RFID WIRELESS DOOR SYSTEM

ASHVAANY EGAMBARAM

UNIVERSITI MALAYSIA PAHANG

"I hereby acknowledge that the scope and quality of this thesis is qualified for the award of the Bachelor Degree of Electrical Engineering (Electronics)"

Signature : ______________________________

Name : MOHD ZAMRI BIN IBRAHIM

Date : 6 NOVEMBER 2008

RFID WIRELESS DOOR SYSTEM

ASHVAANY EGAMBARAM

This thesis is submitted as partial fulfillment of the requirements for the award of the Bachelor of Electrical Engineering (Hons.) (Electronics)

Faculty of Electrical & Electronics Engineering
Universiti Malaysia Pahang

NOVEMBER, 2008

Signature : ______________________

Author : <u>ASHVAANY EGAMBARAM</u>

Date : <u>6 NOVEMBER 2008</u>

To my beloved mom

ACKNOWLEDGEMENT

The successful completion of my project can be attributed to the combined efforts made by the individuals I hereby acknowledge. First of all, I would like to express my heartfelt gratitude and thanks to my supervisor Mr. Mohd Zamri Bin Ibrahim, for encouraging me to move ahead and for being with me right from the beginning of the project and guiding me at every step. Last but not least, I also thank my parents and friends for being with me all the time and giving me moral support. Thank you.

ABSTRACT

The goal of this project is to develop a simple wireless door system using two different technologies. One of them is the RFID technology and the other one is an XBee Pro wireless device that conform the ZigBee standard. In most door applications, RFID is utilized as the primary technology for data identification and this data is used as a main source for controlling the operation of a door. In this project, RFID technology is also used for data identification, but the data obtained will be interfaced with XBee Pro wireless technology. Then the data will be sent wirelessly to the receiver XBee Pro which in turn controls the opening and closing of a door. So, the need for wired communication between the RFID system and the door is eliminated. Furthermore, the door can be controlled from long distance as wireless technology is implemented into the project. Intermediate between RFID and XBee Pro wireless technology is a microcontroller that works on the instructions build from the C programming language. The same applied between XBee Pro wireless technology and a door which is represented by a LED in the recipient part.

ABSTRAK

Matlamat projek ini adalah untuk membangunkan sebuah system pintu wayarles mudah yang menggunakan dua teknologi berlainan. Satu daripadanya ialah teknologi Pengenalan Frekuensi Radio (RFID) dan peranti wayarles XBee Pro yang mematuhi piawaian ZigBee. Dalam kebanyakan aplikasi-aplikasi pintu, RFID digunakan sebagai teknologi primer untuk pengenalpastian data dan data ini akan digunakan sebagai satu sumber utama untuk pengawalan operasi sebuah pintu. Dalam projek ini, teknologi RFID digunakan juga sebagai pengenalpastian data, tetapi data yang diperolehi akan digunapakai untuk berkomunikasi dengan teknologi wayarles Xbee Pro. Kemudian data akan dihantar secara wayarles kepada penerima XBee Pro yang akan mengawal pembukaan dan penutupan sebuah pintu. Oleh sebab itu, keperluan komunikasi berwayar antara sistem RFID dan pintu dihapuskan. Tambahan pula, pintu boleh dikawal daripada jarak jauh kerana teknologi wayarles dilaksanakan di dalam projek ini. Perantaraan di antara RFID dan teknologi wayarles Xbee Pro adalah satu mikropengawal yang melaksanakan kerja di bawah arahan daripada bahasa pengaturcaraan C. Mikropengawal juga digunakan di antara teknologi wayarles XBee Pro dan sebuah pintu yang diwakili oleh LED dalam bahagian penerima.

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

RFID	=	Radio Frequency Identification
LCD	=	Liquid Crystal Display
MHz	=	Megahertz
GHz	=	Gigahertz
RF	=	Radio Frequency
UART	=	Universal Asynchronous Receive Transmit
PIC	=	Programmable Intelligent Computer
V	=	Volts

LIST OF FIGURES

LIST OF TABLES

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

This chapter explains the background of Radio Frequency Identification (RFID) technology and XBee Wireless technology that is used in RFID Wireless Door System. The development of this project entirely depends with the assistance of RFID technology, microcontrollers, XBee PRO Wireless technology and programming language. The problem statement of this project, the objective of the project and project scopes are also elaborated in this chapter.

Radio Frequency Identification (RFID) technology is expanding rapidly with its applications in a wide range of area. RFID technology consist of RFID reader and RFID tags. RFID tags are also called RFID transponders and they are divided into passive and active RFID tags. In this project, passive RFID tag is used. This technology has become an intermediate in a wide variety of applications such as in industries as an instrument for identification, in automobile manufacturing and homeland security applications. The primary goal of RFID technology is to automatically identify data that are contained in electromagnetic fields. RFID tags do not require any physical contact with the reader for identification process. Most RFID tags are inexpensive and small where it derives its power from the signal produced by the RFID reader.

Wireless devices are becoming vital in our society. Since the beginning of the 21st century, the need for reducing wired connections and consequently the popularity of wireless communication has increased more than ever. Nearly everyone has a cellular phone. Companies are moving towards wireless devices and away from wired products. This has become a choice because people want to have the freedom to be mobile and not to be hard wired to infrastructure. First attempts in commercial computer wireless communication were already in the 80s of 20th century. It is no surprise that requirements gradually arose for commercially equitable, mutually compatible, standardized and possibly widely deployable devices. That is where the foundations of nowadays standards were made. The selection of the Radio Frequency (RF) communication modules used for the wireless transmission part in this project is based on several criteria. They are range of communication, power consumption, ease of integration and the cost. So, for this project I have chosen to use the XBee PRO wireless modules which conform to the IEEE 802.15.4 standard. The IEEE 802.15.4 wireless standard, commonly known as ZigBee is suitable for this project.

1.2 PROBLEM STATEMENT

If an entrance door is far from a main office, the door still can be controlled by a staff inside the office. When the staffs notice people by the door side waiting to come into the office through camera, the staff can place his or her ID card on the RFID reader placed inside the office, thus opening the door. This type of situation can take place when an outsider, customer or a client wants to enter the office. As they don't have any ID card to place on the RFID near the door, the door should be controlled from the office in order for them to enter the office.

1.3 OBJECTIVES

The main purpose of this project is to allow users to be identified securely without being intruded by anyone and transmission of data in long distances to control the opening of magnetic sensor doors. Thus, the objectives of this project are:

a. To develop a secure door security system using Radio Frequency Identification (RFID) technology
b. To develop a secure and effective transmission of data using XBee Pro wireless technology.
c. To design project using C programming language

1.4 SCOPE OF PROJECT

The scope of this project is:

a. Radio Frequency Identification (RFID) is used to trace a tag number and differentiate between a user and non-user.
b. Codes detected from the Radio Frequency Identification (RFID) will be verified and if a user is identified, signal will be sent through Xbee PRO wireless technology to control the opening and closing of magnetic sensor doors.
c. C programming language is used to program the PICs where methods of data comparison and data transmission are developed.

CHAPTER 2

LITERATURE REVIEW

2.1 INTRODUCTION

In this chapter, I have focused on the literature review and methodologies for the development of Radio Frequency Identification (RFID) Door System using Wireless Technology. In literature review, I will discuss on Radio Frequency Identification (RFID) and about Xbee Pro wireless technology that is used in RFID Wireless Door System.

2.2 RADIO FREQUENCY IDENTIFICATION (RFID)

RFID is known as Radio Frequency Identification System. This system is divided into two main components where one depends on the other. It has a RFID reader and a RFID transponder or tag. RFID transponder used in this project is a passive transponder where it doesn't use any internal power supply to activate it. When a RFID transponder is placed near a RFID reader, the reader reads information contained in the transponder without any physical contact. RFID reader will send an electromagnetic radio frequency (RF) signal to the transponder which energizes power inside the transponder. This power is sufficiently enough to sends back information on that transponder to the RFID reader to be processed. The RFID reader used in this project will operate with 5V DC power supply and have a RS232 serial interface with 9600 baud rate [1]. The operating frequency of the RFID reader is 125 KHz with 2cm reading range and 0.1s response time [1].

2.2.1 INTRODUCTION TO RFID

RFID has existed in some form or another for over 40 years. It is a method of automatically identifying an object by wirelessly retrieving information from small transponders, called RFID tags. These tags have an antenna built into them, which allows for reception of radio waves from an RFID reader. There are two types of RFID tags, which are active and passive RFID tags.

2.2.2 HISTORY OF RFID

The history of RFID can be directly related to a similar technology employed by Allies in World War 2 called Identification Friend or Foe (IFF). The function of this technology was to identify whether an incoming plane was a friend or a foe by using coded radar signals. These signals could trigger an aircraft's transponder, and a correct reply indicated a friendly military or civilian aircraft. After the war scientists and researchers began to explore the use of RFID to store and relay information. Radio Frequency Identification presented one major obstacle before it could become a feasible technology. It takes time for scientists and researchers to find a suitable power source for this technology. It took roughly thirty years for technology and research to produce internal power source for RFID tags.

2.2.3 TYPES OF RFID TAGS

As mentioned above, there are two types of RFID tags. Active and passive RFID tags are two different types of tags with their own characteristics. Both types of tags meet the requirements of users. Although both tags can be used with RFID reader, active RFID tag and passive RFID tag are fundamentally different technologies. Both the tags use radio frequency energy to communicate with RFID reader but the method of powering the tags is different. Active RFID tag uses a battery as an internal power source within the tag to continuously power up the tag and its RF communication circuitry, whereas passive RFID tag depends on RF energy transferred from the reader to the tag to power up the tag. Passive RFID tag response to the energy from the reader by absorbing and temporarily stores a very small amount of energy from the reader's signal to generate its own quick response. Thus, passive RFID tags require very strong signal from the reader to operate. The signal strength returned from the tag is limited to very low levels by the limited energy.

On the other hand, active RFID tags allows very low-level signals to be received by the tag because the reader does not have to power up the tag and the tag can generate high-level signals back to the reader. The energy is driven from its internal power source. Moreover, the active RFID tag is continuously powered, whether it is in the reader's field or not.

The communication range for passive RFID tag is limited by two factors. The first factor is passive RFID tag needs very strong signal to be received from the reader to power up the tag. The second factor is very small amount of power is available for the tag to respond to the reader. Both these factors limit the reader to tag distance. So, these factors limit passive RFID tags to operate in 3 meters or less. Depending on the frequency of operation, the communication range may be as short as a few centimeters. Active RFID tag can provide communication range of 100 meters or more.

Being cheaper to manufacture, most RFID tags are of the passive variety. Analysts predict that lowering costs and growing demand of RFID will eventually lead to widespread usage of RFID technology on global scale. The four most common tags in use are categorized by radio frequencies which are low frequency tags (125 or 134.2 kHz), high frequency tags (13.56 MHz), UHF tags (868 to 956MHz), and microwave tags (2.45 GHz).

Either passive or active RFID tag to be chosen for this project wholly depends upon its application for which it will be used. Since the project only needs to read tag from short distance, passive RFID tag is chosen for this project.

2.2.4 RFID REGULATIONS

There is no global body that has set RFID regulation, where every country has their own rules. Low Frequency and High Frequency RFID tags can be used globally without a license, but UHF may not be used globally due to lack of accepted standards. For example, in North America, limitations exist on UHF usage, specifically targeting transmission power [2]. These limitations are not accepted by France because UHF interferes with military bands [2]. There are also regulations for health and environmental issues. In Europe it is illegal to dispose RFID tags because of the possibility of damaging sensitive recycling machineries.

2.3 XBEE WIRELESS TECHNOLOGY

Radio Frequency (RF) communication modules used for the wireless data transmission portion of this project was based on a number of different criteria such as range of communication, power consumption, ease of integration, and cost. The wireless module that I chose for the project is the XBeePRO, which conform to the IEEE 802.15.4 standard and are offered by MaxStream, Inc. IEEE 802.15.4 is a low-rate communications protocol for wireless networks. The protocol is responsible for transferring data between two addressable devices. This protocol uses 900MHz or 2.4GHz (XBee Pro) bands and uses Direct Spread Sequence Spectrum (DSSS) that use very low power consumption for transmission. The maximum transfer rate of XBeePRO is up to 250Kbps at a range of 30 meters.

2.3.1 XBEE MODULE vs. XBEEPRO MODULE

Maxstream's ZigBee-based RF module comes in two types that is the XBee and the XBeePRO. The main difference between the two modules is the range of communication and the power consumption during transmission. Both XBee and XBee Pro modules are 20 pin devices with 2mm spacing. They use 3.3V as power supply for the modules. The XBeePRO is the more powerful of the two, consuming approximately 891mW of power during transmission and communication range up to 1.5km in open areas. The XBee, on the other hand, has more modest power consumption during transmission of approximately 149mW, but it also restricted to communication range of only 100m in open area. Both modules have similar power consumption during reception that is 165mW for the XBee and 182mW for the XBeePRO. As a result, I have decided to use the XBee PRO for my project.

2.3.2 ANTENNAS OF XBEE AND XBEE PRO

The Xbee Radio Frequency (RF) modules come in three different antenna option which are whip antenna, U.FL RF connector and chip antenna. The three antenna options are shown in Figure 2.1 below.

