active tag has its own battery power to contact the reader. Power from the battery is used to run the microchip's circuitry and to broadcast a signal to a reader. An active tag's onboard power source enables the tag to broadcast a signal out at great range by either constantly beaconing a signal or broadcasting only when the reader talks first. Some of the more powerful active tags can communicate up to kilometer. Active tags are much larger and therefore can carry a lot more memory capability. Rather than simply having a unique serial number on the tag, like a passive tag, active tags often carry information such as the full contents of a container, its destination, and its origin. By carrying all the information on the tag, you can retrieve information instantly. For example, soldiers in battle usually can't look up a file associated with a tag on the Internet, so soldiers use hand-held units to scan containers with active tags to find out what's inside. Despite their cost, active tags have proven a significant return on investment (ROI) for many applications. Since the early 1990s, the DoD has put active tags, about the size of a cigarette carton, on containers to track both their contents and their whereabouts.

A passive tag does not require a battery. Rather, a passive tag derives its power from the electromagnetic field created by the signal from the RFID reader to respond to the reader with its information. Because the mandates generally require passive tags, Please refer back to topic 2.3 for detail explanation on how they work.

Modification of data: The ability to modify data depends on the standard that you use. Using the electronic product code standard (EPC), the two classes are

• *Class 0 tags:* These are read-only, which means you have to use the number that the manufacturer writes on the tags.

• *Class 1 tags:* These are read/write tags, which means you can program whatever number you want on them (often called *commissioning* the tag) at your place of business and then read them an infinite number of times — write once, read many (WORM).

Security of data: Depending on the class and generation of the RFID tag, they have the ability to be encrypted so that others with standard RFID readers cannot read the actual data on the tag. For more details about the security of data in RFID systems, see the section, "To EPC or Not to Be: Unraveling the Words, Words, Words of the Electronic Product Code," later in this chapter.

Life span: Having no need for a battery makes the passive tag's life virtually unlimited. Active tags and semi passive tags last as long as their batteries. Refer to Table 3-1 for more details.

Size: Passive tags range in size from Hitachi's mu-chip (μ-chip; about the size of a pin head) to the size of a letter envelope. Active tags range in size from the size of a piece of hard candy to about the size of a carton of cigarettes. The larger the size, the fewer items the tag can easily be affixed to. The smaller the size, the less the read distance.

Reading distance: Passive tags communicate in ranges from a few millimeters (called the *near field*) all the way out to tens of meters. Active tags can communicate more than 100 meters. The big benefit, as I mention earlier, is that you don't have to see a tag to read it. Tags inside containers, behind walls, in briefcases, and so forth can still be read.

Number that can be read at a time: A reader can read hundreds of tags nearly simultaneously.

Potential interference: Various materials such as metals and liquids can interfere with passive tags. Active tags are less susceptible to interference but s till can have issues inside metal containers.

Table 3.1 Comparing the Primary Auto-ID Technologies

	Bar codes	**Contact memory**	**Passive RFID**	**Active RFID**
Modification	Unmodifiable	Modifiable	Modifiable	Modifiable
Security of Data	Minimal security	Highly secure	Ranges from minimal to highly secure	Highly Secure
Amount of data	Linear bar codes can hold 8-30 characters; other 2-D bar codes hold up to 7,200 numbers	Up to 8MB	Up to 64KB	Up to 8MB
Life Span	Short unless laser-etched into metal	Long	Indefinite	3-5 year battery life
Reading distance	Line of sight (3-5 feet)	Contact required	No contact or line of sight	No contact or line of sight; distance up to 100 meters and beyond
Potential interference	Optical barriers such as dirt or objects placed between tag and reader	Contact blockage	Environments or fields that affect transmission of radio frequency	Limited barriers since the broadcast signal from the tag is strong

3.3 Hardware Justification

For hardware justification, comparing of passive and active tag is done through Delphi decision matrix. The important factors considered are:

1. Modification of data
   - Modification of data is an important factor because the tags will probably be used as a matrix card to provide/store patients, workers and inventories ID. It is crucial that the tag can modify with a secure way.

2. Security of data
   - Security is an important factor because the tags should not be change easily by anyone due to patient's health history that stored in the tag play a very crucial role in the system.

3. Amount of data can be stored
   - The amount of data is another major role in hardware selection. A better storage will provide a better information details in the system.

4. Reading distance
   - For this system, the tag chose needs to have the ability to be detected / read from long distance. This is for the ease of detection and the fluency of the system.

5. Life span
   - Every tag is expected to reuse for at least 1-2 years.

6. Potential of interference
   - Hardware should have the minimum of interference during the usage because carriers are expected to all around the hospital.

Table 3.2: Delphi Decision Matrix on Hardware Justification.

Factor	Weight	System			
		Bar codes	**Contact memory**	**Passive RFID**	**Active RFID**
Modification Data	2	3	8	8	8
Security Data	2	3	8	8	8
Amount Of Data Stored	1	3	8	6	8
Reading Distance	2	3	5	7	9
Life Span	1	5	8	9	8
Potential of Interference	2	3	5	7	9
	TOTAL	20	42	45	50

3.4 System Designs

As discussed earlier, another main objective of this project is to produce a system that is to locate tags efficiently during normal and emergency time. The system must be non- time consuming and able to detect the location and time of any tag. Below are three designs for this system, 3 of them come with slight difference in their operation and each system will be tested repetitively so that the system desired will be achieved.

3.4.1 System for Patients Search

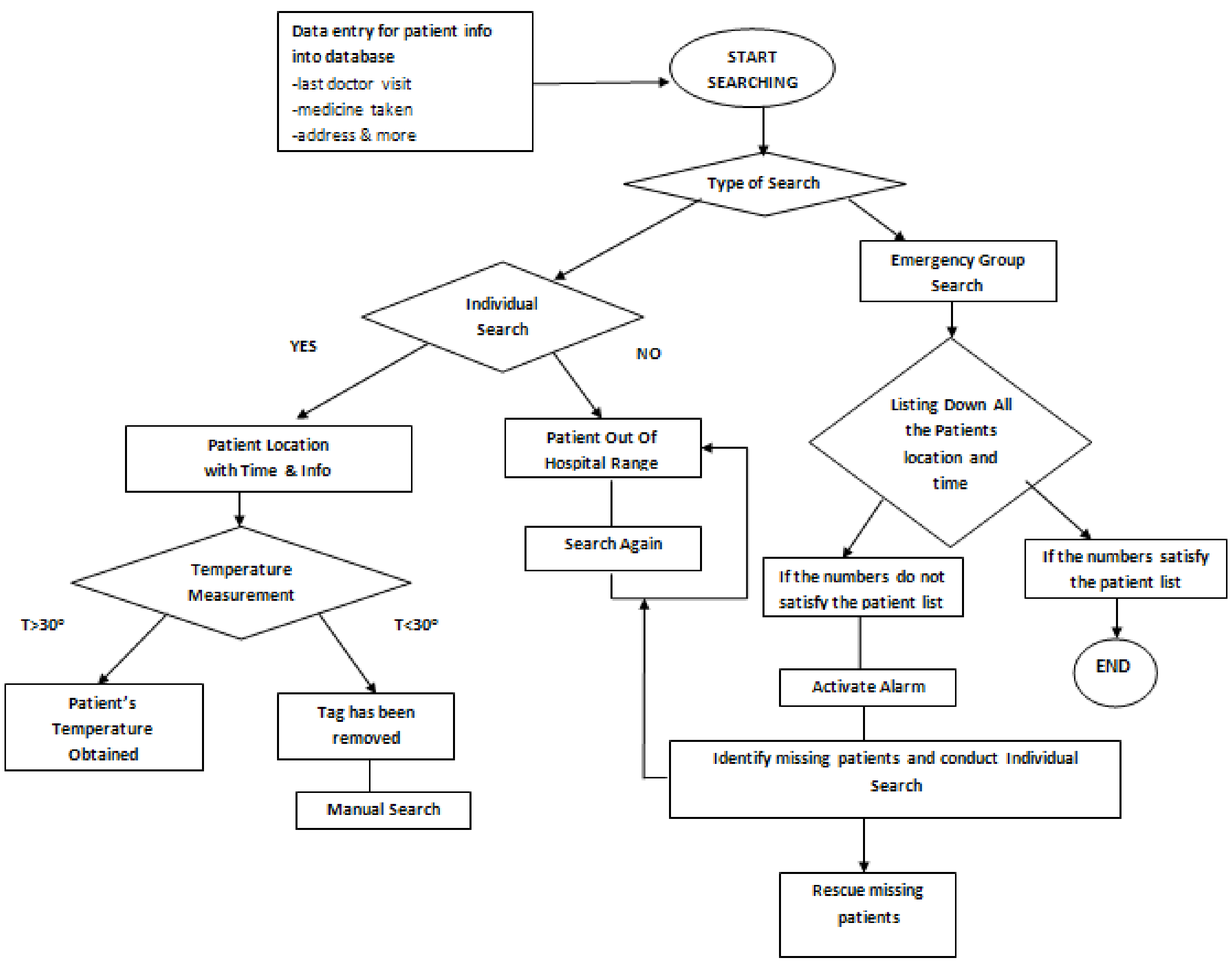


Figure 3.2: RFID System for patients

- Firstly, patient's info will put in to the databases.
- User can choose whether to individual search or emergency group search.
- For individual search, tag id will be inserted to search and if tag is detected, it will tell the location of patient with its current time and info like last visit by doctor and last medicine taken.
- Once tag is detected, patient temperature can be measure as well.
- If measured temperature is less than 30^{o} Celsius meaning patient's tag has been removed and manual searching will be conduct.
- If no result from detection meaning patient is out of hospital range.
- For the emergency group search, it will list down all the people in the hospital based on their location. If number of patients of list satisfies the list, searching will stop
- If number of patients of list does not satisfy the list, alarm will be activated. System will identify the missing patients by conducting individual search and rescue missing patient will be executed.

