SMART-ENABLED SWITCH CONTROL USING ARDUINO

Home automation is one of the recent developments which enables the life to be at ease and secured too. It also helps the aged and physically challenged people to live their life in a better way. Home automation also saves time and energy. This project is aimed at controlling the electrical gadgets using voice / text to switch ON and OFF. An ARDUINO UNO board is used to control the devices under control.

An algorithm is developed to receive the voice / text message and based on which the controlling action is being taken. The algorithm is implemented using embedded programming with ARDUINO. Mobile app called "ARDUINO voice control and Bluetooth terminal" is used to transmit the voice / text command under Bluetooth and HC-05 Bluetooth Module is used to receive the command which is further interfaced with the ARDUINO UNO as input device. The programme gives an output command based on the received input to switch ON/OFF the devices.

In this project both voice and text controls are implemented to switch ON/OFF the devices. For implementation LED's are used as output devices.

INTRODUCTION

OVERVIEW:

Home automation system is getting popular and widely used in a lot of houses worldwide. It has tons of advantages to users even more to the handicapped and/or elderly users in which it will make it easier for them to control their home appliances. Home automation systems can be labeled to two medium in which how it is connected and they are either wired or wirelessly connected. The main difference between these two kinds is that home appliances are linked wirelessly a central controller if it a wireless home automation system. On the other hand, the appliances are connected to a central controller if the medium use wired communication method. Wireless system had been introduced in order to dispose of wired communication among home appliances. Arduino based, Bluetooth based home automation will be applied.

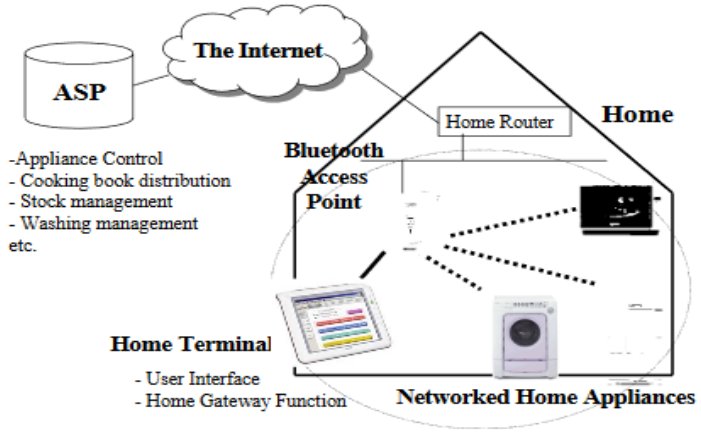
Nowadays, everyone cannot be separated from their smartphones. a number of five thousands individuals from USA, UK, South Korea, India, China, South Africa, Indonesia and Brazil took a survey regarding which was done by Time magazine. The result proved most of them is inseparable from their smartphones, eighty four percent allegedly claimed that survive without their smartphones. Another study shows that seventy five percent of the market share is Android and a total of one hundred and six million android smartphone were shipped in the second half of 2012. Android smartphone became the top operating system in the market in the present time worldwide and it became the most popular operating system known to man

MOTIVATION:

This study will be undertaken to create a home automation system at low cost and easy to create, this will benefit both the manufacturer and the client. It will help the manufacturer by making it easy and cheaper to apply it, and it will also benefit the clients by making it cost effective and the most important advantage is that it will make the house a much more convenient place for the clients especially for the elders and the handicapped.

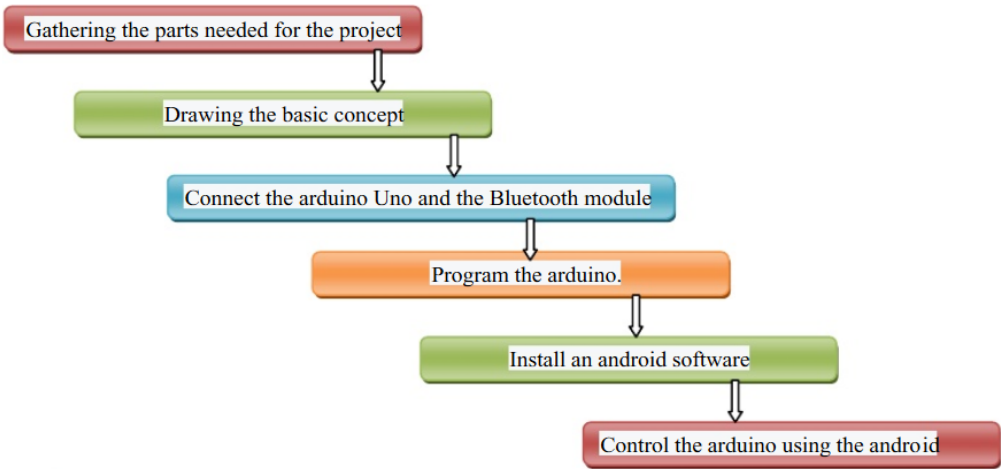
OBJECTIVE:

- To construct a wireless home automation system controlled by a smartphone specifically an android device.
- To design and implement cost effective home automation system yet an efficient one.
- To design a user friendly and a safe system to control home appliances especially aimed to aid the elders and handicapped.



Block Diagram of Usage of Home Appliances

DESCRIPTION



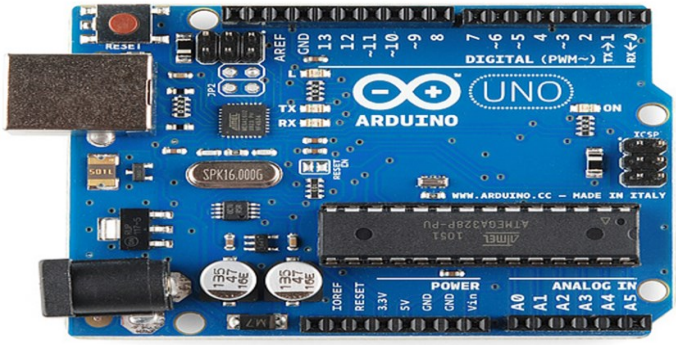
1) ARDUINO UNO:

Arduino is an open-source electronics 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. You can tell your 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 (based on Wiring), and the Arduino Software (IDE), based on Processing.

Over the years Arduino has been the brain of thousands of projects, from everyday objects to complex scientific instruments. A worldwide community of makers - students, hobbyists, artists, programmers, and professionals - has gathered around this open-source platform, their contributions have added up to an incredible amount of accessible knowledge that can be of great help to novices and experts alike.