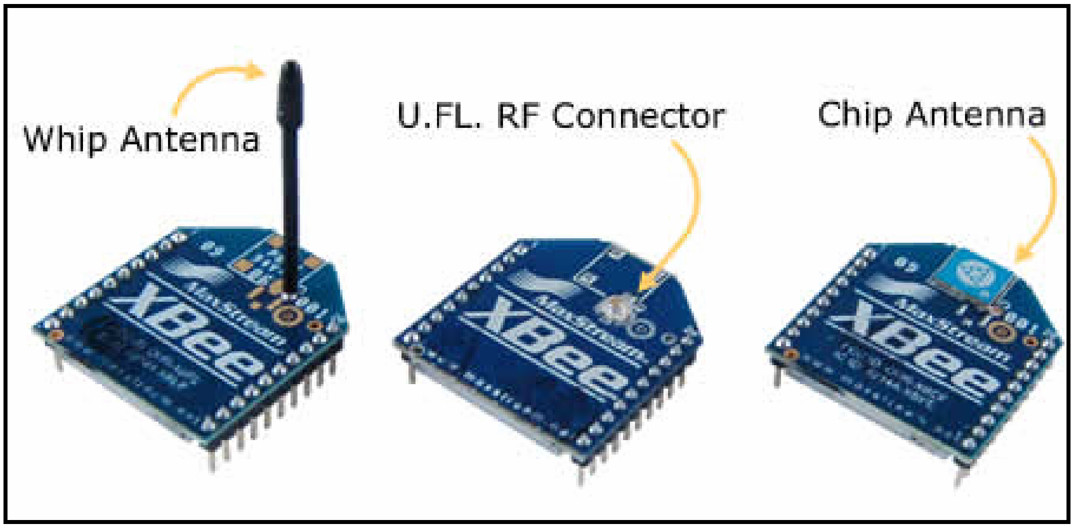
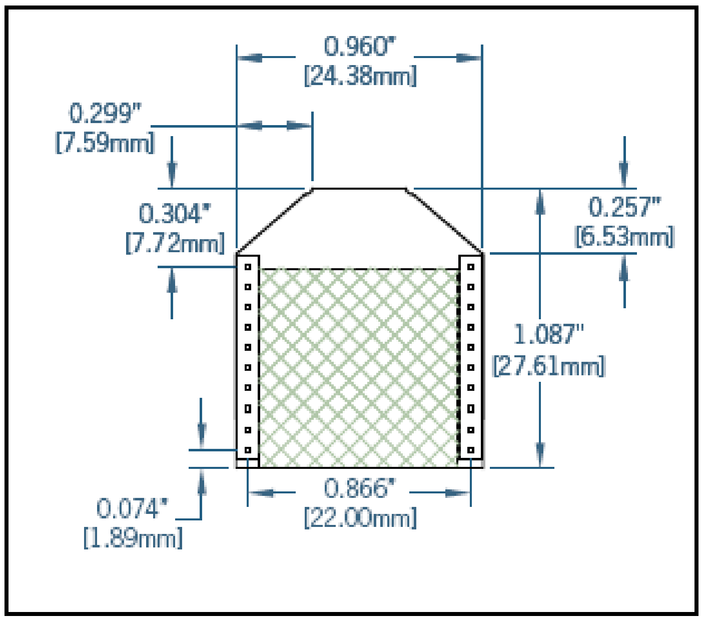
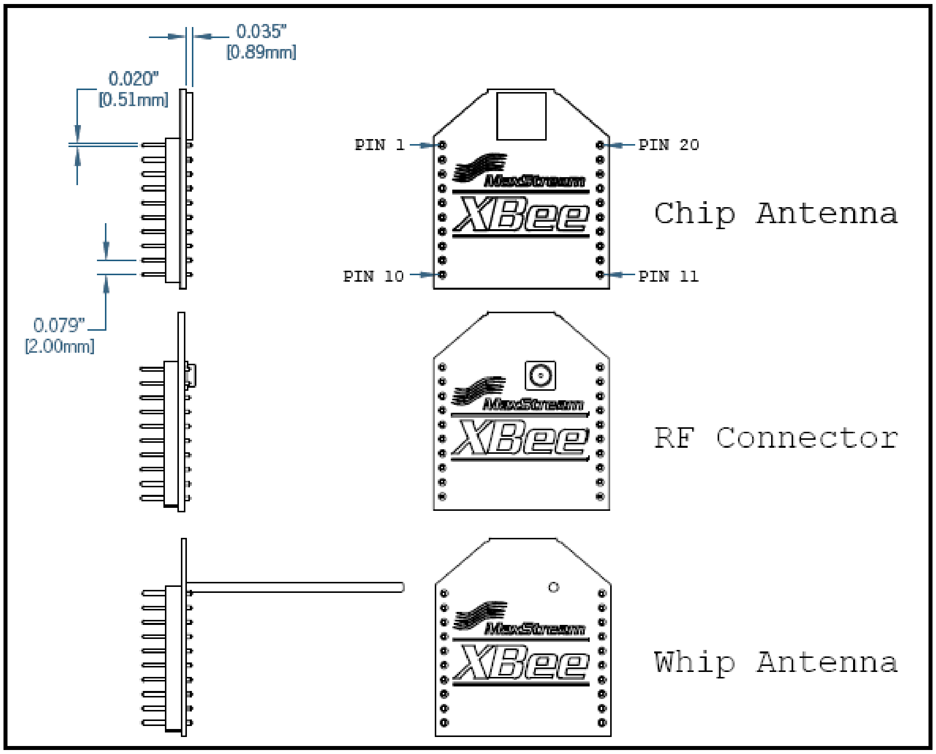


Figure 2.1: Antenna Option of XBee and XBee Pro

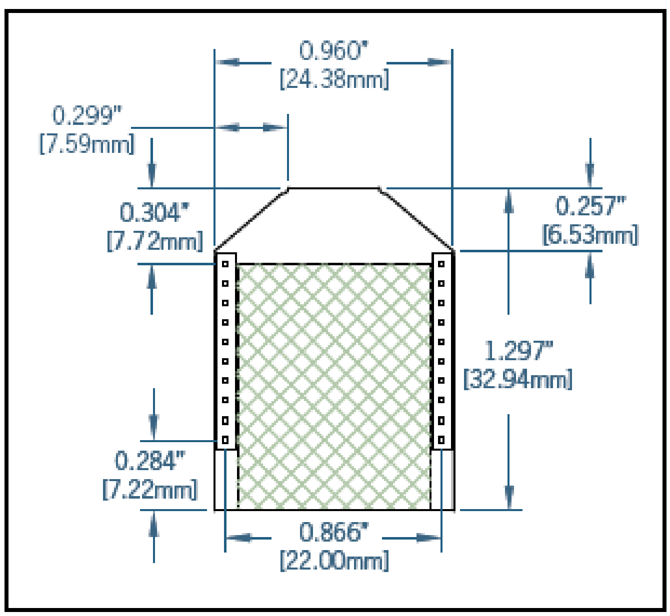
The whip and chip antenna comes readily integrated onto the actual XBee modules, whereas the U.FL RF connector type antenna can be used for connecting an off-chip dipole or other external antenna. Even though a whip antenna gives better performance than the chip antenna, I suspected that it might be harder to work with. Thus, I chose to use the Xbee Pro modules with integrated on-chip antenna. The mechanical drawings of the Xbee and XBee Pro modules are shown in Figure 2.2 below [3].



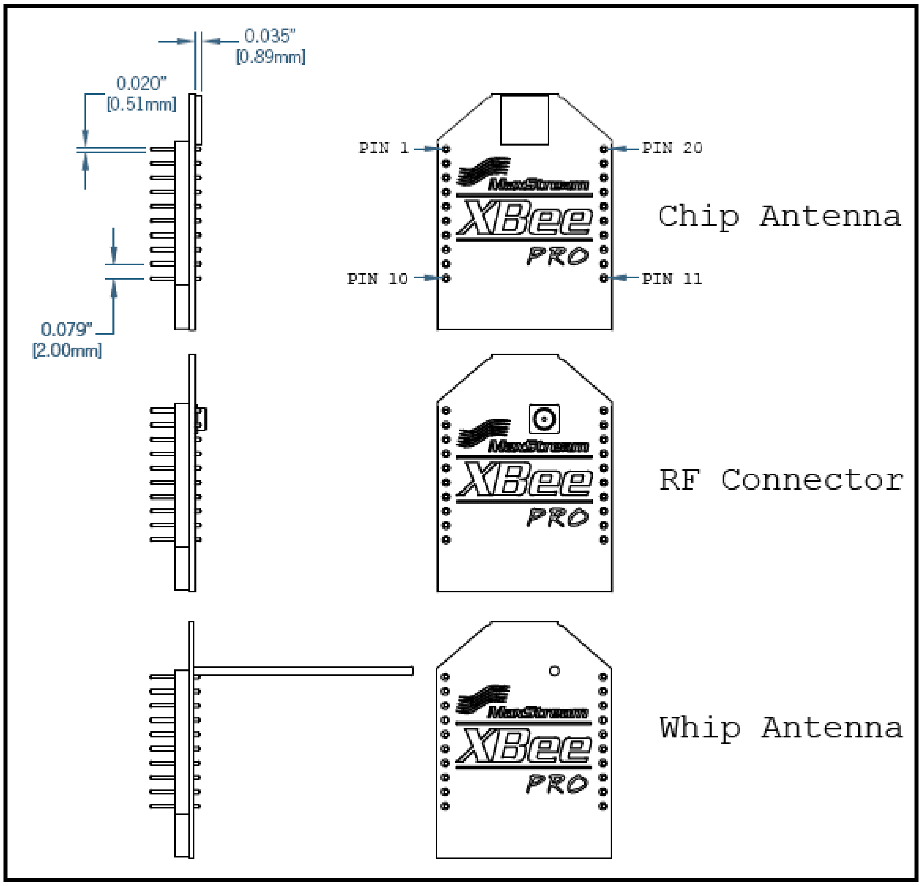
(a)


0.035"
[0.89mm]
0.020"
[0.51mm]
0.079"
[2.00mm]
PIN 1
PIN 20
PIN 10
PIN 11
MaxStream
XBee
Chip Antenna
RF Connector
Whip Antenna

(b)


0.960"
[24.38mm]
0.299"
[7.59mm]
0.304"
[7.72mm]
0.257"
[6.53mm]
1.297"
[32.94mm]
0.284"
[7.22mm]
0.866"
[22.00mm]

(c)



(d)

Figure 2.2: Mechanical Drawings of XBee and XBee Pro Modules

a) Bottom side of Xbee b) Top side of Xbee c) Bottom side of Xbee pro

d) Top side of Xbee Pro

2.3.3 DIFFERENCE BETWEEN XBEE PRO AND 433MHz RF

Many simple RF devices send data freely without considering occurrence of errors or collisions with the data sent from other devices. IEEE 802.15.4 is a fully implemented protocol that ensures data between devices do not collide as much as possible and the packets data arrive without errors. Addressing is a major benefit also.

2.3.4 UART

One of the main advantages of XBee and Xbee Pro is its UART (Universal Asynchronous Receive Transmit) serial interface. This interface makes them ideal for communication with a PC, as well as a PIC microcontroller. Essentially, when operating in its normal receive/ transmit mode, the Xbee and Xbee Pro serve as a wireless serial communication. Thus, in simple applications they can be used as a replacement for serial cable. They can also handle baud rates as high as 115 200.

2.3.5 MODES OF OPERATION FOR XBEE AND XBEE PRO

The Xbee and Xbee Pro have five modes of operation as shown in Figure 2.3 [4].

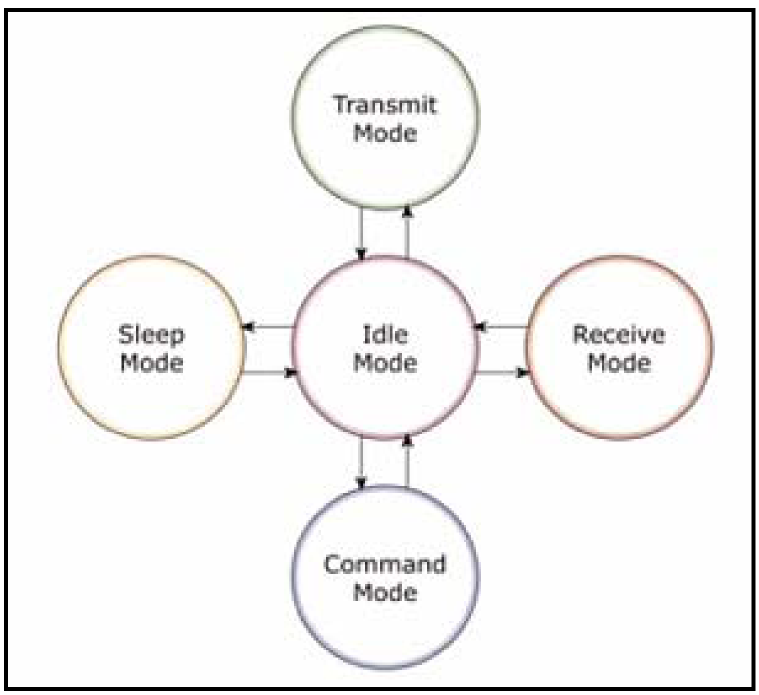


Figure 2.3: Xbee's Mode of Operation

The five modes of operations are idle mode, receive mode, transmit mode, command mode and sleep mode. If RF data comes in through a wireless port, the modules go into receive mode, buffering in the data and then forwarding it to be transmitted. The Xbee automatically switches to transmit mode and transmits the data wirelessly. The third mode a Xbee can enter is the command mode, which allows it to receive simple AT commands from the X-CTU software. This software is specially designed to change the Xbee's configurations such as destination address, baud rate, packet size, transmission channel and many more. The command mode makes the Xbee modules very easy to configure. Finally, the Xbee has a sleep mode, which has a number of different variations. The sleep mode is pin controlled and allows reducing the Xbee's power consumption to less than 33uW.

CHAPTER 3

HARDWARE IMPLEMENTATION

3.1 INTRODUCTION

This project is about inventing a wireless door system using Radio Frequency Identification (RFID) and XBee PRO wireless technology. The purpose of this project is to provide a better, effective and secure wireless door system. The opening and closing of the magnetic door can be controlled without any direct connection to the host system. It is also designed to function successfully when controlled from long distance. Thus, the system provides a wireless control system controlled by the host system.

The system is made up of four parts which are the identification of RFID tag, verification of detected code, transmission of signal when a user is identified and the opening and closing of magnetic sensor doors according to people. Data identification, verification and transmission of signal through XBee PRO wireless module are considered to be the host system of this project. After the identification process, data identified will be compared with data saved earlier in the PIC. The data will be verified whether it is user or non-user. If a user is identified, a signal will be transmitted through Xbee PRO wireless technology.

When a signal is received on the receiving end, it will produce an output on the magnetic sensor door to open it. The receiving end via XBee PRO wireless module is the secondary system of the project. An advantage of this project is the magnetic sensor doors can be controlled from long distance without direct wire connection from the host system.

When a user places his or her identity card on the RFID reader, the RFID reader reads the tag number and sends it to the microcontroller. Inside microcontroller, the code received from the reader and codes saved in it will be compared, consequently displays user name and the tag ID on the LCD display. A signal is then sent through the XBee PRO wireless technology which controls the opening of magnetic sensor door. A block diagram of the system is shown in Figure 3.1.

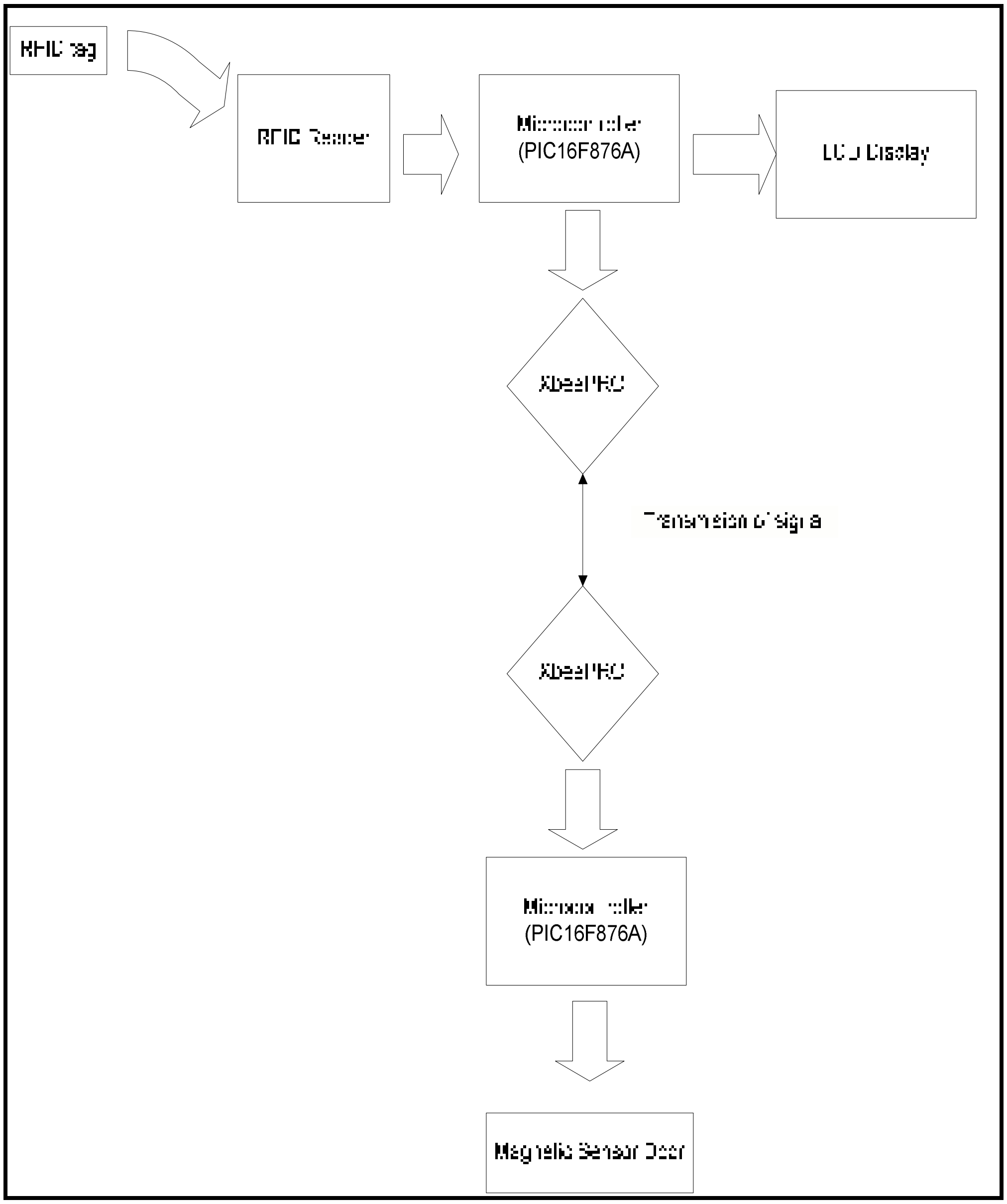
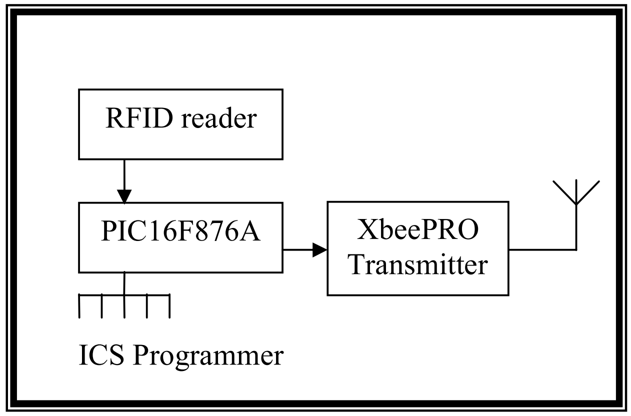
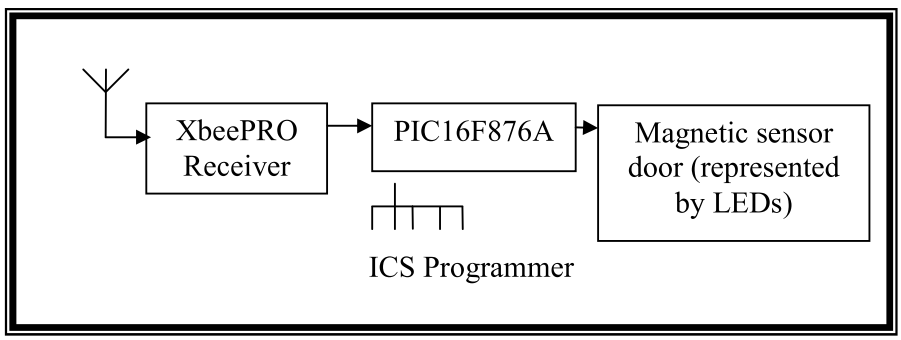