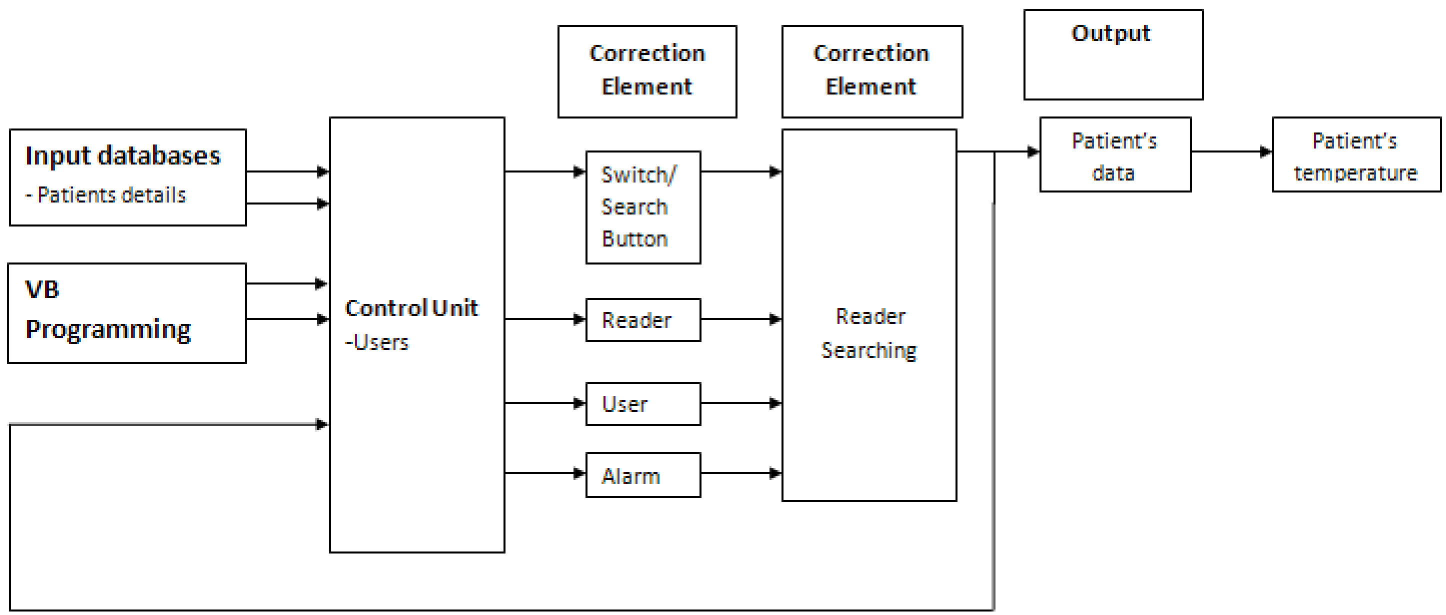


Figure 3.3: Block diagram for monitoring system

3.5 Tools

Required hardware

The system comprises appropriate hardware (supplied by ActiveWave) and in-house software for interaction. The hardware includes:

- Card Tag: is assigned to each holder as their identification card. The tag has an internal memory in which the holder's ID is stored;
- Standard reader: The reader serves as an interface between the hardware component and the host computer. Whenever a tag is waken up by the field generator, it transmits its ID to the reader. The reader then sends to the host the information captured from the tag;
- An automatic fire alarm system that is designed to detect the unwanted presence of fire by monitoring environmental changes associated with combustion. In general, a fire alarm system is either classified as automatically actuated, manually actuated, or both.

Appendix provides more details about the above cited hardware.

Required Software

- Microsoft Visual Basic : to create user interface for the program
- PHP : to create user interface for the program
- MySQL: runs as a server providing multi-user access to a number of databases.
- XAMPP: an open source cross-platform web server for creating and manipulating databases in MySQL.

CHAPTER 4
RESULT AND DISCUSSION

4.1 Hardware

For the choosing of hardware, it is finalized that this system will be using active RFID (refer to Table 3.2: Delphi Decision Matrix on Hardware Justification). The main reason for this decision is because the active tags are easier to be obtained (from collaboration with Electrical Engineering Department.). Active tags are also chosen because it provides greater range of detection. It is also easier to be detected, thus making this system more efficient. The only problem with active tags is that the durability of the tag itself, since the battery life is unpredictable and sometimes the power can run out very fast. Therefore, more research needs to be done in order to improve the durability of the tags. There is also a need to optimize the whole system including readers and tags detection capability. To optimize the hardware, the steps below need to be taken:

1. Reader transmission and tag detection optimization
   - Design of hospital – study of the design of hospital area in order to know the geographical position of the seats and the most suitable position to put the reader.
   - Interference caused by other source e.g, RF, EM – it is important to know recognize or identify the other possible source of radio frequency or electromagnetic as to reduce interference.

2. Tag's battery durability
   - Active tag circuitry – a study needs to be done on the circuitry of the tags in order to fully understand the function of each component and improvement can be taken to improve its durability.

- Battery position in the tag – from the study of the tag circuitry, a closer look on the positioning of the battery can be done in order to improve its durability.
- Rechargeable tag – the last option that can be done to improve the tag durability.

4.2 Hardware Testing

This main purpose of the test is to verify the maximum distance that the tag can responds to the reader's call. Firstly, the battery condition of the tag is verified. Although the battery is low, it is considered okay to be used for the test Secondly, it has been verified that there is an establish connection or communication between the tag and the reader (refer to Figure 12). Both the first and second part is verified using the Query command in Programming Station.

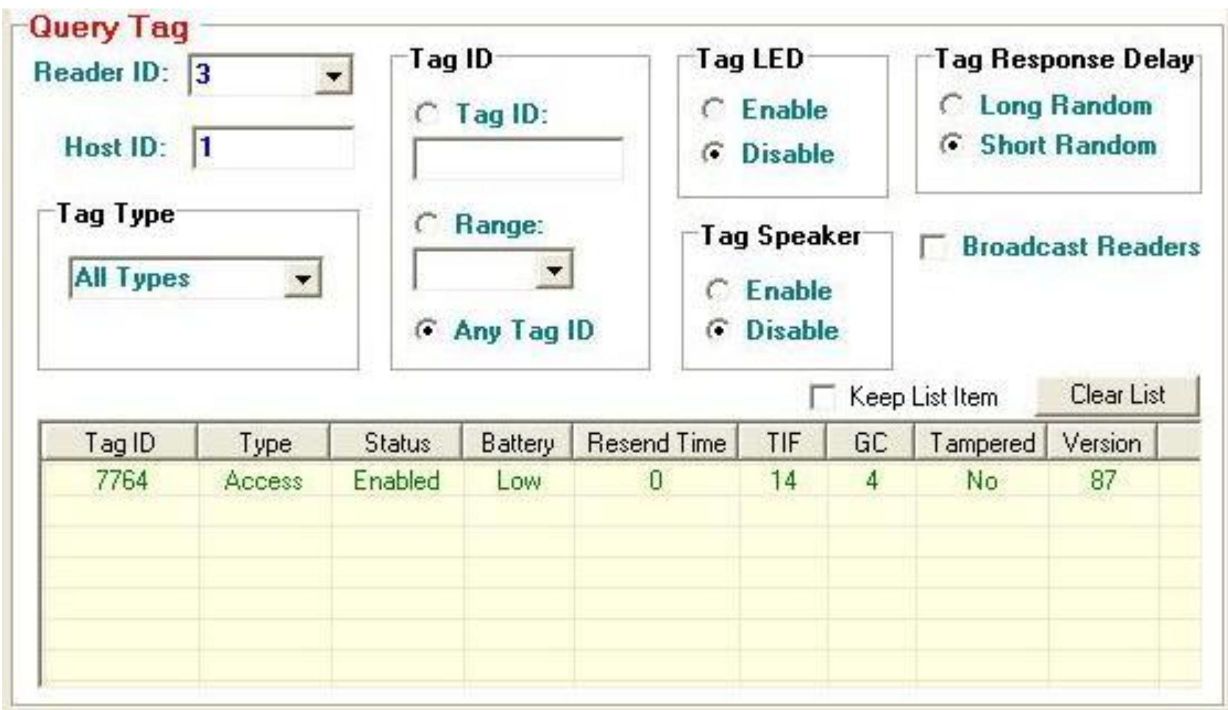


Figure 4.1: Query of tag for hardware testing

For the third part, the distance between the reader and the tag is varied, and the time taken for the tag to responds to reader's call is taken. The test is being done in a room (confined space). The result of this experiment is as shown below in Table 4.1. From this is experiment, it can be concluded that the maximum distance that can be covered by the system is about 20 meters. Despite the result, more tests and experiments should be conducted in order to know the coverage distance of the reader and tag communications in various environments.

