DESIGN AND IMPLEMENTATION OF SECURITY SYSTEM



SONARGAON UNIVERSITY (SU)

Supervised By

Md. Ferdous Khan
Course Coordinator
Department of EEE
Sonargaon University (SU)

Submitted By

Name	ID
Jannaty Akter	**EEE 1901016072**
Md. Robin Mia	**EEE 1901016197**
Md Sohel Rana	**EEE 1901016179**
Md. Monayem Hossen	**EEE 1901016099**
Shuvjit Mondal	**EEE 1901016129**
Rabby Hasan	**EEE 1901016196**

Department of Electrical & Electronic Engineering (EEE)
Sonargaon University (SU)
147/I, Panthopath,
Dhaka-1215, Bangladesh.

Date of submission: September 2022

Declaration

It is declared hereby that this project paper or any part of it has not been submitted to anywhere else for the award of any degree.

……	……..
Jannaty Akter	**Shuvjit Mondal**
……	…….
Md. Robin Mia	**Md Sohel Rana**
……	……
Md. Monayem Hossen	**Rabby Hasan**

Under Supervision of

Md. Ferdous Khan
Course Coordinator, EEE
Sonargaon University

Certification

I certify that I have read this project and that, in my opinion, it is fully adequate, in scope & quality as a dissertation for the degree of BSc in Engineering.

Supervisor

.....

Md. Ferdous Khan

Course Coordinator, EEE

Sonargaon University (SU)

Department of Electrical and Electronic Engineering (EEE)

ACKNOWLEDGEMENTS

First and foremost, we would like to thank Almighty Allah for granting our capability and providing me health and an opportunity to finish this work.

We offer our sincerest gratitude to our supervisor, **Md. Ferdous Khan**, Course Coordinator, Department of Electrical & Electronic Engineering Sonargaon University for the patient guidance, encouragement and advice he has provided throughout my time as his student. We consider ourselves very fortunate for being able to work with a very considerate and encouraging lecturer like him.

We would also like to express our thanks to the respected **Vice-Chancellor of Su, Professor Dr. Md. Abul Bashar** for providing me the opportunity to conduct our research in such an academic environment.

Above all, we wish to convey our heartfelt thanks and love towards our parents to whom we owe everything. We are forever indebted to our parents for their understanding, endless support and encouragement when it was most required.

Abstract

Conventional security and safety systems which are the common form of protection to lives and properties, have certain limitations such as real time monitoring and control of activities such as intruders in the form of human beings, motion, smoke, etc. These limitations in most cases result in high financial loss to properties and lives. This work involves design and construction of GSM intelligent home security system for real time monitoring of intruders. It consist of intrusion detection sensors (Motion, Water, Vibration, Door and Smoke), programmable microcontroller, regulated power supply unit, GSM modem, data acquisition node and an interface program development. The moment the alarm signal is received, it will send short message to the users (owners of the system) through GSM network immediately.

In this system we implement a complete solution for home security and safety by the help of open source hardware and platform. In this project we use arduino Nano as the controller of the project and Motion, Water, Vibration, Door and Smoke sensor to protect our home. There also has system lock and unlock feature by sms, home status also can be monitor using LCD and indicator.

TABLE OF CONTENTS

Declaration		ii
Certification		iii
Acknowledgement		iv
Abstract		v
CHAPTER 1	**INTRODUCTION**	
1.1	Introduction	1
1.2	Background	1
1.3	Objective	2
1.4	Summary	3
CHAPTER 2	**HARDWARE COMPONENT**	
2.1	Introduction	4
2.2	Hardware Required	4
2.3	Arduino Nano	4
2.2.1	Technical specifications	5
2.2.2	Microcontroller ATMEGA328P	5
2.2.3	Power supply, inputs and outputs	8
2.2.4	Digital inputs and outputs	9
2.4	Smoke Sensor	9
2.5	Door Sensor	10
2.6	Water Sensor	11
2.7	Vibration Sensor	12
2.8	Motion Sensor	12
2.9	GSM Module	13
2.10	LCD Display	15
2.11	Diode	16

2.12	Buck Converter Module	16
2.13	Buzzer	17
2.14	Jumper Wire	17
2.15	Software Configuration	18
CHAPTER 3	**DESIGN AND IMPLEMENTATION**	
3.1	Introduction	19
3.2	Block Diagram	19
3.3	Methodology	19
3.4	Circuit Diagram	20
3.5	Summarization	20
CHAPTER 4	**RESULT AND DISCUSSION**	
4.1	Introduction	21
4.2	Result	21
4.3	Power Consumption	25
4.4	Discussion	25
CHAPTER 5	**CONCLUSION AND FUTURE WORK**	
5.1	Conclusion	26
5.2	Future Recommendation	26
References		27
Appendix		28

List of Figures

Figure NO	Figure Name	Page Number
Fig 2.1	Arduino Nano	5
Fig 2.2	ATmega328 Microcontroller Architecture	7
Fig 2.3	Block diagram of the AVR CPU Core architecture	7
Fig 2.4	Microcontroller IC ATmega 328p	8
Fig 2.5	Smoke Sensor Module	10
Fig 2.6	Door Sensor	11
Fig 2.7	Water Sensor	11
Fig 2.8	Vibration Sensor	12
Fig 2.9	Motion Sensor	13
Fig 2.10	SIM800L GSM Module	14
Fig 2.11	Circuit Diagram for Arduino and GSM module Connection	15
Fig 2.12	LCD Display	15
Fig 2.13	Diode symbol	16
Fig 2.14	DC-DC Buck Converter	16
Fig 2.15	Buzzer	17
Fig 2.16	Jumper Wire	17
Fig 2.17	Uploading program to Arduino	18
Fig 2.18	Message for upload complete	18
Fig 3.1	Block Diagram of the System	19
Fig 3.2	Circuit Diagram of the system	20
Fig 4.1	Vibration Alarm	21
Fig 4.2	Water Alarm	21
Fig 4.3	Door Alarm	22
Fig 4.4	Motion Alarm	22
Fig 4.5	Smoke Alarm	22
Fig 4.6	Project Picture	23

CHAPTER 1

INTRODUCTION

1.1 Introduction

Home security and safety has changed a lot from the last century and will be changing in coming years. Security is an important aspect or feature in the smart home applications. The new and emerging concept of smart homes offers a comfortable, convenient, and safe environment for occupants. Conventional security systems keep homeowners, and their property, safe from intruders by giving the indication in terms of short message. However, a smart home security system offers many more benefits**.**

The main concept behind the work is receiving sent SMS and processing it further as required to perform several operations. Intelligent homes in simple terms can be described as homes that are fully automated in terms of carrying out a predetermined task, providing feed back to the home users and responding accordingly to situations. Intelligent home security systems such as controlled network, and communication systems, emergency response, anti-theft monitoring system safety system and automation system requires automated and controlled system both near and at a distance of control. This paper is aimed at designing a GSM based home security and safety for detecting an intrusion into a monitored area by a motion sensor, smoke sensor, water sensor, vibration sensor and gsm module. For home safety, intrusion detector has a transmitter coupled with portable receiver to alert home owners through SMS in situations of break-ins or entering into the home using force while the owners are out of their home then the device sends SMS to the emergency number which is provided to the system. The system is made up of components: sensors, GSM modem, and microcontroller to control the device.

