DEVELOPMENT OF WIRELESS SENSOR NETWORK FOR PUBLIC TRANSPORTATION

AMRAN BIN NAZARI

ELECTRICAL & ELECTRONICS ENGINEERING

UNIVERSITI TEKNOLOGI PETRONAS

SEPTEMBER 2012

Development of Wireless Sensor Network for Public Transportation

by

Amran bin Nazari

Dissertation submitted in partial fulfilment of
the requirements for the
Bachelor of Engineering (Hons)
(Electrical & Electronics Engineering)

SEPTEMBER 2012

Universiti Teknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

Development of Wireless Sensor Network for Public Transportation

by

Amran bin Nazari

A project dissertation submitted to the
Electrical & Electronics Engineering Programme
Universiti Teknologi PETRONAS
in partial fulfillment of the requirement for the
BACHELOR OF ENGINEERING (Hons)
(ELECTRICAL & ELECTRONICS ENGINEERING)

Approved:

____

(Dr. Azlan bin Awang)
Project Supervisor

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK

September 2012

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.

____

AMRAN BIN NAZARI

ABSTRACT

RFID (Radio Frequency Identification) and WSN (Wireless sensor network) have a wide variety of applications and huge potentials. RFID is used to provide detection and identification while WSN can monitor environment condition in a big area. Thus, integrating these two technologies will increase their capabilities and functionality. This project investigates the integration of RFID and WSN methodology to create a smart and reliable bus tracking system.

The author plans to create a system where the traffic of busses in a bus station can be displayed continuously. People can check whether the bus is on schedule or delayed. The estimated time of arrival of the next bus will also be estimated based on the bus schedule. This information will be transmitted to the wireless display around the bus station. Therefore, the main focus of this project is to develop a device that is able to monitor the device by integrating wireless sensor network with RFID. From this project, we could know the location and identity of a targeted vehicle.

ACKNOWLEDGEMENTS

This research project would not have been possible without the support of many people. The author would like to thank the Almighty God for giving him the strength and time to complete the project. The author would like to convey thanks to the Universiti Teknologi PETRONAS for providing the financial means and facilities. The author also wishes to express his gratitude for assistance, support and guidance of his supervisor, Dr. Azlan bin Awang. Special thanks also to all his graduate friends for sharing the literature and invaluable assistance. The author wishes to express his love and gratitude to her beloved families; for their understanding & endless love, through the duration of his studies.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Background of Study

With the revolution of technology in every aspect of any medium, there are many methods for bus service management to consider in monitoring the buses travel time and at the same time improves the service and customer satisfaction. Popular examples of technologies available are Global Positioning System (GPS) and Closed-Circuit Television (CCTV). The GPS receiver needs 4 satellites to estimate the bus position in 3-dimensions. The receiver identifies the location by estimating their distance from a satellite. Although the technique gives a very high precision, GPS receivers need a clear view of the sky or the signal will be attenuated. That is why GPS mainly used outdoors and they usually perform badly in forested areas, near tall buildings and inside buildings. This limits the functionality of GPS in a bus station and around the city. CCTV can be installed at every entry and image processing techniques can be used to detect the arrival of buses. Image processing techniques can also be performed to detect the bus on the road. However, tests have shown the technique is not reliable in tracking buses (~20% precision) [1].

Thus, RFID provides a good alternative to limitation of these technologies. In the 1940s, the US department of defence (DoD) developed the first RFID used to distinguish between friendly and enemy aircraft by using transponder [2]. Since then, the technology has been evolving and found many new applications to help daily life. Among them are Travel, Agriculture, Retail, Smart Plates & Edible RFID Tags, Navigation Systems for the Visually Impaired, Clothes, and Waste Disposal. RFID offers the tracking solutions needed in the bus station to detect every bus entry by using real-time tracking and identification [3]. This technique can be further improved by integrating RFID with WSN. By using technology the range of RFID detection will be increased significantly.

1.2 Problem Statement

Among reasons that affects the travel time of buses are heavy traffic, heavy rain and accidents. People have to wait patiently without knowing either the bus will be on time or not. The bus service management also usually prepares bus schedules in a trip sheets manually. This technique is time consuming and inaccurate. Although public transport is cheap, it is does not provide a quality and convenient service to their customer. Especially in terms of punctuality and number of service provided. All of these then results to dissatisfaction, stress and inconvenience among bus users every day.

They can be avoided with implementation of accurate real time bus monitoring system. This project has a huge potential to boost the customer experience and also public opinion on public transport that have been branded as unreliable. With this, customers can check for the bus availability and plan ahead their schedule at any time and any day.

1.3 Objective

The main objective of this study is to study and gain knowledge on wireless sensor network. The objectives of this study are as follows:

a) To build a working and low cost prototype to track vehicles.
b) To be able to send data from sensor to a computer.
c) To be able to display data on a computer.
d) To be able to estimate time and distance between user and vehicle.

1.4 Scope of Study

The scopes of this study are:

a) Study on wireless communication.
b) Study on sensors and its implementation using the prototype.
c) Study of a programming language.
d) Construct and design a user friendly interface program.
e) Analyse the result obtained.

1.5 Feasibility of the Project within the Scope and Time Frame

The project will begin by collecting information from books, journals and technical papers on wireless sensor networks, RFID, and its integration. Continuous research and consultation needed for better understanding on this project. A programming language for the project will be learned and applied. There are many products available in the market that is capable to produce the intended system.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction of Wireless Communication

In the early 20th century, Morse code is the first wireless signal transmitted in the world. As the technology grows, wireless communication has become an integral part of our daily life and.

Wireless communication is the exchange of data between multiple points without the need of wire. This technology can cover various distances from a few meters to thousands of kilometres. The best example available is a mobile phone. With the subscription number already reached 4.6 billion in 2010 [4]. The number will only keep on increasing each year. Some other examples are global positioning system (GPS), two-way radio, satellite television, wireless local area networks (LAN).

2. 2 Wireless Sensor Network

WSN (Wireless Sensor Network) has a huge potential and rapidly evolving each day as the number of its applications increasing each day. WSN can be used to monitor and also control a situation. It consists of sensor nodes that can operate automatically without human supervision for a long time and can be installed almost anywhere [5].

There are three main elements in sensor nodes which are detection, computing, and communication. All of the information is then collected by a sink node. Long range communication with a sink node uses multi-hop wireless connectivity to relay the data. Open source operating system (OS) designed specifically for WSN is used to support the task. The challenge for this project is to develop a low power communication with low cost WSN while taking into consideration of the limited battery life of the sensors.

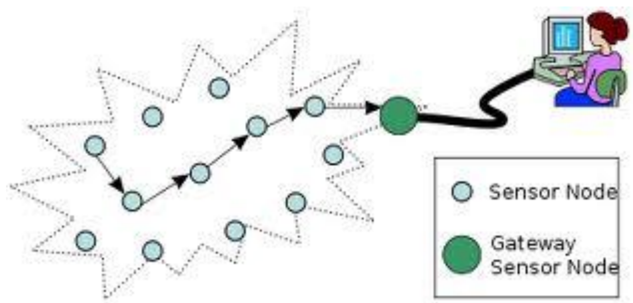

Figure 1: Multi-hop WSN architecture

2.3 Wireless Connectivity

Wireless connectivity has a huge advantage compared to other connections as they are more user friendly, easy to install, access and expand existing network. Three main wireless connections available are WI-Fi (IEEE 802.11 WLAN), Bluetooth (IEEE 802.15.1 WPAN) and ZigBee. Wi-Fi can support devices within a network up to maximum 128 devices and relay data up to 30 m, but require higher power and cost more compared to Bluetooth or ZigBee [6]. Bluetooth is a replacement of cable for mobile devices. Wireless personal area network (WPAN) is used as a protocol for Bluetooth and offer short-range communication. Bluetooth can support only 8 devices within a network and relay data up to 10 m.

ZigBee Standard offer cheaper system compared to Wi-Fi and Bluetooth [7]. ZigBee follows IEEE 802.15.4 as its base. ZigBee standard is divided into two main components which are Zigbee and IEEE 802.15.4. Zigbee specifies the application and network layers while the IEEE 802.15.4 specifies medium access control and physical layers. IEEE 802.15.4 normally operates in 2.4 GHz industrial, scientific and medical

(ISM) band. It can support data transmission up to 250 kb/s between 10 to 70 m. Zigbee can accommodate more than 65,000 nodes.