Table 4.1: Hardware testing result

Distance (meter)	**Tag's Detection (Yes/No)**	**Time Taken (sec)**
2.0	Yes	5
4.0	Yes	5
6.0	Yes	6
8.0	Yes	6
10.0	Yes	8
12.0	Yes	10
14.0	Yes	14
16.0	Yes	16
18.0	Yes	20
20.0	Yes	30
22.0	No	-
24.0	No	-

4.3 System Designed

From the system designed for the hospital system, the system has brings a more flexibility in term of searching for tag holder in a shorter time rather to be using normal method. For people searching, it can be done by individual search and group searching depending on the situation. The system has provided a better automatic system, without the human having to do searching manually. A justification is referred in order to design the best system to be implemented for the project.

1. Time efficiency
   - One of the main objectives of this project is to provide a efficient searching system that will make human job faster and easier in searching. Therefore, it is an important factor so that this objective is achieved.
2. Reliability of system
   - The system designed should provide reliability in terms of detecting each tag that in different location of the hospital. This is crucial since without a reliable detection, this system won't function properly thus not meeting its objectives.
3. System controllability
   - Controllability of system defines the system easiness to be adjusted or manipulated in case of any changes occurred. For example, if the patient is released from hospital, the management should be able to delete the data and to reuse the tag for other patients.
4. Cost
   - The system designed should not exceed the budget limited for both FYP 1 and FYP 2 project which is RM 500.00.
5. Maintenance
   - System maintenance is chosen as a one of a factor because the system needs to be maintained every once in a while for its efficiency.

4.4 Software

As discussed in previous chapter, there are three main softwares or programs that are required in order for the system to work efficiently. The first program which is Programming Station is developed by Activewave Inc. and is already a proven program. The same goes to the second program, the Patient List Database which is an open source program developed by Apache. Therefore, only the Patient searching-In Program which is self-developed by the author will be discussed in this section.

4.4.1 Searching Check-In Program

This program is very crucial for the searching system since the built interface will communicate and react with both the reader and the patient list database. Therefore, it is considered as the key program that will determine the successfulness of this project. The program interface is as shown by Figure 11. The functions of each numbered icon (refer to Figure 4.1) are as follows:

Emergency Search (Search by group)

1. Config:
   - To choose ip addresses for different reader in different direction.
2. Connect:
   - To establish connection between the Patient List Program and the RFID reader.
3. Start:
   - To prompt the readers to start calling or looking for tags.
4. Stop:
   - To stop the readers from calling or looking for tags.

5. Result:
   - This is where the patients location, patients name and time will shown which detected by readers.

Individual Search (Temperature retriever)

6. Temperature:
   - The temperature box on the right will be appeared by clicking on the button.
7. Enter:
   - Entering patient id.
8. Get
   - Information about patient name, location and body temperate will be displayed.

Patient Data History

9. History
   - Individual patient's history such as last doctor visit and patient's diagnosis can be displayed
10. Add/Remove
    - Early registration can be done for the admission of new patients in the hospital.
11. AddHistory
    - To key in/edit patient health information for future references.

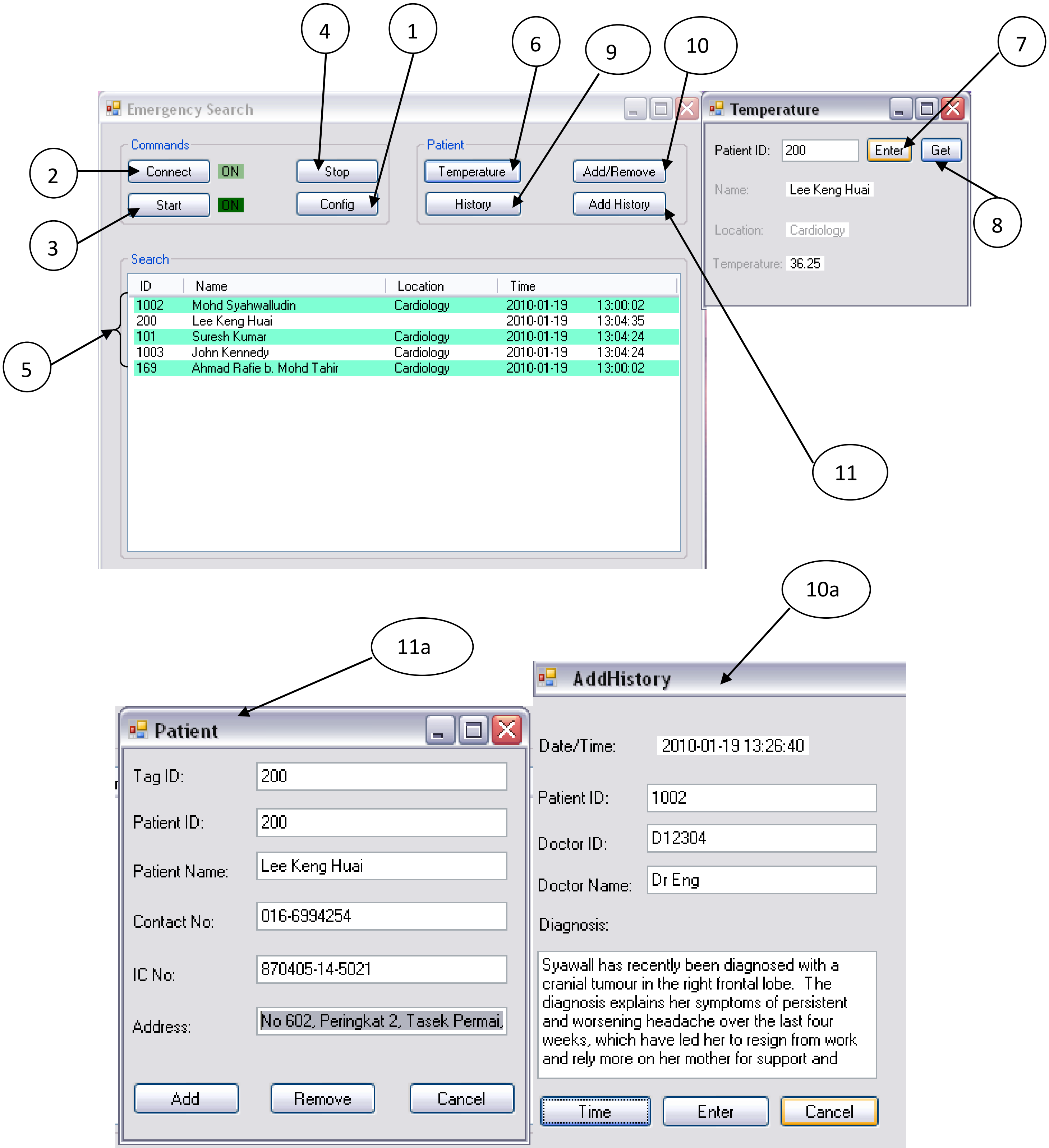


Figure 4.2: Patient Searching List Database

4.4.2 Patient List Database

The Patient List Database is databases consist of list of patient in the hospital including each tag id, patient name, contact number and address. This database is built using MySQL and XAMPP:

1. MySQL: runs as a server providing multi-user access to a number of databases.
2. XAMPP: an open source cross-platform web server from Apache for creating and manipulating databases in MySQL.