Figure 3.1: Basic Block Diagram of the RFID Door System using Wireless Technology

Figure 3.2 below shows how components are connected to complete this project:



(a)



(b)

Figure 3.2: Diagram of the RFID Wireless Door System a) The host system, b) The secondary system

First, both PICs 16F876A at the host and secondary system are programmed using In Circuit Serial Programmer, (ICSP). The PIC at the host system will be programmed with all users' identification codes. When a user places the identity card on the RFID reader, the RFID reader reads the tag code and sends it to the microcontroller PIC16F87A. Inside the microcontroller, the code received from the reader and identification codes preprogrammed earlier will be compared, consequently displaying user name and the tag ID on LCD display. If the code received from RFID reader is equal to one of the preprogrammed codes, a signal is transmitted through XBeePRO wireless module. The XBeePRO wireless module at the secondary system will receive signal from host system and controls the opening and closing of the magnetic sensor door. Figure 3.3 shows the hardware flow chart for this system.

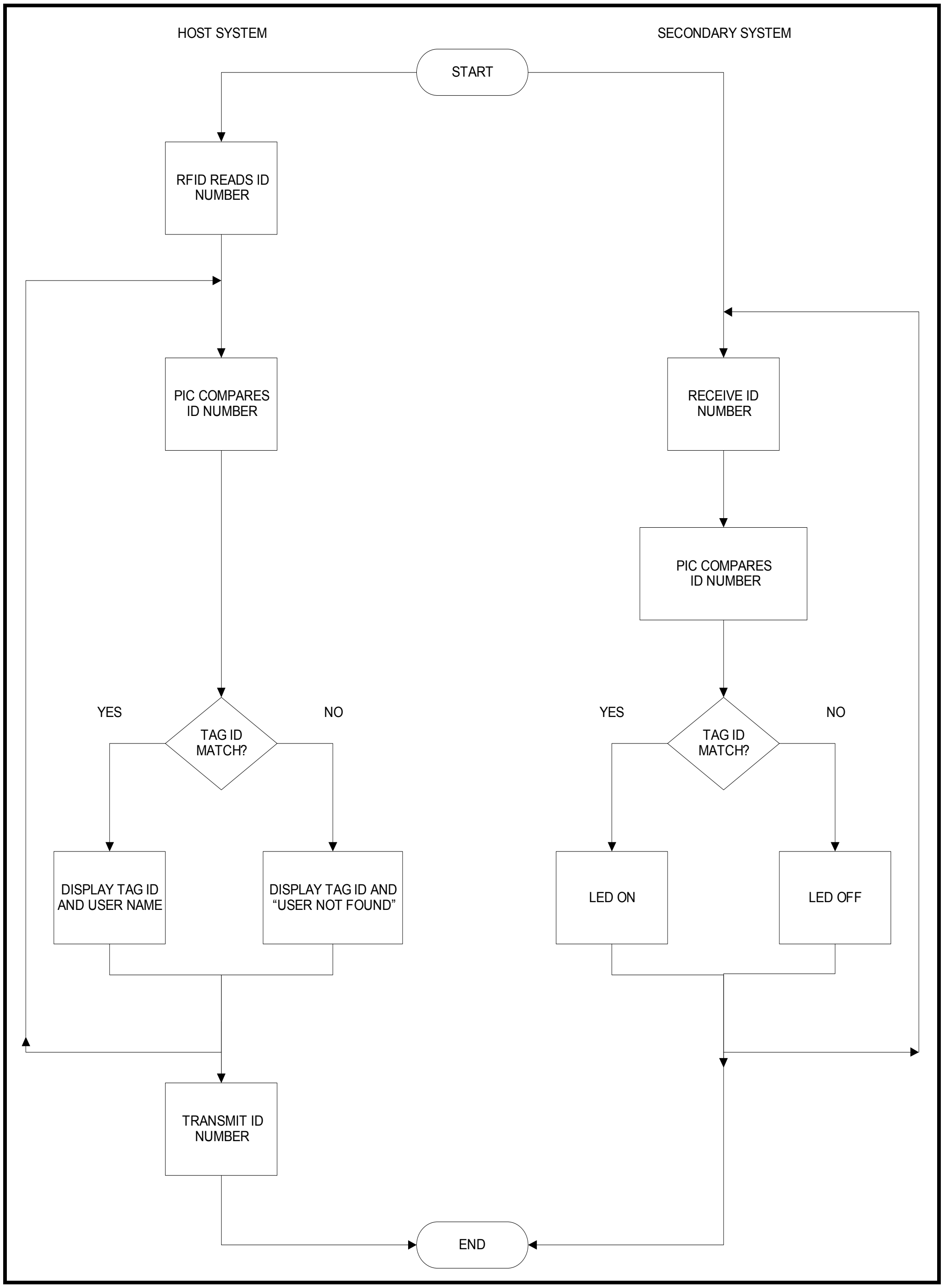


Figure 3.3: Hardware Flow Chart of RFID Wireless Door System

3.2 COMPONENT'S LIST

Table 3.1: Components Used For the Project

PIC16F876A	28 PINS
2X16 LCD DISPLAY	
XBEE PRO	
ADAPTER SOCKET(DC)	
BUZZER	4.7KΩ
DIODE	IN4007
DIODE	IN4148
LED	GREEN
LED	RED
CRYSTAL	20MHz
CAPACITOR	22pF
CAPACITOR	0.1uF
CAPACITOR	100uF
VOLTAGE REGULATOR	LM1117
VOLTAGE REGULATOR	LM7805
PUSH BUTTON	
RESISTOR	330Ω
RESISTOR	4.7kΩ
RESISTOR	1kΩ
RESISTOR	10kΩ
TRANSISTOR	2N2222
IC BASE	28 PINS
SCREWS & NUTS	3MM
PCB HEADER	
HEAT SINK	
RFID READER	
RFID TAGS	

PIC16F876A

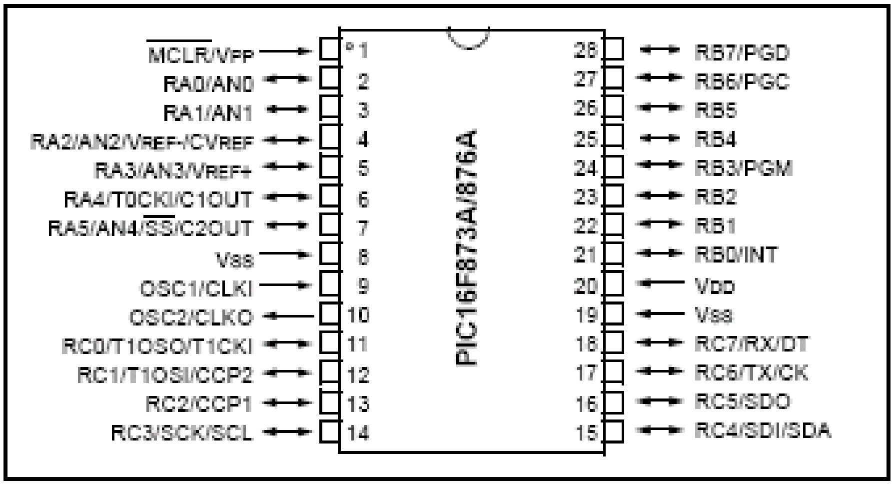


Figure 3.4: Pin Layout of PIC16F876A

Table 3.1 lists all components used in this project. The microcontroller used to operate circuits in this project is Microchip's PIC16LF876A. The pin layout is shown in Figure 3.4 [5]. The CMOS flash-based 8-bit microcontroller is a high-performance Reduced Instruction Set Computer (RISC) consisting of only 35 single word instructions. The microcontroller supports both digital and analog inputs, allowing easy interfacing with digital and analog sensors. The programming language used to program the PIC is very similar to standard (ANSI) C. This made developing the software an easier task since I have taken prior coursework involving the C programming language. A summary of features given by the manufacturer include: 256 bytes of EEPROM data memory, self programming, an In-Circuit Debugger (ICD), 2 Comparators, 5 channels of a 10-bit Analog-to-Digital (A/D) converter, 2 capture/compare/PWM functions, a synchronous serial port that can be configured as either 3-wire Serial Peripheral Interface (SPI) or the 2-wire Inter-Integrated Circuit (I²C) bus, and a Universal Asynchronous Receiver Transmitter (USART) [6].

The PIC16F876A was chosen, in particular, because of its low power consumption, small size and memory size of 256 bytes EEPROM data memory. This microcontroller has a wide operating voltage range of 2.0V to 5.5V. The LF model typically draws a supply current of only 1.6mA. Also, the available 28-pin package allowed the overall circuit design to be small and compact.

3.3 HARDWARE OF TRANSMITTER (HOST SYSTEM)

The hardware portion of the transmitter side, that is for the host system involves various types of modules in order for the system to function or operate. They are the UIC00A Programmer module, the power module, the reset module, the output's module, the LCD module, the RFID module, the XBee Pro transmitter circuit module and the PIC's module itself. All these modules have to be combined and connected into one single circuit to operate the system. The complete circuit of the host system is included in the Appendix A for reference.

3.3.1 UIC00A PROGRAMMER MODULE

Figure 3.5 on the next page shows the hardware connections of UIC00A programmer module that need to be soldered on the board in order to load program onto microcontroller's memory.

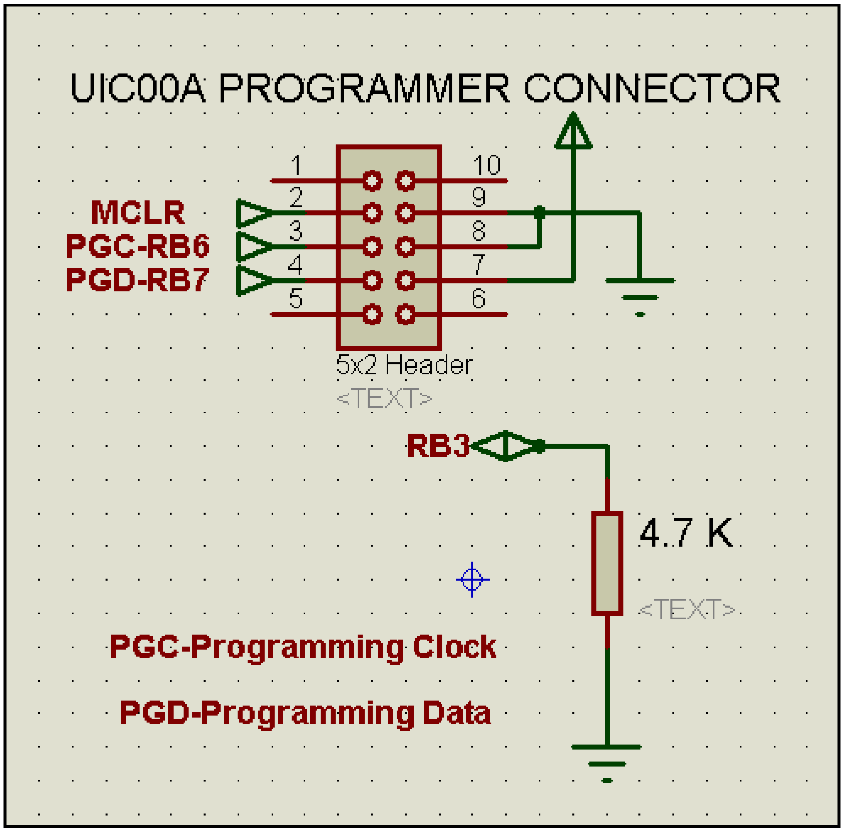


Figure 3.5: UIC00A Programmer Module

As shown in the figure above, pins MCLR, RB6 and RB7 of PIC16F876A should be connected to the In-Circuit Programmer (UIC00A) to program the PIC microcontroller. Pin RB3 is pulled low to disable low voltage programming as the programmer is using high voltage programming. Pin MCLR of microcontroller is connected so that when applied, the programmer goes into programming mode. The power and ground pins are connected to supply voltage and grounding respectively. Programming clock (PGC) and Programming data (PGD) are connected to pins RB6 and RB7 respectively. PGC is a unidirectional synchronous serial clock line from programmer to the microcontroller. PGD is a bidirectional synchronous serial data line.

3.3.2 POWER MODULE

Figure 3.6 below shows a power module connected to the host system:

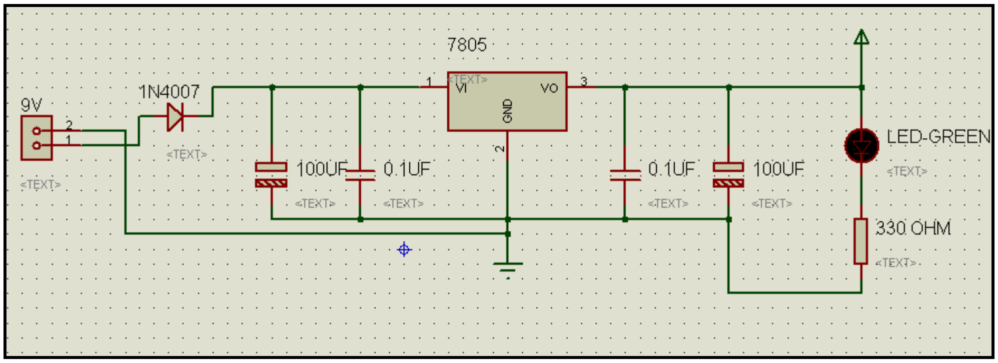


Figure 3.6: Power Module

The microcontroller should be connected to a power module as shown above. The power module supplies a regulated +5V to power up the system. A 9V is supplied to the system and to be regulated by the LM7805 to produce a regulated +5V. The diode 1N4007 is connected as shown to protect the circuit from wrong polarity supply. The capacitors are connected to stabilize voltage at the input and output side of the LM7805 voltage regulator. On the other hand, the LED is connected to indicate the power status of the circuit and the 330 ohm resistor is to protect the LED from burning.

