UNIVERSITI TEKNOLOGI MALAYSIA

DECLARATION OF THESIS / UNDERGRADUATE PROJECT REPORT AND COPYRIGHT

Author's full name : **MUHAMMAD AFIQ BIN BAHARIN**

Date of Birth : **27th May 1990**

Title : **STUDENT ATTENDANCE SYSTEM BY FINGERPRINT**

Academic Session : 2013/2014

I declare that this thesisis classified as:

☐	**CONFIDENTIAL**	(Contains confidential information under the Official Secret Act 1972)*
☐	**RESTRICTED**	(Contains restricted information as specified by the organization where research was done)*
☑	**OPEN ACCESS**	I agree that my thesis to be published as online open access (full text)

I acknowledged that Universiti Teknologi Malaysia reserves the right as follows:

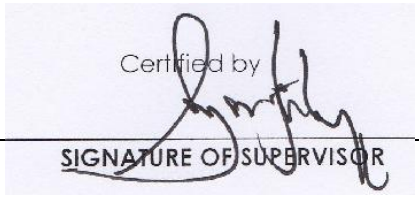
1. The thesisis the property of Universiti Teknologi Malaysia
2. The Library of Universiti Teknologi Malaysia has the right to make copies for the purpose of research only.
3. The Library has the right to make copies of the thesis for academic exchange.

SIGNATURE

900527-14-5243

(NEW IC NO/PASSPORT)

Date: **18th JUNE 2014**



SIGNATURE OF SUPERVISOR

PROF. DR. RUZAIRI BIN HJ. ABD RAHIM

NAME OF SUPERVISOR

Date: **18th JUNE 2014**

NOTES: * If the thesis is CONFIDENTAL or RESTRICTED, please attach with the letter from the organization with period and reasons for confidentiality or restriction.

"I hereby declare that I have read this thesis and in my opinion this thesis is sufficient in terms of scope and quality for the award of the degree of Bachelor of Engineering (Electrical – Mechatronic)."

Signature : .....

Name : PROF. DR. RUZAIRI BIN HJ. ABD RAHIM

Date : JUNE 2014

STUDENT ATTENDANCE SYSTEM BY FINGERPRINT

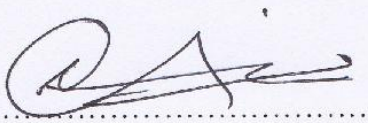
MUHAMMAD AFIQ BIN BAHARIN

This thesis was submitted in fulfillment of the requirements for the award of the Degree in Bachelor of Engineering (Electrical – Mechatronics)

Faculty of Electrical Engineering
Universiti Teknologi Malaysia

JUNE 2014

"I declare that this thesis entitled "Student Attendance System by Fingerprint" is the result of my own research except as cited in the references. The thesis has not been accepted for any other degree and is not currently submitted in candidature of any other degree".

Signature : 

Name : MUHAMMAD AFIQ BIN BAHARIN

Date : JUNE 2014

Specially dedicated to my beloved mother, Mrs. Shariah binti Sarijo, my dad, Mr. Baharin bin Aminuddin, my brother, Mohd Aizat bin Baharin and all my friends who have encouraged, support, counseled and inspired me throughout the journey of this project.

ACKNOWLEDGEMENT

First and foremost, all praise to Allah, the Almighty, for His blessings and guidance for giving me inspiration and strength to complete this project. I would also like to express my gratitude to those who gave me the possibility whether direct or indirectly, to finish this projecting the given time.

I am deeply indebted to my supervisor, Professor Dr. Ruzairi bin Hj Abd Rahim for the knowledge and assistance that he always give throughout the progress of this project. Thank you Professor Dr. Ruzairi and I am glad to work under your guidance.

My appreciation also goes to my family, who has always support me and encourage me with their love for all these years. Especially to my brother, who alwaysshared some of his excellent opinion or idea for this project. Without them, I would probably been stress out and give up.

Not to forget, to my friends; Nur Afifah,Amsyar, Khazin, Alif and SEM member's batch 2011/2014 which always support me in this project. I want to thank them all for all their interest, help and support. Thank you very much.

ABSTRACT

Attendance system is a system that records the attendance of their employee's time entered and left the office or factories. In Malaysia, school activities such as curriculum and co curriculum are compulsory. It is one of the ministry of Education Malaysia policies is to provide holistic and balance curriculum and co curriculum to expend the diversity of the student potential of individual to excel. To achieve that goal, the need to increase the quality in monitoring, reporting and marking for curriculum and co curriculum program. However, most of the attendance systems that are being used in school still are in traditional method. There is no standardization from the government in this system. Taking attendance by roll calling is a failure. It gives a huge burden to the teacher and a lot of time waste over responses of student. Then, the recording needs to analyze and report to admin. Waste of paper happen in this process. These propose a research on developing and design a system that can automatically record student's attendance then somehow calculate the percentage of student's attendance and their mark. Therefore, the goal of this project is to design an automated attendance system that can record and analysis student's attendance and then generate the percentage of student's attendance and their mark using biometric. Biometric is use because it is the part of our body and it is most secure identification. For this project, fingerprint will be chosen as identification. This project also needs to design the Graphical User Interface for the system so that it can be used by end user easily. Hope, this project could reduce the work burden of the teacher and also able to improve the quality in monitoring, reporting and marking for curriculum and co curriculum program and also be used in Malaysia.

ABSTRAK

Sistem kehadiran adalah sistem yang merekodkan kehadiran masa pekerja memasuki dan meninggalkan pejabat atau kilang. Di Malaysia, aktiviti-aktiviti sekolah seperti kurikulum dan ko kurikulum adalah wajib. Ia merupakan salah satu dasar Kementerian Pelajaran Malaysia untuk menyediakan keseimbangan kurikulum dan ko kurikulum yang holistik untuk kepelbagaian potensi individu pelajar untuk cemerlang. Untuk mencapai matlamat itu, keperluan untuk meningkatkan kualiti dalam pemantauan, pelaporan dan permarkahan untuk kurikulum dan ko kurikulum program. Walau bagaimanapun, kebanyakan sistem kehadiran yang digunakan di sekolah masih berada dalam kaedah tradisional. Tiada keseragaman dari kerajaan dalam sistem ini. Mengambil kehadiran secara memanggil nama adalah kegagalan. Ia memberikan beban yang besar kepada guru dan banyak masa terbazir menunggu jawapan dari pelajar. Kemudian, rekod kehadiran tadi perlu dianalisis dan dilaporkan kepada pihak pengurusan. Pembaziran kertas berlaku dalam proses ini. Ini mencadangkan penyelidikan kepada pembangunan dan reka bentuk satu sistem yang boleh secara automatik merekod kehadiran pelajar kemudian mengira peratusan kehadiran pelajar dan markah mereka. Oleh itu, matlamat projek ini adalah untuk merekabentuk sistem kehadiran automatik yang boleh merekod kehadiran pelajar kemudian menganalisis dan menjana peratusan kehadiran pelajar dan markah mereka menggunakan biometrik. Biometrik digunakan kerana ia adalah sebahagian daripada badan kita dan ia adalah pengenalan yang paling selamat. Untuk projek ini, cap jari akan dipilih sebagai pengenalan. Projek ini juga perlu untuk mereka bentuk Antara Muka Pengguna Grafik (GUI) untuk system ini supaya ia boleh digunakan oleh pengguna secara mudah. Berharap, projek ini dapat mengurangkan beban kerja guru

dan juga dapat meningkatkan kualiti dalam pemantauan, pelaporan dan permarkahan untuk program kurikulum dan ko kurikulum dan juga boleh digunakan di Malaysia.

TABLE OF CONTENT

LIST OF TABLE

LIST OF FIGURES

LIST OF ABBREVIATIONS

GUI	-	Graphical User Interface
APA	-	American Payroll Association
RFID	-	Radio Frequencies Identification
GPS	-	Global Positioning System
VB	-	Visual Basic
CCD	-	Charged-Couple Device
CMOS	-	Complementary Metal-Oxide Semiconductor
UART	-	Universal Asynchronous Receiver Transmitter
PC	-	Personal Computer
COTS	-	Commercial Off-The-Shelf
USB	-	Universal Serial Bus

LIST OF APPENDICES

Chapter 1

Introduction

1.0 Overview

This chapter will explain briefly about the background, problem statement, objective and the scope of the project. The summary and the outline of the thesis are also included in this chapter.

1.1 Introduction of Attendance System

Attendance system is a system that records the attendance of their employee's time entered and left the office or factories. Employer have a huge burden on how to keeping track on their employee. This system ensures the company to know what does the employee do and how disciplines are they. This times and attendance system let the employer know the exact amount of wages for their employee in the same time it make it difficult to cheat on giving wages due to the employee also know their working hour. This system also can motivate to the employee to come to

work early and indirectly increase company productivity. Furthermore, this attendance system can help employer planning on the future outcome of their company.

For an organization like school, teachers need to track student attendance by calling name one by one or counting person by person every day in every starting school day. Than the teachers need to calculate the percentage of every student attendance at the end of the month. Again, the records need to rewrite in every student record book to show to their parent about their children activities in school.

In Malaysia, school activities such as curriculum and co curriculum are compulsory. Student must come to the school on the weekend to actively join club and sport. The students have to bring along student activities book to record the attendance which is sign by the supervisor teacher of the organization. This student activities book will be gathering up and then need to calculate the percentage of student attendance at the end of the year. This percentage will differentiate the mark of the student.

1.2 Background of Problem

One of the ministry of Education Malaysia policies is to provide holistic and balance curriculum and co curriculum to expend the diversity of the student potential of individual to excel.(1) To achieve that goal, the need to increase the quality in monitoring, reporting and marking for curriculum and co curriculum program.

However, most of the attendance systems that are being used in school still are in traditional method. There is no standardization from the government in this

system. Taking attendance by roll calling is a failure. It gives a huge burden to the teacher and a lot of time waste over responses of student. Then, the recording needs to analyze and report to admin. Waste of paper happen in this process.

These propose a research on developing and design a system that can automatically record student's attendance then somehow calculate the percentage of student's attendance and their mark and with a software development to easily view and access the data.

1.3 Objective of Study

An attendance system is important in monitoring student activities. It is also important to show the recording and analysis to the student's parent. Therefore this research aims to achieve the following objectives:

i. To perform literature survey on the history and the existing attendance system.
ii. To design an automated attendance system that can record and analysis student's attendance and then generate the percentage of student's attendance and their mark.
iii. To design a Graphical User Interface for the system so that it can be used by end user easily.
iv. To demonstrate the system's capabilities in identify the correct identification of the user.

1.4 Scope of Study

The scopes for this research are to design an attendance system for recording student's attendance and automatically calculate their attendance's percentage and their mark. To record student's attendance, fingerprint device are needed to recognize and identify the identification of the student. Fingerprint device is used as an input reader for this research. This is included in this scope of project. And lastly, for the purpose to use by end user, a software development isincluding in designing the Graphical User Interface (GUI) for easier access on the system.

1.5 Summary

The flow and the progress of the project are summarized into a flow chart attached into Appendix A.

Chapter 2

Literature Review

2.0 Overview

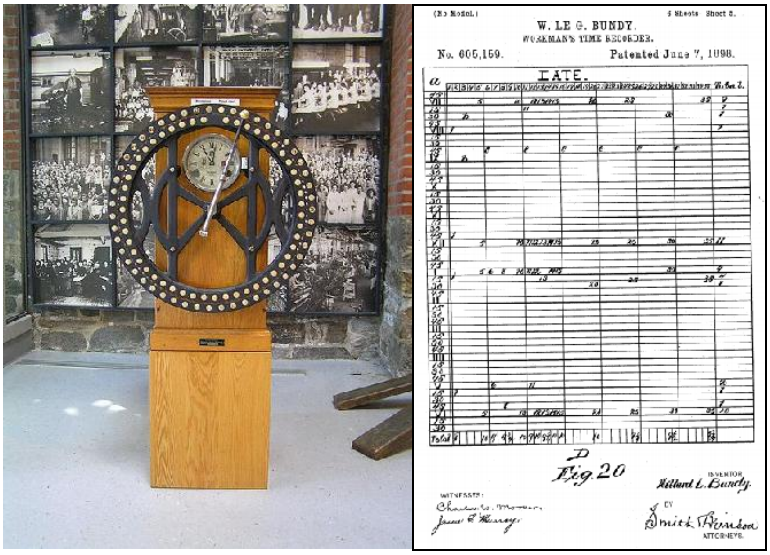
In this chapter will discuss the history of attendance system and the automated attendance system. The commonly device that may be used in automated system and the comparison between fingerprint device and RFID device are also been described to further understand the project system. This chapter also be explaining about the component and devices used in the system.

2.1 History of attendance system

In early 1800s, new technological advances forces major changes to the economies of Europe and the United State ushered in sweeping changes for both manufacturing and transportation. With the arrival of machine based manufacturing, the traditional family farms and small family cottage industries that produce small

amount of good moving from their rural farm home to the cities to work in industrialized factories. To get extra money for their family, children often work right alongside to their parent. It give benefits to the factory owner due to the child labor is cheap. However, the government starts stepped in and help to improve factory condition and regulate on how many hour people were made to work especially for children. So, factory owner needed a way to keep track of worker hours. (2)

It started on 1888, Willard Le Grand Bundy has patented a first and new invention on time and attendance recording call Bundy Key Recorder shown in Figure 2.1(a)(3). The only purpose of that equipment is to record time of the employee entered and left the factory. The employee need to insert a set of key into a fixed position than it will print number on a paper which is in fixed space shown in Figure 2.1(b). This gave the factory owner the official record on working hours of the employee. The owner of the factory can calculate the amount of wages should be given to the employee accurately. The employee also can know the exact working hour making the employers difficult to cheat on paying the wages. It also can motivate the employee to comes early indirectly increase factory productivity.



(a) (b)

Figure 2.1: (a) Bundy Key Recorder (b) Paper printed number

In the year 1984, Daniel M. Cooper has invented an improved version of the first time clock created by Willard Bundy called Rochester shown in Figure 2.2 (4). Each employee would be given a call time card. The equipment stamps the exact date and time on the time card when the employee insert the card to the machine. At the end of the month, employer would collect the entire employee's time card and calculate the amount of wages should be given to the employee.

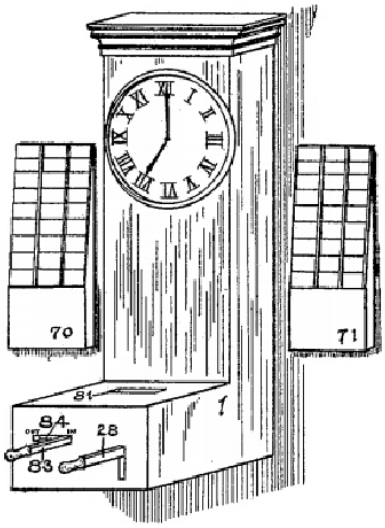


Figure 2.2: The Rochester Time Clock

Now days, several technologies have been develop so the system can be operate automatically. It ensures the organization to easily record and analyse the time and attendance of their employees. This automatic system can use electronic tag, magnetic stripe cards and biometrics (hand, finger, facial and etc.) by touching or swiping to identify the user and record the attendance and time. The recorded data than automatically be transfer to the computer with some process calculations to generates information about the time and attendance of the user. This automated system can reduce the risk of error that commonly done by using manual system and it is easy to analyse.

2.2 Automated Attendance System

Automated attendance system is a system that record time of employee entered or left with using electronic devices such as electronic tag, magnetic stripe, barcode scanner, or biometric[1] by touching or swiping the scanner or reader to identify the identities of the user and then record the attendance and times(5). The recorded data then automatically transfer to the computer software and do some process calculations to generate information about the time and attendance of the user in a split second. It also can generate the wages of the employee.

2.2.1 Advantages of Automatic System than Manual System

A lot of advantages can get by using automated system rather than manual system. In this section will discuss three main advantages which are reducing human error, increasing productivity of the company and save company's money.

Reduce human error

A manual attendance system require employee to report their worked hour per day after the work day. Unfortunately, these system exposures to inaccurate time tracking. The American Payroll Association (APA) estimates that the rate of human error in time card preparation is between 1% and 8%(6). As for example, the handwriting in the time sheet is illegible and difficult to determine the hours that the employee work. Miscalculate happen.

[1] Biometric is referring to the identification of human characteristic by their traits or behavior.

This result inaccurate time will affect the employee wages either it higher or lower. By using automated attendance system, it can reduce the risk of human error and easy to handle. The automated system will automatically calculate the hours of employee work per day.(7)

Increase productivity

In manual system, employer need to collect the time card and calculate for every employee otherwise the employee does not get their wages. This system gives a burden on a worker to calculate employee wages. Tendency to give wages late to the employee is high. It will drop employee's motivation and reduce company productivity. With automated attendance system, the process will be smooth and the operation is more efficient and convenient(8). This will increase the productivity of a company. With this, employer can detect staffing overhead and overcame the problem immediately. Management with using this system can be lot more effective.

Green Ecosystem

Manually, recording attendance are using paper card. The paper card needs to be given to all employees. Compared to automated system, with using only electronic device and software, it appears to be a paperless system. It can help in reducing pollution and global warming. Furthermore, this will increase efficiency and save a lot of money.

2.2.2 Commonly device use in attendance system

There are several devices can be used to identify the identification of the user. There are electronic tagging, magnetic stripe and biometric.

Electronic tag

Electronic tag is a tracking and identifying electronic device attached to a person or vehicle. There are two devices that make the system working, GPS and radio frequencies identification(RFID). For GPS, the system is similar to the GPS automobiles that use satellite to locate the position of the tag. The satellite will transmit the information on the coordinates of the tag to the central hub which is the owner of the tag. RFID is a wireless non-contact use of radio frequency electromagnetic fields to transfer data. There are two types of RFID tag which are passive RFID tags and active RFID tags. Figure 2.3 shows the example of electronic tag in market.

Figure 2.3: Example of Electronic Tag

For passive RFID tags, it does not have its own power supply. It require signal from external source to activate the transmission information from the tag to a reader terminal. It can only use in a short distance within a meter. To make it be read in a further distance, battery assisted need to be inserted to increase the signal strength(9). Active RFID tags contain internal power source so it can be transmit data in a longer distance.

Magnetic stripe cards

Magnetic stripe usually see at the back of a credit card. This stripe is made up from of tiny-based magnetic particles in a plastic like film. Each particle is a very tiny bar magnet about 20 millionth of an inch long(10). Magnetic stripe can be written due to the tiny magnet that can be magnetized in either North Pole or South Pole direction. Magnetic stripe reader is use to read data from a magnetic card. It is microcontroller-based devices that contain signal amplifiers and line drivers. Magnetic stripe consist 3 level of track. Usually only two level tracks is use and the extra track is reserve if the other track is spoilt or broken. Figure 2.4 shows the example of magnetic stripe card.

