ACTIVE INFRARED MOTION DETECTOR FOR HOUSE SECURITY SYSTEM

MIOR MOHAMMAD HAFIIZH BIN ABD. RANI

UNIVERSITI MALAYSIA PAHANG

MIOR MOHAMMAD HAFIIZH BIN ABD. RANI	BACHELOR OF ELECTRICAL ENGINEERING (POWER SYSTEMS)	2007	UMP

UNIVERSITI MALAYSIA PAHANG

BORANG PENGESAHAN STATUS TESIS♦

JUDUL: **ACTIVE INFRARED MOTION DETECTOR FOR HOUSE SECURITY SYSTEM**

SESI PENGAJIAN: **2007/2008**

Saya **MIOR MOHAMMAD HAFIIZH BIN ABD. RANI (840308-01-6183)**
(HURUF BESAR)

mengaku membenarkan tesis (Sarjana Muda/~~Sarjana~~ /~~Doktor Falsafah~~)* ini disimpan di Perpustakaan dengan syarat-syarat kegunaan seperti berikut:

1. Tesis adalah hakmilik Universiti Malaysia Pahang (UMP).
2. Perpustakaan dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. **Sila tandakan (√)

☐	**SULIT**	(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)
☐	**TERHAD**	(Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)
√	**TIDAK TERHAD**	

Disahkan oleh:

_____ (TANDATANGAN PENULIS)

_____ (TANDATANGAN PENYELIA)

Alamat Tetap:

NO. 1-4 QRTS. KASTAM DIRAJA MALAYSIA, JALAN BAKRI, 84000 MUAR, JOHOR.

MOHD REDZUAN BIN AHMAD
(Nama Penyelia)

Tarikh: **28 NOVEMBER 2007**

Tarikh: : **28 NOVEMBER 2007**

CATATAN:
- * Potong yang tidak berkenaan.
- ** Jika tesis ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali tempoh tesis ini perlu dikelaskan sebagai atau TERHAD.
- ♦ Tesis dimaksudkan sebagai tesis bagi Ijazah doktor Falsafah dan Sarjana secara Penyelidikan, atau disertasi bagi pengajian secara kerja kursus dan penyelidikan, atau Laporan Projek Sarjana Muda (PSM).

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

Signature : _____

Name : MOHD REDZUAN BIN AHMAD

Date : 28 NOVEMBER 2007

ACTIVE INFRARED MOTION DETECTOR FOR HOUSE SECURITY SYSTEM

MIOR MOHAMMAD HAFIIZH BIN ABD. RANI

This thesis is submitted as partial fulfillment of the requirements for the award of the Bachelor of Electrical Engineering (Power Systems)

Faculty of Electrical & Electronics Engineering
Universiti Malaysia Pahang

NOVEMBER, 2007

"All the trademark and copyrights use herein are property of their respective owner. References of information from other sources are quoted accordingly; otherwise the information presented in this report is solely work of the author."

Signature : _____

Author : MIOR MOHAMMAD HAFIIZH BIN ABD. RANI

Date : 28 NOVEMBER 2007

To my beloved mother, father, sisters, and brother

ACKNOWLEDGEMENT

In the name of Allah, The Most Loving and The Most Compassionate

I would like to take this opportunity to extend my deepest gratitude to the following persons who helped me a lot in this project, which enabled me to complete my research project in time as a partial of the requirement for the Bachelor of Electrical Engineering (Power Systems).

First and foremost, a special thank to my supervisor En. Mohd Redzuan bin Ahmad, who helped me a lot for the project research, for all the support, continuous patience, and supervision given throughout the project.

I would like to give my appreciation to Mr. Muhamad Khairul Azmi Bin Ahmad Zake and Mr. Shuhairie for their valuable time spent on guiding me during the completion of this project. Their guidance and helps is indeed very much appreciated.

To important lecturer, Mr. Ahmad Johari Bin Mohamad who has taken effort in advising and supporting me, you deserve a sincere Thank You from me. Apart from that, I would like to thank Mr. Addie Irdawan Hashim for his guidance and coordination in this final year project.

Thank you to my classmates with their encouragement and help. Last but not least, my special thanks, to my family members for their continuous support and advice from the early stage of my studies.

ABSTRACT

Nowadays, house security system becomes the best solution to overcome house intrusion problem when user is not in house. As we know, there are many types of house security system which is too expensive and difficult to use. For that reason, an effective house security system at low cost is built where user can also program the security system by their own. This project is focusing on developing a house security system with an active infrared motion detector which is controlled by microcontroller Intel 8051. The overall project is divided into two parts. The first part is concern on the hardware development where all electronics component are connected via the circuit design using wrapping technique. An active infrared, the magnetic sensor, and keypad are the input components while buzzer, indicator, and LCD display are the output components where it's all controlled by controller circuit. The second part is base on software programming to operate the hardware structure. Program for security system based on microcontroller Intel 8051 assembly language is assemble using ASM51 assembler to get the binary file thus, to load into external memory of the hardware structure via serial communication. The process of downloading and executing the program is done using HyperTerminal's communication software to the microcontroller serial port. In order to achieve the best house security system, more detectors or sensors can be connected to the microcontroller output port where it can be reprogram by user using their personal computer at home. As the result, the infrared motion detector is capable to detect motion while the microcontroller is capable to control the whole operation of the security system.

ABSTRAK

Dewasa kini sistem penggera keselamatan rumah merupakan salah satu cara penyelesaian kepada masalah pencerohon rumah yang boleh berlaku sewaktu ketiadaan tuan rumah. Sepertimana yang diketahui, terdapat pelbagai sistem penggera keselamatan rumah di pasaran yang agak mahal dan tidak mampu dimiliki oleh pengguna. Oleh kerana itu, sistem penggera keselamatan yang efektif dan murah di mana pengguna boleh memprogram sendiri sistem keselamatan rumah mereka. Projek ini tertumpu kepada sebuah sistem penggera keselamatan rumah yang dapat mengesan pergerakan melalui pengesan sinar merah yang dikawal oleh pengawalan mikro cip Intel 8051. Secara keseluruhannya, projek ini dibahagikan kepada dua bahagian. Bahagian pertama tertumpu kepada pembangunan litar elektronik di mana kesemua komponen-komponen elektronik disambung berdasarkan gambar rajah litar yang dilukis. Bahagian kedua pula tertumpu kepada program untuk mengoperasikan litar elektronik yang telah dibina. Program sistem keselamatan rumah yang telah disemak dan ditukar kepada bahasa kod nombor akan dipindahkan ke dalam memori data struktur elektronik yang telah dibina menggunakan komunikasi sesiri komputer. Proses memasukkan program data ini ke dalam mikro cip pengawal litar dilakukan menerusi sistem komunikasi yang dikenali sebagai HyperTerminal yang sedia ada dalam komputer. Dalam usaha mendapatkan keputusan sistem keselamatan yang lebih baik, pelbagai alat pengesan lain boleh dimuatkan pada kelauaran mikro cip pengawal Intel 8051 dan boleh diprogram semula oleh pengguna melalui komputer di rumah. Secara kesimpulan projek ini, sinar merah yang digunakan mampu mengesan pergerakan manakala mikro cip pengawal pula dapat mengawal kesemua operasi sistem keselamatan yang dibangunkan.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

PIR	-	Passive Infrared
IR	-	Infrared
DPTR	-	Data Pointer
SFRs	-	Special Function Registers
LCD	-	Liquid-Crystal Displays
ROM	-	Read Only Memory
EPROM	-	Erasable Read Only Memory
RAM	-	Random Access Memory
GND	-	Ground
OSC	-	Oscillator
LED	-	Light Emitting Diode
MSB	-	Most Significant Bit
LSB	-	Least Significant Bit
PC	-	Personal Computer

LIST OF SYMBOLS

μ	-	Micro
K	-	Kilo
kHz	-	Kilo Hertz
T_x	-	Transmit
R_x	-	Receive
V	-	Volts
dc	-	Direct Current
cm	-	Centimeter
L	-	Load
E	-	Execute

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Background

Motion detection is the action of sensing physical movement in a given area. Motion can be detected by measuring change in speed or vector of an object in the field of view. This can be achieved either by mechanical devices that physically interact with the field or by electronic devices that quantifies and measures changes in the given environment.

There are two device of motion detection which is the mechanical device, and the other one is electronic device. In the mechanical device, a tripwire is a simple form of motion detection. If a moving objects steps into the tripwire's field of view then a simple sound device like bells may alert the user. Mechanical motion detection devices can be simple to implement, but at the same time, it can be defeated easily by interrupting the devices' mechanics like "cutting the wire”.

While in electronic device, the electronic motion sensing such as motion detectors, can prevent such mechanical intervention. The principal methods by which motion can be electronically identified are optical detection and acoustical detection. Infrared light or laser technology may be used for optical detection.

Motion detection devices, such as motion detectors, have sensors that detect movement and send a signal to a sound device that produces an alarm or switch on an image recording device. There are motions detector which employ cameras connected to a computer which stores and manages captured images to be viewed over a computer network.

The applications for such detection are detection of unauthorized entry, detection of cessation of occupancy of an area to extinguish lighting, and detection of a moving object which triggers a camera to record subsequent events. The motion detector is thus a basic idea of electronic security systems.

1.2 Project objective

The objective of this project is to build a house security system using microcontroller Intel 8051 based on active infrared motion detection.

1.3 Project scope

This project concentrates on a development of an active infrared motion detector for house security system. To develop the whole project, it consists of three methods which are the concept of security system, the electrical structure, and the software programming.

The concept of security system is on the detection of movement using active sensor to trigger alarm controlled by the microcontroller Intel 8051. The electrical structure consist of two systems which are the active infrared circuit that used to detect or sense motion and the microcontroller Intel 8051 circuit, used to control the whole operation of the security system. While the software programming is base on

the microcontroller Intel 8051 instruction sets. It contains a program designed for a security system as an interaction to operate the electrical structure.

1.4 Literature review

Motion detectors are mainly used in security systems [4]. It is typically positioned near exterior doorways or windows of a building to monitor the area around it. Since motion detectors are so flexible and have so many uses, it offers feelings of protection and security for the average homeowner as well as commercial organizations [4].

An electronic motion detector is a device used to detect any physical movement in a given area and transforms motion into an electric signal. It consist of sensor that electrically connected to other devices such as security system, lighting, audio alarms, and other applications. Motion sensors are used in a wide variety of applications and as a result, many different types of motion sensors are available including the infrared sensor.

Infrared sensors are widely known in the arts of intrusion detection and in fire or smoke detection. It is a device that often used in automatic light switches and security systems to turn on a light or to activate some other form of alarm or warning indicator when a person enters a monitored area [4]. The infrared sensors have basically two forms: active and passive. [4]

An active infrared detector includes a radiation source and an infrared sensor which is sensitive to interruptions in the radiation sensed from the source. [4] These detectors are used as intrusion detectors by providing a path of radiation from the source to the sensor in a place where the path is likely to be interrupted by an intruder.

The proposed active infrared method of motion detection has the advantage of fast speed response of a relatively large sensor. This advantage permits simpler optical system design, especially for wide fields of view. Besides, it is insensitivity to mechanical and acoustic noise, which presents substantial problems in the passive infrared (PIR) sensors. Low production cost is another advantage of these active infrared detectors. [4]

Passive infrared motion detection detects heat energy radiated or emitted by an object, such as a body of a person, moving across a field of view of a heat sensor of the motion detection system. It is generally use an optical collection system and multiple sensing elements of alternating polarity to create a detection pattern in the volume of interest.

PIR detectors employ a group of radiation sensors coupled through amplifiers to a logic circuit. The radiation sensors detect changes in ambient infrared radiation. The detection system has an electrical circuit operatively coupled to the heat sensor for producing a detection signal in response to the heat sensor detecting a change of temperature caused by the body heat of a person entering the detection pattern.

PIR motion detectors are perhaps the most frequently used home security device. [4] Passive IR motion detectors are usually designed to provide an indication to an alarm panel in response to detecting IR that is indicative of motion of the object. The alarm panel is responsive to receipt of the breach indication to cause an alarm condition to occur.

The other motion detector used in security system is an ultrasonic motion detector. It is commonly used for automatic door openers and security alarms [4]. It is inexpensive and can operate with narrow beam-widths. The ultrasonic transducers are the sensor that used in ultrasonic motion detector. It can be used to detect motion in an area where there are not supposed to be any moving objects. This type of motion detector is most commonly used in burglar alarm systems since they are very effective in this application [4].

In an ultrasonic motion detector, there are two transducers; one emits an ultrasonic wave and the other picks up reflections from the different objects in the area. The reflected waves arrive at the receiver in constant phase if none of the objects in the area are moving. If something moves, the received signal is shifted in phase. A phase comparator detects the shifted phase and sends a triggering pulse to the alarm.

Ultrasonic motion detectors have certain advantages and disadvantages when compared with other types of motion detectors. The main advantage is that they are very sensitive and extremely fast acting. However, the largest problem with this type of motion detector is that it sometimes responds to normal environmental vibration that can be caused by a passing car or a plane overhead. Besides, the installation options on this type of motion detector are limited because ultrasonic beams are easily blocked by thin materials, including paper. False triggering is easily caused by reflections from blowing curtains, pets, and flying insects.

While the passive infrared motion detectors offers problem where it can be falsely triggered by warm air movement or other disturbances that can alter the infrared radiation levels in an area. In order to prevent this problem, newer systems use two infrared sensors, which monitor different zones within a protected area. Logic within system triggers the alarm only when the two zones are activated in sequence, as would occur if a person walked through the protected area.

For that reason, the purpose of using the active infrared as a sensor to detect motion for this project is surely on the advantage offers by the sensor. Its capability on detecting motion with a simple design at lowest cost is needed to build an effective house security system based on motion detection.

1.5 Thesis outline

Chapter 1 explains the background of motion detection, the project objective, the project scope, and the literature review of motion detector for security system. The concepts of motion detector are the major element as a guide for the development of the security system.

Chapter 2 explains the systems involved for the development of the active infrared motion detector for house security system. The understanding of three systems which are the active infrared, motion detector, and house security system stated in this chapter needed for the development of the whole project.

Chapter 3 focuses on the methodologies for the development of the electrical structure and the implementations of microcontroller programming. It gives a brief review on the concept of active infrared motion detector, the electrical structure for hardware development, and the programming for the operation of the security system.

Chapter 4 discusses on the results obtained of the whole project. All discussions are concentrating on the result and performance of the security system. The discussion is valuable for future development of the security system.

Chapter 5 discusses about the conclusion on development of the active infrared motion detector for house security system. The recommendations and modification required on this project is stated in this chapter for further development.