Performance	Zigbee	Bluetooth	Wi-Fi
Working frequency	2.4GHz 868/915MHz	2.4GHz	2.4Ghz
System resource	4Kbyte~32Kbyte	250Kbyte	1Mbyte
Communication range	0.1~1.5km	0.1km	0.1km
Data rate	250 Kbps	1 Mbps	11 Mbps
Maximum network nodes	65536	8	32
Wake-up time	30ms	10s	3s
Encryption	128 bits AES	128 bits	SSID
Low power consumption	Support	No support	No support

Table 1: Comparisons between wireless communications

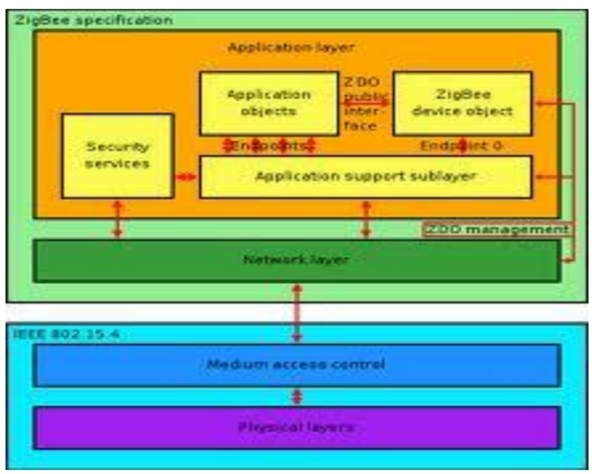

Figure 2: Zigbee Protocol

2.4 RFID Technology

RFID can identify a variety of objects because of its ability to track and get information from tagged item [8]. RFID consists of tags, reader, and a host computer.

RFID tag has an ability to store identification information and stores memory such as height, weight, type and temperature of tagged items. The tags can be attached or even planted into objects making it convenient for its user.

The tag can be either passive (no battery) or active (need batteries) with extra capabilities such as read or write and higher range. The passive tag reflects radio frequency (RF) signal transmitted from reader while the active tag able to send its own signal at a bigger range. Passive tag is only capable of one way communication, but active tag is capable of two way communication.

By using wireless transmission, an RFID reader can read and/or write information from tags. RFID tag cannot communicate with other RFID tag as it is in a single hop. An RFID reader transmits data gathered to host computer or RFID middleware. RFID middleware responsible in gathering and processing the information while the host computer analyses information obtained.

RFID tags operate in three frequency ranges. They are low frequency (LF), high frequency (HF), and ultra-high frequency (UHF). Low frequency tags are great for most applications and cheap. It has shorter reading ranges and low reading speeds compared to higher frequency tags. However, Fluids or metals do not affect LF tags unlike higher frequency tags. LF tags are usually used in two ranges of frequencies; 125–134.2 kHz and 140–148.5 kHz.

High frequency tags have medium reading speeds and ranges, but costs more than LF tags. The most common frequency used in HF is 13.56MHz. Ultra-high frequency tags have the highest reading speeds and ranges between all tags. UHF tags are the most expensive tag. They can go from 3 to 6 meters for passive tags and more than 30 meters for active tags. However, UHF tags are very sensitive to fluids and metals. UHF tags common frequency are 868MHz (Europe), 915MHz (USA), 950MHz (Japan), and 2.45GHz. Frequencies of UHF tags call for a permit and it vary between every country unlike frequencies of LF and HF tags which can be used freely worldwide.

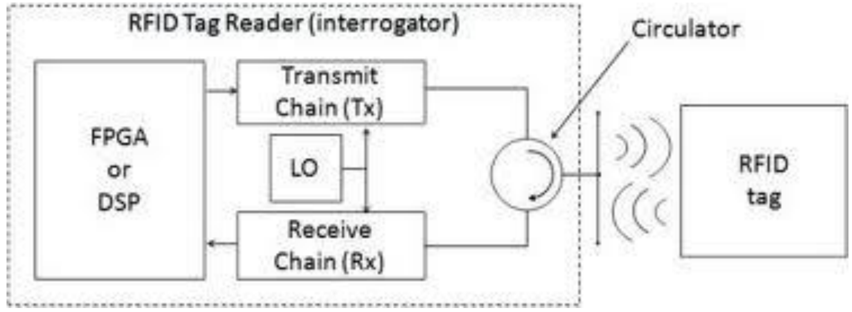

Figure 3: RFID System Block Diagram

2.5 Integration of RFID with WSN

After conducting some research in the integration of RFID and WSN, There were three main methods to consider. They are Integrated RFID Tags with Sensors, Integrating RFID Readers with Wireless Sensor Nodes, and Mix Architecture.

2.5.1 Integrated RFID Tags with Sensors

Integrated RFID tags with sensors are divided into two. It either can communicate only with RFID readers or can communicate with each other creating an ad hoc network. Main features and some examples of these two categories will be explained in detail.

2.5.1.1 Integrated Sensor-Tags with Limited Communicating Capabilities

Most of RFID tags have included sensors in their design. This ensures sensor reading and transmission process to a reader possible. However, The RFID tag performance when given sensing capabilities will not be satisfactory. The reader-tag system uses the same protocols and mechanisms for reading tag IDs and for collecting sensed data causing blurred line between the two networks.

Integrated sensor-tags work as normal RFID tags while integrated sensors are responsible to gather sensed information. The integration is based on changing the sensor signal by using Analog/Digital module and the subsequent data is forwarded by the readers to the base station [9]. There are many ways to integrate RFID sensor-tags. They can be further divided into three types of RFID integration; active, semi-active, and passive.

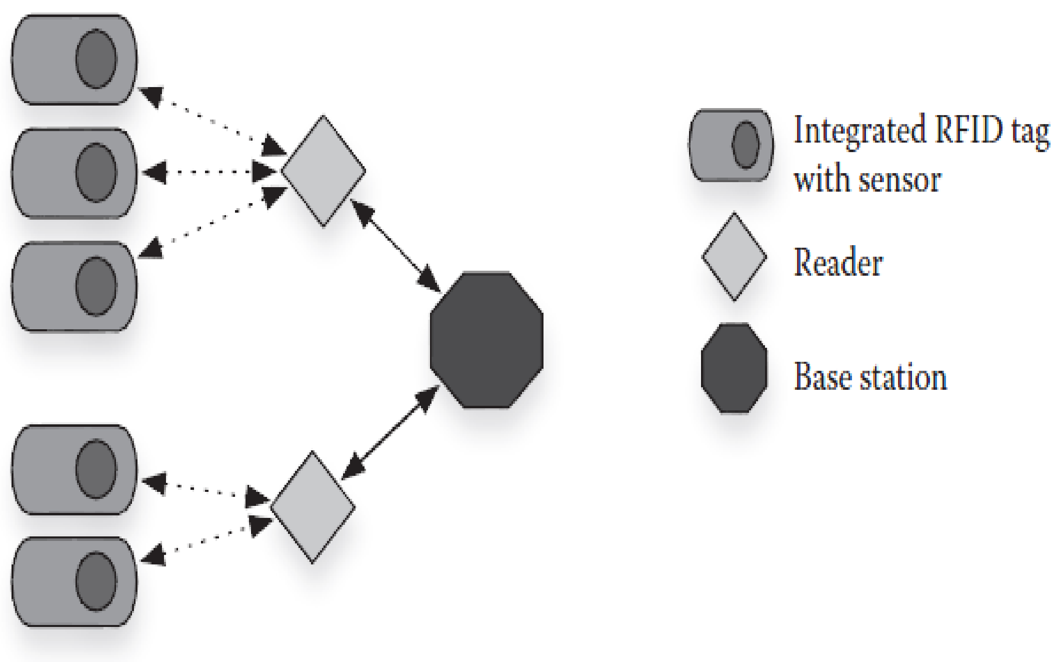

Figure 4: Integration Architecture of Sensor-tags, RFID Reader, and Base Station

2.5.1.1.1 Active Sensor-Tags

Batteries are used to power up active sensor-tags circuitry. It has a long range (around 30 m) high data and activity rates. Since the RFID sensor-tag uses battery to operate, it has limited lifetime, higher cost and more weight.