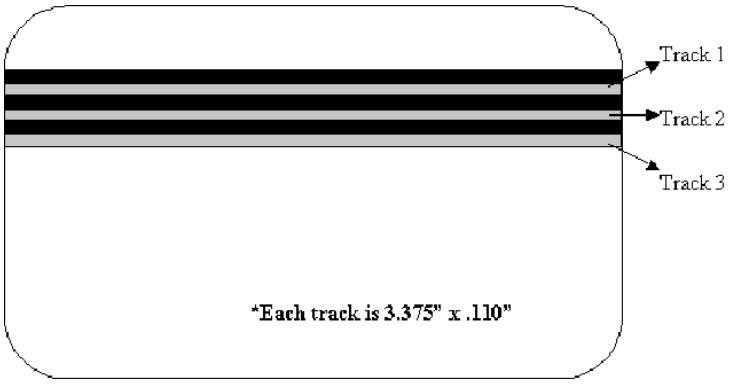


Figure 2.4: Magnetic Stripe Card

Biometrics

Biometrics is refers to the identification of human by their characteristics or trait(11). Biometric identifiers are the distinctive, measureable characteristics used to label and describe individual. It can be classified into two groups which are physiological characteristic and behavioural characteristic. Physiological characteristic are related to the shape of the body including the fingerprint, face recognition, DNA, palm print, iris recognition, retina and scent. While behavioural characteristics are related to the pattern of behaviour of a person such as voices.

2.3 Comparison between RFID and finger print scanner

There two methods on how to read an identification of a person which by using RFID scanner or biometric device by using finger print scanner. Both devices can differentiate the identification by reading code in the RFID tag by RFID scanner or from our finger print then the scanner will encode some codes which are difference from other user. During this research, there are several pros and cons that will find in the finding. From this research, we can divide into three categories which are securities level, cost effectiveness and complexity of the programming and coding.

2.3.1 Securities Level

RFID system is using tag that will receive a signal and transmit back the signal to a scanner. That process will generate some code to the system. So, RFID system must use some sort of tag. In our case, RFID card will be used as a tag. This card will represent the identification of the owner. In order to apply the system, users need to swap or touch the RFID card to the scanner. For that reason, the identification is not secure. The scanner can read the RFID card even though the user is not the owner of the card. The RFID card can be passing to other person to swap or touch the scanner or it may be stole and use for the bad purpose just like in the movie. Furthermore, the RFID card also may be loss or leave at home or misplace. Although the RFID card can be renew, it is a waste of time and cost.

Compare to the finger print scanner, it is much more secure than the RFID system. It does not need tag or card. It just only needs the scanner and the person itself. We all know that our finger print is different to others. In that case, other person cannot register or apply to the system. So, no passing identification cases happen. Cases such as misplace or loss will not be tolerated due to the finger are part of our body except some bad incident happen to the user that may lost their finger or finger print. Therefore, using finger print scanner has high security level.

2.3.2 Cost Effectiveness

There are a lot of company that sold both RFID scanner and finger print scanner. For this discussion, we choose CYTRON as a supplier of the device due to the distance of the shop to UTM. Even the price is a little higher compare to other company but with including the traveling cost

CYTRON might get lower. The survey found out that there is several type of device that is suitable for this research.

For RFID scanner, the price range is between RM100 to RM140 not including the RFID card of the price RM3 per unit. Let say the system is operating for 50 users. So, we need 50 RFID card at the price of RM150. The total price needed for using RFID system is RM290 (estimate by using the highest price of the RFID scanner). Meanwhile, the price for finger print scanner in the range between RM450 to RM500 per unit. If we compare both devices, system with using RFID device is more cost effective than finger print scanner.

2.3.3 Complexity of the Programming and Coding

Microsoft Visual Studio 2010 software with Visual Basic (VB) language is used to design the GUI for this system. It is well known as user friendly software which it can be reducing the complexity in designing the system with the aid of toolbox bar in the software. Other than designing the GUI, signals to communicate the fingerprint device need to be fully understand. It is all depend on how much bit requires during transmitting and what signal will be receive back from the device. In that case, the complexity of the programming and coding is much easier.

Table 2.1: Comparison of RFID Scanner and Fingerprint Scanner

Low security system Allow buddy punching	SECURITY	High security system Eliminate buddy punching
Low cost	COST	High cost
Medium	COMPLEXITY OF THE SYSTEM	Medium

Table 2.1 above shows the comparison between RFID scanner and finger print scanner. From this, for the system that security is not that important RFID scanner can be choose due to the low cost. While, for the system that need high security system fingerprint scanner need to be choose even the price is high.

2.4 How fingerprint device works?

Traditionally, to capture an image of fingerprint a finger is inked and stamped on a paper. Then, the image is digitized by scanner. Modern technology has developed a device that can be capture the fingerprint image and automatically transfer to the database. Inside of the fingerprint device is installed a sensor that will capture the fingerprint image(12). There are various sensor in market that can be used to do those works which are optical sensor, capacitive sensor and solid state sensor.

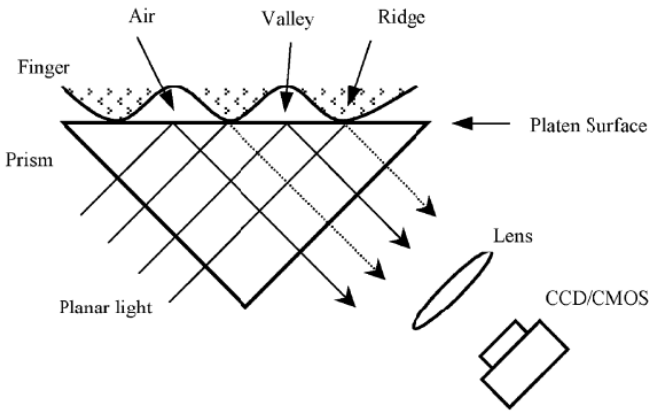


Figure 2.5: Optical Sensor

Optical fingerprint capture is typically base on the frustrated total internal refraction phenomenon as shown in Figure 2.5 above. When the finger touched the platen, the refractive index is difference between the ridge and valley. The light that passes through the glass upon valleys is totally refracted and the light that passes through the glass upon ridge is not refracted. The refracted light is focused by a lens onto a CCD or CMOS camera where the image is captured.

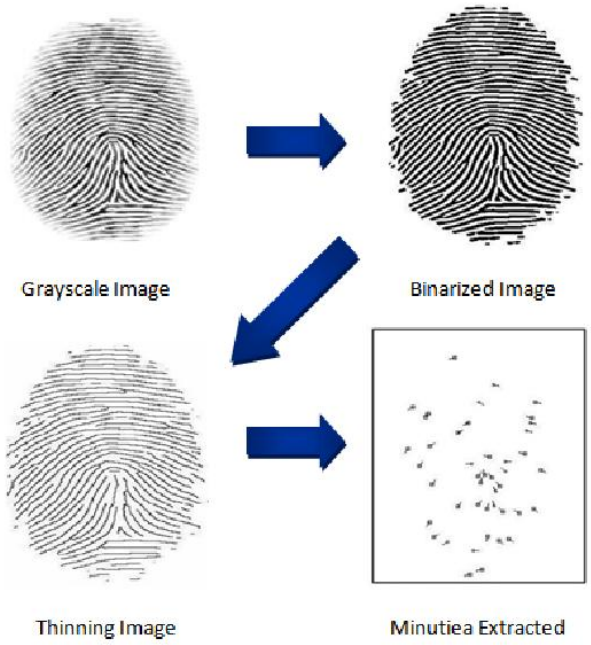


Figure 2.6: Image of Fingerprint throughout the Process

Figure 2.6 shows the process happen from capturing the image until the data stored in database. The image captured by the sensor is in grayscale image. To make it easier to analyze, binarization method is used. It will turn the grayscale image to a

binary image by fixing the threshold value. Then it continues by thinning the binarized image. This process is to reduce the ridge line into single pixel. Then, the minutiae are extracting from the thinning layer. The point can be determined by these ridge patterns which are bifurcation and ending ridge shown in Figure 2.7 below(13). The minutiae location and the minutiae angles are derived after minutiae extraction. The minutiae data is stored on matrix format. Then it continues to further process whether to store in the database or matching template from the database(14).

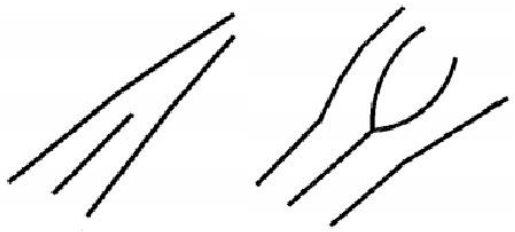
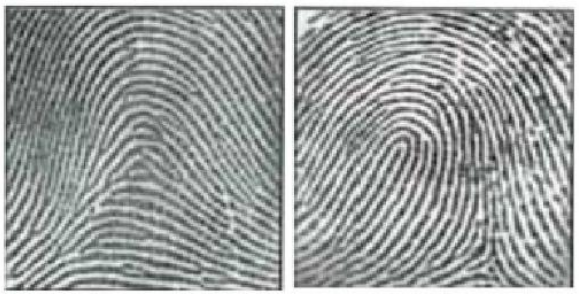


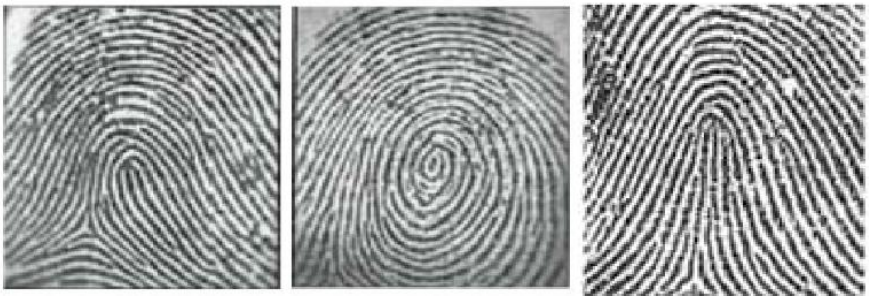
Figure 2.7: (a) Ended ridge (b) Bifurcation Ridge

2.5 Classification of Fingerprint

According to the current most widely used Galton-Henry system, the fingerprint is divided into five classifications shown in Figure 2.8 which are arch, tented arch, left loop right loop and lastly whorl(15).



Arch Left loop



Right loop Whorl tented Arch

Figure 2.8: 5 Classification of fingerprint

Arch: Fingerprint lines start from side of the finger and end at the other side, do not return and on the core points and delta point.

Tented Arch: Like an arch fingerprint, but graphic Center upward rise in the vertical direction, equivalent to a core and a delta on the same vertical line.

Left Loop: A circular pattern is thata fingerprint lines access from one direction then back from the same direction after a rotation around. To the left is Left Loop. There is a core and a delta at the lower left.

Right Loop: To the right is Right Loop. There is a core and a delta at the lower right.

Whorl: At least one fingerprint stripe rotate into a closed curve around the center, there are two core points in center, a triangular point on each side when the cores are not in the same vertical line, here will form a double helix

Chapter 3

Research Methodology

3.0 Overview

This chapter will discuss on research methodology. The discussion covers about the selected device and the process involved which includes circuit designs and software development. These steps were planned out carefully to ensure that the whole project progress smoothly with minimal setbacks. A flow chart of the project was made to have a clear time setting throughout the whole project and so that the project will be finished in time. UART (UC00A UART to USB converter) was used as the interface between the fingerprint device and the database.

3.1 Hardware Development

The main research for this project is the system. So this project will use COTS. To develop an attendance system, UART (UC00A UART to USB converter) is chosen as medium to interface between the fingerprint device and the database. The fingerprint device used SNR-FPR-SM-630.

3.1.1 SNR-FPR-SM630

SM630 Optic Fingerprint Verification Module shown at Figure 3.1 is manufactured by Miaxis Biometrics Co, Ltd. It consists of optical fingerprint sensor, high performance DSP processor and Flash. It has multifunction such as fingerprint login, fingerprint deletion, fingerprint verification, fingerprint upload, fingerprint download and etc. Compared to products of similar nature, SM630 has the following unique features:

Figure 3.1: SNR-FPR-SM630 device

- **High Adaptation to Fingerprints**
  When reading fingerprint images, it has self-adaptive parameter adjustment mechanism, which improves imaging quality for both dry and wet fingers. It can be applied to wider public.

- **Algorithm with Excellent Performance**
  SM630 module algorithm is specially designed according to the image generation theory of the optical fingerprint collection device. It has excellent correction & tolerance to deformed and poor-quality fingerprint.

- **Easy to Use and Expand**

  User can easily develop powerful fingerprint verification application systems based on the rich collection of controlling command provided by SM630 module. All the commands are simple, practical and easy for development.

- **Low Power Consumption**

  Operation current is less than 80mA, especially good for battery power occasions.

- **Integrated Design**

  Fingerprint processing components and fingerprint collection components areintegrated in the same module. The size is small. And there are only 4 cablesconnecting with HOST, much easier for installation and use.

3.1.2 UART UC00A (UART to USB Converter)

UART (Universal Asynchronous Receiver Transmitter) is a microchip that convert parallel signal to serial signal or vice versa. It is usually a part of an integrated circuit used for communication over a peripheral device port. There are several types of UART that being use in market such as Zilog Z8440, Motorola 6850 and Rockwell 65C52(16).UART UC00A is chosen because of the features that are compatible with this project. With a low price, it is suitable for research purpose. This UART UC00A need 5V supply to operate but no external source required due to standard output voltage of USB port on the computer is 5V.

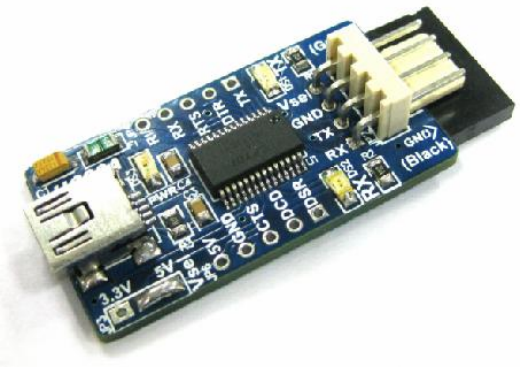


Figure 3.2: UART UC00A (UART to USB Converter)

3.2 Software Development

Part of the software design was the GUI that integrates the fingerprint device with a PC(17). The purpose of this is so that it is easier for the user to conduct the recording process as the image captured from the fingerprint device will be displayed on the PC monitor.

3.2.1 Flow Chart of the programming

It started with the main menu, which are user need to select admin for log in as administration or select student for student log in. then it will jump to the selected part shows in Figure 3.3.

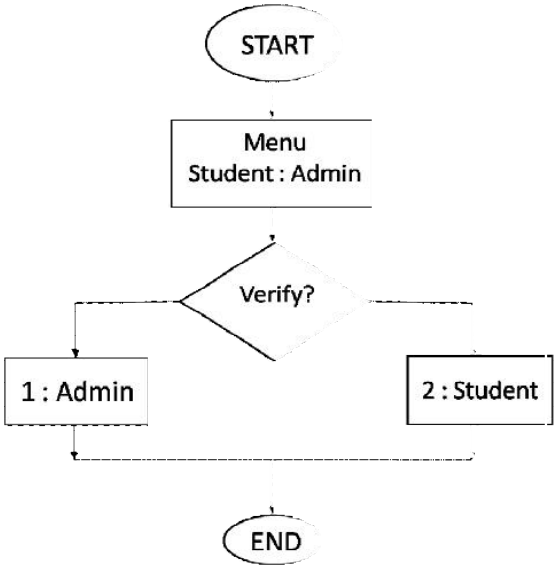


Figure 3.3: Flowchart for Main Menu

When admin were selected, it will jump to the admin subroutine shows in Figure 3.4 below. Then the user needs to log in by entering their username and password. If verified, it will continue to the admin menu otherwise the error will display and the user needs to re-enter again the username and the password. In admin menu, there are four options to choose which is edit, view, new registration or logout. All option will jump to their subroutine and then it will return back to the admin menu.

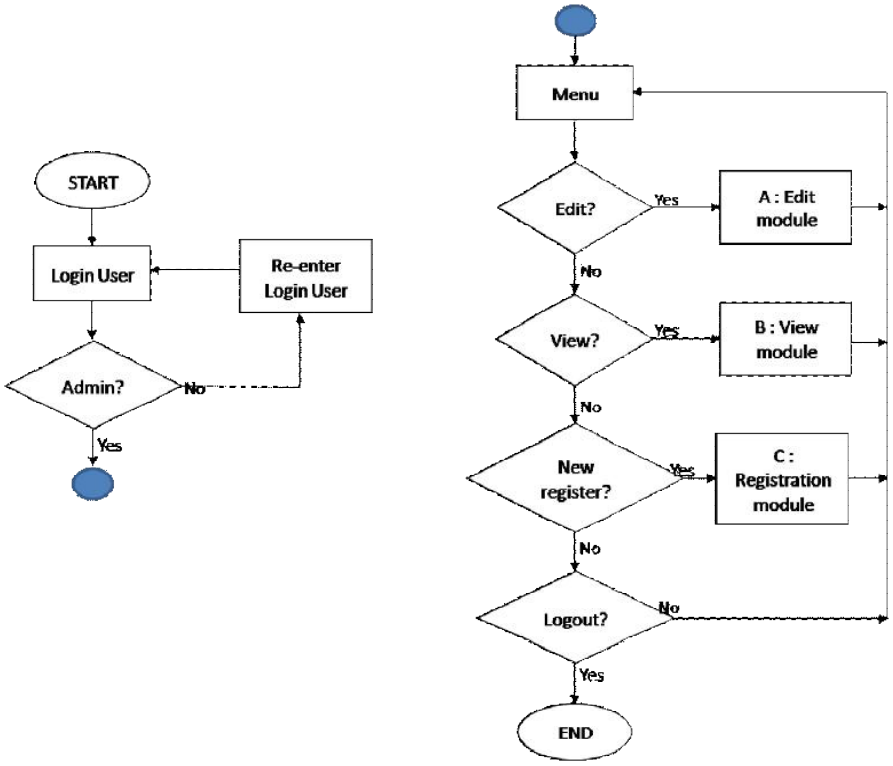


Figure 3.4: Flowchart for Administration Subroutine

In student subroutine, the user also needs to insert username and password to log in as a student. Compare to admin subroutine, student subroutine can only view their profile and record. There only two option in student subroutine menu which is either view profile or view record as shown in Figure 3.5 below.