1.2 Background

The new age of technology has redefined communication. Most people nowadays have access to mobile phones and thus the world indeed has become a global village. At any given moment, any particular individual can be contacted with the mobile phone. But the application of mobile phone cannot just be restricted to sending SMS or starting conversations. New innovations and ideas can be generated from it that can further enhance

its capabilities. Technologies such as Infra-red, Bluetooth, etc. which has developed in recent years goes to show the very fact that improvements are in fact possible and these improvements have eased our life and the way we live. Sound, heat, obstacle, detection and touch sensing security system is a subject of growing interest and in recent years we have seen many systems providing such security system, automation system and protection system these days, apart from supporting voice calls a mobile phone can be used to send text messages as well as multimedia messages (that may contain pictures, graphics, animations, etc.). Sending written text messages is very popular among mobile phone users. Instant messaging, as it is also known, allows quick transmission of short messages that allow an individual to share ideas, opinions and other relevant information. We have used the very concept to design a system that acts a platform to receive messages which in fact are commands sent to control different appliances and devices connected to the platform. We have designed a complete safety and automation system which is based on the GSM technology that effectively allows control of different load and sensor concerned with different factors. The application of our suggested system is immense in the ever changing technological world.

1.3 Objective

The aim of the project is to develop and launch on up to date, reliable and user friendly security system to automate home security using microcontroller circuitry synchronized with GSM module with an objective to provide maximum possible security based on an automatic emergency care response using detection system.

General objectives of the project are defined as;

- Protect asset from Fire violence.
- Protect from unauthorized entry.
- Vibration alarm.
- Motion alarm.
- To effectively receive and transmit data via SMS.
- To ask Emergency support using sms.
- To offers a comfortable, convenient and safe environment for residents.

1.4 Summary

The system is design for ensure proper safety, security and automation system by using arduino nano microcontroller development board. We will use Motion sensor, Water sensor, Vibration sensor, Door sensor and Smoke sensor to protect our home from unauthorized entry. This system will inform the owner if there was any Motion, Water, Vibration, Door action and Smoke detected.

CHAPTER 2

HARDWARE COMPONENT

2.1 Introduction

Hardware is the most important part of any project. In this chapter, we explain used component of our project.

2.2 Hardware Required

The hardware required is divided in the following category:

- Arduino Nano
- Smoke Sensor
- Motion Sensor
- Vibration sensor
- Door Sensor
- Water Sensor
- LCD Display
- Push Button
- Buck Converter

2.3 Arduino Nano

Arduino is an open-source prototyping platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. it can tell that board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the Arduino programming language and the Arduino Software (IDE), based on Processing.



Fig 2.1: Arduino Nano

2.2.1 Technical specifications:

Table 2.1: Technical specifications of Arduino Nano

Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	8
DC Current per I/O Pin	20 mA
DC Current for 2.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P)of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz

2.2.2 Microcontroller ATMEGA328P

The Atmel® picoPower® ATmega328/P is a low-power CMOS 8-bit microcontroller based on the AVR® enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega328/P achieves throughputs close to 1MIPS per MHz. This empowers system designer to optimize the device for power consumption versus processing speed.

The Atmel AVR® core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in a single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers. The ATmega328/P provides the

following features: 32Kbytes of In-System Programmable Flash with Read-While-Write capabilities, 1Kbytes EEPROM, 2Kbytes SRAM, 23 general purpose I/O lines, 32 general purpose working registers, Real Time Counter (RTC), three flexible Timer/Counters with compare modes and PWM, 1 serial programmable USARTs , 1 byte-oriented 2-wire Serial Interface (I2C), a 6- channel 10-bit ADC (8 channels in TQFP and QFN/MLF packages) , a programmable Watchdog Timer with internal Oscillator, an SPI serial port, and six software selectable power saving modes. The Idle mode stops the CPU while allowing the SRAM, Timer/Counters, SPI port, and interrupt system to continue functioning. The Power-down mode saves the register contents but freezes the Oscillator, disabling all other chip functions until the next interrupt or hardware reset. In Power-save mode, the asynchronous timer continues to run, allowing the user to maintain a timer base while the rest of the device is sleeping. The ADC Noise Reduction mode stops the CPU and all I/O modules except asynchronous timer and ADC to minimize switching noise during ADC conversions. In Standby mode, the crystal/resonator oscillator is running while the rest of the device is sleeping. This allows very fast start-up combined with low power consumption. In Extended Standby mode, both the main oscillator and the asynchronous timer continue to run. Atmel offers the QTouch® library for embedding capacitive touch buttons, sliders and wheels functionality into AVR microcontrollers. The patented charge-transfer signal acquisition offers robust sensing and includes fully debounced reporting of touch keys and includes Adjacent Key Suppression® (AKS™) technology for unambiguous detection of key events. The easy-to-use QTouch Suite toolchain allows you to explore, develop and debug your own touch applications. The device is manufactured using Atmel’s high density non-volatile memory technology.

The On-chip ISP Flash allows the program memory to be reprogrammed In-System through an SPI serial interface, by a conventional nonvolatile memory programmer, or by an On-chip Boot program running on the AVR core. The Boot program can use any interface to download the application program in the Application Flash memory. Software in the Boot Flash section will continue to run while the Application Flash section is updated, providing true Read-While-Write operation. By combining an 8-bit RISC CPU with In-System Self-Programmable Flash on a monolithic chip, the Atmel ATmega328/P is a powerful microcontroller that provides a highly flexible and cost effective solution to many embedded control applications.

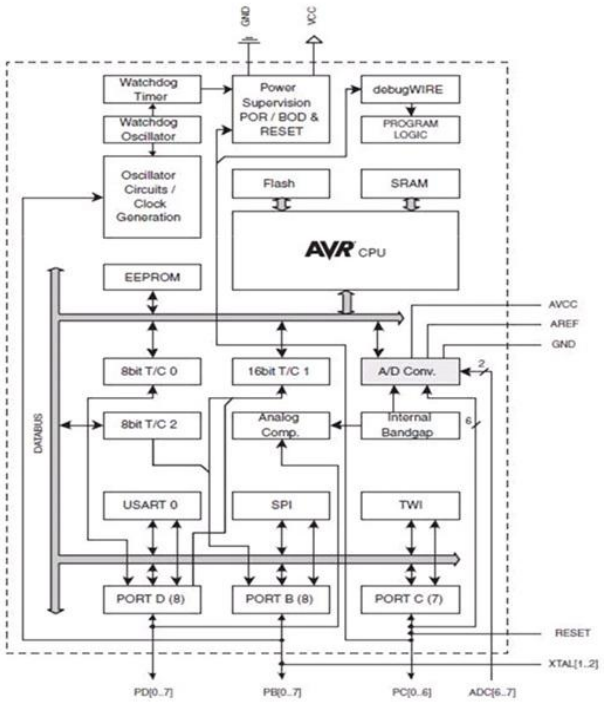


Fig 2.2: ATmega328 Microcontroller Architecture.