The first design example is by using a Sensor-Embedded Radio Frequency Identification (SE-RFID) System based on active RFID tags [10]. The system's sensors can model the external data individually and periodically without the need of a reader in the tag activation area. The systems are available in two different architectures. First, we can have multiple sensors rooted in one RFID tag. Next is by rooting each sensor in one RFID.

Some example of commercially available active sensor-tags are vibration sensor-tag (24TAG02V), temperature sensor-tag (24TAG02T) by Bisa Technologies. Next is TELID 310, temperature sensor-tag by Microsensys, and Callistro and Elara and lastly, temperature and humidity sensor-tags by Adage Solutions.

2.5.1.1.2 Passive Sensor-Tags

Passive sensor-tags powered by RFID readers. Thus, it does not have a limited lifetime, smaller and cheaper compared to active sensor-tags. These features can be exploited extensively for many purposes. However, for it to function properly, it has to be placed near to a RFID reader.

A design example by combining a passive sensor-tag temperature and photo sensors can be used for environmental monitoring [11]. An external ISM band of the RF signal is used to power up the sensor-tag. Next design has a long range of more than 10 m [12]. It operates at 2.45 GHz and 915 MHz ISM bands. It is made of a separated micro strip antenna and a passive voltage multiplying circuit.

Passive sensor-tags readily available developed by OKI, NYK Logistics Japanese, HILLS, Microsensys (TELID 210) and Alien Technology (ALB-2484).

2.5.1.1.3 Semi-Active Sensor-Tags

When RF power is enough to run the semi-passive sensor-tags, it behaves like a passive RFID tag. If not, it works by using batteries in a semi-active mode.

KSW-Microtec made semi-active RFID sensor-tag called VarioSens. The sensor-tag complies with ISO 15693 standard and runs at 13.56 MHz. It is an improved version of Tempsens. VarioSens has 1024 bytes memory (can hold 720 temperature readings) while TempSens has only 292 bytes memory (64 readings). Moreover, it provides increased data security and ability to define authorizations for reading, writing, and erasing data.

The other commercially available semi-active RFID sensor-tag are ThermAssureRF by Evidencia and SensIC RFID ASIC by Phase IV engineering Inc.

2.5.1.2 Integrated Sensor-Tags with Extended Communicating Capabilities

Integrated sensor-tags with limited communicating abilities can only communicate with RFID readers which restrict their abilities. To enhance the performance, we can integrate sensor nodes with RFID tags. As a result, interaction with each other as well as with other wireless devices is possible. This type of integration surpasses the communication limitation of previous integration type as they can form a cooperative ad hoc network for communication.

One of the integrated sensor-tag available in the market is iRFID tag by Machine Talker. iRFID tag is active tags with integrated build in sensors. It can sense various things, for example, light, temperature and vibration. It runs at 900 MHz and communicates via a proprietary air-interface protocol. iRFID tags automatically create a wireless mesh network and able to transfer data between themselves when they are in their own range with each other. iRFID tags can communicate up to 200 m. It is also possible for it to communicate with other devices through WiFi or wired networking protocols.

Another type of Integrated RFID sensor-tags available in the market are CoBIs RFID tag, Temp Tale RF-enabled (TTRF) by Sensitech , a WiFi-based active RFID tag by Aeroscout and An enhanced type of RFID tags by NTT lab.

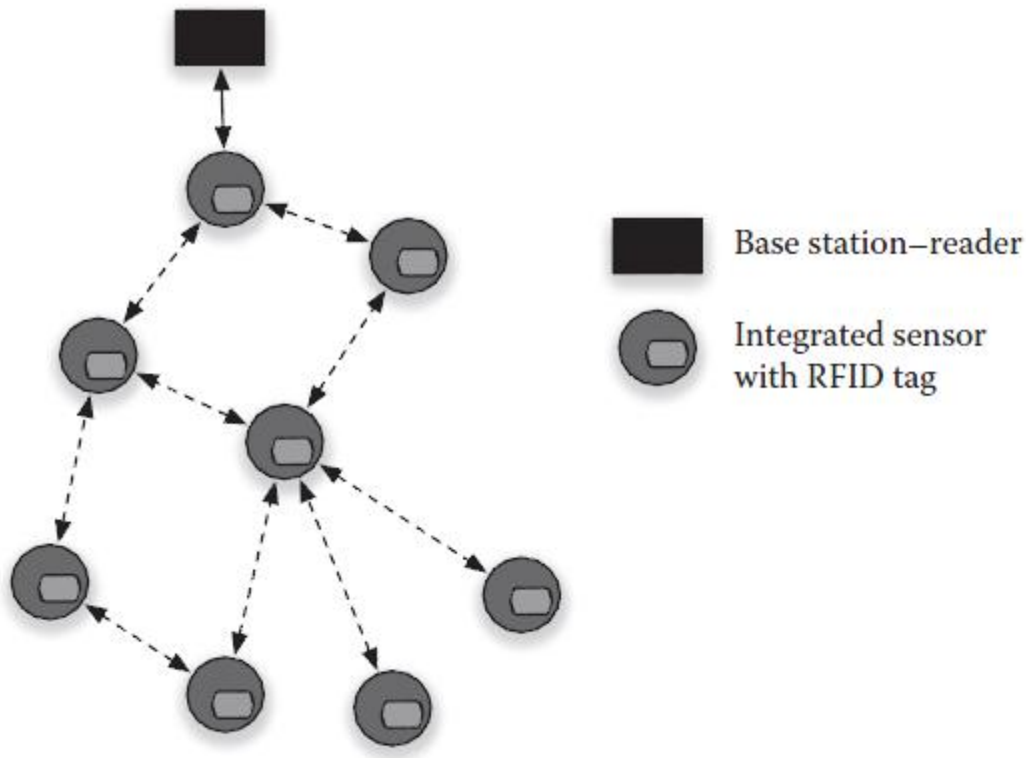

Figure 5: Integrated Sensor-Tags with Extended Communicating Capabilities

2.5.2 Integrating RFID Readers with Wireless Sensor Nodes

Integration can also be achieved by integrating RFID readers with sensor nodes. This method uses three types of devices which are sink-base station, integrated RFID readers-sensor nodes and RFID tags. This type of integration is also known as “a smart node.” Smart nodes can send data or configured as relay nodes of a WSN. Communication with each other is possible as they can create an ad hoc network. The integrated RFID reader-sensor node is capable to relay information to the right destination and work as a router. The smart nodes functions as an information/data collector from RFID tags in their communication range. The information/data will then be sent to the sink-base station. Here, all the data is collected and processed by an operator.