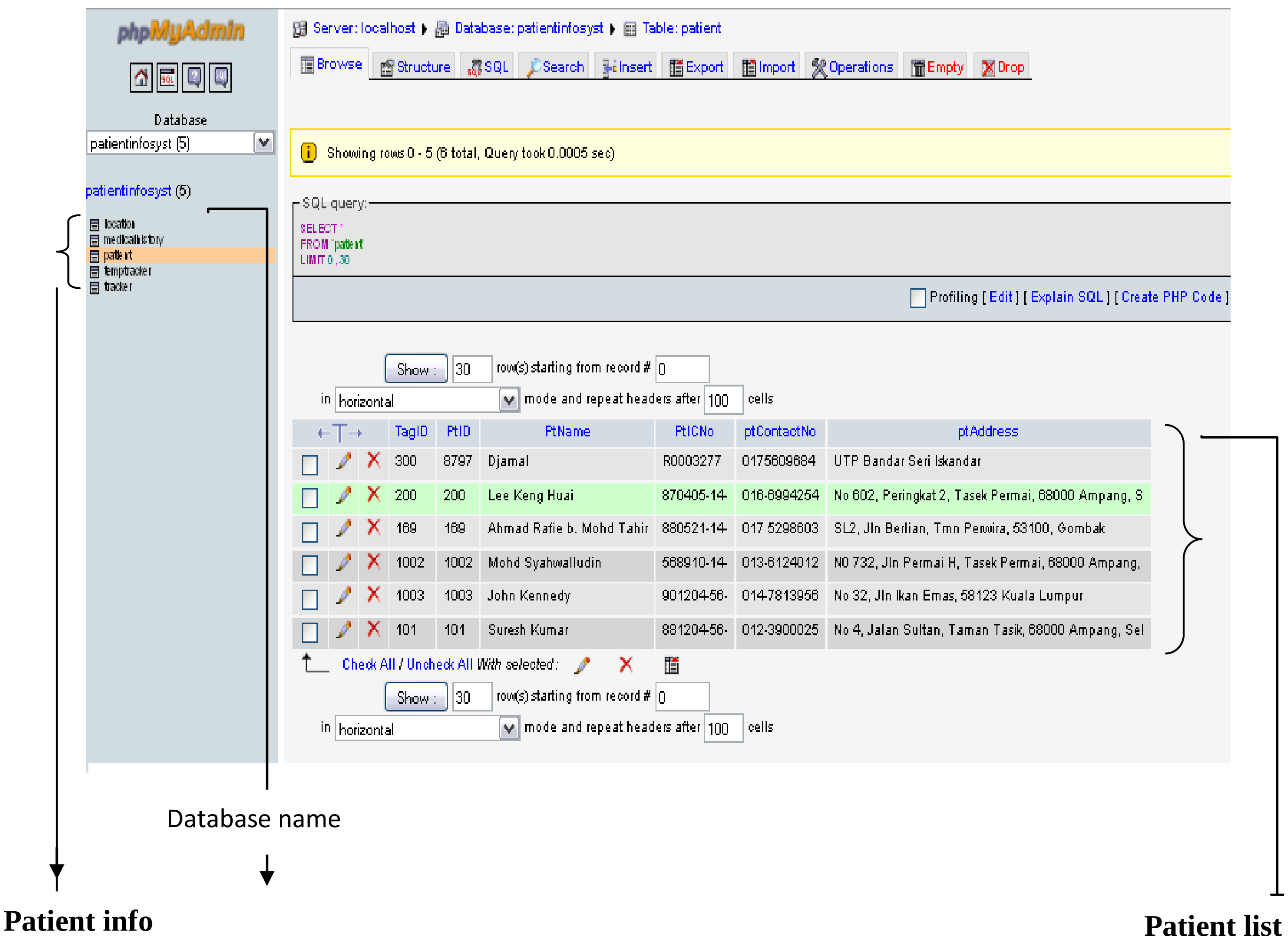


Figure 4.3: Patient List Database

4.5 Theoretical Evaluation

4.5.1 Interrogation Field Strength H_{min}

H_{min} is the minimum field strength, at a maximum distance x between the reader and transponder, that supply enough voltage for the operation of the transponder. This value can be computed by using the equation below:

$$H_{min} = \frac{u_2 \times \sqrt{\omega^2\left(\frac{L_2}{R_L} + \frac{R_2}{\omega_0^2 L_2}\right)^2 + \left(\frac{\omega_0^2 - \omega^2}{\omega_0^2} + \frac{R_2}{R_L}\right)^2}}{\omega \times \mu_0 \times A \times N}$$

Where

u_2: High Frequency input voltage

μ_0: Permeability constant

ω: The angular frequency of the magnetic field (reader transmission frequency)

ω_0: The resonant frequency of the transponder

L_2: Transponder coil inductance

R_2: Transponder input resistance

RL: Transponder load resistor

N: Number of windings of the transponder coil

A: The cross-sectional area of the transponder coil

Large H_{min} means that the transponder needs a lot of power to start proceeding. Improving H_{min} can be feasible by increasing transponder's area and windings and by decreasing transponder's input and load resistance and transponder coil inductance.

The values used are: N = 4, A = 0.05*0.08 m², u_2 = 5V, L_2 = 3.5 µH, R_2 = 5**Ω**, R_L =1.5k **Ω**, ω = 433 MHz, ω_0 = 916 MHz, μ_0=1,

This gives H_{min} = 2.01

4.5.2 Energy Range

After distinguishing H_{min}, the energy range for a certain reader can be assessed. The formula below will be used for that purpose:

$$x = \sqrt{\sqrt[3]{\left(\frac{I \times N_1 \times R^2}{2 \times H_{min}}\right)} - R^2}$$

Where

I: Electric current

N_1: The number of windings of the transmitter antenna

R: The radius of the circle coil antenna

H_{min}: The interrogation field strength of a transponder

x: maximum readable distance between the transponder and the antenna.

The values used are: $N_1 = 4$, $R = 10$ mm, $H_{min} = 2$ $u_2 = 5V$, $L_2 = 3.5$ µH, $R_2 = 5\Omega$, $R_L = 1.5k\ \Omega$, $\omega = 433$ MHz, $\omega_0 = 916$ MHz, $\mu_0 = 1$, I = 5A

This gives x ≈ 21 m

CHAPTER 5

CONCLUSION AND RECOMMENDATION

In a nutshell it can be concluded that this project main objective is to create a monitoring and searching system for the Alzheimer patients. The system created must be able to record and publish patients in a particular location in a shortest time and most importantly it must be efficient. This is possible by using Radio Frequency Identification technology. RFID allows identification of specific tags through data exchanging with radio frequency being transmitted and received. Scanning can be done at a greater distance and with less effort than barcode scanning, offering great efficiencies. RFID can deliver information in real time according to real life events.

The hardware of the system is borrowed from UTP Electrical Engineering Department. The main interface, which is the Patients Check-In Program, is built using Visual Basic. The hardware and software are then combined to produce a complete system. Based on result of test that has been made on the monitoring system, it can be concluded that the monitoring system built are working perfectly, thus meeting the objectives of the project. However, the hardware part of the system still needs improvements mainly in the tag's battery area. As a conclusion, this project has been successful in meeting its design objective and hopefully, it will become a platform for future monitoring system in every hospital.

REFERENCES:

Shey Ahson, Mohammad Ilyas. March 2008 in “Interactive models for RFID” Applications Technology Security and Privacy

Stephen B.Miles, Samjay E.Sarma and John R.Williams. June 2008 RFID Techology Application

V. Daniel Hunt, Albert Puglia, Mike PugliaRFID. 2007 , A Guide to Radio Frequency Identification by John Wiley & Sons

Yanna Hao, Zhiliang Hong , 2007. Development of a Low-cost Active RFID Platform.

Stevan Preradovic, Nemai C. Karmakar, Isaac Balbin. 2008. RFID Transponder. Melbourne

Steven Shepard .2005 RFID: Radio Frequency Identification.

Paolo Magrassi. 2001. "A World of Smart Objects: The Role Of Auto Identification Technologies".

Monahan, Torin. 2006. Sureveillance & Security. Technological Politics ans Power in EverydayLife. New York: Routledge

Harvey Lehpamer. 2008. RFID design Principle. Norwood: Artech House Inc.

