CHAPTER 1

INTRODUCTION

1.1 Introduction

The foremost aim of technology has been to increase efficiency and decrease effort. With the advent of 'Internet of Things' in the last decade, we have been pushing for ubiquitous computing in all spheres of life. It thus is of extreme importance to simplify human interfacing with technology. Automation is one such area that aims that achieves simplicity whilst increasing efficiency. Voice controlled Home Automation System aims to further the cause of automation so as to achieve the goal of simplicity.

The primitive man realized that an effective way to communicate with one another is through voice. With minimum effort, ideas could be narrated with relative ease. When the first computers came around, achieving the level of sophistication so as to narrate commands using voice to a machine was only realized in science fiction. However, with tremendous breakthroughs in the field, we are at the precipice of truly using voice to interface with devices. Using this effective yet ingrained form of communication we would humanize technology to a great extent. Voice controlled House Automation System deploys the use of voice to control devices.

1.2 Objectives of my Project

There are several objectives to conduct this research, They are:

a) The objective of this project is to implement a low-cost home automation system.

b) Reliable home automation system that can be used to remotely switch on or off any household device using a microcontroller.

c) The end goal beyond this project would be a product that would hopefully allow people to connect to many other home devices through wireless device.

1.3 Motivation

In 2016, the CEO of Facebook, Mark Zuckerberg has built an Artificial Intelligence (AI) voice-controlled assistant for his home. His inspiration came from the character "Jarvis" from the movie called Iron-Man. He has described that it is like a digital butler who can speak, play music, control lights and toasters. It can also say who is at the door. This project has made a movie character almost come to real. This excellent work of his has inspired us to do the project of home automation . There are many smart home appliances like underground refrigerator, smart closet etc. used in few luxurious houses of western countries. We believe a day will come when almost all houses will be a smart house. This project is a small step to reach this goal.

Almost every house has smart phones, smart television, smart watch, smart refrigerator, smart washing machine, smart garage which ultimately leads to a smart home. From there it is predictable that one day almost every house will be a smart house with automatic control system.

1.4 Prerequisites

The background knowledge that were required to complete my project are:

1. Need basic knowledge about Arduino microcontroller.
2. Need to know about c programming language.
3. Basic knowledge about electrical power supply.
4. Basic knowledge about Electromagnetic switch.
5. Need to know about Bluetooth Communication & AMR audio codec.

1.5Proposed System

This wireless Arduino based system includes controlling of home appliances like light, fan, air conditioner, television etc. The project is composed of:

1. Controlling of appliances like light, table fan, TV, air conditioner, curtains will be controlled with Android phone through Bluetooth communication using Bluetooth module. This is used inside the house only.
2. Arduino Uno R3 in Windows operating system.
3. HC – 05 Bluetooth Module will transmit and receive data between Arduino and smartphone
4. A relay switch for the appliances.
5. Finally all appliances are controlled by Voice Application and Android Application through Android mobile phone.

1.6 Outline of Chapters

In this chapter we have introduced about objectives, prerequisites and proposed System. These are very necessary of an introductory part. In this chapter we have tried to give a short description of total projects. Next chapter we will discuss about literature review and project backgrounds.

CHAPTER 2

LITERATURE REVIEW AND PROJECT BACKGROUNDS

2.1 Literature Review

Home automation or smart homes can be described as introduction of technology within the home environment to provide convenience, comfort, security and energy efficiency to its occupants [8].

There are many other projects done on home automation in different countries. They are all different from each other in designs; features, devices, elements and algorithm. They were designed according to specific needs and availability of components in the respective areas. Some of them are cheap; some of them are very expensive. Availability of both hardware and software is necessary to work. After a long searching, we have found a lot of articles. Searching for security purpose articles, we also found some projects done for garage security. These are mainly done in western countries. Many projects are done only for security purpose with Arduino or Raspberry Pi. Again, the projects are done only for controlling home appliances using Arduino or Raspberry Pi.

Face recognition is another excellent and smart way that serves security purpose. We have found projects for door security using face recognition using Raspberry Pi. We avoided this part for the security purpose because error occurs more in face recognition than fingerprint recognition. Fingerprint has high accuracy [9]. They didn't explicitly mention about the security purpose or Raspberry Pi. They have only mentioned about the techniques of recognition. Different people have described the procedure of recognition in different ways. Basically all of them have tried to minimize errors for computer to recognize face [10] [11].

Three researchers of Malaysia proposed a web-based indoor air quality system with GSM and Arduino. The system consists of gas sensor, temperature and humidity sensor, particle dust sensor and wireless sensor network (WSN) node as a wireless transmitter. A desktop computer acts as the base station [12].
According to Chen Shih-Chung, the systems proposed by him is designed that can be easily be adapted for various applications such as control of machines in machining industries, automotive industry, navigating mobile wireless nodes, automating offices etc. [13].

Andrew, the writer of the book "Raspberry Pi Home Automation with Arduino", introduced Raspberry Pi and hoe to use it for home automation. He described the use of Raspberry Pi with Arduino for Linux operating system. The book describes some home appliances automatic control. First, he described how to install all the necessary equipment and all required conditions. Firstly, he gave the history of Arduino and Raspberry Pi with all sockets, required shield specifications and all necessary ports with power supply. We were able to find necessary data of Arduino since were used it. Good examples of thermometer, opening and closing of curtain based on light and temperature data are given. On the other hand, he didn't show any example related to security of home [14].

Annan Zhu, Peijie Lin and Shuying Cheng of Fuzhou University of China described the remote control system of home appliances using android phone through GSM network (2012

International Conference on Control Engineering and Communication Technology).They focused on the design of Android terminal, the communication between ARM and GSM module. Minimizing the difficulty in supplying the appropriate low-voltage DC for MCU and wireless module by a single live wire was also one of the tasks. Here we have found only the controlling of appliances using android, nothing more than that [15].

An article of Singapore by the authors Thomas Gonnot, Won-Jae Yi, Ehsan Monsef and Jafar Saniie showed a protocol standard for home automation system called Home Automation Device Protocol (HADP). Wi-Fi, Bluetooth 4.2, ZigBee IP, 6LoWPAN, IEEE 802.15.4 standards, and Ethernet network layer supporting IPv6 protocol were their components. Mainly they proposed a protocol if-this-then-that. So it connected many devices together using WIFI connection [16].

K. M. Abubeker, Jose J Edathala, Shinto Sebastian from India introduced PIR sensors and an intelligent power saving mode in ATM counter. This uses pyro-electric infrared sensors to detect pedestrians and the ATM users. The system is controlled by the real time clock RTC DS 1307 to differentiate the day and night time with a surveillance video. This gives an excellent security to the ATM counter [17].

Again, there is another article to prevent theft in home by P. Satya Ravi Teja, V. Kushal, A. SaiSrikar titled “Photosensitive security system for theft detection and control using GSM technology”. They did it using LDR (Light Dependent Resistor) based sensor which acts as an electronic eye for detecting the theft or attempt, and a signaling procedure based on SMS using GSM (Global Systems for Mobile communications) technology. It is also quiet cheap [18].

These are the few previous researches done on similar topic. It is mentioned earlier that most of them lack either the security system or the controlling system. We avoided the face recognition system for home security because people are trying to minimize a lot of error in recognition of face. The face has to be at a particular angle so that the computer is able to recognize. Therefore, fingerprint recognition module is more reliable for door security. Some of these projects are done with Arduino, some of them are done with Raspberry Pi. The components, like sensors and shields are also of different models. Our aim is to combine those systems together i.e. controlling home appliances and security system with Arduino keeping it as cheap as possible.

2.2 Project Backgrounds

The concept of home automation is actually quite old. Science fiction from the first half of the 20th century dealt heavily with idea of home automation, and it has become a popular trope within Sci-Fi movies, books, TV shows, and artwork. It's in the realm of fiction that most people probably first encountered home automation, and it can often feel bizarre to see it manifesting as a reality. The truth is, however, is that people have been working to make smart homes a reality almost since the very first appearance of smart homes in the realm of fiction [4]. The technology that allows for home automation, and wireless automation in particular, came as a result of Nikola Tesla, who invented the first wireless remote control as far back as 1898, which used radio waves to control a small boat. This, coupled with the rise of home appliances in the early 1900s set the stage for what would later become the potential of automated home systems. The first attempt to create a home automation system was completed in the 1930s, but the technology wasn't quite ready, and the product was never commercialized. As technology progressed, things began to become more and more integrated in the field of electronics, with innovations such as the microprocessor allowing for unprecedented speed in devices of relatively small size. In 1998 the first proof-of-concept smart home was unveiled, proving to the world that a home that was fully integrated with wireless automation was possible. This technology has only continued to progress, and more and more aspects of the home have become available for total integration. In recent years smart home technology has seen a massive increase in popularity and commercial success. This is largely due to the spread of wireless automation that works through Wi-fi and cell data, as well as the increase in the number of people who have personal devices.

With modern technology it's now possible to control your lighting, watering systems, security, electrical usage, and much more, all from your personal device, whether or not you are even at home. From being a common in Science Fiction, to being a heralded possibility, to a full-fledged reality, home automation is an exciting new branch of technology that promises to make our lives easier, more efficient, and more interesting by the day.