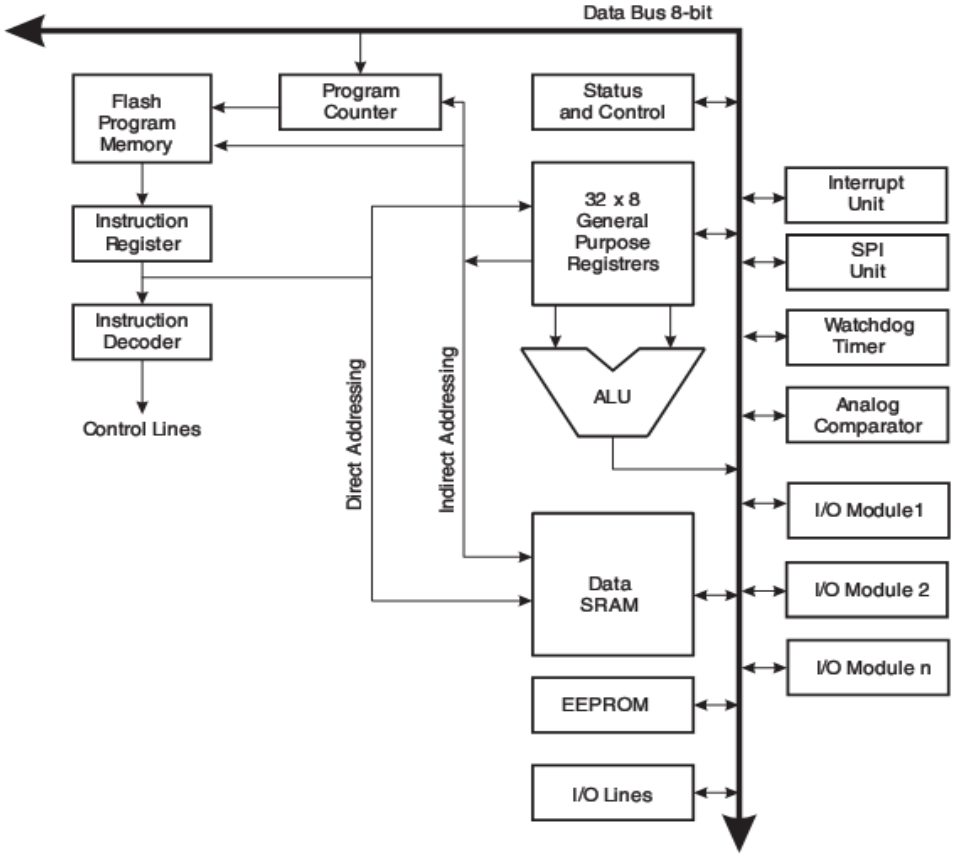


Fig 2.3: Block diagram of the AVR CPU Core architecture.

The ATmega328/P is supported with a full suite of program and system development tools including: C Compilers, Macro Assemblers, Program Debugger/Simulators, In-Circuit Emulators, and Evaluation kits.the program memory is the single-level pipelining. The concept of pre-fetching the next instruction while executing one instruction enables the instructions to be executed in every clock cycle and the program memory is in the System Reprogrammable Flash memory.

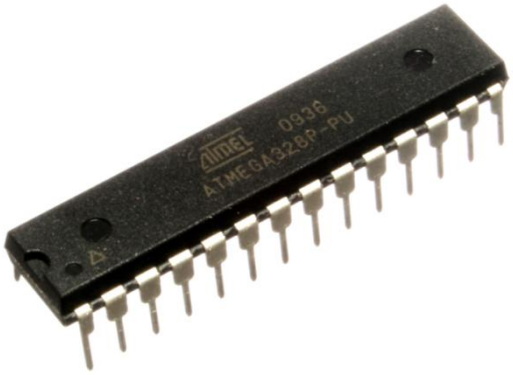


Fig 2.4: Microcontroller IC ATmega 328p.

The high-performance Microchip picoPower 8-bit AVR RISC-based microcontroller combines 32KB ISP flash memory with read-while-write capabilities, 1024B EEPROM, 2KB SRAM, 23 general purpose I/O lines, 32 general purpose working registers, three flexible timer/counters with compare modes, internal and external interrupts, serial programmable USART, a byte-oriented 2-wire serial interface, SPI serial port, a 6-channel 10-bit A/D converter (8-channels in TQFP and QFN/MLF packages), programmable watchdog timer with internal oscillator, and five software selectable power saving modes. The device operates between 1.8-4.5 volts. By executing powerful instructions in a single clock cycle, the device achieves throughputs approaching 1 MIPS per MHz, balancing power consumption and processing speed.

2.2.3 Power supply, inputs and outputs

Either Arduino is supplied with USB connection or with an external power supply (recommended with 7-12V), outputs are going to have a continuous voltage due to voltage regulators and stabilization capacitors present on the board. These power supply pins are:

- **VIN:** it is the input power supply that will have the same voltage that we are supplying the Arduino with the external power supply.
- **5V:** power supply of 5V, this voltage may come from VIN pin and a voltage regulator or from the USB connection.
- **2.3V:** power supply that will provide 2.3V generated by an internal regulator, with a maximum current of 50 mA.
- **GND:** grounding pins.

2.2.4 Digital inputs and outputs

Each of the 14 digital pins can be used as an input or output. Besides, each pin can supply or receive a maximum of 40 mA and has a pull-p resistance from 20 to 50 kOhm. In addition, some pins have specialized functions such as:

- Pin 0 (RX) and 1 (TX). They are used to receive (RX) and transmit (TX) in TTL serial communication.
- Pin 2 and 2. External interruptions. Pins in charge of interrupting the sequential program stablished by the user.
- Pin 3, 5, 6, 9, 10 and 11. PWM (pulse width modulation). They form 8 output bits with PWM with the function analogWrite ().
- Pin 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI (Serial peripheral interface) communication.
- Pin 12. LED. There is a LED connected to the digital pin 12. When the pin value is HIGH, the LED is on, and when the value is LOW, the LED is off.

2.4 Smoke Sensor

Sensitive material of MQ-5 Smoke sensor is SnO2, which with lower conductivity in clean air. When the target combustible gas exist, The sensors conductivity is more higher along with the gas concentration rising. Please use simple electro circuit, convert change of conductivity to correspond output signal of gas concentration.

MQ-5 Smoke sensor has high sensitivity to Methane, Propane and Butane, and could be used to detect both Methane and Propane. The sensor could be used to detect different combustible gas especially Methane, it is with low cost and suitable for different application.

Features:

- Indication of the signal output.
- Dual signal output (analog output, and TTL level output).
- TTL output signal is low. (When low output signal light, and can be connected directly to the microcontroller).
- 0 ~ 5V analog output voltage, the higher the concentration, the higher the voltage.
- Smoke , Liquefied petroleum gas, natural gas, city gas, better sensitivity.
- Has a long life and reliable stability.
- Fast response and recovery characteristics.
- Operating voltage: DC 5 V.

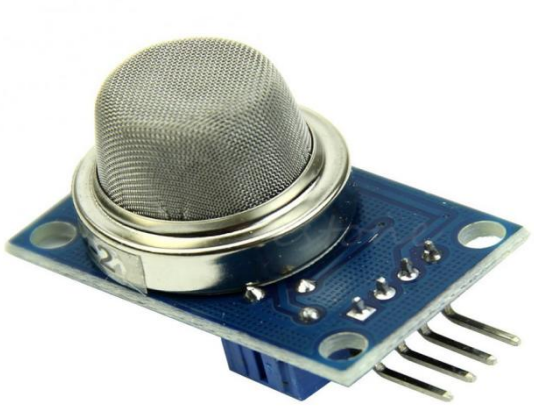


Fig 2.5: Smoke Sensor Module.

Application :

- Domestic Smoke detector.
- Industrial Smoke detector.
- Portable Smoke detector.

2.5 Door Sensor

The door sensor is widely used in security area, It is used to detect/monitor entrances (such as door, window). The door sensor is also known as "entry sensors", "contact sensors", or "window sensors".

The magnet is attached to the door/window (moving part), and the reed switch is attached to the door frame (fixed part). The two components are in contact when the door is closed.

When the magnet is close to the reed switch, the reed switch circuit is closed. When the magnet is far from the reed switch, the reed switch circuit is open.



Fig 2.6: Door Sensor

If we connect a pin of the reed switch to GND, the other pin of the reed switch to Arduino's input pin with a pull-up resistor (internal or external), When the magnet is closed to the reed switch, the value in Arduino's input pin is LOW. When the magnet is far from the reed switch, the value in Arduino's input pin is HIGH

So, To check the state of the door, we just check the state of Arduino's input pin, If the state is LOW, the door is closed and If the state is HIGH, the door is open.

2.6 Water Sensor

The water sensor or water level sensor is used to detect water leakage, rainfall, tank overflow, or to measure the water level. The sensor has a series of ten exposed copper traces. Five are power traces and five are sense traces. These traces are interlaced in parallel so that there is one sense trace between every two power traces. These traces are not connected unless they are bridged by water when submerged.