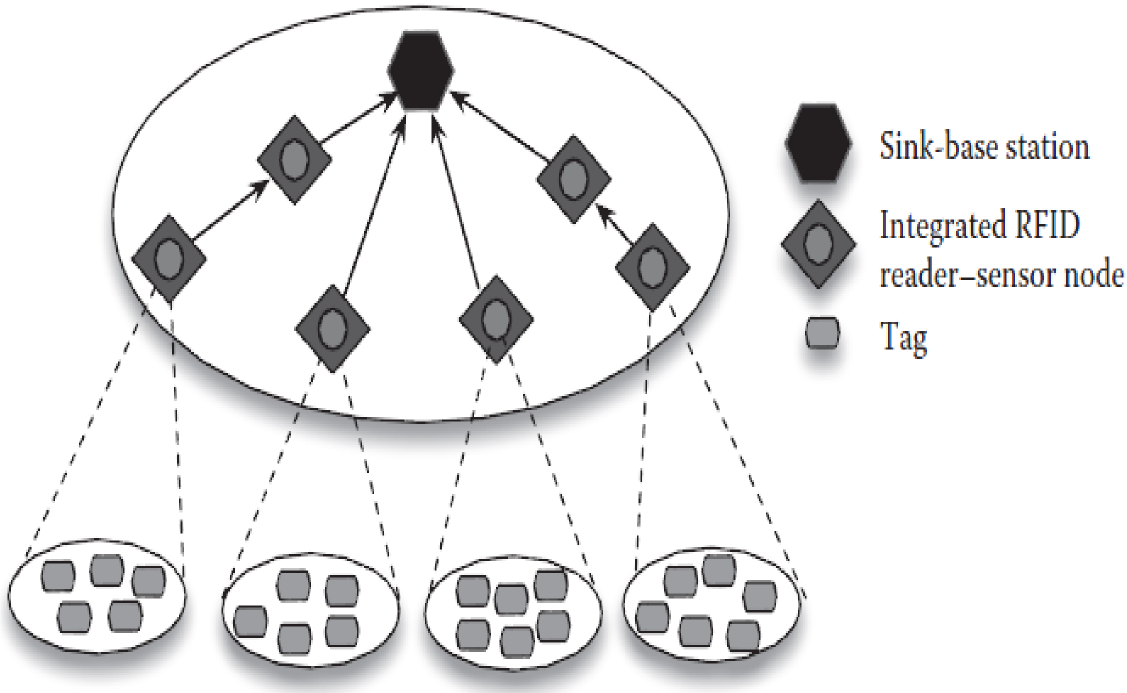

Figure 6: Integrated RFID Readers with Wireless Sensor Nodes

The restrictions present in normal RFID readers such as passive operation, mobility problems, and the antenna's position restrict their application possibilities. The problem previously faced does not present in smart nodes as they are smaller, cheaper, and have high mobility. But, since the communication pattern is from many to one, it will cause disturbance in the energy difference between smart nodes. Since the integration has a fixed communication range, the forwarded traffic will rise significantly because of shorter distance to the base station. Consequently, nodes near the base station will run out of batteries faster and causing a problem to the system [13].

To counter the problem, we can add more reader in the area close to the base station. However, as a result the cost of the system will be higher and causing more collisions. These problems can be overlooked as they enhance the lifetime of the system.

Another type of integration model enables data collection from RFID tags for a large distance. This will overcome the problem encountered in normal RFID system. The method is by connecting RF transceiver to RFID reader. This method can relay information between 100-200m. Thus, a new network node that able to correctly send information can be created. RF reader and an RF transceiver are needed at each node. Then, a microcontroller is added to the mix to organize function in every node. The full node configuration comprises of a microcontroller, an RF transceiver, an RF antenna, an RFID reader, an RFID antenna, and a battery. The MICA2 platform from Crossbow technologies can be used for this type of configuration.

Integrated RFID readers with wireless sensor nodes that commercially available are SkyeRead M1-mini by SkyeTek and ALR-9770 by AlienTechnology.

2.5.3 Mix Architecture

RFID tags and sensor nodes in the mix architecture are separated unlike previous method. They work individually but at the same time work together in an integrated network. The great thing about this method is that it does not require any hardware integration design. Because they are working separately, there is a chance for interference to occur in the system. The problem can be avoided but may result to increase in overhead.

This method uses three types of devices which are the smart stations, sensor nodes and RFID tags. A smart consist of an RFID reader, a microprocessor, and a network interface. Smart stations have no power limitations and can collect information from RFID tags and sensor nodes. It is then transmitted to a local host or to a remote LAN.

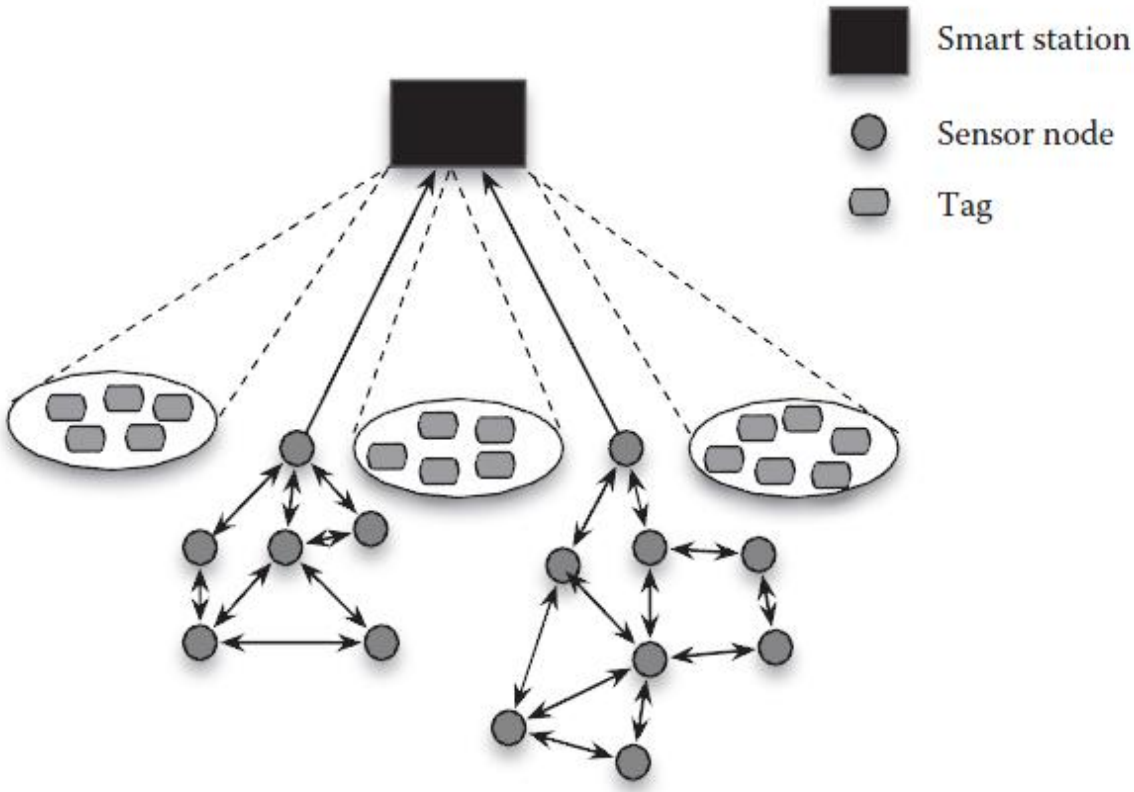

Figure 7: Mix Architecture

RFID tags and sensor nodes can send to the base station. Since smart stations have no power limitations, we can also set up Internet protocol architecture. As a result, smart station is capable to process data and routing and transporting protocols. One of the protocol examples is TCP. In addition, 802.11/WiFi communication protocol is deployable in various environments.

SARIF is an example of RFID and WSNs that use the mix architecture [14]. This system consists of an integration server, RFID networks, and a WSN. The integration server manages tasks of the WSN and the RFID network. The RFID network consists of an information server, RFID readers, and tags. The sensor network consists of a gateway and sensor nodes. Data associated with the RFID is sent by information server by communicating with the integration server. The integration server starts a task in the network based on the information obtained from the information server. The integration server can also assign tasks to RFID network.

There are a few products that can support the integration of RFID and WSNs according to the mix architecture on the market. One of the examples is RFID Anywhere. It has various useful features that support the architecture.