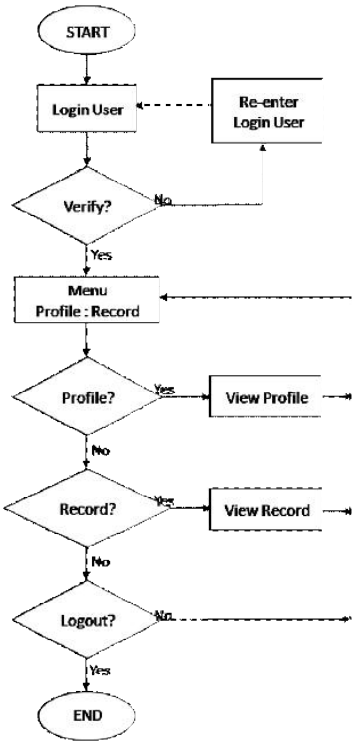


Figure 3.5: Flowchart for Student Subroutine

3.2.2 GUI Concept

GUI is used to make the system easy to handle by end user. To make a user-friendly system, the GUI needs to be plan perfectly. Microsoft Visual Studio 2010 is used to designing the GUI. This is the concept used for this project shown in Figure 3.6.

Figure 3.6: GUI Concept

3.3 Experiment

Several matter need to be caution and test to ensure the overall system is operate according to plan. It divided into two which are securities and human error. In securities, the fingerprint device needs to be tested to prevent it to be hacked. Firstly, the device will be tested with difference finger and with difference people. Secondly, the device need to be tested in which finger is most suitable to use. It will be tested by tap in all five fingers repeatedly. Lastly, the most important test in securities is the tendencies of the device recognize the originality of the fingerprint. It will be tested by making fake fingerprint by printing the fingerprint on paper or print the fingerprint on tape. This fake fingerprint will than tap on the device. In human error, it may happen when the end user mistaken in entering the data, the device is not connected with the computer or the most important the user accidently reset the whole system. These need to be caution in developing the GUI.

3.4 Conclusion

In this chapter, all the process involved in constructing the Automated attendance system were discussed and described in detail in order for the reader have a further understanding of the system.

Chapter 4

Experiment and Result

4.0 Overview

This chapter will discuss on experiment and the result of the experiment. The discussion covers about the procedures on conducting experiment, the result of the experiment and the conclusion of the experiment. This experiment wascarried out carefully to ensure the system is in high performance and a quality system.

4.1 Experiment and Precaution

In developing a good embedded and a quality system, several matter need to be caution and test to ensure the overall system is operate according to plan. It divided into two categories which are securities and human error.

4.2 Test for Securities

In securities, the fingerprint device needs to be tested to prevent it to be hacked or operate by difference user. In this category, it will be divided into three experiments which were tested to undergoing several circumstances in daily life. The first experiment is to test the device capability in differentiate differences fingerprint. Second, to test which fingers is the most suitable and accurate to use for the system. The last experiment in this categories is to test the tendencies of the device recognize the originality of the fingerprint.

4.2.1 Test the capability of the device differentiate differences fingerprint

To do this experiment, the fingerprint scanner (SNR-FPR-SM-630), the UART to USB converter (UC00A) and 20 fingerprints is needed. A simple basic GUI was developed to operate this experiment as shows in Figure 4.1 below.The source code for the system can be seen on Appendix D.

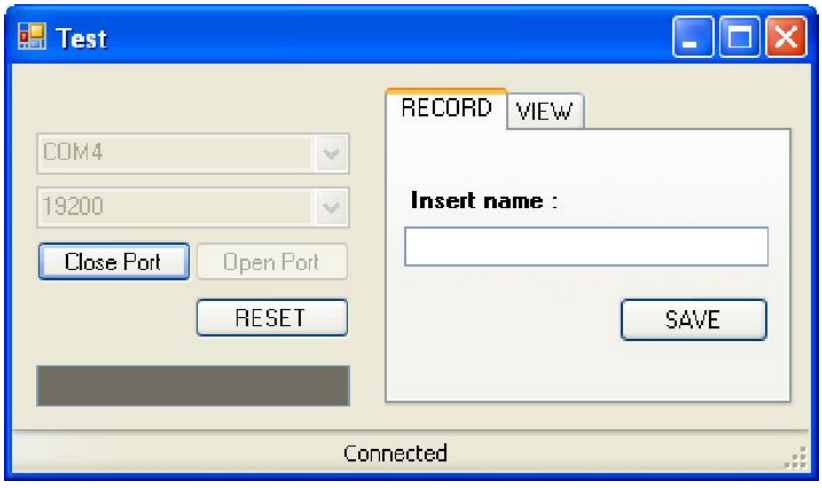


Figure 4.1: Software for testing the device.

The procedures of the experiment are shows below:-

Procedure taken in recording fingerprint

a. The deviceis connected to the system.
b. ‘RECORD’ tab was selectedfrom the display window.
c. User’s name was inserted in the blank box and ensures no repeated name as shown in Figure 4.2 below. ‘SAVE’ button was clicked.

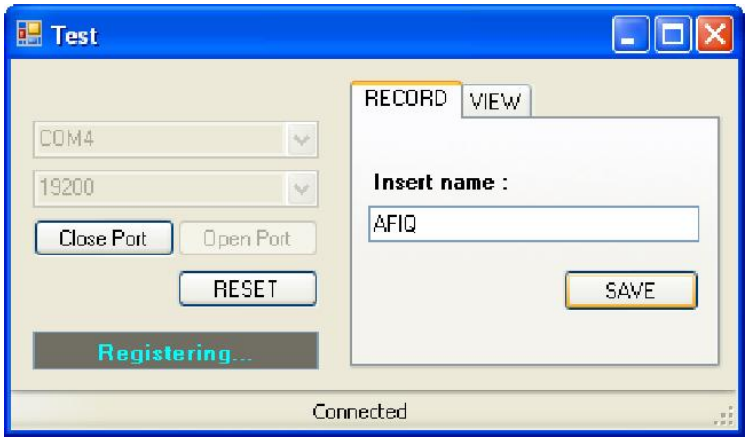


Figure 4.2: Registering a fingerprint window.

d. A ‘beep’ sound appeared from the scanner and the scanner lighted up.
e. User’s fingerprint then tabbedon the scanner and the scanner started recording and verified twice as shows in Figure 4.3 below.

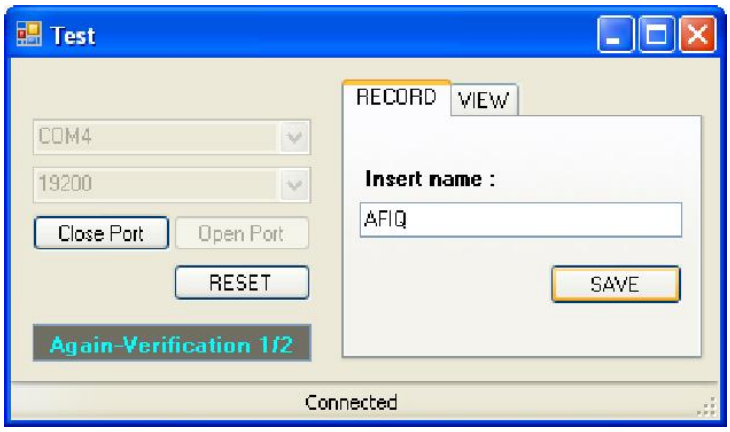


Figure 4.3: Verifying the fingerprint

f. A message box appeared to inform the registration is success as shows in Figure 4.4 below. If fail, procedure c will be repeated.

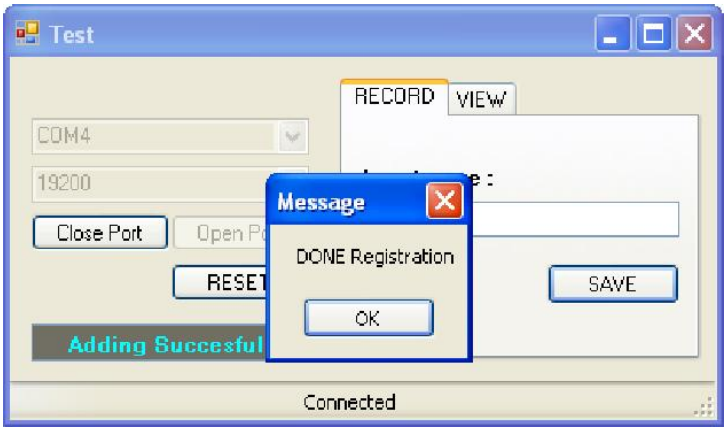


Figure 4.4: Message box shows the registration is done.

g. 'OK' button was clicked on the message box and the process is repeatedto record new fingerprint.

Procedure taken in testing the device in differentiate fingerprint.

a. The device is connected to the system ensure that the device is connected to the system.
b. 'VIEW' tab was selected from the display window as shows in Figure 4.5 below.

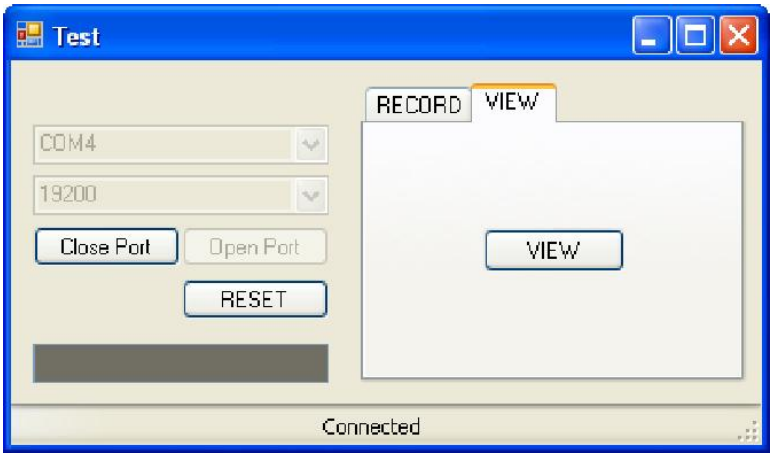


Figure 4.5: View tab selected.

c. ‘VIEW’ buttonwas clicked.

d. A ‘beep’ sound appeared from the scanner and the scanner lighted up.

e. Theuser’s fingerprint is tab on the scanner and the scanner verified the fingerprint.

f. A message box appeared to inform the user’s name as show in Figure 4.6 below.

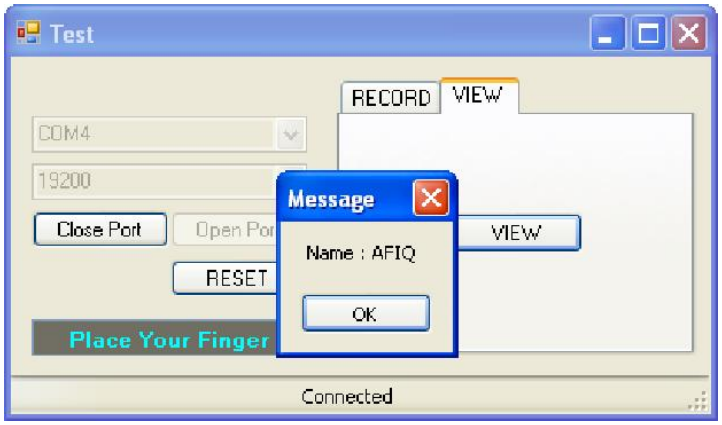


Figure 4.6: Message box displaying the user name.

g. Procedure ‘c’is repeatedif invalid.

h. The name appears in the message box was recorded in Table 4.1 (refer on Appendix B). The process than repeated with 19 other fingerprint randomly and for eachfingerprint was verified 3 times.

From the result recorded, there shows an equal name for every fingerprint tested on the device. It can conclude that this device is capable in differentiate fingerprint. It is also suitable for the system due to the device can manage to match on 1:N fingerprint.

Table 4.1: Record for Experiment 1

BIL	NAME	1ST VERIFICATION	2ND VERIFICATION	3RD VERIFICATION
1	**AFIQ**	AFIQ	AFIQ	AFIQ
2	**NUR AFIFAH**	NUR AFIFAH	NUR AFIFAH	NUR AFIFAH
3	**SHAFIQAH**	SHAFIQAH	SHAFIQAH	SHAFIQAH
4	**AKMAL**	AKMAL	AKMAL	AKMAL
5	**ABDUL AFIF**	ABDUL AFIF	ABDUL AFIF	ABDUL AFIF
6	**ALIF**	ALIF	ALIF	ALIF
7	**MOHD FIRDAUS**	MOHD FIRDAUS	MOHD FIRDAUS	MOHD FIRDAUS
8	**AMSYAR**	AMSYAR	AMSYAR	AMSYAR
9	**ARIFF**	ARIFF	ARIFF	ARIFF
10	**FEKRIE**	FEKRIE	FEKRIE	FEKRIE
11	**THARMIZI**	THARMIZI	THARMIZI	THARMIZI
12	**RAZALI**	RAZALI	RAZALI	RAZALI
13	**KHAZIN**	KHAZIN	KHAZIN	KHAZIN
14	**NAQIB**	NAQIB	NAQIB	NAQIB
15	**AZIZAH**	AZIZAH	AZIZAH	AZIZAH
16	**NORFATIHAH**	NORFATIHAH	NORFATIHAH	NORFATIHAH
17	**ALIP**	ALIP	ALIP	ALIP
18	**ALIAS**	ALIAS	ALIAS	ALIAS
19	**BAHARIN**	BAHARIN	BAHARIN	BAHARIN
20	**NAZ**	NAZ	NAZ	NAZ
21	**FIZAH**	FIZAH	FIZAH	FIZAH
22	**KAMAL**	KAMAL	KAMAL	KAMAL
23	**ASMAH**	ASMAH	ASMAH	ASMAH
24	**SITI**	SITI	SITI	SITI
25	**KARIM**	KARIM	KARIM	KARIM

4.2.2 Test the most suitable and accurate fingers to use for the system

This experiment will be use the same device and system in the previous experiment. For the fingerprint, five selected person was chosen. All five fingers are used in these experiments which are thumb, index finger, middle finger, ring finger and baby finger. The procedures are as follow:-

1. Records fingerprint
    a. The device is connected to the system.
    b. ‘RECORD’ tab was selectedfrom the display window.

c. User's name and type of finger (example: Afiq middle) was inserted in the blank box and ensures no repeated name as shown in Figure 4.7 below. 'SAVE' button was clicked.

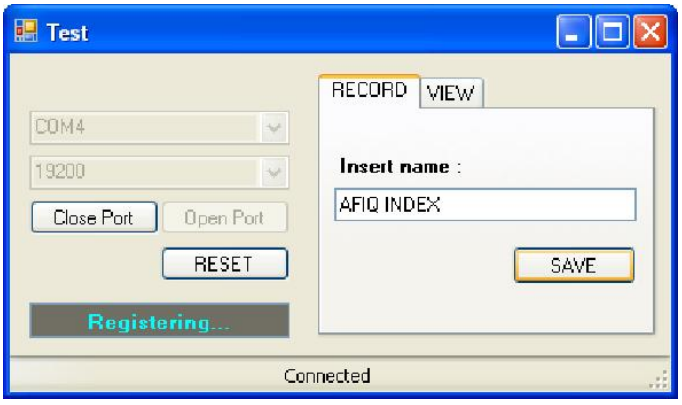


Figure 4.7: Registering a fingerprint window.

d. A 'beep' sound appeared from the scanner and the scanner lighted up.

e. User's fingerprint then tabbed on the scanner and the scanner started recording and verifying twice as shows in Figure 4.8 below.

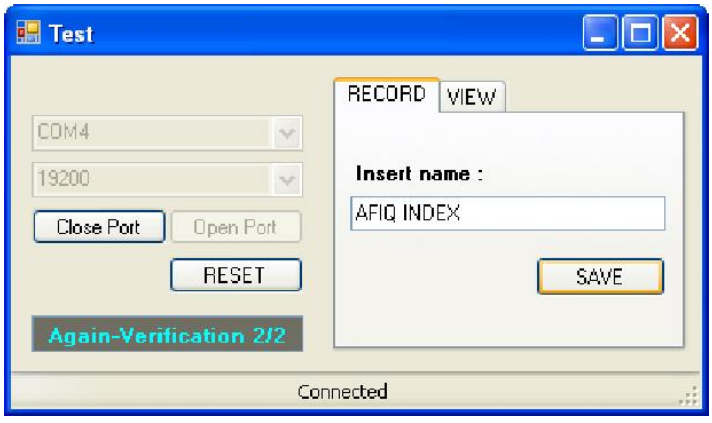


Figure 4.8: Verifying the fingerprint

f. A message box appeared to inform the registration is success as shows in Figure 4.9 below. If fail, procedure c will be repeated.

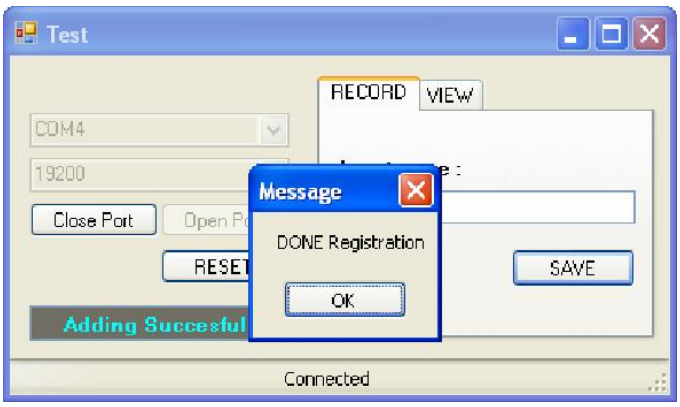


Figure 4.9: Message box shows the registration is done.

g. 'OK' button was clicked on the message box and the process is repeated to record new fingerprint.

2. Verifying the fingerprint.
   a. The device is connected to the system.
   b. 'VIEW' tab was selected from the display window and 'VIEW' button was clicked.
   c. A 'beep' sound appeared from the scanner and the scanner lighted up.
   d. The user's fingerprint is tab on the scanner and the scanner verified the fingerprint.
   e. A message box appeared to inform the user's name as show in Figure 4.10 below.

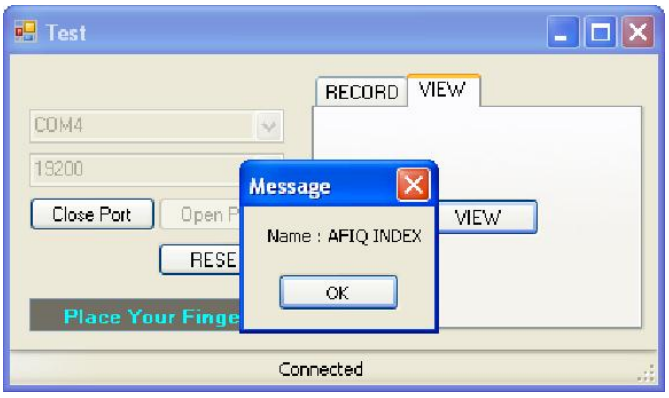


Figure 4.10: Message box displaying the user name.

f. Message in the message box appeared was record in Table 4.2 below. (Success = 1 or Fail = 0).
g. The procedure was repeated with others fingerprint randomly and for each fingerprint was verified 3 times.

