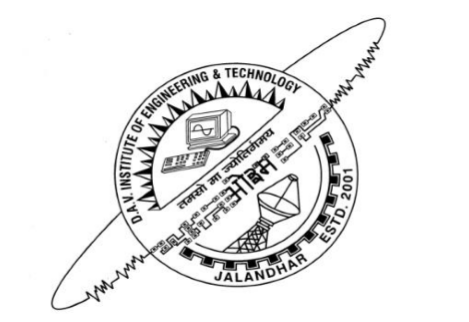
**DAV INSTITUTE OF ENGINEERING AND**

**TECHNOLOGY, JALANDHAR**

****

**MAJOR PROJECT (BTEC-705)**

**PROJECT NAME –** Voice Controlled Home Appliances

**PROJECT COORDINATOR -** Ms. Poonam Sethi

**PROJECT GUIDE –** Mr. Hari Singh

**PROJECT BY -**

Shubham Nag 237/13 ECE-4B

Anshul Mahajan 054/13 ECE-4B

Gurwinder Singh 062/13 ECE-4B

**VOICE CONTROLLED HOME APPLIANCES**

**CANDIDATE’S DECLARATION**

We hereby certify that the work which is being presented in the major project entitled “VOICE CONTROLLED HOME APPLIANCES” by “Shubham Nag, AnshulMahajan&Gurwinder Singh” (Electronics & Communication Engineering) is an authentic record of our own work carried out during a period from July 2016 to Dec 2016 under the supervision of PROF. HARI SINGH and coordination of MS. POONAM SETHI.

Signature of Student(s)

This is to certify that the above statement made by the candidates is correct to best of my/our knowledge.

Signature of Supervisor Signature of Coordinator

**ACKNOWLEDGMENT**

A journey becomes easier when we travel together interdependence is certainly more valuable than independence. This dissertation is a result of a yearlonghard work and blessings of ALMIGHTY.

The most important person I would like to thank is my guide Prof. Hari Singh, Professor, Department of Electronics & Communication Engineering, DAV Institute of Engineering & Technology, Jalandhar, for her guidance , inspiration and many useful skills he has taught us, including technical writing and critical thinking. We will never forget this support and consideration during my tough times. Probably he doesn’t even know how much we have learnt from him.

We also wish to extend our thanks to our project coordinator Ms. PoonamSethi and all the departmental teachers for their insightful comments and constructive suggestion to improve the quality of project work.

We are extremely thankful to Dr. Manoj Kumar (Principal, DAVIET, Jalandhar) and DAVIET management committee to provide the excellent infrastructure and resources to work without which this would have not been possible.

**ABSTRACT**

The key objective of our system design is to provide easy means for normal, handicapped and old age persons to control and operate appliances. Since handicapped and old age people need attendants most of the time to look for their appliance switching needs, we require a system which is affordable, mobile and simple to implement. This system is designed using Android which will send the captured voice input to Arduino UNO via Bluetooth module. An android application is made which uses Google’s voice recognition and Arduino Uno processes the voice input to control the home appliances.

Here in our project, we have used Arduino UNO R3 along with voice recognition of an android phone and the connection between them is established using Bluetooth module (HC-05).

A 4-channel relay is used to drive the ac appliances by providing the arduino output to it along with the ac supply.

**CONTENTS**

**Candidate’s Declaration**

**Acknowledgment**

**Abstract**

**1. INTRODUCTION . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .. . . . . . 1-4**

* **Synopsis**
* **Literature Review**

**2. PROPOSED SYSTEM . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .. . . . . . . 5-19**

* **List of components**
* **Android**
* **Bluetooth module**
* **Arduino uno**
* **Four channel relay**
* **MIT App inventor**
* **Arduino software**

**3. WORKING . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 20-25**

* **Flowchart**
* **Connecting Bluetooth**
* **Giving commands**
* **Output**

**4. RESULT AND DISCUSSION . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .26-27**

* **Result in noiseless environment**
* **Result in noisy environment**
* **Future applications**

**5. CONCLUSION . . . . . . . . . . . . . . . . . . . . . . . . . . … .. . . .. .. . . . . . …. . 28**

**REFERENCES**

**LIST OF FIGURES**

|  |  |
| --- | --- |
| **FIGURE NUMBER** | **DESCRIPTION** |
| Figure 2.1 | Block Diagram of the System |
| Figure 2.2 | Interface for the Voice Control Application |
| Figure 2.3 | Bluetooth module |
| Figure 2.4 | Interfacing with arduino |
| Figure 2.5 | Arduino UNO board |
| Figure 2.6 | Specifications of Arduino UNO |
| Figure 2.7 | Four channel relay |
| Figure 2.8 | Relay working |
| Figure 2.9 | App inventor architecture |
| Figure 2.10 | Arduino interface |
| Figure 3.1 | Flowchart |
| Figure 3.2 | Searching |
| Figure 3.3 | Connected |
| Figure 3.4 | Mic activation |
| Figure 3.5 | light on command |
| Figure 3.6 | Tv On |
| Figure 3.7 | Fan On |
| Figure 3.8 | Light Off |
| Figure 3.9 | Light on |

**CHAPTER 1**

**INTRODUCTION**

**1. 1 SYNOPSIS**

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 House 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.

The advantages of using voice as an interfacing medium are multifold. Firstly we would do away with or significantly decrease the need of training for operating technology. Secondly, the simplification of services would entail a wider adoption of existing technology and would help people with varied disabilities access the same technology. We have deployed an Android Application as user front end primarily because of the ease at which the platform provides us with means to use complex technology and due to the widespread adoption in the mobile industry. Android is being used as the operating system for over 80% of the smartphones.

Voice controlled House Automation System leverages the power of Arduino to provide a holistic voice controlled automation system. Using Natural Language Processing and the available hardware in most smartphones, it translates voice to be used for controlling electrical devices.

**1.2 REVIEW OF LITERATURE**

1. Design of an Intelligent Voice Controlled Home Automation System

(International Journal of Computer Applications (0975 – 8887) Volume 121 – No.15, July 2015)

By- SonaliSen, ShamikChakrabarty, RaghavToshniwal, AnkitaBhaumik

Department of Computer Science, St. Xavier’s College, kataKol

They undertook a viable solution the need of automation at the very basic level, that is, in our homes. According to them, The project will enable us to bring every appliance at every corner of our home under our control from a single point without having to get up and manually switch on or off the appliance. The use of a Bluetooth module assists the use of this system from various locations in our house.The system is further simplified by allowing appliances to be controlled by our voice. The user need not have to have to immense knowledge over the language of English. Just by saying the appliance name and the corresponding number assigned to that particular appliance, and telling it to switch on or off will enable the user to have complete control over any appliance without any effort.

Android applications are very simple and user friendly allowing the user to understand its functionalities in very little time. Hence, the use of android application in this system allows a user to easily learn the process and get accustomed to the functions. Moreover, the entire system is very flexible and scalable. Any number of appliances can be added as and when required. Hence, the systems finds use not only in houses but also in many offices where appliances such as fans or lights on multiple floors can be controlled by a person on any of the floors, saving manual labour and human effort to switch on or off

the electronic appliances, thereby saving time.