CHAPTER 3

BRIEF INTRODUCTION OF ALL THE COMPONENTS

3.1 Introduction

Before beginning the project It is very important to know all the information about both hardware and software specifications. The components we are using are as follows:

3.1.1 Hardware Requirements

a. Arduino Uno R3 microcontroller
b. HC – 05 Bluetooth Module
c. 10 KΩ Resistor
d. 1 KΩ Resistor X 4
e. BC547 NPN Transistor X 4 — 5V 4-channel relay
f. 1N4007 Diode X 4
g. 5 V Relay X 4
h. Prototyping board (Bread board)
i. Connecting wires
j. External Power supply
k. Smartphone or tablet (Bluetooth enabled)

3.1.2 Software Requirements

a. Arduino IDE
b. Proteus 8
c. Android application

3.2 Arduino UNO

The Arduino UNO is an open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino.cc. The board is equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards (shields) and other circuits. The board has 14 Digital pins, 6 Analog pins, and programmable with the Arduino IDE (Integrated Development Environment) via a type B USB cable. It can be powered by a USB cable or by an external 9 volt battery, though it accepts voltages between 7 and 20 volts. It is also similar to the Arduino Nano and Leonardo. The hardware reference design is distributed under a Creative Commons Attribution Share-Alike 2.5 license and is available on the Arduino website. "Uno" means one in Italian and was chosen to mark the release of Arduino Software (IDE) 1.0. The Uno board and version 1.0 of Arduino Software (IDE) were the reference versions of Arduino, now evolved to newer releases. The Uno board

is the first in a series of USB Arduino boards, and the reference model for the Arduino platform. The ATmega328 on the Arduino Uno comes preprogrammed with a bootloader that allows uploading new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol [19].

The Uno also differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it uses the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter [19].

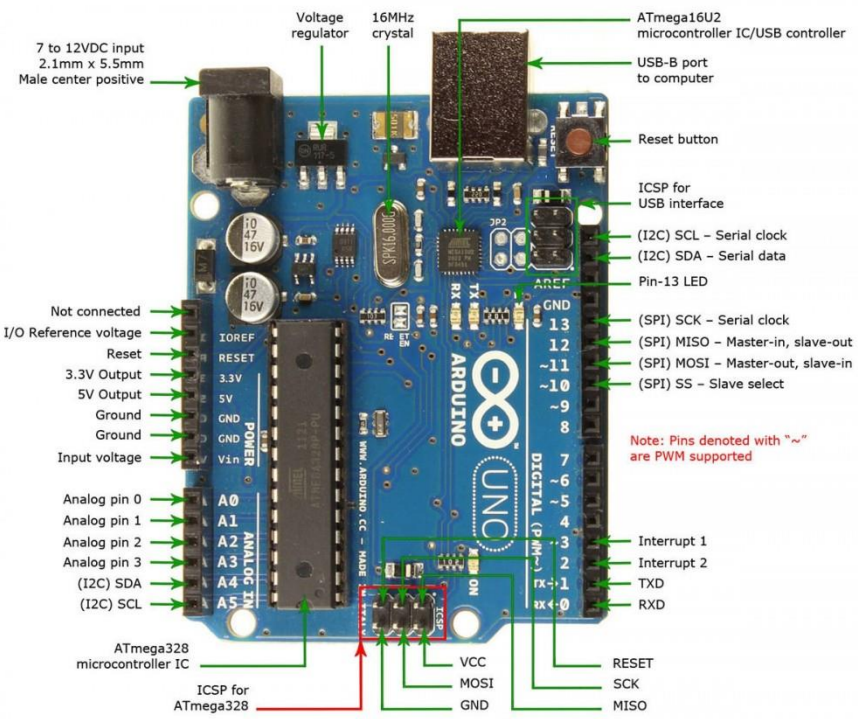

Figure 3.1: Arduino UNO R3 with pinouts [20].

Table 3.1: Technical Specification of Arduino Uno [21].

Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 Ma
DC Current for 3.3V Pin	50mA
Flash Memory	32 KB of which 0.5KB
SRAM	2 KB
EEPROM	1 KB
Clock Speed	16 MHz
Length	68.6 mm
Weight	25g

3.3 Introduction of the modules

Brief Introductions of communicating modules are given below.

3.3.1 HC – 05 Bluetooth Module

The HC-05 is a very cool module which can add two-way (full-duplex) wireless functionality to our projects. we can use this module to communicate between two microcontrollers like Arduino or communicate with any device with Bluetooth functionality like a Phone or Laptop. There are many android applications that are already available which makes this process a lot easier. The module communicates with the help of USART at 9600 baud rate hence it is easy to interface with any microcontroller that supports USART. We can also configure the default values of the module by using the command mode. So, if you looking for a Wireless module that could transfer data from your computer or mobile phone to microcontroller or vice versa then this module might be the right choice for you. However, do not expect this module to transfer multimedia like photos or songs; you might have to look into the CSR8645 module for that [22].

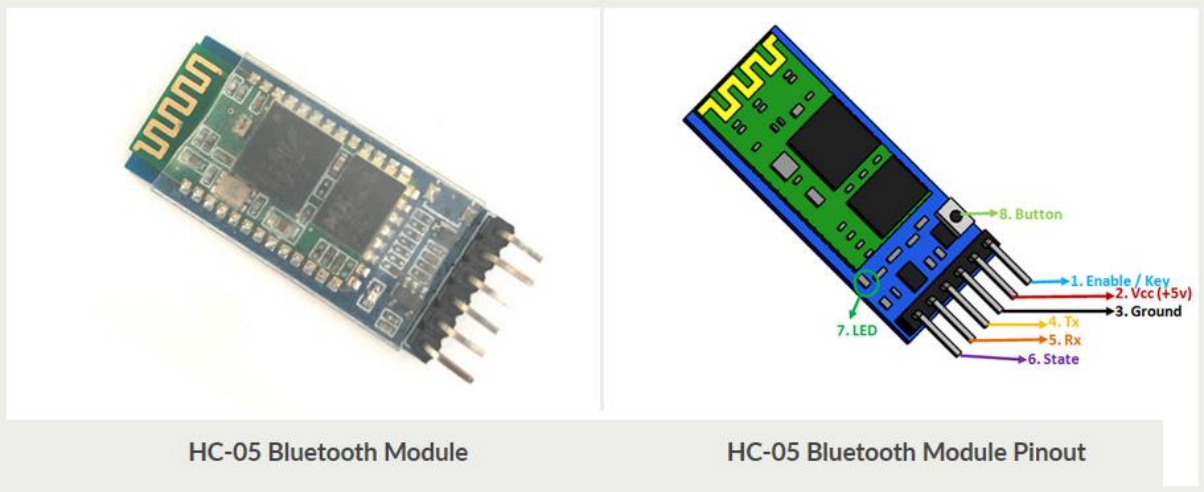

Figure 3.2: HC – 05 Bluetooth Module [22].

HC-05 Technical Specifications

- Serial Bluetooth module for Arduino and other microcontrollers
- Operating Voltage: 4V to 6V (Typically +5V)
- Operating Current: 30mA
- Range: <100m
- Works with Serial communication (USART) and TTL compatible
- Follows IEEE 802.15.1 standardized protocol
- Uses Frequency-Hopping Spread spectrum (FHSS)
- Can operate in Master, Slave or Master/Slave mode
- Can be easily interfaced with Laptop or Mobile phones with Bluetooth
- Supported baud rate: 9600,19200,38400,57600,115200,230400,460800.

3.3.2 Relay Module 5V 4-channel

Relays are electromechanical devices that use an electromagnet to operate a pair of movable contacts from an open position to a closed position. The advantage of relays is that it takes a relatively small amount of power to operate the relay coil, but the relay itself can be used to control motors, heaters, lamps or AC circuits which themselves can draw a lot more electrical power. The design and types of relay switching circuits is huge, but many small electronic projects use transistors and MOSFETs as their main switching device as the transistor can provide fast DC switching (ON-OFF) control of the relay coil from a variety of input sources so here is a small collection of some of the more common ways of switching relays [23].

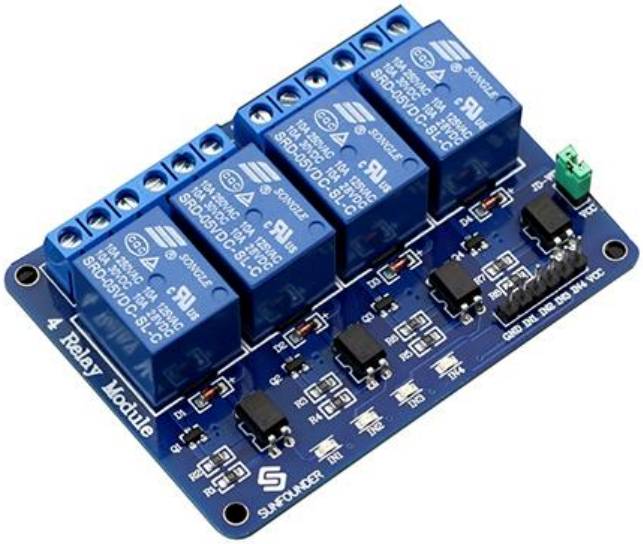

Figure 3.3: A 5V 4-channel relay module [24].

It is a 5V 4-channel relay board used to control various appliances. It can be used with or without microcontrollers. Each 5V relay needs 20mA driving current. It has LEDs for indication of output status. From the picture below, you can see that when the signal port is at low level, the signal light will light up and the optocoupler 817c (it transforms electrical signals by light and can isolate input and output electrical signals) will conduct, and then the transistor will conduct, the relay coil will be electrified, and the normally open contact of the relay will be closed. When the signal port is at high level, the normally closed contact of the relay will be closed. So you can connect and disconnect the load by controlling the level of the control signal port [24].