Michael Buettner, David Wetherall. 2008. An Empirical Study of UHF RFID Performance.California

Stevan Preradovic, Nemai C. Karmakar, Isaac Balbin. 2008. RFID Transponder. MelbourneAustralia

Bjorn Nilsson, Lars Bengtsson, Urban Bilstrup, Per-Arne Wiberg and Bertil Svensson. 2006. Towards an Energy Efficient Protocol for Active RFID. Sweden

Lu Yan, Yan Zhang, Laurence T. Yang, Huansheng Ning. 2008. The Internet of Things. NewYork: Auerbach Publications

Peter Bernardin, Meng Yee, Thomas Ellis. 1998. Cell Radius Inaccuracy: A New Measure of Coverage Reliability, Texas

Patrick J. Sweeney (2005). RFID for Dummies. John Wiley & Sons

http://www.rfidjournal.com/article/articleview/4311/1/82/

Activewave - *www.activewaveinc.com*

Microchip - *www.microchip.com*

RFID journal - *www.rfidjournal.com/*

http://en.wikipedia.org/wiki/Radio-frequency_identification

http://www.rfidgazette.org/2004/07/rfid_in_the_hos.html

http://www.supplyinsight.com/RFID_in_Hospital_Equipment_Tracking.htm

http:// www.suite101.com/article.cfm/alzheimers_disease/93895

http:// www.psfk.com/2009/08/using-rfid-to-track-hospital-patients.html

APPENDICES

‘

Appendix 1: Gantt Chart of FYP II

No	Activities / Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	Simulation														
2	Submission of progress report 2														
3	Simulation of thermal expansion development and distribution														
4	Submission of progress report 2														
5	Seminar														
6	Review of project objectives and results														
7	Final report preparation														
8	Poster exhibition														
9	Submission of dissertation final draft														
10	Oral presentation														
11	Submission of dissertation (hard bound)														

Appendix 2: Tags

A tag is the piece of hardware that is used to identify items and people. It has the capability to store basic data such as ID and the type of the tag to uniquely identify each tag. There are various types of tags designed for various applications. Depending on the type of tags, the memory space ranges from 0 to 256 Kbits. Having the dimensions of a credit card but a little bit thicker, tags can be customized and used as employees’ badges in a secure, access control environment. These tags can also be attached to a flat surface of any merchandise or equipment. warehouse, or other type of facility. For example, the tags can be used to track employees in a petro-chemical plant, find specific item in a warehouse, identify people and track their body temperature in hospital, etc…

Tags usually remain in sleep mode until they receive a wake up command (call or query) from a reader. If they are within the communication range, and the command is destined to them (specific address or broadcast), then they awaken and transmit their ID and other information (such as type, battery status, signal strength aka RSSI, etc). They can also be configured to automatically wake up at predefined time intervals, transmit their information to the system, then go back to sleep to conserve battery life.

Unique anti-collision algorithms are used to ensure that all tag data is received, even when multiple tags are transmitting at the same time. To guarantee that all data remains accurate, all tag packets use reliable cyclic redundancy checks. Tags communicate with the reader using dual-frequency method: they use one frequency for transmit and another frequency to receive making the system faster and more reliable.

Fig 4: Patient Tag –used as patient tag in a secure, access-controlled environment.

Appendix 3: Reader

The reader consists of a real time processor, operating system, virtual portable memory, and transceiver in one small self-contained module that is easily installed in the ceiling or in any other convenient location. It has a built in field generator that generates radio signal to which each tag located within the field range should respond. The reader can listen to tag beacons (when more than one tag responds to the reader call), and can be used to identify individual tags.

In addition to its basic uses that include identifying and calling tags, the reader is also used to configure the tags. That allows the operator to assign to each tag an individual ID, and a type (can be Access, Inventory, Asset or Factory). Anytime a call or query function is generated, a full duplex communication is established between the reader and the tag. The reader also provides immunity to noise and interference. The working frequencies of the system are based on the international platform of RFiD systems. The long read range RFiD systems employ a midrange frequency, and a unique complex software algorithm to provide noise immunity and error-free operation in high-interference environments. There are two types of readers: Standard Reader/Long-range Reader and Programming Station/Short-range Reader.

Fig 5: Standard/Long-range reader

Appendix 4: Automated Alarm

Operation

The principle components of a S.A.B. Unit are a small battery and a switching unit such as a relay, as seen below.

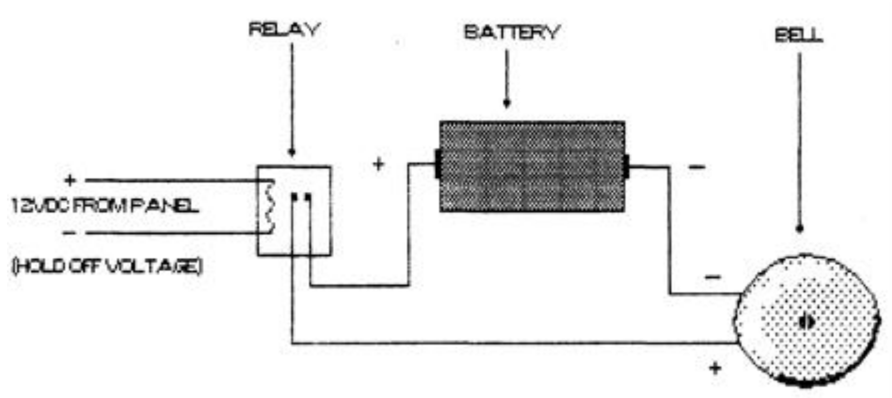


Fig 7: Principle of alarm component

A constant 12VDC supply is taken from the control panel and is connected to a relay such that the relay switch is "open circuit" until the power is removed when the relay switch will become "closed circuit" and thus sound the bell.

The battery used is a rechargeable nickel-cadmium type and is recharged by the 12VDC hold-off voltage from the panel so it is not necessary to replace after use.

Wiring S.A.B Units

There are quite a number of different makes and models of S.A.B units available, however they all operate on the same principle as described on the previous page.

Below is shown a diagram of a universal unit that is one which can be used in any system.

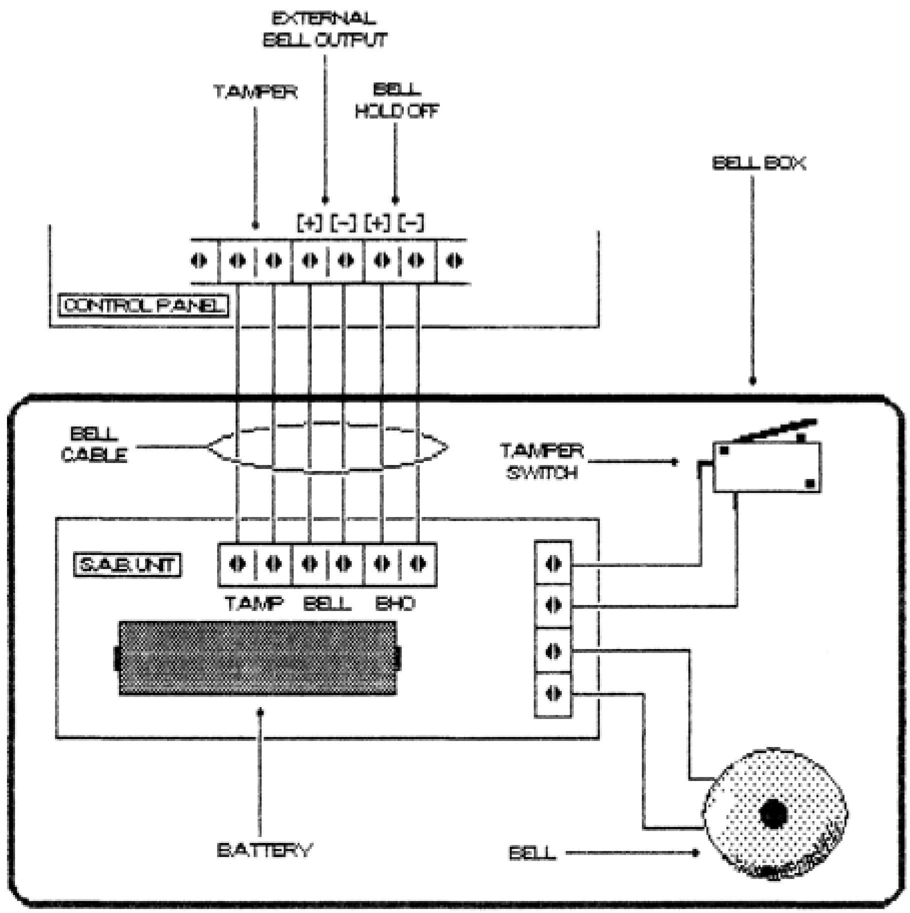


Fig 8: universal unit for alarm system

The following connections to the control panel will be necessary for the unit to operate:

BHO (Bell Hold Off)

The self actuating bell unit requires a constant 12VDC to keep the bell from sounding. This voltage is supplied by the control panel from the 12VDC aux supply, however most panels will have a separate supply marked BHO.

Obviously if the cable is cut the ni-cad battery will take over and ring the bell.

Bell Trigger Connection From Panel

This is taken from the external bell output on the panel, and will ring the bell directly when the alarm system activates during emergency group search.

Tamper

As with all wiring in an alarm system a tamper loop is required, and in this case is terminated at the S.A.B. Unit. This will indicate to the panel if the wire is cut, the tamper switch on the bell box is open op. The 12VDC hold-off is removed.