CHAPTER 2

ACTIVE INFRARED MOTION DETECTOR FOR HOUSE SECURITY SYSTEM

2.1 Introduction

There are two systems on a development of the active infrared motion detector for house security system. The first system is an active infrared motion detector and the other one is the controller system. The first system concentrates on a development of an active infrared motion detector. It is consists of three element, which are the active infrared, the motion detector, and the house security system. The motion detector circuit will be control by the second system which is the microcontroller Intel 8051 based system. The combination between this two systems will built a project called an active infrared motion detector for house security system.

2.2 Active infrared (IR)

Infrared is an electromagnetic spectrum at a wavelength that is longer than visible light. It cannot be seen but it can be detected. Objects that generate heat also generate infrared radiation and those objects include animals and the human body whose radiation is strongest at a wavelength of 9.4μ meter. Infrared in this range will not pass through many types of material that pass visible light such as ordinary

window glass and plastic. However it can pass through, with some attenuation, material that is opaque to visible light such as germanium and silicon.

The active infrared sensors use invisible light to scan a defined area. In active infrared systems, there are two-piece elements which are consisting of an infrared transmitter and an infrared receiver. There is a 3/8 inch infrared beam between the transmitter which is placed on one side of the trail and the receiver which is placed on the other side of the trail. The transmitter and the receiver can be separated by as much as 150 feet.

The transmitter emits a beam of light into the scan zone. The light, which is reflected by the background returns to the receiver, which constantly monitors the scan zone. When a person or object enters the zone the infrared light is interrupted. It then sends a signal to the controller system, which is wired into the door controls. One variation of this operating mode is called ‘background suppression’. This is when the receiver only detects a change in the reflected light when a person or object enters the scan zone thus causing a reflectance variation of the light, sending a signal to the microcontroller thus trigger the alarm of the security system.

2.3 Motion detector

A motion detector is a device that contains a motion sensor and is either integrated with or connected to other devices that alert the user of the pre-sense of motion. An electronic motion detector contains a motion sensor that transforms the detection of motion into an electric signal. The electric signal can be connected to a burglar alarm system which is used to alert the home owner or security service after it detects motion.

An example of sensor that used in security system is an active sensor. Active sensors in motion detectors system are commonly used inside homes for a security system. An active motion detector emits optics or sound waves and measures

feedback to detect motion. The simplest type of active motion detector is commonly used in commercial doorways to trigger a doorbell.

A device is fixed to one side of the doorway, an optical sensor to the other. A beam of light will passes from the device through the sensor. When someone enters the establishment, the beam is broken, triggering the doorbell thus warn user for the intrusion. For that reason, active motion detectors can be purchased for home improvement security system. It is inexpensive devices that can add for more security to a home and provide peace of mind for home owners.

2.4 House security system

House security system is one of security that truly related to burglar or safety alarm system. Burglar and safety alarms are found in electronic form nowadays. Sensors are connected to a control unit via either a low-voltage hardwire which in turn connects to a means for announcing the alarm to elicit response.

In a new construction systems are predominately hardwired for economy while in retrofits wireless systems may be more economical and certainly quicker to install. Some systems are dedicated to one mission; handle fire, intrusion, and safety alarms simultaneously.

In common security system, the lights are triggered by motion gives the impression to user that someone is at home and able to see the burglar. Infrared motion detectors placed in house security system in crucial areas of the house can detect any burglars and alert the home owner or police.

The first security system invented, house alarms were triggered by the release of a pressure button fitted into a door or window frame. This basic alarm was

fundamentally flawed as the entire intruder needed to do to silence the alarm was to close the door or window.

While various systems on the market ranging from inexpensive house security alarms to highly sophisticated systems requiring professional installation. All modern alarms are based on the same foundation, the electric circuit which is completed either when the door is opened or closed depending on the security system designed.

The alarm is triggered when the circuit is altered and will not be silenced until a code is punched into the control panel. The most expensive and complicated alarm systems might also involve a combination of motion sensors and pressure pads to ensure even the most cunning intruder doesn't get his hands on treasures.

2.5 Microcontroller Intel 8051

The microcontroller 8051 is an 8-bit machine. Its memory is organized in bytes and practically all its instruction deal with byte quantities. It uses an Accumulator as the primary register for instruction results. Other operands can be accessed using one of the four different addressing modes available: register implicit, direct, indirect or immediate. Operands reside in one of the five memory spaces of the 8051.

The five memory spaces of the 8051 are the Program Memory, External Data Memory, Internal Data Memory, Special Function Registers and Bit Memory.

The Program Memory space contains all the instructions, immediate data and constant tables and strings. It is principally addressed by the 16-bit Program Counter (PC), but it can also be accessed by a few instructions using the 16-bit Data Pointer (DPTR). The maximum size of the Program Memory space is 64K bytes.

The External Data Memory space contains all the variables, buffers and data structures that can not fit on-chip. It is principally addressed by the 16-bit Data Pointer (DPTR), although the first two general purposes register R0 and R1 of the currently selected register bank can access a 256-byte bank of External Data Memory. The maximum size of the External Data Memory space is 64K bytes. The external data memory can only be accessed using the indirect addressing mode with the DPTR, R0 or R1.

The Internal Data Memory space is functionally the most important data memory space. In it resides up to four banks of general purpose registers, the program stack, 128 bits of the 256-bit memory, and all the variables and data structures that are operated on directly by the program. The maximum size of the Internal Data Memory space is 256-bytes. The register, indirect and direct addressing modes can be used in different parts of the Internal Data Memory space.

The Special Function Register space contains all the on-chip peripheral input and output registers as well as particular registers that need program access. The maximum number of Special Function Registers (SFRs) is 128, though the actual number on an 8051 family member depends on the number and type of peripheral functions integrated on. The SFRs can only be accessed using the direct addressing mode while the upper 128 bytes of the Internal Data Memory can only be accessed using the Indirect addressing mode.

The Bit Memory space is used for storing bit variables and flags. There are specific instructions in the 8051 that operate only in the Bit Memory space. The maximum size of the Bit Memory space is 256-bits. Bits can only be accessed using the bit instructions and the direct addressing mode.

2.6 Summary

The understanding on the elements of the systems involved for this project which are the active infrared, the motion detector, and the house security system is needed before proceed for the design and development process. This is important to ensure the project done according to the main idea of each part discuss in this chapter.

CHAPTER 3

METHODOLOGY

3.1 Introduction

There are several steps to be applied in designing an active infrared motion detector for house security system. The relevant information is gathered through literature review from previous chapter.

Data on motion detection and security system projects has been collected where the theoretical design is studied based on the motion detector for security concept. The understanding on the electrical structure for the hardware development is needed for the design circuit process of the motion detector and the basic security circuit.

The next is the hardware development according to the circuit designed. This process is just only being proceed if each part of the circuit being improved is valid, else, it will be repeated until it is valid as the theoretical. Once the hardware development circuits have the output as the expected, then, the comparison for both hardware and theoretical analysis will be done.

Next is the step where software structure is developed for the security system to be interface with the hardware development. While the final step of this research is on applying the whole project to the real house entrance like doors and windows.

3.2 The concept of motion detector for security system

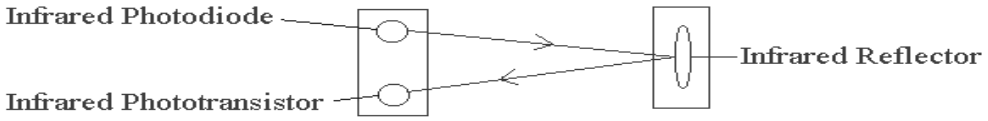


Figure 3.1 Active infrared motion detectors

Figure 3.2 shows the concept of an active infrared motion detector for a security system. In the active system each sensor consists of two housings. The first housing contains an infrared-emitting diode and an infrared-sensitive phototransistor as the infrared detector. The other housing contains an infrared reflector to reflect the infrared signal. When positioned in front of an entrance to a protected area, the two housings establish an invisible beam.

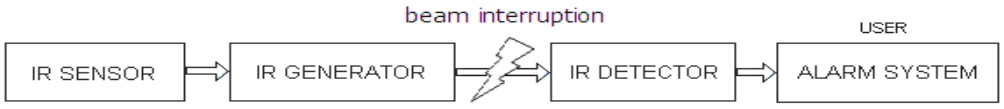


Figure 3.2 The concept of infrared motion detector for security system

A person who enters the area will interrupts the beam causing an alarm to be triggered. For this type of motion detector uses the basic concept of the active infrared motion detector. An interruption in the signal modulated pulsating beam transmit by an infrared diode while receive by an infrared detector will set ‘on’ or ‘off’ the alarm of the security system.

3.3 Hardware development

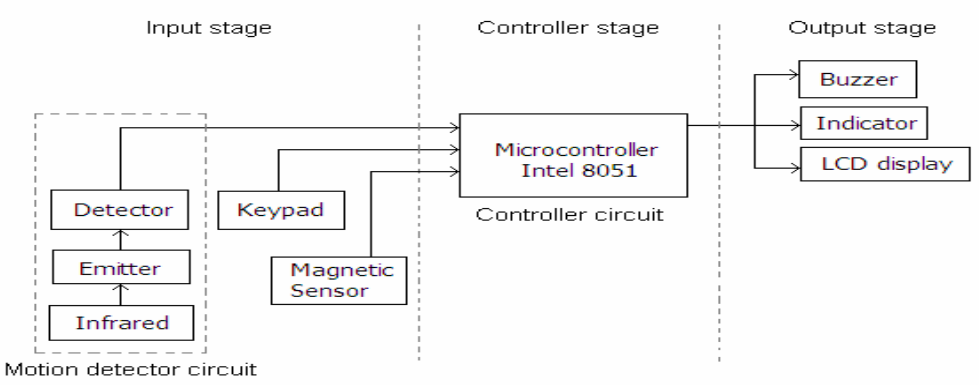


Figure 3.3 Block diagram for the house security system

The hardware development is divided into three stages as shown in block diagram above. The inputs stage of the security system is the motion detector circuit, keypad, and magnetic sensor. The second stage is the controller unit which is the microcontroller Intel 8051. The purpose of using microcontroller is to control the whole system operation by sending data to the output stage which is the LCD display, indicator, and buzzers.

3.3.1 Motion detector circuit

In designing the infrared motion detector circuit, it is based on two basic principle of active infrared motion detector which is the infrared transmitter and infrared receiver as shown in Figure 3.4.

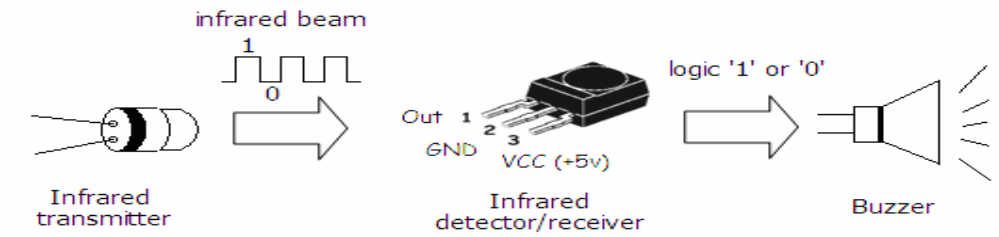


Figure 3.4 Basic principle of infrared operation

3.3.2 Infrared transmitter

For the infrared transmitter which is also known as emitter circuit, it is on a basic design of timer 555 astable operation. The output of timer is connected to the infrared transmitter is used to produce pulse using an astable timer circuit.

In astable circuit operation, pulse will continually generated until the power supplied through the circuit is removed. The astable circuit produces a continuous train of pulses at any frequency required. This means that the 555 timer can operate repeatedly; it will switch ‘on’ and ‘off’ continually to generate data for the infrared transmission.

3.3.3 Infrared receiver

The infrared receiver which is also known as infrared detector receives the data transmitted by the infrared transmitter circuit. It is a simple electronics device on detecting infrared signal.

This infrared detector can be directly connected into the controller circuit to produce logic high ‘1’ or low ‘0’ from the output terminal thus activate or deactivate the controller system operation. The range of infrared detector components according to datasheet stated that the infrared detector can fully operates on detecting the infrared signal of 38 to 45 kHz.

3.3.4 4 × 4 hexadecimal keypad

The purpose of using 4 × 4 hexadecimal keypad in the project is as an input where secure code entries to activate of deactivate the security system operation. The keypad provides eight interface pins, where one pin for each row and column of the keypad matrix. This 4 × 4 hexadecimal keypad is connected to the keypad encoder (MM74C922) to control the keypad bouncing in the hardware development.

3.3.5 Keypad encoder (MM74C922)

The MM74C922 key encoders provide all the necessary logic to fully encode an array of 4 × 4 hexadecimal keypad. It is used to encode the data received from the keypad code entry thus convert into binary code. These binary code is in hexadecimal number is required in data bus of the microcontroller system.

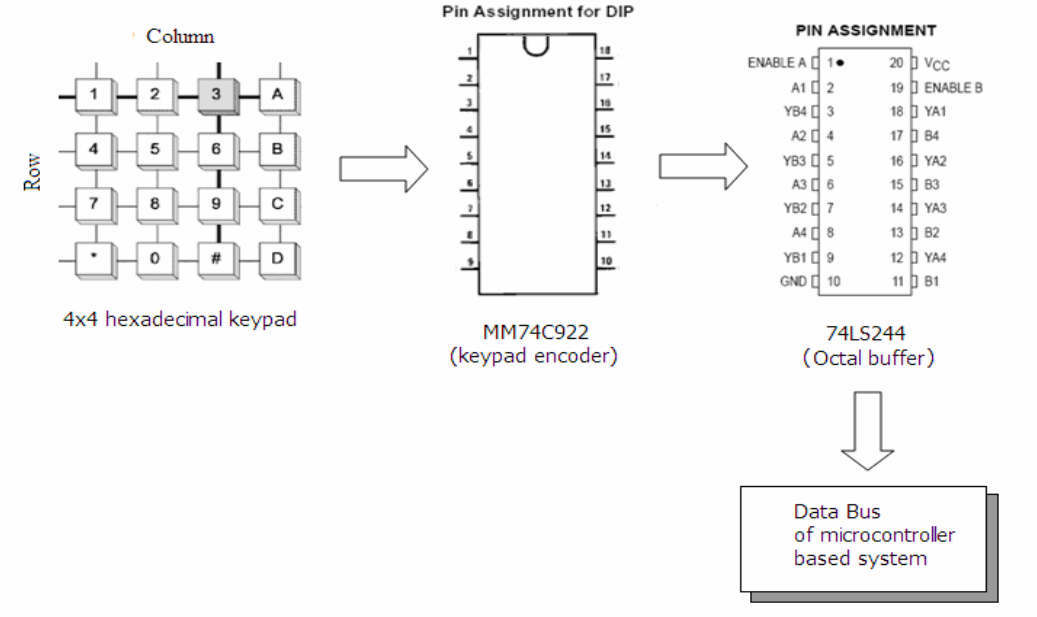


Figure 3.5 Interaction between keypad, keypad encoder, and octal buffer

The figure 3.5 above shows the block diagram on the interaction between the 4 × 4 hexadecimal keypad, keypad encoder (MM74C922), and the octal buffer (74LS244). The keypad is used as a switch to give logic configuration to the keypad encoder.