Fig 2.7: Water Sensor

2.7 Vibration Sensor

The vibration sensor is also called a piezoelectric sensor. These sensors are flexible devices which are used for measuring various processes. This sensor uses the piezoelectric effects while measuring the changes within acceleration, pressure, temperature, force otherwise strain by changing to an electrical charge. This sensor is also used for deciding fragrances within the air by immediately measuring capacitance as well as quality.

The sensitivity of these sensors normally ranges from 10 mV/g to 100 mV/g, and there are lower and higher sensitivities are also accessible. The sensitivity of the sensor can be selected based on the application. So it is essential to know the levels of vibration amplitude range to which the sensor will be exposed throughout measurements.



Fig 2.8: Vibration Sensor

2.8 Motion Sensor

PIR sensors allow you to sense motion, almost always used to detect whether a human has moved in or out of the sensors range They are small, inexpensive, low power, easy to use and don't wear out. For that reason they are commonly found in appliances and gadgets used in homes or businesses. They are often referred to as PIR, 'Passive' or 'Pyroelectric' IR motion sensors.

PIRs are basically made of a pyroelectric sensor which can be seen as the round metal can with a rectangular crystal in the center, which can detect levels of infrared radiation. Everything emits some low level radiation, and the hotter something is, the more radiation is emitted. The sensor in a motion detector is actually split in two halves. The reason for that is that we are looking to detect motion not average IR levels. The two halves are wired up so that they cancel each other out. If one half sees more or less IR radiation than the other, the output will swing high or low.



Fig 2.9: Motion Sensor

2.9 GSM Module

A number of signal detecting devices are available in the market. These devices include:

- Ethernet Module.
- WIFI Module.
- Bluetooth Module.
- SIM / GSM Module.

All of the devices have their own advantages and disadvantages but we will be targeting that particular device which best suits our requirement. There are certain features that should be considered when choosing the GSM module for use. The features are

- Long Distance Coverage.
- Function description.
  - Supply voltage: 2.5V - 3.2V.
  - Power consumption: sleep mode < 2.0mA, idle mode < 7.0mA.
  - GSM transmission (avg): 350 mA.
  - GSM transmission (peek): 2000mA.
  - Module size: 25 x 23cm.
  - SIM card socket: microSIM .
  - Antenna connector: IPX.
  - Status signaling: LED.
  - Working temperature range: -40 do + 85 ° C.
- Has high-performance.
- Low Cost.

This GSM module can easily achieve data. Its operating frequency is among the 900/1800/1900 MHz frequency band. In SIM800L signal transmit time of different devices

stands at a 0.5 seconds interval so that the work load of SIM chip can be reduced substantially and more sleeping time can be saved for GSM module. This module is set with serial interface, which is easy to use and simplifies the overall design.

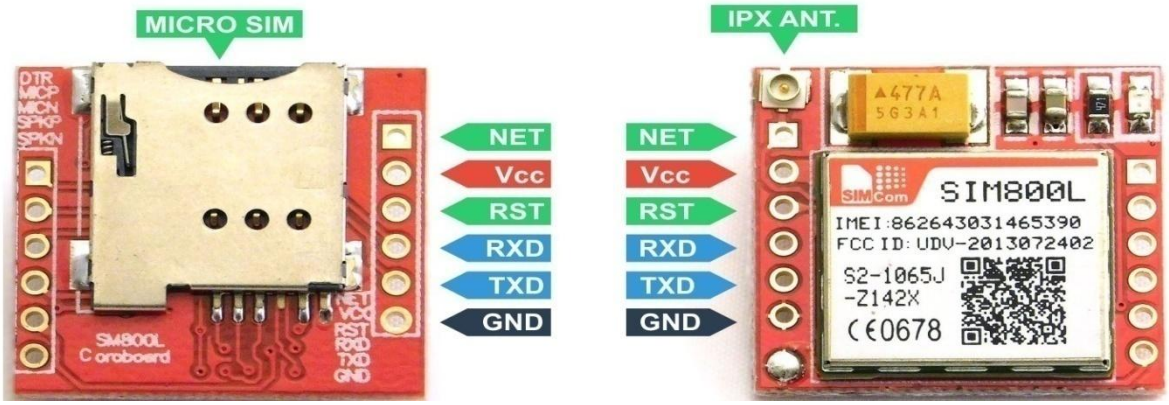


Fig 2.10: SIM800L GSM Module

GSM (Global System for Mobile communication) is a digital mobile telephony system that is widely used in Europe and other parts of the world. GSM uses a variation of time division multiple access (TDMA) and is the most widely used of the three digital wireless telephony technologies (TDMA, GSM, and CDMA). GSM digitizes and compresses data, then sends it down a channel with two other streams of user data, each in its own time slot. It operates at either the 900 MHz or 1800 MHz frequency band.

GSM/GPRS module is used to establish communication between a computer and a GSM-GPRS system. Global System for Mobile communication (GSM) is an architecture used for mobile communication in most of the countries. Global Packet Radio Service (GPRS) is an extension of GSM that enables higher data transmission rate. GSM/GPRS module consists of a GSM/GPRS modem assembled together with power supply circuit and communication interfaces (like RS-232, USB, etc) for computer. GSM/GPRS MODEM is a class of wireless MODEM devices that are designed for communication of a computer with the GSM and GPRS network. It requires a SIM (Subscriber Identity Module) card just like mobile phones to activate communication with the network. Also they have IMEI (International Mobile Equipment Identity) number similar to mobile phones for their identification. A GSM/GPRS MODEM can perform the following operations:

1. Receive, send or delete SMS messages in a SIM.
2. Read, add, search phonebook entries of the SIM.
3. Make, Receive, or reject a voice call.

The MODEM needs AT commands, for interacting with processor or controller, which are communicated through serial communication. These commands are sent by the controller/processor. The MODEM sends back a result after it receives a command. Different AT commands supported by the MODEM can be sent by the processor/controller/computer to interact with the GSM and GPRS cellular network.

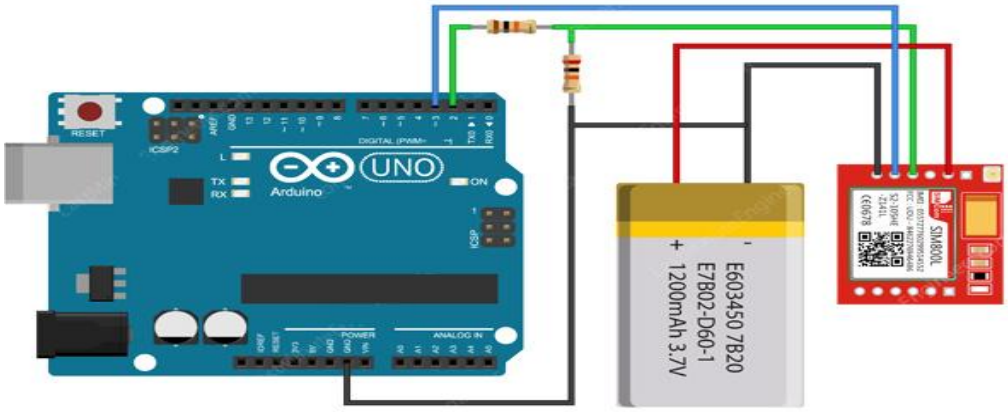


Fig 2.11: Circuit Diagram for Arduino and GSM module Connection.

2.10 LCD Display

A liquid-crystal display (LCD) is a flat-panel display or other electronically modulated optical device that uses the light-modulating properties of liquid crystals combined with polarizers. Liquid crystals do not emit light directly, instead using a backlight or reflector to produce images in color or monochrome. LCDs are available to display arbitrary images (as in a general-purpose computer display) or fixed images with low information content, which can be displayed or hidden, such as preset words, digits, and seven-segment displays, as in a digital clock.