1. Controlling Home Appliances Remotely through Voice Command

(International Journal of Computer Applications (0975 – 888) Volume 48– No.17, June 2012)

By- Faisal Baig Electrical Department Federal Urdu University of Arts, Science and Technology Islamabad, Pakistan ,Saira Beg Computer Science Department COMSATS Institute of Information Technology Islamabad, Pakistan , Muhammad Fahad Khan Computer Science Department Federal Urdu University of Arts, Science and Technology Islamabad, Pakistan

The main object of Home automation is to provide a wireless communication link of home appliances to the remote user. The main objective of this work is to make such a system which controls the home appliances remotely.

The communication link between the appliances and remote user plays an impotent roll in automation. In this study we proposed a system that control electric appliance via voice when the user is in remote area, and also it controls the appliances through home mobile.

1. Voice Controlled Smart Home

(International Journal of Emerging Technology and Advanced Engineering Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 5, Issue 1, January 2015))

By- Amrutha S1 , Aravind S2 , Ansu Mathew3 , Swathy Sugathan4 , Rajasree R5 , Priyalakshmi S6 1 Student, Sree Buddha College of Engineering for Women, Pathanamthitta, Kerala, India 2,6Asst.Prof, Sree Buddha College of Engineering for Women, Pathanamthitta, Kerala, India

The control unit for the embedded section is any microcontroller. In this we use an ARM based LPC2148 microcontroller. Applications and home appliances are the last step of the proposed system. Each device connected with the network must have a switching circuit to turn it ON/OFF. The status of each device is monitored by using toggle circuit. The control part will on/off the device according to the control characters.

The main advantage of the system is that it does require training of voices for only one time. At the same time MATLAB software has been used to support humancomputer interactions to realize multiple functions. The wireless part of the system has been implemented by using Zigbee RF modules. Hence, the system is highly efficient and it consumes low power. This system can be easily extended to remotely control the home appliances through smart devices like iPhone/ iPod and others phones so that one can remotely check the status of the home appliances and turn ON or OFF the same. The proposed system is for three home appliances. But, it can be easily expanded to include more home appliances.

1. Zigbee Based Voice Controlled Wireless Smart Home System

(International Journal of Wireless & Mobile Networks (IJWMN), Vol. 6, No. 1, February 2014)

By- ThorayaObaid

Home automation is the automatic or semi-automatic control and monitoring of household appliances and residential house features like doors, gate and even the windows. This is a demonstration of how to design and build a multi-purpose wireless system that can switch OFF and ON any electrical household appliance depending on the voice produced by the user. ThorayaObaid et al. [1] proposed a voice controlled wireless smart home system for elderly and disabled people. The proposed system has two main components namely voice recognition system, and wireless system. Lab View software has been used to implement the voice recognition system. On the other hand, ZigBee wireless modules have been used to implement the wireless system. Based on the received data at the wireless receiver associated with the appliances desired switching operations are performed. The proposed system is a low cost and low power system because ZigBee is used. Additionally the proposed system needs to be trained of voice command only once.

1. Voice Recognition Wireless Home Automation System Based On Zigbee

(IOSR Journal of Electronics and Communication Engineering (IOSR-JECE), Volume 6, Issue 1 (May. - Jun. 2013), PP 65-75.)

By- Dhawan S. Thakur and Aditi Sharma

Another home automation system was proposed by Dhawan S. Thakurand and Aditi Sharma [2]. That demonstrates a system that can be integrated as a single portable unit and allows one to wirelessly control lights, fans, air conditioners, television sets, security cameras, electronic doors, computer systems, audio/visual equipment‘s etc. and turn on or off any appliance that is plugged into a wall outlet, get the status of different sensors and take decision accordingly.The overall system is controlled from a microphone which is connected with HM 2007 speech recognition chip. This chip sends the voice commands in binary sequence to microcontroller. The base station unit takes decision and sends the commands to remote station by ZigBee transceiver.

**CHAPTER-2**

**PROPOSED SYSTEM**

Here in our project, we have used Arduino UNO R3 along with voice recognition of an android phone and the connection between them is established using Bluetooth module (HC-05).

First an android applicationis made which can accept voice as input and transfer that input to the arduino. The connection between Arduino UNO and the android application of android phone is established using Bluetooth module (HC-05). The android application uses Google’s speech recognition to accept and process user’s commands. The voice commands are provided with particular functionality and when received by Arduino UNO, it operates accordingly.

Software used for programming of Arduino UNO is Arduino 1.6.0and interfacing was done using USB connection.

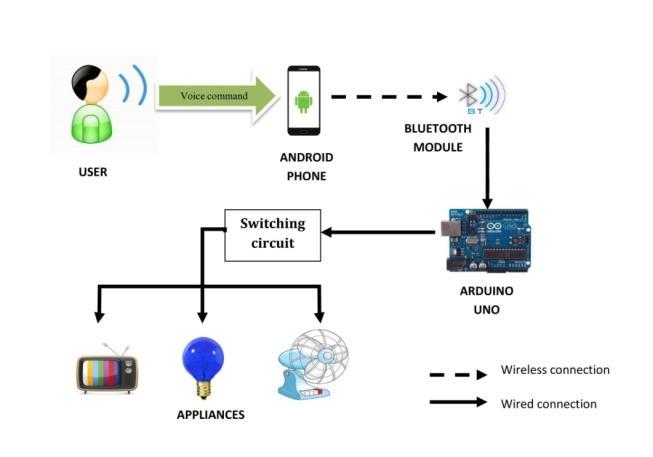
1. ARDUINO UNO It is a microcontroller board based on the ATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started.
2. BLUETOOTH MODULE (HC-05)It is an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup.The HC-05 Bluetooth Module can be used in a Master or Slave configuration, making it a great solution for wireless communication. This serial port Bluetooth module is fully qualified Bluetooth V2.0+EDR.

**LIST OF COMPONENTS:**

1. Arduino Uno
2. Bluetooth module (HC-05)
3. 4-channel relay
4. Wires
5. Breadboard
6. Light bulb
7. Sockets
8. 12V adapter
9. PCB board
10. LED’s

**SYSTEM DESIGN**

**2.1 System Components(HARDWARE)**

****

**Figure 2.1: Block Diagram of the System**

The Voice-operated Android and Arduino Home automation system uses an Android based Bluetooth enabled phone for its application and the Arduino Uno as the microcontroller. The key components of this system are:

* + Android based phone
  + Bluetooth module
  + Arduino Uno
  + Relay boards

**2.1.1 Android Based Phone**

Android is a mobile operating system (OS) based on the Linux kernel and currently developed by Google. With a user interface based on direct manipulation, the OS uses touch inputs that loosely correspond to real-world actions, like swiping, tapping, pinching, and reverse pinching to manipulate on-screen objects, and a virtual keyboard. We have used the Android platform because of its huge market globally and it’s easy to use user interface

Applications on the Android phones extend the functionality of devices and are written primarily in the Java programming language using the Android software development kit (SDK). The voice recognizer which is an in built feature of Android phones is used to build an application which the user can operate to automate the appliances in his house. The user interface of the application is shown below:

**Android**

• Android is an open-source operating system which means that any manufacturer can use it in their phones free of charge.