While the encoder will encode the data configuration from the keypad into the binary code based on datasheet of the encoder. These binary codes will be stabilized using the octal buffer to be process in the microcontroller system. These data will be process by the microcontroller thus running the system operation based on software designed for the security system.

3.3.6 Octal buffer (74LS244)

A buffer is simply a unity-gain amplifier, usually with very high input impedance and very low output impedance. It allows a connection of heavily loaded to another load which requires a lot of current like capacitive load.

The octal buffer is used to isolate one device from another by eliminating loading effects. It provides a ‘buffer’ between the two devices. The 74LS244 are octal buffers and line drivers designed to be employed as the memory address drivers, clock drivers, and bus-oriented transmitters/ receivers which provide improvement on the controller board density.

3.3.7 Microcontroller circuit (Intel 8051 – expended mode)

The controller systems that use to control the motion detector system and other electronic devices are the microcontroller Intel 8051 – expended mode. In expended mode configuration, external ROM and RAM are used to add the data memory to be more than internal memory provided by the Intel manufacturer.

The purpose of using an expended mode for the project is to expend more data available on developing and designing an excellent operation of the security system.

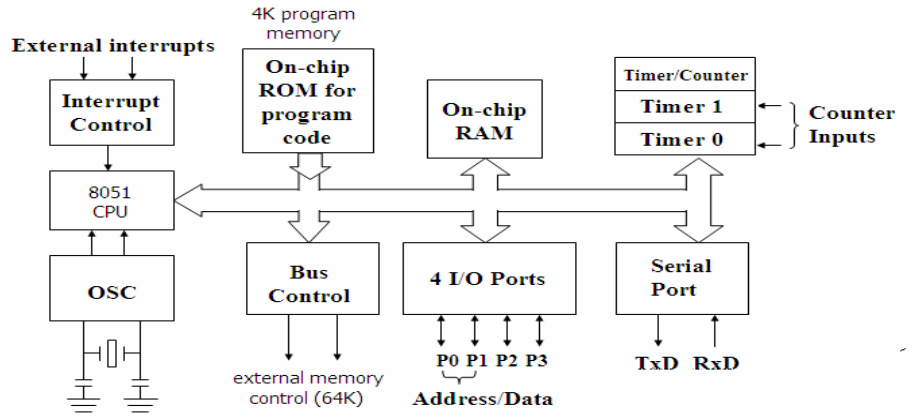


Figure 3.6 Intel 8051 block diagram

The microcontroller Intel 8051 is a widely used on controlling devices and instruments due to its flexibility, existence of multiple producers. It consists of a simple architecture introduced by Intel in the year of 80's. It produced with 128 bytes of RAM, 4K bytes of on-chip ROM, 2 timers, one serial port, and 8-bits wide of ports.

This type of microcontroller become very popular after Intel allowed another manufacturer to make and market any flavor of the 8051, and still remains compatible with 8051. There are various versions of 8051 with different speeds and amount of the on-chip ROM which is valuable on controlling a system designed.

3.4 Software development

The software structure is using the assembly language where a set of program based on security system is assembled using ASM51 assembler. The 8051 Cross Assembler (ASM51) takes an assembly language source file created with a text editor and translates it into a machine language object file. This translation process is done in two passes over the source file.

During the first pass, the Cross Assembler builds a symbol table and labels that used in the source file. While the second pass of the Cross Assembler is actually translates the source file into the machine language object file. During the second pass, the listing file of the assembled is generated for the analysis purpose.

A security program as shown in flow chart below were written in notepad based on 8051 instruction set before assembling process to get the binary code. This binary code is then used to be load into the location of the memory thus operates the hardware developed.

The flowchart as shown in Figure 3.7 is a basic designed for the security system operation. In this security system, the security code is set as ‘1985’ to activate or deactivate security system. When user closed the door, the system will activate where the magnetic switch and infrared is in active condition.

To deactivate the system where opening door without alarm, user must enter the deactivation code. The chance to deactivate the system is once before keypad locked, where user can not enter the code anymore. If deactivation code was correct, green LED will ‘on’ while alarm will ‘off’ condition. System at this time is successfully deactivated and user can open the door without alarm.

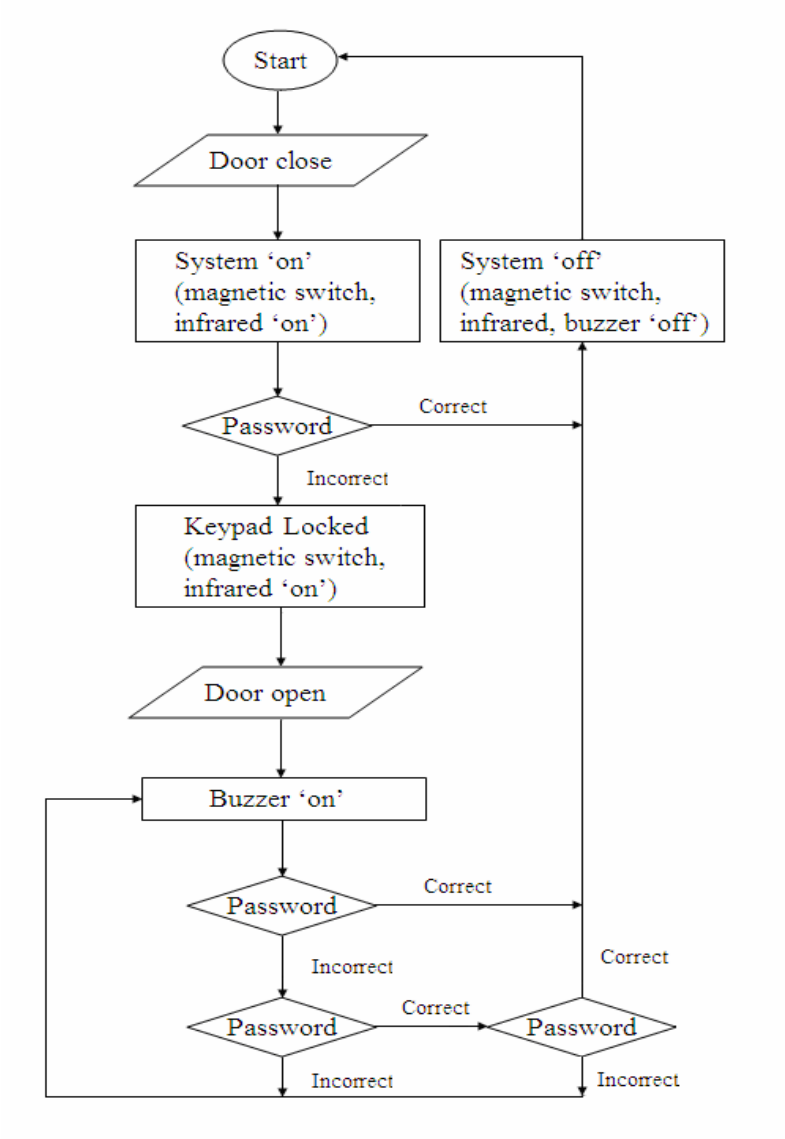


Figure 3.7 Flow chart of software design for the security system

If deactivation code enters was incorrect, system is still activate and for that, red LED and alarm will 'on' where the system is in a warning mode. The system will keep in this mode until the correct deactivation code entered.

When the correct deactivation codes enter, the reconfirm code is needed where user need to reenter the correct deactivation code before the system successfully deactivate; the green LED will 'on' and the alarm will 'off' condition.

The reconfirm code is designed in the system to make more security for the system developed. If intruder open the magnetic switch without entered the correct security code, thus the alarm will ‘on’. User must enter the correct deactivation code to turn ‘off’ the alarm system. The activation and deactivate code for this system are ‘1985’ and the ‘enter’ button codes are ‘A’ for every codes entered.

3.5 Summary

There are three elements discussed in this chapter which are the concept of motion detector for security system, the hardware development, and the software designed for the security system. Each part of these elements is related to each other. The understanding of the concepts and methods on developing the project is very important to achieve the main objective for the whole project.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Introduction

This chapter will briefly discuss on the results and discussions of the hardware development. There are three major modules in developing the hardware structure which are the active infrared motion detector circuit, keypad module circuit, and the most important; the controller circuit

4.2 Motion detector circuit

The hardware development of the active infrared motion detector system is divided into two parts, which are the infrared emitter circuit and the infrared detector component.

4.2.1 The emitter

The emitter which also known as infrared transmitter circuit was build as shown in Figure 4.1. A probe or the oscilloscope is attached to the infrared LED

while the adjustable resistor is varied to calibrate the emitter to transmit 38 kHz square wave. The Figure 4.1 shows the final configuration of the emitter circuit:

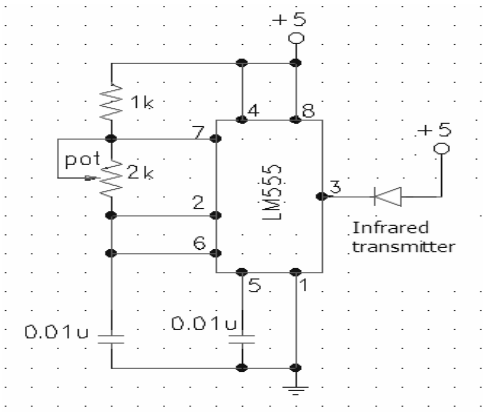


Figure 4.1 The emitter circuit

Figure 4.2 shows the result obtained from the oscilloscope where the frequency generated by the emitter circuit is 38.67 kHz square wave. At this generated frequency, the voltage maximum produced is 1.60V and the peak to peak voltage is 1.16V.

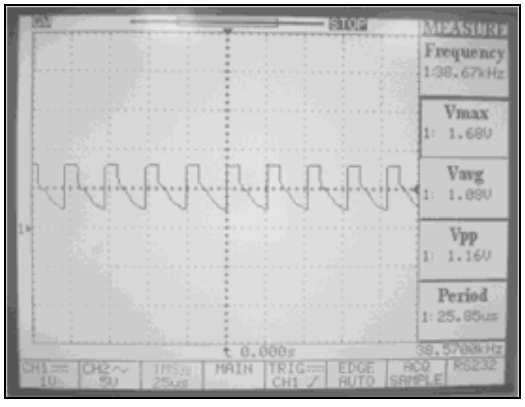


Figure 4.2 The 38 kHz frequency generated by the emitter circuit

4.2.2 The detector

Figure 4.3 shows the detector component which is used for the infrared detector. This electronic device operates at 5 volts dc supply connected to pin 3 while the grounding is connected to pin 2.

When the infrared detector received the signal, it generates logic low ‘0’ to the output pin 1. While detecting motion or a solid object passed through the infrared beam, logic high ‘1’ will be generated at the output pin 1 of the infrared detector component.

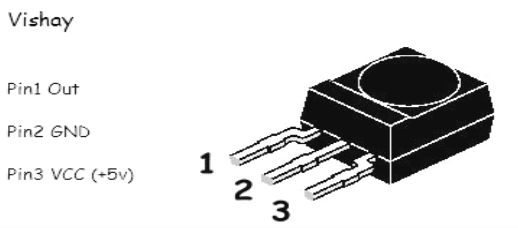


Figure 4.3 The infrared detector component

Figure 4.4 shows the module of the motion detector circuit. On the right side of the picture shows the infrared transmitter circuit while the other side is the controller circuit with the infrared detector.

Figure 4.4 The motion infrared motion detector module

The measured range between the transmitter and the detector component is 100 cm; based on the most door entrance. The generated frequency was measured using the oscilloscope (on the top of the Figure 4.4) is 38.67 kHz and it is suitable for the infrared components to be function as stated in the datasheet of the infrared detector.

Figure 4.5 Infrared signal received by the infrared detector

When no object or motion passed through the invisible beam generated by the infrared transmitter circuit, logic low; data '0' generated for the output of the detector will send to the LED of the trainer which is used as the output indicator. As shown in Figure 4.5 above, LED is 'off' state.

Figure 4.6 Object detected; no signal received by the infrared detector

While an object or motion passed through the invisible beam between the infrared transmitter and infrared detector component, logic high ‘1’ is generated from the pin 1 of the detector component to turn ‘on’ the LED of the trainer as shown in Figure 4.6 above. These means the infrared is capable to detect motion when intruder passed thru the invisible infrared beam.

4.2.3 Results for the motion detector system

For infrared motion detector circuit, it consists of two units which are the transmitter and receiver. These two units can be placed separately or next to each other using an infrared reflector. Power sources that needs for the emitter circuit and detector component are 5 volts dc.

The output of the infrared detector is connects to any LED as a module before connected to the microcontroller Intel 8051 at pin 13. This pin is an external interrupt pin where logic high ‘1’ is needed to execute the interrupt program in the security system. The range of the infrared detection is up to 100cm for the 38 kHz generated frequency by the transmitter (emitter) circuit.

4.3 Keypad module circuit

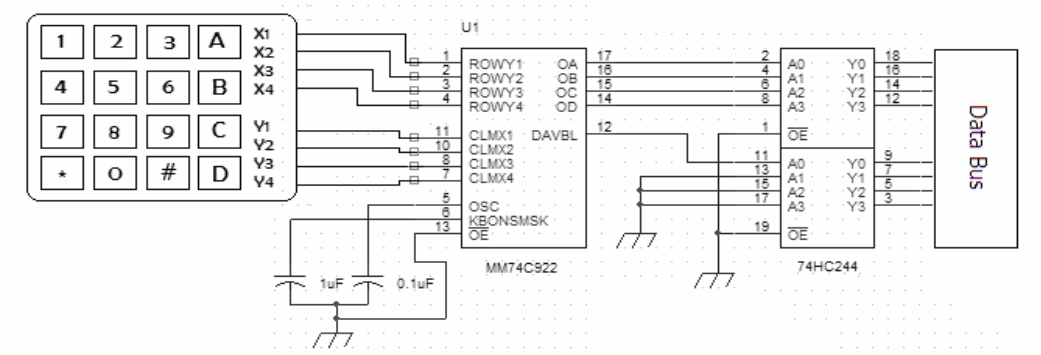


Figure 4.7 Keypad connections as an input of the controller circuit

For the keypad module, the hexadecimal number is defined using manual testing to configure the number between keypad and the binary number. The pin number 17, 15, 13, and 11 is set as the most significant bit (MSB) while the pin number 8, 6, 4, and 2 is set as the least significant bit (LSB) of the octal buffer 74244. For that, the MSB value for the binary number should be 1 while the LSB value is depends on the data input from the keypad encoder MM74C922.