3.3.3 RESET MODULE

Reset module for the host system is shown in Figure 3.7 below:

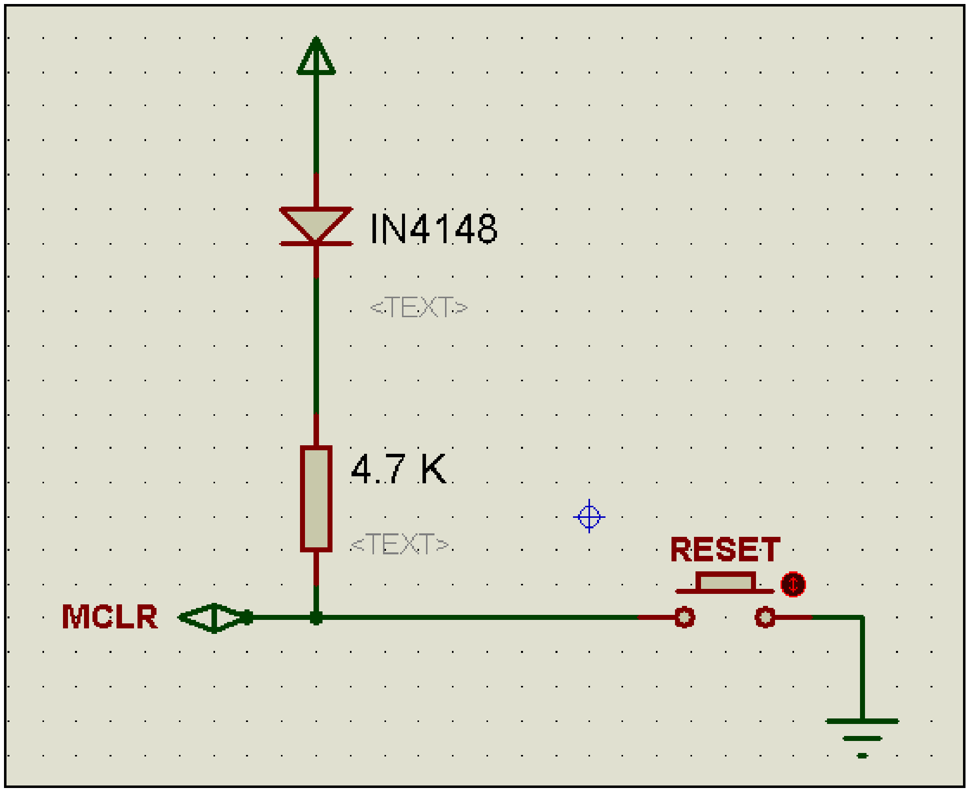


Figure 3.7: Reset Module

The reset module allows the microcontroller to be initialized properly.

3.3.4 OUTPUT'S MODULE

Figure 3.8 below shows the connections used to connect all the outputs for the host system:

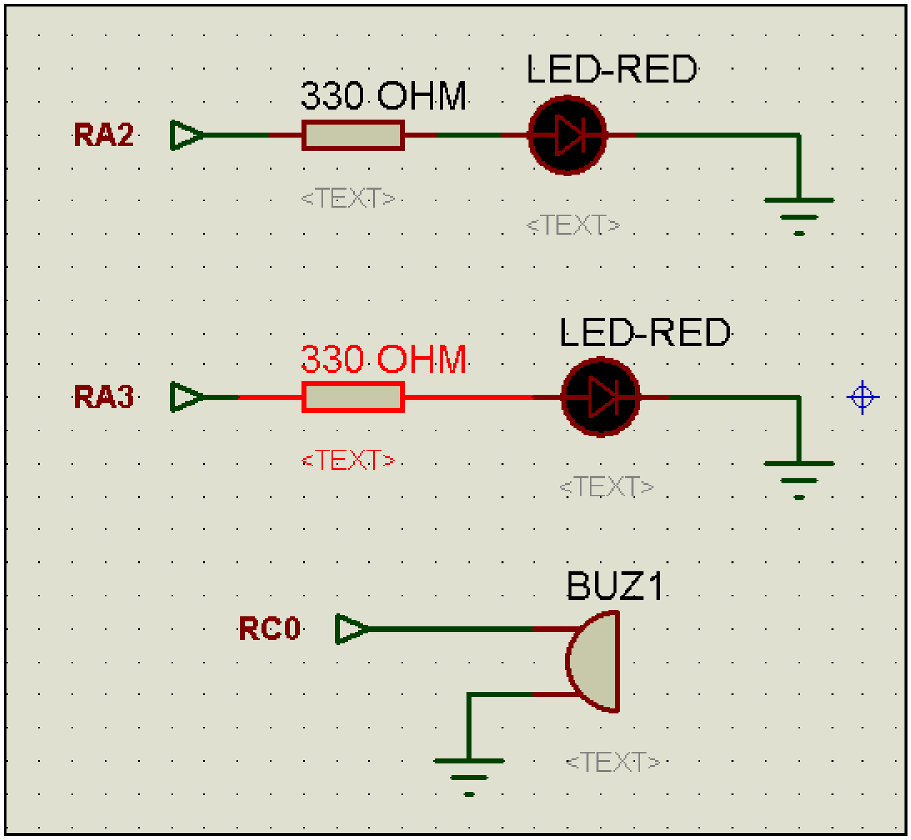


Figure 3.8: Transmitter's Output module

The LEDs are connected to pin RA2 and RA3 of microcontroller respectively to represents outputs. The function of the 330 ohm resistors is to protect the LEDs from over current that will burn the LEDs. When the output is logic 1, the LEDs will ON, while when the output is logic 0, the LEDs will OFF. The buzzer is connected to pin RC0 of the microcontroller. The buzzer will be activated when logic 1 is sent and deactivated when logic 0 is sent.

3.3.5 LCD MODULE

The LCD's module is shown in Figure 3.9 below:

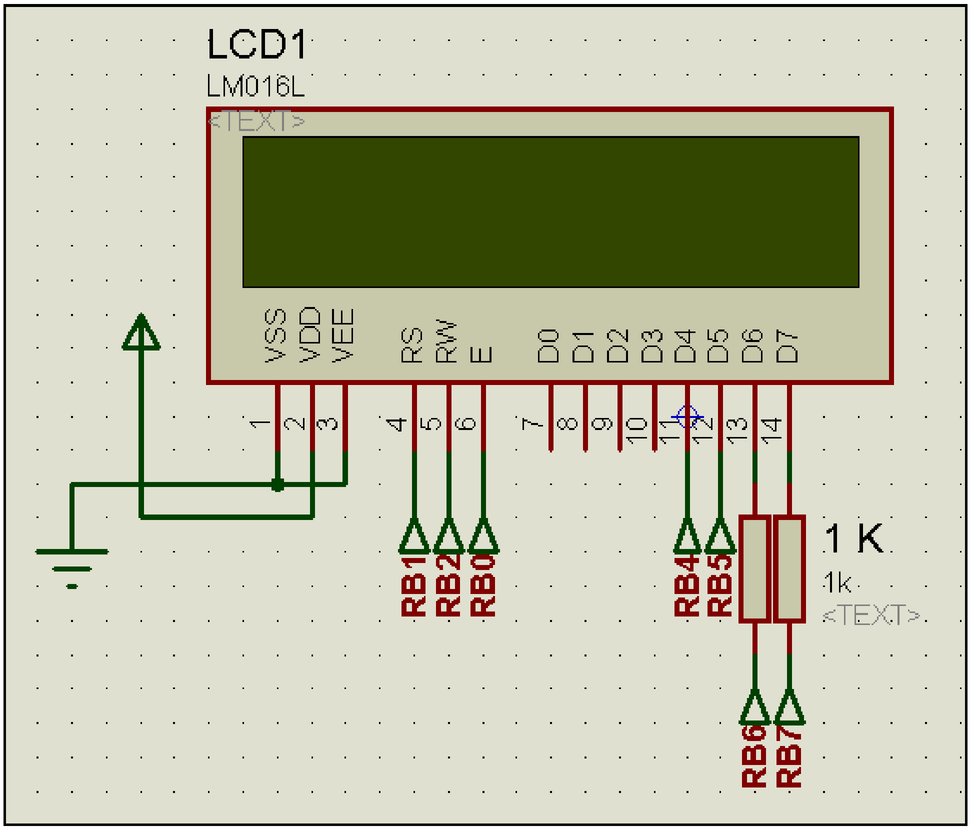


Figure 3.9: LCD Module

The LCD used for this project is a 2x16 character LCD. Vss and V_{EE} pins of the LCD are connected to ground while pin V_{DD} is connected to power supply of +5V. Pins RB0, RB1 and RB2 of microcontroller are connected to the enable (E), select register (RS) and select read or write (RW) pins of LCD respectively. As to operate the LCD in 4-bit mode, data bus pins from D4 to D7 is connected to pins RB4 to RB7 of microcontroller. Pins 15 and 16 of LCD represents for backlight positive input and backlight negative input. Thus, pin 15 is connected to power supply and pin 16 is grounded.

3.3.6 RFID MODULE

Figure 3.10 below shows the circuit connection for the Radio Frequency Identification (RFID) module:

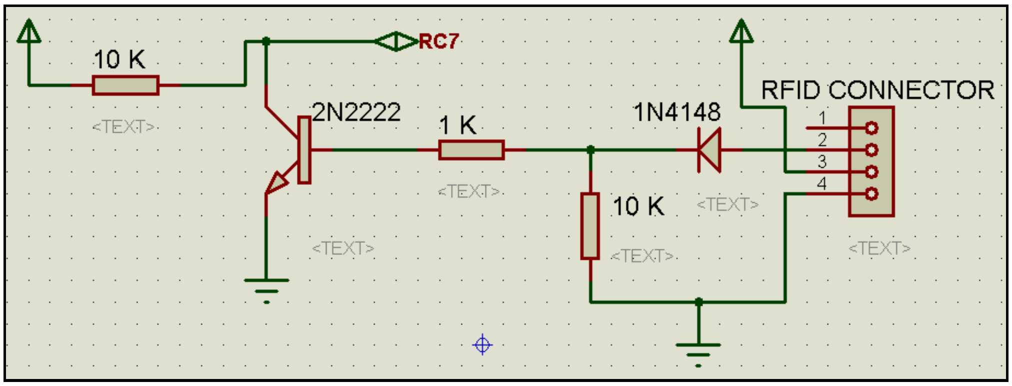


Figure 3.10: RFID Module

Vcc of RFID is connected to power supply and ground of RFID is connected to common ground. The Tx of RFID is connected to pin RC7 through the circuit shown above. The output of the RFID reader is serial UART in logic +10V/-10V and the baud rate is 9600 bps. The circuit shown is to convert +10V/-10V logic to +5V/-5V logic for PIC microcontroller.

3.3.7 XBEE PRO TRANSMITTER MODULE

Figure 3.11 below shows how to connect the XBee Pro wireless module to the PIC microcontroller:

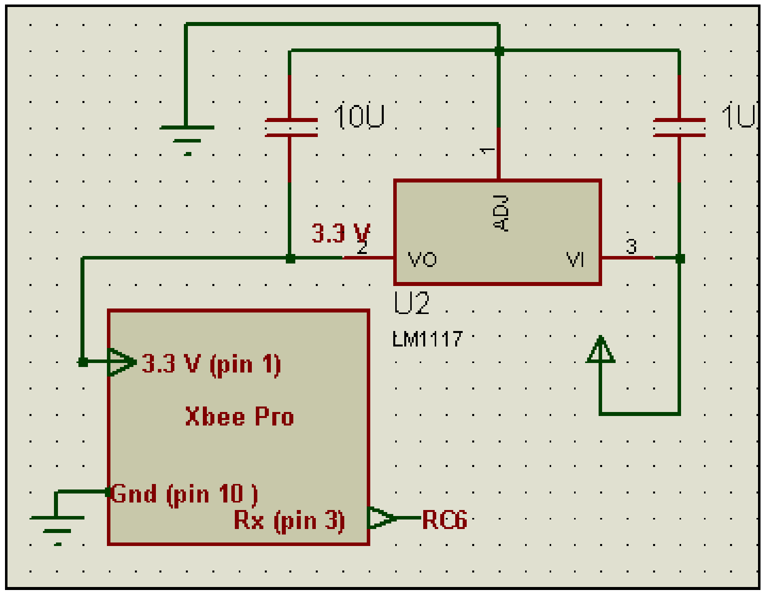


Figure 3.11: XBee Pro Transmitter Module

The XBee Pro transmitter module needs a 3.3V as its power supply. Any voltage more than this voltage level could fry the wireless module. Thus, a voltage regulator of LM1117 is used to regulate +5V into +3.3V and provides voltage to the XBee Pro wireless module. For the XBee Pro to transmit data from the host system, it should receive data from pin RC6 of microcontroller.

3.3.8 PIC's MODULE

Figure 3.12 below shows the connection of the PIC16F87A with other components on the host system:

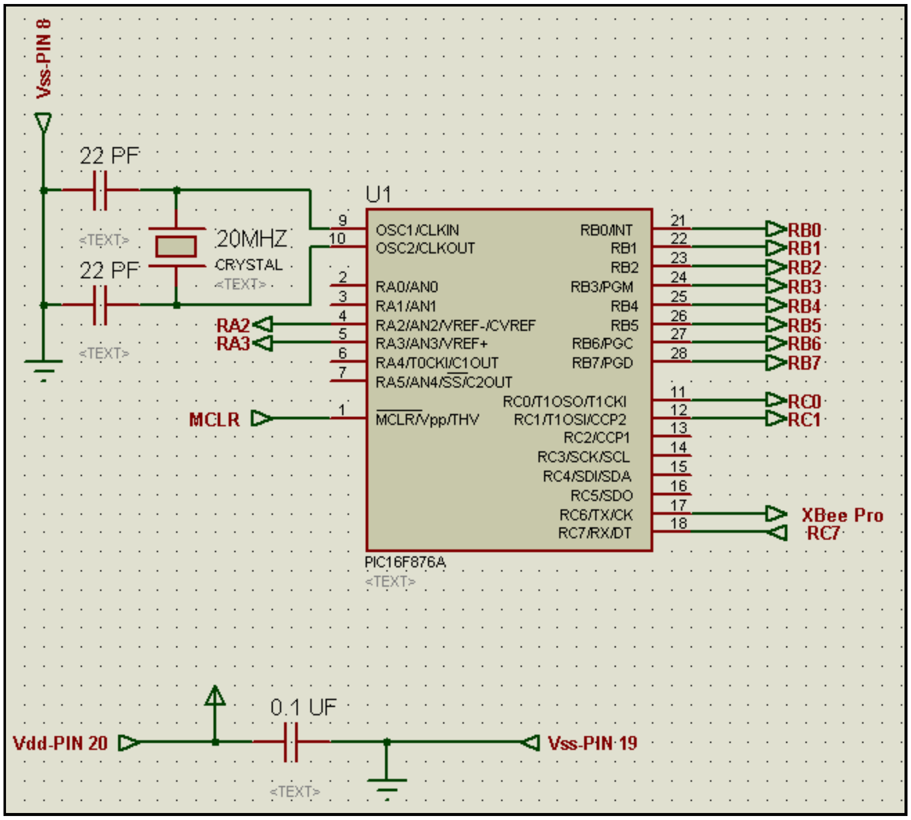


Figure 3.12: Transmitter's PIC Module

All pins of the PIC microcontroller is connected to its corresponding modules. The clock section governs all system timing and thus is really responsible for the proper operation of all system hardware. The clock section usually consists of a crystal oscillator and clock circuitry set up to operate the processor at its specified clock rate. The main purpose of the oscillator in microcontrollers is to provide a reliable clock for the controller processes. At the most basic level, the clock provides a timing interval to account for circuit rise times and to allow data to stabilize before that data is processed.