Arduino was born at the Ivrea Interaction Design Institute as an easy tool for fast prototyping, aimed at students without a background in electronics and programming. As soon as it reached a wider community, the Arduino board started changing to adapt to new needs and challenges, differentiating

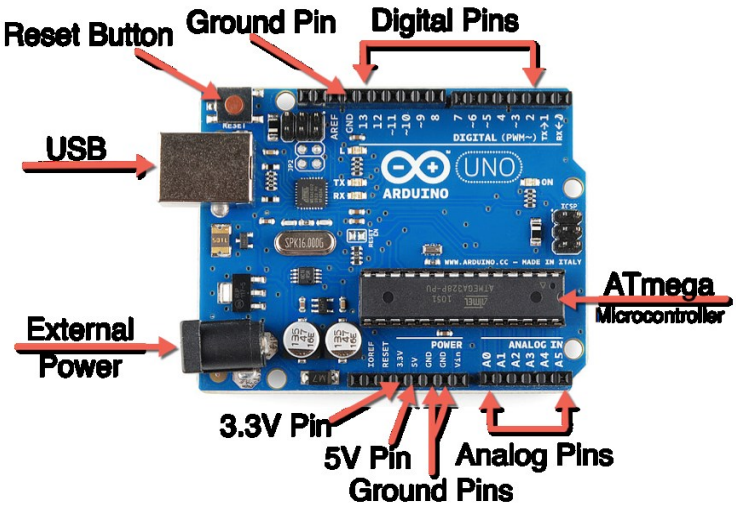
its offer from simple 8-bit boards to products for IoT applications, wearable, 3D printing, and embedded environments. All Arduino boards are completely open-source, empowering users to build them independently and eventually adapt them to their particular needs. The software, too, is open-source, and it is growing through the co hers, it's conveniently based on the Processing programming environment, so students learning to program in that environment the Arduino IDE work



Arduino UNO microcontroller Board

The Arduino Software (IDE) is easy-to-use for beginners, yet flexible enough for advanced users to take advantage of as well. For teachers, it's conveniently based on the Processing programming environment, so students learning to program in that environment will be familiar with how the Arduino IDE works. The Arduino board can be programmed to do anything by simply programming the microcontroller on board using a set of instructions for which, the Arduino board consists of a USB plug to communicate with your computer and a bunch of connection sockets that can be wired to external devices like motors, LEDs etc.

The aim of Arduino is to introduce the world of electronics to people who have small to no experience in electronics like hobbyists, designers, artists etc.



Pin out Configuration

2) BLUETOOTH MODULE

Bluetooth is an industrial specification for WPAN (Wireless Personal Area Network), which enables the transmission of data and voice between different devices through a radiofrequency link in the 2.4 GHz ISM band. The main objectives to be achieved with this standard are facilitate communication between mobile devices, remove cables and connectors between them and offer the possibility to create small wireless networks facilitating the synchronization of data between personal equipment. All Bluetooth devices have a unique address of 48 bits and a device name that allows the identification of each other.

Devices that incorporate this protocol can communicate with one another when within its reach. The communications are by radiofrequency so that the devices do not have to be aligned and can even be in separate rooms if the transmission power is sufficient. These devices are classified as Class 1, Class 2 and Class 3 depending on the transmission power.

Class	**Maximum Power (mW)**	**Maximum Power (dBm)**	**Range**
Class 1	100 mW	20 dBm	100 m
Class 2	2.5 mW	4 dBm	10 m
Class 3	1 mW	0 dBm	1 m

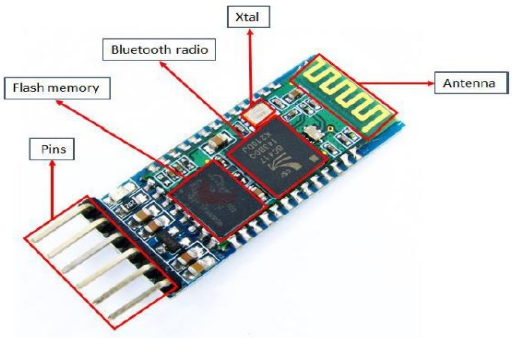
There are two main Bluetooth modules compatibles with Arduino and other microcontrollers: HC-05 and HC-06. These are both of Class 2 and ease to use thanks to its SPP (*Serial Port Protocol*).

Bluetooth module HC-06 is only capable to work as Slave, it means that it can be connected only to a Master and it has a reduced set of instructions, which it can attend. On the other hand, Bluetooth module HC-05 is capable both to work as Slave and as Master, it can be connected to a more than one Slave and receive and request information from all of them, arbitrating the transfer of information (maximum 7 slaves) and can attend to a greater number of configuration commands.

Bluetooth communication

After a global communication overview, it is decided to use Bluetooth for the communication between Arduino and mobile application, more specifically HC-05 module.

This module was finally chosen to meet all application requirements. It is a wireless technology; the sender and the receiver are not physically connected, but trough electromagnetic waves. It has a range of 10 meters, enough to be able to control the variables using mobile phone, since this type of control make sense when the user is inside the house. Finally, the only additional requirement to make the communication between both devices is that the mobile phone has Bluetooth connectivity, which nowadays is a standard characteristic of mobiles.



Main characteristics:

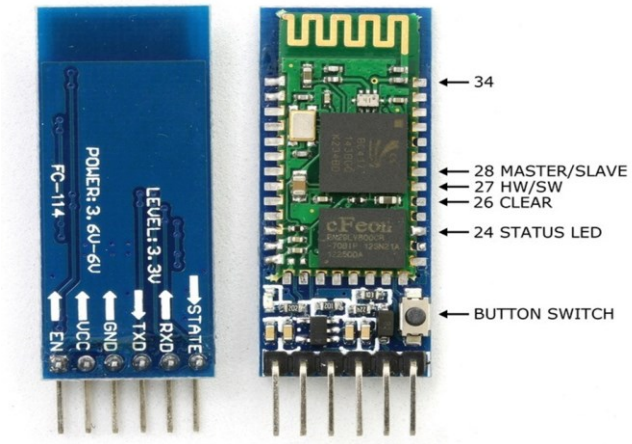
- Configurable as Slave and as Master.
- Radio chip: CSR BC417143
- Frequency: 2.4 GHz ISM band
- Modulation: GFSK (*Gaussian Frequency Shift Keying*).
- Built-in PCB antenna.
- Emission Power: ≤ 4 dBm (Class 2).
- Range: 10 m.
- Speed: Asynchronous: 2.1 Mbps/160 kbps; Synchronous: 1 Mbps/1 Mbps
- Security: Authentication and encryption.
- Current consumption: 50 mA.
- Profiles: Serial port Bluetooth.
- Supply voltage: 3.6 V – 6 V.
- Operation temperature: -20 ºC – 75 ºC.

HC-05 Bluetooth Module:

The HC-05 Bluetooth Module has 6 pins-Vcc, GND, TX, RX, Key, and LED. It comes pre-programmed as a slave, so there is no need to connect the Key pin, unless you need it change it to Master Mode.