Pin Description

Input:

VCC: Positive supply voltage
GND: Ground
IN1--IN4: Relay control port

Output: Connect a load, DC 30V/10A, AC 250V/10A

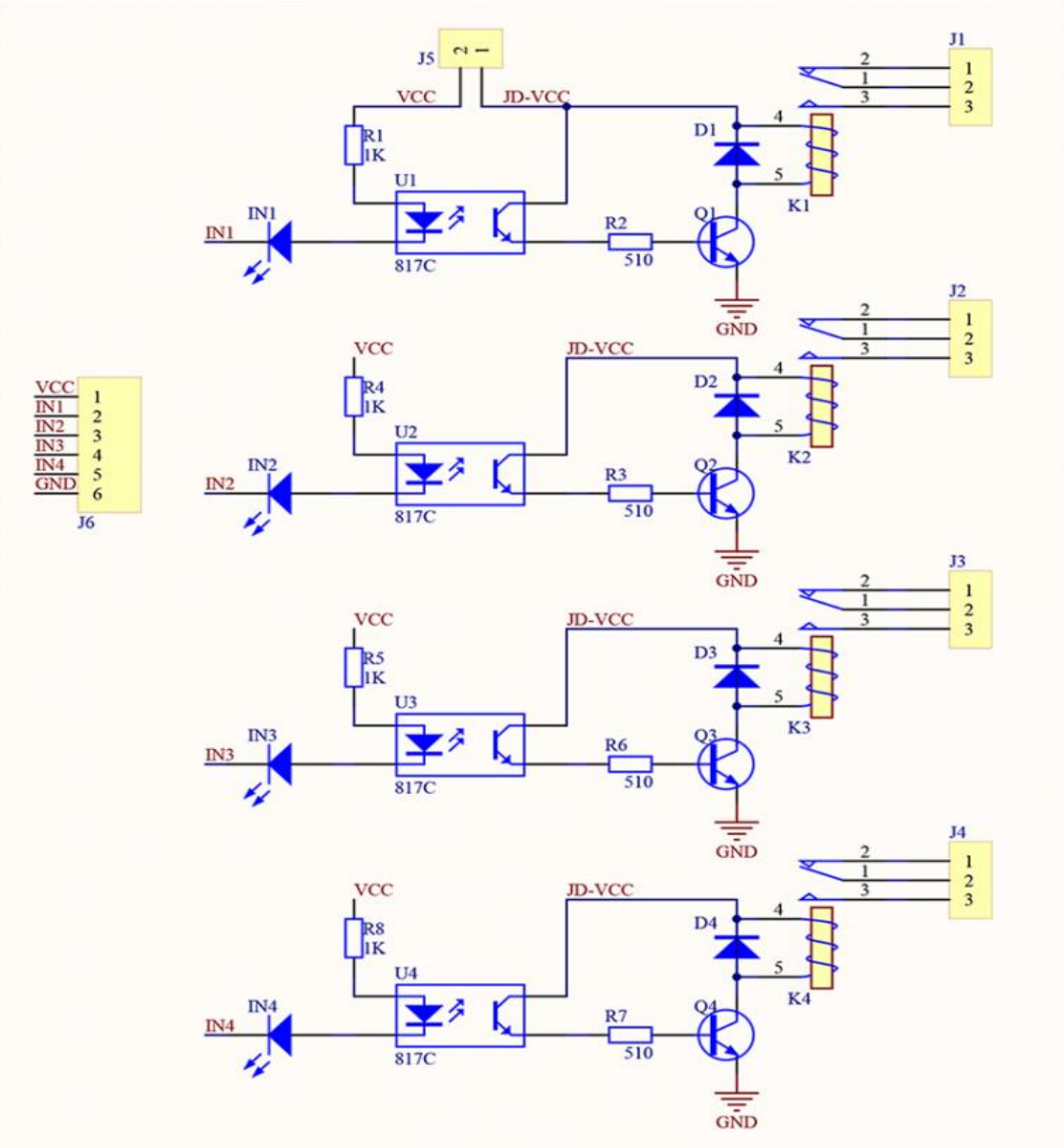

Figure 3.4: Schematic of 4-Channel Relay module [24].

3.4 Introduction to the Arduino IDE

The Arduino is a fantastic single-board microcontroller solution for many DIY projects, and, in this blog, we will look at the Integrated Development Environment, or IDE, that is used to program it! First, you must download the IDE and install it. Start by visiting Arduino's software page. The IDE is available for most common operating systems, including Windows, Mac OS X, and Linux, so be sure to download the correct version for your OS. If you are using Windows 7 or older, do not download the Windows app version, as this requires Windows 8.1 or Windows 10. Once the installer has downloaded, go ahead and install the IDE [25].

3.4.1 The Arduino IDE

Projects made using the Arduino are called sketches, and such sketches are usually written in a cut-down version of C++ (a number of C++ features are not included). Because programming a microcontroller is somewhat different from programming a computer, there are a number of device-specific libraries (e.g., changing pin modes, output data on pins, reading analog values, and timers). This sometimes confuses users who think Arduino is programmed in an "Arduino language." However, the Arduino is, in fact, programmed in C++. It just uses unique libraries for the device.

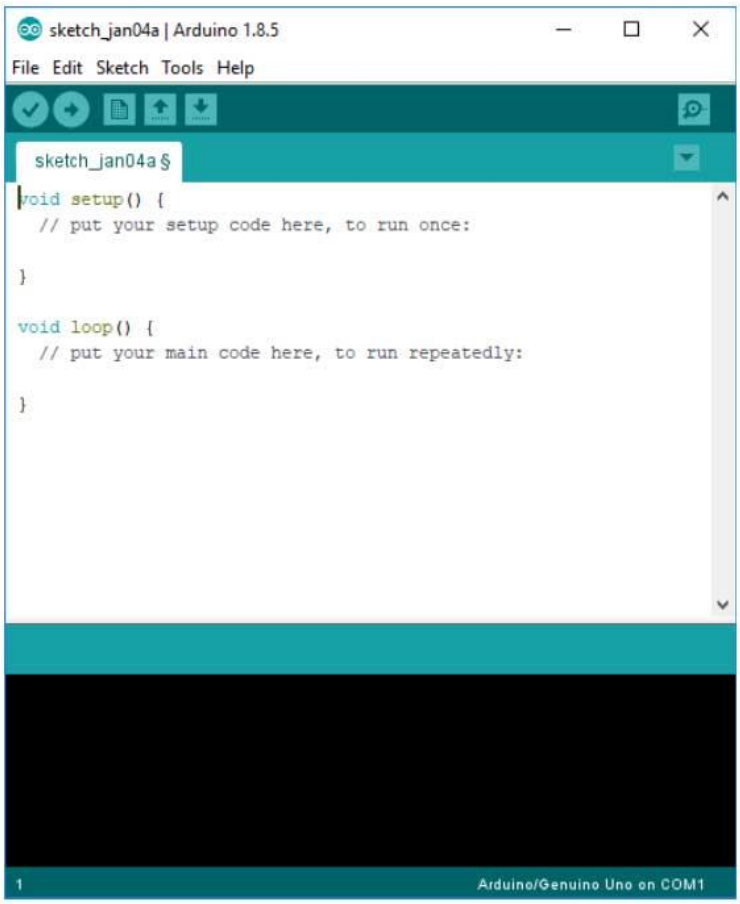

Figure 3.5: The Arduino IDE in its default state [25].

The 6 Buttons

Figure 3.6: The Button Bar [25].

1. The check mark is used to verify your written code.
2. The arrow sign uploads your code to the Arduino.
3. The third option is used to create a new file.
4. The upward arrow is used to open an existing Arduino project.
5. The downward arrow is used to save the current file.
6. The far right button is a serial monitor, which is useful for sending data from the Arduino to the PC for debugging purposes.

There are plenty of other features available to consider on the IDE. But, having used many different types of microcontrollers and having been involved in multiple programming environments, it is shocking how simple the Arduino and its IDE is! In less than two minutes, you can get a simple C++ program uploaded onto the Arduino and have it running.

3.4.2 Proteus 8

Proteus is a simulation and design software tool developed by Lab center Electronics for Electrical and Electronic circuit design. It also possesses 2D CAD drawing feature. It deserves to bear the tagline “From concept to completion”.

About Proteus

It is a software suite containing schematic, simulation as well as PCB designing.

- ISIS is the software used to draw schematics and simulate the circuits in real time. The simulation allows human access during run time, thus providing real time simulation.
- ARES is used for PCB designing. It has the feature of viewing output in 3D view of the designed PCB along with components.
- The designer can also develop 2D drawings for the product.

Features

ISIS has wide range of components in its library. It has sources, signal generators, measurement and analysis tools like oscilloscope, voltmeter, ammeter etc., probes for real time monitoring of the parameters of the circuit, switches, displays, loads like motors and lamps,

discrete components like resistors, capacitors, inductors, transformers, digital and analog Integrated circuits, semi-conductor switches, relays, microcontrollers, processors, sensors etc. ARES offers PCB designing up to 14 inner layers, with surface mount and through hole packages. It is embedded with the foot prints of different category of components like ICs, transistors, headers, connectors and other discrete components. It offers Auto routing and manual routing options to the PCB Designer. The schematic drawn in the ISIS can be directly transferred ARES.