The internal oscillator circuit is used to generate the device clock. The device clock is required for the device to execute instructions and for the peripherals to function. There are up to eight different modes which the oscillator may have. The oscillator mode is selected by the device configuration bits. The device configuration bits are nonvolatile memory locations and the operating mode is determined by the value written during device programming.

The crystal is connected to OSC1 and OSC2 pins to establish oscillation. The PIC microcontroller design requires the use of parallel cut crystal. Using the different cut crystal may give frequency out of manufacturer's specification. The values of capacitors used are identical to the range tested below in the Table 3.2. High capacitance increases the stability of the oscillator but also increase the startup time. Figure 3.11 above shows the crystal oscillator circuit also.

Table 3.2: Typical capacitor selection for crystal oscillator

Mode	**Frequency**	**C1**	**C2**
Low Frequency (LP)	32kHz	68-100pF	68-100pF
	200kHz	15-30pF	15-30pF
Crystal/ Oscillator (XT)	100kHz	68-150pF	15-30pF
	2MHz	15-30pF	15-30pF
	4MHz	15-30pF	15-30pF
High Speed (HS)	8MHz	15-30pF	15-30pF
	10MHz	15-30pF	15-30pF
	20MHz	15-30pF	15-30pF

Using a high frequency crystal oscillator and dividing it down to a lower frequency provides for greater stability. I used a 20MHz crystal which will be divided by microcontroller to operate at 5MHz frequency at room temperature.

3.4 HARDWARE OF RECEIVER (SECONDARY SYSTEM)

The hardware portion of the receiver side, that is for the secondary system also involves various types of modules in order for the system to function or operate. They are the UIC00A Programmer module, the power module, the reset module, the output's module, the XBee Pro circuit module and the PIC's module itself. All these modules are connected in the same way as in the host system except for the XBee Pro receiver circuit module, the output's module and the PIC's module. Thus, in the section that follows, I will only explain about the XBee Pro receiver circuit module, the output's module and the PIC's module. The modules have to be combined and connected into one single circuit to operate the system. The complete circuit of the secondary system is included in the Appendix B for reference.

3.4.1 XBEE PRO RECEIVER MODULE

Figure 3.13 below shows the connection for XBee Pro receiver module:

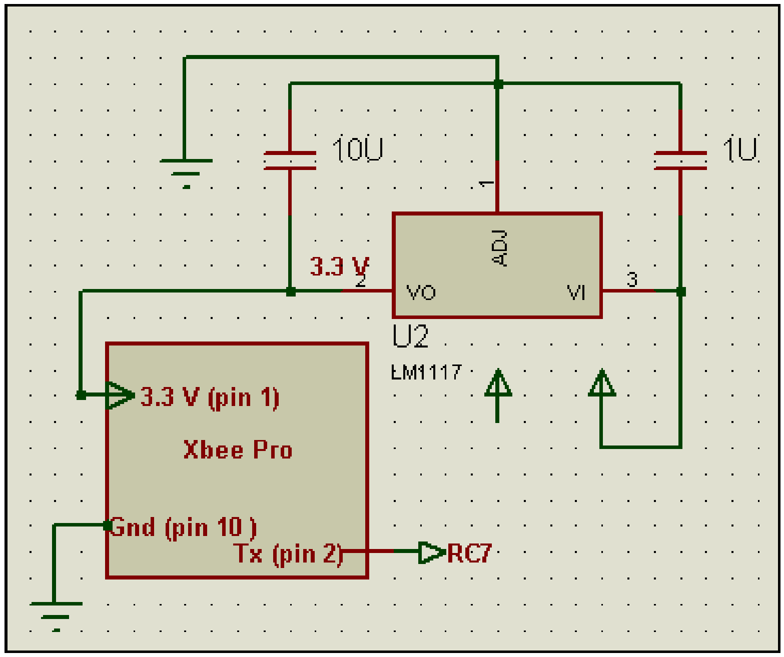


Figure 3.13: XBee Pro Receiver Module

XBee Pro transmitter module needs a 3.3V as its power supply. Any voltage more than this voltage level could fry the wireless module. Thus, a voltage regulator of LM1117 is used to regulate +5V into +3.3V and provides voltage to the XBee Pro wireless module. After the XBee Pro receives data from the host system, it should transmit data to pin RC7 of microcontroller.

3.4.2 OUTPUT'S MODULE

Figure 3.14 below shows the connections used to connect all the outputs for the secondary system:

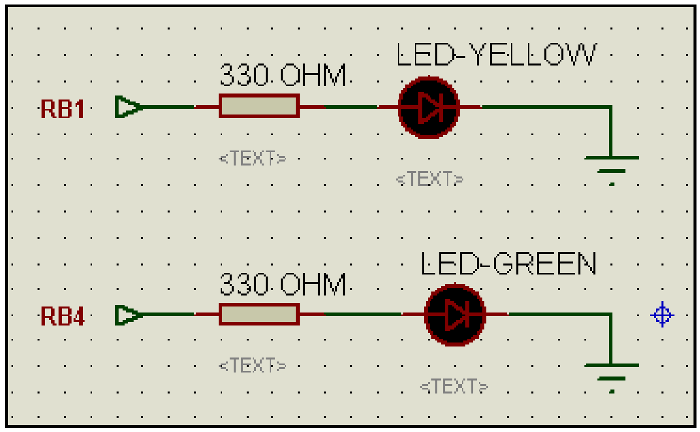


Figure 3.14: Receiver's Output Module

The LEDs are connected to pin RB1 and RB4 of microcontroller respectively to represents outputs. Here I have represented the LEDs for the magnetic sensor door. The function of the 330 ohm resistors is to protect the LEDs from over current that will burn the LEDs. When the output is logic 1, the LEDs will ON, while when the output is logic 0, the LEDs will OFF.

3.4.3 PIC's MODULE

Figure 3.15 below shows the connection of the PIC16F87A with other components on the secondary system:

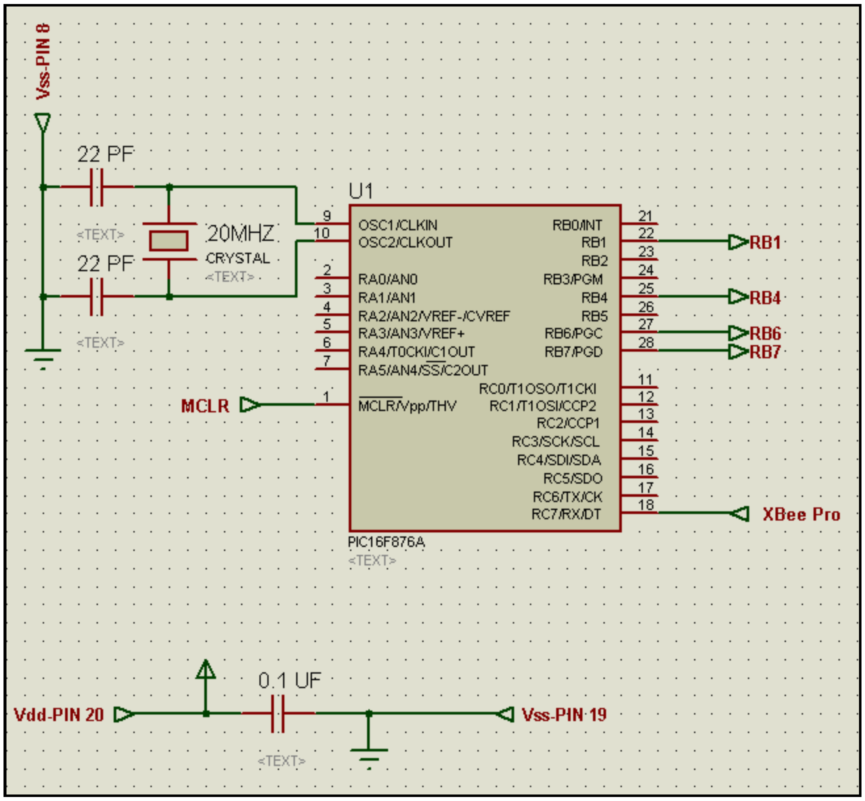


Figure 3.15: Receiver's PIC Module

All pins of the PIC microcontroller is connected to its corresponding modules. The clock section governs all system timing and thus is really responsible for the proper operation of all system hardware. The clock section usually consists of a crystal oscillator and clock circuitry set up to operate the processor at its specified clock rate. Using a high frequency crystal oscillator and dividing it down to a lower frequency provides for greater stability. I used a 20MHz crystal which will be divided by microcontroller to operate at 5MHz frequency room temperature.

CHAPTER 4

SOFTWARE IMPLEMENTATION

4.1 INTRODUCTION

Once the circuit was designed, soldered with necessary components and wired with wrapping wires, they need to be tested. Before testing, the microcontroller units of PIC16F876A needed to be programmed to guide the operation of all components in the system. The operations that should be performed by the system is to receive data from RFID reader, indicate outputs of the system and functions in either transmitting data to or receiving data from the XBee Pro wireless modules. The C programming language is mainly used in this project to develop the methods of data comparison and data transmission. Other than that, it is used for indicating outputs and controls the other components of the system.

4.2 BACKGROUND

As mentioned before, the microcontroller units for the designed circuit is the PIC16F876A. The initials "PIC" represent a family of microcontrollers manufactured mainly by Microchip Technology. The name PIC is an acronym for "Programmable Intelligent Computer". PIC's are popular because of their wide availability and they are relatively low cost processors. There are many families of PIC's, including the 16F, 18F and others. Each family has slightly different capabilities and packages in the aspect of pin layouts and spacing. The PIC16F876A was selected as it offers enough memory requirements for this project. The pin diagram for the PIC16F876A can be found on Figure 3.4 of chapter 3. The microcontroller must be programmed in order for it to function to its intended functionality. When the PICs and all other components were soldered and wired, the first step to be completed is installing the UIC00A compatible PICkit 2 programming software.

UIC00A is a PIC-USB programmer. It is designed to program the microcontrollers. On board ICSP (In Circuit Serial Programming) connector offers flexible methods to load program. It supports on board programming which eliminate the frustration of plug-in and plug-out programming of PICs. This allows the microcontroller to be quickly programmed and debugged the source code while the target PIC on the board itself. Since USB port have become a popular and widely used on laptops and desktop PCs, UIC00A is available for plug and play with USB connection. This programmer obtain its power directly from USB connection, thus no external power supply is required. Connect one end of USB cable to USB port at laptop or PC, and then connect another end of mini USB cable to UIC00A USB port. One side of programming cable is connected to box header of UIC00A and the other side to the box header on the project's board itself for the PIC to be programmed.

The first step before loading program into PIC is writing the codes. To accomplish this task, a program called PIC C Compiler was used. This program allows the code written in C programming language. The code must be compiled using PIC C Compiler's built in compiler to change the C code to hexadecimal (HEX) format, which is what the PIC understands and can run. Therefore it is important to compile after any and all changes are made to the code. After the code was developed and compiled using PIC C Compiler, the software PICkit 2 was used to write the code to the PIC.

The PICkit 2 programming software is launched. The following programming interface appears as shown in Figure 4.1 and notifies that the PICkit 2 the target device is found and connected.

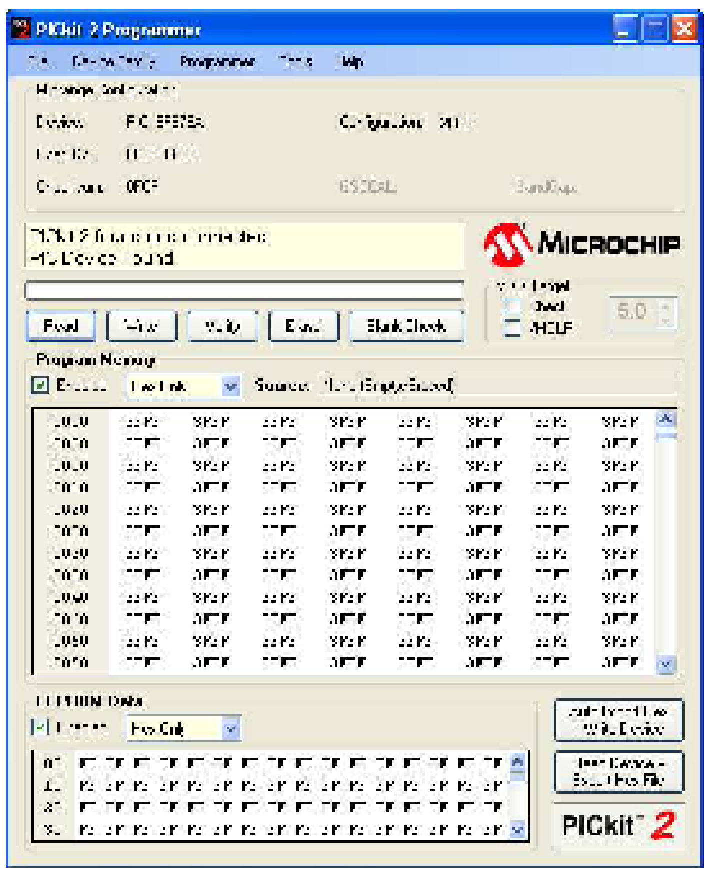


Figure 4.1: PICkit 2 Programmer Interface

Then go to the file icon and select the Import Hex, browse the location of the hex file and open to start importing the hex file. After the hex file has been imported, the PIC can be programmed by clicking the Write icon. The PIC will be erased and programmed with the new hex code imported. The operation status will be displayed and the status bar will turn green if writing is successful as shown in Figure 4.2 below.

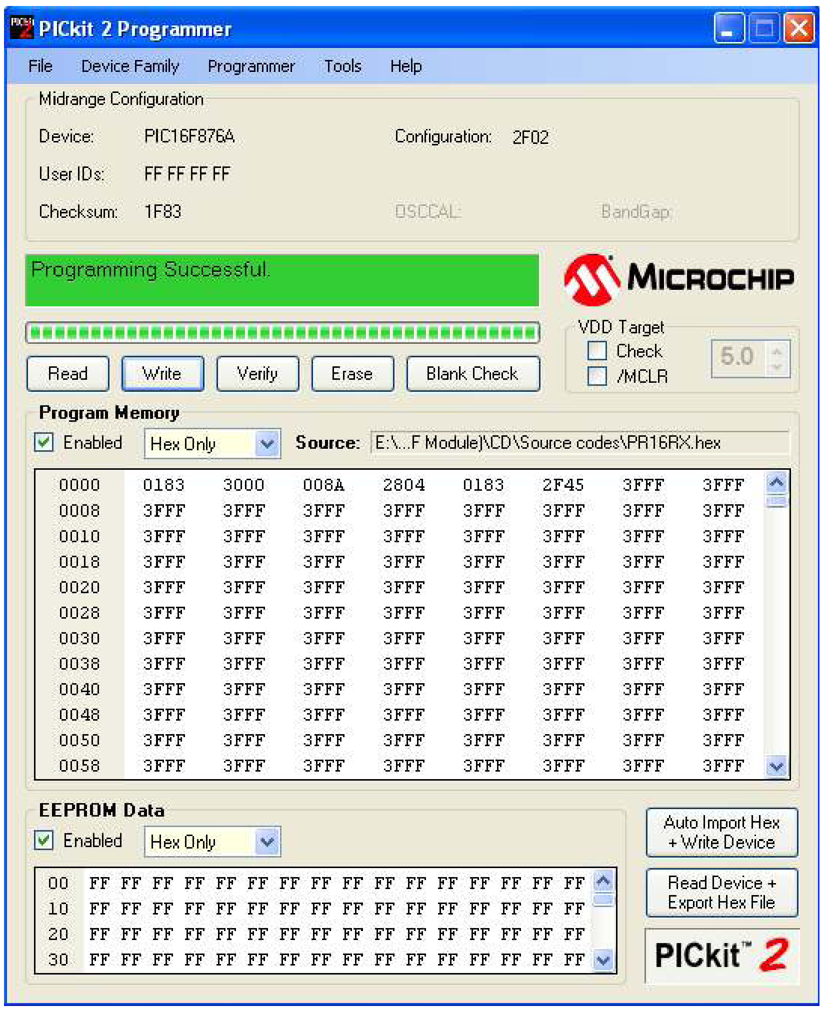


Figure 4.2: PICkit 2 Programmer Interface Indication

4.3 OVERVIEW OF THE SOFTWARE IMPLEMENTATION

First, both PICs 16F876A at the host and secondary system are programmed using In Circuit Serial Programmer, (ICSP). The PIC at the host and secondary system will be preprogrammed with all users' identification codes. When a person places the identity card or tag on the RFID reader, the RFID reader reads the tag code. Received identity card's code is sent to the microcontroller PIC16F87A. Inside the microcontroller, the code received from the RFID reader will be compared with identification codes preprogrammed earlier and whether the codes match or not is verified. If a user is identified, the identification code and user name will be displayed on the LCD display. If a non-user is identified, the identification code and user name of "not found" will be displayed on the LCD display. After displaying details on the LCD display, the code received from the RFID reader, both users and non-users code are sent to the XBee Pro to be transmitted.