• It was built to be truly open.

• Android is built on the open Linux Kernel.

Furthermore, it utilizes a custom JAVA virtual machine that was designed to optimize memory and hardware resources in a mobile environment.

**Android Application on Mobile Phones**

• An android app is meant for phones with an android based operating systems. They can be downloaded from the android app Market which is pre-loaded on every android phone.

• Maps, Drive and Play Movies are some examples.

**Android Application Operated Bluetooth**

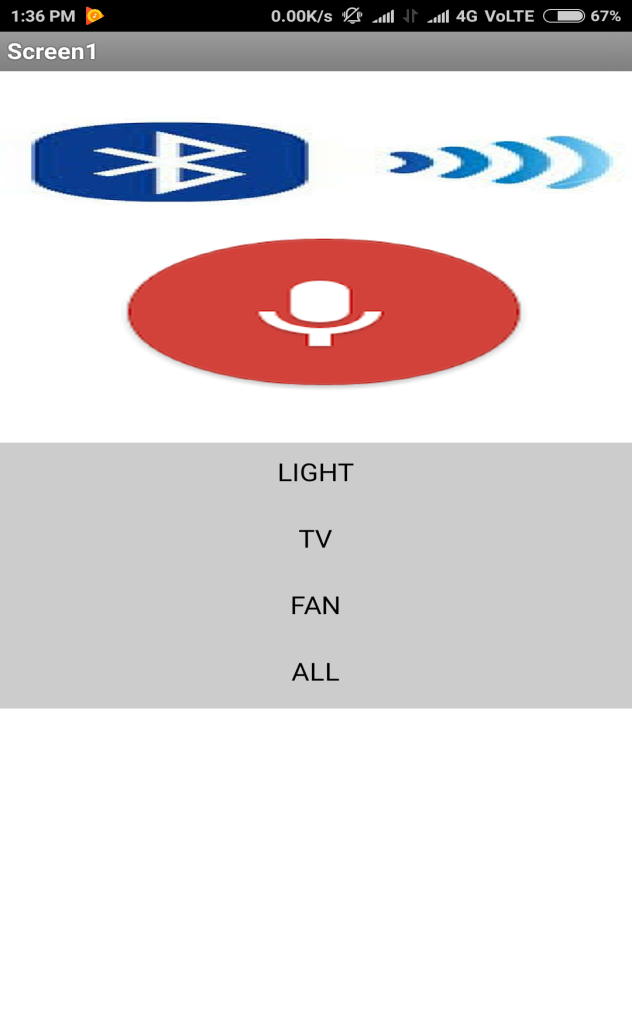
• The Android platform includes support for the Bluetooth network stack, which allows a device to wirelessly exchange data with other Bluetooth devices.

• The application framework provides access to the Bluetooth functionality through the Android Bluetooth APIs.

**Android Application for Home Automation**

• Control home electrical system using smart phone with android application.

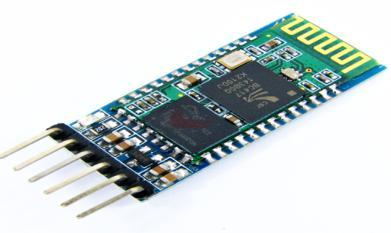
• Blue tooth Module’s outputs to directly drive loads like bulbs, Lamps, Sockets.



**Figure 2.2: Interface for the Voice Control Application**

The microphone button is tapped and the voice command is given to switch the corresponding device on/off. The voice recognizer listens and converts what is said to the nearest matching words or text. The Bluetooth adapter present in the phone is configured to send this text to the Bluetooth module on the Arduino Uno board that would in turn control the electrical appliances through the relay boards.

**2.1.2 Bluetooth Module**

**

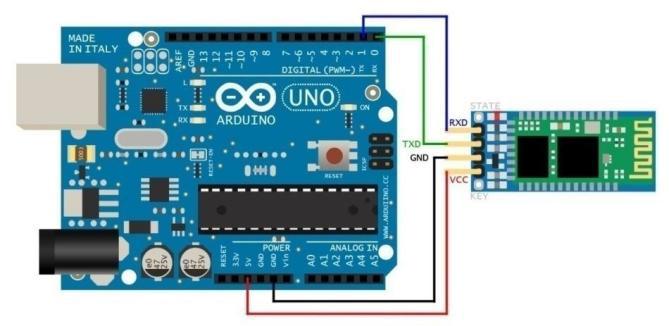
**Figure 2.3: Bluetooth module**

Bluetooth is a wireless technology standard for exchanging data over short distances (using short-wavelength UHF radio waves in the ISM band from 2.4 to 2.485 GHz) from fixed and mobile devices, and building personal area networks (PANs) . The Bluetooth module being used allows us to transmit and receive signals . It receives the text from the Android phone and transmits it to the serial port of the Arduino Uno.

Specifications:-

• For the communication between mobile phone and microcontroller Bluetooth module(HC-05) is used.

* HC-05 is low power 1.8V operation and is easy to use with Bluetooth SPP (serial port protocol).
* Serial port Bluetooth module have a Bluetooth 2.0+EDR (enhanced data rate), 3Mbps modulation with complete 2.4GHZ radio transceiver and baseband.
* Using Bluetooth profile and android platform architecture different type of Bluetooth applications can be developed.
* Typical -80dBm sensitivity
* Up to +4dBm RF transmit power
* Low Power 1.8V Operation ,1.8 to 3.6V I/O
* PIO control
* UART interface with programmable baud rate
* With integrated antenna
* With edge connector
* Default Baud rate: 3840

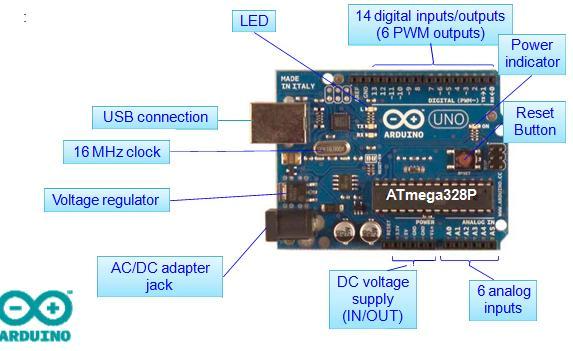


**Figure 2.4: Interfacing with arduino**

**2.1.3 Arduino Uno R3**

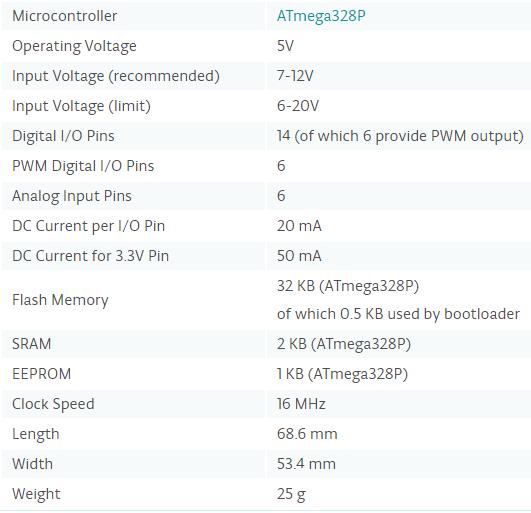
• The Arduino Uno is a microcontroller board based on the ATmega328P.