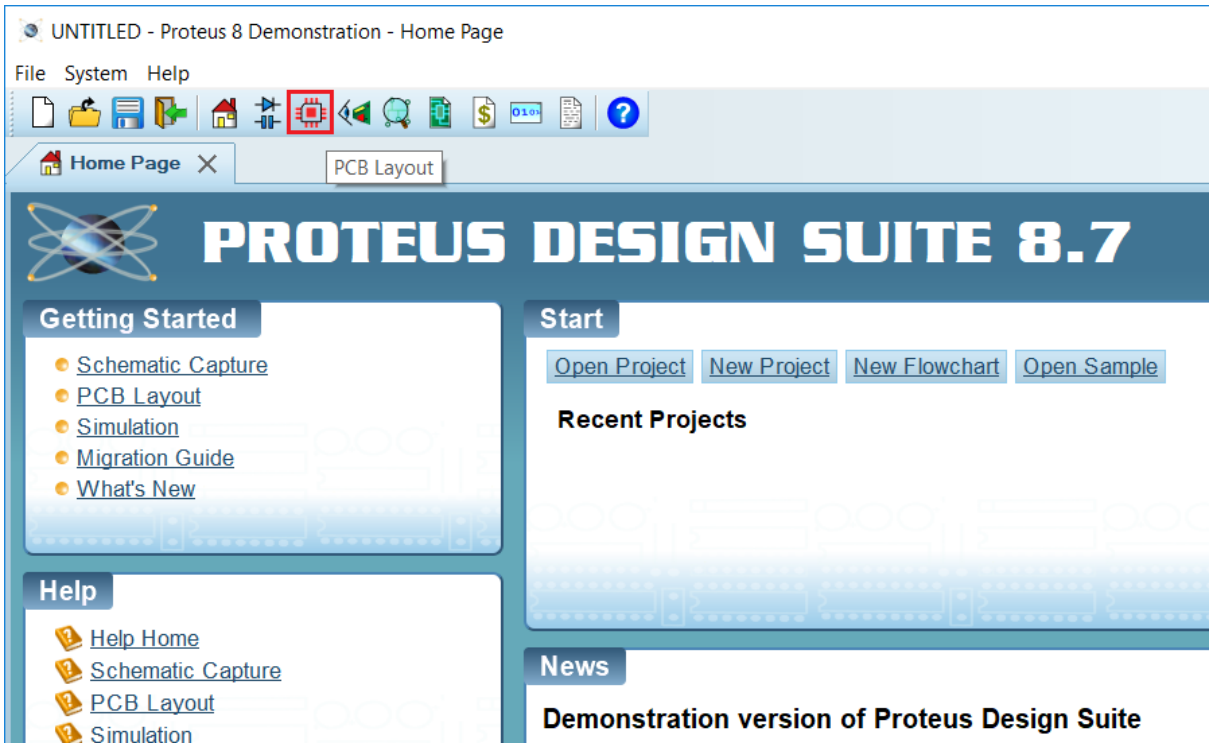

Figure 3.7: Proteus design suite 8.7

CHAPTER 4

METHODOLOGY

4.1 Introduction

We have designed the system structure, According to the proposed system, shown in the block diagram. We have designed the model in such a way that it can be kept at a safe place inside the house. All programming and components installation are done and tested inside the laboratory and in home. There are a lot of components and wires that we have used for the system. This is done in the easiest and lowest cost as possible. However, the system is flexible and can be customized by the user. Changing one of the components setup has to be compatible with the right software available. Every components used in this system was programmed and tested separately for safety measures and matching with the right driver. Each component was programmed separately with Arduino UNO using Arduino IDE. Also they were run in different computers. Later on all were combined in a single Arduino IDE. It is not possible to run the system without computer.

This project is divided into two parts: hardware implementation and software implementation

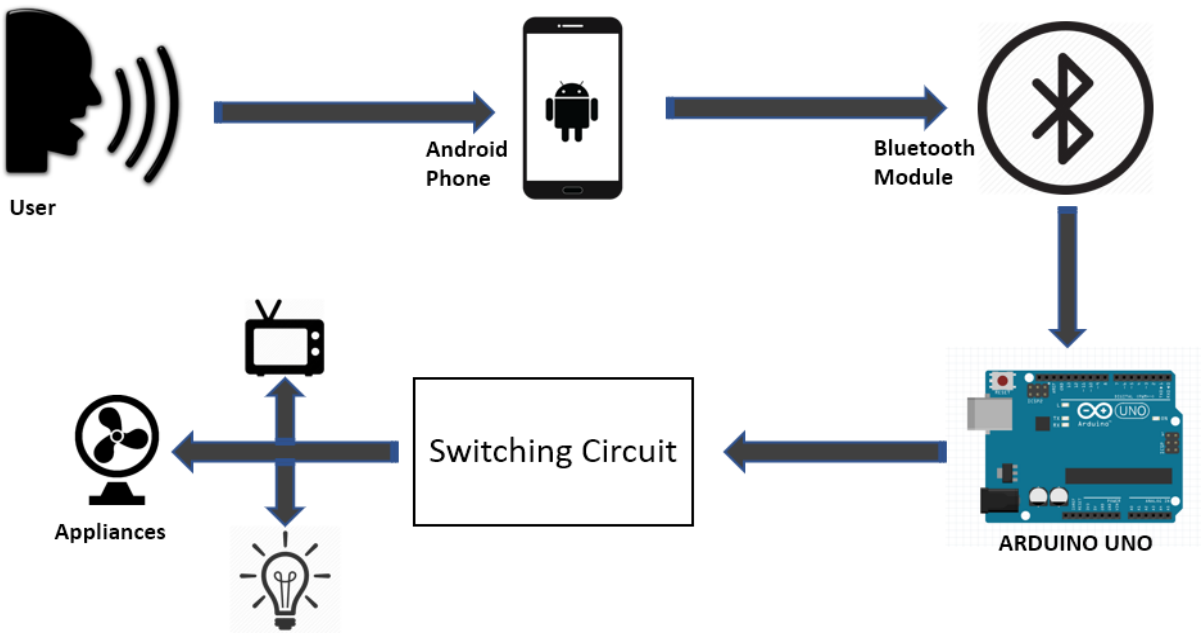

Figure 4.1: Block Diagram of the Home Automation.

4.2 Hardware Architecture & Implementation

In this section, we will discuss how all of the components are interconnected between Arduino and other components.

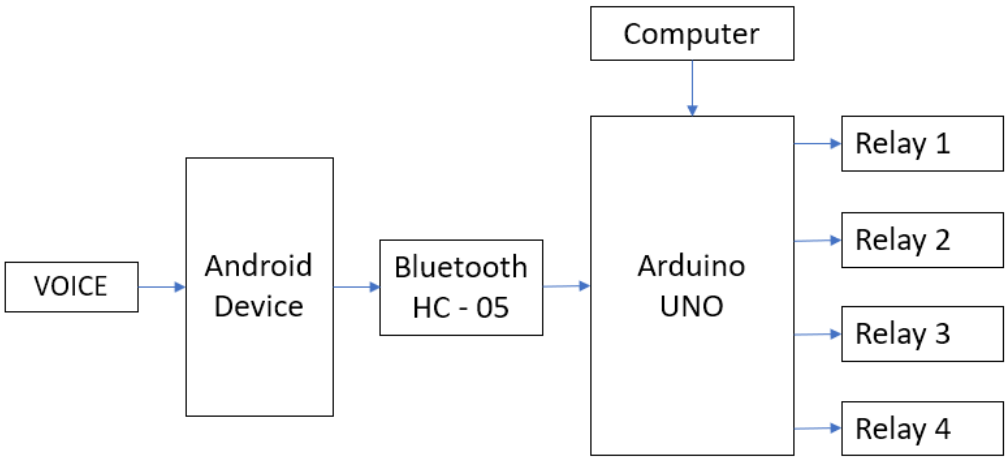

Figure 4.2: Block Diagram of the system.

4.2.1 Modules

All modules are connected with the microcontroller through wires. All input voltages are applied from the microcontroller with the computer. It is also possible to power them externally using 5V AC to DC power adapter. But it is the safest way to power the input module from the Arduino.
They are described below with diagram. In this section hardware implementation of all sensors are described below:

A. HC-05 Bluetooth Module:

Bluetooth module plays a very important role in interfacing the home appliances with the Android phone but it has only four pins for connection. Pin connections are given below:

Table 4.1: Bluetooth module pin interface with Arduino.

Arduino Pins	**Bluetooth Pins**
RX (PIN 0)	TX
TX (PIN 1)	RX
5V	V_{CC}
GND	GND

➢ HC-05 Bluetooth Module Interfacing with Arduino UNO

HC-05 is a Bluetooth device used for wireless communication with Bluetooth enabled devices (like smartphone). It communicates with microcontrollers using serial communication (USART). Default settings of HC-05 Bluetooth module can be changed using certain AT commands. As HC-05 Bluetooth module has 3.3 V level for RX/TX and microcontroller can detect 3.3 V level, so, there is no need to shift TX voltage level of HC-05 module. But we need to shift the transmit voltage level from microcontroller to RX of HC- 05 module.

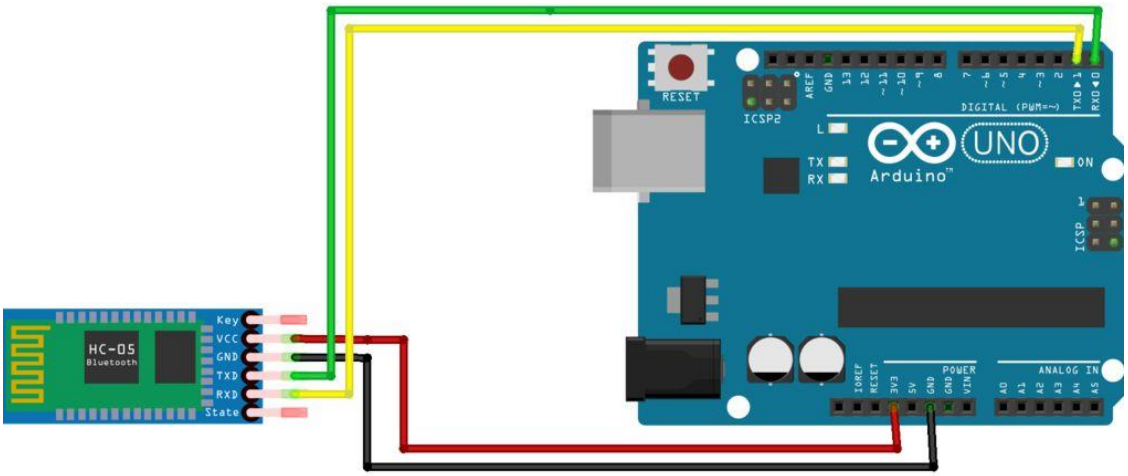

Figure 4.3: Bluetooth Module Interfacing with Arduino UNO [26].

In our system, We will not use Arduino's RX pin as a receiver pin. We set a constant pin to receive signal from Bluetooth module which is pin 2. But its not necessary to use TX pin of the Arduino, because we have nothing to receive from our microcontroller to HC – 05. Although we set a constant TX pin for Arduino Which is pin 3.

So our pin interface will be looked like This:

Table 4.2: Bluetooth Module pin interface in our system.