Table 4.2: Result on testing themost suitable fingers to use for the system

No	NAME	THUMB			INDEX			MIDDLE			RING			BABY		
1	AFIQ	1	1	1	1	1	1	1	1	1	1	1	O	1	O	1
2	ALIAS	1	1	1	1	1	1	1	1	1	O	1	1	1	1	O
3	AKMAL	1	1	1	1	1	O	1	O	1	O	O	1	O	1	O
4	NUR AFIFAH	1	1	1	O	O	1	1	O	1	1	O	O	O	1	O
5	ASMAH	1	1	O	1	1	O	1	1	O	O	1	O	O	O	O
	Percentage	93.3%			73.3%			80.0%			46.7%			40.0%		

Based on the result recorded in Table 4.2 above, thumb is the highest percentage that the device can recognize by 93.3% and the lowest finger is baby finger by 40%. This results shown that the largest surface area of the fingerprint, the easier the device can recognize. From this experiment, we can conclude that using thumb is the most suitable and have highest accuracy than others fingers on recognition of the fingerprint for the system.

4.2.3 Test the tendencies of the device being hacked

This experiment will be use the same device and system in the previous experiment. For the comparison, this experiment will be use fake fingerprint. There are two methods on faking a fingerprint that are well known to hacking the system which by printing the fingerprint on a piece of paper as shown in Figure 4.11 (a) and by ink the finger and tap it on a cello tape as shows as Figure 4.11(b).

(a) (b)

Figure 4.11: (a) Printed fingerprint(b) Inked fingerprint on cello tape.

Both methods will be used in this experiment. 3 fake fingerprints had been made for each method. The procedures are as follow:-

1. Records fingerprint

   The procedure in records the fingerprint is the same as experiment before.

2. Verifying the fake fingerprint.
   a. The device is connected to the system.
   b. 'VIEW' tab was selected from the display window and 'VIEW' button was clicked.
   c. A 'beep' sound appeared from the scanner and the scanner lighted up.
   d. The fake fingerprint (printed fingerprint) was tapped on the scanner and the scanner verified the fingerprint.
   e. A message box will appeared. The message in the message box was recordedin Table 4.3 below.
      (Success = 1 or Fail = 0).
   f. The process was repeated by using others fake fingerprint and was repeated 5 times for each fake fingerprint randomly.

Table 4.3: Result on testing the tendencies of the device being hacked

No	Method	Fake 1			Fake 2			Fake 3		
1	Cello Tape	O	O	O	O	O	O	O	O	O
2	Printed	O	O	O	O	O	O	O	O	O
Percentage of success		0.0%			0.0%			0.0%		

Based on the result from table 4.3 above, all test using fake fingerprints were failed. The fingerprint scanner that we use in this system (SNR-FPR-SM630) is an optical sensor (discuss in chapter 2.4). The fake fingerprint used in this experiment is flat and no ridge and valley due to the printed paper and the cello tape is plain surface. So, all the light that passes through the glass is not refracted and the CCD or CMOS camera cannot capture the image and process. In conclusion, the device can prevent the system being hacked by using these two methods.

4.3 Survey on Human Error

In human error, it may happen when the end user mistaken in entering the data such as the device is not connected with the computer or the most important the user accidently reset the whole system. These need to be caution in developing the GUI. In such way, to identify the mistake that will happen when the end user operate, a survey will be held in testing the system operation by select a random person to use this system with a minimum instruction on what to do. All the important behavior will be record and improvement need to be done. This survey will be done periodically to enhance a better GUI.

Chapter 5

Limitation, Recommendation and Conclusion

5.0 Overview

This chapter will discuss on limitation and recommendation for this project. The discussion will cover on the specifications, the performance and the output obtain from the system. It will also cover on recommendation in future planning for development.

5.1 Output Result

From this research, software that will automatically calculate student's attendance mark had been successfully developed and the hardware had been assembled as shown in Figure 5.1 below. This project was developed accordingly based on the test and survey obtained in experiment discussed in chapter 4. A setup file has been successfully developed for installation to make it easier for the end user.

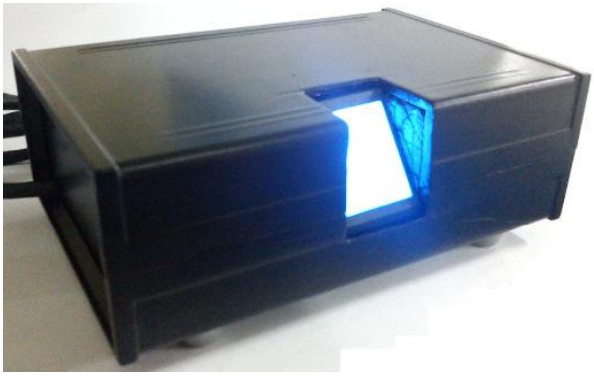


Figure 5.1: Hardware of the system

5.1.1 Manual to install the software

For installing the system, click the Student Attendance System.msi icon on the setup folder as shown in Figure 5.2 below then wait for the setup window that will appear as shown in Figure 5.3. Follow the instruction until the installation is finish.

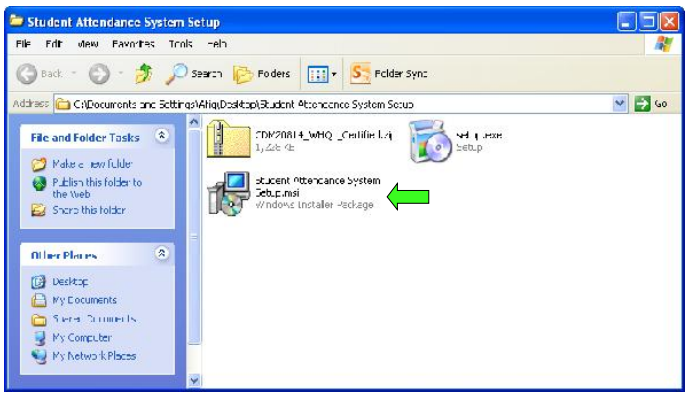


Figure 5.2: Student Attendance System Setup File

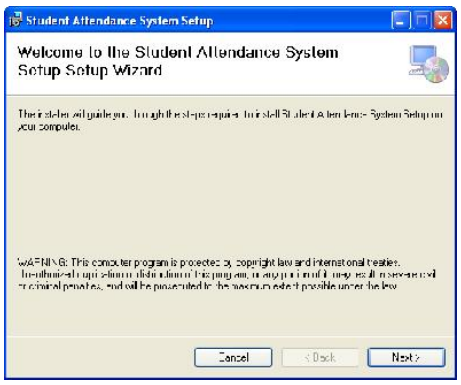


Figure 5.3: Student Attendance System Setup Window

When the installation is finish, the software will run automatically and a splash screen will appeared as shown in Figure 5.4 below determined that the software in running and will be follow up by the starting GUI shows in Figure 5.5.

Figure 5.4: Splash Screen

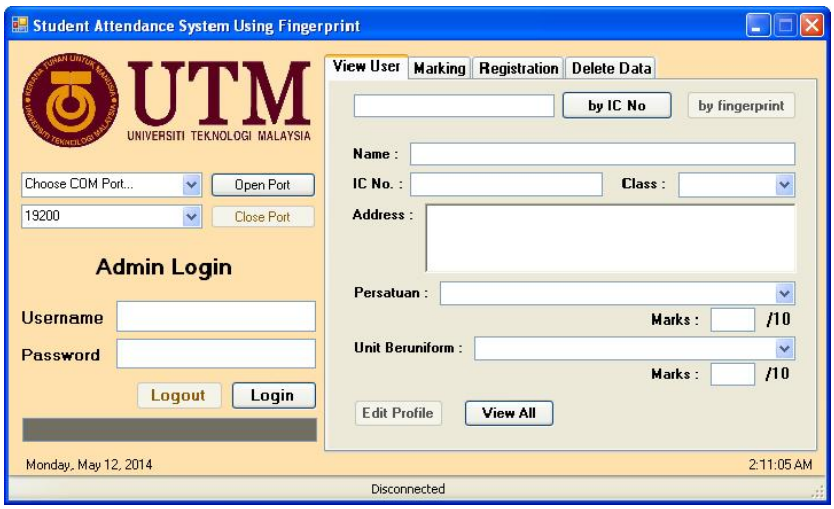


Figure 5.5: Starting GUI for the system

For the first time user, there need to install the hardware by unzipped the CDM20814_WHQL_Certified.zip folder and install the software for hardware unless the computer will not recognize.

5.1.2 User's manual

For the start, only the 'VIEW' and 'MARKING' tab is enableto view unless the admin is login. At the bottom of the window (status bar) will show the device status. At start, the device is not connected and 'Disconnected' will be display on the status bar. This will disable the entire button that will connect to the device. To connect the system to the device, follow the step below:-

Steps to Open Serial Port

1. Make sure all connections are correctly connected and fingerprint reader is switched on.
2. Firstly, choose the correct COM port.The COM port used for the device will be appeared automatically.
3. Select 19200 bps for the Baud Rate and click "Open Port". Figure 5.6 below shows the port opening setting.

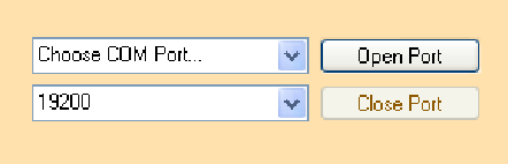


Figure 5.6: Port opening setting

4. Bottom of the window will shows "Connected" if the application establishes a connection successfully.

When the connection is successful, the entire button will enable and the device will work properly. For register and delete a record, the admin need to login first to enabled all tab by inserting the user name and password at admin login setting shows in Figure 5.7 below and click 'Login' button.

Figure 5.7: Admin login setting

To register a student, click at 'Registration' tab. If the tab is disable, please login as admin. To register, follow the step below:-

Register a student record

a. Please ensure that the device is connected to the system.
b. Select a 'Registration' tab on the display window.
c. Fill in the entire blank boxes with the student profile then click 'REGISTER' button.
d. A 'beep' sound will appear from the scanner and the scanner will light up.
e. Put the user fingerprint on the scanner and the scanner will start record and verify twice. Example shows in Figure 5.8 below.

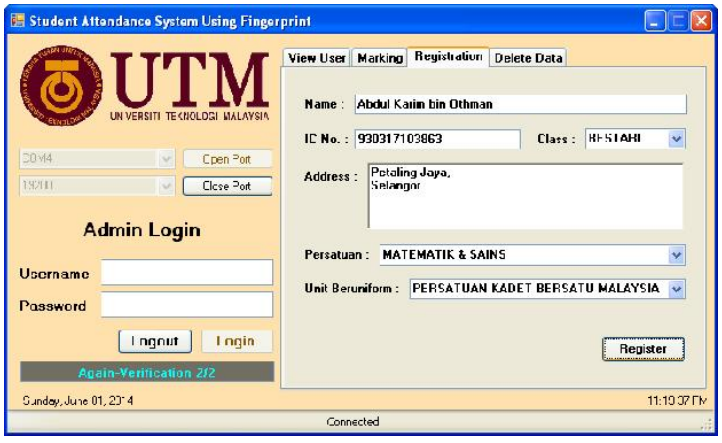


Figure 5.8: Example of registering a record

f. A message box will be appearing as shown in Figure 5.9 below to inform the registration is success or failure. If fail, repeat procedure c.

Figure 5.9: Message box shows done registration

g. Click 'OK' button on the message box and repeat the process to record new fingerprint.

To view student profile and marks, 'View User' tab need to be clicked. There are two options in viewing the record. View record by scan the fingerprint and by identification number. The two options can be select at the top of the tab display as shown in Figure 5.10 below.

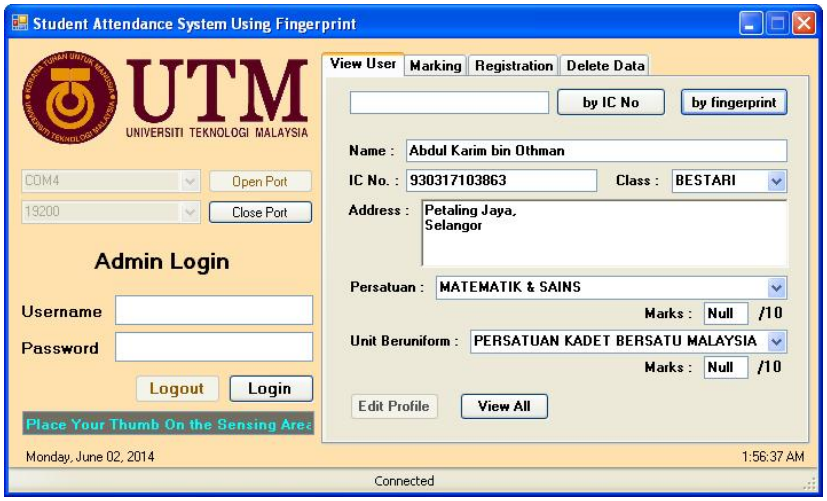


Figure 5.10: 'View User' tab window display

The 'Null' that appear at the marks box determine that the club never held a meeting otherwise it will appear the mark of the attendance over ten according on how much meeting it held. The equation is:-

$$Marks = \frac{No.of\ attendance}{No.of\ meeting} \times 10$$

This system can also view all student marks by clicking 'View All' button at the bottom of the tab. By clicking the button, a new window will appear as shown in Figure 5.11 below. The analysis can be made from this window. All the data regarding the attendance of the student will be seen directly. It is also can be printed directly by clicking the 'Print' button on the right below of the window.

Figure 5.11: Student's data window

To record attendance, click 'MARKING' tab and the tab will display marking option as shown in Figure 5.12 below. There are two options which are record attendance by fingerprint ('MARK' button) or by identification number ('By IC No.' button). The radio buttons need to be select between 'Persatuan' and 'Unit Beruniform' before clicking the button.

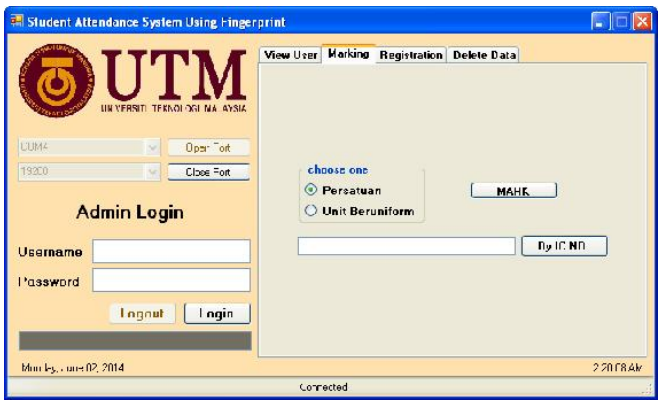


Figure 5.12: 'Marking' tab window display

The attendance will record automatically and a message box will appear to determine the recording is success or fail. Figure 5.13 below shows the example of the message box.

Figure 5.13: Example of a success marking

To delete a record, click at 'DELETE' tab. Figure 5.14 shows the display window when 'DELETE' tab was clicked.

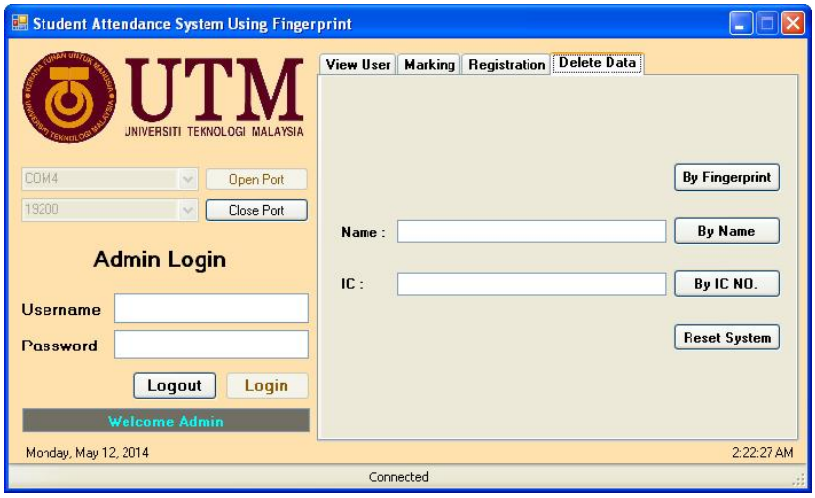


Figure 5.14: Display window when 'DELETE' tab was clicked

Based on the figure, there are four buttons can be click which means it has four options can be chosen. To delete a record by scanning a fingerprint, click 'By Fingerprint' button and the device will light up ready to scan and delete the fingerprint.

The second option is deleting a record by name. The blank box beside the 'By Name' button need to be filled with the correct name before click the button otherwise a message box will appear to tell there is no user. It is the same as the third option which is deleting a record by identification number. The blank box besides the 'By IC No.' button needs to be filled with a correct identification number to delete a record.

The last option is to reset or to delete the entire record. By clicking 'RESET SYSTEM' button, it will automatically delete all the data in the system which means it will reset the system.

5.2 Specifications

This project specification is obtained from the datasheet of the circuit used in this system. Table 5.1 below shows the specification for this project.

Table 5.1: Specification of the system

Storage Capacity (Device)	80 fingerprints
System Storage Capacity	> 2000 users
Security Level	7/10
Baud Rate	19200
Matching/Search Time	< 1.0 sec
False Accept Rate (FAR)	< 0.0001%
False Reject Rate (FRR)	< 0.01%

Based on the table above, the fingerprint template that can be stored in the device is 80 fingerprints. Hence, for the system storage, it can manage to store more than 2000 users due to the storage method is using computer which it has larger memory. The 7/10 security level is based on the tendencies of the device being hacked and it has been tested in experiment discussed in chapter 4. Furthermore, the system itself is secure by admin registration. For this system, it using 19200 baud rate to communicate between the device and computer. Matching fingerprint and its template only takes less than 1 second. FAR is the tendency the device accepting the false fingerprint and FRR is the tendency the device reject the correct fingerprint. For FAR is less than 0.0001% and for FRR is less than 0.01%.

5.3 Limitation

Every system and device has their own limitation including for this system. Based on the specification discussed before, the storage capacity is limited to 80 fingerprints and it is not suitable for a big group such as school. Due the template is stored in the device, the capability is limited to specific group and it is difficult if the device is broken. If it happens, the entire fingerprint template may be loss.

This designed system has a limited in assigning the co curriculum group. The system was designed with a specific group and it is not flexible and if it has any addition regarding a new group, the systems need to have overall changes in the programing.

5.4 Future development

In future development, the monitoring and attendance system can be extended by using both RFID and biometric. As for suggestion, by using RFID tagging system, the student activity can be automatically recorded when entering or living the room or area. It also can be further develop in monitoring system. The movement of the user can be seen in a layout of the area by using the same system.

5.5 Conclusion

In conclusion, the overall project research in develop an attendance system for student co curriculum has successfully developed. Although it has limitation, hope this project give an idea in continuing developing the system in this area for its benefit.