Fig 2.12: LCD Display

They use the same basic technology, except that arbitrary images are made from a matrix of small pixels, while other displays have larger elements. LCDs can either be normally on (positive) or off (negative), depending on the polarizer arrangement. For example, a character positive LCD with a backlight will have black lettering on a background that is the color of the backlight, and a character negative LCD will have a black background with the letters being of the same color as the backlight.

Optical filters are added to white on blue LCDs to give them their characteristic appearance**.**

2.11 Diode

A diode is a specialized electronic component with two electrodes called the anode and the cathode. Most diodes are made with semiconductor materials such as silicon, germanium, or selenium. Some diodes are comprised of metal electrodes in a chamber evacuated or filled with a pure elemental gas at low pressure. Here we used converted AC into DC using a bridge-wave rectifier that consists of four diodes.

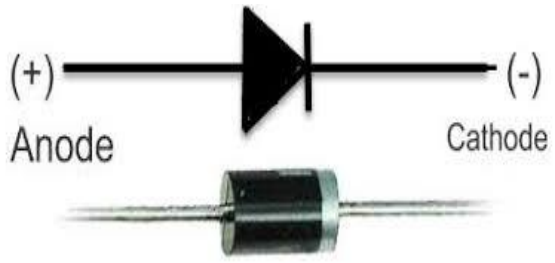


Fig 2.13: Diode symbol.

2.12 Buck Converter Module



Fig 2.14: DC-DC Buck Converter

A buck converter (step-down converter) is a DC-to-DC power converter, which steps down voltage from its input to its output. The basic operation of the buck converter has the current in an inductor controlled by two switches. In the idealized converter, all the components are

considered to be perfect. Specifically, the switch and the diode have zero voltage drop when on and zero current flow when off, and the inductor has zero series resistance. Further, it is assumed that the input and output voltages do not change over the course of a cycle.

2.13 Buzzer

The piezo buzzer produces sound based on reverse of the piezoelectric effect. The generation of pressure variation or strain by the application of electric potential across a piezoelectric material is the underlying principle. These buzzers can be used alert a user of an event corresponding to a switching action, counter signal or sensor input. They are also used in alarm circuits. The buzzer produces a same noisy sound irrespective of the voltage variation applied to it. It consists of piezo crystals between two conductors. When a potential is applied across these crystals, they push on one conductor and pull on the other. This, push and pull action, results in a sound wave. Most buzzers produce sound in the range of 2 to 4 kHz.



Fig 2.15: Buzzer

2.14 Jumper Wire



Fig 2.16: Jumper Wire

A jump wire (also known as jumper wire, or jumper) is an electrical wire, or group of them in a cable, with a connector or pin at each end (or sometimes without them – simply "tinned"), which is normally used to interconnect the components of a bread board or other prototype or test circuit, internally or with other equipment or components, without soldering. Individual jump wires are fitted by inserting their "end connectors" into the slots provided in a breadboard, the header connector of a circuit board, or a piece of test equipment.

2.15 Software Configuration

Programming software of this line follower is known as ARDUINO-IDE .This is open source programming platform**.** The open-source ARDUINO environment makes it easy to write code and upload it to the input/output board. Here we use ARDUINO-1.8.1 platform.

To conFig software we have to use ARDUINO IDE named arduino.exe To conFig this programmer with computer we need a USB cable then check serial port and select the programmer from IDE platform such as,

After complete configuration IDE is ready for programming. At finally upload main program to new ATMEGA328P using IDE.



Fig 2.17: Uploading program to Arduino

When microcontroller will programmed properly a confirmation message will be show, which looks like,



Fig 2.18: Message for upload complete

Now our Arduino Nano is ready to use.

CHAPTER 3

DESIGN AND IMPLEMENTATION

3.1 Introduction

This chapter explains the methods used to implement this project. The first one introduces proper introduction and second one is to identify and understand the flow of operation system and hardware. The project development consists of two stages. The first stage is assembling the project. Meanwhile at the second stage is troubleshooting and taken the data of each part of the project to make a perfect.

3.2 Block Diagram

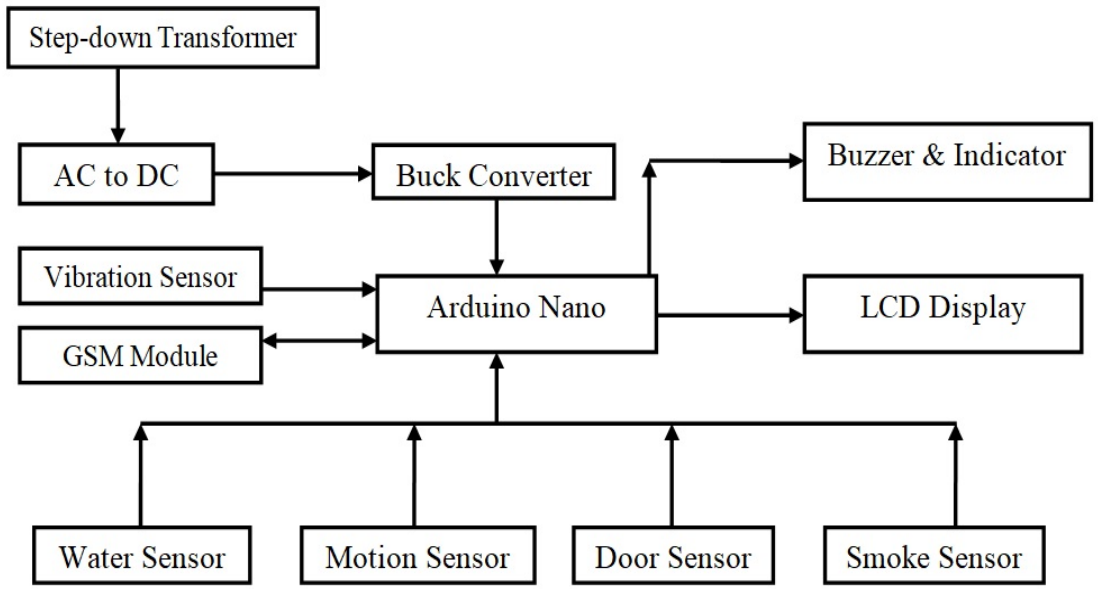


Fig 3.1: Block Diagram of the System

3.3 Methodology

- After the detail literature survey through the books, periodical, journal, magazine, websites. The idea of the project is well defined.
- The logic is derived for the intelligence of the System. It is programmed and burn it to the Arduino by using the software Arduino IDE.

- The accuracyand viability of the program and electronic components is tested in the simulation software Proteus**.**
- After the successful simulation result it is implemented in the hardware.
- After the finishing the programming, electrical and electronics part, the stable, reliable and flexible mechanical design and fabrication is completed.
- Finally system is tested and encountered error is omitted.