Arduino Pins	**Bluetooth Pins**
RX (PIN 2)	TX
TX (PIN 3)	RX
5V	V_{CC}
GND	GND

B. Relay Module

All the components are with the relay. The relay is on after all modules are high. In case of turning on light, when the input voice matches the relay is on and the light turned on. In case of all sensors, when they are high the relay turns on to send text message to the phone. Relay, obviously as a switch is connected with all the home appliances. Also controlling of relay is possible through the GSM. We have also controlled the room curtains with the help of relay. The connection is done according to the schematic diagram.

In relation to mains voltage, relays have 3 possible connections:

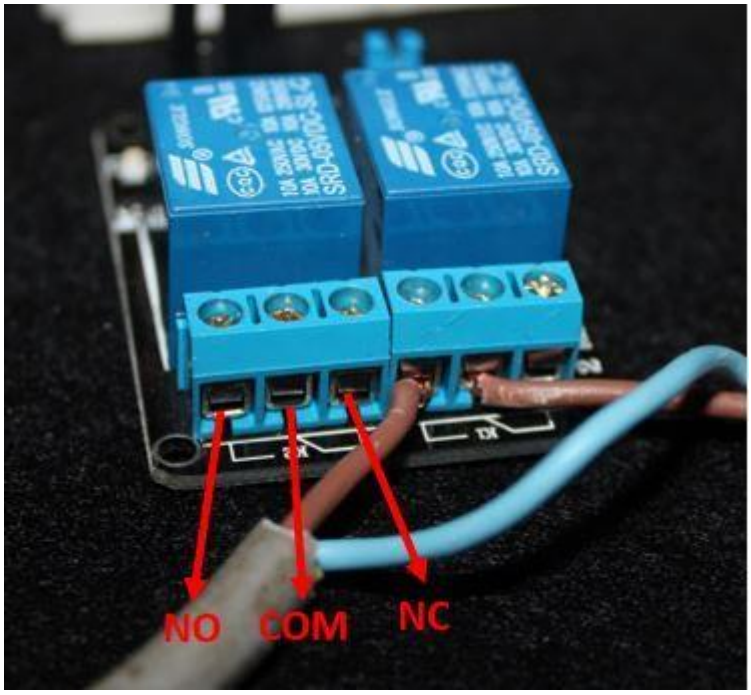

Figure 4.4: Connections in Relay [27].

a) **COM**: common pin
b) **NO (Normally Open):** there is no contact between the common pin and the normally open pin. So, when you trigger the relay, it connects to the COM pin and supply is provided to a load
c) **NC (Normally Closed):** there is contact between the common pin and the normally closed pin. There is always connection between the COM and NC pins, even when the relay is turned off. When you trigger the relay, the circuit is opened and there is no supply provided to a load [27].

➢ 4-Channel Relay Module Interfacing with Arduino UNO

a) Logic GND: This will be connected to GND on your Arduino.
b) Input 1 (IN 1): This will be connected to digital pin on your Arduino, or leave it unconnected if you do not want to use this channel.
c) Input 2 (IN 2): This will be connected to the digital pin on your Arduino, or leave it unconnected if you do not want to use this channel.
d) Input 3 (IN 3): This will be connected to the digital pin on your Arduino, or leave it unconnected if you do not want to use this channel.
e) Input 4 (IN 4): This will be connected to the digital pin on your Arduino, or leave it unconnected if you do not want to use this channel.
f) Logic VCC: This will be connected to the 5v pin of the Arduino o power the 4 relay module [28].

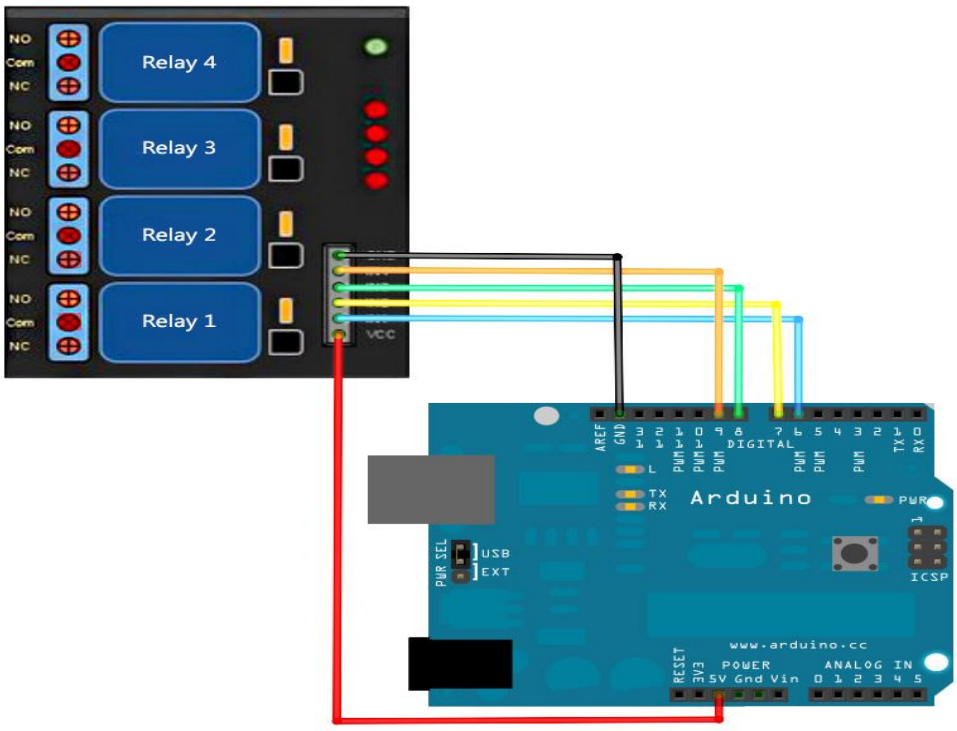

Figure 4.5: 4-Channel Relay Module Interfacing with Arduino UNO [28].

Pin connections for this 4-channel Relay module are given below:

Table 4.3: Relay Module pin interface With Arduino.

Arduino Pins	Relay Module Pins
GND	GND
6	IN1
7	IN2
8	IN3
9	IN4
5V	VCC

But in our System, the pin interfaces are little bit different. Here, we used Arduino's pins 4,5,6,7 instead of 6,7,8,9. The Bluetooth module pin interfaces with Arduino in our system will be looked like following table.

Table 4.4: Relay Module pin interface With Arduino.

Arduino Pins	Relay Module Pins
GND	GND
4	IN1
5	IN2
6	IN3
7	IN4
5V	VCC

We can also power the 4-relay module using the external power by giving voltage from 5 to 24V DC. But in this case we need to make sure Which relay module we are using. It is possible to use a AC to DC power adapter or DC source to power the Relay switch.

Table 4.5: All Module's Pins with Arduino UNO

SL	Modules	Arduino UNO
1	Bluetooth's RX	4
2	Bluetooth's TX	2
3	Relay 1	4
4	Relay 2	5
5	Relay 3	6
6	Relay 4	7
7	Bluetooth, Relay (GND)	GND
8	Bluetooth, Relay (VCC)	5V

There we can use AC to DC power adapter, Or 5V DC source for the Arduino. Bluetooth module and Relay Module get their power respectively from the Arduino. We can also use AC power source to the Relay switch, But we will use DC source for my Project safety. message will be sent or received. To maintain a stable condition these extra modules are. High or low voltage and current configuration can damage the whole system. If there is any chance of deducting power supply from the system, the whole system will collapse. To avoid such problem battery and adapter are used. Therefore, there will be no problem in the security system

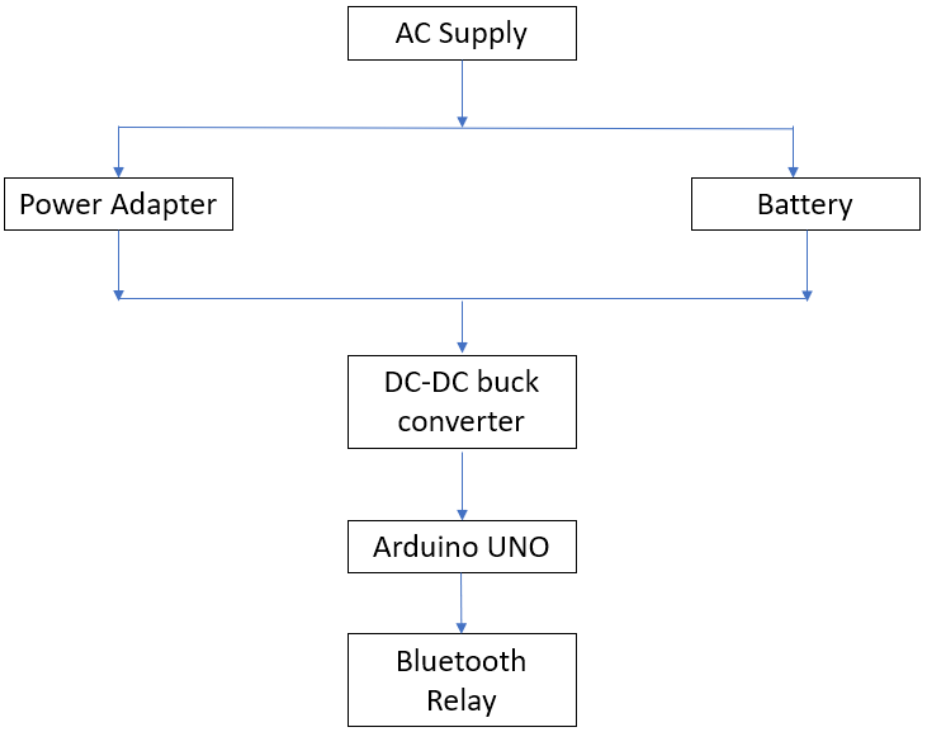

Figure 4.6: Power supply block diagram of the home automation system.

4.3 Software Implementation

The software we used is Arduino IDE 1.8.7. All code is written in a single IDE called sketch. All the components are Arduino compatible so we have included respective Arduino Adafruit Library. For this section we have also divided the working procedure into three parts: modules, Android Application and Voice controller.