The major difference between Master and Slave modes is that, in Slave mode the Bluetooth module cannot initiate a connection, it can however accept incoming connections. After the connection is established the Bluetooth module can transmit and receive data regardless of the mode it is running in. If you are using a phone to connect to the Bluetooth module, you can simply use it in the Slave mode. The default data transmission rate is 9600kbps.

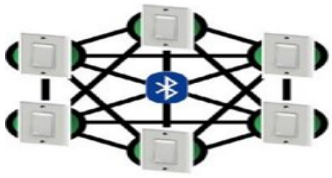
The range for Bluetooth communication is usually 30m or less. The module has a factory set pin of "1234" which is used while pairing the module to a phone.



This Bluetooth module can easily achieve serial wireless data transmission. Its operating frequency is among the most popular 2.4GHz ISM frequency band (i.e. Industrial, scientific and medical). It adopts Bluetooth 2.0+EDR standard. In Bluetooth 2.0, signal transmit time of different devices stands at a 0.5 seconds interval so that the workload of Bluetooth chip can be reduced substantially and more sleeping time can be saved for Bluetooth.

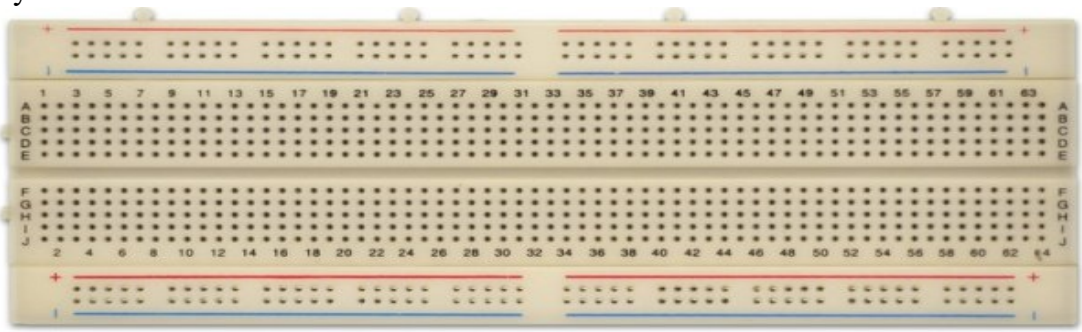
This module is set with serial interface, which is easy to use and simplifies the overall design/development cycle

Bluetooth technology has been one of the critical innovations to home automation systems or smart living.it is a remote technology created to take the place of wired devices to wireless one which links gadgets like cell phones and pc's (laptops/desktops).albeit "link substitution" could make a point to point connection, Bluetooth permits remort gadgets to have the ability to connect to one another inside reach .the system of a set of Bluetooth gadgets is called "piconet" ,which is anideal technology to system a brilliant advanced home.



3) Bread Board:

A breadboard is a construction base for prototyping of electronics. Originally it was literally a bread board, a polished piece of wood used for slicing bread. In the 1970s the solder less breadboard (a.k.a. plug board, a terminal array board) became available and nowadays the term "breadboard" is commonly used to refer to these.

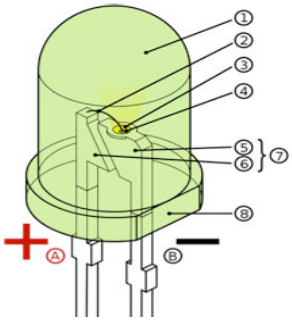


Because the solder less breadboard does not require soldering, it is reusable. This makes it easy to use for creating temporary prototypes and experimenting with circuit design. For this reason, solder less breadboards are also popular with students and in technological education. Older breadboard types did not have this property. A stripboard (Vero board) and similar prototyping printed circuit boards, which are used to build semi-permanent soldered prototypes or one-offs, cannot easily be reused. A variety of electronic systems may be prototyped by using breadboards, from small analog and digital circuits to complete central processing units (CPUs).

4) LED LIGHTS:

A light-emitting diode (LED) is a two-lead semiconductor light source. It is a p–n junction diode that emits light when activated. When a suitable current is applied to the leads, electrons are able to recombine with electron holes within the device, releasing energy in the form of photons. This effect is called electroluminescence, and the color of the light (corresponding to the energy of the photon) is determined by the energy band gap of the semiconductor. LEDs are typically small (less than 1 mm2) and integrated optical components may be used to shape the radiation pattern.

Appearing as practical electronic components in 1962, the earliest LEDs emitted low-intensity infrared light. Infrared LEDs are still frequently used as transmitting elements in remote-control circuits, such as those in remote controls for a wide variety of consumer electronics. The first visible-light LEDs were of low intensity and limited to red. Modern LEDs are available across the visible, ultraviolet, and infrared wavelengths, with very high brightness.



Typical Through-Hole LED

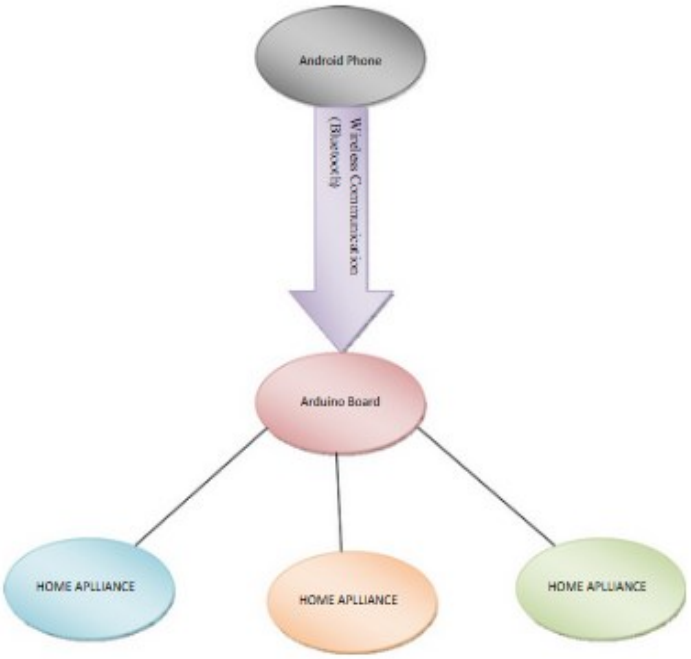
1. Epoxy Lens/Case
2. Wire Bond
3. Reflective Cavity
4. Semi-conductor Die
5. Anvil
6. Post
7. Leadframe
8. Flat-Spot

Early LEDs were often used as indicator lamps for electronic devices, replacing small incandescent bulbs. They were soon packaged into numeric readouts in the form of seven-segment displays and were commonly seen in digital clocks. Recent developments have produced LEDs suitable for environmental and task lighting. LEDs have led to new displays and sensors, while their high switching rates are useful in advanced communications technology.