CHAPTER 3

METHODOLOGY

3.1 Procedure

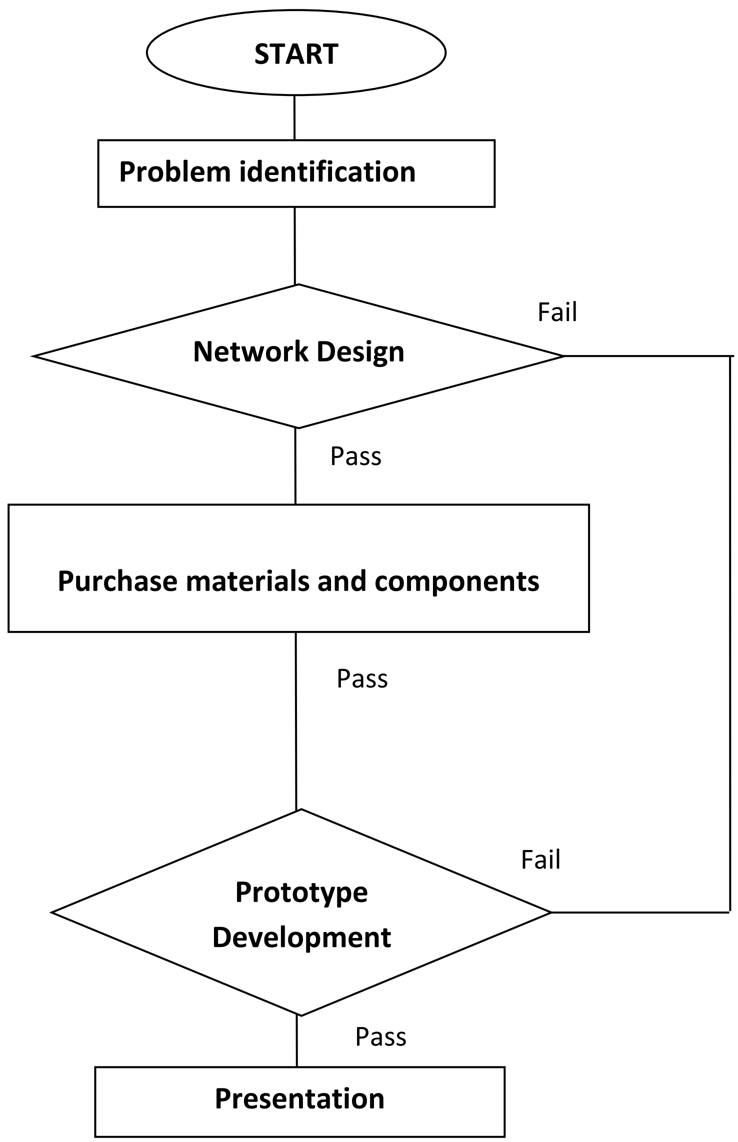

Figure 8: Project Flow Chart

3.2 Project Activities

The most important thing for this project is to identify which hardware to be used. There are a few criteria that have to be met which are cost, availability and compatibility. Next, is to design the network before the prototype can be developed. The result obtained from the prototype will be analysed and recorded.

3.3 Tools

3.3.1 Hardware

- RFID Reader
- FRID Card
- XBee Starter Kit with XBee module (SKXBEE)
- UART to RS232 converter
- Breadboard power stick
- Breakout USB Type A Female
- Breadboard
- Power adapter DC 12V
- Jumper male to male
- Jumper male to female
- Straight pin header 1x40 ways
- Soldering equipments
- Computer

RFID Reader RFID Tag SKXBEE



UART to RS232 converter Breadboard USB Type A Female



Breadboard power stick Breakout Power adapter DC 12V Jumper male to male


Jumper male to female Straight pin header 1x40 ways

Figure 9: List of Hardware

3.3.2 Software

- SKXBEE driver
  USB driver for SKXBEE

- XCTU 5.2.7.5
  Software to configure XBees

- Hyperterminal
  To set up hyperterminal in PC for displaying data received from the RFID reader

- Microsoft Visual Basic Express 2010
  Creating GUI (Graphical User Interface) for the hardware

3.4 Block Diagram of Overall System

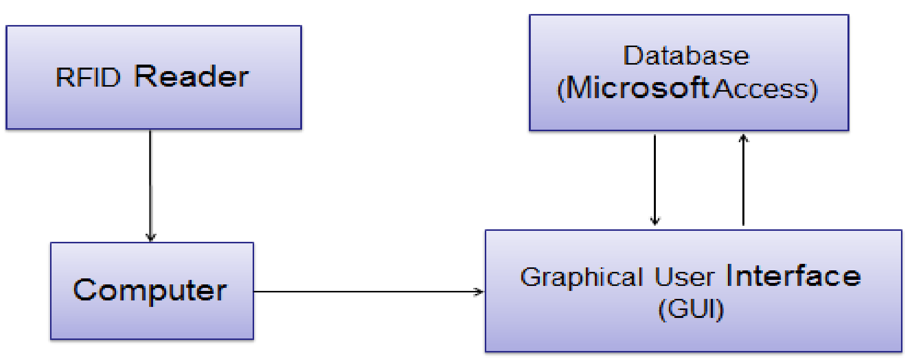

Figure 10: Block Diagram of Overall System

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Modelling

In designing the prototype, it is essential to design the system design first. Thus, designing a power efficient system is a priority. The protocol should also be rugged to be deployed in a large area.

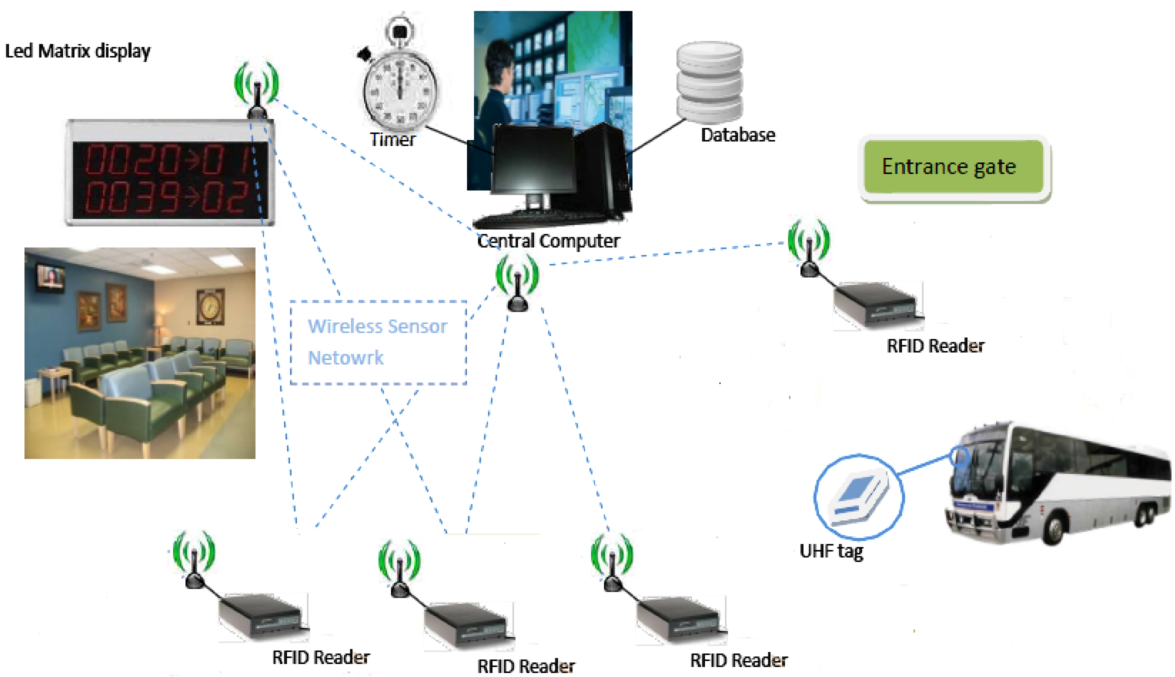

Figure 11: Integrated WSN design

The power consumptions of the wireless network need to be optimized. It can be done by reducing:

- Sending time
- Power while sending
- Listening time
- Switching time

Typically, Sending is the greatest power consuming process. This is because the power can be in between 10 - 100mW. By reducing the sending time it will greatly save overall energy consumption.

4.2 Prototype

The RFID reader is connected to a PC directly through RS232 using the DB9 (Male). As a result, tag's IDs scanned can be sent to the computer. Wireless RFID reader eliminates cable length constraint thus enable the RFID reader to be placed at a further distance from the PC. All the information from the scanned tag's IDs goes to a PC through the wireless link. This subsequently increases the RFID reader portability. The wireless module used in this project is an XBee starter kit because it simplifies the process of setting up the circuit of the wireless RFID reader. It consists of on board logic shifter that allows UART communication with 5V device and RS232 chip combined USB port that allows direct communication between computer and XBee. The prototype of the project is as shown below:

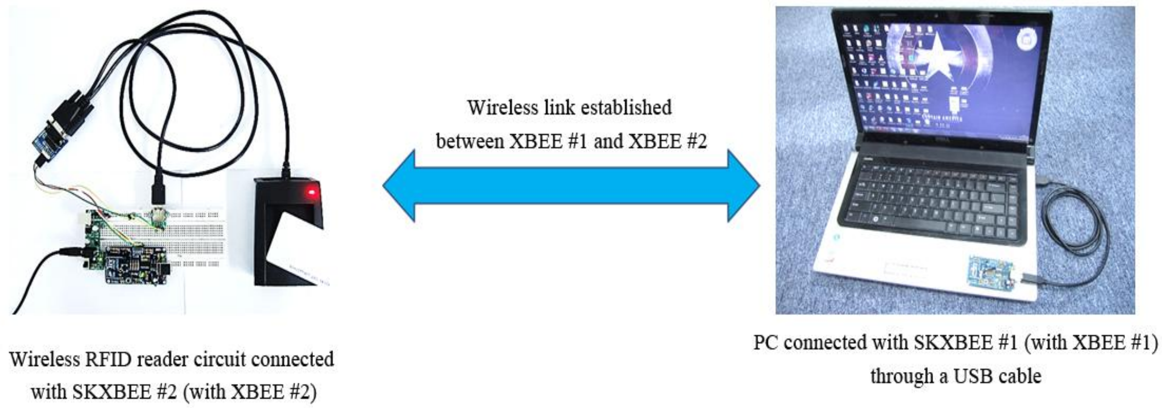

Figure 12: Flow Diagram of Integrated WSN

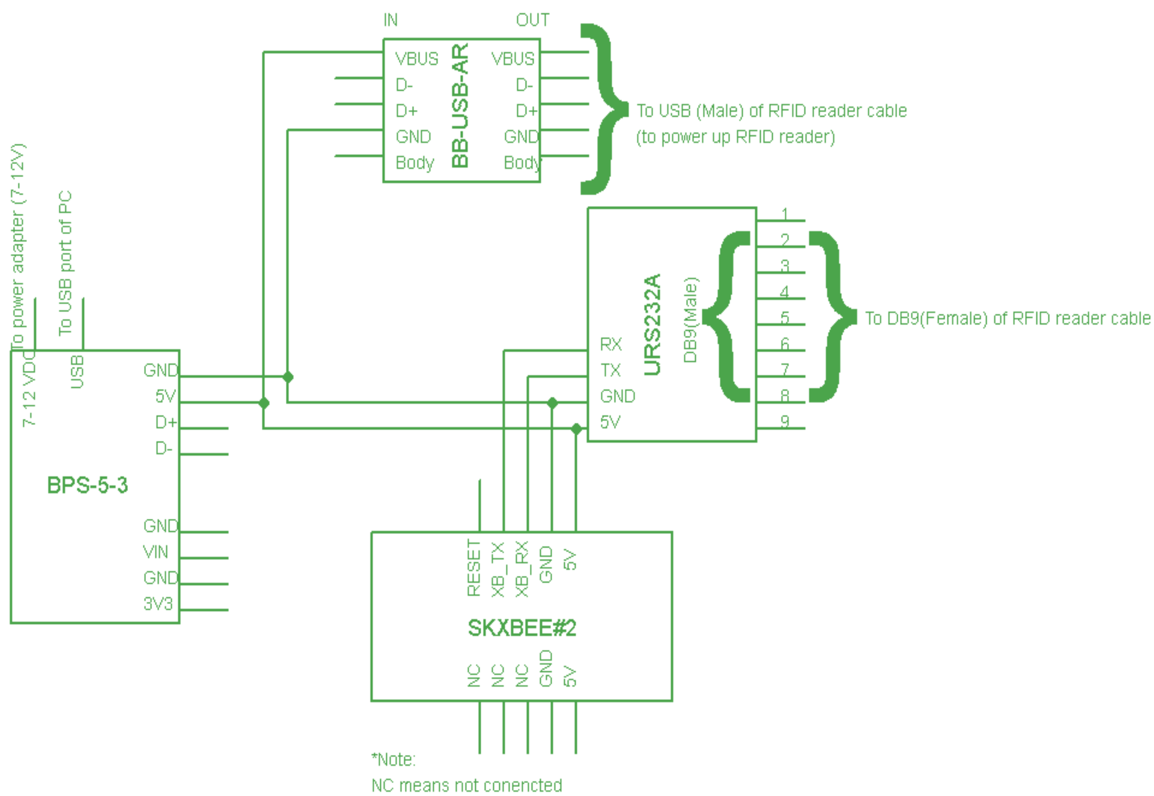

Figure 13: Circuit Schematic Diagram

The prototype capability is tested after setting up all the hyperterminal for both XBees using hyperterminal software. The RFID card ID was displayed in the hyperterminal as shown below which confirms the ID displayed to the ID of RFID card used in the trial.

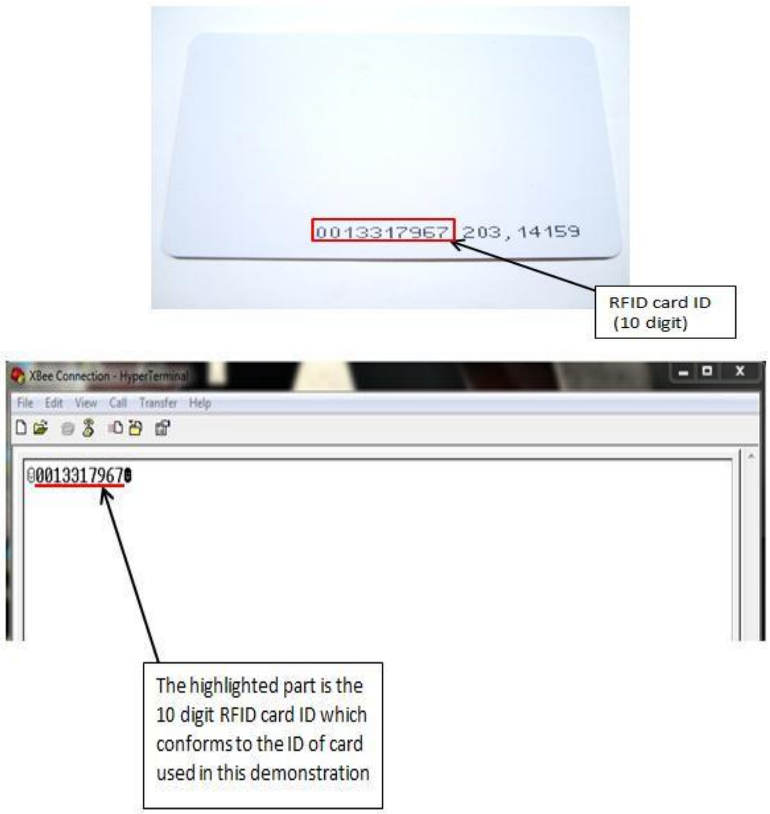

Figure 14: RFID ID Display

4.3 Graphical User Interface (GUI)

The Graphical User Interface (GUI) for this system is developed by using Microsoft Visual Basic Express 2010. All the coding done can be referred at Appendices section. The GUI is divided into three sections; welcome page, bus list and bus schedule. On the welcome page, the user cannot do anything and can only know the time and date located at the bottom right of the GUI. By clicking enter; the user will be guided into a new window (Appendix 1).

This window will display a list of busses available and their destination. For this GUI I put only four busses in the list. The user needs to select a bus service from the list and click on the select button to proceed. If the user tries to select more than one, the program will automatically change to the last selected item on the list. Plus, if the user tries to proceed without selecting any service available on the list, the program will display a pop out window which will remind the user to select a service (Appendix 2).

Figure 15: Welcome Page Window

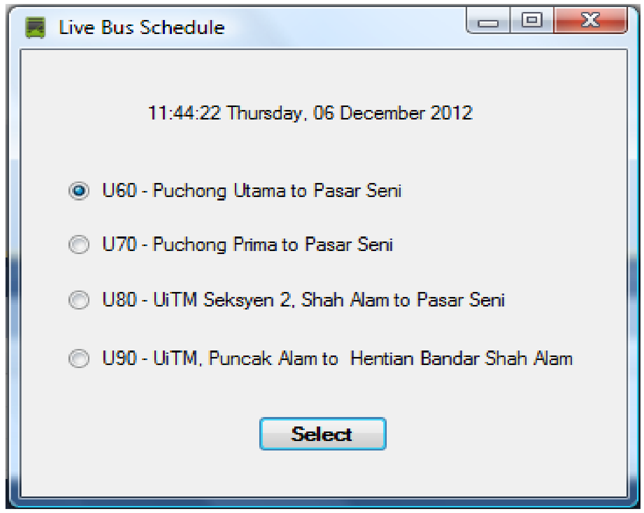

Figure 16: Bus List Window

The user can see some important information on the selected bus service in the bus schedule window. First, the user can see when the bus schedule and their status. The user will be notified if the bus departed or delayed at the status section. Next, the user can see when the next bus is coming. Both of the information is using a timer function to give real time updates on the bus status. Lastly, a time and date function for the user to cross check the time for the next bus arrival and the bus schedule in the list (Appendix 3).