The diagram on the previous page shows the wiring arrangement between the panel and the S.A.B. Unit. In this case six wires are needed but for most systems this can be reduced to only four.

Appendix 5: Visual Basic Programming of Search Patient System

```
Imports System.Data
Imports System
Imports System.Runtime.InteropServices
Imports Microsoft.VisualBasic
Imports AW_API_NET
Imports System.IO
Imports MySql.Data.MySqlClient
Imports System.Threading

Public Class Form1

  Dim Hconn As IntPtr
  Dim readerIP(20) As Byte
    Dim readerPort As UInt16
    Dim commPort As UInt32
    Dim commBaud As UInt32
    Public myPKTID As Integer
        Dim registered As Boolean
    Public ActiveWaveAPI As AW_API_NET.APINetClass = New
AW_API_NET.APINetClass
 Public ReaderEventHandler As AW_API_NET.fReaderEvent
  Public TagEventHandler As AW_API_NET.fTagEvent
    Dim pubReaderID As Integer
    Dim tagSelect As AW_API_NET.rfTagSelect_t

 Dim TrdTemp As Thread

   Dim strCn As String = "Database=PatientInfoSyst;Data
Source=localhost;User Id=root;Password="
 Dim tagID As ListViewItem
 Private Thd As System.Threading.Thread

  Private Sub Form1_Load(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles MyBase.Load


System.Windows.Forms.Control.CheckForIllegalCrossThreadCalls =
False


  readerPort = Convert.ToUInt16(10001)
  commPort = Convert.ToUInt32(1)
  commBaud = Convert.ToUInt32(115200)
  ReaderEventHandler = New AW_API_NET.fReaderEvent(AddressOf
Me.OnReaderEvent)  'AddressOf Me.OnReaderEvent
  TagEventHandler = New AW_API_NET.fTagEvent(AddressOf
Me.OnTagEvent)

  registered = False
```

```
  BtnStart.Enabled = False
  BtnStop.Enabled = False
  BtnTemp.Enabled = False

  LvwPatient.Columns.Clear()
  LvwPatient.Columns.Add("ID", 50, HorizontalAlignment.Left)
  LvwPatient.Columns.Add("Name", 180, HorizontalAlignment.Left)
  LvwPatient.Columns.Add("Location", 100,
HorizontalAlignment.Left)
  LvwPatient.Columns.Add("Time", 160, HorizontalAlignment.Left)

  DelTracker()
  TmrDispSearch.Start()

 End Sub

  Private Function OnReaderEvent(ByVal readerEvent As
AW_API_NET.rfReaderEvent_t) As Integer

       Dim ipStr As String = ""

        'Beep()
   If
readerEvent.eventType.Equals(Convert.ToUInt16(AW_API_NET.APIConst
s.RF_SCAN_NETWORK)) Then
      For i As Integer = 0 To readerEvent.ip.Length - 1
          ipStr += Convert.ToChar(readerEvent.ip(i))
      Next i
      IPListBox.Items.Add(ipStr)
  ElseIf
readerEvent.eventType.Equals(Convert.ToUInt16(AW_API_NET.APIConst
s.RF_OPEN_SOCKET)) Then
      For i As Integer = 0 To readerEvent.ip.Length - 1
          ipStr += Convert.ToChar(readerEvent.ip(i))
      Next i
      'AddMsg("Socket Opened IP = " + ipStr)
  ElseIf
readerEvent.eventType.Equals(Convert.ToUInt16(AW_API_NET.APIConst
s.RF_CLOSE_SOCKET)) Then
      For i As Integer = 0 To readerEvent.ip.Length - 1
          ipStr += Convert.ToChar(readerEvent.ip(i))
      Next i
      'AddMsg("Socket Closed IP = " + ipStr)

      Dim index As Integer
      If ipStr.Length > 0 Then
          index = IPListBox.FindStringExact(ipStr)
          If index >= 0 Then
              IPListBox.Items.RemoveAt(index)
          End If
      End If
  ElseIf
readerEvent.eventType.Equals(Convert.ToUInt16(AW_API_NET.APIConst
s.RF_STD_FGEN_POWERUP)) Then
      'FGenIDTextBox.Text = readerEvent.fGenerator.ToString()
      'AddMsg("STD FGen Powered UP")
```

```
  ElseIf
readerEvent.eventType.Equals(Convert.ToUInt16(AW_API_NET.APIConst
s.RF_READER_POWERUP)) Then
      'ReaderIDTextBox.Text = readerEvent.reader.ToString()
      'tText(readerEvent.reader.ToString())
      'AddMsg("Reader Powered UP")
  ElseIf
readerEvent.eventType.Equals(Convert.ToUInt16(AW_API_NET.APIConst
s.RF_QUERY_STD_FGEN)) Then
      Dim str As String

      'AddMsg(readerEvent.eventType.ToString)
      'AddMsg(AW_API_NET.APIConsts.RF_READER_POWERUP.ToString)

      'AddMsg("STD FGEN Query __________")
      str = readerEvent.smartFgen.fsValue
      'AddMsg("FS Value = " + str)
      str = readerEvent.smartFgen.txTime
      'AddMsg("TX Time = " + str)
      str = readerEvent.smartFgen.waitTime
      'AddMsg("Wait Time = " + str)
      str = readerEvent.smartFgen.assignRdr
      'AddMsg("Assigned Rdr = " + str)
  ElseIf
readerEvent.eventType.Equals(Convert.ToUInt16(AW_API_NET.APIConst
s.RF_GET_RDR_FS)) Then
      'FSTextBox.Text = readerEvent.smartFgen.fsValue.ToString()
      'AddMsg("Reader ID:" + readerEvent.reader.ToString() + "
FS:" + readerEvent.smartFgen.fsValue.ToString())
  ElseIf
readerEvent.eventType.Equals(Convert.ToUInt16(AW_API_NET.APIConst
s.RF_SET_RDR_FS)) Then
      'AddMsg("Reader FS was set successfuly")
  ElseIf
readerEvent.eventType.Equals(Convert.ToUInt16(AW_API_NET.APIConst
s.RF_SCAN_IP)) Then
      ipStr = GetStringIP(readerEvent.ip)
      If ipStr.Length > 0 Then
          If IPListBox.FindStringExact(ipStr) = -1 Then
              IPListBox.Items.Add(ipStr)
          End If
          'AddMsg("ScanIP IP=" + ipStr)
      End If
  End If

  ReportReaderEvent(readerEvent)

  Return 0
 End Function

 Private Function OnTagEvent(ByVal tagEvent As
AW_API_NET.rfTagEvent_t) As Integer

  Dim ipStr As String
  ipStr = ""
  Beep()
```

```
  If
tagEvent.eventType.Equals(Convert.ToUInt16(AW_API_NET.APIConsts.R
F_TAG_READ)) Then
      Dim str As String
      Dim n As Integer
      n = CInt(Convert.ToInt16(tagEvent.tag.dataLen))
      For i As Integer = 0 To n - 1
          str = tagEvent.tag.data(i)
          ipStr += str + " "
      Next
      'AddMsg("Data :" + ipStr + vbCrLf)
  End If

  ReportTagEvent(tagEvent)

  Return 0
End Function

Private Sub ReportReaderEvent(ByRef readerEvent As
AW_API_NET.rfReaderEvent_t)
  If readerEvent.errorStatus.ToString() = 0 Then
      LblConnectStatus.Text = "ON"
      pubReaderID = readerEvent.reader.ToString()
      BtnStart.Enabled = True
      BtnTemp.Enabled = True
  End If

End Sub

Private Sub ReportTagEvent(ByRef tagEvent As
AW_API_NET.rfTagEvent_t)
  Dim msg As String

  msg = "eventType " + tagEvent.eventType.ToString() + vbCrLf _
      + vbTab + "errorStatus = " +
tagEvent.errorStatus.ToString() + vbCrLf _
      + vbTab + "host = " + tagEvent.host.ToString() + vbCrLf _
      + vbTab + "reader = " + tagEvent.reader.ToString() + vbCrLf
            + vbTab + "fGenerator = " +
tagEvent.fGenerator.ToString() + vbCrLf _
      + vbTab + "eventStatus = " +
tagEvent.eventStatus.ToString() + vbCrLf _
      + vbTab + "cmdType = " + tagEvent.cmdType.ToString() +
vbCrLf _
      + vbTab + "RSSI = " + tagEvent.RSSI.ToString() + vbCrLf _
      + vbTab + "tagID = " + tagEvent.tag.id.ToString() + vbCrLf
      + vbTab + "tagType = " + tagEvent.tag.tagType.ToString() +
vbCrLf _
      + vbTab + "pktID = " + tagEvent.pktID.ToString() + vbCrLf


  If tagSelect.selectType =
Convert.ToUInt32(AW_API_NET.APIConsts.RF_SELECT_FIELD) Then
      If tagEvent.tag.id > 0 Then
          'MessageBox.Show(tagEvent.tag.id.ToString)
          insertInfo(tagEvent.tag.id.ToString,
tagEvent.reader.ToString)
```

```
      End If
  ElseIf tagSelect.selectType =