LEDs have many advantages over incandescent light sources, including lower energy consumption, longer lifetime, improved physical robustness, smaller size, and faster switching. Light-emitting diodes are used in applications as diverse as aviation lighting, automotive headlamps, advertising, general lighting, traffic signals, camera flashes, lighted wallpaper and medical devices. They are also significantly more energy efficient and, arguably, have fewer environmental concerns linked to their disposal.

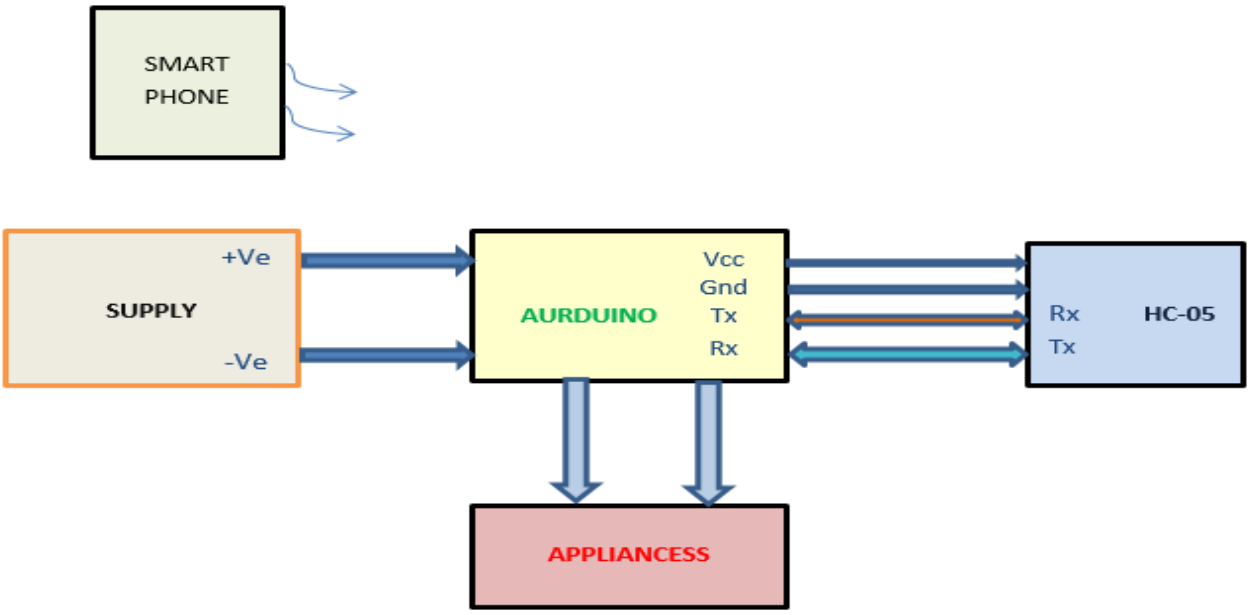
Unlike a laser, the color of light emitted from an LED is neither coherent nor monochromatic, but the spectrum is narrow with respect to human vision, and for most purposes the light from a simple diode element can be regarded as functionally monochromatic.

SMART HOME ENVIRONMENT



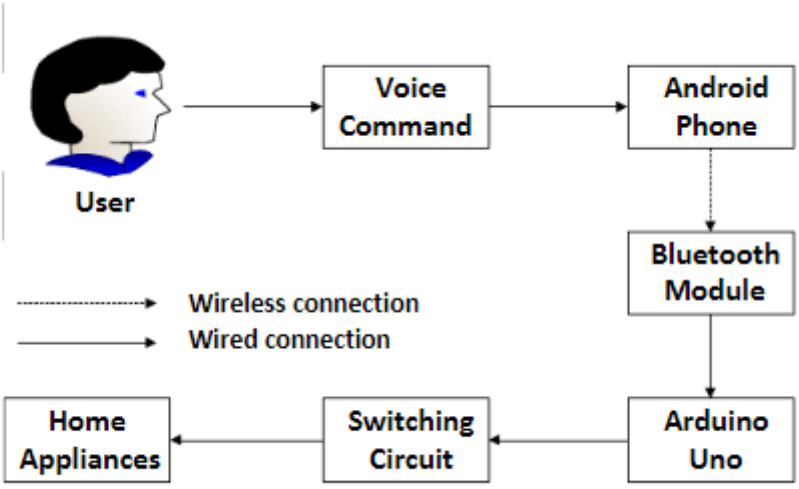
Smart Environment

Smart Automation

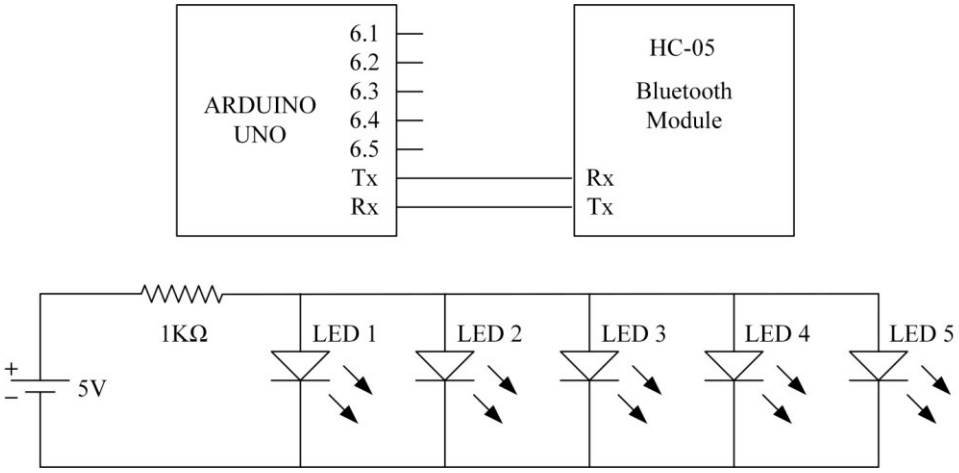


Smart Home System Block Diagram

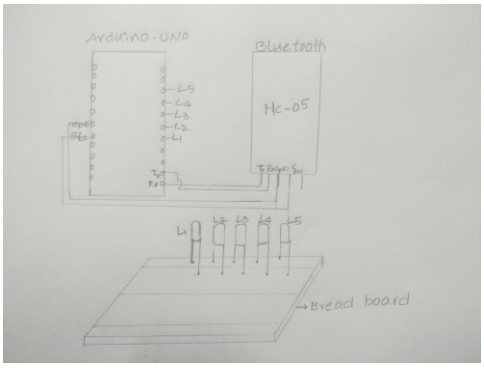
In this Figure, voice is voiced into the smart phone and the supply is given to the arduino board which is connected to the Bluetooth module and the arduino is embedded with a particular programme which allows the usage of the appliances.