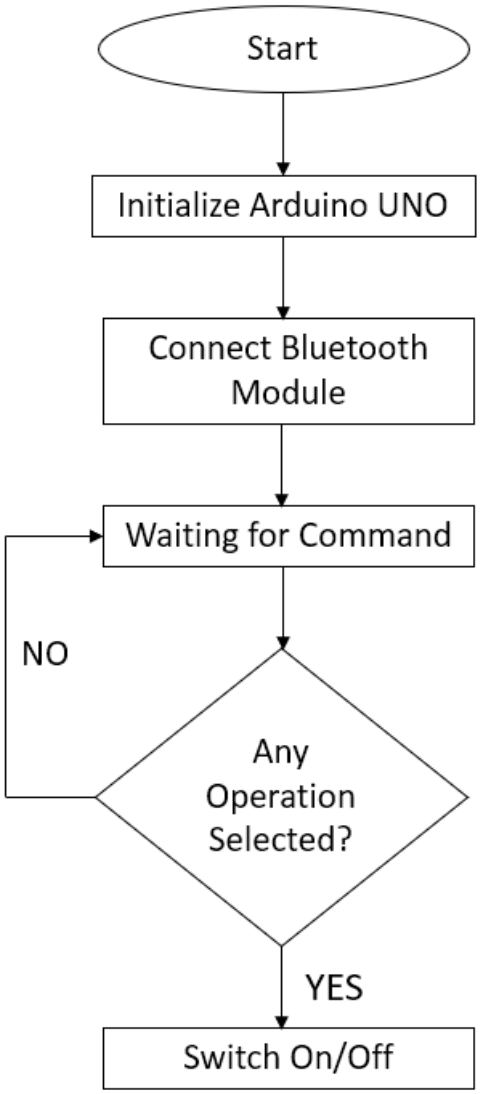

Figure 4.7: Flowchart of the home automation system.

4.3.1 Modules

We have mentioned the names of the other modules of the system earlier. Their software implementations are described below:

A. HC – 05 Bluetooth Module

The default baud rate for AT mode is 38400. First we have to manually enter the AT mode. It keeps reading data after entering. The Bluetooth module relays the command and sends it to the Arduino UNO. Determining communication speed we used the function setup( ) in place of begin( ) and findBaud(b). To send a command to the module the function cmd( ) is used. The key (cmdPin) pin is activated to put the module in command mode where 'AT' commands are recognized. The default functions of mode start and end are used for the speed 38400. In our system, we control the home appliances through Android Application via Bluetooth. The Android App interfaces with microcontroller via Bluetooth. Switching of home appliances is its main function. In this system we only give command from the Android to the system. Both the Voice Control Application and Android Application are used to control home appliances. No command is returned through the Bluetooth.

B. 4 Channel Relay

There is no programming for the relay. It is a part of hardware connection only. It just get active or inactive using digitalWrite(); function.

4.3.2 Android Application

In this system we have the Android application to control all the home appliances. From Android phone we select any home appliance from the options that appear in the App then we select ON or OFF. This can be done only when the user is inside the house. There should be Wi-Fi connection for the App. The user cannot run the App from outside the house even if there is internet connection in the phone. It is related with the Bluetooth module. It allows establishing point-to-point connection with Bluetooth support devices. This technology is known by Android's support for the Bluetooth network stack which permits to exchange data wirelessly.

The Android Software Development Kit (SDK) provides all necessary tools to develop Android Application (API). This application is a Java based program. The Android uses .apk file to install the application. The code is written in Android Studio IDE. All appliances buttons list will appear first. Then the user has to choose an option. Later the action button ON and OFF appears. There are 2 layouts of the code structure, two Class code and user permission code. These are written in Android Studio IDE. The code is written according to the appearance of the options in the phone [29].

To open device lists:

importandroid.widget.Button;

importandroid.widget.ListView;

To create variables for Bluetooth:

privateBluetoothAdaptermyBluetooth = null;

privateSet.pairedDevice;

After initialization, methods of Java is written ending with user permission code. Then the code is written in Arduino IDE. The code starts with initializing characters as 'String' [30].

4.3.3 Voice Controlling Android Application

In this project we have also used Android Voice Application to control the home appliances. All home appliances can be turned ON or OFF by sending voice command. We are not calling it a Voice Recognition system because this App allows everyone to speak. This is kept easy and flexible to for the user that can used with any microcontroller. The system receives voice command from anyone. The Google Assistant Voice Application receives the sound wave as a "String" through paired Bluetooth Serial Modules and converts it into text. Then the command is processed for the relay to work.

For example, if we press the microphone icon and say "turn on light", the app will recognize the command and the transfers it to the Bluetooth Module. Also, the command gets displayed on the screen for our reference.

When the string "turn on light" is detected by the app, it will send the string as "*turn on light#". So, the actual message received by the Bluetooth Module is in the format of "*Message#". The reason for padding the '*' and '#' at the begging and end of the string is to identify the starting and ending of the message.

We are able to delete the '#' from the string but left out the '*' in order to identify the starting of the string. The received message is compared with some predefined strings and if the message matches with any of them, then corresponding action like turning on or turning off the load happens.

We have used the following commands: "turn on AC", "turn off AC", "turn on light", "turn off light", "turn on TV", "turn off TV", "turn on fan", "turn off fan", "turn on all" and "turn off all".

CHAPTER 5

RESULT AND ANALYSIS

5.1 Results

After connecting and programming all the components with the, we conducted the experiment. We have run all the components according to the proposed system. We have designed a prototype of a house placing inside room and outside door. All modules and microcontrollers are kept together with a lot of wires. This part is the main center of the home automation system. The sensors are placed inside the room (Figure 5.1). All commands are given from Android App and voice command.

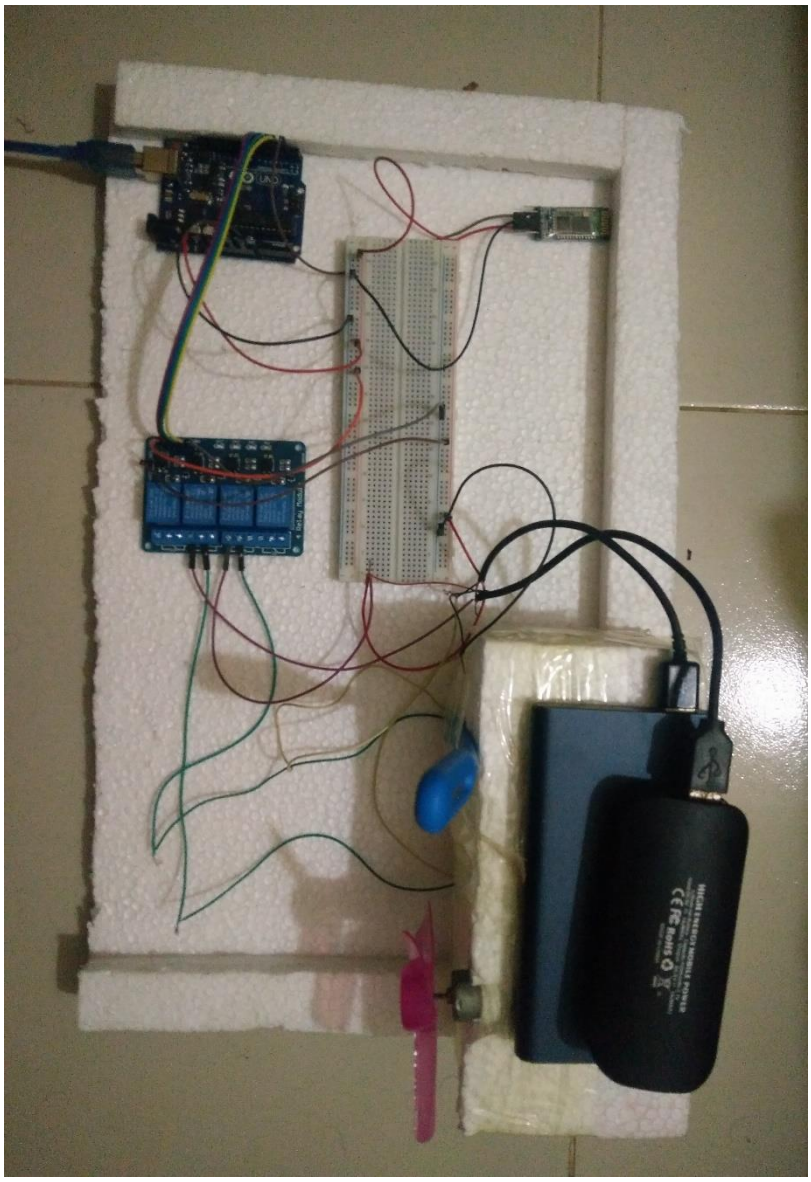

Figure 5.1: Top view of the home automation system showing different modules.