Table 4.1: The keypad configuration number

Keypad	Binary number	Keypad	Binary number	Keypad	Binary number	Keypad	Binary number
1	10H	2	14H	3	18H	A	1CH
4	11H	5	15H	6	19H	B	1DH
7	12H	8	16H	9	1AH	C	1EH
*	13H	0	17H	#	1BH	D	1FH

A set of program to identify the binary number of the keypad is written (Appendix D), assembled, and loaded into the controller circuit. The tested process to identify the configuration between the binary number and the keypad itself is done one by one to get the results as shown in Table 4.1.

4.4 Controller circuit

The controller circuit is using Intel 8051 microcontroller based system. There are 3 major phase in designing the controller circuit for the security system. The first phase is a design of a simple serial communication 8051-based system. The purpose of this phase is to provide a basic understanding of the 8-bit microcontroller Intel 8051.

It is on the use of serial communication software to communicate between personal computer (PC) and the electrical device of the hardware circuit using serial port. Serial port is used because it is easy to use and its only has line to transmit and receive data for a communication process.

The second phase is a design circuit for a simple input and output interface using 7-segment display. In this phase, hyper terminal is used to download and execute the microcontroller program after assembled process done using ASM51, the Intel 8051 assembler. The purpose of the assemble process is to convert the sets of Intel 8051 program to a binary file to be load into the location address of the external memory (RAM) of the electrical structure.

The EPROM programmer and EPROM eraser is no longer use to download the hex-file. Further, it will be much easier for downloading the program into the hardware design circuit. The final phase is a design circuit for the basic security system using 8051 microcontroller.

4.4.1 Results for the controller system

The results for the controller circuit is based on three phase done. The first phase is focus on a hardware development for simple serial communication 8051 – based system, while the second phase is on testing the input/ output port by interfacing it directly to the 7-segment display. The last phase is the objective of this project which is on development of the security system.

4.4.2 Phase 1 (Simple serial communication 8051 – based system)

At first, the based circuit of expended mode with other electronic devices was design on a strip board as a basic hardware development of the controller circuit. All

the electronic components are connected using wire-wrapping technique, based on the design circuit to build a simple communication 8051 – based system.

A set of program for the monitor program (Appendix A) were assemble using ASM51 assembler before burned into the external memory EPROM. PORT 1 of the microcontroller is used as an output port, where it is directly connected to the seven segment display as a preparation for the next phase.

The 8051-based system can communicate with PC (personal computer) using the EIA232 connectors via MAX233 line drivers. The communication software that used to view the monitor program is called HyperTerminal. All the setting of the HyperTerminal as followed:

- Baud rate = 9600 baud
- Data bits = 8bits
- Parity = none
- Stop bits = none
- Start bits = 1

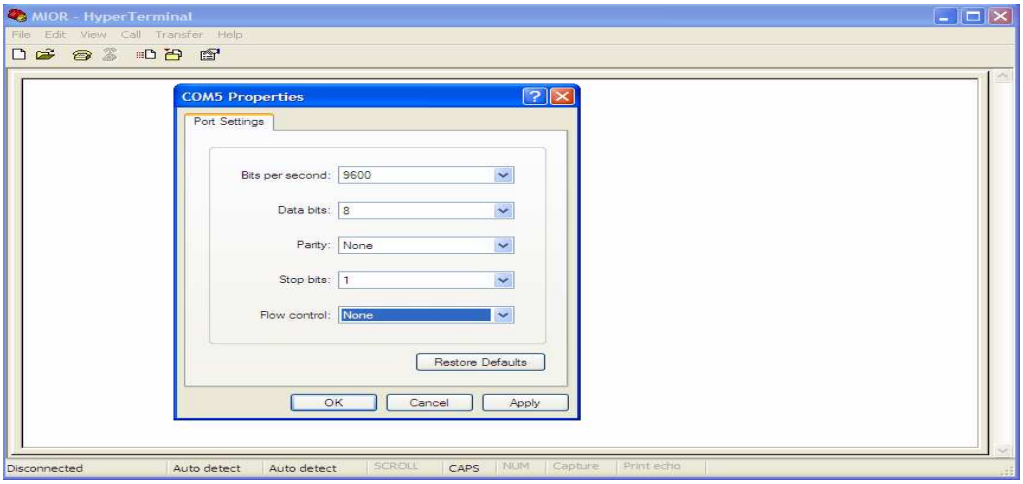


Figure 4.8 The HyperTerminal setting

After the setting is done, the based circuit is ‘on’ and it is ready for a loading and executing process. The HyperTerminal communication software will view monitor program as shown in Figure 4.9 below:

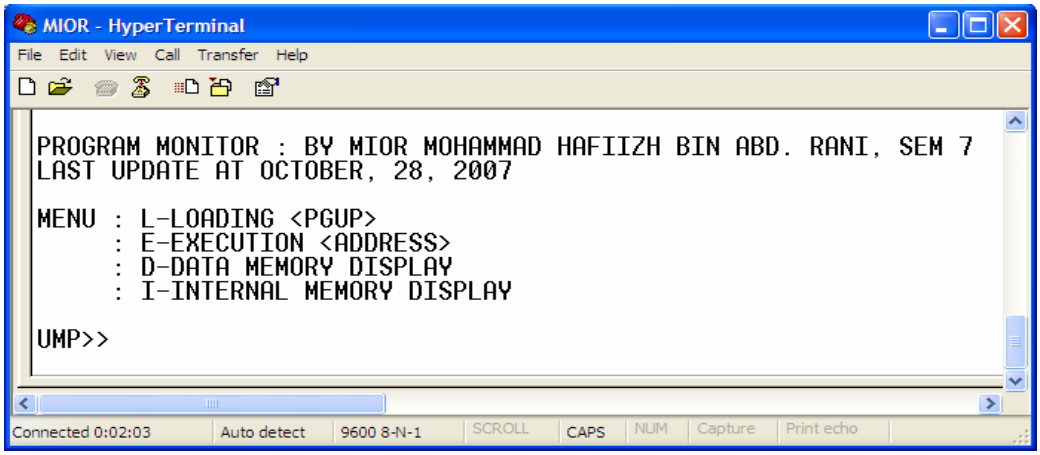


Figure 4.9 The monitoring program

From the monitoring program shown in Figure 4.9 above, a set of binary program for the controller 8051 system can easily upload to the RAM. Symbol ‘L’ is refer to load process where program is loading from PC to the location address of the RAM, where it must be stated in the program written. The symbol of ‘E’ refers to execute the program address to command the system to start from which address of the program.

4.4.3 Phase 2 (Simple input/output interface – 7 segment display)

For the second phase, PORT 1 of the microcontroller is set as the output, where it is ready to identify either the 8051-based system was fully function by interfaced to the 7-segment display. The port is an output drain. For the reason, it requires a 10kΩ pull-up resistor to each output pin to limit current, thus protecting the 7-segment display from damage.

The program for displaying numbers was written on a notepad before saved as the extension of asm file. The asm file (Appendix B) is assemble using ASM51 (Intel 8051 assembler) and inspect the existence of binary file as the file will be used in the system.

After all was done, the ‘L’ command is now used to load object code file from HyperTerminal software. If the loading process is completed, the 7 – segment display is ready to function by the execution program using the ‘E’ command.

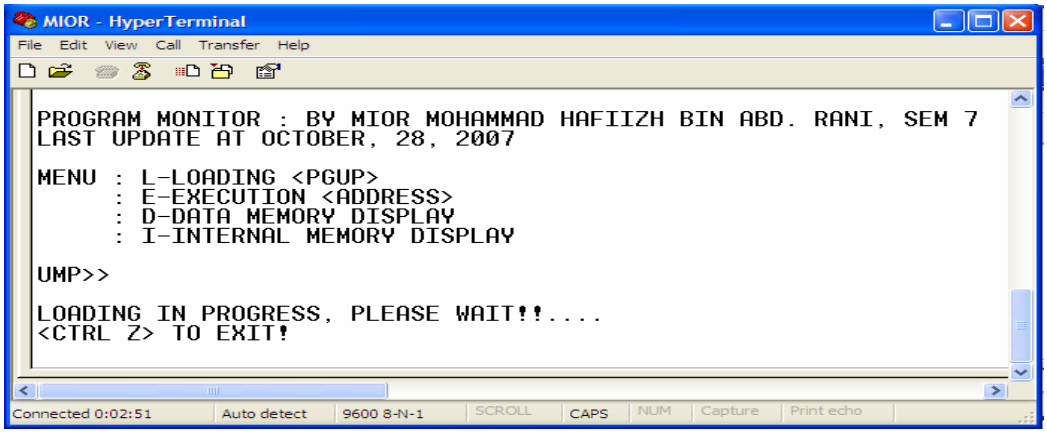


Figure 4.10 Loading the binary file process

The steps of the loading and executing program from PC to the hardware board using HyperTerminal software as follows:

- Firstly, monitoring program is viewed by HyperTerminal software
- Then press ‘L’ to load the binary program from PC to the system
- After loading progress finish, press ‘E’ for execution address. In this system, the address for RAM stated in the program is 8000.

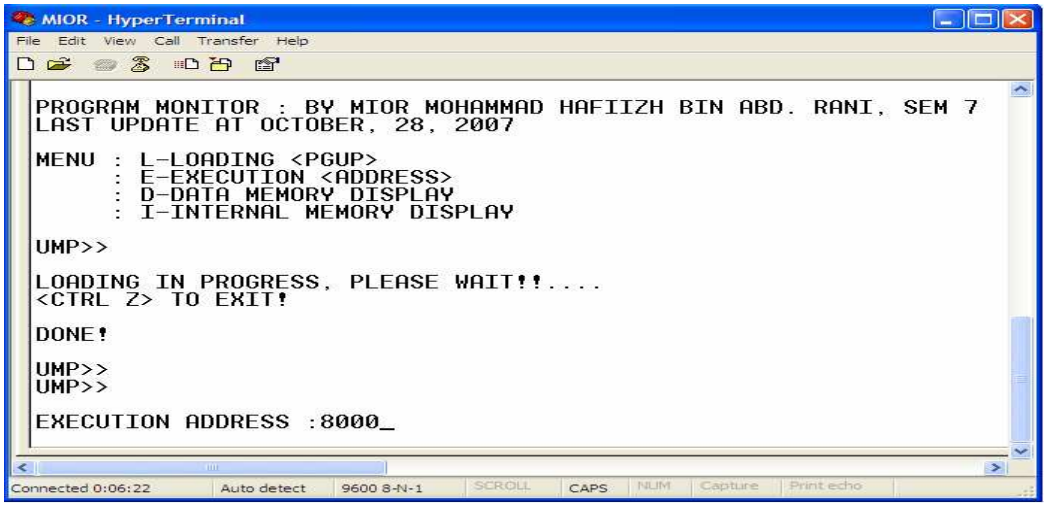


Figure 4.11 The execution address

After the execution process, system will started display the numbers based on program loaded. This system will keep displaying the program until the reset button is pushed; the running program will automatically lost because the load program into the RAM is just removed. These mean the programs will volatile when the system was reset. To display again the number, the above steps must be follows to load the program and execute the system.

4.4.4 Phase 3 (The security system)

The phase 3 is the final stage for this project. An additional circuit is added to the based system. For additional circuit is using the electronics components as follows:

- LED
- Buzzer
- Octal buffer
- Magnetic switch
- 4 × 4 hexadecimal keypad
- MM74C922 keypad encoder

The 4 × 4 hexadecimal keypad used to enter the security code. In this system, the hardware bouncing is to detect hexadecimal keypad. The keypad encoder is used as a driver to encode the data read from the 4 × 4 hexadecimal keypad. Magnetic switch is use as a sensor (input for this system).

The LED is use as an indicator to show either the data available during entering the code while buzzer is use as an alarm for this security system. For the security system developed, a set of program for the security system is assembled based on the flowchart stated in the software structure. Appendix C are the security program based on the software structure stated in methodology chapter.

The security code is set as ‘1985’ to activate or deactivate security system. When user close the door, the system will turn ‘on’ where magnetic switch and infrared is in active condition.

When the correct activation codes are entered, the green LED will turn ‘on’ (this means the system will ‘off’ condition). At this time, user can open the door and no alarm will turn ‘on’. If intruder open the magnetic switch without entered the correct security code, the alarm will turn ‘on’.

User must entered deactivation code to turn ‘off’ the system. The activation and deactivate code for this system are ‘1985’ and the ‘enter’ button for each codes entered are ‘A’.

4.5 Discussions

The used of MAX233 is to convert the TTL logic to EIA232 standard to ensure that data can be transmitted serially up to 15 meter. The 11.0592MHz crystal is used to generate the clock and suite with the serial port that used 9600 baud rate. HyperTerminal program is used to communicate with the serial port.

The monitor program is burned into the external memory EPROM 2764. All programs must be loaded into RAM 6264 before execution process using the monitor program. DB9 is used via reben cable to communicate between 8051-based system and the personal computer.

Assembly language is used to assemble program from any available editor such as notepad to produce list file and hex file. Only port 1 is available because the 8051-based system is using external memories. HyperTerminal is used to download and execute the program into RAM 2764. The program loaded must be in hex file. The data in RAM will be disappeared if the power supply is off. This situation happened because RAM is volatile.

Timing for reading the keypad data is not consistent. For that reason, user must enter the code carefully and slowly. Using hardware to control keypad bouncing made the system become complicated. It needs to be determining the connection between hardware and test it step by step.

When designed software for the system, it has too many errors occur. Because of that the control data port by port is needed to make sure the software program based on the flow diagram designed for the security system.

In the software development, the MOD51 is used to define the specific location that required of some instruction set, for example, PORT 1 is in location address at 90H in SFR RAM Address. This MOD51 converts the word to byte specification which is needed in the assembling process.

4.6 Summary

The result of each elements stated from the previous chapter has been discussed in this chapter. The analysis is done on discussing method to solve problem occur thus give an advantage for the future development. The motion detector and the controller circuit are succeeding developed thus capable to achieve the objective of this project.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Summary of work

One of the major goals of this infrared motion detector for house security system project is to build an effective home security system at lowest cost. The home security system is design using microcontroller Intel 8051 to give as much reliability as possible, within the basic constraints of the security process. As expected conclusion, this project is useful because it offers flexibility and effective on detecting motion for house security system.

5.2 Future recommendations

For the future plan of this project, it is recommended other candidate to do more studies on the related information in order to develop a new design of house security system that match up with other updated house security available in the market. Two matters are recommended to be added in the security systems which are the wireless security system and the alarm monitoring system.

The wireless home security has many advantages compared to wired security system. Moreover, the alarm monitoring system can be added to the basic security

system in order to make the system more reliable and effective thus the objective of house security is clarify.