The code transmitted by Xbee Pro will be received by another XBee Pro module on the receiving end of the secondary system. This code is sent to the PIC16F876A microcontroller to be compared again as done before in the host system. The comparison method is done again for security purposes so that data sent from the host system is not lost during wireless transmission. Here, when a user is identified, the magnetic sensor door is opened represented with an LED in this project. Thus, the door can be controlled even it is far from the host system. Figure 4.3 shows the software flow chart for this system.

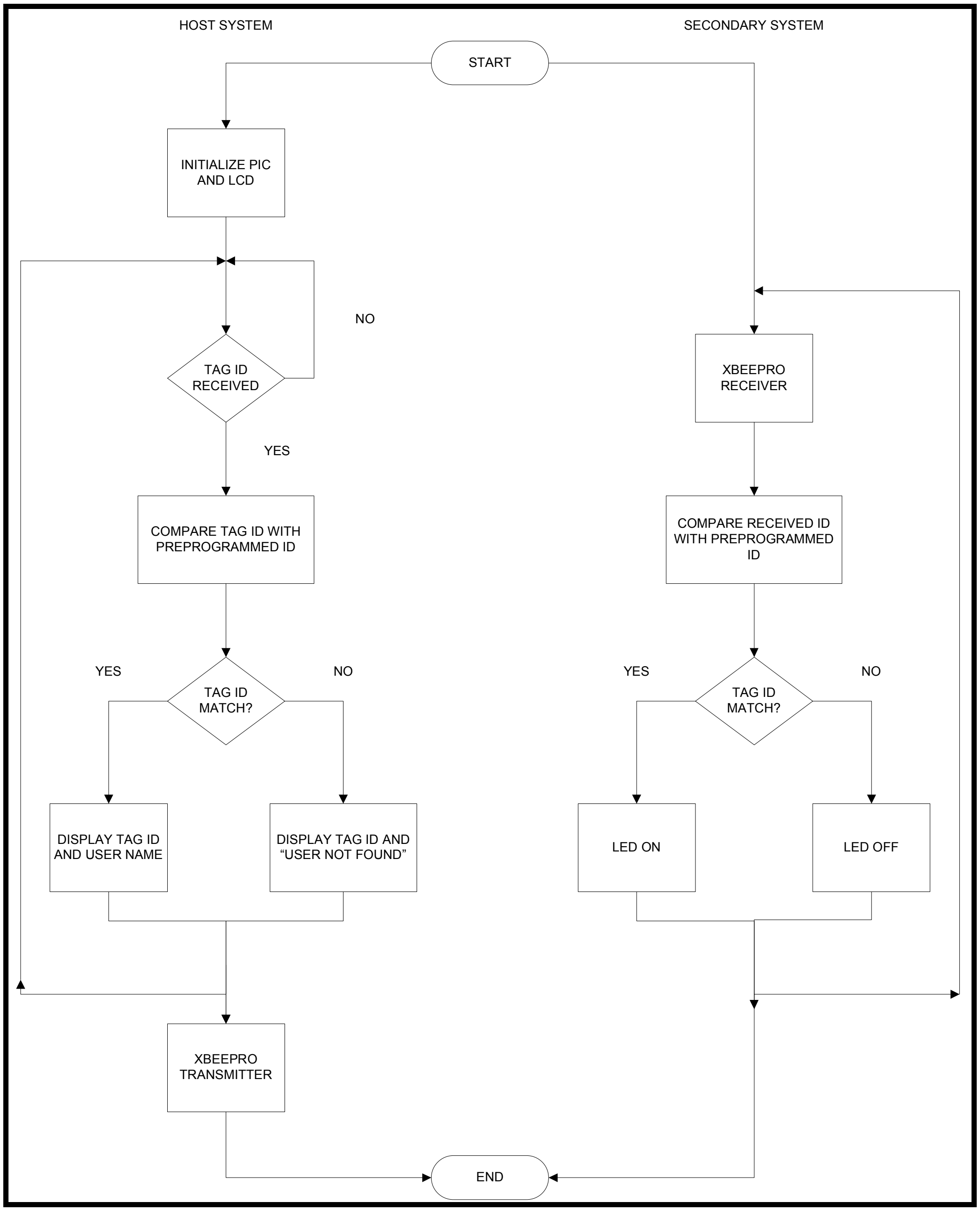


Figure 4.3: Software Flow Chart of RFID Wireless Door System

4.4 C PROGRAMMING LANGUAGE

C programming language is a computer programming language that creates lists of instructions for a computer or a microcontroller to follow. C is one of thousands of programming languages currently in use. C has been around for several decades and has won widespread acceptance because it gives programmers maximum control and efficiency. The full source code for the project is presented in Appendix C and Appendix D.

4.4.1 OUTPUT ON LCD DISPLAY

In this project, the first output of the system is an LCD display that will display the information about the system. The LCD display will display phrases such as “RFID Wireless door System”, ‘Place Your ID on the Reader”, “User ID:” and “User Name:” In order for the LCD display to display the stated phrases, the PIC should be programmed with appropriate C programming language. The source code is shown in Figure 4.4 below.

```
printf(lcd_putc, "   RFID Wireless  ");

printf(lcd_putc, "Door System");

printf(lcd_putc, " Place your ID ");

printf(lcd_putc, " On The Reader");

printf(lcd_putc, "User ID:");

printf(lcd_putc," User Name:");
```

Figure 4.4: C Code of LCD’s Output

4.4.2 RECEIVE DATA FROM RFID READER

The main operation that should be done in this project is to receive data from RFID reader, compare it and send the received data wirelessly. Therefore, the data coming in from RFID reader is very important as this will assist to proceed to other operations such as data comparison and data transmission. If data cannot be received from RFID reader, the whole system will not function. Thus, for effectively receive data from the RFID reader; source code shown in Figure 4.5 is used.

```
for(j=0;j<10;j++)

   code[j]=i=fgetc();
```

Figure 4.5: C Code to Receive RFID Data

The source code shown above in Figure 4.4 explains that ten characters should be received from the reader where the ten characters represent a code. The "i" and "j" are characters respectively and the command "fgetc()" represents the incoming data from RFID reader.

4.4.3 DATA COMPARISON

After data have been received from the RFID reader, the data should be compared with the ID code programmed earlier in the PIC to verify whether the ID represents a user or non user. If a user is identified, the user's name will be displayed in the LCD display and if a non-user is identified user not found will be displayed. Figure 4.6 below shows the source code for comparing the received data with the ID number programmed earlier in the PIC.

```
if(!strcmp(code,passa))
{
  printf(lcd_putc,"\f user name:\n Ashvaany");
}
else if (!strcmp(code,passb))
{
  printf(lcd_putc,"\f user name:\n Egambaram");
}
else
{
  printf(lcd_putc,"\f user name:\n not found");
}
```

Figure 4.6: C Code to Compare Data

The command "passa" and "passb" are ID numbers that is declared earlier and compared with the received "code" from RFID reader.

4.4.4 DATA TRANSMISSION

The received data from RFID reader will be transmitted to the Xbee Pro wireless module. So the code will be transmitted by the Xbee Pro to the secondary system from the host system. Figure 4.7 below show a command to transmit the code through Xbee Pro wireless module.

```
printf(code);
```

Figure 4.7: C Code to Transmit Data

4.4.5 DATA RECEPTION

Data transmitted by the host system will be received on the secondary system by another Xbee Pro wireless module. Figure 4.8 shows the source code for receiving data from the Xbee Pro at the secondary system. The command "kbhit()" is an interrupt instruction where data will be read as soon there is data coming in from the Xbee Pro wireless module.

```
if (kbhit())
{
  for(j=0;j<10;j++)
  code[j]=i=fgetc();
}
```

Figure 4.8: C code to Receive Data

CHAPTER 5

RESULTS AND DISCUSSIONS

5.1 INTRODUCTION

This chapter covers the simulation results and the functionality of hardware of a RFID Wireless Door System. Software called Proteus 7 Professional is used to simulate the microcontroller. The simulated microcontroller runs firmware just as in real life and it can interact with large range of peripheral circuit models and powerful debugging tools.

5.2 SIMULATION RESULT FOR THE HOST SYSTEM

When the system is initialized, the LCD display will conduct a user by displaying “RFID Wireless door System” as shown in Figure 5.1.

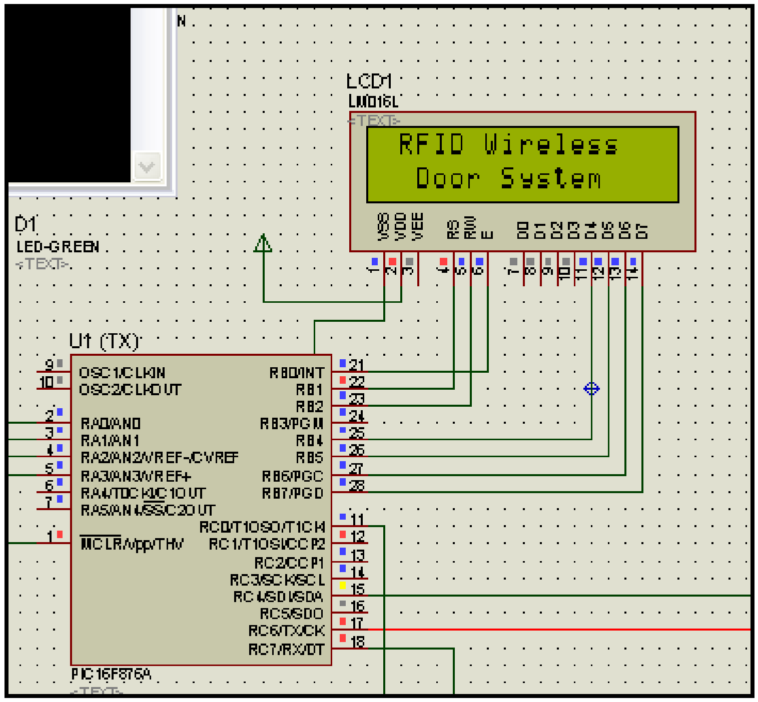


Figure 5.1: First Simulation Output of LCD

After displaying about the system, it will display “Place Your ID on the Reader” as shown in Figure 5.2. As soon as this phrase is displayed, the user has to place his identity card near the RFID reader.

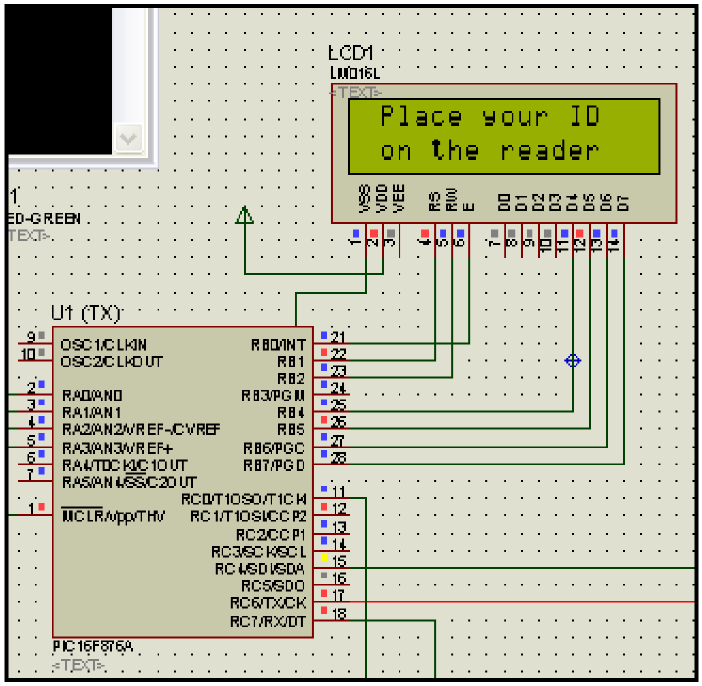


Figure 5.2: Second Simulation Output of LCD

The RFID is represented as a virtual terminal for simulation purposes. Thus, the user will enter the identity cards number into the virtual terminal. After the ID code is inserted, “User ID” and User Name” will be displayed on the LCD display as shown in Figure 5.3.

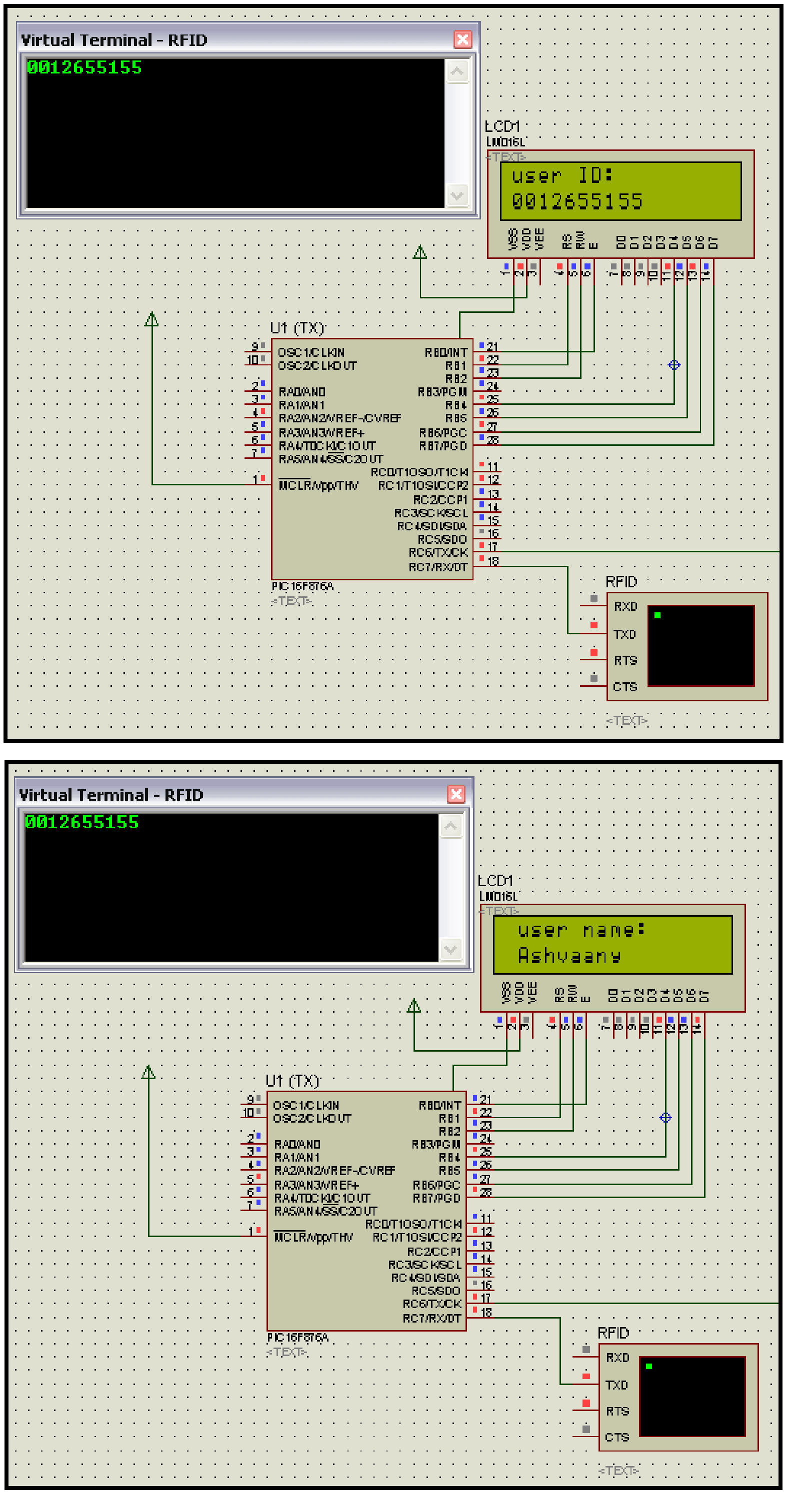


Figure 5.3: Final Simulation Output of LCD

5.3 SIMULATION RESULT FOR THE SECONDARY SYSTEM

Obtained ID number from the RFID reader is sent to the secondary system through Xbee Pro wireless module but in simulation, a wire is connected from the host system to the secondary system. The secondary system will receive the code through another Xbee Pro wireless module. A virtual terminal will display the identity code received and if a user is verified by the microcontroller, LED will ON. The LED represents a magnetic sensor door in this project. The virtual terminal display and LED is shown in Figure 5.4. Figure 5.5 shows the simulation result of the whole system.