Chapter 6

Project Management

6.0 Overview

This chapter will discuss on the management for the whole process in developing this project. It will cover on research scope, research time and research budget. Research scope and research time had been tabulated on Gantt chart in project schedule to give a clear view and guideline in time management of this project. The cost of this project also will be discussed in this chapter.

6.1 Project Schedule

In completed the FYP1, this project is divided into three categories which are project background study andliterature review, research methodologyand designing system.Based on the Gantt chart for semester one shown in Table 6.1 below,the first and the second task was lag behind for a week. It is because the report needs to be

corrected before continuing the next task and it drag the second task for a week. However, for the next task and fort is commence as proposed on date.

Table 6.1: Gantt chart for semester one

	Sep				Oct				Nov				Dec				Jan			
Week	1	2	3	4	5	6	7	8	9		10	11	12	13	14	15	16	17	18	19
Task																				
Project Backgrounds Study & Literature Review			Proposed																	
			Actual																	
Research Methodology																				
Designing System																				
FYP1 presentation																				
FYP1 Report Submission																				
Attending Visual Studio courses																				

Table 6.2: Gantt chart for semester two

	Feb				Mar				Apr				May				June			
Week	1	2	3	4	5	6	7	8	9		10	11	12	13	14	15	16	17	18	19
Task																				
Purchasing component																				
Hardware assembling																				
Programming, Testing & Experiment																				
Data analysis																				
Data verification																				
Preparation for Exhibition																				
Exhibition (MERCED '14)																				
FYP2 Report Draf																				
Thesis Submission																				

Table 6.2 shows the Gantt chart for semester two. The purchasing of the component lags a week due to the market survey process to ensure a minimal project

cost. However, in hardware assembling, it finishes early from proposed time and the next task can start earlier.

6.2 Overall Cost

This project is focus on software development where no need hardware development. Because of that, this project will use COTS where it can be found online or at electronic shop. After market survey process, SNR-FPR-SM630 Fingerprint Reader (UART) had been choose due to the lower cost depend on others product with the same specification. Furthermore, the store that sells the product is near and they give one year warranty. In addition of the circuit, UC00A (USB to UART Converter) is needed to connect the scanner to the computer.

Table 6.3: Overall cost for the device

NO.	Component	Prices (RM)
1	SNR-FPR-SM630 Fingerprint Reader (UART)	450.00
2	UC00A USB to UART Converter	39.00
3	USB MiniB Cable	3.50
4	Plastic Box	7.00
	TOTAL	**RM 499.50**

Table 6.3 shows the overall cost for this project. The cost to build up this device is RM499.50.

REFERENCES

[1] Bahagian Perancangan dan Penyelidikan Dasar Pendidikan, Kementerian Pelajaran Malaysia".PELAN STRATEGIK INTERIM KEMENTERIAN PELAJARAN MALAYSIA 2011 - 2020". s.l. : Perpustakaan Negara Malaysia , 2012. ISBN 978-983-9522-26-6.

[2] Morrow, Jeffery. "History of Times and attendance system". Redcort Software. [Online] August 26, 2011. http://www.redcort.com/blog/2011/08/history-of-time-and-attendance-systems.

[3] Bundy, Willard Le Grand."worksman Times Recorder". US393205 A United Stated, November 20, 1888.

[4] Cooper, Daniel M."Workman's Time Recorder". US528223 A U.S., October 30, 1894.

[5] "Touch screen terminals used to record staff attendance". study, Protefil Group case.

[6] Richard. Co AdvantageHR Blog. "the benefits of Automating Time & Attendance". [Online] April 8, 2013. http://blog.coadvantage.com/2013/04/08/the-benefits-of-automating-time-attendance-management/.

[7] Smith, Christine."Advantages of Automating your Time and Attendance System". [Catalog] s.l. : Gate KeeperBusiness Solution, 2011 2011.

[8] HRMS, Sage. Slide Share. "Punching in to Improve Work Management". [Online] Jun 10, 2013. http://www.slideshare.net/SageHRMS/top-seven-benefits-of-an-automated-time-and-attendance-system.

[9] "RFID essentials". Bill Glover, Himanshu Bhatt. s.l. : O'Reilly Media, Inc., 2006. ISBN 0-596-00944-5.

[10] "The long life and imminent death of the mag-stripe card". Svigals, Jerome. s.l. : IEEE Spectrum, Vol. June 2012.

[11] "Biometric Authentication". Weaver, A.C. p. 96-97. DOI 10.1109/MC.2006.47, 2006.

[12] Tom's Guide Staff, Ryan Goodrich."Fingerprint Scanners: What They Are and How They Work". OCTOBER 10, 2013.

[13]" Latent Fingerprints, Setting Standards In The Comparison and Identification". Thornton, John. s.l. : 84th Annual Training Conference of the California State Division of IAI.

[14] "Minutia vs. Pattern Based Fingerprint Templates". 2003.

[15]"A World History of Fingerprint". Heidn, R. s.l. : Chinese People Public Security University Press, 2008, Vols. LISBN: 978-7-81109-789-4.

[16] Wikipedia. [Online] [Cited: November 4, 2013.] http://en.wikipedia.org/wiki/Universal_asynchronous_receiver/transmitter.

[17] "Graphical user interfaces". Martinez, W. L. s.l. : WIREs Comp Stat, 2011.

[18] Strovis Payroll. "Save Time and Money with Automated Time and Attendance Systems". [Online] March 11, 2011. http://strovispayroll.com/payrollblog/?p=64.

[19] Cytron Technologies Sdn. Bhd.FINGERPRINT READER (UART). Malaysia : Cytron Technologies Sdn. Bhd., May 2011.

[20] "The State of RFID Implementation and Its Policy Implications". IEEE-USA. USA : IEEE-USA, 2009.

[21] CYTRON.SM630 "Fingerprint Verification Module". Malaysia : CYTRON, 2008.

[22] "Employee Attendance Management System Using Fingerprint Recognition". Hemlata Patel, Pallavi Asrodia. s.l. : International Journal of Electrical, electronic and Computer Engineering, 2012. ISSN No. (Online) : 2277-2626.

[23] "Fingerprint Verification based on Gabor Filter" . Lavabya B. N., K B Raja, Venugopal K R. Bangalore : International Journal of Computer Science and Information Security, 2009, Vol. 6. ISSN 1947-5500.

[24] "Innovation in Fingerprint Capture Device". Lawrence O'Gorman, Xiongwu Xia. USA : Pergamon, 2001.

[25] "A comparison of common programming languages used in" . Mathieu Fourment, Michael R Gillings. Australia : BioMed Central, 2008.

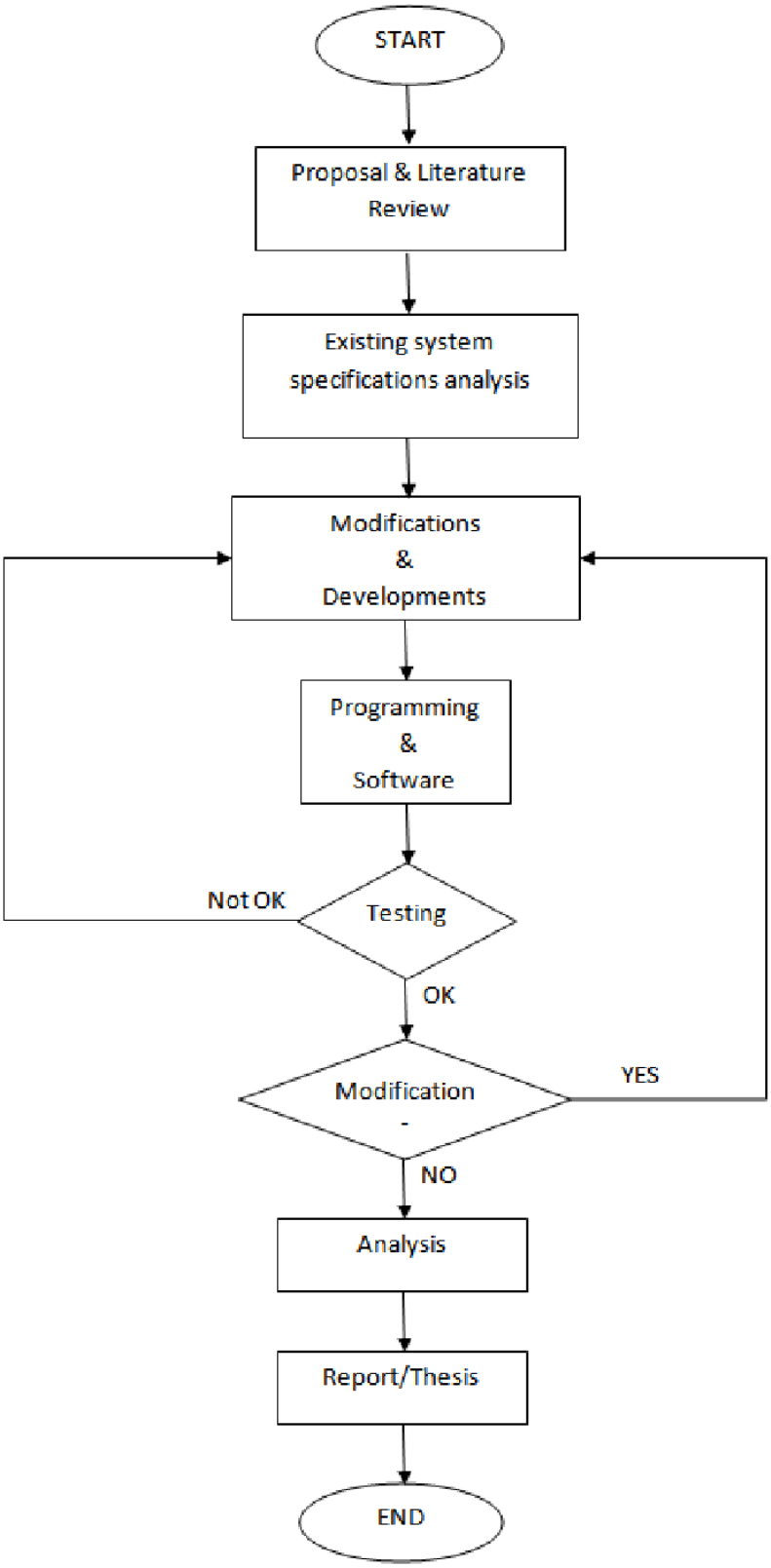
[26] O'Gorman, Lawrence.An Overview of Fingerprint Verification Technologies. s.l. : Veridicon Inc., 1998.

[27] "FINGERPRINT RECOGNITION USING MINUTIA SCORE" . RAVI. J, K. B. RAJA, VENUGOPAL. K. R. Bangalore : International Journal of Engineering Science and Technology, 2009, Vol. 1. ISSN: 0975-5462.

APPENDIXES

APPENDIX A

Project Flowchart



APPENDIX B

Source Code for Experiment

```
PublicClassForm1

Dim DataStudent AsString = "D:\project psm\testing.xlsx"
Dim error1 AsShort
Dim IDNoValid AsString
Dim Getrecord AsInteger
Dim portisopen AsBoolean
Dim nameStudent AsString

PrivateSub cboPort_DropDown(ByVal sender AsObject, ByVal e As System.EventArgs) Handles cboPort.DropDown

For i AsInteger = 0 ToMy.Computer.Ports.SerialPortNames.Count - 1     'check serialport avaibility and
cboPort.Items.Clear()                                                'add the port found as an item
cboPort.Items.Add(My.Computer.Ports.SerialPortNames(i))
Next

EndSub
''
'' Open Port Button
''
PrivateSub cmdPortOpen_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles cmdPortOpen.Click
If cboPort.Text = "Choose COM Port..."Then
MsgBox("Please select COM Port", vbExclamation, "Message")  'if port not chosen prompt user msgbox
Else
            SerialPort.PortName = cboPort.Text                  'get and set portname
            SerialPort.BaudRate = cboBaudRate.Text              'get and set baudrate
            SerialPort.Encoding = System.Text.Encoding.ASCII    'set encoding

Try
SerialPort.Open()                       'open serialport
                StatusPanel.Text = "Connected"'display status as connected
portisopen = True'indicator for opened port
Catch ex AsException
MsgBox(cboPort.Text &" already in used by other application.", vbExclamation, "Message")    'if port is not ready to open
prompt msgbox
Exit Sub
EndTry
'enable or disable control
            cmdPortClose.Enabled = True
            tabRecord.Enabled = True
            tabView.Enabled = True
            cmdPortOpen.Enabled = False
            cboPort.Enabled = False
            cboBaudRate.Enabled = False
EndIf

'Serial port opened and perform initialization to fingerprint reader
If portisopen = TrueThen

'Setting Timeout'
Dim a AsInteger
DimCommandSetTimeOut() AsByte = {&HF5, &H2E, &H0, &H7, &H0, &H0, &H29, &HF5}
SerialPort.DiscardInBuffer()                    'clear buffer
For i = 0 To 7
SerialPort.Write(CommandSetTimeOut, i, 1)   'send command through serialport
Next i
            a = 0
While a = 0
                a = checkbuffer(8)                          'check if 8 bytes in buffer
EndWhile
Dim number AsByte()
Dim receivedData AsString = ""
receivedData += SerialPort.ReadExisting()       'read serialport buffer
number = StrToByteArray(receivedData)           'string to byte array conversion

'Setting No Repetition'
DimCommandSetNoRepeat() AsByte = {&HF5, &H2D, &H0, &H1, &H0, &H0, &H2C, &HF5}
SerialPort.DiscardInBuffer()                    'clear buffer
For i = 0 To 7
SerialPort.Write(CommandSetNoRepeat, i, 1)  'send command through serialport
Next i
            a = 0
While a = 0
                a = checkbuffer(8)                          'check if 8 bytes in buffer
EndWhile
receivedData += SerialPort.ReadExisting()       'read serialport buffer
number = StrToByteArray(receivedData)           'string to byte array conversion
EndIf
EndSub
PublicSharedFunctionStrToByteArray(ByVal str AsString) AsByte()
Dim encoding AsNewSystem.Text.ASCIIEncoding()
Returnencoding.GetBytes(str)
EndFunction

'check buffer'
PublicFunctioncheckbuffer(ByVal lenght AsByte) AsByte

Dim flag AsBoolean
If SerialPort.BytesToRead < lenght Then'buffer not reaching desired capacity
flag = 0
```

```
Return flag                             'return 0
Else'buffer has enough data
flag = 1
Return flag                             'return 1
EndIf
EndFunction
''
'' check status
''
PublicFunctioncheckstatus() AsByte

Dim number AsByte()                         'define number as byte array
Dim receivedData AsString = ""'a string to temporary store everything from buffer
receivedData = ""'clear string
receivedData += SerialPort.ReadExisting()   'read buffer
number = StrToByteArray(receivedData)       'string to byte array conversion

IfLen(BitConverter.ToString(number)) = 23 Then
'lenght af the string = 23 eg. "F5-01-00-01-03-00-03-F5"
SelectCasenumber(1)   'select byte 2
'if we count character by character there are 23 characters which means we didn't receive more or less bytes
'Matching 1:N
Case 12                     ' 0X0C is decimal 12 reply of 1:N
Ifnumber(4) = 3 Then' successful    'select byte 5
Ifnumber(3) = 99 Then
Return 99   'admin
Else
Returnnumber(3)
EndIf

ElseIfnumber(4) = 8 Then'failed     'select byte 5
                        txtStatus.Text = "Time Out"
Return 0
ElseIfnumber(4) = 5 Then
                        txtStatus.Text = "User Not Found"
Return 0
EndIf

'Get the total fingerprint stored
Case 9
                    txtStatus.Text = "No. of fingerprints collected = "&number(3)
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                    error1 = 0
Return error1
'Add fingerprint operation (1)'
Case 1
SelectCasenumber(4)   'select byte 5
Case0  'successful'
                            txtStatus.Text = "Again-Verification 1/2"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                            error1 = 0
Return error1
Case 1
                            txtStatus.Text = "Error!"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                            error1 = 1
Return error1
Case 4
                            txtStatus.Text = "Full!"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                            error1 = 1
Return error1
Case 7
                            txtStatus.Text = "Error: User exists"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                            error1 = 1
Return error1
Case 8
                            txtStatus.Text = "Time out"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                            error1 = 1
Return error1
Case Else
                            txtStatus.Text = "Unknown Error!"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                            error1 = 1
Return error1
EndSelect
'Add fingerprint operation (2)'
Case 2
SelectCasenumber(4)   'select byte 5
Case0  'successful
                            txtStatus.Text = "Again-Verification 2/2"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                            error1 = 0
Return error1
Case 1
                            txtStatus.Text = "Error!"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                            error1 = 1
```

```
Return error1
Case 4
                                txtStatus.Text = "Full!"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                                error1 = 1
Return error1
Case 7
                                txtStatus.Text = "Error: User exists"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                                error1 = 1
Return error1
Case 8
                                txtStatus.Text = "Time out"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                                error1 = 1
Return error1
Case Else
                                txtStatus.Text = "Unknown Error!"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                                error1 = 1
Return error1
EndSelect
Case 3        'Add fingerprint operation (3)'
SelectCasenumber(4)   'select byte 5
Case0  'successful
                                txtStatus.Text = "Adding Succesfull"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                                error1 = 0
Return error1

Case 1
                                txtStatus.Text = "Error!"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                                error1 = 1
Return error1
Case 4
                                txtStatus.Text = "Full!"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                                error1 = 1
Return error1
Case 7
                                txtStatus.Text = "Error: User exists"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                                error1 = 1
Return error1
Case 8
                                txtStatus.Text = "Time out"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                                error1 = 1
Return error1
Case Else
                                txtStatus.Text = "Unknown Error!"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                                error1 = 1
Return error1
EndSelect
'Delete a fingerprint
Case 4
SelectCasenumber(4)   'select byte 5
Case0  'sucessful
'clear_record()
                                txtStatus.Text = " Record has been cleared"
txtStatus.Refresh()
'save_record()
                                error1 = 0
Return error1
Case 1
                                txtStatus.Text = "Error!"
txtStatus.Refresh()
                                error1 = 1
Return error1
EndSelect
EndSelect
Else
            txtStatus.Text = "something wrong"'this shouldn't happen
Return error1 = 1
EndIf
Return error1
EndFunction
''              FINGERPRINT COMMAND
'' add new record
PrivateSubAddNewRecord()
Dim i AsShort
Dim a AsShort
Dim error1 AsShort
Dim CHK1 AsInteger
Dim CHK2 AsInteger
Dim CHK3 AsInteger
        txtStatus.Text = "Place Your Finger"
```

```
txtStatus.Refresh()
        CHK1 = IDNoValid Xor&H1 Xor&H3
        CHK2 = IDNoValid Xor&H2 Xor&H3
        CHK3 = IDNoValid Xor&H3 Xor&H3
Dim user_1() AsByte = {&HF5, &H1, &H0, IDNoValid, &H3, &H0, CHK1, &HF5}
Dim user_2() AsByte = {&HF5, &H2, &H0, IDNoValid, &H3, &H0, CHK2, &HF5}
Dim user_3() AsByte = {&HF5, &H3, &H0, IDNoValid, &H3, &H0, CHK3, &HF5}
SerialPort.DiscardInBuffer()                'clear buffer
For i = 0 To 7
SerialPort.Write(user_1, i, 1)
'send 1st command through serial port
Next i
        a = 0
While a = 0
            a = checkbuffer(8)                      'check buffer
EndWhile
        error1 = checkstatus()                      'check reply
If error1 = 0 Then'proceed if no error returns
SerialPort.DiscardInBuffer()            'clear buffer
For i = 0 To 7
SerialPort.Write(user_2, i, 1)
'send 2nd command through serialport
Next i
            a = 0
While a = 0
                a = checkbuffer(8)                  'check buffer
EndWhile
            error1 = checkstatus()                  'check reply
If error1 = 0 Then'proceed if no error returns
SerialPort.DiscardInBuffer()        'clear buffer
For i = 0 To 7
SerialPort.Write(user_3, i, 1)
'send 3rd command through serialport
Next i
                a = 0
While a = 0
                    a = checkbuffer(8)              'check buffer
EndWhile
                error1 = checkstatus()              'check reply
EndIf
EndIf

EndSub

'' Matching fingerprint
''
PrivateSubMachtingFinger()
        Getrecord = Nothing
        txtStatus.Text = "Place Your Finger"
txtStatus.Refresh()

Dimbuff() AsByte = {&HF5, &HC, &H0, &H0, &H0, &H0, &HC, &HF5}

Dim i AsShort
Dim a AsShort
SerialPort.DiscardInBuffer()        'clear buffer
For i = 0 To 7
SerialPort.Write(buff, i, 1)    'send command through serialport
Next i

While a = 0
            a = checkbuffer(8)              'check buffer
EndWhile
        Getrecord = checkstatus()           'check reply

EndSub
''  reset fingerprint
PrivateSubDeleteAllFingerprint()

System.Threading.Thread.Sleep(1000)  ''delay
txtStatus.Refresh()
'Global delete to reset whole system
Dim a AsByte
DimdeleteAll() AsByte = {&HF5, &H5, &H0, &H0, &H0, &H0, &H5, &HF5}    'command
SerialPort.DiscardInBuffer()            'clear buffer
For i = 0 To 7
SerialPort.Write(deleteAll, i, 1)
'send command through serialport
Next i
        a = 0
While a = 0
            a = checkbuffer(8)                  'check buffer
EndWhile

        error1 = checkstatus()                  'check reply
txtStatus.Refresh()
EndSub
PrivateFunctionRegistration()
Dim complete AsString = False
Dim MyExcel AsNew Excel.Application
MyExcel.Workbooks.Open(DataStudent)
'insert
MyExcel.Sheets("Sheet1").activate()
MyExcel.Range("B2").Activate()
Do
If MyExcel.ActiveCell.Value >NothingOr MyExcel.ActiveCell.Text >NothingThen
MyExcel.ActiveCell.Offset(1, 0).Activate()
Else
Exit Do
```

```
EndIf
Loop
        MyExcel.ActiveCell.Value = txtRecord.Text     'insert name to excel
'close
MyExcel.Application.ActiveWorkbook.Save()
MyExcel.Workbooks.Close()
        MyExcel = Nothing
MsgBox("DONE Registration", MsgBoxStyle.OkOnly, "Message")
        txtRecord.Text = Nothing
Return complete = True
EndFunction
PrivateFunctionCheckValidID() AsString
        IDNoValid = Nothing
Dim MyExcel AsNew Excel.Application
MyExcel.Workbooks.Open(DataStudent)
MyExcel.Sheets("Sheet1").activate()
MyExcel.Range("B2").Activate()
Do
If MyExcel.ActiveCell.Value >NothingOr MyExcel.ActiveCell.Text >NothingThen
MyExcel.ActiveCell.Offset(1, 0).Activate()     'turun bwh stu kotak
Else
MyExcel.ActiveCell.Offset(0, -1).Activate()
                IDNoValid = MyExcel.ActiveCell.Value
Exit Do
EndIf
Loop
MyExcel.Application.ActiveWorkbook.Save()
MyExcel.Workbooks.Close()
        MyExcel = Nothing
Return IDNoValid
EndFunction
PrivateSubViewRecord()
Dim MyExcel AsNew Excel.Application
MyExcel.Workbooks.Open(DataStudent)
MyExcel.Sheets("Sheet1").activate()
MyExcel.Range("A1").Activate()
MyExcel.ActiveCell.Offset(Getrecord, 1).Activate()
nameStudent = MyExcel.ActiveCell.Value
MyExcel.Application.ActiveWorkbook.Save()
MyExcel.Workbooks.Close()
        MyExcel = Nothing
MsgBox("Name : "& nameStudent, MsgBoxStyle.OkOnly, "Message")
EndSub
''  Save button
PrivateSub btnSave_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnSave.Click
Dim Completed AsBoolean = False
        txtStatus.Text = "Registering..."
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
If txtRecord.Text = NothingThen
MsgBox("Please Fill in all require fields.", MsgBoxStyle.Exclamation, "Message")
            txtStatus.Text = "Failed"
Return
EndIf
        IDNoValid = CheckValidID()
AddNewRecord()
If error1 = 1 Then
MsgBox("User already exist", MsgBoxStyle.Exclamation, "Message")
            txtStatus.Text = "Failed"
Exit Sub
EndIf
Registration()
Return
EndSub
''  View button
PrivateSub btnView_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnView.Click
MachtingFinger()
If Getrecord = 0 Then
MsgBox("Invalid Please try again", vbExclamation, "Message")
Exit Sub
Else
ViewRecord()
EndIf
EndSub
PrivateSub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) HandlesMyBase.Load
        tabRecord.Enabled = False
        tabView.Enabled = False
EndSub
PrivateSub cmdPortClose_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles cmdPortClose.Click
SerialPort.Close()                  'close serialport
        StatusPanel.Text = "Disconnected"'display disconnected
'enable/disable/reset control
        cboPort.Enabled = True
        cboBaudRate.Enabled = True
        cmdPortOpen.Enabled = True
        tabRecord.Enabled = False
        tabView.Enabled = False
        cmdPortClose.Enabled = False
portisopen = False
        cboPort.Text = "Choose COM Port..."
        cboBaudRate.Text = "19200"
EndSub
''      button reset
PrivateSub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
DeleteAllFingerprint()
EndSub
EndClass
```