5.2.1 Wireless house security system

Over the past several years, the face of home security has changed dramatically. Thus, new technology has introduced; the wireless home security systems which capable to secure a lawn, driveway, porch, and other outdoor areas of home. Many home owners prefer wireless home security systems over hard-wired systems because of its versatility. With the numerous accessories available, it is easy to extend the wireless home security system to the outdoors area.

There are many advantages on choosing the wireless home security system to cover the outdoor areas. The low cost and flexibility makes wireless home security systems ideal for outdoor use.

Wireless home security systems allow home owners to modify their systems easily when the continuous changes in outdoors, such as growing trees or new landscaping. Since there are no wires to take into consideration, it is easy to move the monitoring and motion detectors for wireless home security systems to accommodate a newly area of house security.

5.2.2 Alarm monitoring system

An alarm monitoring system means user property to watch over their home area by an external center. Home owner can easily know their home security system at any time of day or night. Some monitored alarm systems include a personal attack button, which if pressed, will initiate an immediate call to the police.

Similarly, if the alarm is triggered by the smoke detector, the fire service will be immediately dispatched. In the case of a power failure, home security system will fall back onto its battery reserves to keep on the operation. If the phone line is cut, the alarm will be signaled immediately at the external monitoring center.

5.2.3 Costing and commercialization

The overall cost of the whole project is based on the hardware development. As discussed in previous chapter, the hardware development consist of two systems and for that, costing of the whole project is surely depends of the electronic devices used for the development process.

Table 5.1 shows the overall cost for the hardware development of this project. For the motion detector system, it is consists of six electronic components which are the infrared transmitter, 555 timer, capacitor, resistor, potentiometer, and the infrared detector component, while the others are components used for the controller system.

Table 5.1: Total cost on the development of the security system

Electronic devices	Specification	Quantity	Cost (RM)
Infrared transmitter	-	1	3.00
Infrared detector	PIC-1018SL (Waitrony)	1	10.00
Crystal	11.0952 MHz	1	8.00
Latch	74373	1	1.35
Decoder	74138	1	0.95
EPROM	2764	1	8.90
RAM	6264	1	7.50
Serial interface	MAX 233	1	9.50
Microcontroller	Intel 8051	1	20.00
Wire Wrapping	30 meter	1	8.00
Buzzer	3-24 Volt	1	1.50

Keypad	Hexadecimal 4×4	1	23.00
Magnetic switch	Electromagnetic sensor	1	12.00
AND gate IC	7408	1	0.60
OR gate IC	7404	1	0.60
DB9	Female	1	0.60
Reben Cable	3 ways for 1 meter	1	5.00
Timer	LM 555	1	0.85
Buffer	74LS244	1	1.50
Keypad encoder	MM74C922	1	18.00
IC base	8 pins	1	0.12
IC base	14 pins	2	0.30
IC base	18 pins	1	0.20
IC base	20 pins	3	0.23
IC base	28 pins	2	1.00
IC base	40 pins	1	0.90
LED	Red	1	0.11
LED	Yellow	1	0.12
LED	Green	1	0.12
Potentiometer	5kΩ	1	0.80
Resistor	100Ω	1	0.10
Resistor	1kΩ	1	0.10
Resistor	4.7kΩ	12	1.50
Resistor	10kΩ	1	0.11
Capacitor	0.01x10-6 F	2	0.30
Capacitor	1x10-6 F	1	0.10
Capacitor	0.1x10-6 F	2	0.20
Independent strip board	-	2	8.00
Total cost			**RM 155.16**

The total cost for the development of the security system is RM 155.16. These value is quite cheap compared to other security system and for that reason, it have the value to be commercialize in market nowadays.

Besides, home securities become important and necessary since long time ago for home owner in securing their home properties. The continuous technology is still developing an effective security system to full fill the need of home owner. Thus, this project is proudly useful for them on securing their home without thinking of the price anymore.

REFERENCES

[1] I. Scott MacKenzie (1992). The 8051 Microcontroller: Prentice Hall.

[2] David Calcutt, Fred Cowan, Hassan Parchizadeh (1994). 8051 Microcontrollers; An Applications Based Introduction: Newnes.

[3] David Calcutt, Fred Cowan, Hassan Parchizadeh (1997). 8051 Microcontrollers; Hardware, Software, and Applications: ARNOLD.

[4] Namita Ahuja, Robert Mitchell, Karl Schrader, Michael Perry, Joe Ha (1996) Final project: *An Ambient-Light-Ignoring Infrared Active Motion Detector*
Available at http://www.ee.washington.edu/conselec/Sp96/projects

[5] Wikipedia, Online encyclopedia, Citing Internet sources
URL http://wikipedia.org

[6] How Infrared motion detector components work, Citing Internet sources
URL http://www.glolab.com/pirparts/infrared.html

[7] Active infrared motion detector and method for detecting movement, Citing Internet sources
URL http://www.patentstorm.us/patents/4896039.html

APPENDIX A

Program for monitoring program (Phase 1)

```
;#########################################################################
;TITLE: SERIAL MONITOR PROGRAM
;DATE: 17HB.SEPTEMBER.2007
;BY     : MIOR MOHAMMAD HAFIIZH BIN ABD. RANI
;ID     : EE05005
;#########################################################################

              $MOD51
LF            EQU   0AH
CTRLZ         EQU   1AH
COUNTER       EQU   0BH
CR            EQU   0DH
BYT           EQU   0EH
BUF1          EQU   0FH
ASC           EQU   30H
STACK         EQU   10H
SPACE         EQU   20H
LED           EQU   2000H

              ORG         0000H
              LJMP        START

              ORG         03H
              LJMP        9B00H

              ORG         0030H
START:        MOV         SP,#STACK
              MOV         SCON,#01011010B
              MOV         TCON,#11010010B
```

```
        MOV     TMOD,#20H
        MOV     TH1,#0FDH
        MOV     TL1,#0FDH

        SETB    TR1
        MOV     A,#0

        LCALL   PUT_CHAR
        MOV     DPTR,#MSG1
        LCALL   PUT_IN

MAIN:   MOV     A,#LF
        LCALL   PUT_CHAR
        MOV     DPTR,#MSG3
        LCALL   PUT_IN
        LCALL   GETCHAR
        XRL     A,#'L'
        JNZ     MAIN1
        MOV     A,#LF
        LCALL   PUT_CHAR
        MOV     DPTR,#MSG4
        LCALL   PUT_IN

        LCALL   HEXFILE
        AJMP    MAIN

MAIN1:  XRL     A,#'L'
        XRL     A,#'E'
        JNZ     MAIN2
        AJMP    EXEC

MAIN2:  XRL     A,#'E'
        XRL     A,#'D'
        JNZ     MAIN3
```

```
                AJMP    MEM

MAIN3:          XRL     A,#'D'
                XRL     A,#'I'
                JNZ     MAIN
                AJMP    I_MEM

MEM:            MOV     DPTR,#MSG7
                LCALL   PUT_IN

                MOV     DPTR,#MSG8
                LCALL   PUT_IN

                LCALL   GET_ADDR
                LCALL   OUT_SPC
                LCALL   OUT_STAR

                MOVX    A,@DPTR
                LCALL   OUT_BYTE
                INC     DPTR
                MOV     A,#LF
                LCALL   PUT_CHAR
                LCALL   DELAY
                AJMP    MAIN

I_MEM:          MOV     DPTR,#MSG9
                LCALL   PUT_IN
                MOV     DPTR,#MSG10
                LCALL   PUT_IN

                LCALL   BYTE_IN1
                MOV     R0,A
                LCALL   OUT_SPC
                LCALL   OUT_STAR
```

```
                MOV     A,@R0
                LCALL   OUT_BYTE
                MOV     A,#LF
                LCALL   PUT_CHAR
                LCALL   DELAY
                AJMP    MAIN

;*****************************************************************

EXEC:           MOV     DPTR,#MSG2
                LCALL   PUT_IN
                LCALL   BYTE_IN1
                MOV     DPH,A
                LCALL   BYTE_IN1
                MOV     DPL,A
                PUSH    DPL
                PUSH    DPH
                RET

;*****************************************************************

HEXFILE:        LCALL   GETCHAR
                XRL     A,#CTRLZ
                JZ      TO_EXIT
                XRL     A,#CTRLZ
                CJNE    A,#':',HEXFILE
                LCALL   BYTE_IN

                JZ      TO_MAIN
                MOV     COUNTER,A
                LCALL   WORD
                LCALL   BYTE_IN
```

```
AGAIN:      LCALL     BYTE_IN
            MOVX      @DPTR,A
            INC       DPTR
            DEC       COUNTER
            MOV       A,COUNTER
            JZ        HEXFILE
            SJMP      AGAIN

TO_MAIN:    MOV       DPTR,#MSG5
            LCALL     PUT_IN
            RET

TO_EXIT:    CLR       C
            MOV       DPTR,#MSG6
            LCALL     PUT_IN
            RET

;*********************************************************************

BYTE_IN:    LCALL     SER_IN
            ANL       A,#7FH
            LCALL     ASC_HEX
            SWAP      A
            MOV       BUF1,A

            LCALL     SER_IN
            ANL       A,#7FH
            LCALL     ASC_HEX
            ADD       A,BUF1
            RET

;*********************************************************************

BYTE_IN1:
```

```
                LCALL   SER_IN
                ANL     A,#7FH
                LCALL   SER_OUT
                LCALL   DELAY
                LCALL   ASC_HEX
                SWAP    A
                MOV     BUF1,A

                LCALL   SER_IN
                ANL     A,#7FH
                LCALL   SER_OUT
                LCALL   DELAY
                LCALL   ASC_HEX
                ADD     A,BUF1
                RET

;*****************************************************************

WORD:           PUSH    ACC
                LCALL   BYTE_IN
                MOV     DPH,A
                LCALL   BYTE_IN
                MOV     DPL,A
                POP     ACC
                RET

;*****************************************************************

ASC_HEX:        CJNE    A,#'9',NINE
                JMP     LESS

NINE:           JC      LESS
                ADD     A,#9H
```

```
LESS:       ANL         A,#0FH
            CLR         C
            RET

;*********************************************************************

HEX_ASC:    PUSH        ACC
            MOV         A,ASC
            CJNE        A,#9,NINE1
            JMP         NINE2

NINE1:      JC          NINE2
            ADD         A,#7H

NINE2:      ADD         A,#30H
            CALL        SER_OUT
            POP         ACC
            RET

;*********************************************************************
GETCHAR:    LCALL       SER_IN
            ANL         A,#7FH
            RET

;*********************************************************************
PUT_CHAR:
            CJNE        A,#LF,LABL1
            MOV         A,#CR
            LCALL       SER_OUT
            MOV         A,#LF

LABL1:      LCALL       SER_OUT
            RET
```

```
;*********************************************************************
HEXASC:     ANL         A,#0FH
            ADD         A,#-10
            JC          HXASC1
            ADD         A,#(10+'0')
            RET

HXASC1:     ADD         A,#'A'
            RET

;*********************************************************************
OUT_BYTE:
            PUSH        ACC
            SWAP        A
            LCALL       OUT_HEX
            POP         ACC
            LCALL       OUT_HEX
            RET

OUT_HEX:    LCALL       HEXASC
            LCALL       PUT_CHAR
            RET

;*********************************************************************
PUT_IN:     CLR         A
            MOVC        A,@A+DPTR
            JZ          PULN1
            LCALL       PUT_CHAR
            INC         DPTR
            SJMP        PUT_IN
PULN1:      RET

;*********************************************************************
SER_OUT:    JNB         TI,SER_OUT
```

```
                CLR         TI
                MOV         SBUF,A
                RET

;*********************************************************************
SER_IN:         JNB         RI,SER_IN
                CLR         RI
                MOV         A,SBUF
                RET

;*********************************************************************
OUT_SPC:        MOV         A,#SPACE
                LCALL       PUT_CHAR
                RET

;*********************************************************************

OUT_STAR:
                MOV         A,#'*'
                LCALL       PUT_CHAR
                RET

;*********************************************************************
GET_ADDR:
                LCALL       BYTE_IN1
                MOV         DPH,A
                LCALL       BYTE_IN1
                MOV         DPL,A
                RET

;*********************************************************************

DELAY:          PUSH        ACC
                MOV         R6,#50H
```

```
                MOV         R5,#00H

LOOP1:          DJNZ        R5,LOOP1
                DJNZ        R6,LOOP1
                POP         ACC
                RET

;*****************************************************************

MSG1:DB     LF,LF,'PROGRAM MONITOR : BY MIOR MOHAMMAD
HAFIIZH BIN ABD. RANI, SEM 7'
        DB      LF,'LAST UPDATE AT OCTOBER, 28, 2007',LF
        DB      LF,'MENU : L-LOADING <PGUP>'
        DB      LF,'     : E-EXECUTION <ADDRESS>'
        DB      LF,'     : D-DATA MEMORY DISPLAY'
        DB      LF,'     : I-INTERNAL MEMORY DISPLAY',LF,0
MSG2:DB     LF,LF,'EXECUTION ADDRESS :',0
MSG3:DB     'UMP>>',0
MSG4:DB     LF,'LOADING IN PROGRESS, PLEASE WAIT!!....'
        DB      LF,'<CTRL Z> TO EXIT!',LF,0
MSG5:DB     LF,'DONE!',LF,0
MSG6:DB     LF,'TERMINATED BY USER! CAUTION!',LF,0
MSG7:DB     LF,LF,'DATA MEMORY DISPLAY',0
MSG8:DB     LF,LF,'ADDRESS : ',0
MSG9:DB     LF,LF,'INTERNAL MEMORY DISPLAY',0
MSG10:      DB      LF,LF,'ADDRESS : ',0

                END
```

APPENDIX B

Program for 7-segment display (Phase 2)

```
        $MOD51
        ORG   8000H

        MOV   A,#0
PLTEST: MOV   P1,A
AGAIN:  MOV   DPTR,#TABLE
NEXT:   CLR   A
        MOVC  A,@A+DPTR
        JZ    AGAIN
        CPL   A
        MOV   P1,A
        INC   DPTR
DELAY:  MOV   R2,#8
INLP1:  MOV   R1,#255
INLP2:  MOV   R0,#255
INLP3:  DJNZ  R0,INLP3
        DJNZ  R1,INLP2
        DJNZ  R2,INLP1
        SJMP  NEXT

TABLE:  DB    0FFH,86H,83H,0C0H,0B0H,0C0H,0B0H,0A4H
        DB    0FFH,86H,83H,0C0H,0B0H,0C0H,99H,82H,00H

        END
```