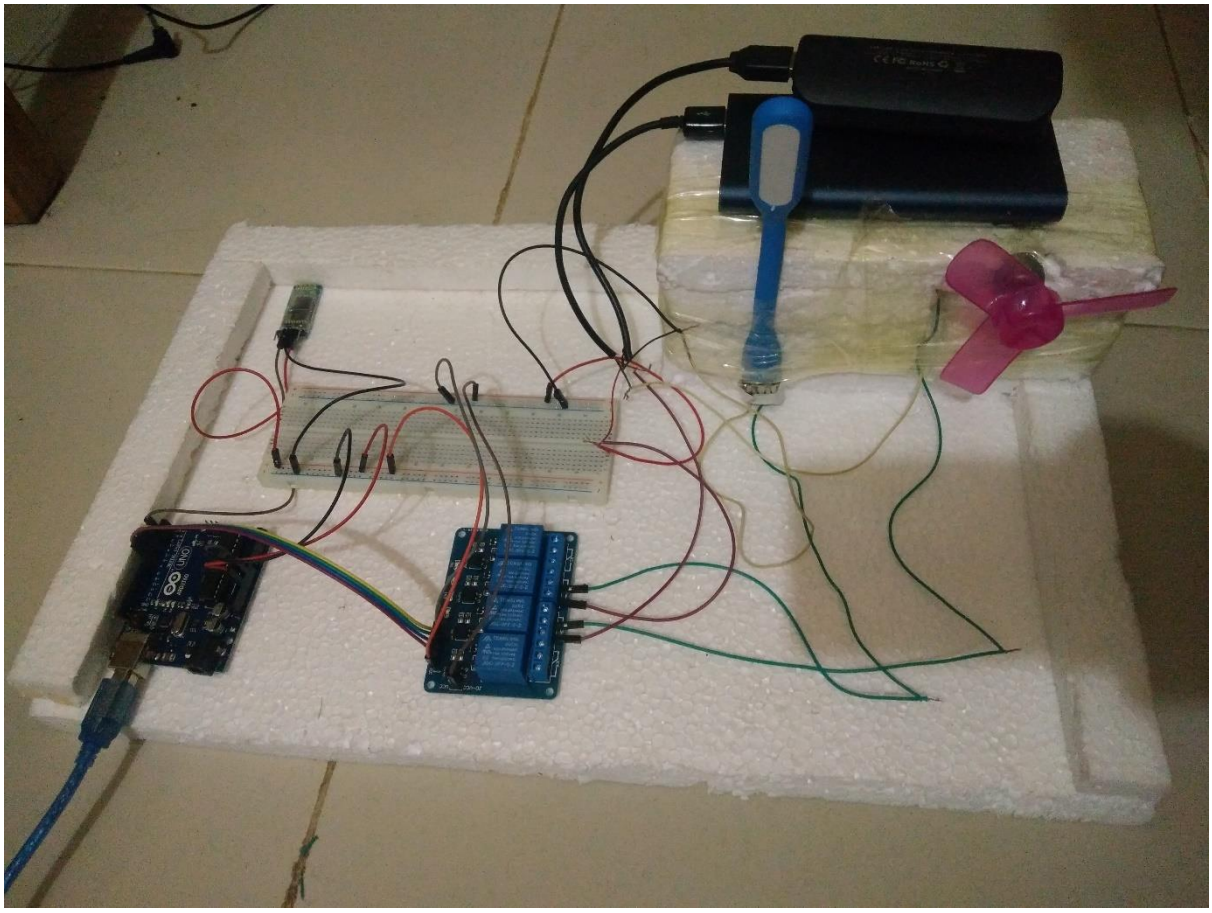

Figure 5.2: Front view of Designed Project

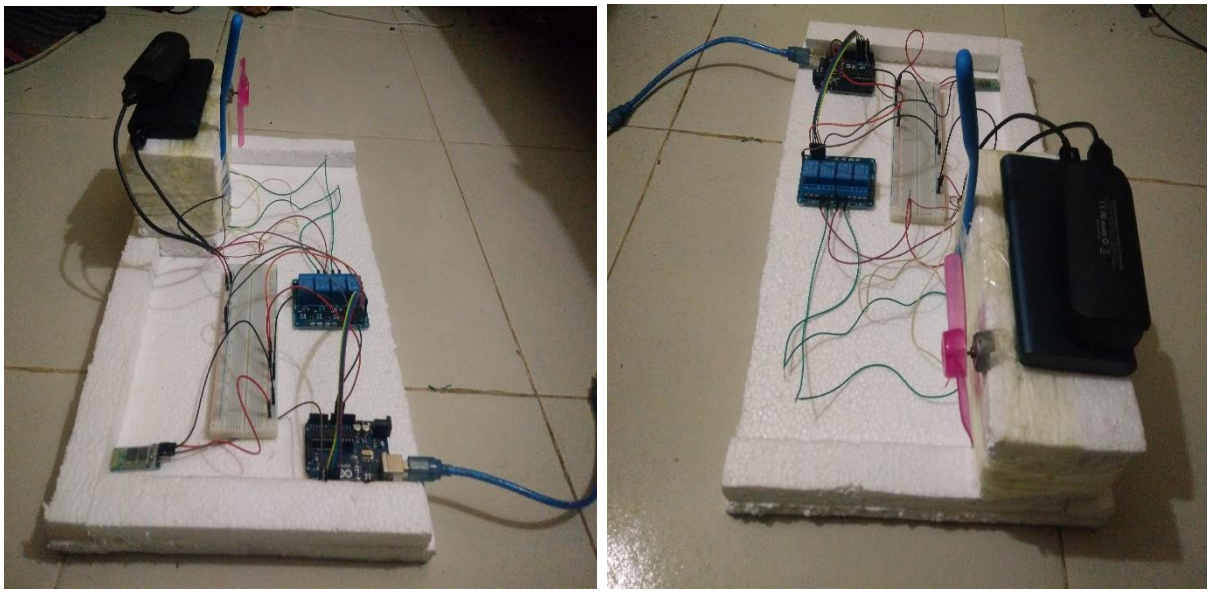

Figure 5.3: Side views of Designed Project

Fan is turned ON and OFF using voice command (Figure 5.2).

(a) (b)

Figure 5.4: (a) Fan OFF. (b) Fan ON.

Light is turned ON and OFF using voice command (Figure 5.3).

(a) (b)

Figure 5.5: (a) Light OFF. (b) Light ON.

project is controlling appliances with Bluetooth and Voice App. We have used specific characters for switching on and off the light, fan and other home appliances. The mobile receives the command as string with start and stop bits indicating. It takes around 2 seconds to process the command and work.

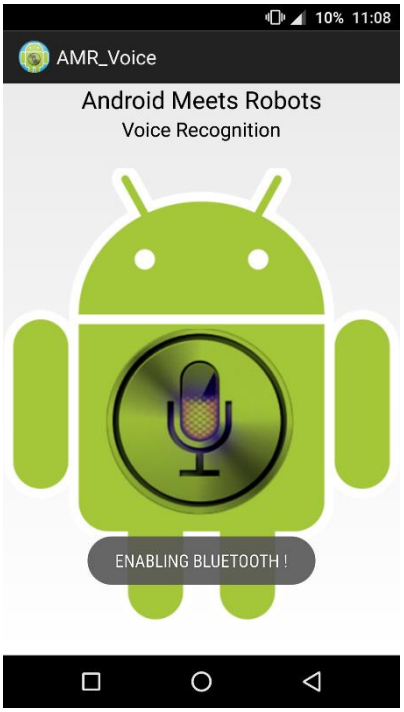

Figure 5.6: Initializing Voice Recognition App.

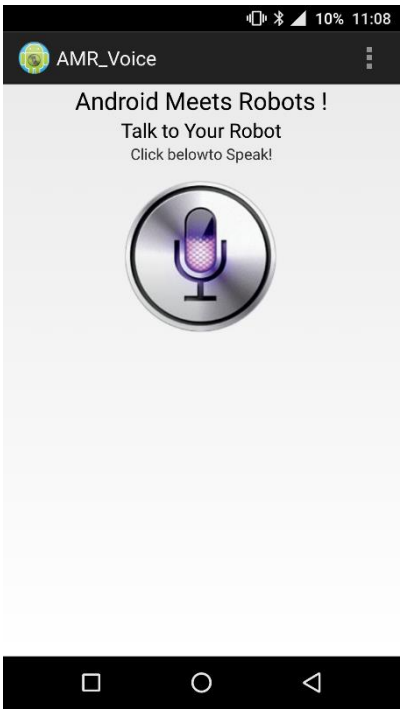

Figure 5.7: Initializing Voice Recognition App.

5.2 Analysis

This is not a complete project. This is a prototype of another larger system for lager house. There are a lot of important matters to observe in this project. After performing all the tasks we have seen that the voltage and current is not the same always as given in the components' specifications. We have used a lot of devices that that need high and constant supply. Otherwise there is delay in the task. There is also a risk of destroying the devices if there is very high voltage supply.

The number of wires that we have used in this project doesn't make significant power loss but the modules need constant power supply. To maintain a constant power supply, we had to add DC-DC buck converter and an adapter. There will be a problem if there is no electricity. In processing voice command, it has maximum 4 seconds delay.

The whole program is written in a single Arduino IDE so it is very easy for the user to change any kind of function. Overall the use of this automation system is easy, flexible and reliable. We can easily add extra features with system. If we analyze about the expenses, this system is very efficient and reliable.

Our expenses only for the security system of the house in this project:

Table 5.1: Expense of the security system.

Components	**Price**
Arduino UNO R3	TK 520
HC – 05 Bluetooth Module	TK 350
Prototyping Board	TK 180
Relay	TK 410
Total	**Tk 1460**

CHAPTER 6

CONCLUSION

6.1 Discussion

From the project carried out, we find the system effectively low cost and user friendly. The whole house remains under the user's control all the time. In future we may find some devices that are more reliable, faster and cheaper. We have tried to make a good controlling and security system. The components that we have used can be changed with the latest device but it should have the right software and the right driver.

All the tasks of this project are done successfully. We were able to fulfill our goals as proposed in this system. We had our limitations in time and expenses but we hope that it will serve as basis of other latest AI systems as that of western countries.

Almost all scientific and latest technologies have both good and bad sides. That doesn't mean we should avoid technology. This type of work inspires us to do better for our country. Smart Technology is a blessing for our country. We should try to avoid the bad consequences and use it for our betterment.

6.2 Limitations

There are some limitations observed throughout our project. If we want to make our project internet based, our home appliances are totally controlled through internet access. As we control the whole system through internet, we need to confirm high speed of internet. Otherwise, the system delay will occur as Ethernet shield will not be working as a network provider to the circuit.

The project relies on power supply. So if the power supply fails, the internet connection will be halted. Then database access will be stopped. For this, without security system the whole system will not be worked. In our project, security system is powered by another power source for security safety.