Convert.ToUInt32(AW_API_NET.APIConsts.RF_SELECT_TAG_ID) Then
      If tagEvent.tag.id > 0 Then
          'MessageBox.Show(tagEvent.tag.id.ToString + "    " +
tagEvent.tag.temp.temperature.ToString)
          insertTemp(tagEvent.tag.id.ToString,
tagEvent.reader.ToString, tagEvent.tag.temp.temperature.ToString)
      End If
  End If

  'AddMsg(msg)
 End Sub

Private Sub insertTemp(ByVal tid As String, ByVal rdr As String,
ByVal temp As String)

  Dim name As String = ""
  Dim loc As String = ""
  Dim id As String = ""

  Dim cn As New MySqlConnection(strCn)

  Dim DelTempCmd As New MySqlCommand("DELETE FROM tempTracker",
cn)
  If cn.State = ConnectionState.Closed Then
      cn.Open()
  End If
  DelTempCmd.ExecuteNonQuery()
  If cn.State = ConnectionState.Open Then
      cn.Close()
  End If

  Dim IdNameCmd As New MySqlCommand("SELECT * FROM patient WHERE
TagID = '" & tid & "'", cn)
  Dim IdNameRdr As MySqlDataReader
  If cn.State = ConnectionState.Closed Then
      cn.Open()
  End If
  IdNameRdr = IdNameCmd.ExecuteReader
  While IdNameRdr.Read
      id = IdNameRdr.Item("PtID")
      name = IdNameRdr.Item("ptName")
  End While
  If cn.State = ConnectionState.Open Then
      cn.Close()
  End If

  Dim LocCmd As New MySqlCommand("SELECT LocName FROM location
WHERE ReaderID = '" & rdr & "'", cn)
  Dim LocRdr As MySqlDataReader
  If cn.State = ConnectionState.Closed Then
      cn.Open()
  End If
  LocRdr = LocCmd.ExecuteReader
  While LocRdr.Read
      loc = LocRdr.GetString(0)
```

```
  End While
  If cn.State = ConnectionState.Open Then
      cn.Close()
  End If

  Dim insertCmd As New MySqlCommand("INSERT INTO tempTracker
(PtID, PtName, LocName, Temperature) VALUES ('" & id & "', '" &
      name & "', '" & loc & "', '" & temp & "')", cn)
  If cn.State = ConnectionState.Closed Then
      cn.Open()
  End If
  insertCmd.ExecuteNonQuery()
  If cn.State = ConnectionState.Open Then
      cn.Close()
  End If

  Temperature.updateTemp()
  Temperature.Update()
 End Sub

Private Sub insertInfo(ByVal tid As String, ByVal rdr As String)
  Dim name As String = ""
  Dim loc As String = ""
  Dim id As String = ""

  Dim cn As New MySqlConnection(strCn)

  Dim IdNameCmd As New MySqlCommand("SELECT * FROM patient WHERE
TagID = '" & tid & "'", cn)
  Dim IdNameda As New MySqlDataAdapter
  Dim IdNameds As New DataSet
  Dim IdNamedt As New DataTable
  Dim IdNameCount As Integer = 0
  Dim IdNameRdr As MySqlDataReader

  If cn.State = ConnectionState.Closed Then
      cn.Open()
  End If
  IdNameda.SelectCommand = IdNameCmd
  IdNameda.Fill(IdNameds, "Patient")
  IdNamedt = IdNameds.Tables("Patient")
  IdNameCount = IdNamedt.Rows.Count
  If cn.State = ConnectionState.Open Then
      cn.Close()
  End If

  If IdNameCount > 0 Then

      If cn.State = ConnectionState.Closed Then
          cn.Open()
      End If
      IdNameRdr = IdNameCmd.ExecuteReader()
      While IdNameRdr.Read
          id = IdNameRdr.Item("PtID")
          name = IdNameRdr.Item("PtName")
      End While
      If cn.State = ConnectionState.Open Then
```

```
            cn.Close()
        End If

        Dim LocCmd As New MySqlCommand("SELECT LocName FROM
        location WHERE ReaderID = '" & rdr & "'", cn)
        Dim LocRdr As MySqlDataReader
        If cn.State = ConnectionState.Closed Then
            cn.Open()
        End If
        LocRdr = LocCmd.ExecuteReader
        While LocRdr.Read
            loc = LocRdr.GetString(0)
        End While
        If cn.State = ConnectionState.Open Then
            cn.Close()
        End If

        Dim IdSearchCmd As New MySqlCommand("SELECT * FROM tracker
        WHERE PtID = '" & id & "'", cn)
        Dim IdSearchda As New MySqlDataAdapter
        Dim IdSearchds As New DataSet
        Dim IdSearchdt As New DataTable
        Dim IdSearchCount As Integer = 0
        cn.Open()
        Try
            IdSearchda.SelectCommand = IdSearchCmd
            IdSearchda.Fill(IdSearchds, "searchresult")
            IdSearchdt = IdSearchds.Tables("searchresult")
            IdSearchCount = IdSearchdt.Rows.Count()
        Catch ex As Exception
            MsgBox("Error: " & ex.Source & ": " & ex.Message,
        MsgBoxStyle.OkOnly, "Connection Error !!")
        End Try
        If ConnectionState.Open Then
            cn.Close()
        End If

        If IdSearchCount = 0 Then
            Dim InsertCmd As New MySqlCommand("INSERT INTO tracker
        (PtID, PtName, LocName, CurrTime) VALUES ('" & id & "', '"
        & name & "', '" & loc & "', '" & Format(DateTime.Now,
        "yyyy-MM-dd       HH:mm:ss") & "')", cn)
            If cn.State = ConnectionState.Closed Then
                cn.Open()
            End If
            InsertCmd.ExecuteNonQuery()
            If cn.State = ConnectionState.Open Then
                cn.Close()
            End If
        Else
            Dim UpdateCmd As New MySqlCommand("UPDATE tracker SET
        LocName = '" & loc & "', CurrTime = '" &
Format(DateTime.Now, "yyyy-MM-dd HH:mm:ss") & "' WHERE PtID = '"
& id & "'", cn)
            If cn.State = ConnectionState.Closed Then
                cn.Open()
            End If
```

```
            UpdateCmd.ExecuteNonQuery()
            If cn.State = ConnectionState.Open Then
                cn.Close()
            End If
        End If

    End If

  End Sub

   Public Function GetStringIP(ByVal ip As Byte()) As String

    Dim p As Integer
    Dim s As String
    Dim ct As Integer

    ct = 0
    p = 0
    s = ""
    While (Convert.ToBoolean((ct <= 3)) AndAlso
  Convert.ToBoolean((p < 20)) AndAlso Convert.ToBoolean((ip(p) <>
  0)))
        If ip(p) <> 46 Then
            s += Convert.ToString(ip(p) - 48) '- 48
            p += 1
        Else
            ct += 1
            p += 1
            s += "."
        End If

    End While
    Return s
   End Function


Private Sub BtnConfig_Click(ByVal sender As System.Object, ByVal e
  As System.EventArgs) Handles BtnConfig.Click
    frmConfig.Show()
End Sub

Private Sub BtnConnect_Click(ByVal sender As System.Object, ByVal e
  As System.EventArgs) Handles BtnConnect.Click
    scanIP()
    LblConnectStatus.BackColor = Color.DarkSeaGreen
End Sub

Private Sub scanIP()
    Dim iRet As Integer
    Dim ip(20) As Byte
    Dim stripx As String

    readIp()

    If (myPKTID >= 223) Then
        myPKTID = 1
    Else
```

```
        myPKTID = myPKTID + 1
    End If

    If registered = False Then

        ActiveWaveAPI.rfRegisterReaderEvent(ReaderEventHandler)

        ActiveWaveAPI.rfRegisterTagEvent(TagEventHandler)

        registered = True
    End If

    For c As Integer = 0 To frmConfig.lbIp.Items.Count - 1
        stripx = frmConfig.lbIp.Items.Item(c).ToString

        For i As Integer = 0 To stripx.ToString.Length - 1
            ip(i) = Convert.ToByte(stripx.ToString.Chars(i))
        Next i
        iRet = ActiveWaveAPI.rfScanIP(ip,
Convert.ToUInt16(myPKTID))

    Next c
    OpenConn()
End Sub

 Private Sub OpenConn()
    Dim iRet As Integer
    Dim ip(20) As Byte
    Dim cIP(20) As Char
    Dim stripx As String = "192.168.10.26"

    If (myPKTID >= 223) Then