APPENDIX C

Program for the security system (Phase 3)

```
            $MOD51

KEYPAD      EQU   4000H

            ORG   8000H

MULA:       ;SETB IE.7
            ;SETB IE.0
            ;SETB IP.0

SCAN1:      JB    P1.5,SCAN1
            CLR   P1.2
DELAY9:     MOV   R2,#8
INLP151:    MOV   R1,#100
INLP251:    MOV   R0,#100
INLP351:    DJNZ  R0,INLP351
            DJNZ  R1,INLP251
            DJNZ  R2,INLP151
            SETB  P1.6
            SETB  P1.0
            SETB  P1.1
            SETB  P1.2
            CLR   P1.3
            SETB  P1.5
            CLR   P1.4
            MOV   R4,#00H

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

```
UJI:    MOV  DPTR,#KEYPAD
        MOVXA,@DPTR
        CJNE  A,#10H,DUA  ;FIRST CORRECT CODE
        MOV  A,#01H
        LJMP  MASUK1
DUA:    CJNE  A,#14H,TIGA
        MOV  R4,#02H
        LJMP  BUNYI
TIGA:   CJNE  A,#18H,EMPAT
        MOV  R4,#03H
        SJMP  BUNYI
EMPAT:  CJNE  A,#11H,LIMA
        MOV  R4,#04H
        SJMP  BUNYI
LIMA:   CJNE  A,#15H,ENAM      ;FOURTH CORRECT CODE
        MOV  A,#05H
        LJMP  MASUK4
ENAM:   CJNE  A,#19H,TUJUH
        MOV  R4,#06H
        SJMP  BUNYI
TUJUH:  CJNE  A,#12H,LAPAN
        MOV  R4,#07H
        SJMP  BUNYI
LAPAN:  CJNE  A,#16H,SMBLN     ;THIRD CORRECT CODE
        MOV  A,#08H
        LJMP  MASUK3
SMBLN:  CJNE  A,#1AH,KOSONG    ;SECOND CORRECT CODE
        MOV  A,#09H
        LJMP  MASUK2
KOSONG: CJNE  A,#17H,KEYA
        MOV  R4,#10H
        SJMP  BUNYI
KEYA:   CJNE  A,#1CH,STAR
        LJMP  BACA
```

```
STAR:      CJNE  A,#13H,HTRIK
           MOV   R4,#10H
           SJMP  BUNYI
HTRIK:     CJNE  A,#1BH,KEYB
           MOV   R4,#10H
           SJMP  BUNYI
KEYB:      CJNE  A,#1DH,KEYC
           MOV   R4,#10H
           SJMP  BUNYI
KEYC:      CJNE  A,#1EH,KEYD
           MOV   R4,#10H
           SJMP  BUNYI
KEYD:      CJNE  A,#1FH,UJI
           MOV   R4,#10H
           SJMP  BUNYI

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

MASUK1:    MOV   DPTR,#9701H
           MOVX@DPTR,A
           SJMP  BUNYI
MASUK2:    MOV   DPTR,#9702H
           MOVX@DPTR,A
           SJMP  BUNYI
MASUK3:    MOV   DPTR,#9703H
           MOVX@DPTR,A
           SJMP  BUNYI
MASUK4:    MOV   DPTR,#9704H
           MOVX@DPTR,A
           SJMP  BUNYI

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

BUNYI:     SETB  P1.3
```

```
DELAY:      MOV   R2,#8
INLP1:      MOV   R1,#100
INLP2:      MOV   R0,#100
INLP3:      DJNZ  R0,INLP3
            DJNZ  R1,INLP2
            DJNZ  R2,INLP1
            CLR   P1.3
            LJMP  UJI

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

BACA:       SETB  P1.3
DELAY2:     MOV   R2,#8
INLP12:     MOV   R1,#100
INLP22:     MOV   R0,#100
INLP32:     DJNZ  R0,INLP32
            DJNZ  R1,INLP22
            DJNZ  R2,INLP12
            CLR   P1.3

            CLR   A
            MOV   DPTR,#9701H
            MOVXA,@DPTR
            CJNE  A,#01H,ALERT3
            CLR   A
            MOVX@DPTR,A
            CLR   A
            MOV   DPTR,#9702H
            MOVXA,@DPTR
            CJNE  A,#09H,ALERT3
            CLR   A
            MOVX@DPTR,A
            CLR   A
            MOV   DPTR,#9703H
```

```
            MOVXA,@DPTR
            CJNE  A,#08H,ALERT3
            CLR   A
            MOVX@DPTR,A
            CLR   A
            MOV   DPTR,#9704H
            MOVXA,@DPTR
            CJNE  A,#05H,ALERT3

            CLR   A
            MOVX@DPTR,A
            CLR   A
            MOV   A,R4
            CJNE  A,#00H,ALERT3
            CLR   A
            MOV   R4,A
            CLR   P1.0
            SETB  P1.6
            SJMP  NEXT

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

NEXT:       CLR   P1.0
DELAY5:     MOV   R2,#8
INLP15:     MOV   R1,#100
INLP25:     MOV   R0,#100
INLP35:     DJNZ  R0,INLP35
            DJNZ  R1,INLP25
            DJNZ  R2,INLP15
            LJMP  MULA

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

ALERT3:     SETB  P1.4
```

```
        CLR   P1.1
        MOV   R2,#8
INLP94: MOV   R1,#100
INLP95: MOV   R0,#100
INLP96: DJNZ  R0,INLP96
        DJNZ  R1,INLP95
        DJNZ  R2,INLP94
        SETB  P1.1
        CLR   P1.4
SCAN13: JNB   P1.5,SCAN13
        CLR   P1.6
        SJMP  NEXT1

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

NEXT1:  MOV   R4,#00H
UJI1:   MOV   DPTR,#KEYPAD
        MOVXA,@DPTR
        CJNE  A,#10H,DUA2       ;FIRST CORRECT CODE
        MOV   A,#01H
        LJMP  MASUK12
DUA2:   CJNE  A,#14H,TIGA2
        MOV   R4,#02H
        LJMP  BUNYI2
TIGA2:  CJNE  A,#18H,EMPAT2
        MOV   R4,#03H
        SJMP  BUNYI2
EMPAT2: CJNE  A,#11H,LIMA2
        MOV   R4,#04H
        SJMP  BUNYI2
LIMA2:  CJNE  A,#15H,ENAM2      ;FOURTH CORRECT CODE
        MOV   A,#05H
        LJMP  MASUK42
ENAM2:  CJNE  A,#19H,TUJUH2
```

```
            MOV   R4,#06H
            SJMP  BUNYI2
TUJUH2:     CJNE  A,#12H,LAPAN2
            MOV   R4,#07H
            SJMP  BUNYI2
LAPAN2:     CJNE  A,#16H,SMBLN2      ;THIRD CORRECT CODE
            MOV   A,#08H
            LJMP  MASUK32
SMBLN2:     CJNE  A,#1AH,KOSONG2     ;SECOND CORRECT CODE
            MOV   A,#09H
            LJMP  MASUK22
KOSONG2:    CJNE  A,#17H,KEYA2
            MOV   R4,#10H
            SJMP  BUNYI2
KEYA2:      CJNE  A,#1CH,STAR2
            LJMP  BACA2
STAR2:      CJNE  A,#13H,HTRIK2
            MOV   R4,#10H
            SJMP  BUNYI2
HTRIK2:     CJNE  A,#1BH,KEYB2
            MOV   R4,#10H
            SJMP  BUNYI2
KEYB2:      CJNE  A,#1DH,KEYC2
            MOV   R4,#10H
            SJMP  BUNYI2
KEYC2:      CJNE  A,#1EH,KEYD2
            MOV   R4,#10H
            SJMP  BUNYI2
KEYD2:      CJNE  A,#1CH,UJI1
            MOV   R4,#10H
            SJMP  BUNYI2

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

```
MASUK12:    MOV   DPTR,#9701H
            MOVX@DPTR,A
            SJMP  BUNYI2
MASUK22:    MOV   DPTR,#9702H
            MOVX@DPTR,A
            SJMP  BUNYI2
MASUK32:    MOV   DPTR,#9703H
            MOVX@DPTR,A
            SJMP  BUNYI2
MASUK42:    MOV   DPTR,#9704H
            MOVX@DPTR,A
            SJMP  BUNYI2


;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

BUNYI2:     SETB  P1.3
DELAY7:     MOV   R2,#8
INLP14:     MOV   R1,#100
INLP24:     MOV   R0,#100
INLP34:     DJNZ  R0,INLP34
            DJNZ  R1,INLP24
            DJNZ  R2,INLP14
            CLR   P1.3
            LJMP  UJI1

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;


BACA2:      SETB  P1.3
DELAY8:     MOV   R2,#8
INLP13:     MOV   R1,#100
INLP23:     MOV   R0,#100
INLP33:     DJNZ  R0,INLP33
```

```
DJNZ  R1,INLP23
DJNZ  R2,INLP13
CLR   P1.3

CLR   A
MOV   DPTR,#9701H
MOVXA,@DPTR
CJNE  A,#01H,ALERT2
CLR   A
MOVX@DPTR,A
CLR   A
MOV   DPTR,#9702H
MOVXA,@DPTR
CJNE  A,#09H,ALERT2
CLR   A
MOVX@DPTR,A
CLR   A
MOV   DPTR,#9703H
MOVXA,@DPTR
CJNE  A,#08H,ALERT2
CLR   A
MOVX@DPTR,A
CLR   A
MOV   DPTR,#9704H
MOVXA,@DPTR
CJNE  A,#05H,ALERT2

CLR   A
MOVX@DPTR,A
CLR   A
MOV   A,R4
CJNE  A,#00H,ALERT2
CLR   A
MOV   R4,A
```

```
                SJMP    NEXT2

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

NEXT2:          SETB    P1.6
                CLR     P1.0
DELAY81:        MOV     R2,#8
INLP751:        MOV     R1,#100
INLP651:        MOV     R0,#100
INLP551:        DJNZ    R0,INLP551
                DJNZ    R1,INLP651
                DJNZ    R2,INLP751
                SETB    P1.0
                LJMP    MULA

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

ALERT2:         CLR     P1.2
DELAY6:         MOV     R2,#8
INLP19:         MOV     R1,#100
INLP29:         MOV     R0,#100
INLP39:         DJNZ    R0,INLP39
                DJNZ    R1,INLP29
                DJNZ    R2,INLP19
                SETB    P1.2
                LJMP    UJI

                END
```

APPENDIX D

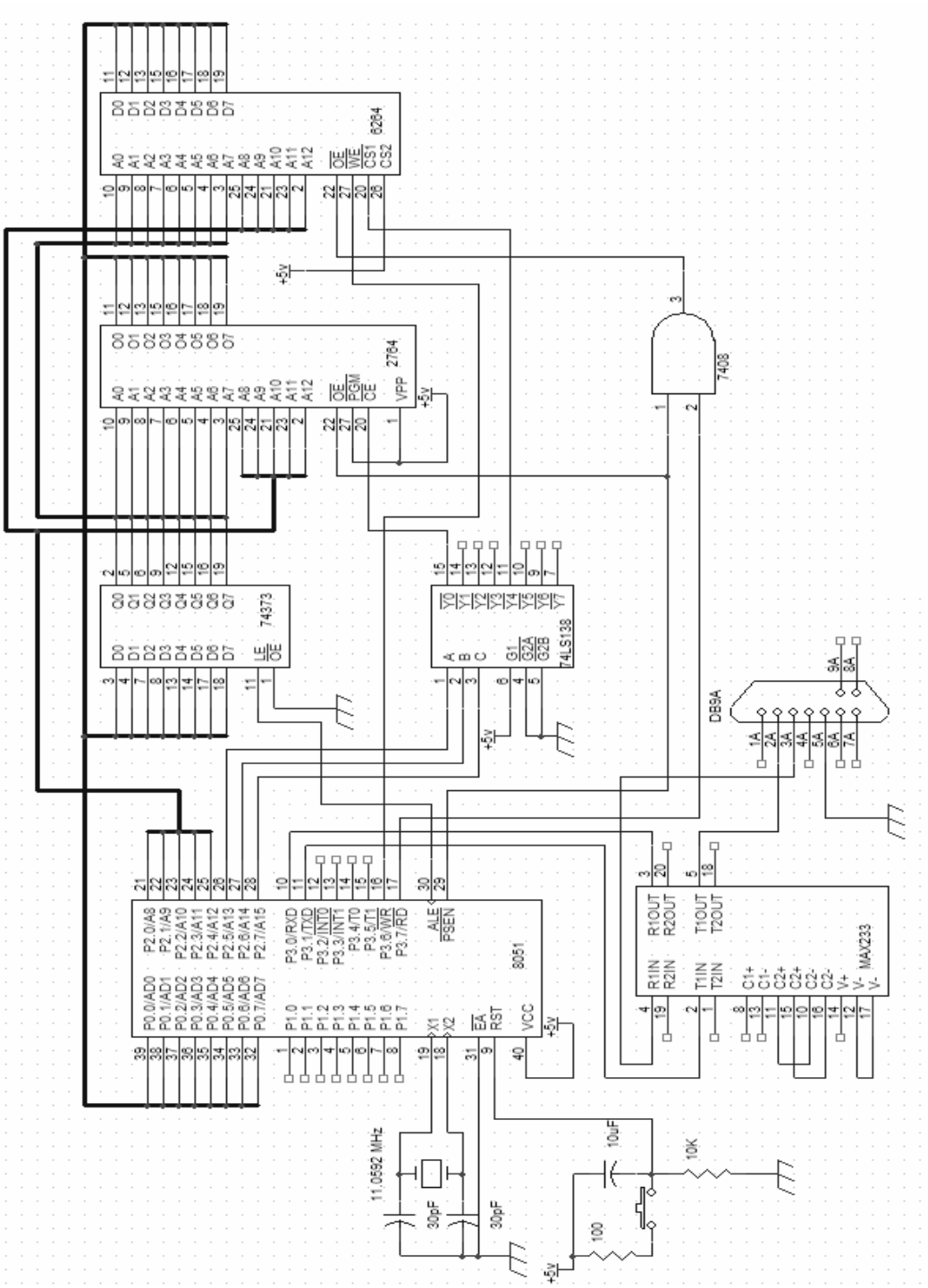
Program for encoding keypad module

```
                $MOD51

KEYPAD          EQU     4000H

                ORG     8000H

MULA:           ;SETB IE.7
                ;SETB IE.0
                ;SETB IP.0

                CLR     P1.0            ;INITIALIZE PORT
                CLR     P1.1
                CLR     P1.2
                CLR     P1.3
                SETB    P1.4
                MOV     R4,#00H

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

UJI:            MOV     DPTR,#KEYPAD
                MOVXA,@DPTR
                CJNE    A,#1FH,UJI      ;CORRECT CODE

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

BUNYI:          SETB    P1.3
DELAY:          MOV     R2,#8
INLP1:          MOV     R1,#100
INLP2:          MOV     R0,#100
```

```
INLP3:  DJNZ  R0,INLP3
        DJNZ  R1,INLP2
        DJNZ  R2,INLP1
        CLR   P1.3
        LJMP  UJI

        END
```