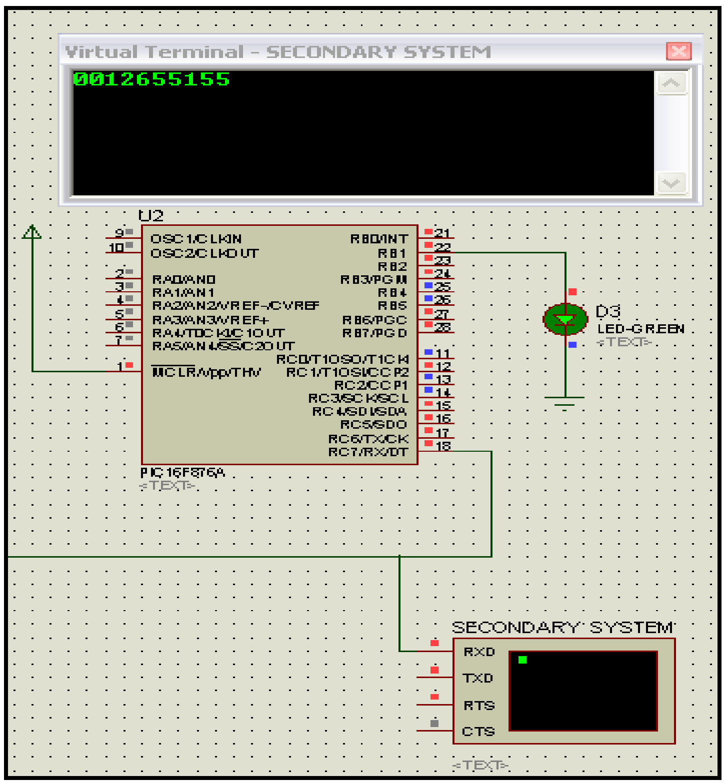


Figure 5.4: Secondary System's Simulation Result

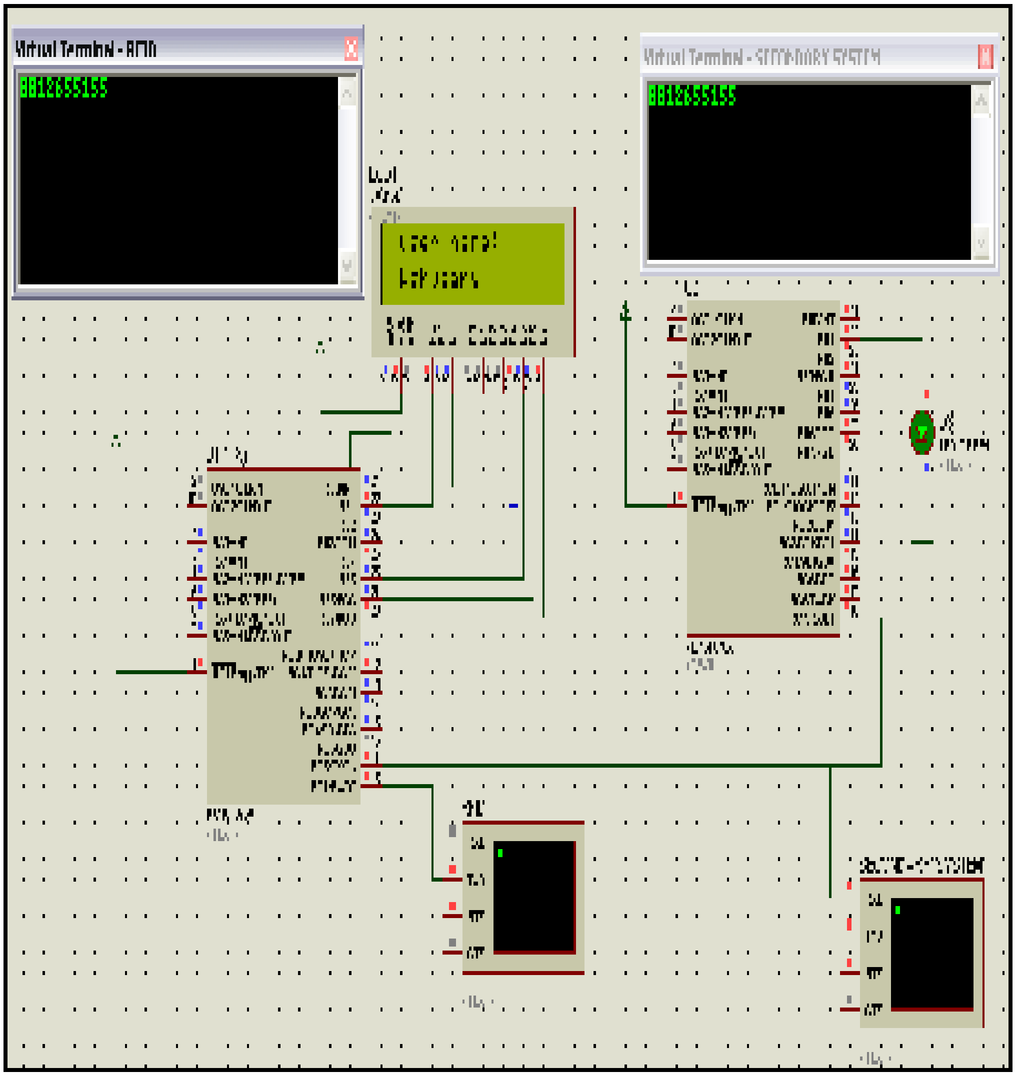


Figure 5.5: Simulation Result of the System

5.4 HARDWARE RESULT FOR THE HOST SYSTEM

The same procedures take place for hardware result. First the LCD display displays “RFID Wireless Door System”. Then, “Place Your ID on the Reader” will be displayed as shown in Figure 5.6.

Figure 5.6: Output on LCD Display

When the identity card is placed near the RFID reader, the card's ID number is read and displayed on LCD display followed by displaying user's name as shown in Figure 5.7 and Figure 5.8 below.

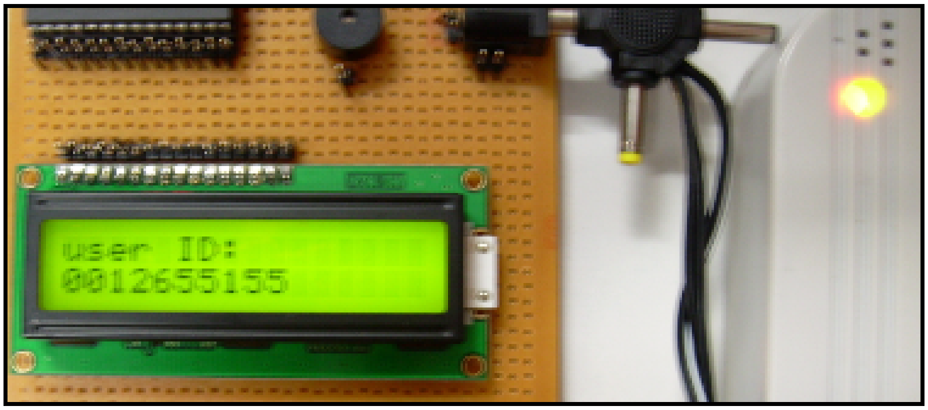


Figure 5.7: Identification of User ID

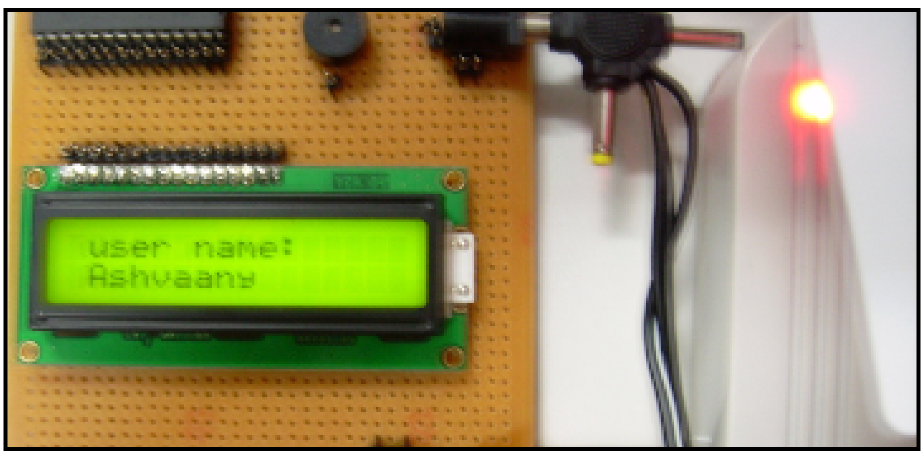


Figure 5.8: Identification of User Name

5.5 HARDWARE RESULT FOR THE SECONDARY SYSTEM

Obtained ID number from the RFID reader is sent to the secondary system through Xbee Pro wireless module. The secondary system will receive the code through another Xbee Pro wireless module. The identity code is received and if a user is verified by the microcontroller, LED will ON. The LED represents a magnetic sensor door in this project. Figure 5.9 shows the LED ON when the correct ID code received and Figure 5.10 shows the result for the whole RFID Wireless Door System.

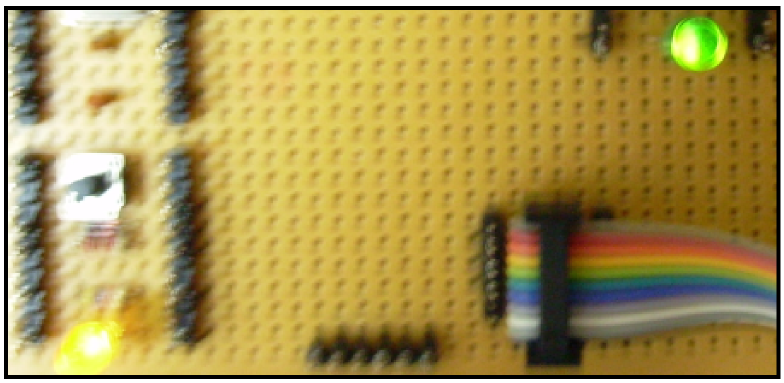


Figure 5.9: Result of the Secondary System

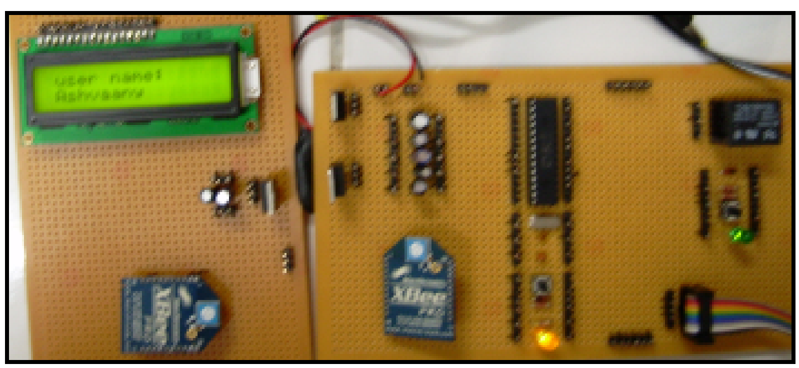


Figure 5.10: Result of the System

5.6 DISCUSSIONS

In this project, two main technologies are used, namely Radio Frequency Identification (RFID) and Xbee Pro Wireless technology. In Radio Frequency Identification (RFID), the RFID reader will read data from RFID tag and displays the identified code on LCD display. The type of tag chosen is the passive RFID tag. Passive RFID tags have no internal power supply. Instead, a small electric current is created in the antenna when an incoming signal reaches it. This current provides enough power to briefly activate the tag, usually just long enough to relay simple information, such as an ID number. Because passive RFID tags do not contain any internal power supply, they can be very small in size, sometimes thinner than a piece of paper. The communication range for the passive RFID tags is very short. As there is no need for long communication range between the reader and the tag, passive RFID tag is chosen.

Xbee Pro wireless module operates at a frequency of 2.4GHz, has a communication range of 1.5km outdoor and 100m indoor. This is much higher than Xbee where it operates at the same frequency of 2.4GHz, but has shorter communication range of 100m outdoor and 30 m indoor. Thus, Xbee Pro is chosen as the wireless medium for data transmission rather than Xbee so that controlling a door is not limited within few hundred meters only.

CHAPTER 6

CONCLUSION AND RECOMMENDATIONS

6.1 CONCLUSION

It can be concluded that all the three objectives of this project is achieved. The outcome for this project is able to fulfill scope and the expected results are obtained. The RFID Wireless Door System can be implemented in offices and also places where manual door opening is practiced. Radio Frequency Identification (RFID) is used for tracing ID numbers. By using the PIC16F876A microcontroller, the traced ID number can be verified and effectively transmitted through Xbee Pro wireless module. C programming language perfectly assists in receiving data from RFID reader, comparing the received data and also to transmit the received data.

In this report, fair progress had been obtained, largely in the design and development aspect of the RFID Wireless Door System. The whole of this report serves to bring sufficient knowledge and information regarding design for C programming, circuit design and project's hardware development.

6.2 RECOMMENDATIONS

The Proteus 7 Professional is effective software which simulates microcontroller to run firmware just as in real life and it can interact with peripheral circuits and powerful debugging tools. However Proteus 7 Professional has some limitation, it can't produce noises which occurs in actual hardware's output. Any problem that occurs in hardware might be due to noise in the circuit. Thus, simulation results can be used to as references.

In future development of this project:

i) A door can be included into the hardware to indicate the real functionality of the project.

ii) The operation of the door can also be limited according to time such as operation hour from 7am to 7pm. This can be done by including a real-time clock into the circuit.

iii) Other than that, to be more secure, finger print technology and image recognition can be used instead of RFID technology for data identification.

iv) For transmitting data between Xbee Pro wireless modules, data encryption methods such as XTEA encryption method can be used so that data is not lost during transmission.

6.3.1 COSTING AND COMMERCIALIZATION

RFID Wireless Door System is a system in which a door can be controlled by anyone with a correct ID number. There is no need for the user to be right near the door to control the door. The user can control the door from long distance if this system is used. At present there is no such device or system in market that allows a user to control a door from long distance.