APPENDIX C

Source Code of the System

```
PublicClassForm1

Dim Amanah AsString = "AMANAH"
Dim Bestari AsString = "BESTARI"
Dim Cemerlang AsString = "CEMERLANG"
Dim BMelayu AsString = "BAHASA MELAYU"
Dim BInggeris AsString = "BAHASA INGGERIS"
Dim MatSains AsString = "MATEMATIK & SAINS"
Dim rekaCipta AsString = "REKACIPTA"
Dim PKBM AsString = "PERSATUAN KADET BERSATU MALAYSIA"
Dim KRS AsString = "KADET REMAJA SEKOLAH"
Dim KPA AsString = "KADET PERTAHANAN AWAM"
Dim DataStudent AsString = "C:\Student Attendance System Setup\Database\datastudent.xls"
Dim SetNo AsInteger
Dim IDNoValid AsString
Dim portisopen AsBoolean
Dim Getrecord AsInteger
Dim deleteName AsString
Dim deleteIC AsDouble
Dim error1 AsShort = 0
Dim success, success1, success2, success3 AsInteger

PrivateSub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) HandlesMyBase.Load
        RegTab.Enabled = False
        btnViewFinger.Enabled = False
        DeleteDataTab.Enabled = False
        MarkButton.Enabled = False
        btnLogout.Enabled = False
        btnResetSystem.Enabled = False
        btnDeleteFinger.Enabled = False
        RegisterButton.Enabled = False
        btnEdit.Enabled = False
        Timer1.Enabled = True
        DateToday.Text = Format(Now, "D") ' Today
EndSub
''
''  button register
''
PrivateSub Button5_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles RegisterButton.Click
Dim Completed AsBoolean = False
        txtStatus.Text = "Registering..."
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
Errortesttxtbox()
If success >NothingThen
MsgBox("Please Fill in all require fields.", MsgBoxStyle.Exclamation, "Message")
Return
EndIf
        IDNoValid = CheckValidID()
AddNewRecord()
If error1 = 1 Then
MsgBox("User already exist", MsgBoxStyle.Exclamation, "Message")
Exit Sub
EndIf
Registration()
Return
EndSub
''
''  Registration
''
PrivateFunctionRegistration()
Dim complete AsString = False
Dim SelectPersatuanReg AsString
Dim SelectUniformReg AsString = Nothing
Dim MyExcel AsNew Excel.Application
MyExcel.Workbooks.Open(DataStudent)

'insert
MyExcel.Sheets("StudentID").activate()
MyExcel.Range("B2").Activate()

Do
If MyExcel.ActiveCell.Value >NothingOr MyExcel.ActiveCell.Text >NothingThen
MyExcel.ActiveCell.Offset(1, 0).Activate()     'turun bwh stu kotak
Else
Exit Do
EndIf
Loop

        MyExcel.ActiveCell.Value = txtName.Text     'insert name to excel
MyExcel.ActiveCell.Offset(0, 1).Activate()
        MyExcel.ActiveCell.Value = txtIC.Text       'insert IC nom to excel
MyExcel.ActiveCell.Offset(0, 1).Activate()
        MyExcel.ActiveCell.Value = txtAddress.Text  'insert Address to excel
MyExcel.ActiveCell.Offset(0, 1).Activate()
        MyExcel.ActiveCell.Value = txtselectClass.Text       'choose class
MyExcel.ActiveCell.Offset(0, 1).Activate()
        MyExcel.ActiveCell.Value = txtselectPersatuan.Text  'insert PERSATUAN to excel
```

```
MyExcel.ActiveCell.Offset(0, 1).Activate()
        MyExcel.ActiveCell.Value = txtselectUniform.Text  'insert UNIFORM to excel

        SelectPersatuanReg = txtselectPersatuan.Text
MyExcel.Sheets(SelectPersatuanReg).activate()  'insert data to persatuan
MyExcel.Range("B3").Activate()
Do
If MyExcel.ActiveCell.Value >NothingOr MyExcel.ActiveCell.Text >NothingThen
MyExcel.ActiveCell.Offset(1, 0).Activate()     'turun bwh stu kotak
Else
Exit Do
EndIf
Loop
        MyExcel.ActiveCell.Value = txtName.Text     'insert name to excel

If txtselectUniform.SelectedItem = PKBM Then
            SelectUniformReg = "PKBM"
ElseIf txtselectUniform.SelectedItem = KRS Then
            SelectUniformReg = "KRS"
ElseIf txtselectUniform.SelectedItem = KPA Then
            SelectUniformReg = "KPA"
EndIf

MyExcel.Sheets(SelectUniformReg).activate()    'insert data to iniform
MyExcel.Range("B3").Activate()

Do
If MyExcel.ActiveCell.Value >NothingOr MyExcel.ActiveCell.Text >NothingThen
MyExcel.ActiveCell.Offset(1, 0).Activate()     'turun bwh stu kotak
Else
Exit Do
EndIf
Loop

        MyExcel.ActiveCell.Value = txtName.Text
MyExcel.ActiveCell.Offset(0, 1).Activate()

'close
MyExcel.Application.ActiveWorkbook.Save()
MyExcel.Workbooks.Close()
        MyExcel = Nothing

MsgBox("DONE Registration", MsgBoxStyle.OkOnly, "Message")
        txtName.Text = Nothing
        txtIC.Text = Nothing
        txtAddress.Text = Nothing
        txtselectClass.Text = Nothing
        txtselectPersatuan.Text = Nothing
        txtselectUniform.Text = Nothing
        txtStatus.Text = ""

Return complete = True
EndFunction

PrivateFunctionCheckValidID() AsString
        IDNoValid = Nothing

Dim MyExcel AsNew Excel.Application
MyExcel.Workbooks.Open(DataStudent)
MyExcel.Sheets("StudentID").activate()
MyExcel.Range("B2").Activate()
Do
If MyExcel.ActiveCell.Value >NothingOr MyExcel.ActiveCell.Text >NothingThen
MyExcel.ActiveCell.Offset(1, 0).Activate()     'turun bwh stu kotak
Else
MyExcel.ActiveCell.Offset(0, -1).Activate()
                IDNoValid = MyExcel.ActiveCell.Value
Exit Do
EndIf
Loop
MyExcel.Application.ActiveWorkbook.Save()
MyExcel.Workbooks.Close()
        MyExcel = Nothing
Return IDNoValid
EndFunction

PrivateFunctionErrortesttxtbox()
success = Nothing
If txtselectClass.Text = NothingThen
            success1 = 1
Else
            success1 = 0
EndIf
If txtselectPersatuan.Text = NothingThen
            success2 = 1
Else
            success2 = 0
EndIf
If txtselectUniform.Text = NothingThen
            success3 = 1
Else
            success3 = 0
EndIf
success = success1 + success2 + success3
Return success
EndFunction
''
''                      VIEW MARK
''
```

```
''  View All
''
PrivateSub Button3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnViewAll.Click
Dim openform AsNewForm2'define a new form
        txtStatus.Text = ""'clear txtstatus
openform.Show()
EndSub
''
'' View by fingerprint
''
PrivateSub Button10_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnViewFinger.Click
Dim getMarkPersatuan AsInteger = Nothing
Dim getMarkUniform AsInteger = Nothing
Dim UniformSheet AsString = Nothing
Dim percentPersatuan AsDouble = Nothing
Dim percentUniform AsInteger = Nothing
Dim a AsShort = 0

        TxtNameView.Text = ""
        TxtICView.Text = ""
        TxtAddressView.Text = ""
        TxtClassView.Text = ""
        TxtPersatuanView.Text = ""
        TxtUniformView.Text = ""
        txtMarkPersatuan.Text = ""
        txtMarkUniform.Text = ""
MachtingFinger()

If Getrecord = 0 Then
MsgBox("Invalid Please try again", vbExclamation, "Message")
Exit Sub
Else
ViewRecord()
EndIf
EndSub
''
'' view by IC
''
PrivateSub btnViewIC_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnViewIC.Click
Dim a AsShort
        TxtNameView.Text = ""
        TxtICView.Text = ""
        TxtAddressView.Text = ""
        TxtClassView.Text = ""
        TxtPersatuanView.Text = ""
        TxtUniformView.Text = ""
        txtMarkPersatuan.Text = ""
        txtMarkUniform.Text = ""

If txtSetViewIC.Text = ""Then
MsgBox("Please insert an IC No.", vbExclamation, "Message")
Exit Sub
Else
Dim MyExcel AsNew Excel.Application
MyExcel.Workbooks.Open(DataStudent)
MyExcel.Sheets("StudentID").activate()
MyExcel.Range("C2").Activate()

Do
If MyExcel.ActiveCell.Value = txtSetViewIC.Text Then
MyExcel.ActiveCell.Offset(0, -2).Activate()
                    Getrecord = MyExcel.ActiveCell.Value
                    error1 = 0
Exit Do
Else
MyExcel.ActiveCell.Offset(1, 0).Activate()     'turun bwh stu kotak
                    a = a + 1
If a = 40 Then
MsgBox("Please insert a corect IC No.", vbExclamation, "Message")
                        error1 = 1
Exit Do
EndIf
EndIf
Loop
MyExcel.Application.ActiveWorkbook.Save()
MyExcel.Workbooks.Close()
            MyExcel = Nothing
If error1 = 0 Then
ViewRecord()
EndIf
EndIf
        txtSetViewIC.Text = ""
EndSub
''
'' function view record
''
PrivateSubViewRecord()
Dim getMarkPersatuan AsInteger = Nothing
Dim getMarkUniform AsInteger = Nothing
Dim UniformSheet AsString = Nothing
Dim percentPersatuan AsDouble = Nothing
Dim percentUniform AsInteger = Nothing
Dim a AsShort = 0
Dim MyExcel AsNew Excel.Application
MyExcel.Workbooks.Open(DataStudent)
MyExcel.Sheets("StudentID").activate()
MyExcel.Range("A1").Activate()
MyExcel.ActiveCell.Offset(Getrecord, 1).Activate()
```

```
        TxtNameView.Text = MyExcel.ActiveCell.Value
MyExcel.ActiveCell.Offset(0, 1).Activate()
        TxtICView.Text = MyExcel.ActiveCell.Value
MyExcel.ActiveCell.Offset(0, 1).Activate()
        TxtAddressView.Text = MyExcel.ActiveCell.Value
MyExcel.ActiveCell.Offset(0, 1).Activate()
        TxtClassView.Text = MyExcel.ActiveCell.Value
MyExcel.ActiveCell.Offset(0, 1).Activate()
        TxtPersatuanView.Text = MyExcel.ActiveCell.Value
MyExcel.ActiveCell.Offset(0, 1).Activate()
        TxtUniformView.Text = MyExcel.ActiveCell.Value
MyExcel.ActiveCell.Offset(0, 1).Activate()

'' Get Persatuan Mark
MyExcel.Sheets(TxtPersatuanView.Text).activate()
MyExcel.Range("C2").Activate()
        a = 0
Do
If MyExcel.ActiveCell.Value = NothingThen
Exit Do
Else
MyExcel.ActiveCell.Offset(0, 1).Activate()
                a = a + 1
EndIf
Loop

If a = 0 Then
            txtMarkPersatuan.Text = "Null"
Else
MyExcel.Range("B3").Activate()
Do
If MyExcel.ActiveCell.Value = TxtNameView.Text Then
MyExcel.ActiveCell.Offset(0, 1).Activate()
Exit Do
Else
MyExcel.ActiveCell.Offset(1, 0).Activate()     'turun bwh stu kotak
EndIf
Loop

For i = 0 To a
getMarkPersatuan = getMarkPersatuan + MyExcel.ActiveCell.Value
MyExcel.ActiveCell.Offset(0, 1).Activate()
Next i
percentPersatuan = (getMarkPersatuan / a) * 10
            txtMarkPersatuan.Text = percentPersatuan  '' display mark
EndIf

''Get Uniform Mark
If TxtUniformView.Text = PKBM Then
            UniformSheet = "PKBM"
ElseIf TxtUniformView.Text = KRS Then
            UniformSheet = "KRS"
ElseIf TxtUniformView.Text = KPA Then
            UniformSheet = "KPA"
EndIf

MyExcel.Sheets(UniformSheet).activate()
MyExcel.Range("C2").Activate()
        a = 0
Do
If MyExcel.ActiveCell.Value = NothingThen
Exit Do
Else
MyExcel.ActiveCell.Offset(0, 1).Activate()
                a = a + 1
EndIf
Loop
If a = 0 Then
            txtMarkUniform.Text = "Null"
Else

MyExcel.Range("B3").Activate()
Do
If MyExcel.ActiveCell.Value = TxtNameView.Text Then
MyExcel.ActiveCell.Offset(0, 1).Activate()
Exit Do
Else
MyExcel.ActiveCell.Offset(1, 0).Activate()     'turun bwh stu kotak
EndIf
Loop

For i = 0 To a
getMarkUniform = getMarkUniform + MyExcel.ActiveCell.Value
MyExcel.ActiveCell.Offset(0, 1).Activate()
Next i
percentUniform = (getMarkUniform / a) * 10
            txtMarkUniform.Text = percentUniform
EndIf
MyExcel.Application.ActiveWorkbook.Save()
MyExcel.Workbooks.Close()
        MyExcel = Nothing
EndSub
''
''Edit Profile
''
PrivateSub Button8_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnEdit.Click
Dim SetPersatuan AsString
Dim SetUniform AsString
Dim NameOri AsString
```

```
Dim nameChange AsString

If Getrecord = 0 Then
MsgBox("Invalid! Please try again.", vbExclamation, "Message")
Exit Sub
Else
Dim MyExcel AsNew Excel.Application
MyExcel.Workbooks.Open(DataStudent)
MyExcel.Sheets("StudentID").activate()
MyExcel.Range("A1").Activate()

MyExcel.ActiveCell.Offset(Getrecord, 1).Activate()
            NameOri = MyExcel.ActiveCell.Value

If NameOri = TxtNameView.Text Then'checked name
MyExcel.ActiveCell.Offset(0, 1).Activate()
nameChange = 0
Else
                MyExcel.ActiveCell.Value = TxtNameView.Text     'change when it is difference
MyExcel.ActiveCell.Offset(0, 1).Activate()

nameChange = 1
EndIf


If MyExcel.ActiveCell.Value = TxtICView.Text Then
MyExcel.ActiveCell.Offset(0, 1).Activate()
Else
                MyExcel.ActiveCell.Value = TxtICView.Text
MyExcel.ActiveCell.Offset(0, 1).Activate()
EndIf

If MyExcel.ActiveCell.Value = TxtAddressView.Text Then
MyExcel.ActiveCell.Offset(0, 1).Activate()
Else
                MyExcel.ActiveCell.Value = TxtAddressView.Text
MyExcel.ActiveCell.Offset(0, 1).Activate()
EndIf

If MyExcel.ActiveCell.Value = TxtClassView.Text Then
MyExcel.ActiveCell.Offset(0, 1).Activate()
Else
                MyExcel.ActiveCell.Value = TxtClassView.Text
MyExcel.ActiveCell.Offset(0, 1).Activate()
EndIf

            SetPersatuan = MyExcel.ActiveCell.Value
MyExcel.ActiveCell.Offset(0, 1).Activate()
            SetUniform = MyExcel.ActiveCell.Value

If nameChange = 1 Then'if name changes
MyExcel.Sheets(SetPersatuan).activate()
MyExcel.Range("B3").Activate()
Do
If MyExcel.ActiveCell.Value = NameOri Then
                        MyExcel.ActiveCell.Value = TxtNameView.Text 'change old name in

Exit Do
Else
MyExcel.ActiveCell.Offset(1, 0).Activate()     'turun bwh stu kotak
EndIf
Loop

If SetUniform = PKBM Then
                    SetUniform = "PKBM"
ElseIf SetUniform = KRS Then
                    SetUniform = "KRS"
ElseIf SetUniform = KPA Then
                    SetUniform = "KPA"
EndIf

MyExcel.Sheets(SetUniform).activate()
MyExcel.Range("B3").Activate()
Do
If MyExcel.ActiveCell.Value = NameOri Then
                        MyExcel.ActiveCell.Value = TxtNameView.Text 'change old name in

Exit Do
Else
MyExcel.ActiveCell.Offset(1, 0).Activate()     'turun bwh stu kotak
EndIf
Loop
EndIf

MyExcel.Application.ActiveWorkbook.Save()
MyExcel.Workbooks.Close()
            MyExcel = Nothing
EndIf
MsgBox("Edit profile done.", MsgBoxStyle.OkOnly, "Message")

EndSub
''
''
''
PrivateSub btnMarkIC_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnMarkIC.Click
'compare
Dim a AsInteger = 0
        SetNo = Nothing
If SelectPersatuan.Checked = TrueThen
```

```
            SetNo = 4
ElseIf SelectUniform.Checked = TrueThen
            SetNo = 5
Else
MsgBox("Please select fields", MsgBoxStyle.Exclamation, "Message")
Exit Sub
EndIf

If txtMarkIC.Text = ""Then
MsgBox("Please insert an IC No.", vbExclamation, "Message")
Exit Sub
Else
Dim MyExcel AsNew Excel.Application
MyExcel.Workbooks.Open(DataStudent)
MyExcel.Sheets("StudentID").activate()
MyExcel.Range("C2").Activate()

Do
If MyExcel.ActiveCell.Value = txtMarkIC.Text Then
MyExcel.ActiveCell.Offset(0, -2).Activate()
                    Getrecord = MyExcel.ActiveCell.Value
                    error1 = 0
Exit Do
Else
MyExcel.ActiveCell.Offset(1, 0).Activate()     'turun bwh stu kotak
                    a = a + 1
If a = 40 Then
MsgBox("Please insert a corect IC No.", vbExclamation, "Message")
                       error1 = 1
Exit Do
EndIf
EndIf
Loop