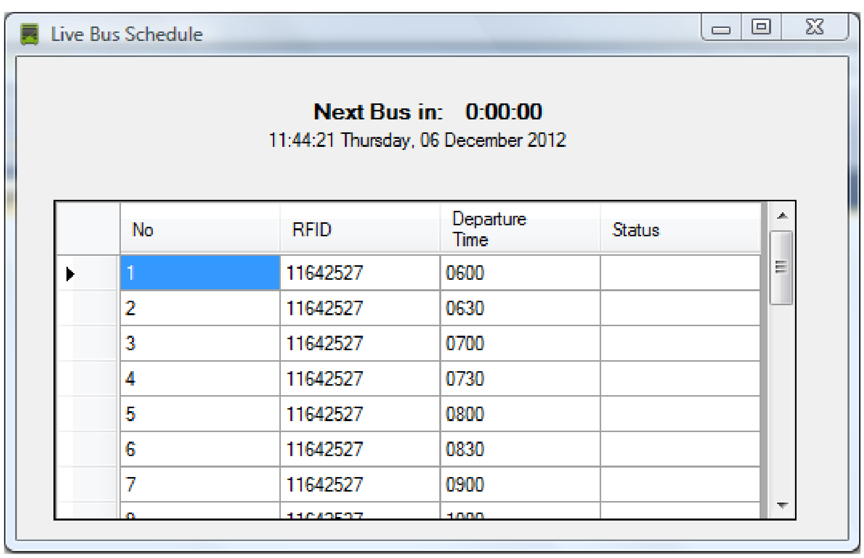

Figure 17: Bus Schedule Window

4.4 Discussion

This paper presents the design and implementation of a real time monitoring of buses, by integrating WSN and RFID. The architecture of the system is to provide the platform for flexibility and user customizations. There are a number of problems encountered throughout completing the project.

Firstly, the initial learning stage of the project when the author needs to learn and understand the Visual Basic software and configuring the hardware. Next, to develop an interface that allows communication between software and hardware. Therefore, in order to overcome the problems encountered, the author had to learn the basics of Visual Basic 2008 software by referring to the relevant textbooks available in the library and tutorials on the internet. The author also consults for advice from the expert in the programming language.

The proposed system succeeded in improving ways of obtaining bus traffic in a bus station by incorporating it with a new technology to a whole new level. It managed to display the schedule, bus status, estimated time for the next bus and transmits the RFID tag data wirelessly on a real time basis.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

The project objective is to develop systems that can track buses by using integrated WSN and RFID. The system should be able to display the bus schedule, status and estimated time of arrival for the next bus. The objective is achieved as the working prototype developed can perform the tracking and monitoring needed. The project is completed within the time frame. As a result, a more reliable and flexible system for bus monitoring is successfully developed.

5.2 Recommendations

A bigger and better system must be developed to eliminate the limitations present in the prototype. The system must be able to communicate within a large radius to be implemented in real life situation. A flexible database system of the bus schedule and better software must be developed to deal with the complex and unpredictable events. The system also should be able to be accessed everywhere using the internet. This will add a portability feature to the system existing prototype.

REFERENCES

[1] Cracknell, Mark; McCarthy, John; Renaud, Derek; , "Image detection in the real world," *Road Transport Information and Control - RTIC 2008 and ITS United Kingdom Members' Conference, IET* , vol., no., pp.1-4, 20-22 May 2008.

[2] Ollivier, M.M.; , "RFID-a new solution technology for security problems," *Security and Detection, 1995., European Convention on* , vol., no., pp.234-238, 16-18 May 1995.

[3] Sheng, Q.Z.; Xue Li; Zeadally, S.; , "Enabling Next-Generation RFID Applications: Solutions and Challenges," *Computer* , vol.41, no.9, pp.21-28, Sept. 2008

[4] Robust demand for mobile phone service will continue, UN agency predicts, *UN News Centre*, February 15, 2010.

[5] Culler, D.; Estrin, D.; Srivastava, M.; , "Guest Editors' Introduction: Overview of Sensor Networks," *Computer* , vol.37, no.8, pp. 41- 49, Aug. 2004.

[6] Baker, N.; , "ZigBee and Bluetooth strengths and weaknesses for industrial applications," *Computing & Control Engineering Journal* , vol.16, no.2, pp.20-25, April-May 2005.

[7] Baronti, P.; Pillai, P.; Chook, V.W.C.; Chessa, S.; Gotta, A.; Hu, Y.F.; , "Wireless sensor networks: A survey on the state-of-the-art and the IEEE802.15.4 and ZigBee standard", *Computer Communication*, Vol.30, pp 1655-1695, 2007.

[8] Suhong, L.; Visich, J.K.; Basheer, M.K.; Zhang, C.; , "Radio frequency identification technology: Applications, technical challenges and strategies", *Sensor Review*, Vol.23, pp 193-202, 2006.

[9] Hai Liu; Bolic, M.; Nayak, A.; Stojmenovic, I.; , "Taxonomy and Challenges of the Integration of RFID and Wireless Sensor Networks," *Network, IEEE* , vol.22, no.6, pp.26-35, November-December 2008.

[10] Hai Deng; Varanasi, M.; Swigger, K.; Garcia, O.; Ogan, R.; Kougianos, E.; , "Design of Sensor-Embedded Radio Frequency Identification (SE-RFID) Systems,"

Mechatronics and Automation, Proceedings of the 2006 IEEE International Conference on , vol., no., pp.792-796, 25-28 June 2006.

[11] Namjun Cho; Seong-Jun Song; Sunyoung Kim; Shiho Kim; Hoi-Jun Yoo; , "A 5.1-μW UHF RFID tag chip integrated with sensors for wireless environmental monitoring," *Solid-State Circuits Conference, 2005. ESSCIRC 2005. Proceedings of the 31st European* , vol., no., pp. 279- 282, 12-16 Sept. 2005.

[12] Kitayoshi, H.; Sawaya, K.; , "Long range passive RFID-tag for sensor networks," *Vehicular Technology Conference, 2005. VTC-2005-Fall. 2005 IEEE 62nd* , vol.4, no., pp. 2696- 2700, 25-28 Sept., 2005.

[13] Guangsong Yang; Mingbo Xiao; Chaoyang Chen; , "A Simple Energy-Balancing Method in RFID Sensor Networks," *Anti-counterfeiting, Security, Identification, 2007 IEEE International Workshop on* , vol., no., pp.306-310, 16-18 April 2007.

[14] Jaekyu Cho; Yoonbo Shim; Taekyoung Kwon; Yanghee Choi; , "SARIF: A novel framework for integrating wireless sensor and RFID networks," *Wireless Communications, IEEE* , vol.14, no.6, pp.50-56, December 2007.

APPENDICES

Appendix 1

Window 1: Welcome page

This is the main page for users to access the program.

```
Public Class Form1

   Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
      'ínitiate Timer 1 and will continually update the timer at every 1 second
      Timer1 = New Timer
      AddHandler Timer1.Tick, AddressOf Timer1_Tick
      Timer1.Start()
   End Sub

   Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
      'Open up Form 2 when clicked
      Form2.Show()
   End Sub

   Private Sub TextBox2_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles TextBox2.TextChanged
      'The clock is displayed in TextBox2
   End Sub

   Private Sub Timer1_Tick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Timer1.Tick
      'Obtain current time and arrange it to be displayed in "hour:minute:second day,
month, year"
      TextBox2.Text = DateTime.Now.ToString("hh:mm:ss dddd, dd MMMM yyyy")
   End Sub
End Class
```

Appendix 2

Window 2: Bus List

The user is given a choice to access information from one of four bus services available.

```
Public Class Form2

   Private Sub Form2_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
      'ínitiate Timer 1 and will continually update the timer at every 1 second
      Timer1 = New Timer
      AddHandler Timer1.Tick, AddressOf Timer1_Tick
      Timer1.Start()

   End Sub

   Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
      'Shows Selection of bus services available
      'Radio button allows user to select only one bus service at a time

      If RadioButton1.Checked = True Then

         Form3.Show()

      ElseIf RadioButton2.Checked = True Then

         Form4.Show()

      ElseIf RadioButton3.Checked = True Then

         Form4.Show()

      ElseIf RadioButton4.Checked = True Then

         Form4.Show()

      Else

      'If the user tries to proceed without selecting a bus service, a reminder will be given
         MsgBox("Please select a schedule")

      End If

   End Sub
```

```
   Private Sub TextBox1_TextChanged(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles TextBox1.TextChanged
       'The clock is displayed in TextBox1
   End Sub

   Private Sub Timer1_Tick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Timer1.Tick
       'Obtain current time and arrange it to be displayed in "hour:minute:second day,
month, year"
       TextBox1.Text = DateTime.Now.ToString("hh:mm:ss dddd, dd MMMM yyyy")
   End Sub
End Class
```