• It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button.

• Simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

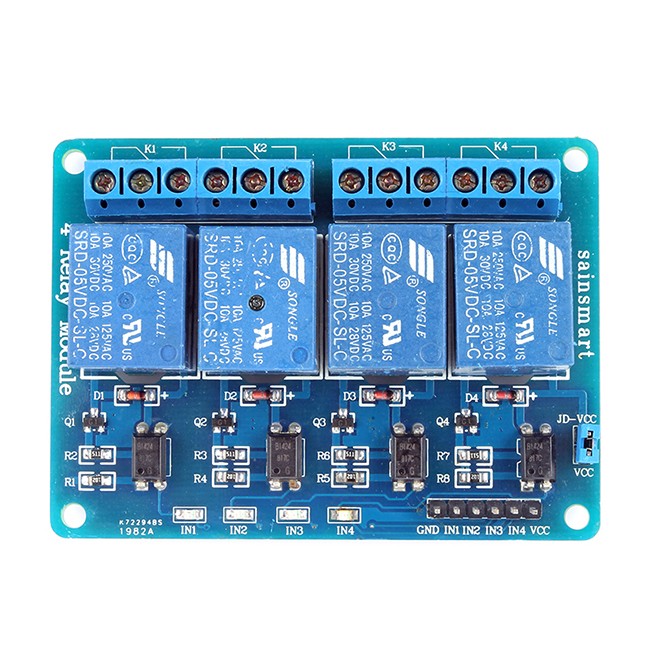
**Figure 2.5: Arduino UNO board**

**Specifications…**



**Figure 2.6: Specifications of Arduino UNO**

**2.1.4 Four channel relay:**

****

**Figure 2.7: Four channel relay**

This is a 5V 4-Channels Relay module, It can be controlled directly by a wide range of microcontrollers such as Arduino, AVR, PIC, ARM and MSP430. 4 relays are included in this module, with “NC” ports means “Normally connected to COM” and “NO” ports means “Normally open to COM”. This module also equipped with 4 LEDS to show the status of relays.

Features 4 mechanical relays with status indicator LED

• Both “NC” and “NO” ports for each relay

• Specification Module Type: Control

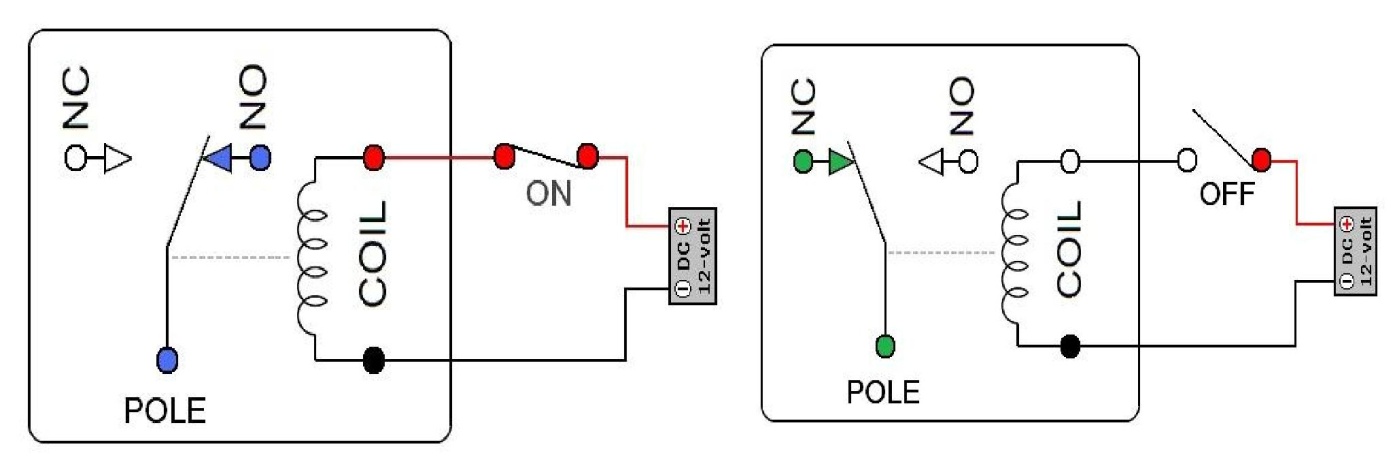
• Weight: 70.00g

• Board Size: 8 x 4.8 x 2cm

• Version: 1

• Operation Level: Digital 5V

• Power Supply: External 5V

**Working of relay**

**Figure 2.8: Relay working**

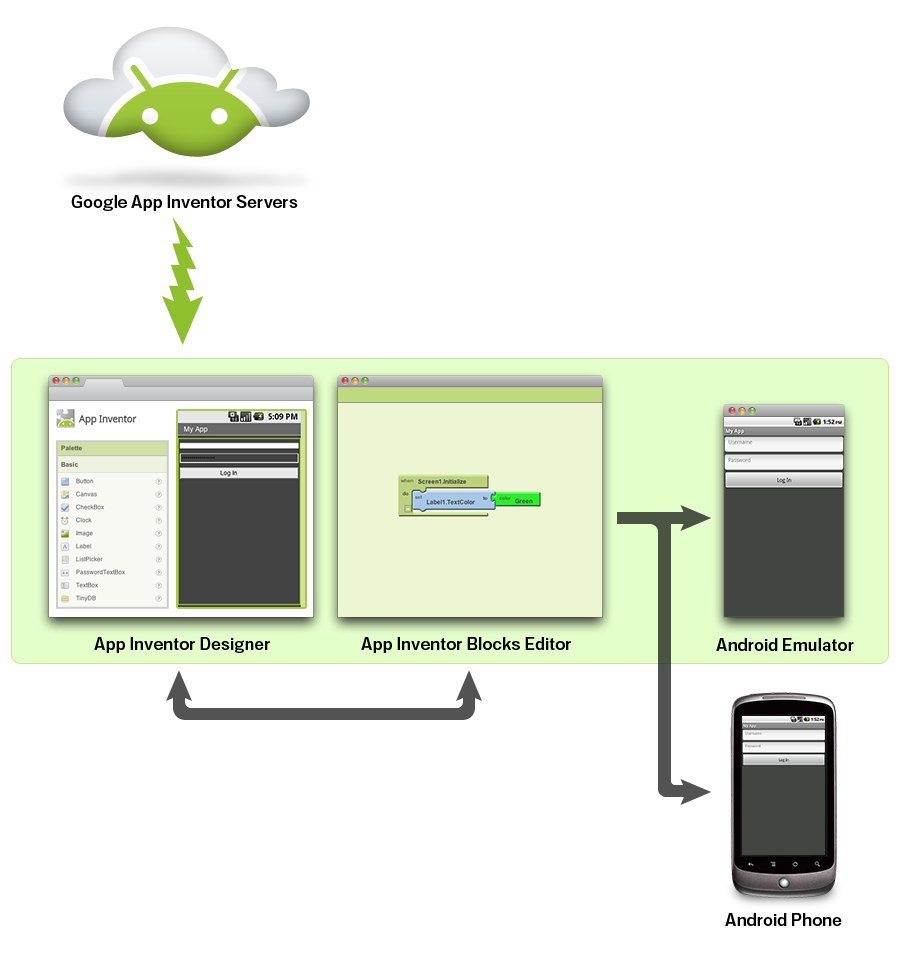
A relay is an [electromagnetic](http://www.explainthatstuff.com/magnetism.html) switch operated by a relatively small [electric](http://www.explainthatstuff.com/electricity.html) current that can turn on or off a much larger electric current. The heart of a relay is an electromagnet (a coil of wire that becomes a temporary [magnet](http://www.explainthatstuff.com/magnetism.html) when electricity flows through it). You can think of a relay as a kind of electric [lever](http://www.explainthatstuff.com/toolsmachines.html): switch it on with a tiny current and it switches on ("leverages") another appliance using a much bigger current. Why is that useful? As the name suggests, many sensors are incredibly sensitivepieces of [electronic](http://www.explainthatstuff.com/electronics.html) equipment and produce only small electric currents. But often we need them to drive bigger pieces of apparatus that use bigger currents. Relays bridge the gap, making it possible for small currents to activate larger ones. That means relays can work either as switches (turning things on and off) or as amplifiers (converting small currents into larger ones).