MyExcel.Application.ActiveWorkbook.Save()
MyExcel.Workbooks.Close()
            MyExcel = Nothing
If error1 = 0 Then
markingData()
                txtMarkIC.Text = ""
EndIf

EndIf



EndSub

''  marking attendance
''
''
PrivateSub Button5_Click_1(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MarkButton.Click
'Dim IDName As String
'Dim SetPersatuanUniform As String
'Dim A As Integer

'compare
        SetNo = Nothing
If SelectPersatuan.Checked = TrueThen
            SetNo = 4
ElseIf SelectUniform.Checked = TrueThen
            SetNo = 5
Else
MsgBox("Please select fields", MsgBoxStyle.Exclamation, "Message")
Exit Sub
EndIf

MachtingFinger()

If Getrecord = 0 Then
MsgBox("Invalid! Please try again.", vbExclamation, "Message")
Exit Sub
Else
markingData()
EndIf
EndSub
''
''
PrivateSubmarkingData()
Dim IDName AsString
Dim SetPersatuanUniform AsString
Dim A AsInteger


Dim MyExcel AsNew Excel.Application'bukak excel
MyExcel.Workbooks.Open(DataStudent)

MyExcel.Sheets("StudentID").activate()
MyExcel.Range("A1").Activate()
MyExcel.ActiveCell.Offset(Getrecord, 1).Activate()

        IDName = MyExcel.ActiveCell.Text

MyExcel.ActiveCell.Offset(0, SetNo).Activate()
        SetPersatuanUniform = Nothing
If SelectPersatuan.Checked = TrueThen
            SetPersatuanUniform = MyExcel.ActiveCell.Text
ElseIf SelectUniform.Checked = TrueThen
If MyExcel.ActiveCell.Text = PKBM Then
                SetPersatuanUniform = "PKBM"
```

```
ElseIf MyExcel.ActiveCell.Text = KRS Then
                SetPersatuanUniform = "KRS"
ElseIf MyExcel.ActiveCell.Text = KPA Then
                SetPersatuanUniform = "KPA"
EndIf
EndIf

MyExcel.Sheets(SetPersatuanUniform).activate()
MyExcel.Range("B3").Activate()

        A = 1
Do
If MyExcel.ActiveCell.Value = IDName Then
Exit Do
Else
MyExcel.ActiveCell.Offset(1, 0).Activate()     'turun bwh stu kotak
                A = A + 1
EndIf
Loop
MyExcel.ActiveCell.Offset(0, 1).Activate()

'MarkingRecord
MyExcel.Range("C2").Activate()
Do
If MyExcel.ActiveCell.Value = Today Then
MyExcel.ActiveCell.Offset(A, 0).Activate()
Exit Do
ElseIf MyExcel.ActiveCell.Value = NothingThen
                MyExcel.ActiveCell.Value = Today
MyExcel.ActiveCell.Offset(A, 0).Activate()
Exit Do
Else
MyExcel.ActiveCell.Offset(0, 1).Activate()
EndIf
Loop
        MyExcel.ActiveCell.Value = "1"

'close
MyExcel.Application.ActiveWorkbook.Save()
MyExcel.Workbooks.Close()
        MyExcel = Nothing

If SetPersatuanUniform = "PKBM"Then
            SetPersatuanUniform = PKBM
ElseIf SetPersatuanUniform = "KRS"Then
            SetPersatuanUniform = KRS
ElseIf SetPersatuanUniform = "KPA"Then
            SetPersatuanUniform = KPA
Else
            SetPersatuanUniform = SetPersatuanUniform
EndIf

MsgBox("Done  Marking.."&  vbNewLine  &"Tq  "&  IDName  &  vbNewLine  &"Mark  on  "&  SetPersatuanUniform,  MsgBoxStyle.OkOnly,
"Message")
EndSub
''
''                      DELETE DATA
''
''  delete data by name
''
PrivateSub Button12_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnDeleteName.Click
Dim a AsShort
deleteName = txtDeleteName.Text
If txtDeleteName.Text = ""Then
MsgBox("please insert name to delete", vbExclamation, "Message")
Exit Sub
EndIf
Dim MyExcel AsNew Excel.Application
MyExcel.Workbooks.Open(DataStudent)         'open excel
MyExcel.Sheets("StudentID").activate()
MyExcel.Range("B2").Activate()

Do
If MyExcel.ActiveCell.Value = deleteName Then
MyExcel.ActiveCell.Offset(0, -1).Activate()
                Getrecord = MyExcel.ActiveCell.Value
                error1 = 0
Exit Do
Else
MyExcel.ActiveCell.Offset(1, 0).Activate()     'turun bwh stu kotak
                a = a + 1
If a = 40 Then
MsgBox("Please insert a corect Name", vbExclamation, "Message")
                    error1 = 1
Exit Do
EndIf
EndIf
Loop

MyExcel.Application.ActiveWorkbook.Save()
MyExcel.Workbooks.Close()
        MyExcel = Nothing
If error1 = 0 Then
DeleteARecord()
DeleteData()
EndIf

        txtDeleteName.Text = ""
EndSub
```

```
''
''  delete by IC
''
PrivateSub btnDeleteIC_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnDeleteIC.Click
Dim a AsShort
deleteIC = txtDeleteIC.Text
If txtDeleteIC.Text = ""Then
MsgBox("Please insert IC No. to delete", vbExclamation, "Message")
Exit Sub
EndIf
Dim MyExcel AsNew Excel.Application
MyExcel.Workbooks.Open(DataStudent)         'open excel
MyExcel.Sheets("StudentID").activate()
MyExcel.Range("C2").Activate()

Do
If MyExcel.ActiveCell.Value = deleteIC Then
MyExcel.ActiveCell.Offset(0, -1).Activate()
deleteName = MyExcel.ActiveCell.Value
MyExcel.ActiveCell.Offset(0, -1).Activate()
                Getrecord = MyExcel.ActiveCell.Value
                error1 = 0
Exit Do
Else
MyExcel.ActiveCell.Offset(1, 0).Activate()     'turun bwh stu kotak
                a = a + 1
If a = 40 Then
MsgBox("Please insert a corect Name", vbExclamation, "Message")
                    error1 = 1
Exit Do
EndIf
EndIf
Loop
MyExcel.Application.ActiveWorkbook.Save()
MyExcel.Workbooks.Close()
        MyExcel = Nothing
If error1 = 0 Then
DeleteARecord()
DeleteData()
EndIf
        txtDeleteIC.Text = ""
EndSub
''
''  delete via fingerprint
''
PrivateSub Button11_Click_1(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnDeleteFinger.Click
MachtingFinger()

If Getrecord <> 0 Then


Dim MyExcel AsNew Excel.Application
MyExcel.Workbooks.Open(DataStudent)         'open excel
MyExcel.Sheets("StudentID").activate()
MyExcel.Range("A1").Activate()
MyExcel.ActiveCell.Offset(Getrecord, 1).Activate()

deleteName = MyExcel.ActiveCell.Value
MyExcel.ActiveCell.Offset(0, 1).Activate()

MyExcel.Application.ActiveWorkbook.Save()
MyExcel.Workbooks.Close()
            MyExcel = Nothing

DeleteARecord()
DeleteData()
EndIf

EndSub
''
''function delete data
''
PrivateFunctionDeleteData()
Dim deletePersatuan AsString = Nothing
Dim deleteUniform AsString = Nothing
Dim MyExcel AsNew Excel.Application
        txtStatus.Text = "Deleting data.."
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
MyExcel.Workbooks.Open(DataStudent)         'open excel
MyExcel.Sheets("StudentID").activate()
MyExcel.Range("B2").Activate()

Do
If MyExcel.ActiveCell.Value = deleteName Then
                MyExcel.ActiveCell.Value = ""
Exit Do
Else
MyExcel.ActiveCell.Offset(1, 0).Activate()     'turun bwh stu kotak
EndIf
Loop
MyExcel.ActiveCell.Offset(0, 1).Activate()
        MyExcel.ActiveCell.Value = ""'delete IC
MyExcel.ActiveCell.Offset(0, 1).Activate()
        MyExcel.ActiveCell.Value = ""'delete address
MyExcel.ActiveCell.Offset(0, 1).Activate()
        MyExcel.ActiveCell.Value = ""'delete class
'get data
MyExcel.ActiveCell.Offset(0, 1).Activate()
```

```
deletePersatuan = MyExcel.ActiveCell.Value
        MyExcel.ActiveCell.Value = ""'delete persatuan
MyExcel.ActiveCell.Offset(0, 1).Activate()
deleteUniform = MyExcel.ActiveCell.Value
        MyExcel.ActiveCell.Value = ""'delte uniform

MyExcel.Sheets(deletePersatuan).activate()
MyExcel.Range("B3").Activate()
Do
If MyExcel.ActiveCell.Value = deleteName Then
Exit Do
Else
MyExcel.ActiveCell.Offset(1, 0).Activate()     'turun bwh stu kotak
EndIf
Loop

Do
If MyExcel.ActiveCell.Value = NothingThen
Exit Do
Else
                MyExcel.ActiveCell.Value = ""
MyExcel.ActiveCell.Offset(0, 1).Activate()
EndIf
Loop

If deleteUniform = PKBM Then
deleteUniform = "PKBM"
ElseIf deleteUniform = KRS Then
deleteUniform = "KRS"
ElseIf deleteUniform = KPA Then
deleteUniform = "KPA"
EndIf
MyExcel.Sheets(deleteUniform).activate()
MyExcel.Range("B3").Activate()

Do
If MyExcel.ActiveCell.Value = deleteName Then
Exit Do
Else
MyExcel.ActiveCell.Offset(1, 0).Activate()     'turun bwh stu kotak
EndIf
Loop

Do
If MyExcel.ActiveCell.Value = NothingThen
Exit Do
Else
                MyExcel.ActiveCell.Value = ""
MyExcel.ActiveCell.Offset(0, 1).Activate()
EndIf
Loop
MyExcel.Application.ActiveWorkbook.Save()
MyExcel.Workbooks.Close()
        MyExcel = Nothing

        txtStatus.Text = "deleted "& deleteName

Return 0
EndFunction

PrivateSub Timer1_Tick(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Timer1.Tick
        clocktime.Text = TimeOfDay
EndSub
''
'' Open Port Button
''
PrivateSub cmdPortOpen_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles cmdPortOpen.Click
If cboPort.Text = "Choose COM Port..."Then
MsgBox("Please select COM Port", vbExclamation, "Message")  'if port not chosen prompt user msgbox
Else
            SerialPort.PortName = cboPort.Text                  'get and set portname
            SerialPort.BaudRate = cboBaudRate.Text              'get and set baudrate
            SerialPort.Encoding = System.Text.Encoding.ASCII    'set encoding

Try
SerialPort.Open()                       'open serialport
                StatusPanel.Text = "Connected"'display status as connected
portisopen = True'indicator for opened port
Catch ex AsException
MsgBox(cboPort.Text &" already in used by other application.", vbExclamation, "Message")    'if port is not ready to open
prompt msgbox
Exit Sub
EndTry
'enable or disable control
            cmdPortClose.Enabled = True
            btnViewFinger.Enabled = True
            RegisterButton.Enabled = True
            btnDeleteFinger.Enabled = True
            MarkButton.Enabled = True
            cmdPortOpen.Enabled = False
            cboPort.Enabled = False
            cboBaudRate.Enabled = False
EndIf

'Serial port opened and perform initialization to fingerprint reader
If portisopen = TrueThen

'Setting Timeout'
Dim a AsInteger
```

```
DimCommandSetTimeOut() AsByte = {&HF5, &H2E, &H0, &H7, &H0, &H0, &H29, &HF5}
SerialPort.DiscardInBuffer()                    'clear buffer
For i = 0 To 7
SerialPort.Write(CommandSetTimeOut, i, 1)   'send command through serialport
Next i
            a = 0
While a = 0
                a = checkbuffer(8)                          'check if 8 bytes in buffer
EndWhile
Dim number AsByte()
Dim receivedData AsString = ""
receivedData += SerialPort.ReadExisting()       'read serialport buffer
number = StrToByteArray(receivedData)           'string to byte array conversion

'Setting No Repetition'
DimCommandSetNoRepeat() AsByte = {&HF5, &H2D, &H0, &H1, &H0, &H0, &H2C, &HF5}
SerialPort.DiscardInBuffer()                    'clear buffer
For i = 0 To 7
SerialPort.Write(CommandSetNoRepeat, i, 1)  'send command through serialport
Next i
            a = 0
While a = 0
                a = checkbuffer(8)                          'check if 8 bytes in buffer
EndWhile
receivedData += SerialPort.ReadExisting()       'read serialport buffer
number = StrToByteArray(receivedData)           'string to byte array conversion
EndIf
EndSub
'check buffer'
PublicFunctioncheckbuffer(ByVal lenght AsByte) AsByte

Dim flag AsBoolean
If SerialPort.BytesToRead < lenght Then'buffer not reaching desired capacity
flag = 0
Return flag                             'return 0
Else'buffer has enough data
flag = 1
Return flag                             'return 1
EndIf
EndFunction
PrivateSub cboPort_DropDown(ByVal sender AsObject, ByVal e As System.EventArgs) Handles cboPort.DropDown

For i AsInteger = 0 ToMy.Computer.Ports.SerialPortNames.Count - 1     'check serialport avaibility and
cboPort.Items.Clear()                                               'add the port found as an item
cboPort.Items.Add(My.Computer.Ports.SerialPortNames(i))
Next

EndSub
PublicSharedFunctionStrToByteArray(ByVal str AsString) AsByte()
Dim encoding AsNewSystem.Text.ASCIIEncoding()
Returnencoding.GetBytes(str)
EndFunction
''
'' Close Port Button
''
PrivateSub cmdPortClose_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles cmdPortClose.Click

SerialPort.Close()                  'close serialport
        StatusPanel.Text = "Disconnected"'display disconnected

'enable/disable/reset control
        cboPort.Enabled = True
        cboBaudRate.Enabled = True
        cmdPortOpen.Enabled = True
        RegisterButton.Enabled = False
        btnDeleteFinger.Enabled = False
        MarkButton.Enabled = False
        btnViewFinger.Enabled = False
        cmdPortClose.Enabled = False
portisopen = False
        cboPort.Text = "Choose COM Port..."
        cboBaudRate.Text = "19200"
EndSub
''
'' Button Login
''
PrivateSub btnLogin_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnLogin.Click
Dim password AsString
Dim username AsString
Dim passwordcmpre AsString
Dim usernamecmpre AsString

If txtPassword.Text <>""And txtUserName.Text <>""Then'check both fields are not blank

username = txtUserName.Text         'typed username assign to variable "username"
password = txtPassword.Text         'typed password assign to variable "password"
username = username.ToUpper()       'both convert all characters into uppercase (not case sensitive)
password = password.ToUpper()

Dim MyExcel AsNew Excel.Application'bukak excel
MyExcel.Workbooks.Open("C:\Student Attendance System Setup\Database\admin.xls") 'C:\Student Attendace Database\

MyExcel.Sheets("Sheet1").activate()
MyExcel.Range("B1").Activate()
usernamecmpre = MyExcel.ActiveCell.Value
MyExcel.ActiveCell.Offset(1, 0).Activate()    'turun bwh stu kotak
passwordcmpre = MyExcel.ActiveCell.Value

If usernamecmpre = username And passwordcmpre = password Then'password and username matched
```

```
                txtStatus.Text = "Welcome Admin"
txtStatus.Refresh()
                RegTab.Enabled = True
                DeleteDataTab.Enabled = True
                btnResetSystem.Enabled = True
                btnEdit.Enabled = True
                btnLogout.Enabled = True
                btnLogin.Enabled = False
Else
                txtStatus.Text = "Username or/and password incorrect"
txtStatus.Refresh()
EndIf
'close Excel
MyExcel.Application.ActiveWorkbook.Save()
MyExcel.Workbooks.Close()
            MyExcel = Nothing
            txtUserName.Text = Nothing'typed username assign to variable "username"
            txtPassword.Text = Nothing
Else
MsgBox("Please Fill in all require fields.", MsgBoxStyle.Exclamation, "Message") 'prompt user to fill in all require fields
EndIf
EndSub
''
''Button Logout
''
PrivateSub Button11_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnLogout.Click
        RegTab.Enabled = False
        DeleteDataTab.Enabled = False
        btnResetSystem.Enabled = False
        btnLogout.Enabled = False
        btnEdit.Enabled = False
        btnLogin.Enabled = True

        txtStatus.Text = "Thank you"
txtStatus.Refresh()
EndSub
''
'' check status