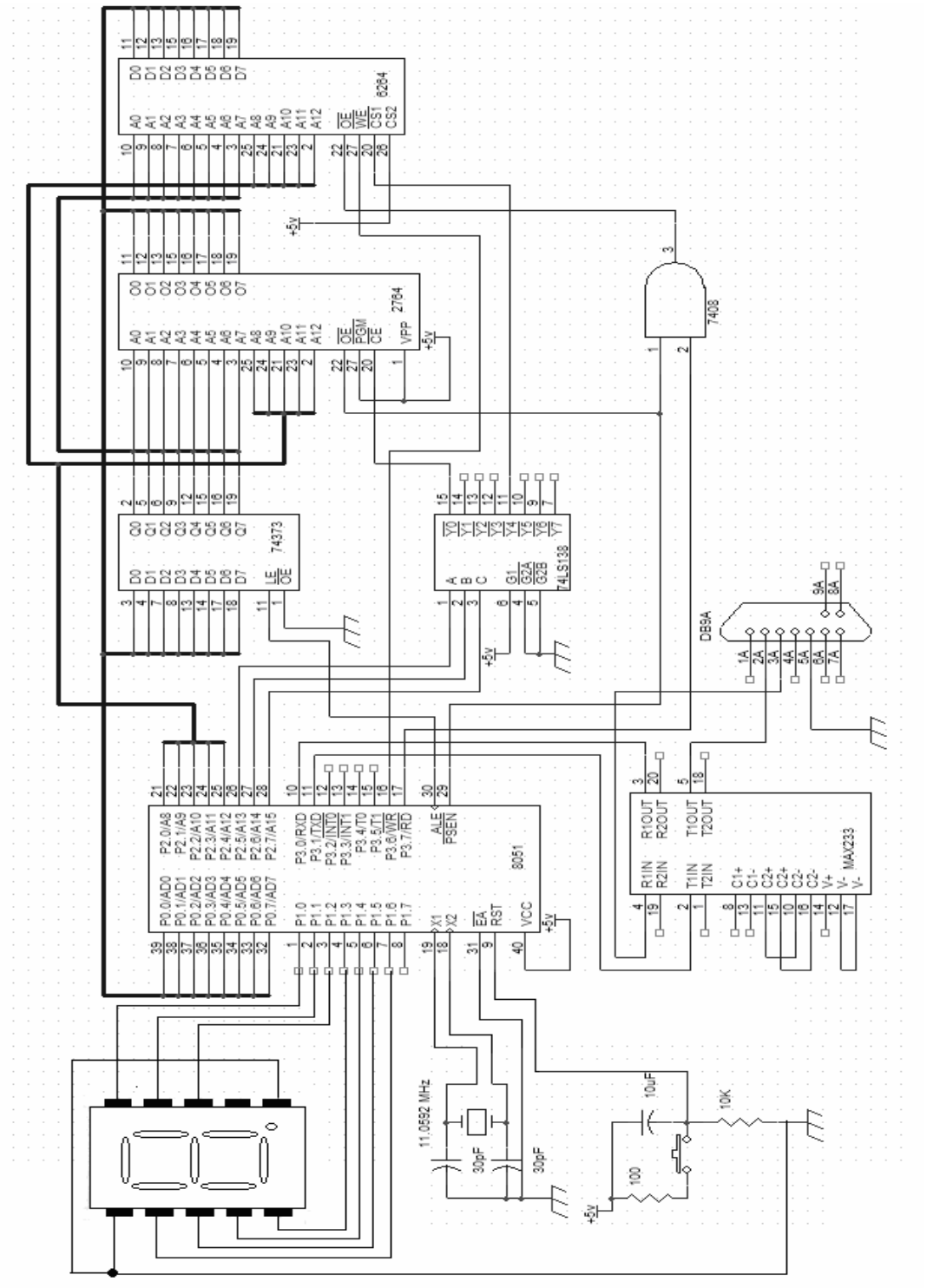
APPENDIX E

Phase 1 (Simple serial communication)



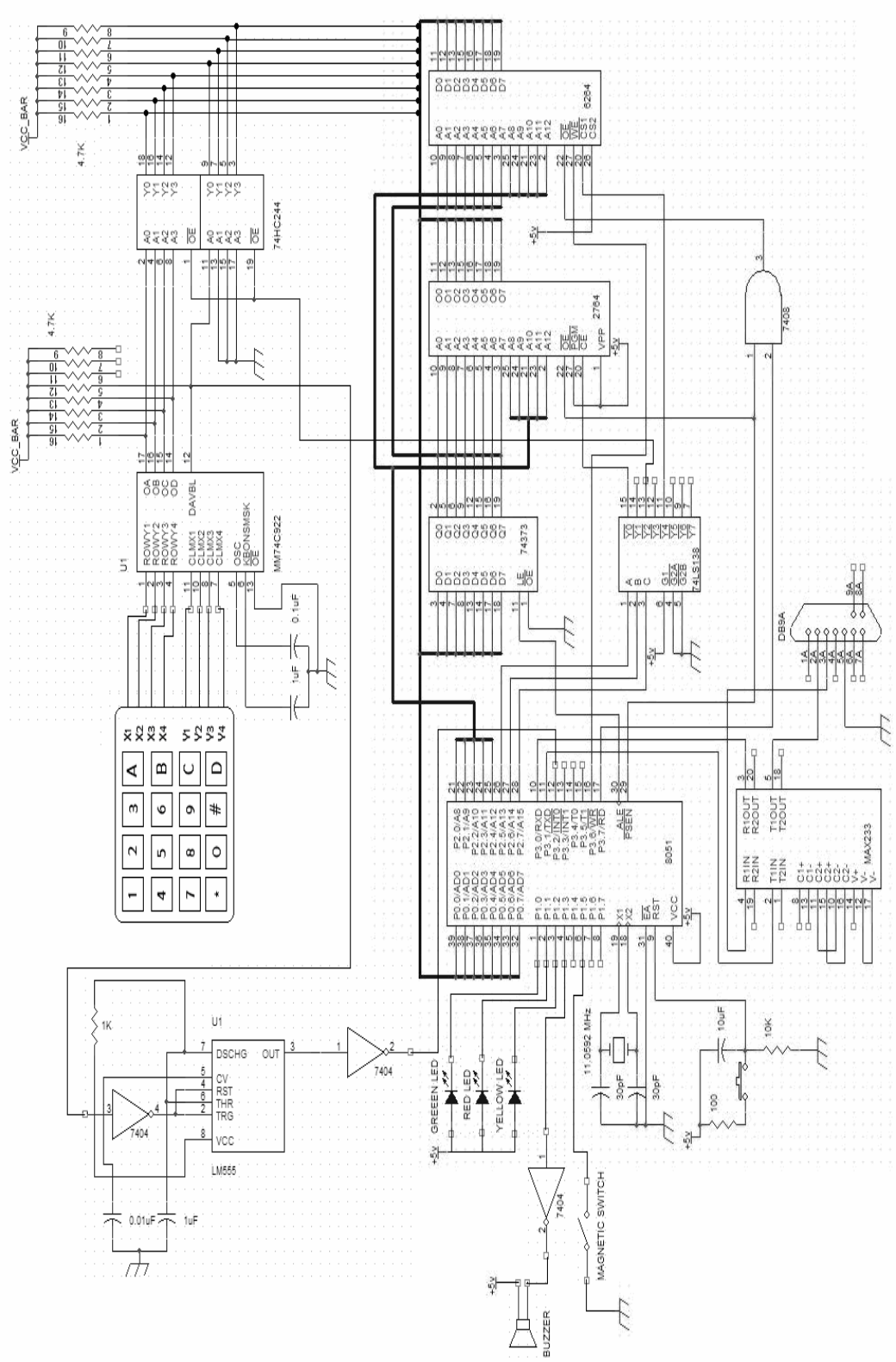
APPENDIX F

Phase 2 (Simple input/output interface – 7 segment display)



APPENDIX G

Phase 3 (The security system)



APPENDIX H

User Manual for the security system

1. A program based on security design circuit is written.
2. It is then saved in <filename>.asm in the ASM51 folder. In this ASM51 folder consist of ASM51 assembler with address mod which is important in assembling program based on 8051 controller.
3. Click on the command prompt.
4. Identify the location of ASM51 assembler – command “CD ASM51” is used to access the folder. As an example in Figure A, the ASM51 folder is located in Local Disk (C:) > Documents and Settings folder > Admin folder. The ASM51 folder which is in Admin can be access using command “CD ASM51”.

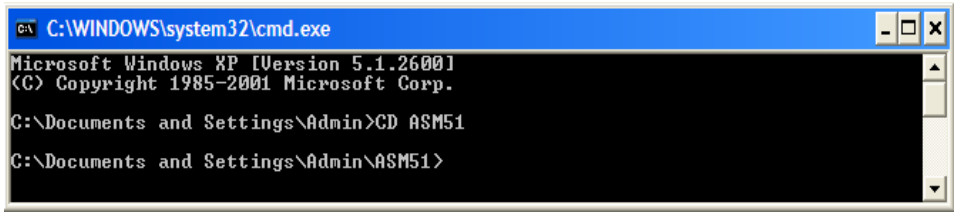


Figure A

5. Then assemble the <filename>.asm using command “ASM 51 <filename>.asm”.
6. Program succeeds assembled with no error as shown in Figure B. If the <filename>.asm succeeds assembled, two files in notepad format will produced in ASM51 folder. The files are <filename>.HEX and <filename>.LST. The <filename>.HEX is the binary code that will be used in loading process.

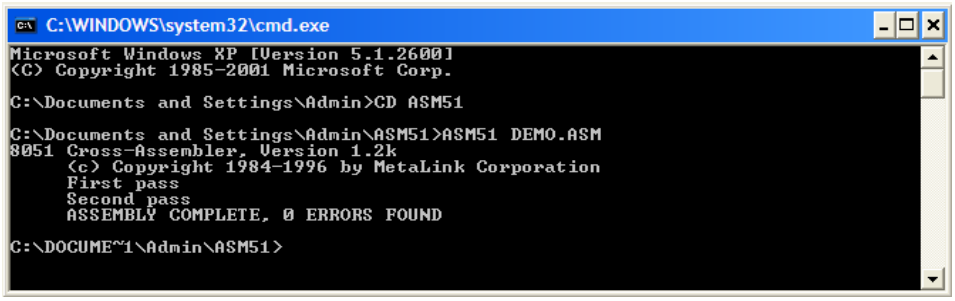


Figure B

7. Then connect the hardware board with 5V dc supply.
8. Connect serial wire (DB9 female) from the hardware board to the DB9 male of the personal computer (PC).
9. Click at HyperTerminal software.
10. The type any name/ title as shown in Figure C.

Figure C

11. Click OK.
12. Next, make sure setting as shown in Figure D.

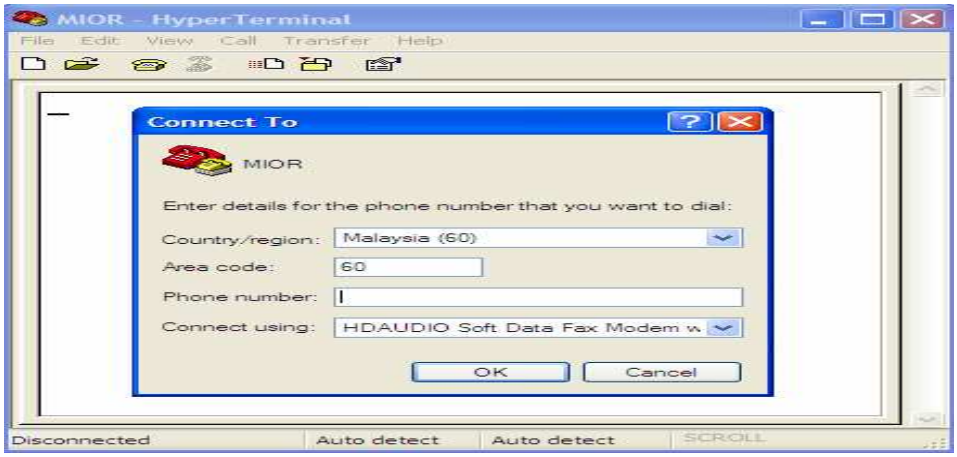


Figure D

13. Choose port – example COM5 (any COM related to port that DB9's connected to) Figure E.

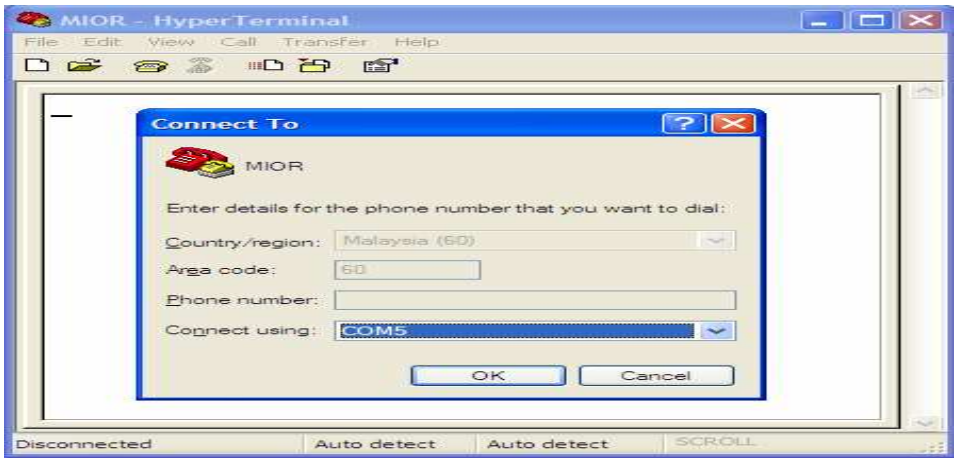


Figure E

14. Click OK.
15. The selected COM properties must be set as Figure F.

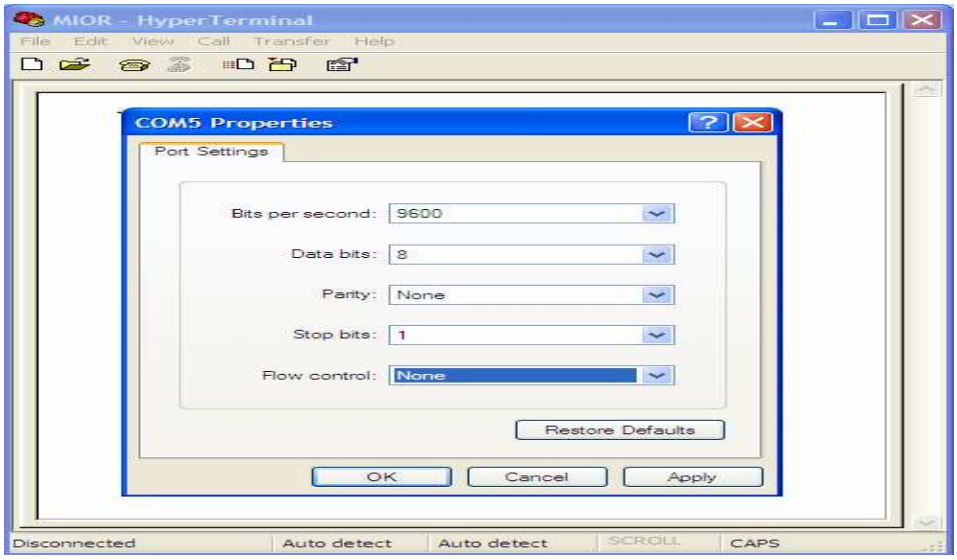


Figure F

16. Click OK.
17. After that, press reset button on hardware board.
18. Monitoring Program as shown in Figure G will be viewed.