Relays are switches that open and close circuits electromechanically or electronically. Relays control one electrical circuit by opening and closing contacts in another circuit. As relay diagrams show, when a relay contact is normally open (NO), there is an open contact when the relay is not energized. When a relay contact is Normally Closed (NC), there is a closed contact when the relay is not energized. In either case, applying electrical current to the contacts will change their state. Relays are generally used to switch smaller currents in a control circuit and do not usually control power consuming devices except for small motors and Solenoids that draw low amps. Nonetheless, relays can "control" larger voltages and amperes by having an amplifying effect because a small voltage applied to a relays coil can result in a large voltage being switched by the contacts. Protective relays can prevent equipment damage by detecting electrical abnormalities, including overcurrent, undercurrent, overloads and reverse currents. In addition, relays are also widely used to switch starting coils, heating elements, pilot lights and audible alarms.

**2.2 System Components(SOFTWARE)**

**2.2.1 MIT App Inventor**

App Inventor lets you develop applications for Android phones using a web browser and either a connected phone or emulator. The App Inventor servers store your work and help you keep track of your projects.

  
 **Figure 2.9: App inventor architecture**

You build apps by working with:

* The App Inventor Designer*,* where you select the components for your app.
* The App Inventor Blocks Editor*,* where you assemble program blocks that specify how the components should behave. You assemble programs visually, fitting pieces together like pieces of a puzzle.

Your app appears on the phone step-by-step as you add pieces to it, so you can test your work as you build. When you're done, you can package your app and produce a stand-alone application to install.

If you don't have an Android phone, you can build your apps using the Android emulator*,* software that runs on your computer and behaves just like the phone.

The App Inventor development environment is supported for Mac OS X, GNU/Linux, and Windows operating systems, and several popular Android phone models. Applications created with App Inventor can be installed on any Android phone.

**2.2.2 ARDUINO**

The Arduino Integrated Development Environment - or Arduino Software (IDE) - contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino and Genuino hardware to upload programs and communicate with them.

**Writing Sketches**

Programs written using Arduino Software (IDE) are called sketches. These sketches are written in the text editor and are saved with the file extension .ino. The editor has features for cutting/pasting and for searching/replacing text. The message area gives feedback while saving and exporting and also displays errors. The console displays text output by the Arduino Software (IDE), including complete error messages and other information. The bottom righthand corner of the window displays the configured board and serial port. The toolbar buttons allow you to verify and upload programs, create, open, and save sketches, and open the serial monitor.

|  |  |
| --- | --- |
| https://www.arduino.cc/en/uploads/Guide/play.png | *Verify*  Checks your code for errors compiling it. |
| https://www.arduino.cc/en/uploads/Guide/export.png | *Upload*  Compiles your code and uploads it to the configured board. |
| https://www.arduino.cc/en/uploads/Guide/new.png | *New*  Creates a new sketch. |
| https://www.arduino.cc/en/uploads/Guide/open.png | *Open*  Presents a menu of all the sketches in your sketchbook. Clicking one will open it within the current window overwriting its content.  Note: due to a bug in Java, this menu doesn't scroll; if you need to open a sketch late in the list, use the File | Sketchbookmenu instead. |
| https://www.arduino.cc/en/uploads/Guide/save.png | *Save*  Saves your sketch. |
| https://www.arduino.cc/en/uploads/Guide/serial_monitor.png | *Serial Monitor*  Opens the [serial monitor](https://www.arduino.cc/en/Guide/Environment#serialmonitor). |

Additional commands are found within the five menus: File, Edit, Sketch, Tools, Help. The menus are context sensitive, which means only those items relevant to the work currently being carried out are available.

**File**

* *New*   
  Creates a new instance of the editor, with the bare minimum structure of a sketch already in place.
* *Open*   
  Allows to load a sketch file browsing through the computer drives and folders.
* *Open Recent*   
  Provides a short list of the most recent sketches, ready to be opened.
* *Sketchbook*   
  Shows the current sketches within the sketchbook folder structure; clicking on any name opens the corresponding sketch in a new editor instance.
* *Examples*   
  Any example provided by the Arduino Software (IDE) or library shows up in this menu item. All the examples are structured in a tree that allows easy access by topic or library.
* *Close*   
  Closes the instance of the Arduino Software from which it is clicked.
* *Save*   
  Saves the sketch with the current name. If the file hasn't been named before, a name will be provided in a "Save as.." window.
* *Save as...*   
  Allows to save the current sketch with a different name.
* *Page Setup*   
  It shows the Page Setup window for printing.
* *Print*   
  Sends the current sketch to the printer according to the settings defined in Page Setup.
* *Preferences*   
  Opens the Preferences window where some settings of the IDE may be customized, as the language of the IDE interface.
* *Quit*   
  Closes all IDE windows. The same sketches open when Quit was chosen will be automatically reopened the next time you start the IDE.

**Edit**

* *Undo/Redo*   
  Goes back of one or more steps you did while editing; when you go back, you may go forward with Redo.
* *Cut*   
  Removes the selected text from the editor and places it into the clipboard.
* *Copy*   
  Duplicates the selected text in the editor and places it into the clipboard.
* *Copy for Forum*   
  Copies the code of your sketch to the clipboard in a form suitable for posting to the forum, complete with syntax coloring.
* *Copy as HTML*   
  Copies the code of your sketch to the clipboard as HTML, suitable for embedding in web pages.
* *Paste*   
  Puts the contents of the clipboard at the cursor position, in the editor.
* *Select All*   
  Selects and highlights the whole content of the editor.
* *Comment/Uncomment*   
  Puts or removes the // comment marker at the beginning of each selected line.
* *Increase/Decrease Indent*   
  Adds or subtracts a space at the beginning of each selected line, moving the text one space on the right or eliminating a space at the beginning.
* *Find*   
  Opens the Find and Replace window where you can specify text to search inside the current sketch according to several options.
* *Find Next*   
  Highlights the next occurrence - if any - of the string specified as the search item in the Find window, relative to the cursor position.
* *Find Previous*   
  Highlights the previous occurrence - if any - of the string specified as the search item in the Find window relative to the cursor position.