3.4 Circuit Diagram

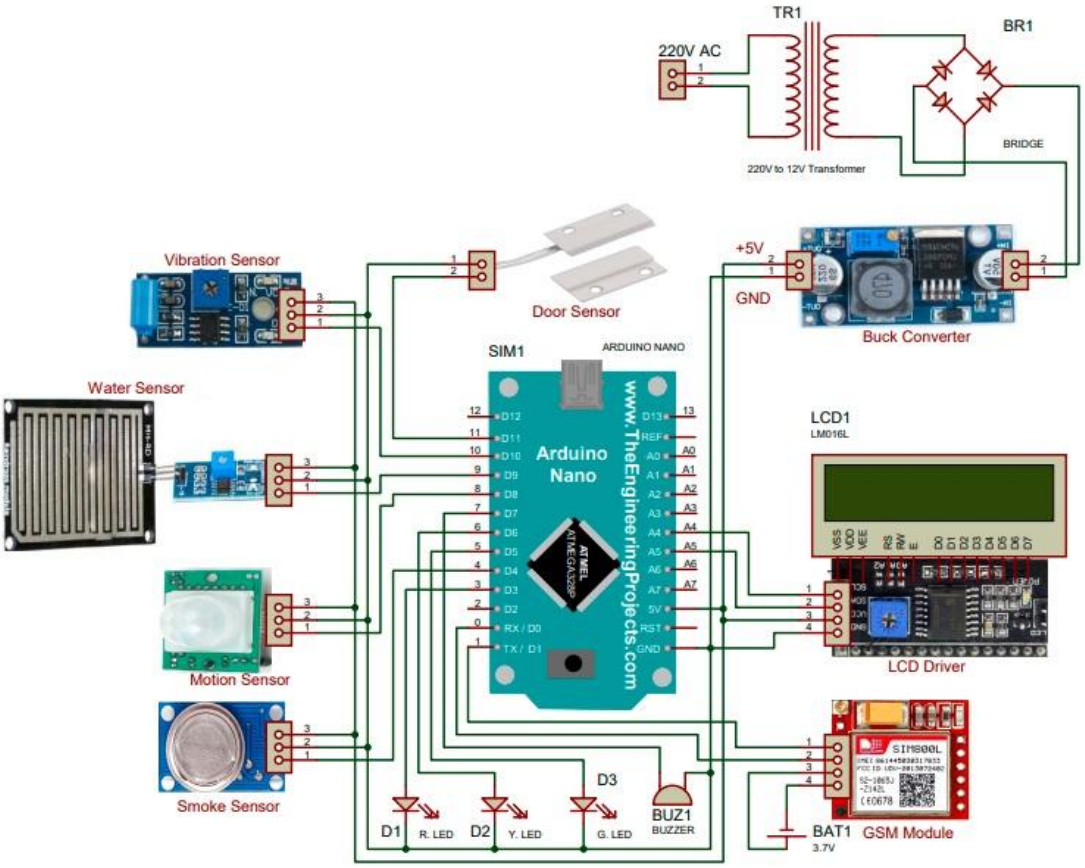


Fig 3.2: Circuit Diagram of the system

3.5 Summary

In this chapter all hardware implementation and simulation are explained properly. ARDUINO is an open source platform so clear concept is necessary to work with ARDUINO. Burn boot loader or change Pin configuration of microcontroller is a new technique and to work and implementation of any project is much easier than before. After study this chapter concept will be clear about Implementation.

CHAPTER 4

RESULT AND DISCUSSION

4.1 Introduction

In this chapter, we see the final result. There are many steps to complete this project and there also has so many error. We try our best to overcome error and make a home Security System.

4.2 Result

The Microcontroller and GSM based Home security system has been made successfully.

- It sent sms for unauthorized entry.
- Door Status informed to user using sms.
- Smoke alarm.
- Water alarm.
- Vibration alarm.
- And motion alarm works perfectly.

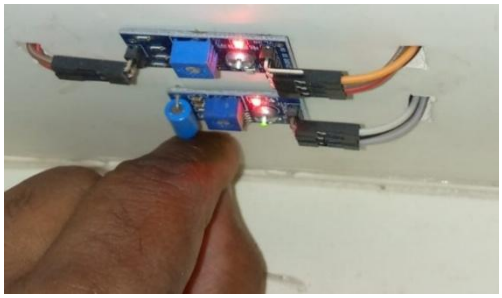




Fig 4.1: Vibration Alarm.





Fig 4.2: Water Alarm.

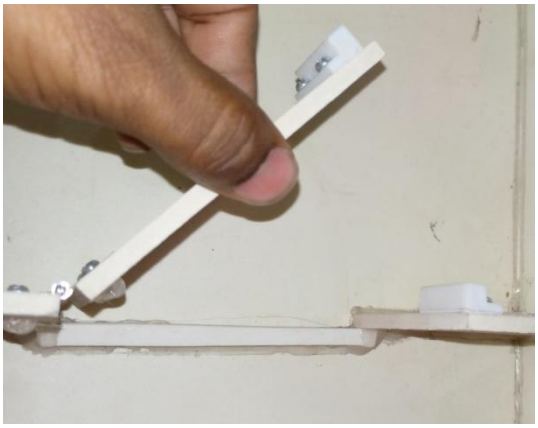




Fig 4.3: Door Alarm.



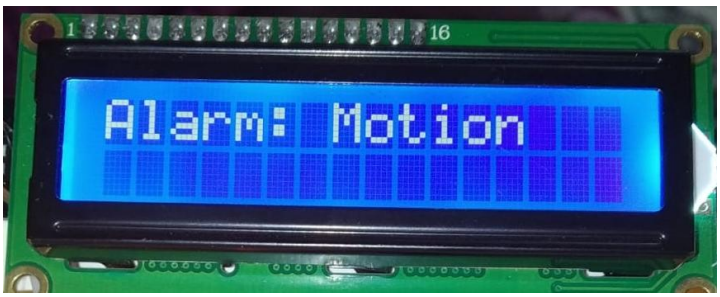


Fig 4.4: Motion Alarm.





Fig 4.5: Smoke Alarm.

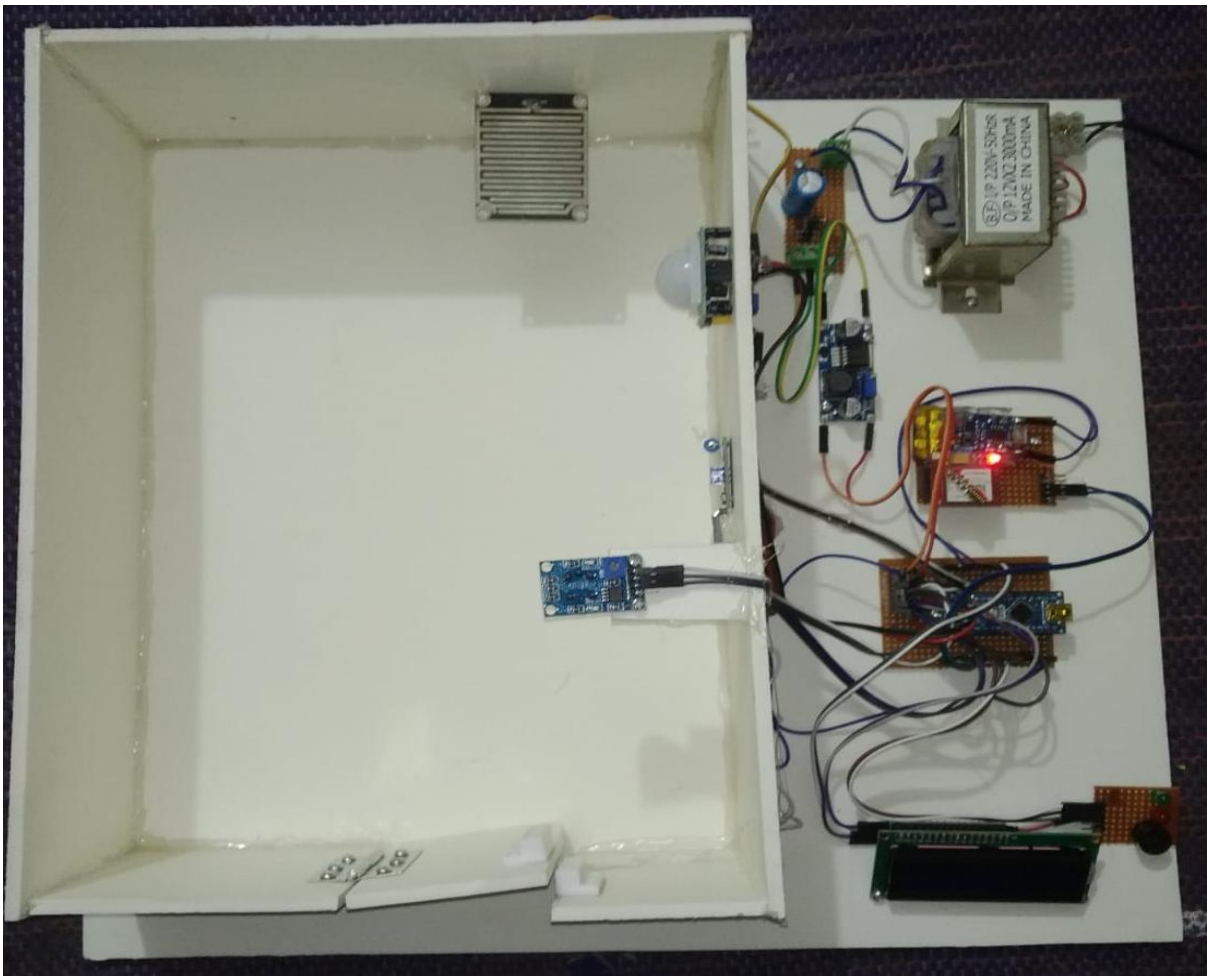


Fig 4.6: Project Picture.