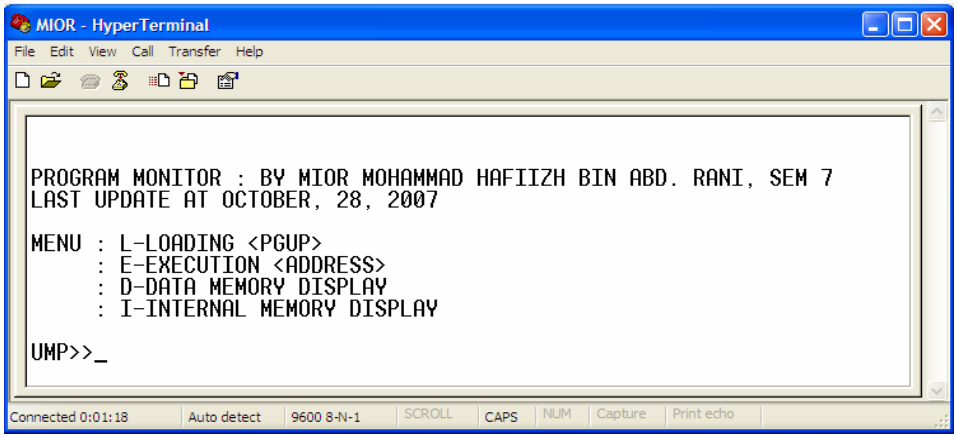


Figure G

19. Then press ‘L’ thus Figure H will be viewed for loading process.

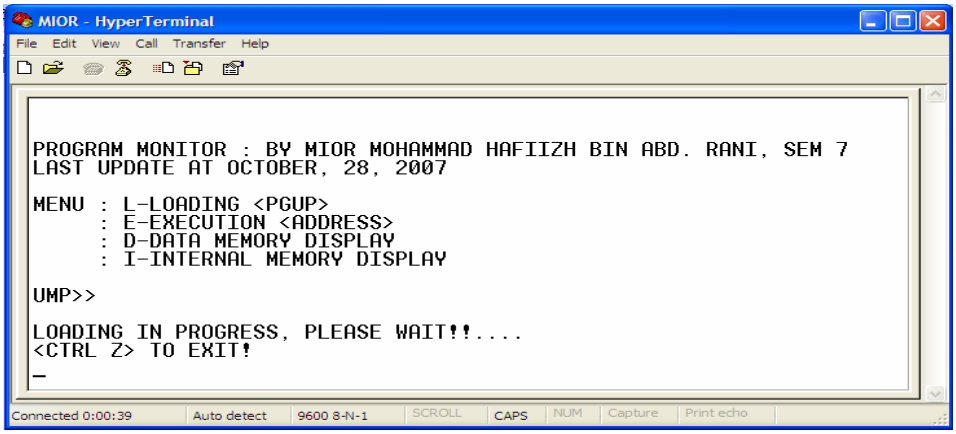


Figure H

20. To start load the <filename>.HEX, click to “Transfer and Send Text File…” as shown in Figure I.

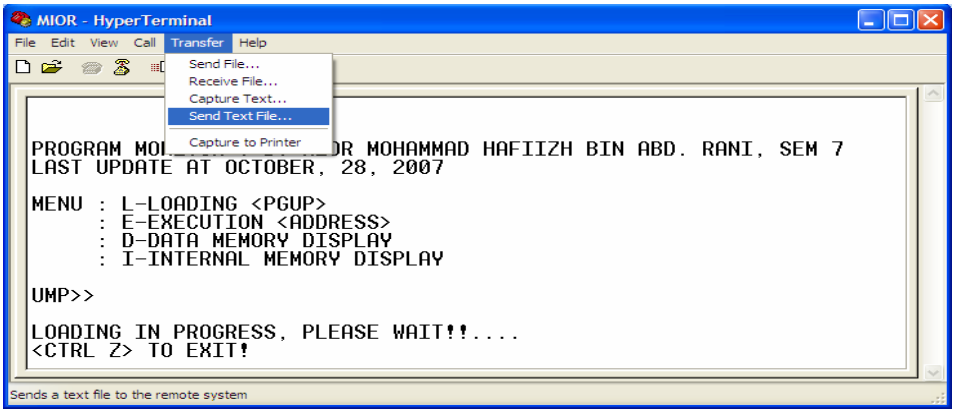


Figure I

21. New window will viewed as Figure J. Choose the <filename>.HEX then click ‘Open’.

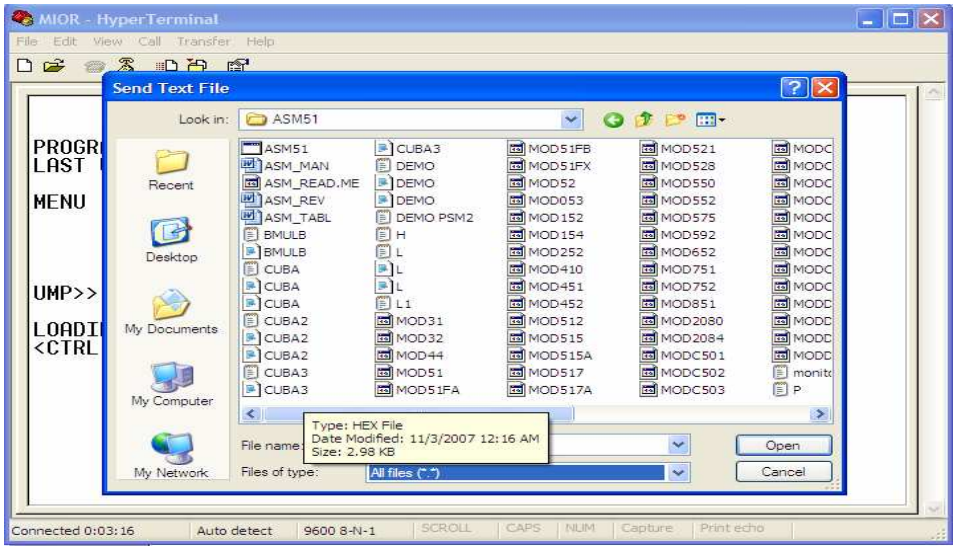


Figure J

22. Figure K will be viewed means loading the <filename>.HEX to the address of RAM succeed.

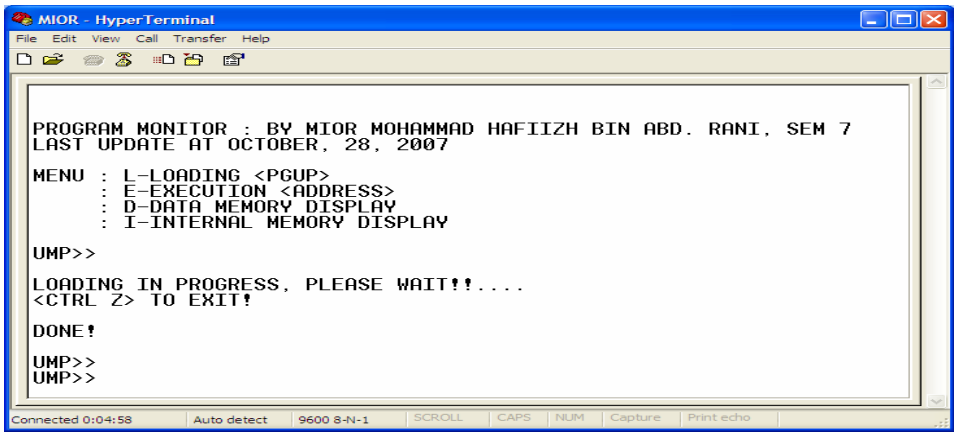


Figure K

23. Then press ‘E’ for the execution address. The execution address is based on address of program wrote in <filename>.asm.
24. As an example, in this security system, the 8000 is the execution address as shown in Figure L.

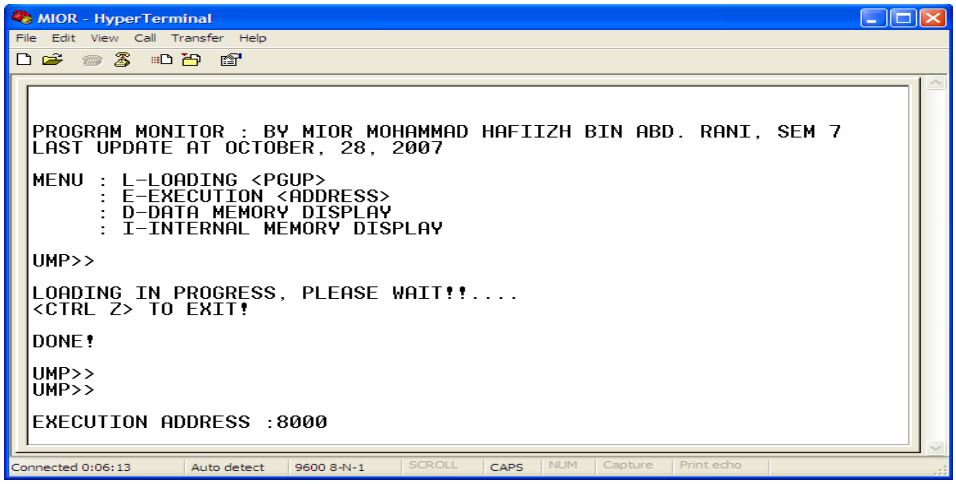


Figure L

25. Reconnect DB9 wire from hardware board.
26. The security system is now ready to be use!
27. Below are the flowchart (Figure M) and the description of the security system based on program written.

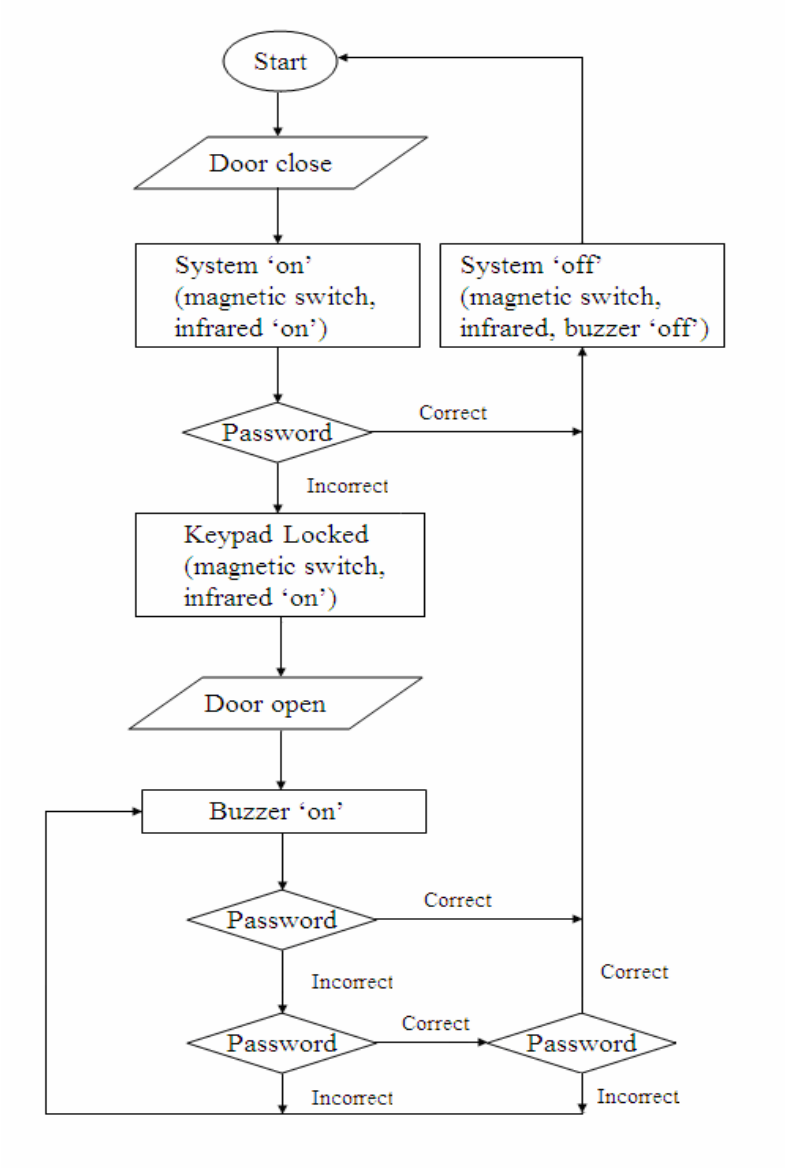


Figure M Flow chart of software design for security system

The security system description

The flowchart as shown in Figure M above is a basic designed for the security system operation. In this security system, the security code is set as '1985' to activate or deactivate security system. When user closed the door, the system will activate where the magnetic switch and infrared is in active condition.

To deactivate the system where opening door without alarm, user must enter the deactivation code. The chance to deactivate the system is once before keypad locked, where user can not enter the code anymore. If deactivation code was correct, green LED will ‘on’ while alarm will ‘off’ condition. System at this time is successfully deactivated and user can open the door without alarm.

If deactivation code enters was incorrect, system is still activate and for that, red LED and alarm will ‘on’ where the system is in a warning mode. The system will keep in this mode until the correct deactivation code entered.

When the correct deactivation codes enter, the reconfirm code is needed where user need to reenter the correct deactivation code before the system successfully deactivate; the green LED will ‘on’ and the alarm will ‘off’ condition.

The reconfirm code is designed in the system to make more security for the system developed. If intruder open the magnetic switch without entered the correct security code, thus the alarm will ‘on’. User must enter the correct deactivation code to turn ‘off’ the alarm system. The activation and deactivate code for this system are ‘1985’ and the ‘enter’ button codes are ‘A’ for every codes entered.