Electrical and Electronics Circuit Overview



Circuit diagram of the breadboard which consists of 5 LED's, 1 5V Supply, 1 Resistor of 1K Ohm, and the Arduino Uno and the Bluetooth module and connected to each other on their two ports Tx and Rx



PROCEDURE

Connection are given as per the circuit diagram.
(i) Bluetooth's RX Terminal is connected to TX terminal of the Arduino board.
(ii) Bluetooth's TX Terminal is connected to RX terminal of the Arduino board.
(iii) Bluetooth Gnd to Arduino board Gnd.
(iv)Vcc of the Bluetooth Module is Arduino Board's Vcc.
(v)All the Connections in Bluetooth are placed on the breadboard.
(vi)LED Lights are place on the breadboard.
(vii)LED's are numbered as per the programming.
(viii) All LED's are connected to Terminals of the Arduino Board as per the numbering in the programming.
(ix)To Check Whether the Arduino Is working on not, we use the programming for a single LED.
(x)To Check Whether the Bluetooth Module is Working or not we use the one example programme.
(xi) We Debug the programme into Bluetooth and Arduino Using the Arduino Cable.
(xii) The Connection from the Arduino board to Bluetooth module to LED Lights is given with use of Jumper Cables (or) wires.
(xiii)For the working of LED's also Programme is given.
(xiv)For All the process and the working, we use an Bluetooth App(Arduino Voice Control App And Bluetooth Terminal App).
(xv)Bluetooth Terminal App Is connected to the Bluetooth Module with help of a Smartphone.
(xvi)The Command inputted in the Programme Is Voiced into the Arduino Control App In the Command Form.
(xvii)The Result of the command is the glowing of the LED Light.

PROGRAMMING AND SIMULATION

Arduino IDE

The Arduino IDE is a multi-platform development platform for creating embedded programs that run on the many various Arduino microcontrollers. As seen in Figure, it is rather basic in its functionality when compared to many other modern IDEs.
Basic structure of a sketch:
A sketch can be divided in three parts.
1. Name variable
In the first part elements of the program are named (This will be explained in program no.
3). This part is not absolutely necessary.
2. Setup (absolutely necessary for the program)
The setup will be performed only once. Here you are telling the program for example what
Pin (slot for cables) should be an input and what should be an output on the boards.
Defined as Output: The pin should put out a voltage. For example: With this pin a LED is
meant to light up.
Defined as an Input: The board should read out a voltage. For example: A switch is
actuated. The board recognized this, because it gets a voltage on the Input pin.
3. Loop (absolutely necessary for the program)
The board will continuously repeat this loop part. It assimilates the sketch from beginning to end and starts again from the beginning and so on.
Deployment of the code to the microcontroller in the Arduino IDE is quite streamlined. One just needs to attach an USB cable to the to the USB port of the Arduino, pick the right board model from the IDE and upload the code. The entire Arduino IDE with its many utility libraries is actually just a user-friendly wrapper around the AVR-GCC compiler, which handles compiling the source code to a

format that the Arduino's on-board ATmega can understand. After the code has been checked for errors and compiled into binary, another utility program by the name of AVRDUDE (short for AVR Downloader/UploaDEr) handles the uploading and saving of the program to the Arduino itself. Many advanced users choose to bypass relative unwieldiness of the Arduino IDE by using their preferred text editor/IDE, and just use the AVR-GCC and AVRDUDE as stand-alone programs.

Installation

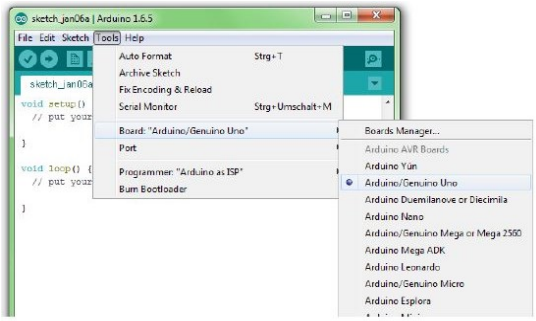
Now one after another the Arduino software and the USB driver for the board have to be installed.

Installation and set up of the Arduino software

1. Download the Arduino software on www.arduino.cc and install it on the computer (The microcontroller NOT connected to the PC). After that, you open the software file and start the program named arduino.exe.

Two set ups on the program are important and should be considered.

a) The board that you want to connect, has to be selected on the arduino software.

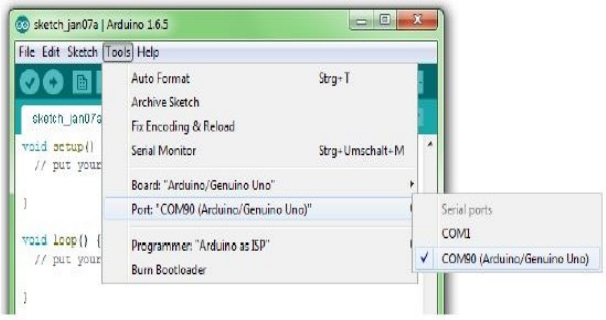


Selection of Arduino Board

b) You have to choose the right "Serial-Port", to let the Computer know to which port the board has been connected. That is only possible if the USB driver has been installed correctly. It can be checked this way:

At the moment, the Arduino isn't connected to the PC. If you now choose "Port", under the field "Tool", you will already see one or more ports here (COM1/ COM2/ COM3…).

The quantity of the shown ports does not depend on the quantity of the USB ports on the computer. When the board is connected to the computer, YOU WILL FIND ONE MORE PORT.



Selection of Arduino Serial Communication port

Installation of the USB driver

How it should be:

1. You connect the board to the computer.
2. The Computer recognizes the board and suggests installing a driver automatically.

ATTENTION: Wait a second! Most of the time the computer can't find the driver automatically to install it. You might choose the driver by your own to install it. It can be found in the Arduino file under “Drivers”. Control: At the control panel of the Computer you can find the “Device manager”. If the board has been installed successfully, it should appear here. When the installation has failed, there is either nothing special to find or you will find an unknown USB device with a yellow exclamation mark. In this case: Click on the unknown device and choose “update USB driver”. Now you can start over with the manual installation

Arduino Embedded C++ Programming

C++ is an object-oriented superset of the C language, originally developed by Bjarne Stroustrup from 1979. (Learncpp.com, 2007). Since its inception, it has influenced many other programming languages, such as C# and Java. It is mostly designed for system programming and embedded, performance-intensive systems. (B. Stroustrup, 2014.)
While C++ is said to be a superset of the C language, the language used to program the Arduino family of microcontroller can be thought of as a subset of C++. While many of the familiar paradigms of C++ programming, such as object-oriented programming, are mostly too memory intensive for embedded usage, the Arduino language is basically just a set of C/C++ functions.
The following is an example of a bare-minimum Arduino program:

```
1 void setup() {
2   // put your setup code here, to run once:
3
4 }
5
6 void loop() {
7   // put your main code here, to run repeatedly:
8
9 }
```

Arduino Embedded C++

The setup- and loop functions are the only two required functions that must always exist in a valid Arduino program. The setup function is called once, when the microcontroller is first powered up, at the beginning of the program. The program then proceeds to run forever inside the function called loop.