Though it needs less circuitry but its cost is not in minimal range. To get facility, users need to expense for this. The cost of installing a home automation system can be to a certain extent expensive. But it depends on the apparatus. The more sophisticated the system is the more expensive it will be.

This real time server base system is limited to only one person which means only one person can operate the system at a time.

If there is any break due to rupturing of cables or the fibers then the total system will be crashed. So, this will not be the case of radio signals or the other signals. There will be a difficulty of receiving signal.

If the individual does not handle the equipment safely or make use of the exact key to carry out the operations, human error can occur. Human faults also direct to the destructions of the device. Then there will be system collides.

Home automation has numerous drawbacks. For having home automated system, people will be lazier. That ultimately might end up with making great harm in human social and professional life.

In exceptionally uncommon cases, the unwavering quality of the home computerized gadgets fluctuates (decreases). It depends generally on the innovation utilized and the progressions being finished.

6.3 Future Scope

As we have mentioned earlier this thesis is not a complete project. This is just a basic structure of another complete system. We have done all the basic necessities of a typical house. The tasks that we have done are not the only tasks the components are able to do. There are a lot of other scopes for this project.

More appliances can be added in this system with a powerful relay module. Garage automatic door system can be added for extra security. All available smart devices can interface with this system including a car.

DS1307 Real Time Clock module is a very important device. This project could have a cloud database to save all the data. Readings from all the sensors with date and time can be saved. Room air purifier can be added to this system to make it more efficient. We can add a surveillance camera outside the house for extra security.

Solar power system can make this system extra cheap and durable. Then the system can run with the solar power.

Automatic door Control can also be done by using this project. Using finger print we may able to control our door. And it will be helpful to detect thief.

We can Also detect the amount of dust in our house using a optical dust detector. Using gas detector and buzzer we can make a automatic fire alarming System.

This project can be developed by replacing text communication with voice communication. The notification system cab changed to voice communication system.

REFERENCES

[1] Mohamed Abd El-Latif Mowad, Ahmed Fathy, Ahmed Hafez "Smart Home Automated Control System Using Android Application and Microcontroller" International Journal of Scientific & Engineering Research, Volume 5, Issue 5, May-2014 ISSN 2229-5518.

[2] Armando Roy Delgado, Rich Picking and Vic Grout"Remote- Controlled Home Automation Systems withDifferent Network Technologies" Centre for Applied Internet Research (CAIR), University of Wales, NEWI, Wrexham, UK.

[3] Obaid, T. et al. 2014. "ZIGBEE BASED VOICE CONTROLLED WIRELESS SMART HOME SYSTEM", International Journal of Wireless & Mobile Networks (IJWMN), Vol. 6, No. 1, Pg. 47-5

[4] The history of home automation, http://www.mccormickconcepts.com/easyblog-frontpage/The-history-of-home-automation.html (Accessed: 2018).

[5] history of remote control, https://science.howstuffworks.com/innovation/repurposed-inventions/history-of-remote-control.htm (Accessed: 2018).

[6] Md. Wasif Islam, Bishawjit Roy, Farhan Bin Mehtab, Nazira Humaira preeti, "Design of Arduino Based Home Automation System".

[7] Mark Zuckerberg debuts AI assistant voiced by Morgan Freeman, https://money.cnn.com/2016/12/20/technology/zuckerberg-jarvis-morgan-freeman/index.html (Accessed 2018).

[8] Wikipedia. Home automation, " http://en.wikipedia.org/wiki/Home_automation ".

[9] fingerprint vs facial recognition, "https://www.bayometric.com/fingerprint-vs-facial-recognition" .

[10] Lingfeng Zhang, Pengfei Dou, Shishir K. Shah, Ioannis A. Kakadiaris, "Hierarchical multi-label framework for robust face recognition", *Biometrics (ICB) 2015 International*
Conference on, pp. 127-134, 2015.

[11] Yu Zhang, Dit-Yan Yeung, "Semisupervised Generalized Discriminant Analysis", *Neural Networks IEEE Transactions on*, vol. 22, pp. 1207-1217, 2011, ISSN 1045-9227.

[12] S. M. Saad, A. R. Mohd Saad, A. M. Y. Kamarudin, A. Zakaria and A. Y. M. Shakaff, "Indoor air quality monitoring system using wireless sensor network (WSN) with web interface," *2013 International Conference on Electrical, Electronics and System Engineering (ICEESE)*, Kuala Lumpur, 2013, pp. 60-64.

[13] Shih-Chung Chen, Chung-Min Wu, Yeou-Jiunn Chen, Jung-Ting Chin, Yu-Yin Chen, "Smart home control for the people with severe disabilities", *Applied System Innovation (ICASI) 2017 International Conference on*, pp. 503-506, 2017.

[14] Dennis, A. K. (2013). " Raspberry-Pi Home Automation with Arduino". United Kingdom, UK: Packt Publishing Ltd.

[15] Z. Annan, S. Lin, C. Shuying, " Design and Realization of Home Appliances Control System Based on The Android Smartphone", International Conference on Control

Engineering and Communication Technology, China, 2012.

[16] Piyare, R., Tazil, M.: Bluetooth based home automation system using cell phone. In Consumer Electronics (ISCE), 2011 IEEE 15th International Symposium on, pp. 192-195.(2011).

[17] K. M. Abubeker, J. J. Edathala and S. Sebastian, "Some investigations on 'PIR' SENSOR and their application on significant energy savings in ATM counter," *2013 IEEE Conference on Information & Communication Technologies*, JeJu Island, 2013, pp. 37-41.

[18] P. S. R. Teja, V. Kushal, A. S. Srikar and K. Srinivasan, "Photosensitive security system for theft detection and control using GSM technology," *2015 International Conference on Signal Processing and Communication Engineering Systems*, Guntur, 2015, pp. 122-125.

[19] Arduino Uno , Available: "https://en.wikipedia.org/wiki/Arduino_Uno#cite_note-2" (Accessed: 2018).

[20] Arduino Uno R3, Available:

http://fullparts.com/media/catalog/product/cache/1/thumbnail/1000x/17f82f742ffe127f42dca9de82fb58b1/a/r/arduino_3.jpg.

[21] Arduino UNO Specification, Available:

https://www.researchgate.net/profile/Rasel_Sultan/publication/311421499/figure/fig1/AS:52599317728@1502418068248/Arduino-UNO-Board-Specification-Fig-2-Arduino-UNO-Board-Gyro-Accelerometer.png.

[22] HC – 05 Bluetooth module, Available: "https://components101.com/wireless/hc-05-bluetooth-module".

[23] Relay Switch Circuit, https://www.electronics-tutorials.ws/blog/relay-switch-circuit.html (Accessed: 2018).

[24] 4 Channel 5V Relay Module, Available:

http://wiki.sunfounder.cc/index.php?title=4_Channel_5V_Relay_Module\

[25] Introduction to the Arduino IDE, Available:

https://www.digikey.com/en/maker/blogs/2018/introduction-to-the-arduino-ide

[26] Bluetooth Module Interfacing with Arduino Available:

https://www.instructables.com/id/Remotely-Control-LED-using-HC-05-Bluetooth-Arduino/

[27] Connections in Relay, Available: https://i0.wp.com/randomnerdtutorials.com/wp-content/uploads/2016/11/relay-labeled.jpg?w=372&ssl=1

[28] Relay Module Interfacing with Arduino UNO, http://microcontrollerslab.com/relay-module-interfacing-arduino-arduino/

[29] Android SDK, Available https://developer.android.com/

[30] HC – 05 Command list, Available: https://www.teachmemicro.com/hc-05-bluetooth-command-list/

[31] proteus, Available: http://www.circuitstoday.com/category/proteus

APPENDIX

Code Segment

```
#include <SoftwareSerial.h>
const int rxPin = 2;
const int txPin = 3;
SoftwareSerial mySerial(rxPin, txPin);
int ac=4;
int light=5;
int fan=6;
int tv=7;
String data;
void setup()
{
  Serial.begin(9600);
  mySerial.begin(9600);
  pinMode(ac, OUTPUT);
  pinMode(light, OUTPUT);
  pinMode(fan, OUTPUT);
  pinMode(tv, OUTPUT);
  digitalWrite(ac, HIGH);
  digitalWrite(light, HIGH);
  digitalWrite(fan, HIGH);
  digitalWrite(tv, HIGH);
}
void loop()
{
   int i=0;
   char ch=0;
   data="";
```

```
 while(1)
 {
  while(mySerial.available()<=0);
  ch = mySerial.read();
  if(ch=='#')
  break;
  data+=ch;
 }
Serial.println(data);

 if(data=="*turn on AC")
 {
  digitalWrite(ac,LOW);
  Serial.println("ac on");
 }
 else if(data=="*turn off AC")
 {
  digitalWrite(ac,HIGH);
  Serial.println("ac off");
 }
 else if(data=="*turn on light")
 {
  digitalWrite(light,LOW);
  Serial.println("light on");
 }
 else if(data=="*turn off light")
 {
  digitalWrite(light,HIGH);
  Serial.println("light off");
 }
```

```
else if(data=="*turn on fan")
{
  digitalWrite(fan,LOW);
  Serial.println("fan on");
}
else if(data=="*turn off fan")
{
  digitalWrite(fan,HIGH);
  Serial.println("fan off");
}
else if(data=="*turn on TV")
{
  digitalWrite(tv,LOW);
  Serial.println("tv on");
}
else if(data=="*turn on TV")
{
  digitalWrite(tv,HIGH);
  Serial.println("tv off");
}
else if(data=="*turn on all")
{
  digitalWrite(ac,LOW);
  digitalWrite(light,LOW);
  digitalWrite(fan,LOW);
  digitalWrite(tv,LOW);
  Serial.println("all on");
}
else if(data=="*turn off all")
{
```

```
  digitalWrite(ac,HIGH);
  digitalWrite(light,HIGH);
  digitalWrite(fan,HIGH);
  digitalWrite(tv,HIGH);
  Serial.println("all off");

}
}
```