''
PublicFunctioncheckstatus() AsByte

Dim number AsByte()                         'define number as byte array
Dim receivedData AsString = ""'a string to temporary store everything from buffer
receivedData = ""'clear string
receivedData += SerialPort.ReadExisting()   'read buffer
number = StrToByteArray(receivedData)       'string to byte array conversion

IfLen(BitConverter.ToString(number)) = 23 Then'lenght af the string = 23 eg. "F5-01-00-01-03-00-03-F5"
SelectCasenumber(1)    'select byte 2            'if we count character by character there are 23 characters which means we
didn't receive more or less bytes
'Matching 1:N
Case 12                     ' 0X0C is decimal 12 reply of 1:N
Ifnumber(4) = 3 Then' successful    'select byte 5
Ifnumber(3) = 99 Then
Return 99   'admin
Else
Returnnumber(3)
EndIf

ElseIfnumber(4) = 8 Then'failed     'select byte 5
                        txtStatus.Text = "Time Out"
Return 0
ElseIfnumber(4) = 5 Then
                        txtStatus.Text = "User Not Found"
Return 0
EndIf

'Get the total fingerprint stored
Case 9
                    txtStatus.Text = "No. of fingerprints collected = "&number(3)
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                    error1 = 0
Return error1
'Add fingerprint operation (1)'
Case 1
SelectCasenumber(4)   'select byte 5
Case0  'successful'
                            txtStatus.Text = "Again-Verification 1/2"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                            error1 = 0
Return error1
Case 1
                            txtStatus.Text = "Error!"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                            error1 = 1
Return error1
Case 4
                            txtStatus.Text = "Full!"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                            error1 = 1
Return error1
Case 7
                            txtStatus.Text = "Error: User exists"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
```

```
                                error1 = 1
Return error1
Case 8
                                txtStatus.Text = "Time out"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                                error1 = 1
Return error1
Case Else
                                txtStatus.Text = "Unknown Error!"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                                error1 = 1
Return error1
EndSelect
'Add fingerprint operation (2)'
Case 2
SelectCasenumber(4)   'select byte 5
Case0  'successful
                                txtStatus.Text = "Again-Verification 2/2"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                                error1 = 0
Return error1
Case 1
                                txtStatus.Text = "Error!"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                                error1 = 1
Return error1
Case 4
                                txtStatus.Text = "Full!"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                                error1 = 1
Return error1
Case 7
                                txtStatus.Text = "Error: User exists"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                                error1 = 1
Return error1
Case 8
                                txtStatus.Text = "Time out"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                                error1 = 1
Return error1
Case Else
                                txtStatus.Text = "Unknown Error!"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                                error1 = 1
Return error1
EndSelect
'Add fingerprint operation (3)'
Case 3
SelectCasenumber(4)   'select byte 5
Case0  'successful
                                txtStatus.Text = "Adding Succesfull"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                                error1 = 0
Return error1

Case 1
                                txtStatus.Text = "Error!"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                                error1 = 1
Return error1
Case 4
                                txtStatus.Text = "Full!"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                                error1 = 1
Return error1
Case 7
                                txtStatus.Text = "Error: User exists"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                                error1 = 1
Return error1
Case 8
                                txtStatus.Text = "Time out"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                                error1 = 1
Return error1
Case Else
                                txtStatus.Text = "Unknown Error!"
txtStatus.Refresh()
System.Threading.Thread.Sleep(1000)
                                error1 = 1
Return error1
EndSelect
'Delete a fingerprint
Case 4
SelectCasenumber(4)   'select byte 5
```

```
Case0  'sucessful
'clear_record()
                            txtStatus.Text = " Record has been cleared"
txtStatus.Refresh()
'save_record()
                            error1 = 0
Return error1
Case 1
                            txtStatus.Text = "Error!"
txtStatus.Refresh()
                            error1 = 1
Return error1
EndSelect
EndSelect
Else
            txtStatus.Text = "something wrong"'this shouldn't happen
Return error1 = 1
EndIf
Return error1
EndFunction
''
'' delete user
''
PrivateSubDeleteARecord()

Dim a AsByte
Dim CHK AsInteger
        CHK = Getrecord Xor&H4
'record 1 chosen
Dimdelete() AsByte = {&HF5, &H4, &H0, Getrecord, &H0, &H0, CHK, &HF5}
SerialPort.DiscardInBuffer()            'clear buffer
For i = 0 To 7
SerialPort.Write(delete, i, 1)     'send delete command
Next i
        a = 0
While a = 0
            a = checkbuffer(8)                  'check buffer
EndWhile
checkstatus()

EndSub
''
'' button reset system
''
PrivateSub btnResetSystem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnResetSystem.Click
Dim result AsInteger = MsgBox("Are you sure to reset the system?", MsgBoxStyle.YesNo, "Message")

If result = DialogResult.No Then
Exit Sub
ElseIf result = DialogResult.Yes Then
            txtStatus.Text = "Resetting System.."
txtStatus.Refresh()
DeleteAllFingerprint()
DeleteAllData()
MessageBox.Show("System has been reset")
            txtStatus.Text = ""
EndIf


EndSub
''
''  function delete all data
''
PrivateSubDeleteAllData()
Dim MyExcel AsNew Excel.Application
MyExcel.Workbooks.Open(DataStudent)
MyExcel.Sheets("StudentID").activate()
MyExcel.Range("B2").Activate()

For a = 0 To 30
For i = 0 To 5
                MyExcel.ActiveCell.Value = ""
MyExcel.ActiveCell.Offset(0, 1).Activate()
Next i
MyExcel.ActiveCell.Offset(1, -6).Activate()
Next a

DimSheet1() AsString = {"BAHASA MELAYU", "BAHASA INGGERIS", "MATEMATIK & SAINS", _
"REKACIPTA", "PKBM", "KRS", "KPA"}

For i = 0 To 6
MyExcel.Sheets(Sheet1(i)).activate()
MyExcel.Range("C2").Activate()
For c = 0 To 15
                MyExcel.ActiveCell.Value = ""
MyExcel.ActiveCell.Offset(0, 1).Activate()
Next c
MyExcel.ActiveCell.Offset(1, -17).Activate()
For a = 0 To 20
For c = 0 To 15
                    MyExcel.ActiveCell.Value = ""
MyExcel.ActiveCell.Offset(0, 1).Activate()
Next c
MyExcel.ActiveCell.Offset(1, -16).Activate()
Next a
Next i
MyExcel.Application.ActiveWorkbook.Save()
MyExcel.Workbooks.Close()
        MyExcel = Nothing
```

```
EndSub
''
''              FINGERPRINT COMMAND
''
'' add new record
''
PrivateSubAddNewRecord()
Dim i AsShort
Dim a AsShort
Dim error1 AsShort
Dim CHK1 AsInteger
Dim CHK2 AsInteger
Dim CHK3 AsInteger
        txtStatus.Text = "Place Your Thumb On the Sensing Area"
txtStatus.Refresh()
        CHK1 = IDNoValid Xor&H1 Xor&H3
        CHK2 = IDNoValid Xor&H2 Xor&H3
        CHK3 = IDNoValid Xor&H3 Xor&H3

Dim user_1() AsByte = {&HF5, &H1, &H0, IDNoValid, &H3, &H0, CHK1, &HF5}
Dim user_2() AsByte = {&HF5, &H2, &H0, IDNoValid, &H3, &H0, CHK2, &HF5}
Dim user_3() AsByte = {&HF5, &H3, &H0, IDNoValid, &H3, &H0, CHK3, &HF5}
SerialPort.DiscardInBuffer()                'clear buffer
For i = 0 To 7
SerialPort.Write(user_1, i, 1)         'send 1st command through serialbport
Next i
        a = 0
While a = 0
            a = checkbuffer(8)                      'check buffer
EndWhile
        error1 = checkstatus()                      'check reply
If error1 = 0 Then'proceed if no error returns
SerialPort.DiscardInBuffer()            'clear buffer
For i = 0 To 7
SerialPort.Write(user_2, i, 1)     'send 2nd command through serialport
Next i
            a = 0
While a = 0
                a = checkbuffer(8)                  'check buffer
EndWhile
            error1 = checkstatus()                  'check reply
If error1 = 0 Then'proceed if no error returns
SerialPort.DiscardInBuffer()        'clear buffer
For i = 0 To 7
SerialPort.Write(user_3, i, 1) 'send 3rd command through serialport
Next i
                a = 0
While a = 0
                    a = checkbuffer(8)              'check buffer
EndWhile
                error1 = checkstatus()              'check reply
EndIf
EndIf

EndSub
''
'' Matching fingerprint
''
PrivateSubMachtingFinger()
        Getrecord = Nothing
        txtStatus.Text = "Place Your Thumb On the Sensing Area"
txtStatus.Refresh()

Dimbuff() AsByte = {&HF5, &HC, &H0, &H0, &H0, &H0, &HC, &HF5}

Dim i AsShort
Dim a AsShort
SerialPort.DiscardInBuffer()        'clear buffer
For i = 0 To 7
SerialPort.Write(buff, i, 1)    'send command through serialport
Next i

While a = 0
            a = checkbuffer(8)              'check buffer
EndWhile
        Getrecord = checkstatus()           'check reply

EndSub
''
''  reset fingerprint
''
PrivateSubDeleteAllFingerprint()

System.Threading.Thread.Sleep(1000)  ''delay
txtStatus.Refresh()
'Global delete to reset whole system
Dim a AsByte
DimdeleteAll() AsByte = {&HF5, &H5, &H0, &H0, &H0, &H0, &H5, &HF5}    'command
SerialPort.DiscardInBuffer()            'clear buffer
For i = 0 To 7
SerialPort.Write(deleteAll, i, 1)   'send command through serialport
Next i
        a = 0
While a = 0
            a = checkbuffer(8)                  'check buffer
EndWhile

        error1 = checkstatus()                  'check reply
txtStatus.Refresh()
```

```
EndSub

'' keypressing not allowed
PrivateSub txtselectClass_KeyDown(ByVal sender AsObject, ByVal e As System.Windows.Forms.KeyEventArgs) Handles
txtselectClass.KeyDown
        e.SuppressKeyPress = True
EndSub
PrivateSub txtselectPersatuan_KeyDown(ByVal sender AsObject, ByVal e As System.Windows.Forms.KeyEventArgs) Handles
txtselectPersatuan.KeyDown
        e.SuppressKeyPress = True
EndSub
PrivateSub txtselectUniform_KeyDown(ByVal sender AsObject, ByVal e As System.Windows.Forms.KeyEventArgs) Handles
txtselectUniform.KeyDown
        e.SuppressKeyPress = True
EndSub
PrivateSub TxtNameView_KeyDown(ByVal sender AsObject, ByVal e As System.Windows.Forms.KeyEventArgs) Handles
TxtNameView.KeyDown
If btnEdit.Enabled = FalseThen
            e.SuppressKeyPress = True
EndIf
EndSub
PrivateSub TxtICView_KeyDown(ByVal sender AsObject, ByVal e As System.Windows.Forms.KeyEventArgs) Handles TxtICView.KeyDown
If btnEdit.Enabled = FalseThen
            e.SuppressKeyPress = True
Else
If e.KeyCode = Keys.D0 Or e.KeyCode = Keys.D1 Or e.KeyCode = Keys.D2 Or e.KeyCode = Keys.D3 Or _
            e.KeyCode = Keys.D4 Or e.KeyCode = Keys.D5 Or e.KeyCode = Keys.D6 Or e.KeyCode = Keys.D7 Or _
            e.KeyCode = Keys.D8 Or e.KeyCode = Keys.D9 Then
Else
                e.SuppressKeyPress = True
EndIf
EndIf
EndSub
PrivateSub TxtAddressView_KeyDown(ByVal sender AsObject, ByVal e As System.Windows.Forms.KeyEventArgs) Handles
TxtAddressView.KeyDown
If btnEdit.Enabled = FalseThen
            e.SuppressKeyPress = True
EndIf
EndSub
PrivateSub TxtClassView_KeyDown(ByVal sender AsObject, ByVal e As System.Windows.Forms.KeyEventArgs) Handles
TxtClassView.KeyDown
        e.SuppressKeyPress = True
EndSub
PrivateSub txtMarkPersatuan_KeyDown(ByVal sender AsObject, ByVal e As System.Windows.Forms.KeyEventArgs) Handles
txtMarkPersatuan.KeyDown
        e.SuppressKeyPress = True
EndSub
PrivateSub txtMarkUniform_KeyDown(ByVal sender AsObject, ByVal e As System.Windows.Forms.KeyEventArgs) Handles
txtMarkUniform.KeyDown
        e.SuppressKeyPress = True
EndSub
PrivateSub txtDeleteIC_KeyDown(ByVal sender AsObject, ByVal e As System.Windows.Forms.KeyEventArgs) Handles
txtDeleteIC.KeyDown
If e.KeyCode = Keys.D0 Or e.KeyCode = Keys.D1 Or e.KeyCode = Keys.D2 Or e.KeyCode = Keys.D3 Or _
            e.KeyCode = Keys.D4 Or e.KeyCode = Keys.D5 Or e.KeyCode = Keys.D6 Or e.KeyCode = Keys.D7 Or _
            e.KeyCode = Keys.D8 Or e.KeyCode = Keys.D9 Then
Else
            e.SuppressKeyPress = True
EndIf
EndSub
PrivateSub TxtUniformView_KeyDown(ByVal sender AsObject, ByVal e As System.Windows.Forms.KeyEventArgs) Handles
TxtUniformView.KeyDown
        e.SuppressKeyPress = True
EndSub
PrivateSub TxtPersatuanView_KeyDown(ByVal sender AsObject, ByVal e As System.Windows.Forms.KeyEventArgs) Handles
TxtPersatuanView.KeyDown
        e.SuppressKeyPress = True
EndSub
PrivateSub txtSetViewIC_KeyDown(ByVal sender AsObject, ByVal e As System.Windows.Forms.KeyEventArgs) Handles
txtSetViewIC.KeyDown
If e.KeyCode = Keys.D0 Or e.KeyCode = Keys.D1 Or e.KeyCode = Keys.D2 Or e.KeyCode = Keys.D3 Or _
            e.KeyCode = Keys.D4 Or e.KeyCode = Keys.D5 Or e.KeyCode = Keys.D6 Or e.KeyCode = Keys.D7 Or _
            e.KeyCode = Keys.D8 Or e.KeyCode = Keys.D9 Then
Else
            e.SuppressKeyPress = True
EndIf
EndSub
PrivateSub txtIC_KeyDown(ByVal sender AsObject, ByVal e As System.Windows.Forms.KeyEventArgs) Handles txtIC.KeyDown
If e.KeyCode = Keys.D0 Or e.KeyCode = Keys.D1 Or e.KeyCode = Keys.D2 Or e.KeyCode = Keys.D3 Or _
            e.KeyCode = Keys.D4 Or e.KeyCode = Keys.D5 Or e.KeyCode = Keys.D6 Or e.KeyCode = Keys.D7 Or _
            e.KeyCode = Keys.D8 Or e.KeyCode = Keys.D9 Then
Else
            e.SuppressKeyPress = True
EndIf
EndSub
PrivateSub txtMarkIC_KeyDown(ByVal sender AsObject, ByVal e As System.Windows.Forms.KeyEventArgs) Handles txtMarkIC.KeyDown
If e.KeyCode = Keys.D0 Or e.KeyCode = Keys.D1 Or e.KeyCode = Keys.D2 Or e.KeyCode = Keys.D3 Or _
           e.KeyCode = Keys.D4 Or e.KeyCode = Keys.D5 Or e.KeyCode = Keys.D6 Or e.KeyCode = Keys.D7 Or _
           e.KeyCode = Keys.D8 Or e.KeyCode = Keys.D9 Then
Else
            e.SuppressKeyPress = True
EndIf
EndSub
EndClass
```