SMART AUTOMATION CODINGS

- *Initialization of LED pins*

```
int led1 = 2;
int led2 = 3;
int led3 = 4;
int led4 = 5;
int led5 = 6;
```

- *Set Serial communication Baud rate to begin with Arduino IDE*

```
Serial.begin(9600);
```

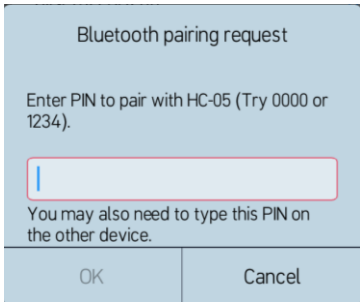
- *Arduino IDE PinMode configuration*

```
                                        void setup()
                                        {
pinMode(led1, OUTPUT);
pinMode(led2, OUTPUT);
pinMode(led3, OUTPUT);
pinMode(led4, OUTPUT);
pinMode(led5, OUTPUT);
}
```

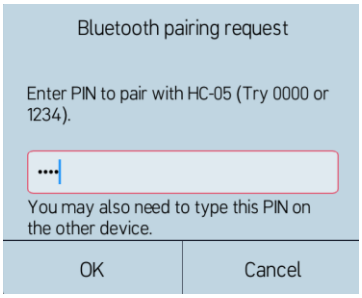
- *Smart automation Embeddded C++ conditions for Arduino IDE*

```
                                        void loop()
                                        {
                                                if(Serial.available()){
                                                while(Serial.available())
                                                {
                                                char inChar = (char)Serial.read(); //read the input
                                        inputString += inChar;
                                                        }
                                        Serial.println(inputString);
                                        while (Serial.available() > 0)
                                        if(inputString == "conditions for each LED's"){
                                        digitalWrite(led, HIGH);
                                        Serial.println("light's ON");
                                                }
                                        if(inputString == "conditions for each LED's"){
                                        digitalWrite(led, LOW);
                                        Serial.println("light's OFF");
                                                }
                                        }
```