**Sketch**

* *Verify/Compile*   
  Checks your sketch for errors compiling it; it will report memory usage for code and variables in the console area.
* *Upload*   
  Compiles and loads the binary file onto the configured board through the configured Port.
* *Upload Using Programmer*   
  This will overwrite the bootloader on the board; you will need to use Tools > Burn Bootloader to restore it and be able to Upload to USB serial port again. However, it allows you to use the full capacity of the Flash memory for your sketch. Please note that this command will NOT burn the fuses. To do so a *Tools -> Burn Bootloader* command must be executed.
* *Export Compiled Binary*   
  Saves a .hex file that may be kept as archive or sent to the board using other tools.
* *Show Sketch Folder*   
  Opens the current sketch folder.
* *Include Library*   
  Adds a library to your sketch by inserting #include statements at the start of your code. For more details, see [libraries](https://www.arduino.cc/en/Guide/Environment#libraries) below. Additionally, from this menu item you can access the Library Manager and import new libraries from .zip files.
* *Add File...*   
  Adds a source file to the sketch (it will be copied from its current location). The new file appears in a new tab in the sketch window. Files can be removed from the sketch using the tab menu accessible clicking on the small triangle icon below the serial monitor one on the right side o the toolbar.

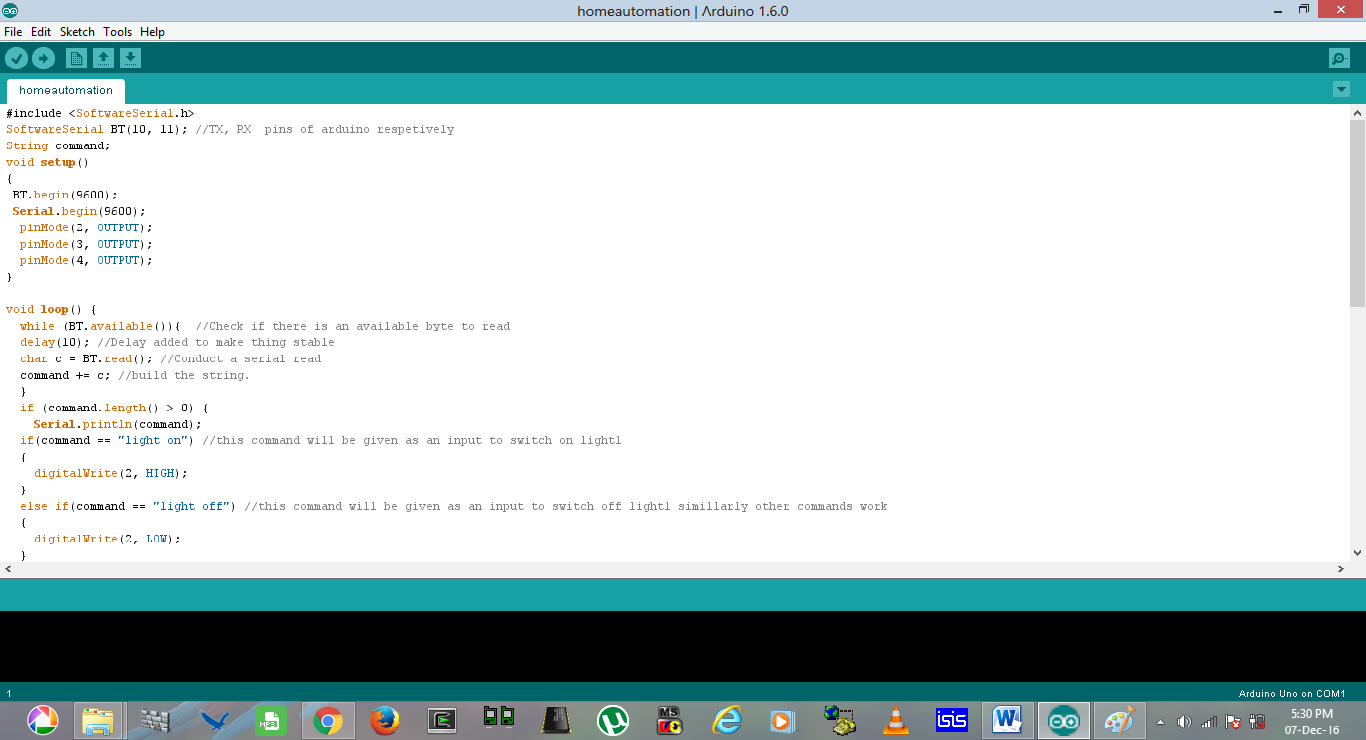
**Tools**

* *Auto Format*   
  This formats your code nicely: i.e. indents it so that opening and closing curly braces line up, and that the statements inside curly braces are indented more.
* *Archive Sketch*   
  Archives a copy of the current sketch in .zip format. The archive is placed in the same directory as the sketch.
* *Fix Encoding & Reload*   
  Fixes possible discrepancies between the editor char map encoding and other operating systems char maps.
* *Serial Monitor*   
  Opens the serial monitor window and initiates the exchange of data with any connected board on the currently selected Port. This usually resets the board, if the board supports Reset over serial port opening.
* *Board*   
  Select the board that you're
* *Port*   
  This menu contains all the serial devices (real or virtual) on your machine. It should automatically refresh every time you open the top-level tools menu.

Help

Here you find easy access to a number of documents that come with the Arduino Software (IDE). You have access to Getting Started, Reference, this guide to the IDE and other documents locally, without an internet connection. The documents are a local copy of the online ones and may link back to our online website.

* *Find in Reference*   
  This is the only interactive function of the Help menu: it directly selects the relevant page in the local copy of the Reference for the function or command under the cursor.



**Figure 2.10: Arduino interface**

**CHAPTER- 3**

**WORKING**

In this project, we learned to design a relay and HC-05 Bluetooth based Android Mobile controlled wireless Home Automation system with AVR ATmega328P microcontroller . Here, we used the Android Mobile as the input device to control the appliances .  
  
The communication between HC-05 Bluetooth Module and Android Mobile takes place through wireless Bluetooth technology. And the communication between HC-05 Bluetooth Module and ATmega328P microcontroller takes place through UART serial communication protocol. The HC-05 Bluetooth Module and Android Mobile are connected through Bluetooth. User enters the control signal from Android Mobile through Bluetooth Terminal App and the Android Mobile transmits the control signal to the HC-05 Bluetooth Module through its Bluetooth . The HC-05 Bluetooth Module receives the control signal and transmits it to the ATmega32 microcontroller through UART.  
  
The ATmega32 microcontroller receives the control signal and processes it and sends the required control signal to the Relay Driver. Relay Driver will turn On or Off the home appliances according the control signal received from the ATmega16 microcontroller. The ATmega16 microcontroller also sends the appliance status to the Android Mobile through Bluetooth .

The signal can be sent as voice in noiseless environment and control keys in noisy environment.