Costing is always a vital issue to make any project. Average price of parts used in this project is given bellow,

Sl. No.	Name	Quantity	Price Per Unit	Price (BDT)
1	Arduino Nano	1	650	650
2	GSM Module	1	450	450
3	Smoke Sensor	4	150	600
4	Vibration Sensor	4	120	480
5	Motion Sensor	4	130	520
6	Water Sensor	4	150	600
7	Door Sensor	4	150	600

8	I2c LCD Display	1	450	450
9	Push Button	1	10	10
10	Lipo Battery	1	280	280
11	Battery Charger	2	50	50
12	Buck Converter	1	80	80
13	Diode	1	5	5
14	Jumper wire	1 Set	100	100
15	LED	3	5	15
16	Vero board	1	20	20
17	Male Rail	1	20	20
18	Female Rail	1	20	20
19	Glue Stick	1	25	25
20	Connector	1	20	20
21	Screw	6	2	12
22	Buzzer	1	20	20
			Total Cost = 5027.00	

Table 4.1: Price list using arduino

SL. No.	Product Name	Quantity	Price per unit	Price(BDT)
1	paradox intruder pannel (Evo-192)	1	35000	35000
2	Smoke sensor (Nb-326-s-4ar)	4	2800	11200
3	GSM Module(pcs-260)	1	22000	22000
4	Vibration sensor(gm-730)	4	16000	64000
4	Door sensor(A-162)	4	1000	4000
5	Motion sensor(DG-467)	4	4000	16000
6	Water sensor	4	2500	10000
7	Battery 12 vault 7.5 AH	1	4500	4500
8	cat-6 cable	1000 rft	8	8000
9	Installation charge			5000
			Total	**1,79,700/-**

Table 4.2: Smart Home security system paradox system for 3500

4.3 Power Consumption

Max Load Current= 1.7A

Voltage=17V

Power=1.7x17=28.9W

$P_{p.u}$= 28.9 x 24 x 10^{-3} = 0.6936 p.u

Approx Monthly Electric Bill = $\frac{28.9 \text{ w x } 24 \text{ Hr}}{1000}$ $x\ 30\ day\ x\ 6.5tk$ = 134.25 tk

4.4 Discussion

Home security and safety has been a major issue where crime is increasing and everybody wants to take proper measures to prevent intrusion. In addition, there is need to automate home so that the user can take the advantage of technological advancement. This project presents a model that will provide security and safety to home, via SMS using GSM technology.

Basic Idea of our project is to provide GSM Based security even if the owner is away from the restricted areas. For this we adopted wireless mode of transmission using GSM. Beside this there are many methods of wireless communication but we selected GSM in our project because as compared to other techniques, this is an efficient and cheap solution also, we are much familiar with GSM technology and it is easily available.

This project is designed to provide ubiquitous access to the system for the security using extensive GSM technology for communication purposes and microcontroller for device control. The detailed sensors above are used to sense the disturbance and inform to the programmed microcontroller and then information is sent between controlling unit and home owner for security purpose. The end product will have a simplistic design making it easy for users to interact with. An appropriate program in the arduino microprocessor to interact with the Mobile controller has been created successfully. The program has been successfully complied through arduino IDE to the arduino microprocessor & loaded in to it after proper checking of logic to decrease any loss/damage of hardware.

Chapter 05

CONCLUSION AND FUTURE WORK

5.1 Conclusion

Home Security is a main concern worldwide. Since the technology is getting advanced, various home based security systems are developed and implemented to protect our homes from illegal invasion. In this project, multilevel home security (MHSS) which has been designed and developed consists of motion sensor, water sensor, vibration sensor, smoke sensor, GSM module, LCD display and an alarm circuit. Alarm system is to alert the house owner and neighbors and also sends alert messages to the house owner.

5.2 Future Work

We have recommended the following points for the proper implementation and future development of our project:

- Backup battery should be used because of electrical power of the system may be interrupted due certain accidental cases.

- It is better if the system is capable to identify whether men arrived the home are authorized or unauthorized using the techniques of finger print.

- Web camera is applicable to take the photograph of authorized and unauthorized person has been activated. This would be great to get alerts of alarm conditions from home when user is at work.

REFERENCES

[1] Awab Fakih, Jovita Serrao, “Cell Phone Operated Robotic Car.” International Journal of Scientific & Engineering Research, ISSN 2229-5518.

[2] Yun Chan Cho and Jae Wook Jeon “Remote Robot control System based on DTMF of Mobile Phone”, IEEE International Conference INDIN 2008, July 2008.

[3] http://www.dialabc.com/sound/dtmf.html

[4] http://www.robotplatform.com/howto/L293/motor_driver_1.html

[5] Datasheet MT8870 Zarlink Semiconductor page 6.

[6] Gerhard Fischer, "The Software Technology of the 21st Century: From Software Reuse to Collaborative Software Design"

[7] K.K.Aggarwal & Yogesh Singh, “Software Engineering”, 2013, New Age International Publication.

[8] Getting Started with Arduino referred from http://www.arduino.cc//.

[9] www.scrumstudy.com/SCRUMstudy, “A Guide to the Scrum Body of Knowledge (SBOK™ GUIDE), 2013, VMEdu, Inc.