SMART SIMULATION

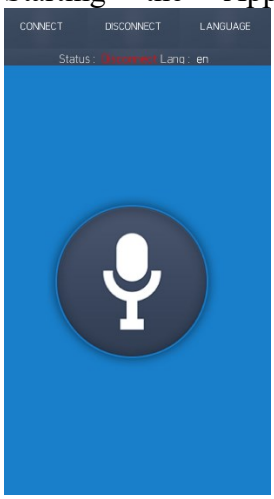


Initialization of HC-05 with smartphone



Passcode verification HC-05 with smartphone

a. Starting the App

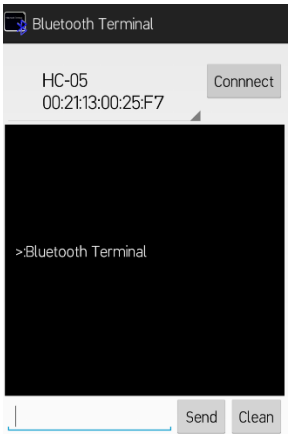


b. Selecting Bluetooth HC-05

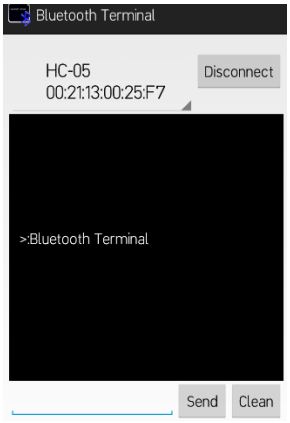
c. Connected with HC-05

Connection of HC-05 with Arduino voice controlled App

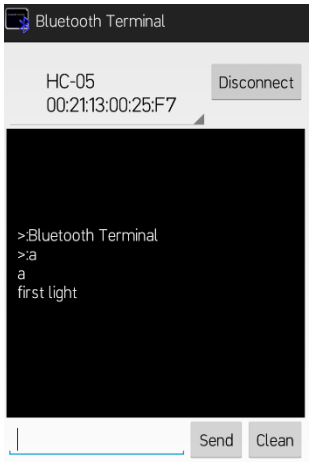
a. Starting the App



b. Connecting with HC-05



c. Sending character to switch ON the LED



d. Sending character to switch OFF the LED

Connection of HC-05 with Bluetooth Terminal App

SMART HARDWARE SETUP

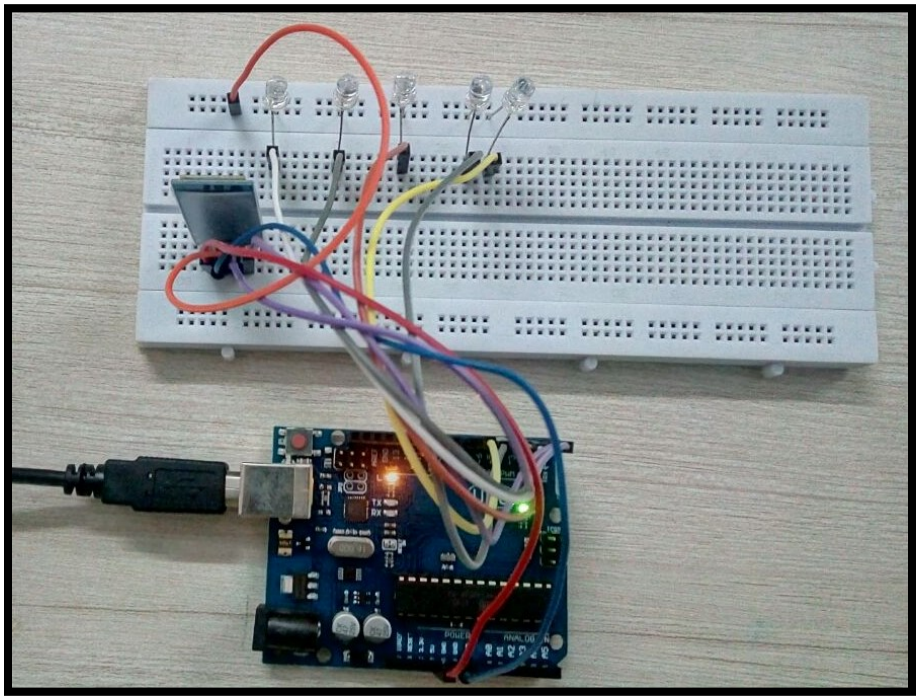
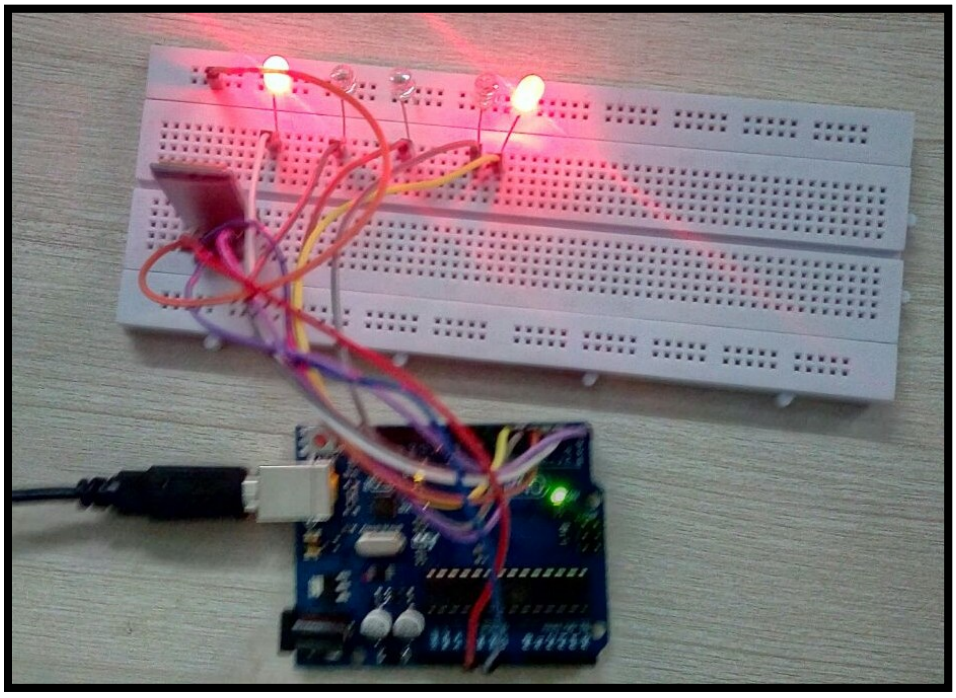
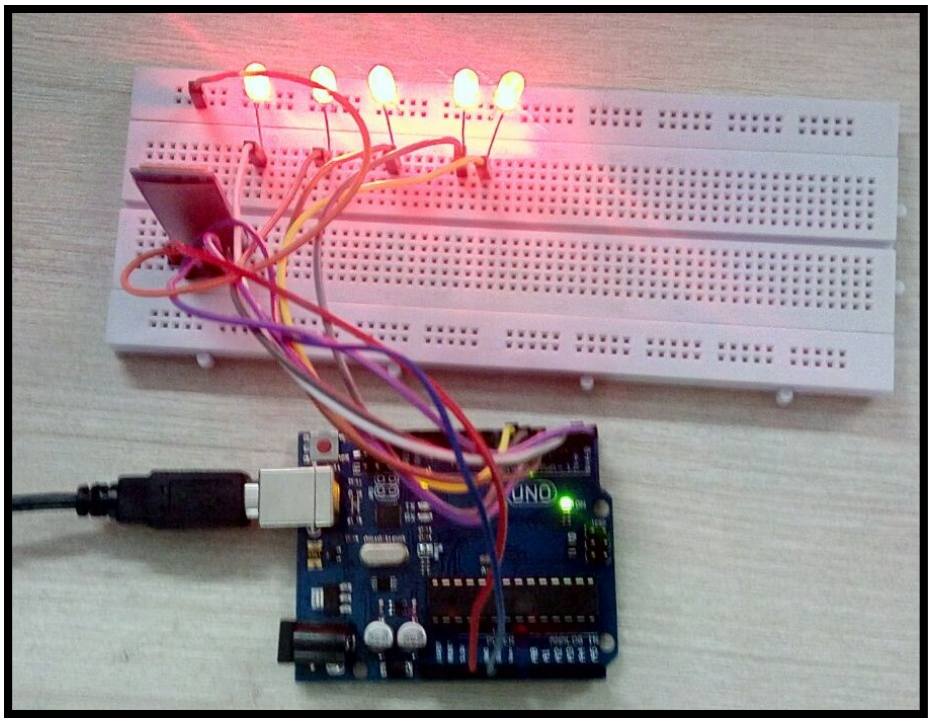


Figure. Hardware setup



Hardware output based on Voice and also Text based controlling setup



Smart Hardware output

Conclusion:

It can be concluded that "**VOICE AND TEXT CONTROLLED SWITCH USING ARDUINO**" was a success. This system consists of an Arduino Uno board, a Bluetooth Module an android phone, power sockets, home appliances and android application it is user friendly and cost effective.

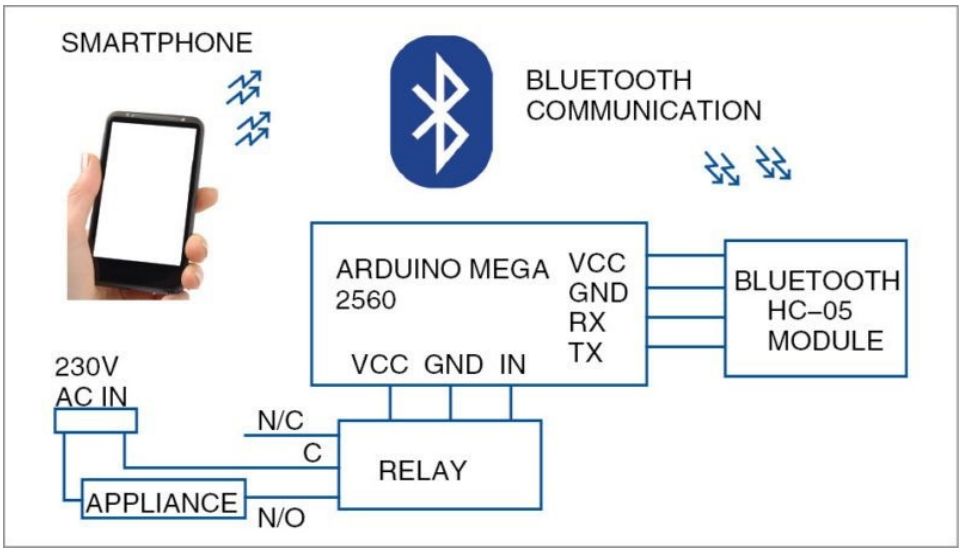
Also it can be concluded that the objectives of this project has been successfully met and they are as follows:

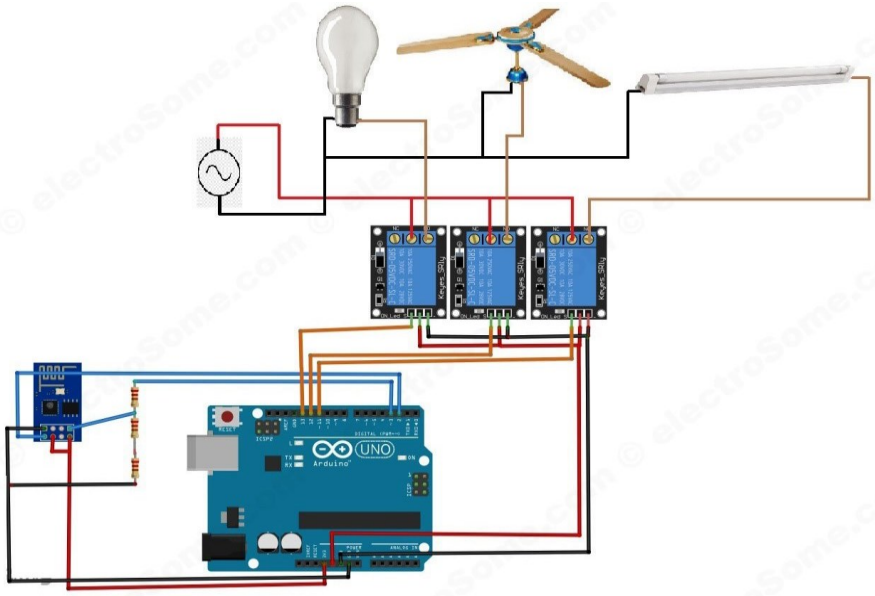
- Constructed a wireless home automation system controlled by a smart phone specifically an android device.
- Designed and implement cost effective home automation system yet an efficient one.
- Designed a user friendly and a safe system to controlled home appliances especially aimed to aid the elders and handicapped.

Future works:

There are some recommendations for future works, some of them are

- Better to use relay modules and connect it directly than using normal relays with breadboard.
- Try to find a way to amplify bluetooth module signal to work in grater distance.
- Test each and every component before using them especially the relays for safety purpose.
- The heart beat is being monitored continuously via heart beat sensor circuit. If suppose person forgets to switch of a device such as T.V, A.C ,etc. and falls asleep, then the proposed system is useful..
- Though there are various systems for home automation already available but none of them provision like this.
- It will be helpful in saving energy and bring down the annual cost of an individual's electricity bills thus helping both ways.





- By the use of Voice Control, Robots can also be controlled with the help of application that are helpful in their movement and their actions.

- It is very helpful for the use of handicapped, as it is can help the movement of their chair with the help of Bluetooth installed in the android phone.

- The system needs a continuous power supply to be practical or else we might not be able to control the appliances
.
- Hence, best way to design the system efficiency would be to implement both the automated control and manual control through switches at a time.

RESULTS AND DISCUSSION

RESULT:

Home Automation is undeniably a resource which can make a home environment automated. People can control their electrical devices via these Home Automation devices and set up the controlling actions in the computer. We think this product have high potential for marketing in the future.

To be able to design a product using the current technology that will be beneficial to the lives of others is a huge contribution to the community. This paper presents the design and implementation of a low cost but yet flexible and secure cell phone based home automation system. The design is based on a standalone Arduino BT board and the home appliances are connected to the input/ output ports of this board via relays. The communication between the cell phone and the Arduino BT board is wireless. This system is designed to be low cost and scalable allowing variety of devices to be controlled with minimum changes to its core. Password protection is being used to only allow authorized users from accessing the appliances at home.

DISCUSSION:

Discussed home Automation systems have been done and their all common, features, advantages and disadvantages are highlighted. All above discussed system have a main module that is connected with home appliances. Different types of communication techniques are used to transmit commands from the user
interface to the main controller board.

Bluetooth based home automation system gives complete control over home appliance as long as the user is in range of Bluetooth network. The Bluetooth system uses a PC or smartphone as receiver device. It has a high communication rate, great security and low cost, so it can be implemented as a real time system. Bluetooth network has limited range of 10 meters if the smartphone is out of range, then it will not be able to control the home appliances, this is one of the main disadvantages of Bluetooth based home automation system.

Voice recognition based home automation systems are most useful for handicapped and elderly persons, who wants to control home appliances by speaking voice commands. Voice of every human contains unique features so such system has greater security. An operating system for Android smartphones has built-in voice recognition feature that can use for voice recognition tool for home automationsystem.

APPENDIX

```
/*
Arduino Turn LED On/Off using voice strings
Created April 24, 2018
Navaprakash, Researcher (India)
Originally: https://github.com/npk1
*/

char junk;
String inputString="";
int led1 = 2;                  // Initialization of Digital pins
int led2 = 3;
int led3 = 4;
int led4 = 5;
int led5 = 6;
void setup()                   // run once, when the sketch starts
{
 Serial.begin(9600);           // set the baud rate to 9600,
 pinMode(led1, OUTPUT);        // Declaration of output
 pinMode(led2, OUTPUT);
 pinMode(led3, OUTPUT);
 pinMode(led4, OUTPUT);
 pinMode(led5, OUTPUT);
}

void loop()
{
  if(Serial.available()){
  while(Serial.available())
    {
      char inChar = (char)Serial.read(); //read the input
      inputString += inChar;        //make a string of the characters coming on serial
    }
    Serial.println(inputString);
    while (Serial.available() > 0)
    { junk = Serial.read() ; }      // clear the serial buffer
    if(inputString == "a"){         //in case of 'a' turn the LED on
      digitalWrite(led1, HIGH);
      Serial.println("first light");
    }if(inputString == "b"){   //in case of 'b' turn the LED off
      digitalWrite(led2, HIGH);
      Serial.println("second light");
    }if(inputString == "c"){        //in case of 'c' turn the LED on
      digitalWrite(led3, HIGH);
      Serial.println("third light");
    }if(inputString == "d"){   //incase of 'd' turn the LED off
      digitalWrite(led4, HIGH);
      Serial.println("forth light");
    }if(inputString == "e"){        //in case of 'e' turn the LED on
      digitalWrite(led5, HIGH);
      Serial.println("fifth light");}
    }if(inputString == "6"){   //incase of '6' turn the LED off
      digitalWrite(led1, LOW);
      Serial.println("first light off");
    }if(inputString == "7"){        //in case of '7' turn the LED on
      digitalWrite(led2, LOW);
      Serial.println("second light off");
    }if(inputString == "8"){   //incase of '8' turn the LED off
      digitalWrite(led3, LOW);
      Serial.println("third light off");
    }if(inputString == "9"){        //in case of '9' turn the LED on
      digitalWrite(led4, LOW);
      Serial.println("forth light off");
    }if(inputString == "1"){   //incase of '1' turn the LED off
      digitalWrite(led5, LOW);
      Serial.println("fifth light off");}
    inputString = "";
  }
```