**Voice commands:-**

Here, we will use one voice command from Android Mobile to control one appliance. Different commands will be used to turn on and off the appliance.

|  |  |
| --- | --- |
| **Control voice input** | **Appliance Status Change** |
| Light on | Lights On |
| Light off | Lights  Off |
| Tv on | Tv  On |
| Tv off | TvOff |
| Fan on | Fan On |
| Fan of | Fan off |
| All on | All appliances On |
| All of | All aplliancesOff |

**Control keys:-**  
Here, we will use one key from Android Mobile to control one appliance. Same key will be used to turn on and off the appliance. Each time we press the same key, the appliance changes its status. So, for four appliances, we will use 3 keys from the Android Mobile. Apart from these 3 keys, we will use 1 more key to turn on and off all appliances at a time. Now, press different keys from the Android Mobile and control the appliances as you want.

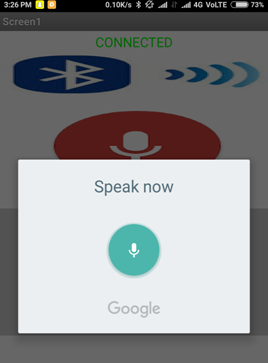
|  |  |
| --- | --- |
| **Control keys** | **Appliance Status Change** |
| Light | Light On/Off |
| Tv | Tv On/Off |
| Fan | Fan  On/Off |
| All | All Appliances  On/Off |
| **3.1 Flowchart:** |  |

**Figure 3.1: Flowchart**

**3.2 Different stages of working:-**

**3.2.1 Bluetooth connection**

The bluetooth module is pwered on so that the android app may detect the signal provided by the hc 05.The user switches on the mobile and connects with hc 05 with bluetooth.the app shows if the module is connected or not.

**Figure 3.2: Searching** **Figure 3.3: Connected**

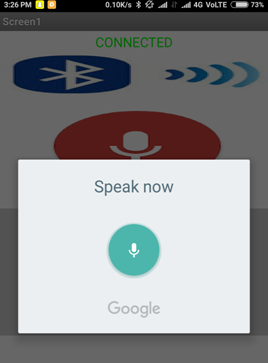
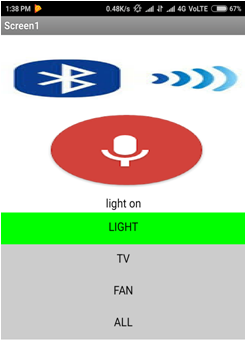
The hc05 may ask for a password on first time connection ,the password is generally

**0000 or 1234.**For more protection the passwoed can be changed by oprating the hc05 in the administerative mode.

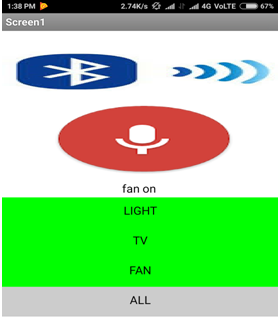
**3.2.2 Giving voice commands**

The user gives the voice command as per the requirements.the user can tap buttons or make voice commands by taping on the mic.The commands are sent to the hc05 using serial communication and the information is decoded to give desired output.

Upon decoding the microcontroller sends the command to the allocated relay swith to turn on or turn off the appliances.

**Figure 3.4: Mic activation Figure 3.5: light on command**

**Figure 3.6: Tv on Figure 3.7: fan on**

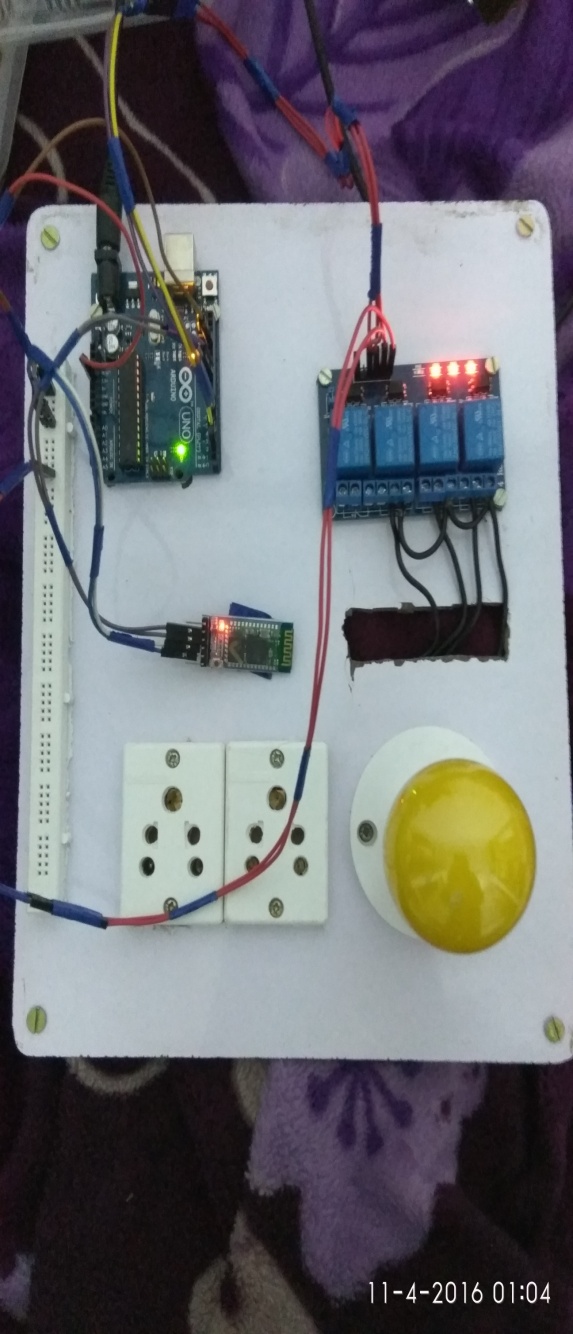
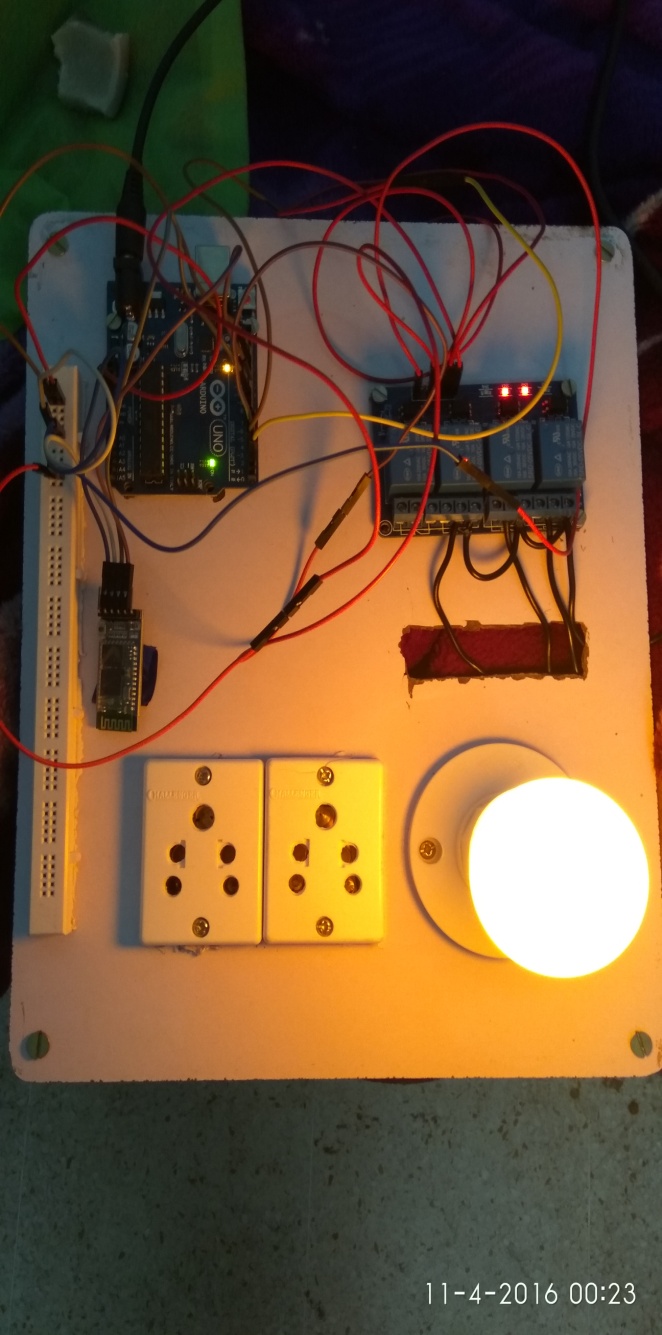
**3.2.2 OUTPUT ON THE APPLIANCES**