Almost all premises and offices do have their own door security system using RFID technology, where the door will open when a user is identified through RFID but whenever there is a client or a customer wants to enter their office, a staff should go near the door and place his ID on the RFID reader in order for the door to open. As can be seen, this method will consume more time, both for the client waiting outside and also for the staff. This, in return reduces the production quantity, quality and waste of time for their business.

Thus, RFID Wireless Door System can be implemented where it's unique in wirelessly transmitting data to open the door further from the host system thereby making things happen quicker. The estimated cost of the device would be approximately RM 565. Table 6.1 shows the summary of cost for components and devices used in this project.

Table 6.1: Component's Cost

Component	Specification	Quantity	Estimated cost
PIC16F876A	28 PINS	2	RM 50
2X16 LCD DISPLAY		1	RM 30
XBEE PRO		2	RM 250
ADAPTER SOCKET(DC)		1	RM 0.50
BUZZER	4.7KΩ	1	RM 2
DIODE	IN4007	1	RM 0.10
DIODE	IN4148	2	RM 0.20
LED	GREEN	1	RM 0.13
LED	RED	2	RM 0.26
CRYSTAL	20 MHz	2	RM 1
CAPACITOR	22pF	6	RM 0.42
CAPACITOR	0.1uF	2	RM 0.36
CAPACITOR	100uF	3	RM 0.20
VOLTAGE REGULATOR	LM1117	2	RM 2
VOLTAGE REGULATOR	LM7805	2	RM 2
PUSH BUTTON		2	RM 1
RESISTOR	330Ω	5	RM 0.30
RESISTOR	4.7kΩ	4	RM 0.24
RESISTOR	1kΩ	3	RM 0.18
RESISTOR	10kΩ	2	RM 0.12
TRANSISTOR	2N2222	1	RM 0.50
IC BASE	28 PINS	2	RM 0.70
SCREWS & NUTS	3MM	8	-
PCB HEADER		5	RM 4
HEAT SINK		1	RM 0.90
RFID READER		1	RM 198
RFID TAGS		2	RM 20
TOTAL AMOUNT			**RM 565.11**

This amount is considered as quite high. But this is a one time investment. Once implemented it would be cost effective for the user of this system as it reduces the manual work, so the cost of the device can be recovered in no time. Also if the product is manufactured in bulk, it would reduce the price of the device thereby making it more cost effective.

REFERENCES

[1] Cytron Technologies. “RFID READER”
Available at: *www.cytron.com.my/listProductCategory*

[2] Wikipedia-the free encyclopedia. “RFID”
Available at: *http://en.wikipedia.org/wiki/RFID*

[3] © 2006 MaxStream, Inc.
Available at: *www.maxstream.net*

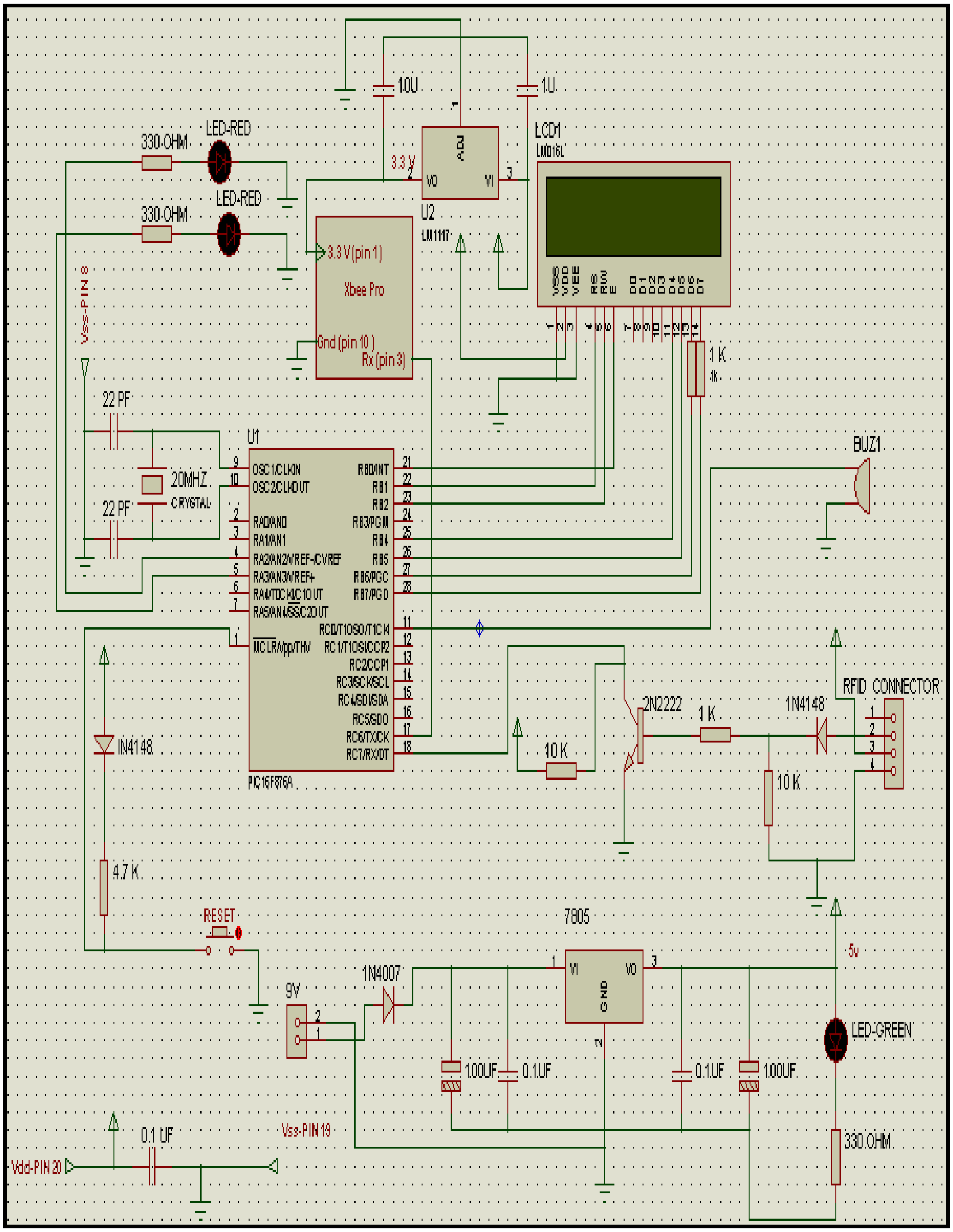
[4] Product Manual Xbee-OEM
Available at: *www.maxstream.net*

[5] PIC16F87XA Data Sheet
Available at: *http://ww1.microchip.com/downloads/en/DeviceDoc/39582b.pdf*

[6] PIC16F87XA Data Sheet
Available at: *http://www.microchip.com/stellent/idcplg?IdcService*

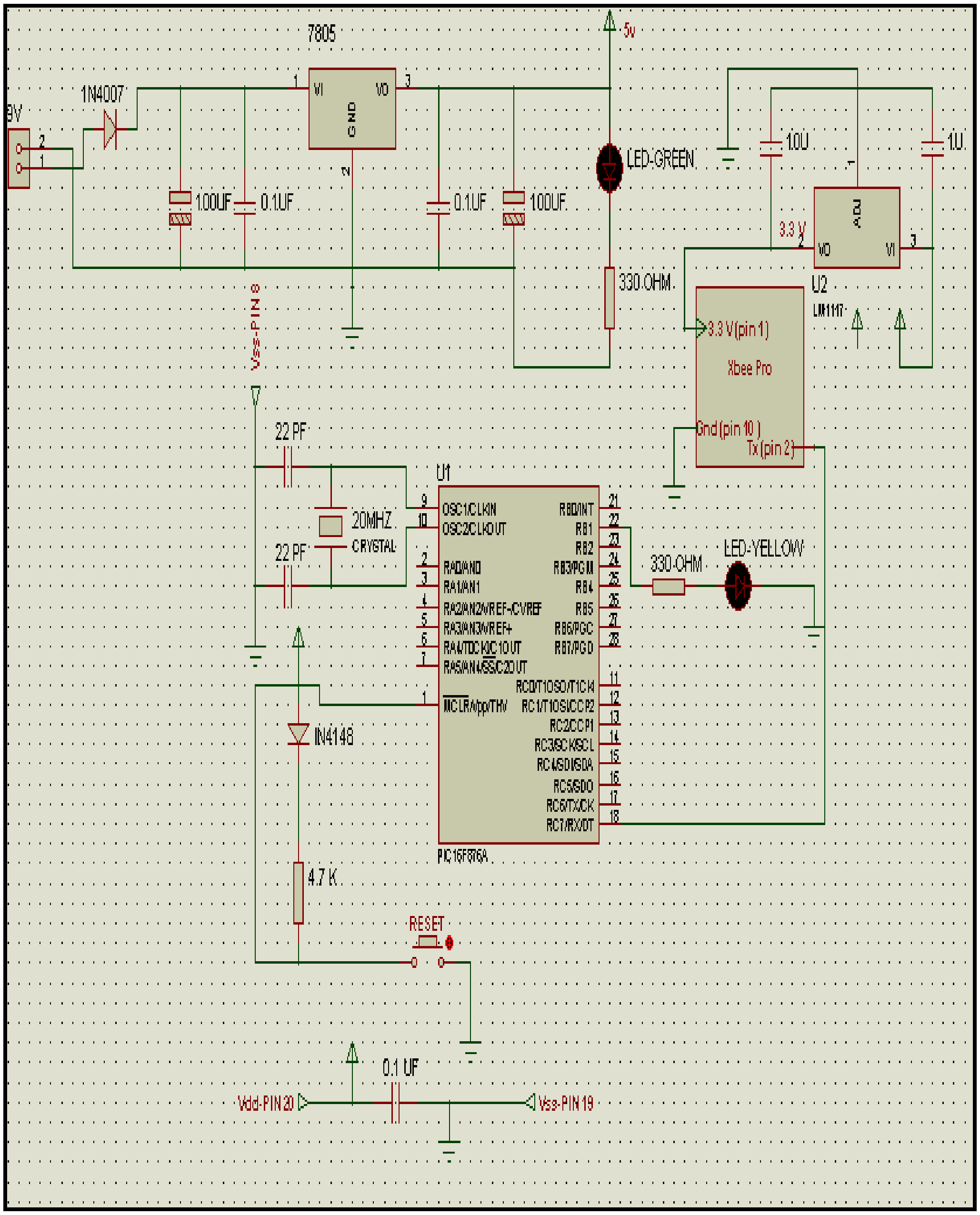
APPENDIX A

CIRCUIT – HOST SYSTEM



APPENDIX B

CIRCUIT – SECONDARY SYSTEM



APPENDIX C

C CODE – HOST SYSTEM

```
#include <16F876A.h>
#include <string.h>

/*configuration*/
#FUSES NOWDT                    //No Watch Dog Timer
#FUSES HS                       //Crystal Osc (<= 4mhz)
#FUSES PUT                           //Power Up Timer
#FUSES NOPROTECT                //Code not protected from
reading
#FUSES NODEBUG                  //No Debug mode for ICD
#FUSES NOBROWNOUT               //Don't Reset when brownout
detected
#FUSES NOLVP                    //No low voltage prgming,
                                B3(PIC16) or B5(PIC18) used for
                                I/O
#FUSES NOCPD                    //No EE protection
#FUSES NOWRT                    //Program memory not write
protected

#use delay(clock=20000000)
#use rs232(baud=9600,xmit=PIN_C6,rcv=PIN_C7,PARITY=N)
//RFID io
#define   rs  RB1               //RS pin of the LCD display
#define   rw  RB2               //R/W pin of the LCD display
#define   e   RB0               //E pin of the LCD display
#define   b_light  RC1          //backlight of the LCD
                                display (1 to on backlight)
#define   buzzer   RC0          //buzzer (1 to on buzzer)
#define   button1  RA0          //button (active low)
#define   button2  RA1          //button (active low)
#define   lcd_data PORTB //LCD 4-bit data PORT
```

```
#define   led1     RA2   //led 1 (active high)
#define  led2      RA3   //led 2 (active high)
#define use_portb_lcd TRUE

#include "lcd.c"
void main()
{
   char i,j;
   char code[11];
   char passa[11] = {"0012655155"};
   char passb[11] = {"1234567890"};

   setup_adc_ports(NO_ANALOGS);
   setup_adc(ADC_OFF);
   setup_spi(SPI_SS_DISABLED);
   setup_timer_0(RTCC_INTERNAL);
   setup_timer_1(T1_DISABLED);
   setup_timer_2(T2_DISABLED,0,1);

   set_tris_a(0x00);
   set_tris_b(0x00);
   set_tris_c(0xA0);
```

```
lcd_init();


while (TRUE){
printf(lcd_putc, " RFID Wireless ");
printf(lcd_putc,"\n");
printf(lcd_putc, "  Door System  ");
delay_ms(2000);
printf(lcd_putc,"\f");
printf(lcd_putc, " Place your ID ");
printf(lcd_putc,"\n");
printf(lcd_putc, " on the reader");

for(j=0;j<10;j++)
code[j]=i=fgetc();
printf(lcd_putc,"\f");
printf(lcd_putc, "user ID:\n");
printf(lcd_putc,code);
output_high(PIN_C0);
output_high(PIN_A2);
delay_ms(1000);
output_low(PIN_A2);
output_low(PIN_C0);

 if(strcmp(code,passa))
 {
   output_high(PIN_A3);
   printf(lcd_putc,"\f user name:\n Ashvaany");
   printf(code);
   delay_ms(500);
   output_low(PIN_A3);
 }
```

```
 else if (strcmp(code,passb))
 {
   output_high(PIN_A3);
   printf(lcd_putc,"\f user name:\n Egambaram");
   printf(code);
   delay_ms(500);
   output_low(PIN_A3);
 }
 else
 {
   output_low(PIN_A3);
   printf(lcd_putc,"\f user name:\n not found");
   printf(code);
 }

}
```

APPENDIX D

C CODE – SECONDARY SYSTEM

```
#include <16F876A.h>
#include <string.h>


#fuses HS,NOLVP,NOWDT,NOPROTECT
#use delay(clock=20000000)
#use rs232(baud=9600, xmit=PIN_C6, rcv=PIN_C7, PARITY=N)

void main() {

char i,j;

char code[10];

char code1[11] = {"0012655155"};

char code2[11] = {"1234567890"};




   setup_adc_ports(NO_ANALOGS);
   setup_adc(ADC_OFF);
   setup_spi(SPI_SS_DISABLED);
   setup_timer_0(RTCC_INTERNAL);
   setup_timer_1(T1_DISABLED);
   setup_timer_2(T2_DISABLED,0,1);
```

```
 set_tris_b(0x00);
 set_tris_c(0x80);




while(TRUE){


     if (kbhit())
     {
       for(j=0;j<10;j++)
       code[j]=i=fgetc();
     }

if
((code[0]==code1[0])&&(code[1]==code1[1])&&(code[2]==code
1[2])&&(code[3]==code1[3])&&(code[4]==code1[4])&&(code[5]
==code1[5])&&(code[6]==code1[6])&&(code[7]==code1[7])&&(c
ode[8]==code1[8])&&(code[9]==code1[9]))

{
       output_high(PIN_B1);
       output_high(PIN_B2);
       delay_ms(3000);
       output_low(PIN_B2);
       output_low(PIN_B1);
}
```

```
  else if
((code[0]==code2[0])&&(code[1]==code2[1])&&(code[2]==code
2[2])&&(code[3]==code2[3])&&(code[4]==code2[4])&&(code[5]
==code2[5])&&(code[6]==code2[6])&&(code[7]==code2[7])&&(c
ode[8]==code2[8])&&(code[9]==code2[9]))

  {
         output_high(PIN_B1);
         output_high(PIN_B2);
         delay_ms(3000);
         output_low(PIN_B2);
         output_low(PIN_B1);
  }

   else
  {
   output_low(PIN_B1);
   output_low(PIN_B2);
  }
  code[j]=0;

  }

 }
```