Appendix 3

Window 3: Bus Schedule

The user can access information of the bus status, current time, schedule and arrival time of the next bus in this window.

```
Imports System.Data.OleDb
Public Class Form3
   Private alarmTime As Date
   Private count As Integer = 0
   Dim data_get As String
   Private r As Integer = 1
   Dim txTest As String

   Private Sub Form3_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
      'TODO: This line of code loads data into the
'Database11DataSet._U60___Puchong_Utama_to_Pasar_Seni' table. You can move, or
remove it, as needed.

Me.U60___Puchong_Utama_to_Pasar_SeniTableAdapter.Fill(Me.Database11DataSet._
U60___Puchong_Utama_to_Pasar_Seni)
      'ínitiate Timer 1 and will continually update the timer at every 1 second
      Timer1 = New Timer
      AddHandler Timer1.Tick, AddressOf Timer1_Tick
      Timer1.Start()

      'Run SetALarmTime function
      SetALarmTime()

      SerialPort1.PortName = "COM17" 'Set SerialPort1 to the selected COM port at
startup
      SerialPort1.BaudRate = 9600 'Set Baud rate to the selected value
      SerialPort1.Parity = IO.Ports.Parity.None
      SerialPort1.StopBits = IO.Ports.StopBits.One
      SerialPort1.DataBits = 8
      SerialPort1.Open()'Open our serial port

      'ínitiate Timer 3
      Timer3.Start()
   End Sub

   Private Sub Countdown()
      Me.alarmTime = Date.Now.AddMinutes(1)
```

```
        'adds 1 minutes into timer 2
        Me.Timer2.Start()
    End Sub

    Private Sub Timer1_Tick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Timer1.Tick
        'Obtain current time and arrange it to be displayed in "hour:minute:second day,
month, year"
        TextBox1.Text = DateTime.Now.ToString("hh:mm:ss dddd, dd MMMM yyyy")
    End Sub

    Private Sub Timer2_Tick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Timer2.Tick
        If alarmTime < Date.Now Then
        'If the 1 minute is up, the timer is stopped and a message is displayed
            Me.Timer2.Stop()
            MessageBox.Show("Time's up.")
            If data_get <> "" Then
            'Update database "On Schedule"
                UpdateDatabase2(Me.r)
                data_get = Nothing
            Else
            'Update database "Delayed"
                UpdateDatabase(Me.r)
            End If

            If (Me.count < 19) Then
            'Set to how many times you want the timer to run
                SetALarmTime()
            End If
        Else
            Dim remainingTime As TimeSpan = Me.alarmTime.Subtract(Date.Now)
            'The timer will show a countdown in "hours:minutes:seconds" format
            Me.Label1.Text = String.Format("{0}:{1:d2}:{2:d2}", _
                                         remainingTime.Hours, _
                                         remainingTime.Minutes, _
                                         remainingTime.Seconds)
        End If
    End Sub

    Private Sub SetALarmTime()
        Me.alarmTime = Date.Now.AddMinutes(1)
        'Use this to determine your timespan use addseconds or addminutes methods
        Me.Timer2.Start()
        Me.count = Me.count + 1
    End Sub
```

```
    Private Sub serial_disconnect()
        SerialPort1.Close() 'Close our Serial Port
    End Sub

    Private Sub SerialPort1_DataReceived(ByVal sender As Object, ByVal e As
System.IO.Ports.SerialDataReceivedEventArgs) Handles SerialPort1.DataReceived
        ReceivedText(SerialPort1.ReadExisting()) 'Automatically called every time a data
is received at the serialPort
    End Sub

    Private Sub ReceivedText(ByVal [text] As String)
        txTest = [text]
        data_get = txTest
       'Store RFID ID number information as string
    End Sub

    Private Sub Timer3_Tick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Timer3.Tick
       'Continually try to get data from serial port
        If data_get <> "" Then
            Label3.Text = data_get
            If alarmTime < Date.Now Then
        'Display new data on Label3
                Label3.Text = ""
            End If
        End If
    End Sub
    Private Sub database(ByVal idx As Integer)
        'Establish connection with the database
        Dim ds As New DataSet()
        Dim con As New
OleDb.OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=|DataDirectory|\Database11.accdb")

        con.Open()
        Dim da As New OleDb.OleDbDataAdapter("SELECT * FROM [U60 - Puchong
Utama to Pasar Seni]", con)
        Dim cb As New OleDbCommandBuilder(da)
        'Update the database

        da.InsertCommand = cb.GetInsertCommand(True)
        da.Fill(ds)
        Dim delay As String = "Delayed"
        ds.Tables("Table").Rows(idx)(3) = delay
        da.Update(ds, "Table")
```

```
        con.Close()
        Me.r = idx + 1

Me.U60___Puchong_Utama_to_Pasar_SeniTableAdapter.Fill(Me.Database11DataSet._
U60___Puchong_Utama_to_Pasar_Seni)
    End Sub

    Private Sub UpdateDatabase(ByVal idx As Integer)
        Dim connetionString As String
        Dim connection As OleDbConnection
        Dim oledbAdapter As New OleDbDataAdapter
        Dim sql As String
        connetionString = "Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=|DataDirectory|\Database11.accdb"
        connection = New OleDbConnection(connetionString)
        sql = "update [U60 - Puchong Utama to Pasar Seni] set Status = 'delayed' where
[ID] = " & idx.ToString()
        Try
            connection.Open()
            oledbAdapter.UpdateCommand = connection.CreateCommand
            oledbAdapter.UpdateCommand.CommandText = sql
            oledbAdapter.UpdateCommand.ExecuteNonQuery()
            MsgBox("Row(s) Updated !! ")
            Me.r = idx + 1
           'Update the database
Me.U60___Puchong_Utama_to_Pasar_SeniTableAdapter.Fill(Me.Database11DataSet._
U60___Puchong_Utama_to_Pasar_Seni)
        Catch ex As Exception
            MsgBox(ex.ToString)
        End Try
    End Sub

    Private Sub database2(ByVal idx As Integer)
        'Establish connection with the database
        Dim ds As New DataSet()
        Dim con As New
OleDb.OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=|DataDirectory|\Database11.accdb")

        con.Open()
        Dim da As New OleDb.OleDbDataAdapter("SELECT * FROM [U60 - Puchong
Utama to Pasar Seni]", con)
        Dim cb As New OleDbCommandBuilder(da)
        'Update the database

        da.InsertCommand = cb.GetInsertCommand(True)
```

```
        da.Fill(ds)
        Dim onTime As String = "On Schedule"
        ds.Tables("Table").Rows(idx)(3) = onTime
        da.Update(ds, "Table")
        con.Close()
        Me.r = idx + 1

Me.U60___Puchong_Utama_to_Pasar_SeniTableAdapter.Fill(Me.Database11DataSet._
U60___Puchong_Utama_to_Pasar_Seni)
    End Sub
    Private Sub UpdateDatabase2(ByVal idx As Integer)
        'Establish connection with the database
        Dim connetionString As String
        Dim connection As OleDbConnection
        Dim oledbAdapter As New OleDbDataAdapter
        Dim sql As String
        connetionString = "Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=|DataDirectory|\Database11.accdb"
        connection = New OleDbConnection(connetionString)
        sql = "update [U60 - Puchong Utama to Pasar Seni] set Status = 'On Schedule'
where [ID] = " & idx.ToString()
        Try
            connection.Open()
            oledbAdapter.UpdateCommand = connection.CreateCommand
            oledbAdapter.UpdateCommand.CommandText = sql
            oledbAdapter.UpdateCommand.ExecuteNonQuery()
            MsgBox("Row(s) Updated !! ")
            Me.r = idx + 1
            'Update the database
Me.U60___Puchong_Utama_to_Pasar_SeniTableAdapter.Fill(Me.Database11DataSet._
U60___Puchong_Utama_to_Pasar_Seni)
        Catch ex As Exception
            MsgBox(ex.ToString)
        End Try
    End Sub
End Class
```