On the basis of the various inputs to the microcontroller the various aplliances are operated.

The output of the relay is applied to the relay which in turn drives the current to the appliances.

The relay can operate in NO or NC modes .NO stands for narmally open and NC cmeans normally closed.We can use any of the two modes as per our requirement.

.

** **

**Figure 3.8:Light off Figure 3.9: light on**

**CHAPTER-4**

**RESULT AND DISCUSSIONS**

When finally the project was completed following observations were made:-

1.The project works normally for voice input under noiseless environment.various commands used were recognised by the microcontroller and the related appliance was turned on or off.

|  |  |
| --- | --- |
| **Control voice input** | **Appliance Status Change** |
| Light on | Lights On |
| Light off | Lights  Off |
| Tv on | Tv  On |
| Tv off | TvOff |
| Fan on | Fan On |
| Fan of | Fan off |
| All on | All appliances On |
| All of | All aplliancesOff |

2.In noisy environemt the voice was not interpreted properly and that’s why we propose the use of keys in the app for such situations.

|  |  |
| --- | --- |
| **Control keys** | **Appliance Status Change** |
| Light | Light On/Off |
| Tv | Tv On/Off |
| Fan | Fan  On/Off |
| All | All Appliances  On/Off |

**Applications and Future Scope :**

1. **USE IN HOSPITALS:** The system can be used in hospitals for patients in non-ambulatory conditions to control appliances such as TV, CEILING FAN, FAN HEATER, LIGHT etc. This can be done by the patient itself by using an android phone without the help of any attendant. This automation process will reduce the human work and will benefit the hospitals.

The system can also be modified to make systems that can make the patient’s bed movements automatic. Sometimes it is preferred by the patient to lay fully and sometimes partially. Thus the bed motors can be controlled using the automation system.

The whole system can also be used to automate other instruments, doors, lights and other appliances. Some of the advantages of the automation systems in hospitals are :

* **LabourSavings**  
  Using automation to replace manually intensive tasks that are better done by machine can be a big time saver. It doesn’t have to eliminate employees, but rather elevate them into higher-functioning roles that make use of the clinical expertise they have been trained for.
* **Improved Quality and Consistency**  
  Automation tools are not subject to human error or fatigue, so they can help provide a consistent basis of care activities. A Texas hospital study found that greater automation in the areas of medical records, order entry, and decision support appeared to result in a reduction in deaths, complications and cost.
* **Reduced Waste**  
  Use of paper and spreadsheets and other workarounds needed for an overfull workload can lead to a lot of waste. For example, rather than playing phone tag with a discharged patient in the free minutes between hospital nursing duties, automation can help get nurses and patients connected more efficiently.
* **Higher Throughput**  
  A nurse supported by automation tools can handle a larger population of patients at one time. Instead of scaling up and down your headcount as patient volumes grow and shrink, an automated platform can scale flexibly to address groups of all sizes.

1. **MOBILE BOARD:** The system can be used as a mobile appliance control board. This board will have three to four socket ports that can be operated by using android application. We can take this mobile board anywhere, switch in MOBILE CHARGERS, LAPTOP CHARGERS, TV, etc. into the socket ports of this mobile board by just providing it power supply. The same system can be made by the project components we have used. The mobile board rely on the same working as our project. This a very practical application of the project as it is easy to make and is cheap.
2. **HOME AUTOMATION:**Home automation is the residential extension of building automation and involves the control and automation of lighting, heating (such as smart thermostats), ventilation, air conditioning (HVAC), and security, as well as home appliances such as washer/dryers, ovens or refrigerators/freezers that use WiFi for remote monitoring. Modern systems generally consist of switches and sensors connected to a central hub sometimes called a "gateway" from which the system is controlled with a user interface that is interacted either with a wall-mounted terminal, mobile phone software, tablet computer or a web interface, often but not always via internet cloud services.The system can be used to provide home automation solutions for normal, handicapped and old age people. They can simply give commands by voice to control home appliances.

**CHAPTER -5**

**CONCLUSION**

The “Voice controlled home appliances” project which is used for controlling devices through human voice is proposed and implemented and several changes can be done in this to suite different applications and scenarios. The system is made up of the components proposed earlier and are made working to provide the ease of controlling appliances through voice. For relevance touch switches in the application are also provided in case of noisy environment. The key objective of our system design was to provide easy means for normal, handicapped and old age persons to control and operate home appliances and we succeeded in that. This system is designed using Android which sends the captured voice input to Arduino UNO via Bluetooth module. An android application is made which uses Google’s voice recognition andArduino Uno processes the voice input to control the home appliances.The connection between Arduino UNO and the android application of android phone is established using Bluetooth module (HC-05). The android application uses Google’s speech recognition to accept and process user’s commands. The voice commands are provided with particular functionality and when received by Arduino UNO, it operates accordingly.

The whole system worked properly and is made possible to meet today’s automation demands. In future, the system have many applications as such in home automation, use in hospitals, as a mobile board and many more. The whole aim of providing the ease of controlling the appliances with voice to normal and special people is achieved.

**REFERENCES**

1. International Journal of Computer Applications (0975 – 8887) Volume 121 – No.15, July 2015
2. International Journal of Computer Applications (0975 – 888) Volume 48– No.17, June 2012
3. IOSR Journal of Electronics and Communication Engineering (IOSR-JECE), Volume 6, Issue 1 (May. - Jun. 2013), PP 65-75
4. International Journal of Wireless & Mobile Networks (IJWMN), Vol. 6, No. 1, February 2014
5. International Journal of Emerging Technology and Advanced Engineering Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 5, Issue 1, January 2015)
6. http://appinventor.mit.edu/
7. http://www.instructables.com/
8. https://www.arduino.cc/