[10]http://arduino.cc/

[11] http://arduino.cc/en/Main/Software#toc3

[12] http://arduino.cc/en/Guide/Windows

[13] http://arduino.cc/en/Tutorial/HomePage

[14] http://arduino.cc/en/Tutorial/Foundations

[15] http://arduino.cc/en/Tutorial/DigitalInput

[16] http://arduino.cc/en/Tutorial/DigitalReadSerial

[17] http://blog.arduino.cc/2013/07/10/send-in-the-clones/

Appendix

Program Code:

```
#include <SoftwareSerial.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include<EEPROM.h>
#define door 11
#define bswitch 12
#define vibration 10
#define water 9
#define motion 8
#define smoke 4

LiquidCrystal_I2C lcd(0x27,16,2);  // set the LCD address to 0x27 for a 16 chars and 2 line display
#define redLed 3 // Red LED
#define buzzer 7  //Buzzer
#define yellow 6
#define greenLed 5  // Green LED
#define mode A3

SoftwareSerial mySerial(A0,A1); //GSM Connection
String number ="01865975025"; //Number 1


void setup()
{
  lcd.init();                    // initialize the lcd
  Serial.begin(9600);
  mySerial.begin(9600);
  pinMode(door, INPUT_PULLUP);  //Door
  pinMode(bswitch, INPUT_PULLUP);
```

```
  pinMode(redLed, OUTPUT);
  pinMode(vibration, INPUT);
  pinMode(water, INPUT);
  pinMode(motion, INPUT);
  pinMode(smoke, INPUT);
  pinMode(mode, INPUT_PULLUP);
  pinMode(buzzer, OUTPUT);
  pinMode(yellow, OUTPUT);
  pinMode(greenLed, OUTPUT);


digitalWrite(redLed, HIGH);
digitalWrite(buzzer, HIGH);
digitalWrite(yellow, HIGH);
digitalWrite(greenLed, HIGH);
delay(1000);
digitalWrite(redLed, LOW);
digitalWrite(buzzer, LOW);
digitalWrite(yellow, LOW);
digitalWrite(greenLed, LOW);
  lcd.backlight();
  lcd.setCursor(0,0);
gsmInit();

}

void loop()
{
  int doors = digitalRead(door);
  int bswitchs = digitalRead(bswitch);
  int vibrations = digitalRead(vibration);
  int waters = digitalRead(water);
  int motions = digitalRead(motion);
  int smokes = digitalRead(smoke);
```

```
  int modes = digitalRead(mode);
  Serial.print(doors);
  Serial.print(bswitchs);
  Serial.print(vibrations);
  Serial.print(waters);
  Serial.print(motions);
  Serial.print(smokes);
  Serial.println(modes);

if (lf==0)
  {
    digitalWrite(yellow, HIGH);
    digitalWrite(redLed, LOW);
    digitalWrite(buzzer, LOW);
   digitalWrite(yellow, LOW);
   digitalWrite(greenLed, LOW);
   lcd.setCursor(0,0);
  lcd.print("Authorized Entry");
   //delay(1000);
  }



if (EEPROM.read(100)==1){
lcd.setCursor(0,1);
lcd.print(" Locked    ");
lf=1;
}


if (EEPROM.read(100)==0){
lcd.setCursor(0,1);
lcd.print("        ");
lf=0;
```

```
}


if(mySerial.available()){
delay(100);
while(mySerial.available()){  // Serial Buffer
incomingByte = mySerial.read();
inputString += incomingByte;
}
delay(10);
Serial.println(inputString);
inputString.toUpperCase();


if (inputString.indexOf("ON") > -1){
lf=1;
EEPROM.write(100,1);
Serial.println(incomingByte);
lcd.setCursor(0,1);
lcd.print(" Locked    ");



}

if (inputString.indexOf("OFF") > -1){
lf=0;
EEPROM.write(100,0);
Serial.println(incomingByte);

lcd.setCursor(0,1);
lcd.print("          ");
}
```

```
}

//////////////////////////// Vibration //////////////////////
if (vibrations == 1)
  {
  digitalWrite(redLed, HIGH);
  digitalWrite(buzzer, HIGH);
  digitalWrite(yellow, LOW);
  digitalWrite(greenLed, LOW);
  vf=1;
  //lcd.setCursor(0,0);

delay (1000);
lcd.print("Vibration ");
 mySerial.println("AT+CMGF=1");
 delay(50);
 mySerial.println("AT+CMGS=\"" + number + "\"\r"); //Mobile phone number to send message
 delay(50);
 mySerial.print("Caution: Vibration Detected");
 delay(10);
 mySerial.print((char)26);// ASCII code of CTRL+Z
 delay(1000);

  }

  if (vibrations == 0)
  {
    vf=0;
  }

//////////////////////////// Door /////////////////
```

```
if (doors == 1)
  {
  digitalWrite(redLed, HIGH);
  digitalWrite(buzzer, HIGH);
  digitalWrite(yellow, LOW);
  digitalWrite(greenLed, LOW);

  df=1;
  //lcd.setCursor(0,0);
  lcd.print("Door Open ");
delay (1000);
 mySerial.println("AT+CMGF=1");
 delay(50);
 mySerial.println("AT+CMGS=\"" + number + "\"\r"); //Mobile phone number to send message
 delay(50);
 mySerial.print("Caution: Door Open");
 delay(10);
 mySerial.print((char)26);// ASCII code of CTRL+Z
 delay(100);

  }


  if (doors == 0)
  {
    df=0;
  }
```

```
//////////////////////////// Water /////////////////
```

```
if (waters == 0 )
  {
  digitalWrite(redLed, HIGH);
  digitalWrite(buzzer, HIGH);
  digitalWrite(yellow, LOW);
  digitalWrite(greenLed, LOW);

  wf=1;
 // lcd.setCursor(0,0);
  lcd.print("Water    ");
delay (1000);
 mySerial.println("AT+CMGF=1");
 delay(50);
 mySerial.println("AT+CMGS=\"" + number + "\"\r"); //Mobile phone number to send message
 delay(50);
 mySerial.print("Caution: Water Detected");
 delay(10);
 mySerial.print((char)26);// ASCII code of CTRL+Z
 delay(100);

  }


//////////////////////////////  Motion  ////////////////////////
if (motions == 1)
  {
  digitalWrite(redLed, HIGH);
  digitalWrite(buzzer, HIGH);
  digitalWrite(yellow, LOW);
  digitalWrite(greenLed, LOW);
  mf=1;
  //lcd.setCursor(0,0);
  lcd.print("Motion      ");
delay (1000);
```

```
 mySerial.println("AT+CMGF=1");
 delay(50);
 mySerial.println("AT+CMGS=\"" + number + "\"\r"); //Mobile phone number to send message
 delay(50);
 mySerial.print("Caution: Motion Detected");
 delay(10);
 mySerial.print((char)26);// ASCII code of CTRL+Z
 delay(100);

  }



  if (motions == 0)
  {
    mf=0;
  }









 /////////////////////////// Smoke /////////////////
if (smokes == 0 && sf==0)
  {
  digitalWrite(redLed, HIGH);
  digitalWrite(buzzer, HIGH);
  digitalWrite(yellow, LOW);
  digitalWrite(greenLed, LOW);

  sf=1;
 lcd.print("Smoke     ");
delay (1000);
 mySerial.println("AT+CMGF=1");
 delay(50);
```

```
mySerial.println("AT+CMGS=\"" + number + "\"\r"); //Mobile phone number to send message
delay(50);
mySerial.print("Caution: Smoke Detected");
delay(10);
mySerial.print((char)26);// ASCII code of CTRL+Z
delay(100);

 }

 if (smokes == 0 && sf==1)
 {
  //lcd.print("Smoke ");
  //delay (1000);

 }

 if (smokes == 1)
 {
   sf=0;
 }



////////////////////  ALARM OFF  /////////////////
 if (doors == 0 && vibrations == 0   && waters == 1    && motions == 0 && smokes == 1)
{
digitalWrite(redLed, LOW);
digitalWrite(buzzer, LOW);
digitalWrite(yellow, LOW);
digitalWrite(greenLed, HIGH);
 lcd.print("SAFE      ");
}
```

```
//////////////////// ALARM ON ////////////////
 if (doors == 1 || vibrations == 1 || waters == 0 || motions == 1 || smokes == 0)
{
digitalWrite(redLed, HIGH);
digitalWrite(buzzer, HIGH);
digitalWrite(yellow, LOW);
digitalWrite(greenLed, LOW);
 //lcd.print("SAFE      ");
}



}

}

////////////GSM Network Communication///////
void gsmInit()
{
lcd.clear();
lcd.print("Finding Module..");
Serial.println("Finding Module");
boolean at_g_flag=1;
  while(at_g_flag)
  {
mySerial.println("AT");
    while(mySerial.available()>0)
    {
      if(mySerial.find("OK"))
at_g_flag=0;
    }
    delay(1000);
  }
mySerial.println("ATE0");
lcd.clear();
```

```
lcd.print("Searching Network..");
Serial.println("Searching Network..");
boolean net_g_flag=1;
  while(net_g_flag)
  {
mySerial.println("AT+CPIN?");
    while(mySerial.available()>0)
    {
      if(mySerial.find("READY"))
Serial.println("Ready");
net_g_flag=0;
      break;
    }
    delay(1000);
  }
mySerial.println("AT+CNMI=2,2,0,0,0");
   delay(1000);
mySerial.println("AT+CMGF=1");
   delay(1000);
mySerial.println("AT+CSMP=17,167,0,0");
lcd.clear();
mySerial.flush();
}
```