        myPKTID = 1
    Else
        myPKTID = myPKTID + 1
    End If
    iRet = ActiveWaveAPI.rfOpenSocket(readerIP, readerPort, False,
Convert.ToUInt16(AW_API_NET.APIConsts.ALL_IPS),
Convert.ToUInt16(myPKTID))



    If (myPKTID >= 223) Then
        myPKTID = 1
    Else
        myPKTID = myPKTID + 1
    End If
    iRet = ActiveWaveAPI.rfResetReader(UInt16.Parse(1),
UInt16.Parse(0), UInt16.Parse(0),
Convert.ToUInt16(AW_API_NET.APIConsts.ALL_READERS),
Convert.ToUInt16(myPKTID))

'*****************************************************************
**********
    'added option
```

```
End Sub

Private Sub readIp()
  Try
      Using sr As StreamReader = New StreamReader("config.txt")
          Dim line As String
          line = sr.ReadLine()
          If Mid(line, 1, 3) = "ip:" Then
              frmConfig.lbIp.Items.Add(Mid(line, 4))
          End If
          While Not line Is Nothing
              line = sr.ReadLine()
              If Mid(line, 1, 3) = "ip:" Then
                  frmConfig.lbIp.Items.Add(Mid(line, 4))
              End If
          End While
          sr.Close()
      End Using
  Catch E As Exception
      ' Let the user know what went wrong.
      Console.WriteLine("The file could not be read:")
      Console.WriteLine(E.Message)
  End Try
 End Sub

Private Sub CloseConn()

  Dim iRet As Integer
  Dim ip(20) As Byte
  Dim cIP(20) As Char



  If (myPKTID >= 223) Then
      myPKTID = 1
  Else
      myPKTID = myPKTID + 1
  End If

  'If AllIPRadioButton.Checked Then
  iRet = ActiveWaveAPI.rfCloseSocket(readerIP,
Convert.ToUInt16(AW_API_NET.APIConsts.ALL_IPS))
  IPListBox.Items.Clear()

End Sub

 Private Sub emergSearch()
  Dim iRet As Integer
  Dim tagList(50) As UInt32
  Dim rdrID As UInt16
  Dim longInterval As Boolean
  Dim RdrCmdType As Integer

  rdrID = UInt16.Parse(0)

  RdrCmdType = APIConsts.ALL_READERS
```

```
  tagSelect.tagList = tagList

  tagSelect.tagList(0) = UInt32.Parse(0)
  tagSelect.numTags = Convert.ToUInt32(50)

  tagSelect.selectType =
Convert.ToUInt32(AW_API_NET.APIConsts.RF_SELECT_FIELD)

  tagSelect.tagType = APIConsts.ACCESS_TAG

  If (myPKTID >= 223) Then
      myPKTID = 1
  Else
      myPKTID = myPKTID + 1
  End If

  longInterval = False

  iRet = ActiveWaveAPI.rfCallTags(UInt16.Parse(1), rdrID,
UInt16.Parse(0), UInt16.Parse(0), tagSelect, True, longInterval,
Convert.ToUInt16(RdrCmdType), Convert.ToUInt16(myPKTID))
  'iRet = ActiveWaveAPI.rfQueryTags(UInt16.Parse(1), rdrID,
UInt16.Parse(0), tagSelect, True, longInterval,
Convert.ToUInt16(RdrCmdType), Convert.ToUInt16(myPKTID))
  'iRet = ActiveWaveAPI.rfGetTagTemp(UInt16.Parse(1), rdrID,
UInt16.Parse(0), tagSelect, True, longInterval,
Convert.ToUInt16(RdrCmdType), Convert.ToUInt16(myPKTID))

End Sub

 Private Sub TmrEmSearch_Tick(ByVal sender As System.Object,
ByVal e      As System.EventArgs) Handles TmrEmSearch.Tick
  emergSearch()
  End Sub

Private Sub BtnStart_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles BtnStart.Click
  TmrEmSearch.Start()
  LblStartStatus.BackColor = Color.DarkGreen
  LblStartStatus.Text = "ON"
  BtnStop.Enabled = True
End Sub

Private Sub BtnTemp_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles BtnTemp.Click
  TmrEmSearch.Stop()
  Temperature.Show()
End Sub

Private Sub getTemp()
  If Me.InvokeRequired Then
      Me.Invoke(New MethodInvoker(AddressOf getTemp))
  Else
      Dim obj As New Temperature
      obj.Show()
  End If
End Sub
```

```
Public Sub tempCheck(ByVal id As String)
  Dim iRet As Integer
  Dim tagList(50) As UInt32
  Dim rdrID As UInt16
  Dim longInterval As Boolean
  Dim RdrCmdType As Integer

  rdrID = UInt16.Parse(0)

  RdrCmdType = AW_API_NET.APIConsts.ALL_READERS

  tagSelect.tagList = tagList

  If (Temperature.TxtPatientID.Text.Equals("")) Then
      MsgBox("No Tag ID", MsgBoxStyle.OkOnly, "Error Msg")
      Return
  Else
      tagSelect.tagList(0) = Convert.ToUInt32(id)
      tagSelect.numTags = Convert.ToUInt32(1)
  End If

  tagSelect.selectType =
Convert.ToUInt32(AW_API_NET.APIConsts.RF_SELECT_TAG_ID)

  tagSelect.tagType = AW_API_NET.APIConsts.ACCESS_TAG

  If (myPKTID >= 223) Then
      myPKTID = 1
  Else
      myPKTID = myPKTID + 1
  End If

  longInterval = False

  'iRet = ActiveWaveAPI.rfCallTags(UInt16.Parse(1), rdrID,
      UInt16.Parse(0), UInt16.Parse(0), tagSelect, True,
      longInterval, Convert.ToUInt16(RdrCmdType),
            Convert.ToUInt16(myPKTID))
  'iRet = ActiveWaveAPI.rfQueryTags(UInt16.Parse(1), rdrID,
      UInt16.Parse(0), tagSelect, True, longInterval,
      Convert.ToUInt16(RdrCmdType), Convert.ToUInt16(myPKTID))
  iRet = ActiveWaveAPI.rfGetTagTemp(UInt16.Parse(1), rdrID,
      UInt16.Parse(0), tagSelect, True, longInterval,
Convert.ToUInt16(RdrCmdType), Convert.ToUInt16(myPKTID))
       End Sub

       Private Sub BtnAdd_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles BtnAdd.Click
  Patient.Show()
 End Sub

 Private Sub BtnAddHistory_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles BtnAddHistory.Click
  AddHistory.Show()
End Sub
```

```
  Private Sub dispEmSearch()
    If Me.InvokeRequired Then
        Me.Invoke(New MethodInvoker(AddressOf dispEmSearch))
    Else
        Dim i As Integer = 0
        Dim cn As New MySqlConnection(strCn)
        Dim cmd As New MySqlCommand("SELECT * FROM tracker", cn)

        Dim myReader As MySqlDataReader

        LvwPatient.Items.Clear()

        If cn.State = ConnectionState.Closed Then
            cn.Open()
        End If
        myReader = cmd.ExecuteReader()
        While myReader.Read
            tagID =
  LvwPatient.Items.Add(myReader.Item("PtID").ToString)
            tagID.SubItems.Add(myReader.Item("PtName").ToString)
            tagID.SubItems.Add(myReader.Item("LocName").ToString)
            tagID.SubItems.Add(Format(myReader.Item("CurrTime"),
        "yyyy-MM-dd         HH:mm:ss"))
        End While
        cmd.Connection.Close()
        If cn.State = ConnectionState.Open Then
            cn.Close()
        End If
        'coloring backgroung
        While i <= LvwPatient.Items.Count - 1
            If i Mod 2 = 0 Then
                LvwPatient.Items(i).BackColor = Color.Aquamarine
            Else
                LvwPatient.Items(i).BackColor = Color.White
            End If
            i = i + 1
        End While

    End If
  End Sub

  Sub dispEmSearchThread()
    dispEmSearch()
  End Sub

    Private Sub TmrDispSearch_Tick(ByVal sender As System.Object,
  ByVal e As System.EventArgs) Handles TmrDispSearch.Tick
    Thd = New Thread(AddressOf dispEmSearchThread)
    Thd.Start()
  End Sub

Private Sub DelTracker()
    Dim cn As New MySqlConnection(strCn)
    Dim cmd As New MySqlCommand("DELETE FROM tracker", cn)
    If cn.State = ConnectionState.Closed Then
        cn.Open()
    End If
```

```
    cmd.ExecuteNonQuery()
    If cn.State = ConnectionState.Open Then
        cn.Close()
    End If
End Sub

Private Sub BtnStop_Click(ByVal sender As System.Object, ByVal e As
  System.EventArgs) Handles BtnStop.Click
    TmrEmSearch.Stop()
      End Sub
  End Class
```